

UNIVERSITÉ DE MONTRÉAL

**CONTRIBUTION À L'OPTIMISATION NON-DIFFÉRENTIABLE
ET À LA DÉCOMPOSITION EN PROGRAMMATION
MATHÉMATIQUE**

CLAUDE BERGER

**DÉPARTEMENT DE MATHÉMATIQUES
ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL**

**THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION
DU DIPLÔME DE PHILOSOPHIAE DOCTOR (Ph.D.)
(MATHÉMATIQUES DE L'INGÉNIEUR)**

AOÛT 1996



National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-26412-2

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée:

**CONTRIBUTION À L'OPTIMISATION NON-DIFFÉRENTIABLE
ET À LA DÉCOMPOSITION EN PROGRAMMATION
MATHÉMATIQUE**

présentée par: BERGER Claude

en vue de l'obtention du diplôme de: Philosophiae Doctor

a été dûment acceptée par le jury d'examen constitué de:

M. SAVARD Gilles, Ph.D., président

M. SOUMIS François, Ph.D., membre et directeur de recherche

M. LOULOU Richard, Ph.D., codirecteur de recherche

M. DESROSIERS Jacques, Ph.D., membre

M. VIAL Jean-Philippe, Docteur, membre externe

REMERCIEMENTS

Je désire remercier mes directeurs de recherche Richard Loulou et François Soumis pour l'encadrement, le support financier et l'environnement de travail stimulant qu'ils m'ont fournis tout au long de mon doctorat.

Le Conseil de Recherches en Sciences Naturelles et en Génie (CRSNG) a aussi contribué, par la bourse qu'il m'a octroyée, au financement de ces travaux et je l'en remercie.

Je remercie également Saeb El Hachem pour les intéressantes discussions que nous avons eues et pour m'avoir fait découvrir les méthodes de faisceaux. Je me suis toujours rappelé, particulièrement dans les moments difficiles, de certains de ses conseils sur la façon de mener un projet de thèse. Ceci m'a permis de trouver la volonté de m'accrocher, lorsque tout tremblait, et de mener à terme ce projet de doctorat.

Je remercie aussi le Professeur Krzysztof C. Kiwiel qui a lu une version préliminaire de ce document et qui a porté à mon attention une erreur importante. Il m'a aussi apporté plusieurs suggestions.

Gilles Savard m'a aussi fait des suggestions intéressantes qui ont contribué à l'amélioration de ma thèse. Je tiens à l'en remercier.

La production de ce document a nécessité un important travail de traitement

de texte. Je désire remercier Francine Benoit, qui en a produit la plus grande partie, pour son beau travail. C'est ce qui m'a permis de faire mes premiers pas avec $\text{\LaTeX}$. Je remercie aussi Nicole Paradis qui, par ses conseils, a souvent su me dépanner avec $\text{\LaTeX}$. Cela m'a permis de faire la mise en page finale en un temps raisonnable.

Tous mes remerciements aux membres du jury qui ont accepté de lire et de critiquer cette thèse.

Enfin, un gros merci à tous ceux et celles qui ont contribué d'une manière ou d'une autre à la réalisation de mon doctorat.

RÉSUMÉ

Cette thèse reconsidère les problèmes de programmation linéaire de grande taille munis de la structure duale angulaire par blocs. Ceux-ci peuvent être transformés, par projection sur les variables liantes, en un problème équivalent d'optimisation convexe non-différentiable consistant en la minimisation d'une somme de fonctions polyédriques convexes sur son domaine. Malheureusement, les méthodes classiques de décomposition ne traitent que minimalement la non-différentiabilité de ce problème équivalent, cherchant davantage à l'éviter qu'à l'attaquer de front.

Le point de vue adopté dans cette thèse est qu'en reconnaissant pleinement que l'on cherche à résoudre un problème équivalent d'optimisation non-différentiable, des améliorations peuvent être obtenues en exploitant les idées et algorithmes de ce domaine relativement nouveau, notamment les méthodes de faisceaux. Celles-ci possèdent la plupart des ingrédients des méthodes qu'elles remplacent. Elles comportent en plus des mécanismes particuliers destinés à déjouer les pièges de la non-différentiabilité.

Etonnamment, bien que la décomposition de problèmes de programmation mathématique soit probablement la principale source de problèmes d'optimisation non-différentiable, il n'existe que très peu de publications traitant de l'application des méthodes de faisceaux à ces problèmes. Nous avons cherché, dans cette thèse, à suppléer à ce manque de connaissances en utilisant une méthode de faisceaux dans le cadre d'une approche primale de décomposition appliquée à nos problèmes. Ceci nous a amené à concevoir une nouvelle variante des méthodes de faisceaux. Bien que la majorité des résultats de convergence soient valides pour le problème

plus général de la minimisation de la somme d'un nombre fini de fonctions convexes fermées sur un domaine convexe fermé quelconque (non défini explicitement par des fonctions), il faudrait modifier l'algorithme pour le rendre capable de résoudre tous les problèmes de cette classe. Ce domaine peut être complexe et n'est pas traité implicitement par l'algorithme comme dans les cas habituels où il s'agit d'ensembles simples tels ceux découlant de la non-négativité des variables. Nous avons supposé l'existence d'oracles qui engendrent des hyperplans de séparation lorsque la solution du problème de recherche d'une direction d'amélioration est à l'extérieur du domaine du problème. Contrairement à l'hypothèse habituelle des méthodes de faisceaux, nous n'avons pas supposé que les fonctions considérées sont à valeurs finies dans R^n , ce qui restreint davantage le domaine de l'optimisation. Nous avons traité ce problème en considérant explicitement une formulation équivalente de ce dernier dans laquelle les contraintes ensemblistes sont remplacées par un terme de pénalisation qui n'est rien d'autre que la fonction indicatrice du domaine. Malgré que cette équivalence soit bien connue, aucune méthode de faisceaux ne l'exploite. Ceci peut être expliqué par le fait que les fonctions convexes n'ont pas de sous-différentiel à l'extérieur de leur domaine. Or, les méthodes de faisceaux supposent l'existence d'un oracle produisant un sous-gradient à chaque point déterminé par la solution du problème de recherche d'une direction d'amélioration. Ces sous-gradients servent à affiner la représentation approchée des fonctions. Ce sont aussi des ε -sous-gradients de la fonction économique au meilleur point trouvé et c'est cette observation qui est utilisée par les méthodes de faisceaux. En constatant que les hyperplans de séparation produits par les oracles de notre méthode permettent de définir des ε -sous-gradients de la fonction économique du problème équivalent, nous pouvons adapter les méthodes existantes de manière à traiter ce cas. Enfin, le terme de pénalisation du problème de recherche d'une direction n'est plus nécessairement quadratique et la norme utilisée n'est pas forcément euclidienne (ni quadratique).

Ceci nous a conduit à déterminer la formulation duale, plus pratique, du problème de recherche d'une direction. Ce problème, faisant intervenir une norme dite duale, est une généralisation du problème obtenu avec les méthodes classiques.

Préalablement à l'obtention de l'algorithme, nous avons considéré deux approches différentes, mais non indépendantes, de détermination de directions d'amélioration. Ces deux approches, soit celles des régions de confiance et des pénalisations, utilisent la fonction norme afin de restreindre l'optimisation à une région de l'espace où nous espérons que les modèles des fonctions ont de meilleures chances d'être de bonne qualité. Nous avons considéré des normes quelconques et constaté que les normes qui semblent les plus avantageuses pour une approche ne le sont pas pour l'autre.

L'algorithme que nous avons conçu est basé sur l'approche des pénalisations. Toutefois, les tests numériques ont été faits à l'aide de l'approche des régions de confiance obtenue en adaptant l'algorithme. Des tests préliminaires effectués sur des problèmes de programmation linéaire réels issus de la modélisation énergétique ont permis une première calibration de l'algorithme et de comparer diverses stratégies et options disponibles. Les normes ℓ_1 et ℓ_∞ ont été testées, et les résultats obtenus sont encourageants. Ils démontrent une très nette supériorité de la méthode sur celle de Benders. Ils montrent aussi que la méthode fonctionne mieux avec la norme ℓ_1 . Bien que la méthode soit beaucoup plus stable et rapide que celle de Benders, elle présente tout de même le comportement asymptotique des méthodes de plans coupants.

ABSTRACT

In this thesis, we reconsider the class of linear programming problems exhibiting the dual block angular structure. Using a projection over the coupling variables, we can transform any problem of this class into an equivalent convex, but nondifferentiable, optimization problem where the sum of a finite number of polyhedral convex functions is minimized over its domain. Unfortunately, the classical decomposition methods of mathematical programming do not fully exploit the nondifferentiability of this equivalent problem.

We firmly believe that the classical approaches can be improved by using the ideas and methods from the field of nondifferentiable optimization, and from the philosophy of the bundle methods in particular. These methods have the basic ingredients of the decomposition algorithms but they are also characterized by special mechanisms to counteract the nondifferentiability's pitfalls.

It is surprising that despite being the main source of nondifferentiable optimization problems, there are only a few reported applications of bundle methods to any kind of decomposition. A goal of this thesis is to fill in the gaps in this area for the case of a primal decomposition approach. In order to do this, it was necessary to develop a new bundle method. Although most of the convergence results remain valid for the more general problem of the minimization of the sum of a finite number of closed convex extended functions over an arbitrary convex set (i.e. not explicitly defined by functions), the method cannot handle this class of problems unless we do some modification to the algorithm. Unlike the usual assumptions in mathematical programming, this set may be complex and cannot be handled implicitly,

and the functions defining the problem were not assumed to take only finite values. We assumed the existence of oracles that generate separating hyperplanes when the solution of the direction finding problem is not feasible for the original problem.

Instead of minimizing the equivalent nondifferentiable optimization problem, the algorithm minimizes another equivalent problem obtained by removing the set constraint and replacing it by a penalty term in its objective function. The penalty term is the indicator function of the set. No known bundle method is based on this well known equivalence. This can probably be explained by the fact that bundle methods assume the existence of an oracle generating a subgradient at each point found by the direction finding problem. Indicator functions have empty subdifferentials outside the feasible set. However, each subgradient is also an ε -subgradient to the best feasible solution found so far and this is what is really needed by a bundle method. Since the separating hyperplanes generated by our oracles might be used to define ε -subgradients of the objective function, it becomes possible to adapt many existing bundle methods to this case.

We also consider more general stabilizing terms in the direction finding problem of our method. It is not necessarily a quadratic expression of the euclidean norm. This lead us to determine the more practical dual form of the direction finding problem, expressed in terms of the so called dual norm.

Prior to the definition of the new algorithm, we have considered two different but related approaches to the problem of determining an improvement direction. Both the approaches, trust region and penalization, are based on a norm function whose aim is to restrict the optimization to a region of the space where, hopefully, the approximate descriptions of the functions are reasonably accurate. Arbitrary

norms were considered. However, we found that norms which are attractive for one approach are usually not for the other.

Our algorithm is based on the penalization viewpoint, but our numerical tests were done using a trust region adaptation of the method. They allowed us to calibrate the algorithm's parameters and to compare various strategies and options. In particular, the ℓ_1 and ℓ_∞ norms were compared and the preliminary results are very promising. It was found that our implementation usually performs better with the ℓ_1 norm. The results also show a very clear superiority of the method over the Benders' one. Finally, inspite of an improved stability behaviour and better empirical convergence, the method retained the asymptotical convergence properties of the cutting plane methods.

TABLE DES MATIÈRES

REMERCIEMENTS	iv
RÉSUMÉ	vi
ABSTRACT	ix
TABLE DES MATIÈRES	xii
LISTE DES TABLEAUX	xvi
LISTE DES FIGURES	xvii
LISTE DES ANNEXES	xviii
 CHAPITRE 1 Introduction	 1
 CHAPITRE 2 Le problème: définition et quelques cas particuliers ..	 8
2.1 Le problème	8
2.2 Cas particuliers	12
2.3 Cas plus général	18
 CHAPITRE 3 Travaux apparentés	 21
 CHAPITRE 4 Accélération et stabilisation des algorithmes de plans sécants en programmation mathématique convexe	 35

4.1 Méthodes de plans sécants: description et lacunes	35
4.2 Stabilisation des méthodes de plans sécants: cas non contraint ..	41
4.3 Stabilisation des méthodes de plans sécants: cas des contraintes explicites	45
4.4 Stabilisation des méthodes de plans sécants: cas des contraintes ensemblistes.....	48
4.5 La méthode BOXSTEP	52
4.6 Propriétés des fonctions $v_i(\cdot)$ du chapitre 2	55
 CHAPITRE 5 Construction et mise-à-jour des problèmes de recherche d'une direction d'amélioration pour le problème de la minimisation d'une somme de fonctions convexes.....	 63
5.1 Introduction	63
5.2 Linéarisation des ensembles et fonctions convexes définissant le problème	65
5.3 Détermination d'une direction d'amélioration: approche des régions de confiance.....	69
5.4 Détermination du problème dual: approche des régions de confiance.....	74
5.5 Détermination d'une direction d'amélioration: approche des pénalisations.....	76

5.6 Détermination du problème dual: approche des pénalisations	79
5.7 La stratégie d'accumulation	88
5.8 La stratégie d'accumulation/épuration	92
5.9 La stratégie d'agrégation	95
 CHAPITRE 6 Un algorithme de descente pour la minimisation d'une somme de fonctions convexes	 108
6.1 Introduction	108
6.2 L'algorithme avec la stratégie d'agrégation	112
6.3 Remarques sur l'algorithme	116
6.4 Résultats préliminaires	121
6.5 Convergence de l'algorithme utilisant la stratégie d'agrégation ..	138
6.6 L'algorithme pour les stratégies d'accumulation et d'accumulation/épuration	162
 CHAPITRE 7 Expérimentation numérique	 167
7.1 Introduction	167
7.2 Le programme CMinSfpc	168
7.3 Les problèmes tests	170

7.4 L'approche des fonctions à valeurs finies	173
7.4.1 Rayons invariants	173
7.4.2 Ajustement dynamique du paramètre γ	176
7.4.3 La stratégie d'accumulation/épuration	181
7.4.4 La convergence asymptotique	182
7.4.5 Deux procédures d'enrichissement	183
7.5 L'approche des plans coupants	185
7.5.1 Rayons invariants et méthode de Benders	186
7.5.2 Ajustement dynamique du paramètre γ	189
7.5.3 La convergence asymptotique	192
7.6 Les derniers problèmes	193
7.7 Conclusion	195
 CHAPITRE 8 Conclusions et travaux futurs	 198
RÉFÉRENCES	202
ANNEXES	221

LISTE DES TABLEAUX

7.1	Quelques caractéristiques des problèmes tests.....	171
7.2	Nombre de variables des problèmes équivalents	172
D.1	Comportement de l'algorithme appliqué au problème DQCNY3 pour des rayons constants sous la stratégie d'accumulation	245
D.2	Performances de l'algorithme avec ajustement dynamique de γ sous la stratégie d'accumulation.....	246
D.3	Performances de l'algorithme avec ajustement dynamique de γ sous la stratégie d'accumulation/épuration	247
D.4	Comportement asymptotique de l'algorithme appliqué au problème DQCNY5_QC	248
D.5	Performances de CMinSfpc en présence d'enrichissement	249
D.6	Comparaison de la méthode des faisceaux avec γ constant à la méthode de Benders	250
D.7	Performances de l'algorithme avec plans coupants et ajustement dy- namique de γ sous la stratégie d'accumulation	251
D.8	Performances de l'algorithme avec plans coupants et ajustement dy- namique de γ sous la stratégie d'accumulation/épuration	252
D.9	Comportement asymptotique de l'algorithme appliqué au problème DQCNY5_QC en utilisant la norme ℓ_1	253
D.10	Performances de l'algorithme avec ajustement dynamique de γ pour les plus gros problèmes.....	254
D.11	Comportement asymptotique de l'algorithme appliqué au problème DQCNY_9QC' en utilisant la norme ℓ_1	255

LISTE DES FIGURES

6.1	Représentation géométrique du procédé d'agrégation.....	136
B.1	Interprétation géométrique de la fonction d'appui.....	229

LISTE DES ANNEXES

ANNEXE A Rappels d'algèbre linéaire et de programmation linéaire	221
ANNEXE B Rappels d'analyse convexe	223
ANNEXE C Démonstration du lemme 5.9.8	240
ANNEXE D Résultats numériques	245

CHAPITRE 1

Introduction

Le besoin d'optimiser le fonctionnement de systèmes complexes a conduit à la naissance de la recherche opérationnelle et, en particulier, de la programmation mathématique. Lorsque l'on utilise ces outils, on est amené à construire un ou plusieurs modèles mathématiques dans le but de décrire le système étudié ainsi que les actions que l'on peut poser pour le modifier. Ces modèles sont habituellement constitués de plusieurs contraintes reflétant des considérations technologiques, financières, organisationnelles, environnementales, de mise en marché, et autres, du système en question. Celles-ci relient les variables sur lesquelles on peut agir. Une ou plusieurs fonctions économiques que l'on cherche à optimiser permettent alors de déterminer une ou plusieurs solutions, ou manières d'affecter les ressources limitées du système, aux activités qui compétitionnent entre elles.

L'évolution de la technologie informatique et des méthodes de la recherche opérationnelle a permis, au fil des années, d'élargir le domaine d'application de la programmation mathématique en permettant, entre autres, de traiter des problèmes de plus en plus gros et complexes. Cependant, malgré ces progrès constants, le chercheur opérationnel s'est toujours buté à l'obstacle que constitue la grande taille de plusieurs problèmes. Celle-ci constitue souvent un obstacle à leur résolution par les méthodes générales, particulièrement s'il y a des non-linéarités. Heureusement, la plupart des problèmes de grande taille possèdent des structures particulières dont on peut tirer profit. Celles-ci permettent la décomposition des problèmes en un ensemble de plus petits problèmes pouvant être résolus isolément, mais de manière coordonnée. C'est le domaine de la décomposition en programmation mathématique.

Cette branche de la programmation mathématique est née au début des années 60 avec la publication des méthodes, toujours populaires aujourd'hui, de décomposition de Dantzig et Wolfe (1960, 1961), puis de Benders (1962). Ces méthodes et leurs variantes doivent une bonne partie de leur popularité au fait qu'elles permettent de traiter des problèmes d'une très grande importance pratique et économique tels ceux qui se posent à une entreprise possédant plusieurs divisions autonomes soumises à des contraintes budgétaires globales (c'est-à-dire devant partager certaines ressources communes) ou encore lorsque ces divisions doivent utiliser une partie de leurs ressources propres pour la réalisation d'activités communes. Les mêmes problèmes se posent à un pays divisé en régions possédant une certaine autonomie.

La décennie des années 60 est sans doute la période la plus fertile de la jeune histoire des méthodes de décomposition. Les méthodes conçues à cette époque, ainsi que la plupart de celles qui ont suivies, ne traitent que des problèmes définis à l'aide de fonctions différentiables. Bien que l'on n'en soit pas toujours conscient, ces méthodes introduisent des fonctions possédant des dérivées discontinues. Elles nous amènent à traiter des problèmes équivalents d'optimisation non-différentiable. Ceux-ci sont plus complexes et possèdent de moins bonnes propriétés analytiques que les problèmes de programmation mathématique différentiable.

Les méthodes classiques de décomposition, même lorsqu'elles ont été obtenues en transformant le problème original en un problème équivalent d'optimisation non-différentiable, n'ont pas été conçues comme des méthodes d'optimisation non-différentiable. Elles ne traitent que minimalement la non-différentiabilité, cherchant davantage à l'éviter qu'à l'attaquer de front.

Le point de vue adopté dans cette thèse est qu'en reconnaissant pleinement que l'on cherche à résoudre un problème équivalent d'optimisation non-différentiable, des améliorations peuvent être obtenues en exploitant les idées de ce domaine relativement nouveau, apparu durant les années 60 mais ayant vraiment pris son essor durant les années 70 avec la publication des premières méthodes de faisceaux. Celles-ci, lorsqu'appliquées aux problèmes non-différentiables équivalents traités par certaines approches de décomposition résultent en des algorithmes plus généraux possédant la plupart des ingrédients des méthodes qu'elles remplacent. Les différences proviennent du fait que les méthodes de faisceaux comportent des mécanismes particuliers destinés à déjouer les pièges de la non-différentiabilité. La fonction économique de leurs problèmes de recherche d'une direction d'amélioration, le problème maître des méthodes de décomposition, possède un terme stabilisateur, habituellement le carré de la norme euclidienne du déplacement.

Etonnamment, bien que la décomposition de problèmes de programmation mathématique soit probablement la principale source de problèmes d'optimisation non-différentiable, il n'existe que très peu de publications traitant de l'application des méthodes de faisceaux à ces problèmes. La seule étude venue à notre connaissance est celle de Medhi (1987) qui a utilisé une méthode de faisceaux connue dans le cadre d'une approche duale de décomposition appliquée à des problèmes de programmation linéaire exhibant la structure angulaire par blocs. Ses tests ont montré une nette amélioration par rapport aux performances de l'algorithme de Dantzig et Wolfe.

Nous avons cherché, dans cette thèse, à suppléer à ce manque de connaissances en utilisant une méthode de faisceaux dans le cadre d'une approche primale de décomposition appliquée à des problèmes de programmation linéaire munis de la structure duale angulaire par blocs typique des problèmes traités par la méthode de Benders. Contrairement à Medhi, nous avons dû concevoir une nouvelle méthode de faisceaux pour parvenir à nos fins. Celle-ci est une généralisation de la méthode proximale de faisceaux de Kiwiel (1990). Elle minimise la somme d'un nombre fini de fonctions polyédriques convexes sur un domaine polyédrique convexe. Etant donné que la majorité des résultats nécessaires à l'établissement de la convergence de la méthode sont valides pour le cas plus général de la minimisation de la somme d'un nombre fini de fonctions convexes fermées sur un domaine convexe fermé quelconque (non défini explicitement par des fonctions), nous avons écrit la thèse comme si c'était ce problème que nous voulions résoudre. Le domaine réalisable peut être complexe et n'est pas traité implicitement par l'algorithme comme dans les cas habituels où il s'agit d'ensembles simples tels ceux découlant de la non-négativité des variables. Nous avons supposé l'existence d'oracles qui engendrent des hyperplans de séparation lorsque la solution du problème de recherche d'une direction d'amélioration est à l'extérieur du domaine du problème. Contrairement à l'hypothèse habituelle des méthodes de faisceaux, nous n'avons pas supposé que les fonctions considérées sont à valeurs finies dans $\mathbb{R}^n$. Ceci restreint davantage le domaine de l'optimisation car elle ne peut être faite que là où les fonctions sont définies. Nous avons traité ce problème en considérant explicitement une formulation équivalente de ce dernier dans laquelle les contraintes ensemblistes sont remplacées par un terme de pénalisation qui n'est rien d'autre que la fonction indicatrice du domaine. Malgré que cette équivalence soit bien connue, aucune méthode de faisceaux ne l'exploite. Ceci peut être expliqué par le fait que les fonctions convexes

n'ont pas de sous-différentiel à l'extérieur de leur domaine. Or, les méthodes de faisceaux supposent l'existence d'un oracle produisant un sous-gradient à chaque point déterminé par la solution du problème de recherche d'une direction d'amélioration. Ces sous-gradients servent à affiner la représentation approchée des fonctions. Ce sont aussi des ε -sous-gradients de la fonction économique au meilleur point trouvé et c'est cette observation qui est utilisée par les méthodes de faisceaux. En constatant que les hyperplans de séparation produits par les oracles de notre méthode permettent de définir des ε -sous-gradients de la fonction économique du problème équivalent nous pouvons adapter les méthodes existantes de manière à traiter ce cas. Enfin, le terme de pénalisation du problème de recherche d'une direction n'est plus nécessairement quadratique et la norme utilisée n'est pas forcément euclidienne (ni quadratique). Ceci nous a conduit à déterminer la formulation duale, plus pratique, du problème de recherche d'une direction. Ce problème, faisant intervenir une norme dite duale, est une généralisation du problème obtenu avec les méthodes classiques.

La mise en oeuvre de la méthode a nécessité l'écriture d'un programme C utilisant le code commercial de programmation linéaire Cplex (1992). Des tests préliminaires effectués sur des problèmes de programmation linéaire réels issus de la modélisation énergétique ont permis une première calibration de l'algorithme et de comparer diverses stratégies et options disponibles. Ils ont aussi montré que la méthode est efficace et beaucoup plus performante que la méthode de Benders pour ces problèmes.

L'organisation de la thèse est la suivante. Le prochain chapitre présente les problèmes de programmation linéaire desquels on tire, par projection, un problème équivalent d'optimisation convexe non-différentiable qui consiste à minimiser une

fonction polyédrique convexe sur son domaine. On constate que plusieurs problèmes pratiques se posant en modélisation énergétique et environnementale sont des cas particuliers de ce problème. Un problème d'optimisation convexe non-différentiable semblable est aussi obtenu lorsque l'on applique une approche de décomposition par les ressources à un problème de programmation convexe séparable avec contraintes couplantes.

Le chapitre 3 est un survol de la littérature des principales méthodes de décomposition en programmation mathématique et des méthodes d'optimisation non-différentiable pouvant éventuellement être utilisées pour résoudre les problèmes d'optimisation convexe non-différentiable équivalents. Pour ces dernières, l'accent a été mis sur les méthodes de faisceaux.

Les formulations non-différentiable équivalentes se prêtent assez naturellement à une résolution à l'aide de méthodes de plans coupants. Le chapitre 4 présente les faiblesses de ces méthodes ainsi que des façons de les stabiliser. Ceci nous permet d'introduire certains concepts importants des méthodes de faisceaux (pas de descente, pas nul, centre de stabilité). On y discute aussi de certaines manières de traiter les contraintes. Celles-ci étant souvent implicites (i.e. non définies par des fonctions connues explicitement), nous proposons de les éliminer en incluant la fonction indicatrice de l'ensemble réalisable dans la fonction économique du problème équivalent. Cette approche est exploitée dans les deux chapitres suivants qui constituent la partie la plus importante de la thèse. Ceux-ci traitent le problème de la minimisation d'une somme de fonctions convexes, pouvant prendre la valeur $+\infty$ à certains points, dans un ensemble convexe fermé.

Le chapitre 5 est consacré à l'étude du problème de la recherche de directions d'amélioration. Nous obtenons les formulations primale et duale de ce problème sous l'approche des régions de confiance puis sous celle des pénalisations. Ces formulations sont plus générales que celles utilisées dans les méthodes connues de faisceaux. Le reste du chapitre montre comment on peut utiliser ces formulations sous les stratégies d'accumulation, d'accumulation/épuration, puis d'agrégation de l'information.

Le chapitre 6 présente et démontre la convergence d'un algorithme de faisceaux basé sur l'approche des pénalisations sous l'hypothèse de polyédralité des fonctions et ensembles considérés. Une variante de celui-ci a été testée sur des problèmes réels. Le chapitre 7 décrit cette variante et les problèmes testés, puis fait l'analyse des résultats numériques.

La thèse se termine au chapitre 8 avec la conclusion et des suggestions pour des améliorations possibles et des travaux ultérieurs.

Le document comporte aussi 4 annexes. Les deux premières résument des résultats et définitions connus utilisés surtout dans les chapitres 5 et 6. L'annexe C présente la preuve d'un résultat du chapitre 5 tandis que l'annexe D contient les tableaux des résultats analysés au chapitre 7.

La plupart des notations utilisées dans cette thèse sont ou bien d'usage courant ou bien définies à l'annexe B. Remarquons cependant que la notation $\| \cdot \|$ est utilisée pour désigner une norme quelconque et que la fonction $\langle \cdot, \cdot \rangle : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ sera utilisée pour désigner le produit scalaire euclidien, i.e. $\langle a, b \rangle = a^T b$, $\forall a, b \in \mathbb{R}^n$.

CHAPITRE 2

Le problème: définition et quelques cas particuliers

Nous présentons dans ce chapitre le problème de programmation linéaire qui est à l'origine des travaux faisant l'objet de cette thèse. Nous montrons comment on peut le reformuler en un problème équivalent d'optimisation convexe non-différentiable consistant à minimiser une somme de fonctions non-différentiables sur un ensemble convexe. Nous discutons ensuite de son application à des problèmes pratiques à caractère économique issus, entre autres, de la modélisation énergétique et environnementale. Nous terminons en montrant que des problèmes plus généraux de programmation non-linéaire convexe peuvent aussi être reformulés comme un problème de minimisation d'une somme de fonctions convexes non-différentiables sur un ensemble convexe. La méthode proposée dans cette thèse pourra être appliquée à tous les problèmes de programmation linéaire considérés dans ce chapitre.

2.1 Le problème

Le problème global se formule comme suit:

$$\begin{aligned}
 & \underset{x,y}{\text{minimiser}} \sum_{i=1}^r \langle c_i, x_i \rangle + \langle c, y \rangle \\
 & \text{sous les contraintes} \\
 & \quad C_0 y \geq b_0 \\
 & \quad A_i x_i + C_i y \geq b_i, \quad \forall i \in \{1, \dots, r\} \\
 & \quad B_i x_i \geq d_i, \quad \forall i \in \{1, \dots, r\} \\
 & \quad x_i \geq 0, \quad \forall i \in \{1, \dots, r\} \\
 & \quad y \geq 0,
 \end{aligned} \tag{2.1}$$

où

$$\begin{aligned}
 c_i &\text{ est de dimension } n_i \times 1, & \forall i \in \{1, \dots, r\} \\
 c &\text{ est de dimension } n_0 \times 1, \\
 A_i &\text{ est de dimension } m_i \times n_i, & \forall i \in \{1, \dots, r\} \\
 B_i &\text{ est de dimension } p_i \times n_i, & \forall i \in \{1, \dots, r\} \\
 C_i &\text{ est de dimension } m_i \times n_0, & \forall i \in \{0, 1, \dots, r\} \\
 b_i &\text{ est de dimension } m_i \times 1, & \forall i \in \{0, 1, \dots, r\}
 \end{aligned}$$

et

$$d_i \text{ est de dimension } p_i \times 1.$$

Les vecteurs x_i sont de dimension $n_i \times 1, i \in \{1, \dots, r\}$, tandis que le vecteur y est de dimension $n_0 \times 1$.

Ce problème exhibe la structure angulaire par blocs étudiée par Dantzig et Wolfe (1960, 1961) et on peut le résoudre en appliquant leur algorithme de décomposition par les prix, les contraintes $A_i x_i + C_i y \geq b_i, \forall i \in 1, \dots, r$, constituant les contraintes liantes ou couplantes. L'approche est intéressante, lorsque ces contraintes sont relativement peu nombreuses, mais souffre de deux inconvénients majeurs:

- Il n'est pas possible de tirer profit de la structure particulière de la sous-matrice constituée des matrices $C_i, i \in \{1, \dots, r\}$ (qui apparaît dans le problème maître); et
- Chaque sous-problème ne sert qu'à engendrer des propositions (colonnes) transmises au problème maître. C'est ce dernier qui détermine la pondération des propositions produisant une solution optimale, les solutions optimales des sous-problèmes ne constituant généralement pas une solution réalisable pour

le problème original, même lorsqu'elles sont obtenues avec le vecteur des prix optimaux.

Le problème (2.1) possède également la structure du problème étudié par Benders (1962) dans le cas particulier d'un problème de programmation linéaire à variables continues où la matrice associée aux variables $x = (x_1, \dots, x_r)$ possède une structure angulaire par blocs.

On reformule légèrement le problème en créant de nouvelles variables s_i et en posant

$$s_i + C_i y = 0, \quad \forall i \in \{1, \dots, r\}, \quad (2.2)$$

ce qui nous permet d'obtenir la formulation équivalente ci-dessous:

$$\begin{aligned} & \underset{x, y, s}{\text{minimiser}} \quad \sum_{i=1}^r \langle c_i, x_i \rangle + \langle c, y \rangle \\ & \text{sous les contraintes} \\ & \quad C_0 y \geq b_0 \\ & \quad A_i x_i - s_i \geq b_i, \quad \forall i \in \{1, \dots, r\} \\ & \quad B_i x_i \geq d_i, \quad \forall i \in \{1, \dots, r\} \\ & \quad s_i + C_i y = 0, \quad \forall i \in \{1, \dots, r\} \\ & \quad x_i \geq 0, \quad \forall i \in \{1, \dots, r\} \\ & \quad y \geq 0. \end{aligned} \quad (2.3)$$

Le problème devenant décomposable lorsque la valeur des variables $s = (s_1, \dots, s_r)$ est fixée, cette dernière formulation est équivalente à la suivante:

$$\underset{s=(s_1, \dots, s_r)}{\text{minimiser}} \quad v(s) \triangleq v_0(s) + \sum_{i=1}^r v_i(s_i) \quad (2.4)$$

où

$$\begin{aligned}
 v_0(s) &\triangleq \underset{y}{\text{minimiser}} \langle c, y \rangle \\
 &\text{sous les contraintes} \\
 &C_0 y \geq b_0 \\
 &C_i y = -s_i, \quad \forall i \in \{1, \dots, r\} \\
 &y \geq 0,
 \end{aligned} \tag{2.5}$$

et

$$\begin{aligned}
 v_i(s_i) &\triangleq \underset{x_i}{\text{minimiser}} \langle c_i, x_i \rangle \\
 &\text{sous les contraintes} \\
 &A_i x_i \geq b_i + s_i, \\
 &B_i x_i \geq d_i, \\
 &x_i \geq 0,
 \end{aligned} \tag{2.6}$$

Pour que le problème (2.4) soit bien défini, il faut travailler avec des fonctions convexes “étendues”, c’est-à-dire prenant leurs valeurs dans $[-\infty, +\infty]$ (bornes comprises) puisque l’une des fonctions valeurs $v_i(\cdot)$ peut prendre la valeur $+\infty$ lorsque s n’est pas réalisable pour l’un des problèmes (2.5) ou (2.6) la définissant. Les valeurs des fonctions $v_0(\cdot)$ et $v_i(\cdot)$ seront prises égales à $-\infty$ lorsque les problèmes (2.5) ou (2.6) les définissant ne sont pas bornés inférieurement.

Le formulation (2.4) correspond à une projection (Geoffrion, 1970b) sur les variables liantes. Bien qu’ayant procédé un peu différemment, c’est l’approche qu’ont trouvée Ho et Loute (1978). Remarquons qu’elle n’est pas affublée des deux tares mentionnées précédemment à propos de la décomposition par les prix de Dantzig et Wolfe. En effet, le sous-problème (2.5) permet d’exploiter la structure de la sous-matrice constituée des matrices $C_i, i \in \{1, \dots, r\}$, contrairement à l’approche de Dantzig et Wolfe où elle se trouve dans le problème maître en compagnie d’autres

colonnes (nous verrons à la prochaine section que le problème (2.5) est souvent un problème de flot à coût minimum). De plus, il n'est plus nécessaire de faire des pondérations de propositions afin de récupérer une solution réalisable (et optimale) lorsque l'algorithme s'arrête: l'union des solutions des sous-problèmes de la dernière itération constitue une solution pour le problème complet. Enfin, la décomposition par les quantités est une approche primale, ce qui veut dire que, contrairement à une approche duale comme la décomposition par les prix, on peut interrompre les calculs en tout temps et se satisfaire de la dernière solution obtenue puisqu'elle est réalisable.

2.2 Cas particuliers

Notre intérêt pour le problème (2.1) découle de la généralité des matrices C_i . En considérant des matrices C_i particulières, il est possible d'engendrer une grande classe de problèmes, dont le suivant qui est à l'origine de notre étude.

$$\begin{aligned}
 & \underset{x_i, s_{ij}, \forall i, j}{\text{minimiser}} \sum_{i=1}^r \langle c_i, x_i \rangle + \sum_{i=1}^r \sum_{\substack{j=1 \\ j \neq i}}^r \langle p_{ij}, s_{ij} \rangle \\
 & \text{sous les contraintes} \\
 & \quad A_i x_i - \sum_{\substack{j=1 \\ j \neq i}}^r E_{ij}^{-1} s_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^r s_{ji} \geq b_i, \quad \forall i \in \{1, \dots, r\} \\
 & \quad B_i x_i \geq d_i, \quad \forall i \in \{1, \dots, r\} \\
 & \quad x_i \geq 0, \quad \forall i \in \{1, \dots, r\} \\
 & \quad s_{ij} \geq 0, \quad \forall i, j \in \{1, \dots, r\}, i \neq j.
 \end{aligned} \tag{2.7}$$

Remarque: Les variables x_i et s_{ij} sont vectorielles.

Le problème (2.7) est celui qui se pose lorsque r régions ou agents économiques peuvent s'échanger une partie de leur production afin de satisfaire leurs demandes respectives à moindre coût. Les contraintes techniques de chacun des agents sont représentées par les polytopes $L_i \triangleq \{x_i : B_i x_i \geq d_i, x_i \geq 0\}, i \in \{1, \dots, r\}$, et ils doivent chacun satisfaire à une demande $b_i, i \in \{1, \dots, r\}$. De plus, chaque agent peut expédier une partie de sa production à un (ou plusieurs autres) agent(s). Le vecteur d'échange s_{ij} représente alors les quantités reçues de chaque bien par l'agent j en provenance de l'agent i . Cependant, des pertes peuvent survenir durant le transport, de sorte que l'on associera une matrice diagonale E_{ij} , représentant les efficacités de transport, au vecteur s_{ij} . S'ils choisissent de coopérer, ces r agents doivent alors résoudre le problème de programmation linéaire (2.7).

Les variables d'échange s_{ij} remplacent la variable y du problème (2.1) tandis que les matrices C_i sont des assemblages de sous-matrices E_{ij}^{-1} et de matrices identité I . Remarquons que le problème (2.7) ne possède pas l'équivalent des contraintes $C_0 y \geq b_0$ du problème (2.1).

Dans ce cas particulier, le sous-problème (2.5) a la structure d'un problème de flot (avec gain ou, plutôt, perte) à coût minimum:

$$\begin{aligned}
 v_0(s) &\triangleq \underset{s_{ij}, \forall i,j}{\text{minimiser}} \sum_{i=1}^r \sum_{\substack{j=1 \\ j \neq i}}^r \langle p_{ij}, s_{ij} \rangle \\
 &\text{sous les contraintes} \\
 &\quad - \sum_{\substack{j=1 \\ j \neq i}}^r E_{ij}^{-1} s_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^r s_{ji} = -s_i, \quad \forall i \in \{1, \dots, r\} \\
 &\quad s_{ij} \geq 0, \quad \forall i, j \in \{1, \dots, r\}, i \neq j.
 \end{aligned} \tag{2.8}$$

Le problème (2.7) s'applique à un grand nombre de problèmes réels dont plusieurs se présentent en modélisation énergétique et/ou environnementale. Toutefois, il peut arriver que cette structure particulière ne soit pas suffisamment générale. C'est le cas notamment lorsque le modèle comporte une description du secteur de production d'électricité. En effet, les quantités d'électricité échangées n'interviennent pas seulement dans les contraintes de bilan, ou contraintes de conservation de flot, selon la terminologie des réseaux, mais aussi dans des contraintes de pointe (ou de réserve) assurant une capacité excédentaire de production afin de faire face aux pannes aléatoires des équipements de production ainsi qu'aux aléas de la demande. C'est le cas du modèle MARKAL (Fishbone et Abilock, 1981; Fishbone et al., 1983). De plus, même lorsqu'il n'y a pas de contrainte de pointe, certaines variables d'échange peuvent intervenir dans plus d'une contrainte. Ainsi, lorsque deux producteurs d'électricité s'échangent de l'énergie selon un contrat de puissance, la variable d'échange correspondante intervient dans les contraintes de bilan de chacune des tranches horaires.

Le problème global a alors la structure ci-dessous:

$$\begin{aligned}
 & \underset{x_i, s_{ij}, \forall i, j}{\text{minimiser}} \sum_{i=1}^r \langle c_i, x_i \rangle + \sum_{i=1}^r \sum_{\substack{j=1 \\ j \neq i}}^r \langle p_{ij}, s_{ij} \rangle \\
 & \text{sous les contraintes} \\
 & \quad A_i^1 x_i - \sum_{\substack{j=1 \\ j \neq i}}^r E_{ij}^{-1} s_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^r s_{ji} \geq b_i^1, \quad \forall i \in \{1, \dots, r\} \\
 & \quad A_i^2 x_i - \sum_{\substack{j=1 \\ j \neq i}}^r F_{ij} s_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^r G_{ji} s_{ji} \geq b_i^2, \quad \forall i \in \{1, \dots, r\} \\
 & \quad B_i x_i \geq d_i, \quad \forall i \in \{1, \dots, r\} \\
 & \quad x_i \geq 0, \quad \forall i \in \{1, \dots, r\} \\
 & \quad s_{ij} \geq 0, \quad \forall i, j \in \{1, \dots, r\}, i \neq j,
 \end{aligned} \tag{2.9}$$

et l'on définit les variables multi-dimensionnelles s_i par

$$s_i = \begin{pmatrix} s_i^1 \\ s_i^2 \end{pmatrix} \quad \forall i \in \{1, \dots, r\}$$

où

$$s_i^1 = \sum_{\substack{j=1 \\ j \neq i}}^r E_{ij}^{-1} s_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^r s_{ji}, \quad \forall i \in \{1, \dots, r\},$$

et

$$s_i^2 = \sum_{\substack{j=1 \\ j \neq i}}^r F_{ij} s_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^r G_{ji} s_{ji}, \quad \forall i \in \{1, \dots, r\},$$

En introduisant ces vecteurs s_i et leurs définitions dans le problème (2.9), on obtient un problème décomposable lorsque $s = (s_1, \dots, s_r)$ est fixé. Le problème équivaut à résoudre

$$\underset{s=(s_1, \dots, s_r)}{\text{minimiser}} v(s) \triangleq v_0(s) + \sum_{i=1}^r v_i(s_i) \quad (2.10)$$

où

$$v_0(s) \triangleq \underset{s, \forall i, j}{\text{minimiser}} \sum_{i=1}^r \sum_{\substack{j=1 \\ j \neq i}}^r \langle p_{ij}, s_{ij} \rangle$$

sous les contraintes

$$\begin{aligned} - \sum_{\substack{j=1 \\ j \neq i}}^r E_{ij}^{-1} s_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^r s_{ji} &= -s_i^1, & \forall i \in \{1, \dots, r\} \\ - \sum_{\substack{j=1 \\ j \neq i}}^r F_{ij} s_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^r G_{ji} s_{ji} &= -s_i^2, & \forall i \in \{1, \dots, r\} \\ s_{ij} &\geq 0, & \forall i, j \in \{1, \dots, r\}, i \neq j \end{aligned} \quad (2.11)$$

et

$$\begin{aligned}
 v_i(s_i) &\triangleq \underset{x_i}{\text{minimiser}} \langle c_i, x_i \rangle \\
 &\text{sous les contraintes} \\
 &A_i^1 x_i \geq b_i^1 + s_i^1, \\
 &A_i^2 x_i \geq b_i^2 + s_i^2, \\
 &B_i x_i \geq d_i, \\
 &x_i \geq 0.
 \end{aligned} \tag{2.12}$$

Remarques: Les contraintes $A_i^2 x_i \geq b_i^2 + s_i^2$ des problèmes (2.12), $i \in \{1, \dots, r\}$, sont habituellement peu nombreuses. De même, les contraintes $-\sum_{\substack{j=1 \\ j \neq i}}^r F_{ij} s_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^r G_{ji} s_{ji} = -s_i^2$, $i \in \{1, \dots, r\}$ du problème (2.11) sont elles aussi peu nombreuses.

L'algorithme que nous allons présenter pourra résoudre le problème (2.1) ainsi que les variantes (2.7) et (2.9) de celui-ci.

Dans un modèle technico-économique global non régionalisé, on peut habituellement distinguer plusieurs secteurs. Ainsi, les indices $i \in \{1, \dots, r\}$, peuvent, dans un modèle énergétique comme MARKAL, être associés aux secteurs représentés tels la production d'électricité, le raffinage du pétrole, la production des autres produits énergétiques, et divers segments des secteurs industriel, résidentiel, commercial et institutionnel, et des transports. Le modèle n'étant pas régionalisé, il n'y a pas de structure de réseau semblable à celle des problèmes (2.7) et (2.9). Le modèle a été conçu autour du concept de "pot commun": tout se passe comme si les quantités produites de chaque bien énergétique, par les diverses technologies incluses dans le modèle, sont acheminées vers un point central (le pot commun) d'où elles sont

ensuite allouées à chacun des usages. Dans ce cas, et en tenant compte des remarques précédentes sur la présence de contraintes spéciales (comme les contraintes de pointe), le problème est le suivant:

$$\begin{aligned}
 & \underset{x_i, exp_i, rec_i, \forall i}{\text{minimiser}} \sum_{i=1}^r \langle c_i, x_i \rangle + \sum_{i=1}^r \langle p_i, exp_i \rangle + \sum_{i=1}^r \langle q_i, rec_i \rangle \\
 & \text{sous les contraintes} \\
 & \quad A_i^1 x_i - E_i^{-1} exp_i + rec_i \geq b_i^1, \quad \forall i \in \{1, \dots, r\} \\
 & \quad A_i^2 x_i - F_i exp_i + G_i rec_i \geq b_i^2, \quad \forall i \in \{1, \dots, r\} \quad (2.13) \\
 & \quad B_i x_i \geq d_i, \quad \forall i \in \{1, \dots, r\} \\
 & \quad \sum_{i=1}^r (exp_i - rec_i) \geq 0, \\
 & \quad x_i \geq 0, exp_i \geq 0, rec_i \geq 0, \forall i \in \{1, \dots, r\}
 \end{aligned}$$

Remarque: Les variables vectorielles exp_i et rec_i sont associées aux expéditions et aux réceptions par l'agent i de chaque forme énergétique. Plus précisément, exp_i est la quantité reçue, par le pot commun, des diverses quantités d'énergie expédiées par l'agent i . Puisqu'il peut y avoir pertes durant le transport, alors $E_i^{-1} exp_i$ représente les expéditions de l'agent i vers le pot commun.

Tout comme on l'a fait pour le problème (2.9), en définissant les vecteurs s_i par

$$s_i = \begin{pmatrix} s_i^1 \\ s_i^2 \end{pmatrix}$$

où

$$s_i^1 = E_i^{-1} exp_i - rec_i, \quad \forall i \in \{1, \dots, r\},$$

et

$$s_i^2 = F_i exp_i - G_i rec_i, \quad \forall i \in \{1, \dots, r\},$$

on constate que le problème devient décomposable lorsque $s = (s_1, \dots, s_r)$ est fixé:

$$\underset{s=(s_1, \dots, s_r)}{\text{minimiser}} v(s) \triangleq v_0(s) + \sum_{i=1}^r v_i(s_i) \quad (2.14)$$

où

$$v_0(s) \triangleq \underset{\exp_i, \text{rec}_i, \forall i}{\text{minimiser}} \sum_{i=1}^r \langle p_i, \exp_i \rangle + \sum_{i=1}^r \langle q_i, \text{rec}_i \rangle$$

sous les contraintes

$$\begin{aligned} \sum_{i=1}^r (\exp_i - \text{rec}_i) &\geq 0, \\ -E_i^{-1} \exp_i + \text{rec}_i &= -s_i^1, & \forall i \in \{1, \dots, r\} \\ -F_i \exp_i + G_i \text{rec}_i &= -s_i^2, & \forall i \in \{1, \dots, r\} \\ \exp_i &\geq 0, \text{rec}_i \geq 0, \forall i \in \{1, \dots, r\}, \end{aligned} \quad (2.15)$$

et

$$v_i(s_i) \triangleq \underset{x_i}{\text{minimiser}} \langle c_i, x_i \rangle$$

sous les contraintes

$$\begin{aligned} A_i^1 x_i &\geq b_i^1 + s_i^1, \\ A_i^2 x_i &\geq b_i^2 + s_i^2, \\ B_i x_i &\geq d_i, \\ x_i &\geq 0. \end{aligned} \quad (2.16)$$

Les problèmes (2.13) et (2.15) possèdent ici le bloc de contraintes $C_0 y \geq b_0$ du problème (2.1). L'algorithme que nous présenterons au chapitre 5 pourra aussi résoudre cette variante du problème (2.1).

2.3 Cas plus général

Bien que les travaux effectués dans le cadre de cette recherche aient été motivés par le problème (2.1), il est important de préciser que les diverses variantes de

l'algorithme général proposé au chapitre 6 permettent de résoudre des problèmes plus généraux.

Le problème équivalent (2.4) a été obtenu en appliquant la stratégie de projection à une reformulation du problème (2.1). Un problème équivalent un peu plus général peut être obtenu en appliquant une approche de décomposition par les ressources à un problème de programmation convexe séparable avec contraintes couplantes.

Considérons le problème de programmation convexe

$$\begin{aligned}
 &\underset{x=(x_1, \dots, x_r)}{\text{minimiser}} \quad f(x) \triangleq \sum_{i=1}^r f_i(x_i) \\
 &\text{sous les contraintes} \\
 &\quad \sum_{i=1}^r g_i(x_i) \leq b \\
 &\quad x_i \in X_i, \quad \forall i \in \{1, \dots, r\}
 \end{aligned} \tag{2.17}$$

où les fonctions $f_i : \mathbb{R}^{n_i} \rightarrow \mathbb{R}$ et $g_i : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^m, i \in \{1, \dots, r\}$, ainsi que les sous-ensembles $X_i \subseteq \mathbb{R}^{n_i}, i \in \{1, \dots, r\}$, sont convexes.

C'est le problème de la gestion d'un système formé de r sous-systèmes interconnectés dans lequel le vecteur $b \in \mathbb{R}^m$ est la quantité de ressources disponibles. Pour chaque sous-système i , la fonction $g_i(x_i)$ mesure la quantité de ressources consommées par celui-ci lorsque le vecteur de ses niveaux d'activités est x_i , et $f_i(x_i)$ en est le coût de fonctionnement.

Sous l'approche de la décomposition par les ressources, on cherche à décomposer le vecteur b en vecteurs $b_i, i \in \{1, \dots, r\}$ affectés à chacun des r sous-systèmes. Plus

précisément, si

$$S = \{(b_1, \dots, b_r) \in \mathbb{R}^m \times \dots \times \mathbb{R}^m : \sum_{i=1}^r b_i = b\}$$

et si

$$v_i(b_i) \triangleq \min\{f_i(x_i) : g_i(x_i) \leq b_i, x_i \in X_i\}, \quad \forall i \in \{1, \dots, r\}$$

alors résoudre le problème (2.17) équivaut à résoudre le problème de programmation convexe suivant:

$$\underset{(b_1, \dots, b_r) \in S}{\text{minimiser}} \ v(b_1, \dots, b_r) \triangleq \sum_{i=1}^r v_i(b_i) \quad (2.18)$$

Dans sa forme actuelle, l'algorithme général proposé au chapitre 6 ne peut être utilisé pour le résoudre.

CHAPITRE 3

Travaux apparentés

Le besoin de résoudre des problèmes réels importants amène très souvent le chercheur opérationnel à construire des problèmes de programmation mathématique de très grande taille. La grande taille d'un problème constitue souvent un obstacle à sa résolution par les méthodes générales de la programmation mathématique, particulièrement s'il comporte des non-linéarités. Heureusement, la plupart de ces problèmes possèdent des structures particulières dont on peut tirer profit. C'est ce qui a motivé les chercheurs à imaginer des méthodes permettant de remplacer la résolution de gros problèmes par la résolution coordonnée de plus petits problèmes, obtenus en les brisant. C'est le domaine de la décomposition en programmation mathématique. On peut aussi vouloir faire appel à des méthodes de décomposition dans le but d'améliorer les temps de résolution de certains problèmes. Ceci découle de l'observation que les temps de calcul des problèmes croissent plus vite que leurs tailles. En les brisant en plus petits problèmes, on peut alors espérer freiner cette croissance du temps de calcul à condition que le mécanisme de coordination ne soit pas trop coûteux. Malheureusement, ce deuxième objectif n'est pas toujours atteint.

Les méthodes de décomposition de Benders (1962) et de Dantzig et Wolfe (1960 et 1961) sont probablement les deux premières à avoir été conçues. Bien que plusieurs autres algorithmes aient été proposés depuis, elles sont encore très populaires aujourd'hui. La décennie 60 a été très fertile dans ce domaine et on trouve une bonne synthèse de la production de cette période dans l'excellent ouvrage de Lasdon

(1970) qui est encore pertinent. Minoux (1983) présente un traitement plus moderne d'un nombre plus limité de méthodes. Geoffrion (1970b et 1970c) a effectué une classification de ces techniques qui permet de mieux en saisir les différences fondamentales.

Bien que l'on n'en soit pas toujours conscient, les méthodes de décomposition introduisent des fonctions possédant des dérivées discontinues (Lemaréchal, 1989). Elles nous amènent à traiter des problèmes d'optimisation non-différentiable. Ceux-ci sont plus complexes et possèdent de moins bonnes propriétés analytiques que les problèmes de programmation mathématique différentiable.

Nous nous limiterons dans cette discussion aux méthodes de décomposition adaptées aux problèmes de programmation mathématique convexe. Plusieurs d'entre elles, dont les méthodes de projection et de décomposition par les prix ou par les ressources, nous amènent à résoudre un problème semblable au suivant:

$$\underset{x \in V}{\text{minimiser}} \ f(x) \triangleq \sum_{i=1}^r f_i(x) \quad (3.1)$$

où les fonctions f_i sont convexes et non-différentiables, et V est un ensemble convexe (pouvant être $\mathbb{R}^n$). On remarque que les problèmes (2.4) et (2.18) sont de ce genre.

Les approches duales, ou de décomposition par les prix, maximisent la fonction duale qui est concave. C'est le problème (3.1) lorsque le problème est complètement décomposable (ou, en anglais, "separable"). La méthode de Dantzig et Wolfe (1960 et 1961), pour le problème de programmation linéaire avec structure angulaire par blocs, est de ce genre. Bien qu'ayant été justifiée différemment par ses auteurs, Minoux (1983) montre qu'on la retrouve si l'on tente de maximiser la fonction duale à l'aide d'une méthode de plans coupants. La méthode a été généralisée par Dantzig

(1963) au problème de programmation convexe complètement décomposable. Lasdon (1968) a proposé de résoudre le problème (3.1) à l'aide d'une méthode de plus forte pente. Pour le cas linéaire, Grinold (1972), après avoir établi une équivalence entre la méthode de plus forte pente appliquée au problème (3.1) et la méthode primale-duale du simplexe appliquée à une reformulation du problème original, a obtenu une méthode duale généralisant l'algorithme de Balas (1966). Les méthodes d'Abadie et Williams (1963) et de Bell (1965) sont aussi des décompositions par les prix.

Les approches primales, ou de décomposition par les ressources, remplacent le problème original par un problème équivalent où les fonctions f_i du problème (3.1) sont les fonctions valeur (on dit encore fonctions de perturbation) associées à chacun des sous-systèmes du problème global supposé complètement décomposable. Nous avons vu à la section 2.3 comment obtenir ce problème équivalent. Les fonctions f_i peuvent prendre la valeur $+\infty$ si elles sont évaluées à l'extérieur de leur domaine. Le problème (3.1) possède alors des contraintes de partage du membre de droite des ressources communes entre les différents sous-systèmes. Celles-ci définissent le sous-ensemble V de $\mathbb{R}^n$. Elles ont comme avantage de produire des solutions intermédiaires réalisables, ce qui rend possible l'interruption prématurée des calculs. Zschau (1966), pour le problème de programmation linéaire avec structure angulaire par blocs, résout le problème (3.1) avec une méthode complexe qui est une adaptation des méthodes de plus forte pente. Silverman (1972a et 1972b) a généralisé les travaux de Zschau au cas convexe complètement décomposable. Geoffrion (1970a) a proposé trois méthodes pour résoudre le problème convexe complètement décomposable: une méthode de plans coupants, une méthode de plus forte pente, et une méthode de partitionnement de l'espace. Enfin, Grinold (1972) a obtenu une méthode de plus forte pente beaucoup simple que celle de Zschau.

L'approche de projection est utile lorsque le problème comporte des variables liantes. Elle consiste à fixer les valeurs de celles-ci afin de rendre le problème décomposable. C'est ce que nous avons fait au chapitre 2 avec les variables s_i du problème (2.3). C'est une approche très pratique pour le cas où des variables couplantes lient tous les sous-systèmes du problème. Les variables du problème (3.1) sont alors les variables couplantes et on dit que l'on projette le problème sur ces variables. L'algorithme de Benders (1962) procède ainsi. Le problème du chapitre 2 en était un de programmation linéaire, mais la méthode de Benders permet de traiter des problèmes mixtes dans lesquels la partie facile est linéaire continue, l'autre partie faisant intervenir des variables couplantes à valeurs discrètes et/ou incluses dans des termes non-linéaires. Une fois la projection faite, Benders résout le problème (3.1) avec une méthode de plans coupants.

On constate, de ce rapide survol, que deux approches sont fréquemment utilisées pour résoudre le problème équivalent (3.1): les méthodes de plans coupants et les méthodes de plus forte pente. Ceci correspond à deux points de vue différents. Le premier est global en ce sens que les hyperplans produits sont, par la convexité des fonctions, toujours situés sous l'épigraphe de la fonction approchée. Le second est local: on exploite la convexité des fonctions pour déterminer une direction de plus forte descente en minimisant la valeur de la dérivée directionnelle en un point x donné. Celle-ci existe alors pour toutes les directions $d \in \mathbb{R}^n$ lorsque le point x est dans l'intérieur relatif du domaine de la fonction. Elle vaut alors

$$\nabla f(x : d) = \sup \{ \langle s, d \rangle : s \in \partial f(x) \} \quad (3.2)$$

et elle prend toujours une valeur finie lorsque x est dans l'intérieur du domaine de f (Rockafellar, 1970: théorème 23.4). Sous cette dernière approche, la direction de

déplacement d est obtenue en résolvant le problème de minimax

$$\min_{d \in \mathbb{R}^n} \max_{s \in \partial f(x)} \langle s, d \rangle. \quad (3.3)$$

Le point de vue global produit des méthodes généralement plus simples et élégantes que celles associées au point de vue local. Les méthodes de plus forte pente sont en effet réputées pour leurs mauvaises propriétés de convergence, et ce, même lorsqu'elles sont appliquées à des problèmes d'optimisation différentiable (voir l'analyse de Zangwill (1969)).

Les méthodes classiques de décomposition, même lorsqu'elles manipulent des fonctions non-différentiables, n'ont pas été conçues comme des méthodes d'optimisation non-différentiable. Elles ne traitent que minimalement la non-différentiabilité du problème, cherchant davantage à l'éviter qu'à l'attaquer de front.

