POLYPUBLIE
Polytechnique Montréal

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

Titre: Title:	Théorie de l'optimisation sans dérivées dans le cas discontinu
Auteur: Author:	Pierre-Yves Bouchet
Date:	2023
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Bouchet, P.-Y. (2023). Théorie de l'optimisation sans dérivées dans le cas discontinu [Thèse de doctorat, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/56997/

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/56997/
Directeurs de recherche: Advisors:	Charles Audet, & Loïc Bourdin
Programme: Program:	Doctorat en mathématiques

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal

https://publications.polymtl.ca

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Théorie de l'optimisation sans dérivées dans le cas discontinu

PIERRE-YVES BOUCHET
Département de mathématiques et de génie industriel

Thèse présentée en vue de l'obtention du diplôme de *Philosophiæ Doctor*
Mathématiques

Décembre 2023

© Pierre-Yves Bouchet, 2023.

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Cette thèse intitulée :

Théorie de l'optimisation sans dérivées dans le cas discontinu

présentée par **Pierre-Yves BOUCHET**
en vue de l'obtention du diplôme de *Philosophiæ Doctor*
a été dûment acceptée par le jury d'examen constitué de :

Sébastien LE DIGABEL, président
Charles AUDET, membre et directeur de recherche
Loïc BOURDIN, membre et codirecteur de recherche
Youssef DIOUANE, membre
Clément ROYER, membre externe

DÉDICACE

À Alain Damasio.
À sa vision du lien au réel
érigée comme art de vivre.
« Chercher ce point ultime de haute vitalité
où la membrane qui nous sépare et nous protège
des variations incessantes du dehors
atteint l'extrême finesse qui précède sa déchirure. »

REMERCIEMENTS

Aux membres du jury appréciant ma thèse, via le temps qu'ils accordent à ma recherche ; aux institutions de Polytechnique Montréal et du GERAD, de par l'environnement de travail qu'elles m'ont fourni ;
je suis reconnaissant de leur implication, rendant possible la reconnaissance de ce manuscrit.

À Charles Audet, mon directeur de thèse, en réponse à la qualité du mentorat qu'il m'a offert, à la confiance qu'il m'a accordée pour me laisser définir des objectifs de recherche ambitieux, ainsi qu'à l'immense soutien mathématique, financier et humain qu'il m'a apporté ;
à Loïc Bourdin, mon co-directeur de thèse, pour son implication sans failles dans ce projet, l'ampleur de ses apports et la richesse des échanges que nous avons eus ;
j'adresse mes sincères remerciements. Ce fut un honneur de travailler à leurs côtés.

À Solène, Ludovic, Romain, Jean, Waïl, Théo, à la chaleur qu'ils apportaient au labo, à leur capacité à transformer une pause café en source intarissable d'idées ;
à Capucine, Yahya, Youness, Laurent, Gaël, Aude, Marion, Alex, Alexis, mes solides points d'ancrage au Québec depuis mon arrivée ;
à Ilaria, Raphaël, Ève, Baptiste, Guillaume, Thibaut, Amélie, pour le lien fort et les superbes moments issus du même plaisir de tracter la vie au bout des doigts, et pour m'avoir suivi dans des pérégrinations parfois assez folles ;
à Marie-Ange, Sophie, Mathieu, Yann, Gaëlle, Romain, Charles, Paul, Anissa, Manon, Arthur, à l'amitié que je leur porte et qui survit à la distance ;
à Alexandre, à Philippe, pour tout ce que nous partageons depuis tellement d'années ;
à ma famille, toujours proche, même de loin ;
à tous ceux que j'oublie de remercier ici ;
merci d'avoir été présents.

À ces deux merveilleuses maximes qui ont porté la rédaction de cette thèse ; mes hommages.

> *Lorsque j'échouais à mettre en train un nouveau récit, je pensais : « Ne t'en fais pas. Ce qu'il faut, c'est écrire une seule phrase vraie. Écris la phrase la plus vraie que tu connaisses. » Alors j'écrivais une phrase vraie, et continuais à partir de là.*
> *J'avais déjà appris à toujours m'arrêter avant d'assécher le puits de mon inspiration, pour laisser la source remplir le réservoir pendant la nuit.*
>
> « Paris est une fête », Ernest Hemingway.

RÉSUMÉ

Cette thèse vise à développer des algorithmes d'optimisation aptes à minimiser une fonction objectif présentant des discontinuités. Ce manuscrit propose d'abord une introduction aux concepts fondamentaux de l'optimisation discontinue, accompagné d'une revue de littérature, puis reprend le contenu de nos travaux de recherche sur le comportement de la *méthode de recherche directe* (« Direct Search Method », `DSM`) dans ce contexte.

Notre première contribution est de rouvrir une discussion sur un travail pionnier dans l'étude de `DSM` dans un contexte discontinu, « Analysis of direct searches for discontinuous functions », Mathematical Programming Volume 133, pp. 299–325, 2012. Cette référence a identifié une condition technique sur le comportement de `DSM` qui permet d'étendre les résultats connus dans le cas continu vers le cas discontinu. Malheureusement, le théorème énoncé par les auteurs pour s'assurer que cette condition technique est vérifiée comporte une erreur. Nous mettons en évidence ce problème via un contre-exemple, puis adaptons `DSM` afin de le surmonter.

Nous poursuivons avec une étude plus poussée des propriétés théoriques de `DSM` dans des cadres présentant des discontinuités potentielles. Pour ce faire, nous identifions une nouvelle adaptation de `DSM` permettant de garantir qu'elle génère une solution locale au problème d'optimisation. Nous prouvons également que ce résultat est plus fort que ceux énoncés dans d'autres travaux de la littérature, et obtenu sous des hypothèses de travail plus faibles.

Enfin, nous introduisons le cadre de l'*optimisation partitionnée*, pour permettre à `DSM` de traiter efficacement des problèmes de grande dimension dans lesquels les discontinuités de la fonction objectif ne dépendent que de quelques-unes des variables. Nous optimisons ces variables par `DSM`, et en parallèle nous traitons toutes les autres avec un algorithme d'optimisation classique. Nous commençons par développer le cadre théorique, et nous prouvons que cette stratégie identifie une solution locale du problème. Ensuite, nous proposons plusieurs tests numériques pour confirmer qu'il est plus efficace de suivre cette stratégie que d'optimiser toutes les variables par une méthode d'optimisation sans dérivées.

En complément, nous terminons par une illustration de l'optimisation partitionnée appliquée à une classe de problèmes issus du contrôle optimal, où seuls quelques états affectent un coût de Mayer discontinu et où toutes les autres variables (d'état et de contrôle) affectent un coût de Lagrange lisse.

ABSTRACT

This thesis lies in the field of derivative-free optimization, and aims to extend the usual methods in the literature to the cases where the objective function is possibly discontinuous. This manuscript first introduces the fundamental concepts involved in discontinuous optimization, accompanied by a literature review. Then, it summarizes the content of my research papers about the behaviour of the *Direct Search Method* (`DSM`) in this context.

The first part of this thesis discusses a former work that contributed to shape discontinuous optimization, "Analysis of direct search methods for discontinuous functions", Mathematical Programming Volume 133, pp. 299–325, 2012. The authors identified a technical condition about `DSM` which allows to relax the continuity assumption in most results from the literature, so they remain valid in the discontinuous case. Nevertheless, this technical condition may be difficult to satisfy and the related theorem provided by the authors is incorrect. The thesis highlights this error via a counterexample, and corrects it by adapting `DSM`.

The second part of the manuscript strengthens the theoretical properties for `DSM`. It is proven that under an adaptation explicitly constructed, `DSM` generates a local solution to the optimization problem, and this result holds true even in discontinuous cases. It is also proven that the assumptions this adaptation relies on are weaker and more generic than in former work from the literature.

Finally, the manuscript introduces the framework of *partitioned optimization* to allow `DSM` to solve large scale problems where the discontinuities in the objective function depend only on a subset of the variables. The idea is that in such cases, it is more efficient to address these variables via `DSM` and to leave all the others to an usual algorithm relying on derivatives. This hybrid strategy still identifies a local solution to the optimization problem. It is also proven that the required structure on the optimization problem is common, as it is indeed met in several classes of problems. Last, some numerical examples are provided, highlighting that the strategy under discussion is indeed more efficient than regular methods from the literature.

In addition, the manuscript proposes an illustration of the application of partitioned optimization to a class of optimal control problems where only the Mayer cost (depending on the state of the sytem at a few times) has discontinuities, while the Lagrange cost (depending on all others variables) is smooth.

TABLE DES MATIÈRES

DÉDICACE iv

REMERCIEMENTS v

RÉSUMÉ vi

ABSTRACT vii

TABLE DES MATIÈRES viii

LISTE DES TABLEAUX xi

LISTE DES FIGURES xii

PLAN DU MANUSCRIT ET DÉPENDANCE DES CHAPITRES xv

CHAPITRE 1 INTRODUCTION 1
- 1.1 Bref survol de l'historique de l'optimisation 1
  - 1.1.1 Optimisation calculatoire : des premiers travaux à Newton 1
  - 1.1.2 Optimisation algorithmique : l'essor de l'informatique 2
  - 1.1.3 Cas particulier de l'optimisation sans dérivées 3
- 1.2 Définitions et concepts fondamentaux en optimisation discontinue 5
  - 1.2.1 Notions liées à la continuité et discontinuité d'une fonction 5
  - 1.2.2 Notions d'analyse par différentiabilité généralisée 6
  - 1.2.3 Problème sans dérivées, conditions d'optimalité envisageables 8
- 1.3 Objectifs de recherche 9

CHAPITRE 2 ÉTAT DE L'ART 11
- 2.1 Présentation succincte de l'optimisation sans dérivées 11
  - 2.1.1 Problèmes à ne pas résoudre par des méthodes sans dérivées 11
  - 2.1.2 Différentes dénominations dans différentes littératures 13
  - 2.1.3 Notions d'analyse d'algorithmes sans dérivées 14
- 2.2 Invalidité des algorithmes sans dérivées dans des cas discontinus 15
  - 2.2.1 Méthodes heuristiques 15
  - 2.2.2 Méthodes fondées sur les modèles 18
  - 2.2.3 Méthodes de recherche directe (`DSM`) 23

CHAPITRE 3 CORRECTIF À UN RÉSULTAT DE LA LITTÉRATURE 28
3.1 Synthèse de « Analysis of direct searches for discontinuous functions » 28
3.1.1 Établissement de conditions nécessaires d'optimalité 29
3.1.2 Affirmation de la convergence de DSM vers un point vérifiant des conditions nécessaires d'optimalité 30
3.2 Erreur dans l'affirmation de la convergence de DSM 31
3.2.1 Construction d'un contre-exemple 32
3.2.2 Analyse de la preuve erronée 35
3.3 Modification de DSM corrigeant l'erreur 37
3.3.1 Modification de DSM par ajout d'une étape revealing : rDSM 37
3.3.2 rDSM garantit des conditions nécessaires d'optimalité 38
3.3.3 rDSM répond au contre-exemple 43

CHAPITRE 4 SIMPLIFICATION DU CADRE THÉORIQUE ET AMÉLIORATION DES RÉSULTATS DE CONVERGENCE DE rDSM 44
4.1 Complément à l'analyse de convergence de rDSM 44
4.1.1 Optimalité locale sous l'hypothèse d'unicité du point raffiné 45
4.1.2 Aspects restrictifs dans le cadre d'étude de rDSM 45
4.2 Extension de rDSM : étape de covering remplaçant l'étape revealing 48
4.2.1 Algorithme cDSM 48
4.2.2 Allègement du cadre théorique requis pour l'analyse de cDSM 52
4.2.3 Analyse de convergence de cDSM vers des optima locaux 54
4.2.4 Généraliser plus le cadre d'étude affaiblit la convergence 57

CHAPITRE 5 AMÉLIORATION DE cDSM SUR CERTAINS PROBLÈMES PAR OPTIMISATION PARTITIONNÉE 60
5.1 Reformulation d'un problème via l'optimisation partitionnée 61
5.1.1 Présentation formelle du cadre de l'optimisation partitionnée 61
5.1.2 Préservation des solutions entre la reformulation et le problème initial 63
5.1.3 Approches similaires dans la littérature 63
5.2 Application de cDSM au problème reformulé 65
5.2.1 Convergence de cDSM vers des solutions au problème initial 65
5.2.2 Pistes de généralisations possibles de l'analyse 68
5.3 Remarques sur l'optimisation partitionnée et extensions potentielles 69
5.3.1 Commentaires sur le cadre actuel 69
5.3.2 Définitions moins strictes de la fonction d'oracle 70
5.3.3 Méthodes hybrides 72

CHAPITRE 6 MISE EN APPLICATION DE L'OPTIMISATION PARTITIONNÉE 74
6.1 Optimisation partitionnée pour simplifier des problèmes bruités 74
6.1.1 Bruit mono-variable 75
6.1.2 Bruit radial 77
6.1.3 Bruit affecté par une combinaison non-linéaire de deux variables . . . 78
6.1.4 Bruit de dimension 2 dans un problème de dimension 3 80
6.2 Étude numérique du gain apporté à `DSM` par l'optimisation partitionnée . . . 83
6.2.1 Bruit mono-variable en dimension 101 84
6.2.2 Bruit radial en dimension 101 85
6.2.3 Bruit affecté par une combinaison non-linéaire de 101 variables 86
6.2.4 Bruit de dimension 10 dans un problème de dimension 100 87

CHAPITRE 7 APPLICATION AU CONTRÔLE OPTIMAL 89
7.1 Problème de contrôle optimal et discontinuités potentielles 89
7.1.1 Problème de contrôle optimal, coût de Mayer discontinu 90
7.1.2 État de l'art sur le contrôle optimal en présence de discontinuités . . 91
7.1.3 Classe simple de problèmes discontinus : problèmes à plateaux 92
7.2 Utilisation de l'optimisation partitionnée en contrôle optimal 94
7.2.1 Validité du cadre de l'optimisation partitionnée 94
7.2.2 Méthode numérique de résolution via l'optimisation partitionnée . . . 96
7.3 Mise en application sur des problèmes à plateaux 99
7.3.1 Solution analytique d'un problème de double intégrateur 99
7.3.2 Solution numérique d'un problème de Zermelo à plateaux circulaires . 102
7.3.3 Solution numérique d'un problème de Zermelo à plateaux verticaux . 104
7.4 Pistes d'améliorations des processus de résolution 106
7.4.1 Étude de sensibilité du sous-problème 106
7.4.2 Relaxation de la contrainte spatiale sur les états fixés 107

CHAPITRE 8 CONCLUSION 109
8.1 Synthèse des travaux 109
8.2 Limites théoriques et pratiques 110
8.3 Extensions possibles 111
8.4 Tableau récapitulatif des contributions et des extensions possibles 114

RÉFÉRENCES 115

LISTE DES TABLEAUX

Tableau 0.1 Notations non usuelles utilisées dans ce manuscrit xiv
Tableau 0.2 Contenu des chapitres et dépendances de lecture. xvi
Tableau 4.1 Intérêt de chaque étape de `cDSM`, et pistes d'amélioration pratique. . . 59
Tableau 6.1 Huit exécutions de `cDSM` au chapitre 6.1.1. Ici, $\hat{x}^k \triangleq x^k$. 76
Tableau 6.2 Huit exécutions de `cDSM` au chapitre 6.1.2. Ici, $\hat{x}^k \triangleq x^k - \sqrt{2}$. 77
Tableau 6.3 Huit exécutions de `cDSM` au chapitre 6.1.3. Ici, $\hat{x}^k \triangleq x^k - 4$. 79
Tableau 6.4 Huit exécutions de `cDSM` au chapitre 6.1.4. Ici, $\hat{x}^k \triangleq \|x^k\|_\infty$. 81
Tableau 8.1 Synthèse des travaux discutés et extensions possibles. 114

LISTE DES FIGURES

Figure 2.1 Schéma d'un algorithme génétique. 16
Figure 2.2 Détail et illustration de NM. 17
Figure 2.3 Manque d'efficacité de NM à l'approche d'une discontinuité. 18
Figure 2.4 Illustration d'une Tr. 19
Figure 2.5 Manque d'efficacité d'une Tr sur un cas discontinu. 20
Figure 2.6 Illustration du krigeage et d'Ego. 21
Figure 2.7 Manque d'efficacité du krigeage et d'Ego sur un cas discontinu. 21
Figure 2.8 Manque de pertinence d'un gradient simplexe sur un cas discontinu. . . 22
Figure 2.9 Ensembles de directions admissibles sur le treillis adaptatif de Mads. . . 24
Figure 3.1 Illustration d'ensembles ayant l'ECP ou non. 30
Figure 3.2 Fonction objectif $f_{\mathtt{CE}}$ du contre-exemple. 32
Figure 3.3 Quatre premières itérations de DSM sur le contre-exemple. 33
Figure 3.4 Échec de DSM à visiter chaque ensemble de continuité. 35
Figure 3.5 Itération 0 de rDSM résolvant le contre-exemple avec $r = 2$. 43
Figure 4.1 Répartition de tirages uniformes indépendants dans la boule unité. . . . 46
Figure 4.2 Mise en échec de rDSM lorsque plusieurs points raffinés existent. 47
Figure 4.3 Illustration d'ensembles amples ou non. 53
Figure 5.1 Fonction plus simple à minimiser via l'optimisation partitionnée. 61
Figure 6.1 Fonction φ (tronquée), partition de $\mathbb{Y}$ et image de γ au chapitre 6.1.1. . 76
Figure 6.2 Fonctions ε et Φ au chapitre 6.1.1. 76
Figure 6.3 Fonction φ, partition de $\mathbb{Y}$ et image de γ au chapitre 6.1.2. 78
Figure 6.4 Fonctions ε et Φ au chapitre 6.1.2. 78
Figure 6.5 Fonction φ, partition de $\mathbb{Y}$ et image de γ au chapitre 6.1.3. 80
Figure 6.6 Fonctions ε et Φ au chapitre 6.1.3. 80
Figure 6.7 Coupes de la fonction φ au chapitre 6.1.4. 82
Figure 6.8 Partition de $\mathbb{Y}$ et image de γ au chapitre 6.1.4. 82
Figure 6.9 Fonctions ε, M et Φ au chapitre 6.1.4. 82
Figure 6.10 Gains apportés par l'optimisation partitionnée au chapitre 6.2.1. 84
Figure 6.11 Gains apportés par l'optimisation partitionnée au chapitre 6.2.2. 85
Figure 6.12 Gains apportés par l'optimisation partitionnée au chapitre 6.2.3. 86
Figure 6.13 Gains apportés par l'optimisation partitionnée au chapitre 6.2.4. 88
Figure 6.14 Surplus de gain apporté par un solveur performant au chapitre 6.2.4. . 88
Figure 7.1 Cible circulaire et illustration de trajectoires approchant son centre. . . 92

Figure 7.2 Fonction *partie entière* et ses lissages. 93
Figure 7.3 Exemples de plateaux de différents coûts. 93
Figure 7.4 Modélisation du courant de la rivière dans le problème de Zermelo. . . 102
Figure 7.5 Cibles discrètes à atteindre sur la rivière. 103
Figure 7.6 Résultats numériques sur le problème de cibles discrètes de Zermelo. . . 103
Figure 7.7 Qualité mesurée des bollards sur la rive nord de la rivière. 105
Figure 7.8 Résultats numériques sur le problème de plateaux de Zermelo. 105

LISTE DES NOTATIONS NON USUELLES DE MATHÉMATIQUES

Le tableau ci-dessous récapitule l'ensemble des notations mathématiques possiblement non usuelles utilisées dans ce manuscrit. Nous espérons que le manque de formalisme dans ce tableau ne perturbe pas la compréhension. Cette page ne vise pas à introduire formellement ces notations ; toutes sont définies formellement dans le document au moment où elles apparaissent pour la première fois.

Tableau 0.1 Notations non usuelles utilisées dans ce manuscrit

Notation	Signification
$\triangleq$	Égal par définition (dit autrement, défini comme égal à)
$\doteq$	Redéfinition
$D^{-\top}$	Inverse de la transposée de la matrice inversible D, $D^{-\top} \triangleq (D^\top)^{-1}$
$\overline{\mathbb{R}}$, $\overline{\mathbb{N}}$	Respectivement, $\mathbb{R} \cup \{\pm\infty\}$ et $\mathbb{N} \cup \{+\infty\}$
$\overline{\mathbb{R}}^n$	Espace des vecteurs de $n \geq 1$ éléments de $\overline{\mathbb{R}}$
$\mathbb{R}^*$, $\mathbb{N}^*$	Respectivement, $\mathbb{R} \setminus \{0\}$ et $\mathbb{N} \setminus \{0\}$
$\mathbb{S}^n$	Sphère unité de $\mathbb{R}^n$, pour une norme donnée
$\mathrm{cl}(\mathcal{X})$, $\mathrm{int}(\mathcal{X})$, $\partial\mathcal{X}$	Respectivement, fermeture, intérieur et frontière de l'ensemble $\mathcal{X}$
$\mathrm{adh}(\cdot)$	Ensemble des valeurs d'adhérences d'une suite donnée
$\mathcal{B}_r(x)$	Boule ouverte de rayon r centrée en x
$\mathcal{B}_r$	Boule ouverte de rayon r centrée en l'origine
$\#\mathcal{X}$	Cardinal de l'ensemble $\mathcal{X}$
$2^{\mathcal{X}}$	Ensemble des parties de $\mathcal{X}$
$\mathcal{X}_1 + \mathcal{X}_2$	Somme de Minkowski entre les ensembles $\mathcal{X}_1$ et $\mathcal{X}_2$
$f(\mathcal{X})$	Ensemble des images de $\mathcal{X}$ par la fonction f
$f^{-1}(\mathcal{Y})$	Ensemble des antécédents de $\mathcal{Y}$ par la fonction f
$\mathcal{I}_{\mathcal{X}}$	Fonction indicatrice de $\mathcal{X}$
$\sqcup$	Union disjointe
$f_{\vert\mathcal{X}}$	Restriction de la fonction f à $\mathcal{X}$
$\mathcal{V}_f(x)$	Ensemble $\mathcal{V}_f(x) \triangleq \cap_{r>0}\ \mathrm{cl}(f(\mathcal{B}_r(x)))$
$\mathrm{round}_\delta^{\mathcal{M}}(\cdot)$	Fonction d'arrondi sur le treillis $\mathcal{M}(\delta)$
$\mathrm{dist}(x, \mathcal{X})$	Distance du point x à l'ensemble $\mathcal{X}$
$\lfloor\cdot\rfloor$, $\lceil\cdot\rceil$	Respectivement, fonction de partie entière et de partie plafond
$\lfloor\cdot\rceil$	Fonction $\lfloor x \rceil \triangleq \lfloor x \rfloor$ si $x \leq 0$ et $\lfloor x \rceil \triangleq \lceil x \rceil - 1$ si $x > 0$

PLAN DU MANUSCRIT ET DÉPENDANCE DES CHAPITRES

Ce manuscrit est composé de huit chapitres, résumés ci-dessous.

Le chapitre 1 est une introduction à la problématique de cette thèse : l'optimisation sans dérivées (DFO) dans un contexte discontinu. Le chapitre 1.1 précise ce qu'est la DFO, et dans quels contextes des discontinuités peuvent apparaître. Le chapitre 1.2 poursuit avec une revue des notions mathématiques fondamentales en DFO. Enfin, le chapitre 1.3 formalise les objectifs de recherche ayant motivé cette thèse et auxquels ce manuscrit répond. La lecture de ce chapitre est optionnelle, car seul le problème d'optimisation (**P**) formalisé au chapitre 1.2.3 intervient dans la suite de ce manuscrit.

Le chapitre 2 passe en revue la littérature en DFO, avec une emphase sur le contexte discontinu. Le chapitre 2.1 cerne les limites du domaine en identifiant les problèmes sur lesquels la littérature s'accorde à dire que la DFO n'est pas appropriée. Ensuite, le chapitre 2.2 démontre le manque d'efficacité de toutes les familles de méthodes communes de la DFO dans des contextes discontinus. La famille des *méthodes de recherche directes* (DSM) est développée plus en détails car nous croyons qu'elle est la plus prometteuse dans ce contexte. Ce chapitre peut être lu sans pré-requis, et est optionnel à la compréhension de la suite.

Ensuite, le chapitre 3 fait la transition entre l'état de l'art et les nouveautés apportées par cette thèse. L'objectif est de démontrer que l'un des résultats de l'article « Analysis of direct searches for discontinuous functions », une référence pionnière dans l'étude de DSM dans un contexte discontinu, contient une erreur dans l'un de ses théorèmes. Notre contribution est d'identifier cette erreur et de la corriger. Le chapitre 3.1 synthétise le raisonnement mené par « Analysis of direct searches for discontinuous functions ». Le chapitre 3.2 met en évidence la partie erronée du raisonnement. Enfin, le chapitre 3.3 propose une adaptation de DSM évitant l'écueil affectant l'analyse précédente. L'algorithme modifié, rDSM, dépend de DSM mais est présenté sans dépendance au chapitre 2. Le chapitre 3 peut être lu sans pré-requis.

L'analyse de rDSM est enrichie au chapitre 4. Le chapitre 4.1 prouve que rDSM offre en réalité un résultat plus fort que ce que « Analysis of direct searches for discontinuous functions » tentait d'établir. Nous prouvons la convergence vers une solution locale du problème (**P**), et nous identifions également comment lever les hypothèses restrictives que cette convergence requiert. Ces généralisations sont effectuées au chapitre 4.2. Nous y introduisons l'algorithme cDSM comme généralisation de rDSM, et nous proposons une analyse de convergence dont nous prouvons qu'elle est la plus générale possible. Il est conseillé de maîtriser les enjeux du chapitre 3 et de comprendre l'algorithme rDSM avant de commencer la lecture du chapitre 4.

Le chapitre 5 poursuit avec une stratégie d'optimisation dite *partitionnée* pour élargir les frontières de la DFO. Par une technique de reformulation d'un problème, la DFO peut parfois traiter des problèmes de dimension trop grande pour les approches usuelles. Le chapitre 5.1 présente cette reformulation. Une solution du problème reformulé permet de trouver une solution du problème initial. De fait, au chapitre 5.2 nous appliquons cDSM au problème reformulé pour en obtenir des solutions. Enfin, le chapitre 5.3 identifie des pistes pour simplifier l'applicabilité de la reformulation en pratique. Malgré sa dépendance à l'algorithme cDSM, le chapitre 5 est écrit pour être indépendant du chapitre 4 à la lecture.

Le chapitre 6 consiste en un versant numérique et pratique de l'étude de la reformulation proposée au chapitre 5. D'abord, le chapitre 6.1 illustre comment reformuler un problème de DFO suivant cette stratégie, via quatre exemples. Ensuite, le chapitre 6.2 complexifie ces quatre exemples, et compare les performances de solveurs performants de la DFO sur ces problèmes à une implémentation simple de cDSM résolvant la reformulation. Nous mettons ainsi en évidence le gain sensible de performance apporté par la reformulation. Le chapitre 6 fait de nombreuses références au chapitre 5, aussi la lecture préalable de ce dernier est conseillée.

Enfin, le chapitre 7 propose une seconde illustration de l'intérêt de la reformulation. Nous l'appliquons ici à des problèmes issus du contrôle optimal présentant des discontinuités. Le contrôle optimal et sa littérature dédiée aux discontinuités sont présentés au chapitre 7.1. L'application de la stratégie du chapitre 5 suit au chapitre 7.2. Une illustration sur trois exemples est proposée au chapitre 7.3. Enfin, des pistes d'amélioration de la performance dans le contexte spécifique du contrôle optimal sont identifiées au chapitre 7.4. Il est conseillé de maîtriser le contenu du chapitre 5 pour comprendre le chapitre 7 en profondeur.

Le manuscrit se termine sur le chapitre 8, qui offre une synthèse des travaux discutés et des possibles travaux futurs qui pourraient en découler. Ce chapitre peut être lu sans avoir connaissance du détail du contenu des chapitres le précédant.

Tableau 0.2 Contenu des chapitres et dépendances de lecture.

(§) Chapitre et thème abordé	Dépendances
(1) Introduction : enjeux et objectifs	/
(2) État de l'art : manque de méthodes gérant des discontinuités	/
(3) Correctif état de l'art : correction d'une erreur dans une référence	/
(4) Nouvelles garanties théoriques : analyse poussée des méthodes	(3)
(5) Optimisation partitionnée : repousser les limites de la DFO	/
(6) Illustrations : mise en application de l'optimisation partitionnée	(5)
(7) Contrôle optimal : exemple d'application exigeante	(5)
(8) Synthèse des travaux : état actuel et extensions possibles	/

CHAPITRE 1 INTRODUCTION

Les fondations, là-bas. (...) Et juste là, un petit géranium. Ouais, ça va être bien, ça va être très bien même. Bon, bien sûr, là faut imaginer. Allez, j'y vais ! À dans un mois, maximum ! Le temps de trouver Panobémémix, et de revenir.

« Astérix et Obélix : Mission Cléopâtre », tirade de Jamel Debbouze écrite par Alain Chabat.

Le concept d'*optimisation* désigne une branche des mathématiques visant à calculer une solution la plus satisfaisante possible à un problème donné. Un bref historique de l'optimisation, pour cerner les enjeux auxquels elle répond, est donné au chapitre 1.1. Ensuite, le chapitre 1.2 formalise les notions mathématiques requises à la formalisation de ce qu'est l'optimisation. Le chapitre 1.3 suit en mettant en évidence les questions ouvertes à laquelle la thèse répond et annonce la structure de ce manuscrit.

1.1 Bref survol de l'historique de l'optimisation

1.1.1 Optimisation calculatoire : des premiers travaux à Newton

Les premiers résultats d'optimisation connus remontent à la fin de l'Antiquité. Nous avons connaissance de travaux d'Archimède puis d'Héron à la fin du Ier siècle, portant sur la réflexion d'un objet dans un miroir. Ils calculèrent que le plus court chemin reliant un observateur, le miroir, et l'objet reflété est exactement le chemin que prend le regard de l'observateur pour voir l'objet à travers le miroir. Cette branche des mathématiques est donc très ancienne. Cependant, son premier essor notable eut lieu au XVIIème siècle, suite à la découverte de la notion de *dérivée* intuitée par Fermat puis grandement étudiée par Newton et Leibniz. Newton, qui la nomma *fluxion*, comprit et énonça le fait suivant :

> « Une quantité qui est devenue la plus grande ou la moindre qu'il se peut, n'augmente ni ne diminue, c'est-à-dire, ne flue ni en avant ni en arrière dans cet instant ; car si elle augmente, c'est une marque qu'elle était plus petite et que tout à l'heure elle va être plus grande qu'elle n'était, ce qui est contre la supposition, et c'est le contraire si elle diminue. Ainsi, trouvez sa fluxion et supposez-la nulle. »

Cet extrait conceptualise une notion devenue cruciale en optimisation : étant donné un problème consistant à minimiser une quantité donnée, une réponse proposée à ce problème doit annuler la fluxion pour potentiellement être une solution. Ceci offre une stratégie efficace pour résoudre de nombreux problèmes. En effet, le seul pré-requis de cette méthode est que le problème étudié consiste en la minimisation d'une quantité dont on sait calculer la fluxion.

L'optimisation devint alors un outil puissant en sciences physiques, car la notion de fluxion y trouve une interprétation naturelle. En ce sens, l'optimisation est depuis lors un domaine des mathématiques dites *appliquées* (ou plus précisément, des *mathématiques applicables*). Elle permet d'exploiter le *principe de moindre action*, énoncé par Maupertuis au milieu du XVIIIème siècle et pressenti par Leibniz auparavant :

> « Lorsqu'il arrive un changement dans la nature, la quantité d'action instantanée nécessaire pour ce changement est la plus petite possible. »

Il s'agit d'un principe *calculatoire*, stipulant que les calculs effectués par les physiciens consistent en l'annulation de la fluxion de l'action. Par exemple, ce principe guide la mécanique d'un solide, en affirmant que son énergie totale doit varier de façon minimale.

Newton n'était cependant pas pleinement satisfait de sa notion de fluxion. De même, Leibniz observait des imperfections dans sa notion analogue de *quotient ultime de deux accroissements évanescents*. Ces deux concepts furent finalement unifiés et formalisés par Cauchy et Weierstrass au XIXème siècle. Ils mirent en évidence deux faits importants. Le premier est une méthode de calcul pour exprimer la dérivée d'une fonction ; ainsi le critère de « fluxion nulle » de Newton a un intérêt pratique car le calcul de la dérivée est explicite. Le second est que la notion de dérivée n'est pas universelle, au sens où toutes les fonctions n'admettent pas de dérivée. Malgré tout, la dérivée est un outil fondamental dans d'innombrables travaux en mathématiques et en science physique. Par conséquent, les cas où elle n'est pas définie sont plus difficiles à traiter, et historiquement bien moins étudiés. Pourtant, de nombreux problèmes concrets comportent des difficultés mettant en échec le calcul des dérivées. Dans certains de ces problèmes, la dérivée n'existe même pas en théorie ; dans d'autres, même si elle existe, il est déraisonnable de la calculer car ceci est trop difficile ou trop coûteux.

1.1.2 Optimisation algorithmique : l'essor de l'informatique

L'optimisation connut un second essor au milieu du XXème siècle, motivé par l'apparition puis la complexification des problématiques en ingénierie et par le développement de l'informatique. La plupart des méthodes modernes résultent de ce second essor. Elles sont de nature *algorithmiques* et non plus calculatoires, puisqu'un ordinateur peut exécuter très efficacement un algorithme bien implémenté. L'optimisation est maintenant majoritairement une discipline de conception d'algorithmes pouvant être exécutés par un ordinateur pour résoudre des problèmes complexes. Parmi les applications actuelles, nous pouvons citer :

— le domaine du transport, pour minimiser des coûts de déplacement à travers un réseau ;

— la planification de la gestion de ressources, pour décider quand allouer des ressources et en quelle quantité afin d'éviter au maximum des pénuries ;

— l'ingénierie, où des objectifs de coût, de robustesse ou autres sont omniprésents ;

— la finance, pour maximiser ses gains via de bons choix d'investissement, en connaissant l'état actuel du marché et en prédisant l'état futur ;

— les statistiques, l'intelligence artificielle et les modèles mathématiques en général, qui dépendent de nombreux paramètres qu'il convient de régler au mieux.

1.1.3 Cas particulier de l'optimisation sans dérivées

Le cadre d'étude nommé *optimisation sans dérivées* apparut plus tardivement, mais est lui aussi riche d'applications réelles. Celles-ci ont à la fois motivé son développement et confirmé sa maturité. La décennie 1950-60 vit naître certaines méthodes sans dérivées au détour d'expérimentations avec l'informatique alors naissante, puis la majorité des travaux précurseurs visant explicitement à éviter le calcul de dérivées arrivèrent dans la décennie 1980-90. Depuis, le domaine est en expansion constante et a maintenant atteint une grande maturité. Citons par exemple les applications suivantes, toutes résolues via des méthodes sans dérivées par des chercheurs du laboratoire m'ayant hébergé durant ma thèse :

— le logiciel **MATLAB**, très populaire en ingénierie, contient des fonctions encodant la méthode mathématique de *Runge-Kutta*. Cette méthode dépend de paramètres, qu'il est possible d'optimiser par des méthodes sans dérivées. Lorsqu'on lance la méthode sur une classe de problèmes spécifique, les paramètres optimisés pour cette classe sont 7 à 300 fois meilleurs que les paramètres par défaut de **MATLAB** [7] ;

— pour gérer le réseau électrique alimentant le Québec, le fournisseur Hydro-Québec dispose d'outils analysant le niveau d'usure des génératrices. Les conditions difficiles rendent la rupture des pièces difficile à prévoir, coûteuse à gérer, et cruciale à anticiper. Il faut donc définir à l'avance des scénarios de remplacement des pièces. Pour ce faire, Hydro-Québec et son équivalent français EDF ont développé un simulateur prédisant les conséquences à long terme d'une stratégie de remplacement. L'horizon de prédiction est d'environ 25 ans. Sur une telle échelle, la moindre variation dans la stratégie peut avoir de grandes conséquences. La recherche de la stratégie minimisant le coût de gestion calculé par le simulateur a donc nécessité une technique sans dérivées [3] ;

— un phénomène récurrent en ingénierie des turbomachines est le risque de rupture de pièces, à cause de vibrations importantes à la fréquence de résonance. Des techniques existent pour dimensionner les moteurs afin d'éviter ces ruptures. Elles sont fiables, mais coûteuses. Des techniques sans dérivées permettent de les exploiter le plus efficacement possible. Une thèse a été dédiée à ce sujet, impliquant un chercheur en optimisation sans dérivées et un autre en mécanique pour encadrer une étudiante maîtrisant les

deux aspects [71]. L'une des difficultés notables dans cette application était la présence de discontinuités dans le modèle mathématique (qui représentait les phénomènes de rupture) qu'il fallait détecter afin de s'en éloigner suffisamment.

Le cadre de travail usuel en optimisation sans dérivées est celui visant à minimiser des fonctions non différentiables mais malgré tout *continues*. En effet, sous l'hypothèse de continuité, il est possible de généraliser efficacement la notion de dérivée. Ainsi, les techniques issues de l'optimisation avec dérivées admettent une généralisation au cas sans dérivées mais avec continuité. La théorie dans le cas continu est donc bien comprise.

Le travail mené dans cette thèse se situe dans le cadre où la notion de continuité n'est pas applicable. En d'autres mots, le contexte considéré est celui de l'*optimisation discontinue*. Ce cadre est encore très récent, car certains travaux considérés comme fondateurs datent des années 2010. Parmi les problèmes cités plus haut, certains ont requis des techniques d'optimisation discontinue. En particulier le troisième, car les discontinuités observées ne sont pas des artifices mathématiques mais correspondent à des problèmes réels : elles traduisent des dimensionnements de la pièce menant à des ruptures. Très souvent, lorsqu'un problème consiste à dimensionner une pièce mécanique, la présence de discontinuités dans le modèle mathématique traduit un risque de rupture de la pièce. De manière plus générale, une discontinuité dans un modèle mathématique traduit souvent un changement de comportement dans la réalité physique décrite par le modèle. Pour des considérations pratiques, il est donc désirable d'avoir des certitudes que la méthode d'optimisation employée est toujours capable de détecter ces discontinuités. Pour ce faire, les *dérivées généralisées* peuvent apporter un élément de réponse. Une autre solution est de connaître la sensibilité de la solution calculée, en évaluant suffisamment de solutions potentielles voisines à celle renvoyée. Ceci offre des garanties qu'aucune discontinuité potentielle proche de la solution n'a échappé à l'analyse.

L'objectif de cette thèse est de contribuer à la compréhension du cadre théorique de l'optimisation discontinue. Nous étudions le comportement de méthodes classiques de l'optimisation sans dérivées dans ce nouveau cadre. Nous observons des différences avec le cadre continu, ce qui met en évidence les problèmes engendrés par l'absence de continuité. Ensuite, nous généralisons les méthodes afin de surmonter ces problèmes. Enfin, nous montrons que même en présence de discontinuités, il est parfois possible d'identifier et d'exploiter certaines structures dans un problème d'optimisation pour maximiser l'efficience des méthodes de résolution. Nous illustrons comment repérer et exploiter une telle structure dans plusieurs classes de problèmes d'optimisation et de contrôle optimal.

1.2 Définitions et concepts fondamentaux en optimisation discontinue

Commençons par définir le cadre d'étude dans lequel s'inscrit cette thèse. Nous rappelons d'abord, au chapitre 1.2.1, quelques notions liées à la discontinuité d'une fonction. Ensuite, le chapitre 1.2.2 formalise la notion de dérivée puis de quelques dérivées généralisées classiques. Le cadre général de l'optimisation sans dérivées suit au chapitre 1.2.3.

1.2.1 Notions liées à la continuité et discontinuité d'une fonction

Nous rappelons ici la notion de continuité d'une fonction, avant de poursuivre avec des notions liées aux fonctions discontinues.

Continuité. Une fonction $f : \mathbb{R}^n \to \mathbb{R}$ est *continue* en $x \in \mathbb{R}^n$ lorsque

$$\forall \varepsilon > 0, \quad \exists \delta > 0 : \qquad |f(x+h) - f(x)| \leq \varepsilon \quad \text{pour tout } h \in \mathbb{R}^n \text{ tel que } \|h\| \leq \delta.$$

Continuité de Lipschitz. Une notion légèrement plus restrictive est la notion de *Lipschitz-continuité*. La fonction f est Lipschitz-continue en x avec constante de Lipschitz $\mathcal{L} \geq 0$ lorsque

$$\exists \delta > 0 : \qquad |f(x+h) - f(x)| \leq \mathcal{L}\,\|h\| \quad \text{pour tout } h \in \mathbb{R}^n \text{ tel que } \|h\| \leq \delta.$$

Semi-continuité. Une fonction $f : \mathbb{R}^n \to \mathbb{R}$ est *semi-continue* (ou plus précisément, *semi-continue inférieurement*) en $x \in \mathbb{R}^n$ lorsque

$$\forall \varepsilon > 0, \quad \exists \delta > 0 : \qquad f(x+h) \geq f(x) - \varepsilon \quad \text{pour tout } h \in \mathbb{R}^n \text{ tel que } \|h\| \leq \delta.$$

Une fonction semi-continue peut ne pas être continue, par exemple la fonction $\mathcal{I}_{\mathbb{R}_+^*} : \mathbb{R} \to \mathbb{R}$ définie par $\mathcal{I}_{\mathbb{R}_+^*}(x) \triangleq 0$ si $x \leq 0$ et $\mathcal{I}_{\mathbb{R}_+^*}(x) = 1$ si $x > 0$ est semi-continue mais discontinue.

Domaine effectif, ensembles de continuité. Le *domaine effectif* d'une fonction $f : \mathbb{R}^n \to \overline{\mathbb{R}} \triangleq \mathbb{R} \cup \{\pm\infty\}$ est l'ensemble $X \triangleq \{x \in \mathbb{R}^n : f(x) \neq \pm\infty\}$. Une partition de X en *ensembles de continuité* de f est une partition $X = \sqcup_{i=1}^N X_i$, où $N \in \overline{\mathbb{N}}^* \triangleq \mathbb{N}^* \cup \{+\infty\}$, telle que $f_{|X_i}$ est continue pour tout $i \in [\![1, N]\!]$. Il n'y a pas unicité d'une telle partition.

Valeurs limites en un point. Pour une fonction $f : \mathbb{R}^n \to \overline{\mathbb{R}}^m$, l'ensemble des *valeurs limites* de f en $x \in \mathbb{R}^n$ est l'ensemble $\mathcal{V}_f(x) \subseteq \overline{\mathbb{R}}^m$ des valeurs limites atteignables par $f(x')$ lorsque $x' \to x$, défini par $\mathcal{V}_f(x) \triangleq \cap_{r>0} \operatorname{cl}(f(\mathcal{B}_r(x)))$ (pour la topologie usuelle de $\overline{\mathbb{R}}^m$).

1.2.2 Notions d'analyse par différentiabilité généralisée

L'analyse par différentiabilité étudie les variations d'une fonction $f : \mathbb{R}^n \to \mathbb{R}$ afin d'en déduire des propriétés sur les valeurs prises par f autour d'un point de référence. Il s'agit d'une notion centrale en optimisation. Elle formalise la remarque de Newton à propos de conditions qu'une réponse proposée à un problème d'optimisation doit vérifier pour potentiellement être une solution. Ici nous donnons d'abord la notion de dérivée et ses extensions usuelles comme les dérivées directionnelles. Nous poursuivons avec deux généralisations restant valides dans des cas où la dérivée usuelle n'est pas définie. Pour chacune de ces notions, nous mettons également en évidence le fait que l'intuition de Newton à propos de la « fluxion devant être nulle » est exacte, et se traduit mathématiquement. Cette présentation n'a pas vocation à être exhaustive, et beaucoup d'autres notions de dérivées généralisées existent. Cependant, celles discutées ici sont les plus populaires en optimisation sans dérivées.