Le point de vue adopté dans cette thèse est qu'en reconnaissant pleinement que l'on cherche à résoudre un problème équivalent d'optimisation non-différentiable, des améliorations peuvent être obtenues en exploitant les idées de ce domaine relativement nouveau.

Les deux points de vue dégagés précédemment sont présents également en optimisation non-différentiable. Les deux premières méthodes connues en sont des exemples. Bien que proposée pour le cas différentiable, la méthode des plans sécants de Kelley (1960) et de Cheney-Goldstein (1959) fonctionne même si les fonctions ne sont pas différentiables. Elle consiste à approcher les fonctions par des linéarisations tangentielles affînées itérativement. Elle est réputée pour sa faible vitesse de convergence. Celle-ci peut être attribuée au fait que les déplacements proposés tendent

à produire des solutions assez éloignées de la région de l'espace où les linéarisations ont été obtenues et où les approximations peuvent être considérées fiables.

Les méthodes de sous-gradients, développées en ex Union-Soviétique (Polyak, 1967 et 1969; Ermoliev, 1966), sont probablement les premiers représentants de l'approche locale. Elles généralisent les méthodes de gradient en prenant comme direction de déplacement à chaque itération un sous-gradient (plutôt que le gradient) au point courant. De plus, les pas de déplacement sont déterminés, *a priori*, exogènement. Leur très grande simplicité constitue certainement leur principal attrait. Leur convergence est toutefois excessivement lente, de sorte que plusieurs autres mécanismes de choix des paramètres de déplacement ont été proposés (Shor, 1968; Goffin, 1977; Held, Wolfe et Crowder, 1974) dans le but d'en accélérer la convergence. Shor (1970) a aussi décrit une variante à métrique variable de la méthode. Minoux (1983) présente un survol de ces méthodes. Outre leurs mauvaises propriétés de convergence, elles n'ont pas de critère fiable d'arrêt. Contrairement à ce qui se passe avec des fonctions différentiables, il n'est pas possible d'utiliser la norme des sous-gradients dans un test d'arrêt. En effet, il se peut que tout en ayant atteint la solution optimale, l'algorithme soit incapable de trouver un sous-gradient nul ou suffisamment petit.

Mahey (1982 et 1986) a utilisé l'algorithme du sous-gradient dans plusieurs méthodes de décomposition par les prix et/ou par les ressources afin de résoudre le problème de programmation linéaire avec structure angulaire par blocs. Il a aussi imaginé des algorithmes composites alternant entre des itérations de l'algorithme du sous-gradient et des itérations d'un algorithme de plans coupants. Le but visé était d'exploiter les oscillations de la méthode du sous-gradient autour de l'optimum pour identifier les points extrêmes actifs de la solution optimale du problème.

Outre leurs problèmes de convergence, les méthodes de sous-gradients et de plans sécants ont aussi en commun de ne pas être des méthodes de descente dans lesquelles les solutions produites à chaque itération sont réalisables et font décroître la valeur de la fonction économique. Remarquons cependant que pour des pas suffisamment petits la méthode de sous-gradients produit des solutions qui font décroître la distance qui les sépare de l'ensemble des solutions optimales (Lemaréchal, 1989: lemme 5.1).

Les années 70 ont vu naître une classe de méthodes destinées à résoudre des problèmes d'optimisation non-différentiable tout en préservant la propriété de descente. On les désigne souvent sous le vocable de *méthodes de faisceaux* bien que ce terme soit approprié pour une partie seulement d'entre elles. Les premières méthodes proposées (Lemaréchal, 1975; Wolfe, 1975; Mifflin, 1977a), aussi appelées méthodes de sous-gradients conjugués, sont de nature locale et traitent surtout le problème non contraint. Elles construisent des approximations, améliorées itérativement, du sous-différentiel de la fonction économique au point courant. Ces approximations sont affinées tant et aussi longtemps qu'elles ne permettent pas d'obtenir une direction de descente (pas nécessairement de plus forte pente) assurant une décroissance "suffisante" de la valeur de la fonction économique. On obtient celle-ci en recherchant le sous-gradient de plus faible norme (euclidienne) de l'approximation courante du sous-différentiel. Ceci est fait en résolvant un problème de programmation quadratique de structure particulière. La direction est alors l'opposé de ce sous-gradient. Dans ces méthodes, les sous-gradients accumulés sont considérés comme des sous-gradients au point courant, même si ce n'est pas le cas. Pour cette raison, elles comportent des mécanismes d'élimination de vieux sous-gradients. Elles sont assez sensibles à ces mécanismes car une élimination trop rapide ou radicale de ceux-ci peut entraver leur convergence. Polak, Mayne et Wardi (1983) ont aussi présenté

un algorithme ayant des caractéristiques générales similaires. Lemaréchal (1975) et Wolfe (1975) ont proposé des modifications de leurs méthodes dans le but de préserver la convergence tout en ne conservant qu'un nombre fini de sous-gradients. Ils y parviennent en effectuant une agrégation des sous-gradients accumulés avant de procéder à une épuration de ceux-ci. Le "sous-gradient agrégé" est ajouté à la liste des sous-gradients restants suite à l'épuration. La direction trouvée sert à effectuer une optimisation unidimensionnelle. Chacun des algorithmes mentionnés possède sa propre procédure de recherche linéaire. Elles ont en commun de ne pas rechercher un pas "optimal" car cette stratégie est non convergente en optimisation non différentiable. Lemaréchal (1981) fait une discussion des techniques de recherche linéaire en optimisation non-différentiable.

Les méthodes précédentes sont basées sur l'hypothèse que les sous-gradients engendrés en des points différents sont des sous-gradients au point courant, ce qui n'est pas le cas. Par contre, chacun de ces sous-gradients est un ε -sous-gradient de la fonction en tout point x où son sous-différentiel n'est pas vide, à condition que la valeur de la constante ε soit suffisamment élevée. On la détermine par différence entre la valeur de la fonction en ce point et la valeur, au même point, de la linéarisation construite avec le sous-gradient (engendré ailleurs). Des méthodes exploitant cette idée ont été proposées (Lemaréchal, 1978a; Lemaréchal, Strodiot et Bihain, 1981; Strodiot, Nguyen et Heukemes, 1983). On les qualifie de méthodes d' ε -plus-forte descente (" ε -steepest descent"). A chaque itération, elles construisent une approximation interne de l' ε -sous-différentiel de la fonction au point courant et y recherchent l' ε -sous-gradient de plus faible norme (euclidienne). Les problèmes quadratiques à résoudre comportent une contrainte additionnelle garantissant que la solution trouvée est bien un ε -sous-gradient. Celle-ci remplace le mécanisme d'épuration des méthodes précédentes. Si la valeur optimale de la fonction économique de ce

problème quadratique est nulle, alors la solution courante x^k est ε -optimale. Le principal inconvénient de l'approche vient de la difficulté qu'il y a à définir des règles d'ajustement de la valeur ε^k du paramètre ε à l'itération k qui garantissent la convergence vers une solution optimale (plutôt qu' ε -optimale). Lemaréchal, Strodriot et Bihain (1981) ont proposé une procédure d'agrégation des sous-gradients permettant d'exploiter leur méthode sans devoir accumuler toute l'information engendrée.

Une troisième classe de méthodes de descente, basée cette fois-ci sur le point de vue global, est issue d'une méthode proposée par Lemaréchal (1978b) pour le cas non contraint. Il s'agit d'ajouter un terme quadratique, le carré de la norme du déplacement, à la fonction économique du problème de programmation linéaire des méthodes de plans sécants de Kelley-Cheney-Goldstein. Ce terme agit comme une pénalisation qui empêche l'algorithme de proposer des solutions trop éloignées de la région où l'approximation polyédrique de la fonction est la meilleure. La recherche linéaire est ici très simple. On accepte le nouveau point proposé s'il produit une décroissance "suffisante" de la fonction économique. Sinon, la solution courante ne change pas et on engendre un sous-gradient, et une coupe, au point proposé. Le dual du problème de recherche d'une direction d'amélioration est un problème de programmation quadratique où l'on minimise une fonction formée de deux termes: l'un correspondant au carré de la norme d'une combinaison convexe des ε -sous-gradients et l'autre à la combinaison convexe correspondante des erreurs de linéarisation au point courant. La méthode a été généralisée au cas contraint par Mifflin (1982). Kiwiel (1983, 1984 et 1985) a par la suite modifié ces méthodes en proposant des procédures d'agrégation et d'épuration des sous-gradients afin de préserver la convergence tout en ne requérant qu'une capacité finie de stockage de l'information.

L'ouvrage de Kiwiel (1985) contient une analyse détaillée de toutes les méthodes discutées ici. Plusieurs autres extensions de ces méthodes sont discutées par Kiwiel (1989a). Parmi, celles-ci, mentionnons la version proximale de l'algorithme du sous-gradient agrégé dans laquelle le poids associé au terme quadratique de la fonction économique du problème de recherche d'une direction de descente n'est plus fixé à 1, mais peut varier à chaque itération. Cette méthode (Kiwiel, 1990) est nettement plus rapide que celle du sous-gradient agrégé (Kiwiel, 1983). Schramm et Zowe (1992) ont aussi proposé une méthode semblable. Hiriart-Urruty et Lemaréchal (1993b) présentent une analyse plus générale de ces méthodes "proximales". Lemaréchal (1992) observe qu'elles peuvent être utilisées dans une approche duale de décomposition. Il met en évidence les liens qui existent entre celles-ci et l'algorithme de proximité, proposé par Martinet (1970) puis popularisé par Rockafellar (1976), et la méthode du lagrangien augmenté (Hestenes, 1969; Powell, 1969).

La plupart des méthodes précédentes n'exploitent pas la structure du problème lorsqu'on les applique à la résolution du problème (3.1). Kiwiel (1987) a proposé une méthode, inspirée de sa méthode de sous-gradients agrégés, pour résoudre le problème (3.1) lorsque toutes les fonctions sont à valeurs finies. Elle généralise une méthode de décomposition proposée par Nurminsky et Balabanov (1983) pour résoudre des problèmes de programmation linéaire de grande taille issus de la modélisation énergétique. Elle ne semble pas très efficace car la détermination d'une direction se fait en résolvant non pas un, mais r problèmes de programmation quadratique (i.e. un pour chacune des r fonctions f_i du problème (3.1)).

Enfin, Lemaréchal, Nurminskii et Nesterov (1995) ont récemment proposé une quatrième classe de méthodes de faisceaux basées sur le point de vue global. Ces méthodes travaillent sur les ensembles niveaux ("level sets"). Elles minimisent la

longueur du déplacement sous la contrainte que la fonction polyédrique approchant la fonction convexe à minimiser prenne une valeur inférieure à un certain seuil ajusté itérativement par l'algorithme. Cet ajustement peut être fait très simplement, du fait de la relative "stabilité" des ensembles niveaux. Il s'agit des premières méthodes de faisceaux pour lesquelles des estimés de complexité ont été proposés.

S'inspirant des travaux de Clarke (1975) qui a généralisé et/ou adapté de nombreux résultats d'analyse convexe au cas des fonctions localement lipschitziennes (définies à l'annexe B), quelques auteurs (Feuer, 1974; Mifflin, 1977a et 1977b; Bihain, 1984) ont proposé des méthodes de faisceaux pour des problèmes de minimisation formulés à l'aide de fonctions appartenant à des sous-ensembles de l'ensemble des fonctions localement lipschitziennes. Ces classes de fonctions incluent les fonctions convexes. Plusieurs des méthodes mentionnées précédemment peuvent être adaptées à certains de ces sous-ensembles.

Bien que la décomposition de problèmes de programmation mathématique soit probablement la principale source de problèmes d'optimisation non-différentiable, il n'existe que très peu de publications traitant de l'application des méthodes de faisceaux à ces problèmes. Robinson (1986) a considéré un cas particulier du problème (3.1) où l'ensemble V est remplacé par des contraintes linéaires de partage des ressources. Le problème de programmation linéaire avec structure angulaire par blocs peut être reformulé comme un tel problème. La formulation duale de celui-ci possède la structure du problème (3.1), où $V = \mathbb{R}^n$. Il a proposé d'utiliser une méthode d' ε -plus-forte descente pour le résoudre et fait remarquer que la connaissance d'une solution duale optimale ne garantit pas la découverte d'une solution primale optimale. Medhi (1987 et 1994), en utilisant l'approche de Robinson, a déterminé, *a posteriori*, des estimés de l'erreur sur la valeur optimale de la fonction

économique ainsi que de l'écart entre les objectifs des fonctions primale et duale évalués aux solutions primale et duale approximatives. Il a testé l'approche sur des problèmes de programmation linéaire ayant la structure angulaire par blocs. Son algorithme a mieux fait que la mise en oeuvre avancée de l'algorithme de Dantzig et Wolfe par Ho et Louie (1981 et 1983). Enfin, Robinson (1989) a déterminé des conditions devant être satisfaites par le problème pour que l'algorithme d' ε -plus-forte pente converge.

Loulou, Savard et Lavigne (1992) ont aussi présenté une méthode de décomposition, pour le moment heuristique, destinée à résoudre des problèmes de programmation linéaire correspondants à des modèles où il peut y avoir échanges entre les différents sous-systèmes. La méthode a été conçue pour résoudre le problème (2.7). Elle a des points communs avec celle de Nurminsky et Balabanov (1983) mais elle en diffère par ses approximations non polyédriques, inspirées de la méthodologie PIES (Hogan, 1975). Ces méthodes, ainsi que celles de Mahey (1982 et 1986), sont primales-duales.

Les méthodes que nous allons concevoir sont des extensions de l'algorithme de Kiwiel (1990) au problème de la minimisation d'une somme de fonctions convexes, pouvant prendre la valeur infinie, sur un ensemble convexe général (non défini explicitement par des fonctions). Le terme de pénalisation du problème de recherche d'une direction ne sera pas nécessairement quadratique et la norme utilisée ne sera pas forcément euclidienne. La méthode classique de décomposition la plus rapprochée est peut-être la technique d'approximations tangentielles de Geoffrion (1970a).

On peut montrer (Hiriart-Urruty et Lemaréchal, 1993b) que la direction proposée par le problème maître des méthodes de faisceaux issues des travaux de Lemaréchal (1978b) peut aussi être obtenue en résolvant un problème équivalent obtenu en éliminant le terme quadratique de la fonction économique et en l'utilisant pour construire une contrainte délimitant une région de confiance. Les solutions proposées par ces méthodes ne sont que les points qui minimisent l'approximation polyédrique (externe) du problème dans cette région de confiance. C'est en ces endroits que sont engendrées les coupes (les sous-gradients) qui améliorent les approximations. Goffin et Vial (1990), puis Goffin, Haurie et Vial (1992) ont proposé plutôt de déterminer les prochains points en minimisant le volume d'un ensemble (compact) de localisation défini comme étant la partie de l'épigraphe de l'approximation de la fonction économique située sous la meilleure valeur trouvée de la fonction économique. Cet ensemble contient toujours les solutions optimales du problème. Le but de la méthode est de produire les coupes les plus profondes possible afin d'obtenir de grandes réductions du volume de l'ensemble de localisation. Pour y parvenir, ils suggèrent de les engendrer aux points déterminés par le centre analytique, un concept introduit par Sonnevend (1985), de l'ensemble de localisation. Celui-ci est déterminé par résolution d'un problème de programmation non-linéaire (différentiable) résolu à l'aide d'un algorithme projectif (de Ghellinck et Vial, 1986) qui est une variante de la méthode de points intérieurs de Karmarkar (1984). Cette méthode a été utilisée pour la décomposition de problèmes de programmation linéaire (Goffin, Haurie, Vial et Zhu, 1993), pour la résolution de problèmes de programmation géométrique (Bahn, Goffin, Vial et Du Merle, 1994), de programmation stochastique (Bahn, Du Merle, Goffin et Vial, 1994), et de multiflots non-linéaires (Goffin, Gondzio, Sarkissian et Vial, 1994). Remarquons qu'une méthode de plans sécants proposée par Elzinga et Moore (1975) est du même esprit. Le centre de l'ensemble de localisation est évalué plus simplement. Il ne jouit cependant pas de la propriété d'unicité du

centre analytique. La méthode s'inspire d'idées de Nemhauser et Widhelm (1971) qui ont remarqué que l'utilisation de coupes non-normalisées a quelquefois comme inconvénient de rendre actives des coupes superflues, forçant alors l'algorithme à effectuer des itérations additionnelles afin de les rendre inactives. Ils attribuent ce comportement au fait que des coupes non-normalisées ne permettent pas d'aller très profondément à l'intérieur de la région de l'espace qu'elles délimitent. On vient à bout du problème en normalisant les coupes. Les résultats préliminaires de O'Neill et Widhelm (1975) ne sont toutefois pas très concluants.

Cette thèse n'est pas basée sur les idées du paragraphe précédent.

CHAPITRE 4

Accélération et stabilisation des algorithmes de plans sécants en programmation mathématique convexe

La formulation équivalente (2.4)-(2.6) du problème de programmation linéaire (2.1) se prête naturellement à une résolution par un algorithme de plans sécants. Ce chapitre contient une brève analyse critique de cette méthodologie. Il présente des concepts proposés dans le cadre des méthodes de faisceaux afin de stabiliser les méthodes de plans coupants et d'en faire des méthodes de descente. Ces concepts sont utilisés dans trois méthodes schématiques destinées à résoudre autant de classes de problèmes, la dernière méthode étant originale et représentative des problèmes qui nous intéressent. Les deux prochains chapitres consisteront en une mise-en-oeuvre, à l'aide d'approximations polyédriques, de cette troisième méthode. L'application des méthodes obtenues au problème de programmation linéaire (2.1) du chapitre 2 ne sera possible que si l'on sait exploiter les particularités des fonctions $v_i(\cdot)$ définies en (2.5)-(2.6). Pour cette raison, le chapitre se termine par une étude de certaines propriétés de ces fonctions.

4.1 Méthodes de plans sécants: description et lacunes

Nous avons vu que le problème de programmation linéaire de grande dimension qui nous intéresse possède une formulation équivalente qui consiste à minimiser une fonction convexe linéaire par morceaux, et donc, non différentiable. Comme il est

possible de considérer des variations du problème comportant aussi des contraintes dans la formulation équivalente, nous nous intéresserons au problème suivant:

$$\begin{aligned} & \underset{x}{\text{minimiser}} \ f(x) \\ & \text{sous les contraintes} \\ & \qquad g_j(x) \leq 0, \quad j \in \{1, \dots, m\}, \end{aligned} \tag{4.1}$$

où les fonctions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ et $g_j : \mathbb{R}^n \rightarrow \mathbb{R}, j = 1, \dots, m$, sont convexes (non nécessairement différentiables). Nous désignerons par $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ la fonction vectorielle associée aux m contraintes du problème.

Les algorithmes de décomposition s'appliquant aux problèmes de programmation linéaire de grandes dimensions étant pour la plupart des méthodes où l'on engendre systématiquement des colonnes ou des contraintes, dites coupes, nous nous intéresserons plus particulièrement aux *méthodes de plans sécants* ("cutting plane methods").

Ces méthodes, bien que proposées initialement pour résoudre des problèmes de programmation mathématique convexe où les fonctions sont différentiables, furent en fait les premières à pouvoir résoudre une grande classe de problèmes en optimisation non-différentiable. Elles datent des années 1959-60.

Puisque le problème (4.1) équivaut à

$$\begin{aligned} & \underset{x, x_{n+1}}{\text{minimiser}} \ x_{n+1} \\ & \text{sous les contraintes} \\ & \qquad f(x) - x_{n+1} \leq 0, \\ & \qquad g_j(x) \leq 0, \qquad j \in \{1, \dots, m\}, \end{aligned} \tag{4.2}$$

alors les méthodes de plans sécants destinées à le résoudre sont conçues pour résoudre

le problème un peu plus général qui suit:

$$\begin{aligned}
 & \underset{x}{\text{minimiser}} \langle c, x \rangle \\
 & \text{sous les contraintes} \\
 & g_j(x) \leq 0, \quad j \in \{1, \dots, m\}, \\
 & x \in S,
 \end{aligned} \tag{4.3}$$

où S est un sous-ensemble convexe compact et non-vidé de $\mathbb{R}^n$. Le plus souvent, S est un polytope défini par $S \triangleq \{x \in \mathbb{R}^n : Ax \leq b\}$. Nous supposons que le problème est réalisable.

Le problème (4.3) peut être traité par un algorithme proposé simultanément, et indépendamment, par Kelley (1960) et Cheney-Goldstein (1959). C'est l'algorithme KCG présenté ci-dessous:

ALGORITHME KCG
(Méthode des plans sécants
de Kelley et de Cheney-Goldstein)

(0) (INITIALISATION) Poser $k = 1$.

(1) (DÉTERMINATION DE LA PROCHAINE SOLUTION APPROCHÉE)

Résoudre le problème de programmation linéaire ci-dessous:

$$\begin{aligned}
 & \underset{x}{\text{minimiser}} \langle c, x \rangle \\
 & \text{sous les contraintes} \\
 & \langle \alpha^\ell, x \rangle \leq \beta^\ell, \quad \ell \in \{1, \dots, k-1\}, \\
 & x \in S.
 \end{aligned} \tag{4.4}$$

et désigner par x^k la solution optimale trouvée.

Remarque: Le premier groupe de contraintes n'existe pas lorsque $k = 1$.

(2) (TEST D'ARRÊT)

Si $g(x^k) \leq 0$, alors x^k est une solution optimale du problème original (4.3) et on arrête.

(3) (AMÉLIORATION DE LA REPRÉSENTATION APPROCHÉE DE L'ENSEMBLE DES SOLUTIONS RÉALISABLES)

Soit $r \in \{1, \dots, m\}$ l'indice de la (ou d'une) contrainte la plus violée. Former le problème (4.4) de l'itération $k + 1$ en ajoutant la contrainte suivante au problème courant:

$$g_r(x^k) + \langle s_r(x^k), x - x^k \rangle \leq 0,$$

c'est-à-dire

$$\langle \alpha^k, x \rangle \leq \beta^k,$$

où

$$\alpha^k = s_r(x^k) \in \partial g_r(x^k)$$

et

$$\beta^k = \langle s_r(x^k), x^k \rangle - g_r(x^k).$$

Faire $k \leftarrow k + 1$ et retourner à l'étape 1.

Remarque: Le sous-gradient $s_r(x^k)$ définissant la contrainte à ajouter ne peut être nul car si tel était le cas, x^k minimiserait $g_r(\cdot)$ et le problème original serait impossible, ce qui est contraire à notre hypothèse.

Dans ce qui suit, nous supposerons que les fonctions sont convexes à valeurs finies. Remarquons que, par la première partie du théorème B.5 (de l'annexe B), l'étape 3 de l'algorithme KCG est bien définie.

Supposons que le critère d'arrêt de l'étape 2 ne soit jamais satisfait. L'algorithme engendre alors une suite $\{x^k\}_{k=1}^{\infty}$ dans un ensemble compact S , et on peut en extraire au moins une sous-suite convergente. Le résultat suivant énonce la convergence de la méthode.

Théorème 4.1.1 *Supposons que les fonctions convexes $g_j, j \in \{1, \dots, m\}$, du problème (4.3) soient à valeurs finies et supposons que l'algorithme KCG engendre une suite de points $\{x^k\}$. Alors tout point d'accumulation de cette suite est une solution optimale du problème (4.3).*

Démonstration: Luenberger (1984) et Zangwill (1969). ■

Cet algorithme, comme toutes les autres méthodes de plans sécants a besoin pour converger d'une hypothèse de compacité. C'est ce qui garantit l'existence de points d'accumulation de la suite $\{x^k\}$ produite, de même que l'existence de solutions finies pour les sous-problèmes (4.4).

Les méthodes de plans sécants ont pour particularité d'être instables. Cette instabilité se manifeste d'au moins deux façons: (i) un comportement "erratique" caractérisé par des changements brutaux de direction entre deux itérations successives, comportement très apparent lors des premières itérations alors que les solutions obtenues par résolution des sous-problèmes (4.4) sont presque toujours

systématiquement collées à la frontière de l'ensemble S ; et (ii) une absence de monotonie des valeurs de la fonction économique et/ou de la distance du point courant à l'ensemble des solutions optimales.

Cette instabilité n'est qu'exacerbée lorsqu'on élimine l'hypothèse de compacité du sous-ensemble S . En effet, l'application de l'algorithme KCG au problème ainsi obtenu devient problématique du fait de la possibilité de l'inexistence de solutions optimales finies aux sous-problèmes (4.4), particulièrement pour de faibles valeurs de k , et ce, même si le problème original possède une solution optimale finie.

La plupart des auteurs ne se sont pas préoccupés de cette difficulté: ils la contournent en affirmant que les problèmes réels prennent des valeurs finies que l'on peut encadrer *a priori* par des bornes, les solutions non bornées étant le plus souvent le fait d'une erreur de modélisation. Ils définissent alors l'ensemble compact S à l'aide de ces bornes et appliquent l'algorithme KCG ou une autre méthode de plans sécants.

Bien que ceci soit la plupart du temps vrai, il n'est pas toujours évident de fixer les bornes de chacune des variables à des valeurs qui ne soient pas exagérément éloignées. Nous aimerions pouvoir résoudre le problème (4.3) et lui trouver une solution optimale finie s'il en possède une, ou dire qu'il n'en possède pas sans devoir recourir à la compacité de l'ensemble S .

4.2 Stabilisation des méthodes de plans sécants: cas non contraint

Hiriart-Urruty et Lemaréchal (1993b) ont fait un survol plus approfondi des problèmes d'instabilité des algorithmes de plans sécants. Ils ont aussi présenté les ingrédients de base d'un algorithme stabilisé dans lequel on retrouve les concepts de *région de confiance* ("trust region") et de *descente*. Ayant constaté que les suites de points provoquant une décroissance monotone de la fonction économique exhibent habituellement un comportement plus "régulier", ils proposent de limiter l'optimisation à une région réduite de l'espace tant qu'une décroissance raisonnable n'a pas été réalisée. De cette façon, ils parviennent à traiter des problèmes (non contraints) sans devoir recourir à l'artifice de la compacité de l'ensemble S .

Le problème qu'ils traitent est le suivant:

$$\underset{x \in \mathbb{R}^n}{\text{minimiser}} f(x)$$

où, comme dans le problème (4.1), $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est une fonction convexe non nécessairement différentiable. A chaque itération, que l'on désignera par la lettre k , les méthodes proposées sont caractérisées par:

- un *modèle* convexe φ^k , habituellement une fonction linéaire par morceaux, de la fonction économique f ;
- un *centre de stabilité* x^k , notre meilleur candidat au titre de solution optimale, dans un voisinage duquel on espère que le modèle φ^k de la fonction f est adéquat;

- une norme $\| \cdot \|_k$ qui peut changer avec les itérations; et
- un candidat y^{k+1} obtenu en recherchant un compromis entre réduire la valeur du modèle φ^k de la fonction économique f , i.e. tel que

$$\varphi^k(y^{k+1}) < \varphi^k(x^k),$$

et ne pas s'éloigner trop du centre de stabilité x^k , i.e. tel que

$$\| y^{k+1} - x^k \|_k$$

est petit.

Le point y^{k+1} deviendra notre prochain centre de stabilité, i.e. $x^{k+1} = y^{k+1}$, lorsque la diminution $\varphi^k(x^k) - \varphi^k(y^{k+1})$ prédite par le modèle n'est pas trop différente de la diminution réelle $f(x^k) - f(y^{k+1})$ de la fonction économique. Plus spécifiquement, on exigera que la diminution réelle soit au moins égale à une fraction $\alpha > 0$ de la diminution prédite. Si tel est le cas, le modèle φ^k est considéré comme suffisamment adéquat dans un voisinage de x^k et y^{k+1} deviendra notre prochain centre de stabilité. Sinon, on améliore le modèle qui deviendra φ^{k+1} et on ne change pas le centre de stabilité, i.e. $x^{k+1} = x^k$. Ces considérations sont résumées dans l'algorithme suivant:

ALGORITHME SCHEMATIQUE STABILISE I

(Problème non contraint)

(0) (INITIALISATION) Choisir une solution initiale $x^1 \in \mathbb{R}^n$, un coefficient de descente $\alpha \in]0, 1[$ et un seuil d'arrêt $\underline{\delta} \geq 0$. Poser $k = 1$.

(1) (CONSTRUCTION D'UN MODÈLE φ^k ET CHOIX D'UNE NORME $\| \cdot \|_k$)

(2) (OBTENTION D'UNE NOUVELLE SOLUTION CANDIDATE) Résoudre le problème associé au modèle stabilisé afin d'obtenir une solution y^{k+1} pas trop éloignée du centre de stabilité x^k . Evaluer $\delta^k = \varphi^k(x^k) - \varphi^k(y^{k+1}) \geq 0$.

Si $\delta^k \leq \underline{\delta}$, on arrête.

(3) (EVALUATION DE LA QUALITÉ DU MODÈLE DE LA FONCTION ÉCONOMIQUE)

Si

$$f(x^k) - f(y^{k+1}) \geq \alpha \delta^k,$$

poser $x^{k+1} = y^{k+1}$; sinon poser $x^{k+1} = x^k$. Remplacer k par $k + 1$ et retourner à l'étape 1.

Remarque: En pratique, les modèles φ^k seront des approximations polyédriques obtenues en ajoutant au modèle précédent φ^{k-1} une pièce linéaire engendrée au point y^k , et on aura $\varphi^k(x^k) = f(x^k)$ et $\varphi^k(y^j) = f(y^j), \forall j \leq k$.

Remarque: Le paramètre α doit être fixé à une valeur comprise entre 0 et 1. Il semble que des valeurs élevées soient préférables lorsque la fonction est différentiable et que de faibles valeurs doivent être privilégiées lorsque la fonction ne l'est pas.

Mifflin (1982), pour son algorithme, recommande de choisir $\alpha < 1/2$ lorsque la fonction économique est continuellement différentiable. Lemaréchal (1981) recommande $\alpha = 0,7$ lorsque f est continuellement différentiable et $\alpha = 0,2$ sinon. Madsen (1975) optimise une fonction non-différentiable et prend $\alpha = 0,01$. Schramm et Zowe (1992) dans leurs tests avec des fonctions non-différentiables ont choisi des valeurs comprises entre 0,01 et 0,1.

Définition. Nous dirons que l'algorithme effectue un *pas de descente* lorsque le centre de stabilité est changé. Lorsque ce n'est pas le cas, l'algorithme effectue un *pas nul*.

Il semble que la stabilisation de l'algorithme des plans sécants passe par l'utilisation combinée des concepts de descente et de région de confiance. En effet, la descente seule n'est pas suffisante pour garantir la convergence d'un algorithme de descente. Il existe plusieurs exemples en optimisation non-différentiable où des extensions de la méthode de plus forte pente, initialement proposée par Cauchy (1847), convergent vers un point qui n'est pas une solution optimale du problème. Ainsi, Lemaréchal (1989) en présente un dans le cas nonlinéaire convexe. Il n'est pas nécessaire que la fonction économique soit partout nonlinéaire pour que ce phénomène se produise. L'exemple de Wolfe (1974) avec une fonction affine par morceaux le prouve. Les méthodes de plus forte pente ne sont pas les seules à souffrir de ce problème. Ainsi, Fletcher (1972) présente l'exemple d'une fonction polynomiale continue traitée avec la méthode de Newton: la suite convergente $\{f(x^k)\}$ décroît strictement d'une itération à l'autre sans toutefois que la suite $\{x^k\}$ converge. Cette suite possède deux points d'accumulation qui ne sont pas des solutions optimales,

mêmes locales, du problème.

L'utilisation du concept de région de confiance sans se préoccuper de descente est possible si l'on conserve l'hypothèse de compacité du sous-ensemble S (Fukushima, 1983 et 1984). La compacité de S sert à empêcher des variations trop extrêmes de la suite déjà erratique des y^{k+1} , ce qui tend à prouver que les méthodes résultantes ne sont pas stables.

4.3 Stabilisation des méthodes de plans sécants: cas des contraintes explicites

En définissant la fonction convexe $G : \mathbb{R}^n \rightarrow \mathbb{R}$ par

$$G(x) \triangleq \max\{g_j(x), j \in \{1, \dots, m\}\}, \quad \forall x \in \mathbb{R}^n,$$

on ramène le problème (4.1) à un problème à une seule contrainte:

$$\begin{aligned} & \underset{x}{\text{minimiser}} \ f(x) \\ & \text{sous la contrainte} \\ & \qquad G(x) \leq 0. \end{aligned} \tag{4.5}$$

Résoudre ce problème équivaut à résoudre le problème ci-dessous:

$$\underset{x \in \mathbb{R}^n}{\text{minimiser}} \ h(x) \tag{4.6}$$

où

$$h(x) \triangleq \max\{f(x) - f^*, G(x)\} \tag{4.7}$$

et f^* est la valeur optimale de la fonction économique du problème (4.5).

La valeur minimale de la fonction convexe $h : \mathbb{R}^n \rightarrow \mathbb{R}$ est zéro. On ne peut cependant pas résoudre le problème (4.6) car, sauf pour de très rares exceptions, on ne connaît pas *a priori* la valeur de f^* .

On peut toutefois, $\forall x, y \in \mathbb{R}^n$, définir la fonction $H : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ par

$$H(y; x) \triangleq \max\{f(y) - f(x), G(y)\}.$$

Remarquons que $H(x; x) = 0$ lorsque x est réalisable. Si tel est le cas, alors $H(y; x) < H(x; x) = 0$ implique que $f(y) < f(x)$ et $G(y) < 0$, ce qui signifie que y est une solution réalisable de plus faible coût que x . Pour cette raison, on qualifie H de *fonction d'amélioration*.

Kiwiel (1985) a démontré, sous l'hypothèse que la condition de Slater est vérifiée (i.e. $\exists \bar{x} \in \mathbb{R}^n$ tel que $G(\bar{x}) < 0$), que $\bar{x} \in \mathbb{R}^n$ est une solution optimale du problème (4.5) si et seulement si

$$\min\{H(y; \bar{x}) : y \in \mathbb{R}^n\} = H(\bar{x}; \bar{x}) = 0, \quad (4.8)$$

c'est-à-dire si et seulement si

$$0 \in \partial H(\bar{x}; \bar{x}), \quad (4.9)$$

où $H(\cdot; \bar{x})$ est une fonction convexe de $\mathbb{R}^n$ vers $\mathbb{R}$, $\bar{x}$ étant fixé.

On peut alors adapter l'algorithme de la section précédente en substituant un modèle Ψ^k de la fonction $H(\cdot; x^k)$ au modèle φ^k utilisé pour le cas non contraint. A l'itération k , on posera

$$\Psi^k(x) = \max\{\varphi^k(x) - \varphi^k(x^k), \gamma^k(x)\}, \quad \forall x \in \mathbb{R}^n,$$

où $\gamma^k : \mathbb{R}^n \rightarrow \mathbb{R}$ est le modèle de la fonction G à la $k^{\text{ème}}$ itération. L'algorithme général est le suivant:

ALGORITHME SCHEMATIQUE STABILISE II

(Problème avec contraintes explicites)

- (0) (INITIALISATION) Choisir une solution initiale réalisable $x^1 \in \mathbb{R}^n$, un coefficient de descente $\alpha \in]0, 1[$ et un seuil d'arrêt $\underline{\delta} \geq 0$. Poser $k = 1$.
- (1) (CONSTRUCTION DU MODÈLE Ψ^k ET CHOIX D'UNE NORME $\| \cdot \|_k$) Obtenir les modèles φ^k et γ^k en améliorant les modèles φ^{k-1} et γ^{k-1} avec l'information additionnelle récoltée au point y^k . Poser

$$\Psi^k(x) = \max\{\varphi^k(x) - \varphi^k(x^k), \gamma^k(x)\}.$$

- (2) (OBTENTION D'UNE NOUVELLE SOLUTION CANDIDATE) Résoudre le problème associé au modèle stabilisé afin d'obtenir une solution y^{k+1} pas trop éloignée du centre de stabilité x^k .
- (3) (TEST D'ARRÊT) Evaluer $\delta^k = \Psi^k(x^k) - \Psi^k(y^{k+1}) \geq 0$.
Si $\delta^k \geq \underline{\delta}$, on arrête.
- (4) (EVALUATION DE LA QUALITÉ DU MODÈLE Ψ^k) Si

$$H(x^k; x^k) - H(y^{k+1}; x^k) = -H(y^{k+1}; x^k) \geq \alpha\delta^k,$$

poser $x^{k+1} = y^{k+1}$, sinon poser $x^{k+1} = x^k$.

- (5) (MISE-À-JOUR DU COMPTEUR D'ITÉRATIONS) Remplacer k par $k + 1$ et retourner à l'étape 1.

4.4 Stabilisation des méthodes de plans sécants: cas des contraintes ensemblistes

Considérons un problème semblable aux problèmes (4.1) et/ou (4.5) où l'ensemble des solutions réalisables n'est plus défini à l'aide de fonctions convexes, mais avec un ensemble (convexe). Plus précisément, soit le problème

$$\underset{x \in C}{\text{minimiser}} f(x) \quad (4.10)$$

où $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est convexe *étendue* et C est un ensemble convexe fermé *quelconque*.

L'hypothèse selon laquelle C est quelconque est importante. Elle diffère du cas usuel en programmation mathématique où des contraintes du type $x \in C$ sont utilisées pour décrire des ensembles simples comme ceux découlant de la non-négativité des variables ou de l'imposition de bornes à certaines variables. Dans ce cas, plusieurs algorithmes traitent implicitement ces contraintes.

Les ensembles C considérés ici sont plus complexes et *a priori* inconnus. Il pourrait s'agir par exemple du domaine d'une fonction valeur comme celles considérées au chapitre 2. Ce domaine est défini par un très grand nombre de contraintes linéaires *a priori* inconnues.

On peut, en utilisant la proposition B.13 de l'annexe B remplacer le problème (4.10) par le problème non contraint

$$\underset{x \in \mathbb{R}^n}{\text{minimiser}} \chi(x) \triangleq f(x) - f^* + I_D(x) \quad (4.11)$$

où $\chi : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est une fonction convexe étendue telle que $\chi(x) = +\infty$ si $x \notin D \triangleq (C \cap \text{dom } f)$. Tout comme dans la section précédente, la valeur minimale de la fonction convexe à minimiser est nulle. On ne peut cependant pas résoudre le problème (4.11) car, sauf pour de très rares exceptions, on ne connaît pas *a priori* la valeur de f^* .

Ce problème non contraint ne peut être traité par l'algorithme schématique I à moins de choisir, à l'étape d'initialisation, un $x^1 \in D$, car sinon le test de l'étape 3 est indéfini lorsque $y^2 \notin D$ (en effet, l'expression $+\infty - \infty$ n'est pas définie en arithmétique étendue). De plus, l'amélioration du modèle de la fonction économique devient problématique lorsque ce modèle est une approximation polyédrique obtenue par linéarisation à l'aide de sous-gradients, aux points $y^{k+1}, k \geq 1$. En effet, la fonction $\chi(\cdot)$ n'a pas de sous-différentiel aux points $y \notin D$ (théorème B.5(d)).

L'approche de la section précédente, basée sur la minimisation de la fonction d'amélioration n'est pas valable dans ce cas ci, car

$$H(y; x) = \max\{f(y) - f(x), I_D(y)\} \geq 0, \quad \forall x, y \in D.$$

L'algorithme schématique II s'arrêterait dès qu'une solution réalisable aurait été trouvée car la condition de Slater n'est pas vérifiée par la fonction indicatrice I_D . De plus, il faudrait aussi exiger que $y^2 \in D$ pour que les expressions et le test de l'étape 4 soient définis.

Nous proposons de modifier l'algorithme schématique I de façon à obtenir une méthode de directions réalisables dans laquelle tous les centres de stabilité x^k sont réalisables. Pour cela, nous supposerons l'existence d'un oracle qui permet

d'améliorer l'approximation $\Gamma^{k-1} : \mathbb{R}^n \rightarrow \mathbb{R}$ de la fonction indicatrice I_D lorsque $y^k \notin D$ et qui améliore l'approximation ϕ^{k-1} de la fonction économique f lorsque $y^k \in D$. L'algorithme tente de résoudre le problème (4.11) en substituant à la fonction étendue χ le modèle $\chi^k : \mathbb{R}^n \rightarrow \mathbb{R}$ défini par

$$\chi^k(x) \triangleq \phi^k(x) - \phi^k(x^k) + \Gamma^k(x), \quad \forall x \in \mathbb{R}^n. \quad (4.12)$$

L'algorithme est le suivant:

ALGORITHME SCHEMATIQUE STABILISE III

(Problème avec contraintes ensemblistes)

(0) (INITIALISATION) Choisir une solution initiale réalisable $x^1 \in D \triangleq (C \cap \text{dom} f)$, un coefficient de descente $\alpha \in]0, 1[$ et un seuil d'arrêt $\underline{\delta} \geq 0$. Poser $k = 1$.

(1) (CONSTRUCTION DU MODÈLE χ^k ET CHOIX D'UNE NORME $\|\cdot\|_k$)

Si $y^k \in D$ alors poser $\Gamma^k = \Gamma^{k-1}$ et obtenir le modèle ϕ^k en améliorant le modèle ϕ^{k-1} avec l'information additionnelle récoltée au point y^k ;

sinon (i.e. $y^k \notin D$), poser $\phi^k = \phi^{k-1}$ et obtenir le modèle Γ^k en améliorant le modèle Γ^{k-1} avec l'information additionnelle récoltée au point y^k .

Poser

$$\chi^k(x) = \phi^k(x) - \phi^k(x^k) + \Gamma^k(x), \quad \forall x \in \mathbb{R}^n.$$

(2) (OBTENTION D'UNE NOUVELLE SOLUTION CANDIDATE) Résoudre le problème associé au modèle stabilisé afin d'obtenir une solution y^{k+1} pas trop éloignée du centre de stabilité x^k .

(3) (EVALUATION DE LA QUALITÉ DU MODÈLE DE L'ENSEMBLE)

Si

$$y^{k+1} \in D,$$

aller à l'étape 4.

Sinon aller à l'étape 6.

(4) (TEST D'ARRÊT) Evaluer

$$\delta^k = \chi^k(x^k) - \chi^k(y^{k+1}) = \phi^k(x^k) - \phi^k(y^{k+1}) + \Gamma^k(x^k) - \Gamma^k(y^{k+1}) \geq 0.$$

Si $\delta^k \leq \underline{\delta}$, on arrête.

(5) (EVALUATION DE LA QUALITÉ DU MODÈLE DE LA FONCTION ÉCONOMIQUE)

Si

$$\chi(x^k) - \chi(y^{k+1}) = f(x^k) - f(y^{k+1}) \geq \alpha \delta^k,$$

poser $x^{k+1} = y^{k+1}$ et aller à l'étape 7.(6) (INADÉQUATION DE L'UN DES MODÈLES) Poser $x^{k+1} = x^k$.(7) (MISE-À-JOUR DU COMPTEUR D'ITÉRATIONS) Remplacer k par $k + 1$ et retourner à l'étape 1.

Remarque: Il est nécessaire d'utiliser une procédure de Phase I afin de déterminer une solution initiale $x^1 \in D$.

Remarque: Il est important, pour que la méthode fonctionne, qu'elle puisse déterminer un point réalisable en un nombre fini d'itérations à toutes les fois qu'elle

produit un point y^{k+1} non-réalisable. Si Γ^{k-1} est une approximation polyédrique externe, il peut être nécessaire de multiplier cette fonction par une constante supérieure à 1 afin d'obtenir Γ^k à l'étape (1) de l'algorithme.

Les prochains chapitres présenteront et analyseront la convergence d'un algorithme de ce genre.

4.5 La méthode BOXSTEP

Marsten, Hogan et Blankenship (1975) ont présenté une méthode, connue sous le nom de méthode BOXSTEP, présentant certaines similitudes avec les algorithmes schématiques stabilisés I et III. Le problème traité est

$$\underset{x \in C}{\text{minimiser}} f(x) \quad (4.13)$$

où $f : C \rightarrow \mathbb{R} \cup \{+\infty\}$ est convexe et C est un sous-ensemble convexe *compact* de $\mathbb{R}^n$.

Sous sa forme la plus simple, la méthode n'utilise pas de modèle de la fonction économique f et de l'ensemble C . Elle est basée sur l'hypothèse que, $\forall \bar{x} \in C$ et $\gamma > 0$, la restriction du problème (4.13) à l'ensemble $C \cap \{x : \|x - \bar{x}\|_\infty \leq \gamma\}$ est beaucoup plus facile à résoudre que le problème original.

Méthode BOXSTEP

(0) (INITIALISATION) Choisir une solution initiale $x^1 \in C$, une taille $\gamma > 0$ de la région de confiance, et un seuil d'arrêt $\varepsilon \geq 0$. Poser $k = 1$.

(1) (DÉTERMINATION DU PROCHAIN CENTRE DE STABILITÉ) Déterminer une solution ε -optimale x^{k+1} du problème ci-dessous:

$$\begin{aligned} & \underset{x}{\text{minimiser}} \ f(x) \\ & \text{sous les contraintes} \\ & \quad \|x - \bar{x}\|_{\infty} \leq \gamma, \\ & \quad x \in C. \end{aligned} \tag{4.14}$$

(2) (TEST D'ARRÊT) Si

$$f(x^{k+1}) \geq f(x^k) - \varepsilon,$$

on arrête.

Sinon, remplacer k par $k + 1$ et retourner à l'étape 1.

Le résultat suivant énonce la convergence de l'algorithme.

Théorème 4.5.1 *Si la fonction f est fermée dans C , alors l'algorithme s'arrête avec une solution σ -optimale après un nombre fini d'itérations, où*

$$\sigma = 2\varepsilon/\lambda, \quad \lambda = \min\{\gamma/\delta, 1\}$$

et

$$\delta = \max\{\|x - y\|_2 : x, y \in C\}.$$

Le fait d'utiliser directement f et C dans le sous-problème (4.14) rend inutile le recours à des modèles de f et de C , ainsi que les étapes d'évaluation des modèles que comportent les algorithmes schématiques stabilisés. La nouvelle solution produite par le problème (4.14) est la meilleure que l'on puisse obtenir, ce qui, avec les méthodes schématiques, peut nécessiter plusieurs itérations où l'on améliore les modèles. Ceci nous assure d'une descente si la solution n'est pas $2\varepsilon/\lambda$ -optimale et nous autorise à déplacer la région de confiance si le test d'arrêt de l'étape 2 n'est pas satisfait.

En pratique, Marsten, Hogan et Blankenship (1975) utilisent des approximations polyédriques de f et de C dans une méthode conventionnelle de plans sécants (de type KCG) utilisée à l'étape 1 de l'algorithme. Leurs résultats expérimentaux pour des problèmes de décomposition en programmation linéaire montrent que l'algorithme ainsi obtenu est très efficace. Bien qu'ils ne fassent pas varier la taille de la région de confiance (i.e. la valeur du paramètre γ) d'une itération à l'autre, leurs résultats montrent très clairement que des valeurs de γ trop faibles (amenant l'algorithme à simuler le comportement des méthodes de plus forte pente) ou trop grandes (cas des méthodes de plans coupants classiques) sont associées à une convergence très lente, ce qui milite en faveur de l'utilisation de procédures ajustant la valeur du paramètre γ^k d'une itération à l'autre. Ces résultats très encourageants nous incitent à croire que la mise-en-oeuvre des algorithmes schématiques stabilisés présentés dans ce chapitre peut résulter en des méthodes efficaces de résolution de problèmes de programmation mathématique de grandes dimensions, tels ceux décrits au chapitre 2.

4.6 Propriétés des fonctions $v_i(\cdot)$ du chapitre 2

L'application de méthodes de plans sécants, stabilisés ou non, au problème de programmation linéaire du chapitre 2 requiert l'exploitation des particularités des fonctions valeur qui y ont été définies. Nous étudions ici quelques propriétés des fonctions v_i définies à la section 2.1. Nous verrons qu'elles sont convexes et affines par morceaux dans leurs domaines respectifs et que l'on sait évaluer leurs dérivées directionnelles ainsi que certains de leurs sous-gradients.

Pour chaque indice $i \in \{1, \dots, r\}$ définissons comme suit un ensemble S_i :

$$S_i \triangleq \{s_i : \text{le problème (2.6) est réalisable}\} = \{s_i : v_i(s_i) < +\infty\}.$$

De plus, définissons l'ensemble S_0 par

$$\begin{aligned} S_0 &\triangleq \{s_0 = (s_1, \dots, s_r) : \text{le problème (2.5) est réalisable}\} \\ &= \{s_0 = (s_1, \dots, s_r) : v_0(s_0) < +\infty\}. \end{aligned}$$

Lemme 4.6.1 *Chacune des fonctions $v_i(\cdot)$ est convexe et affine par morceaux dans son domaine S_i , $i \in \{0, 1, \dots, r\}$.*

Démonstration: La convexité découle du lemme 2.3 de Shapiro (1979) tandis que la dernière partie du résultat découle de l'application répétée du théorème 19.4 de Rockafellar (1970). ■

Proposition 4.6.2 *La fonction économique $v(\cdot)$ du problème équivalent (2.4) est polyédrique convexe lorsque le problème (2.1) est réalisable.*

Démonstration: La proposition découle du corollaire A8 et de la proposition A9 de Wets (1966). ■

Proposition 4.6.3 *Il existe $s_i \in S_i$ tel que $v_i(s_i) = -\infty$ si et seulement si $v_i(s_i) \equiv -\infty, \forall s_i \in S_i$.*

Démonstration: Wets (1966), lemme A1. ■

Considérons maintenant les formulations duales (4.15) et (4.16), des sous-problèmes (2.5) et (2.6):

$$\begin{aligned} & \underset{\pi_0, \mu}{\text{maximiser}} \quad b_0^T \pi_0 - \sum_{i=1}^r s_i^T \mu_i \\ & \text{sous les contraintes} \\ & \quad \pi_0^T C_0 + \sum_{i=1}^r \mu_i^T C_i \leq c^T, \\ & \quad \pi_0 \geq 0 \end{aligned} \tag{4.15}$$

et

$$\begin{aligned} & \underset{\pi_i, \nu_i}{\text{maximiser}} \quad (b_i + s_i)^T \pi_i + d_i^T \nu_i \\ & \text{sous les contraintes} \\ & \quad \pi_i^T A_i + \nu_i^T B_i \leq c_i^T, \\ & \quad \pi_i \geq 0, \nu_i \geq 0. \end{aligned} \tag{4.16}$$

Les dérivées directionnelles des fonctions $v_i(\cdot)$ dépendent des solutions optimales des problèmes duaux (4.16), $i \in \{1, \dots, r\}$. Il est facile de démontrer le résultat suivant en utilisant le théorème A.1.

Lemme 4.6.4 *Supposons que $v_0(s)$ ou que $v_i(s_i), i \in \{1, \dots, r\}$, prenne une valeur finie.*

- Si $w = (w_1, \dots, w_r) \neq 0$ est une direction réalisable de v_0 en s , alors $\nabla v_0(s : w)$ existe et est donnée par

$$\nabla v_0(s : w) = \max\{-\mu^T w : (\pi_0, \mu) \text{ est une solution optimale du problème dual (4.15)}\}.$$

- Si $w_i \neq 0$ est une direction réalisable de v_i en $s_i, i \in \{1, \dots, r\}$, alors $\nabla v_i(s_i : w_i)$ existe et sa valeur est donnée par

$$\begin{aligned} \nabla v_i(s_i : w_i) &\triangleq \nabla v_i((s_i, d_i) : (w_i, 0)) \\ &= \max\{\pi_i^T w_i : (\pi_i, \nu_i) \text{ est une solution optimale du problème dual (4.16)}\}. \end{aligned}$$

En fait, les solutions duales optimales des problèmes (2.5) et (2.6) servent à définir des sous-gradients des fonctions $v_i(\cdot)$ en des points $s_i \in S_i, i \in \{0, 1, \dots, r\}$. On démontre facilement la proposition suivante en constatant que l'inégalité du sous-gradient est vérifiée.

Proposition 4.6.5 *Les deux énoncés suivants sont vrais:*

- Si (π_i^*, ν_i^*) est un vecteur de valeurs duales optimales associé au problème de programmation linéaire (2.6) définissant $v_i(s_i), s_i \in S_i, i \neq 0$, alors π_i^* est un sous-gradient de $v_i(\cdot)$ en s_i .
- Si (π_0^*, μ^*) , où $\mu^* = (\mu_1^*, \dots, \mu_r^*)$, est un vecteur de valeurs duales optimales associé au problème de programmation linéaire (2.5) définissant $v_0(s_0), s_0 \in S_0$, alors $-\mu^*$ est un sous-gradient de $v_0(\cdot)$ en s_0 .

Enfin, il découle du théorème B.8, que

$$\partial v(s) = \partial v_0(s) + \partial v_1(s) + \dots + \partial v_r(s)$$

si le problème (2.1) est réalisable.

En supposant que le problème (2.1) possède une solution optimale finie, on a $v(s) > -\infty, \forall s$, par la proposition 4.6.3, de sorte que les fonctions $v_0(s)$ et $v_i(s_i)$ sont elles aussi bornées inférieurement. Cependant, certaines valeurs de s peuvent rendre non-réalisable l'un des problèmes (2.5) ou (2.6), ce qui a pour effet de rendre l'une des fonctions $v_i, i \in \{0, 1, \dots, r\}$, et du coup v , égale à $+\infty$.

On ne peut donc choisir n'importe comment les variables $s = (s_1, \dots, s_r)$. Il faut restreindre l'optimisation au domaine de v , c'est-à-dire à $\text{dom } v \triangleq \{s \in \mathbb{R}^n : v(s) < +\infty\}$. Pour cela, il faut au moins que les problèmes (2.5) et (2.6), $i \in \{0, 1, \dots, r\}$, aient tous un ensemble de solutions non vide et que l'intersection de ces ensembles soit non vide. Les deux propositions suivantes donnent des conditions nécessaires et suffisantes d'existence d'une solution réalisable pour chacun de ces problèmes pour des valeurs de s fixées. Elles découlent directement de l'application du lemme de Farkas et Minkowski (présenté à l'annexe A).

Proposition 4.6.6 *Le problème de programmation linéaire (2.5) définissant $v_0(s)$ est réalisable si et seulement si*

$$\bar{\pi}_0^T b_0 - \sum_{i=1}^r \bar{\mu}_i^T s_i \leq 0$$

pour tout

$$(\bar{\pi}_0, \bar{\mu}_1, \dots, \bar{\mu}_r) \in \mathfrak{F}_0$$

où

$$\mathfrak{S}_0 \triangleq \{(\pi_0, \mu_1, \dots, \mu_r) : \pi_0^T C_0 + \sum_{i=1}^r \mu_i^T C_i \leq 0, \pi_0 \geq 0\}.$$

Proposition 4.6.7 *Soit $i \in \{1, \dots, r\}$. Le problème de programmation linéaire (2.6) définissant $v_i(s_i)$ est réalisable si et seulement si*

$$\bar{\pi}_i^T (b_i + s_i) + \bar{\nu}_i^T d_i \leq 0$$

pour tout

$$(\bar{\pi}_i, \bar{\nu}_i) \in \mathfrak{S}_i$$

où

$$\mathfrak{S}_i \triangleq \{(\pi_i, \nu_i) : \pi_i^T A_i + \nu_i^T B_i \leq 0, (\pi_i, \nu_i) \geq 0\}.$$

Les ensembles $\mathfrak{S}_i, i \in \{0, 1, \dots, r\}$, sont des cônes polyédriques. Ils ont chacun un nombre fini de générateurs:

$$(\pi_0^\ell, \mu_1^\ell, \dots, \mu_r^\ell), \quad \ell \in \{1, \dots, q_0\}, \quad \text{pour } \mathfrak{S}_0,$$

et

$$(\pi_i^\ell, \nu_i^\ell), \quad \ell \in \{1, \dots, q_i\}, \quad \text{pour } \mathfrak{S}_i, \quad i \in \{1, \dots, r\}.$$

Les conditions de la proposition 4.6.6 sont vérifiées si et seulement si

$$\langle \pi_0^\ell, b_0 \rangle - \sum_{i=1}^r \langle \mu_i^\ell, s_i \rangle \leq 0, \quad \forall \ell \in \{1, \dots, q_0\},$$

alors que celles de la proposition 4.6.7 le sont si et seulement si

$$\langle \pi_i^\ell, b_i + s_i \rangle + \langle \nu_i^\ell, d_i \rangle \leq 0, \quad \forall \ell \in \{1, \dots, q_i\}.$$

Or, les générateurs des cônes polyédriques $\mathfrak{S}_i, i \in \{0, 1, \dots, r\}$, sont précisément les rayons extrémaux des polytopes V_i définis par les contraintes des problèmes duaux (4.15) et (4.16).

Le problème qui se pose alors est celui de la détermination du rayon extrême approprié lorsqu'un problème (2.5) ou (2.6) est irréalisable.

Proposition 4.6.8 *Si, pour $s = \bar{s}$, le problème de programmation linéaire (2.5) définissant la fonction $v_0(\cdot)$ n'est pas réalisable, i.e. si $v_0(\bar{s}) = +\infty$, alors le vecteur $(\bar{\pi}_0, \bar{\mu}_1, \dots, \bar{\mu}_r)$ de valeurs duales optimales du problème de phase I est un rayon extrême de V_0 tel que*

$$\langle b_0, \bar{\pi}_0 \rangle - \sum_{i=1}^r \langle \bar{s}_i, \bar{\mu}_i \rangle > 0.$$

Proposition 4.6.9 *Si, pour $i \in \{1, \dots, r\}$ et $s_i = \bar{s}_i$, le problème de programmation linéaire (2.6) définissant la fonction $v_i(\cdot)$ n'est pas réalisable, i.e. si $v_i(\bar{s}_i) = +\infty$, alors le vecteur $(\bar{\pi}_i, \bar{\nu}_i)$ de valeurs duales optimales du problème de phase I est un rayon extrême de V_i tel que*

$$\langle b_i + \bar{s}_i, \bar{\pi}_i \rangle + \langle d_i, \bar{\nu}_i \rangle > 0.$$

Ce sont les inégalités de ces deux dernières propositions qui serviront à engendrer les coupes requises par notre algorithme lorsqu'un sous-problème n'est pas réalisable.

La méthode utilisée pour résoudre le problème équivalent modifie itérativement le vecteur s , de sorte que les sous-problèmes (2.5) et (2.6), $i \in \{1, \dots, r\}$, ne changent que dans leurs seconds membres. Ceci signifie que la solution duale optimale d'un sous-problème à l'itération k est réalisable à l'itération $k + 1$. Il semble alors souhaitable d'utiliser la méthode duale du simplexe pour réoptimiser les sous-problèmes, ce qui nous incite à rechercher une autre manière d'engendrer un rayon extrême lorsqu'un sous-problème est impossible.

Désignons par B la base courante d'un sous-problème au moment où l'algorithme dual du simplexe détecte l'incompatibilité des contraintes. Soit u^ℓ la $\ell^{\text{ème}}$ ligne de l'inverse B^{-1} de cette base, ℓ étant l'indice de la ligne sur laquelle l'algorithme dual essaie de pivoter. Il est facile de montrer que $-u^\ell$ est un rayon extrême du polyèdre V des contraintes du problème dual (voir Luenberger [1984], page 98) associé à ce sous-problème. On a alors le résultat ci-dessous.

Proposition 4.6.10 *Supposons que l'on résolve les sous-problèmes (2.5) et (2.6) à l'aide de la méthode duale du simplexe. Si, pour $s = \bar{s}$,*

- *le problème de programmation linéaire (2.5) ou (2.6) définissant la fonction $v_i(\cdot)$ n'est pas réalisable, $i \in \{1, \dots, r\}$, et*
- *u^ℓ est la $\ell^{\text{ème}}$ ligne de l'inverse de la base courante, ℓ étant l'indice de la ligne sur laquelle la méthode duale essaie de pivoter lorsqu'elle détecte l'incompatibilité des contraintes,*

alors

(a) $-u^\ell = (\pi_0^\ell, \mu_1^\ell, \dots, \mu_r^\ell)$ est un rayon extrême de V_0 , si $i = 0$, tel que

$$\langle b_0, \pi_0^\ell \rangle - \sum_{i=1}^r \langle \bar{s}_i, \mu_i^\ell \rangle > 0$$

ou

(b) $-u^\ell = (\pi_i^\ell, \nu_i^\ell)$ est un rayon extrême de V_i , si $i > 0$, tel que

$$\langle b_i + \bar{s}_i, \pi_i^\ell \rangle + \langle d_i, \nu_i^\ell \rangle > 0.$$

CHAPITRE 5

Construction et mise-à-jour des problèmes de recherche d'une direction d'amélioration pour le problème de la minimisation d'une somme de fonctions convexes

5.1 Introduction

Nous avons vu au chapitre 2 comment un problème de programmation linéaire possédant la structure du problème (2.1) peut être reformulé comme un problème équivalent de programmation convexe non-différentiable. Ce problème équivalent, le problème (2.4), consiste à minimiser une somme de fonctions valeurs dans le domaine de la fonction résultante qui, par les résultats du chapitre 4, est polyédrique convexe.

Nous nous attaquerons dans ce chapitre, et au suivant, à la résolution de ce problème en considérant le problème de programmation convexe un peu plus général suivant:

$$\underset{x \in C \cap \text{dom } f}{\text{minimiser}} \quad f(x) \triangleq \sum_{i=1}^r f_i(x) \quad (5.1)$$

où chaque fonction $f_i : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, $i \in \{1, \dots, r\}$, est convexe propre et fermée (c'est-à-dire semi-continue inférieure), et C est un ensemble convexe fermé quelconque. Nous supposons que $D \triangleq C \cap \text{dom } f$ possède un intérieur non-vide.

L'hypothèse selon laquelle les fonctions f_i sont fermées nous assure que les ensembles $\text{dom } f_i$ sont (convexes) fermés. L'ensemble D l'est donc lui aussi.

Dans ce chapitre, nous cherchons à opérationnaliser les concepts du chapitre 4, et plus particulièrement ceux de la section 4.4, dans le but de préciser l'étape 2 de l'algorithme schématique III, c'est-à-dire de déterminer un nouveau candidat au titre de centre de stabilité.

Il est important de remarquer que le problème (5.1) est plus général que celui traité par la plupart des méthodes d'optimisation convexe non-différentiable. En effet, la vaste majorité de ces méthodes suppose que $D = \mathbb{R}^n$ ou que $\text{dom } f = \mathbb{R}^n$ tandis que C est défini **explicitement** avec des fonctions convexes à valeurs finies comme dans les problèmes (4.1) et (4.5). Lorsque C n'est pas défini explicitement, on suppose habituellement qu'il ne crée pas de complications additionnelles et on l'inclut dans la définition du problème de recherche d'une direction d'amélioration, le problème maître. C'est ce que font, par exemple, Hiriart-Urruty et Lemaréchal (1993b) avec leur généralisation de l'algorithme de Kiwiel (1992).

Comme au chapitre 4, le problème de la recherche d'une direction d'amélioration est construit à l'aide de modèles des fonctions et ensembles du problème. Les modèles que nous utiliserons sont des approximations polyédriques. Deux approches seront utilisées pour définir ce problème. Sous l'approche des **régions de confiance**, un sous-ensemble de $\mathbb{R}^n$ est délimité explicitement autour du centre de stabilité courant afin de restreindre l'optimisation à une région de $\mathbb{R}^n$ où les modèles sont susceptibles d'être bons. Ceci est fait à l'aide d'une contrainte additionnelle construite avec la fonction norme. Sous l'approche des **pénalisations**, on ajoute un terme à la fonction économique du problème maître qui agira comme un ressort afin de limiter le déplacement. Ce terme est lui aussi construit avec la fonction norme.

Ces deux approches ne sont pas nouvelles (on en trouvera une bonne discussion dans l'ouvrage d'Hiriart-Urruty et Lemaréchal (1993b)); ce qui l'est, par contre, c'est la généralité des normes considérées. En effet, contrairement aux méthodes de faisceaux qui utilisent majoritairement les normes euclidienne et quadratique, nous considérons des normes quelconques. Nous verrons que certaines normes sont mieux adaptées que d'autres à une approche particulière.

On se démarque également des approches habituelles en programmation mathématique en ce sens que l'on préfère transformer le problème (5.1) en un problème d'optimisation sans contrainte où les ensembles sont remplacés par des fonctions indicatrices.

Le chapitre s'inspire largement, en les généralisant, de certaines parties de l'ouvrage de Kiwiel (1985). Il se termine en montrant comment des problèmes maîtres équivalents peuvent être obtenus et mis-à-jour sous les stratégies d'accumulation/épuration et d'agrégation de l'information de Kiwiel (1985), pour le cas de l'approche des pénalisations.

5.2 Linéarisation des ensembles et fonctions convexes définissant le problème

L'algorithme que nous voulons construire doit, à chaque itération, déterminer un nouveau candidat au titre de centre de stabilité. Ce candidat est obtenu en résolvant un problème maître construit avec l'information accumulée au cours du déroulement de l'algorithme.

L'information utilisée lors de la $k^{\text{ème}}$ itération est segmentée et indicée à l'aide des ensembles $J_{f_i}^k, J_{c_i}^k, i \in \{1, \dots, r\}$ et J_c^k . Désignons par x^k le centre de stabilité et par y^{k+1} la solution du problème maître de l'itération k .

Si $y^{k+1} \in \text{dom } f_i$, l'algorithme évalue $f_i^{k+1} \triangleq f_i(y^{k+1})$ et détermine un sous-gradient $g_i^{k+1} \in \partial f_i(y^{k+1})$. L'ensemble $J_{f_i}^k \subseteq \{j \leq k : y^j \in \text{dom } f_i\}$ est un sous-ensemble des indices des précédents candidats qui faisaient partie du domaine de la fonction f_i .

Si $y^{k+1} \notin \text{dom } f_i$, alors l'algorithme doit engendrer $(a_i^{k+1}, b_i^{k+1}) \in \mathbb{R}^n \times \mathbb{R}$ tel que

$$C_i \triangleq \text{dom } f_i \subseteq \{x \in \mathbb{R}^n : \langle a_i^{k+1}, x \rangle \leq b_i^{k+1}\} \not\ni y^{k+1}.$$

L'ensemble $J_{c_i}^k \subseteq \{j \leq k : y^j \notin \text{dom } f_i\}$ est un sous-ensemble des indices des candidats précédents pour lesquels un hyperplan séparateur a dû être engendré.

Enfin, si $y^{k+1} \notin C$, l'algorithme engendre aussi un vecteur $(a^{k+1}, b^{k+1}) \in \mathbb{R}^n \times \mathbb{R}$ tel que

$$C \subseteq \{x \in \mathbb{R}^n : \langle a^{k+1}, x \rangle \leq b^{k+1}\} \not\ni y^{k+1}.$$

L'ensemble $J_c^k \subseteq \{j \leq k : y^j \notin C\}$ est le sous-ensemble des indices des candidats ne faisant pas partie de C utilisé par le problème maître de l'itération k .

Il n'est pas nécessaire, pour le moment, de savoir comment les coupes sont obtenues. Des précisions seront apportées au prochain chapitre.

Remarquons que

$$J_{f_i}^k \cap J_{c_i}^k = \emptyset, \quad \forall i \in \{1, \dots, r\}$$

et que

$$J_{f_i}^k \cup J_{\alpha_i}^k \subseteq \{1, \dots, k\}, \quad \forall i \in \{1, \dots, r\}.$$

La dernière relation indique qu'il est possible que l'on ait détruit une partie de l'information déjà trouvée.

De plus, nous exigeons que

$$J_{f_i}^k \neq \emptyset, \quad \forall i \in \{1, \dots, r\}, \quad \forall k$$

car nous désirons établir une méthode primale.

Les quantités

$$(f_i^j, g_i^j), \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\}$$

$$(a_i^j, b_i^j), \quad j \in J_{\alpha_i}^k, \quad i \in \{1, \dots, r\}$$

et

$$(a^j, b^j), \quad j \in J_c^k$$

servent à définir des approximations polyédriques des fonctions et ensembles définissant le problème.

Pour chaque fonction $f_i : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ définissons les linéarisations $\underline{f}_i(\cdot|y^j) : \mathbb{R}^n \rightarrow \mathbb{R}$, $j \in J_{f_i}^k$, par

$$\underline{f}_i(x|y^j) \triangleq f_i(y^j) + \langle g_i^j, x - y^j \rangle, \quad \forall x \in \mathbb{R}^n \quad (5.2)$$

ainsi que son approximation polyédrique $\hat{f}_i^k$ par

$$\hat{f}_i^k(x) \triangleq \max\{\underline{f}_i(x|y^j) : j \in J_{f_i}^k\}, \quad \forall x \in \mathbb{R}^n. \quad (5.3)$$

L'approximation polyédrique $\hat{f}^k$ de la fonction économique f est donné par

$$\hat{f}^k(x) \triangleq \sum_{i=1}^r \hat{f}_i^k(x), \quad \forall x \in \mathbb{R}^n. \quad (5.4)$$

L'approximation polyédrique C_i^k de $C_i = \text{dom } f_i$, $i \in \{1, \dots, r\}$, est définie par

$$C_i^k \triangleq \{x \in \mathbb{R}^n : \langle a_i^j, x \rangle \leq b_i^j, \quad \forall j \in J_\alpha^k\} \quad (5.5a)$$

et l'approximation polyédrique $\hat{I}_\alpha^k$ de la fonction indicatrice I_α est

$$\hat{I}_\alpha^k(x) \triangleq \max\{I_\alpha^j(x) \triangleq \langle a_i^j, x \rangle - b_i^j : j \in J_\alpha^k\}, \quad \forall x \in \mathbb{R}^n. \quad (5.6a)$$

De même, on définit les approximations polyédriques C^k et $\hat{I}_c^k$ de C et de I_c par

$$C^k \triangleq \{x \in \mathbb{R}^n : \langle a^j, x \rangle \leq b^j, \quad \forall j \in J_c^k\} \quad (5.5b)$$

et

$$\hat{I}_c^k(x) \triangleq \max\{I_c^j(x) \triangleq \langle a^j, x \rangle - b^j : j \in J_c^k\}, \quad \forall x \in \mathbb{R}^n. \quad (5.6b)$$

L'ensemble $D = C \cap \text{dom } f$ des solutions réalisables est approché par

$$D^k \triangleq C^k \cap C_1^k \cap \dots \cap C_r^k \quad (5.7)$$

tandis que l'approximation polyédrique $\hat{I}_D^k$ de la fonction indicatrice I_D est

$$\hat{I}_D^k(x) \triangleq \max\{\hat{I}_c^k(x), \hat{I}_{c1}^k(x), \dots, \hat{I}_{cr}^k(x)\}, \quad \forall x \in \mathbb{R}^n. \quad (5.8)$$

Les linéarisations externes C^k et C_i^k , $i \in \{1, \dots, r\}$, sont justifiées par le théorème B.1 tandis que la linéarisation externe de la fonction f_i repose sur l'existence de sous-gradient(s) en tout point de $\text{dom } f_i$, $i \in \{1, \dots, r\}$.

Le théorème B.5 nous assure que les fonctions f_i possèdent au moins un sous-gradient en tout point de l'intérieur relatif de leurs domaines, c'est-à-dire $ri(\text{dom } f_i)$, et que leurs sous-différentiels sont vides à l'extérieur de leurs domaines.

Nous ferons l'hypothèse que chacune des fonctions f_i est sous-différentiable en tout point de son domaine, ce qui nous permet d'exploiter la convexité de f_i et d'obtenir la représentation équivalente suivante:

$$f_i(x) = \max_{y \in \text{dom } f_i} \{f_i(y) + \langle g_i(y), x - y \rangle : g_i(y) \in \partial f_i(y)\}, \quad \forall x \in \text{dom } f_i \quad (5.9)$$

qui constitue la linéarisation externe de f_i .

5.3 Détermination d'une direction d'amélioration: approche des régions de confiance

Puisque nous ne connaissons pas *a priori* les sous-différentiels complets des fonctions f_i en chaque point $y \in \text{dom } f_i$, $i \in \{1, \dots, r\}$, ni les ensembles de tous les demi-espaces fermés contenant C et $\text{dom } f_i$, $i \in \{1, \dots, r\}$, l'algorithme que nous construirons ne travaillera qu'avec des versions approchées du problème à l'aide d'une stratégie de relaxation (Geoffrion, 1970a). Ceci nous suggère de résoudre le problème suivant à l'itération k :

$$\underset{x \in D^k}{\text{minimiser}} \quad \hat{f}^k(x) = \sum_{i=1}^r \hat{f}_i^k(x) \quad (5.10a)$$

c'est-à-dire

$$\begin{aligned} &\underset{x, v}{\text{minimiser}} \quad \sum_{i=1}^r v_i \\ &\text{sous les contraintes} \\ &\quad f_i^j + \langle g_i^j, x - y^j \rangle \leq v_i, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \\ &\quad \langle a_i^j, x \rangle \leq b_i^j, \quad j \in J_{a_i}^k, \quad i \in \{1, \dots, r\} \\ &\quad \langle a^j, x \rangle \leq b^j, \quad j \in J_c^k. \end{aligned} \quad (5.10b)$$

Cette approche peut sembler intéressante, mais elle souffre d'un grave problème. Si la solution optimale y^{k+1} (en supposant qu'il en existe une) du problème (5.10a) n'est pas dans D , il se peut qu'il faille effectuer une infinité d'itérations rien que pour atteindre D , c'est-à-dire engendrer une suite infinie $\{y^\ell\}_{\ell > k}$ telle que

$$y^\ell \notin D, \quad \forall \ell > k$$

et

$$\lim_{\ell \rightarrow \infty} y^\ell = \bar{y} \in D \quad (\text{en fait fr } D, \text{ la frontière de } D).$$

Cette situation est désagréable car l'algorithme ne fait alors qu'enrichir les ensembles J_c^k et/ou $J_{\alpha_i}^k$, $i \in \{1, \dots, r\}$, et n'améliore pas le modèle de la fonction économique f . Une telle méthode n'est donc pas généralement convergente. Il serait intéressant de pouvoir engendrer une solution réalisable en un nombre fini d'itérations lorsque la méthode nous fait quitter D . Pour cela, il semble raisonnable de résoudre le problème suivant à l'itération k , si l'ensemble convexe D possède un point intérieur:

$$\begin{aligned} & \underset{x, \sigma_f, \sigma_D}{\text{minimiser}} \quad \sigma_f + \sigma_D \\ & \text{sous les contraintes} \\ & \quad \hat{f}^k(x) - f(x^k) \leq \sigma_f, \\ & \quad \langle a_i^j, x \rangle - b_i^j \leq \sigma_D, \quad j \in J_{\alpha_i}^k, \quad i \in \{1, \dots, r\} \\ & \quad \langle a^j, x \rangle - b^j \leq \sigma_D, \quad j \in J_c^k \\ & \quad \sigma_D \geq 0. \end{aligned} \tag{5.11}$$

Cette idée consiste à remplacer la résolution du problème

$$\underset{x \in \mathbb{R}^n}{\text{minimiser}} \quad \varphi^k(x) \triangleq f(x) - f(x^k) + I_D(x) \tag{5.12}$$

présenté au chapitre 4 par celle du problème suivant:

$$\text{minimiser } \hat{\varphi}^k(x) \triangleq \hat{f}^k(x) - f(x^k) + \hat{I}_D^k(x) \quad (5.13)$$

où $\hat{I}_D^k$ est une approximation polyédrique de la fonction indicatrice I_D telle que $\hat{I}_D^k(x) \geq 0, \forall x \in \mathbb{R}^n$ (ceci découle de la non-négativité de la variable σ_D).

Remarquons qu'en tout point réalisable $x \in D$, on a $I_D(x) = 0$. De plus, si $\varphi^k(x) < 0$, alors $x \in D$ et $f(x) < f(x^k)$, ce qui signifie que x est un point réalisable de coût inférieur à x^k .

On remarque aussi que le problème (5.13) est équivalent au problème suivant:

$$\text{minimiser } \hat{\varphi}^k(x^k + d) \quad (5.14)$$

lorsque l'on effectue le changement de variable $x = x^k + d$.

Les problèmes (5.13) et (5.14), tout comme les autres formulations, reposent sur des approximations polyédriques enrichies d'itération en itération selon une méthode de plans coupants. Or, comme nous l'avons vu au chapitre précédent, les méthodes de plans sécants sont réputées pour leur instabilité. De plus, bien que toujours réalisables, il n'est pas certain que les problèmes (5.14) engendrés à chaque itération aient toujours une solution optimale finie. Pour ces raisons, nous stabiliserons le comportement de l'algorithme en limitant le domaine d'optimisation à une région (compacte) de confiance qui ne sera déplacée que lorsque l'on aura déterminé un nouveau centre de stabilité exhibant un coût "raisonnablement" inférieur à celui du centre de stabilité courant x^k .

Remarque: Il est probablement préférable, pour le cas de fonctions et ensembles convexes non-polyédriques, de multiplier la fonction $\hat{f}_D^k(x)$ du problème (5.13) par une constante $c^k > 0$ suffisamment grande si l'on doit retrouver un point réalisable en un nombre fini d'étapes à chaque fois qu'un algorithme utilisant ce problème détermine une solution non-réalisable pour le problème (5.1). ■

Pour celà, nous remplaçons le problème (5.14) par le suivant:

$$\begin{aligned} & \underset{d \in \mathbb{R}^n}{\text{minimiser}} \quad \hat{\varphi}^k(x^k + d) \\ & \text{sous la contrainte} \\ & \|d\| \leq \gamma_k \end{aligned} \tag{5.15}$$

c'est-à-dire

$$\underset{d, v, \sigma_f, \sigma_D}{\text{minimiser}} \quad \sigma_f + \sigma_D \tag{5.16a}$$

sous les contraintes

$$\sum_{i=1}^r v_i - \sigma_f \leq f(x^k) \tag{5.16b}$$

$$\langle g_i^j, d \rangle - v_i \leq -f_{ij}^k, \quad j \in J_{fi}^k, \quad i \in \{1, \dots, r\} \tag{5.16c}$$

$$\langle a_i^j, d \rangle - \sigma_D \leq b_{ij}^k, \quad j \in J_{ci}^k, \quad i \in \{1, \dots, r\} \tag{5.16d}$$

$$\langle a^j, d \rangle - \sigma_D \leq b_j^k, \quad j \in J_c^k \tag{5.16e}$$

$$\|d\| \leq \gamma_k, \tag{5.16f}$$

$$\sigma_D \geq 0 \tag{5.16g}$$

où

$\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}_+$ est une norme quelconque,

$\gamma_k > 0$ est la taille de la région de confiance

$$f_{ij}^k \triangleq f_i^j + \langle g_i^j, x^k - y^j \rangle, \quad j \in J_{fi}^k, \quad i \in \{1, \dots, r\} \tag{5.17a}$$

$$b_{ij}^k \triangleq b_i^j - \langle a_i^j, x^k \rangle, \quad j \in J_{ci}^k, \quad i \in \{1, \dots, r\} \quad (5.17b)$$

et

$$b_j^k \triangleq b^j - \langle a^j, x^k \rangle, \quad j \in J_c^k. \quad (5.17c)$$

Remarquons que, par la convexité des fonctions f_i , on a

$$f_{ij}^k \leq f_i(x^k), \quad \forall j \in J_{fi}^k, \quad i \in \{1, \dots, r\} \quad (5.18a)$$

tandis que

$$b_{ij}^k \geq 0, \quad \forall j \in J_{ci}^k, \quad i \in \{1, \dots, r\} \quad (5.18b)$$

et

$$b_j^k \geq 0, \quad \forall j \in J_c^k \quad (5.18c)$$

puisque le centre de stabilité x^k est réalisable.

Remarquons aussi que l'on aurait pu remplacer la contrainte $\|d\| \leq \gamma^k$ des problèmes (5.15) et (5.16) par $h(\|d\|) \leq \gamma_k$ où $h : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ est une fonction bijective croissante telle que $h(0) = 0$. Ceci équivaut à poser la contrainte $\|d\| \leq h^{-1}(\gamma_k)$ qui n'est pas plus générale que celle des problèmes (5.15) et (5.16).

On constate, enfin, que la valeur optimale

$$\sigma^k \triangleq \sigma_f^k + \sigma_D^k \quad (5.19)$$

de la fonction économique du problème (5.16) est toujours inférieure ou égale à zéro (il suffit d'observer que $d = 0 \in \mathbb{R}^n$, $\sigma_f = \sigma_D = 0$, et $v_i = \max\{f_{ij}^k : j \in J_{fi}^k\}$, $\forall i$, en est une solution réalisable).

Le problème (5.16) a le défaut de posséder un nombre pouvant être élevé de contraintes linéaires. Il est donc intéressant d'en connaître le dual.

5.4 Détermination du problème dual: approche des régions de confiance

La formulation duale du problème (5.16) s'obtient en procédant de la manière habituelle. En associant les multiplicateurs de Lagrange

$$\begin{aligned} \eta &\geq 0 && \text{à la contrainte (5.16b)} \\ \pi_{ij} &\geq 0 && \text{aux contraintes (5.16c)} \\ \mu_{ij} &\geq 0 && \text{aux contraintes (5.16d)} \\ \omega_j &\geq 0 && \text{aux contraintes (5.16e)} \\ \rho &\geq 0 && \text{à la contrainte (5.16f)} \\ \text{et } \nu &\geq 0 && \text{à la contrainte (5.16g),} \end{aligned}$$

on construit le lagrangien du problème (5.16). La minimisation de celui-ci sur les variables primales nous permet d'obtenir la fonction duale. En appliquant les résultats de l'annexe B sur les normes duales et en maximisant la fonction duale, on obtient le problème dual ci-dessous, où $\|\cdot\|$ désigne la fonction norme duale associée à la norme $\|\cdot\|$.

$$\begin{aligned} \text{maximiser}_{\pi, \mu, \omega} \quad & \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} [f_{ij}^k - f_i(x^k)] - \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} b_{ij}^k - \sum_{j \in J_c^k} \omega_j b_j^k \\ & - \gamma_k \left\| \sum_{i=1}^r \left(\sum_{j \in J_{f_i}^k} \pi_{ij} g_i^j + \sum_{j \in J_{c_i}^k} \mu_{ij} a_i^j \right) + \sum_{j \in J_c^k} \omega_j a^j \right\|^* \end{aligned} \quad (5.20a)$$

sous les contraintes

$$\sum_{j \in J_{f_i}^k} \pi_{ij} = 1, \quad i \in \{1, \dots, r\} \quad (5.20b)$$

$$\sum_{i=1}^r \sum_{j \in J_{\alpha_i}^k} \mu_{ij} + \sum_{j \in J_c^k} \omega_j \leq 1 \quad (5.20c)$$

$$\pi_{ij} \geq 0, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \quad (5.20d)$$

$$\mu_{ij} \geq 0, \quad j \in J_{\alpha_i}^k, \quad i \in \{1, \dots, r\} \quad (5.20e)$$

$$\omega_j \geq 0, \quad j \in J_c^k. \quad (5.20f)$$

Les calculs intermédiaires permettant d'aboutir à ce résultat ont été omis car ils seront faits un peu plus loin dans le cas d'un problème un peu plus difficile.

Proposition 5.4.1 *Le problèmes (5.16) et (5.20) sont duaux.*

Les quantités $f_i(x^k) - f_{ij}^k \geq 0$, $j \in J_{f_i}^k$, sont les erreurs de linéarisation en x^k des $|J_{f_i}^k|$ hyperplans d'appui à l'épigraphe de la fonction f_i , $i \in \{1, \dots, r\}$. Les quantités b_{ij}^k , $j \in J_{\alpha_i}^k$, sont les erreurs de linéarisation en x^k des $|J_{\alpha_i}^k|$ hyperplans situés sous l'épigraphe de la fonction indicatrice I_{α_i} , $i \in \{1, \dots, r\}$, tandis que les constantes $b_j^k \geq 0$, $j \in J_c^k$, sont les erreurs de linéarisation en x^k des $|J_c^k|$ hyperplans situés sous l'épigraphe de I_c . Enfin, le dernier terme de la fonction économique (5.20a) est la norme duale de la combinaison convexe des ε -sous-gradients en x^k des fonctions f_i et I_{α_i} , $i \in \{1, \dots, r\}$, et I_c engendrés jusqu'ici (nous verrons au chapitre 6 que les g_i^j , a_i^j et a^j sont des ε -sous-gradients de ces fonctions).

Par conséquent, le problème dual (5.20) minimise la somme des erreurs de linéarisation des fonctions et de la norme d'un "sous-gradient agrégé". Ces quantités étant toutes positives, le mieux que l'on puisse espérer est d'annuler la valeur de la fonction économique. Ceci se produit lorsque $0 \in \mathbb{R}^n$ fait partie de l'enveloppe convexe des ε -sous-gradients (il s'agit plutôt d'une relaxation du concept d'enveloppe

convexe, à cause de la présence de l'inégalité à la contrainte (5.20c)) et que les poids de l'une des combinaisons "convexes" permettant d'obtenir le vecteur nul permettent aussi d'annuler la somme des erreurs de linéarisation.

5.5 Détermination d'une direction d'amélioration: approche des pénalisations

Si la norme utilisée dans le problème (5.16) est la norme ℓ_1 (ou ℓ_∞) alors la norme ℓ_∞ (ou ℓ_1) apparaît dans la fonction économique (5.20a) du problème dual, en vertu de la proposition B.22. Ces deux problèmes sont donc des problèmes de programmation linéaire que l'on sait résoudre.

Par contre, si le problème primal (5.16) est formulé avec une norme quadratique, il en est aussi de même (par le corollaire B.25) du problème dual. En pratique, ce dernier n'est pas simple à résoudre. Si l'on pouvait faire apparaître la norme au carré dans le problème dual, celui-ci deviendrait un problème de programmation quadratique (avec une structure particulière de contraintes) pour lequel il existe des algorithmes très efficaces (Kiwiel, 1989b et 1994; Ruszczynski, 1986; Stoer, 1971; Frangioni, 1996).

Il est donc intéressant de considérer des formulations alternatives du problème primal. L'une d'elles consiste à ajouter un terme de pénalisation à la fonction économique. On pourrait, par exemple, résoudre le problème suivant:

$$\underset{d \in \mathbb{R}^n}{\text{minimiser}} \quad \hat{\varphi}^k(x^k + d) + \Psi_k h(\|d\|) \quad (5.21)$$

c'est-à-dire

$$\underset{d, v, \sigma_f, \sigma_D}{\text{minimiser}} \sigma_f + \sigma_D + \Psi_k h(\|d\|) \quad (5.22a)$$

sous les contraintes

$$\sum_{i=1}^r v_i - \sigma_f \leq f(x^k) \quad (5.22b)$$

$$\langle g_i^j, d \rangle - v_i \leq -f_{ij}^k, \quad j \in J_{fi}^k, \quad i \in \{1, \dots, r\} \quad (5.22c)$$

$$\langle a_i^j, d \rangle - \sigma_D \leq b_{ij}^k, \quad j \in J_{ci}^k, \quad i \in \{1, \dots, r\} \quad (5.22d)$$

$$\langle a^j, d \rangle - \sigma_D \leq b_j^k, \quad j \in J_c^k \quad (5.22e)$$

$$-\sigma_D \leq 0 \quad (5.22f)$$

où $\Psi_k > 0$ et $h : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ est une fonction convexe croissante dérivable telle que $h(0) = h'(0) = 0$ et dont la dérivée est bijective (exemple: $h(x) = x^q, \forall q > 1$). Remarquons que la convexité de la fonction économique est préservée puisque la fonction norme est croissante.

On peut, à la manière d'Hiriart-Urruty et Lemaréchal (1993b), établir l'équivalence des formulations (5.15) et (5.21) et obtenir un résultat similaire à leur proposition XV.2.2.3. Cette équivalence est basée sur le fait que ces deux problèmes produisent les mêmes solutions optimales lorsque les constantes γ_k et Ψ_k sont bien choisies.

Proposition 5.5.2 *Substituons la contrainte $h(\|d\|) \leq \gamma_k$ à la contrainte $\|d\| \leq \gamma_k$ du problème (5.15). Alors*

- (a) *Pour tout $\gamma_k > 0$, il existe $\Psi_k \geq 0$ tel que toute solution optimale du problème (5.15) soit aussi une solution optimale du problème (5.21).*

(b) Pour tout $\Psi_k \geq 0$ tel que le problème (5.21) possède une solution optimale finie, il existe $\gamma_k \geq 0$ tel que cette solution soit aussi une solution optimale du problème (5.15).

Remarquons que, bien que le problème (5.15) possède toujours une solution optimale finie, il n'est pas nécessairement ainsi du problème (5.21). Pour ce dernier, l'existence d'une solution optimale finie dépend du choix de la fonction h et de la constante Ψ_k . Ainsi, si $h(x) = x, \forall x \in \mathbb{R}_+$, il se peut qu'il existe $(\tilde{d}, \tilde{v}, \tilde{\sigma}_f, \tilde{\sigma}_D)$ réalisable pour le problème (5.22a)–(5.22f) telle que

$$\tilde{\sigma}_f + \tilde{\sigma}_D + \Psi_k \|\tilde{d}\| < 0$$

et telle que chacun des membres de gauche des contraintes de ce problème soit négatif ou nul.

Dans ce cas, $t(\tilde{d}, \tilde{v}, \tilde{\sigma}_f, \tilde{\sigma}_D)$ est une solution réalisable $\forall t > 0$ et

$$\lim_{t \rightarrow \infty} \left(t(\tilde{\sigma}_f + \tilde{\sigma}_D) + \Psi_k \|t\tilde{d}\| \right) = \left(\tilde{\sigma}_f + \tilde{\sigma}_D + \Psi_k \|\tilde{d}\| \right) \lim_{t \rightarrow \infty} t = -\infty.$$

Par contre, avec $h(x) = x^q$, le problème (5.21) a une solution optimale finie pour toute valeur de $\Psi_k > 0$ lorsque $q > 1$.

5.6 Détermination du problème dual: approche des pénalisations

Le problème (5.22a)–(5.22f) peut posséder un grand nombre de contraintes. Il est intéressant d'en connaître le dual. En associant les multiplicateurs de Lagrange

$$\begin{aligned} \eta &\geq 0 && \text{à la contrainte (5.22b)} \\ \pi_{ij} &\geq 0 && \text{aux contraintes (5.22c)} \\ \mu_{ij} &\geq 0 && \text{aux contraintes (5.22d)} \\ \omega_j &\geq 0 && \text{aux contraintes (5.22e)} \\ \text{et} \quad \rho &\geq 0 && \text{à la contrainte (5.22f),} \end{aligned}$$

nous obtenons le *lagrangien*

$$\begin{aligned} L(d, v, \sigma_f, \sigma_D, \eta, \pi, \mu, \omega, \rho) &= (1 - \eta)\sigma_f + \left(1 - \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} - \sum_{j \in J_c^k} \omega_j - \rho\right)\sigma_D \\ &+ \sum_{i=1}^r \left(\eta - \sum_{j \in J_{f_i}^k} \pi_{ij}\right)v_i \\ &+ \left\langle \sum_{i=1}^r \left(\sum_{j \in J_{f_i}^k} \pi_{ij} g_i^j + \sum_{j \in J_{c_i}^k} \mu_{ij} a_i^j \right) + \sum_{j \in J_c^k} \omega_j a^j, d \right\rangle + \Psi_k h(\|d\|) \\ &- \eta f(x^k) + \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} f_{ij}^k - \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} b_{ij}^k - \sum_{j \in J_c^k} \omega_j b_j^k \end{aligned}$$

avec lequel on définit la *fonction duale*

$$D(\eta, \pi, \mu, \omega, \rho) = \min_{d, v, \sigma_f, \sigma_D} L(d, v, \sigma_f, \sigma_D, \eta, \pi, \mu, \omega, \rho)$$

et le problème dual

$$\text{maximiser } D(\eta, \pi, \mu, \omega, \rho)$$

sous les contraintes

$$\eta \geq 0, \pi \geq 0, \mu \geq 0, \omega \geq 0, \rho \geq 0.$$

Les variables σ_f, σ_D et $v_i, i \in \{1, \dots, r\}$ étant libres, on aura $D(\eta, \pi, \mu, \omega, \rho) = -\infty$ lorsque l'une des trois conditions suivantes n'est pas vérifiée:

$$\eta = 1 \quad (5.23a)$$

$$\sum_{i=1}^r \sum_{j \in J_{\epsilon_i}^k} \mu_{ij} + \sum_{j \in J_{\epsilon}^k} \omega_j \leq 1 \quad (5.23b)$$

$$\eta - \sum_{j \in J_{f_i}^k} \pi_{ij} = 0, \quad i \in \{1, \dots, r\}. \quad (5.23c)$$

Sous ces conditions, on a

$$\sum_{j \in J_{f_i}^k} \pi_{ij} = 1, \quad i \in \{1, \dots, r\} \quad (5.24)$$

et la fonction duale devient

$$\begin{aligned} D(\eta, \pi, \mu, \omega, \rho) = & \sum_{i=1}^r \left[\sum_{j \in J_{f_i}^k} \pi_{ij} (f_{ij}^k - f_i(x^k)) - \sum_{j \in J_{\epsilon_i}^k} \mu_{ij} b_{ij}^k \right] \\ & - \sum_{j \in J_{\epsilon}^k} \omega_j b_j^k + \min_{d \in \mathbb{R}^n} \left\{ \langle p, d \rangle + \Psi_k h(\|d\|) \right\} \end{aligned} \quad (5.25)$$

où

$$p = \sum_{i=1}^r \left(\sum_{j \in J_{f_i}^k} \pi_{ij} g_i^j + \sum_{j \in J_{\epsilon_i}^k} \mu_{ij} a^j \right) + \sum_{j \in J_{\epsilon}^k} \omega_j a^j. \quad (5.26)$$

Considérons le problème d'optimisation

$$\min_{d \in \mathbb{R}^n} \left\{ \langle p, d \rangle + \Psi_k h(\|d\|) \right\}. \quad (5.27)$$

Si $p = 0$, alors $d^* = 0$ est l'unique solution optimale du problème (5.27). Supposons que $p \neq 0$. Dans ce cas, d^* en est une solution optimale (finie) si et seulement si

$$\begin{aligned} 0 & \in p + \Psi_k \partial h(\|\cdot\|)(d^*) && \text{par le théorème B.19} \\ & = p + \Psi_k h'(\|d^*\|) \partial \|\cdot\|(d^*) && \text{par le théorème B.9.} \end{aligned}$$

Ceci est vrai si et seulement si il existe $y \in \partial \|\cdot\|(d^*)$ tel que

$$p + \Psi_k h'(\|d^*\|) y = 0.$$

Mais $p \neq 0$ implique que $h'(\|d^*\|) = 0$ (d'où $d^* \neq 0$) et $y \neq 0$, de sorte que

$$y = \frac{-p}{\Psi_k h'(\|d^*\|)}.$$

La proposition B.26 nous permet d'écrire

$$y \in B^* = \{t \in \mathbb{R}^n : \|t\|^* \leq 1\} \text{ et } \langle y, d^* \rangle = \|d^*\|,$$

d'où

$$\langle -p, d^* \rangle = \Psi_k h'(\|d^*\|) \|d^*\|. \quad (5.28)$$

De plus, $\|y\|^* \leq 1$, puisque $y \in B^*$, et on a

$$\|p\|^* \leq \Psi_k h'(\|d^*\|). \quad (5.29)$$

Par conséquent

$$\begin{aligned} \Psi_k h'(\|d^*\|) \|d^*\| &= \langle -p, d^* \rangle && \text{par (5.28)} \\ &\leq \|p\|^* \|d^*\| && \text{par (B8)} \\ &\leq \Psi_k h'(\|d^*\|) \|d^*\| && \text{par (5.29)} \end{aligned}$$

d'où

$$\langle p, d^* \rangle = -\|p\|^* \|d^*\| \quad (5.30)$$

et

$$\|p\|^* = \Psi_k h'(\|d^*\|) \quad (5.31)$$

de sorte que

$$\|d^*\| = h'^{-1} \left(\frac{\|p\|^*}{\Psi_k} \right) \quad (5.32)$$

et

$$\langle p, d^* \rangle + \Psi_k h(\|d^*\|) = -\|p\|^* h'^{-1} \left(\frac{\|p\|^*}{\Psi_k} \right) + \Psi_k h \left(h'^{-1} \left(\frac{\|p\|^*}{\Psi_k} \right) \right). \quad (5.33)$$

Le problème dual est alors

$$\begin{aligned} \text{maximiser}_{\pi, \mu, \omega} \quad & \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} [f_{ij}^k - f_i(x^k)] - \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} b_{ij}^k - \sum_{j \in J_c^k} \omega_j b_j^k \\ & - \|p\|^* h'^{-1} \left(\frac{\|p\|^*}{\Psi_k} \right) + \Psi_k h \left(h'^{-1} \left(\frac{\|p\|^*}{\Psi_k} \right) \right) \end{aligned}$$

sous les contraintes

$$\begin{aligned} \sum_{j \in J_{f_i}^k} \pi_{ij} &= 1, \quad i \in \{1, \dots, r\} \\ \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} + \sum_{j \in J_c^k} \omega_j &\leq 1 \\ p - \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} g_i^j - \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} a_i^j - \sum_{j \in J_c^k} \omega_j a^j &= 0 \\ \pi_{ij} &\geq 0, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \\ \mu_{ij} &\geq 0, \quad j \in J_{c_i}^k, \quad i \in \{1, \dots, r\} \\ \omega_j &\geq 0, \quad j \in J_c^k. \end{aligned} \tag{5.34}$$

Remarques:

- (i) Bien que les calculs ayant conduit à cette formulation supposaient que $p \neq 0$, il est facile de vérifier qu'elle est valide pour $p = 0$ (d'où $d^* = 0$, $h(\|d^*\|) = h'(\|d^*\|) = 0$).
- (ii) Afin de simplifier l'écriture de la fonction économique, nous avons préféré ne pas remplacer p par l'expression (5.26). En pratique, on élimine la contrainte en p en faisant la substitution dans la fonction économique.

Bref, nous avons démontré le résultat suivant.

Proposition 5.6.1 *Les problèmes (5.22a)–(5.22f) et (5.34) sont duaux.*

Remarquons que, exception faite des termes mettant en jeu la norme duale, les problèmes (5.34) et (5.20) sont semblables. La fonction économique du problème (5.34) peut être simplifiée si la fonction $h : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ possède des propriétés additionnelles.

Supposons par exemple que $h(r) = r^q$, $\forall r \geq 0$, avec $q \geq 1$ (cas d'une fonction positivement homogène de degré q). Le cas où $q = 1$ est à éviter car la fonction h n'a alors pas de dérivée bijective. Si $q > 1$, l'expression (5.33) devient

$$\langle p, d^* \rangle + \Psi_k h(\|d^*\|) = - \left(\frac{q-1}{q} \right) \frac{1}{(q\Psi_k)^{\frac{1}{q-1}}} (\|p\|^*)^{\frac{q}{q-1}}$$

et la fonction économique du problème (5.34) est alors

$$\begin{aligned} \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} [f_{ij}^k - f_i(x^k)] &= \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} b_{ij}^k - \sum_{j \in J_c^k} \omega_j b_j^k \\ &- \left(\frac{q-1}{q} \right) \frac{1}{(q\Psi_k)^{\frac{1}{q-1}}} (\|p\|^*)^{\frac{q}{q-1}} \end{aligned} \quad (5.35)$$

Si la norme utilisée est la norme ℓ_q alors la norme duale est, par la proposition B.23, la norme ℓ_s où

$$s = \frac{q}{q-1},$$

de sorte que l'expression (5.35) est une expression polynomiale de degré s .

En optimisation non-différentiable convexe, on utilise la plupart du temps la norme euclidienne. Il est, dans ce cas, judicieux de choisir $h(r) = r^2$. On a alors $q = 2$ et l'expression (5.35) devient

$$\sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} [f_{ij}^k - f_i(x^k)] - \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} b_{ij}^k - \sum_{j \in J_c^k} \omega_j b_j^k - \frac{1}{4\Psi_k} \|p\|^2 \quad (5.36)$$

Les normes ℓ_1 et ℓ_∞ sont aussi quelquefois utilisées, le plus souvent pour des problèmes particuliers (Bartels, Conn et Charalambous, 1978; Bartels, Conn et

Sinclair, 1978; Streit, 1986). Il est intéressant dans ce cas de poser $h(r) = r$ car le problème (5.22a)–(5.22f) en est alors un de programmation linéaire. Le problème dual n'est plus le problème (5.34) car $q = 1$ et h' n'est pas bijective.

Il se peut alors, si la constante $\Psi_k > 0$ n'est pas suffisamment grande, que le problème (5.22a)–(5.22f) n'aie pas d'optimum fini. Supposons que ce ne soit pas le cas. On a alors, en utilisant l'expression (5.28),

$$\langle p, d^* \rangle + \Psi_k h(\|d^*\|) = -\Psi_k \|d^*\| + \Psi_k \|d^*\| = 0$$

de sorte que la fonction duale (5.25) se simplifie pour devenir

$$D(\eta, \pi, \mu, \omega, \rho) = \sum_{i=1}^r \left[\sum_{j \in J_{f_i}^k} \pi_{ij} (f_{ij}^k - f_i(x^k)) - \sum_{j \in J_{c_i}^k} \mu_{ij} b_{ij}^k \right] - \sum_{j \in J_c^k} \omega_j b_j^k. \quad (5.37)$$

De plus, l'expression (5.29) devient

$$\|p\|^* \leq \Psi_k. \quad (5.38)$$

En utilisant (5.37), (5.38) et les contraintes (5.23), on obtient le problème dual suivant:

$$\text{maximiser } \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} [f_{ij}^k - f_i(x^k)] - \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} b_{ij}^k - \sum_{j \in J_c^k} \omega_j b_j^k$$

sous les contraintes

$$\begin{aligned} \sum_{j \in J_{f_i}^k} \pi_{ij} &= 1, \quad i \in \{1, \dots, r\} \\ \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} + \sum_{j \in J_c^k} \omega_j &\leq 1 \\ \left\| \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} g_i^j + \sum_{i=1}^r \sum_{j \in J_{c_i}^k} \mu_{ij} a_i^j + \sum_{j \in J_c^k} \omega_j a^j \right\|^* &\leq \Psi_k \\ \pi_{ij} &\geq 0, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \\ \mu_{ij} &\geq 0, \quad j \in J_{c_i}^k, \quad i \in \{1, \dots, r\} \\ \omega_j &\geq 0, \quad j \in J_c^k. \end{aligned} \quad (5.39)$$

Ce problème en est un de programmation linéaire lorsque l'on utilise l'une des normes ℓ_1 ou ℓ_∞ . Il a été obtenu en faisant l'hypothèse que le problème primal (5.22a)–(5.22f) possède une solution optimale finie. Si ce n'est pas le cas, alors la fonction économique du problème primal n'est pas bornée inférieurement dans la région de l'espace correspondant aux solutions réalisables. Le problème dual n'est alors pas réalisable et la constante Ψ_k est trop petite. Il faut alors l'augmenter et résoudre à nouveau le problème (5.39). On répète le procédé jusqu'à ce que ce problème devienne réalisable. Il est probablement plus avantageux, lorsque $h(r) = r$ et que l'on utilise l'une des normes ℓ_1 ou ℓ_∞ , de recourir aux formulations (5.16) et (5.20) de l'approche des régions de confiance.

Le prochain résultat est une adaptation du lemme 5.2.3 de Kiwiel (1985). Il est valable sous les hypothèses initiales à propos de la fonction h et nous en omettons la preuve.

Lemme 5.6.4

- (a) Il existe toujours une valeur $\underline{\Psi}_k > 0$ telle que le problème (5.22a)–(5.22f) possède une solution optimale finie, $\forall \Psi_k \geq \underline{\Psi}_k$.
- (b) $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ est une solution optimale du problème (5.22a)–(5.22f) si et seulement si il existe des multiplicateurs de Lagrange $\pi_{ij}^k, \mu_{ij}^k, \omega_j^k$ et des vecteurs p_f^k, p_D^k et p^k de $\mathbb{R}^n$ vérifiant:

$$\left. \begin{aligned} \pi_{ij}^k &\geq 0, \quad j \in J_{fi}^k, \quad i \in \{1, \dots, r\} \\ \mu_{ij}^k &\geq 0, \quad j \in J_{ci}^k, \quad i \in \{1, \dots, r\} \\ \omega_j^k &\geq 0, \quad j \in J_c^k \end{aligned} \right\} \quad (5.40a)$$

$$\sum_{j \in J_f^k} \pi_{ij}^k = 1, \quad i \in \{1, \dots, r\} \quad (5.40b)$$

$$\sum_{i=1}^r \sum_{j \in J_a^k} \mu_{ij}^k + \sum_{j \in J_c^k} \omega_j^k \leq 1 \quad (5.40c)$$

$$\sum_{i=1}^r v_i^k - \sigma_f^k = f(x^k) \quad (5.40d)$$

$$\left(\langle g_i^j, d^k \rangle - v_i^k + f_{ij}^k \right) \pi_{ij}^k = 0, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \quad (5.40e)$$

$$\left(\langle a_i^j, d^k \rangle - \sigma_D^k - b_{ij}^k \right) \mu_{ij}^k = 0, \quad j \in J_a^k, \quad i \in \{1, \dots, r\} \quad (5.40f)$$

$$\left(\langle a^j, d^k \rangle - \sigma_D^k - b_j^k \right) \omega_j^k = 0, \quad j \in J_c^k \quad (5.40g)$$

$$\left(1 - \sum_{i=1}^r \sum_{j \in J_a^k} \mu_{ij}^k - \sum_{j \in J_c^k} \omega_j^k \right) \sigma_D^k = 0 \quad (5.40h)$$

$$p_f^k = \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij}^k g_i^j \quad (5.40i)$$

$$p_D^k = \sum_{i=1}^r \sum_{j \in J_a^k} \mu_{ij}^k a_i^j + \sum_{j \in J_c^k} \omega_j^k a^j \quad (5.40j)$$

$$p^k = p_f^k + p_D^k \quad (5.40k)$$

$$\|p^k\|^* = \Psi_k h'(\|d^k\|) \text{ et } \|d^k\| = h'^{-1} \left(\frac{\|p^k\|^*}{\Psi_k} \right) \quad (5.40l)$$

$$p^k \in -\Psi_k h'(\|d^k\|) \partial \| \cdot \| (d^k) \\ \text{ou } d^k \in \text{zéro} \{ \Psi_k h'(\|d\|) \partial \| \cdot \| (d) + p^k \} \quad (5.40m)$$

$$\sigma_f^k = \langle p_f^k, d^k \rangle + \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij}^k (f_{ij}^k - f_i(x^k)) \quad (5.40n)$$

$$\sigma_D^k = \langle p_D^k, d^k \rangle - \sum_{i=1}^r \sum_{j \in J_a^k} \mu_{ij}^k b_{ij}^k - \sum_{j \in J_c^k} \omega_j^k b_j^k \quad (5.40o)$$

$$\sigma^k = \sigma_f^k + \sigma_D^k = -\|p^k\|^* h'^{-1} \left(\frac{\|p^k\|^*}{\Psi_k} \right) + \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij}^k (f_{ij}^k - f_i(x^k)) \\ - \sum_{i=1}^r \sum_{j \in J_a^k} \mu_{ij}^k b_{ij}^k - \sum_{j \in J_c^k} \omega_j^k b_j^k \quad (5.40p)$$

$$\langle g_i^j, d^k \rangle - v_i^k \leq -f_{ij}^k, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \quad (5.40q)$$

$$\langle a_i^j, d^k \rangle - \sigma_D^k \leq b_{ij}^k, \quad j \in J_{\alpha_i}^k, \quad i \in \{1, \dots, r\} \quad (5.40r)$$

$$\langle a^j, d^k \rangle - \sigma_D^k \leq b_j^k, \quad j \in J_c^k. \quad (5.40s)$$

(c) Les multiplicateurs π_{ij}^k, μ_{ij}^k et ω_j^k vérifiant les expressions (5.40) si et seulement si (π^k, μ^k, ω^k) constitue une solution optimale du problème (5.34).

Remarque: Lorsque $h(r) = r^2$ et que la norme euclidienne est utilisée, les expressions (5.40l), (5.40m) et (5.40p) deviennent, en vertu du corollaire B.25,

$$\begin{aligned} \|p^k\| &= 2\Psi_k \|d^k\| \quad \text{et} \quad \|d^k\| = \frac{\|p^k\|}{2\Psi_k} \\ p^k &= -2\Psi_k d^k, \quad \text{c'est-à-dire} \quad d^k = \frac{-p^k}{2\Psi_k} \end{aligned}$$

et

$$\begin{aligned} \sigma^k &= -\frac{\|p^k\|^2}{2\Psi_k} + \sum_{i=1}^r \sum_{j \in J_{\alpha_i}^k} \pi_{ij}^k [f_{ij}^k - f_i(x^k)] \\ &\quad - \sum_{i=1}^r \sum_{j \in J_{\alpha_i}^k} \mu_{ij}^k b_{ij}^k - \sum_{j \in J_c^k} \omega_j^k b_j^k. \end{aligned}$$

En posant $\Psi_k = \frac{1}{2}$, $C = \mathbb{R}^n$ (d'où $J_c^k = \emptyset$) et $r = 1$, on obtient les résultats de Kiwiel (1985).

Remarque: Dans le cas plus général où $h(r) = r^q$, $q \in \{2, 3, 4, \dots\}$, et que la norme ℓ_q est utilisée, les expressions (5.40l), (5.40m) et (5.40p) deviennent

$$\begin{aligned} \|p^k\|_s &= q\Psi_k (\|d^k\|_q)^{q-1} \quad \text{où} \quad s = q/(q-1) \\ p^k &= -q\Psi_k \delta^k \quad \text{où} \quad \delta^k \in \mathbb{R}^n \text{ tel que } \delta_j^k = |d_j^k|^{q-1} \operatorname{sgn}(d_j^k) \\ &\quad \text{et} \quad \operatorname{sgn}(r) = \begin{cases} +1 & \text{si } r \geq 0 \\ -1 & \text{sinon} \end{cases} \\ d_j^k &= \left| \frac{p_j^k}{q\Psi_k} \right|^{\frac{1}{q-1}} \operatorname{sgn}(-p_j^k), \quad j \in \{1, \dots, r\} \end{aligned}$$

et

$$\begin{aligned}\sigma^k &= \frac{-(\|p^k\|_s)^s}{(q\Psi_k)^{\frac{1}{q-1}}} + \sum_{i=1}^r \sum_{j \in J_{fi}^k} \pi_{ij}^k [f_{ij}^k - f_i(x^k)] \\ &\quad - \sum_{i=1}^r \sum_{j \in J_{\alpha i}^k} \mu_{ij}^k b_{ij}^k - \sum_{j \in J_{\varepsilon}^k} \omega_j^k b_j^k.\end{aligned}$$

Le reste de ce chapitre est consacré à l'étude de trois stratégies de gestion de l'information accumulée dans le cadre de l'approche des pénalisations. Nous présentons la stratégie classique d'accumulation, la stratégie d'accumulation/épuration et, enfin, la stratégie d'agrégation de l'information.

5.7 La stratégie d'accumulation

La formulation du problème (5.22a)–(5.22f) repose sur l'utilisation des linéarisations $\underline{f}_i(\cdot|y^j)$, $I_{\alpha i}^j(\cdot)$ et $I_{\varepsilon}^j(\cdot)$ définies en (5.2) et en (5.6). À partir de maintenant, nous travaillerons plutôt avec les déplacements d . En posant $x = x^k + d$ et en se rappelant que $f_i^j \triangleq f_i(y^j)$, on définit les linéarisations suivantes:

$$\begin{aligned}f_{ijk}(d) &\triangleq \underline{f}_i(x^k + d|y^j) \\ &= f_{ij}^k + \langle g_i^j, d \rangle, \quad j \in J_{fi}^k, \quad i \in \{1, \dots, r\} \\ &\text{par (5.2) et (5.17a)}\end{aligned}\tag{5.41a}$$

$$\begin{aligned}I_{ijk}(d) &\triangleq \langle a_i^j, x^k + d \rangle - b_i^j \\ &= \langle a_i^j, d \rangle - b_{ij}^k, \quad j \in J_{\alpha i}^k, \quad i \in \{1, \dots, r\} \\ &\text{par (5.17b)}\end{aligned}\tag{5.41b}$$

et

$$\begin{aligned}
 I_{ojk}(d) &\triangleq \langle a^j, x^k + d \rangle - b^j \\
 &= \langle a^j, d \rangle - b_j^k, \quad j \in J_c^k \\
 &\text{par (5.17c).}
 \end{aligned} \tag{5.41c}$$

On a alors

$$\hat{f}_i^k(x^k + d) = \max\{f_{ijk}(d) : j \in J_{fi}^k\}, \quad i \in \{1, \dots, r\} \tag{5.42a}$$

$$\hat{I}_{ci}^k(x^k + d) = \max\{I_{ijk}(d) : j \in J_{ci}^k\}, \quad i \in \{1, \dots, r\} \tag{5.42b}$$

et

$$\hat{I}_c^k(x^k + d) = \max\{I_{ojk}(d) : j \in J_c^k\}. \tag{5.42c}$$

On obtient la *stratégie d'accumulation* de l'information (plus précisément des ε -sous-gradients) lorsque les ensembles J_{fi}^k , J_{ci}^k et J_c^k sont de taille maximale. C'est la stratégie utilisée par la méthode de plans coupants de Kelley (1960) et Cheney-Goldstein (1959) ainsi que par plusieurs algorithmes de décomposition inspirés de cette méthode. Leurs preuves de convergence sont simples et exploitent l'accumulation de l'information.

La direction d^k obtenue par résolution du problème (5.22a)–(5.22f), ou par application de la condition (5.40m) après résolution du problème dual (5.34), détermine un point $y^{k+1} \triangleq x^k + d^k$ qui deviendra, s'il est "suffisamment bon", le prochain centre de stabilité x^{k+1} . On posera $x^{k+1} = y^{k+1}$ et $\delta^k = 1$ si $y^{k+1} \in D$ et provoque une décroissance "suffisante" de la valeur de la fonction économique (se référer au chapitre 4). On posera $x^{k+1} = x^k$ et $\delta^k = 0$, si ce n'est pas le cas. On aura donc

$$x^{k+1} = x^k + \delta^k d^k \tag{5.43}$$

où $\delta^k \in \{0, 1\}$.

Le problème (5.22a)–(5.22f) de l'itération $k + 1$ est obtenu en ajoutant r ou $r + 1$ coupes additionnelles (selon que y^{k+1} est dans C ou pas) et en mettant à jour les linéarisations (5.41) qui le définissent.

Pour chaque indice $i \in \{1, \dots, r\}$, on distingue deux cas selon que $y^{k+1} \in \text{dom } f_i$ ou pas.

Premier cas: $y^{k+1} \in \text{dom } f_i$

L'approximation polyédrique $\hat{f}_i^k$ de f_i est améliorée par l'ajout de la coupe

$$\langle g_i^{k+1}, d \rangle - v_i \leq f_{i,k+1}^{k+1}$$

où $g_i^{k+1} \in \partial f_i(y^{k+1}), \quad f_i^{k+1} = f_i(y^{k+1})$

et $f_{i,k+1}^{k+1} = f_i^{k+1} - \delta^k \langle g_i^{k+1}, d^k \rangle.$

On pose $J_{fi}^{k+1} = J_{fi}^k \cup \{k + 1\}$ et les scalaires $f_{ij}^k, j \in J_{fi}^k$, deviennent

$$f_{ij}^{k+1} = f_{ij}^k + \delta^k \langle g_i^j, d^k \rangle \quad \text{par (5.17a) et (5.43).}$$

On pose $J_{ci}^{k+1} = J_{ci}^k$ et les scalaires $b_{ij}^k, j \in J_{ci}^k$, sont changés pour

$$b_{ij}^{k+1} = b_{ij}^k - \delta^k \langle a_i^j, d^k \rangle \quad \text{par (5.17b) et (5.43).}$$

Deuxième cas: $y^{k+1} \notin \text{dom } f_i$

On a $\delta^k = 0$ et $x^{k+1} = x^k$. On pose $J_{ci}^{k+1} = J_{ci}^k \cup \{k + 1\}$ et l'approximation

polyédrique $\widehat{I}_{\alpha_i}^k$ de I_{α_i} est améliorée en ajoutant la coupe

$$\langle a_i^j, d \rangle - \sigma_D \leq b_{i,k+1}^{k+1}$$

où $b_{i,k+1}^{k+1} = b_i^{k+1} - \langle a_i^{k+1}, x^{k+1} \rangle = b_i^{k+1} - \langle a_i^{k+1}, x^k \rangle$

et $(a_i^{k+1}, b_i^{k+1}) \in \mathbb{R}^n \times \mathbb{R}$ sont à définir un demi-espace fermé contenant $\text{dom } f_i$ mais pas y^{k+1} .

On pose $b_{ij}^{k+1} = b_{ij}^k, \forall j \in J_{\alpha_i}^k, J_{fi}^{k+1} = J_{fi}^k$ et $f_{ij}^{k+1} = f_{ij}^k, \forall j \in J_{fi}^k$.