Dérivée, dérivée directionnelle, gradient. Une fonction $f : \mathbb{R} \to \mathbb{R}$ continue en $x \in \mathbb{R}$ est dite *dérivable en x* lorsque la quantité

$$f'(x) \triangleq \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

existe et est finie. La valeur $f'(x)$ est alors nommée *dérivée de f en x*. Lorsque f est définie de $\mathbb{R}^n$ dans $\mathbb{R}$, la notion se généralise au sens des *dérivées partielles*. Pour tout $i \in [\![1, n]\!]$, la *i-ème dérivée partielle* de f en $x \in \mathbb{R}^n$ est

$$f'_i(x) \triangleq \lim_{h \to 0} \frac{f(x + he_i) - f(x)}{h},$$

où e_i désigne le i-ème vecteur de la base canonique de $\mathbb{R}^n$. La fonction est alors dite dérivable lorsque la quantité $f'_i(x)$ existe et est finie pour tout $i \in [\![1, n]\!]$. On forme alors le *gradient de f en x* comme le vecteur

$$\nabla f(x) \triangleq \begin{bmatrix} f'_1(x) \\ \vdots \\ f'_n(x) \end{bmatrix}.$$

L'intuition de Newton citée plus haut trouve alors une formalisation via le gradient.

Proposition 1.1. Soit $x \in \mathbb{R}^n$ et $f : \mathbb{R}^n \to \mathbb{R}$ dérivable en x. Si x est un minimiseur local de f, alors $\nabla f(x) = 0$.

Dérivée de Dini. Il existe des variantes de ces notions, comme celle de *dérivée directionnelle de Dini de f en x dans la direction $d \in \mathbb{S}^n$* (où $\mathbb{S}^n$ désigne la sphère unité de $\mathbb{R}^n$),

$$f'_{\mathtt{D}}(x;d) \triangleq \lim_{h \searrow 0} \frac{f(x+hd)-f(x)}{h}.$$

La dérivée directionnelle de Dini peut exister en un point en lequel les dérivées partielles ne le sont pas. Avec cette définition, il est possible d'adapter la proposition 1.1 comme suit :

Proposition 1.2. Soit $x \in \mathbb{R}^n$ et $f : \mathbb{R}^n \to \mathbb{R}$ dérivable en $(x;d)$ au sens de Dini pour tout $d \in \mathbb{S}^n$. Si x est un minimiseur local de f, alors $f'_{\mathtt{D}}(x;d) \geq 0$ pour tout $d \in \mathbb{S}^n$.

Hessien. Si f admet des *dérivées partielles secondes* en x (définies comme les dérivées partielles de f'_i en x, pour tout $i \in [\![1,n]\!]$), on définit le *hessien* de f en x comme

$$\nabla^2 f(x) \triangleq \begin{bmatrix} (f'_1)'_1(x) & \dots & (f'_n)'_1(x) \\ \vdots & \ddots & \vdots \\ (f'_1)'_n(x) & \dots & (f'_n)'_n(x) \end{bmatrix}.$$

Le hessien permet d'établir un résultat plus fort qu'aux propositions 1.1 et 1.2. Ces dernières donnent des *conditions nécessaires d'optimalité*, alors que la proposition suivante donne des *conditions suffisantes d'optimalité*.

Proposition 1.3. Soit $x \in \mathbb{R}^n$ et $f : \mathbb{R}^n \to \mathbb{R}$ admettant un hessien en x. Si $\nabla f(x) = 0$ et $\nabla^2 f(x)$ est défini positif, alors x est un minimiseur local de f.

Dérivées généralisées

Lorsqu'une fonction continue n'est pas dérivable, la proposition 1.1 n'est plus applicable. La proposition 1.2 peut parfois rester valide, mais beaucoup de fonctions continues mettent en échec les deux propositions. Pour pallier à ce manque, des généralisations de la dérivée existent. L'une des plus communes est la *dérivée de Clarke* [32], généralisant celle de Dini.

Définition 1.1 (Dérivée de Clarke). Pour tout $x \in \mathbb{R}^n$ et $d \in \mathbb{S}^n$, la dérivée de Clarke de $f : \mathbb{R}^n \to \mathbb{R}$ en $(x;d)$ est

$$f^{\circ}_{\mathtt{C}}(x;d) \triangleq \limsup_{y \to x,\ h \searrow 0} \frac{f(y+hd)-f(y)}{h}.$$

La dérivée de Clarke requiert la Lipschitz-continuité de f en x pour garantir le renvoi de valeurs finies. Cette notion mène à une proposition analogue à la proposition 1.2.

Proposition 1.4. Soit $x \in \mathbb{R}^n$ et $f : \mathbb{R}^n \to \mathbb{R}$ Lipschitz-continue en x. Si x est un minimiseur local de f, alors $f_{\mathtt{C}}^{\circ}(x; d) \geq 0$ pour tout $d \in \mathbb{S}^n$.

Enfin, une autre dérivée généralisée notable est la *dérivée de Rockafellar* [99, 116], car elle est pensée pour une application aux fonctions possiblement discontinues.

Définition 1.2 (Dérivée de Rockafellar). Pour tout $x \in \mathbb{R}^n$ et $d \in \mathbb{S}^n$, la dérivée de Rockafellar de $f : \mathbb{R}^n \to \mathbb{R}$ en $(x; d)$ est

$$f_{\mathtt{R}}^{\circ}(x; d) \triangleq \limsup_{\substack{y \to x,\ h \searrow 0, \\ f(y) \to f(x)}} \frac{f(y + hd) - f(y)}{h}.$$

La dérivée de Rockafellar requiert uniquement que $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ soit semi-continue en $x \in X$ pour avoir du sens. On a de plus la condition nécessaire d'optimalité suivante.

Proposition 1.5. Soit $x \in \mathbb{R}^n$ et $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ semi-continue en $x \in X$. Si x est un minimiseur local de f, alors $f_{\mathtt{R}}^{\circ}(x; d) \geq 0$ pour tout $d \in \mathbb{S}^n$.

Notons toutefois qu'il n'existe pas d'équivalent à la proposition 1.3 lorsque f n'est pas dérivable. Il est donc difficile de certifier qu'un point est un minimiseur local dans ce cas.

1.2.3 Problème sans dérivées, conditions d'optimalité envisageables

Une formulation très générique d'un problème d'optimisation sans dérivées (en anglais *derivative-free optimization*, souvent abrégé `DFO`) est donnée par le cadre suivant.

Cadre 1.1. **Cadre de la `DFO`.** On considère le problème d'optimisation

$$\underset{x \in \mathbb{X}}{\text{minimiser}} \quad f(x) \quad \text{sous contrainte} \quad x \in \Omega, \tag{P}$$

où l'*espace des variables* $\mathbb{X}$ est un espace normé, la *fonction objectif* $f : \mathbb{X} \to \overline{\mathbb{R}}$ admet un domaine effectif $X \subseteq \mathbb{X}$ non vide, et $\Omega \subseteq \mathbb{X}$ est *l'ensemble des points réalisables*. On suppose de plus que $f_{|\Omega}$ est bornée inférieurement et que ses ensembles de niveau sont bornés, que $f_{|X}$ est semi-continue, et que Ω est fermé et intersecte X.

Dans la plupart des cas usuels, l'espace $\mathbb{X}$ est $\mathbb{R}^n$ muni de sa norme infinie, pour un certain $n \in \mathbb{N}^*$ que l'on nomme la *dimension du problème*. Cette thèse est écrite autant que possible dans ce cas, mais nous devons revenir au cas général dans les chapitres 5 et 7.

En présence de contraintes ($\Omega \neq \mathbb{R}^n$), les propositions 1.4 et 1.5 doivent être adaptées en conséquence. Ceci est discuté au chapitre 3. Lorsque la fonction objectif est continue, la proposition 1.4 (après adaptation) fournit un critère pour chercher des solutions au problème (**P**). En effet, la proposition 1.4 est une condition nécessaire d'optimalité, donc la première étape dans la résolution du problème (**P**) consiste à chercher des points en lesquels la proposition 1.4 est vérifiée. Une vérification directe est difficile en général, car la valeur exacte de la dérivée de Clarke est difficile à calculer, mais il est aisé d'en obtenir une minoration sur laquelle on a un bon contrôle. Ainsi, l'approche commune en `DFO` consiste à concevoir des méthodes générant des points en lesquels la minoration de la dérivée de Clarke est positive. Cela nous garantit que la dérivée de Clarke est elle-même positive en ces points, et donc que la proposition 1.4 tient. Il est en général difficile d'obtenir une condition suffisante d'optimalité qui permettrait réciproquement de déterminer si un point est un minimiseur local de f, aussi la littérature en `DFO` discute peu de cet objectif supplémentaire.

1.3 Objectifs de recherche

Lorsque f n'est pas continue, il existe une analogie directe à la stratégie utilisée dans le cas continu. En effet, la proposition 1.5 est également une condition nécessaire d'optimalité. Cependant, elle est difficile à exploiter en pratique. Contrairement à la dérivée de Clarke, la dérivée de Rockafellar n'a pas de minoration facilement exploitable. Un des travaux pionniers de l'optimisation discontinue, paru en 2010, visait à surmonter cette difficulté [116]. Cependant, la problématique reste ouverte car nous avons observé que l'analyse proposée par [116] comporte une erreur. Le premier objectif de cette thèse est d'identifier cette erreur, puis de la corriger en adaptant la méthode proposée par [116].

Le second objectif de cette thèse est d'établir une nouvelle stratégie permettant de garantir une optimalité locale. Nous généralisons une méthode usuelle de la `DFO` pour lui donner des garanties de calculer une solution locale au problème (**P**), même lorsque la fonction objectif est discontinue. Pour ce faire, nous nous assurons que la méthode évalue un ensemble dense de points autour de la solution renvoyée. Nous précisons également sous quelles hypothèses nous obtenons ce résultat, et nous montrons que nos hypothèses sont les plus faibles possibles. Ainsi, notre résultat est le plus général possible. De plus, aucun travail précédent n'a obtenu de résultat aussi fort dans un cadre aussi générique.

En troisième objectif, cette thèse vise ensuite à augmenter la dimension maximale que la DFO peut traiter, même en présence de discontinuités. Nous identifions une structure potentielle d'un problème d'optimisation qui peut être exploitée pour accélérer la résolution. Plus précisément, nous montrons que même lorsque la fonction objectif est mal structurée, potentiellement discontinue, il arrive que seules quelques-unes des variables soient responsables de la non-dérivabilité rendant le problème difficile. Si de plus ces variables sont identifiées, nous proposons une stratégie *hybride* : il s'agit d'optimiser ces variables par un algorithme de DFO, et en parallèle d'optimiser toutes les autres via une méthode de l'optimisation usuelle exploitant les dérivées. Nous prouvons que procéder ainsi préserve la garantie de notre méthode de calculer une solution locale au problème.

Enfin, en quatrième objectif, nous mettons en évidence le fait que cette stratégie hybride est plus performante qu'une approche purement sans dérivées, et également plus satisfaisante qu'une tentative de forcer l'usage d'une méthode issue de l'optimisation avec dérivées. Nous illustrons ce fait sur plusieurs classes de problèmes de DFO de grande dimension, puis sur des problèmes issus d'un autre domaine de l'optimisation ; le contrôle optimal. La présence de discontinuités dans un problème de contrôle optimal est mal maîtrisée par la littérature. De plus, la DFO usuelle est incapable de répondre à ce manque car le contrôle optimal traite de problèmes de dimension infinie. Cependant, nous montrons que grâce à notre méthode, nous pouvons exploiter à la fois les forces de la littérature en contrôle optimal et celles de la DFO. Nous optimisons les quelques variables (en nombre fini, usuellement très faible) impactant les discontinuités par un algorithme de DFO, et en parallèle nous laissons l'optimisation de toutes les autres à n'importe quelle méthode bien choisie dans la littérature sur le contrôle optimal. Nos tests numériques montrent que cette approche hybride résout des problèmes de contrôle optimal présentant des discontinuités, et qu'elle est la seule à le faire de façon fiable.

CHAPITRE 2 ÉTAT DE L'ART

C'est une grave erreur que d'échafauder une théorie avant d'avoir rassemblé tous les matériaux nécessaires ; cela ne peut que fausser le jugement.

« Sherlock Holmes : Une étude en rouge », Sir Arthur Conan Doyle.

Le cadre d'étude de la DFO (le problème (**P**) décrit au cadre d'étude 1.1) est très large, et englobe un grand nombre de problèmes. Pour autant, la DFO ne doit jamais être la première solution envisagée pour résoudre un problème donné. Nous commençons ici par discuter de ce fait et donner d'abord quelques pistes de compréhension pour s'initier au domaine, au chapitre 2.1. Ensuite, au chapitre 2.2, nous listons plusieurs approches existantes en DFO et, pour chacune, nous montrons que le cas discontinu peut casser ses propriétés de convergence.

2.1 Présentation succincte de l'optimisation sans dérivées

Ce chapitre cerne ce que la DFO vise à traiter ou non, puis identifie deux notions importantes du domaine. Le chapitre 2.1.1 délimite les frontières du domaine et des problèmes considérés, et le chapitre 2.1.2 poursuit avec une discussion sur les différentes dénominations du domaine dans tous ses champs d'application. Ensuite, le chapitre 2.1.3 introduit les notions de *suites raffinantes* et de *cône hypertangent*, centrales dans l'analyse de plusieurs algorithmes de DFO.

2.1.1 Problèmes à ne pas résoudre par des méthodes sans dérivées

Problèmes lisses ou pouvant être lissés. Lorsqu'un problème est *lisse*, c'est-à-dire que les dérivées existent, il faut privilégier une méthode d'optimisation les exploitant. Toutefois, lorsqu'un problème est *non-lisse* (les dérivées n'existent donc pas), il est parfois *lissable*, au sens qu'il peut être altéré en un nouveau problème lisse. Si le problème lissé a ses solutions proches de celles du problème initial, alors une bonne stratégie est de résoudre le problème lissé par une méthode avec dérivées pour obtenir une quasi-solution du problème initial. Ceci arrive par exemple lorsque la fonction objectif correspond à un signal bruité et qu'on parvient à éliminer le bruit. Par conséquent, même si un problème est non-lisse, une méthode sans dérivées n'est pas conseillée si le problème peut être lissé en un problème simple et représentatif. Que les vraies dérivées existent ou non, s'il existe de bons substituts à celles-ci qui soient simples à calculer, alors il faut envisager de les donner à un algorithme pouvant les exploiter. Cette approche requiert toutefois un lissage du problème qui soit simple à résoudre et représentatif du problème initial.

Problèmes combinatoires, problèmes discrets. La formulation du problème (**P**) n'impose pas que l'espace des variables soit un continuum. Un cas simple d'espace dit *discret* est l'ensemble $\mathbb{N}$ des entiers naturels, pour contraindre les variables à être entières. Un cas plus complexe est l'espace des permutations de $[\![1, N]\!]$ (où $N \in \mathbb{N}^*$), souvent rencontré pour les problèmes de tournées de véhicules par exemple. La notion de dérivée n'a pas de sens sur un espace discret, aussi tout problème à variables discrètes est un problème sans dérivées. Malgré tout, un tel problème peut contenir énormément de structure. Ainsi, des problèmes combinatoires ou des problèmes discrets présentant une structure suffisamment simple et explicite sont à résoudre par des méthodes issues des littératures dédiées plutôt que par un algorithme de `DFO`.

Problème de grande dimension rapportée à l'horizon temporel. Les méthodes sans dérivées souffrent de la *malédiction de la dimension*, qui stipule que la performance des méthodes se dégrade lorsque la dimension du problème augmente trop. Plusieurs logiciels de `DFO` annoncent une valeur critique autour de 50 variables [19, 123]. Au-delà, la résolution par une méthode sans dérivées risque d'échouer en pratique. Lorsqu'on résout un problème réel à plus de 50 variables, l'horizon temporel peut donc être un facteur extrêmement limitant. Si d'anciennes instances ont été étudiées au préalable, il peut être plus efficace de se fier à ces informations passées pour assister un algorithme de `DFO` que d'attendre qu'il résolve le nouveau problème tout seul. Aussi, si la fonction objectif n'est pas trop coûteuse à évaluer, il peut être judicieux d'envisager une approche statistique pour inférer rapidement une solution. En d'autres termes, si l'horizon temporel est limité, il faut exploiter un maximum d'information pour tenter d'obtenir de l'information à donner à un algorithme de `DFO`. Il est également possible d'accélérer un algorithme en exploitant le parallélisme ; mais cela requiert une expertise technique car le parallélisme soulève des difficultés d'implémentation.

Synthèse des cas sur lesquels ne pas utiliser la `DFO`. Le problème (**P**) présuppose très peu de structure ; mais lorsqu'on considère une instance précise, il faut s'interroger sur ce qu'on connaît du problème. Plus un problème admet de structure explicite, plus il est plausible qu'un algorithme performant et dédié à cette structure existe dans la littérature. Nous référons au livre [86] pour une revue générale des algorithmes pour problèmes structurés et au solveur **IPOPT** [118] pour résoudre des problèmes lisses. La résolution par un algorithme de `DFO` ne devrait être considérée que si la réponse à toutes les questions suivantes est « non ».

— Le problème admet-il une structure exploitable et identifiée dans la littérature ?

— Existe-il un lissage simple et n'altérant pas significativement les solutions ?

— Le temps de calcul de la fonction objectif est-il faible, comparé à la dimension ?

2.1.2 Différentes dénominations dans différentes littératures

L'optimisation sans dérivées porte plusieurs noms et est riche de plusieurs littératures. Nous les survolons ici pour en extraire les similarités et des références classiques.

Optimisation sans dérivées. Cette terminologie communément retenue en mathématiques décrit la principale difficulté inhérente au problème (**P**) : il doit être résolu par une méthode n'exploitant pas de dérivées car la fonction objectif a un comportement imprévisible. Cette thèse est écrite sous cette dénomination, et le chapitre 2.2 discute plus en détails de l'application de cette littérature dans les cas discontinus. Les livres [22, 35] sont également écrits sous cette dénomination.

Optimisation de simulation. En ingénierie, un problème récurrent consiste à imposer un certain comportement à un système physique. Ceci se fait usuellement en proposant une modélisation mathématique de la réponse du système lorsqu'on fixe ses degrés de liberté. Au lieu de réellement étudier le système dans plusieurs configurations possibles, on laisse le modèle mathématique simuler la réponse du système. On parle alors d'*optimisation de simulation* lorsqu'on cherche les valeurs des degrés de liberté minimisant l'écart entre la réponse du système prédite par le modèle et celle désirée. Le livre [101] discute de cette problématique. Même si la terminologie diffère, ceci est de l'optimisation sans dérivées. Le modèle du système est souvent complexe et difficile à interpréter. Il est donc difficile d'expliquer comment la sortie du modèle évolue en fonction des entrées. En d'autres termes, la dérivée de la sortie par rapport aux entrées est difficile à obtenir.

Optimisation de boîte noire. Lorsqu'un processus est complexe et difficile à analyser, il est décrit comme une *boîte noire*. Dans un tel processus, seules les entrées sont bien identifiées. Il y a de l'incertitude ou de l'incompréhension dans le détail du processus et dans l'interprétation de la valeur de sortie. Par exemple, un modèle de simulation physique est souvent une boîte noire. Le système est complexe, donc la modélisation mathématique comporte des opérations intriquées rendant l'analyse difficile. Un autre exemple de boîte noire serait une expérience de laboratoire. Par extension, on parle parfois de *boîte grise* [6] lorsqu'on a de l'information claire sur une partie du processus uniquement, ou lorsque tout le processus est bien connu mais est très complexe. Les boîtes noires sont de plus en plus fréquentes en sciences [4], et l'*optimisation de boîte noire* s'est développée en réponse à cette hausse. La référence [74] propose une taxonomie des contraintes rencontrées dans les différentes littératures sur le sujet, la revue de littérature [73] fait un bon survol des méthodes existantes,

et le livre [17] fait le lien entre l'optimisation sans dérivées et l'optimisation de boîte noire. Il s'agit du même problème en théorie, mais en pratique il faut voir l'optimisation de boîte noire comme un cas particulier notable. Le temps d'exécution d'un processus boîte noire est souvent long (pouvant aller jusqu'à plusieurs semaines dans des problèmes industriels), pour n'importe quelle entrée. Ainsi, une méthode d'optimisation de boîte noire doit éviter au maximum d'exécuter le processus.

Optimisation difficile. Un certain nombre de chercheurs en optimisation développent des techniques nommées *métaheuristiques*. Une méthode est *heuristique* lorsqu'elle n'a pas de garantie théorique de générer un point satisfaisant certaines conditions d'optimalité pour le problème d'optimisation traité. Une *métaheuristique* est une heuristique dont certaines étapes ne sont pas explicites, mais décrites en terme de grandes lignes directrices à suivre. Lorsqu'un utilisateur souhaite résoudre un problème donné, il peut définir ces étapes comme il le souhaite. La méthode résultante est alors une heuristique propre au problème, mais également une instance de la métaheuristique. Elle peut donc être performante si la métaheuristique est bien conçue. Le livre [41] discute en détail de plusieurs métaheuristiques. La référence [108] propose une taxonomie des méthodes et y incorpore des techniques issues de l'*apprentissage machine*. La littérature étudiant les métaheuristiques parle de problèmes *difficiles* pour signifier qu'il n'y a pas de méthode de résolution simple. Il s'agit, très souvent, de problèmes combinatoires. Certaines métaheuristiques sont utilisées en `DFO`, mais les deux littératures traitent rarement des problèmes similaires.

2.1.3 Notions d'analyse d'algorithmes sans dérivées

Les notions de *suite raffinante* [17, définition 8.4], [13, définition 3.2] et de *cône hypertangent à* Ω *en* $x \in \Omega$ [17, définition 6.7], rappelées ci-dessous, sont centrales en `DFO`.

Définition 2.1 (Suite raffinante). Soit un algorithme `A` résolvant le problème (**P**) et générant une suite $(x^k; \delta^k; \mathcal{D}^k)_{k\in\mathbb{N}}$; où pour tout $k \in \mathbb{N}$, $x^k \in \mathbb{R}^n$ est le meilleur point connu au début de l'itération k, $\delta^k \in \mathbb{R}_+^*$ est le pas de recherche à l'itération k et $\mathcal{D}^k \subset \mathbb{R}^n$ est un ensemble de directions de recherche. Une *suite raffinante* $(x^k; \delta^k; \mathcal{D}^k)_{k\in K^*}$, indexée par $K^* \subseteq \mathbb{N}$, est alors telle que $(x^k)_{k\in K^*}$ converge vers un *point raffiné* x^*, et l'itération k est un échec ($x^{k+1} = x^k$) pour tout $k \in K^*$, et $(\delta^k)_{k\in K^*} \to 0$. L'ensemble D^* des *directions raffinées* est alors l'ensemble des directions $d \in \mathbb{S}^n$ telles qu'il existe une suite $(\frac{d^k}{\|d^k\|})_{k\in K'} \to d$, où $K' \subseteq K^*$ et $d^k \in \mathcal{D}^k$ pour tout $k \in K'$. On note $\mathbb{G}_{\mathtt{A}}^*$ l'ensemble des éléments $(K^*; x^*; D^*)$ où K^* indexe une suite raffinante générée par `A`.

Beaucoup d'algorithmes génèrent au moins une suite raffinante dans le cadre d'étude 1.1.

Définition 2.2 (Cône hypertangent). Le cône hypertangent à Ω en $x \in \Omega$ est défni comme

$$\mathcal{H}(x) \quad \triangleq \quad \{d \in \mathbb{S}^n : \quad \exists \varepsilon > 0,\ \forall (y,u,t) \in (\mathcal{B}_\varepsilon(x) \cap \Omega) \times \mathcal{B}_\varepsilon(d) \times [0,\varepsilon[, \quad y + tu \in \Omega\}\,.$$

Les résultats de convergence qui suivent dans ce chapitre sont donnés en terme de directions raffinées appartenant au cône hypertangent à Ω au point raffiné.

2.2 Invalidité des algorithmes sans dérivées dans des cas discontinus

Dans ce chapitre, nous passons en revue les trois principales classes de méthodes de la littérature en `DFO`, afin de montrer que des discontinuités peuvent les mettre en défaut. La première classe est celle des heuristiques, discutée au chapitre 2.2.1. Il s'agit de méthodes simples à utiliser mais n'ayant pas de garantie théorique. La seconde classe est celle des méthodes fondées sur des modèles, présentée au chapitre 2.2.2. Ces méthodes construisent itérativement un modèle approximant la fonction objectif à partir de l'historique des points déjà évalués. Enfin, la troisième classe est celle des méthodes de recherche directe, introduite au chapitre 2.2.3. L'objectif de ces méthodes est de minimiser la fonction objectif sans consommer de budget d'évaluations dans le but de construire un modèle de la fonction ou de ses dérivées. Les prochains chapitres de cette thèse se concentrent sur les méthodes directes.

2.2.1 Méthodes heuristiques

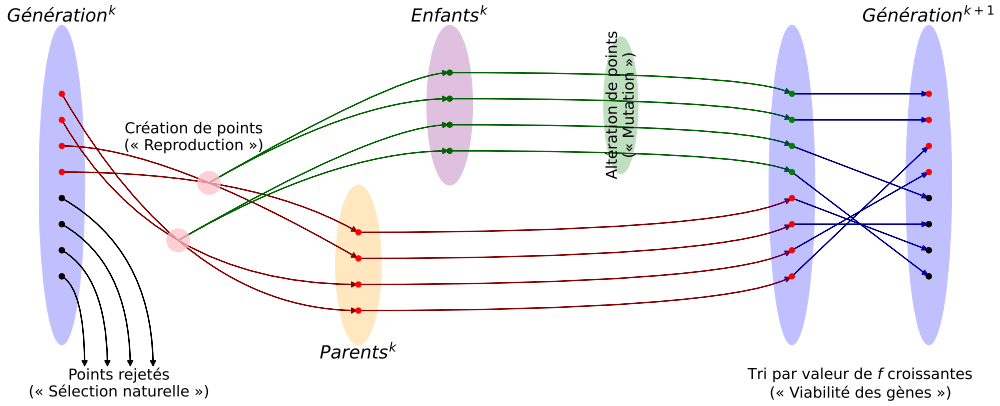
Malgré leur manque de garanties théoriques, les heuristiques sont populaires car elles sont simples et peuvent tout de même être efficaces en pratique. En `DFO`, il existe deux classes principales d'heuristiques : les métaheuristiques et l'algorithme Nelder-Mead.

Métaheuristiques

Les *métaheuristiques* ont déjà été introduites au chapitre 2.1.2. L'objectif ici est de montrer qu'elles ne devraient pas être privilégiées pour résoudre des problèmes d'optimisation sans dérivées. On se concentre pour cela sur l'*algorithme génétique* (*Genetic Algorithm*, `GA` [65]), l'une des métaheuristiques les plus populaires et grandement étudiée [21, 70]. Cet algorithme, illustré par la figure 2.1, fonctionne par analogie avec la théorie de l'évolution de Darwin.

L'efficacité de `GA` dépend de la bonne conception des étapes de « reproduction » et de « mutation ». Elles doivent permettre l'exploration de nouveaux points et l'amélioration des points actuels. Or, pour ce faire il faut de l'information sur le problème. Donc pour construire un bon `GA`, il faut exploiter de l'information qui aurait peut-être pu servir à la

FIGURE 2.1 Schéma d'un algorithme génétique.



place à concevoir un algorithme avec de meilleures garanties. Il existe des variantes de `GA` ayant des garanties de convergence [37, 63], mais ces variantes perdent la simplicité qui représente l'attrait principal de `GA` et s'inspirent fortement d'algorithmes disposant eux-mêmes de propriétés de convergence. Il y a donc une redondance avec les algorithmes originels.

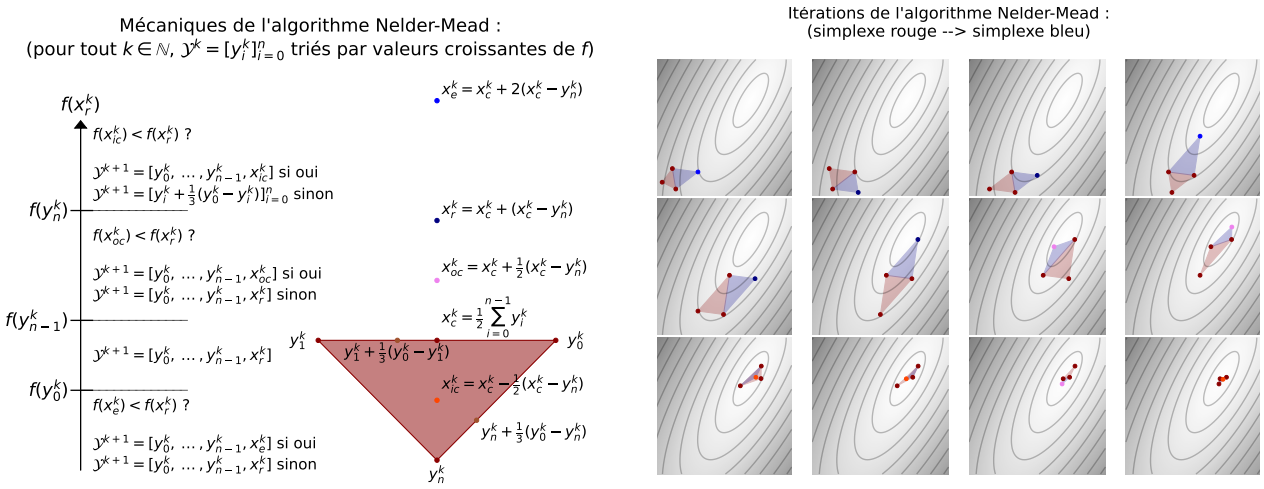
Le choix final d'utiliser `GA` ou non est laissé à l'utilisateur ; mais il faut avoir conscience qu'à niveau d'information égal, un algorithme doté de garanties de convergence peut généralement être rendu au moins aussi performant. Pour cette raison, un nombre croissant de chercheurs s'accordent à dire que `GA` est rarement le meilleur choix pour résoudre un problème d'optimisation sans dérivées [2, 120]. Ce défaut affecte également une grande partie des autres métaheuristiques. Ceci est d'autant plus vrai dans le cas discontinu, car révéler les discontinuités requiert une conception particulière pour laquelle les étapes libres des métaheuristiques ne sont pas pensées. Il faudrait par exemple conserver un historique de tous les anciens points évalués et l'enrichir autour de certains points qu'on identifie comme importants, ce qui va à l'encontre de la volonté de simplicité des métaheuristiques.

Notons toutefois que les métaheuristiques peuvent servir en soutien de méthodes sans dérivées [16]. Les métaheuristiques manquent souvent de performance en *intensification* (la capacité à améliorer un point donné en l'altérant légèrement) mais sont généralement efficaces en *exploration* (l'aptitude à découvrir des ensembles de points prometteurs encore inexplorés par la méthode). Ainsi, une méthode sans dérivées pensée pour l'intensification peut être enrichie par l'ajout d'une métaheuristique exploratoire. Par exemple, à chaque itération d'une méthode sans dérivées, on peut en complément effectuer une itération de l'algorithme `Vns` (« Variable Neighborhood Search » [59, 81]), comme fait dans [9]. `Vns` est peu coûteux et pensé pour l'exploration, et contrairement à `GA`, les étapes laissées libres dans sa définition peuvent être définies efficacement sans information sur le problème.

Algorithme de Nelder-Mead

L'algorithme de Nelder-Mead (NM) [85] est une autre heuristique très populaire en DFO. Ceci s'explique par sa grande simplicité, autant dans la compréhension de la méthode que dans son implémentation. De plus, malgré cette simplicité, NM est souvent performant sur des problèmes non-lisses assez simples. La référence [121] propose un bon résumé de l'historique de NM, de sa nature heuristique mise en évidence par des contre-exemples [80], et de ses applications à succès dans de nombreux cas pratiques. NM a également été réétudié pour lui enlever sa nature heuristique [20, 44, 51]. L'algorithme consiste en la définition d'un *simplexe*[1] qui est déformé itérativement, par extensions ou contractions en fonction du profil de la fonction objectif donné par ses valeurs en les points du simplexe. La figure 2.2 détaille le fonctionnement de l'algorithme et en illustre le comportement sur un exemple simple.

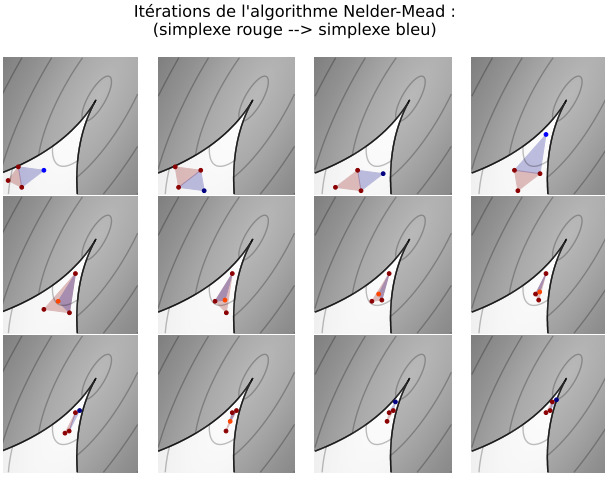
FIGURE 2.2 Détail et illustration de NM.



Dans les cas discontinus, la performance de NM peut se dégrader. Ce cas n'est pas analysé dans la littérature, mais la figure 2.3 met en évidence le principal problème en altérant l'exemple de droite de la figure 2.2. Lorsque le simplexe approche une solution proche d'une discontinuité, cette dernière agit comme une contrainte. NM n'accepte que des points améliorant la valeur de la fonction objectif et ne dispose d'aucune stratégie pour détecter la discontinuité. Le seul moyen de converger est alors de le faire sans franchir la discontinuité, ce qui peut être difficile si le simplexe est mal positionné. Sur la figure, la quatrième itération est un grand succès ; mais à partir de la cinquième itération, NM passe la majeur partie de ses itérations à générer des points juste au-delà de la discontinuité. Pour avancer, NM doit rendre le simplexe presque plat, et presque perpendiculaire à la direction à suivre, pour converger.

1. Un simplexe dans $\mathbb{R}^n$ consiste en $n+1$ points dont l'enveloppe convexe est d'intérieur non vide. Un simplexe est à l'espace ce qu'un triangle non aplati est au plan.

FIGURE 2.3 Manque d'efficacité de NM à l'approche d'une discontinuité.



2.2.2 Méthodes fondées sur les modèles

L'une des hypothèses les plus discutables en optimisation sans dérivées est de supposer que la fonction objectif est une boîte noire, sans aucune structure. Faisons l'analogie avec un signal bruité : le signal semble très erratique localement à cause du bruit, mais la tendance devient observable en analysant le signal globalement. De même, il est commun que la fonction objectif présente une tendance générale malgré des difficultés locales. L'objectif des méthodes présentées ici est de capter cette tendance. Nous montrons que même si ces méthodes ont des garanties de convergence et sont efficaces dans les cas continus ; elles sont aisément mises en échec sur des cas discontinus. Nous pouvons conjecturer qu'à moins de retravailler significativement les méthodes, il faudrait probablement connaître explicitement les points de discontinuité pour s'affranchir de l'hypothèse de continuité. Infirmer cette conjecture pourrait être une avancée notable dans la gestion des discontinuités en DFO.

Régions de confiance

Les méthodes de *régions de confiance* (*Trust-Region*, Tr [34, 122], [17, chapitre 11], [35, chapitre 10]) sont très étudiées dans de nombreux domaines de l'optimisation. Notamment, dans les cas non-lisses [5, 12, 54, 78, 91]. L'étude des Tr est trop riche pour être discutée de manière exhaustive. De plus, l'algorithme complet décrivant ces méthodes est technique [52, algorithme 4.1], [17, algorithme 11.1] et une bonne construction de certaines de ces étapes fait l'objet de travaux entiers [84, 93, 115]. L'algorithme Tr n'est pas donné dans cette thèse, mais la figure 2.4 en illustre le fonctionnement. Le comportement de la Tr [52, algorithme 4.1] lorsque la fonction objectif est non-lisse est connu [52], et peut être résumé comme suit.

Théorème 2.1. **[52, Théorème 4.3].** Si $f_{|X}$ vérifie [52, hypothèses 4.1, 4.2, 4.3], alors $f_{\mathtt{C}}^{\circ}(x^*; d^*) \geq 0$ pour tout $(x^*; d^*) \in X \times (D^* \cap \mathcal{H}(x^*))$ tels que $(K^*; x^*; D^*) \in \mathbb{G}^*_{\mathtt{Tr}}$.

Les méthodes de régions de confiance construisent des modèles locaux. Le comportement de f est étudié autour d'un point, sur un voisinage assez large pour capter une tendance. À chaque itération $k \in \mathbb{N}$, étant donné $\mathcal{T}_{\mathtt{S}}^k \subset \mathbb{R}^n$ un ensemble de points d'échantillonnage passés, on construit $\widetilde{f}^k$ à partir de $\mathcal{T}_{\mathtt{S}}^k \subset \mathbb{R}^n$ comme un modèle interpolant f en des points de $\mathcal{T}_{\mathtt{S}}^k$ bien choisis proches de $x_r^k \in \operatorname{argmin} f(\mathcal{T}_{\mathtt{S}}^k)$. On se fie alors au modèle $\widetilde{f}^k$ sur l'ensemble de la boule fermée $\operatorname{cl}(\mathcal{B}_{\delta^k}(x_r^k))$, où $\delta^k > 0$, pour calculer minimiseur de $\widetilde{f}^k$ sur $\operatorname{cl}(\mathcal{B}_{\delta^k}(x_r^k))$ noté $t^k \in \operatorname{argmin} \widetilde{f}^k(\operatorname{cl}(\mathcal{B}_{\delta^k}(x_r^k)))$. On définit alors $\mathcal{T}_{\mathtt{S}}^{k+1} \triangleq \mathcal{T}_{\mathtt{S}}^k \cup \{t^k\}$. Selon la concordance entre $\widetilde{f}^k(t^k)$ et $f(t^k)$, on peut soit conclure que le modèle est localement pertinent et alors autoriser $\delta^{k+1} \geq \delta^k$, soit remarquer que le modèle est erroné et alors forcer $\delta^{k+1} < \delta^k$. Ces méthodes privilégient l'intensification à l'exploration.

Un cas particulier notable des `Tr` est l'ensemble des méthodes dites *de Powell*. La thèse [92] en offre un bon survol et le solveur **PRIMA** [123] les implémente. Les méthodes de Powell construisent différents modèles (linéaires, quadratiques, ou pseudo-quadratiques) et traitent différents types de contraintes. L'analyse de convergence des méthodes reste à faire lorsque f n'est pas lisse, mais [92, chapitre 8.2.1] conjecture que cette analyse sera similaire à celles des méthodes originelles dont les méthodes de Powell s'inspirent.

Enfin, le potentiel des `Tr` dans un cas discontinu n'est pas clair. La figure 2.5 illustre comment une `Tr` simple peut être induite en erreur par une discontinuité. Il est probablement possible de les adapter, mais l'adaptation ne semble pas directe. Une autre approche serait de supposer que les points de discontinuité sont connus, car on pourrait alors adapter les modèles pour tenir compte d'éventuelles discontinuités entre les points d'interpolation.

FIGURE 2.4 Illustration d'une `Tr`.

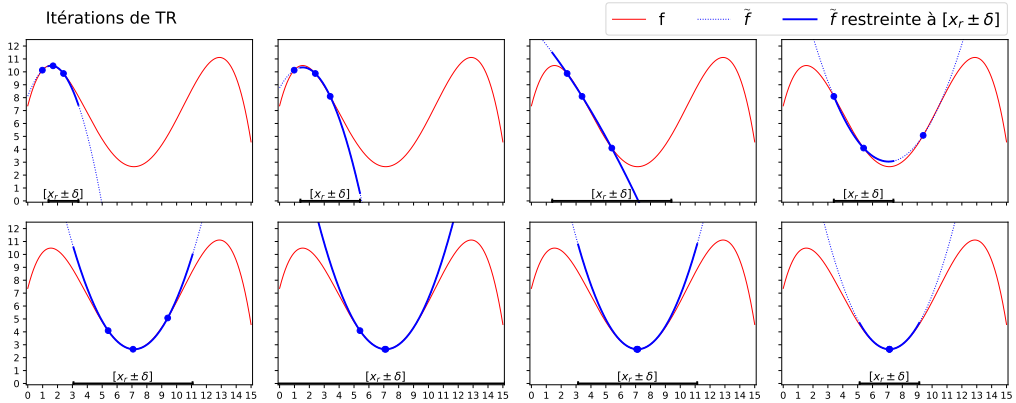
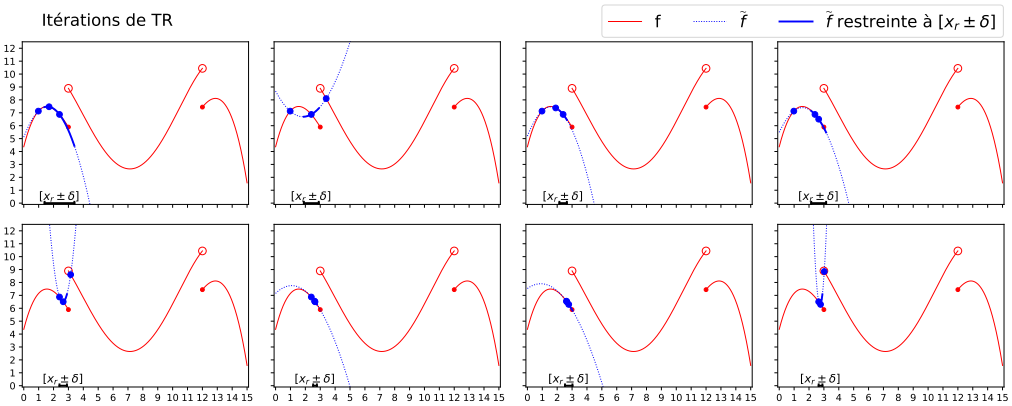


FIGURE 2.5 Manque d'efficacité d'une `Tr` sur un cas discontinu.



Algorithme bayésien : EGO, extensions, et autres algorithmes

Les méthodes bayésiennes [29, 50, 53] développent itérativement une *surface de réponse* $\widetilde{f}^k$ captant la forme globale de la fonction objectif f à partir d'un ensemble $\mathcal{T}_{\mathtt{S}}^k \subset \mathbb{R}^n$ de points d'échantillonnage. La surface de réponse interpole f sur $\mathcal{T}_{\mathtt{S}}^k$, et extrapole f sur $\mathbb{R}^n \setminus \mathcal{T}_{\mathtt{S}}^k$. Une estimation de la variance statistique $\sigma^k(x)$ associée à l'extrapolation $\widetilde{f}^k(x)$ est également disponible pour tout $x \in \mathbb{R}^n$. Cette variance est en général estimée en tout $x \in \mathbb{R}^n$ via la variabilité locale de la surface de réponse et via la distance aux points d'interpolation. À chaque itération $k \in \mathbb{N}$, un nouveau point d'échantillonnage est choisi suivant une *fonction d'acquisition* φ^k qui encode un compromis entre garantie d'intensification (donnée par $\widetilde{f}^k$) et potentiel exploratoire (décrit par σ^k). Les méthodes bayésiennes sont donc pensées pour l'exploration, autant voire plus que pour l'intensification.

Une méthode bayésienne très populaire est l'*Efficient Global Optimization* (`Ego` [69]). Cet algorithme calcule $\widetilde{f}$ par *krigeage* [72, 106, 119] et définit φ par un *expected improvement* [69]. La figure 2.6 illustre comment le krigeage réagit à l'ajout de points d'échantillonnage, et comment la variance associée traduit de l'incertitude loin de ces points. La figure de droite illustre la façon dont `Ego` exploite le krigeage. La fonction d'acquisition estime que les points d'intéret sont ceux en lesquels σ est grande et $\widetilde{f}$ pas trop élevée, et ceux en lesquels σ est très faible mais $\widetilde{f}$ est basse. Il n'y a pas de preuve de convergence pour `Ego`, mais [69] conjecture qu'`Ego` garantit à minima que l'algorithme explore tout l'espace.

Conjecture 2.2. **[69, Conjecture 1].** Si X est compact et $f_{|X}$ est continue, alors `Ego` génère un ensemble de points d'échantillonnage dense dans X.

Une extension d'`Ego` nommée *Trust-Region* ***Ego*** (`TrEgo`) est proposée en [39]. Celle-ci ajoute à `Ego` des éléments issus des régions de confiance pour renforcer l'analyse.

Théorème 2.3. **[39, Théorème 4.3].** Si f est Lipschitz-continue, alors $f^{\circ}_{\mathtt{C}}(x^*; d^*) \geq 0$ pour tout $(x^*; d^*) \in X \times (D^* \cap \mathcal{H}(x^*))$ tels que $(K^*; x^*; D^*) \in \mathbb{G}^*_{\mathtt{TrEgo}}$.

Les méthodes fondées sur `Ego` peuvent cependant être mises en échec lorsque f n'est pas continue, car l'hypothèse de continuité est au coeur de l'idée d'extrapolation par distance aux points d'échantillonnage. La figure 2.7 montre comment `Ego` peut persister dans une extrapolation de f peu représentative à proximité des points de discontinuité.

FIGURE 2.6 Illustration du krigeage et d'`Ego`.

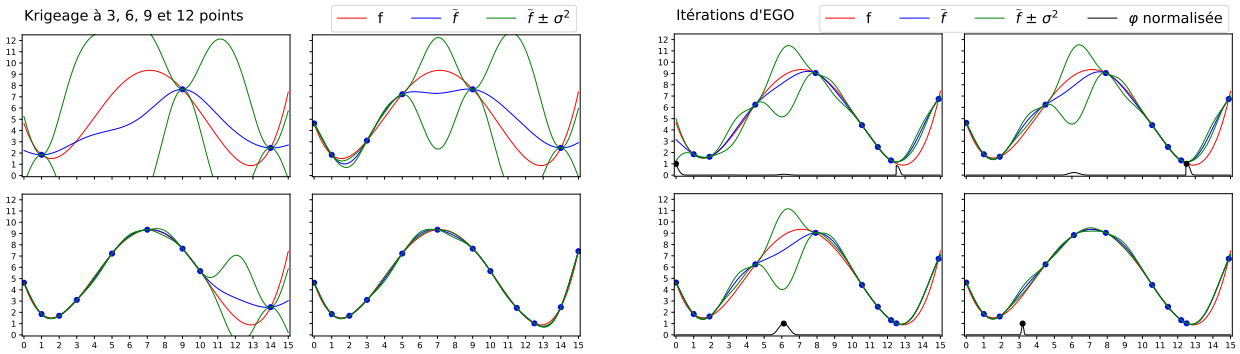
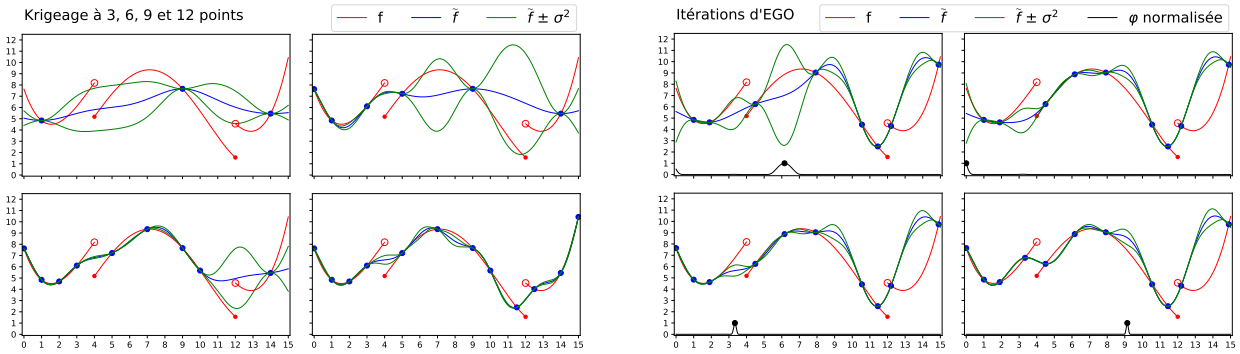


FIGURE 2.7 Manque d'efficacité du krigeage et d'`Ego` sur un cas discontinu.



De nombreux autres algorithmes bayésiens existent, fondés sur d'autres choix pour φ [75], mais souffrent du même défaut. Quelques références traitent des extensions à des fonctions mal structurées, comme [110] qui propose un moyen de détecter des *contraintes cachées* (des points hors de X, donc en lesquels f renvoie $+\infty$). Des algorithmes bayésiens certifiant la détection d'une solution globale existent également [43].

Gradients simplexes, Hessiens simplexes, recherches linéaires

En optimisation lisse, les méthodes de *descente de gradient* et de *descente hessienne* sont très efficaces [100]. Dans le cas non-lisse, des équivalents existent en substituant le gradient et le Hessien par respectivement un *gradient simplexe* [96] et un *Hessien simplexe* [61]. Certaines références parlent également d'*oracle d'ordre zéro* [42]. Les méthodes résultantes [17, chapitre 10] sont des méthodes d'intensification efficaces, mais pas pensées pour l'exploration.

Un gradient simplexe de f en $x \in \mathbb{R}^n$ s'obtient via une matrice $\mathcal{D} \triangleq \begin{bmatrix} d_1 \dots d_n \end{bmatrix}$ inversible, et se construit comme

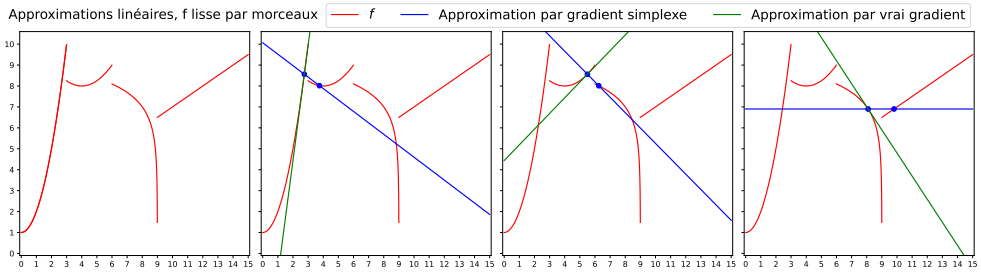
$$\nabla_{\mathsf{s}} f(x; \mathcal{D}) \triangleq \mathcal{D}^{-\top} \delta_{\mathsf{s}}(x; \mathcal{D}), \quad \text{où} \quad \delta_{\mathsf{s}}(x; \mathcal{D}) \triangleq \begin{bmatrix} f(x + d_1) - f(x) \\ \vdots \\ f(x + d_n) - f(x) \end{bmatrix}.$$

Le hessien simplexe suit une logique de construction similaire, qu'on ne discute pas ici. L'algorithme de descente de gradient simplexe, et le résultat de convergence associé, sont alors similaires à une descente de gradient dans le cas lisse : au point raffiné, toutes les directions hypertangentes ont un produit scalaire positif avec le gradient.

Théorème 2.4. **[17, Théorème 10.6].** Si f est dérivable et que les gradients simplexes suivent les hypothèses techniques de [17, théorème 10.6], alors l'algorithme de descente de gradient simplexe génère un point x^* vérifiant $(\nabla f(x^*) \mid d) \geq 0$ pour tout $d \in \mathcal{H}(x^*)$.

La figure 2.8 montre qu'une discontinuité de f peut mener à un gradient simplexe peu pertinent. En bien ou en mal selon les cas, les discontinuités faussent l'estimation d'une direction de décroissance. Ce phénomène n'est pas propre aux fonctions discontinues, mais les discontinuités peuvent l'amplifier. Pour la même raison, un algorithme de recherche linéaire (*line search* [26, 55, 56]) plus avancé peut aussi être mis en défaut par des discontinuités.

FIGURE 2.8 Manque de pertinence d'un gradient simplexe sur un cas discontinu.



2.2.3 Méthodes de recherche directe (DSM)

Lorsque la fonction objectif est très coûteuse à évaluer, il n'est pas toujours conseillé de calculer des modèles ou des estimations de gradients. Estimer un gradient par différences finies requiert $n+1$ évaluations de la fonction objectif, où n est la dimension du problème, et ne donne une information valide que localement. Partant du constat que le coût d'évaluation est élevé en situations réelles, la *méthode de recherche directe* (*Direct Search Methods*, notée DSM, [17, partie 3] et [35, chapitre 7]) s'est développée avec l'objectif d'économiser un maximum d'évaluations de la fonction objectif. La méthode de recherche directe n'évalue jamais la fonction objectif dans le but d'en concevoir un modèle.

Cette méthode est itérative, et chaque itération est usuellement constituée de deux étapes. La principale est l'étape de *sonde* (en anglais *poll*, notée `poll`), qui constitue le coeur de l'analyse théorique. La seconde est l'étape d'*exploration* (en anglais *search*, notée `search`), introduite pour donner de la flexibilité aux méthodes. La `poll` est une étape assez rigide, alors que la `search` peut contenir n'importe quelle méthode laissée au choix des usagers. L'état de l'art a pendant un temps accepté la croyance que DSM traite nativement les discontinuités, mais nous revenons sur cette croyance à la fin de chapitre.

Recherche par coordonnées et Recherche par motifs

Le précurseur de DSM est l'algorithme *Coordinate Search* (Cs, [46]). À chaque itération $k \in \mathbb{N}$, Cs évalue les $2n$ points voisins du meilleur point actuel, $x^k \in \Omega$, suivant les directions canoniques et un pas $\delta^k \in \mathbb{R}_+^*$. L'ensemble des points de `poll` est donc $\mathcal{T}_{\mathtt{P}}^k \triangleq \{x^k\} + \delta^k \mathbb{D}_{\mathtt{Cs}}$, où $\mathbb{D}_{\mathtt{Cs}} \triangleq \{\pm e_i : e_i \text{ dans la base canonique de } \mathbb{R}^n\}$. Si $\mathcal{T}_{\mathtt{P}}^k \cap \Omega \neq \emptyset$ et si $t^k \in \operatorname{argmin} f(\mathcal{T}_{\mathtt{P}}^k \cap \Omega)$ vérifie $f(t^k) < f(x^k)$, alors Cs définit $x^{k+1} \triangleq t^k$ et $\delta^{k+1} \triangleq \delta^k$. Autrement, Cs définit $x^{k+1} \triangleq x^k$ et $\delta^{k+1} \triangleq \frac{1}{2}\delta^k$. On a alors le résultat suivant.

Théorème 2.5. **[17, Théorème 3.4].** Si f est Lipschitz-continue, alors $f_{\mathtt{C}}^\circ(x^*; d^*) \geq 0$ pour tout $(x^*; d^*) \in X \times (D^* \cap \mathcal{H}(x^*))$ tels que $(K^*; x^*; D^*) \in \mathbb{G}_{\mathtt{Cs}}^*$.

Cs est cependant peu satisfaisant, pour deux raisons. D'abord, il force la suite $(\delta^k)_{k\in\mathbb{N}}$ à être décroissante, ce qui inhibe tout potentiel exploratoire efficace. Ensuite, et surtout, il limite ses recherches aux directions canoniques. Il vient que l'ensemble des directions raffinées potentielles est restreint à $D^* = \mathbb{D}_{\mathtt{Cs}}$. Une extension importante de Cs corrigeant ces deux défauts est l'algorithme *Generalized Pattern Search* (Gps, [111]). Gps choisit l'ensemble de directions à considérer à chaque itération parmi une liste prédéfinie, aussi riche que désiré, et augmente le pas de recherche en cas de succès au cours d'une itération. En d'autres termes,

`Gps` est paramétré par une liste $\{\mathcal{D}_i\}_{i=0}^{N-1}$ de $N \in \mathbb{N}^*$ sous-ensembles finis de $\mathbb{R}^n$, et construit l'ensemble de `poll` à chaque itération $k \in \mathbb{N}$ comme $\mathcal{T}_{\mathtt{P}}^k \triangleq \{x^k\} + \delta^k \mathcal{D}_{(k \text{ modulo } N)}$. De là, si $\mathcal{T}_{\mathtt{P}}^k \cap \Omega \neq \emptyset$ et si $t^k \in \operatorname{argmin} f(\mathcal{T}_{\mathtt{P}}^k \cap \Omega)$ vérifie $f(t^k) < f(x^k)$, alors `Gps` définit $x^{k+1} \triangleq t^k$ et $\delta^{k+1} \triangleq 2\delta^k$. Autrement, `Gps` définit $x^{k+1} \triangleq x^k$ et $\delta^{k+1} \triangleq \frac{1}{2}\delta^k$. L'ensemble des directions raffinées potentielles est donc $\mathbb{D}_{\mathtt{Gps}} \triangleq \{\frac{d}{\|d\|} : d \in \cup_{i=1}^N \mathcal{D}_i\}$.

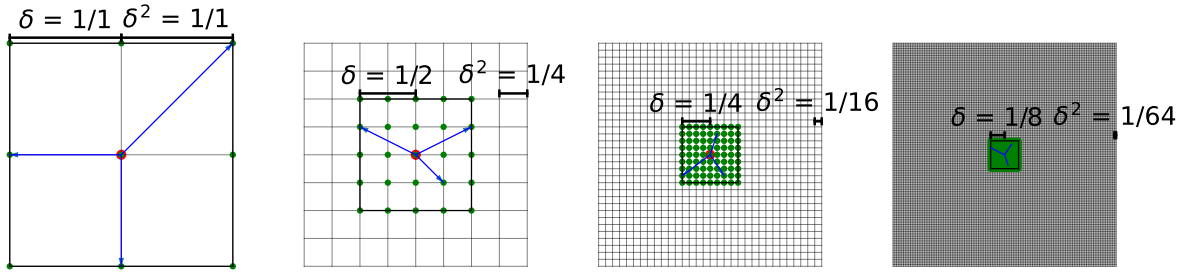
`Gps` a une propriété de convergence similaire à `Cs`, mais qui porte sur un ensemble de directions plus riche que dans le résultat lié à `Cs`. L'ensemble D^* des directions raffinées est au moins la normalisation de l'un des $(\mathcal{D}_i)_{i=1}^N$, et au plus $\mathbb{D}_{\mathtt{Gps}}$. Toutefois, `Gps` a le défaut de rester dépendant de l'ensemble $\cup_{i=1}^N \mathcal{D}_i$, qui est choisi à priori et limite donc la richesse de D^*.

Théorème 2.6. **[17, Théorème 7.7].** Si f est Lipschitz-continue, alors $f_{\mathtt{C}}^\circ(x^*; d^*) \geq 0$ pour tout $(x^*; d^*) \in X \times (D^* \cap \mathcal{H}(x^*))$ tels que $(K^*; x^*; D^*) \in \mathbb{G}_{\mathtt{Gps}}^*$.

Algorithme `Mads`

L'algorithme *Mesh Adaptive Direct Search* (`Mads` [14]) corrige les défauts de `Gps`. `Mads` construit automatiquement l'ensemble de directions de `poll` à chaque itération. L'algorithme n'est donc pas dépendant d'un choix fait à priori. Pour ce faire, `Mads` choisit les directions de `poll` sur un *treillis adaptatif*, comme illustré en figure 2.9.

FIGURE 2.9 Ensembles de directions admissibles sur le treillis adaptatif de `Mads`.



Le treillis adaptatif est une fonction $\mathcal{M} : \mathbb{R}_+^* \to 2^{\mathbb{R}^n}$ donnant des discrétisations de $\mathbb{R}^n$. La figure 2.9 illustre le treillis $\mathcal{M} : \delta \mapsto \min(\delta, \delta^2)\mathbb{Z}^2$. On a également représenté le *cadre de sonde* paramétré par δ, défini par $\mathrm{cl}(\mathcal{B}_\delta)$ en norme infinie, et on note $\mathcal{M}_\nu(\delta) \triangleq \mathcal{M}(\delta) \cap \mathrm{cl}(\mathcal{B}_\nu)$ pour tout $(\delta; \nu) \in (\mathbb{R}_+^*)^2$. À δ donné, l'algorithme `Mads` originel considère l'ensemble des directions admissibles $\mathcal{M}_\delta(\delta)$.[2] Ainsi, le nombre de directions potentielles croît lorsque δ décroît, et `Mads` peut considérer un ensemble de directions devenant dense dans $\mathbb{S}^n$ après normalisation. L'ensemble des directions raffinées potentielles est donc $\mathbb{D}_{\mathtt{Mads}} \triangleq \mathbb{S}^n$.

2. Nous choisissons ici de prendre le treillis dans l'espace des directions et pas dans l'espace des variables. Ceci diffère de la littérature, mais simplifie notre présentation et n'altère pas l'analyse.

`Mads` autorise aussi une étape de `search`. Cette étape est optionnelle à chaque itération, mais elle permet de tester d'autres stratégies (par exemple, heuristiques) en parallèle. La seule condition est que l'ensemble de directions considéré soit $\mathcal{D}_{\mathtt{S}}^k \subset \mathcal{M}(\delta^k)$.

Une version simple de `Mads` est alors décrite comme suit. Initialiser $x^0 \in X \cap \Omega$ et $\delta^0 = 1$. Pour tout $k \in \mathbb{N}$, définir $\mathcal{D}_{\mathtt{S}}^k \subset \mathcal{M}(\delta^k)$ (potentiellement vide) et $\mathcal{D}_{\mathtt{P}}^k \subset \mathcal{M}(\delta^k) \cap \mathrm{cl}(\mathcal{B}_{\delta^k})$ (non vide). Définir $\mathcal{T}^k \triangleq \{x^k\} + (\mathcal{D}_{\mathtt{S}}^k \cup \mathcal{D}_{\mathtt{P}}^k)$, et $t^k \in \operatorname{argmin} f(\mathcal{T}^k)$. Si $f(t^k) < f(x^k)$, définir $x^{k+1} \triangleq t^k$ et $\delta^{k+1} \triangleq 2\delta^k$, sinon, définir $x^{k+1} \triangleq x^k$ et $\delta^{k+1} \triangleq \frac{1}{2}\delta^k$. Une version avancée et performante de `Mads` est implémentée dans le solveur **NOMAD** [19].

`Mads` admet le résultat de convergence suivant. La formulation est proche des résultats de `Cs` et `Gps`, mais il est possible de se persuader que D^* peut égaler $\mathbb{D}_{\mathtt{Mads}} = \mathbb{S}^n$ avec une bonne construction de $(\mathcal{D}^k)_{k\in\mathbb{N}}$. Divers schémas existent, notamment celui d'`OrthoMads` [1]. Ainsi, `Mads` garantit des conditions nécessaires d'optimalité dans toutes les directions.

Théorème 2.7. **[17, Théorème 7.7].** Si f est Lipschitz-continue, alors $f_{\mathtt{C}}^\circ(x^*; d^*) \geq 0$ pour tout $(x^*; d^*) \in X \times (D^* \cap \mathcal{H}(x^*))$ tels que $(K^*; x^*; D^*) \in \mathbb{G}_{\mathtt{Mads}}^*$.

Algorithme de décroissance suffisante

Une alternative à `Mads` permet de retrouver un théorème similaire au théorème 2.7 sans discrétiser l'espace des directions. L'algorithme de *décroissance suffisante* (`SuDe`, [35, chapitre 7.7]) ne repose pas sur un treillis adaptatif pour choisir l'ensemble des directions de `poll`. Pour un pas de sonde $\delta \in \mathbb{R}_+^*$ donné, les directions sont tirées comme $\mathcal{D}_{\mathtt{P}}^k \subset \mathrm{cl}(\mathcal{B}_\delta)$. Une nouvelle contrainte est qu'un point candidat n'est considéré comme meilleur que la solution courante que lorsque la décroissance sur f est suffisamment importante. `SuDe` requiert une *fonction de forçage* $\rho : \mathbb{R}_+^* \to \mathbb{R}_+^*$ croissante et telle que $\rho(\nu) \in o(\nu)$ lorsque $\nu \searrow 0$.[3]

À toute itération $k \in \mathbb{N}$, `SuDe` considère $\mathcal{D}_{\mathtt{S}}^k \subset \mathbb{R}^n$ (potentiellement vide) et $\mathcal{D}_{\mathtt{P}}^k \subset \mathrm{cl}(\mathcal{B}_{\delta^k})$ (non vide), et définit $\mathcal{T}^k \triangleq \{x^k\} + (\mathcal{D}_{\mathtt{S}}^k \cup \mathcal{D}_{\mathtt{P}}^k)$. Si $\mathcal{T}^k \cap \Omega \neq \emptyset$, `SuDe` note $t^k \in \operatorname{argmin} f(\mathcal{T}^k \cap \Omega)$, puis n'autorise $x^{k+1} \triangleq t^k$ et $\delta^{k+1} \triangleq 2\delta^k$ que si $f(t^k) \leq f(x^k) - \rho(\delta^k)$; et autrement il pose $x^{k+1} \triangleq x^k$ et $\delta^{k+1} \triangleq \frac{1}{2}\delta^k$. L'ensemble des directions raffinées potentielles est $\mathbb{D}_{\mathtt{SuDe}} \triangleq \mathbb{S}^n$.

Cet algorithme dispose d'une garantie de convergence similaire à celle de `Mads`. De plus, tout comme `Mads`, `SuDe` peut générer des suites raffinantes telles que $D^* = \mathbb{D}_{\mathtt{SuDe}} = \mathbb{S}^n$.

Théorème 2.8. **[35, th. 7.5 et 7.12].** Si f est Lipschitz-continue, alors $f_{\mathtt{C}}^\circ(x^*; d^*) \geq 0$ pour tout $(x^*; d^*) \in X \times (D^* \cap \mathcal{H}(x^*))$ tels que $(K^*; x^*; D^*) \in \mathbb{G}_{\mathtt{SuDe}}^*$.

3. La condition $\rho(\nu) \in o(\nu)$ lorsque $\nu \searrow 0$ n'intervient dans la littérature étudiant `SuDe`, que dans l'étude du cas continûment différentiable. Aussi, par la suite nous imposons à la place $\lim_{\nu\searrow 0} \rho(\nu) = 0$.

Expression de `Mads` et de `SuDe` comme un unique algorithme

Les algorithmes `Mads` et `SuDe` sont conceptuellement très proches, au point qu'il est possible de les exprimer comme deux variantes d'un unique algorithme. Cette structure générale est esquissée par [116, algorithme 1]. Dans cette thèse nous nous concentrons sur une adaptation de leur construction donnée par l'algorithme 1. Les différences par rapport à [116, algorithme 1] sont mineures ; ce sont des détails techniques (que nous ne discutons pas ici) pour éviter des difficultés théoriques.

Algorithme 1 `DSM` générique pour résoudre le problème (**P**).

Initialisation :
initialiser $(x^0; \delta^0) \in (X \cap \Omega) \times \mathbb{R}_+^*$, et poser $\underline{\delta}^0 \triangleq \delta^0$;
définir $\rho : \mathbb{R}_+ \to \mathbb{R}_+$ et $\mathcal{M} : \mathbb{R}_+ \to 2^{\mathbb{R}^n}$ et $\Lambda \subset]0, 1[$ et $\Upsilon \subset [1, +\infty[$ suivant, au choix,
— *`DSM` à treillis adaptatif* : $\rho(\nu) \triangleq 0$ et $\mathcal{M}(\nu) \triangleq \min\{\nu, \frac{\nu^2}{\delta^0}\} M\mathbb{N}^p$ pour tout $\nu \in \mathbb{R}_+$, où $M \in \mathbb{R}^{n \times p}$ est une base positive de $\mathbb{R}^n$ (avec $p > n$), et $\Lambda \subseteq \{\tau^\ell : \ell \in [\![1, m]\!]\}$ et $\Upsilon \subseteq \{\tau^\ell : \ell \in [\![-m, 0]\!]\}$, où $\tau \in \mathbb{Q} \cap]0, 1[$ et $m \in \mathbb{N}^*$;
— *`DSM` à décroissance suffisante* : ρ nulle en 0 et strictement croissante et $\mathcal{M}(\nu) \triangleq \mathbb{R}^n$ pour tout $\nu \in \mathbb{R}_+$, et $\Lambda \subseteq [\underline{\lambda}, \overline{\lambda}]$ et $\Upsilon \subseteq [\underline{\upsilon}, \overline{\upsilon}]$, où $0 < \underline{\lambda} \leq \overline{\lambda} < 1 \leq \underline{\upsilon} \leq \overline{\upsilon} < +\infty$;

pour $k \in \mathbb{N}$ **faire** :
exploration (notée `search`) :
définir $\mathcal{D}_{\mathtt{S}}^k \subseteq \mathcal{M}(\underline{\delta}^k)$ et $\mathcal{T}_{\mathtt{S}}^k \triangleq \{x^k\} + \mathcal{D}_{\mathtt{S}}^k$; si $\mathcal{T}_{\mathtt{S}}^k \cap \Omega \neq \emptyset$ et $t_{\mathtt{S}}^k \in \operatorname{argmin} f(\mathcal{T}_{\mathtt{S}}^k \cap \Omega)$ vérifie $f(t_{\mathtt{S}}^k) < f(x^k) - \rho(\underline{\delta}^k)$, définir $t^k \triangleq t_{\mathtt{S}}^k$ et ne pas faire la `poll` ;
sonde (notée `poll`) :
définir $\emptyset \neq \mathcal{D}_{\mathtt{P}}^k \subseteq \mathcal{M}_{\delta^k}(\underline{\delta}^k)$ et $\mathcal{T}_{\mathtt{P}}^k \triangleq \{x^k\} + \mathcal{D}_{\mathtt{P}}^k$; si $\mathcal{T}_{\mathtt{P}}^k \cap \Omega \neq \emptyset$ et $t_{\mathtt{P}}^k \in \operatorname{argmin} f(\mathcal{T}_{\mathtt{P}}^k \cap \Omega)$ vérifie $f(t_{\mathtt{P}}^k) < f(x^k) - \rho(\underline{\delta}^k)$, définir $t^k \triangleq t_{\mathtt{P}}^k$, sinon définir $t^k \triangleq x^k$;
mise à jour des paramètres :
si $t^k \neq x^k$, définir $\delta^{k+1} \in \delta^k \Upsilon$, sinon définir $\delta^{k+1} \in \delta^k \Lambda$; définir $\underline{\delta}^{k+1} \triangleq \min_{\ell \leq k+1} \delta^k$; définir $x^{k+1} \triangleq t^k$.

`DSM` est flexible dans les choix des ensembles $\mathcal{D}_{\mathtt{S}}^k$ et $\mathcal{D}_{\mathtt{P}}^k$ à chaque itération $k \in \mathbb{N}$, ainsi que dans les choix des paramètres. Moult références discutent de bons choix de $\mathcal{D}_{\mathtt{S}}^k$ [9, 16] et de $\mathcal{D}_{\mathtt{P}}^k$ [1, 18, 60, 62, 90, 97, 114]. Des choix éprouvés sont donnés dans [17, chapitre 8] pour `DSM` à treillis adaptatif, et dans [35, chapitre 7] pour `DSM` à décroissance suffisante. Cependant, le théorème suivant est vrai pour tout choix algorithmique valide.

Théorème 2.9. **Corollaire des théorèmes 2.7 et 2.8.** Si f est Lipschitz-continue, alors $f_{\mathtt{C}}^\circ(x^*; d^*) \geq 0$ pour tout $(x^*; d^*) \in X \times (D^* \cap \mathcal{H}(x^*))$ tels que $(K^*; x^*; D^*) \in \mathbb{G}_{\mathtt{DSM}}^*$.

Des extensions de `DSM` gérant plus efficacement les contraintes existent, comme [12, 15] reposant sur le principe proposé par [48]. Nous les mentionnons car elles servent au chapitre 7, mais nous ne les discutons pas plus ici.

Invalidité de DSM sur des cas discontinus

Lorsque la fonction f est discontinue, le théorème 2.9 ne tient plus. La référence [116] fut un premier pas vers un nouveau théorème, pensant pouvoir adapter le théorème 2.9 directement en passant de la dérivée de Clarke à celle de Rockafellar. En d'autres termes, l'objectif de [116] était de montrer que DSM génère des points raffinés vérifiant la proposition 1.5. Cependant, l'analyse développée dans cet article comporte une erreur qui rend invalide l'un des résultats proposés. Nous discutons de ce fait plus en détails au chapitre 3. Nous prouvons que lorsque f n'est pas continue, DSM peut ne générer aucun point raffiné satisfaisant la proposition 1.5 donnée au chapitre 1.2.2. Par conséquent, au moment de la rédaction de cette thèse (fin 2023), DSM n'a aucune garantie de convergence dans le cas discontinu, et il doit forcément être adapté pour obtenir une analyse satisfaisante. De fait, actuellement ni Cs et Gps, ni Mads, ni SuDe n'ont d'analyse de convergence satisfaisante dans le cas discontinu. Nous traitons cette question au chapitre 4.

Un autre travail précurseur de l'étude des cas discontinus est [8], qui introduit l'algorithme DiscoMads (*discontinuities* Mads). La finalité de DiscoMads est de ne pas renvoyer de point trop proche d'une discontinuité. Pour cela, il attend en paramètre un rayon de sécurité $r > 0$ fixé, et à chaque itération il effectue une étape supplémentaire nommée *révélatrice* (revealing dans la référence) : à δ fixé quelconque, l'étape revealing considère un ensemble $\mathcal{D}_{\mathtt{R}} \subset \mathcal{M}_r(\delta)$ de directions. Ainsi, lorsque l'algorithme génère un point raffiné x^*, toute discontinuité à distance inférieure à r de x^* peut être révélée. DiscoMads ne permet pas de renforcer l'analyse de convergence, mais l'idée derrière l'étape revealing est une base intéressante que nous utilisons dans cette thèse pour établir notre propre analyse.

CHAPITRE 3 CORRECTIF À UN RÉSULTAT DE LA LITTÉRATURE

Que me soient donnés la force de supporter ce qui ne peut être changé, le courage d'altérer ce qui peut l'être, mais aussi la sagesse de distinguer l'un de l'autre.

« Pensées pour moi-même », Marc Aurèle.

Le chapitre 2 démontre que l'optimisation discontinue est un domaine difficile. Dans ce contexte, DSM se démarque des autres méthodes de la DFO. Parmi les méthodes ayant des garanties de convergence, ce sont les seules ne tentant pas d'approximer la fonction objectif par un modèle continu. Il est donc naturel de supposer que cette méthode est la plus propice à gérer les cas discontinus. L'un des travaux fondateurs de l'étude de DSM en contexte discontinu est « Analysis of direct searches for discontinuous functions » [116], qui visait à prouver que DSM ne requiert aucune adaptation. Malheureusement, cette thèse doit revenir sur [116] car l'un de ses derniers théorèmes ([116, théorème 4.1]) est erroné. Le contenu de ce chapitre est extrait de notre référence [11], acceptée pour publication dans la revue *Mathematical Programming* et en cours d'édition au moment de l'écriture de cette thèse.

Nous commençons par rappeler le raisonnement mené par les auteurs de « Analysis of direct searches for discontinuous functions », en précisant les enjeux, les différents résultats établis pour y répondre, et les techniques utilisées pour les établir. Ceci fait l'objet du chapitre 3.1. Nous attirons ensuite l'attention sur le résultat erroné et sur les conséquences de cette erreur sur la fin du raisonnement. Pour ce faire, nous construisons un contre-exemple pour illustrer le problème et nous étudions en détail la preuve du théorème incorrect pour cibler l'argument invalide. Le chapitre 3.2 se concentre sur ces questions. Enfin, nous proposons une modification de DSM afin d'éviter cette écueil. Nous montrons qu'il est alors possible de suivre une analyse proche de celle menée dans [116] pour retrouver des résultats similaires à ceux auparavant erronés, sous réserve que l'on accepte une hypothèse supplémentaire. Ce travail est effectué au chapitre 3.3.

3.1 Synthèse de « Analysis of direct searches for discontinuous functions »

« Analysis of direct searches for discontinuous functions » [116] avait deux objectifs. Le premier consistait à rendre la dérivée de Rockafellar exploitable dans l'étude du problème (**P**), pour établir des conditions nécessaires d'optimalité lorsque la fonction objectif est discontinue. Nous revenons sur cet objectif au chapitre 3.1.1, qui a été atteint. Le second était de prouver que DSM génère au moins un point raffiné vérifiant cette proposition. Comme nous le montrons au chapitre 3.1.2, cet objectif n'est malheureusement pas validé.

3.1.1 Établissement de conditions nécessaires d'optimalité

Pour établir la proposition 1.5 dans le cas contraint, [116] commence par introduire, en tout point $x \in X$ et toute direction $d \in \mathbb{S}^n$, la *sous-dérivée supérieure* qu'on note

$$f^-(x;d) \triangleq \limsup_{\substack{y \to x,\ h \searrow 0,\\ f(y) \to f(x),\ y \in \Omega}} \inf_{\substack{u \to d,\\ y + hu \in \Omega}} \frac{f(y+hu) - f(y)}{h},$$

et un analogue que l'on pourrait nommer *sur-dérivée supérieure* dans le cas contraint,

$$f^+(x;d) \triangleq \limsup_{\substack{y \to x,\ h \searrow 0,\\ f(y) \to f(x),\ y \in \Omega}} \sup_{\substack{u \to d,\\ y + hu \in \Omega}} \frac{f(y+hu) - f(y)}{h};$$

pour enfin établir la dérivée de Rockafellar dans le cas contraint,

$$f^\circ_{\mathtt{R}}(x;d) \triangleq \limsup_{\substack{y \to x,\ h \searrow 0,\\ f(y) \to f(x),\ y \in \Omega,\ y + hd \in \Omega}} \frac{f(y+hd) - f(y)}{h}.$$

Il est alors prouvé que si f est semi-continue en un point $x \in X$ et que $f^+(x;d) < +\infty$ pour une direction $d \in \mathbb{S}^n$, alors $f^-(x;d) = f^+(x;d) = f^\circ_{\mathtt{R}}(x;d)$. Ce résultat généralise le cas non contraint, prouvé dans [99]. Enfin, [116] prouve également que lorsque d appartient à $\mathrm{cl}(\mathcal{H}(x))$, le cône tangent à Ω en x, alors $f^-(x;d)$ est la limite inférieure d'une suite $f^\circ_{\mathtt{R}}(x;u^k)$ où $u^k \in \mathcal{H}(x)$ pour tout $k \in \mathbb{N}$ et converge vers d. Cela mène à la première conclusion importante de [116] : un résultat de convergence valide pour toute direction $d \in \mathrm{cl}(\mathcal{H}(x))$.

Théorème 3.1. **[116, Théorème 3.2].** Lorsque DSM résout le problème (**P**), pour tout triplet $(K^*; x^*; D^*) \in \mathbb{G}^*_{\mathtt{DSM}}$ tel que $(f(x^k))_{k \in K^*} \to f(x^*)$ et pour tout $d^* \in D^* \cap \mathcal{H}(x^*)$ admettant un voisinage $\mathcal{U}$ dans $\mathbb{S}^n$ tel que $\sup f^+(x^*;\mathcal{U}) < +\infty$, on a $f^\circ_{\mathtt{R}}(x^*;d^*) \geq 0$.

Rien ne laisse penser que ce théorème est erroné et aucune erreur n'a été détectée dans sa preuve. Nous le tenons donc pour acquis dans cette thèse. Nous avons déjà statué au chapitre 2.2.3 que l'ensemble D^* des directions raffinées peut égaler $\mathbb{S}^n$; aussi le principal inconvénient de ce théorème est la condition technique $(f(x^k))_{k \in K^*} \to f(x^*)$.

Ensuite, [116] propose un second théorème, visant à établir des conditions suffisantes pour garantir $(f(x^k))_{k \in K^*} \to f(x^*)$. Nous en parlons au chapitre 3.1.2, car il s'agit de la partie du raisonnement qui est erronée.

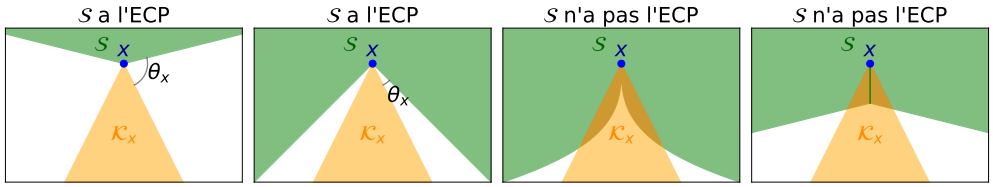
3.1.2 Affirmation de la convergence de DSM vers un point vérifiant des conditions nécessaires d'optimalité

Même lorsque la fonction objectif est semi-continue, il est possible qu'une suite $(x^k)_{k\in\mathbb{N}}$ d'éléments de X convergeant vers un point $x^* \in X$ ne vérifie pas $(f(x^k))_{k\in\mathbb{N}} \to f(x^*)$. Prenons par exemple $f \triangleq \mathcal{I}_{\mathbb{R}_+^*} : \mathbb{R} \to \mathbb{R}$, définie par $\mathcal{I}_{\mathbb{R}_+^*}(x) \triangleq 0$ si $x \in \mathbb{R}_-$ et $\mathcal{I}_{\mathbb{R}_+^*}(x) \triangleq 1$ si $x \in \mathbb{R}_+^*$; et $x^* \triangleq 0$ approché par la suite $x^k \triangleq 2^{-k}$ pour tout $k \in \mathbb{N}$. Ainsi, le pré-requis du théorème 3.1 stipulant que DSM doit générer une suite raffinante vérifiant $(f(x^k))_{k\in K^*} \to f(x^*)$ n'est pas automatique. Pour le garantir, [116] propose une hypothèse supplémentaire sur les ensembles de continuité de f et prétend qu'alors DSM génère des points raffinés approchés depuis l'ensemble de continuité de f les contenant. Ce résultat est cependant erroné, comme illustré au chapitre 3.2.

L'hypothèse proposée par [116] repose sur la *propriété de cône extérieur* (*Exterior Cone Property, ECP*), que nous rappelons à la définition 3.1 et illustrons en figure 3.1. Nous adaptons ici les notations de [116] pour coïncider avec celles que nous utilisons par la suite.

Définition 3.1. **ECP, adaptée de [116, définition 4.1].** Un ensemble $\mathcal{S} \subseteq \mathbb{R}^n$ a la *propriété de cône extérieur* si pour tout $x \in \partial\mathcal{S}$, il existe un cône $\mathcal{K}$ d'intérieur non vide, un voisinage $\mathcal{O}$ de 0 et un angle $\theta_x \in]0, \pi[$ tel que $\Theta(e - x, s - x) \geq \theta_x$ pour tout $(e, s) \in (\mathcal{E}_x \setminus \{x\}) \times (\mathcal{S}_x \setminus \{x\})$, où $\Theta(\cdot, \cdot)$ désigne l'angle saillant entre deux vecteurs et $\mathcal{E}_x \triangleq (\mathcal{O}_x \cap \mathcal{K}_x) \subset (\mathbb{R}^n \setminus \mathcal{S})$ et $\mathcal{S}_x \triangleq \mathcal{S} \cap \mathcal{O}_x$, et $\mathcal{O}_x \triangleq \{x\} + \mathcal{O}$ et $\mathcal{K}_x \triangleq \{x\} + \mathcal{K}$.

FIGURE 3.1 Illustration d'ensembles ayant l'ECP ou non.



L'ECP est alors utilisée pour poser l'hypothèse suivante sur f.

Hypothèse 3.1. **[116, hypothèse 4.1].** Pour tout $x \in X \cap \Omega$, il existe un voisinage $\mathcal{X}$ de x admettant une partition $\mathcal{X} = \sqcup_{i=0}^{N} \mathcal{X}_i$ (où $N \in \mathbb{N}^*$ est fini) vérifiant $\mathcal{X}_0 \triangleq (\mathcal{X} \setminus (X \cup \Omega))$ et, pour tout $i \in [\![1, N]\!]$, $\mathrm{int}(\mathcal{X}_i) \neq \emptyset$ et $\mathcal{X}_i$ a l'ECP et $f_{|\mathcal{X}_i}$ est Lipschitz-continue.

Sous l'hypothèse 3.1, [116] prétend le résultat suivant.

Énoncé 3.1. **[116, théorème 4.1 et corollaire 4.1].** Lorsque DSM résout le problème (**P**) sous l'hypothèse 3.1, tout $(K^*; x^*; D^*) \in \mathbb{G}^*_{\texttt{DSM}}$ tel que $D^*_{K'}$ est dense dans $\mathbb{S}^n$ pour tout $K' \subseteq K^*$ (où $D^*_{K'}$ est l'ensemble des directions raffinées associées à K') vérifie

1. soit $x^* \in \cup_{i=1}^N \text{int}(\mathcal{X}_i)$, et alors $f^\circ_{\mathsf{C}}(x^*; d) \geq 0$ pour tout $d \in \text{cl}(\mathcal{H}(x^*))$;
2. soit $x^* \notin \cup_{i=1}^N \text{int}(\mathcal{X}_i)$, et alors il existe $K' \subseteq K^*$ et un indice $j \in [\![1, N]\!]$ tel que $x^k \in \text{cl}(\mathcal{X}_j)$ pour tout $k \in K'$ et $x^k \in \text{int}(\mathcal{X}_j)$ pour une infinité de valeurs de $k \in K^*$ et $x^k \in \mathbb{R}^n \setminus \text{cl}(\mathcal{X}_j)$ pour tout $k \in K'$;

où on note $\mathcal{X} = \sqcup_{i=0}^N \mathcal{X}_i$ un voisinage de x^* tel que défini par l'hypothèse 3.1. Si de plus $N = 2$, alors l'indice j dans les résultats précédents est l'indice tel que $x^* \in \mathcal{X}_j$, et de plus $(f(x^k))_{k \in K^*} \to f(x^*)$.

Ce résultat clôture le raisonnement mené par [116], puisqu'il donne des conditions sur f garantissant le pré-requis $(f(x^k))_{k \in K^*} \to f(x^*)$ impliqué dans le théorème 3.1. L'analyse est encore améliorable, notamment sur le présupposé que $x^* \in X$, sur la restriction au cas $N = 2$, et sur l'hypothèse de densité de l'ensemble des directions raffinées associé à n'importe quel $K' \subseteq K^*$. Cependant, le résultat contient un autre problème plus profond sur lequel nous nous concentrons maintenant. Le point crucial de l'énoncé est l'affirmation que DSM génère des points dans chacun des deux ensembles de continuité de f lorsque $N = 2$. Or, ceci est malheureusement erroné.

Remarque 3.1. L'hypothèse portant sur la densité de l'ensemble des directions raffinées associé à n'importe quel $K' \subseteq K^*$ n'est jamais vérifiée en pratique. Fixons $\mathcal{U}$ un ouvert de $\mathbb{S}^n$, et définissons $K' \subseteq K^*$ comme l'ensemble des itérations $k \in K^*$ dans lesquelles aucune direction d'évaluation n'appartient à $\mathcal{U}$. Alors K' contredit l'hypothèse. L'énoncé 3.1 est donc déjà inutilisable pour cette raison. Cependant, ce défaut important de l'énoncé 3.1 n'est pas celui sur lequel nous nous attardons, car il est issu d'une interprétation dégénérée de la notion de suite raffinante. Il est contre-productif de sélectionner délibérément une suite raffinante de manière à éviter d'avoir un ensemble de directions raffinées. En comparaison, l'erreur que nous mettons en évidence est plus critique, car elle est issue d'une faiblesse inhérente à DSM.