Il faut aussi s'occuper des coupes associées à l'ensemble C . Ici aussi il faut distinguer deux cas selon que $y^{k+1} \in C$ ou pas. La mise-à-jour de l'ensemble J_c^k et des coupes définies par les $(a^j, b^j) \in \mathbb{R}^n \times \mathbb{R}$ se fait de la même manière que dans le cas des $J_{\alpha_i}^k$ et (a_i^j, b_i^j) . Enfin, la contrainte (5.22b) est révisée en remplaçant son second membre $f(x^k)$ par $f(x^{k+1})$.

Remarque: Les formules de mise-à-jour des

$$f_{ij}^k, \quad j \in J_{fi}^k, \quad i \in \{1, \dots, r\}$$

$$b_{ij}^k, \quad j \in J_{\alpha_i}^k, \quad i \in \{1, \dots, r\}$$

et $b_j^k, \quad j \in J_c^k$

rendent inutile le stockage des solutions intermédiaires y^j produites par l'algorithme.

L'accumulation systématique de toutes les informations produites par l'algorithme constitue un grave problème. Théoriquement, une méthode basée sur cette stratégie requiert une capacité infinie de stockage. De plus, le problème (5.22a)–(5.22f) devient de plus en plus difficile et long à résoudre. La méthode est, pour ces raisons, peu attrayante. Dans ce qui suit, nous présentons deux stratégies conduisant à des méthodes qui ne sont pas affublées de ces tares.

5.8 La stratégie d'accumulation/épuration

Lorsque le nombre de coupes, ou de colonnes du problème dual, est trop élevé, il est pratique courante d'en éliminer. Cette élimination est très souvent faite selon des règles heuristiques qui peuvent, par malchance, nuire à la convergence de l'algorithme. Le résultat suivant, peut être utilisé pour contrôler la taille du problème (5.22a)–(5.22f). Il s'agit d'une simple adaptation des deux dernières parties du lemme 5.2.3 de Kiwiel (1985) et nous n'en donnons pas la démonstration.

Lemme 5.8.5 *Il existe une solution optimale (π^k, μ^k, ω^k) du problème dual (5.34) telle que les ensembles*

$$\hat{J}_{fi}^k \triangleq \{j \in J_{fi}^k : \pi_{ij}^k > 0\}, \quad i \in \{1, \dots, r\} \quad (5.44a)$$

$$\hat{J}_{\alpha i}^k \triangleq \{j \in J_{\alpha i}^k : \mu_{ij}^k > 0\}, \quad i \in \{1, \dots, r\} \quad (5.44b)$$

$$\text{et} \quad \hat{J}_c^k \triangleq \{j \in J_c^k : \omega_j^k > 0\} \quad (5.44c)$$

vérifient

$$\sum_{i=1}^r \left(|\hat{J}_{fi}^k| + |\hat{J}_{\alpha i}^k| \right) + |\hat{J}_c^k| \leq n + r + 1. \quad (5.45)$$

Une telle solution peut être obtenue par résolution du problème de programmation linéaire suivant à l'aide de la méthode du simplexe:

$$\text{maximiser}_{\pi, \mu, \omega} \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} [f_{ij}^k - f_i(x^k)] - \sum_{i=1}^r \sum_{j \in J_{a_i}^k} \mu_{ij} b_{ij}^k - \sum_{j \in J_c^k} \omega_j b_j^k$$

sous les contraintes

$$\begin{aligned} \sum_{j \in J_{f_i}^k} \pi_{ij} &= 1, \quad i \in \{1, \dots, r\} \\ \sum_{i=1}^r \sum_{j \in J_{a_i}^k} \mu_{ij} + \sum_{j \in J_c^k} \omega_j &\leq 1 \\ \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \pi_{ij} g_i^j + \sum_{i=1}^r \sum_{j \in J_{a_i}^k} \mu_{ij} a_i^j + \sum_{j \in J_c^k} \omega_j a_j^j &= p^k \\ \pi_{ij} &\geq 0, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \\ \mu_{ij} &\geq 0, \quad j \in J_{a_i}^k, \quad i \in \{1, \dots, r\} \\ \omega_j &\geq 0, \quad j \in J_c^k \end{aligned} \tag{5.46}$$

où p^k est défini en (5.40m).

De plus, les solutions $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ du problème primal (5.22a)–(5.22f) sont aussi des solutions optimales du problème réduit ci-dessous:

$$\text{minimiser}_{d, v, \sigma_f, \sigma_D} \sigma_f + \sigma_D + \Psi_k h(\|d\|)$$

sous les contraintes

$$\begin{aligned} \sum_{i=1}^r v_i - \sigma_f &\leq f(x^k) \\ \langle g_i^j, d \rangle - v_i &\leq -f_{ij}^k, \quad j \in \hat{J}_{f_i}^k, \quad i \in \{1, \dots, r\} \\ \langle a_i^j, d \rangle - \sigma_D &\leq b_{ij}^k, \quad j \in \hat{J}_{a_i}^k, \quad i \in \{1, \dots, r\} \\ \langle a^j, d \rangle - \sigma_D &\leq b_j^k, \quad j \in \hat{J}_c^k \\ \sigma_D &\geq 0. \end{aligned} \tag{5.47}$$

Observons que si le vecteur p^k obtenu en (5.40m) est unique, les problèmes (5.34) et (5.46) ont les mêmes solutions optimales.

Ce lemme est d'un grand intérêt car il pave la voie à la conception d'algorithmes nécessitant de l'espace mémoire pour stocker pas plus de $n + r + 2$ contraintes pour le problème (5.22a)–(5.22f) ou pas plus de $n + r + 1$ variables pour le problème dual (5.34). Si l'algorithme de résolution du problème (5.34) produisait une solution optimale ayant plus de $n + r + 1$ variables non nulles, il suffirait d'évaluer p^k avec celle-ci puis de résoudre le problème (5.46) à l'aide de la méthode du simplexe afin de récupérer une solution vérifiant la condition (5.45).

La stratégie d'accumulation/épuration fonctionne comme suit. À l'itération k , on résout le problème (5.22a)–(5.22f) et on obtient une solution optimale $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ ainsi que des multiplicateurs de Lagrange optimaux (π^k, μ^k, ω^k) pour le problème (5.34). Le lemme 5.8.5 nous assure que $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ est aussi une solution optimale du problème réduit (5.47) dont la structure est celle du problème (5.22a)–(5.22f). S'il est nécessaire d'effectuer une autre itération, on remplace d'abord les ensembles J_{fi}^k , J_{ci}^k et J_c^k par les ensembles définis en (5.44). Ceci revient à remplacer le problème (5.22a)–(5.22f) par le problème réduit (5.47). Le problème (5.22a)–(5.22f) de l'itération $k + 1$ est alors obtenu, comme dans la stratégie d'accumulation, en lui ajoutant r ou $r + 1$ coupes additionnelles et en mettant à jour les coefficients des linéarisations qui le définissent.

Les ensembles $\hat{J}_{fi}^k$, $\hat{J}_{ci}^k$ et $\hat{J}_c^k$ permettent de définir les approximations polyédriques courantes $\hat{f}_e^k$ et $\hat{I}_{De}^k$ de f et de I_D :

$$\hat{f}_e^k(x^k + d) = \sum_{i=1}^r \hat{f}_{ie}^k(x^k + d) \quad (5.48a)$$

$$\hat{I}_{De}^k(x^k + d) = \max \left\{ \hat{I}_{ce}^k(x^k + d), \hat{I}_{cle}^k(x^k + d), \dots, \hat{I}_{cre}^k(x^k + d), 0 \right\} \quad (5.48b)$$

où

$$\hat{f}_{ie}^k(x^k + d) = \max \left\{ f_{ijk}(d) : j \in \hat{J}_{fi}^k \right\}, \quad i \in \{1, \dots, r\} \quad (5.48c)$$

$$\hat{I}_{cie}^k(x^k + d) = \max \left\{ I_{ijk}(d) : j \in \hat{J}_{ci}^k \right\}, \quad i \in \{1, \dots, r\} \quad (5.48d)$$

et

$$\hat{I}_{ce}^k(x^k + d) = \max \left\{ I_{ojk}(d) : j \in \hat{J}_c^k \right\}. \quad (5.48e)$$

Ceci permet d'obtenir l'approximation polyédrique courante $\hat{\varphi}_e^k(x^k + d)$ de la fonction $\varphi(x^k + d)$ définie en (5.12):

$$\hat{\varphi}_e^k(x^k + d) = \hat{f}_e^k(x^k + d) - f(x^k) + \hat{I}_{De}^k(x^k + d). \quad (5.49)$$

Le problème (5.47) est alors

$$\underset{d \in \mathbb{R}^n}{\text{minimiser}} \quad \hat{\varphi}_e^k(x^k + d) + \Psi_k h(\|d\|) \quad (5.50)$$

et ses solutions optimales $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ sont telles que

$$\sigma^k = \sigma_f^k + \sigma_D^k = \hat{\varphi}_e^k(x^k + d^k).$$

5.9 La stratégie d'agrégation

Si le nombre de variables est très élevé, les problèmes (5.47) de la stratégie d'accumulation/épuration peuvent encore être trop gros. On peut alors préférer la *stratégie d'agrégation* de l'information. Celle-ci construit un problème réduit en agrégeant les contraintes du problème (5.22a)–(5.22f) à l'aide de leurs multiplicateurs de Lagrange après une étape préalable de mise en échelle.

Soit (π^k, μ^k, ω^k) une solution optimale du problème (5.34) et définissons les quantités suivantes:

$$\nu_{fi}^k \triangleq \sum_{j \in J_{fi}^k} \pi_{ij}^k \geq 0, \quad i \in \{1, \dots, r\}, \quad (5.51a)$$

$$\nu_{ci}^k \triangleq \sum_{j \in J_{ci}^k} \mu_{ij}^k \geq 0, \quad i \in \{1, \dots, r\}, \quad (5.51b)$$

et

$$\nu_c^k \triangleq \sum_{j \in J_c^k} \omega_j^k \geq 0. \quad (5.51c)$$

Des contraintes du problème (5.34), on trouve

$$\nu_{fi}^k = 1, \quad \forall i \in \{1, \dots, r\} \quad (5.52a)$$

et

$$\sum_{i=1}^r \nu_{ci}^k + \nu_c^k \leq 1. \quad (5.52b)$$

Les quantités définies en (5.51) servent à obtenir les quantités $\tilde{\mu}^k$ et $\tilde{\omega}^k$ par mise en échelle des μ^k et ω^k . Il faut que

$$\mu_{ij}^k = \nu_{ci}^k \tilde{\mu}_{ij}^k, \quad j \in J_{ci}^k, \quad i \in \{1, \dots, r\} \quad (5.53a)$$

$$\omega_j^k = \nu_c^k \tilde{\omega}_j^k, \quad j \in J_c^k \quad (5.53b)$$

$$\tilde{\mu}_{ij}^k \geq 0, \quad j \in J_{ci}^k, \quad i \in \{1, \dots, r\} \quad (5.53c)$$

$$\tilde{\omega}_j^k \geq 0, \quad j \in J_c^k \quad (5.53d)$$

$$\sum_{j \in J_{ci}^k} \tilde{\mu}_{ij}^k = 1, \quad i \in \{1, \dots, r\} \quad (5.53e)$$

et

$$\sum_{j \in J_c^k} \tilde{\omega}_j^k = 1. \quad (5.53f)$$

Si $\nu_c^k \neq 0$ et $\nu_{ci}^k \neq 0, \forall i \in \{1, \dots, r\}$, les conditions (5.53a) et (5.53b) suffisent. Si $\nu_c^k = 0$, alors $\omega_j^k = 0, \forall j \in J_c^k$, et les expressions (5.53b) sont vérifiées par tout vecteur $\tilde{\omega}^k$ vérifiant (5.53d) et (5.53f). La même remarque s'applique au calcul des $\tilde{\mu}_{ij}^k$ lorsque $\nu_{ci}^k = 0$.

On remarque que les valeurs duales π^k sont déjà normalisées (on peut supposer que $\tilde{\pi}^k = \pi^k$). Les vecteurs $\pi^k, \tilde{\mu}^k$ et $\tilde{\omega}^k$ étant normalisés, on peut alors définir des coupes agrégées.

Définition: On appelle *coupes agrégées* les couples $(p_{fi}^k, \tilde{f}_{pi}^k), (p_{ci}^k, \tilde{b}_{pi}^k), i \in \{1, \dots, r\}$, et $(p_c^k, \tilde{b}_p^k)$ définis par

$$(p_{fi}^k, \tilde{f}_{pi}^k) \triangleq \sum_{j \in J_{fi}^k} \pi_{ij}^k(g_i^j, f_{ij}^k), \quad i \in \{1, \dots, r\} \quad (5.54a)$$

$$(p_{ci}^k, \tilde{b}_{pi}^k) \triangleq \sum_{j \in J_{ci}^k} \tilde{\mu}_{ij}^k(a_i^j, b_{ij}^k), \quad i \in \{1, \dots, r\} \quad (5.54b)$$

et

$$(p_c^k, \tilde{b}_p^k) \triangleq \sum_{j \in J_c^k} \tilde{\omega}_j^k(a^j, b_j^k). \quad (5.54c)$$

Lemme 5.9.6 Les coupes agrégées définies par les expressions (5.54) sont telles que

$$p^k = \sum_{i=1}^r (p_{fi}^k + \nu_{ci}^k p_{ci}^k) + \nu_c^k p_c^k \quad (5.55)$$

et

$$\sigma^k = -\|p^k\|^\alpha h'^{-1}\left(\frac{\|p^k\|^\alpha}{\Psi_k}\right) - \tilde{\alpha}^k \quad (5.56)$$

où

$$\tilde{\alpha}^k = \sum_{i=1}^r [f_i(x^k) - \tilde{f}_{pi}^k] + \sum_{i=1}^r \nu_{ci}^k \tilde{b}_{pi}^k + \nu_c^k \tilde{b}_p^k \quad (5.57)$$

Démonstration: L'expression (5.55) découle de (5.40i), (5.40j), (5.40k), des expressions (5.53a) et (5.53b), et des définitions (5.54).

Les expressions (5.56) et (5.57) découlent de (5.40p), des expressions (5.53a) et (5.53b), des définitions (5.54) et (5.40b). ■

Remarque: Dans le cas de l'approche des régions de confiance, l'expression (5.56) est remplacée par

$$\sigma^k = -\gamma_k \|p^k\|^* - \tilde{\alpha}^k.$$

Le résultat suivant présente le sous-problème agrégé qui justifie la stratégie d'agrégation. Il s'agit d'une adaption du lemme 5.2.4 de Kiwiel (1985) et nous omettons sa démonstration.

Lemme 5.9.7 *Les solutions optimales du problème (5.22a)–(5.22f) sont des solutions optimales du problème agrégé ci-dessous:*

$$\begin{aligned}
 & \underset{d, v, \sigma_f, \sigma_D}{\text{minimiser}} \sigma_f + \sigma_D + \Psi_k h(\|d\|) \\
 & \text{sous les contraintes} \\
 & \sum_{i=1}^r v_i - \sigma_f \leq f(x^k) \\
 & \langle p_{fi}^k, d \rangle - v_i \leq -\tilde{f}_{pi}^k, \quad i \in \{1, \dots, r\} \\
 & \langle p_{ci}^k, d \rangle - \sigma_D \leq \tilde{b}_{pi}^k, \quad i \in \{1, \dots, r\} \\
 & \langle p_c^k, d \rangle - \sigma_D \leq \tilde{b}_p^k \\
 & \langle g_i^j, d \rangle - v_i \leq -f_{ij}^k, \quad j \in \tilde{J}_{fi}^k, \quad i \in \{1, \dots, r\} \\
 & \langle a_i^j, d \rangle - \sigma_D \leq b_{ij}^k, \quad j \in \tilde{J}_{ci}^k, \quad i \in \{1, \dots, r\} \\
 & \langle a^j, d \rangle - \sigma_D \leq b_j^k, \quad j \in \tilde{J}_c^k \\
 & \sigma_D \geq 0
 \end{aligned} \tag{5.58}$$

où les sous-ensembles

$$\begin{aligned}
 \tilde{J}_{fi}^k & \subseteq J_{fi}^k, \quad i \in \{1, \dots, r\} \\
 \tilde{J}_{ci}^k & \subseteq J_{ci}^k, \quad i \in \{1, \dots, r\}
 \end{aligned}$$

et

$$\tilde{J}_c^k \subseteq J_c^k,$$

peuvent être vides.

Lorsque la fonction économique du problème (5.22a)–(5.22f) est strictement convexe, celui-ci possède une *unique* solution optimale. Il en est également ainsi du problème (5.58). Ces deux problèmes étant munis de la même fonction économique, ils doivent alors, par le lemme précédent, posséder la même solution optimale. Si leur fonction économique n'est pas strictement convexe, il faut alors montrer que

les solutions optimales $(\hat{d}^k, \hat{v}^k, \hat{\sigma}_f^k, \hat{\sigma}_D^k)$ du problème (5.58) sont aussi des solutions optimales du problème (5.22a)–(5.22f) avant de conclure que les deux problèmes ont les mêmes ensembles de solutions optimales.

Le résultat suivant, dont la démonstration se trouve à l'annexe C, complète le lemme 5.9.7.

Lemme 5.9.8 *Les solutions optimales du problème (5.58) sont aussi des solutions optimales du problème (5.22a)–(5.22f).*

En combinant les deux derniers lemmes, on obtient le résultat suivant:

Proposition 5.9.9 *Les problèmes (5.22a)–(5.22f) et (5.58) sont équivalents (c'est-à-dire qu'ils ont les mêmes solutions optimales).*

Les contraintes agrégées du problème (5.58) utilisent les *linéarisations agrégées* suivantes:

$$\tilde{f}_i^k(x) = \tilde{f}_{pi}^k + \langle p_{fi}^k, x - x^k \rangle, \quad i \in \{1, \dots, r\} \quad (5.59a)$$

$$\tilde{I}_{ci}^k(x) = -\tilde{b}_{pi}^k + \langle p_{ci}^k, x - x^k \rangle, \quad i \in \{1, \dots, r\} \quad (5.59b)$$

et

$$\tilde{I}_c^k(x) = -\tilde{b}_p^k + \langle p_c^k, x - x^k \rangle. \quad (5.59c)$$

Lemme 5.9.10 *Les linéarisations agrégées (5.59) sont des combinaisons convexes des linéarisations $f_{ijk}(x - x^k)$, $I_{ijk}(x - x^k)$ et $I_{ojk}(x - x^k)$ définies en (5.41). De*

plus,

$$\tilde{f}_i^k(x) \leq f_i(x), \quad \forall x \in \mathbb{R}^n, \quad i \in \{1, \dots, r\} \quad (5.60a)$$

$$\tilde{I}_{ci}^k(x) \leq I_{ci}(x), \quad \forall x \in \mathbb{R}^n, \quad i \in \{1, \dots, r\} \quad (5.60b)$$

$$\tilde{I}_c^k(x) \leq I_c(x), \quad \forall x \in \mathbb{R}^n \quad (5.60c)$$

et

$$\begin{aligned} \tilde{\varphi}^k(x) &\triangleq \tilde{f}^k(x) - f(x^k) + \max\{\tilde{I}_c^k(x), \tilde{I}_{c1}^k(x), \dots, \tilde{I}_{cr}^k(x)\} \\ &\leq \varphi^k(x), \quad \forall x \in \mathbb{R}^n. \end{aligned} \quad (5.60d)$$

Démonstration:

$\forall i \in \{1, \dots, r\}$, on a

$$\begin{aligned} \tilde{f}_i^k(x) &= \sum_{j \in J_{f_i}^k} \pi_{ij}^k \left(f_{ij}^k + \langle g_i^j, x - x^k \rangle \right) && \text{par (5.59a) et (5.54a)} \\ &= \sum_{j \in J_{f_i}^k} \pi_{ij}^k f_{ijk}(x - x^k) && \text{par (5.41a)} \\ &\leq \sum_{j \in J_{f_i}^k} \pi_{ij}^k f_i(x) \\ &= f_i(x) && \text{par (5.52a) et (5.51a).} \end{aligned}$$

De même, on démontre les inégalités (5.60b) et (5.60c). L'inégalité (5.60d) découle directement des trois précédentes. ■

La stratégie d'agrégation fonctionne comme suit. À l'itération k , on résout le problème (5.22a)–(5.22f) et on obtient une solution optimale $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ ainsi que des multiplicateurs de Lagrange optimaux (π^k, μ^k, ω^k) pour le problème (5.34). Le lemme 5.9.7 nous assure que $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ est aussi une solution optimale du

problème agrégé (5.58) dont la structure est celle du problème (5.22a)–(5.22f). S'il est nécessaire d'effectuer une autre itération, on forme d'abord le problème agrégé (5.58) que l'on substitue au problème (5.22a)–(5.22f). Le problème (5.22a)–(5.22f) de l'itération $k + 1$ est alors obtenu, comme dans la stratégie d'accumulation, en lui ajoutant r ou $r + 1$ coupes additionnelles et en mettant à jour les coefficients des linéarisations désignées par les sous-ensembles $\tilde{J}_{fi}^k$, $\tilde{J}_{ci}^k$, $i \in \{1, \dots, r\}$, et $\tilde{J}_c^k$ du lemme 5.9.7.

Remarquons que les linéarisations agrégées (5.59) peuvent aussi s'écrire comme suit:

$$\tilde{f}_i^k(x) = f_{pi}^{k+1} + \langle p_{fi}^k, x - x^{k+1} \rangle, \quad i \in \{1, \dots, r\} \quad (5.61a)$$

$$\tilde{I}_{ci}^k(x) = -b_{pi}^{k+1} + \langle p_{ci}^k, x - x^{k+1} \rangle, \quad i \in \{1, \dots, r\} \quad (5.61b)$$

$$\tilde{I}_c^k(x) = -b_p^{k+1} + \langle p_c^k, x - x^{k+1} \rangle \quad (5.61c)$$

où

$$f_{pi}^{k+1} \triangleq \tilde{f}_i^k(x^{k+1}) = \tilde{f}_{pi}^k + \delta^k \langle p_{fi}^k, d^k \rangle, \quad i \in \{1, \dots, r\} \quad (5.62a)$$

$$b_{pi}^{k+1} \triangleq -\tilde{I}_{ci}^k(x^{k+1}) = \tilde{b}_{pi}^k - \delta^k \langle p_{ci}^k, d^k \rangle, \quad i \in \{1, \dots, r\} \quad (5.62b)$$

$$b_p^{k+1} \triangleq -\tilde{I}_c^k(x^{k+1}) = \tilde{b}_p^k - \delta^k \langle p_c^k, d^k \rangle. \quad (5.62c)$$

En utilisant les expressions (5.61), on obtient alors la formulation du problème (5.22a)–(5.22f) de l'itération $k + 1$:

minimiser $\sigma_f + \sigma_D + \Psi_{k+1}h(\|d\|)$
 d, v, σ_f, σ_D

sous les contraintes

$$\begin{aligned}
 \sum_{i=1}^r v_i - \sigma_f &\leq f(x^{k+1}) \\
 \langle p_{fi}^k, d \rangle - v_i &\leq -f_{pi}^{k+1}, \quad i \in \{1, \dots, r\} \\
 \langle p_{ci}^k, d \rangle - \sigma_D &\leq b_{pi}^{k+1}, \quad i \in \{1, \dots, r\} \\
 \langle p_c^k, d \rangle - \sigma_D &\leq b_p^{k+1} \\
 \langle g_i^j, d \rangle - v_i &\leq -f_{ij}^{k+1}, \quad j \in \tilde{J}_{fi}^{k+1}, \quad i \in \{1, \dots, r\} \\
 \langle a_i^j, d \rangle - \sigma_D &\leq b_{ij}^{k+1}, \quad j \in \tilde{J}_{ci}^{k+1}, \quad i \in \{1, \dots, r\} \\
 \langle a^j, d \rangle - \sigma_D &\leq b_j^{k+1}, \quad j \in \tilde{J}_c^{k+1} \\
 \sigma_D &\geq 0
 \end{aligned} \tag{5.63}$$

où, $\forall i \in \{1, \dots, r\}$, on a

$$\begin{aligned}
 \tilde{J}_{fi}^{k+1} &\subseteq J_{fi}^k \cup \{k+1\} \\
 \tilde{J}_{ci}^{k+1} &\subseteq J_{ci}^k \cup \{k+1\} \\
 k+1 &\in \tilde{J}_{fi}^{k+1} \cup \tilde{J}_{ci}^{k+1} \\
 k+1 &\notin \tilde{J}_{fi}^{k+1} \cap \tilde{J}_{ci}^{k+1} = \emptyset
 \end{aligned}$$

et

$$\tilde{J}_c^{k+1} \subseteq \tilde{J}_c^k \cup \{k+1\} \text{ avec } k+1 \in \tilde{J}_c^{k+1} \text{ si } y^{k+1} \notin C.$$

Par conséquent, le problème de détermination d'une direction d'amélioration de l'itération k de la stratégie d'agrégation est:

minimiser $\sigma_f + \sigma_D + \Psi_k h(\|d\|)$
 d, v, σ_f, σ_D

sous les contraintes

$$\begin{aligned}
 \sum_{i=1}^r v_i - \sigma_f &\leq f(x^k) \\
 \langle p_{fi}^{k-1}, d \rangle - v_i &\leq -f_{pi}^k, \quad i \in \{1, \dots, r\} \\
 \langle p_{ci}^{k-1}, d \rangle - \sigma_D &\leq b_{pi}^k, \quad i \in \{1, \dots, r\} \\
 \langle p_c^{k-1}, d \rangle - \sigma_D &\leq b_p^k \\
 \langle g_i^j, d \rangle - v_i &\leq -f_{ij}^k, \quad j \in \tilde{J}_{fi}^k, \quad i \in \{1, \dots, r\} \\
 \langle a_i^j, d \rangle - \sigma_D &\leq b_{ij}^k, \quad j \in \tilde{J}_{ci}^k, \quad i \in \{1, \dots, r\} \\
 \langle a^j, d \rangle - \sigma_D &\leq b_j^k, \quad j \in \tilde{J}_c^k \\
 \sigma_D &\geq 0.
 \end{aligned} \tag{5.64}$$

Le problème étant de même structure que le problème (5.22a)–(5.22f), on peut lui appliquer le procédé de mise-en-échelle (5.51)–(5.53) afin d'obtenir, à l'aide des définitions (5.54) les nouvelles contraintes agrégées, où π_{pi}^k , μ_{pi}^k , $i \in \{1, \dots, r\}$, et ω_p^k sont leurs multiplicateurs de Lagrange:

$$(p_{fi}^k, \tilde{f}_{pi}^k) = \pi_{pi}^k (p_{fi}^{k-1}, f_{pi}^k) + \sum_{j \in \tilde{J}_{fi}^k} \pi_{ij}^k (g_i^j, f_{ij}^k), \quad i \in \{1, \dots, r\} \tag{5.65a}$$

$$(p_{ci}^k, \tilde{b}_{pi}^k) = \mu_{pi}^k (p_{ci}^{k-1}, b_{pi}^k) + \sum_{j \in \tilde{J}_{ci}^k} \mu_{ij}^k (a_i^j, b_{ij}^k), \quad i \in \{1, \dots, r\} \tag{5.65b}$$

$$(p_c^k, b_c^k) = \omega_p^k (p_c^{k-1}, b_p^k) + \sum_{j \in \tilde{J}_c^k} \omega_j^k (a^j, b_j^k). \tag{5.65c}$$

La proposition 5.9.9 affirme que le problème (5.22a)–(5.22f) équivaut au suivant:

$$\text{minimiser } \hat{\varphi}_a^k(x^k + d) + \Psi_k h(\|d\|) \tag{5.66a}$$

où

$$\hat{\varphi}_a^k(x) = \hat{f}_a^k(x) - f(x^k) + \hat{I}_{D_a}^k(x) \quad (5.66b)$$

$$\hat{f}_a^k(x) = \sum_{i=1}^r \hat{f}_{ia}^k(x) \quad (5.66c)$$

$$\hat{f}_{ia}^k(x) = \max \left\{ \hat{f}_i^{k-1}(x), f_{ijk}(x - x^k) : j \in \tilde{J}_{fi}^k \right\}, \quad i \in \{1, \dots, r\} \quad (5.66d)$$

$$\hat{I}_{D_a}^k(x) = \max \left\{ \hat{I}_{ca}^k(x), \hat{I}_{c1a}^k(x), \dots, \hat{I}_{cra}^k(x), 0 \right\} \quad (5.66e)$$

$$\hat{I}_{C_a}^k(x) = \max \left\{ \tilde{I}_c^{k-1}(x), I_{ojk}(x - x^k) : j \in \tilde{J}_c^k \right\} \quad (5.66f)$$

et

$$\hat{I}_{cia}^k(x) = \max \left\{ \tilde{I}_{ci}^{k-1}(x), I_{ijk}(x - x^k) : j \in J_{ci}^k \right\}, \quad i \in \{0, \dots, r\}. \quad (5.66g)$$

Lemme 5.9.11 Si $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ est une solution optimale du problème (5.64), alors

$$\begin{aligned} \sigma^k = \sigma_f^k + \sigma_D^k &= \hat{\varphi}_a^k(x^k + d^k) \\ &= \langle p^k, d^k \rangle + \sum_{i=1}^r \left[\hat{f}_{pi}^k - f_i(x^k) - \nu_{ci}^k \tilde{b}_{pi}^k \right] - \nu_c^k \tilde{b}_p^k. \end{aligned} \quad (5.67)$$

Démonstration: Si $(\eta^k, \pi^k, \mu^k, \omega^k, \delta^k) \geq 0$ est le vecteur des valeurs duales optimales associées aux contraintes du problème (5.64), alors

$$\begin{aligned} \eta^k &= 1 \\ \delta^k &= 1 - \sum_{i=1}^r \nu_{ci}^k - \nu_c^k \end{aligned}$$

et on peut écrire les conditions de complémentarité en utilisant les expressions (5.53a) et (5.53b):

$$\sum_{i=1}^r v_i^k - \sigma_f^k - f(x^k) = 0 \quad (5.68a)$$

$$-\left(1 - \sum_{i=1}^r \nu_{ci}^k - \nu_c^k\right) \sigma_D^k = 0 \quad (5.68b)$$

$$\pi_{pi}^k \left[\langle p_{fi}^{k-1}, d^k \rangle - v_i^k + f_{pi}^k \right] = 0, \quad i \in \{1, \dots, r\} \quad (5.69a)$$

$$\pi_{ij}^k \left[\langle g_i^j, d^k \rangle - v_i^k + f_{ij}^k \right] = 0, \quad j \in \hat{J}_{fi}^k, \quad i \in \{1, \dots, r\} \quad (5.69b)$$

$$\nu_{ci}^k \tilde{\mu}_{pi}^k \left[\langle p_{ci}^{k-1}, d^k \rangle - \sigma_D^k - b_{pi}^k \right] = 0, \quad i \in \{1, \dots, r\} \quad (5.70a)$$

$$\nu_{ci}^k \tilde{\mu}_{ij}^k \left[\langle a_i^j, d^k \rangle - \sigma_D^k - b_{ij}^k \right] = 0, \quad j \in \hat{J}_{ci}^k, \quad i \in \{1, \dots, r\} \quad (5.70b)$$

$$\nu_c^k \tilde{\omega}_p^k \left[\langle p_c^{k-1}, d^k \rangle - \sigma_D^k - b_p^k \right] = 0 \quad (5.71a)$$

$$\nu_c^k \tilde{\omega}_j^k \left[\langle a^j, d^k \rangle - \sigma_D^k - b_j^k \right] = 0, \quad j \in \hat{J}_c^k. \quad (5.71b)$$

En additionnant entre elles les expressions (5.69), puis les expressions (5.70), puis enfin les expressions (5.71), et en utilisant les formules de mise-à-jour (5.65) et les propriétés (5.52a), (5.53c) et (5.53f), on obtient:

$$\begin{aligned} \langle p_{fi}^k, d^k \rangle - v_i^k + \tilde{f}_{pi}^k &= 0, \quad i \in \{1, \dots, r\} \\ \nu_{ci}^k \left[\langle p_{ci}^k, d^k \rangle - \sigma_D^k - \tilde{b}_{pi}^k \right] &= 0, \quad i \in \{1, \dots, r\} \\ \nu_c^k \left[\langle p_c^k, d^k \rangle - \sigma_D^k - \tilde{b}_p^k \right] &= 0. \end{aligned}$$

Additionnant ces trois derniers groupes d'équations aux expressions (5.68), on trouve en utilisant l'expression (5.55) du lemme 5.9.6:

$$\sigma_f^k + \sigma_D^k = \langle p^k, d^k \rangle + \sum_{i=1}^r \left(\tilde{f}_{pi}^k - f_i(x^k) - \nu_{ci}^k \tilde{b}_{pi}^k \right) - \nu_c^k \tilde{b}_p^k.$$

■

La stratégie d'agrégation ne requiert que les $2r + 1$ contraintes agrégées du problème (5.64) ainsi qu'au moins une des r ou $r + 1$ nouvelles contraintes ajoutées à l'itération courante au point y^{k+1} . Il ne faut toutefois pas s'attendre à une convergence rapide si on fait cela.

Il est recommandé de conserver le plus grand nombre possible de contraintes non-agrégées afin de profiter des bénéfices d'une bonne approximation du problème.

La stratégie d'agrégation est attribuée à Kiwiel (1983) qui l'a d'abord proposée pour le cas non contraint. Il l'a par la suite adaptée au cas contraint (Kiwiel, 1985a, 1985b) et utilisé sous plusieurs approches de résolution de problèmes de programmation convexe et non-convexe. Schramm et Zowe (1992) ont combiné l'agrégation à l'approche des pénalisations pour obtenir une méthode performante de minimisation sans contrainte de fonctions convexes.

Nous verrons au prochain chapitre comment les idées de ce chapitre peuvent être utilisées dans un algorithme de plans sécants stabilisés utilisant l'approche des pénalisations dans le but de résoudre le problème (5.1).

CHAPITRE 6

Un algorithme de descente pour la minimisation d'une somme de fonctions convexes

6.1 Introduction

Nous présentons dans ce chapitre une méthode de descente pour résoudre le problème suivant:

$$\underset{x \in C \cap \text{dom } f}{\text{minimiser}} \quad f(x) = \sum_{i=1}^r f_i(x) \quad (6.1)$$

où chaque fonction $f_i : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, $i \in \{1, \dots, r\}$, est polyédrique convexe, $\text{dom } f \triangleq \{x \in \mathbb{R}^n : f(x) < +\infty\} = \bigcap_{i=1}^r \text{dom } f_i$, et C est un ensemble polyédrique convexe. Nous supposons que l'ensemble convexe fermé $D \triangleq C \cap \text{dom } f$ possède un intérieur non vide. Afin de pouvoir éventuellement réutiliser une partie des résultats de ce chapitre, nous traiterons le problème (6.1) comme si chaque fonction f_i , $i \in \{1, \dots, r\}$, était convexe propre et fermée (i.e. semi-continue inférieure), $\text{dom } f_i$ est fermé $\forall i \in \{1, \dots, r\}$, et C est un ensemble convexe fermé. Il faudra aussi supposer que les fonctions f_i , $i \in \{1, \dots, r\}$, possèdent un sous-différentiel non vide en tout point de leur domaine.

Il est courant, en optimisation non-différentiable, de faire les hypothèses suivantes:

- (i) les fonctions sont à valeurs finies dans $\mathbb{R}^n$, et

- (ii) il existe un oracle, ou une boîte noire, acceptant en entrée un point $x \in \mathbb{R}^n$ et produisant à sa sortie la valeur de la fonction en ce point et un sous-gradient quelconque de la fonction en ce point.

L'existence de cet oracle est fondamentale. Cette hypothèse est basée sur l'observation que l'on obtient souvent comme sous-produit de l'évaluation d'une fonction en un point un sous-gradient de celle-ci en ce même point. Les méthodes de faisceaux sont conçues pour exploiter cette hypothèse: elles engendrent des hyperplans d'appui à la fonction à l'aide de l'inégalité du sous-gradient et des sous-gradients obtenus.

Ces hypothèses ne sont toutefois pas adaptées à notre problème. En effet, les fonctions f_i , $i \in \{1, \dots, r\}$, ne sont pas définies partout dans $\mathbb{R}^n$ (elles prennent donc la valeur $+\infty$ à l'extérieur de leur domaine) et leurs sous-différentiels sont vides à l'extérieur de leurs domaines (théorème B.5). De plus, contrairement à l'hypothèse habituelle, nous travaillons avec un ensemble convexe C général, i.e. non nécessairement défini avec des fonctions.

Comment, dans ces conditions, engendrer des plans sécants permettant de séparer un point non réalisable de C ou du domaine d'une fonction f_i ?

Il faudra supposer l'existence d'oracles plus complexes qui, lorsqu'ils essaient d'évaluer une fonction f_i en un point $x \in \mathbb{R}^n$, produisent un hyperplan de séparation entre x et $\text{dom } f_i$ lorsque $x \notin \text{dom } f_i$, ou un hyperplan d'appui à l'épigraphe de la fonction f_i lorsque $x \in \text{dom } f_i$. Il faudra aussi un oracle produisant un hyperplan de séparation entre x et C lorsque $x \notin C$.

En fait, nous exigerons des oracles qu'ils ne produisent pas des hyperplans quelconques mais plutôt des σ -hyperplans à l'aide de σ -demi-espaces fermés.

Définition. Soit un ensemble convexe $F \subset \mathbb{R}^n$, $x \in \mathbb{R}^n \setminus F$ et $\sigma \geq 0$. On dit que $(a, b) \in \mathbb{R}^n \times \mathbb{R}$, définit un σ -demi-espace fermé séparant x de F si

$$F \subseteq \{y \in \mathbb{R}^n : \langle a, y \rangle - \sigma \leq b\} \not\ni x.$$

Nous verrons un peu plus loin que tout hyperplan de séparation peut être défini à l'aide d'un σ -demi-espace fermé.

Nous remplacerons donc les hypothèses habituelles de l'optimisation non-différentiable convexe par celle de l'existence des $r + 1$ oracles suivants:

- Pour chaque fonction f_i , $i \in \{1, \dots, r\}$, il existe un oracle qui, en essayant d'évaluer la fonction en un point $x \in \mathbb{R}^n$, produit aussi

un hyperplan d'appui à l'épigraphe de la fonction si $x \in \text{dom } f_i$,

ou

un σ -hyperplan de séparation entre x et $\text{dom } f_i$ si $x \notin \text{dom } f_i$.

- Il existe un oracle qui produit un σ -hyperplan de séparation entre x et C lorsque $x \notin C$.

Contrairement au chapitre précédent où nous avons construit des sous-problèmes de recherche de direction en utilisant une fonction $h : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ générale,

nous nous restreindrons dans ce chapitre au cas où elle est positivement homogène d'ordre $q > 1$, i.e.

$$h(x) = ax^q, \quad \forall x \in \mathbb{R}_+$$

où $a > 0$.

De plus, puisque cette fonction apparaît dans l'expression de la fonction économique du sous-problème multipliée par une constante $\Psi_k > 0$, nous supposons que la constante a est incluse dans la constante Ψ_k et nous ne considérerons plus que la fonction suivante:

$$h(x) = x^q, \quad \forall x \in \mathbb{R}_+.$$

Pour une raison dont la justification sera donnée plus loin, nous exigerons que le problème de recherche d'une direction d'amélioration comporte toujours les contraintes engendrées au plus récent point y^{k+1} réalisable. Ainsi, à l'itération k , on aura

$$k' \in J_{fi}^k, \quad \forall i \in \{1, \dots, r\}$$

tel que

$$k' \leq k$$

$$y^{k'} \in D$$

et

$$y^\ell \notin D, \quad \forall \ell \in \{k' + 1, \dots, k\} \text{ si } k' < k.$$

Nous commencerons par l'étude d'un algorithme basé sur la stratégie d'agrégation. Nous verrons plus loin comment le modifier afin d'utiliser la stratégie d'accumulation/épuration (i.e. de sélection).

6.2 L'algorithme avec la stratégie d'agrégation

ALGORITHME

(0) (INITIALISATION)

Choisir une solution initiale réalisable $x^1 \in D \triangleq C \cap \text{dom } f$, un poids initial $\Psi_1 > 0$, une borne inférieure $\Psi_{\min} > 0$ pour les poids, une constante d'arrêt $\varepsilon_s \geq 0$ et un coefficient de descente $m \in (0, 1)$. Poser $y^1 = x^1$ et

$$p_{fi}^0 = g_i^1 \in \partial f_i(x^1), f_{pi}^1 = f_{i1}^1 = f_i(x^1), J_{fi}^1 = \{1\}, i \in \{1, \dots, r\} \quad (6.2a)$$

$$J_{Ci}^1 = \emptyset, i \in \{1, \dots, r\} \quad (6.2b)$$

$$J_c^1 = \emptyset \quad (6.2c)$$

Fixer la valeur de l'estimé de la variation de la fonction économique à $+\infty$, i.e. $\varepsilon_v^1 = +\infty$.

Poser $k = k' = 1$ et $K = \emptyset$.

(1) (RECHERCHE D'UNE DIRECTION)

Évaluer les multiplicateurs (π^k, μ^k, ω^k) solution du problème dual (6.3) ci-dessous:

$$\begin{aligned} \text{maximiser}_{\pi, \mu, \omega} \quad & \sum_{i=1}^r \left(\pi_{pi} [f_{pi}^k - f_i(x^k)] + \sum_{j \in J_{fi}^k} \pi_{ij} [f_{ij}^k - f_i(x^k)] \right) \\ & - \sum_{i=1}^r \left(\mu_{pi} b_{pi}^k + \sum_{j \in J_{ci}^k} \mu_{ij} b_{ij}^k \right) - \left(\omega_p b_p^k + \sum_{j \in J_c^k} \omega_j b_j^k \right) \\ & + \frac{1}{(q \Psi_k)^{\frac{1}{q-1}}} \left(\frac{1}{q} - 1 \right) (\|p\|^*)^{\frac{q}{q-1}} \end{aligned}$$

sous les contraintes

$$\begin{aligned} \pi_{pi} + \sum_{j \in J_{fi}^k} \pi_{ij} &= 1, \quad i \in \{1, \dots, r\} \\ \sum_{i=1}^r \left(\mu_{pi} + \sum_{j \in J_{fi}^k} \mu_{ij} \right) + \left(\omega_p + \sum_{j \in J_c^k} \omega_j \right) &\leq 1 \\ p - \sum_{i=1}^r \left(\pi_{pi} p_{fi}^{k-1} + \sum_{j \in J_{fi}^k} \pi_{ij} g_i^j \right) \\ &- \sum_{i=1}^r \left(\mu_{pi} p_{ci}^{k-1} + \sum_{j \in J_{ci}^k} \mu_{ij} a_i^j \right) - \left(\omega_p p_c^{k-1} + \sum_{j \in J_c^k} \omega_j a^j \right) = 0 \\ \pi_{pi} \geq 0, \mu_{pi} \geq 0, \quad i \in \{1, \dots, r\} \quad (\mu_{pi} = 0 \text{ si } J_{ci}^k = \emptyset) \\ \omega_p &\geq 0 \\ \pi_{ij} \geq 0, \quad j \in J_{fi}^k, \quad i \in \{1, \dots, r\} \\ \mu_{ij} \geq 0, \quad j \in J_{ci}^k, \quad i \in \{1, \dots, r\} \\ \omega_j &\geq 0, \quad j \in J_c^k. \end{aligned} \tag{6.3}$$

Évaluer des multiplicateurs $\nu_{ci}^k, i \in \{1, \dots, r\}, \nu_c^k$ et $(\tilde{\mu}^k, \tilde{\omega}^k)$ vérifiant les expressions (5.51a)–(5.51c) et (5.53a)–(5.53f). Évaluer les quantités

$$\begin{aligned} (p_{fi}^k, \tilde{f}_{pi}^k), \quad i \in \{1, \dots, r\} \\ (p_{ci}^k, \tilde{b}_{pi}^k), \quad i \in \{1, \dots, r\} \end{aligned}$$

et

$$(p_c^k, \tilde{b}_p^k)$$

avec les expressions (5.65a)–(5.65c).

Évaluer p^k, σ^k et $\tilde{\alpha}^k$ avec les expressions (5.55)–(5.57) et utiliser (5.40l) pour obtenir d^k , i.e.

$$d^k \in \text{zéro} \left[\Psi_k q \|d\|^{q-1} \partial \|\cdot\| (d) + p^k \right] \tag{5.40l}$$

(2) (TEST D'ARRÊT)

Poser

$$w^k = \sigma^k + \Psi_k \|d^k\|^q. \quad (6.4)$$

Si $-w^k \leq \varepsilon_s$, on arrête.

(3) (RECHERCHE LINÉAIRE: PAS NUL)

Poser

$$y^{k+1} = x^k + d^k.$$

Si $y^{k+1} \notin D$, poser $\delta^k = 0$ (pas nul), $\alpha(x^k, y^{k+1}) = +\infty$ et aller à l'étape 6.

(4) (ÉVALUATION DE LA QUALITÉ DU MODÈLE DE LA FONCTION ÉCONOMIQUE)

Si l'inégalité

$$f(y^{k+1}) \leq f(x^k) + m\sigma^k \quad (6.5)$$

n'est pas vérifiée, alors poser

$$\delta^k = 0 \text{ (pas nul),}$$

$$\alpha(x^k, y^{k+1}) \triangleq f(x^k) - \underline{f}(x^k|y^{k+1}) \geq 0$$

où

$$\begin{aligned} \underline{f}(x^k|y^{k+1}) &= f(y^{k+1}) + \langle g^{k+1}, x^k - y^{k+1} \rangle \\ &= f(y^{k+1}) - \langle g^{k+1}, d^k \rangle \end{aligned}$$

$$g^{k+1} = \sum_{i=1}^r g_i^{k+1} \in \partial f(y^{k+1})$$

et

$$g_i^{k+1} \in \partial f_i(y^{k+1}), \quad i \in \{1, \dots, r\},$$

puis aller à l'étape 6.

(5) (PAS DE DESCENTE)

Poser $\delta^k = 1$ (pas de descente) et $K \leftarrow K \cup \{k\}$.

(6) (GESTION DU FAISCEAU)

Choisir

$$J_{fi}^{k+1} \subseteq J_{fi}^k \text{ tel que } k' \in J_{fi}^{k+1} \text{ et } J_{ci}^{k+1} \subseteq J_{ci}^k, \forall i \in \{1, \dots, r\}$$

et

$$J_c^{k+1} \subseteq J_c^k.$$

Faire $k' \leftarrow k + 1$ si $y^{k+1} \in D$.

(7) (MISE-À-JOUR DES LINÉARISATIONS)

Poser

$$x^{k+1} = x^k + \delta^k d^k.$$

Pour tout $i \in \{1, \dots, r\}$,

$$f_{pi}^{k+1} = \tilde{f}_{pi}^k + \delta^k \langle p_{fi}^k, d^k \rangle \quad (5.62 \text{ a})$$

$$b_{pi}^{k+1} = \tilde{b}_{pi}^k - \delta^k \langle p_{ci}^k, d^k \rangle \quad (5.62 \text{ b})$$

Si $y^{k+1} \in \text{dom } f_i$

$$\text{Poser } f_{ij}^{k+1} = f_{ij}^k + \delta^k \langle g_i^j, d^k \rangle, \quad \forall j \in J_{fi}^{k+1}$$

Trouver $g_i^{k+1} \in \partial f_i(y^{k+1})$ et évaluer

$$f_{i,k+1}^{k+1} = f_i(y^{k+1}) - \delta^k \langle g_i^{k+1}, d^k \rangle.$$

$$\text{Poser } J_{fi}^{k+1} \leftarrow J_{fi}^{k+1} \cup \{k+1\}.$$

$$\text{Poser } b_{ij}^{k+1} = b_{ij}^k - \delta^k \langle a_i^k, d^k \rangle, \quad \forall j \in J_{ci}^{k+1}.$$

Si $y^{k+1} \notin \text{dom } f_i$

$$\text{Poser } b_{ij}^{k+1} = b_{ij}^k, \quad \forall j \in J_{ci}^{k+1}$$

$$\text{et } f_{ij}^{k+1} = f_{ij}^k, \quad \forall j \in J_{fi}^{k+1}$$

$$\text{Poser } J_{ci}^{k+1} \leftarrow J_{ci}^{k+1} \cup \{k+1\} \text{ et trouver } (a_i^{k+1}, b_i^{k+1}) \in \mathbb{R}^n \times \mathbb{R},$$

à l'aide de l'oracle, afin de définir un σ_D^k -demi-espace fermé

séparant y^{k+1} de $\text{dom } f_i$.

$$\text{Poser } b_{i,k+1}^{k+1} = b_i^{k+1} - \langle a_i^{k+1}, x^k \rangle.$$

$$\text{Poser } b_p^{k+1} = \tilde{b}_p^k - \delta^k \langle p_c^k, d^k \rangle \quad (5.62 \text{ c})$$

Si $y^{k+1} \in C$

$$\text{poser } b_j^{k+1} = b_j^k - \delta^k \langle a^j, d^k \rangle, \quad \forall j \in J_c^{k+1},$$

sinon

$$\text{poser } b_j^{k+1} = b_j^k, \quad \forall j \in J_c^{k+1}$$

$$\text{poser } J_c^{k+1} \leftarrow J_c^{k+1} \cup \{k+1\}$$

Trouver $(a^{k+1}, b^{k+1}) \in \mathbb{R}^n \times \mathbb{R}$, à l'aide de l'oracle, afin de définir un σ_D^k -demi-espace fermé contenant C , mais pas y^{k+1} .

$$\text{Poser } b_{k+1}^{k+1} = b^{k+1} - \langle a^{k+1}, x^k \rangle.$$

(8) (MISE-À-JOUR DES POIDS)

Si $\delta^k = 1$ (pas de descente), alors choisir $\Psi_{k+1} \in [\Psi_{\min}, \Psi_k]$ et poser $\varepsilon_v^{k+1} = \varepsilon_v^k$.

Sinon, i.e. si $\delta^k = 0$ (pas nul), poser

$$\varepsilon_v^{k+1} = \min\{\varepsilon_v^k, \|p^k\|^* + \tilde{\alpha}^k\} \quad (6.6)$$

Si $y^{k+1} \in D$,

poser $\Psi_{k+1} = \Psi_k$, ou choisir $\Psi_{k+1} \geq \Psi_k$ si

$$\alpha(x^k, y^{k+1}) > \varepsilon_v^{k+1}. \quad (6.7)$$

Si $(y^{k+1} \notin D)$, poser $\Psi_{k+1} = \Psi_k$.

(9) (BOUCLE) Poser $k \leftarrow k+1$ et retourner à l'étape 1.

6.3 Remarques sur l'algorithme

6.3.1 L'idée générale de la procédure de mise-à-jour des poids de l'étape 8 de l'algorithme est de permettre un accroissement de la taille de la "région de confiance"

autour du centre de stabilité (en réduisant la valeur du poids) lorsque l'algorithme effectue des pas de descente. Inversement, on réduit la taille de la "région de confiance" (en accroissant la valeur du poids) lorsque des pas nuls réalisables sont effectués. Lorsque cette dernière situation se produit et que le candidat y^{k+1} est réalisable, deux cas se présentent:

- (i) L'erreur en x^k (donnée par $\alpha(x^k, y^{k+1})$) de la plus récente linéarisation, en y^{k+1} , de la fonction d'amélioration n'est pas très grande et on ne change pas la valeur du poids.
- (ii) L'erreur en x^k de la plus récente linéarisation est trop grande et on permet une réduction de la taille de la "région de confiance" par accroissement du poids.

C'est le paramètre ε_v^{k+1} qui sert à déterminer si l'erreur de linéarisation est tolérable ou non. Sa mise-à-jour, lorsqu'il y a pas nul, i.e.

$$\varepsilon_v^{k+1} = \min\{\varepsilon_v^k, \|p^k\|^* + \tilde{\alpha}^k\} \quad (6.6)$$

permettra de démontrer que la méthode est convergente lorsque l'algorithme effectue un nombre fini de pas de descente suivis d'une infinité de pas nuls dont on peut extraire une infinité de candidats y^{k+1} réalisables. Le test (6.7) servira à s'assurer que les poids Ψ_k n'augmentent pas trop rapidement.

On ne change pas la valeur du poids Ψ_k si un pas nul non-réalisable vient d'être effectué car dans le cas de fonctions et ensembles polyédriques, l'algorithme finira par nous ramener dans le domaine réalisable en un nombre fini d'étapes. Un traitement plus élaboré est cependant nécessaire pour le cas non-polyédrique. Nous en reparlerons plus loin.

6.3.2 (Pertinence de la variation des poids) L'algorithme ici présenté permet la variation des poids Ψ_k d'une itération à une autre. Ceci est dû au fait que la stratégie consistant à les garder constants est dangereuse. En effet, les poids risquent d'être souvent trop élevés ou trop faibles.

Lorsque le poids Ψ_k prend une valeur très grande, le sous-problème (5.64) (ou le problème équivalent (5.66a)) produit des solutions (σ^k, d^k) où $|\sigma^k|$ et $\|d^k\|$ prennent de faibles valeurs. Les pas de descente sont alors nombreux et très courts, et la convergence est lente. L'algorithme tend alors à se comporter comme une méthode de plus forte pente, approche réputée pour sa convergence problématique, même en optimisation convexe différentiable (Zangwill, 1969).

Plusieurs exemples démontrant la non-convergence de la méthode de plus forte pente en optimisation convexe non-différentiable ont été proposés, dont ceux de Feuer (1974), Dem'yanov et Malozemov (1974), Wolfe (1974, 1975), Zowe (1985) et Lemaréchal (1989).

Pour cette raison, l'algorithme peut faire décroître la valeur du poids lorsqu'un pas de descente vient d'être effectué: on suppose alors que la qualité du modèle est bonne, ce qui justifie un accroissement de la taille de la "région de confiance" par l'intermédiaire d'une réduction du poids Ψ_k .

D'autre part, une faible valeur du poids Ψ_k provoque la production de solutions (σ^k, d^k) où $|\sigma^k|$ et $\|d^k\|$ prennent des valeurs élevées. L'algorithme aura tendance à faire succéder chaque pas de descente par plusieurs pas nuls. Un pas nul étant une indication de la faible qualité du modèle associé au problème, l'algorithme autorise une réduction de la taille de la "région de confiance" dans un tel cas. Ceci se fait

par accroissement de la valeur du paramètre Ψ_k .

Nous avons volontairement omis de donner des règles précises concernant la façon de mettre les poids à jour car c'est encore un domaine ouvert de recherche. On ne sait que fort peu de choses à ce sujet et il reste encore beaucoup de travaux empiriques et théoriques à effectuer avant d'être plus catégoriques.

Kiwiel (1990) ainsi que Schramm et Zowe (1992) ont conçu des procédures plus précises et détaillées que la nôtre à ce sujet. Hiriart-Urruty et Lemaréchal (1993b) ont proposés des conditions générales originales pour assurer la convergence d'un algorithme inspiré de celui de Kiwiel (1990). Ils présentent aussi une discussion plus étoffée des inconvénients occasionnés par des valeurs extrêmes des poids.

6.3.3 L'exigence de production, à l'étape 7, de σ_D^k -demi-espaces fermés par les oracles, n'en est pas vraiment une. En fait, tout hyperplan séparant y^{k+1} de C ou de $\text{dom } f_i$ peut servir à produire un σ_D^k -demi-espace fermé. En effet, si $y^{k+1} \notin C$ et si (a^{k+1}, b^{k+1}) sépare y^{k+1} de C , alors

$$\langle a^{k+1}, x \rangle \leq b^{k+1}, \quad \forall x \in C$$

et

$$\langle a^{k+1}, y^{k+1} \rangle > b^{k+1}.$$

Si

$$\langle a^{k+1}, y^{k+1} \rangle - \sigma_D^k > b^{k+1}$$

alors (a^{k+1}, b^{k+1}) définit un σ_D^k -demi-espace fermé. Par contre, si

$$\langle a^{k+1}, y^{k+1} \rangle - \sigma_D^k \leq b^{k+1}$$

alors

$$\eta \langle a^{k+1}, y^{k+1} \rangle - \sigma_D^k > \eta b^{k+1}, \quad \forall \eta > M,$$

où

$$M = \sigma_D^k / (\langle a^{k+1}, y^{k+1} \rangle - b^{k+1}) > 0$$

et $\eta(a^{k+1}, b^{k+1})$ défini un σ_D^k -demi-espace-fermé. On procède de la même manière avec les ensembles $\text{dom } f_i$.

6.3.4 L'erreur en x^k de la plus récente linéarisation, en y^{k+1} , de la fonction d'amélioration, désignée par $\alpha(x^k, y^{k+1})$, ne prend une valeur finie que s'il y a pas de descente à l'itération k (i.e. $k \in K$). On pourrait définir $\alpha'(x^k, y^{k+1})$ en généralisant la définition de $\alpha(x^k, y^{k+1})$ de manière à ce que cette quantité prenne une valeur finie à toutes les itérations:

$$0 \leq \alpha'(x^k, y^{k+1}) \triangleq \sum_{i=1}^r (\alpha_{f_i}(x^k, y^{k+1}) + \alpha_{C_i}(x^k, y^{k+1})) + \alpha_C(x^k, y^{k+1})$$

où

$$\alpha_{f_i}(x^k, y^{k+1}) = \begin{cases} f_i(x^k) - f_i(y^{k+1}) - \langle g_i^{k+1}, x^k - y^{k+1} \rangle & \text{si } y^{k+1} \in \text{dom } f_i \\ 0 & \text{sinon,} \end{cases}$$

$$\alpha_{C_i}(x^k, y^{k+1}) = \begin{cases} 0 & \text{si } y^{k+1} \in \text{dom } f_i \\ b_i^{k+1} - \langle a_i^{k+1}, x^k \rangle & \text{sinon} \end{cases}$$

et

$$\alpha_C(x^k, y^{k+1}) = \begin{cases} 0 & \text{si } y^{k+1} \in C \\ b^{k+1} - \langle a^{k+1}, x^k \rangle & \text{sinon.} \end{cases}$$

Remarquons que nous obtenons l'expression de l'étape 4 lorsque $k \in K$. La démonstration de la convergence de la méthode n'a été faite qu'avec la définition originale des étapes 3 et 4 de l'algorithme.