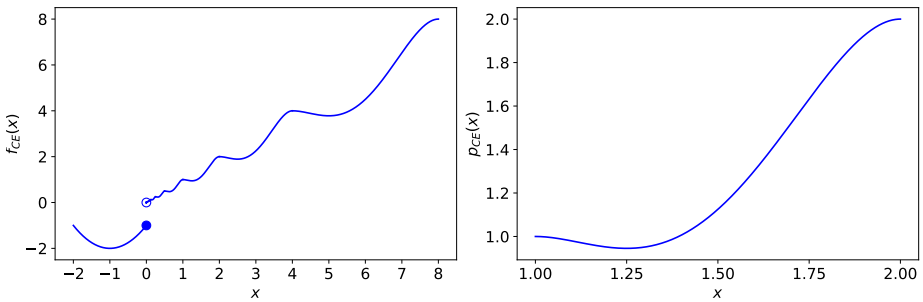
3.2 Erreur dans l'affirmation de la convergence de DSM

Le problème dans le raisonnement proposé dans « Analysis of direct searches for discontinuous functions » a déjà été illustré dans la référence [11], l'un des articles écrits durant cette thèse. Aussi, le contenu de ce chapitre reprend le contenu de [11, sections 2 et 3].

3.2.1 Construction d'un contre-exemple

Le contre-exemple que nous construisons ici prouve que l'énoncé 3.1 est erroné. Nous proposons une instance du problème (**P**) avec une fonction objectif $f_{\mathtt{CE}} : \mathbb{R}^2 \to \mathbb{R}$ présentant une discontinuité en 0 satisfaisant l'hypothèse 3.1 avec $N = 2$, et un ensemble $\Omega = \mathbb{R}^2$. La fonction $f_{\mathtt{CE}}$ est représentée dans le cadre de gauche de la figure 3.2.

FIGURE 3.2 Fonction objectif $f_{\mathtt{CE}}$ du contre-exemple.



Construction de la fonction objectif

Sur $\mathbb{R}_-$, $f_{\mathtt{CE}}$ est définie comme $f_{\mathtt{CE}} : x \in \mathbb{R}_- \mapsto (x+1)^2 - 2$. Sur $\mathbb{R}_+^*$, $f_{\mathtt{CE}}$ est construite à partir du polynôme $p_{\mathtt{CE}} : [1,2] \to \mathbb{R}$ suivant, illustré dans le cadre de droite de la figure 3.2,

$$p_{\mathtt{CE}}(x) \triangleq -18 + 60x - 69x^2 + 34x^3 - 6x^4 \text{ pour tout } x \in [1,2],$$

vérifiant $p_{\mathtt{CE}}(1) = 1$, $p_{\mathtt{CE}}(2) = 2$, $p'_{\mathtt{CE}}(1){=}p'(\frac{5}{4}){=}p'_{\mathtt{CE}}(2) = 0$, $p''_{\mathtt{CE}}(1) < 0$, $p''_{\mathtt{CE}}(2) < 0$, $p''_{\mathtt{CE}}(\frac{5}{4}) > 0$. Ainsi, $p_{\mathtt{CE}}$ a un unique minimum local en $x = \frac{5}{4}$, avec $p_{\mathtt{CE}}(x) = \frac{121}{128}$, et deux maxima locaux aux bornes de l'intervalle $[1,2]$. La fonction $f_{\mathtt{CE}}$ est alors construite sur $\mathbb{R}_+^*$ par mises à l'échelle de $p_{\mathtt{CE}}$: $f_{\mathtt{CE}}$ égale $2p_{\mathtt{CE}}$ sur $2[1,2[$, égale $p_{\mathtt{CE}}$ sur $[1,2[$, égale $\frac{p_{\mathtt{CE}}}{2}$ sur $\frac{1}{2}[1,2[$, et plus généralement égale $2^\ell p_{\mathtt{CE}}$ sur $2^\ell[1,2[$ avec $\ell \in \mathbb{Z}$. Ainsi, sur tout intervalle $2^\ell[1,2]$ avec $\ell \in \mathbb{Z}$, $f_{\mathtt{CE}}$ a exactement un minimum local en $x = 2^\ell\frac{5}{4}$, de valeur objectif $2^\ell\frac{121}{128}$.

Le contre-exemple consiste donc en l'instance suivante du problème (**P**) :

$$\underset{x \in \mathbb{R}}{\text{minimiser}} \quad f_{\mathtt{CE}}(x), \qquad \text{où} \qquad f_{\mathtt{CE}} : x \mapsto \begin{cases} (x+1)^2 - 2 & \text{si } x \leq 0, \\ 2^\ell p_{\mathtt{CE}}\left(\dfrac{x}{2^\ell}\right) & \text{si } x \in 2^\ell[1,2[\text{ avec } \ell \in \mathbb{Z}. \end{cases} \tag{$\mathbf{P}_{\mathtt{CE}}$}$$

Il s'agit bien une instance du problème (**P**), et les hypothèses requises par « Analysis of direct searches for discontinuous functions » sont toutes satisfaites ici.

Proposition 3.2. La fonction $f_{\texttt{CE}}$ est semi-continue, a ses ensembles de niveau bornés, et vérifie l'hypothèse 3.1.

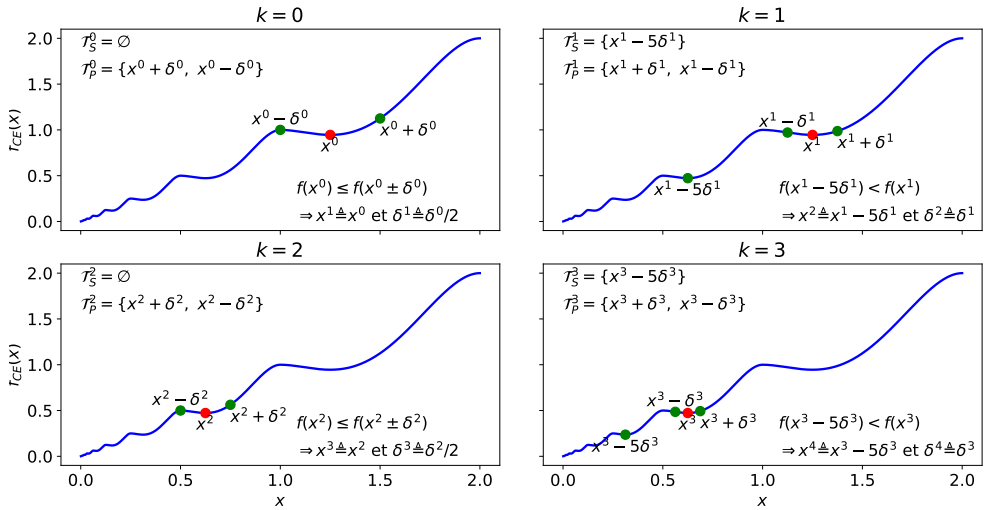
Démonstration. La fonction $f_{\texttt{CE}}$ est semi-continue et a ses ensembles de niveau bornés. De plus, l'hypothèse 3.1 tient en tout $x \neq 0$, et en $x = 0$ le voisinage $\mathcal{X} \triangleq]-2, 2[$ peut être partitionné en $\mathcal{X} = \mathcal{X}_1 \sqcup \mathcal{X}_2$ où $\mathcal{X}_1 \triangleq]-2, 0]$ et $\mathcal{X}_2 \triangleq]0, 2[$ tels que $\mathrm{int}(\mathcal{X}_i) \neq \emptyset$ et $f_{\texttt{CE}|\mathcal{X}_i}$ est Lipschitz-continue et $\mathrm{cl}(\mathcal{X}_i)$ vérifie l'ECP, pour $i \in \{1, 2\}$. □

Construction de l'instance de DSM

L'instance de DSM retenue pour résoudre le problème ($\mathbf{P}_{\texttt{CE}}$) est la suivante. L'initialisation est faite avec $(x^0; \delta^0) \triangleq (\frac{5}{4}; \frac{1}{4})$. Nous choisissons une instance de DSM à treillis adaptatif, avec $M \triangleq [-1, 1]$ et $\tau = \frac{1}{2}$, $\Upsilon \triangleq \{1\}$, $\Lambda \triangleq \{\frac{1}{2}\}$. L'étape de `search` est négligée aux itérations paires ($\mathcal{D}_{\mathrm{S}}^k \triangleq \emptyset$ pour tout $k \in 2\mathbb{N}$), et évalue une direction autrement : $\mathcal{D}_{\mathrm{S}}^k \triangleq \{-5\delta^k\}$ pour tout k tel que $k-1 \in 2\mathbb{N}$. L'ensemble des directions de `poll` est $\mathcal{D}_{\mathrm{P}}^k \triangleq \{-1, 1\}$ pour tout $k \in \mathbb{N}$.

Ces paramètres sont choisis de telle sorte qu'à chaque itération paire, $k = 2q$ avec $q \in \mathbb{N}$, la solution courante x^{2q} est le minimum de $f_{\texttt{CE}}$ sur l'intervalle $2^{-q}[1, 2]$, l'étape de `poll` échoue car tous les points qu'elle génère sont dans cet interalle, et l'étape de `search` tournant à l'itération suivante ($k = 2q + 1$) génère son point candidat exactement au minimum de $f_{\texttt{CE}}$ sur l'intervalle $2^{-(q+1)}[1, 2]$). Par conséquent, $(x^k)_{k\in\mathbb{N}}$ converge vers l'origine, $(\delta^k)_{k\in\mathbb{N}} \to 0$, et les directions $d_1 = -1$ et $d_2 = 1$ sont considérées à chaque itération paire. Nous illustrons ces affirmations en figure 3.3 et les formalisons dans le lemme 3.3.

FIGURE 3.3 Quatre premières itérations de DSM sur le contre-exemple.



Lemme 3.3. Les suites $(x^k)_{k\in\mathbb{N}}$ et $(\delta^k)_{k\in\mathbb{N}}$ satisfont

$$\forall q \in \mathbb{N}, \quad x^{2q} = 2^{-q}\frac{5}{4} = x^{2q+1}, \quad \delta^{2q} = 2^{-q}\frac{1}{4} = 2\delta^{2q+1}.$$

De plus, à chaque itération paire, $k = 2q$ avec $q \in \mathbb{N}$, l'étape de `poll` échoue après avoir considéré les directions $d_1 \triangleq -1$ et $d_2 \triangleq 1$.

Démonstration. Cette preuve fonctionne par récurrence sur $q \in \mathbb{N}$.

L'itération 0 commence avec $x^0{=}\frac{5}{4}$ et $\delta^0{=}\frac{1}{4}$. L'étape de `search` est négligée car 0 est pair. Le point x^0 est le minimum de $f_{\mathtt{CE}}$ sur $[1,2]$, et ainsi la `poll` mène à $f_{\mathtt{CE}}(t_{\mathtt{P}}^0) \geq f_{\mathtt{CE}}(x^0)$ (car elle évalue $\mathcal{T}_{\mathtt{P}}^0 \triangleq \{x^0+\delta^0 d_1, x^0+\delta^0 d_2\} \subset [1,2]$). L'itération 0 est donc un échec. Cela implique $x^1 \triangleq x^0 = \frac{5}{4}$ et $\delta^1 \triangleq \frac{1}{2}\delta^0 = 2^{-1}\frac{1}{4}$. Le résultat désiré est donc vrai pour $q = 0$.

Considérons $q \in \mathbb{N}^*$ et supposons que le résultat est vrai pour $q{-}1$. Par hypothèse, l'itération $2q{-}1$ est partie de $x^{2q-1} = x^{2(q-1)+1} = 2^{-(q-1)}\frac{5}{4}$ et $\delta^{2q-1} = \delta^{2(q-1)+1} = 2^{-q}\frac{1}{4}$. Puisque $2q{-}1$ est impair, la `search` a généré le point $t_{\mathtt{S}}^{2q-1} \triangleq x^{2q-1} - 5\delta^{2q-1} = 2^{-q}\frac{5}{4}$. Ce point satisfait $f_{\mathtt{CE}}(t_{\mathtt{S}}^{2q-1}) = \frac{f_{\mathtt{CE}}(x^{2q-1})}{2} < f_{\mathtt{CE}}(x^{2q-1})$. L'itération $2q{-}1$ était donc un succès, donc l'itération $2q$ part de $x^{2q} \triangleq t_{\mathtt{S}}^{2q-1} = 2^{-q}\frac{5}{4}$ et $\delta^{2q} \triangleq \delta^{2q-1} = 2^{-q}\frac{1}{4}$. Cette itération est d'indice pair, donc il n'y a pas d'étape de `search`. La solution courante, x^{2q}, est le minimum de $f_{\mathtt{CE}}$ sur $2^{-q}[1,2]$, et ainsi la `poll` mène à $f_{\mathtt{CE}}(t_{\mathtt{P}}^{2q}) \geq f_{\mathtt{CE}}(x^{2q})$ (car elle évalue $\mathcal{T}_{\mathtt{P}}^{2q} \triangleq \{x^{2q}+\delta^{2q}d_1, x^{2q}+\delta^{2q}d_2\} \subset 2^{-q}[1,2]$). L'itération $2q$ est donc un échec. Cela implique $x^{2q+1} \triangleq x^{2q} = 2^{-q}\frac{5}{4}$ et $\delta^{2q+1} \triangleq \frac{1}{2}\delta^{2q} = 2^{-(q+1)}\frac{1}{4}$. Le résultat désiré est donc vrai pour $q \in \mathbb{N}^*$ s'il est vrai pour $q{-}1$. □

La suite $(x^k;\delta^k;\mathcal{D}^k)_{k\in\mathbb{N}}$ résultant de l'exécution de `DSM` sur le problème $(\mathbf{P}_{\mathtt{CE}})$ vérifie toutes les conditions requises par [116, théorème 4.1] et [116, corollaire 4.1].

Proposition 3.4. La suite $(x^k;\delta^k;\mathcal{D}^k)_{k\in\mathbb{N}}$ est une suite raffinante, avec l'origine comme point raffiné, et l'ensemble des directions raffinées associé est $D^* \triangleq \{-1,1\}$.

Démonstration. Le lemme 3.3 prouve que la suite $(x^k)_{k\in\mathbb{N}}$ converge vers $x^* \triangleq 0$ et que la suite $(\delta^k)_{k\in\mathbb{N}}$ converge vers 0. De plus, pour tout $k \in 2\mathbb{N}$, l'étape de `poll` échoue après avoir évalué $x^k + \delta^k d$ pour tout $d \in \mathcal{D}_{\mathtt{P}}^k = \{-1,1\}$. Ainsi, $K^* \triangleq \mathbb{N}$ indexe une suite raffinante, et l'ensemble des directions raffinées est $D^* = \{-1,1\}$. □

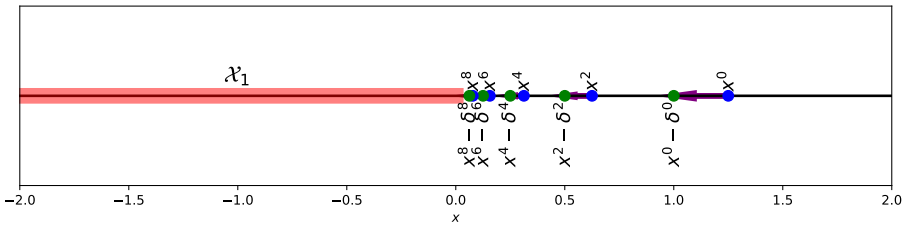
Observation que le contre-exemple contredit le théorème

Les propositions 3.2 et 3.4 prouvent que le contre-exemple satisfait toutes les conditions énoncées dans l'énoncé 3.1. l'énoncé 3.1 prétend donc qu'il existe une suite raffinante (indexée par $K' \subseteq \mathbb{N}$) et $i \in \{1, 2\}$ tels que

$$(a): \quad \forall k \in K', \begin{cases} x^k \in \mathrm{cl}(\mathcal{X}_i), \\ x^k + \delta^k d_1 \in \mathbb{R} \setminus \mathrm{cl}(\mathcal{X}_i), \\ x^k + \delta^k d_2 \in \mathrm{int}(\mathcal{X}_i), \end{cases} \qquad \text{et} \qquad (b): \quad \lim_{k \in K} f_{\texttt{CE}}(x^k) = f_{\texttt{CE}}(x^*).$$

La suite $(x^k)_{k\in\mathbb{N}}$ ayant tous ses éléments dans $\mathcal{X}_2 =]0, 2[$, et donc aucun dans $\mathcal{X}_1 =]-2, 0]$, il nous faut considérer $i \triangleq 2$ pour espérer satisfaire le résultat (a). De plus, il nous faut considérer $K^* \subseteq 2\mathbb{N}$, car sinon K^* n'indexe pas une suite raffinante. Or, aucun $K' \subseteq K^*$ ne vérifie le résultat (a). En effet, pour tout $k \in K^*$, le point $x^k + \delta^k d_1 = x^k - \delta^k = 2^{-q}$ est dans $\mathcal{X}_2$ (où $q = \frac{k}{2}$), et donc il ne rencontre jamais l'ensemble $\mathbb{R} \setminus \mathrm{cl}(\mathcal{X}_2)$. La figure 3.4 illustre ce phénomène. De plus, la suite $(f_{\texttt{CE}}(x^k))_{k\in K^*}$ converge vers 0 tandis que $f_{\texttt{CE}}(x^*) = -1$. De fait, le résultat (b) est également erroné.

FIGURE 3.4 Échec de DSM à visiter chaque ensemble de continuité.



3.2.2 Analyse de la preuve erronée

Le contre-exemple proposé à la section précédente met en évidence un problème dans l'énoncé 3.1. Cette section vise à analyser argument par argument la preuve donnée par [116], afin de localiser cette erreur. On se place dans le cadre proposé par l'énoncé 3.1.

La preuve proposée par [116] est divisée en six paragraphes que nous décrivons ci-dessous. Un septième mène à [116, corollaire 4.1]. L'erreur réside dans le cinquième paragraphe.

Notons $i \in [\![1, N]\!]$ tel que $x^* \in \mathcal{X}_i$. D'abord, [116] restreint la preuve au cas de DSM à treillis adaptatif, en clamant que la preuve s'adapte aisément au cas de DSM à décroissance suffisante. Ensuite, [116] traite le cas où $x^* \in \mathrm{int}(\mathcal{X}_i)$, qui est similaire aux analyses menées sous une hypothèse de continuité de f. Ceci fait, [116] considère alors le cas où $x^* \in \partial\mathcal{X}_i$.

Le sous-cas où $x^k \in \partial\mathcal{X}_i$ pour tout $k \in K^*$ à partir d'un certain rang est simple. Dans le sous-cas restant, [116] extrait une sous-suite (indexée par $K' \subseteq K^*$) telle que $x^k \in \operatorname{int}(\mathcal{X}_i)$ pour tout $k \in K'$. Puisque l'ensemble des directions raffinées associées à K' est supposé dense dans $\mathbb{S}^n$, on peut extraire une seconde sous-suite (indexée par $K'' \subseteq K'$) telle que, pour tout $k \in K''$, certains points de `poll` sont dans $\operatorname{int}(\mathcal{X}_i)$. Ces étapes semblent correctes.

Ensuite, au cinquième paragraphe, [116] envisage de prouver par contradiction que certains des points de `poll` associés aux itérations $k \in K''$ ne sont pas dans $\operatorname{cl}(\mathcal{X}_i)$. Pour ce faire, [116] définit une fonction $\widetilde{f}$ étendant $f_{|\mathcal{X}_i}$ à $\mathbb{R}^n$ de manière Lipschitz-continue et telle que $\widetilde{f}$ est localement strictement décroissante en tout $x \in \partial\mathcal{X}_i$ dans toutes les directions pointant dans un cône extérieur à $\mathcal{X}_i$ en x. Ces étapes semblent correctes. Cependant, [116] considère alors une direction d dans laquelle $\widetilde{f}$ est strictement décroissante en x^*, et prétend que la dérivée de Clarke $\widetilde{f}^\circ_{\mathsf{C}}(x^*; d)$ est donc strictement négative. L'erreur dans la preuve réside dans cet argument.[1] L'extrait suivant, tiré de [116], met en évidence cette affirmation.

> **Erreur au cinquième paragraphe de la preuve de [116, théorème 4.1] (transcription dans les notations de cette thèse)** *Let us assume that all* `poll` *points associated with the refining subsequence belong to* $\operatorname{cl}(\mathcal{X}_i)$*. We will see that this leads us to a contradiction. [...] Let* $\widetilde{f}$ *be the extended function. We then obtain that*
>
> $$\widetilde{f}^\circ_{\mathsf{C}}(x^*; d) \;\geq\; [\ldots] \;\geq\; 0,$$
>
> *for all refining directions* d*, which is a contradiction since these directions are dense in the unit sphere and* $\widetilde{f}$ *is locally strictly decreasing from* x^* *along all directions in a cone of nonempty interior.*

Enfin, le sixième paragraphe (qui considère l'affirmation précédente comme vraie) déduit l'existence d'une troisième sous-suite (indexée par $K''' \subseteq K''$) telle que pour tout $k \in K'''$, l'étape de `poll` génère un point candidat $t \triangleq x^k + \delta^k d^k$ vérifiant $t \notin \operatorname{cl}(\mathcal{X}_i)$. Ceci conclut la preuve de [116, théorème 4.1]. Enfin, un septième paragraphe mène à [116, corollaire 4.1]. Dans le cas où $N = 2$, [116] déduit que certains points de `poll` sont dans $\mathcal{X}_2$ et d'autres dans $\mathcal{X}_1$ pour tout $k \in K'''$. Ainsi, la limite de $(f(x^k))_{k \in K'''}$ égale $f(x^*)$ par semi-continuité de f en x^* et le fait que la `poll` échoue à tout itération $k \in K'''$. Malheureusement, ces deux dernières étapes sont incorrectes si l'affirmation erronée n'est pas tenue pour vraie.

1. Le fait qu'une fonction f est strictement décroissante un point x dans une direction d n'implique pas que $f^\circ_{\mathsf{C}}(x; d) < 0$. Par exemple, $f : x \in \mathbb{R} \mapsto -|x|$ vérifie $f^\circ_{\mathsf{C}}(x; 1) = f^\circ_{\mathsf{C}}(x; -1) = 1$ en $x = 0$.

3.3 Modification de DSM corrigeant l'erreur

Le contre-exemple décrit précédemment illustre un défaut de DSM dans le cas discontinu. Pour $K^* \subseteq \mathbb{N}$ donné indexant une suite raffinante, il est possible que la suite des solutions courantes $(x^k)_{k \in K^*}$ approche le point raffiné x^* par un et un seul ensemble de continuité de f. L'algorithme ne détecte donc pas les discontinuités en x^*, même si x^* n'est pas dans l'ensemble de continuité de f contenant toutes les solutions courantes. Toutefois, il est possible de corriger DSM pour éviter ce défaut, comme proposé au chapitre 3.3.1. Nous proposons ensuite une analyse de convergence de cette modification de DSM, au chapitre 3.3.2. Enfin, nous montrons au chapitre 3.3.3 que notre modification de DSM traite le contre-exemple.

Précisons que ce chapitre reprend le contenu de [11, section 4], mais la lecture de l'analyse est optionnelle pour la compréhension de la suite de cette thèse. L'algorithme développé et étudié ci-après est généralisé au chapitre 4, et l'analyse menée dans ce chapitre repose sur des hypothèses plus fortes que dans [116], alors que celle du chapitre 4 requiert des hypothèses plus faibles. Ainsi, seul le chapitre 3.3.1 est important dans la suite de ce manuscrit.

3.3.1 Modification de DSM par ajout d'une étape revealing : rDSM

Le correctif à DSM proposé dans ce chapitre consiste en l'ajout d'une étape *révélatrice* à DSM, notée revealing et inspirée par l'algorithme DiscoMads [8, 71]. DiscoMads est une variante de Mads pensée pour un objectif différant du nôtre ([8] vise à éviter de générer des points raffinés proches d'une discontinuité), mais l'idée de l'étape revealing peut être reprise et adaptée. À chaque itération, l'étape revealing sélectionne une direction sur l'intersection entre le treillis et une boule de rayon constant. Par conséquent, si les directions de l'étape revealing sont bien choisies, le nouvel algorithme rDSM (revealing DSM) peut évaluer un ensemble de points dense dans chaque ensemble de continuité de f proche d'un point raffiné. Ainsi, contrairement à DSM, rDSM vérifie l'énoncé 3.1 clamé par [116].

L'algorithme rDSM est formalisé dans l'algorithme 2, et est très proche de l'algorithme DSM. Les différences sont l'ajout de l'étape de revealing, avec ses paramètres $r \in \mathbb{R}_+^*$ et $(\mathcal{D}_\ell)_{\ell \in \mathbb{N}}$ en initialisation, et l'ensemble $K^* \subseteq \mathbb{N}$ généré explicitement. Pour un treillis $\mathcal{M}$ donné, nous notons $\text{round}_\delta^{\mathcal{M}}(\cdot)$ la fonction d'*arrondi* sur $\mathcal{M}(\delta)$, pour tout $\delta \in \mathbb{R}_+^*$.

La suite $(\mathcal{D}_\ell)_{\ell \in \mathbb{N}}$ peut être construite en définissant chaque terme comme un tirage de la distribution de probabilité uniforme et indépendante sur $\text{cl}(\mathcal{B}_r)$. Une autre stratégie est d'utiliser des *suites pseudo-aléatoires* [58] de manière similaire à la génération de directions de poll dans [1]. Avec ces stratégies, il n'est pas nécessaire de construire la suite à priori ; chaque terme peut n'être défini qu'au moment où il intervient pour la première fois.

Algorithme 2 `rDSM` générique pour résoudre le problème (**P**), généralisant `DSM`.

Initialisation :
initialiser $(x^0; \delta^0) \in (X \cap \Omega) \times \mathbb{R}_+^*$, et poser $\underline{\delta}^0 \triangleq \delta^0$;
définir $\rho : \mathbb{R}_+ \to \mathbb{R}_+$ et $\mathcal{M} : \mathbb{R}_+ \to 2^{\mathbb{R}^n}$ et $\Lambda \subset]0,1[$ et $\Upsilon \subset [1,+\infty[$ suivant, au choix,
— *`DSM` à treillis adaptatif* : $\rho(\nu) \triangleq 0$ et $\mathcal{M}(\nu) \triangleq \min\{\nu, \frac{\nu^2}{\delta^0}\} M\mathbb{N}^p$ pour tout $\nu \in \mathbb{R}_+$, où $M \in \mathbb{R}^{n\times p}$ est une base positive de $\mathbb{R}^n$ (avec $p > n$), et $\Lambda \subseteq \{\tau^\ell : \ell \in [\![1, m]\!]\}$ et $\Upsilon \subseteq \{\tau^\ell : \ell \in [\![-m, 0]\!]\}$, où $\tau \in \mathbb{Q} \cap]0,1[$ et $m \in \mathbb{N}^*$;
— *`DSM` à décroissance suffisante* : ρ nulle en 0 et strictement croissante et $\mathcal{M}(\nu) \triangleq \mathbb{R}^n$ pour tout $\nu \in \mathbb{R}_+$, et $\Lambda \subseteq [\underline{\lambda}, \overline{\lambda}]$ et $\Upsilon \subseteq [\underline{\upsilon}, \overline{\upsilon}]$, où $0 < \underline{\lambda} \leq \overline{\lambda} < 1 \leq \underline{\upsilon} \leq \overline{\upsilon} < +\infty$;
initialiser $K^* \triangleq \emptyset$ et $\ell \triangleq 0$;

pour $k \in \mathbb{N}$ **faire** :
exploration (notée `search`) :
définir $\mathcal{D}_{\mathtt{S}}^k \subseteq \mathcal{M}(\underline{\delta}^k)$ et $\mathcal{T}_{\mathtt{S}}^k \triangleq \{x^k\} + \mathcal{D}_{\mathtt{S}}^k$; si $\mathcal{T}_{\mathtt{S}}^k \cap \Omega \neq \emptyset$ et $t_{\mathtt{S}}^k \in \operatorname{argmin} f(\mathcal{T}_{\mathtt{S}}^k \cap \Omega)$ vérifie $f(t_{\mathtt{S}}^k) < f(x^k) - \rho(\underline{\delta}^k)$, poser $t^k \triangleq t_{\mathtt{S}}^k$ et ne pas faire la `revealing` et la `poll` ;
révélation (notée `revealing`) :
poser $\mathcal{D}_{\mathtt{R}}^k \triangleq \operatorname{round}_{\underline{\delta}^k}^{\mathcal{M}}(\mathcal{D}_\ell)$ et $\mathcal{T}_{\mathtt{R}}^k \triangleq \{x^k\} + \mathcal{D}_{\mathtt{R}}^k$; si $\mathcal{T}_{\mathtt{R}}^k \cap \Omega \neq \emptyset$ et $t_{\mathtt{R}}^k \in \operatorname{argmin} f(\mathcal{T}_{\mathtt{R}}^k \cap \Omega)$ vérifie $f(t_{\mathtt{R}}^k) < f(x^k) - \rho(\underline{\delta}^k)$, poser $t^k \triangleq t_{\mathtt{R}}^k$ et ne pas faire la `poll` ;
sonde (notée `poll`) :
définir $\emptyset \neq \mathcal{D}_{\mathtt{P}}^k \subseteq \mathcal{M}_{\delta^k}(\underline{\delta}^k)$ et $\mathcal{T}_{\mathtt{P}}^k \triangleq \{x^k\} + \mathcal{D}_{\mathtt{P}}^k$; si $\mathcal{T}_{\mathtt{P}}^k \cap \Omega \neq \emptyset$ et $t_{\mathtt{P}}^k \in \operatorname{argmin} f(\mathcal{T}_{\mathtt{P}}^k \cap \Omega)$ vérifie $f(t_{\mathtt{P}}^k) < f(x^k) - \rho(\underline{\delta}^k)$, poser $t^k \triangleq t_{\mathtt{P}}^k$, sinon poser $t^k \triangleq x^k$;
mise à jour des paramètres :
si $t^k \neq x^k$, poser $\delta^{k+1} \in \delta^k \Upsilon$, sinon poser $\delta^{k+1} \in \delta^k \Lambda$; poser $\underline{\delta}^{k+1} \triangleq \min_{\ell \leq k+1} \delta^k$; poser $x^{k+1} \triangleq t^k$; si $\underline{\delta}^{k+1} < \underline{\delta}^k$, redéfinir $K^* \doteq K^* \cup \{k\}$ et $\ell \doteq \ell + 1$.

L'étape `revealing` intensifie fortement le long de directions de recherche données avant d'en changer. Par construction, les directions considérées ne changent que lorsque le pas de sonde atteint une valeur plus faible qu'à toutes les itérations précédentes.

3.3.2 `rDSM` garantit des conditions nécessaires d'optimalité

L'intégralité de l'analyse de convergence de `DSM` s'applique également à `rDSM`. En effet, l'étape `revealing` peut être vue comme une `search` particulière. Cependant, l'étape `revealing` permet de renforcer l'analyse en garantissant que le pré-requis $f(x^k) \to f(x^*)$ demandé par le théorème 3.1 est vérifié ; sous une légère adaptation de l'hypothèse 3.1 (pour des raisons techniques discutées ci-après) et une nouvelle hypothèse sur le comportement de `rDSM`. Notre analyse ne tient que si `rDSM` génère exactement un point raffiné. Ces deux nouvelles hypothèses sont données ci-dessous.

Pour poser l'altération de l'hypothèse 3.1, nous proposons une simplification de l'ECP.

Définition 3.2 (Propriété de cône intérieur (*Interior Cone Property*, ICP) et définition équivalente de l'ECP). Un ensemble $\mathcal{S} \subseteq \mathbb{R}^n$ a la propriété de cône intérieur (ICP) si

$$\forall x \in \mathrm{cl}(\mathcal{S}), \quad \exists \begin{cases} \mathcal{U} \text{ un ouvert non vide de } \mathbb{S}^n \\ \mathcal{K} \triangleq \mathbb{R}_+^* \mathcal{U} \text{ le cône généré par } \mathcal{U}, \quad \mathcal{K}_x \triangleq \{x\} + \mathcal{K} \\ \mathcal{O} \text{ un ouvert de } \mathbb{R}^n \text{ contenant } 0, \quad \mathcal{O}_x \triangleq \{x\} + \mathcal{O} \end{cases} : \quad (\mathcal{K}_x \cap \mathcal{O}_x) \subseteq \mathcal{S}.$$

De même, $\mathcal{S}$ a l'ECP si et seulement si $\mathbb{R}^n \setminus \mathcal{S}$ a l'ICP.

Hypothèse 3.2. **[Adaptation de l'hypothèse 3.1]** Pour tout $x \in X \cap \Omega$, il existe une partition $\mathcal{B}_R(x) = \sqcup_{i=0}^N \mathcal{X}_i$ (où $R \in]0, r]$ et $N \in \mathbb{N}^*$ est fini), avec $\mathcal{X}_0 \triangleq (\mathcal{B}_R(x) \setminus (X \cup \Omega))$, et vérifiant, pour tout $i \in [\![1, N]\!]$, $\mathcal{X}_i$ a l'ICP et $f_{|\mathcal{X}_i}$ est Lipschitz-continue.

Hypothèse 3.3. La suite $(x^k)_{k\in\mathbb{N}}$ générée par `rDSM` converge vers un point $x^* \in X$.

L'hypothèse 3.2 est très proche de l'hypothèse 3.1. Les différences sont une restriction du voisinage ($\mathcal{B}_R(x)$ au lieu d'un voisinage quelconque), le fait que $\mathrm{int}(\mathcal{X}_i)$ n'est plus imposé, et l'application de l'ICP à chacun des $\mathcal{X}_i$ au lieu de l'ECP. La première est nécessaire car un voisinage quelconque peut ne pas être un ouvert (c'est simplement un ensemble contenant un ouvert contenant x), ce qui aurait pu casser les preuves à venir. La seconde vient du fait que $\mathrm{int}(\mathcal{X}_i) \neq \emptyset$ pour tout $i \in [\![1, N]\!]$ est une conséquence de l'ICP de $\mathcal{X}_i$.[2] Enfin, nous imposons l'ICP à $\mathcal{X}_i$ au lieu de l'ECP car nous avons constaté qu'une interprétation dégénérée de l'hypothèse 3.1 pouvait accepter des discontinuités qu'on aurait dû rejeter.[3] Ceci n'est plus possible avec notre correctif, et nous prouvons dans le lemme 3.8 que le changement de l'ECP de la définition 3.1 vers l'ICP de la définition 3.2 est transparent.

L'hypothèse 3.3 n'est généralement pas restrictive en pratique, car le comportement de `rDSM` le plus plausible est la convergence de la suite des solutions courantes vers un point du domaine effectif de f. Le contre-exemple introduit au chapitre 3.3.3 vérifie également cette hypothèse. Nous prouvons que sous cette hypothèse, l'ensemble K^* généré par `rDSM` indexe une suite raffinante. Ceci permet d'affirmer les propriétés suivantes.

2. La définition appliquée en n'importe quel $x \in \mathrm{cl}(\mathcal{X}_i)$ donne $\mathcal{K}_x \cap \mathcal{O}_x$ ouvert inclus dans $\mathcal{X}_i$.

3. Prenons $f : x \in \mathbb{R}^2 \mapsto 0$ si $x \in \mathrm{epi}(\sqrt{|\cdot|})$ et 1 sinon. Les ensembles de continuité de f sont $\mathcal{X}_1 \triangleq \mathrm{epi}(\sqrt{|\cdot|})$ et $\mathcal{X}_2 \triangleq \mathbb{R}^2 \setminus \mathcal{X}_1$. Une telle discontinuité est censée être rejetée car $\mathcal{X}_2$ n'a pas l'ECP en l'origine. Or, on peut re-partitionner $\mathcal{X}_2$ en $\mathcal{X}_2 = \mathcal{X}_{2,m} \sqcup \mathcal{X}_{2,p}$ où $\mathcal{X}_{2,m} \triangleq \{(x_1, x_2) \in \mathcal{X}_2 : x_1 \leq 0\}$ et $\mathcal{X}_{2,p} \triangleq \{(x_1, x_2) \in \mathcal{X}_2 : x_1 > 0\}$. ainsi, $\mathcal{X}_1 \sqcup \mathcal{X}_{2,m} \sqcup \mathcal{X}_{2,p}$ forme toujours une partition d'un voisinage de l'origine en ensembles de continuité de f, et chacun des ensembles de partition possède l'ECP.

Lemme 3.5. **Généralisation de [8, lemme 4.12] à rDSM.** rDSM résolvant le problème (P) génère $(\mathcal{D}_{\mathtt{R}}^k)_{k\in\mathbb{N}}$ et $K^* \subseteq \mathbb{N}$ tels que $(\mathcal{D}_{\mathtt{R}}^k)_{k\in K^*}$ est dense dans $\mathrm{cl}(\mathcal{B}_r)$.

Démonstration. Notons $\ell : k \in K^* \mapsto \#(K^* \cap [\![0, k]\!])$, la fonction qui calcule la position de tout élément $k \in K^*$ dans l'ensemble K^* ordonné. On a $\ell(K^*) \triangleq \{\ell(k) : k \in K^*\} = \mathbb{N}$. De plus, par construction de rDSM, nous avons $\mathcal{D}_{\mathtt{R}}^k = \mathrm{round}_{\underline{\delta}^k}^{\mathcal{M}}(\mathcal{D}_{\ell(k)})$ pour tout $k \in K^*$, et $(\underline{\delta}^k)_{k\in K^*} \to 0$ par construction de K^*. Ainsi, $\sup_{d\in\mathcal{D}_{\mathtt{R}}^k,\ d'\in\mathcal{D}_{\ell(k)}} \|d' - d\| \to 0$ (c'est trivial dans le cas de rDSM à décroissance suffisante car round est alors l'identité, et prouvé de manière similaire à [8, lemme 4.12] dans le cas de rDSM à treillis adaptatif). Le résultat suit par densité de $(\mathcal{D}_\ell)_{\ell\in\mathbb{N}}$ dans $\mathrm{cl}(\mathcal{B}_r)$, qui est requise en préambule de rDSM. □

Remarquons que [8, lemme 4.12] comporte également une erreur, car il affirme le résultat pour n'importe quelle suite raffinante. Ceci est faux, pour la raison déjà donnée dans la remarque 3.1. Néanmoins, considérer la suite raffinante indexée par K^* évite le problème.

Lemme 3.6. Considérons la suite $(x^k; \delta^k; \mathcal{D}^k)_{k\in\mathbb{N}}$ et l'ensemble $K^* \subseteq \mathbb{N}$ générés par rDSM (où $\mathcal{D}^k \triangleq \mathcal{D}_{\mathtt{S}}^k \cup \mathcal{D}_{\mathtt{R}}^k \cup \mathcal{D}_{\mathtt{P}}^k$ pour tout $k \in \mathbb{N}$) résolvant le problème (P). Sous l'hypothèse 3.3, K^* indexe une suite raffinante de point raffiné x^*, d'ensemble D^* des directions raffinées dense dans $\mathbb{S}^n$, et d'ensemble $\mathcal{T}^* \triangleq \cup_{k\in K^*}\mathcal{T}^k$ (où $\mathcal{T}^k \triangleq \{x^k\} + \mathcal{D}^k$ pour tout $k \in \mathbb{N}$ est l'ensemble des points évalués à l'itération k) dense dans $\mathrm{cl}(\mathcal{B}_R(x^*))$.

Démonstration. Prouvons que K^* indexe une suite raffinante. Par construction, K^* ne contient que des indices d'itérations d'échec ; et $(x^k)_{k\in K^*}$ converge (via l'hypothèse 3.3) ; et $(\delta^k)_{k\in K^*}$ converge vers 0 car $\underline{\delta}^k \to 0$ ([17, théorème 8.1] et [35, théorème 7.1]) et pour tout $k \in K^*$, $\delta^k \leq \frac{1}{\inf \Upsilon}\underline{\delta}^k$; et la densité de D^* dans $\mathbb{S}^n$ découle directement du lemme 3.5. Ensuite, prouvons la densité de $\mathcal{T}^*$ dans $\mathrm{cl}(\mathcal{B}_R(x^*))$. Soit $x \in \mathrm{cl}(\mathcal{B}_R(x^*))$, $d \triangleq x - x^*$, et $K' \subseteq K^*$ tel qu'il existe $(d^k)_{k\in K'}$ convergeant vers d avec $d^k \in \mathcal{D}_{\mathtt{R}}^k$ pour tout $k \in K'$, et soit $t^k \triangleq x^k + d^k \in \mathcal{T}^*$ pour tout $k \in K'$. Alors $\|t^k - x\| \leq \|x^k - x^*\| + \|d^k - d\| \to 0$, ce qui conclut la preuve. □

On peut maintenant établir un correctif à l'énoncé 3.1 vérifiant tous les éléments qu'il voulait établir. On doit pour cela altérer l'hypothèse 3.1 en l'hypothèse 3.2 (cette altération est technique et non restrictive), et surtout supposer l'unicité du point raffiné via l'hypothèse 3.3 (là encore, peu restrictive en pratique). Mis en contrepartie, on gagne une définition explicite de la suite raffinante à considérer ; la suite indexée par K^*, qui est explicitement généré par rDSM. Aussi, l'hypothèse très restrictive discutée dans la remarque 3.1 n'est plus requise.

Théorème 3.7. Considérons la suite $(x^k; \delta^k; \mathcal{D}^k)_{k\in\mathbb{N}}$ et l'ensemble $K^* \subseteq \mathbb{N}$ générés par `rDSM` résolvant le problème (**P**). Sous l'hypothèse 3.3, K^* indexe une suite raffinante de point raffiné $x^* \in X$ et d'ensemble D^* des directions raffinées dense dans $\mathbb{S}^n$. Si de plus l'hypothèse 3.2 est vérifiée, alors en notant $\mathcal{B}_R(x^*) = \sqcup_{i=0}^N \mathcal{X}_i^*$ issus de son application en x^* et $\mathcal{X}^* \in \{\mathcal{X}_i,\ i \in [\![1, N]\!]\}$ tel que $x^* \in \mathcal{X}^*$, on a

1. soit $x^* \in \operatorname{int}(\mathcal{X}^*)$, et alors $f_{\mathsf{C}}^\circ(x^*; d) \geq 0$ pour tout $d \in \operatorname{cl}(\mathcal{H}(x^*))$;
2. soit $x^* \in \partial\mathcal{X}^*$, et alors $\mathcal{T}^* \triangleq \cup_{k\in K^*} \mathcal{T}^k$ est dense dans $\operatorname{int}(\mathcal{X}_i)$ pour tout $i \in [\![1, N]\!]$, et $(x^k)_{k\in K^*}$ admet une sous-suite incluse dans $\operatorname{cl}(\mathcal{X}^*)$, et $(f(x^k))_{k\in K^*} \to f(x^*)$.

Démonstration. Comme dans la preuve originelle de [116], cette preuve est restreinte au cas de `rDSM` à treillis adaptatif. L'adaptation au cas de `rDSM` à décroissance suffisante requiert de surmonter quelques subtilités techniques, mais n'en discutons pas ici car le théorème 3.7 est de toute manière remplacé par un résultat plus satisfaisant au chapitre 4.

Le cas $x^* \in \operatorname{int}(\mathcal{X}^*)$ est similaire aux études supposant la continuité de f. On se concentre donc sur le cas où $x^* \in \partial\mathcal{X}^*$. Rappelons que pour tout $i \in [\![1, N]\!]$, $\operatorname{int}(\mathcal{X}_i) \neq \emptyset$. Notons aussi que $x^* \in \operatorname{cl}(\operatorname{int}(\mathcal{X}^*))$ comme conséquence de l'ICP appliquée à $\mathcal{X}^*$ en x^*, car nous avons alors $x^* \in \operatorname{cl}(\mathcal{E}_{x^*} \cap \mathcal{B}_R(x^*))$ et $\mathcal{E}_{x^*} \cap \mathcal{B}_R(x^*) \subseteq \operatorname{int}(\mathcal{X}^*)$. Rappelons enfin que $\mathcal{X}^* \subseteq \Omega$ par construction dans l'hypothèse 3.2.

Le lemme 3.6 garantit que K^* indexe une suite raffinante, que $\mathcal{T}^*$ est dense dans $\operatorname{int}(\mathcal{X}_i)$ pour tout $i \in [\![1, N]\!]$, et que D^* est dense dans $\mathbb{S}^n$. Ainsi, il ne reste qu'à prouver l'existence de $f^* \triangleq \lim_{k\in K^*} f(x^k)$ et l'égalité $f^* = f(x^*)$, même lorsque $N \neq 2$. D'abord, $(f(x^k))_{k\in K^*}$ est décroissante et bornée, donc elle converge, donc f^* existe. Ensuite, $f^* \geq f(x^*)$ par semi-continuité de f. Enfin, supposons que $f^* > f(x^*)$, pour établir une contradiction. On a $f(x^k) \geq f(x^*) + \varepsilon$ pour un certain $\varepsilon > 0$ et tout $k \in K^*$. Alors $\inf f(\mathcal{T}^* \cap \Omega) \geq f(x^*) + \varepsilon$, car $f(x) \geq f(x^k)$ pour tout $k \in K^*$ et $x \in \mathcal{T}^k \cap \Omega$. Ainsi, $\inf f(\mathcal{T}^* \cap \operatorname{int}(\mathcal{X}^*)) \geq f(x^*) + \varepsilon$. Définissons enfin une suite $(y_\ell)_{\ell\in\mathbb{N}}$ d'éléments de $\mathcal{T}^* \cap \operatorname{int}(\mathcal{X}^*)$ convergeant vers x^*, ce qui est possible car $x^* \in \operatorname{cl}(\operatorname{int}(\mathcal{X}^*))$ et $\mathcal{T}^*$ est dense dans $\operatorname{int}(\mathcal{X}^*)$. La contradiction apparaît alors comme suit : $\liminf_{\ell\in\mathbb{N}} f(y_\ell) \geq f(x^*) + \varepsilon$ alors que $f_{|\mathcal{X}^*}$ est continue en x^*. □

Le théorème 3.7 garantit $\lim_{k\in K^*} f(x^k) = f(x^*)$. Ainsi, le théorème 3.1 est applicable en x^*, ce qui certifie que x^* vérifie des conditions nécessaires d'optimalité. Le théorème 3.7 affirme également que `rDSM` sonde chaque ensemble de continuité de f au voisinage de x^*, pour n'importe quelle valeur de N alors que l'énoncé 3.1 ne l'affirme que dans le cas $N = 2$. Enfin, Le théorème 3.7 assure la densité de l'ensemble des directions raffinées dans la sphère unité. Ceci est requis par le théorème 3.1 mais pas clamé par l'énoncé 3.1.

La principale restriction du théorème 3.7 est finalement sa dépendance en l'hypothèse 3.3. Cependant, celle-ci est vérifiée dans une grande majorité de cas pratiques (à notre connaissance, seuls des contre-exemples théoriques construits *ad hoc* ont déjà mené à `DSM` générant plus d'un point raffiné ; ou à un point raffiné en lequel f renvoie $+\infty$).

Terminons cette analyse par la preuve que notre nouvelle définition de l'ECP (comme complémentaire à notre définition de l'ICP dans la définition 3.2) est bien équivalente à la définition de l'ECP proposée par [116].

Lemme 3.8. L'ECP de la définition 3.1 est équivalente à l'ECP de la définition 3.2.

Démonstration. Notons (ECP) la propriété de cône extérieur telle que posée dans la définition 3.2, et $[ECP]$ la propriété de cône extérieur telle qu'introduite par [116] (rappelée en définition 3.1). Soit $\mathcal{S} \subseteq \mathbb{R}^n$. Notons $\mathcal{S}^c \triangleq \mathbb{R}^n \setminus \mathcal{S}$. Pour tout $(x, y) \in (\mathbb{R}^n)^2$, on note $[x, y] \triangleq \{x + t(y - x) : t \in [0, 1]\}$ le segment reliant x et y. On note également $\Theta(\cdot, \cdot)$ l'angle saillant entre deux vecteurs.

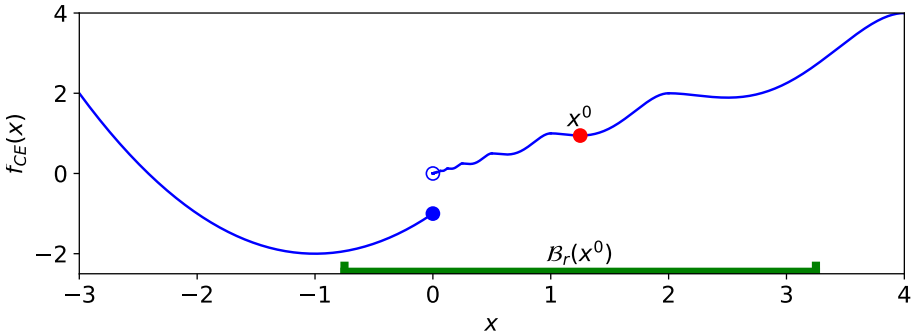
Supposons que $\mathcal{S}$ a l'$[ECP]$. On remarque qu'il est clair que l'(ICP) d'un ensemble tient en tout point de l'intérieur de cet ensemble, ainsi $\mathcal{S}^c$ a l'(ICP) en tout point de son intérieur. Par conséquent, pour prouver que $\mathcal{S}$ a l'(ECP), il suffit de montrer que $\mathcal{S}^c$ a également l'(ICP) en tout point de sa frontière. Soit $x \in \partial\mathcal{S}^c$ et considérons le cône $\mathcal{K}$, le voisinage $\mathcal{O}$ et l'ensemble $\mathcal{E}_x$ issus de l'application de l'$[ECP]$ de $\mathcal{S}$ en x. Soit $\mathcal{U} \triangleq \operatorname{int}(\mathcal{K} \cap \mathbb{S}^n) \neq \emptyset$ (pour la topologie de $\mathbb{S}^n$) et $\mathcal{K}' \triangleq \mathbb{R}_+^* \mathcal{U}$. Alors $\mathcal{S}^c$ a l'(ICP) en x, en considérant $\mathcal{U}$, $\mathcal{O}$ et $\mathcal{K}'$ dans la définition, puisque $(\mathcal{K}'_x \cap \mathcal{O}_x) \subseteq \mathcal{E}_x \subseteq \mathcal{S}^c$ par construction. Ainsi, l'(ICP) de $\mathcal{S}^c$ est avérée en tout $x \in \partial\mathcal{S}^c$. Ainsi, $\mathcal{S}^c$ a l'(ICP), et donc $\mathcal{S}$ a l'(ECP) comme désiré.

Supposons à la place que $\mathcal{S}$ a l'(ECP), et donc que $\mathcal{S}^c$ a l'(ICP). Soit $x \in \partial\mathcal{S}^c = \partial\mathcal{S}$, et considérons l'ensemble $\mathcal{U}$ et le voisinage $\mathcal{O}$ issus de l'application de l'(ICP) de $\mathcal{S}^c$ en x. Sans perte de généralité, supposons que $\mathcal{U} \neq \mathbb{S}^n$, de sorte que $\partial\mathcal{U} \neq \emptyset$ dans la topologie induite par $\mathbb{S}^n$. Soit $\varepsilon \in \mathbb{R}_+^*$, suffisamment petit pour que $\mathcal{U}' \triangleq \{u \in \mathcal{U} : \Theta(u, v) \geq \varepsilon,\ \forall v \in \partial\mathcal{U}\} \neq \emptyset$. Soit $\mathcal{K}' \triangleq \mathbb{R}_+^* \mathcal{U}'$. Soit $\mathcal{O}_x \triangleq \{x\} + \mathcal{O}$ et $\mathcal{K}'_x \triangleq \{x\} + \mathcal{K}'$ et $\mathcal{E}_x \triangleq (\mathcal{O}_x \cap \mathcal{K}'_x)$ et $\mathcal{S}_x^c \triangleq \mathcal{S}^c \cap \mathcal{O}_x$. D'abord, on constate que $\mathcal{E}_x \subseteq A^c$ par construction. Ensuite, pour tout $(e, s) \in (\mathcal{E}_x \setminus \{x\}) \times (\mathcal{S}_x^c \setminus \{x\})$, on définit $y \in \partial\mathcal{K}'_x \cap [e, s]$, pour arriver à $\Theta(e - x, s - x) = \Theta(e - x, y - x) + \Theta(y - x, s - x) = \Theta(\frac{e-x}{\|e-x\|}, \frac{y-x}{\|y-x\|}) + \Theta(y - x, s - x) \geq \varepsilon + 0$. Ainsi, $\mathcal{S}$ a l'$[ECP]$ en x, en considérant $\mathcal{K}'$, $\theta_x \triangleq \varepsilon$, et $\mathcal{O}$. Cela prouve que l'(ECP) implique l'$[ECP]$. $\square$

3.3.3 rDSM répond au contre-exemple

Terminons ce chapitre en nous assurant que rDSM répond au contre-exemple construit au chapitre 3.2.1. La figure 3.5 illustre le cas où le paramètre r est choisi à $r \triangleq 2$. On voit que dès la première itération, il y a une possibilité que la revealing génère un point permettant à rDSM de franchir la discontinuité en $x = 0$. La proposition 3.9 formalise cette affirmation. Le cas $r \in \mathbb{R}_+^*$ quelconque pourrait également être prouvé, mais la preuve est technique et son intérêt est limité.

FIGURE 3.5 Itération 0 de rDSM résolvant le contre-exemple avec $r = 2$.



Proposition 3.9. Soit l'instance de DSM du chapitre 3.3.3 résolvant le problème ($\mathbf{P}_{\mathtt{CE}}$), et étendue en une instance de rDSM via une sonde revealing de rayon $r \triangleq 2$. Alors l'algorithme génère le point raffiné $x^* \triangleq -1$, le minimum global de $f_{\mathtt{CE}}$.

Démonstration. Remarquons que $f_{\mathtt{CE}}(x) \leq 0$ si et seulement si $x \in I \triangleq [-\sqrt{2}-1, 0]$. Ainsi, s'il existe $\kappa \in \mathbb{N}$ tel que $x^\kappa \in I$, alors toutes les solutions courantes x^k avec $k \geq \kappa$ sont aussi dans I. Par conséquent, $(x^k)_{k \geq \kappa}$ converge nécessairement vers $x^* = -1$, car il s'agit du seul point de I vérifiant des conditions d'optimalité. Il suffit donc de prouver qu'un tel indice κ existe. Or, si κ n'existe pas, alors par construction la suite raffinante indexée par K^* est telle que $(x^k)_{k \in K^*}$ reste confinée dans l'intervalle $J \triangleq]0, x^0] =]0, \frac{5}{4}]$. Or, pour tout $x \in J$, on a $\mathrm{cl}(\mathcal{B}_r(x)) \cap I \neq \emptyset$. Par conséquent, par construction, il existe $\kappa \in \mathbb{N}$ tel que l'étape revealing génère un point dans I à l'itération κ, ce qui mène à une contradiction. $\square$

CHAPITRE 4 SIMPLIFICATION DU CADRE THÉORIQUE ET AMÉLIORATION DES RÉSULTATS DE CONVERGENCE DE rDSM

- Mais alors, osai-je commenter, vous êtes encore loin de la solution ...

- J'en suis très près, dit Guillaume, je ne sais simplement pas de laquelle.

« Le nom de la rose », Umberto Eco.

L'algorithme rDSM défini au chapitre 3 est une première extension de DSM permettant de résoudre le problème (**P**). Comme nous l'avons montré, rDSM génère au moins un point raffiné, et s'il y en a exactement un, ce point vérifie des conditions nécessaires d'optimalité pour le problème (**P**). Ce résultat étend naturellement les théorèmes habituels issus de la littérature sur le cas continu. Cependant, cette analyse de rDSM est incomplète. Nous montrons au chapitre 4.1 que sous l'hypothèse d'unicité du point raffiné, ce point est en réalité une solution locale au problème (**P**). Nous proposons ensuite une extension de rDSM pour supprimer le besoin d'unicité du point raffiné. Le nouvel algorithme (cDSM, covering DSM) remplace l'étape revealing par une nouvelle étape nommée covering, et est formalisé au chapitre 4.2. Nous montrons également que cDSM ne requiert que des hypothèses strictement plus faibles que celles requises par « Analysis of direct searches for discontinuous functions » et par rDSM pour obtenir un résultat strictement plus fort : tous les points raffinés générés par cDSM sont des solutions locales au problème (**P**), dans un sens généralisé traitant également le cas des points hors de X.

4.1 Complément à l'analyse de convergence de rDSM

Rappelons le cadre requis pour établir le théorème 3.7 (prouvant que le point raffiné de rDSM vérifie des conditions nécessaires d'optimalité).

Cadre 4.1. **Cadre d'étude de rDSM.** Considérons le problème (**P**) dans le cadre 1.1, résolu par rDSM. On suppose que la suite $(x^k)_{k\in\mathbb{N}}$ générée par rDSM converge vers un point $x^* \in X$ (hypothèse 3.3). On suppose aussi qu'il existe $R \in \,]0, r]$ tel que $\mathcal{B}_R(x^*)$ admet une partition $\mathcal{B}_R(x^*) = \sqcup_{i=0}^{N} \mathcal{X}_i$, où $\mathcal{X}_0 \triangleq (\mathcal{B}_R(x^*) \setminus (X \cup \Omega))$, $N \in \mathbb{N}^*$, et pour tout $i \in [\![1, N]\!]$, $\mathcal{X}_i$ a l'ICP et $f_{|\mathcal{X}_i}$ est Lipschitz-continue (hypothèse 3.2 en x^*).

Dans ce cadre, il est possible d'arriver à une conclusion plus forte que celle énoncée par le théorème 3.7. Ce dernier dit que x^* vérifie des conditions nécessaires d'optimalité. En comparaison, le chapitre 4.1.1 établit que x^* est en réalité une solution locale au problème (**P**). Nous discutons ensuite des aspects restrictifs de ce cadre au chapitre 4.1.2.

4.1.1 Optimalité locale sous l'hypothèse d'unicité du point raffiné

Le lemme 3.6 établit que l'ensemble $\mathcal{T}^* \triangleq \cup_{k\in K^*}\mathcal{T}^k$ est dense dans $\mathcal{B}_R(x^*)$, où pour tout $k \in \mathbb{N}$, $\mathcal{T}^k$ est l'ensemble des points évalués par `rDSM` à l'itération k. Cependant, on peut aussi en déduire la densité de $\mathcal{T}^*$ dans chacun des ensembles de continuité de f au voisinage de x^*. Ceci n'a pas été énoncé au chapitre 3. Le lemme 4.1 formalise un résultat technique proche de cette affirmation.

Lemme 4.1. Dans le cadre 4.1, $\mathcal{X}_i \subseteq \mathrm{cl}(\mathcal{T}^* \cap \mathcal{X}_i \setminus \{x\})$ pour tout $i \in [\![1, N]\!]$ et tout $x \in \mathcal{X}_i$.

Démonstration. Soit $i \in [\![1, N]\!]$ et $x \in \mathcal{X}_i$. Le lemme 3.6 implique que $\mathcal{E} \subseteq \mathrm{cl}(\mathcal{T}^* \cap \mathcal{E})$ pour tout ouvert $\mathcal{E} \subseteq \mathcal{B}_R(x^*)$. Or, l'ICP appliquée à $\mathcal{X}_i$ en x donne l'ouvert $\mathcal{E}_x \subseteq \mathcal{X}_i$ vérifiant $x \in \mathrm{cl}(\mathcal{E}_x)$. Ainsi, il existe un ouvert $\mathcal{E}'_x \subseteq \mathcal{E}_x$ tel que $x \in \mathrm{cl}(\mathcal{E}'_x) \setminus \mathcal{E}'_x$, et on en déduit $x \in \mathrm{cl}(\mathcal{T}^* \cap \mathcal{E}'_x) \subseteq \mathrm{cl}(\mathcal{T}^* \cap \mathcal{X}_i)$ comme conséquence du lemme 3.6. □

On peut alors en déduire un théorème remplaçant le théorème 3.7.

Théorème 4.2. Dans le cadre d'étude 4.1, x^* est une solution locale au problème (**P**).

Démonstration. Nous avons bien $x^* \in \Omega$, comme limite de la suite $(x^k)_{k\in\mathbb{N}}$ d'éléments du fermé Ω. Prouvons alors que $f(x) \geq f(x^*)$ pour tout $x \in \mathcal{B}_R(x^*) \cap \Omega$. Ceci est trivial pour tout $x \notin X$, donc considérons $x \in \mathcal{X}_i \subseteq \Omega$ pour un certain $i \in [\![1, N]\!]$. Par le lemme 4.1, il existe une suite $(y_\ell)_{\ell\in\mathbb{N}}$ d'éléments de $\mathcal{T}^* \cap \mathcal{X}_i \setminus \{x\}$ convergeant vers x. Notons, pour tout $\ell \in \mathbb{N}$, $\kappa(\ell) \in K^*$ tel que $y_\ell \in \mathcal{T}^{\kappa(\ell)}$, qui indexe une itération d'échec par construction de K^*. Par conséquent, on a $f(y_\ell) \geq f(x^{\kappa(\ell)}) - \rho(\underline{\delta}^{\kappa(\ell)})$ pour tout $\ell \in \mathbb{N}$. De plus, $(f(y_\ell))_{\ell\in\mathbb{N}} \to f(x)$ par continuité de $f_{|\mathcal{X}_i}$. Enfin, $\kappa(\ell) \to +\infty$ puisque $\mathcal{T}^k$ est fini pour tout $k \in \mathbb{N}$, et donc $(x^{\kappa(\ell)})_{\ell\in\mathbb{N}} \to x^*$ et $(\underline{\delta}^{\kappa(\ell)})_{\ell\in\mathbb{N}} \to 0$. On déduit de tout ceci que $f(x) = \lim_{\ell\in\mathbb{N}} f(y_\ell) \geq \lim_{\ell\in\mathbb{N}} f(x^{\kappa(\ell)}) - 0 \geq f(x^*)$ par semi-continuité de f. □

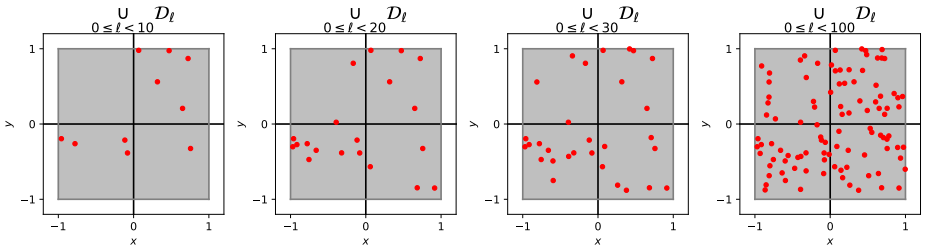
4.1.2 Aspects restrictifs dans le cadre d'étude de `rDSM`

Le cadre d'étude 4.1 est restrictif sur trois aspects. D'abord, en pratique, l'étape `revealing` est peu exploitable car peu compatible avec le critère d'arrêt usuel de `rDSM`. La construction de l'étape `revealing` ne peut considérer que peu de directions de recherche avant que le critère d'arrêt de `rDSM` ne soit atteint. Ensuite, l'hypothèse 3.3 est souvent avérée dans les cas réels mais est simple à briser dans la théorie. Enfin, l'ICP des ensembles de continuité de $f_{|\Omega}$ est difficilement vérifiable puisque f est souvent une boîte noire.

Faible nombre de directions considérées en cas d'interruption anticipée de `rDSM`

L'étape `revealing` traduit mal en pratique sa propriété de densité des directions de recherche dans $\mathrm{cl}(\mathcal{B}_r)$. Prenons une instance de `rDSM` avec $\delta^0 \triangleq 1$ et δ^k divisé par deux à chaque itération d'échec et doublé à chaque succès. Construisons l'étape `revealing` via une suite $(\mathcal{D}_\ell)_{\ell\in\mathbb{N}}$ de tirages indépendants de la distribution uniforme sur $\mathrm{cl}(\mathcal{B}_r)$, où $r \triangleq 1$. Pour tout $k \in \mathbb{N}$, l'étape `revealing` considère alors l'ensemble de directions $\mathcal{D}_{\mathtt{R}}^k \triangleq \mathcal{D}_{\ell(k)}$, où la suite $(\ell(k))_{k\in\mathbb{N}}$ est incrémentée lorsque δ^k atteint la première fois une nouvelle puissance négative de 2. Sous le critère d'arrêt « $\delta^k \leq 2^{-20}$ », l'étape `revealing` ne peut donc considérer que les 20 premiers éléments de $(\mathcal{D}_\ell)_{\ell\in\mathbb{N}}$ avant l'interruption de l'algorithme. Or, ceci n'est pas suffisant pour recouvrir $\mathrm{cl}(\mathcal{B}_r)$ de façon satisfaisante. Même 30 voire 100 tirages restent insuffisants, comme illustré en figure 4.1. En général, pour pouvoir considérer L éléments de $(\mathcal{D}_\ell)_{\ell\in\mathbb{N}}$, il faut baisser le critère d'arrêt à « $\delta^k \leq 2^{-L}$ », alors qu'un choix de $L > 6$ est prohibé car il mène usuellement à une exécution très longue et en grande partie inutile.

FIGURE 4.1 Répartition de tirages uniformes indépendants dans la boule unité.



Absence de densité des points évalués en présence de plusieurs points raffinés

Lorsque `rDSM` génère plusieurs points raffinés, la densité de l'ensemble des points évalués autour de chacun n'est plus garantie. Prouvons ceci via l'exemple suivant, illustré en figure 4.2.

Exemple 4.1. Considérons $f \triangleq \|\cdot\|_\infty$ dans $\mathbb{R}^2$. L'instance de `rDSM` part de $x^0 \triangleq -3\mathbb{1}$, où $\mathbb{1} \triangleq (1,1)$, avec $\delta^0 \triangleq 1$ et $\Lambda \triangleq \{\frac{1}{2}\}$ et $\Upsilon \triangleq \{1\}$. L'étape de `search` est exécutée aux itérations $k = 3q \in 3\mathbb{N}$, pour évaluer $t_{\mathtt{S}}^{3q} \triangleq (-1)^q\,(1+2^{-q})\,\mathbb{1}$. L'étape de `poll` considère l'ensemble de points $\mathcal{T}_{\mathtt{P}}^k \triangleq \{x^k\} + \{(\pm\delta^k, 0), (0, \pm\delta^k)\}$ pour tout $k \in \mathbb{N}$. L'étape `revealing` repose sur une suite $(\mathcal{D}_\ell)_{\ell\in\mathbb{N}}$ dense dans $\mathrm{cl}(\mathcal{B}_r)$ où $r \triangleq 1$ telle que, pour tout $q \in \mathbb{N}$, $\mathcal{D}_{4q} \subset \mathbb{R}_+ \times \mathbb{R}_+$, $\mathcal{D}_{4q+1} \subset \mathbb{R}_- \times \mathbb{R}_+$, $\mathcal{D}_{4q+2} \subset \mathbb{R}_- \times \mathbb{R}_-$, $\mathcal{D}_{4q+3} \subset \mathbb{R}_+ \times \mathbb{R}_-$.

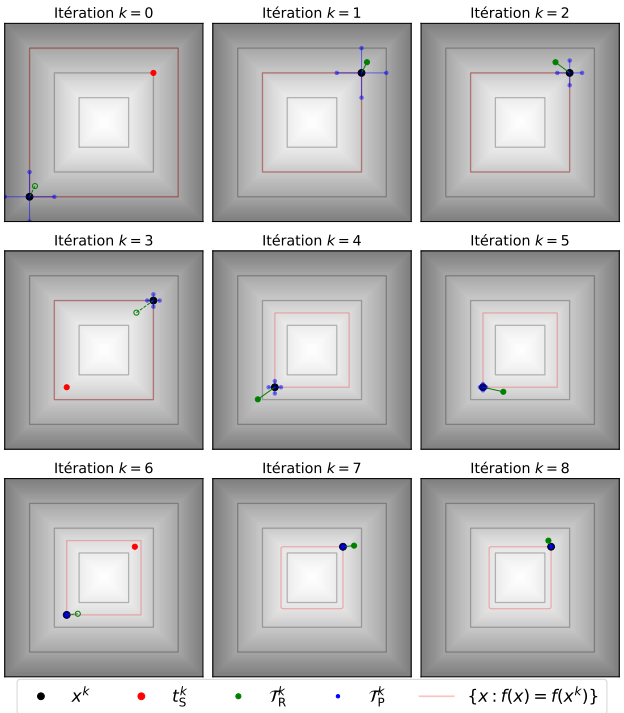
Par construction, `rDSM` a ici le comportement suivant, pour tout $q \in \mathbb{N}$.[1]

$$\begin{cases} x^{3q} &= (-1)^{q-1}\left(1+2^{-(q-1)}\right)\mathbb{1}, & \delta^{3q} &= 4^{-q}, & \mathcal{D}_{\mathtt{R}}^{3q} &= \mathcal{D}_{2q}, & \text{succès de } \mathtt{search}, \\ x^{3q+1} &= (-1)^{q}\quad(1+2^{-q})\,\mathbb{1}, & \delta^{3q+1} &= 4^{-q}, & \mathcal{D}_{\mathtt{R}}^{3q+1} &= \mathcal{D}_{2q}, & \text{itération d'échec}, \\ x^{3q+2} &= (-1)^{q}\quad(1+2^{-q})\,\mathbb{1}, & \delta^{3q+2} &= \tfrac{1}{2}4^{-q}, & \mathcal{D}_{\mathtt{R}}^{3q+2} &= \mathcal{D}_{2q+1}, & \text{itération d'échec}. \end{cases}$$

Ici, `rDSM` génère deux points raffinés, $x_+^* \triangleq \mathbb{1}$ et $x_-^* \triangleq -\mathbb{1}$, et aucun ne minimise $\|\cdot\|_\infty$. L'étape de `search` à toute itération $k \in 3\mathbb{N}$ est un succès, et donc seule la `search` est effectuée à ces itérations. Comme la boule $\mathrm{cl}(\mathcal{B}_r(x_+^*))$ égale l'ensemble $[0,2]^2$, les seuls points évalués dans $\mathrm{cl}(\mathcal{B}_r(x_+^*))$ sont les points de `search` aux itérations $\{3q : q \text{ pair}\}$, les points de `poll` aux itérations $\{3q+1, 3q+2 : q \text{ pair}\}$, et les points de `revealing` aux itérations $\{3q+1, 3q+2 : q \text{ pair}\}$. L'analyse pour x_-^* est symétrique. On a donc

$$\mathrm{cl}\left(\bigcup_{k\in\mathbb{N}} \mathcal{T}^k\right) \cap \left\{x \in \mathrm{cl}(\mathcal{B}_r(x_+^*)) : x - x_+^* \in \mathbb{R}_- \times \mathbb{R}_-\right\} = \emptyset,$$
$$\mathrm{cl}\left(\bigcup_{k\in\mathbb{N}} \mathcal{T}^k\right) \cap \left\{x \in \mathrm{cl}(\mathcal{B}_r(x_-^*)) : x - x_-^* \in \mathbb{R}_+ \times \mathbb{R}_+\right\} = \emptyset.$$

FIGURE 4.2 Mise en échec de `rDSM` lorsque plusieurs points raffinés existent.



1. La preuve se fait par une récurrence simple, mais longue à rédiger et d'intérêt limité. Comme de coutume dans un tel cas, nous clamons que *la preuve est laissée comme exercice au lecteur*.

4.2 Extension de rDSM : étape de covering remplaçant l'étape revealing

L'algorithme rDSM peut être adapté pour surmonter le besoin d'unicité du point raffiné. Nous altérons l'étape revealing en une nouvelle étape que nous nommons covering. Il en résulte un nouvel algorithme (covering DSM, cDSM), introduit au chapitre 4.2.1. Nous donnons ensuite le cadre théorique le plus large possible pour garantir la convergence de cDSM, au chapitre 4.2.2. Enfin, nous donnons au chapitre 4.2.3 une analyse de convergence de cDSM dans ce cadre, qui étend le théorème 4.2 au cas où plusieurs points raffinés existent.

4.2.1 Algorithme cDSM

Expression de l'algorithme cDSM

L'algorithme cDSM est formalisé comme suit. Il consiste en une altération de rDSM, où l'étape revealing est remplacée par l'étape covering. Le nouveau paramètre en entrée, l'*oracle de covering* $\mathbb{O}$, est discuté plus loin dans ce chapitre.

Algorithme 3 cDSM générique pour résoudre le problème (**P**), généralisant rDSM.

Initialisation :
initialiser $(x^0; \delta^0) \in (X \cap \Omega) \times \mathbb{R}_+^*$, et poser $\underline{\delta}^0 \triangleq \delta^0$;
définir $\rho : \mathbb{R}_+ \to \mathbb{R}_+$ et $\mathcal{M} : \mathbb{R}_+ \to 2^{\mathbb{R}^n}$ et $\Lambda \subset]0,1[$ et $\Upsilon \subset [1,+\infty[$ suivant, au choix,
— *DSM à treillis adaptatif* : $\rho(\nu) \triangleq 0$ et $\mathcal{M}(\nu) \triangleq \min\{\nu, \frac{\nu^2}{\delta^0}\} M\mathbb{N}^p$ pour tout $\nu \in \mathbb{R}_+$, où $M \in \mathbb{R}^{n\times p}$ est une base positive de $\mathbb{R}^n$ (avec $p > n$), et $\Lambda \subseteq \{\tau^\ell : \ell \in [\![1, m]\!]\}$ et $\Upsilon \subseteq \{\tau^\ell : \ell \in [\![-m, 0]\!]\}$, où $\tau \in \mathbb{Q} \cap]0,1[$ et $m \in \mathbb{N}^*$;
— *DSM à décroissance suffisante* : ρ nulle en 0 et strictement croissante et $\mathcal{M}(\nu) \triangleq \mathbb{R}^n$ pour tout $\nu \in \mathbb{R}_+$, et $\Lambda \subseteq [\underline{\lambda}, \overline{\lambda}]$ et $\Upsilon \subseteq [\underline{\upsilon}, \overline{\upsilon}]$, où $0 < \underline{\lambda} \leq \overline{\lambda} < 1 \leq \underline{\upsilon} \leq \overline{\upsilon} < +\infty$;
définir $r \in \mathbb{R}_+^*$ et $\mathbb{O}$ un oracle de covering (voir définition 4.1), et poser $\mathcal{T} \triangleq \emptyset$;

pour $k \in \mathbb{N}$ **faire** :
exploration (notée search en anglais) :
définir $\mathcal{D}_{\mathsf{S}}^k \subseteq \mathcal{M}(\underline{\delta}^k)$ et $\mathcal{T}_{\mathsf{S}}^k \triangleq \{x^k\} + \mathcal{D}_{\mathsf{S}}^k$; si $\mathcal{T}_{\mathsf{S}}^k \cap \Omega \neq \emptyset$ et $t_{\mathsf{S}}^k \in \operatorname{argmin} f(\mathcal{T}_{\mathsf{S}}^k \cap \Omega)$ vérifie $f(t_{\mathsf{S}}^k) < f(x^k) - \rho(\underline{\delta}^k)$, poser $t^k \triangleq t_{\mathsf{S}}^k$ et ne pas faire la covering et la poll ;
recouvrement (notée covering en anglais) :
poser $\mathcal{D}_{\mathsf{C}}^k \triangleq \mathbb{O}(x^k; \underline{\delta}^k; \mathcal{T})$ et $\mathcal{T}_{\mathsf{C}}^k \triangleq \{x^k\} + \mathcal{D}_{\mathsf{C}}^k$; si $\mathcal{T}_{\mathsf{C}}^k \cap \Omega \neq \emptyset$ et $t_{\mathsf{C}}^k \in \operatorname{argmin} f(\mathcal{T}_{\mathsf{C}}^k \cap \Omega)$ vérifie $f(t_{\mathsf{C}}^k) < f(x^k) - \rho(\underline{\delta}^k)$, poser $t^k \triangleq t_{\mathsf{C}}^k$ et ne pas faire la poll ;
sonde (notée poll en anglais) :
définir $\emptyset \neq \mathcal{D}_{\mathsf{P}}^k \subseteq \mathcal{M}_{\delta^k}(\underline{\delta}^k)$ et $\mathcal{T}_{\mathsf{P}}^k \triangleq \{x^k\} + \mathcal{D}_{\mathsf{P}}^k$; si $\mathcal{T}_{\mathsf{P}}^k \cap \Omega \neq \emptyset$ et $t_{\mathsf{P}}^k \in \operatorname{argmin} f(\mathcal{T}_{\mathsf{P}}^k \cap \Omega)$ vérifie $f(t_{\mathsf{P}}^k) < f(x^k) - \rho(\underline{\delta}^k)$, poser $t^k \triangleq t_{\mathsf{P}}^k$, sinon poser $t^k \triangleq x^k$;
mise à jour des paramètres :
si $t^k \neq x^k$, poser $\delta^{k+1} \in \delta^k \Upsilon$, sinon poser $\delta^{k+1} \in \delta^k \Lambda$; poser $\underline{\delta}^{k+1} \triangleq \min_{\ell \leq k+1} \delta^k$; poser $x^{k+1} \triangleq t^k$; redéfinir $\mathcal{T} \doteq \mathcal{T} \cup \{x \in \mathcal{T}_{\mathsf{S}}^k \cup \mathcal{T}_{\mathsf{C}}^k \cup \mathcal{T}_{\mathsf{P}}^k : x \text{ a été évalué}\}$.

L'oracle de `covering` $\mathbb{O}$ garantit que l'ensemble des points évalués par `cDSM` est dense au voisinage de tout point raffiné. Des expressions pratiques de $\mathbb{O}$ suivent dans ce chapitre.

Propriété 4.3 (*dense covering property* garantie par l'étape `covering`). Pour tout point raffiné x^* généré par `cDSM`, l'ensemble $\mathcal{T}$ vérifie

$$\mathcal{B}_r(x^*) \subseteq \mathrm{cl}\,(\mathcal{T})\,. \tag{DCP}$$

Définition d'un oracle de `covering`

La véracité de la propriété 4.3 ne peut être observée qu'à posteriori, après une exécution d'une infinité d'itérations de `cDSM`. Cependant, il est possible de garantir à priori (avant l'exécution de `cDSM`) que l'algorithme la vérifiera à posteriori. Nous formalisons ceci dans la proposition 4.4, qui dépend de la structure d'*oracle de* ***`covering`*** définie comme suit.

Définition 4.1 (oracle de `covering`). Pour $\mathcal{M} : \mathbb{R}_+ \to 2^{\mathbb{R}^n}$ et $r > 0$ donnés, une fonction $\mathbb{O} : \mathbb{R}^n \times \mathbb{R}_+^* \times 2^{\mathbb{R}^n} \to 2^{\mathbb{R}^n}$ est un *oracle de* `covering` si $\mathbb{O}(y;\nu;\mathcal{S}) \subseteq \mathcal{M}_r(\nu)$ pour tout point $y \in \mathbb{R}^n$ et tout rayon $\nu \in \mathbb{R}_+^*$ et tout ensemble $\mathcal{S} \subseteq \mathbb{R}^n$, et si

$$\lim_{k\in\mathbb{N}} \max_{d\in\mathrm{cl}(\mathcal{B}_r)} \mathrm{dist}\left(y^k + d, \bigcup_{\ell<k} \mathcal{S}^\ell\right) = 0$$

pour toute suite $(y^k;\nu^k)_{k\in\mathbb{N}}$ de $\mathbb{R}^n \times \mathbb{R}_+^*$ avec $(y^k)_{k\in\mathbb{N}}$ convergente et $\nu^k \to 0$, où $\mathcal{S}^{-1} \subseteq \mathbb{R}^n$ et $\mathcal{S}^k \triangleq \{y^k\} + \mathbb{O}(y^k;\nu^k;\mathcal{S}^{k-1})$ pour tout $k \in \mathbb{N}$ et où $\mathrm{dist}(x,\mathcal{S}) \triangleq \inf_{a\in\mathcal{S}} \|x-a\|$ pour tout $x \in \mathbb{R}^n$ et $\mathcal{S} \subseteq \mathbb{R}^n$.

Proposition 4.4. Si $\mathbb{O}$ satisfait la définition 4.1 et si `cDSM` a son étape de `covering` construite suivant le schéma

$$\forall k \in \mathbb{N}, \qquad \mathcal{D}_{\mathtt{C}}^k \triangleq \mathbb{O}\left(x^k, \underline{\delta}^k, \bigcup_{\ell<k} \mathcal{T}^\ell\right),$$

où $\mathcal{T}^\ell$ est l'ensemble des points évalués à l'itération ℓ, pour tout $\ell \in \mathbb{N}$, alors l'exécution de `cDSM` satisfait la propriété 4.3.

Démonstration. Nous notons ici $\mathcal{M}_r(\nu;y) \triangleq \{y\} + \mathcal{M}_r(\nu)$ pour tout $(\nu;y) \in \mathbb{R}_+^* \times \mathbb{R}^n$. On se place dans le cadre d'étude de la proposition 4.4. Soit $K^* \subseteq \mathbb{N}$ indexant une suite raffinante générée par `cDSM`, et x^* le point raffiné associé. Prouvons que $\mathcal{T} \cap \mathcal{B}_\varepsilon(x) \neq \emptyset$, pour tout $x \in \mathcal{B}_r(x^*)$ et tout $\varepsilon > 0$. Soit $x \in \mathcal{B}_r(x^*)$ et $\varepsilon > 0$. On exploite alors la définition d'un oracle de `covering` avec la suite $(x^k;\underline{\delta}^k;\mathcal{T}_{\mathtt{C}}^k)_{k\in K^*}$ en lieu de $(y^k;\underline{\delta}^k;\mathcal{S}^k)_{k\in\mathbb{N}}$.

Nous obtenons alors $\operatorname{dist}(\mathcal{M}_r(\underline{\delta}^k, x^k), \cup_{\ell<k}\mathcal{T}^\ell) \leq \operatorname{dist}(\mathcal{M}_r(\underline{\delta}^k, x^k), \cup_{\ell\in K^*\cap[\![0,k[\![}\mathcal{T}^\ell_{\mathtt{C}}) < \frac{\varepsilon}{2}$ pour tout $k \in K^*$ suffisamment grand. De plus, nous avons aussi $\operatorname{dist}(x, \mathcal{M}_r(x^k, \underline{\delta}^k)) < \frac{\varepsilon}{2}$ pour tout $k \in K^*$ suffisamment grand, par construction de $\mathcal{M}(\underline{\delta}^k)$. Prenons alors $k \in K^*$ satisfaisant ces deux conditions, définissons y comme l'arrondi de x sur $\mathcal{M}_r(\underline{\delta}^k, x^k)$ et c comme l'arrondi de y sur $\cup_{\ell<k}\mathcal{T}^\ell$. On a alors $\|x - c\| \leq \|x - y\| + \|y - c\| < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon$, ce qui montre que $c \in \mathcal{B}_\varepsilon(x) \cap \mathcal{T}$ et conclut ainsi la preuve. □

Construction d'un oracle de `covering`

Tout oracle de `covering` mène à la propriété 4.3, mais en pratique un oracle de `covering` doit satisfaire deux propriétés supplémentaires. D'abord, il doit générer peu de points, pour éviter que l'étape de `covering` associée ne soit trop coûteuse par itération. Ensuite, il doit garantir une bonne répartition des points qu'il propose à toute itération (et non pas seulement asymptotiquement), pour éviter le défaut de l'étape `revealing` illustré en figure 4.1. Pour répondre à tous ces prérequis supplémentaires, nous proposons l'oracle de `covering` suivant.

$$\forall (y; \nu; \mathcal{S}) \in \mathbb{R}^n \times \mathbb{R}^*_+ \times 2^{\mathbb{R}^n}, \qquad \mathbb{O}_{\max\operatorname{dist}}(y; \nu; \mathcal{S}) \triangleq \operatorname*{argmax}_{d\in\mathcal{M}(\nu)\cap\operatorname{cl}(\mathcal{B}_r)} \operatorname{dist}(y + d, \mathcal{S}). \quad (\mathbb{O}_{\max\operatorname{dist}})$$

Cet oracle de `covering` force la définition de $\mathcal{D}^k_{\mathtt{C}}$ à toute itération $k \in \mathbb{N}$ comme une direction $d^k_{\mathtt{C}}$ dans $\mathcal{M}(\underline{\delta}^k) \cap \operatorname{cl}(\mathcal{B}_r)$ telle que $t^k_{\mathtt{C}} \triangleq x^k + d^k_{\mathtt{C}}$ soit le plus loin possible de l'ensemble des points d'évaluation passés. Dans le cas où deux directions ou plus vérifient cette propriété, on peut toutes les prendre ou en choisir une au hasard. Cette construction est pensée pour viser un recouvrement uniforme de la boule $\operatorname{cl}(\mathcal{B}_r(x^k))$ à chaque itération $k \in \mathbb{N}$.

Proposition 4.5. La fonction $\mathbb{O}_{\max\operatorname{dist}}$ est un oracle de `covering`.

Démonstration. Pour tout point $y \in \mathbb{R}^n$, tout rayon $\nu \in \mathbb{R}^*_+$ et tout ensemble $\mathcal{S} \subseteq \mathbb{R}^n$, on a $\mathbb{O}_{\max\operatorname{dist}}(y, \nu, \mathcal{S}) \subseteq \mathcal{M}_r(\nu)$, donc le premier prérequis de la définition 4.1 est vérifié. Pour vérifier le second prérequis, considérons $y \in \mathbb{R}^n$ et $(y^k; \nu^k) \in \mathbb{R}^n \times \mathbb{R}^*_+$ pour tout $k \in \mathbb{N}$ tels que $y^k \to y$ et $\nu^k \to 0$. Sans perte de généralité, supposons que $y^k \in \mathcal{B}_r(y)$ pour tout $k \in \mathbb{N}$. Définissons $\mathcal{S}^{-1} \triangleq \emptyset$ et $\mathcal{S}^k \triangleq \{y^k\} + \mathbb{O}_{\max\operatorname{dist}}(y^k, \nu^k, \mathcal{S}^{k-1})$ et $t^k \in \mathcal{S}^k$, pour tout $k \in \mathbb{N}$. Alors $\max_{d\in\mathcal{M}(\nu^k)\cap\operatorname{cl}(\mathcal{B}_r)} \operatorname{dist}(y^k + d, \cup_{\ell<k}\mathcal{S}^\ell) = \operatorname{dist}(t^k, \cup_{\ell<k}\mathcal{S}^\ell) \leq \min_{\ell<k}\|t^k - t^\ell\|$ pour tout $k \in \mathbb{N}$. De plus, $(t^k)_{k\in\mathbb{N}}$ a tous ses éléments inclus dans le compact $\operatorname{cl}(\mathcal{B}_{2r}(x))$, et donc $\min_{\ell<k}\|t^k - t^\ell\| \to 0$ lorsque $k \to +\infty$. Cela prouve que le second prérequis de la définition 4.1 est vérifié. □

Pour `cDSM` à treillis adaptatif, calculer $\mathbb{O}_{\max\operatorname{dist}}(y; \nu; \mathcal{S})$ est un problème combinatoire pour n'importe quel $(y; \nu; \mathcal{S}) \in \mathbb{R}^n \times \mathbb{R}^*_+ \times 2^{\mathbb{R}^n}$, puisque $\mathcal{M}(\nu)$ est un ensemble discret. Ce pro-

blème est difficile lorsque $\mathcal{S}$ contient beaucoup d'éléments et que ν est petit. Un problème très proche, le *facility dispersion problem* (nommé aussi *obnoxious facility location* [109]) est NP-difficile. Il est cependant possible de surmonter cette difficulté. Le schéma reste valide si $\mathbb{O}_{\max \operatorname{dist}}$ est remplacé par $\mathbb{O}_{\max \operatorname{dist}_\alpha}$ pour n'importe quel $\alpha \in]0,1]$, où $\mathbb{O}_{\max \operatorname{dist}_\alpha}(y;\nu;\mathcal{S})$ renvoie un élément d_α tel que $\operatorname{dist}(y+d_\alpha,\mathcal{S}) \geq \alpha \max_{d\in\mathcal{M}(\nu)\cap\operatorname{cl}(\mathcal{B}_r)} \operatorname{dist}(y+d,\mathcal{S})$. La preuve de la proposition 4.4 s'adapte aisément au cas où $\mathbb{O}_{\max \operatorname{dist}_\alpha}$ est considéré à la place de $\mathbb{O}_{\max \operatorname{dist}}$, et la littérature sur le *facility dispersion problem* mentionne des heuristiques efficaces pour construire $\mathbb{O}_{\max \operatorname{dist} \frac{1}{2}}$ [95]. On peut aussi résoudre le problème sur l'espace continu (en autorisant $d \in \operatorname{cl}(\mathcal{B}_r)$ au lieu de $d \in \mathcal{M}_r(\nu) = \mathcal{M}(\nu) \cap \operatorname{cl}(\mathcal{B}_r)$) puis arrondir la solution sur $\mathcal{M}_r(\nu)$.