6.4 Résultats préliminaires

Lemme 6.4.1 *Supposons que l'algorithme ne se soit pas arrêté avant la $k^{\text{ème}}$ itération.*

Alors

$$(p_{fi}^k, \tilde{f}_{pi}^k) \in \text{conv} \{(g_i^j, f_{ij}^k) : j \in J_{fi}^1 \cup \dots \cup J_{fi}^k\}, i \in \{1, \dots, r\} \quad (6.8a)$$

$$(p_{ci}^k, \tilde{b}_{pi}^k) \in \text{conv} \{(a_i^j, b_{ij}^k) : j \in J_{ci}^1 \cup \dots \cup J_{ci}^k, \text{ si } J_{ci}^k \neq \emptyset, i \in \{1, \dots, r\}\} \quad (6.8b)$$

$$(p_c^k, \tilde{b}_p^k) \in \text{conv} \{(a^j, b_j^k) : j \in J_c^1 \cup \dots \cup J_c^k\}, \text{ si } J_c^k \neq \emptyset. \quad (6.8c)$$

De plus, si $k > 1$

$$(p_{fi}^{k-1}, f_{pi}^k) \in \text{conv} \{(g_i^j, f_{ij}^k) : j \in J_{fi}^1 \cup \dots \cup J_{fi}^{k-1}\}, i \in \{1, \dots, r\} \quad (6.9a)$$

$$(p_{ci}^{k-1}, b_{pi}^k) \in \text{conv} \{(a_i^j, b_{ij}^k) : j \in J_{ci}^1 \cup \dots \cup J_{ci}^{k-1}\} \\ \text{si } J_{ci}^{k-1} \neq \emptyset, i \in \{1, \dots, r\} \quad (6.9b)$$

$$(p_c^{k-1}, b_p^k) \in \text{conv} \{(a^j, b_j^k) : j \in J_c^1 \cup \dots \cup J_c^{k-1}\} \text{ si } J_c^{k-1} \neq \emptyset. \quad (6.9c)$$

Démonstration:

Nous procéderons par induction. Soit

$$J_i^k \triangleq J_{fi}^1 \cup \dots \cup J_{fi}^k = \{j_1, \dots, j_\ell\} \subseteq \{1, \dots, k\}$$

l'ensemble des numéros d'itérations, parmi les k premières, où un sous-gradient $g_i^j \in \partial f_i(j^j)$ a été engendré. Remarquons que

$$J_i^1 = J_{fi}^1 = \{1\}, \quad \forall i \in \{1, \dots, r\}$$

et que

$$J_i^k = J_i^{k-1} \cup \{k\} \quad \text{si } J_i^k \neq J_i^{k-1}.$$

Soit une solution duale optimale (π^k, μ^k, ω^k) associée au problème (6.3) de l'itération k . Nous procéderons en complétant le vecteur $(\pi^k, \tilde{\mu}^k, \tilde{\omega}^k)$ défini en (5.51a)–(5.51c) et (5.53a)–(5.53f) puis en construisant un vecteur de poids $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k)$ à l'aide de $(\pi^k, \tilde{\mu}^k, \tilde{\omega}^k)$. Ces poids seront des combinaisons convexes qui rendent vrais les énoncés du lemme. Posons

$$\begin{aligned} \bar{\pi}_{i1}^1 &= 1 \\ \pi_{ij}^k &= 0 \quad \text{si } j \in J_i^k \setminus J_{fi}^k, \quad k > 1 \\ \bar{\pi}_{ij}^k &= \pi_{ij}^k + \pi_{pi}^k \bar{\pi}_{ij}^{k-1} \quad \text{pour } j \in J_i^{k-1}, \quad k > 1 \\ \bar{\pi}_{ik}^k &= \pi_{ik}^k \quad \text{si } k \in J_i^k \end{aligned}$$

Si $k = 1$, alors

$$\begin{aligned} (p_{fi}^k, \tilde{f}_{pi}^k) &= \pi_{pi}^1(p_{fi}^0, f_{pi}^1) + \pi_{i1}^1(g_i^1, f_{i1}^1) \quad \text{par (5.65a)} \\ &= \pi_{pi}^1(g_i^1, f_{i1}^1) + \pi_{i1}^1(g_i^1, f_{i1}^1) \quad \text{par (6.2a)} \\ &= (g_i^1, f_{i1}^1) \quad \text{par (5.51a) et (5.52a)} \end{aligned}$$

et l'expression (6.8a) est vraie. De plus

$$\begin{aligned} f_{pi}^{k+1} &= f_{pi}^2 = \tilde{f}_{pi}^1 + \langle p_{fi}^1, x^2 - x^1 \rangle \quad \text{par (5.62a)} \\ &= f_{i1}^1 + \langle g_i^1, x^2 - x^1 \rangle \quad \text{par (6.8a)} \\ &= f_{i1}^2 \quad \text{par (5.17a) et (5.43)} \end{aligned}$$

ce qui vérifie l'expression (6.9a) pour $k = 2$.

Supposons maintenant que l'expression (6.9a) soit vérifié pour un $k > 2$. Alors

$$\begin{aligned}
 (p_{fi}^k, \tilde{f}_{pi}^k) &= \pi_{pi}^k(p_{fi}^{k-1}, f_{pi}^k) + \sum_{j \in J_{fi}^k} \pi_{ij}^k(g_i^j, f_{ij}^k) && \text{par (5.65a)} \\
 &= \pi_{pi}^k(p_{fi}^{k-1}, f_{pi}^k) + \sum_{j \in J_{fi}^k} \pi_{ij}^k(g_i^j, f_{ij}^k) && \text{car } \pi_{ij}^k = 0, \forall j \in J_i^k \setminus J_{fi}^k \\
 &= \pi_{pi}^k \sum_{j \in J_i^{k-1}} \bar{\pi}_{ij}^{k-1}(g_i^j, f_{ij}^k) + \sum_{j \in J_i^k} \pi_{ij}^k(g_i^j, f_{ij}^k) && \text{par (6.9a)} \\
 &= \begin{cases} \sum_{j \in J_i^{k-1}} (\pi_{pi}^k \bar{\pi}_{ij}^{k-1} + \pi_{ij}^k)(g_i^j, f_{ij}^k) & \text{si } J_i^{k-1} = J_i^k \\ \pi_{ik}^k(g_i^k, f_{ik}^k) + \sum_{j \in J_i^{k-1}} (\pi_{pi}^k \bar{\pi}_{ij}^{k-1} + \pi_{ij}^k)(g_i^j, f_{ij}^k) & \text{sinon} \end{cases} \\
 &= \begin{cases} \sum_{j \in J_i^{k-1}} \bar{\pi}_{ij}^k(g_i^j, f_{ij}^k) & \text{si } J_i^{k-1} = J_i^k \\ \sum_{j \in J_i^k} \bar{\pi}_{ij}^k(g_i^j, f_{ij}^k) & \text{sinon} \end{cases} \\
 &= \sum_{j \in J_i^k} \bar{\pi}_{ij}^k(g_i^j, f_{ij}^k).
 \end{aligned}$$

De plus, lorsque $k \in J_i^k$, on a

$$\begin{aligned}
 \sum_{j \in J_i^k} \bar{\pi}_{ij}^k &= \bar{\pi}_{ik}^k + \sum_{j \in J_i^{k-1}} \bar{\pi}_{ij}^k = \pi_{ik}^k + \sum_{j \in J_i^{k-1}} (\pi_{ij}^k + \pi_{pi}^k \bar{\pi}_{ij}^{k-1}) \\
 &= \sum_{j \in J_i^k} \pi_{ij}^k + \pi_{pi}^k \sum_{j \in J_i^{k-1}} \bar{\pi}_{ij}^{k-1} \\
 &= \sum_{j \in J_i^k} \pi_{ij}^k + \pi_{pi}^k && \text{par (6.9a)} \\
 &= 1 && \text{par (5.51a) et (5.52a)}
 \end{aligned}$$

Ce dernier résultat est encore vrai lorsque $k \notin J_i^k$, de sorte que l'on a vérifié la relation (6.8a) pour k . Enfin, on a

$$\begin{aligned}
 f_{pi}^{k+1} &= \tilde{f}_{pi}^k + \langle p_{fi}^k, x^{k+1} - x^k \rangle && \text{par (5.62a)} \\
 &= \sum_{j \in J_i^k} \bar{\pi}_{ij}^k f_{ij}^k + \langle \sum_{j \in J_i^k} \bar{\pi}_{ij}^k g_i^j, x^{k+1} - x^k \rangle && \text{par (6.8a)} \\
 &= \sum_{j \in J_i^k} \bar{\pi}_{ij}^k (f_{ij}^k + \langle g_i^j, x^{k+1} - x^k \rangle) \\
 &= \sum_{j \in J_i^k} \bar{\pi}_{ij}^k f_{ij}^{k+1} && \text{par (5.17a) et (5.43) .}
 \end{aligned}$$

ce qui vérifie l'expression (6.9a) pour $k + 1$ et qui termine un cycle d'induction.

Les autres expressions se démontrent de la même manière. Il faut toutefois s'assurer que les ensembles J_{α}^k et J_c^k ne sont pas vides (ils sont vides au début des calculs). ■

L'analyse qui suit est basée sur le fait que les sous-gradients $g_i^j \in \partial f_i(y^j)$ sont des ε -sous-gradients de f_i au centre de stabilité courant x^k .

Supposons que l'algorithme ne se soit pas arrêté avant l'itération k et définissons les **erreurs de linéarisation**.

$$\alpha_{f_i,j}^k = f_i(x^k) - f_{ij}^k, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \quad (6.10a)$$

$$\alpha_{f_i,p}^k = f_i(x^k) - f_{pi}^k, \quad i \in \{1, \dots, r\} \quad (6.10b)$$

$$\tilde{\alpha}_{f_i,p}^k = f_i(x^k) - \tilde{f}_{pi}^k, \quad i \in \{1, \dots, r\} \quad (6.10c)$$

$$\alpha_{f,p}^k = \sum_{i=1}^r \alpha_{f_i,p}^k \quad (6.10d)$$

$$\tilde{\alpha}_{f,p}^k = \sum_{i=1}^r \tilde{\alpha}_{f_i,p}^k \quad (6.10e)$$

Le résultat suivant est une adaptation du lemme 5.4.2 de Kiwiel (1985).

Lemme 6.4.2 *À la $k^{\text{ème}}$ itération de l'algorithme, on a*

$$g_i^j \in \partial_\varepsilon f_i(x^k), \quad \text{où} \quad \varepsilon = \alpha_{f_i,j}^k \geq 0, \quad j \in J_{f_i}^k, \quad i \in \{1, \dots, r\} \quad (6.11a)$$

$$p_{f_i}^{k-1} \in \partial_\varepsilon f_i(x^k), \quad \text{où} \quad \varepsilon = \alpha_{f_i,p}^k \geq 0, \quad i \in \{1, \dots, r\} \quad (6.11b)$$

$$p_{f_i}^k \in \partial_\varepsilon f_i(x^k), \quad \text{où} \quad \varepsilon = \tilde{\alpha}_{f_i,p}^k \geq 0, \quad i \in \{1, \dots, r\} \quad (6.11c)$$

$$p_f^{k-1} = \sum_{i=1}^r p_{f_i}^{k-1} \in \partial_\varepsilon f(x^k), \quad \text{où} \quad \varepsilon = \alpha_{fp}^k \geq 0 \quad (6.11d)$$

$$p_f^k = \sum_{i=1}^r p_{f_i}^k \in \partial_\varepsilon f(x^k), \quad \text{où} \quad \varepsilon = \tilde{\alpha}_{fp}^k \geq 0 \quad (6.11e)$$

Le prochain résultat affirme que les hyperplans de séparation entre un point $x \notin F$ et un ensemble convexe fermé F permettent de définir des ε -sous-gradients de la fonction indicatrice I_F en des points $x^k \in F$. Il fait appel au concept d'ensemble ε -normal présenté à l'annexe B. Il rend possible l'extension du lemme 6.4.1.

Proposition 6.4.3 *Si, à l'itération k*

(i) *l'oracle associé à l'ensemble convexe fermé C engendre une paire $(a^{k+1}, b^{k+1}) \in \mathbb{R}^n \times \mathbb{R}$, alors*

$$a^{k+1} \in \partial_\varepsilon I_c(x^k) = N_\varepsilon(C; x^k) \quad (6.12a)$$

où

$$\varepsilon = b^{k+1} - \langle a^{k+1}, x^k \rangle = b_{k+1}^{k+1} \geq 0. \quad (6.12b)$$

(ii) *l'oracle associé à l'une des fonctions f_i , $i \in \{1, \dots, r\}$, engendre une paire $(a_i^{k+1}, b_i^{k+1}) \in \mathbb{R}^n \times \mathbb{R}$, alors*

$$a_i^{k+1} \in \partial_\varepsilon I_{\text{dom } f_i}(x^k) = N_\varepsilon(\text{dom } f_i; x^k) \quad (6.13a)$$

où

$$\varepsilon = b_i^{k+1} - \langle a_i^{k+1}, x^k \rangle = b_{i,k+1}^{k+1} \geq 0. \quad (6.13b)$$

Démonstration:

Nous ne ferons la démonstration que pour la première affirmation, la seconde se démontrant de la même manière. À la $k^{\text{ème}}$ itération, le centre de stabilité est $x^k \in D = C \cap \text{dom } f$ et l'algorithme engendre un point y^{k+1} . Le couple (a^{k+1}, b^{k+1}) n'est engendré que si $y^{k+1} \notin C$. Dans ce cas, on a

$$x^k \in C \subseteq H_{a^{k+1}, b^{k+1}} = \{z \in \mathbb{R}^n : \langle a^{k+1}, z \rangle \leq b^{k+1}\} \not\ni y^{k+1}.$$

Remarquons que b^{k+1} n'est pas l'unique valeur qui fasse l'affaire. On a aussi

$$x^k \in C \subseteq H_{a^{k+1}, r} = \{z \in \mathbb{R}^n : \langle a^{k+1}, z \rangle \leq r\} \not\ni y^{k+1}, \quad \forall r \in [\sigma_c(a^{k+1}), b^{k+1}].$$

où $\sigma_c : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est la fonction d'appui de l'ensemble C définie à l'annexe B.

Par conséquent,

$$\langle a^{k+1}, x^k \rangle \leq \sigma_c(a^{k+1}) \leq b^{k+1} < \langle a^{k+1}, y^{k+1} \rangle$$

de sorte que

$$\begin{aligned} \sigma_c(a^{k+1}) \leq b^{k+1} &= \langle a^{k+1}, x^k \rangle + (b^{k+1} - \langle a^{k+1}, x^k \rangle) \\ &= \langle a^{k+1}, x^k \rangle + \varepsilon \end{aligned}$$

avec $\varepsilon = b^{k+1} - \langle a^{k+1}, x^k \rangle \geq 0$, d'où $a^{k+1} \in \partial_\varepsilon I_c(x^k)$, par le corollaire B.18. De plus,

$$\begin{aligned} \varepsilon &= b^{k+1} - \langle a^{k+1}, x^k \rangle \\ &= b^{k+1} - \langle a^{k+1}, x^{k+1} \rangle \quad \text{car } x^{k+1} = x^k \text{ puisque } y^{k+1} \notin C \\ &= b_{k+1}^{k+1} \quad \text{par (5.17c)} \\ &\leq 0 \quad \text{car } x^{k+1} \in C. \end{aligned}$$

■

Remarque: Si (a^{k+1}, b^{k+1}) définit un hyperplan séparant y^{k+1} de C , alors $\eta(a^{k+1}, b^{k+1})$ définit le même hyperplan, quel que soit la valeur du paramètre $\eta > 0$. Par conséquent,

$$\alpha^{k+1} \in \partial_\varepsilon I_c(x^k) = \partial_\varepsilon I_c(x^{k+1}), \quad \forall \eta > 0$$

où

$$\alpha^{k+1} = \eta a^{k+1}$$

et

$$\varepsilon = \eta(b^{k+1} - \langle a^{k+1}, x^k \rangle).$$

En faisant tendre η vers 0, on a

$$0 \in \partial I_c(x^k),$$

ce qui est toujours vrai.

En faisant varier le paramètre η , on peut obtenir des ε et des ε -sous-gradients aussi grands ou petits que l'on veut. Afin de rendre l'algorithme moins sensible à la "taille" des $(a^{k+1}, b^{k+1}) \in \mathbb{R}^n \times \mathbb{R}$ produits par l'oracle, on peut normaliser ces quantités en choisissant $M > 0$ et

$$\eta \leq \frac{M}{\|(a^{k+1}, b^{k+1})\|}.$$

Lemme 6.4.4 *À la $k^{\text{ème}}$ itération de l'algorithme, on a*

$$a_i^j \in \partial_\varepsilon I_{\text{dom } f_i}(x^k), \quad \text{où } \varepsilon = b_{ij}^k \geq 0, \quad \forall j \in J_{ci}^k, \quad \forall i \in \{1, \dots, r\} \quad (6.14a)$$

$$a^j \in \partial_\varepsilon I_c(x^k), \quad \text{où } \varepsilon = b_j^k \geq 0, \quad \forall j \in J_c^k, \quad \forall i \in \{1, \dots, r\} \quad (6.14b)$$

$$p_{ci}^k \in \partial_\varepsilon I_{\text{dom } f_i}(x^k), \quad \text{où } \varepsilon = \tilde{b}_{pi}^k \geq 0, \quad \forall i \in \{1, \dots, r\} \quad (6.14c)$$

$$p_c^k \in \partial_\varepsilon I_c(x^k), \quad \text{où } \varepsilon = \tilde{b}_p^k \geq 0 \quad (6.14d)$$

$$p_{ci}^{k-1} \in \partial_\varepsilon I_{\text{dom } f_i}(x^k), \quad \text{où } \varepsilon = b_{pi}^k \geq 0, \quad \forall i \in \{1, \dots, r\} \quad (6.14e)$$

$$p_c^{k-1} \in \partial_\epsilon I_c(x^k), \text{ où } \epsilon = b_p^k \geq 0 \quad (6.14f)$$

$$p_D^k = p_c^k + \sum_{i=1}^r p_{ci}^k \in \partial_\epsilon I_D(x^k), \text{ où } \epsilon = \tilde{b}_{pD}^k \triangleq \tilde{b}_p^k + \sum_{i=1}^r \tilde{b}_{pi}^k \quad (6.14g)$$

$$p_D^{k-1} \in \partial_\epsilon I_D(x^k), \text{ où } \epsilon = b_{pD}^k \triangleq b_p^k + \sum_{i=1}^r b_{pi}^k \quad (6.14h)$$

Démonstration:

Nous savons, par la proposition 6.4.3, que

$$a^j \in \partial_\epsilon I_c(x^j), \text{ où } \epsilon = b_j^j = b^j - \langle a^j, x^j \rangle, \forall j \in J_c^k.$$

Par conséquent, $\forall x \in \mathbb{R}^n$, on a

$$\begin{aligned} I_c(x) &\geq I_c(x^j) + \langle a^j, x - x^j \rangle - b_j^j \\ &= \langle a^j, x - x^j \rangle - b^j + \langle a^j, x^j \rangle && \text{car } x^j \in C \\ &= \langle a^j, x - x^k \rangle - b^j + \langle a^j, x^j \rangle + \langle a^j, x^k - x^j \rangle \\ &= \langle a^j, x - x^k \rangle - [b^j - \langle a^j, x^k \rangle] \\ &= I_c(x^k) + \langle a^j, x - x^k \rangle - b_j^k && \text{par (5.17c) et} \\ &&& \text{parce que } x^k \in C, \end{aligned}$$

et l'énoncé (6.14b) est vérifié. Utilisant cet énoncé et les poids $\bar{\omega}_j^k$ de l'énoncé (5.8c), on obtient, en posant $J_o^k \triangleq J_c^1 \cup \dots \cup J_c^k, \forall x \in \mathbb{R}^n$

$$\begin{aligned} I_c(x) &= \sum_{j \in J_o^k} \bar{\omega}_j^k I_c(x) \\ &\geq \langle \sum_{j \in J_o^k} \bar{\omega}_j^k a^j, x - x^k \rangle - \sum_{j \in J_o^k} \bar{\omega}_j^k b_j^k \\ &= \langle p_c^k, x - x^k \rangle - \tilde{b}_p^k. \end{aligned}$$

En posant $x = x^k$, on obtient $\tilde{b}_p^k \geq 0$ et l'énoncé (6.14d) est démontré.

De même, en utilisant l'énoncé (6.14b) et les poids $\bar{\omega}_j^{k-1}$ de l'énoncé (6.9c), on démontre l'énoncé (6.14f).

Les énoncés (6.14a), (6.14c) et (6.14e) se démontrent de la même manière que les énoncés (6.14b), (6.14d) et (6.14f) et nous ne répéterons pas la démonstration.

L'énoncé (6.14g) se démontre en constatant que

$$\begin{aligned} I_D(x) &= \max\{I_c(x), I_{\text{dom } f_1}(x), \dots, I_{\text{dom } f_r}(x)\} \\ &= I_C(x) + \sum_{i=1}^r I_{\text{dom } f_i}(x), \quad \forall x \in \mathbb{R}^n. \end{aligned}$$

Utilisant les énoncés (6.14c) et (6.14d), on obtient

$$\begin{aligned} I_D(x) &\geq I_C(x^k) + \sum_{i=1}^r I_{\text{dom } f_i}(x^k) + \langle p_c^k + \sum_{i=1}^r p_{ci}^k, x - x^k \rangle - (\tilde{b}_p^k + \sum_{i=1}^r \tilde{b}_{pi}^k) \\ &= I_D(x^k) + \langle p_D^k, x - x^k \rangle - (\tilde{b}_p^k + \sum_{i=1}^r \tilde{b}_{pi}^k). \end{aligned}$$

L'énoncé (6.14h) se démontre de la même façon en utilisant les énoncés (6.14e) et (6.14f). ■

Les quantités b_{ij}^k , b_j^k , $\tilde{b}_{pi}^k$, $\tilde{b}_p^k$, b_{pi}^k , b_p^k , $\tilde{b}_{pD}^k$ et p_{pD}^k sont les erreurs de linéarisation associées aux fonctions indicatrices.

Lemme 6.4.5 *Si*

$$\begin{aligned} \tilde{\alpha}^k &\triangleq \sum_{i=1}^r [f_i(x^k) - \tilde{f}_{pi}^k] \\ &\quad + \sum_{i=1}^r \nu_{ci}^k \tilde{b}_{pi}^k + \nu_c^k \tilde{b}_p^k \end{aligned} \tag{6.15}$$

alors

$$p^k \in \partial_\epsilon \varphi^k(x^k) = \partial_\epsilon \varphi(x^k) \tag{6.16}$$

où

$$\varepsilon = \tilde{\alpha}^k \geq 0.$$

Démonstration:

Les définitions (6.10) et les lemmes 6.4.2 et 6.4.4 nous permettent d'écrire

$$f_i(x) - f_i(x^k) \geq \langle g_i^j, x - x^k \rangle - [f_i(x^k) - f_{ij}^k], \quad \forall j \in J_{f_i}^k, i \in \{1, \dots, r\}$$

$$f_i(x) - f_i(x^k) \geq \langle p_{f_i}^{k-1}, x - x^k \rangle - [f_i(x^k) - f_{pi}^k], \quad \forall i \in \{1, \dots, r\}$$

$$I_{\text{dom } f_i}(x) \geq \langle a_i^j, x - x^k \rangle - b_{ij}^k, \quad \forall j \in J_{ci}^k, i \in \{1, \dots, r\}$$

$$I_{\text{dom } f_i}(x) \geq \langle p_{ci}^{k-1}, x - x^k \rangle - b_{pi}^k, \quad \forall i \in \{1, \dots, r\}$$

$$I_c(x) \geq \langle a^j, x - x^k \rangle - b_j^k, \quad \forall j \in J_c^k$$

$$I_c(x) \geq \langle p_c^{k-1}, x - x^k \rangle - b_p^k.$$

Soient les constantes $\nu_{ci}^k, i \in \{1, \dots, r\}$, ν_c^k et les multiplicateurs normalisés $(\pi^k, \tilde{\mu}^k, \tilde{\omega}^k)$ obtenus par l'algorithme à l'étape 1. Multiplions chacune des inégalités ci-dessus par le multiplicateur correspondant. Ces multiplicateurs étant positifs, les inégalités ne changent pas de sens et on obtient, en les additionnant et en utilisant les expressions (5.53a)–(5.53f) et (5.65a)–(5.65c):

$$f_i(x) - f_i(x^k) \geq \langle p_{f_i}^k, x - x^k \rangle - [f_i(x^k) - \tilde{f}_{pi}^k], \quad \forall i \in \{1, \dots, r\}$$

$$I_{\text{dom } f_i}(x) \geq \langle p_{ci}^k, x - x^k \rangle - \tilde{b}_{pi}^k, \quad \forall i \in \{1, \dots, r\}$$

$$I_c(x) \geq \langle p_c^k, x - x^k \rangle - \tilde{b}_p^k.$$

En multipliant ces dernières inégalités par les constantes $\nu_{f_i}^k$ (valant 1), $\nu_{ci}^k, i \in$

$\{1, \dots, r\}$, et ν_c^k appropriées, et en additionnant le tout, on trouve

$$\begin{aligned}
& \sum_{i=1}^r [f_i(x) - f_i(x^k)] + \sum_{i=1}^r \nu_{\alpha_i}^k I_{\text{dom } f_i}(x) + \nu_c^k I_c(x) \\
& \geq \langle \sum_{i=1}^r (p_{fi}^k + \nu_{\alpha_i}^k p_{\alpha_i}^k) + \nu_c^k p_c^k, x - x^k \rangle \\
& \quad - \sum_{i=1}^r [f_i(x^k) - \tilde{f}_{pi}^k] - \sum_{i=1}^r \nu_{\alpha_i}^k \tilde{b}_{pi}^k - \nu_c^k \tilde{b}_p^k \\
& = \langle p^k, x - x^k \rangle - \tilde{\alpha}^k, \quad \forall x \in \mathbb{R}^n \quad \text{par (5.55) et (6.15)}
\end{aligned}$$

De plus, $\forall x \in \mathbb{R}^n$, on a

$$\begin{aligned}
\varphi^k(x) &= f(x) - f(x^k) + I_D(x) \\
&= \sum_{i=1}^r [f_i(x) - f_i(x^k)] + I_c(x) + \sum_{i=1}^r I_{\text{dom } f_i}(x) \\
&\geq \sum_{i=1}^r [f_i(x) - f_i(x^k)] + \nu_c^k I_c(x) + \sum_{i=1}^r \nu_{\alpha_i}^k I_{\text{dom } f_i}(x) \quad \text{par (5.51a)-(5.51c)} \\
&\quad \text{et (5.52b)} \\
&\geq \langle p^k, x - x^k \rangle - \tilde{\alpha}^k.
\end{aligned}$$

Par conséquent,

$$\varphi^k(x) \geq \varphi^k(x^k) + \langle p^k, x - x^k \rangle - \tilde{\alpha}^k, \quad \forall x \in \mathbb{R}^n$$

car $\varphi^k(x^k) = 0$. En posant $x = x^k$, on obtient $\tilde{\alpha}^k \geq 0$.

Remarquons que

$$\varphi(x) = \varphi^k(x) + f(x^k) - f^*, \quad \forall x \in \mathbb{R}^n$$

de sorte que

$$\partial_\varepsilon \varphi(x) = \partial_\varepsilon \varphi^k(x), \quad \forall x \in \mathbb{R}^n, \quad \forall \varepsilon \geq 0.$$

■

Le problème (6.3) qui doit être résolu à l'étape 1 de l'algorithme est le dual du problème (5.66a) qui consiste à déterminer

$$d^k \in \operatorname{argmin} \{ \widehat{\varphi}_a^k(x^k + d) + \Psi^k \|d\|^q \}.$$

Or, le théorème B.19 nous apprend que d^k est une solution optimale du problème non contraint (5.66a) si et seulement si

$$0 \in \partial \left(\widehat{\varphi}_a^k(x^k + \cdot) + \Psi^k \|\cdot\|^q \right) (d^k),$$

i.e., par le théorème B.7, si et seulement si

$$0 \in \partial \widehat{\varphi}_a^k(x^k + d^k) + \Psi^k \partial \|\cdot\|^q(d^k),$$

ou encore, par le théorème B.9, si et seulement si

$$0 \in \partial \widehat{\varphi}_a^k(x^k + d^k) + \Psi^k q \|d^k\|^{q-1} \partial \|\cdot\|(d^k).$$

Ceci signifie que

$$\partial \widehat{\varphi}_a^k(x^k + d^k) \cap [-\Psi^k q \|d^k\|^{q-1} \partial \|\cdot\|(d^k)] \neq \emptyset.$$

Or, nous savons par le lemme 5.6.4 que

$$p^k \in -\Psi^k q \|d^k\|^{q-1} \partial \|\cdot\|(d^k).$$

Le prochain résultat nous apprend que

$$p^k \in \partial \widehat{\varphi}_a^k(x^k + d^k),$$

de sorte que p^k fait partie de l'intersection.

Lemme 6.4.6 *Le vecteur p^k vérifie*

$$p^k \in \partial \widehat{\varphi}_a^k(x^k + d^k). \tag{6.17a}$$

De plus, si $\widehat{\varphi}_{ap}^k$ désigne la fonction résultant, a posteriori, de l'agrégation du problème (5.64) associé à la fonction φ_a^k , alors

$$\widehat{\varphi}_{ap}^k(x^k + d^k) = \widehat{\varphi}_a^k(x^k + d^k) \quad (6.17b)$$

et

$$p^k \in \partial \widehat{\varphi}_{ap}^k(x^k + d^k). \quad (6.17c)$$

Démonstration:

Puisque d^k est une solution du problème (5.66a), il existe aussi $v^k \in \mathbb{R}^r$, $\sigma_f^k \in \mathbb{R}$ et $\sigma_D^k \in \mathbb{R}$ tels que $(d^k, v^k, \sigma_f^k, \sigma_D^k)$ soit une solution optimale du problème (5.64). Par conséquent,

$$\begin{aligned} \widehat{\varphi}_a^k(x^k + d) &= \sigma_f + \sigma_D \\ &= \sum_{i=1}^r \left(\max \{ \langle p_{fi}^{k-1}, d \rangle + f_{pi}^k; \langle g_i^j, d \rangle + f_{ij}^k : j \in J_{fi}^k \} - f_i(x^k) \right) \\ &\quad + \max \{ \langle p_{ci}^{k-1}, d \rangle - b_{pi}^k; \langle a_i^j, d \rangle - b_{ij}^k : j \in J_{ci}^k, i = 1, \dots, r; \\ &\quad \langle p_c^{k-1}, d \rangle - b_p^k; \langle a^j, d \rangle - b_j^k : j \in J_c^k \} \\ &\geq \sum_{i=1}^r \left(\langle \pi_{pi}^k p_{fi}^{k-1} + \sum_{j \in J_{fi}^k} \pi_{ij}^k g_i^j, d \rangle + \pi_{pi}^k f_{pi}^k + \sum_{j \in J_{fi}^k} \pi_{ij}^k f_{ij}^k - f_i(x^k) \right) \\ &\quad + \max \{ \langle \tilde{\mu}_{pi}^k p_{ci}^{k-1} + \sum_{j \in J_{ci}^k} \tilde{\mu}_{ij}^k a_i^j, d \rangle - \tilde{\mu}_{pi}^k b_{pi}^k - \sum_{j \in J_{ci}^k} \tilde{\mu}_{ij}^k b_{ij}^k : i = 1, \dots, r; \\ &\quad \langle \tilde{\omega}_p^k p_c^{k-1} + \sum_{j \in J_c^k} \tilde{\omega}_j^k a^j, d \rangle - \tilde{\omega}_p^k b_p^k - \sum_{j \in J_c^k} \tilde{\omega}_j^k b_j^k \} \\ &= \sum_{i=1}^r \left(\langle p_{fi}^k, d \rangle + \tilde{f}_{pi}^k - f_i(x^k) \right) \\ &\quad + \max \{ \langle p_{c1}^k, d \rangle - \tilde{b}_{p1}^k, \dots, \langle p_{cr}^k, d \rangle - \tilde{b}_{pr}^k, \langle p_c^k, d \rangle - \tilde{b}_p^k \} \\ &\quad \text{par (5.65a)–(5.65c)} \end{aligned}$$

$$\begin{aligned}
&\geq \sum_{i=1}^r \left(\langle p_{fi}^k, d \rangle + \tilde{f}_{pi}^k - f_i(x^k) \right) \\
&\quad + \sum_{i=1}^r \nu_{ci}^k (\langle p_{ci}^k, d \rangle - \tilde{b}_{pi}^k) + \nu_c^k (\langle p_c^k, d \rangle - \tilde{b}_p^k) \\
&\hspace{15em} \text{par (5.51b), (5.51c) et (5.52b)} \\
&= \langle p^k, d \rangle + \sum_{i=1}^r \left[\left(\tilde{f}_{pi}^k - f_i(x^k) \right) - \nu_{ci}^k \tilde{b}_{pi}^k \right] - \nu_c^k \tilde{b}_p^k \quad \text{par (5.55)} \\
&= \langle p^k, d \rangle + \hat{\varphi}_a^k(x^k + d^k) - \langle p^k, d^k \rangle \quad \text{par (5.67)}
\end{aligned}$$

On a donc

$$\hat{\varphi}_a^k(x^k + d) \geq \hat{\varphi}_a^k(x^k + d^k) + \langle p^k, d - d^k \rangle, \quad \forall d \in \mathbb{R}^n$$

d'où

$$p^k \in \partial \hat{\varphi}_a^k(x^k + d^k).$$

La fonction $\hat{\varphi}_{ap}^k$ est associée au problème (5.58). Par le lemme 5.9.7, on a

$$\hat{\varphi}_{ap}^k(x^k + d^k) = \varphi_a^k(x^k + d^k)$$

où d^k fait à la fois partie d'une solution optimale du problème (5.64) et d'une solution optimale du problème (5.58). De plus, on peut choisir $\hat{J}_{fi}^k = \emptyset$, $\forall i \in \{1, \dots, r\}$. Par conséquent, on a

$$\begin{aligned}
\hat{\varphi}_{ap}^k(x^k + d) &= \sum_{i=1}^r \left(\langle p_{fi}^k, d \rangle + \tilde{f}_{pi}^k - f_i(x^k) \right) \\
&\quad + \max \{ \langle p_{c1}^k, d \rangle - \tilde{b}_{p1}^k, \dots, \langle p_{cr}^k, d \rangle - \tilde{b}_{pr}^k, \langle p_c^k, d \rangle - \tilde{b}_p^k \} \\
&\geq \langle p^k, d \rangle + \hat{\varphi}_a^k(x^k + d^k) - \langle p^k, d^k \rangle \quad \text{par le raisonnement précédent} \\
&= \hat{\varphi}_{ap}^k(x^k + d^k) + \langle p^k, d - d^k \rangle \quad \text{par (6.17b)}
\end{aligned}$$

et

$$p^k \in \partial \hat{\varphi}_{ap}^k(x^k + d^k).$$

■

Puisque

$$\begin{aligned}
 w^k &= \hat{\varphi}_a^k(x^k + d^k) + \Psi_k \|d^k\|^q \\
 &= \min\{\hat{\varphi}_a^k(x^k + d) + \Psi_k \|d\|^q : d \in \mathbb{R}^n\} \\
 &\leq \hat{\varphi}_a^k(x^k + d) + \Psi_k \|d\|^q, \quad \forall d \in \mathbb{R}^n
 \end{aligned}$$

alors

$$w^k - \Psi_k \|d\|^q \leq \hat{\varphi}_a^k(x^k + d), \quad \forall d \in \mathbb{R}^n \quad (6.18a)$$

et

$$w^k - \Psi_k \|d^k\|^q = \hat{\varphi}_a^k(x^k + d^k). \quad (6.18b)$$

De plus, le lemme 6.4.6 ainsi que la discussion qui le précède, nous disent que p^k est un sous-gradient des membres de gauche et de droite de l'inégalité (6.18a). La résolution du problème (5.66a) équivaut à faire croître w^k le plus possible, c'est-à-dire jusqu'à ce que le graphe de la fonction $w^k - \Psi_k \|d\|^q$ s'appuie sur le graphe de $\hat{\varphi}_a^k(x^k + d)$. La figure ci-dessous illustre cette idée.

La ligne hachurée, tangente aux deux courbes, a p^k comme sous gradient. Elle illustre le processus d'agrégation où la fonction $\hat{\varphi}_a^k(x^k + \cdot)$ est remplacée par une fonction affine produisant la (les) même(s) solution(s) optimale(s).

Lemme 6.4.7 *À la $k^{\text{ème}}$ itération de l'algorithme, on a*

$$\sigma^k \triangleq \sigma_f^k + \sigma_D^k = - \left[\frac{(\|p^k\|^*)^{\frac{q}{q-1}}}{(q\Psi_k)^{\frac{1}{q-1}}} + \tilde{\alpha}^k \right] \quad (6.19a)$$

$$= - \left[q\Psi_k \|d^k\|^q + \tilde{\alpha}^k \right] \quad (6.19b)$$

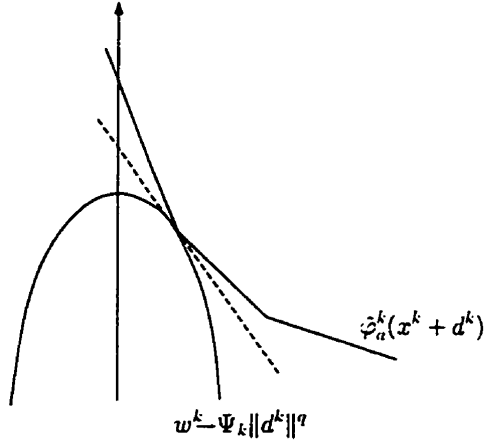


Figure 6.1 Représentation géométrique du procédé d'agrégation

et

$$\sigma^k \leq w^k \leq \frac{q-1}{q} \sigma^k \leq 0. \quad (6.20)$$

Démonstration:

L'expression (6.19a) découle de (5.56) et de la définition (6.15) de $\tilde{\alpha}^k$. L'expression (6.19b) est obtenue en appliquant l'égalité (5.40k) à (6.19a).

La première des inégalités (6.20) découle de (6.4) et du fait que $\Psi^k \|d\|^q \geq 0$, $\forall d \in \mathbb{R}^n$. La seconde provient de (6.4), (6.19b) et de la non-négativité de $\tilde{\alpha}^k$:

$$\begin{aligned} w^k &= -[\Psi_k(q-1)\|d^k\|^q + \tilde{\alpha}^k] \\ &= \frac{q-1}{q} \sigma^k - \frac{\tilde{\alpha}^k}{q} \\ &\leq \frac{q-1}{q} \sigma^k. \end{aligned}$$

Enfin, la dernière inégalité provient de la non-négativité des deux termes définissant σ^k dans les expressions (6.19a) et (6.19b). ■

Par le lemme 6.4.7, on a

$$\sigma^k \geq \tilde{\alpha}^k \quad (6.21a)$$

et

$$-\sigma^k \geq \frac{(\|p^k\|^*)^{\frac{q}{q-1}}}{(q\Psi_k)^{\frac{1}{q-1}}}, \quad (6.21b)$$

d'où

$$\|p^k\|^* \leq (q\Psi_k)^{\frac{1}{q}} (-\sigma^k)^{\frac{q-1}{q}}. \quad (6.22)$$

Grâce au lemme 6.4.5, on peut écrire

$$\begin{aligned} \varphi^k(x) &\geq \varphi^k(x^k) + \langle p^k, x - x^k \rangle - \tilde{\alpha}^k, & \forall x \in \mathbb{R}^n \\ &\geq \langle p^k, x - x^k \rangle + \sigma^k & \text{par (6.21a)} \\ & & \text{et parce que } \varphi^k(x^k) = 0 \\ &\geq -\|p^k\|^k \|x - x^k\| + \sigma^k & \text{par (B8)} \\ &\geq -(q\Psi_k)^{\frac{1}{q}} (-\sigma^k)^{\frac{q-1}{q}} \|x - x^k\| + \sigma^k & \text{par (6.22)} \end{aligned}$$

Puisque $\varphi^k(x) = f(x) - f(x^k) + I_D(x)$, $\forall x \in \mathbb{R}^n$, alors on a

$$f(x) \geq f(x^k) - (q\Psi_k)^{\frac{1}{q}} (-\sigma^k)^{\frac{q-1}{q}} \|x - x^k\| + \sigma^k, \quad \forall x \in D \quad (6.23)$$

et le terme

$$-(q\Psi_k)^{\frac{1}{q}} (-\sigma^k)^{\frac{q-1}{q}} \|x - x^k\| + \sigma^k$$

de l'inégalité (6.23) est un indicateur d'optimalité du centre de stabilité courant.

Remarque: Par (6.4), (6.19a), (5.40k) et (6.20), on a

$$-w^k = \left(\frac{q-1}{q}\right) \frac{(\|p^k\|^*)^{\frac{q}{q-1}}}{(q\Psi_k)^{\frac{1}{q-1}}} + \tilde{\alpha}^k \geq 0. \quad (6.24)$$

Le premier terme du membre de droite est une fonction croissante de $\|p^k\|^*$ qui devient nulle lorsque $p^k = 0 \in \mathbb{R}^n$. Sa valeur indique donc jusqu'à quel point p^k diffère du vecteur nul. De plus, le lemme 6.4.5 nous apprend que le second terme $\tilde{\alpha}^k$ mesure la distance de p^k à $\partial\varphi^k(x^k)$.

Or, on sait que $\bar{x}$ est optimal si et seulement si $\bar{p} = 0 \in \partial\varphi(\bar{x})$ (théorème B.19). Par conséquent, $-w^k$ est un indicateur composite dont la première partie "mesure" la distance entre p^k et $0 \in \mathbb{R}^n$ tandis que la seconde "mesure" la distance entre p^k et $\partial\varphi^k(x^k)$. En minimisant $-w^k$, l'algorithme cherche à déterminer un x^k tel que $p^k = 0 \in \partial\varphi(x^k)$ et w^k se trouve à mesurer l'optimalité de x^k .

6.5 Convergence de l'algorithme utilisant la stratégie d'agrégation

À moins d'indication contraire, nous supposons tout au long de cette section que $\varepsilon_s = 0$.

Lemme 6.5.1 *Si l'algorithme s'arrête à la $k^{\text{ème}}$ itération, alors x^k est une solution optimale du problème (6.1).*

Démonstration:

Le critère d'arrêt étant

$$-w^k \leq \varepsilon_s = 0,$$

on doit, par le lemme 6.4.7, avoir $w^k = 0$, c'est-à-dire en utilisant l'égalité (6.24)

$$\tilde{\alpha}^k = 0 \quad \text{et} \quad \|p^k\|^* = 0$$

ou encore

$$\tilde{\alpha}^k = 0 \quad \text{et} \quad p^k = 0.$$

Par conséquent,

$$0 \in \partial\varphi(x^k) \quad (\text{lemme 6.4.5})$$

et x^k est une solution optimale (théorème B.19). ■

Nous supposerons à partir de maintenant que l'algorithme effectue une infinité d'itérations. L'analyse que nous en ferons examinera deux situations, soit celle où l'algorithme effectue une infinité de pas de descente et celle où il n'effectue qu'un nombre fini de pas de descente.

Lemme 6.5.2 (*Kiwiel, 1990: lemme 3.1*)

S'il existe $\tilde{x} \in D$ tel que

$$f(x^k) \geq f(\tilde{x}), \quad \forall k \tag{6.25}$$

alors

$$\sum_{k=1}^{\infty} \delta^k \|\sigma^k\| \leq \frac{f(x^1) - f(\tilde{x})}{m} < +\infty \tag{6.26}$$

et $\{\sigma^k\}_K \rightarrow 0$ si K est de taille infinie, i.e. si l'algorithme effectue une infinité de pas de descente.

Démonstration:

On a

$$f(x^{k+1}) \leq f(x^k) + m\delta^k \sigma^k, \quad \forall k$$

où

$$\delta^k = \begin{cases} 1 & \text{si pas de descente à l'itération } k \\ 0 & \text{si pas nul à l'itération } k \end{cases}$$

$$m \in (0, 1)$$

et

$$\sigma^k \leq 0.$$

Par conséquent,

$$0 \leq -m\delta^k\sigma^k \leq f(x^k) - f(x^{k+1}), \quad \forall k.$$

En additionnant ces inégalités, $\forall k \in \{1, 2, \dots\}$, et en utilisant l'inégalité (6.25), on obtient

$$0 \leq -m \sum_{k=1}^{\infty} \delta^k \sigma^k \leq f(x^1) - f(\tilde{x}),$$

d'où

$$-\sum_{k=1}^{\infty} \delta^k \sigma^k = \sum_{k=1}^{\infty} \delta^k |\sigma^k| = \sum_{k \in K} |\sigma^k| \leq \frac{f(x^1) - f(\tilde{x})}{m} < +\infty.$$

■

Remarque: L'inégalité (6.26) nous permet de conclure que la suite $\{\sigma^k\}$ converge vers zéro plus rapidement que la suite $\{1/k\}$ si l'algorithme n'effectue que des pas de descente. La même conclusion s'applique à la suite $\{w^k\}$, en vertu des inégalités (6.20) du lemme 6.4.7. On ne sait toutefois rien dire de la convergence de la suite $\{x^k\}$.

Proposition 6.5.3 *Si la fonction économique f du problème (6.1) est inf-compacte, i.e. si*

$$\{x \in \mathbb{R}^n : f(x) \leq \lambda\}$$

est compact pour toute valeur de $\lambda \in \mathbb{R}$, et si l'algorithme effectue une infinité de pas de descente, alors tous les points d'accumulation de la suite $\{x^k\}$ sont des solutions optimales du problème (6.1).

Démonstration:

Puisque

$$f(x^{k+1}) \leq f(x^k), \quad \forall k$$

et

$$f(x^{k+1}) < f(x^k), \quad \forall k \in K,$$

alors

$$\begin{aligned} x^k \in S &\triangleq \{x \in \mathbb{R}^n : f(x) \leq f(x^1)\} \cap D, \quad \forall k \\ &= \{x \in \mathbb{R}^n : f(x) \leq f(x^1)\} \cap C, \quad \forall k. \end{aligned}$$

Par l'inf-compacité de f , S est compact et la suite $\{x^k\}$ est bornée: elle possède donc au moins une valeur d'adhérence. Il doit également en être ainsi de la sous-suite $\{x^k\}_K$.

Soit $\bar{x} \in S$ une valeur d'adhérence arbitraire de la suite $\{x^k\}_K$. Alors

$$\bar{x} \in S \subseteq D \subseteq C \quad (\text{par la fermeture de } D)$$

et il existe une sous-suite $\{x^k\}_{K'}$, $K' \subseteq K$, telle que

$$\lim_{\substack{k \rightarrow \infty \\ k \in K'}} x^k = \bar{x}.$$

Par le lemme 6.5.2, $\{\sigma^k\}_K \rightarrow 0$, d'où $\{\sigma^k\}_{K'} \rightarrow 0$. Puisque

$$\Psi_k \geq \Psi_{\min}, \quad \forall k$$

le lemme 6.4.7 implique alors que

$$\{\|d^k\|\}_K \rightarrow 0, \{\|p^k\|^*\}_K \rightarrow 0, \{\tilde{\alpha}^k\}_K \rightarrow 0 \text{ et } \{w^k\}_K \rightarrow 0,$$

d'où

$$\{d^k\}_K \rightarrow 0 \in \mathbb{R}^n \quad \text{et} \quad \{p^k\}_K \rightarrow 0 \in \mathbb{R}^n.$$

Mais $p^k \in \partial_{\tilde{\alpha}^k} \varphi(x^k)$, par le lemme 6.4.5. Par conséquent

$$\lim_{\substack{k \rightarrow \infty \\ k \in K'}} (\tilde{\alpha}^k, x^k, p^k) = (0, \bar{x}, 0)$$

car $K' \subseteq K$, de sorte que $0 \in \partial_0 \varphi(\bar{x}) = \partial \varphi(\bar{x})$ par la proposition B.16. Le théorème B.19 nous dit alors que $\bar{x}$ est une solution optimale du problème (6.1). ■

Remarque: L'hypothèse d'inf-compacité ne permet pas d'affirmer que la suite $\{x^k\}$ est convergente. En fait, c'est une question ouverte (Kiwiel, 1990). Cette hypothèse n'est habituellement pas utilisée lorsque la norme choisie est euclidienne (ou plus généralement quadratique) et que $h(r) = r^2$. Nous avons vu que les expressions (5.40m) deviennent alors

$$p^k = -2\Psi_k d^k \quad \text{et} \quad d^k = -p^k / 2\Psi_k.$$

Puisque le lemme 6.4.5 permet d'écrire

$$f(x) \geq f(x^k) + \langle p^k, x - x^k \rangle - \tilde{\alpha}^k, \quad \forall x \in D,$$

on peut substituer $-2\Psi_k d^k$ à p^k et écrire

$$f(x) \geq f(x^k) - 2\Psi_k \langle d^k, x - x^k \rangle - \tilde{\alpha}^k, \quad \forall x \in D.$$

Si le problème possède une solution optimale finie, il existe alors $\tilde{x} \in D$ tel que

$$f(\tilde{x}) \leq f(x^k), \quad \forall k.$$

On peut alors écrire

$$-2\langle d^k, \tilde{x} - x^k \rangle \leq \frac{\tilde{\alpha}^k}{\Psi_k}, \quad \forall k. \quad (6.27)$$

De plus,

$$\begin{aligned} \|\tilde{x} - x^{k+1}\|^2 &= \|\tilde{x} - x^k\|^2 + 2\langle \tilde{x} - x^k, x^k - x^{k+1} \rangle + \|x^k - x^{k+1}\|^2 \\ &= \|\tilde{x} - x^k\|^2 - 2\delta^k \langle \tilde{x} - x^k, d^k \rangle + \delta^k \|d^k\|^2 \\ &\leq \|\tilde{x} - x^k\|^2 + \delta^k \frac{\tilde{\alpha}^k}{\Psi_k} + \delta^k \|d^k\|^2 \quad \text{par (6.27)} \\ &\leq \|\tilde{x} - x^k\|^2 + \delta^k \frac{\tilde{\alpha}^k}{\Psi_k} + 2\delta^k \|d^k\|^2 \\ &= \|\tilde{x} - x^k\|^2 - \delta^k \frac{\sigma^k}{\Psi_k} \quad \text{par (6.19b)} \\ &= \|\tilde{x} - x^k\|^2 + \delta^k \frac{|\sigma^k|}{\Psi_k}. \end{aligned}$$

Puisque $\Psi_k \geq \Psi_{\min} > 0$, $\forall k$, alors pour tout $n > k$, on a

$$\begin{aligned} \|\tilde{x} - x^n\|^2 &\leq \|\tilde{x} - x^k\|^2 + \sum_{i=k}^{\infty} \delta^i \frac{|\sigma^i|}{\Psi_i} \\ &\leq \|\tilde{x} - x^k\|^2 + \frac{1}{\Psi_{\min}} \sum_{i=k}^{\infty} \delta^i |\sigma^i| \\ &< +\infty \end{aligned} \quad (6.28)$$

la dernière inégalité découlant du lemme 6.5.2. Ceci montre que la suite $\{x^k\}$ est bornée. Elle possède donc au moins un point d'accumulation. L'inégalité (6.28) est encore utilisée pour démontrer que la suite complète est convergente (voir par exemple la démonstration du lemme 3.1 de Kiwiel (1990) ou du théorème XV.3.2.2 d'Hiriart-Urruty et Lemaréchal (1993b)).

L'hypothèse d'inf-compacité de la fonction économique a été utilisée par Auslender (1978, 1987). Elle permet une preuve plus simple ne nécessitant pas le recours à la borne inférieure $\Psi_{\min} > 0$ de la suite $\{\Psi_k\}$. Elle ne permet malheureusement pas, jusqu'à présent, de prouver que la suite $\{x^k\}$ est convergente.

Dans le cas général, il ne semble pas possible d'utiliser les expressions (5.40m) et le lemme 6.4.5 pour obtenir une inégalité semblable à (6.27). Il faut vraisemblablement exiger que la fonction $\|d\|^q$ soit fortement convexe (ce qui est le cas lorsqu'une norme quadratique est choisie et que $q \geq 2$) pour obtenir de tels résultats.

Il semble donc y avoir un arbitrage à faire entre les propriétés de la fonction économique f et celles du terme de pénalisation. La minimisation de fonctions convexes générales requiert l'imposition de propriétés fortes (comme la forte convexité) au terme de pénalisation. Par contre, l'amoindrissement des exigences relatives à ce dernier entraîne l'imposition de propriétés plus fortes (comme l'inf-compacité) à la fonction f .

Il est toutefois peut-être possible de démontrer la proposition 6.5.3, sans changer les propriétés du terme de pénalisation, en remplaçant l'hypothèse d'inf-compacité de f par une hypothèse plus faible permettant de considérer une plus vaste classe de fonctions comme l'ont fait Auslender et Crouzeix (1989).

Il reste maintenant à analyser le cas où l'algorithme n'effectue qu'un nombre fini de pas de descente. Il existe alors un indice k_0 tel que

$$x^k = x^{k_0}, \quad \forall k > k_0$$

et l'algorithme effectue une infinité de pas nuls consécutifs après le dernier pas de

descente.

Il n'est pas possible dans ce cas d'utiliser l'inégalité (6.26) pour démontrer que $\sigma^k \rightarrow 0$. Remarquons que la suite $\{x^k\}$ est convergente car

$$\lim_{k \rightarrow \infty} x^k = x^{k_0}.$$

Lemme 6.5.4 *Si $k_1, k_2 \in K$, $k_1 < k_2$ (on pose $k_2 = +\infty$ si k_1 est l'indice du dernier pas de descente), sont les indices de deux pas de descente successifs, alors pour k et ℓ tels que*

$$k_1 < k \leq \ell \leq k_2,$$

on a

$$\begin{aligned} 0 \leq -w^\ell \leq -w^k &\leq \Psi_{k_1+1}(q-1) \left(\frac{\|g^{k_1+1}\|^*}{q\Psi_{k_1+1}} \right)^{\frac{q}{q-1}} \\ &\leq \Psi_{\min}(q-1) \left(\frac{\|g^{k_1+1}\|^*}{q\Psi_{\min}} \right)^{\frac{q}{q-1}} \end{aligned} \quad (6.29)$$

où

$$g^{k_1+1} = \sum_{i=1}^r g_i^{k_1+1} \in \partial f(x^{k_1+1}), \quad (6.30)$$

et

$$g_i^{k_1+1} \in \partial f_i(x^{k_1+1}).$$

Démonstration:

Si $k = \ell$, alors on a trivialement

$$0 \leq -w^\ell = -w^k,$$

et les deux premières inégalités du résultat sont vérifiées.

Si $k < \ell$, alors k est un indice de pas nul et $\Psi_k \leq \Psi_\ell$. La valeur optimale de la fonction économique du problème (5.64) de l'itération k étant w^k , alors la proposition 5.9.9 et la discussion suivant le lemme 5.9.10 nous disent qu'en remplaçant les contraintes du problème (5.64) par celles résultant de l'agrégation de ces dernières ainsi que, possiblement, un sous-ensemble quelconque de ces contraintes, on ne modifie pas la valeur optimale de la fonction économique du problème ainsi modifié ni, d'ailleurs, son ensemble de solutions optimales.

w^k est donc la valeur optimale de la fonction économique du problème (5.64) modifié, i.e. agrégé. Le problème de l'itération $k + 1$ est obtenu en ajoutant des contraintes (correspondant, entre autres, aux hyperplans de séparation associés à y^{k+1}) et en changeant la constante Ψ_k pour $\Psi_{k+1} \geq \Psi_k$ lorsque $y^{k+1} \notin D$. Par conséquent, le problème (5.64) de l'itération $k + 1$ ne peut prendre de valeur optimale inférieure à w^k , et l'on a

$$w^{k+1} \geq w^k.$$

Les deux premières inégalités du résultat sont donc toujours vraies. Supposons que $k = k_1 + 1$. k est alors le plus faible indice d'une itération où x^{k_1+1} est un centre de stabilité. On a

$$x^k \in D$$

et on peut écrire

$$\begin{aligned}
0 \geq w^k &= \min \left\{ \widehat{\varphi}_a^k(x^k + d) + \Psi_k \|d\|^q : d \in \mathbb{R}^n \right\} && \text{par (5.66a)} \\
&= \min \left\{ \sum_{i=1}^r \left[\widehat{f}_{ia}^k(x^k + d) - f_i(x^k) \right] + \widetilde{f}_{Da}^k(x^k + d) + \Psi_k \|d\|^q : d \in \mathbb{R}^n \right\} \\
&&& \text{par (5.66b) et (5.66c)} \\
&\geq \min \left\{ \sum_{i=1}^r \left[\widehat{f}_{ia}^k(x^k + d) - f_i(x^k) \right] + \Psi_k \|d\|^q : d \in \mathbb{R}^n \right\} \\
&&& \text{par (5.66e)} \\
&\geq \min \left\{ \sum_{i=1}^r \left[f_{ikk}(d) - f_i(x^k) \right] + \Psi_k \|d\|^q : d \in \mathbb{R}^n \right\} \\
&&& \text{par (5.66d)} \\
&&& \text{et parce que } k \in J_{fi}^k \\
&= \min \left\{ \sum_{i=1}^r \left[f_i(y^k) + \langle g_i^k, x^k - y^k \rangle + \langle g_i^k, d \rangle - f_i(x^k) \right] + \Psi_k \|d\|^q : d \in \mathbb{R}^n \right\} \\
&&& \text{par (5.41a) et (5.17a)} \\
&= \min \left\{ \left\langle \sum_{i=1}^r g_i^k, d \right\rangle + \Psi_k \|d\|^q : d \in \mathbb{R}^n \right\} && \text{car } x^{k_1+1} = y^{k_1+1} \\
&&& \text{puisque } k_1 \in K \\
&= \min \left\{ \langle g^k, d \rangle + \Psi_k \|d\|^q : d \in \mathbb{R}^n \right\} \\
&= \langle g^k, d^* \rangle + \Psi_k \|d^*\|^q
\end{aligned}$$

où d^* est tel que

$$\begin{aligned}
0 &\in \partial(\langle g^k, \cdot \rangle + \Psi_k \|\cdot\|^q)(d^*) && \text{(théorème B.19)} \\
&= g^k + q\Psi_k \|d^*\|^{q-1} \partial\|\cdot\|(d^*),
\end{aligned}$$

l'égalité résultant de l'application du théorème B.9. Il existe donc $\nu^* \in \partial\|\cdot\|(d^*)$ tel que

$$g^k + q\Psi_k \|d^*\|^{q-1} \nu^* = 0,$$

d'où

$$\nu^* = \frac{-g^k}{q\Psi_k\|d^*\|^{q-1}}.$$

De plus, par la proposition B.26, on a

$$\begin{aligned}\|d^*\| &= \langle \nu^*, d^* \rangle \\ &= \frac{\langle g^k, d^* \rangle}{q\Psi_k\|d^*\|^{q-1}}.\end{aligned}$$

Par conséquent,

$$\begin{aligned}\langle g^k, d^* \rangle + \Psi_k\|d^*\|^q &= -q\Psi_k\|d^*\|^q + \Psi_k\|d^*\|^q \\ &= -\Psi_k(q-1)\|d^*\|^q\end{aligned}$$

et

$$\begin{aligned}0 \geq w^k &\geq -\Psi_k(q-1)\|d^*\|^q \\ &= -\Psi_k(q-1)\left(\frac{\|g^k\|^*}{q\Psi_k}\right)^{\frac{q}{q-1}} && \text{par (5.40l)} \\ &= -\frac{(q-1)}{\Psi_k^{\frac{1}{q-1}}}\left(\frac{\|g^k\|^*}{q}\right)^{\frac{q}{q-1}} \geq -\frac{(q-1)}{\Psi_{\min}^{\frac{1}{q-1}}}\left(\frac{\|g^k\|^*}{q}\right)^{\frac{q}{q-1}} && (\text{où } k = k_1 + 1)\end{aligned}$$

■

Lemme 6.5.5 *Si $k_1, k_2 \in K$, $k_1 < k_2$ (on pose $k_2 = +\infty$ si k_1 est l'indice du dernier pas de descente), sont les indices de deux pas de descente successifs dans K , alors pour k tel que*

$$k_1 < k \leq k_2$$

on a

$$\|d^k\| \leq \frac{1}{\Psi_k^{\frac{1}{q}}(\Psi_{\min})^{\frac{1}{q(q-1)}}}\left(\frac{\|g^{k_1+1}\|^*}{q}\right)^{\frac{1}{q-1}} \quad (6.31)$$

où g^{k_1+1} est défini en (6.30).