Pour `cDSM` à décroissance suffisante (où $\mathcal{M}(\cdot) \triangleq \mathbb{R}^n$), $\mathbb{O}_{\max \operatorname{dist}}(y;\nu;\mathcal{S})$ est la solution d'un problème d'optimisation continue non lisse. Nous pouvons considérer un substitut, comme $\underline{\mathbb{O}_{\max \operatorname{dist}}}(y;\nu;\mathcal{S}) \triangleq \operatorname{argmax}_{d\in\operatorname{cl}(\mathcal{B}_r)} \Sigma_{a\in\mathcal{S}} \frac{-1}{\|y+d-a\|}$ par exemple. Nous pouvons de plus remplacer $\underline{\mathbb{O}_{\max \operatorname{dist}}}$ par $\underline{\mathbb{O}_{\max \operatorname{dist}_\alpha}}$ pour n'importe quel $\alpha \in]0,1]$ fixé, où $\underline{\mathbb{O}_{\max \operatorname{dist}_\alpha}}(y;\nu;\mathcal{S})$ renvoie un élément d_α tel que $\Sigma_{a\in\mathcal{S}} \frac{-1}{\|y+d_\alpha-a\|} \geq \alpha \max_{d\in\operatorname{cl}(\mathcal{B}_r)} \Sigma_{a\in\mathcal{S}} \frac{-1}{\|y+d-a\|}$. Ceci n'altère pas le schéma, et permet l'utilisation d'heuristiques simples pour calculer d_α. Par exemple, une recherche exhaustive sur un treillis suffisamment fin, ou une recherche itérant sur les premiers termes d'une *suite remplissante* telle que la suite pseudo-aléatoire de Halton [58].

L'oracle de `covering` $\mathbb{O}_{\max \operatorname{dist}}$ pourrait servir de base de comparaison avec d'autres oracles, car il n'est certainement pas le seul envisageable. On peut par exemple considérer un schéma exploitant les fonctions d'acquisition des algorithmes bayésiens, en posant

$$\forall (y;\nu;\mathcal{S}) \in \mathbb{R}^n \times \mathbb{R}_+^* \times 2^{\mathbb{R}^n}, \qquad \mathbb{O}_{\max \varphi}(y;\nu;\mathcal{S}) \triangleq \operatorname*{argmax}_{d\in\mathcal{M}(\nu)\cap\operatorname{cl}(\mathcal{B}_r)} \varphi(y+d,\mathcal{S}), \qquad (\mathbb{O}_{\max \varphi})$$

où φ est une fonction d'acquisition (pour des détails, nous référons à la section dédiée aux algorithmes bayésiens du chapitre 2.2.2). Si la conjecture 2.2 est vraie et se généralise au cas discontinu, alors elle mène immédiatement à la proposition 4.4.

Le coût de calcul associé à ces schémas peut paraître rédhibitoire, mais dans un contexte sans dérivées il est négligeable comparé au coût de calcul de $f(x)$ en n'importe quel $x \in X$. L'efficience pratique de ces schémas découle donc du fait qu'ils génèrent peu de points mais bien choisis, et que le calcul de ces points n'implique aucun appel à f. Le coût de chaque itération de `cDSM` engendré par l'étape de `covering` est alors de 1 point à évaluer à chaque itération. En comparaison, l'étape usuelle de `poll` évalue au moins $n+1$ points par itération, puisqu'elle repose sur une base positive de directions de $\mathbb{R}^n$. Minimiser le coût de l'étape de `covering` est probablement le choix le plus pertinent dans de nombreux contextes pratiques. Remarquons de plus que la définition d'un oracle de `covering` autorise $\mathbb{O}$

à renvoyer l'ensemble vide. Nous pouvons donc limiter encore le nombre de points considérés par l'étape de `covering` au cours de l'optimisation. Par exemple, $\mathbb{O}$ peut ne renvoyer un ensemble non vide que lorsque δ^k est suffisamment petit.

Le choix du rayon de `covering` $r \in \mathbb{R}_+^*$ est laissé libre dans l'algorithme, mais il est difficile d'identifier une valeur optimale. Ce choix dépend probablement de l'instance du problème (**P**) considérée. En effet, plus r est petit, plus vite l'étape de `covering` recouvre efficacement $\mathcal{B}_r(x^*)$, mais en contrepartie, plus r est grand et plus `cDSM` peut s'échapper de solutions locales de faible qualité. En l'absence d'information sur le problème, deux choix simples sont $r \triangleq \delta^0$ ou $r \triangleq \frac{\delta^0}{10}$. Nous pouvons également envisager une suite $(r^k)_{k\in\mathbb{N}}$ de rayons au lieu d'un rayon r fixe, sous réserve que $\underline{r} \triangleq \inf_{k\in\mathbb{N}} r^k > 0$ et que (DCP) tienne avec $\underline{r} > 0$.

4.2.2 Allègement du cadre théorique requis pour l'analyse de `cDSM`

Nous mentionnions au chapitre 4.1.2 que le cadre d'étude 4.1 présente quelques défauts. Le principal, lié à l'unicité du point raffiné, est surmonté grâce au remplacement de `rDSM` par `cDSM`. Le remplacement de l'étape `revealing` par l'étape `covering` règle aussi le manque de compatibilité pratique de `rDSM` avec les critères d'arrêt usuels. Il reste donc à corriger le troisième défaut du cadre 4.1, portant sur l'hypothèse 3.2 difficile à garantir en pratique. Dans ce chapitre, nous donnons une nouvelle hypothèse (hypothèse 4.1), strictement plus légère que l'hypothèse 3.2 mais qui permet quand même d'établir une analyse de convergence satisfaisante de `cDSM`. L'analyse est laissée au chapitre 4.2.3.

Dans un contexte sans dérivées où f est une boîte noire, les ensembles de continuité de f peuvent être mal connus. Il est alors difficile de garantir qu'ils vérifient l'ICP demandée par l'hypothèse 3.2. En comparaison, il est possiblement plus aisé de garantir que les ensembles de continuité sont *amples*,[2] au sens de la définition suivante illustrée par la figure 4.3.

Définition 4.2. Un ensemble $\mathcal{S} \subseteq \mathbb{R}^n$ est dit *ample* lorsque $\mathcal{S} \subseteq \mathrm{cl}(\mathrm{int}(\mathcal{S}))$.

Définition 4.3. Un ensemble $\mathcal{S} \subseteq \mathbb{R}^n$ est dit *localement fin* lorsqu'il existe $\mathcal{O} \subseteq \mathbb{R}^n$ ouvert tel que $\mathcal{S} \cap \mathcal{O} \neq \emptyset = \mathrm{int}(\mathcal{S}) \cap \mathcal{O}$.

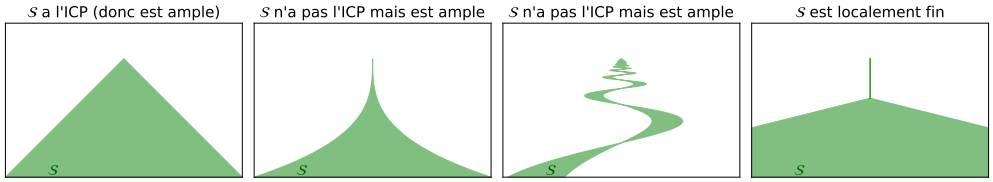
Imposer la notion d'amplitude aux ensembles de continuité de f est le prérequis le moins restrictif possible pour l'analyse de `cDSM`. En effet, les seules ensembles non-amples sont les ensembles que nous nommons *localement fins*, comme prouvé dans la proposition suivante.

2. Nous présentons ici la notion comme si elle était nouvelle ; mais durant la rédaction de ce manuscrit nous avons découvert une référence introduisant la notion d'ensemble *semi-ouvert* [76], équivalente à notre notion d'ensemble ample [76, théorème 1]. Toutefois, nous nous permettons de conserver la notion d'ensemble *ample* telle que développée durant cette thèse, car nous la trouvons plus parlante.

Proposition 4.6. Un ensemble est localement fin si et seulement s'il n'est pas ample.

Démonstration. Soit $\mathcal{S} \subseteq \mathbb{R}^n$. Supposons que $\mathcal{S}$ est localement fin. Prenons $x \in \mathcal{S} \cap \mathcal{O}$, où $\mathcal{O} \subseteq \mathbb{R}^n$ est un ouvert tel que $\mathcal{S} \cap \mathcal{O} \neq \emptyset = \mathrm{int}(\mathcal{S}) \cap \mathcal{O}$. Alors si $\mathcal{S}$ était ample, on aurait $x \in \mathrm{cl}(\mathrm{int}(\mathcal{S})) \cap \mathcal{O}$. Il existerait donc $\varepsilon > 0$ tel que $\mathcal{B}_\varepsilon(x) \subseteq \mathcal{O}$ et $\mathcal{B}_\varepsilon(x) \cap \mathrm{int}(\mathcal{S}) \neq \emptyset$, et de fait $\mathrm{int}(\mathcal{S}) \cap \mathcal{O} \neq \emptyset$, ce qui soulève une contradiction. Ainsi, $\mathcal{S}$ localement fin ne peut pas être ample. Maintenant, supposons à la place que $\mathcal{S} \subseteq \mathbb{R}^n$ n'est pas ample. Alors $\mathcal{O} \triangleq \mathbb{R}^n \setminus \mathrm{cl}(\mathrm{int}(\mathcal{S}))$ est ouvert, et $\mathcal{S} \cap \mathcal{O} \neq \emptyset = \mathrm{int}(\mathcal{S}) \cap \mathcal{O}$ par construction. De fait, $\mathcal{S}$ non ample est localement fin. □

FIGURE 4.3 Illustration d'ensembles amples ou non.



La notion d'ensemble ample est strictement plus générale que celle de l'ICP.

Proposition 4.7. Si $\mathcal{S} \subseteq \mathbb{R}^n$ a l'ICP, alors $\mathcal{S}$ est ample, mais la réciproque est fausse.

Démonstration. Soit $\mathcal{S} \subseteq \mathbb{R}^n$. Supposons que $\mathcal{S}$ a l'ICP et prenons $x \in \mathcal{S}$. Appliquons l'ICP à $\mathcal{S}$ en x. Suivant les notations de la définition, $\mathcal{K}$ est ouvert comme image de $\mathbb{R}_+^* \times \mathcal{U}$ (un ouvert de $\mathbb{R}_+^* \times \mathbb{S}^n$) par l'homéomorphisme $(\lambda, u) \in \mathbb{R}_+^* \times \mathbb{S}^n \mapsto \lambda u \in \mathbb{R}^n \setminus \{0\}$. On a aussi les ensembles $\mathcal{K}_x$ et $\mathcal{O}_x$ tels que $(\mathcal{K}_x \cap \mathcal{O}_x) \subseteq \mathcal{S}$ et $x \in \mathrm{cl}(\mathcal{K}_x \cap \mathcal{O}_x)$. Ainsi, $\mathcal{K}_x \cap \mathcal{O}_x$ est également ouvert, et donc $(\mathcal{K}_x \cap \mathcal{O}_x) \subseteq \mathrm{int}(\mathcal{S})$. On a donc $x \in \mathrm{cl}(\mathrm{int}(\mathcal{S}))$, ce qui clôt la preuve de l'implication directe. Pour contredire la réciproque, considérons $n \triangleq 2$ et $\mathcal{S} \triangleq \mathrm{epi}(\sqrt{|\cdot|})$. Alors $\mathcal{S}$ est ample mais n'a pas l'ICP en $x = (0, 0) \in \mathrm{cl}(\mathcal{S})$. □

Ainsi, en ne supposant pas que les ensembles de continuité de f ont l'ICP mais uniquement qu'ils sont amples, on peut toujours faire tenir le lemme 4.1 (et donc le théorème 4.2). On peut donc affaiblir l'hypothèse 3.2 comme suit.

Hypothèse 4.1. **[Hypothèse 3.2 affaiblie].** Il existe une partition de X en $X = \sqcup_{i=0}^{N} X_i$, avec $N \in \overline{\mathbb{N}}^*$ et $\mathcal{X}_0 \triangleq X \setminus \Omega$, où pour tout $i \in [\![1, N]\!]$, X_i est ample et $f_{|X_i}$ est continue.

Il y a trois différences entre l'hypothèse 4.1 et l'hypothèse 3.2. D'abord, l'hypothèse 4.1 est globale, alors que l'hypothèse 3.2 est locale mais stipulée en x^* supposé connu. Ce changement est transparent. Ensuite, l'hypothèse 3.2 requiert la continuité de Lipschitz de f sur chaque ensemble de continuité, alors que l'hypothèse 4.1 n'impose que la continuité simple. Nous ne nous concentrerons pas plus sur ce changement.[3] Enfin, la principale différence réside dans le fait que l'hypothèse 3.2 impose l'ICP à chacun des ensembles de continuité de f, alors que l'hypothèse 4.1 ne demande qu'un caractère ample. Ce dernier changement est le plus notable, car c'est lui qui justifie notre affirmation que l'hypothèse 4.1 est strictement plus légère que l'hypothèse 3.2. Cette affirmation découle directement de la proposition 4.7.

Le cadre d'étude minimal requis pour l'analyse de `cDSM` est donc le suivant.

Cadre 4.2. **Cadre d'étude de `cDSM`.** Considérons le problème (**P**) dans le cadre 1.1, résolu par `cDSM`. L'ensemble $\mathcal{T}$ des points évalués vérifie la propriété 4.3. On suppose que X admet une partition en $X = \sqcup_{i=0}^{N} X_i$, avec $N \in \overline{\mathbb{N}}^*$ où $\mathcal{X}_0 \triangleq X \setminus \Omega$ et X_i est un ensemble ample de continuité de f pour tout $i \in [\![1, N]\!]$ (hypothèse 4.1).

4.2.3 Analyse de convergence de `cDSM` vers des optima locaux

Comme annoncé au chapitre 4.2.2, le cadre d'étude 4.2 permet d'établir une analyse de convergence satisfaisante pour `cDSM`. Nous la précisons maintenant, en posant et prouvant le théorème suivant.

Théorème 4.8. Dans le cadre d'étude 4.2, `cDSM` génère au moins une suite raffinante, et en considérant n'importe laquelle (notée $K^* \subseteq \mathbb{N}$, de point raffiné noté x^*), on a

$$x^* \in \mathrm{cl}(X) \cap \Omega \quad \text{et} \quad \lim_{k \in K^*} f(x^k) \;=\; \begin{cases} \min f(\mathcal{B}_r(x^*) \cap \Omega) = f(x^*) & \text{si } x^* \in X, \\ \inf f(\mathcal{B}_r(x^*) \cap \Omega) & \text{si } x^* \notin X. \end{cases}$$

Préliminaire topologique liés aux ensembles amples

Nous utilisons dans cette section une notion de densité que nous nommons *intersection dense entre deux ensembles*. Un ensemble $\mathcal{S}_1 \subseteq \mathbb{R}^n$ a une *intersection dense avec* $\mathcal{S}_2 \subseteq \mathbb{R}^n$ lorsque $\mathcal{S}_2 \subseteq \mathrm{cl}(\mathcal{S}_1 \cap \mathcal{S}_2)$. Cette notion étend la notion usuelle de *sous-ensemble dense dans un ensemble* ($\mathcal{S}_1 \subseteq \mathcal{S}_2$ et $\mathcal{S}_2 = \mathrm{cl}(\mathcal{S}_1)$) au cas où $\mathcal{S}_1$ n'est pas inclus dans $\mathcal{S}_2$. On peut alors poser et prouver le lemme suivant.

3. La continuité de Lipschitz n'était requise dans l'hypothèse 3.2 que pour permettre le calcul de dérivées de Clarke dans l'analyse initiale de `rDSM` menée dans [11]. Nous ne calculons pas cette dérivée dans l'analyse de `cDSM`, donc la continuité est allégée en conséquence.

Lemme 4.9. Soit $\mathcal{S}_1 \subseteq \mathbb{R}^n$, $\mathcal{S}_2 \subseteq \mathbb{R}^n$ et $\mathcal{S}_3 \subseteq \mathbb{R}^n$. Alors,

a) si $\mathcal{S}_1$ est ample et $\mathcal{S}_2$ est ouvert, alors $\mathcal{S}_1 \cap \mathcal{S}_2$ est ample ;

b) si $\mathcal{S}_1$ a une intersection dense avec $\mathcal{S}_2$ et $\mathcal{S}_3 \subseteq \mathcal{S}_2$ est ample, alors $\mathcal{S}_1$ a une intersection dense avec $\mathcal{S}_3$;

c) si $\mathcal{S}_1$ a une intersection dense avec $\mathcal{S}_2$ et $\mathcal{S}_2$ est ample, alors pour tout $x \in \mathcal{S}_2$, $\mathcal{S}_1 \backslash \{x\}$ a une intersection dense avec $\mathcal{S}_2$.

Démonstration. **a).** Considérons $\mathcal{S}_1 \subseteq \mathbb{R}^n$ ample et $\mathcal{S}_2 \subseteq \mathbb{R}^n$ ouvert. Pour tout $x \in \mathcal{S}_1 \cap \mathcal{S}_2$, on a $x \in \mathcal{S}_1 \subseteq \mathrm{cl}(\mathrm{int}(\mathcal{S}_1))$, donc il existe $(x^k)_{k\in\mathbb{N}} \to x$ avec $x^k \in \mathrm{int}(\mathcal{S}_1)$ pour tout $k \in \mathbb{N}$. Cela implique que $x^k \in \mathrm{int}(\mathcal{S}_1) \cap \mathcal{S}_2 = \mathrm{int}(\mathcal{S}_1 \cap \mathcal{S}_2)$ pour tout k suffisamment grand, et donc que $x \in \mathrm{cl}(\mathrm{int}(\mathcal{S}_1 \cap \mathcal{S}_2))$ comme désiré.

b). Considérons que $\mathcal{S}_1 \subseteq \mathbb{R}^n$ a une intersection dense avec $\mathcal{S}_2 \subseteq \mathbb{R}^n$, et que $\mathcal{S}_3 \subseteq \mathcal{S}_2$ est ample. Pour tout $x \in \mathcal{S}_3 \subseteq \mathrm{cl}(\mathrm{int}(\mathcal{S}_3))$, considérons $(x^k)_{k\in\mathbb{N}}$ convergeant vers x avec $x^k \in \mathrm{int}(\mathcal{S}_3) \subseteq \mathcal{S}_2 \subseteq \mathrm{cl}(\mathcal{S}_1 \cap \mathcal{S}_2)$ pour tout $k \in \mathbb{N}$. Alors, pour tout $k \in \mathbb{N}$, il existe $(x^k_\ell)_{\ell\in\mathbb{N}} \to x^k$ avec $x^k_\ell \in \mathcal{S}_1 \cap \mathcal{S}_2$ pour tout $\ell \in \mathbb{N}$, et il existe un indice $\ell(k) \in \mathbb{N}$ tel que $\|x^k_{\ell(k)} - x^k\| \leq 2^{-k}$ et $x^k_{\ell(k)} \in \mathrm{int}(\mathcal{S}_3)$. Il vient alors que $x^k_{\ell(k)} \to x$. De plus, $x^k_{\ell(k)} \in \mathcal{S}_1 \cap \mathcal{S}_2 \cap \mathrm{int}(\mathcal{S}_3) \subseteq \mathcal{S}_1 \cap \mathcal{S}_3$ pour tout $k \in \mathbb{N}$, et ainsi $x \in \mathrm{cl}(\mathcal{S}_1 \cap \mathcal{S}_3)$ comme désiré.

c). Considérons que $\mathcal{S}_1 \subseteq \mathbb{R}^n$ a une intersection dense avec $\mathcal{S}_2 \subseteq \mathbb{R}^n$ ample. Soit $x \in \mathcal{S}_2$. Prenons $y \in \mathcal{S}_2$. On a donc $y \in \mathrm{cl}(\mathrm{int}(\mathcal{S}_2))$, donc il existe $(y^k)_{k\in\mathbb{N}}$ convergeant vers y avec $y^k \in \mathrm{int}(\mathcal{S}_2)$ pour tout $k \in \mathbb{N}$. Pour tout $k \in \mathbb{N}$, on peut donc définir $0 < r^k \leq 2^{-k}$ tel que $\mathcal{B}_{r^k}(y^k) \subseteq \mathrm{int}(\mathcal{S}_2)$, et il existe donc $z^k \in \mathcal{B}_{r^k}(y^k) \setminus \{x\}$. Par construction, $z^k \to y$ et $z^k \in \mathcal{S}_1 \cap \mathcal{S}_2 \setminus \{x\}$ pour tout $k \in \mathbb{N}$. Ainsi, $y \in \mathrm{cl}((\mathcal{S}_1 \setminus \{x\}) \cap \mathcal{S}_2)$ comme désiré. □

Préliminaires liés à `cDSM`

L'étape de `covering` est exprimable comme une `search`. De fait, `cDSM` hérite de toutes les propriétés de `DSM`. La proposition 4.10, stipulée en [17, théorème 8.1] pour `DSM` à treillis adaptatif et en [35, corollaire 7.2] pour `DSM` à décroissance suffisante, est donc valide.

Proposition 4.10. Dans le cadre 4.2, `cDSM` génère au moins une suite raffinante.

Cependant, l'étape de `covering` différencie `cDSM` de `DSM`. En effet, `cDSM` dispose de la propriété 4.3, qui permet de clamer le lemme suivant. Ce lemme est très proche du lemme 3.6 donnant la densité des points d'évaluation de `rDSM` dans chaque ensemble de continuité de f autour de l'unique point raffiné de `rDSM` ; mais contrairement au lemme 3.6, le résultat ci-dessous reste vrai lorsqu'il y a plusieurs points raffinés.

Lemme 4.11. Dans le cadre 4.2, pour tout point raffiné x^*, tout $i \in [\![1, N]\!]$ et tout $x \in X_i$, les ensembles $\mathcal{T}$ et $\mathcal{T} \setminus \{x\}$ ont tous les deux une intersection dense avec $\mathcal{B}_r(x^*) \cap X_i$.

Démonstration. Soit x^* un point raffiné généré par `cDSM`. Soit $i \in [\![1, N]\!]$. Alors (DCP) et le lemme 4.9.b) appliqué à $\mathcal{S}_1 \triangleq \mathcal{T}$, $\mathcal{S}_2 \triangleq \mathrm{cl}(\mathcal{T})$ et $\mathcal{S}_3 \triangleq \mathcal{B}_r(x^*)$ prouvent que $\mathcal{T}$ a une intersection dense avec $\mathcal{B}_r(x^*)$. De plus, le lemme 4.9.a) sous l'hypothèse 4.1 affirme que $\mathcal{B}_r(x^*) \cap X_i$ est un sous-ensemble ample de $\mathcal{B}_r(x^*)$. Le premier résultat en découle alors, par application du lemme 4.9.b) à $\mathcal{S}_1 \triangleq \mathcal{T}$, $\mathcal{S}_2 \triangleq \mathcal{B}_r(x^*)$ et $\mathcal{S}_3 \triangleq \mathcal{B}_r(x^*) \cap X_i$. Le second résultat suit par application du lemme 4.9.c) à $\mathcal{S}_1 \triangleq \mathcal{T}$ et $\mathcal{S}_2 \triangleq \mathcal{B}_r(x^*) \cap X_i$. □

On peut alors en déduire la proposition principale découlant de la propriété 4.3.

Proposition 4.12. Dans le cadre 4.2, $\lim_{k \in K^*} f(x^k) \leq f(x)$ pour tout $x \in \mathcal{B}_r(x^*) \cap \Omega$ et tout $(K^*; x^*; \cdot) \in \mathbb{G}^*_{\mathtt{cDSM}}$.

Démonstration. Soit $(K^*; x^*; \cdot) \in \mathbb{G}^*_{\mathtt{cDSM}}$. Observons d'abord que $(f(x^k))_{k \in \mathbb{N}}$ est bornée inférieurement (car $f_{|\Omega}$ l'est et $x^k \in \Omega$ pour tout $k \in \mathbb{N}$) et décroît par construction. Ainsi $f^* \triangleq \lim_{k \in K^*} f(x^k) = \lim_{k \in \mathbb{N}} f(x^k) \neq +\infty$ existe.

Soit $x \in \mathcal{B}_r(x^*) \cap \Omega$. On a $f(x) = +\infty > f^*$ pour tout $x \notin X$, donc on peut se restreindre à supposer également que $x \in X_i$ pour un certain $i \in [\![1, N]\!]$. Soit $(y_\ell)_{\ell \in \mathbb{N}} \to x$ avec $y_\ell \in \mathcal{T} \cap X_i \setminus \{x\}$ pour tout $\ell \in \mathbb{N}$, dont l'existence découle du lemme 4.11. Définissons aussi $\kappa(\ell) \triangleq \min\{k \in \mathbb{N} : y_\ell \in \mathcal{T}^k\}$ pour tout $\ell \in \mathbb{N}$. On a $(\kappa(\ell))_{\ell \in \mathbb{N}} \to +\infty$ puisque $\mathcal{T}^k$ est fini pour tout $k \in \mathbb{N}$. En effet, à supposer le contraire, il existe alors $k \leq \max_{\ell \in \mathbb{N}} \kappa(\ell)$ tel que $\mathcal{T}^k$ contient une infinité d'éléments de la suite $(y_\ell)_{\ell \in \mathbb{N}}$; or $(y_\ell)_{\ell \in \mathbb{N}}$ prend une infinité de valeurs distinctes car $y_\ell \neq x$ pour tout $\ell \in \mathbb{N}$ et $(y_\ell)_{\ell \in \mathbb{N}} \to x$.

Enfin, par construction, nous avons $f(y_\ell) \geq f(t_{\mathtt{C}}^{\kappa(\ell)}) \geq f(x^{\kappa(\ell)+1})$ si l'itération $\kappa(\ell)$ est un succès et $f(y_\ell) \geq f(t_{\mathtt{C}}^{\kappa(\ell)}) \geq f(x^{\kappa(\ell)}) - \rho(\underline{\delta}^{\kappa(\ell)})$ si l'itération $\kappa(\ell)$ est un échec. Ceci implique que $f(y_\ell) + \rho(\underline{\delta}^{\kappa(\ell)}) \geq f(x^{\kappa(\ell)+1})$ pour tout $\ell \in \mathbb{N}$. Le résultat suit, puisque $f(x^{\kappa(\ell)+1}) \to f^*$, et $\rho(\underline{\delta}^{\kappa(\ell)}) \to 0$, et $f(y_\ell) \to f(x)$. □

Démonstration du théorème 4.8

Grâce à la proposition 4.12, la preuve du théorème 4.8 devient plus concise que ce qui a été fait pour prouver le théorème 4.2 relatif à l'étude de `rDSM`.

Démonstration. Plaçons-nous dans le cadre du théorème 4.8. La proposition 4.10 garantit que $\mathbb{G}^*_{\texttt{cDSM}} \neq \emptyset$. Soit $(K^*; x^*; \cdot) \in \mathbb{G}^*_{\texttt{cDSM}}$. On a $x^* \in \mathrm{cl}(X) \cap \Omega$ comme limite de la suite $(x^k)_{k \in K^*}$ où $x^k \in \Omega \cap X$ pour tout $k \in K^*$ et Ω est fermé. Notons $f^* \triangleq \lim_{k \in K^*} f(x^k)$, qui existe et est finie car $(f(x^k))_{k \in \mathbb{N}}$ est décroissante et $f_{|\Omega}$ est bornée inférieurement. Montrons d'abord que $f^* = \inf f(\mathcal{B}_r(x^*) \cap \Omega)$. Nous avons $f^* \geq \inf f(\mathcal{B}_r(x^*) \cap \Omega)$ car $x^k \in \mathcal{B}_r(x^*) \cap \Omega$ pour tout $k \in K^*$ assez grand, et l'autre inégalité est prouvée par l'absurde : en supposant que $f^* > \inf f(\mathcal{B}_r(x^*) \cap \Omega)$, on peut définir $x^\sharp \in \mathcal{B}_r(x^*) \cap \Omega$ tel que $f^* > f(x^\sharp)$, mais alors $f^* > f(x^\sharp) \geq f^*$ par la proposition 4.12. Remarquons que cela clôture déjà la preuve du cas où $x^* \notin X$. Terminons donc le cas restant en supposant de plus que $x^* \in X$. On a alors $f^* \geq f(x^*)$ par semi-continuité de f, et $f^* \leq f(x^*)$ par la proposition 4.12. Il vient que $f(x^*) = f^* = \inf f(\mathcal{B}_r(x^*) \cap \Omega) = \min f(\mathcal{B}_r(x^*) \cap \Omega)$. □

4.2.4 Généraliser plus le cadre d'étude affaiblit la convergence

Le cadre d'étude 4.2 peut être légèrement généralisé encore. Mais en le faisant, la qualité de l'analyse de convergence se détériore. Nous discutons de ce fait dans cette section.

Impossibilité d'affaiblir l'hypothèse 4.1

Le caractère ample des ensembles de continuité de f est critique dans l'analyse, pour que la propriété 4.3 garantisse que `cDSM` évalue des points dans tous les ensembles de continuité. L'exemple suivant montre que négliger l'hypothèse 4.1 brise la convergence vers une solution.

Exemple 4.2. Soit $n = 2$, $\Omega \triangleq \mathbb{R}^n$ et $f : \mathbb{R}^2 \to \mathbb{R}$ définie par

$$\forall x \in X \triangleq \mathbb{R}^2, \qquad f(x) \triangleq \begin{cases} |x_1 - 1| + |x_2| - 1, & \text{si } x \in X_1 \triangleq (\mathbb{R}_- \times \mathbb{R}) \cup (\mathbb{R}^*_+ \times \{0\}), \\ |x_1| + |x_2|, & \text{si } x \in X_2 \triangleq X \setminus X_1. \end{cases}$$

Le problème de la minimisation de f consiste en une instance valide du problème (**P**). Cependant, le cadre d'étude 4.2 n'est pas exploitable car l'hypothèse 4.1 n'est pas satisfaite. En effet, X_1 est localement fin (prendre $\mathcal{O} \triangleq \mathbb{R}^*_+ \times \mathbb{R}$), et donc non-ample. Dans cet exemple, l'unique minimiseur de f est le point $(1, 0) \in X_1 \setminus \mathrm{cl}(\mathrm{int}(X_1))$, tandis que l'origine est le minimum global de $f_{|\mathrm{cl}(\mathrm{int}(X_1)) \cup X_2}$ mais pas de f. Ainsi, n'importe quelle instance de `cDSM` telle que $(\mathcal{D}^k_{\texttt{C}} \cup \mathcal{D}^k_{\texttt{S}} \cup \mathcal{D}^k_{\texttt{P}}) \cap (\mathbb{R}^*_+ \times \{0\}) = \emptyset$ pour tout $k \in \mathbb{N}$ et exécutée depuis $x^0 \triangleq (0, 0)$ ne quitte pas $(0, 0)$, puisqu'elle n'évalue que des points dans $\mathrm{cl}(\mathrm{int}(X_1)) \cup X_2$. Dans cette situation, $(0, 0)$ est donc l'unique point raffiné et appartient à X, mais il n'est pas un minimum local de f.

On pourrait toutefois se contenter de supposer que les ensembles de niveau de f sont localement amples au voisinage d'au moins un point raffiné, et alors clamer le théorème uniquement en ces points raffinés. Cependant, cette hypothèse est difficile à clamer en pratique car les points raffinés que `cDSM` va générer ne sont pas connus à priori.

Affaiblissement du cadre 1.1 de la `DFO`

Affaiblir les hypothèses posées à la définition du problème (**P**) affaiblit le résultat de convergence. Le caractère non vide de X et de $X \cap \Omega$ est critique. Le caractère fermé de Ω est très commun en optimisation, mais il est toutefois aisé d'adapter le théorème 4.8 pour se passer de ce pré-requis. Les hypothèses sur f méritent une discussion plus approfondie.

L'hypothèse stipulant que $f_{|\Omega}$ est bornée inférieurement et a ses ensembles de niveau bornés garantit à priori que `cDSM` génère des points raffinés. Sans cette hypothèse, le théorème devient dépendant d'une hypothèse à posteriori, clamant que l'exécution de `cDSM` a bien généré au moins un point raffiné. En pratique, ceci est simple à conjecturer mais impossible à certifier.

L'hypothèse de semi-continuité de $f_{|X}$ peut être négligée. En effet, sans la semi-continuité, on a toujours $\lim_{k \in K^*} f(x^k) = \inf f(\mathcal{B}_r(x^*) \cap \Omega)$ pour tout $K^* \subset \mathbb{N}$ indexant une suite raffinante (avec x^* le point raffiné associé). Cependant, il faut supposer que f est semi-continue en x^* pour retrouver $\inf f(\mathcal{B}_r(x^*) \cap \Omega) = \min f(\mathcal{B}_r(x^*) \cap \Omega) = f(x^*)$. Ce dernier théorème a une utilité dans les prochains chapitres, aussi nous le formalisons comme suit.

Théorème 4.13. Dans le cadre d'étude 4.2 privé de l'hypothèse de semi-continuité de f, `cDSM` génère au moins une suite raffinante, et en considérant n'importe laquelle (notée $K^* \subseteq \mathbb{N}$, de point raffiné noté $x^* \in \Omega$), on a

$$\lim_{k \in K^*} f(x^k) = \begin{cases} \min f(\mathcal{B}_r(x^*) \cap \Omega) = f(x^*) & \text{si } f \text{ est semi-continue en } x^*, \\ \inf f(\mathcal{B}_r(x^*) \cap \Omega) & \text{sinon.} \end{cases}$$

La preuve consiste en une réécriture directe de la preuve proposée au chapitre 4.2.3. Il suffit pour cela de remarquer que la semi-continuité de f en x^* n'est possible que si $x^* \in X$.

Extensions possibles de l'aspect pratique de `cDSM`

Il semble difficile de pousser l'étude théorique de la convergence de `cDSM` beaucoup plus loin que le théorème 4.13. En effet, il dépend de peu d'hypothèses, et nous avons montré qu'elles sont les plus génériques possibles (en tant qu'hypothèses observables à priori). Ainsi, des travaux futurs pourraient plutôt privilégier des extensions pratiques.

La première option consiste à gérer les contraintes. La formulation actuelle de `cDSM` requiert un point initial $x^0 \in X \cap \Omega$, à savoir un point réalisable du point de vue des contraintes explicites (caractérisant Ω) et cachées (les valeurs de $x \in \mathbb{R}^n$ telles que $f(x) = +\infty$, caractérisant X). Il est difficile de gérer des contraintes cachées lorsque X n'est pas connu à priori, mais les contraintes explicites sont plus prometteuses. Suivant la taxonomie [74], si Ω est caractérisé par des contraintes quantifiables, on peut tolérer $x^0 \notin \Omega$. On peut en effet appliquer `cDSM` en deux phases [17, algorithme 12.1], où la première phase minimise la *fonction de violation de contraintes* [48] pour identifier un point appartenant à Ω.

L'autre piste principale consiste en la définition de bons oracles de `covering`. Tout oracle de `covering` (définition 4.1) doit atteindre l'objectif asymptotique de valider la propriété 4.3. Mais en pratique, il faut aussi que l'oracle ait un intérêt en temps fini, et à chaque itération. Les schémas $\mathbb{O}_{\max \mathrm{dist}}$ et $\mathbb{O}_{\max \varphi}$ sont deux exemples d'oracles de `covering` visant à accomplir un objectif à court terme en complément de l'objectif asymptotique. À toute itération, ils cherchent respectivement à optimiser le recouvrement du voisinage de la solution courante par des points évalués, et à extrapoler f pour prédire quel point est statistiquement pertinent à évaluer.

Le tableau 4.1 récapitule l'intérêt de chacune des étapes de `cDSM`, et met en évidence ce qu'un bon oracle de `covering` pourrait chercher à accomplir à chaque itération.

Tableau 4.1 Intérêt de chaque étape de `cDSM`, et pistes d'amélioration pratique.

Étape	Potentiel de `cDSM`	
	`cDSM` avec oracle quelconque	`cDSM` avec oracle bien défini
`search`	Optionnelle. Permet l'usage d'heuristiques et de stratégies exploratoires.	
`poll`	Optionelle. Mais en pratique, contribue fortement à la convergence.	
`covering`	Requise. Assure l'optimalité locale de tous les points raffinés. Fortes garanties asymptotiques, mais peut manquer d'utilité pratique. Peut être coûteuse par itération.	Requis pour la convergence vers des solutions locales. Faible coût par itération. Optimise le recouvrement du voisinage de la solution courante, et enrichit des modèles de f.

CHAPITRE 5 AMÉLIORATION DE cDSM SUR CERTAINS PROBLÈMES PAR OPTIMISATION PARTITIONNÉE

Je vais lui faire une ordonnance, et une sévère ! Aux quatre coins de Paris qu'on va le retrouver, éparpillé par petits bouts, façon puzzle. Moi, quand on m'en fait trop, je correctionne plus. Je dynamite. Je disperse. Je ventile.

« Les Tontons Flingueurs », tirade de Bernard Blier écrite par Michel Audiard.

Au sortir des chapitres précédents, nous disposons d'un algorithme (cDSM) apte à résoudre le problème (**P**). Sa convergence est formalisée par le théorème 4.8, et la structure requise est décrite par le cadre d'étude 4.2, dont nous avons déjà prouvé qu'il est très large. Toutefois, le théorème 4.8 ne certifie que la convergence de cDSM, et ne donne pas de garanties de performance en pratique. Puisque cDSM est très proche de DSM, nous pouvons conjecturer que le comportement de cDSM en pratique est proche de celui de DSM. Par conséquent, cDSM peut manquer d'efficacité lorsque le problème (**P**) est de trop grande dimension. Dans ce chapitre, nous proposons une approche que nous nommons *optimisation partitionnée* (*Partitioned Optimization*, PO) pour accélérer cDSM lorsqu'elle est applicable.

La PO est envisageable dès que la structure du problème (**P**) serait simplifiée en reliant des variables entre elles ou en fixant certaines d'entre elles. Cela arrive par exemple lorsque

— seules quelques variables affectent l'aspect non lisse de la structure ;

— on considère un processus boîte grise dont la majeure partie est simple à optimiser ;

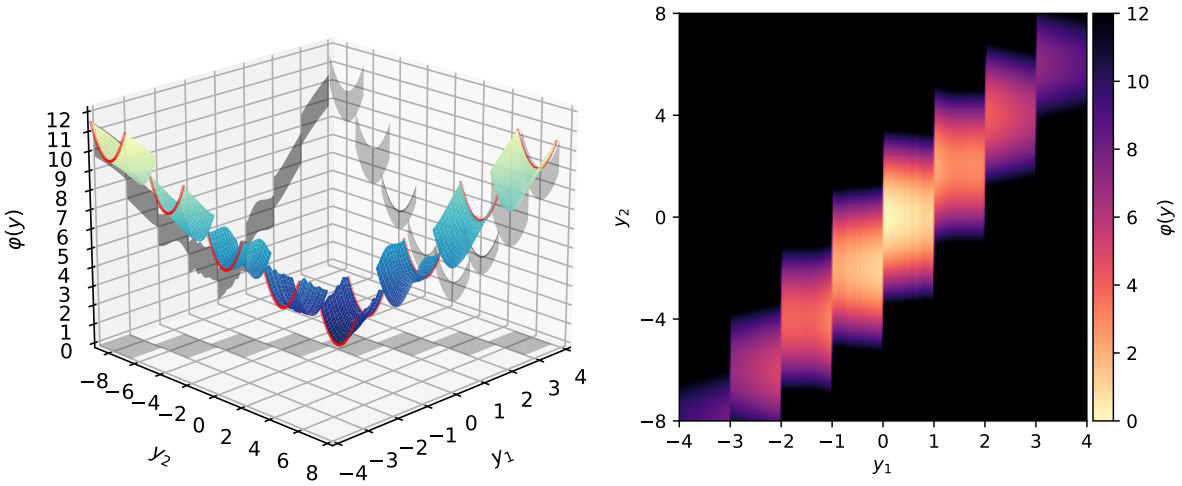
— fixer quelques variables rendrait le problème convexe ;

— le problème consiste en deux sous-problèmes distincts reliés par peu de variables ;

— le problème est discontinu mais seules quelques variables affectent les discontinuités.

L'idée est alors d'ajouter des contraintes reliant les variables entre elles afin de partitionner l'espace des variables suivant les remarques ci-dessus, pour que la fonction objectif restreinte à n'importe lequel des ensembles de partition soit plus simple à minimiser que sur l'espace entier. La difficulté de résolution du problème est alors transférée vers l'identification de l'ensemble de partition minimisant le minimum de la fonction objectif sur l'ensemble. Ce nouveau problème est potentiellement très peu structuré et fortement discontinu, mais de dimension faible. cDSM est donc approprié pour le résoudre. La figure 5.1 illustre une fonction $\varphi : \mathbb{R}^2 \to \mathbb{R}$ qu'il est pertinent de minimiser via la PO. La fonction est discontinue et oscille près de l'origine, mais sa restriction à n'importe quelle droite de $\mathbb{R}^2$ d'abscisse fixée est une fonction quadratique de l'ordonnée. On peut donc partitionner $\mathbb{R}^2$ comme $\mathbb{R}^2 = \sqcup_{y_1 \in \mathbb{R}} \{y_1\} \times \mathbb{R}$, et chercher $y_1 \in \mathbb{R}$ minimisant le minimum de $\varphi_{|\{y_1\} \times \mathbb{R}}$.

FIGURE 5.1 Fonction plus simple à minimiser via l'optimisation partitionnée.



Le cadre de la PO est formalisé au chapitre 5.1. Nous discutons ensuite de l'application de cDSM au problème reformulé par l'optimisation partitionnée à la place du problème initial, au chapitre 5.2. Nous prouvons à cette occasion que les solutions locales au problème reformulé calculées par cDSM permettent d'obtenir directement des solutions locales au problème initial. Enfin, le chapitre 5.3 discute d'extensions possibles du cadre. Notons qu'une illustration de l'application de la PO et une étude numérique des gains de performance qu'elle apporte sont proposées aux chapitres 6 (sur des problèmes de DFO) et 7 (pour une application à des problèmes de contrôle optimal requérant des méthodes sans dérivées).

5.1 Reformulation d'un problème via l'optimisation partitionnée

Les problèmes dont nous discutons dans ce chapitre sont plus structurés que le problème (**P**) générique. Le détail de cette structure requise est précisé au chapitre 5.1.1. Nous y introduisons la *reformulation d'un problème par le cadre de l'optimisation partitionnée*. Les liens entre le problème initial et le problème reformulé sont exposés au chapitre 5.1.2. Enfin, nous comparons notre approche avec des travaux issus de la littérature, au chapitre 5.1.3.

5.1.1 Présentation formelle du cadre de l'optimisation partitionnée

Pour bien distinguer le problème structuré que nous regardons ici du problème générique, nous nous permettons un changement de notations. Dans tout le chapitre 5, le cadre d'étude de l'optimisation partitionnée est le suivant.