Démonstration:

Par le lemme 6.4.7 et l'expression (6.4), on a

$$\begin{aligned} w^k &= \sigma^k + \Psi_k \|d^k\|^q \\ &= -(q-1)\Psi_k \|d^k\|^q - \bar{\alpha}^k \\ &\leq -(q-1)\Psi_k \|d^k\|^q, \quad \forall k. \end{aligned}$$

alors, par le lemme 6.5.4, on a

$$(q-1)\Psi_k \|d^k\|^q \leq -w^k \leq (q-1)\Psi_{\min} \left(\frac{\|g^{k_1+1}\|^*}{q\Psi_{\min}} \right)^{\frac{q}{q-1}},$$

d'où

$$\|d^k\| \leq \frac{1}{\Psi_k^{\frac{1}{q}} (\Psi_{\min})^{\frac{1}{q(q-1)}}} \left(\frac{\|g^{k_1+1}\|^*}{q} \right)^{\frac{1}{q-1}}.$$

■

Remarque: Le fonctionnement du super-oracle qui peut engendrer plus d'un hyperplan de séparation à chaque itération k où $y^{k+1} \notin D$ correspond à la modification de l'algorithme de Kelley (1960) et Cheney-Goldstein (1959) proposée par Wolfe (1961) dans le but d'en accélérer la convergence. Wolfe, en plus d'engendrer une contrainte associée au point y^{k+1} , permet que l'on engendre des contraintes linéaires associées à des points arbitraires additionnels. C'est en fait ce que fait notre algorithme. En effet, si les t , $t \in \{1, \dots, r+1\}$, hyperplans de séparation produits par l'algorithme sont différents, on peut toujours prétendre que chacun d'eux est associé à un point $\tilde{y}^{kj}$, $j \in \{1, \dots, t\}$, différent. Cela est possible parce que l'on exploite la structure de la fonction économique.

Il est facile de démontrer le résultat suivant.

Lemme 6.5.6 *A chaque fois qu'un point y^{k+1} n'est pas réalisable, il ne faut qu'un nombre fini d'itérations pour retrouver un point $y^{k''+1} \in D, k'' > k$ si l'ensemble D est polyédrique et si les hyperplans de séparation sont construits avec les pièces linéaires définissant la frontière de D .*

Remarque: C'est ce qui se passe avec la méthode de Benders.

Remarque: Le problème de la détermination d'une solution intermédiaire $y^{k''+1}$ réalisable en un nombre fini d'itérations après avoir quitté le domaine réalisable est difficile. Sous sa forme actuelle l'algorithme ne peut pas le faire lorsque l'hypothèse de polyédralité est relâchée. Il faut probablement modifier la dernière partie de l'étape 8 pour y parvenir. Un tel changement pourrait utiliser la formulation alternative $\alpha'(x^k, y^{k+1})$ de l'erreur de linéarisation proposée à la remarque 6.3.4 et un test semblable à (6.7) pour déterminer s'il faut accroître ou non un facteur multiplicatif $c^k \geq 1$ associé à la variable σ_D dans la fonction économique du problème primal suivant

$$\underset{x, \sigma_f, \sigma_D}{\text{minimiser}} \sigma_f + c^k \sigma_D + \Psi_k \|d\|^q$$

sous les contraintes

$$\hat{f}^k(x) - f(x^k) \leq \sigma_f,$$

$$\hat{I}_D^k \leq \sigma_D$$

$$\sigma_D \geq 0$$

qui généralise le problème (5.22a)–(5.22f). On obtient ce dernier en posant $c^k = 1, \forall k$. Kiwiel (1991) a utilisé un procédé semblable pour un problème plus simple comportant des contraintes explicites définies avec des fonctions à valeurs finies.

On peut également, lorsque $y^{k+1} = x^k + d^k \notin D$, considérer le point $\tilde{y}^{k+1} = \tilde{x} + \theta(y^{k+1} - \tilde{x})$, $\theta \in (0, 1)$, où $\tilde{x}$ est un point connu de $\text{int } D$. On trouvera alors $\tilde{y}^{k''+1} \in D$, $k'' > k$, en un nombre fini d'étapes si les oracles produisent leurs hyperplans de séparation à l'aide d'une multi-application fermée (Berge, 1959; Zangwill, 1969; Topkis, 1970) et si l'on empêche Ψ_k de croître indéfiniment lorsque $y^{k+1} \notin D$. En effet, on peut alors aisément vérifier, si $y^{k+1} \notin D$, $\forall k > k_0$, que la suite $\{y^{k+1}\}_{k > k_0}$ possède des points d'accumulation et que ceux-ci sont situés sur la frontière de D . Ce procédé invalide cependant la plupart des démonstrations des prochains résultats.

Enfin, on peut aussi considérer l'emploi de méthodes de centre analytique (Goffin et al., 1994). Ceci constitue cependant un changement majeur à la philosophie de l'algorithme.

Lemme 6.5.7 *Si $|K| < +\infty$ et si k_0 est l'indice du dernier pas de descente, alors il existe une constante $R < +\infty$ telle que*

$$\begin{aligned} 0 \leq \alpha(x^k, y^{k+1}) &\triangleq f(x^k) - \underline{f}(x^k; y^{k+1}) \\ &\leq R/\Psi_k^{\frac{1}{q}} \quad \forall k > k_0 \text{ tel que } y^{k+1} \in D, \end{aligned}$$

où

$$\underline{f}(y; y^{k+1}) \triangleq f(y^{k+1}) + \langle g^{k+1}, y - y^{k+1} \rangle, \quad \forall y \in \mathbb{R}^n$$

est la linéarisation de f en y^{k+1} ,

$$g^{k+1} = \sum_{i=1}^r g_i^{k+1} \in \partial f(y^{k+1})$$

et

$$g_i^{k+1} \in \partial f_i(y^{k+1}).$$

Démonstration:

Soit k_0 l'indice du dernier pas de descente. Soit $L \subseteq \{k_0 + 1, k_0 + 2, \dots\}$ l'ensemble des pas nuls tels que

$$k \in L \Rightarrow y^{k+1} \in D.$$

Comme il n'y a rien à montrer si $L = \emptyset$, alors nous avons $|L| > 0$. Pour tout $k \in L$, on a $x^k = x^{k_0+1}$ et

$$\begin{aligned} 0 \leq \alpha(x^k, y^{k+1}) &= \alpha(x^{k_0+1}, y^{k+1}) \\ &= f(x^{k_0+1}) - \underline{f}(x^{k_0+1}; y^{k+1}) \\ &= f(x^{k_0+1}) - \left(f(y^{k+1}) + \langle g^{k+1}, x^{k_0+1} - y^{k+1} \rangle \right) \\ &= f(x^{k_0+1}) - f(x^{k_0+1} + d^k) + \langle g^{k+1}, d^k \rangle \\ &\qquad\qquad\qquad \text{car } y^{k+1} = x^{k_0+1} + d^k \\ &\leq |f(x^{k_0+1}) - f(x^{k_0+1} + d^k)| + \|g^{k+1}\|^* \|d^k\| \\ &\qquad\qquad\qquad \text{par (B8).} \end{aligned}$$

L'ensemble L pouvant être de taille finie, nous supposons dans ce cas, afin de simplifier la démonstration, que son dernier élément est répété une infinité de fois: ceci nous permettra de travailler avec des ensembles infinis.

La suite $\{\Psi_k\}_L$ étant croissante, l'inégalité (6.31) du lemme 6.5.5 nous dit que la suite $\{\|d^k\|\}_L$ est bornée, ce qui signifie que les suites $\{d^k\}_L$ et $\{y^{k+1} = x^{k_0+1} + d^k\}_{k \in L} \subset D$ sont elles aussi bornées. Il existe alors, par le théorème B.4, une constante de Lipschitz $R_1 \geq 0$, valide sur un sous-ensemble convexe compact de D contenant la suite $\{y^{k+1}\}_{k \in L}$, telle que

$$|f(x^{k_0+1}) - f(x^{k_0+1} + d^k)| \leq R_1 \|d^k\|, \quad \forall k \in L.$$

La multi-application ∂f étant localement bornée (proposition B.6), la suite $\{\|g^{k+1}\|^*\}_L$ doit être bornée. Soit $R_2 > 0$ une constante positive bornant supérieurement cette suite. On a alors

$$\begin{aligned} 0 \leq \alpha(x^k, y^{k+1}) &\leq (R_1 + R_2)\|d^k\| \\ &\leq \frac{(R_1 + R_2)}{(\Psi_{\min})^{\frac{1}{q(q-1)}}} \left(\frac{\|g^{k_0+1}\|^*}{q} \right)^{\frac{1}{q-1}} \frac{1}{\Psi_k^{\frac{1}{q}}} \quad \text{par (6.31)} \\ &= R/\Psi_k^{\frac{1}{q}}, \quad \forall k \in L. \end{aligned}$$

■

Proposition 6.5.8 Si $x^k = x^{k_0+1} = \bar{x}$, $\forall k > k_0 \in K$, alors $\{w^k\} \nearrow 0$ et $\{\sigma^k\} \rightarrow 0$.

Démonstration:

Par les règles d'ajustement des poids Ψ_k et des estimés ε_v^k , on a

$$\Psi_{k+1} \geq \Psi_k, \quad \forall k \geq k_0 + 1$$

et

$$\varepsilon_v^{k+1} \leq \varepsilon_v^k, \quad \forall k \geq k_0 + 1.$$

De plus, par le lemme 6.5.4, on a aussi

$$0 \leq -w^{k+1} \leq -w^k, \quad \forall k \geq k_0 + 1.$$

Si $\Psi_k \nearrow +\infty$, alors $\|d^k\| \rightarrow 0$, par le lemme 6.5.5, d'où $d^k \rightarrow 0 \in \mathbb{R}^n$. Nous savons, par (5.401), que

$$\|p^k\|^* = \Psi_k q \|d^k\|^{q-1}, \quad \forall k$$

mais nous ne pouvons rien dire de $\lim \|p^k\|^*$ sans savoir si Ψ_k varie plus ou moins rapidement que $\|d^k\|$.

Comme dans la démonstration du lemme 6.5.7, définissons $L \subseteq \{k_0 + 1, k_0 + 2, \dots\}$ comme étant l'ensemble des indices des pas nuls réalisables, après le $k_0^{\text{ème}}$ et dernier pas de descente, i.e.

$$k \in L \Leftrightarrow y^{k+1} \in D \triangleq C \cap \text{dom } f.$$

Il découle du lemme 6.5.6 que $|L| = +\infty$.

Puisque $\{\Psi_k\} \rightarrow +\infty$, alors nous obtenons du lemme 6.5.7

$$\lim_{\substack{k \rightarrow \infty \\ k \in L}} \alpha(x^k, y^{k+1}) = 0,$$

de sorte que les règles de mise-à-jour de l'étape 8 de l'algorithme impliquent que la suite monotone $\{\varepsilon_v^k\}$ est convergente et a zéro pour limite:

$$\lim_{k \rightarrow \infty} \varepsilon_v^k = 0.$$

Il faut, par conséquent, que

$$\liminf_{k \rightarrow \infty} \{\|p^k\|^* + \tilde{\alpha}^k\} = 0, \quad \text{par (6.6)}$$

ce qui signifie que

$$\liminf_{k \rightarrow \infty} \{\sigma^k\} = 0, \quad \text{par (6.19a)}$$

et que $\{w^k\}_{k > k_0} \nearrow 0$, par (6.20) et la propriété de monotonie découlant de (6.29).

Supposons maintenant que $\{\Psi_k\}_{k > k_0} \uparrow \bar{\Psi} \in (0, +\infty)$.

La suite $\{w^k\}_{k > k_0}$ est, par le lemme 6.5.4, bornée et croissante. Elle est donc convergente (Rudin, 1976: théorème 3.14). Désignons sa limite par $\bar{w} \leq 0$. On doit

aussi avoir

$$\lim_{\substack{k \rightarrow \infty \\ k \in L}} w^k = \bar{w}$$

où L est tel que défini ci-haut.

Les inégalités (6.20) nous disent que la sous-suite $\{\sigma^k\}_L$ est bornée. Elle possède donc au moins une valeur d'adhérence que nous désignerons par $\bar{\sigma} \leq 0$. Il existe donc une sous-suite $\{\sigma^k\}_{L_1}$, $L_1 \subseteq L$, telle que $\{\sigma^k\}_{L_1} \rightarrow \bar{\sigma}$. De plus, on a

$$\bar{\sigma} \leq \bar{w} \leq 0.$$

Puisque

$$w^k = \sigma^k + \Psi_k \|d^k\|^q, \quad \forall k, \quad (6.4)$$

alors

$$\|d^k\| = \left(\frac{w^k - \sigma^k}{\Psi_k} \right)^{\frac{1}{q}}, \quad \forall k$$

et

$$\lim_{\substack{k \rightarrow \infty \\ k \in L_1}} \|d^k\| = \left(\frac{\bar{w} - \bar{\sigma}}{\bar{\Psi}} \right)^{\frac{1}{q}} \in (0, \infty).$$

La suite $\{\|d^k\|\}_{L_1}$ est bornée (Rudin, 1976: théorème 3.2c), ce qui implique que la suite $\{d^k\}_{L_1}$ est elle aussi bornée. Elle possède donc un point d'accumulation $\bar{d}$ et il existe $L_2 \subseteq L_1$ tel que

$$\lim_{\substack{k \rightarrow \infty \\ k \in L_2}} d^k = \bar{d}.$$

Par conséquent,

$$\lim_{\substack{k \rightarrow \infty \\ k \in L_2}} y^{k+1} = \bar{x} + \bar{d} \quad \text{car } y^{k+1} = \bar{x} + d^k, \forall k > k_0.$$

La sous-suite $\{y^{k+1}\}_{k \in L_2}$ étant convergente, elle est bornée, elle est une suite de Cauchy, et on doit avoir

$$\lim_{\ell \rightarrow \infty} \|y^{k(\ell)+1} - y^{k(\ell+1)+1}\| = 0 \quad (6.32)$$

où $k(\ell)$ désigne le $\ell^{\text{ème}}$ élément de la liste L_2 .

Définissons $\varepsilon^{k(\ell)}$, $\ell \in \{1, 2, 3, \dots\}$, par

$$\varepsilon^{k(\ell)} \triangleq \varphi^{k(\ell)}(y^{k(\ell)+1}) - \widehat{\varphi}_a^{k(\ell)}(y^{k(\ell)+1}) \geq 0$$

et $g^{k(\ell)+1}$ avec la définition (6.30).

Par conséquent,

$$\begin{aligned} \varepsilon^{k(\ell)} &= \underline{f}(y^{k(\ell)+1}; y^{k(\ell)+1}) - f(x^{k_0}) - \widehat{\varphi}_a^{k(\ell)}(y^{k(\ell)+1}) \\ &= \underline{f}(y^{k(\ell+1)+1}; y^{k(\ell)+1}) - f(x^{k_0}) - \widehat{\varphi}_a^{k(\ell)}(y^{k(\ell)+1}) \\ &\quad - \langle g^{k(\ell)+1}, y^{k(\ell+1)+1} - y^{k(\ell)+1} \rangle \\ &\leq \widehat{f}_a^{k(\ell+1)}(y^{k(\ell+1)+1}) - f(x^{k_0}) - \widehat{\varphi}_a^{k(\ell)}(y^{k(\ell)+1}) \\ &\quad - \langle g^{k(\ell)+1}, y^{k(\ell+1)+1} - y^{k(\ell)+1} \rangle \\ &\quad \text{car, par l'étape 6 de l'algorithme, on a} \\ &\quad k' = k(\ell) + 1 \in J_{f_i}^{k(\ell)+1}, \forall i \in \{1, \dots, r\} \\ &\leq \widehat{\varphi}_a^{k(\ell)+1}(y^{k(\ell+1)+1}) - \widehat{\varphi}_a^{k(\ell)}(y^{k(\ell)+1}) + \|g^{k(\ell)+1}\|^* \|y^{k(\ell+1)+1} - y^{k(\ell)+1}\| \\ &\quad \text{par (B8)} \\ &= w^{k(\ell+1)} - \Psi_{k(\ell+1)} \|y^{k(\ell+1)+1} - x^{k_0}\|^q \\ &\quad - w^{k(\ell)} + \Psi_{k(\ell)} \|y^{k(\ell)+1} - x^{k_0}\|^q + \|g^{k(\ell)+1}\|^* \|y^{k(\ell+1)+1} - y^{k(\ell)+1}\|. \end{aligned}$$

On a alors

$$\lim_{\ell \rightarrow \infty} \varepsilon^{k(\ell)} = 0$$

car

$$\lim_{\ell \rightarrow \infty} w^{k(\ell)} = \bar{w}$$

$$\lim_{\ell \rightarrow \infty} y^{k(\ell)+1} = \bar{y}$$

$$\lim_{k \rightarrow \infty} \Psi_k = \bar{\Psi}$$

et par (6.32) et la proposition B.6 qui nous dit que les sous-gradients sont localement bornés. Le test (6.5) n'étant jamais satisfait, $\forall \ell$, on doit avoir

$$f(y^{k(\ell)+1}) - f(x^{k(\ell)}) > m\sigma^{k(\ell)}, \forall \ell \in \{1, 2, 3, \dots\}.$$

Puisque

$$\begin{aligned} \varepsilon^{k(\ell)} &= \varphi^{k(\ell)}(y^{k(\ell)+1}) - \varphi^{k(\ell)}(x^{k(\ell)}) - \left[\widehat{\varphi}_a^{k(\ell)}(y^{k(\ell)+1}) - \varphi^{k(\ell)}(x^{k(\ell)}) \right] \\ &= f(y^{k(\ell)+1}) - f(x^{k(\ell)}) - \widehat{\varphi}_a^{k(\ell)}(y^{k(\ell)+1}) \quad \text{car } x^{k(\ell)} = x^{k_0} \in D, \quad \forall \ell \\ &> m\sigma^{k(\ell)} - \sigma^{k(\ell)} \\ &= (m - 1)\sigma^{k(\ell)} \\ &= (1 - m)|\sigma^{k(\ell)}| \end{aligned}$$

et que $m \in (0, 1)$, alors

$$\lim_{\ell \rightarrow \infty} \varepsilon^{k(\ell)} = 0 \quad \Rightarrow \quad \lim_{\ell \rightarrow \infty} \sigma^{k(\ell)} = 0.$$

Mais, par (6.20), ceci entraîne que

$$\lim_{\ell \rightarrow \infty} w^{k(\ell)} = 0,$$

d'où, par la monotonie de $\{w^k\}$,

$$\lim_{k \rightarrow \infty} w^k = 0.$$

Par conséquent, que la suite $\{\Psi_k\}$ diverge à l'infini ou ait $\bar{\Psi} \in (0, \infty)$ pour limite, on a $\{w^k\} \nearrow 0$. ■

Remarque: Si les poids étaient tels que $\{\Psi_k\} \nearrow +\infty$ lorsque l'algorithme effectue une infinité de pas nuls consécutifs, il ne serait pas nécessaire, à l'étape 6 de

l'algorithme, d'imposer la condition

$$k' \in J_{f_i}^{k+1}, \quad \forall i \in \{1, \dots, r\}.$$

Le problème primal (5.64) est alors de plus petite dimension.

Les propositions 6.5.3 et 6.5.8 prouvent la convergence de l'algorithme lorsque la fonction économique est inf-compacte.

Théorème 6.5.9 *Si la fonction économique est inf-compacte, alors la suite $\{x^k\}$ possède au moins un point d'accumulation et chacun de ceux-ci est une solution optimale du problème (6.1).*

Remarque: L'inf-compactité n'intervient que lorsque l'algorithme effectue une infinité de pas de descente.

Nous avons vu à la proposition 6.5.8 que $\{w^k\} \nearrow 0$ et $\sigma^k \rightarrow 0$ lorsque le nombre de pas de descente est fini. Nous avons aussi vu au lemme 6.5.2 que $\{\sigma^k\}_K \rightarrow 0$ lorsqu'il y a une infinité de pas de descente. Dans ce cas, la démonstration de la proposition 6.5.3 nous a appris que $\{w^k\}_K \rightarrow 0$. Il serait intéressant, dans ce dernier cas, de pouvoir en dire un peu plus de la suite $\{w^k\}$, c'est-à-dire d'affirmer que la suite complète est convergente.

Le lemme 6.5.4 nous a aussi appris que

$$0 \geq w^\ell \geq w^k$$

$\forall \ell, k$ tels que $k_1 < k \leq \ell \leq k_2$, où k_1 et k_2 sont les indices de deux pas de descente consécutifs. Malheureusement, ces résultats ne nous disent pas ce qui se

passé immédiatement après un pas de descente. Si nous pouvions montrer que $\lim_{\substack{k \rightarrow \infty \\ k \in K}} w^{k+1} = 0$, ceci prouverait aussi que la suite $\{w^k\}$ est convergente et a zéro pour limite.

Lemme 6.5.10 *Si l'algorithme effectue une infinité de pas de descente, alors $\{w^k\} \rightarrow 0$.*

Démonstration:

Le résultat est évident si l'algorithme n'effectue aucun pas nul. Supposons donc qu'il existe $k \in K$ tel que $k+1 \notin K$.

Le lemme 6.4.6 nous permet de définir la linéarisation $\tilde{\varphi}_a^k$ de $\hat{\varphi}_a^k$ et de $\hat{\varphi}_{ap}^k$ en x^{k+1} :

$$\tilde{\varphi}_a^k(x) \triangleq \hat{\varphi}_a^k(x^{k+1}) + \langle p^k, x - x^{k+1} \rangle \quad (6.33)$$

car $x^{k+1} = y^{k+1} \triangleq x^k + d^k$ puisque $k \in K$. On peut alors écrire

$$\tilde{\varphi}_a^k(x) \leq \hat{\varphi}_{ap}^k(x), \quad \forall x \in \mathbb{R}^n,$$

avec égalité lorsque $x = x^{k+1}$.

De plus, on doit avoir

$$\hat{\varphi}_a^{k+1}(x) \geq \hat{\varphi}_{ap}^k(x), \quad \forall x \in \mathbb{R}^n$$

car $\hat{\varphi}_a^{k+1}$ est obtenu en ajoutant les linéarisations engendrées en x^{k+1} à celles définissant $\hat{\varphi}_{ap}^k$. Par conséquent, si $k \in K$, on a

$$\tilde{\varphi}_a^k(x) \leq \hat{\varphi}_{ap}^k(x) \leq \hat{\varphi}_a^{k+1}(x), \quad \forall x \in \mathbb{R}^n, \quad (6.34)$$

et on peut écrire

$$\begin{aligned}
w^{k+1} &= \min\{\varphi_a^{k+1}(x) + \Psi_{k+1}\|x - x^{k+1}\|^q : x \in \mathbb{R}^n\} && \text{par (5.66a)} \\
&\geq \min\{\tilde{\varphi}_a^k(x) + \Psi_{k+1}\|x - x^{k+1}\|^q : x \in \mathbb{R}^n\} && \text{par (6.34)} \\
&= \min\{\tilde{\varphi}_a^k(x^{k+1}) + \langle p^k, x - x^{k+1} \rangle + \Psi_{k+1}\|x - x^{k+1}\|^q : x \in \mathbb{R}^n\} \\
&&& \text{par (6.33)} \\
&= \tilde{\varphi}_a^k(x^{k+1}) + \langle p^k, x^* - x^{k+1} \rangle + \Psi_{k+1}\|x^* - x^{k+1}\|^q
\end{aligned}$$

où $x^* \in \mathbb{R}^n$ est tel que

$$\begin{aligned}
0 &\in \partial\left(\langle p^k, \cdot - x^{k+1} \rangle + \Psi_{k+1}\|\cdot - x^{k+1}\|^q\right)(x^*) \\
&= p^k + q\Psi_{k+1}\|x^* - x^{k+1}\|^{q-1}\partial\|x^* - x^{k+1}\|.
\end{aligned}$$

Il existe $\nu^* \in \partial\|x^* - x^{k+1}\|$ tel que

$$p^k + q\Psi_{k+1}\|x^* - x^{k+1}\|^{q-1}\nu^* = 0,$$

d'où

$$\nu^* = \frac{-p^k}{q\Psi_{k+1}\|x^* - x^{k+1}\|^{q-1}}.$$

De plus, par la proposition B.26, on a

$$\begin{aligned}
\|x^* - x^{k+1}\| &= \langle \nu^*, x^* - x^{k+1} \rangle \\
&= \frac{-\langle p^k, x^* - x^{k+1} \rangle}{q\Psi_{k+1}\|x^* - x^{k+1}\|^{q-1}}
\end{aligned}$$

de sorte que

$$\langle p^k, x^* - x^{k+1} \rangle = -q\Psi_{k+1}\|x^* - x^{k+1}\|^q.$$

Par conséquent,

$$\begin{aligned}
w^{k+1} &\geq \tilde{\varphi}_a^k(x^{k+1}) - q\Psi_{k+1}\|x^* - x^{k+1}\|^q + \Psi_{k+1}\|x^* - x^{k+1}\|^q \\
&= \tilde{\varphi}_a^k(x^{k+1}) - \Psi_{k+1}(q-1)\|x^* - x^{k+1}\|^q
\end{aligned}$$

$$\begin{aligned}
&= \tilde{\varphi}_a^k(x^{k+1}) - \Psi_{k+1}(q-1) \|d^*\|^q && \text{où } d^* = x^* - x^{k+1} \\
&= \tilde{\varphi}_a^k(x^{k+1}) - \frac{\Psi_{k+1}(q-1)}{(\Psi_k q)^{\frac{q}{q-1}}} (\|p^k\|^*)^{\frac{q}{q-1}} && \text{par (5.401)} \\
&\geq \tilde{\varphi}_a^k(x^{k+1}) - \frac{1}{(q\Psi_k)^{\frac{1}{q-1}}} (\|p^k\|^*)^{\frac{q}{q-1}} && \text{car } \Psi_{k+1} \leq \Psi_k \\
&= -\tilde{\varphi}_a^k(x^k) + \tilde{\varphi}_a^k(x^{k+1}) + \tilde{\varphi}_a^k(x^k) - \frac{1}{(q\Psi_k)^{\frac{1}{q-1}}} (\|p^k\|^*)^{\frac{q}{q-1}}.
\end{aligned}$$

Mais

$$\begin{aligned}
&-\tilde{\varphi}_a^k(x^k) + \tilde{\varphi}_a^k(x^{k+1}) = \langle p^k, x^{k+1} - x^k \rangle = \langle p^k, d^k \rangle, \\
&\tilde{\varphi}_a^k(x^k) = -\tilde{\alpha}^k && \text{par (6.33), (5.67) et (6.15)} \\
&\text{et } -\frac{1}{(q\Psi_k)^{\frac{1}{q-1}}} (\|p^k\|^*)^{\frac{q}{q-1}} = \sigma^k + \tilde{\alpha}^k && \text{par (6.19a) .}
\end{aligned}$$

de sorte que

$$\begin{aligned}
0 \geq w^{k+1} &\geq \langle p^k, d^k \rangle - \tilde{\alpha}^k + \sigma^k + \tilde{\alpha}^k \\
&= 2\sigma^k + \tilde{\alpha}^k && \text{par le lemme 5.9.11} \\
&\geq 2\sigma^k && \text{car } \tilde{\alpha}^k \geq 0.
\end{aligned}$$

On a donc

$$0 \geq \lim_{\substack{k \rightarrow \infty \\ k \in K}} w^{k+1} \geq 2 \lim_{\substack{k \rightarrow \infty \\ k \in K}} \sigma^k = 0,$$

ce qui prouve que $\{w^k\} \rightarrow 0$. ■

Ce dernier résultat, ainsi que la discussion qui le précède, démontrent que, quel que soit le nombre de pas de descente, la suite $\{w^k\}$ est convergente et a zéro pour limite. Ceci nous assure également que le test d'arrêt de l'étape 2 est vérifié après un nombre fini d'itérations lorsque $\varepsilon_S > 0$.

Proposition 6.5.11 *La suite $\{w^k\}$ a zéro pour limite et l'algorithme s'arrête après un nombre fini d'itérations lorsque $\varepsilon_S > 0$.*

6.6 L'algorithme pour les stratégies d'accumulation et d'accumulation/épuration

On peut, dans l'algorithme de la section 6.2, remplacer la stratégie d'agrégation par la stratégie d'accumulation ou la stratégie d'accumulation/épuration. Pour cela, il suffit de remplacer les étapes 1 et 6 par:

(1') (RECHERCHE D'UNE DIRECTION) Évaluer des multiplicateurs (π^k, μ^k, ω^k) solution optimale du problème (5.34):

$$\begin{aligned} \underset{\pi, \mu, \omega, p}{\text{maximiser}} \quad & \sum_{i=1}^r \sum_{j \in J_{fi}^k} \pi_{ij} [f_{ij}^k - f_i(x^k)] - \sum_{i=1}^r \sum_{j \in J_{ci}^k} \mu_{ij} b_{ij}^k - \sum_{j \in J_c^k} \omega_j b_j^k \\ & + \frac{1}{q(\Psi_k)^{\frac{1}{q-1}}} \left(\frac{1}{q} - 1 \right) (\|p\|^*)^{\frac{q}{q-1}} \\ \text{sous les contraintes} \quad & \end{aligned} \tag{5.34}$$

$$\begin{aligned} \sum_{j \in J_{fi}^k} \pi_{ij} &= 1, & i &\in \{1, \dots, r\} \\ \sum_{i=1}^r \sum_{j \in J_{ci}^k} \mu_{ij} + \sum_{j \in J_c^k} \omega_j &\leq 1 \\ p - \sum_{i=1}^r \sum_{j \in J_{fi}^k} \pi_{ij} g_i^j - \sum_{i=1}^r \sum_{j \in J_{ci}^k} \mu_{ij} a_i^j - \sum_{j \in J_c^k} \omega_j a_j^j &= 0 \\ \pi_{ij} &\geq 0, \quad j \in J_{fi}^k, & i &\in \{1, \dots, r\} \\ \mu_{ij} &\geq 0, \quad j \in J_{ci}^k, & i &\in \{1, \dots, r\} \\ \omega_j &\geq 0, \quad j \in J_c^k. \end{aligned}$$

Déterminer les ensembles $\hat{J}_{fi}^k, \hat{J}_{ci}^k, i \in \{1, \dots, r\}$, et $\hat{J}_c^k$ de valeurs duales strictement positives définis par les expressions (5.44a)–(5.44c). Évaluer les constantes $\nu_{ci}^k, i \in \{1, \dots, r\}$, et ν_c^k ainsi que des multiplicateurs normalisés $(\bar{\mu}^k, \bar{\omega}^k)$ vérifiant les expressions (5.51a)–(5.51c) et (5.53a)–(5.53f). Évaluer les quantités

$$(p_{fi}^k, \tilde{f}_{pi}^k), \quad i \in \{1, \dots, r\}$$

$$(p_{ci}^k, \tilde{b}_{pi}^k), \quad i \in \{1, \dots, r\}$$

et

$$(p_c^k, \tilde{b}_p^k)$$

avec les expressions (5.54a)–(5.54c).

Évaluer p^k, σ^k et $\tilde{\alpha}^k$ à l'aide des expressions (5.55)–(5.57) et utiliser (5.40l) pour obtenir d^k , i.e.

$$d^k \in \text{zéro} \left[\Psi_k q \|d\|^{q-1} \partial \|\cdot\| (d) + p^k \right] \quad (5.40l)$$

(6') (GESTION DU FAISCEAU) Choisir $J_{fi}^{k+1}, J_{ci}^{k+1}, i \in \{1, \dots, r\}$, et J_c^{k+1} de façon à ce que

$$\hat{J}_{fi}^k \subseteq J_{fi}^{k+1} \subseteq J_{fi}^k, \quad k' \in J_{fi}^{k+1}, \quad i \in \{1, \dots, r\}$$

$$\hat{J}_{ci}^k \subseteq J_{ci}^{k+1} \subseteq J_{ci}^k, \quad i \in \{1, \dots, r\}$$

et

$$\hat{J}_c^k \subseteq J_c^{k+1} \subseteq J_c^k.$$

Remarque: Le vecteur p^k évalué à l'étape 1' avec l'expression (5.55) est la valeur optimale de la variable vectorielle p du problème (5.34).

Remarque: Si, à l'étape 6', on choisit

$$\begin{aligned} J_{fi}^{k+1} &= \hat{J}_{fi}^k \cup \{k'\}, \quad \forall i \in \{1, \dots, r\}, \quad \forall k \\ J_{ci}^{k+1} &= \hat{J}_{ci}^k, \quad \forall i \in \{1, \dots, r\}, \quad \forall k \\ J_c^{k+1} &= \hat{J}_c^k, \quad \forall k \end{aligned}$$

on a alors la stratégie d'accumulation/épuration.

Par contre, on obtient la stratégie d'accumulation lorsque l'on a

$$\begin{aligned} J_{fi}^{k+1} &= J_{fi}^k, \quad \forall i \in \{1, \dots, r\}, \quad \forall k \\ J_{ci}^{k+1} &= J_{ci}^k, \quad \forall i \in \{1, \dots, r\}, \quad \forall k \\ J_c^{k+1} &= J_c^k, \quad \forall k. \end{aligned}$$

L'algorithme permet toutes les stratégies intermédiaires possibles.

Les énoncés de tous les résultats de ce chapitre, à l'exception du lemme 6.4.6 qui a besoin d'être modifié, demeurent vrais sans aucun changement pour les stratégies d'accumulation et d'accumulation/épuration. Il est toutefois nécessaire de modifier trivialement certaines démonstrations. Nous effectuerons ci-dessous une très rapide revue des principaux changements à apporter aux résultats et démonstrations précédents.

Le lemme 6.4.1 découle directement des définitions (5.54a)–(5.54c) lorsque l'on définit le vecteur de poids $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k)$ par,

$$\begin{aligned} \bar{\pi}_{ij}^k &= \begin{cases} \pi_{ij}^k & \text{si } j \in J_{fi}^k, \\ 0 & \text{si } j \in J_{fi}^1 \cup \dots \cup J_{fi}^{k-1} \setminus J_{fi}^k \end{cases} \quad i \in \{1, \dots, r\} \\ \bar{\mu}_{ij}^k &= \begin{cases} \tilde{\mu}_{ij}^k & \text{si } j \in J_{ci}^k \\ 0 & \text{si } j \in J_{ci}^1 \cup \dots \cup J_{ci}^{k-1} \setminus J_{ci}^k, \end{cases} \quad i \in \{1, \dots, r\} \end{aligned}$$

et

$$\bar{\omega}_j^k = \begin{cases} \bar{\omega}_j^k & \text{si } j \in J_c^k \\ 0 & \text{si } j \in J_c^1 \cup \dots \cup J_c^{k-1} \setminus J_c^k. \end{cases}$$

On peut remplacer le lemme 6.4.6 par le

Lemme 6.6.1 *Le vecteur p^k vérifie*

$$p^k \in \partial \hat{\varphi}_e^k(x^k + d^k). \quad (6.35a)$$

De plus, si $\hat{\varphi}_{ep}^k$ désigne la fonction associée au problème (5.47), i.e. résultant de l'épuration a posteriori (s'il y en a) du problème (5.22a)-(5.22f) associé à la fonction $\hat{\varphi}_e^k$, alors

$$\hat{\varphi}_{ep}^k(x^k + d^k) = \hat{\varphi}_e^k(x^k + d^k) \quad (6.35b)$$

et

$$p^k \in \partial \hat{\varphi}_{ep}^k(x^k + d^k). \quad (6.35c)$$

Démonstration:

Semblable à celle du lemme 6.4.6. Il faut remplacer les mentions aux problèmes (5.66a) et (5.64) par des mentions aux problèmes (5.50) et (5.22a)-(5.22f).

Remarquons aussi qu'étant donné que les problèmes (5.64) et (5.22a)-(5.22f) ont la même structure, l'expression (5.67) est toujours vraie si l'on substitue $\hat{\varphi}_e^k(x^k + d^k)$ à $\hat{\varphi}_a^k(x^k + d^k)$. Ceci permet alors de conclure que $p^k \in \partial \hat{\varphi}_e^k(x^k + d^k)$.

L'égalité (6.35b) découle de la dernière partie du lemme 5.8.5 tandis que (6.35c) découle de (6.35a) en remarquant que le problème équivalent (5.47) contient toutes

les contraintes du problème (5.22a)–(5.22f) ayant un multiplicateur de Lagrange non nul. ■

Par conséquent, les commentaires immédiatement avant et après le lemme 6.4.6 demeurent valides si l'on remplace $\hat{\varphi}_a^k$ par $\hat{\varphi}_e^k$.

Il faut modifier trivialement la démonstration du lemme 6.5.4 en remplaçant les mentions à des résultats de la section 5.9 par des mentions aux résultats correspondants de la section 5.8. Il faut évidemment changer les fonctions $\hat{\varphi}_a^k$, $\hat{f}_{ia}^k$ et $\hat{I}_{D_a}^k$ par $\hat{\varphi}_e^k$, $\hat{f}_{ie}^k$ et $\hat{I}_{D_e}^k$.

Enfin, il faut modifier la démonstration du lemme 6.5.10 comme suit. Il faut d'abord remplacer la définition (6.33) de la fonction $\hat{\varphi}_a^k$ par

$$\tilde{\varphi}_e^k(x) \triangleq \hat{\varphi}_e^k(x^{k+1}) + \langle p^k, x - x^{k+1} \rangle. \quad (6.36)$$

En utilisant l'argumentation du début de la démonstration, on prouve que

$$\tilde{\varphi}_e^k(x) \leq \hat{\varphi}_{ep}^k(x) \leq \hat{\varphi}_e^{k+1}(x), \quad \forall x \in \mathbb{R}^n, \quad (6.37)$$

ce qui permet de remplacer les inégalités (6.34) par (6.37). Il suffit, par la suite, de remplacer les fonctions $\hat{\varphi}_a^{k+1}$, $\tilde{\varphi}_a^k$ et $\hat{\varphi}_a^k$ par $\hat{\varphi}_e^{k+1}$, $\tilde{\varphi}_e^k$ et $\hat{\varphi}_e^k$.

CHAPITRE 7

Expérimentation numérique

7.1 Introduction

Nous avons présenté au chapitre 2 un problème de programmation linéaire de grande taille que nous avons reformulé comme un problème équivalent de programmation convexe non-différentiable consistant à minimiser une somme de fonctions polyédriques convexes sur son domaine. Ceci nous a amené à nous intéresser au problème plus général de la minimisation d'une somme de fonctions convexes (non nécessairement polyédriques) sur un sous-ensemble convexe du domaine de la fonction résultante que l'on est assez naturellement porté à résoudre à l'aide de méthodes de coupes. Les chapitres qui ont suivi ont présenté les concepts de stabilisation des algorithmes de plans sécants et leur mise en oeuvre dans le cadre de nouveaux algorithmes de la famille des méthodes de faisceaux.

Le présent chapitre est une étude préliminaire du comportement numérique de ces algorithmes appliqués à la résolution du problème de programmation linéaire (2.1) du chapitre 2. Pour ce faire, il a fallu concevoir le programme CMinSfpc et engendrer une série de problèmes tests. Celui-ci utilise les résultats de la section 4.6 pour produire les sous-gradients et coupes requises par l'algorithme. Remarquons dès maintenant que le problème équivalent (2.4) ne comporte pas d'ensemble C restreignant le domaine de l'optimisation, d'où l'inutilité de l'oracle qui y est associé. Contrairement à d'autres études (dont celle de Medhi (1987)) nous avons choisi de

ne traiter que des problèmes réels.

Les sections 7.2 et 7.3 décrivent brièvement le programme CMinSfpc et les problèmes tests tandis que les suivantes décrivent et interprètent les résultats de nos tests numériques.

7.2 Le programme CMinSfpc

Le programme CMinSfpc est un programme C, exploitant la modularité de l'optimiseur de programmation linéaire Cplex (1992), résolvant le problème (2.1) à l'aide d'une variante de l'algorithme du chapitre 6 appliquée au problème équivalent (2.4). Il minimise donc une somme de fonctions polyédriques convexes, d'où son nom.

Nous avons vu au chapitre 5 deux approches qui s'offraient à nous pour résoudre le problème (5.1), soit l'approche des régions de confiance et celle des pénalisations. L'approche des régions de confiance est bien adaptée au cas des normes ℓ_1 et ℓ_∞ puisque les problèmes de recherche d'une direction d'amélioration en sont alors de programmation linéaire. Par contre, l'approche est difficile à mettre en oeuvre lorsque l'on utilise d'autres normes car on obtient alors une contrainte non linéaire que l'on est tenté de dualiser par passage à la fonction économique. Mieux vaut, dans ce cas, recourir à la seconde approche, soit l'approche des pénalisations que nous avons développée au chapitre 6. C'est celle qui permet de traiter plus facilement le plus grand nombre de normes.

Cependant, dans la pratique, on est le plus souvent intéressé par un nombre restreint de normes, soit les normes ℓ_∞ , ℓ_1 et ℓ_2 (ou plus généralement par des normes quadratiques). Ceci semble favoriser l'utilisation de l'approche des régions de confiance puisque deux de ces trois normes peuvent être facilement utilisées sous cette approche. A cela, s'ajoute la rareté de codes robustes et performants d'optimisation quadratique adaptés au traitement d'un petit nombre de contraintes de convexité (Kiwiel, 1989b et 1994) qui ne sont malheureusement pas disponibles.

Pour ces raisons et pour des contraintes de temps, il a été décidé de reporter à plus tard l'utilisation des normes quadratiques. Le programme CMinSfpc est donc basé sur l'utilisation de régions de confiance. Il recherche une direction d'amélioration à l'aide du problème (5.16a)–(5.16g), ou de son dual (5.20a)–(5.20f). L'algorithme mis en oeuvre est une adaptation de l'algorithme du chapitre 6 exploitant l'équivalence des deux approches mise en évidence par la proposition 5.5.2. (En fait, là où l'algorithme du chapitre 6 décide d'augmenter la valeur du paramètre de pénalisation, CMinSfpc fait décroître le “rayon” de la région de confiance, et vice-versa). C'est ce qui était entendu par l'expression “variante de l'algorithme du chapitre 6” du début de cette section.

Outre le choix entre les normes ℓ_1 et ℓ_∞ dans le problème primal de recherche d'une direction, le programme CMinSfpc offre le choix entre les stratégies d'accumulation et d'accumulation/épuration de l'information. Des paramètres permettent de contrôler la manière dont les poids (ou plutôt les “rayons” des régions de confiance) varient, et l'utilisateur peut choisir entre une approche de “pénalisations” dans laquelle des variables sont ajoutées aux contraintes liantes de chacun des sous-problèmes afin de ne manipuler que des fonctions à valeurs finies (voir Minoux (1983), section 8.3), et l'approche des plans coupants du chapitre 6 où des plans sont engendrés

afin de ramener les solutions du problème maître dans le domaine des fonctions. Il faudra comprendre, dans la suite de ce chapitre, que l'expression "approche des pénalisations" sera utilisée dans ce sens plutôt que dans celui des chapitres 5 et 6. Deux procédures d'enrichissement local des modèles des fonctions sont possibles. De plus, le programme offre à l'utilisateur la possibilité de résoudre son problème à l'aide de la méthode de Benders (1962) classique. Ces diverses options seront comparées dans les prochaines sections.

7.3 Les problèmes tests

Nous avons décidé de ne tester l'algorithme qu'avec des problèmes réels. Le modèle d'échanges d'électricité entre le Québec et l'état de New-York de Berger, Dubois, Lessard, Haurie et Loulou (1990) a servi à construire les problèmes utilisés pour les tests. Il s'agit d'un modèle de programmation linéaire de long terme pouvant compter 9 périodes de 5 ans qui a servi à étudier les gains résultants de la coopération entre les deux régions sous divers scénarios d'interconnexion entre leurs réseaux électriques. Ce modèle est engendré avec la version canadienne (Berger *et al.*, 1992) du logiciel MARKAL (Fishbone et Abilock, 1981; Fishbone *et al.*, 1983), un outil d'analyse à long terme de systèmes énergétiques et environnementaux régionaux, né d'un projet de l'Agence Internationale de l'énergie.

La flexibilité de MARKAL permet d'engendrer facilement des problèmes comportant de 1 à 9 périodes. Cette flexibilité a été mise à profit pour engendrer les problèmes QCNY1 à QCNY9 dont quelques caractéristiques apparaissent au tableau 7.1. La dernière colonne de ce tableau présente les temps d'optimisation par Cplex

pour le problème non-décomposé original.

Tableau 7.1 Quelques caractéristiques des problèmes tests

Problème	Contraintes	Variables	Coefficients non nuls	Densité	Valeur optimale	Temps UCT (secondes)
QCNY1	280	283	904	1,141 %	25796,536	0.6
QCNY2	599	621	2031	0,546 %	44725,399	2.5
QCNY3	1006	1072	3598	0,334 %	58828,882	6.9
QCNY4	1655	1756	6351	0,219 %	70328,270	19.0
QCNY5	2331	2473	9328	0,162 %	79718,341	39.0
QCNY6	3033	3220	12518	0,128 %	87675,394	62.2
QCNY7	3735	3967	15772	0,106 %	94155,292	119.3
QCNY8	4437	4714	19085	0,091 %	99365,886	159.1
QCNY9	5135	5461	22451	0,080 %	103490,735	234.5

Chacun de ces 9 problèmes peut être décomposé d'au moins trois manières. On peut d'abord considérer la décomposition proposée au chapitre 2 pour laquelle il y a un sous-problème pour chacune des deux régions et un pour coordonner les échanges. Puisque, pour ces problèmes, le sous-problème des échanges est de petite taille, on peut le fusionner à l'un ou l'autre des deux sous-problèmes régionaux, ce qui fournit les deux autres décompositions.

Nous avons par conséquent identifié trois problèmes équivalents d'optimisation convexe non-différentiable pour chacun des 9 problèmes de programmation linéaire du tableau 7.1. Nous désignerons par $DQCNY_i$ le problème équivalent au problème $QCNY_i$, $i \in \{1, \dots, 9\}$, où le modèle des échanges constitue un sous-problème distinct, et par $DQCNY_i\_QC$ et $DQCNY_i\_NY$ ceux où le problème des échanges est fusionné au problème du Québec ou à celui de l'état de New-York.

Ceci fournit donc 27 problèmes de programmation convexe pouvant être traités par CMinSfpc. En pratique, nous ne les avons pas tous testés. A mesure que

des problèmes de plus grande taille étaient considérés, il était décidé de poursuivre ou non les tests avec certaines formulations, de sorte que seuls les problèmes DQCNYi_QC ont tous été traités. Le tableau 7.2 indique, pour chacune des formulations, la taille (c'est-à-dire le nombre de variables s_i) du problème équivalent (non-différentiable).

Tableau 7.2 Nombre de variables des problèmes équivalents

Nombre de périodes (i)	DQCNYi	DQCNYi_QC	DQCNYi_NY
1	16	8	8
2	32	16	16
3	48	24	24
4	70	35	35
5	92	46	46
6	114	57	57
7	136	68	68
8	158	79	79
9	180	90	90

Le reste de ce chapitre est consacré aux tests effectués avec CMinSfpc. Ce programme fait appel à la version 2.1 de Cplex et tous les tests ont été effectués sur une station de travail Sun-Sparc-Station 10/30 disposant de 32 méga-octets de mémoire vive et d'une puissance de calcul de 86 Mips. Les temps UCT (unité centrale de traitement) d'optimisation qui seront donnés excluent les opérations de lecture et de chargement en mémoire des problèmes traités.

La plupart des tests ont été effectués en fixant la constante d'arrêt ε_S de l'algorithme à 0,025, ce qui, en la comparant aux valeurs optimales du tableau 7.1, correspond à des erreurs relatives comprises entre 0,000024% et 0,000097%. Ceci permet d'étudier le comportement asymptotique des algorithmes testés. Remarquons qu'il est très rare d'exiger de telles précisions d'algorithmes de décomposition. L'annexe D contient l'ensemble des résultats.

7.4 L'approche des fonctions à valeurs finies

Lorsque l'utilisateur le désire, le programme CMinSfpc ajoute une variable de surplus et une variable de déficit à chacune des contraintes liantes de chacun des sous-problèmes. De cette façon, ils sont toujours réalisables, les fonctions $v_i(\cdot)$ définies par les problèmes (2.5) et (2.6) sont à valeurs finies et la contrainte (5.20c) du dual du problème de recherche d'une direction est alors superflue. Cette option a été utilisée dans cette section et le coût unitaire de chacune des variables artificielles ajoutées a été pris égal à 10000. A moins d'indication contraire, c'est la stratégie d'accumulation des contraintes qui est utilisée.

7.4.1 Rayons invariants

Nous avons commencé par étudier le comportement de l'algorithme lorsque le "rayon" γ de la région de confiance est fixé à une valeur qui ne change pas.

Le tableau D.1 présente les résultats de deux groupes d'optimisations faites sur le problème (à trois périodes) DQCNY3: l'un où la norme ℓ_∞ est utilisée dans la fonction économique (5.20a) et l'autre où c'est la norme ℓ_1 qui est utilisée. Dans les deux cas, nous avons effectué 18 optimisations avec des valeurs différentes du paramètre γ .

On remarque, comme prévu, que pour chacune des normes, les performances (i.e. nombre d'itérations, colonnes engendrées, ou temps d'optimisation) de l'algorithme se dégradent lorsque la valeur de γ est trop petite ou trop grande. En fait,

pour chaque norme, il existe un intervalle de valeurs de γ produisant les meilleures performances. Cet intervalle est beaucoup plus étroit dans le cas de la norme du maximum (i.e. ℓ_∞).

Nous avons refait cet exercice pour des formulations correspondant à des problèmes comportant de 1 à 5 périodes et ce comportement a toujours été observé. Marsten, Hogan et Blankenship (1975) l'ont aussi remarqué dans le cadre de tests effectués avec la méthode BOXSTEP, qui utilise la norme ℓ_∞ , sur un problème de multifiots de petite taille.

Des valeurs inférieures à celles de l'intervalle "optimal" se traduisent par un accroissement du nombre d'itérations et du temps d'optimisation. On remarque toutefois que le nombre de colonnes engendrées ne croît pas aussi rapidement. Ceci est dû au fait que les bases optimales des sous-problèmes ne changent pas à toutes les itérations de l'algorithme, d'où l'inutilité de réengendrer des colonnes déjà existantes. Dans ce cas, plusieurs itérations peuvent être effectuées avant que l'algorithme quitte la région de validité de la base optimale d'un sous-problème et qu'une nouvelle colonne soit introduite dans le problème maître, i.e. de recherche d'une direction d'amélioration. Le ratio des pas de descente par itération est alors élevé (70% et plus).

Des valeurs supérieures à celles de l'intervalle "optimal" provoquent un accroissement graduel, mais en apparence borné, du nombre d'itérations et de colonnes engendrées. Les temps d'exécution croissent également. Pour les petits problèmes, ils plafonnent mais pour des problèmes de plus grande taille (comme les problèmes DQCNY5 et DQCNY5_QC associés au problème initial QCNY5), ils explosent. Ceci est dû au fait que Cplex a besoin de plus en plus d'itérations pour réoptimiser les

sous-problèmes car les changements des seconds membres sont très importants. De plus, le problème maître semble devenir mal conditionné car il est de plus en plus difficile et long à résoudre (nous reparlerons plus loin de ce problème). Le ratio des pas de descente par itération prend de faibles valeurs (10% et moins).

En général, pour un nombre d'itérations identique, une valeur élevée de γ produira un plus grand nombre de colonnes et un temps d'optimisation plus long. C'est parce que l'on explore une plus grande région de l'espace et que les bases optimales des sous-problèmes varient davantage entre deux itérations successives de l'algorithme.

Des problèmes d'instabilité numérique ont été fréquemment observés pour des valeurs extrêmes de γ . De plus, des valeurs trop petites peuvent occasionner un arrêt prématuré du programme. En effet, il se peut que la petite taille de la région de confiance interdise une décroissance supérieure à ε_S , la constante d'arrêt, de la valeur de la fonction économique. Le programme doit comporter un test vérifiant, lorsque ce cas se présente, si le déplacement proposé correspond à un point situé sur la frontière de la région de confiance et un mécanisme forçant la poursuite des itérations jusqu'à ce que la solution optimale du problème maître soit à l'intérieur de la région de confiance.

Bien que le meilleur temps d'optimisation affiché dans le tableau D.1 ait été obtenu avec la norme ℓ_∞ (pour $\gamma = 0,75$), celui-ci peut, selon le problème, être atteint avec l'une ou l'autre des deux normes. Cependant, la norme ℓ_1 semble la plus sûre. En effet, sauf pour de très faibles valeurs de γ où l'algorithme semble se comporter aussi mal avec les deux normes, les critères de performance que sont le nombre d'itérations, de colonnes engendrées et le temps d'optimisation sont généralement

meilleurs avec la norme ℓ_1 . Celle-ci semble nettement plus efficace pour de fortes valeurs de γ .

Les tests faits sur les problèmes DQCNY i , DQCNY i _QC et DQCNY i _NY, $i \geq 5$, ont montré que les meilleures valeurs de γ sont d'environ 0,9 pour la norme ℓ_∞ et 6,75 pour la norme ℓ_1 . De plus, malgré l'imprécision de l'affirmation qui suit, les performances de l'algorithme ne sont généralement pas "mauvaises" pour $\gamma \in [0,1, 5]$ dans le cas de la norme ℓ_∞ et pour $\gamma \in [1, 50]$ dans le cas de la norme ℓ_1 .

L'algorithme n'ayant pas été testé sur une grande variété de problèmes, il serait téméraire et prématuré d'affirmer qu'une valeur particulière, ou qu'un intervalle particulier de valeurs, de γ soit idéal pour la majorité des problèmes. Par conséquent, pour chaque nouveau problème, nous devrions supposer inconnu l'intervalle des meilleures valeurs de ce paramètre. Ceci, et la dégradation observée des performances lorsque γ prend des valeurs extrêmes, milite grandement en faveur de l'utilisation de procédures dynamiques d'ajustement du "rayon" de la région de confiance.

7.4.2 Ajustement dynamique du paramètre γ

Lorsque les règles de l'étape 8 de l'algorithme le permettent, CMinSfpc modifie comme suit la valeur du "rayon" γ de la région de confiance:

$$\gamma_{k+1} \leftarrow \theta \gamma_k \quad \text{s'il y a pas de descente,}$$

ou

$$\gamma_{k+1} \leftarrow \gamma_k / \theta \quad \text{s'il y a pas nul,}$$

où θ est une constante supérieure à 1 choisie par l'utilisateur du programme. De plus, afin d'empêcher un changement trop rapide de γ , l'utilisateur doit choisir un entier supérieur ou égal à 1, que l'on appelle *facteur d'inertie*.

L'algorithme ne modifiera le paramètre γ que si un nombre minimal, égal au facteur d'inertie, d'itérations consécutives justifiant un même changement a eu lieu. Ainsi, si le facteur d'inertie vaut N , le paramètre γ ne sera augmenté qu'après N pas de descente consécutifs. De même, γ ne sera réduit qu'après N pas nuls consécutifs (vérifiant le test (6.7)). L'utilisation de ce facteur d'inertie a été suggérée par Kiwiel (1990) qui a adopté la valeur 4.

Afin d'éliminer les problèmes d'instabilité numérique résultant de valeurs extrêmes de γ , l'utilisateur doit en définir un intervalle de valeurs admissibles en fixant une borne inférieure γ_{INF} et une borne supérieure γ_{SUP} . Il détermine aussi la valeur initiale γ_0 de γ .

Le tableau D.2 présente quelques statistiques sur les performances de CMinSfpc pour les 15 problèmes tests correspondants à des problèmes comportant de 1 à 5 périodes. Pour tous ces tests, nous avons posé $\theta = 1,33$ et fixé à 4 la valeur du facteur d'inertie. On avait $\gamma_{INF} = 0,1$, $\gamma_{SUP} = 5$ et $\gamma_0 = 0,9$ pour la norme ℓ_∞ , et $\gamma_{INF} = 1$, $\gamma_{SUP} = 50$ et $\gamma_0 = 6,75$ pour la norme ℓ_1 .

Dans la majorité des cas, l'utilisation de la norme ℓ_1 dans la formulation primale du problème maître a produit les meilleurs temps d'optimisation. Le nombre

d'itérations semble être à peu près identique pour les deux normes. On remarque que les formulations DQCNY i _QC produisent constamment de meilleurs résultats que les formulations DQCNY i . Les formulations DQCNY i _NY sont moins intéressantes: bien que supérieures aux formulations DQCNY i , elles produisent presque toujours de moins bons temps de calcul que les formulations DQCNY i _QC. Pour cette raison, nous ne les considérerons plus par la suite. On remarquera aussi que pour la totalité des cas concernant la formulation DQCNY i _QC, c'est la norme ℓ_1 qui a produit les meilleurs temps d'exécution.

En comparant les résultats du problème DQCNY3 à ceux du tableau D.1, on constate que l'utilisation du mécanisme d'ajustement dynamique de γ produit des résultats de très bonne qualité. Il ne parvient cependant pas à faire mieux que les meilleurs résultats du tableau D.1. Dans le cas de la norme ℓ_∞ , on converge en moins de temps et en engendrant moins de colonnes que pour n'importe lequel des 18 cas du tableau D.1. Le nombre d'itérations n'est battu (de justesse) que par l'un de ces 18 cas. Les résultats sont moins spectaculaires pour la norme ℓ_1 . Néanmoins, ils sont intéressants. Des tests supplémentaires effectués avec les problèmes DQCNY1, DQCNY3_QC, DQCNY3_NY, DQCNY5 et DQCNY5_QC ont révélé que cette procédure dynamique produit de très bonnes valeurs des indicateurs de performance sans nécessairement faire mieux à tout coup que la méthode où γ est constant et fixé à la meilleure valeur trouvée. Nous croyons cependant que (1) l'intérêt d'une procédure d'ajustement dynamique résulte de l'ignorance, *a priori*, de la meilleure valeur de γ , et (2) qu'il est probablement possible de faire mieux en fixant plus judicieusement les valeurs du facteur d'inertie, des constantes θ , γ_{INF} , γ_{SUP} et γ_0 , ou en considérant d'autres mécanismes d'ajustement de γ .

Des tests effectués en utilisant plusieurs valeurs de θ indiquent que la valeur

1,33 est excellente. Par contre, nous ne sommes pas arrivés à la même conclusion avec le facteur d'inertie. Nous avons, dans le cas du problème DQCNY4, testé des valeurs du facteur d'inertie comprises entre 1 et 7 pour diverses valeurs de γ et quelques variantes de la procédure d'ajustement de γ sans arriver à des résultats concluants quant à la supériorité d'une valeur particulière du facteur d'inertie. Des tests supplémentaires devraient être entrepris. Pour cette raison, nous avons choisi de le garder fixé à 4. Il serait peut-être intéressant de le rendre lui aussi ajustable dynamiquement.

Tout comme pour les méthodes de coupes classiques (c'est-à-dire non stabilisées), nous avons observé une amélioration rapide de la valeur de la fonction économique au cours des premières itérations et une décroissance lente par la suite. Le phénomène est d'autant plus apparent que le problème de programmation linéaire sous-jacent est gros. Diverses variantes du mécanisme d'ajustement de γ ont été étudiées dans le but de contrer ce problème mais puisqu'aucune d'elles n'a été significativement supérieure au mécanisme du programme, il a semblé préférable de ne pas le modifier tant que des tests plus exhaustifs n'auront pas été conduits. Ces variantes étaient toutes basées sur l'observation que, lorsque la valeur de la fonction économique est assez près de sa valeur optimale, des séquences de plusieurs (dizaines de) pas nuls consécutifs sont fréquentes. Ceci a pour effet d'entraîner assez rapidement la valeur de γ à sa borne inférieure. Les pas de descente consécutifs devenant alors rares, la valeur de γ n'augmente plus, ou presque, et la méthode devient alors une méthode où γ est fixé à une faible valeur, d'où une mauvaise convergence (comme en font foi les premières lignes du tableau D.1). Il a semblé intéressant de faire augmenter plus rapidement et davantage la valeur de γ suite à un pas de descente, une fois que γ a atteint sa valeur minimale.

Enfin, on remarque en examinant les valeurs du ratio “Temps_PM/Temps_SP” du temps total d’optimisation du problème maître sur le temps total d’optimisation des sous-problèmes que l’algorithme se comporte très bien lorsque la valeur de ce ratio est faible (i.e. inférieure à $\frac{1}{2}$) et plutôt bien lorsque ce ratio n’excède pas 1,0. C’est le cas de toutes les formulations à 3 périodes ou moins. Par contre, les formulations DQCNY4 et DQCNY5 exhibent des ratios variant entre 2,66 et 7,13 et les indicateurs de performance se dégradent fortement. Il est intéressant de constater que la formulation alternative DQCNY i _QC est caractérisée par de faibles valeurs de ce ratio, pour toutes valeurs de $i \leq 5$.

La faiblesse des formulations DQCNY i tient au fait que, en ne considérant pas les variables artificielles ajoutées par le programme, les modèles de production d’électricité des deux régions peuvent s’accommoder d’importantes fluctuations de la demande et fournir des solutions optimales (réalisables), alors que ce n’est pas le cas du modèle d’échanges correspondant au problème (2.5) du chapitre 2. En effet, on peut, en simplifiant les choses, voir les valeurs des variables du problème non-différentiable comme les capacités des sources et les demandes des puits d’une espèce de problème de transport (comme le problème (2.8)). Or, la solution du problème maître vérifie rarement les conditions nécessaires pour qu’un problème de transport soit bien défini (comme la somme des capacités des sources est égale à la somme des demandes des puits). Il est donc nécessaire de faire beaucoup d’itérations pour arriver à un tel résultat.

Les formulations alternatives sont supérieures car, en intégrant le sous-modèle des échanges au modèle de l’une des régions, on augmente significativement la proportion des itérations où tous les sous-problèmes sont réalisables et l’information

transmise au problème maître est alors de meilleure qualité. Cette remarque prendra tout son sens dans la prochaine section, où nous n'utiliserons plus de variables artificielles.

Des ratios "Temps_PM/Temps_SP" élevés suggèrent l'utilisation de la stratégie d'accumulation/épuration. En effet, les problèmes maîtres étant de plus petites tailles, ils seront plus faciles à résoudre et la valeur du ratio devrait s'améliorer. Par contre, les solutions seront moins bonnes et le nombre d'itérations pourra augmenter.

7.4.3 La stratégie d'accumulation/épuration

Le programme CMinSfpc met en oeuvre une stratégie d'accumulation/épuration. A chaque itération, il détruit les colonnes du problème maître dual (5.20a)–(5.20f) dont les variables sont nulles et les coûts réduits strictement positifs.

Les résultats de l'optimisation des problèmes $DQCNY_i$ et $DQCNY_i\_QC$, $i \leq 5$, apparaissent au tableau D.3. On remarque, en les comparant à ceux du tableau D.2, que, comme nous nous y attendions, le nombre d'itérations a augmenté pour chaque problème. Cette augmentation est modeste ou nulle pour les plus petits problèmes ($i \leq 2$), mais devient importante pour les plus gros. L'augmentation du nombre de colonnes engendrées est proportionnelle à celle du nombre d'itérations. La stratégie s'est aussi avérée plus coûteuse en terme de temps de calcul: pour la plupart des problèmes, l'augmentation du temps de calcul est plus ou moins proportionnelle à celle du nombre d'itérations.

Le phénomène des longues séquences de pas nuls consécutifs observé près de

l'optimum sous la stratégie d'accumulation est encore plus évident avec la stratégie d'accumulation/épuration et il devient nécessaire, afin d'aboutir rapidement à la précision recherchée, de trouver un mécanisme d'accélération.

Ici aussi, et encore plus que dans le cas de la stratégie d'accumulation, la norme ℓ_1 s'est avérée supérieure à la norme ℓ_∞ .

Suite aux résultats jusqu'ici obtenus, il a été décidé de ne résoudre les problèmes de plus de 5 périodes qu'avec la formulation DQCNY i _QC (en fait, avec une variante de celle-ci).

7.4.4 La convergence asymptotique

Le tableau D.4 illustre le comportement asymptotique de l'algorithme. Il contient le nombre d'itérations et le temps calcul requis pour résoudre le problème DQCNY5_QC avec une erreur absolue de 0,025 et quatre niveaux d'erreur relative (i.e. 0,1%, 0,5%, 1,0% et 2,5%). L'algorithme a été utilisé avec chacune des deux normes ℓ_∞ et ℓ_1 et chacune des deux stratégies de gestion de l'information (la lettre "A" correspond à la stratégie d'accumulation tandis que la lettre "E" est associée à celle d'accumulation/épuration).

Pour chacune des 4 combinaisons considérées, on constate que la recherche d'une précision élevée se traduit par un nombre élevé d'itérations et de longs temps de calcul. Ainsi, si l'on se satisfait d'une erreur relative de 0,5%, on réduit substantiellement le nombre d'itérations (de 45 à 48% dans le cas de la stratégie d'accumulation et de 61 à 79% dans le cas de la stratégie d'accumulation/épuration)

et le temps de calcul (d'approximativement $\frac{2}{5}$ sous la stratégie d'accumulation, et de $\frac{2}{3}$ sous la stratégie d'accumulation/épuration).

On constate aussi que la stratégie d'accumulation/épuration requiert approximativement le même nombre d'itérations que la stratégie d'accumulation pour des erreurs relatives comprises entre 1,0% et 2,5%. Le temps de calcul est alors inférieur (de 13% sous la norme ℓ_∞ et de 6–7% sous la norme ℓ_1). La stratégie d'accumulation est supérieure pour des erreurs relatives inférieures à 0,1%.

Le comportement ainsi mis en évidence s'observe pour tous les gros problèmes traités. Plus le problème est de grande taille et plus il est apparent. Il suggère que l'utilisation de la norme ℓ_1 et de la stratégie d'accumulation/épuration avec une erreur relative qui n'est pas inférieure à 1% produit les meilleurs temps de calcul. Malheureusement, les solutions obtenues ne sont généralement pas réalisables et il est souhaitable, si l'on veut procéder ainsi, de ne plus recourir à l'approche des pénalisations.

7.4.5 Deux procédures d'enrichissement

Le programme CMinSfpc comporte deux options d'enrichissement des modèles des fonctions dans un voisinage de la solution proposée par le problème maître. Sous l'*enrichissement sélectif*, après la résolution d'un sous-problème, le programme effectue des analyses de sensibilité pour chacun des éléments de son second membre correspondant à une contrainte liante. Utilisant les valeurs duales et les $2n_i$ variations trouvées (2 pour chacune des n_i contraintes liantes du sous-problème i), il détermine la contrainte et le sens de la variation de son second membre produisant

la plus forte décroissance de la valeur de la fonction économique du sous-problème. Il réévalue alors la fonction valeur $v_i(\cdot)$ à ce nouveau point (en fait, à une petite distance $\varepsilon > 0$ à l'extérieur de l'intervalle) afin d'engendrer un nouveau sous-gradient. Par conséquent, sous l'enrichissement sélectif, deux sous-gradients peuvent être engendrés à chaque itération par chacun des sous-problèmes.

Sous la procédure d'*enrichissement massif*, l'algorithme réévalue la fonction $v_i(\cdot)$ à chacun des $2n_i$ nouveaux points déterminés par les analyses de sensibilité. Jusqu'à $2n_i + 1$ sous-gradients peuvent ainsi être engendrés par le sous-problème i à chaque itération.

Les problèmes DQCNY1_QC, DQCNY2_QC et DQCNY3_QC ont été résolus en utilisant ces deux procédures d'enrichissement. Le tableau D.5 présente quelques résultats de ces tests (les lettres "I", "S" et "M" correspondent respectivement à un enrichissement inactif, sélectif et massif des modèles). Il révèle, sans surprise, que le nombre d'itérations diminue à mesure que l'on accroît le niveau d'enrichissement et que la proportion des pas de descente s'accroît. Par contre, cette réduction du nombre d'itérations ne suffit pas à compenser pour l'augmentation du temps calcul résultant de l'enrichissement. Le temps de calcul se détériore à mesure que le niveau d'enrichissement augmente. Cette dégradation du temps de calcul semble attribuable au coût fixe élevé de l'appel à la procédure d'optimisation de Cplex (des tests ont montré que le temps de calcul de Cplex peut être représenté par le modèle linéaire $\beta + \alpha \text{ iter}$ où $\beta \approx 7\alpha$, "iter" est le nombre d'itérations du simplexe et α est le temps de calcul pour une itération).

Les mauvaises performances des procédures d'enrichissement peuvent aussi être expliquées par le fait qu'il est inutile d'enrichir les modèles des fonctions lorsque

l'on est loin de l'optimum: les sous-gradients engendrés aux points proposés par le problème maître suffisent à éliminer ces points et leurs voisinages. Il semble préférable de réserver l'enrichissement aux dernières itérations lorsque la solution courante n'est plus très éloignée de l'ensemble des solutions optimales. On pourrait espérer ainsi réduire suffisamment les longues séquences de pas nuls consécutifs observés sous des erreurs relatives minimales dans le but de réduire les temps d'exécution.

7.5 L'approche des plans coupants

Même s'il est possible de transformer artificiellement, comme nous l'avons fait à la section précédente, le problème de manière à ce que les fonctions soient toutes à valeurs finies, notre problème en est d'abord un de minimisation d'une somme de fonctions convexes à valeurs dans $\mathbb{R} \cup \{+\infty\}$. Le traitement des valeurs infinies, i.e. des incompatibilités des sous-problèmes, consiste à engendrer des plans coupants destinés à forcer le problème maître à produire des solutions faisant partie du domaine de l'optimisation. C'est cette approche qui a été explorée dans les précédents chapitres. L'un de ses intérêts réside dans les faiblesses de l'approche des pénalisations.

D'abord, il n'est pas toujours évident de déterminer *a priori* une valeur raisonnable du coût unitaire de pénalisation associé aux variables artificielles. Une valeur trop faible mène à une solution non-optimale (et généralement non-réalisable) pour le problème original tandis qu'une valeur trop élevée conduit à de sérieux problèmes d'instabilité numérique pouvant empêcher l'optimisation.

Ensuite, comme nous l'avons déjà remarqué, l'approche des pénalisations force l'utilisation d'une précision très élevée si l'on doit trouver une solution finale réalisable.

Avec l'approche des plans coupants, ces problèmes disparaissent. Celle-ci nécessite cependant une double utilisation de l'algorithme dans laquelle on détermine, dans une première étape (la Phase I), une solution réalisable, puis dans une seconde étape (la Phase II) une solution ε -optimale. Cette procédure en deux phases peut augmenter le nombre d'itérations et le temps calcul (par rapport à l'approche des pénalisations). Par contre, une fois la Phase I terminée, on peut en tout temps interrompre les calculs et disposer d'une solution réalisable. Cette approche semble donc toute désignée lorsque l'on désire une solution réalisable ε -optimale, avec une valeur de ε pas trop petite.

7.5.1 Rayons invariants et méthode de Benders

L'étude du comportement de l'algorithme lorsque le paramètre γ est fixé ayant été faite sous l'approche des pénalisations, nous nous intéresserons davantage à une méthode bien connue qu'il est possible d'obtenir en apportant des modifications mineures à la codification de la méthode de faisceaux, soit la méthode de Benders. Celle-ci est disponible comme une option du programme CMinSfpc.

Contrairement aux méthodes de faisceaux, la méthode de Benders (1962), comme toutes les méthodes classiques de plans sécants, travaille avec une "région de confiance" de taille infinie. De plus, elle ignore les concepts de pas nul et de pas de descente: il y a un déplacement à chaque itération.

Il est donc possible, en ignorant le test des pas de descente et en posant $\gamma = +\infty$, de transformer l'algorithme de manière à obtenir la méthode de Benders. Ceci a été fait dans le cas de la norme ℓ_∞ en posant γ égal à 10000, une valeur très largement supérieure à la norme des solutions de chacun de nos problèmes tests.

Le tableau D.6 permet, pour trois problèmes, une comparaison des performances de la méthode de Benders à celles des méthodes de faisceaux avec paramètre γ constant fixé à 0,9 (sous la norme ℓ_∞). Le tableau permet aussi la comparaison des deux approches de la méthode de faisceaux. Ainsi, le sigle "FP" dans la colonne "Méthode" désigne l'approche des pénalisations tandis que le code "FC" est associé à l'approche des plans coupants. La lettre "B" désigne la méthode de Benders. Les valeurs entre parenthèses de la colonne "Itérations" sont les nombres d'itérations de Phase I de la méthode de faisceaux avec plans coupants. La dernière colonne du tableau donne le nombre total d'itérations du simplexe ayant été nécessaires pour l'optimisation initiale et les réoptimisations subséquentes des sous-problèmes et du problème maître. Les chiffres entre parenthèses ne concernent que les itérations du problème maître.

Comme il fallait s'y attendre, la méthode de faisceaux avec plans coupants est moins intéressante que celle avec pénalisations. Cependant, pour les trois problèmes tests, les deux méthodes de faisceaux surpassent la méthode de Benders, tant pour le nombre d'itérations, que pour le nombre de colonnes engendrées, que le temps d'optimisation ou que le nombre total d'itérations du simplexe. Ceci provient du fait que le problème maître de la méthode de Benders permet de très grands déplacements, ce qui augmente le nombre d'itérations du simplexe nécessaires à la réoptimisation des sous-problèmes et, de ce fait, le temps de calcul.

Les résultats concernant le problème DQCNY3 sont des plus intéressants. On remarque que le temps de calcul et le nombre total d'itérations du simplexe sont respectivement 51,6 et 37,3 fois supérieurs à ceux de la méthode de faisceaux avec pénalisations et de 18,2 et 21,8 fois supérieurs à ceux de la méthode de faisceaux avec plans coupants. Cette dégradation, qui ne s'observe pas aussi nettement sur les deux problèmes de plus petites tailles, est essentiellement due au problème maître, comme en fait foi la dernière colonne du tableau D.6. La très grande valeur du paramètre γ semble provoquer de l'instabilité numérique qui, à certaines itérations de l'algorithme, force l'optimiseur à faire un très grand nombre d'itérations, comme si sa base initiale était très mauvaise ou inexistante. La norme (du maximum) de la solution optimale du problème étant approximativement égale à 26,7, nous avons cherché à confirmer l'interprétation précédente en refaisant l'optimisation en fixant la valeur du paramètre γ à 35. La dernière ligne du tableau D.6 montre une amélioration de tous les indicateurs de performance: modeste dans le cas du nombre d'itérations de l'algorithme et du nombre de colonnes engendrées, mais spectaculaire dans le cas du temps calcul et du nombre total d'itérations du simplexe. La méthode est tout de même 10,5 fois plus lente que celle de faisceaux avec pénalisations et 3,7 fois plus lente que celle de faisceaux avec plans coupants. Enfin, un autre essai, fait en fixant γ à 100, s'est révélé aussi mauvais que celui où $\gamma = 10000$.

La méthode de Benders semble donc se comporter "raisonnablement" bien lorsque l'on peut borner *a priori* de façon serrée l'intervalle de variation de chacune des composantes de la variable à optimiser, chose difficile à faire. De plus, on ne peut garantir l'obtention d'une solution réalisable avant la dernière itération de l'algorithme. Il a été observé, pour les problèmes traités, que la borne supérieure, égale à la valeur de la meilleure solution réalisable trouvée, vaut $+\infty$ pour la presque totalité des itérations. Ceci signifie que la première solution réalisable n'est trouvée

que vers la fin (dans les 3 ou 5 dernières itérations) des calculs. Il est dans ce cas impossible de faire cesser rapidement les calculs en fixant une erreur élevée car le test d'arrêt ne sera vérifié, à quelques itérations près, qu'après le même nombre d'itérations que pour une petite valeur de l'erreur permise. De plus, l'instabilité numérique observée lors de la résolution du problème maître, ne disparaît que vers la fin des calculs. Impossible non plus d'espérer en éviter une bonne partie en ne demandant qu'une faible précision.

On ne peut cependant pas généraliser ces observations à tous les problèmes. Il semble en effet que les méthodes classiques de plans coupants sont particulièrement efficaces pour résoudre les problèmes pouvant être décomposés en beaucoup de sous-problèmes. C'est le cas notamment des problèmes de multiflot et des problèmes de programmation stochastique (du Merle, Goffin et Vial, 1996). Nos problèmes ne sont pas de ce genre et les résultats avec les trois plus petits d'entre eux sont suffisamment éloquents pour décider de ne pas poursuivre davantage les tests avec la méthode de Benders.

7.5.2 Ajustement dynamique du paramètre γ

Etant donné que la méthode comporte deux phases, la question se pose quant à savoir s'il est préférable de démarrer la phase II en utilisant la valeur finale de phase I du paramètre γ ou une autre valeur. Le programme CMinSfpc offre deux choix: la valeur initiale de phase II du paramètre γ est ou bien la valeur finale de phase I ou bien la valeur initiale γ_0 de phase I.

Ces deux possibilités ont été testées sur tous les problèmes DQCNY $_i$ et

DQCNY i _QC de moins de 6 périodes en utilisant les deux normes et les deux stratégies et il a été trouvé que, pour la plupart d'entre eux, cela ne changeait pas grand chose. Cependant, il nous est arrivé d'observer sur quelques problèmes une amélioration significative du temps de calcul en refixant γ à γ_0 en début de phase II et, pour cette raison, nous avons choisi de ne présenter que des résultats relatifs à ce choix.

Les tableaux D.7 et D.8 présentent les résultats des tests effectués sur ces problèmes en utilisant respectivement les stratégies d'accumulation et d'accumulation/épuration de l'information. Dans ces deux tableaux, les valeurs entre parenthèses des colonnes "Itérations" correspondent au nombre d'itérations nécessaires pour trouver une solution réalisable.

On observe tout d'abord, en comparant le tableau D.7 au tableau D.2, que la méthode de faisceaux avec plans coupants n'est pas nécessairement plus lente que la méthode de faisceaux avec pénalisations. La variante avec pénalisations semble plus rapide pour les petits problèmes ($i \leq 3$) tandis que les deux variantes ont à peu près les mêmes temps d'exécution pour les plus gros problèmes (comme DQCNY4_QC et DQCNY5_QC). Ceci s'explique par le fait que l'évaluation d'un sous-problème non réalisable est habituellement plus rapide lorsqu'il n'y a pas de variables artificielles. En effet, l'optimiseur stoppe les calculs dès qu'il détecte une impossibilité. Avec l'approche des pénalisations, il peut sortir du domaine de la fonction (en encourageant des pénalités) et faire plusieurs itérations avant de déterminer une solution optimale (mais non réalisable pour le problème sans variable artificielle). De plus, ce sous-gradient provenant d'un point assez éloigné de la région réalisable, la coupe qui y correspond élimine probablement moins de solutions non-réalisables que celle résultant de la méthode des plans sécants.

On constate aussi que la phase I est courte. La plupart des itérations servent à améliorer la valeur de la fonction économique et il est alors possible de faire arrêter très rapidement l'algorithme avec une solution réalisable.

Dans le cas de la stratégie d'accumulation/épuration, les résultats sont, comme les tableaux D.8 et D.3 le montrent, plus mitigés. La norme ℓ_1 semble produire de meilleurs résultats avec l'approche des plans coupants pour les plus gros problèmes. Ce n'est pas le cas avec la norme du maximum. Il faudrait conduire d'autres tests afin d'être plus ferme dans nos conclusions.

La comparaison des tableaux D.7 et D.8 montre, sans surprise, que le nombre moyen de pas de descente par itération se dégrade avec le recours à la stratégie d'accumulation/épuration. On constate aussi que les temps de calcul sous cette stratégie se dégradent moins dans le cas de la norme ℓ_1 . Enfin, la phase I ne requiert que peu d'itérations pour chacune des 4 combinaisons de stratégies d'accumulation et de normes. En fait, le recours à la stratégie d'accumulation/épuration n'accroît, dans la majorité des cas, que très peu le nombre d'itérations de la phase I. Utilisant ces faits et le plus faible temps d'optimisation du problème maître, on pourrait probablement obtenir une solution réalisable ε -optimale avec de faibles temps de calcul si on était prêt à choisir des ε correspondants à des erreurs relatives pas trop faibles. Nous revenons plus loin sur ce point.

Une autre façon d'améliorer les temps de calcul consisterait à modifier la stratégie d'accumulation/épuration de manière à ne pas trop éliminer d'information. Ainsi, l'épuration pourrait être faite en ne détruisant que les colonnes du problème (5.20a)–(5.20f) dont les variables sont nulles et les coûts réduits supérieurs à un seuil strictement positif. Actuellement, ce seuil est fixé à zéro. On pourrait aussi ne pas

déclencher l'épuration à chaque itération. En combinant ces deux idées, on peut penser obtenir une stratégie d'accumulation/épuration plus rapide que la stratégie d'accumulation.

7.5.3 La convergence asymptotique

Le tableau D.9 illustre, pour le problème DQCNY5_QC et la norme ℓ_1 , le comportement asymptotique de l'algorithme pour les stratégies d'accumulation ("A") et d'accumulation/épuration ("E"). On y constate à peu près les mêmes comportements que ceux observés au tableau D.4 sous l'approche des pénalisations. Mais cette fois-ci, on peut interrompre en tout temps l'optimisation (après la phase I) et disposer d'une solution réalisable.

La stratégie d'accumulation produit les meilleurs temps de calcul pour de faibles valeurs des erreurs relatives ($\leq 0,5\%$) tandis que les deux stratégies ont sensiblement les mêmes temps d'exécution pour de plus grandes erreurs relatives. La dernière ligne pour chacune des stratégies caractérise la solution initiale de phase II. Il est intéressant de constater qu'elle n'est que 2,72% plus coûteuse qu'une solution optimale. Des précisions initiales d'environ 2% ne sont pas rares.

La qualité des solutions initiales a cependant plus à voir avec le problème traité qu'avec la méthode. Des tests additionnels sur d'autres classes de problèmes sont donc à faire.

7.6 Les derniers problèmes

En substituant les contraintes

$$C_i y \geq -s_i, \quad i \in \{1, \dots, r\}$$

aux contraintes

$$C_i y = -s_i, \quad i \in \{1, \dots, r\}$$

du problème (2.5) défini au chapitre 2, on obtient un nouveau problème équivalent (2.4) dans lequel la fonction $v_0(\cdot)$ est remplacée par la fonction $v'_0(\cdot)$ associée au sous-problème (2.5') ainsi défini.

En procédant ainsi, on peut tirer 27 autres problèmes équivalents (que nous désignerons DQCNY*i*', DQCNY*i*_QC' et DQCNY*i*_NY') des 27 problèmes du tableau 7.2. Des tests préliminaires effectués avec ces nouvelles formulations montrent une amélioration des performances de CMinSfpc pour les plus gros problèmes. Par contre, pour les petits problèmes (de moins de 5 périodes), les indicateurs de performance peuvent se détériorer.

Le plus grand intérêt des formulations DQCNY*i*', DQCNY*i*_QC' et DQCNY*i*_NY' obtenues en utilisant le sous-problème (2.5') provient du fait que les contraintes de ce sous-problème sont plus faciles à satisfaire et que la phase I de l'approche des plans coupants est plus courte. On est alors en mesure d'interrompre plus rapidement l'algorithme et de tout de même disposer d'une solution réalisable.

Ces formulations ont été utilisées pour résoudre les plus gros problèmes. Le tableau D.10 contient des statistiques relatives à la résolution des problèmes

DQCNY i _QC', $i \in \{6, 7, 8, 9\}$ en utilisant la norme ℓ_1 . Chaque problème a été résolu de trois manières différentes, en retenant trois des quatre combinaisons des deux approches ("C" pour plans coupants et "P" pour pénalisations) aux deux stratégies ("A" pour accumulation et "E" pour accumulation/épuration). Les valeurs entre parenthèses indiquent soit le nombre d'itérations de phase I de l'approche des plans coupants, ou soit le nombre de colonnes détruites sous la stratégie d'accumulation/épuration.

On remarque d'abord le faible nombre (8) d'itérations de phase I de l'approche des plans coupants pour chacune des stratégies et chacun des problèmes. Ceci confirme notre remarque précédente et permet une interruption très rapide de l'algorithme avec une solution réalisable.

On constate ensuite que, contrairement à ce qui avait été observé pour les plus petits problèmes, l'approche des plans coupants, avec ou sans épuration de l'information, est supérieure à celle des pénalisations. Cet avantage va en augmentant à mesure que croît la taille du problème. Ainsi, alors que les temps de calcul sont pratiquement les mêmes pour le problème DQCNY6_QC' sous la stratégie d'accumulation, ils sont très nettement en faveur de l'approche des plans coupants (avec moins de la moitié de celui de l'approche des pénalisations) pour le problème DQCNY9_QC'.

Enfin, il est rassurant de constater le faible nombre d'itérations requises pour atteindre la précision demandée (i.e. 0,025), dans le cas de l'approche des plans coupants sous la stratégie d'accumulation. Notons qu'elle correspond à des erreurs relatives inférieures à 0,000029%. Ceci laisse entrevoir un énorme potentiel de réduction du temps calcul si on est prêt à augmenter l'erreur relative. Le tableau

D.11, semblable au tableau D.9, le confirme.

La plupart des paramètres utilisés pour conduire l'algorithme (comme θ , γ_0 , γ_{INF} , γ_{SUP} et le facteur d'inertie) ayant été déterminés en résolvant les plus petits problèmes, il est permis de croire que leurs valeurs ne sont pas les plus appropriées pour les plus gros problèmes (et peut-être même pour les plus petits). De plus, bien d'autres procédures d'ajustement peuvent être imaginées pour faire varier le paramètre γ . Il faut donc voir les résultats de ce chapitre, et particulièrement ceux du tableau D.10, comme préliminaires. Il existe sans doute un important potentiel d'amélioration qui reste à être exploité et de bien meilleurs temps calcul peuvent être espérés.

7.7 Conclusion

En résumé, il ressort des tests précédents qu'à moins de très bien connaître le comportement de CMinSfpc pour un problème donné, il est préférable de recourir à l'option d'ajustement dynamique du rayon de la région de confiance. De plus, les meilleures performances du programme sont habituellement obtenues en utilisant la norme ℓ_1 sous la stratégie d'accumulation. L'approche des plans coupants s'est avérée supérieure à celle des pénalisations pour les problèmes de plus grande taille, i.e. ceux pour lesquels on peut penser recourir à des méthodes de décomposition. Bien que plus stables et rapides que la méthode de Benders, les méthodes mises en oeuvre dans CMinSfpc n'ont pas réussi à éliminer le caractère asymptotique des méthodes de plans sécants. De très bons temps calcul sont obtenus si le critère d'arrêt correspond à une erreur relative de 1% ou plus. Dans ce cas, la stratégie

d'accumulation/épuration fait aussi bien que la stratégie d'accumulation. Plusieurs suggestions ont été émises dans le but d'améliorer les performances du programme.

Nous avons réussi à améliorer significativement les temps de résolution des problèmes en fusionnant le problème (2.5) de coordination des échanges à l'un ou l'autre des deux problèmes régionaux. Il faut toutefois s'attendre à rencontrer de nouveau les difficultés mentionnées à la fin de la sous-section 7.4.2 s'il y a plus de deux régions, comme c'est le cas, par exemple, du modèle d'échanges d'électricité à quatre régions de Berger *et al.* (1990). En effet, pour plus de deux régions, l'harmonisation des échanges entre les régions 2 à r sera longue à établir si le modèle de coordination des échanges est fusionné à celui de la première région. Par contre, cette dernière aura peu de difficulté à établir des échanges réalisables avec les autres régions. Ceci suggère l'idée de construire une méthode stabilisée de décomposition emboîtée ("nested decomposition") où le modèle de chaque région joue à la fois les rôles de problème maître et de sous-problème. Pour ce faire, on pourrait chercher à stabiliser l'algorithme de Kallio et Porteus (1977), conçu pour les problèmes de programmation linéaire avec structure arborescente, en utilisant les idées de notre méthode.

On remarquera que les temps de calcul de CMinSfpc sont supérieurs à ceux du tableau 7.1 où Cplex avait été utilisé pour résoudre les problèmes non-décomposés. Ce n'est pas surprenant, vue la petite taille des problèmes traités. De plus, malgré tous les efforts consacrés à sa codification, il n'en demeure pas moins qu'il s'agit là d'une première mise en oeuvre de la méthode et que des améliorations importantes sont à espérer. Il faut donc accorder plus d'importance à l'étude du comportement de la méthode qu'aux temps de calcul.

Bien que, comme expliqué au paragraphe 7.2, l'on se soit d'abord restreint à l'utilisation des normes ℓ_∞ et ℓ_1 , il existe des raisons de croire que la norme euclidienne, ou plus généralement une norme quadratique, conduise à de meilleures performances, du moins en termes d'itérations.

Celles-ci tournent autour du théorème II.2.2.4 de Hiriart-Urruty et Lemaréchal (1993a) (il s'agit en fait d'un résultat de Polak (1971)) et de leur discussion sur l'utilisation de normes générales dans une méthode de plus forte pente appliquée à la minimisation de fonctions deux fois continuellement différentiable.

Utilisant ce résultat et faisant quelques approximations, ils comparent les normes ℓ_1 et ℓ_2 et trouvent que le taux de convergence se détériore lorsque, dans le cas de la norme ℓ_1 , le nombre de variables augmente, ce qui n'est pas le cas avec la norme euclidienne. De plus, celui-ci est toujours à l'avantage de la norme euclidienne.

Bien entendu, nous ne cherchons pas à minimiser des fonctions deux fois continuellement différentiable et les directions déterminées par notre méthode n'en sont pas toutes de plus forte pente. Néanmoins, ces résultats sont suffisamment éloquents pour justifier des tests ultérieurs avec la norme euclidienne.

CHAPITRE 8

Conclusions et travaux futurs

Nous avons, dans cette thèse, reconsidéré les problèmes de programmation linéaire de grande taille munis de la structure duale angulaire par blocs de Benders (1962). Chaque problème de cette classe peut être transformé, par projection sur les variables liantes, en un problème équivalent d'optimisation convexe non-différentiable consistant en la minimisation d'une somme de fonctions polyédriques convexes sur son domaine que nous avons résolu à l'aide d'une méthode de plans coupants stabilisée.

Plus précisément, nous avons proposé une nouvelle méthode de faisceaux adaptée à ce problème. La méthode pourrait éventuellement être modifiée pour traiter le problème plus général de la minimisation d'une somme de fonctions convexes étendues (non nécessairement polyédriques) sur un ensemble convexe fermé quelconque. La méthode est basée sur la transformation du problème contraint en un problème non contraint obtenu en ajoutant un terme de pénalisation, la fonction indicatrice du domaine de l'optimisation, à la fonction économique originale. Elle exploite le fait que les hyperplans produits par les oracles utilisés permettent de définir des ε -sous-gradients de la fonction indicatrice du domaine de l'optimisation au centre de stabilité courant.

Avant d'obtenir l'algorithme, nous avons considéré deux approches différentes, mais non indépendantes, de détermination de directions d'amélioration. Ces deux

approches, soient celles des régions de confiance et des pénalisations, utilisent la fonction norme afin de restreindre l'optimisation à une région de l'espace où nous espérons que les modèles des fonctions ont de meilleures chances d'être de bonne qualité. Nous avons considéré des normes quelconques et constaté que les normes qui semblent les plus avantageuses pour une approche ne le sont pas pour l'autre. Ainsi, les normes ℓ_1 et ℓ_∞ sont appropriées pour l'approche des régions de confiance car le problème de détermination d'une direction d'amélioration en est alors un de programmation linéaire facile à résoudre. Par contre, ces fonctions ne sont pas intéressantes pour l'approche des pénalisations. Une norme "ronde" comme la norme euclidienne menant à un terme de pénalisation strictement convexe est alors avantageuse.

L'algorithme que nous avons conçu est basé sur l'approche des pénalisations. Cependant, les tests numériques du chapitre 7 ont été faits à l'aide de l'approche des régions de confiance obtenue en adaptant l'algorithme. Nous avons testé les normes ℓ_1 et ℓ_∞ , et les résultats obtenus sont encourageants. Ils démontrent une très nette supériorité de la méthode sur celle de Benders. Ils montrent aussi que la méthode fonctionne mieux avec la norme ℓ_1 . Bien que la méthode soit beaucoup plus stable et rapide que celle de Benders, elle présente tout de même le comportement asymptotique des méthodes de plans coupants.

Les tests ont permis de faire une première calibration des principaux paramètres de la méthode sur une classe de problèmes réels. Beaucoup de travail reste cependant à faire afin de mieux les calibrer. Il est possible aussi que des améliorations importantes de performances puissent être obtenues en dynamisant certains de ces paramètres. Il faudrait aussi tester l'approche des pénalisations en utilisant la norme euclidienne afin de déterminer s'il y a effectivement des avantages à utiliser cette

norme qui nous force à résoudre des problèmes de programmation quadratique particuliers pour obtenir des directions.

Les tests numériques ont mis en évidence la supériorité de certaines formulations du problème non-différentiable équivalent. Il est probable que la méthode ne soit pas suffisamment bien adaptée au problème des échanges entre r agents économiques de la section 2.2 lorsque $r > 2$, comme par exemple pour le problème d'échanges d'électricité de Berger, Dubois, Haurie et Loulou (1990) où l'on a $r = 4$. Des gains de performance sont à espérer d'une méthode de décomposition emboîtée stabilisée dans laquelle, par exemple, le sous-problème de l'agent i sert à construire une approximation polyédrique f_i^k qui sera amalgamée à la représentation exacte f_{i+1} de l'agent $i + 1$. L'optimisation de $f_{i+1} + f_i^k$ permettra de déterminer une approximation f_{i+1}^k envoyée à l'agent $i + 2$, et ainsi de suite. Avec une telle approche, chaque agent a un problème jouant les rôles de problème maître pour le précédent et de sous-problème pour le suivant. Plusieurs mécanismes de ce genre peuvent être imaginés. Quelques méthodes non stabilisées ont déjà été proposées pour des problèmes de programmation linéaire (Ho et Manne, 1974; Ament et al., 1981; Kallio et Porteus, 1977). La méthode de Kiwiel (1987) présente des ressemblances avec une telle approche. Il serait peut-être possible de s'en inspirer en l'améliorant.

Enfin, et surtout, l'algorithme pourrait être amélioré en le modifiant de manière à ce qu'il puisse résoudre le problème général de la minimisation d'une somme de fonctions convexes étendues et fermées sur un ensemble convexe fermé. Ceci pourrait possiblement être fait en utilisant la remarque 6.3.4, les remarques suivants le lemme 6.5.6, et une approche de pénalisation exacte. Ceci permettrait de généraliser les résultats de Kiwiel (1991). Un inconvénient important de notre méthode vient du fait que la fonction indicatrice du domaine D ne fournit que peu d'informations.

On pourrait chercher à la remplacer par une fonction continue à valeurs réelles dans une méthode inspirée de la discussion de la section 4.3. Le meilleur candidat semble être la fonction *jauge* (Rockafellar, 1970; Hiriart-Urruty et Lemaréchal, 1993a) qui est cependant difficile à évaluer. Il est toutefois possible de l'approcher en évaluant plutôt la fonction jauge de l'approximation externe D^* du domaine. On obtient ainsi une approximation minorante qui, si l'on pouvait en estimer la précision, pourrait être utilisée dans une méthode inspirée de celle de Kiwiel (1995).

RÉFÉRENCES

- ABADIE, J.M. et WILLIAMS, A.C. (1963). Dual and parametric methods in decomposition, *Recent Advances in Mathematical Programming*, R.L. Graves, P. Wolfe, McGraw-Hill Book Company, Inc., New York, 149–158.
- AMENT, D., HO, J., LOUTE, E. et REMMELSWAAL, M. (1981). LIFT: A nested decomposition algorithm for solving lower block triangular linear programs, *Large-Scale Linear Programming*, G.B. Dantzig, M.A.H. Dempster, M.J. Kallio, IIASA, Laxenburg, Autriche, Vol. 1, 383–408.
- AUSLENDER, A. (1978). Minimisation de fonctions localement lipschitziennes: Applications à la programmation mi-convexe, mi-différentiable, *Nonlinear Programming 3*, O.L. Mangasarian, R.R. Meyer, S.M. Robinson, Academic Press, New York, 429–460.
- AUSLENDER, A. (1987). Numerical methods for nondifferentiable convex optimization, *Mathematical Programming Study*, Vol. 30, February 1987, 102–126.
- AUSLENDER, A. et CROUZEIX, J.P. (1989). Well behaved asymptotical convex functions, *Analyse non linéaire*, H. Attouch, J.-P. Aubin, F. Clarke, I. Ekeland, Gauthiers-Villars, Paris, 101–121.

- BAHN, O., GOFFIN, J.-L., VIAL, J.-P. et DU MERLE, O. (1994). Experimental behavior of an interior point cutting plane algorithm for convex programming: an application to geometric programming, *Discrete Applied Mathematics*, Vol. 49, 3–23.
- BAHN, O., DU MERLE, O., GOFFIN, J.-L. et VIAL, J.-P. (1994). *A cutting plane method from analytic centers for stochastic programming*, Technical Report, Département d'Économie Commerciale et Industrielle, Université de Genève, Suisse, December 1992, revised January 1994, à paraître dans *Mathematical Programming*.
- BALAS, E. (1966). An infeasibility-pricing decomposition method for linear programs, *Operations Research*, Vol. 14, No. 5, September-October 1966, 847–873.
- BARTELS, R.H., CONN, A.R. et CHARALAMBOUS, C. (1978). On Cline's direct method for solving overdetermined linear systems in the ℓ_∞ sense, *SIAM Journal of Numerical Analysis*, Vol. 15, No. 2, April 1978, 255–270.
- BARTELS, R.H., CONN, A.R. et SINCLAIR, J.W. (1978). Minimization techniques for piecewise differentiable functions: The ℓ_1 solution to an overdetermined linear system, *SIAM Journal of Numerical Analysis*, Vol. 15, No. 2, April 1978, 224–241.
- BELL, E.J., Jr. (1965). *Primal-Dual Decomposition Programming*, Thèse de doctorat, University of California, Berkeley.

- BENDERS, J.F. (1962). Partitioning procedures for solving mixed-variables programming problems, *Numerische Mathematik*, Vol. 4, 238–252.
- BERGE, C. (1959). *Espaces topologiques et fonctions multivoques*, Dunod, Paris.
- BERGER, C., DUBOIS, R., HAURIE, A., LESSARD, E. et LOULOU, R. (1990). Assessing the dividends of power exchange between Quebec and New York State: A systems analysis approach, *International Journal of Energy Research*, Vol. 14, 253–273.
- BERGER, C., DUBOIS, R., HAURIE, A., LESSARD, E., LOULOU, R. et WAAUB, J.-P. (1992). Canadian MARKAL: An advanced linear programming system for energy and environmental modelling, *INFOR*, Vol. 30, No. 3, August 1992, 222–239.
- BERGER, C., DUBOIS, R., HAURIE, A. et LOULOU, R. (1990). Modelling electricity trading in the Northeast, *Energy Supply/Demand Balances: Options and Costs*, International Association for Energy Economics, Proceedings of the Twelfth Annual North American Conference, held in Ottawa, October 1-3, 1990, 304–315.
- BIHAIN, A. (1984). Optimality of upper semidifferentiable functions, *Journal of Optimization Theory and Applications*, Vol. 44, No. 4, December 1984, 545–568.

- CAUCHY, A. (1847). Méthode générale pour la résolution des systèmes d'équations simultanées, *Comptes rendus de l'Académie des Sciences de Paris*, Vol. 25, 536–538.
- CHENEY, E.W. et GOLDSTEIN, A.A. (1959). Newton's method for convex programming and Tchebycheff approximation, *Numerische Mathematik*, Vol. 1, No. 1, 253–268.
- CLARKE, F.H. (1975). Generalized gradients and applications, *Transactions of the American Mathematical Society*, Vol. 205, 247–262.
- CPLEX. (1992). *CPLEX Reference Manual. Using the CPLEX Callable Library and CPLEX Mixed Integer Library*, CPLEX Optimization, Inc., Incline Village, NV 89451-9436, U.S.A.
- DANTZIG, G.B. (1963). *Linear Programming and Extensions*, Princeton University Press, Princeton, New Jersey.
- DANTZIG, G.B. et WOLFE, P. (1960). Decomposition principle for linear programs, *Operations Research*, Vol. 8, 101–111.
- DANTZIG, G.B. et WOLFE, P. (1961). The decomposition algorithm for linear programs, *Econometrica*, Vol. 29, No. 4, October 1961, 767–778.
- DE GHELLINCK, G. et VIAL, J.-P. (1986). A polynomial Newton method for linear programming, *Algorithmica*, Vol. 1, 425–453.

DEM'YANOV, V.F. et MALOZEMOV, V.N. (1974). *Introduction to Minimax*, John Wiley & sons, New York, aussi paru en 1990 chez Dover Publications, Inc., New York.

DU MERLE, O., GOFFIN, J.-L. et VIAL, J.-P. (1996). *On the comparative behavior of Kelley's cutting plane method and the analytic center cutting plane method*, Technical Report 1996.4, Département d'Économie Commerciale et Industrielle, Université de Genève, Suisse, March 1996.

ELZINGA, J. et MOORE, T.G. (1975). A central cutting plane algorithm for the convex programming problem, *Mathematical Programming*, Vol. 8, No. 2, April 1975, 134-145.

ERMOLIEV, Y.M. (1966). Methods of solution of nonlinear extremal problems, *Kibernetika*, Vol. 2, No. 4, 1-17.

FEUER, A. (1974). *An Implementable Mathematical Programming Algorithm for Admissible Fundamental Functions*, Thèse de doctorat, Department of Mathematics, Columbia University, New York.

FISHBONE, L.G. et ABILOCK, H. (1981). MARKAL, a linear-programming model for energy systems analysis: Technical description of the BNL version, *Energy Research*, Vol. 5, 353-375.

- FISHBONE, L.G., GIESEN, G., GOLDSTEIN, G., HYMMEN, H.A., STOCKS, K.J., VOS, H., WILDE, D., ZÖLCHER, R., BALZER, C. et ABILOCK, H. (1983). *User's Guide for MARKAL (BNL/KFA Version 2.0): A Multi-Period, Linear-Programming Model for Energy Systems Analysis*, IEA Energy Technology Systems Analysis Project, Brookhaven National Laboratory, Upton, New York, U.S.A., et Kernforschungsanlage Jülich, Jülich, Germany, Report BNL 51701, July 1, 1983.
- FLETCHER, R. (1972). An algorithm for solving linearly constrained optimization problems, *Mathematical Programming*, Vol. 2. No. 2, April 1972, 133–165.
- FRANGIONI, A. (1996). Solving semidefinite quadratic problems within nonsmooth optimization algorithms, *Computers and Operations Research*, Vol. 23, No. 11, 1099–1118.
- FUKUSHIMA, M. (1983). An outer approximation algorithm for solving general convex programs, *Operations Research*, Vol. 31, No. 1, January-February 1983, 101–113.
- FUKUSHIMA, M. (1984). On the convergence of a class of outer approximation algorithms for convex programs, *Journal of Computational and Applied Mathematics*, Vol. 10, No. 1, February 1984, 147–156.
- GEOFFRION, A.M. (1970a). Primal resource-directive approaches for optimizing nonlinear decomposable systems, *Operations Research*, Vol. 18, No. 3, May-June 1970, 375–403.

- GEOFFRION, A.M. (1970b). Elements of large-scale mathematical programming. Part I: Concepts, *Management Science*, Vol. 16, No. 11, July 1970, 652–675.
- GEOFFRION, A.M. (1970c). Elements of large-scale mathematical programming. Part II: Synthesis of algorithms and bibliography, *Management Science*, Vol. 16, No. 11, July 1970, 676–691.
- GOFFIN, J.-L. (1977). On convergence rates of subgradient optimization methods, *Mathematical Programming*, Vol. 13, No. 3, December 1977, 329–347.
- GOFFIN, J.-L., GONDZIO, J., SARKISSIAN, R. et VIAL, J.-P. (1994). *Solving Nonlinear Multicommodity Flow Problems by the Analytic Center Cutting Plane Method*, GERAD, Ecole des Hautes Etudes Commerciales, Ecole Polytechnique et Université McGill, Montréal, Cahier du GERAD G-94-54, octobre 1994.
- GOFFIN, J.-L., HAURIE, A. et VIAL, J.-P. (1992). Decomposition and nondifferentiable optimization with the projective algorithm, *Management Science*, Vol. 38, No. 2, 284–302.
- GOFFIN, J.-L., HAURIE, A., VIAL, J.-P. et ZHU, D.L. (1993). Using central prices in the decomposition of linear programs, *European Journal of Operational Research*, Vol. 64, 393–409.

- GOFFIN, J.-L., LUO, Z.-Q. et YE, Y. (1994). On the complexity of a column generation algorithm for convex or quasiconvex feasibility problems, *Large Scale Optimization: State of the Art*, W.W. Hager, D.W. Hearn, P.M. Pardalos, Kluwer Academic Publishers, Dordrecht, Boston & London, 182–191.
- GOFFIN, J.-L. et VIAL, J.-P. (1990). Cutting planes and column generation techniques with the projective algorithm, *Journal of Optimization Theory and Applications*, Vol. 65, 409–429.
- GRINOLD, R.C. (1972). Steepest ascent for large scale linear programs, *SIAM Review*, Vol. 14, No. 3, July 1972, 447–464.
- HELD, M., WOLFE, P. et CROWDER, H.P. (1974). Validation of subgradient optimization, *Mathematical Programming*, Vol. 6, 62–88.
- HESTENES, M.R. (1969). Multiplier and gradient methods, *Journal of Optimization Theory and Applications*, Vol. 4, No. 5, 303–320.
- HIRIART-URRUTY, J.-B. (1982). ε -subdifferential calculus, *Convex Analysis and Optimization*, J.-P. Aubin, R.B. Vinter, Pitman, Boston, 43–92.
- HIRIART-URRUTY, J.-B. et LEMARÉCHAL, C. (1993a). *Convex Analysis and Minimization Algorithms I - Fundamentals*, Springer-Verlag, Berlin.
- HIRIART-URRUTY, J.-B. et LEMARÉCHAL, C. (1993b). *Convex Analysis and Minimization Algorithms II - Advanced Theory and Bundle Methods*, Springer-Verlag, Berlin.

- HO, J.K. et LOUTE, E. (1978). On decentralization of multi-regional production-distribution optimization systems, *Decomposition of Multi-Regional Energy Flow Optimization Models*, B., Culot, E., Loute, J., Ho, Center for Operations Research and Econometrics, Université Catholique de Louvain, Louvain-la-Neuve, Belgique, September 1978, Chapter VI, 81–94.
- HO, J.K. et LOUTE, E. (1981). An advanced implementation of the Dantzig-Wolfe algorithm for linear programming, *Large-Scale Linear Programming*, G.B. Dantzig, M.A.H. Dempster, M. Kallio, International Institute for Applied Systems Analysis, Laxenburg, Austria, 425–460.
- HO, J.K. et LOUTE, E. (1983). Computational experience with advanced implementation of decomposition algorithms for linear programming, *Mathematical Programming*, Vol. 27, 283–290.
- HO, J.K. et MANNE, A.S. (1974). Nested decomposition for dynamic models, *Mathematical Programming*, Vol. 6, 121–140.
- HOGAN, W.W. (1975). Energy policy models for Project Independence, *Computers and Operations Research*, Vol. 2, 251–271.
- KALLIO, M. et PORTEUS, E.L. (1977). Decomposition of arborescent linear programs, *Mathematical Programming*, Vol. 13, 348–356.
- KARMAKAR, N.K. (1984). A new polynomial-time algorithm for linear programming, *Combinatorica*, Vol. 4, 373–395.

- KELLEY, J.E. (1960). The cutting-plane method for solving convex programs, *Journal of the Society for Industrial and Applied Mathematics*, Vol. 8, No. 4, March 1960, 703–712.
- KIWIEL, K.C. (1983). An aggregate subgradient method for nonsmooth convex minimization, *Mathematical Programming*, Vol. 27, No. 3, December 1983, 320–341.
- KIWIEL, K.C. (1984). A linearization algorithm for constrained nonsmooth minimization, *System Modelling and Optimization*, P. Thoft-Christensen, Springer-Verlag, Berlin, 311–320.
- KIWIEL, K.C. (1985a). *Methods of Descent for Nondifferentiable Optimization*, Lecture Notes in Mathematics, 1133, Springer-Verlag, Berlin.
- KIWIEL, K.C. (1985b). An algorithm for linearly constrained convex nondifferentiable minimization problems, *Journal of Mathematical Analysis and Applications*, Vol. 105, No. 2, February 1985, 452–465.
- KIWIEL, K.C. (1986). A method for solving certain quadratic programming problems arising in nonsmooth optimization, *IMA Journal of Numerical Analysis*, Vol. 6, 137–152.
- KIWIEL, K.C. (1987). Decomposition method of descent for minimizing the sum of convex nonsmooth functions, *Journal of Optimization Theory and Applications*, Vol. 52, No. 2, February 1987, 255–271.

- KIWIEL, K.C. (1989a). A survey of bundle methods for nondifferentiable optimization, *Proceedings XII International Symposium on Mathematical Programming*, KTK Scientific Publishers, Tokyo, 263–282.
- KIWIEL, K.C. (1989b). A dual method for certain positive semidefinite quadratic programming problems, *SIAM Journal on Scientific and Statistical Computing*, Vol. 10, No. 1, January 1989, 175–186.
- KIWIEL, K.C. (1990). Proximity control in bundle methods for convex nondifferentiable minimization, *Mathematical Programming*, Vol. 46, No. 1, January 1990, 105–122.
- KIWIEL, K.C. (1991). Exact penalty functions in proximal bundle methods for constrained convex nondifferentiable minimization, *Mathematical Programming*, Vol. 52, No. 2, August 1991, 285–302.
- KIWIEL, K.C. (1994). A Cholesky dual method for proximal piecewise linear programming, *Numerische Mathematik*, Vol. 68, No. 3, September 1994, 325–340.
- KIWIEL, K.C. (1995). Approximations in proximal bundle methods and decomposition of convex programs, *Journal of Optimization Theory and Applications*, Vol. 84, No. 3, March 1995, 529–548.
- LASDON, L.S. (1968). Duality and decomposition in mathematical programming, *IEEE Transactions on Systems Science and Cybernetics*, Vol. SSC-4, No. 2, July 1968, 86–100.

- LASDON, L.S. (1970). *Optimization Theory for Large Systems*, MacMillan Publishing Co., Inc., New York.
- LEMARÉCHAL, C. (1975). An extension of Davidon methods to non differentiable problems, *Mathematical Programming Study*, Vol. 3, November 1975, 95–109.
- LEMARÉCHAL, C. (1978a). Bundle methods in nonsmooth optimization, *Nonsmooth Optimization*, C. Lemaréchal, R. Mifflin, Pergamon Press, 79–102.
- LEMARÉCHAL, C. (1978b). *Nonsmooth Optimization and Descent Methods*, International Institute for Applied Systems Analysis, Laxenburg, Austria, Report RR-78-4, March 1978.
- LEMARÉCHAL, C. (1981). A view of line-searches, *Optimization and Optimal Control*, A. Auslender, W. Oettli, J. Stoer, Springer-Verlag, Berlin, 59–78.
- LEMARÉCHAL, C. (1986). An introduction to the theory of nonsmooth optimization, *Optimization*, Vol. 17, No. 6, 827–858.
- LEMARÉCHAL, C. (1989). Nondifferentiable optimization, *Handbooks in Operations Research and Management Science, Volume 1: Optimization*, G.L. Nemhauser, A.H.G. Rinnooy Kan, M.J. Todd, North-Holland, 529–572.
- LEMARÉCHAL, C. (1992). Lagrangian decomposition and nonsmooth optimization: Bundle algorithm, prox iteration, augmented lagrangian, *Nonsmooth Optimization Methods and Applications*, F. Giannessi, Gordon and Breach Science Publishers, 201–216.

- LEMARÉCHAL, C., NEMIROVSKII, A. et NESTEROV, Y. (1995). New variants of bundle methods, *Mathematical Programming*, Vol. 69, 111–147.
- LEMARÉCHAL, C., STRODIOT, J.J. et BIHAIN, A. (1981). On a bundle algorithm for nonsmooth optimization, *Nonlinear Programming 4*, O.L. Mangasarian, R.R. Meyer, S.M. Robinson, Academic Press, Inc., New York, 245–282.
- LOULOU, R., SAVARD, G. et LAVIGNE, D. (1992). *Decomposition of Multi-Player Linear Programs*, GERAD, Ecole des Hautes Etudes Commerciales, Ecole Polytechnique et Université McGill, Montréal, Cahier du GERAD G-92-17, April 1992.
- LUENBERGER, D.G. (1984). *Linear and Nonlinear Programming*, Second Edition, Addison-Wesley Publishing Company.
- MADSEN, K. (1975). An algorithm for minimax solution of overdetermined systems of nonlinear equations, *Journal of the Institute of Mathematics and its Applications*, Vol. 16, 321–328.
- MAHEY, P. (1982). Decomposition of large-scale linear programs by subgradient optimization, *Mat. Aplic. Comp.*, Vol. 1, No. 2, 121–134.
- MAHEY, P. (1986). Méthodes de décomposition et décentralisation en programmation linéaire, *RAIRO Recherche opérationnelle*, Vol. 20, No. 4, novembre 1986, 287–306.

- MARSTEN, R.E., HOGAN, W.W. et BLANKENSHIP, J.W. (1975). The BOXSTEP method for large-scale optimization, *Operations Research*, Vol. 23, No. 3, May-June 1975, 389-405.
- MARTINET, B. (1970). Régularisation d'inéquations variationnelles par approximations successives, *R.I.R.O.*, Revue Française de recherche opérationnelle, Vol. R-3, 154-158.
- MANGASARIAN, O.L. (1969). *Nonlinear Programming*, McGraw-Hill Book Company, New York.
- MEDHI, D. (1987). *Decomposition of Structured Large-Scale Optimization Problems and Parallel Optimization*, Thèse de doctorat, Computer Sciences Department, University of Wisconsin-Madison, Computer Sciences Technical Report #718, September 1987.
- MEDHI, D. (1994). Bundle-based decomposition for large-scale convex optimization: Error estimate and application to block-angular linear programs, *Mathematical Programming*, Vol. 66, No. 1, August 2, 1994, 79-101.
- MIFFLIN, R. (1977a). An algorithm for constrained optimization with semismooth functions, *Mathematics of Operations Research*, Vol. 2, No. 2, May 1977, 191-207.
- MIFFLIN, R. (1977b). Semismooth and semiconvex functions in constrained optimization, *SIAM Journal on Control and Optimization*, Vol. 15, No. 6, November 1977, 959-972.

- MIFFLIN, R. (1982). A modification and an extension of Lemaréchal's algorithm for nonsmooth minimization, *Mathematical Programming Study*, Vol. 17, April 1982, 77–90.
- MINOUX, M. (1983). *Programmation mathématique - Théorie et algorithmes*, Tomes 1 et 2, Dunod, Paris.
- NEMHAUSER, G.L. et WIDHELM, W.B. (1971). A modified linear program for columnar methods in mathematical programming, *Operations Research*, Vol. 19, No. 4, July-August 1971, 1051–1060.
- NURMINSKI, E. et BALABANOV, T. (1983). Decomposition of a large-scale energy model, *Large scale systems*, Vol. 4, 295–308.
- O'NEILL, R.P. et WIDHELM, W.B. (1975). Computational experience with normed and nonnormed column-generation procedures in nonlinear programming, *Operations Research*, Vol. 23, No. 2, March-April 1975, 372–382.
- ORTEGA, J.M. (1972). *Numerical Analysis: A Second Course*, Academic Press, New York and London.
- POLAK, E. (1971). *Computational Methods in Optimization: A Unified Approach*, Academic Press, New York and London.