Cadre 5.1. **Cadre de l'optimisation partitionnée (PO).** Le problème étudié est

$$\underset{y \in \mathbb{Y}}{\text{minimiser}} \quad \varphi(y) \quad \text{sous contrainte} \quad y \in \Omega, \tag{$\mathbf{P}_{\mathtt{PO}}^{\mathtt{ini}}$}$$

où $\mathbb{Y}$ est un espace normé, $\varphi : \mathbb{Y} \to \overline{\mathbb{R}}$ admet un domaine effectif $Y \subseteq \mathbb{Y}$ non vide, et $\Omega \subseteq \mathbb{Y}$ est fermé et intersecte Y. Nous supposons avoir une partition $\mathbb{Y} = \sqcup_{x \in \mathbb{X}} \mathbb{Y}(x)$, où $\mathbb{X}$ est un espace normé de dimension finie, et où

$$\forall x \in X, \quad \exists!\, \gamma(x) \in Y_\Omega(x) : \qquad \inf \varphi(Y_\Omega(x)) \quad = \quad \min \varphi(Y_\Omega(x)) \quad = \quad \varphi(\gamma(x)),$$

où $X \triangleq \{x \in \mathbb{X} : Y_\Omega(x) \triangleq Y(x) \cap \Omega \neq \emptyset\}$ et, pour tout $x \in \mathbb{X}$, $Y(x) \triangleq \mathbb{Y}(x) \cap Y$. On considère que la *fonction oracle* $\gamma : X \to Y$ est accessible, de même que la *fonction d'indice* $\chi : \mathbb{Y} \to \mathbb{X}$ définie pour tout $y \in \mathbb{Y}$ comme l'unique élément $x \in \mathbb{X}$ pour lequel y appartient à $\mathbb{Y}(x)$. Nous introduisons alors la *reformulation de* ($\mathbf{P}_{\mathtt{PO}}^{\mathtt{ini}}$) comme

$$\underset{x \in \mathbb{X}}{\text{minimiser}} \quad \Phi(x), \qquad \text{avec} \qquad \Phi : \begin{cases} \mathbb{X} & \to & \overline{\mathbb{R}} \\ x & \mapsto & \begin{cases} \varphi(\gamma(x)) & \text{si } x \in X, \\ +\infty & \text{sinon.} \end{cases} \end{cases} \tag{$\mathbf{P}_{\mathtt{PO}}^{\mathtt{ref}}$}$$

Pour simplifier la discussion du cadre, nous introduisons également pour tout $x \in \mathbb{X}$ le *sous-problème associé à* ($\mathbf{P}_{\mathtt{PO}}^{\mathtt{ini}}$) *en* x comme

$$\underset{y \in \mathbb{Y}(x)}{\text{minimiser}} \quad \varphi(y) \quad \text{sous contrainte} \quad y \in \Omega, \tag{$\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x)$}$$

qui a une solution globale (unique, valant $\gamma(x)$) de valeur objectif finie lorsque $x \in X$.

L'exploitation algorithmique du cadre requiert en plus les deux hypothèses suivantes.

Hypothèse 5.1. La fonction χ est continue, et $\chi(\mathcal{Y})$ est borné pour tout $\mathcal{Y} \subseteq \mathbb{Y}$ borné.

Hypothèse 5.2. La fonction $\varphi_{|\gamma(X)}$ est bornée inférieurement et a ses ensembles de niveau bornés ; et il existe une partition $X = \sqcup_{i=1}^{N} X_i$, où $N \in \overline{\mathbb{N}}^*$ telle que X_i est ample et $\gamma_{|X_i}$ est uniformément continue et $\varphi_{|\gamma(X_i)}$ est continue, pour tout $i \in [\![1, N]\!]$. De plus, pour tout $x \in X$ et tout $r \in \mathbb{R}_+^*$, il existe un nombre fini d'indices $i \in [\![1, N]\!]$ tels que $X_i \cap \mathcal{B}_r(x) \neq \emptyset$.

5.1.2 Préservation des solutions entre la reformulation et le problème initial

Nous prouvons dans cette section qu'en connaissant une solution (locale ou globale) de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$, on peut retrouver une solution (de même nature) de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$, et réciproquement.

Proposition 5.1. Dans le cadre d'étude 5.1, les deux affirmations suivantes sont vraies :

1. pour tout y^* solution globale de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$, $\chi(y^*)$ est une solution globale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$;
2. pour tout x^* solution globale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$, $\gamma(x^*)$ est une solution globale de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$.

Démonstration. Soit y^* une solution globale de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$, et $x^* \triangleq \chi(y^*)$. Par construction, on a $\Phi(x) = \varphi(\gamma(x)) \geq \varphi(y^*) = \varphi(\gamma(x^*)) = \Phi(x^*)$ pour tout $x \in X$, comme désiré.

Soit x^* une solution globale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$, et $y^* \triangleq \gamma(x^*) \in \Omega$. Pour tout $y \in \Omega$, on a $\varphi(y) \geq \varphi(\gamma(\chi(y))) = \Phi(\chi(y)) \geq \Phi(x^*) = \varphi(y^*)$. Cela prouve le résultat attendu. □

Proposition 5.2. Dans le cadre d'étude 5.1, les deux affirmations suivantes sont vraies :

1. Pour tout y^* solution locale de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$, si $y^* = \gamma(\chi(y^*))$ et γ est continue en $\chi(y^*)$, alors $\chi(y^*)$ est une solution locale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$.
2. Pour tout x^* solution locale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$, si χ est continue en $\gamma(x^*)$, alors $\gamma(x^*)$ est une solution locale de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$.

Démonstration. Soit y^* une solution locale de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$, telle que $y^* = \gamma(\chi(y^*))$ et γ est continue en $\chi(y^*)$. Il existe un voisinage $\mathcal{Y} \subseteq \mathbb{Y}$ de y^* tel que $y^* \in \operatorname{argmin} \varphi(\mathcal{Y} \cap \Omega)$, et un voisinage $\mathcal{X} \subseteq \mathbb{X}$ de x^* tel que $\gamma(x) \in \mathcal{Y}$ pour tout $x \in \mathcal{X} \cap X$. Ainsi, pour tout $x \in \mathcal{X} \cap X$, on a $\Phi(x) = \varphi(\gamma(x)) \geq \varphi(y^*) = \varphi(\gamma(x^*)) = \Phi(x^*)$, et le résultat suit.

Soit x^* une solution locale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ telle que χ est continue en $\gamma(x^*)$. Il existe un voisinage $\mathcal{X} \subseteq \mathbb{X}$ de x^* tel que $\Phi(x) \geq \Phi(x^*)$ pour tout $x \in \mathcal{X}$, et un voisinage $\mathcal{Y} \subseteq \mathbb{Y}$ de $\gamma(x^*)$ tel que $\chi(\mathcal{Y}) \subseteq \mathcal{X}$. Ainsi, $\varphi(y) \geq \varphi(\gamma(\chi(y))) = \Phi(\chi(y)) \geq \Phi(x^*) = \varphi(\gamma(x^*))$ pour tout $y \in \mathcal{Y} \cap \Omega$. Le résultat suit car on a de plus $\gamma(x^*) \in \Omega$. □

5.1.3 Approches similaires dans la littérature

Le cadre de l'optimisation partitionnée est équivalent à celui de l'*optimisation paramétrique* [24, 107], qui étudie le cas où la fonction objectif ou les contraintes dépendent d'un paramètre réglable. Cependant, nous conservons notre différence de terminologie car la dénomination *optimisation partitionnée* rappelle que le problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ n'est pas paramétrique

et que la nature du paramètre du sous-problème ($\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(x)$) découle du choix de la partition de $\mathbb{Y}$, qui n'est pas imposée mais est définie par l'utilisateur. Nous n'avons pas connaissance de référence en optimisation paramétrique donnant des garanties de trouver une solution locale dans l'intégralité du cadre d'étude 5.1. Notons toutefois l'étude de la continuité locale de γ dans un contexte discontinu effectuée par [45], qui pourrait servir dans des travaux futurs liés à l'optimisation partitionnée.

L'optimisation partitionnée est également un cas particulier d'*optimisation bi-niveau* [36], c'est-à-dire l'optimisation de problèmes dans lesquels certaines variables sont fixées comme des solutions d'un sous-problème d'optimisation. Cependant, les deux cadres présentent une différence de contexte. En effet, le problème ($\mathbf{P}_{\texttt{PO}}^{\texttt{ini}}$) ne comporte qu'un seul niveau. De plus, ici la reformulation ($\mathbf{P}_{\texttt{PO}}^{\texttt{ref}}$) et le sous-problème ($\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(x)$) sont les deux facettes d'une restriction du problème ($\mathbf{P}_{\texttt{PO}}^{\texttt{ini}}$), alors que le cadre bi-niveau décrit habituellement une structure meneur-suiveur conflictuelle. La littérature sur l'optimisation bi-niveau discute de problèmes non lisses [83] et d'applications de `DSM` [38], mais nous n'avons pas connaissance de travaux identifiant des solutions locales dans ce cas.

La terminologie *optimisation partitionnée* est déjà employée dans la littérature en `DFO`, comme dans [77] ou dans les méthodes fondées sur l'algorithme DIRECT [68]. Cependant, dans ces références la partition est *discrète*, au sens où ces méthodes partitionnent récursivement un hyperrectangle en hyperrectangles plus petits. Ceci diffère de notre objectif ici, puisque ces partitions ne visent pas à simplifier la structure du problème restreint à chaque ensemble de partition. Par conséquent, notre stratégie de partitionnement *continu* peut être nommé comme de l'optimisation partitionnée sans causer de confusion avec la terminologie existante.

La `PO` a également des liens avec les techniques de décomposition en sous-espaces, utilisées en `DFO` pour traiter les problèmes présentant beaucoup de variables [31, 105]. Ces techniques visent à déterminer des sous-espaces plus pertinents que d'autres pour l'optimisation. Le choix d sous-espace retenu varie d'itération en itération, afin d'asymptotiquement étudier l'espace entier. En un sens, la `PO` peut être vue comme un cas particulier de technique de décomposition en sous-espaces, applicable lorsque les sous-espaces contenant les principales difficultés du problème sont identifiés à priori et qu'il est possible d'effectuer une minimisation de la fonction objectif sur les autres sous-espaces de façon rapide.

La `PO` peut également être rapprochée de l'*optimisation sur variété* [27, 67], pour contribuer à la résolution efficace du sous-problème ($\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(x)$) lorsque chacun des ensembles de partition de $\mathbb{Y}$ est une variété (c'est-à-dire que $\mathbb{Y}(x)$ est homéomorphe à $\mathbb{R}^p$ pour un certain $p \in \mathbb{N}^*$, pour tout $x \in \mathbb{X}$). Nous pouvons conjecturer qu'en pratique, il est commun que les ensembles de partition de $\mathbb{Y}$ soient bel et bien des variétés de $\mathbb{Y}$ lorsque $\mathbb{Y}$ est de dimension finie.

5.2 Application de cDSM au problème reformulé

On se place dans le cadre d'étude 5.1. Nous démontrons au chapitre 5.2.1 que la reformulation $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ est une instance du problème $(\mathbf{P})$. Il vient que l'algorithme cDSM est apte à résoudre le problème $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$, et qu'une solution locale au problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ en découle. Le théorème 5.4 formalise cette affirmation. Une généralisation suit au chapitre 5.2.2.

5.2.1 Convergence de cDSM vers des solutions au problème initial

Ce chapitre vise à clamer et prouver le théorème 5.4, qui stipule qu'on peut obtenir des solutions au problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ en résolvant la reformulation $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ via cDSM.

Analyse de convergence de cDSM

Prouvons d'abord que l'analyse de convergence de cDSM s'applique au problème $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$. Cela signifie que le cadre d'étude 4.2 est valide ici ; à l'exception possible de la semi-continuité de Φ. On peut appliquer le théorème 4.8 si le cadre 4.2 est entièrement valide, ou sa variante donnée au théorème 4.13 si seule la semi-continuité de Φ n'est pas vérifiée. Des exemples au chapitre 6 montrent que Φ peut bel et bien ne pas être semi-continue dans certains cas.

Proposition 5.3. Si le problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ entre dans le cadre d'étude 5.1 et vérifie les hypothèses 5.1 et 5.2, alors la reformulation $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ est garantie de vérifier tous les prérequis du cadre d'étude 4.2 à l'exception de la semi-continuité de Φ.

Démonstration. Considérons le cadre d'étude 5.1 sous les hypothèses 5.1 et 5.2, et montrons que

1. Φ est bornée inférieurement et a ses ensembles de niveau bornés ;
2. il existe une partition de X en $X = \sqcup_{i=1}^{N} X_i$ où $N \in \mathbb{N}^*$ et pour tout $i \in [\![1, N]\!]$, X_i est ample et $\Phi_{|X_i}$ est continue (ici, il n'y a pas de contraintes dans la reformulation $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$, donc dans la notation du cadre 4.2, on a $X_0 \triangleq \emptyset$).

Pour prouver le premier point, posons $\alpha \in \mathbb{R}$. On a $\Phi(x) = \alpha$ si et seulement si $\varphi(\gamma(x)) = \alpha$ si et seulement si $\gamma(x) \in \varphi^{-1}_{|\gamma(X)}(\{\alpha\})$, qui implique que $x \in \chi(\varphi^{-1}_{|\gamma(X)}(\{\alpha\}))$. Or, on a $\varphi^{-1}_{|\gamma(X)}(\{\alpha\})$ borné par l'hypothèse 5.2, donc $\chi(\varphi^{-1}_{|\gamma(X)}(\{\alpha\}))$ est borné par l'hypothèse 5.1. Par conséquent, $\Phi(x) = \alpha$ implique que x appartient à un ensemble borné. De plus, $\inf \Phi(\mathbb{X}) = \inf \Phi(X) = \inf \varphi(\gamma(X)) > -\infty$ donc Φ est bornée inférieurement.

Pour établir le second point, prenons la partition de $X = \sqcup_{i=1}^{N} X_i$ définie dans le cadre 5.1. Pour tout $i \in [\![1, N]\!]$, on a directement X_i ample, et la continuité de Φ découle de $\Phi_{|X_i} = (\varphi_{|\gamma(X_i)}) \circ (\gamma_{|X_i})$ exprimée comme composée de fonctions continues. □

On peut alors en déduire que le théorème 4.13 est applicable. En exploitant également la proposition 5.2, on peut même en déduire le résultat suivant. [1]

Théorème 5.4. Si le problème $(\mathbf{P}_{\texttt{PO}}^{\texttt{ini}})$ entre dans le cadre d'étude 5.1 et vérifie les hypothèses 5.1 et 5.2, alors `cDSM` résolvant le problème reformulé $(\mathbf{P}_{\texttt{PO}}^{\texttt{ref}})$ génère au moins une suite raffinante ; et pour toute suite raffinante (indexée par $K^* \subseteq \mathbb{N}$, de point raffiné noté x^*), x^* admet un voisinage $\mathcal{X}^*$ tel que

$$\emptyset \neq \operatorname{adh}\left((\gamma(x^k))_{k\in K^*}\right) \subseteq \mathcal{Y}^* \cap \Omega \quad \text{et} \quad \lim_{k\in K^*} \varphi(\gamma(x^k)) = \inf \varphi\left(\mathcal{Y}^* \cap \Omega\right),$$

où $\mathcal{Y}^* \triangleq \mathbb{Y}(\mathcal{X}^*)$ est un voisinage de tout point de $\operatorname{adh}((\gamma(x^k))_{k\in K^*})$, et si de plus φ est semi-continue en tout point de $\mathcal{V}_\gamma(x^*)$, alors

$$\varphi(\gamma(x^*)) = \min \varphi\left(\mathcal{Y}^* \cap \Omega\right).$$

Le théorème 5.4 clame que tout point d'accumulation de la suite $(\gamma(x^k))_{k\in K^*}$ est une solution locale à $(\mathbf{P}_{\texttt{PO}}^{\texttt{ini}})$ dans un sens généralisé, et que si de plus φ est semi-continue en tout point de $\mathcal{V}_\gamma(x^*)$ (l'ensemble des valeurs limites possibles de $\gamma(x)$ lorsque $x \to x^*$), alors $\gamma(x^*)$ est une solution locale à $(\mathbf{P}_{\texttt{PO}}^{\texttt{ini}})$ au sens usuel. Pour prouver ce théorème, nous devons d'abord prouver trois lemmes donnés ci-dessous. La preuve suit.

Lemmes préliminaires

Les lemmes 5.5 et 5.6 montrent des propriétés utiles de χ et de γ. Le lemme 5.7 met en évidence l'intérêt de la continuité uniforme de $\gamma_{|X_i}$ et du nombre localement fini d'ensembles de continuité de f dans l'hypothèse 5.2.

Lemme 5.5. Si χ est continue, alors pour tout $\mathcal{X} \subseteq \mathbb{X}$ ouvert, $\mathbb{Y}(\mathcal{X}) \subseteq \mathbb{Y}$ est ouvert.

Démonstration. Supposons χ continue. Soit $\mathcal{X} \subseteq \mathbb{X}$ ouvert et $y \in \mathbb{Y}(\mathcal{X})$. Alors $\chi(y) \in \mathcal{X}$, et il existe $\mathcal{Y} \subseteq \mathbb{Y}$ ouvert contenant y et tel que $\chi(\mathcal{Y}) \subseteq \mathcal{X}$. Ainsi, $y \in \mathcal{Y} \subseteq \mathbb{Y}(\mathcal{X})$. □

1. Rappelons la notation des *valeurs limites* d'une fonction en un point, donnée au chapitre 1.2.1 et qui intervient ici. L'ensemble des *valeurs limites* de γ en $x \in X$ est l'ensemble $\mathcal{V}_\gamma(x) \triangleq \cap_{r>0} \operatorname{cl}(\gamma(\mathcal{B}_r(x) \cap X))$.

Lemme 5.6. La fonction oracle $\gamma : X \to Y$ est une bijection de X dans $\gamma(X)$, et sa réciproque est la restriction de la fonction d'indice $\chi : \mathbb{Y} \to \mathbb{X}$ à $\gamma(X)$.

Démonstration. Pour tout couple $(x_1, x_2) \in X^2$ avec $x_1 \neq x_2$, on a $Y(x_1) \cap Y(x_2) = \emptyset$, et $\gamma(x_1) \in \mathbb{Y}(x_1)$ et $\gamma(x_2) \in \mathbb{Y}(x_2)$. Il vient donc que γ est bijective de X dans $\gamma(X)$. De plus $\gamma(x) \in \mathbb{Y}(x)$ mène à $\chi(\gamma(x)) = x$ par définition de χ, pour tout $x \in X$. □

Lemme 5.7. Sous l'hypothèse 5.2, pour tout $x \in \mathrm{cl}(X)$ et toute suite $(x^k)_{k\in\mathbb{N}}$ d'éléments de X convergeant vers x, il existe $i \in [\![1, N]\!]$ tel que $K_i \triangleq \{i \in [\![1, N]\!] : x^k \in X_i\}$ est infini, et de plus $(\gamma(x^k))_{k\in K_i}$ admet une limite pour tout $i \in [\![1, N]\!]$ tel que K_i est infini.

Démonstration. Soit $x \in \mathrm{cl}(X)$ et $(x^k)_{k\in\mathbb{N}}$ une suite d'éléments de X convergeant vers x. Soit $r \in \mathbb{R}_+^*$ tel que $x^k \in \mathcal{B}_r(x)$ pour tout $k \in \mathbb{N}$, et soit $K_i \triangleq \{i \in [\![1, N]\!] : x^k \in X_i\}$ pour tout $i \in [\![1, N]\!]$. Par l'hypothèse 5.2, l'ensemble $\mathcal{I} \triangleq \{i \in [\![1, N]\!] : X_i \cap \mathcal{B}_r(x) \neq \emptyset\}$ est fini. Ainsi, pour tout $i \notin \mathcal{I}$, $K_i = \emptyset$. Comme de plus $\mathbb{N} = \sqcup_{i\in[\![1,N]\!]} K_i$, il vient qu'au moins l'un des ensembles K_i, $i \in \mathcal{I}$, est infini. De plus, pour tout $i \in [\![1, N]\!]$ tel que K_i est infini, $(\gamma(x^k))_{k\in K_i}$ admet une limite par continuité uniforme de $\gamma_{|X_i}$. □

Démonstration du théorème 5.4

La preuve repose sur le théorème 4.13 appliqué à la reformulation ($\mathbf{P}_{\texttt{PO}}^{\texttt{ref}}$), et l'utilisation de la proposition 2 pour en déduire une solution locale du problème ($\mathbf{P}_{\texttt{PO}}^{\texttt{ini}}$).

Démonstration. **Application du théorème 4.13.** La proposition 5.3 garantit que le théorème 4.13 est applicable. Ainsi, il existe $K^* \subseteq \mathbb{N}$ indexant une suite raffinante générée par `cDSM`. Le voisinage $\mathcal{X}^* \triangleq \mathcal{B}_r(x^*)$ de $x^* \triangleq \lim_{k\in K^*} x^k$ vérifie

$$\lim_{k\in K^*} \Phi(x^k) = \begin{cases} \min \Phi(\mathcal{X}^*) = \Phi(x^*) & \text{si } \Phi \text{ est semi-continue en } x^*, \\ \inf \Phi(\mathcal{X}^*) & \text{sinon.} \end{cases} \tag{5.1}$$

On note $\mathcal{Y}^* \triangleq \mathbb{Y}(\mathcal{X}^*)$. C'est un ouvert de $\mathbb{Y}$, d'après le lemme 5.5.

Points adhérents à $(\gamma(x^k))_{k\in K^*}$. Montrons que $\emptyset \neq \mathrm{adh}((\gamma(x^k))_{k\in K^*}) \subseteq \mathcal{Y}^* \cap \Omega$. Le lemme 5.7 assure que $\mathrm{adh}((\gamma(x^k))_{k\in K^*}) \neq \emptyset$. Notons $y \in \mathrm{adh}((\gamma(x^k))_{k\in K^*})$, et $K \subseteq K^*$ tel que $(\gamma(x^k))_{k\in K} \to y$. On a $\chi(y) = \lim_{k\in K} \chi(\gamma(x^k)) = \lim_{k\in K} x^k$ par continuité de χ et le lemme 5.6. On a alors $\chi(y) = x^*$, et donc $y \in \mathbb{Y}(x^*) \subset \mathcal{Y}^*$. Enfin, $y \in \Omega$ car $\gamma(x^k) \in \Omega$ pour tout $k \in K^*$ et Ω est fermé.

Convergence de $(\varphi(\gamma(x^k)))_{k\in K^*}$. On a $\varphi(\gamma(x^k)) \geq \inf \varphi(\mathcal{Y}^* \cap \Omega)$ pour tout $k \in K^*$, et $\varphi(y) \geq \varphi(\gamma(\chi(y))) = \Phi(\chi(y)) \geq \inf \Phi(\mathcal{X}^*) = \lim_{k\in K^*} \Phi(x^k) = \lim_{k\in K^*} \varphi(\gamma(x^k))$ pour tout $y \in \mathcal{Y}^* \cap \Omega$. Ainsi, $\lim_{k\in K^*} \varphi(\gamma(x^k)) = \inf \varphi\left(\mathcal{Y}^* \cap \Omega\right)$.

Optimalité locale de $\gamma(x^*)$ si φ est semi-continue en tout point de $\mathcal{V}_\gamma(x^*)$. Supposons la semi-continuité de φ en tout point de $\mathcal{V}_\gamma(x^*)$. Prouvons que Φ est semi-continue en x^*. Soit $(z^k)_{k\in\mathbb{N}}$ une suite d'éléments de $\mathcal{X}^* \cap X$ convergeant vers x. Par le lemme 5.7, $\emptyset \neq \mathrm{adh}((\gamma(z^k))_{k\in\mathbb{N}}) \subseteq \mathcal{V}_\gamma(x^*)$. Soit $y \in \mathrm{adh}((\gamma(x^k))_{k\in K^*})$ et $K \subseteq \mathbb{N}$ tel que $\gamma(z^k) \to y$. Alors $\liminf_{k\in K} \Phi(z^k) = \liminf_{k\in K} \varphi(\gamma(z^k)) \geq \varphi(y)$ par semi-continuité de φ en y. De plus, $\chi(y) = x^*$, donc $y \in \mathbb{Y}(x^*)$. Ainsi, $\varphi(y) \geq \varphi(\gamma(x^*)) = \Phi(x^*)$. Il vient qu'alors Φ est semi-continue en x^*. La preuve s'achève alors via l'équation (5.1), qui implique que x^* est une solution locale au problème $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ sur le voisinage $\mathcal{X}^*$, et via la proposition 5.2.2 qui clame alors que $\gamma(x^*)$ est une solution locale au problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$. Il est de plus aisé de prouver que $\gamma(x^*)$ est optimale sur son voisinage $\mathcal{Y}^*$. □

5.2.2 Pistes de généralisations possibles de l'analyse

Il est possible que `cDSM` génère un point raffiné x^* vérifiant $\varphi(\gamma(x^*)) = \min \varphi\left(\mathcal{Y}^* \cap \Omega\right)$ alors que φ n'est pas semi-continue en tout point de $\mathcal{V}_\gamma(x^*)$. Nous illustrons ce fait au chapitre 6.1. Le théorème 5.4 est donc généralisable, comme suit.

Théorème 5.8. Si le problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ entre dans le cadre d'étude 5.1 et vérifie les hypothèses 5.1 et 5.2, alors `cDSM` résolvant le problème reformulé $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ génère au moins une suite raffinante ; et pour toute suite raffinante (indexée par $K^* \subseteq \mathbb{N}$) de point raffiné x^*, x^* admet un voisinage $\mathcal{X}^*$ tel que

$$\emptyset \neq \mathrm{adh}\left((\gamma(x^k))_{k\in K^*}\right) \subseteq \mathcal{Y}^* \cap \Omega \quad \text{et} \quad \lim_{k\in K^*} \varphi(\gamma(x^k)) = \inf \varphi\left(\mathcal{Y}^* \cap \Omega\right),$$

où $\mathcal{Y}^* \triangleq \mathbb{Y}(\mathcal{X}^*)$ est un voisinage de tout point de $\mathrm{adh}((\gamma(x^k))_{k\in K^*})$. De plus,

$$\varphi(\gamma(x^*)) = \min \varphi\left(\mathcal{Y}^* \cap \Omega\right) \qquad \Longleftrightarrow \qquad \forall K \subseteq K^*, \quad \liminf_{k\in K} \varphi(\gamma(x^k)) \geq \varphi(\gamma(x^*)).$$

Ainsi, tout point d'accumulation de la suite $(\gamma(x^k))_{k\in K^*}$ est une solution locale au problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ dans un sens généralisé, et de plus $\gamma(x^*)$ est une solution locale à $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ au sens usuel si et seulement si la dernière condition est vérifiée. Le théorème 5.8 est strictement plus général que le théorème 5.4, mais la preuve est similaire. Cependant, en pratique le théorème 5.4 est possiblement plus utile en pratique, car la semi-continuité de φ en tout point de $\mathcal{V}_\gamma(x^*)$ est parfois simple à étudier à priori alors que la condition $\liminf_{k\in K} \varphi(\gamma(x^k)) \geq \varphi(\gamma(x^*))$ ne peut être observée qu'à posteriori.

5.3 Remarques sur l'optimisation partitionnée et extensions potentielles

Le cadre de l'optimisation partitionnée représente une avancée vers l'augmentation de la dimension maximale que les méthodes sans dérivées peuvent traiter. Cependant, ce cadre est restrictif sur certains aspects, que nous identifions au chapitre 5.3.1. Le chapitre 5.3.2 propose alors des allègements du cadre via une définition moins contraignante de la fonction oracle γ. Enfin, le chapitre 5.3.3 entrevoit le développement de méthodes *hybrides*, exploitant des méthodes de la DFO collaborant avec des méthodes d'autres domaines de l'optimisation.

5.3.1 Commentaires sur le cadre actuel

Dans le cadre de la PO, nous imposons un partitionnement de l'intégralité de $\mathbb{Y}$. Ceci est adaptable, en ne partitionnant que l'espace réalisable $\Omega \subseteq \mathbb{Y}$. On peut poser $\Omega = \sqcup_{x\in\mathbb{X}}\Omega(x)$, et alors définir $X \triangleq \{x \in \mathbb{X} : Y \cap \Omega(x) \neq \emptyset\}$. Les définitions de γ et du reste des éléments du cadre sont alors inchangées. De même, les indices de partitionnement peuvent ne pas parcourir l'intégralité de $\mathbb{X}$. Le cadre autorise $\mathbb{Y}(x) = \emptyset$ pour certains $x \in \mathbb{X}$.

La dimension de $\mathbb{X}$ est importante pour l'applicabilité du cadre de la PO. Le cadre autorise $\mathbb{X}$ à être de dimension 0, mais ce cas est inutile ($\mathbb{X} \triangleq \{0\}$ donc $\mathbb{Y}(0) \triangleq \mathbb{Y}$, donc $\gamma(0)$ est la solution globale du problème, supposé connu). On peut également avoir $\mathbb{X}$ et $\mathbb{Y}$ de même dimension finie. Ce cas a un intérêt théorique, car il englobe par exemple le cas où $\mathbb{X}$ et $\mathbb{Y}$ sont homéomorphes. On pourrait donc étudier l'influence d'une déformation continue de l'espace des variables sur la performance d'un algorithme. Toutefois, la PO est pensée pour $\mathbb{Y}$ de dimension $d_{\mathbb{Y}} \geq 2$, possiblement infinie, et $\mathbb{X}$ est de dimension $0 < d_{\mathbb{X}} \ll d_{\mathbb{Y}}$ finie. De plus, même si $\mathbb{Y}$ peut être quelconque, $\mathbb{X}$ est en général l'espace $\mathbb{R}^{d_{\mathbb{X}}}$ usuel, et en pratique on doit avoir $d_{\mathbb{X}} \leq 50$ pour résoudre la reformulation ($\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}}$) par un solveur de la DFO [19, 123].

Les défauts pratiques de la PO sont les suivants. D'abord, elle requiert de la structure sur le problème ($\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}}$), puisqu'on suppose que le problème est simplifié par la restriction à n'importe quel ensemble de partition. Dans la terminologie de la DFO, la PO s'applique donc aux *boîtes grises* [6]. Ensuite, sur chacun des ensembles de partition, la fonction objectif doit présenter un et un seul minimiseur global, qui doit de plus être explicitement accessible. Toutefois, des pistes de recherche allègent ce prérequis (voir chapitre 5.3.2), et en pratique on peut se contenter d'une approximation du minimiseur global (par exemple fournie par un algorithme d'optimisation globale [66, 87] usuel ou valide dans des cas non-lisses [43, 68]). On pourrait aussi analyser la sensibilité du sous-problème ($\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x)$) pour estimer le comportement local de γ via le théorème des fonctions implicites généralisé au cas non-lisse [98], suivant une idée proposée par [45] pour l'optimisation paramétrique.

Enfin, un défaut théorique du théorème 5.4 est sa dépendance en l'hypothèse 5.2, qui impose beaucoup de structure ad hoc sur le problème et le partitionnement. Par exemple, puisque $\gamma(x) \in \mathbb{Y}(x)$ pour tout $x \in X$ et que les ensembles $\mathbb{Y}(\cdot)$ sont tous disjoints, la continuité de $\gamma_{|\mathcal{X}}$ pour n'importe quel $\mathcal{X} \subseteq X$ impose que $\gamma(x) \in \partial\mathbb{Y}(x)$ pour tout $x \in \mathcal{X}$. Dit autrement, s'il existe $x \in X$ tel que $\gamma(x) \in \mathrm{int}(\mathbb{Y}(x))$, alors $\gamma_{|\mathcal{X}}$ est discontinue en x pour tout $\{x\} \neq \mathcal{X} \subseteq \mathbb{X}$ contenant x. L'hypothèse 5.2 ne peut donc pas tenir en x.

5.3.2 Définitions moins strictes de la fonction d'oracle

La fonction d'oracle γ est la pierre angulaire de la PO. Heureusement, même si elle est centrale, sa définition actuellement stricte peut être allégée. Dans ce chapitre, nous discutons de tels allègements, pour viser une définition bien plus exploitable en pratique. La finalité de ce chapitre est de montrer qu'on pourrait adapter le cadre d'étude 5.1 pour que pour tout $x \in X$, $\gamma(x)$ ne représente qu'une approximation d'une solution locale du sous-problème $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$. Le cadre en serait alors d'autant plus exploitable, puisqu'en pratique on résout souvent le sous-problème $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$ par un algorithme d'optimisation qui ne renvoie que des approximations de solutions locales.

Inexistence ou non-unicité de la solution globale du sous-problème

On peut supprimer l'hypothèse d'existence et d'unicité de la solution globale du sous-problème dans la définition de γ. Pour tout $x \in \mathbb{X}$, on note $\mathcal{S}(x) \subseteq \mathbb{Y}(x)$ l'ensemble des solutions globales de $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$ de valeur objectif finie. On définit ensuite $X \triangleq \{x \in \mathbb{X} : \mathcal{S}(x) \neq \emptyset\}$. Enfin, on définit $\gamma : X \to \mathbb{Y}$ comme un oracle déterministe qui, pour tout $x \in X$, renvoie un élément de $\mathcal{S}(x)$ (cet élément doit toujours être le même pour deux évaluations successives de $\gamma(x)$). Le reste du cadre, des hypothèses et de l'analyse demeurent inchangés.

Oracle renvoyant une solution locale

La définition de l'oracle γ peut être allégée pour ne considérer qu'une solution locale. Pour tout $x \in \mathbb{X}$, on note $\mathcal{S}(x) \subseteq \mathbb{Y}(x)$ l'ensemble des solutions locales de $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$ de valeur objectif finie. On pose $X \triangleq \{x \in \mathbb{X} : \mathcal{S}(x) \neq \emptyset\}$, et on définit $\gamma : X \to \mathbb{Y}$ comme un oracle déterministe qui, pour tout $x \in X$, renvoie un élément de $\mathcal{S}(x)$ (cet élément doit toujours être le même pour deux évaluations successives de $\gamma(x)$). Nous supposons que γ est tel que pour tout $\mathcal{X} \subseteq X$, s'il existe une application $u : \mathcal{X} \to \cup_{x\in\mathcal{X}}\mathcal{S}(x)$ continue vérifiant $u(x) \in \mathcal{S}(x)$ pour tout $x \in \mathcal{X}$, alors $\gamma_{|\mathcal{X}}$ est continue. Cette hypothèse traduit un principe que l'on pourrait énoncer par « γ reste autant que possible dans un bassin

d'attraction donné ». Enfin, pour tout $x \in X$, nous notons $\mathcal{Y}(x)$ le plus grand ouvert de $\mathbb{Y}(x)$ (pour la topologie de $\mathbb{Y}$ induite par $\mathbb{Y}(x)$) tel que $\varphi(\gamma(x)) = \min \varphi(\mathcal{Y}(x) \cap \Omega)$, et on suppose que $\underline{r} \triangleq \inf\{r : (\mathcal{B}_r(\gamma(x)) \cap \mathbb{Y}(x)) \subseteq \mathcal{Y}(x) \text{ pour tout } x \in X\} > 0$. Ceci traduit que, pour tout $x \in X$, le plus grand voisinage sur lequel $\gamma(x)$ est une solution locale de $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$ est $\mathcal{Y}(x)$, et que tous les points de $\mathbb{Y}(x)$ à distance inférieure à $\underline{r}$ de $\gamma(x)$ sont dans $\mathcal{Y}(x)$.

Il faut alors modifier l'étude faite au chapitre 5.1.2 sur les liens entre les solutions du problèmes $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ et celles du problème $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$.

Lemme 5.9. Soit $\mathbb{Y}$ un espace normé partitionné en $\mathbb{Y} = \sqcup_{x \in \mathbb{X}} \mathbb{Y}(x)$, où $\mathbb{X}$ est un espace normé, tel que la fonction $\chi : \mathbb{Y} \to \mathbb{X}$ d'indice est continue. Soit $\mathcal{Y} : \mathbb{X} \to 2^{\mathbb{Y}}$, telle que pour tout $x \in \mathbb{X}$, $\mathcal{Y}(x) \subseteq \mathbb{Y}(x)$ est ouvert en topologie induite par $\mathbb{Y}(x)$. Soit $X \subseteq \mathbb{X}$ ample et $\gamma : X \to \mathbb{Y}$ telle que $\gamma(x) \in \mathcal{Y}(x)$ pour tout $x \in X$. S'il existe un ouvert $\mathcal{X} \subseteq \mathbb{X}$ tel que $\gamma_{|\mathcal{X} \cap X}$ est continue et $\underline{r} \triangleq \inf\{r : (\mathcal{B}_r(\gamma(x)) \cap \mathbb{Y}(x)) \subseteq \mathcal{Y}(x) \text{ pour tout } x \in \mathcal{X}\} > 0$, alors $\gamma(\mathcal{X} \cap X) \subseteq \mathrm{cl}(\mathrm{int}(\mathcal{Y}(\mathcal{X})))$.

Démonstration. Soit $\mathcal{X} \subseteq \mathbb{X}$ ouvert tel que γ est continue sur $\mathcal{X}$ et $\underline{r} > 0$. Dans un premier temps, soit $x \in \mathrm{int}(\mathcal{X} \cap X)$, et $\rho > 0$ tel que $\mathcal{B}_\rho(x) \subseteq \mathcal{X}$. Par continuité de χ sur $\mathbb{Y}$, on a $\chi(y) \to x$ lorsque $y \to \gamma(x)$. Par continuité de γ en x, on a $\gamma(x') \to \gamma(x)$ lorsque $x' \to x$. De ces deux faits, on déduit qu'il existe $\mu > 0$ vérifiant $\chi(y) \in \mathcal{X}$ et $y \in \mathcal{B}_{\underline{r}}(\gamma(\chi(y))) \cap \mathbb{Y}(\chi(y))$ pour tout $y \in \mathcal{B}_\mu(\gamma(x))$. Il vient alors que la boule $\mathcal{B}_\mu(\gamma(x))$ est incluse dans $\cup_{x \in \mathcal{X}} \mathcal{B}_{\underline{r}}(\gamma(x)) \cap \mathbb{Y}(x)$, donc à fortiori dans $\cup_{x \in \mathcal{X}} \mathcal{Y}(x) = \mathcal{Y}(\mathcal{X})$, ce qui prouve que $\gamma(x) \in \mathrm{int}(\mathcal{Y}(\mathcal{X}))$. Dans un second temps, soit $x \in \partial(\mathcal{X} \cap X)$. Puisque $\mathcal{X} \cap X$ est ample (lemme 4.9.a)), on a $x \in \mathrm{cl}(\mathrm{int}(\mathcal{X} \cap X))$. Le résultat suit en considérant une suite d'éléments de $\mathrm{int}(\mathcal{X} \cap X)$ convergeant vers x. □

Proposition 5.10. Dans le cadre d'étude de ce sous-chapitre,

1. Pour tout y^* solution locale de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$, on a $y^* \in \mathcal{S}(x^*)$, où $x^* \triangleq \chi(y^*)$. Si de plus $y^* = \gamma(x^*)$ et γ est continue en x^*, alors x^* est une solution locale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$.
2. Pour tout x^* solution locale de la reformulation $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$, si γ est continue en x^* et χ est continue en $\gamma(x^*)$, alors $\gamma(x^*)$ est une solution locale du problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$.

Démonstration. Soit y^* une solution locale de $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$. Il existe un voisinage $\mathcal{Y} \subseteq \mathbb{Y}$ de y^* tel que $y^* \in \mathrm{argmin}\, \varphi(\mathcal{Y} \cap \Omega)$. Par conséquent, $y^* \in \mathrm{argmin}\, \varphi(\mathcal{Y} \cap Y_\Omega(x^*))$, et donc $y^* \in \mathcal{S}(x^*)$. Si de plus $y^* = \gamma(x^*)$ et γ est continue en x^*, alors il existe un voisinage $\mathcal{X} \subseteq \mathbb{X}$ de x^* tel que pour tout $x \in \mathcal{X} \cap X$, $\gamma(x) \in \mathcal{Y}$. Cela implique que pour tout $x \in \mathcal{X} \cap X$, $\varphi(\gamma(x)) \geq \varphi(y^*) = \varphi(\gamma(x^*))$. Ainsi, x^* est une solution locale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$.

Soit x^* une solution locale de $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ telle que γ est continue en x^* et χ est continue en $\gamma(x^*)$. Il existe un ouvert $\mathcal{X} \subseteq \mathbb{X}$ contenant x^* tel que $\gamma_{|\mathcal{X}\cap X}$ est continue, χ est continue sur $\mathbb{Y}(\mathcal{X})$, et $\Phi(x) \geq \Phi(x^*)$ pour tout $x \in \mathcal{X}$. Par le lemme 5.9, $\mathcal{Y}^* \triangleq \mathrm{int}(\mathcal{Y}(\mathcal{X})) \subseteq \mathbb{Y}$ est un ouvert contenant $\gamma(x^*)$. De plus, $\varphi(y) \geq \varphi(\gamma(\chi(y))) = \Phi(\chi(y)) \geq \Phi(x^*) = \varphi(\gamma(x^*))$ pour tout $y \in \mathcal{Y}^* \cap \Omega$, et $\gamma(x^*) \in \Omega$. Le résultat en découle. □

Ceci corrigé, le théorème 5.4, montrant que `cDSM` résolvant la reformulation $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ mène à des solutions du problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$, est inchangé.

Oracle renvoyant une approximation de solution locale

Enfin, une dernière extension de γ est envisageable. En pratique, $\gamma(x)$ est souvent obtenu pour tout $x \in X$ via un algorithme d'optimisation résolvant le sous-problème $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$. Or, en pratique un algorithme est interrompu après un temps fini, et donc il ne renvoie pas exactement une solution locale. Par conséquent, pour modéliser exactement une situation réelle, il faudrait que $\gamma(x)$ représente une approximation d'une solution à une tolérance fixée. La formalisation de ce cadre n'est pas encore achevée au moment de la rédaction de cette thèse, aussi nous ne poursuivons pas la discussion au-delà de ce point.

5.3.3 Méthodes hybrides

Le cadre de l'optimisation partitionnée peut être vu comme un prototype de méthode *hybride*, utilisant conjointement un algorithme sans dérivées (pour résoudre le problème $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$) et d'autres classes d'algorithmes (pour résoudre le sous-problème $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$ pour tout $x \in X$). Le développement de telles méthodes est un projet à peine commencé au moment de la rédaction de cette thèse, donc la discussion qui suit manque de formalisme. Toutefois, nous croyons déjà au potentiel de telles méthodes. Elles traiteraient les aspects non-lisses du problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ par une méthode sans dérivées pensée pour cet usage, et laisseraient les aspects lisses et structurés à une méthode dédiée et qui peut être choisie indépendemment de la méthode sans dérivées. Ainsi, ces méthodes hybrides pourraient prendre le meilleur des différentes littératures. Le chapitre 7 illustre ce fait sur des applications en contrôle optimal.

Infaisabilité dans le sous-problème et barrière progressive dans la reformulation

Une piste de méthode hybride est de calculer l'infaisabilité du sous-problème $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$ pour alléger la définition de la fonction Φ, actuellement définie à $+\infty$ pour tout $x \in \mathbb{X}$ tel que $\gamma(x)$ ne peut pas être défini. Ceci revient à suivre une stratégie de *barrière extrême* [14], qui restrictive. En comparaison, la *barrière progressive* [15] est plus efficace.

Appliquer une barrière progressive dans notre cadre requiert deux éléments importants. À tout $x \in \mathbb{X}$, on doit pouvoir associer une valeur objectif et une valeur d'infaisabilité. En d'autres termes, on redéfinit la reformulation $(\mathbf{P}_{\texttt{PO}}^{\texttt{ref}})$ comme

$$\underset{x \in \mathbb{X}}{\text{minimiser}} \quad \widetilde{\Phi}(x) \quad \text{sous contrainte} \quad \widetilde{h}(x) \leq 0, \tag{$\widetilde{\mathbf{P}}_{\texttt{OP}}^{\texttt{ref}}$}$$

où $\widetilde{h}(x)$ représente l'infaisabilité du sous-problème $(\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(x))$ et où $\widetilde{\Phi}(x)$ représente la valeur objectif minimale associée à $(\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(x))$ (qu'on n'associera pas à $+\infty$ lorsque $\widetilde{h}(x) > 0$). Ce problème peut alors être résolu par un algorithme de `DFO` exploitant la barrière progressive.