- POLAK, E., MAYNE, D.Q. et WARDI, Y. (1983). On the extension of constrained optimization algorithms from differentiable to nondifferentiable problems, *SIAM Journal on Control and Optimization*, Vol. 21, No. 2, March 1983, 179–203.
- POLYAK, B.T. (1967). A general method of solving extremum problems, *Soviet Mathematics Dokl.*, Vol. 8, No. 3, 593–597.
- POLYAK, B.T. (1969). Minimization of unsmooth functionals, *USSR Computational Mathematics and Mathematical Physics*, Vol. 9, 14–29.
- POWELL, M.J.D. (1969). A method for nonlinear constraints in minimization problems, *Optimization*, R. Fletcher, Academic Press, 283–298.
- ROBINSON, S.M. (1986). Bundle-based decomposition: Description and preliminary results, *System Modelling and Optimization*, A. Prépoka, J. Szelezsan, B. Strazicky, Springer-Verlag, Berlin, 751–756.
- ROBINSON, S.M. (1989). Bundle-based decomposition: Conditions for convergence, *Annales de l'Institut Henri Poincaré: Analyse Non Linéaire*, Vol. 6, 435–447.
- ROCKAFELLAR, O.L. (1970). *Convex Analysis*, Princeton University Press, Princeton, New Jersey.

- ROCKAFELLAR, O.L. (1976). Augmented lagrangians and applications of the proximal point algorithm in convex programming, *Mathematics of Operations Research*, Vol. 1, No. 2, May 1976, 97–116.
- RUDIN, W. (1976). *Principles of Mathematical Analysis*, Third Edition, McGraw-Hill, Inc., New York.
- RUSZCZYNSKI, A. (1986). A regularized decomposition method for minimizing a sum of polyhedral functions, *Mathematical Programming*, Vol. 35, 309–333.
- SCHRAMM, H. et ZOWE, J. (1992). A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results, *SIAM Journal on Optimization*, Vol. 2, No. 1, February 1992, 121–152.
- SHAPIRO, J.F. (1979). *Mathematical Programming: Structures and Algorithms*, John Wiley & Sons.
- SHOR, N.Z. (1968). On the rate of convergence of the generalized gradient method, *Kibernetika*, Vol. 6, No. 2, 98–99.
- SHOR, N.Z. (1970). Convergence rate of the gradient descent method with dilatation of the space, *Cybernetics*, 102–108.
- SILVERMAN, G.J. (1972a). Primal decomposition of mathematical programs by resource allocation: I - Basic theory and a direction-finding procedure, *Operations Research*, Vol. 20, No. 1, January-February 1972, 58–74.

- SILVERMAN, G.J. (1972b). Primal decomposition of mathematical programs by resource allocation: II - Computational algorithm with an application to the modular design problem, *Operations Research*, Vol. 20, No. 1, January-February 1972, 75–93.
- SIMONNARD, M. (1972). *Programmation linéaire, technique du calcul économique - 1. Fondements*, deuxième édition, Dunod, Paris.
- SONNEVEND, G. (1985). An analytical center for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming, *Lecture Notes in Control and Information Sciences* 84, Springer, New York, 866–876.
- STOER, J. (1971). On the numerical solution of constrained least-squares problems, *SIAM Journal on Numerical Analysis*, Vol. 8, 382–411.
- STREIT, R.L. (1986). Solutions of systems of complex linear equations in the ℓ_∞ norm with constraints on the unknowns, *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 1, January 1986, 132–149.
- STRODIOT, J.-J., NGUYEN, V.H. et HEUKEMES, N. (1983). ϵ -optimal solutions in nondifferentiable convex programming, *Mathematical Programming*, Vol. 15, No. 3, April 1983, 307–328.
- TOPKIS, D.M. (1970). Cutting-plane methods without nested constraint sets, *Operations Research*, Vol. 18, No. 3, May-June 1970, 404–413.

- WETS, R.J.B. (1966). Programming under uncertainty: The equivalent convex program, *SIAM Journal*, Vol. 14, No. 1, January 1966, 89–105.
- WOLFE, P. (1961). Accelerating the cutting plane method for nonlinear programming, *SIAM*, Vol. 9, No. 3, September 1961, 481–488.
- WOLFE, P. (1974). A method of conjugate subgradients for minimizing nondifferentiable functions, *Proceedings of Twelfth Annual Allerton Conference on Circuit and System Theory*, Illinois, 8–15.
- WOLFE, P. (1975). A method of conjugate subgradients for minimizing nondifferentiable functions, *Mathematical Programming Study*, Vol. 3, November 1975, 145–173.
- ZANGWILL, W.I. (1969). *Nonlinear Programming: A Unified Approach*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- ZIPKIN, P.H. (1980). Bounds for row-aggregation in linear programming, *Operations Research*, Vol. 28, No. 4, July-August 1980, 903–916.
- ZOWE, J. (1985). Nondifferentiable Optimization, *Computational Mathematical Programming*, NATO ASI Series, Vol. F15, K. Schittkowski, Springer-Verlag, Berlin, Heidelberg, 323–356.
- ZSCHAU, E.V.W. (1966). *A Primal Decomposition Algorithm for Linear Programming*, Thèse de doctorat, Stanford University, December 1966.

Annexe A Rappels d'algèbre linéaire et de programmation linéaire

Lemme de Farkas et Minkowski

Quels que soient la matrice A et le vecteur d , le système d'équations linéaires

$$\begin{aligned} Ax &= d \\ x &\geq 0 \end{aligned}$$

a une solution $\bar{x}$;
si et seulement si

$$u^T d \leq 0 \quad \text{pour tout } \bar{u} \text{ vérifiant } \bar{u}^T A \leq 0.$$

Démonstration: Voir, entre autres, Mangasarian (1969) ou Simonnard (1972). ■

Le prochain résultat nous permet d'évaluer la dérivée directionnelle de la fonction valeur (ou de perturbation) associée à un problème de programmation linéaire.

Théorème A.1 *Supposons que la fonction $v(\cdot)$ définie par*

$$v(b) \triangleq \min_x c^T x$$

sous les contraintes

$$\begin{aligned} Ax &\geq b \\ x &\geq 0, \end{aligned}$$

dont la formulation duale est

$$\max_{\mu} \mu^T b$$

(D)

sous les contraintes

$$\mu^T A \leq c^T$$

$$\mu \geq 0,$$

prenne une valeur finie en b et que $d \neq 0$ soit une direction réalisable de v en b , c'est-à-dire une direction telle que $v(b + \theta d)$ prenne des valeurs finies pour tout $\theta \in [0, \theta^*]$ avec $\theta^* > 0$. Alors la dérivée directionnelle $\nabla v(b : d)$ de v en b dans la direction d existe et sa valeur est donnée par

$$\nabla v(b : d) = \max \{ \mu^T d : \mu \text{ est une solution optimale du problème dual } D \}.$$

Démonstration: Shapiro (1979), théorème 2.2. ■

Note: On notera la similitude existant entre ce résultat et la dernière partie du théorème B.5.

Annexe B Rappels d'analyse convexe

Afin de rendre le texte plus simple à lire, nous avons réduit le nombre de renvois à des définitions et résultats d'analyse convexe qui ne sont peut-être pas connus de tous en regroupant quelques-uns dans cette annexe. Celle-ci comporte deux parties: la première est consacrée à des concepts et résultats plus généraux tandis que la seconde est dédiée à l'étude des fonctions norme.

B.1 Résultats généraux

Théorème B.1 *Un ensemble convexe fermé est égal à l'intersection de tous les demi-espaces fermés qui le contiennent.*

Démonstration: Hiriart-Urruty et Lemaréchal (1993a), théorème III.4.2.4, ou Rockafellar (1970), théorème 11.5. ■

Définition. Une fonction convexe $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est dite *propre* si son épigraphe n'est pas vide et ne contient pas de ligne verticale, i.e. si

- (i) $\exists x \in \mathbb{R}^n$ tel que $f(x) < +\infty$; et
- (ii) $f(x) > -\infty, \quad \forall x \in \mathbb{R}^n$.

Conséquence: Une fonction convexe f est propre si et seulement si l'ensemble convexe

$$\text{dom } f \triangleq \{x \in \mathbb{R}^n : f(x) < +\infty\},$$

appelé *domaine* de f , est non-vidé et la restriction de f à son domaine ne prend que des valeurs finies.

Définition. Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est dite *inf-compacte* si

$$\{x \in \mathbb{R}^n : f(x) \leq \lambda\}$$

est compact pour toute valeur de $\lambda \in \mathbb{R}$.

Définition. Une fonction convexe $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est dite *fermée* si elle est partout semi-continue inférieure, i.e. si

$$\liminf_{y \rightarrow x} f(y) \geq f(x), \quad \forall x \in \mathbb{R}^n$$

ou si son épigraphe est un ensemble fermé.

Proposition B.2 (*Hiriart-Urruty et Lemaréchal, 1993a: proposition IV.2.1.2*) Si $\{f_j\}_{j \in J}$ est une famille de fonctions convexes (fermées) de $\mathbb{R}^n$ vers $\mathbb{R}$ et s'il existe $x_0 \in \mathbb{R}^n$ tel que $\sup\{f_j(x_0) : j \in J\} < +\infty$, alors la fonction f définie par

$$f(x) \triangleq \sup\{f_j(x) : j \in J\}, \quad \forall x \in \mathbb{R}^n$$

est convexe (fermée) dans $\mathbb{R}^n$.

Théorème B.3 (*Hiriart-Urruty et Lemaréchal, 1993a: théorème IV.3.1.5*) Si les fonctions convexes $f_j : \mathbb{R}^n \rightarrow \mathbb{R}$ convergent vers $f : \mathbb{R}^n \rightarrow \mathbb{R}$, i.e. si

$$\lim_{j \rightarrow \infty} f_j(x) = f(x), \quad \forall x \in \mathbb{R}^n,$$

alors la fonction f est convexe et la convergence est uniforme sur tout sous-ensemble compact S de $\mathbb{R}^n$.

Définition. Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite *lipschitzienne* ou *fonction de Lipschitz* s'il existe une constante $L < \infty$, dite *constante de Lipschitz*, telle que

$$|f(x) - f(y)| \leq L\|x - y\|, \quad \forall x, y \in \mathbb{R}^n.$$

Elle est dite *localement lipschitzienne* si, pour tout sous-ensemble borné B de $\mathbb{R}^n$, il existe une constante de Lipschitz $L = L(B) < +\infty$ telle que

$$|f(x) - f(y)| \leq L\|x - y\|, \quad \forall x, y \in B.$$

Par conséquent, les fonctions lipschitziennes et localement lipschitziennes sont continues.

Théorème B.4 (*Hiriart-Urruty et Lemaréchal, 1993a: théorème IV.3.1.2*) Si $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est une fonction convexe et si S est un sous-ensemble convexe compact de $\text{ri dom } f$ (i.e. l'intérieur relatif du domaine de f), alors il existe $L = L(S) \geq 0$ tel que

$$|f(x) - f(y)| \leq L\|x - y\|, \quad \forall x, y \in S.$$

Par conséquent, f est continue et localement lipschitzienne dans l'intérieur relatif de son domaine.

Définition. Étant donné une fonction convexe $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, on appelle *sous-différentiel* de f en x , lorsqu'il existe, l'ensemble $\partial f(x)$ défini par

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

et ses éléments sont les *sous-gradients* de f en x .

Théorème B.5 Soit une fonction convexe $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$

- (a) Si f est à valeurs finies dans $\mathbb{R}^n$, alors $\partial f(x)$ est compact et non-vide, $\forall x \in \mathbb{R}^n$.
- (b) Si f est convexe propre (i.e. à valeurs finies dans son domaine), alors $\partial f(x)$ est compact et non-vide si et seulement si $x \in \text{int}(\text{dom } f)$.
- (c) Si f est convexe propre, alors $\partial f(x) \neq \emptyset$, $\forall x \in \text{ri}(\text{dom } f)$, i.e. $\forall x$ faisant partie de l'intérieur relatif de $\text{dom } f$.
- (d) Si f est convexe propre, alors $\partial f(x) = \emptyset$, $\forall x \notin \text{dom } f$.
- (e) Si $x \in \text{ri}(\text{dom } f)$, alors la dérivée directionnelle $\nabla f(x : d)$ de la fonction f en x dans la direction d est donnée par

$$\nabla f(x : d) = \sup \{\langle s, d \rangle : s \in \partial f(x)\}.$$

Démonstration: Rockafellar (1970), théorème 23.4. ■

Remarque: La dernière partie du résultat peut être comparée au théorème A.1.

Proposition B.6 (*Hiriart-Urruty et Lemaréchal, 1993a: théorème VI.6.2.2; voir aussi Lemaréchal, 1986*) Si $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est une fonction convexe, alors la multi-application ∂f est localement bornée dans l'intérieur de $\text{dom } f$. Ceci veut dire que l'image $\partial f(B)$ de tout sous-ensemble borné $B \subseteq \text{int}(\text{dom } f)$ est un sous-ensemble borné de $\mathbb{R}^n$.

Théorème B.7 (*Hiriart-Urruty et Lemaréchal, 1993a: théorème VI.4.1.1*) Si f_1 et f_2 sont deux fonctions convexes de $\mathbb{R}^n$ vers $\mathbb{R}$, alors

$$\begin{aligned} \partial(t_1 f_1 + t_2 f_2)(x) &= t_1 \partial f_1(x) + t_2 \partial f_2(x) \quad \forall t_1 > 0, t_2 > 0 \\ &\quad \forall x \in \mathbb{R}^n \end{aligned}$$

Théorème B.8 (*Rockafellar, 1970: théorème 23.8*) Si $f_1, \dots, f_m$ sont des fonctions convexes propres de $\mathbb{R}^n$ vers $\mathbb{R} \cup \{+\infty\}$ et si $f \triangleq f_1 + \dots + f_m$, on a alors

$$\partial f(x) \supseteq \partial f_1(x) + \dots + \partial f_m(x), \quad \forall x \in \mathbb{R}^n.$$

Si les ensembles convexes $\text{ri}(\text{dom } f_i)$, $i = 1, \dots, m$, ont un point en commun, alors

$$\partial f(x) = \partial f_1(x) + \dots + \partial f_m(x), \quad \forall x \in \mathbb{R}^n.$$

Si les fonctions f_i , $i = 1, \dots, m$, sont polyédriques, il suffit que les ensembles $\text{dom } f_i$ aient un point en commun pour que l'égalité ci-dessus soit vérifiée.

Théorème B.9 (*Hiriart-Urruty et Lemaréchal, 1993a: théorème VI.4.3.1*) Si $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est une fonction convexe et si $g : \mathbb{R} \rightarrow \mathbb{R}$ est une fonction convexe croissante d'une seule variable, alors

$$\partial(g \circ f)(x) = \left\{ \rho s : \rho \in \partial g(f(x)), s \in \partial f(x) \right\}.$$

Définition. (Hiriart-Urruty et Lemaréchal, 1993a) Étant donné un sous-ensemble non-vide S de $\mathbb{R}^n$, on appelle *fonction d'appui* de S la fonction $\sigma_S : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ définie par

$$\sigma_S(x) \triangleq \sup \{ \langle s, x \rangle : s \in S \}, \quad \forall x \in \mathbb{R}^n.$$

Proposition B.10 *Les fonctions d'appui sont convexes.*

Proposition B.11 (*Hiriart-Urruty et Lemaréchal, 1993a: proposition V.2.1.3*) La fonction d'appui d'un ensemble S prend partout des valeurs finies si et seulement si S est borné.

Construction dans $\mathbb{R}^n$. Étant donné $S \subset \mathbb{R}^n$ et $d \neq 0$, $d \in \mathbb{R}^n$, considérons pour chaque $r \in \mathbb{R}$ le demi-espace fermé ci-dessous:

$$H_{d,r}^- \triangleq \{ z \in \mathbb{R}^n : \langle z, d \rangle \leq r \}.$$

Supposons que nous puissions trouver r suffisamment grand tel que $S \subseteq H_{d,r}^-$. La valeur $\sigma_S(d)$ est le plus petit des r tels que $S \subseteq H_{d,r}^-$. Il s'agit donc d'appuyer sur S l'hyperplan

$$H_{d,r} \triangleq \{ z \in \mathbb{R}^n : \langle z, d \rangle = r \}.$$

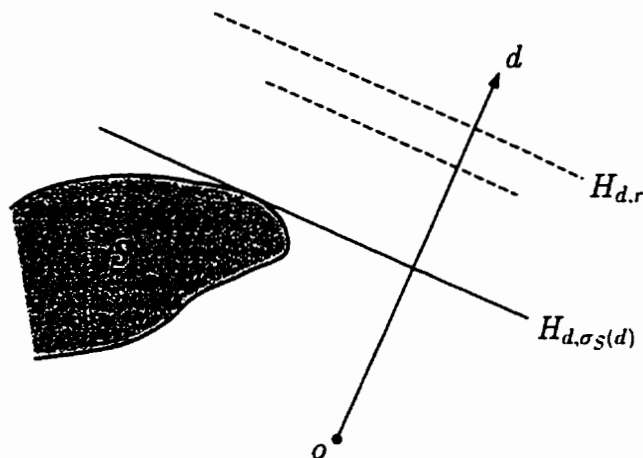


Figure B.1 Interprétation géométrique de la fonction d'appui

Remarque: $\sigma_S(d) = +\infty$ si S n'est pas borné dans la direction d .

Remarque: Lorsque $\sigma_S(d) < +\infty$, $H_{d, \sigma_S(d)}$ est un hyperplan d'appui de S tandis que pour $r > \sigma_S(d)$, on a

$$S \subset H_{d,r}^-.$$

De plus, d est normal à $H_{d,r}$, $\forall r$.

Définition. Étant donné un sous-ensemble non-vidé S de $\mathbb{R}^n$, on appelle *fonction indicatrice* de S la fonction $I_S : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ définie par

$$I_S(x) = \begin{cases} 0 & \text{si } x \in S \\ +\infty & \text{sinon.} \end{cases}$$

Proposition B.12 *Soit un sous-ensemble non-vidé S de $\mathbb{R}^n$. La fonction indicatrice I_S est convexe si et seulement si S est convexe.*

Proposition B.13 *Soit une fonction convexe $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ et un ensemble convexe non-vidé $C \subseteq \mathbb{R}^n$. Si*

$$\text{dom } f \cap C \neq \emptyset$$

alors la fonction $\varphi : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ définie par

$$\varphi(x) \triangleq \begin{cases} f(x) & \text{si } x \in C \\ +\infty & \text{sinon} \end{cases}$$

est convexe.

De plus, $\varphi = f + I_C$.

Définition. (Hiriart-Urruty et Lemaréchal, 1993b) On appelle *conjuguée* d'une fonction convexe $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ ($f \not\equiv +\infty$), la fonction $f^* : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ définie par

$$f^*(s) = \sup \{ \langle s, x \rangle - f(x) : x \in \text{dom } f \}, \quad \forall s \in \mathbb{R}^n.$$

Remarque: Le concept de conjuguée s'applique aussi à des fonctions qui ne sont pas convexes. Dans ce cas, il faut que la fonction soit minorée dans $\mathbb{R}^n$ par une fonction affine.

Proposition B.14 Soit I_C la fonction indicatrice d'un ensemble convexe non-vidé $C \subseteq \mathbb{R}^n$. On a alors

$$\begin{aligned} (I_C)^*(s) &= \sup_{x \in \text{dom } I_C} \{\langle s, x \rangle - I_C(s)\} \\ &= \sup_{x \in C} \langle s, x \rangle \\ &= \sigma_C(s) \end{aligned}$$

i.e.

$$I_C^* = \sigma_C.$$

Définition. Étant donné un ensemble convexe fermé non-vidé $C \subseteq \mathbb{R}^n$ et un point $x \in C$, on appelle *cône normal* de C en x , l'ensemble $N_C(x)$ défini par

$$N_C(x) \triangleq \{s \in \mathbb{R}^n : \langle s, y - x \rangle \leq 0, \quad \forall y \in C\}.$$

Les directions $s \in N_C(x)$ sont dites *normales* à C en $x \in C$: ce sont les directions qui ne font pas d'angle aigu avec chacun des segments de droite originant de x contenus dans C .

Il existe une relation entre la fonction indicatrice d'un ensemble convexe C et son cône normal en chaque point $x \in C$.

Proposition B.15 (Rockafellar, 1970: pages 215-216) Si C est un sous-ensemble convexe fermé non-vidé de $\mathbb{R}^n$, alors

$$\partial I_C(x) = N_C(x), \quad \forall x \in C$$

et

$$\partial I_C(x) = \emptyset, \quad \forall x \in \mathbb{R}^n \setminus C.$$

Ce dernier résultat peut être généralisé. Pour cela, nous aurons besoin de deux nouvelles définitions.

Définition. Soit $\varepsilon \geq 0$ et une fonction convexe $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$. On appelle ε -sous-différentiel de f en x , l'ensemble $\partial_\varepsilon f(x)$ défini par

$$\partial_\varepsilon f(x) \triangleq \{s \in \mathbb{R}^n : f(y) \geq f(x) + \langle s, y - x \rangle - \varepsilon, \quad \forall y \in \mathbb{R}^n\}$$

et ses éléments sont les ε -sous-gradients de f en x .

Remarque: Un analogue du théorème B.5 existe pour la multi-application $\partial_\varepsilon f$ (Lemaréchal, 1986).

Proposition B.16 (Hiriart-Urruty et Lemaréchal, 1993b: proposition XI.4.1.1) Si $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est une fonction convexe fermée et si la suite $\{(\varepsilon^k, x^k, s^k)\}$ a $(\varepsilon, x, s) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n$ pour limite, où $s^k \in \partial_{\varepsilon^k} f(x^k)$, alors $s \in \partial_\varepsilon f(x)$.

Définition. (Hiriart-Urruty, 1982; Strodriot, Nguyen et Heukemes, 1983) Étant donné un ensemble convexe fermé non-vide $C \subseteq \mathbb{R}^n$, un point $x \in C$ et une constante $\varepsilon \geq 0$, on appelle *ensemble ε -normal* de C en x , l'ensemble $N_\varepsilon(C; x)$ défini par

$$N_\varepsilon(C; x) \triangleq \{s \in \mathbb{R}^n : \langle s, y - x \rangle \leq \varepsilon, \quad \forall y \in C\}.$$

Les directions $s \in N_\varepsilon(C; x)$ sont dites ε -normales à C en $x \in C$.

Remarque: Lorsque $\varepsilon = 0$, la définition précédente est celle du cône normal de C en x , i.e.

$$N_0(C; x) = N_C(x), \quad \forall x \in C.$$

De plus, on a $N_C(x) \subseteq N_\varepsilon(C; x)$, $\forall \varepsilon \geq 0$.

Remarque: Contrairement à $N_C(x)$, $N_\varepsilon(C; x)$ n'est pas un cône. Il s'agit toutefois d'un ensemble convexe fermé contenant $N_C(x)$.

Proposition B.17 (*Strodiot, Nguyen et Heukemes, 1983*) Si C est un sous-ensemble convexe fermé non-vide de $\mathbb{R}^n$ et $\varepsilon \geq 0$, alors

$$N_\varepsilon(C; x) = \partial_\varepsilon I_C(x) = \{s \in \mathbb{R}^n : (I_C)^*(s) \leq \langle s, x \rangle + \varepsilon\}, \quad \forall x \in C.$$

Combinant les propositions B.14 et B.17 on obtient le résultat suivant.

Corollaire B.18 *Sous les hypothèses de la proposition B.17, on a*

$$N_\varepsilon(C; x) = \partial_\varepsilon I_C(x) = \{s \in \mathbb{R}^n : \sigma_C(s) \leq \langle s, x \rangle + \varepsilon\}, \quad \forall x \in C,$$

où σ_C désigne la fonction d'appui à C .

Théorème B.19 (*Hiriart-Urruty et Lemaréchal, 1993a: théorème VI.2.2.1*) Si $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est une fonction convexe, alors les trois propriétés suivantes sont équivalentes:

- (i) $x \in \mathbb{R}^n$ minimise f , i.e. $f(y) \geq f(x)$, $\forall y \in \mathbb{R}^n$;
- (ii) $0 \in \partial f(x)$;
- (iii) $\nabla f(x : d) \geq 0$, $\forall d \in \mathbb{R}^n$, où $\nabla f(x : d)$ désigne la dérivée directionnelle de f au point x dans la direction d .

Théorème B.20 (Rockafellar, 1970: théorème 27.4) *Considérons le problème d'optimisation convexe*

$$\inf \{f(x) : x \in C\}$$

où $C \subseteq \mathbb{R}^N$ est un ensemble convexe fermé non-vide, $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est une fonction convexe propre, et $\text{ri}(\text{dom } f) \cap \text{ri}(C) \neq \emptyset$.

Alors $\bar{x} \in C$ minimise f sur C si et seulement si

$$0 \in \partial f(\bar{x}) + N_C(\bar{x}),$$

i.e. si et seulement si

$$0 \in \partial f(\bar{x}) + I_C(\bar{x}).$$

Remarque: La condition $\text{ri}(\text{dom } f) \cap \text{ri}(C) \neq \emptyset$ peut être remplacée par $\text{ri}(\text{dom } f) \cap C \neq \emptyset$ lorsque C est un ensemble polyédrique.

Remarque: Contrairement à ∂f , on a toujours $0 \in \partial I_C(x)$, $\forall x \in C$. Ceci découle du fait que $\partial I_C(x) = N_C(x) \ni 0$, par la proposition B.15.

B.2 Normes et normes duales

Définition. Une *norme* dans $\mathbb{R}^n$ est une fonction $\| \cdot \| : \mathbb{R}^n \rightarrow \mathbb{R}_+ = [0, +\infty]$ munie des trois propriétés suivantes:

- (a) $\|x\| = 0 \Leftrightarrow x = 0$,
- (b) $\|\alpha x\| = |\alpha| \|x\|$, $\forall x \in \mathbb{R}^n$, $\forall \alpha \in \mathbb{R}$; et
- (c) $\|x + y\| \leq \|x\| + \|y\|$, $\forall (x, y) \in \mathbb{R}^n \times \mathbb{R}^n$.

Les prochaines définitions présentent certaines familles de normes.

Définition. Soit $p \in [1, \infty)$. La fonction $\| \cdot \|_p : \mathbb{R}^n \rightarrow \mathbb{R}_+$ définie par

$$\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}}, \quad \forall x \in \mathbb{R}^n \quad (\text{B1})$$

est dite *norme* ℓ_p .

Définition. La fonction $\| \cdot \|_\infty : \mathbb{R}^n \rightarrow \mathbb{R}_+$ définie par

$$\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|, \quad \forall x \in \mathbb{R}^n \quad (\text{B2})$$

est dite *norme* ℓ_∞ .

Remarque: La norme ℓ_∞ est obtenue comme cas limite des normes ℓ_p .

Définition. Si Q est une matrice symétrique définie positive, alors la fonction $\|\cdot\|_Q : \mathbb{R}^n \rightarrow \mathbb{R}_+$ définie par

$$\|x\|_Q = \sqrt{\langle Qx, x \rangle}, \quad \forall x \in \mathbb{R}^n \quad (\text{B3})$$

est dite *norme quadratique*.

Remarques:

- (a) Ortega (1972) qualifie la fonction $\|\cdot\|_Q$ de *norme elliptique*.
- (b) La norme ℓ_2 , ou euclidienne, est une norme quadratique particulière obtenue en posant $Q = I$.

Les normes ℓ_1 , ℓ_2 , ℓ_∞ et quadratiques sont, de très loin, les normes les plus utilisées.

Soit $\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}_+$ une norme quelconque.

Proposition B.21 (*Hiriart-Urruty et Lemaréchal, 1993a: proposition V.3.2.1, page 221*) La fonction $\|\cdot\|^* : \mathbb{R}^n \rightarrow \mathbb{R}_+$ définie par

$$\|s\|^* \triangleq \max \{ \langle s, x \rangle : \|x\| \leq 1, \quad x \in \mathbb{R}^n \} \quad (\text{B4a})$$

$$= \max \{ \langle s, x \rangle : x \in B \}, \quad \forall s \in \mathbb{R}^n \quad (\text{B4b})$$

où

$$B \triangleq \{x \in \mathbb{R}^n : \|x\| \leq 1\} \quad (\text{B5})$$

est une norme dans $\mathbb{R}^n$.

De plus,

$$\|x\| = \max \{\langle s, x \rangle : \|s\|^* \leq 1, s \in \mathbb{R}^n\} \quad (\text{B6a})$$

$$= \max \{\langle s, x \rangle : s \in B^*\} \quad (\text{B6b})$$

où

$$B^* \triangleq \{s \in \mathbb{R}^n : \|s\|^* \leq 1\}. \quad (\text{B7})$$

Définition. La fonction $\|\cdot\|^* : \mathbb{R}^n \rightarrow \mathbb{R}_+$ définie par (B4a)-(B4b) est dite *norme duale* associée à la norme $\|\cdot\|$.

Pour toute norme $\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}_+$, on a une inégalité semblable à celle de Cauchy:

$$\langle s, x \rangle \leq |\langle s, x \rangle| \leq \|s\|^* \|x\|, \quad \forall (s, x) \in \mathbb{R}^n \times \mathbb{R}^n. \quad (\text{B8})$$

Cette relation est une égalité pour tout $(s, x) \in \mathbb{R}^n \times \mathbb{R}^n$, $s \neq 0$ et $x \neq 0$, tel que

$$\frac{s}{\|s\|^*} \in \{t \in B^* : \langle t, x \rangle = \|x\|\} \quad (\text{B9a})$$

ou que

$$\frac{x}{\|x\|} \in \{t \in B : \langle s, t \rangle = \|s\|^*\}. \quad (\text{B9b})$$

Les deux résultats suivants se démontrent aisément.

Proposition B.22 *Si le produit scalaire utilisé est le produit scalaire euclidien, alors les normes ℓ_1 et ℓ_∞ sont duales l'une par rapport à l'autre.*

Proposition B.23 *Si le produit scalaire utilisé est le produit scalaire euclidien, alors la norme duale associée à la norme ℓ_p , $p > 1$, est la norme ℓ_q , où*

$$q = \frac{p}{p-1}.$$

Proposition B.24 *(Hiriart-Urruty et Lemaréchal, 1993a: exemple V.3.2.3, pages 222–223) Si Q est une matrice symétrique définie positive, alors*

$$(\|\cdot\|_Q)^* = \|\cdot\|_{Q^{-1}}.$$

En résumé, on peut toujours, à partir d'une norme, définir une autre norme: sa norme duale. De plus, le dual d'une norme duale est la norme elle-même.

Corollaire B.25 *La norme euclidienne est auto-duale, i.e.*

$$(\|\cdot\|_2)^* = \|\cdot\|_2.$$

Le prochain résultat nous permet de caractériser le sous-différentiel de la fonction norme.

Proposition B.26 (*Hiriart-Urruty et Lemaréchal, 1993a: exemple VI.3.1, pages 258-259*) Si $\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}_+$ est une norme quelconque, alors

$$\partial\|\cdot\|(x) = \{s \in B^* : \langle s, x \rangle = \|x\|\}, \quad \forall x \in \mathbb{R}^n \quad (\text{B10})$$

où B^* est l'ensemble défini par (B7).

Annexe C Démonstration du lemme 5.9.8

Nous présentons ici la démonstration du lemme 5.9.8 qui avait été omise dans le but de faciliter la lecture de la dernière partie du chapitre 5.

Lemme 5.9.8 *Les solutions optimales du problème (5.58) sont aussi des solutions optimales du problème (5.22a)-(5.22f).*

Démonstration.

Soit (π^k, μ^k, ω^k) la solution optimale du problème (5.34) ayant servi à obtenir les quantités $(\tilde{\mu}^k, \tilde{\omega}^k)$ par les expressions (5.53a)-(5.53f). Le vecteur $(\pi^k, \tilde{\mu}^k, \tilde{\omega}^k)$ a servi à construire les coupes agrégées du problème (5.58).

Soit également une solution $(\hat{d}^k, \hat{v}^k, \hat{\sigma}_f^k, \hat{\sigma}_D^k)$ du problème (5.58). Nous montrerons que cette solution est optimale pour le problème (5.22a)-(5.22f) en procédant comme suit: nous prendrons d'abord une solution optimale quelconque $(\hat{\pi}^k, \hat{\mu}^k, \hat{\omega}^k)$ du dual du problème (5.58). À l'aide des solutions $(\hat{\pi}^k, \hat{\mu}^k, \hat{\omega}^k)$ et $(\pi^k, \tilde{\mu}^k, \tilde{\omega}^k)$ nous construirons une solution optimale $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k)$ pour le problème dual complet (5.34). Celle-ci détermine une solution optimale $(\bar{d}^k, \bar{v}^k, \bar{\sigma}_f^k, \bar{\sigma}_D^k)$ du problème (5.22a)-(5.22f). Nous montrerons enfin que

$$(\bar{d}^k, \bar{v}^k, \bar{\sigma}_f^k, \bar{\sigma}_D^k) = (\hat{d}^k, \hat{v}^k, \hat{\sigma}_f^k, \hat{\sigma}_D^k),$$

ce qui prouvera le lemme.

Tout d'abord, remarquons que

$$SRC \subseteq SRA$$

où SRC et SRA sont les ensembles des solutions réalisables des problèmes complet (5.22a)-(5.22f) et agrégé (5.58). En effet, les contraintes du problème (5.58) sont ou bien des contraintes du problème (5.22a)-(5.22f), ou bien des combinaisons convexes de contraintes de ce problème.

Les fonctions économiques des deux problèmes étant identiques, il s'ensuit que la valeur optimale de l'objectif du problème (5.22a)-(5.22f) ne peut être inférieure à celle du problème (5.58).

Les problèmes (5.22a)-(5.22f) et (5.58) étant de même structure, on peut alors appliquer le lemme 5.6.4 au problème agrégé. Il existe donc, par la partie (ii) du lemme 5.6.4, des multiplicateurs de Lagrange $(\hat{\pi}^k, \hat{\mu}^k, \hat{\omega}^k)$, qui, avec $(\hat{d}^k, \hat{v}^k, \hat{\sigma}_f^k, \hat{\sigma}_D^k)$, vérifient les conditions (5.40a)-(5.40s). Ces multiplicateurs sont, par la dernière partie du lemme 5.6.4, une solution optimale du problème dual associé au problème (5.58).

Nous construirons, à l'aide des multiplicateurs $(\hat{\pi}^k, \hat{\mu}^k, \hat{\omega}^k)$ et des multiplicateurs $(\pi^k, \tilde{\mu}^k, \tilde{\omega}^k)$ ayant servi à construire les coupes agrégées, des multiplicateurs $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k)$ réalisables pour le problème dual complet (5.34).

Pour $i \in \{1, \dots, r\}$, définissons

$$\bar{\pi}_{ij}^k = \begin{cases} \hat{\pi}_{pi}^k \pi_{ij}^k + \hat{\pi}_{ij}^k & \text{si } j \in \tilde{J}_{fi}^k \\ \hat{\pi}_{pi}^k \pi_{ij}^k & \text{si } j \in J_{fi}^k \setminus \tilde{J}_{fi}^k \end{cases} \quad (C1)$$

$$\bar{\mu}_{ij}^k = \begin{cases} \hat{\mu}_{pi}^k \tilde{\mu}_{ij}^k + \hat{\mu}_{ij}^k & \text{si } j \in \tilde{J}_{\alpha}^k \\ \hat{\mu}_{pi}^k \tilde{\mu}_{ij}^k & \text{si } j \in J_{\alpha}^k \setminus \tilde{J}_{\alpha}^k. \end{cases} \quad (C2)$$

Définissons aussi

$$\bar{\omega}_j^k = \begin{cases} \hat{\omega}_p^k \tilde{\omega}_j^k + \hat{\omega}_j^k & \text{si } j \in \tilde{J}_c^k \\ \hat{\omega}_p^k \tilde{\omega}_j^k & \text{si } j \in J_c^k \setminus \tilde{J}_c^k. \end{cases} \quad (C3)$$

On a alors $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k) \geq 0$. De plus, en utilisant les définitions (C1) à (C3), puis les expressions (5.52a), (5.51a), (5.53e) et (5.53f), puis enfin les expressions (5.40b) et (5.40c), on trouve que

$$\sum_{j \in J_{f_i}^k} \bar{\pi}_{ij}^k = 1, \quad \forall i \in \{1, \dots, r\}$$

et

$$\sum_{i=1}^r \sum_{j \in J_{\alpha}^k} \bar{\mu}_{ij}^k + \sum_{j \in J_c^k} \bar{\omega}_j^k \leq 1.$$

Ceci montre que $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k)$ est réalisable pour le problème (5.34) complet.

On remarque aussi, en utilisant les définitions (C1) à (C3) et (5.54a)-(5.54c), que

$$\begin{aligned} \bar{p}^k &\triangleq \sum_{i=1}^r \sum_{j \in J_{f_i}^k} \bar{\pi}_{ij}^k g_i^j + \sum_{i=1}^r \sum_{j \in J_{\alpha}^k} \bar{\mu}_{ij}^k a_i^j + \sum_{j \in J_c^k} \bar{\omega}_j^k a^j \\ &= \hat{p}^k \end{aligned} \quad (C4)$$

où $\hat{p}^k$ est le vecteur défini par l'expression (5.40k) associée à la solution optimale $(\hat{d}^k, \hat{v}^k, \hat{\sigma}_f^k, \hat{\sigma}_D^k)$ du problème agrégé (5.58).

De plus, on trouve en utilisant les expressions (C1) à (C4) que la valeur de la fonction économique du problème dual complet (5.34) en $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k)$ est égale à la

valeur optimale de la fonction économique du problème dual agrégé en $(\hat{\pi}^k, \hat{\mu}^k, \hat{\omega}^k)$. Ceci démontre l'optimalité de $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k)$ pour le problème (5.34) complet.

Cette solution optimale permet, par l'expression (5.40m) du lemme 5.6.4, d'obtenir une solution optimale $\bar{d}^k$ du problème (5.21). On obtient une solution optimale du problème (5.22a)-(5.22f) en déterminant $\bar{\sigma}_f^k$ et $\bar{\sigma}_D^k$ à l'aide des expressions (5.40n) et (5.40o), puis en déterminant $\bar{v}^k$ comme suit:

$$\bar{v}_i^k = \max \{ f_{ij}^k + \langle g_i^j, \bar{d}^k \rangle : j \in J_{fi}^k \}, \quad i \in \{1, \dots, r\}. \quad (C5)$$

Puisque

$$\bar{d}^k \in \text{zéro} \{ \Psi_k h'(\|d\|) \partial \| \cdot \| (d) + \bar{p}^k \} \quad \text{par (5.40m) et (C4)}$$

et que $\hat{d}^k$ est déterminé de la même manière, on peut poser

$$\bar{d}^k = \hat{d}^k. \quad (C6)$$

Les expressions (C5) et (C6) montrent que $\bar{v}^k = \hat{v}^k$.

Les scalaires $\bar{\sigma}_f^k$ et $\bar{\sigma}_D^k$ ayant été obtenus à l'aide des expressions (5.40m) et (5.40o), on trouve en utilisant les définitions (C1) à (C4), les définitions (5.54a)-(5.54c) et les égalités (5.52a) que

$$\bar{\sigma}_f^k = \hat{\sigma}_f^k \quad \text{et} \quad \bar{\sigma}_D^k = \hat{\sigma}_D^k. \quad (C7)$$

En résumé, nous avons montré que $(\bar{d}^k, \bar{v}^k, \bar{\sigma}_f^k, \bar{\sigma}_D^k) = (\hat{d}^k, \hat{v}^k, \hat{\sigma}_f^k, \hat{\sigma}_D^k)$ est une solution réalisable et optimale du problème complet (5.22a)-(5.22f). ■

Remarque: La construction de la solution $(\bar{\pi}^k, \bar{\mu}^k, \bar{\omega}^k)$ par les expressions (C1) à (C3) utilise un procédé que Zipkin (1980) appelle *désagrégation par les poids* ("fixed-weight disaggregation").

Tableau D.1: Comportement de l'algorithme appliqué au problème DQCNY3 pour des rayons constants sous la stratégie d'accumulation

Norme ℓ_∞					Norme ℓ_1				
γ	Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/itération	γ	Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/itération
0,025	777	322	97,43	0,903	0,25	1132	203	95,45	0,939
0,05	433	273	59,96	0,810	0,5	620	212	59,60	0,859
0,1	265	242	39,54	0,686	1,0	364	209	38,34	0,744
0,25	151	199	24,65	0,493	1,5	266	192	28,92	0,679
0,5	115	178	18,63	0,368	2,0	227	203	26,91	0,597
0,75	87	180	15,38	0,326	2,5	208	199	25,59	0,541
1,0	100	208	18,47	0,263	4,0	150	184	20,15	0,470
1,5	114	240	22,03	0,168	5,0	129	184	19,27	0,469
2,0	166	388	36,40	0,133	5,5	116	177	18,38	0,487
2,5	184	436	45,00	0,148	6,75	116	193	19,59	0,417
3,0	209	504	54,04	0,115	7,5	124	202	21,09	0,333
4,0	236	580	63,18	0,106	10,0	122	218	20,93	0,281
5,0	251	625	71,59	0,076	15,0	145	249	26,63	0,201
7,5	293	730	92,01	0,082	25,0	148	302	29,85	0,197
10,0	291	739	98,18	0,069	50,0	198	412	44,11	0,112
25,0	355	926	151,39	0,079	100,0	233	576	63,07	0,103
50,0	409	1045	196,91	0,064	500,0	320	789	107,76	0,072
250,0	391	1000	202,52	0,064	2000,0	302	746	107,21	0,073

Tableau D.2: Performances de l'algorithme avec ajustement dynamique de γ
sous la stratégie d'accumulation

Problème	Norme ϵ_∞					Norme ϵ_1				
	Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/ itération	Temps PM/ Temps SP	Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/ itération	Temps PM/ Temps SP
DQCN1	31	41	1,24	0,367	0,5	34	46	1,42	0,424	0,474
DQCN1-QC	17	24	0,58	0,625	0,183	14	22	0,48	0,462	0,105
DQCN1-NY	18	19	0,64	0,706	0,130	17	17	0,55	0,438	0,175
DQCN2	73	135	7,07	0,431	0,450	67	100	5,62	0,470	0,503
DQCN2-QC	41	72	3,22	0,4	0,130	34	74	2,66	0,394	0,106
DQCN2-NY	84	83	6,17	0,361	0,177	65	69	4,78	0,344	0,215
DQCN3	94	160	15,17	0,366	0,629	133	212	21,90	0,455	0,930
DQCN3-QC	59	107	8,62	0,328	0,139	51	84	6,16	0,440	0,089
DQCN3-NY	102	118	13,62	0,248	0,170	105	119	13,27	0,337	0,217
DQCN4	439	653	217,31	0,158	3,380	341	424	127,44	0,429	2,657
DQCN4-QC	168	249	44,24	0,329	0,233	190	254	43,47	0,392	0,223
DQCN4-NY	164	185	38,22	0,362	0,178	299	338	70,94	0,322	0,397
DQCN5	550	787	492,12	0,304	4,181	571	679	543,92	0,423	7,131
DQCN5-QC	280	373	109,35	0,427	0,274	262	348	95,32	0,467	0,215
DQCN5-NY	760	785	321,13	0,252	0,866	702	686	260,29	0,354	0,688

($\theta = 1,33$, facteur d'inertie = 4, $\gamma_0 = 0,9$, $\gamma_{INF} = 0,1$ et $\gamma_{SUP} = 5$ pour la norme ϵ_∞ ; $\gamma_0 = 6,75$, $\gamma_{INF} = 1$ et $\gamma_{SUP} = 50$ pour la norme ϵ_1)

Tableau D.3: Performances de l'algorithme avec ajustement dynamique de γ
sous la stratégie d'accumulation/épuration

Problème	Norme ℓ_∞					Norme ℓ_1						
	Itérations	Colonnes engendrées	Colonnes détruites	Temps UCT (secondes)	Pas de descente/itération	Temps PM/ Temps SP	Itérations	Colonnes engendrées	Colonnes détruites	Temps UCT (secondes)	Pas de descente/itération	Temps PM/ Temps SP
DQCN1	40	61	23	1,63	0,462	0,343	34	44	18	1,34	0,394	0,440
DQCN1_QC	23	34	14	0,79	0,546	0,197	14	22	9	0,49	0,462	0,150
DQCN2	112	185	147	9,62	0,360	0,385	97	127	100	7,41	0,479	0,423
DQCN2_QC	40	73	49	3,10	0,410	0,083	41	74	62	3,14	0,375	0,106
DQCN3	251	336	270	34,09	0,280	0,739	271	317	271	35,37	0,382	1,000
DQCN3_QC	139	209	176	16,55	0,167	0,122	65	107	83	7,80	0,359	0,086
DQCN4	2963	3425	3344	711,22	0,0385	2,435	848	992	934	228,79	0,231	2,347
DQCN4_QC	230	326	273	50,28	0,218	0,113	311	393	363	61,99	0,294	0,144
DQCN5	6712	7150	7039	2950,25	0,050	5,256	1987	2264	2160	957,62	0,185	3,827
DQCN5_QC	425	518	466	137,41	0,323	0,135	509	619	582	162,82	0,305	0,141

($\theta = 1,33$, facteur d'inertie = 4, $\gamma_0 = 0,9$, $\gamma_{INF} = 0,1$ et $\gamma_{SUP} = 5$ pour la norme ℓ_∞ ; $\gamma_0 = 6,75$, $\gamma_{INF} = 1$ et $\gamma_{SUP} = 50$ pour la norme ℓ_1)

Tableau D.4: Comportement asymptotique de l'algorithme appliqué au problème DQCNY5_QC

Stratégie	Précision	Norme ℓ_∞		Norme ℓ_1	
		Itérations	Temps UCT (secondes)	Itérations	Temps UCT (secondes)
A	0,025	280	109,35	262	95,32
	0,1%	251	102,22	241	89,43
	0,5%	146	65,48	144	56,31
	1,0%	91	46,88	102	38,77
	2,5%	85	45,08	68	26,43
E	0,025	425	137,41	509	162,82
	0,1%	396	130,10	488	157,45
	0,5%	91	42,70	197	64,83
	1,0%	91	42,70	102	36,23
	2,5%	87	41,46	66	24,89

Tableau D.5: Performances de CMinSfpc en présence d'enrichissement

Problème	Enrichissement	Norme ℓ_∞				Norme ℓ_1			
		Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/itération	Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/itération
DQCNY1-QC	I	17	24	0,58	0,625	14	22	0,48	0,462
	S	17	33	0,68	0,563	15	38	0,72	0,571
	M	17	358	4,55	0,625	14	186	3,30	0,615
DQCNY2-QC	I	41	72	3,22	0,400	34	64	2,66	0,394
	S	38	106	4,42	0,405	33	90	3,81	0,406
	M	27	1612	50,90	0,577	28	932	34,05	0,444
DQCNY3-QC	I	59	107	8,62	0,328	51	84	6,16	0,440
	S	49	139	11,21	0,354	41	110	8,53	0,500
	M	40	3593	188,48	0,410	38	1596	80,83	0,541

Tableau D.6: Comparaison de la méthode des faisceaux avec γ constant à la méthode de Benders

Problème	Méthode	γ	Itérations	Colonnes engendrées	Temps UCT (secondes)	Itérations du simplexe
DQCNY1	FP	0,9	31	41	1,24	533 (203)
	FC	0,9	76 (30)	97	3,19	697 (297)
	B	10000	85	184	6,16	2135 (332)
DQCNY2	FP	0,9	69	116	6,39	1918 (760)
	FC	0,9	185 (46)	252	17,63	3037 (1728)
	B	10000	226	489	49,73	11862 (2565)
DQCNY3	FP	0,9	86	186	15,51	3691 (1557)
	FC	0,9	242 (84)	395	43,98	6304 (3624)
	B	10000	379	1040	800,89	137706 (101057)
	B	35	350	856	162,72	24854 (8578)

Tableau D.7: Performances de l'algorithme avec plans coupants et ajustement dynamique de γ sous la stratégie d'accumulation

Problème	Norme ℓ_∞					Norme ℓ_1				
	Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/ itération	Temps PM/ Temps SP	Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/ itération	Temps PM/ Temps SP
DQCNV1	80 (25)	94	3,30	0,333	0,530	77 (24)	91	3,22	0,293	0,844
DQCNV1_QC	37 (20)	48	1,42	0,371	0,172	37 (12)	58	1,67	0,229	0,195
DQCNV2	250 (60)	246	20,06	0,419	1,027	223 (58)	232	20,42	0,358	1,562
DQCNV2_QC	72 (35)	112	5,75	0,386	0,095	60 (25)	100	5,16	0,241	0,132
DQCNV3 ⁽¹⁾	295 (87)	378	46,73	0,352	1,167	280 (58)	366	63,32	0,266	2,417
DQCNV3_QC	165 (42)	211	19,78	0,344	0,130	91 (24)	132	12,44	0,371	0,096
DQCNV4_QC	212 (58)	274	48,92	0,438	0,102	181 (50)	258	41,85	0,464	0,123
DQCNV5_QC	349 (102)	491	124,31	0,438	0,143	273 (86)	358	92,82	0,509	0,146

($\theta = 1,33$, facteur d'inertie = 4, $\gamma_0 = 0,9$, $\gamma_{INF} = 0,1$ et $\gamma_{SUP} = 5$ pour la norme ℓ_∞ , $\gamma_0 = 6,75$, $\gamma_{INF} = 1$ et $\gamma_{SUP} = 50$ pour la norme ℓ_1)

Note: ⁽¹⁾ $\gamma_{INF} = 2,5$ pour la norme ℓ_1

Tableau D.8: Performances de l'algorithme avec plans coupants et ajustement dynamique de γ sous la stratégie d'accumulation/épipuration

Problème	Norme ℓ_∞						Norme ℓ_1					
	Itérations	Colonnes engendrées	Colonnes détruites	Temps UCT (secondes)	Pas de descente/itération	Temps PM/ Temps SP	Itérations	Colonnes engendrées	Colonnes détruites	Temps UCT (secondes)	Pas de descente/itération	Temps PM/ Temps SP
DQCNV1	116 (32)	125	54	4,53	0,237	0,563	136 (25)	151	81	5,54	0,187	0,803
DQCNV1-QC	48 (21)	59	15	1,78	0,283	0,198	53 (12)	73	32	2,13	0,157	0,218
DQCNV2	833 (62)	887	719	62,22	0,124	1,240	376 (70)	405	257	29,77	0,222	1,223
DQCNV2-QC	177 (36)	208	123	12,01	0,2	0,151	85 (24)	133	67	6,91	0,169	0,143
DQCNV3 (1)	1245 (83)	1409	1183	155,16	0,106	1,491	583 (64)	745	562	84,27	0,110	1,266
DQCNV3-QC	367 (55)	417	276	39,33	0,225	0,141	154 (24)	196	113	18,56	0,303	0,133
DQCNV4-QC	444 (71)	567	374	93,27	0,231	0,115	301 (48)	351	238	62,99	0,368	0,114
DQCNV5-QC	1005 (44)	1250	863	325,70	0,246	0,128	469 (91)	567	350	141,09	0,450	0,128

($\theta = 1,33$, facteur d'inertie = 4, $\gamma_0 = 0,9$, $\gamma_{INF} = 0,1$ et $\gamma_{SUP} = 5$ pour la norme ℓ_∞ ; $\gamma_0 = 6,75$, $\gamma_{INF} = 1$ et $\gamma_{SUP} = 50$ pour la norme ℓ_1)

Note: (1) $\gamma_{INF} = 4$ pour la norme ℓ_1

Tableau D.9: Comportement asymptotique de l'algorithme appliqué au problème DQCNY5_QC en utilisant la norme ℓ_1

Stratégie	Précision	Itérations	Temps UCT (secondes)
A	0,025	273	92,82
	0,1 %	253	85,61
	0,5 %	197	67,17
	1,0 %	155	55,39
	2,5 %	107	42,57
	2,72%	86	23,39
E	0,025	469	141,09
	0,1 %	432	128,95
	0,5 %	291	90,77
	1,0 %	197	62,16
	2,5 %	122	42,37
	2,72%	91	24,47

($\theta = 1.33$, facteur d'inertie = 4; $\gamma_0 = 6.75$, $\gamma_{INF} = 1$, $\gamma_{SUP} = 50$,

$\gamma = \gamma_0$ au début des deux phases)

Tableau D.10: Performances de l'algorithme avec ajustement dynamique de γ pour les plus gros problèmes

Problème	Approche	Stratégie	Itérations	Colonnes engendrées	Temps UCT (secondes)	Pas de descente/ itération	Temps PM/ Temps SP
DQCNY6_QC'	C	A	400 (8)	600	247,53	0,495	0,166
	C	E	583 (8)	1166 (1101)	342,92	0,441	0,065
	P	A	548	730	240,58	0,431	0,342
DQCNY7_QC'	C	A	484 (8)	807	487,35	0,5	0,258
	C	E	759 (8)	1518 (1448)	643,67	0,448	0,057
	P	A	1093	1551	835,15	0,404	0,808
DQCNY8_QC'	C	A	671 (8)	1166	1035,06	0,517	0,358
	C	E	996 (8)	1992 (1915)	1104,31	0,464	0,067
	P	A	1455	2114	1588,66	0,378	1,070
DQCNY9_QC'	C	A	699 (8)	1262	1346,08	0,458	0,265
	C	E	1452 (8)	2904 (2811)	1970,29	0,445	0,077
	P	A	1858	2646	2926,22	0,369	1,570

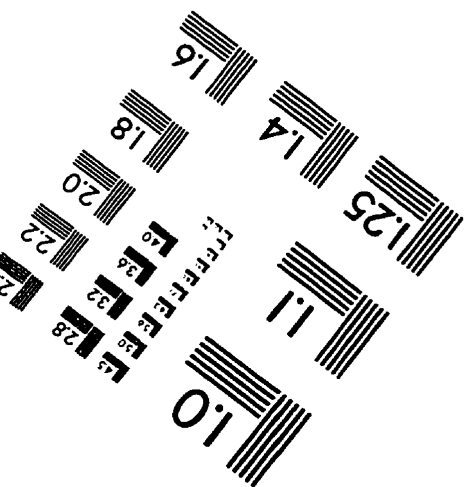
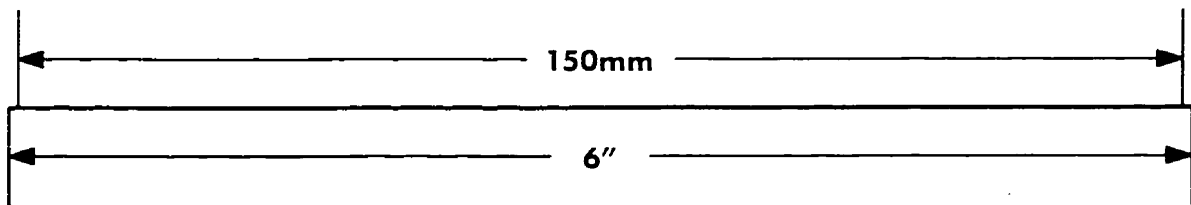
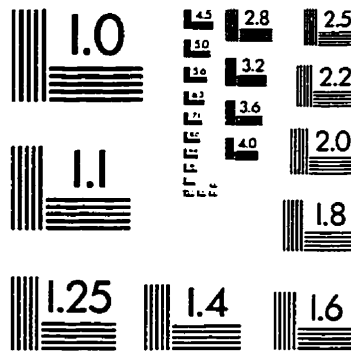
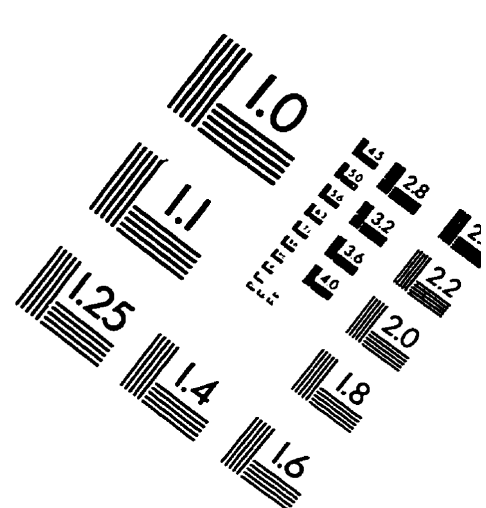
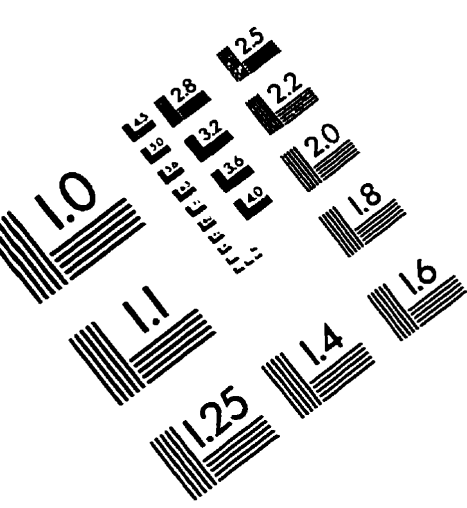
(Norme ℓ_1 , $\theta = 1,33$, facteur d'inertie = 4; $\gamma_0 = 6,75$, $\gamma_{INF} = 1$, $\gamma_{SUP} = 50$, $\gamma = \gamma_0$ au début des deux phases)

Tableau D.11: Comportement asymptotique de l'algorithme appliqué au problème DQCNY9-QC' en utilisant la norme ℓ_1

Stratégie	Précision	Itérations	Temps UCT (secondes)
A	0,025	699	1346,08
	0,1 %	502	947,25
	0,5 %	269	435,32
	1,0 %	158	257,21
	2,5 %	46	92,43
	6,42%	8	3,66
E	2,0	1452	1970,29
	0,1 %	997	1391,50
	0,5 %	434	598,42
	1,0 %	230	323,55
	2,5 %	88	139,00
	6,42%	8	3,62

($\theta = 1,33$, facteur d'inertie = 4; $\gamma_0 = 6,75$, $\gamma_{INF} = 1$, $\gamma_{SUP} = 50$,
 $\gamma = \gamma_0$ au début des deux phases)

IMAGE EVALUATION TEST TARGET (QA-3)



APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989

© 1993, Applied Image, Inc., All Rights Reserved