Définissons d'abord une fonction $h : \mathbb{Y} \to \mathbb{R}_+$ servant de métrique de l'*infaisabilité* d'un point vis-à-vis de Ω. Un choix simple consiste à prendre $h(y) \triangleq \operatorname{dist}(y, \Omega)$ pour tout $y \in \Omega$. Cependant, lorsque $\Omega \triangleq c^{-1}(\mathbb{R}_-^k)$ est défini via $c : \mathbb{Y} \to \mathbb{R}^k$ semi-continue (avec $k \in \mathbb{N}^*$), on préfère la *fonction de violation de contrainte* [48] définie par $h(y) \triangleq \sum_{i=1}^k \max(0, c_i(y))^2$ pour tout $y \in \mathbb{Y}$. Ainsi, pour tout $x \in \mathbb{X}$, on peut définir $\widetilde{h}(x) \triangleq \min h(Y(x))$,[2] qui vaut 0 si et seulement si $(\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(x))$ est réalisable, donc si et seulement si $x \in X$. De là, on peut alors définir $\widetilde{\Phi}(x) \triangleq \varphi(\widetilde{\gamma}(x))$, où $\widetilde{\gamma}(x) = \gamma(x)$ si $x \in X$, et $\widetilde{\gamma}(x)$ est le minimiseur global de $h_{|Y(x)}$ autrement (supposé unique dans cette phrase, mais ceci est aisément relaxable).

Une telle définition de $\widetilde{\gamma}(x)$ est pertinente d'un point de vue appliqué, car lorsqu'on résout le sous-problème $(\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(x))$ numériquement, l'algorithme employé peut chercher en priorité à minimiser l'infaisabilité, puis à minimiser la valeur objectif à partir d'un point d'infaisabilité nulle. Notamment, le solveur **IPOPT** [118] adopte ce comportement.

Optimisation jointe plutôt que bi-niveau

Actuellement, pour tout $x \in X$, nous devons résoudre le sous-problème $(\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(x))$ pour récupérer $\gamma(x)$ et s'en servir dans la reformulation $(\mathbf{P}_{\texttt{PO}}^{\texttt{ref}})$. En ce sens, nous suivons une structure d'optimisation bi-niveau. Or, notre problème diffère du cadre usuel en bi-niveau, puisqu'ici les deux niveaux minimisent la valeur objectif. Une possible méthode hybride serait d'optimiser conjointement l'ensemble de partition considéré et le point considéré à l'intérieur de l'ensemble. Partons de $y^0 \in Y \cap \Omega$ et $x^0 \triangleq \chi(y^0)$. À chaque itération $k \in \mathbb{N}$, l'algorithme `DFO` propose un ensemble $\mathcal{T}^k$ de points autour de x^k. Pour tout $t \in \{x^k\} \cup \mathcal{T}^k$, on effectue quelques itérations d'un algorithme sur le problème $(\mathbf{P}_{\texttt{PO}}^{\texttt{sub}}(t))$ partant de y^k (qui est irréalisable pour ce problème). On récupère un point $\widetilde{\gamma}(t, y^k)$. Enfin, on note $\widetilde{\mathcal{T}}^k \triangleq \{t \in \{x^k\} \cup \mathcal{T}^k : \widetilde{\gamma}(t, y^k) \in \Omega\}$, puis on pose $t^k \in \operatorname{argmin}\{\varphi(\widetilde{\gamma}(t, y^k)) : t \in \widetilde{\mathcal{T}}^k\}$, et on pose $x^{k+1} \triangleq t^k$ et $y^{k+1} \triangleq \widetilde{\gamma}(t^k, y^k)$.

2. Ce minimum peut ne pas être défini, mais l'infimum existe toujours et le raisonnement peut être adapté. Nous ne discuterons pas des cas dégénérées menant à ce problème, car ceci nous emmènerait trop loin.

CHAPITRE 6 MISE EN APPLICATION DE L'OPTIMISATION PARTITIONNÉE

Toute généralisation est une hypothèse. Elle doit toujours, le plus tôt possible et le plus souvent possible, être soumise à la vérification.

« La Science et l'Hypothèse », Henri Poincaré.

Ce chapitre prouve que le cadre de l'optimisation partitionnée élargit le champ des possibles de la DFO. D'abord, le chapitre 6.1 illustre comment l'appliquer en pratique. Pour alléger la présentation, nous nous concentrons sur l'application de la PO pour simplifier des problèmes bruités. Ensuite, le chapitre 6.2 propose quatre problèmes de dimension trop grande pour la DFO usuelle, et compare les performances d'une instance naïve de DSM résolvant les problèmes reformulés avec des solveurs performants de la DFO résolvant les problèmes initiaux. Cette comparaison montre que l'optimisation partitionnée apporte un gain de performance dans la mise en application numérique. Précisons que l'étude de la performance des DSM n'est pas l'objectif de cette section. Précisons aussi que ce chapitre se concentre sur des problèmes de dimension finie. Une application en dimension infinie est illustrée au chapitre 7.

6.1 Optimisation partitionnée pour simplifier des problèmes bruités

Cette section illustre comment l'optimisation partitionnée peut aider à la résolution de problèmes dans lesquels l'objectif φ dépend d'un bruit ε dont on connaît les paramètres. En d'autres termes, lorsqu'on a $\varphi \triangleq \widetilde{\varphi} + \varepsilon \circ f$, où $\widetilde{\varphi} : \mathbb{Y} \to \mathbb{R}$ est simple à minimiser, $\varepsilon : \mathbb{R}^d \to \mathbb{R}$ est non-lisse, et $f : \mathbb{Y} \to \mathbb{R}^d$ est connue. Résoudre le problème $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ est difficile, à cause du bruit ε. Toutefois, ε devient constant si on altère $(\mathbf{P}^{\mathtt{ini}}_{\mathtt{PO}})$ via l'ajout d'une contrainte d'égalité $f(y) = x$, avec $x \in \mathbb{R}^d$ fixé. Cela donne une application du cadre de la PO, en partitionnant $\mathbb{Y}$ via $\mathbb{Y}(x) \triangleq \{y \in \mathbb{Y} : f(y) = x\}$ pour tout $x \in \mathbb{X} \triangleq \mathbb{R}^d$. Pour tout $x \in X$, le sous-problème $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x))$ est alors lisse, et supposé avoir une unique solution globale $\gamma(x)$. Par conséquent, la reformulation $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ consiste en la minimisation de $\Phi \triangleq (x \mapsto \widetilde{\varphi}(\gamma(x)) + \varepsilon(x))$.

L'instance de cDSM retenue pour résoudre $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ est à décroissance suffisante, avec la fonction de forçage $\rho \triangleq (\nu \mapsto (\frac{\nu}{8\delta^0})^3)$ choisie pour être peu restrictive. Le pas initial est $\delta^0 \triangleq 1$, et nous prenons $\Lambda \triangleq \{\frac{1}{2}\}$, et $\Upsilon \triangleq \{1\}$ lorsque $(\mathbf{P}^{\mathtt{ref}}_{\mathtt{PO}})$ dépend d'une seule variable ou $\Upsilon \triangleq \{2\}$ autrement. Nous ne faisons pas d'étape de `search`, et l'étape de `poll` est générée comme dans [1]. Pour simplifier l'implémentation, l'étape de `covering` tire une direction au hasard dans $\mathrm{cl}(\mathcal{B}_r)$ à chaque itération, où nous prenons arbitrairement $r \triangleq \frac{1}{10}$. L'algorithme est arrêté au premier $k \in \mathbb{N}$ tel que $\delta^k \leq 10^{-10}$.

Dans le chapitre 6.1.1, $\widetilde{\varphi}$ est une quadratique dépendant de deux variables et ε dépend de la première variable uniquement. Dans le chapitre 6.1.2, $\widetilde{\varphi}$ est une fonction des coordonnées polaires et ε est un bruit dépendant du rayon. Au chapitre 6.1.3, $\widetilde{\varphi}$ est une fonction de deux variables et ε dépend d'une fonction $f : \mathbb{R}^2 \to \mathbb{R}$ non-linéaire. Enfin, le chapitre 6.1.4 propose un cas où $\widetilde{\varphi}$ est une norme infinie de trois variables et ε dépend de $f : \mathbb{R}^3 \to \mathbb{R}^2$ non-linéaire. Ce problème illustre un cas plus proche de situations réelles, où pour tout $x \in X$, $\gamma(x)$ ne peut être obtenu qu'en résolvant algorithmiquement le sous-problème ($\mathbf{P}^{\mathtt{sub}}_{\mathtt{PO}}(x)$).

6.1.1 Bruit mono-variable

On considère ici la fonction

$$\varphi : \begin{cases} \mathbb{Y} = Y \triangleq \mathbb{R}^2 & \to & \mathbb{R} \\ y = (y_1, y_2) & \mapsto & \varepsilon(y_1) + (y_2 - \sigma(y_1))^2 , \end{cases}$$

où $\varepsilon(x) \triangleq |x| \left(1 + \sin(\frac{2\pi}{x})^2\right)^{\frac{1}{2}} + |\lfloor x \rceil|$ si $x \in \mathbb{R}^*$ et $\varepsilon(0) \triangleq 0$, et $\sigma(x) \triangleq 2 \lfloor x \rceil$, et où $\lfloor x \rceil \triangleq \lfloor x \rfloor$ si $x \in \mathbb{R}_-$ et $\lfloor x \rceil \triangleq \lceil x \rceil - 1$ si $x \in \mathbb{R}^*_+$. L'unique minimiseur global de φ est $y^* \triangleq (0, 0)$, avec $\varphi(y^*) = 0$. Ici, $\mathbb{Y}$ admet une partition en $\mathbb{Y}(x) \triangleq \{x\} \times \mathbb{R}$ pour tout $x \in \mathbb{X} = X \triangleq \mathbb{R}$. Pour tout $x \in \mathbb{X}$, $\varphi_{|\mathbb{Y}(x)}$ est une quadratique dépendant de y_2, tandis que $y_1 = x$ est fixé. Son unique minimiseur global est $\gamma(x) = (x, \sigma(x))$, avec $\varphi(\gamma(x)) = \varepsilon(y_1)$. La figure 6.1 représente φ, la partition de $\mathbb{Y}$ et l'image de γ, et la figure 6.2 représente ε dans le cadre de gauche. La fonction Φ de la reformulation est alors

$$\Phi : \begin{cases} \mathbb{X} & \to & \mathbb{R} \\ x & \mapsto & \varepsilon(x), \end{cases}$$

représentée dans le cadre de droite de la figure 6.2. Son unique minimiseur global est $x^* \triangleq 0$, avec $\Phi(x^*) = 0$, et Φ est discontinue à gauche en x^*.

La `PO` est applicable ici, mais φ n'est pas semi-continue en $(0, -1) \in \mathcal{V}_\gamma(x^*)$. Nous testons huit instances de `cDSM` pour minimiser Φ depuis huit points de départs différents. Les résultats sont donnés dans le tableau 6.1, qui donne le nombre d'itérations requises pour atteindre différents seuils de précision sur la solution. Toutes les instances renvoient $\widetilde{x}^* \in [0, 2E^{-10}]$, donc elles approchent x^* du côté évitant la discontinuité. On a donc $\widetilde{y}^* \triangleq \gamma(\widetilde{x}^*)$ vérifiant, dans les huit cas, $\|\widetilde{y}^* - y^*\| \leq 2E^{-10}$ et $|\varphi(\widetilde{y}^*) - \varphi(y^*)| \leq 2E^{-10}$. Cela valide empiriquement le théorème 5.4. On remarque qu'ici la conclusion du théorème portant sur l'optimalité de $\gamma(x^*)$ est valide, même si φ n'est pas semi-continue en tout point de $\mathcal{V}_\gamma(x^*)$.

TABLEAU 6.1 Huit exécutions de `cDSM` au chapitre 6.1.1. Ici, $\hat{x}^k \triangleq x^k$.

x^0	1er $\hat{x}^k \in [\pm 5E^{-03}]$	1er $\hat{x}^k \in [\pm 5E^{-06}]$	1er $\hat{x}^k \in [\pm 5E^{-09}]$	$\hat{x}^k$ final
$+9.753$	$\hat{x}^{23} = +3.37E^{-03}$	$\hat{x}^{37} = +8.86E^{-07}$	$\hat{x}^{55} = +3.15E^{-09}$	$\hat{x}^{64} = +7.41E^{-12}$
$+\pi$	$\hat{x}^{18} = +3.12E^{-03}$	$\hat{x}^{31} = +3.93E^{-06}$	$\hat{x}^{49} = +1.00E^{-09}$	$\hat{x}^{56} = +6.89E^{-11}$
$+\sqrt{2}$	$\hat{x}^{13} = +4.65E^{-03}$	$\hat{x}^{29} = +2.79E^{-06}$	$\hat{x}^{44} = +3.08E^{-09}$	$\hat{x}^{54} = +5.12E^{-11}$
$+e+1$	$\hat{x}^{17} = +2.08E^{-03}$	$\hat{x}^{24} = +3.01E^{-06}$	$\hat{x}^{45} = +3.29E^{-09}$	$\hat{x}^{53} = +3.45E^{-11}$
-9.753	$\hat{x}^{17} = +4.61E^{-04}$	$\hat{x}^{33} = +3.33E^{-06}$	$\hat{x}^{51} = +3.96E^{-09}$	$\hat{x}^{60} = +4.05E^{-12}$
$-\pi$	$\hat{x}^{20} = +2.94E^{-03}$	$\hat{x}^{33} = +3.46E^{-06}$	$\hat{x}^{43} = +2.33E^{-10}$	$\hat{x}^{54} = +1.16E^{-10}$
$-\sqrt{2}$	$\hat{x}^{06} = +3.45E^{-03}$	$\hat{x}^{27} = +1.13E^{-06}$	$\hat{x}^{43} = +1.51E^{-09}$	$\hat{x}^{52} = +7.67E^{-13}$
$-e-1$	$\hat{x}^{21} = +2.74E^{-03}$	$\hat{x}^{36} = +2.40E^{-06}$	$\hat{x}^{49} = +4.02E^{-09}$	$\hat{x}^{57} = +5.70E^{-11}$

FIGURE 6.1 Fonction φ (tronquée), partition de $\mathbb{Y}$ et image de γ au chapitre 6.1.1.

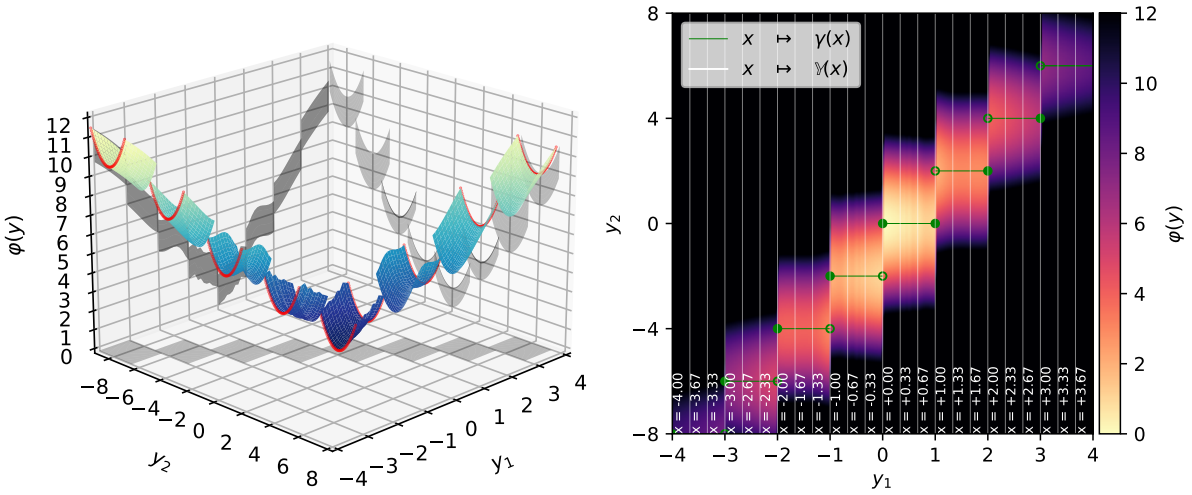
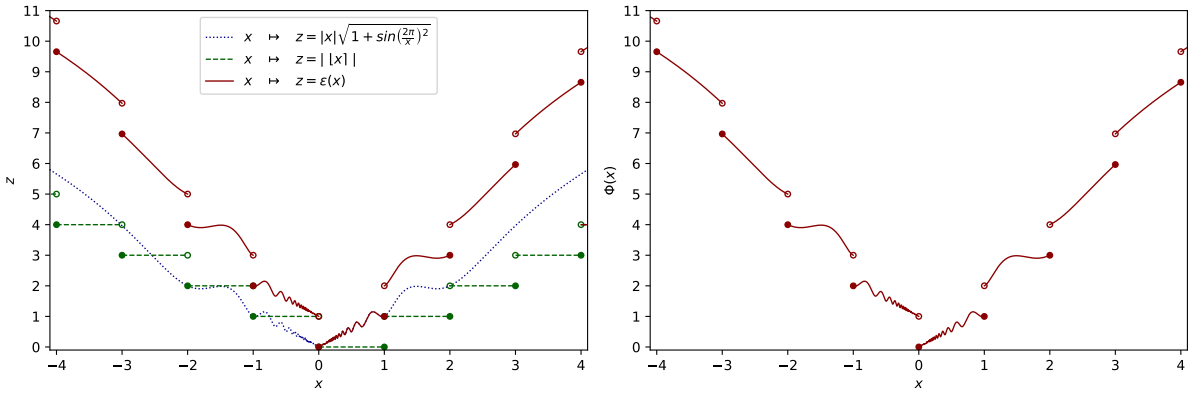


FIGURE 6.2 Fonctions ε et Φ au chapitre 6.1.1.



6.1.2 Bruit radial

On considère ici la fonction φ, donnée en coordonnées polaires par

$$\varphi : \begin{cases} \mathbb{Y} = Y \triangleq \mathbb{R}_+ \times [0, 2\pi[& \to \quad \mathbb{R} \\ \qquad y = (r, \theta) & \mapsto \begin{cases} \varepsilon(r) + \sqrt{r} \sin\left(\dfrac{\theta - \pi - 2\pi \log_2(r)}{2}\right)^2 & \text{si } r > 0, \\ \varepsilon(0) & \text{sinon,} \end{cases} \end{cases}$$

où $\varepsilon(x) \triangleq \sqrt{|x^2 - 2|} + \frac{\sin(10\pi(x-\sqrt{2}))^2}{10}$ pour tout $x \in \mathbb{R}_+$. L'unique minimiseur global de φ est $y^* = (r^*, \theta^*) \triangleq (\sqrt{2}, 0)$, avec $\varphi(y^*) = 0$. Ici, $\mathbb{Y}$ admet une partition en $\mathbb{Y}(x) \triangleq \{x\} \times [0, 2\pi[$ si $x \in \mathbb{R}_+$ et $\mathbb{Y}(x) \triangleq \emptyset$ si $x \in \mathbb{R}_-^*$, indexée par $\mathbb{X} \triangleq \mathbb{R}$ avec $X \triangleq \mathbb{R}_+$. Pour tout $x \in X$, $\varphi_{|\mathbb{Y}(x)}$ est une sinusoïde 2π-périodique suivant θ, tandis que $r = x$ est fixé. Son unique minimiseur global est $\gamma(x) = (x, [\pi + 2\pi \log_2(x)]_{2\pi})$, où $[z]_{2\pi}$ représente le résidu de $z \in \mathbb{R}$ modulo 2π, avec $\varphi(\gamma(x)) = \varepsilon(r)$. La figure 6.3 représente φ, la partition de $\mathbb{Y}$ et l'image de γ, et la figure 6.4 représente ε dans le cadre de gauche. La fonction Φ de la reformulation est alors

$$\Phi_{|X} : \begin{cases} X & \to \quad \mathbb{R} \\ x & \mapsto \quad \varepsilon(x), \end{cases}$$

représentée dans la figure 6.4. Son unique minimiseur global est $x^* \triangleq \sqrt{2}$, avec $\Phi(x^*) = 0$.

La `PO` est applicable ici. Nous testons huit instances de `cDSM` pour minimiser Φ depuis huit points de départ différents. Le tableau 6.2 donne le nombre d'itérations requises pour atteindre différents seuils de précision sur la solution. Toutes les instances approchent x^*, puisqu'elles retournent $\widetilde{x}^* \in [x^* \pm 6E^{-11}]$. Ainsi, elles fournissent toutes $\widetilde{y}^* \triangleq \gamma(\widetilde{x}^*)$ tel que $\|\widetilde{y}^* - y^*\| \leq 6E^{-11}$ et $|\varphi(\widetilde{y}^*) - \varphi(y^*)| \leq 3E^{-5}$. Cela confirme empiriquement la véracité du théorème 5.4. Ici, on est dans le cas où tous les pré-requis du théorèmes 5.4 sont vérifiées.

Tableau 6.2 Huit exécutions de `cDSM` au chapitre 6.1.2. Ici, $\hat{x}^k \triangleq x^k - \sqrt{2}$.

x^0	1er $\hat{x}^k \in [\pm 5E^{-03}]$	1er $\hat{x}^k \in [\pm 5E^{-06}]$	1er $\hat{x}^k \in [\pm 5E^{-09}]$	$\hat{x}^k$ final
0	$\hat{x}^{09} = +4.39E^{-03}$	$\hat{x}^{29} = -8.47E^{-07}$	$\hat{x}^{41} = +2.73E^{-09}$	$\hat{x}^{52} = +5.39E^{-11}$
2^{-5}	$\hat{x}^{09} = +5.08E^{-04}$	$\hat{x}^{27} = +4.42E^{-06}$	$\hat{x}^{44} = -3.57E^{-09}$	$\hat{x}^{54} = +3.83E^{-11}$
$3\sqrt{2}$	$\hat{x}^{17} = -1.14E^{-03}$	$\hat{x}^{35} = -1.51E^{-06}$	$\hat{x}^{51} = -1.95E^{-09}$	$\hat{x}^{60} = +2.94E^{-11}$
4π	$\hat{x}^{25} = +1.37E^{-03}$	$\hat{x}^{39} = +1.50E^{-06}$	$\hat{x}^{57} = +2.05E^{-09}$	$\hat{x}^{66} = -4.99E^{-11}$
5	$\hat{x}^{08} = +6.72E^{-05}$	$\hat{x}^{27} = -1.45E^{-06}$	$\hat{x}^{41} = -1.81E^{-09}$	$\hat{x}^{48} = +5.71E^{-11}$
e	$\hat{x}^{08} = -1.75E^{-03}$	$\hat{x}^{28} = +9.13E^{-07}$	$\hat{x}^{41} = +4.04E^{-09}$	$\hat{x}^{53} = -3.28E^{-11}$
e^2	$\hat{x}^{11} = -5.45E^{-04}$	$\hat{x}^{25} = +4.01E^{-06}$	$\hat{x}^{43} = +5.27E^{-10}$	$\hat{x}^{53} = -5.47E^{-11}$
e^3	$\hat{x}^{30} = +3.83E^{-03}$	$\hat{x}^{44} = +4.66E^{-06}$	$\hat{x}^{59} = -4.19E^{-09}$	$\hat{x}^{69} = +3.72E^{-12}$

FIGURE 6.3 Fonction φ, partition de $\mathbb{Y}$ et image de γ au chapitre 6.1.2.

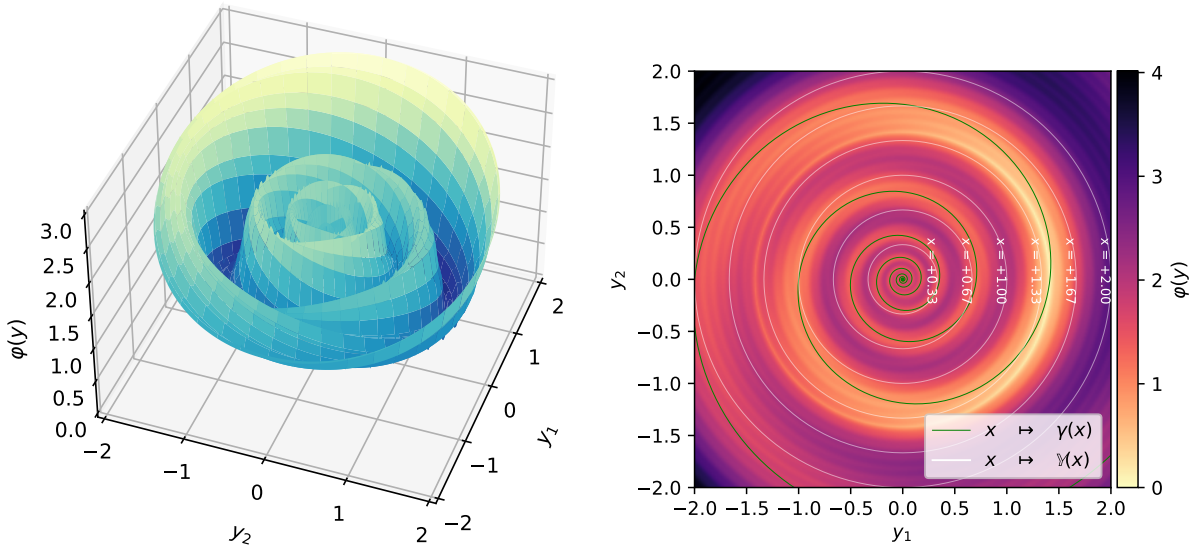
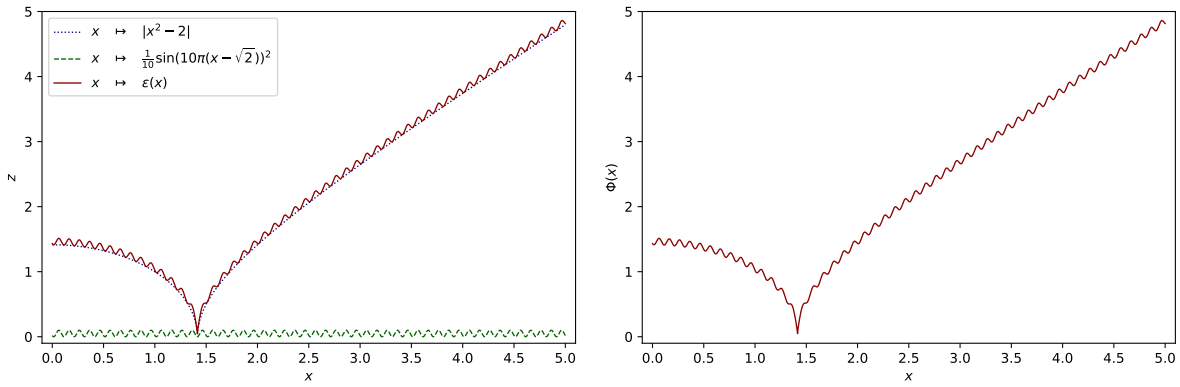


FIGURE 6.4 Fonctions ε et Φ au chapitre 6.1.2.



6.1.3 Bruit affecté par une combinaison non-linéaire de deux variables

On considère ici la fonction

$$\varphi : \begin{cases} \mathbb{Y} \triangleq \mathbb{R}^2 & \to \quad \mathbb{R} \\ y = (y_1, y_2) & \mapsto \begin{cases} \varepsilon(y_1 y_2) + \ln\left(1 + \left(\frac{y_1^2}{y_2^2+1} - 1\right)^2\right) & \text{si } y \in Y \triangleq \mathbb{R}_+ \times \mathbb{R}, \\ +\infty & \text{sinon,} \end{cases} \end{cases}$$

où $\varepsilon(x) \triangleq \exp(\frac{1}{x-4}) + \frac{\sqrt{|x-4|}}{5}$ si $x \neq 4$ et $\varepsilon(4) \triangleq +\infty$. Ici, φ n'a pas de minimum global, mais son infimum est 0. On a $\mathcal{V}_\varphi(y^*) = \{0, +\infty\}$ quand $y \to y^* \approx (2.12, 1.87)$. La valeur exacte de y^* et de certaines directions menant à $\varphi(y) \to 0$ sont données en fin de paragraphe. Ici, $\mathbb{Y}$

admet une partition en $\mathbb{Y}(x) \triangleq \{y = (y_1, y_2) \in \mathbb{Y} : y_1 y_2 = x\}$ pour tout $x \in \mathbb{X} = X \triangleq \mathbb{R}$. Pour tout $x \in \mathbb{X}$, $\varepsilon(y_1 y_2)$ est constant et on peut calculer l'unique minimiseur global de $\varphi_{|\mathbb{Y}(x)}$,[1]

$$\gamma(x) = \left(\sqrt{\frac{1+\sqrt{1+4x^2}}{2}}, x\sqrt{\frac{2}{1+\sqrt{1+4x^2}}}\right), \quad \text{avec} \quad \varphi(\gamma(x)) = \varepsilon(x).$$

La figure 6.5 représente φ, la partition de $\mathbb{Y}$ et l'image de γ, et la figure 6.6 représente ε dans le cadre de gauche. La fonction Φ de la reformulation est alors

$$\Phi : \begin{cases} \mathbb{X} & \to & \mathbb{R} \\ x & \mapsto & \varepsilon(x), \end{cases}$$

représentée dans le cadre de droite de la figure 6.6. Ici, Φ est discontinue en $x = 4$ et n'a pas de minimum global, mais son infimum est 0. De plus, en $x^* \triangleq 4$ on a $\mathcal{V}_\Phi(x^*) = \{0, +\infty\}$ avec $\lim_{x \nearrow x^*} \Phi(x) = 0$ malgré $\lim_{x \searrow x^*} \Phi(x) = +\infty$. On a de même $y^* \triangleq \gamma(x^*)$ et $\varphi(y) \searrow 0$ lorsque $y = \gamma(x)$ avec $x \nearrow x^*$, tandis que $\varphi(y) \nearrow +\infty$ lorsque $y = \gamma(x)$ avec $x \searrow x^*$.

La `PO` est applicable ici. Nous testons huit instances de `cDSM` pour minimiser Φ depuis huit points de départ différents. Le tableau 6.3 donne le nombre d'itérations requises pour atteindre différents seuils de précision sur la solution. Les instances partant de $x^0 < x^*$ renvoient toutes $\tilde{x}^* \in [x^* - 2E^{-10}, x^*]$, qui approche x^* du côté minimisant Φ. Ces instances mènent à $\tilde{y}^* \triangleq \gamma(\tilde{x}^*) \approx y^*$, ce qui confirme empiriquement la véracité du théorème 5.4. L'instance partant de $x^0 \triangleq 3\sqrt{2} \approx 4.24 > x^*$ se comporte de la même manière. Les trois dernières instances, qui commencent avec $x^0 \gg 4$, convergent à la place vers $\tilde{x}^* \approx 9.27$, mais ceci reste cohérent avec le théorème 5.4 puisque ce point est bien un minimiseur local de Φ.

Tableau 6.3 Huit exécutions de `cDSM` au chapitre 6.1.3. Ici, $\hat{x}^k \triangleq x^k - 4$.

x^0	1er $\hat{x}^k \in [\pm 5E^{-03}]$	1er $\hat{x}^k \in [\pm 5E^{-06}]$	1er $\hat{x}^k \in [\pm 5E^{-09}]$	$\hat{x}^k$ final
$-e^2$	$\hat{x}^{107} = -4.17E^{-03}$	$\hat{x}^{120} = -2.20E^{-06}$	$\hat{x}^{138} = -1.68E^{-09}$	$\hat{x}^{146} = -4.65E^{-11}$
$-\pi$	$\hat{x}^{026} = -1.76E^{-03}$	$\hat{x}^{043} = -1.70E^{-06}$	$\hat{x}^{057} = -3.88E^{-09}$	$\hat{x}^{065} = -4.14E^{-11}$
$-\sqrt{2}$	$\hat{x}^{024} = -4.25E^{-04}$	$\hat{x}^{040} = -4.95E^{-06}$	$\hat{x}^{051} = -1.40E^{-09}$	$\hat{x}^{062} = -8.01E^{-12}$
$+e$	$\hat{x}^{009} = -4.68E^{-04}$	$\hat{x}^{026} = -2.78E^{-06}$	$\hat{x}^{043} = -3.13E^{-09}$	$\hat{x}^{051} = -1.06E^{-10}$
$+3\sqrt{2}$	$\hat{x}^{017} = -2.66E^{-03}$	$\hat{x}^{030} = -1.33E^{-07}$	$\hat{x}^{046} = -2.41E^{-09}$	$\hat{x}^{053} = -8.40E^{-11}$
$+2e^2$	/	/	/	$\hat{x}^{201} = +5.27E^{+00}$
$+4\pi$	/	/	/	$\hat{x}^{201} = +5.27E^{+00}$
$+e^3$	/	/	/	$\hat{x}^{201} = +5.27E^{+00}$

1. Partir de $y_1^2/(y_2^2+1) = 1$ pour obtenir $y_2^2 = y_1^2 - 1$ (et donc $y_1^2 \geq 1$ donc $y_1 \neq 0$), avec $(y_1, y_2) \in Y(x)$ pour trouver $y_2 = x/y_1$ et $y_1 \geq 0$. Égaliser les deux expression de y_2^2 mène à $(y_1^2)^2 - y_1^2 - x^2 = 0$, avec une seule solution $y_1^2 = \frac{1}{2}(1+\sqrt{1+4x^2})$ puisque $y_1^2 \geq 1$. De là, y_1 suit (car il doit être positif), et y_2 en découle.

FIGURE 6.5 Fonction φ, partition de $\mathbb{Y}$ et image de γ au chapitre 6.1.3.

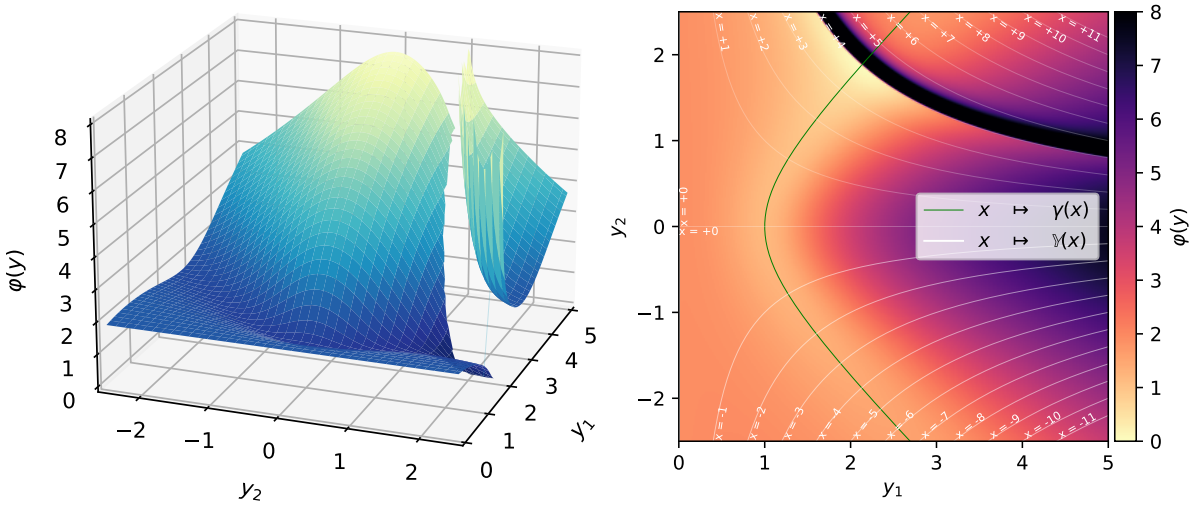
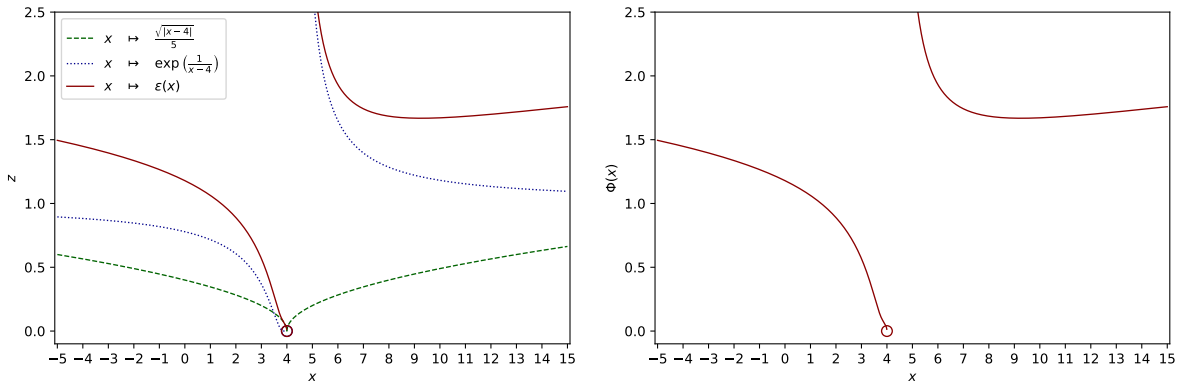


FIGURE 6.6 Fonctions ε et Φ au chapitre 6.1.3.



6.1.4 Bruit de dimension 2 dans un problème de dimension 3

On considère ici la fonction

$$\varphi : \begin{cases} \mathbb{Y} = Y \triangleq \mathbb{R}^3 & \to \quad \mathbb{R} \\ y = (y_1, y_2, y_3) & \mapsto \quad \varepsilon\left(y_2 - y_1^3, y_1 - y_3^3\right) + \|y\|_\infty , \end{cases}$$

où $\varepsilon(x_1, x_2) \triangleq \left(\frac{\sin(10\pi(x_2 - x_1^3))}{5} + \frac{\sin(6\pi(x_2 - e^{-x_1} + 1))}{7} + \frac{\sin(12\pi(x_1^2 + x_2^2)^{\frac{1}{2}})}{11}\right)^2$ pour tout $x \in \mathbb{R}^2$. L'unique minimum global de φ est $y^* \triangleq (0, 0, 0)$, avec $\varphi(y^*) = 0$. Ici, $\mathbb{Y}$ admet une partition suivant $\mathbb{Y}(x) \triangleq \{(y_1, y_2, y_3) \in \mathbb{Y} : y_2 - y_1^3 = x_1 \text{ et } y_1 - y_3^3 = x_2\} = \{(t, t^3 + x_1, \sqrt[3]{t - x_2}) : t \in \mathbb{R}\}$ pour tout $x \in \mathbb{X} = X \triangleq \mathbb{R}^2$, et ainsi $\varphi_{|\mathbb{Y}(x)} = \varepsilon(x) + \max\{|t_x(y)|, |t_x(y)^3 + x_1|, |\sqrt[3]{t_x(y) - x_2}|\}$, où $t_x(y)$ désigne l'unique $t \in \mathbb{R}$ tel que $s_x(t) \triangleq (t, t^3 + x_1, \sqrt[3]{t - x_2}) = y$, pour tout $y \in \mathbb{Y}(x)$.

La minimisation de $\varphi_{|\mathbb{Y}(x)}$ reste difficile, mais ce problème admet la reformulation exacte

$$\begin{array}{ll} \underset{M \geq 0,\ T \in \mathbb{R}}{\text{minimiser}} & \varepsilon(x) + M \\ \text{sous contraintes} & T \in [-M, M]\,, \\ & T \in \left[\sqrt[3]{-M - x_1}, \sqrt[3]{M - x_1}\right], \\ & T \in [-M^3 + x_2, M^3 + x_2]\,, \end{array}$$

dont l'unique solution, notée $(M(x), T(x))$, est simple à obtenir. En effet, $M(x)$ est le plus petit $M \geq 0$ tel que $[-M, M] \cap [\sqrt[3]{-M - x_1}, \sqrt[3]{M - x_1}] \cap [-M^3 + x_2, M^3 + x_2] \neq \emptyset$, et en $M = M(x)$ cette intersection est réduite au singleton $\{T(x)\}$. On peut trouver $M(x)$ par dichotomie,[2] puis en déduire $T(x)$, $\gamma(x) = s_x(T(x))$ et $\varphi(\gamma(x)) = \varepsilon(x) + M(x)$. La figure 6.7 représente φ, la figure 6.8 illustre la partition de $\mathbb{Y}$ et l'image de γ, et la figure 6.9 représente ε et M dans les deux cadres de gauche. La fonction Φ de la reformulation est ici

$$\Phi : \begin{cases} \mathbb{X} & \to & \mathbb{R} \\ x & \mapsto & \varepsilon(x) + M(x), \end{cases}$$

représentée à la figure 6.9. Son unique minimiseur global est $x^* \triangleq (0, 0)$, avec $\Phi(x^*) = 0$.

La `PO` est applicable ici. Nous testons huit instances de `cDSM` pour minimiser Φ depuis huit points de départ différents. Le tableau 6.4 donne le nombre d'itérations requises pour atteindre différents seuils de précision sur la solution. On a $\|\widetilde{x}^* - x^*\| \leq 2E^{-8}$ dans les huit cas. Ainsi, $\widetilde{y}^* \triangleq \gamma(\widetilde{x}^*)$ est tel que $\|\widetilde{y}^* - y^*\| \leq 2E^{-8}$ et $|\varphi(\widetilde{y}^*) - \varphi(y^*)| \leq 2E^{-8}$. Cela confirme empiriquement la véracité du théorème 5.4.

Tableau 6.4 Huit exécutions de `cDSM` au chapitre 6.1.4. Ici, $\hat{x}^k \triangleq \|x^k\|_\infty$.

x^0	1er $\hat{x}^k \leq 5E^{-02}$	1er $\hat{x}^k \leq 5E^{-05}$	1er $\hat{x}^k \leq 5E^{-08}$	$\hat{x}^k$ final
$[-2, 2]$	$\hat{x}^{032} = 3.52E^{-02}$	$\hat{x}^{110} = 4.03E^{-05}$	$\hat{x}^{194} = 4.79E^{-08}$	$\hat{x}^{220} = 1.10E^{-08}$
$[\frac{-1}{100}, e^2]$	$\hat{x}^{037} = 2.02E^{-02}$	$\hat{x}^{100} = 3.99E^{-05}$	$\hat{x}^{176} = 1.10E^{-08}$	$\hat{x}^{198} = 9.03E^{-10}$
$[\frac{-\pi}{2}, \frac{7}{4}]$	$\hat{x}^{027} = 6.34E^{-03}$	$\hat{x}^{095} = 4.60E^{-05}$	$\hat{x}^{177} = 1.23E^{-08}$	$\hat{x}^{195} = 5.51E^{-09}$
$[\frac{-\pi}{4}, e^{\frac{1}{2}}]$	$\hat{x}^{040} = 3.66E^{-02}$	$\hat{x}^{126} = 1.46E^{-05}$	$\hat{x}^{235} = 3.45E^{-08}$	$\hat{x}^{253} = 4.54E^{-09}$
$[\frac{1}{4}, \frac{1}{4}]$	$\hat{x}^{022} = 2.85E^{-02}$	$\hat{x}^{119} = 2.90E^{-05}$	$\hat{x}^{193} = 1.84E^{-08}$	$\hat{x}^{212} = 2.88E^{-09}$
$[\frac{3\pi}{2}, \frac{1}{\sqrt{8}}]$	$\hat{x}^{039} = 3.29E^{-02}$	$\hat{x}^{124} = 1.96E^{-05}$	$\hat{x}^{186} = 4.86E^{-08}$	$\hat{x}^{226} = 5.22E^{-09}$
$[e^2, 2\pi]$	$\hat{x}^{036} = 3.43E^{-02}$	$\hat{x}^{109} = 3.00E^{-05}$	$\hat{x}^{171} = 4.40E^{-08}$	$\hat{x}^{211} = 8.19E^{-09}$
$[e^2, \frac{-1}{11}]$	$\hat{x}^{029} = 1.48E^{-02}$	$\hat{x}^{093} = 4.77E^{-05}$	$\hat{x}^{148} = 2.48E^{-08}$	$\hat{x}^{172} = 1.48E^{-09}$

2. La dichotomie part de $\underline{M} = 0$ et $\overline{M} > M(x)$ et itère tant que $\overline{M} - \underline{M} \geq 10^{-10}$. Alors $M(x) \approx \frac{\underline{M} + \overline{M}}{2}$.

FIGURE 6.7 Coupes de la fonction φ au chapitre 6.1.4.

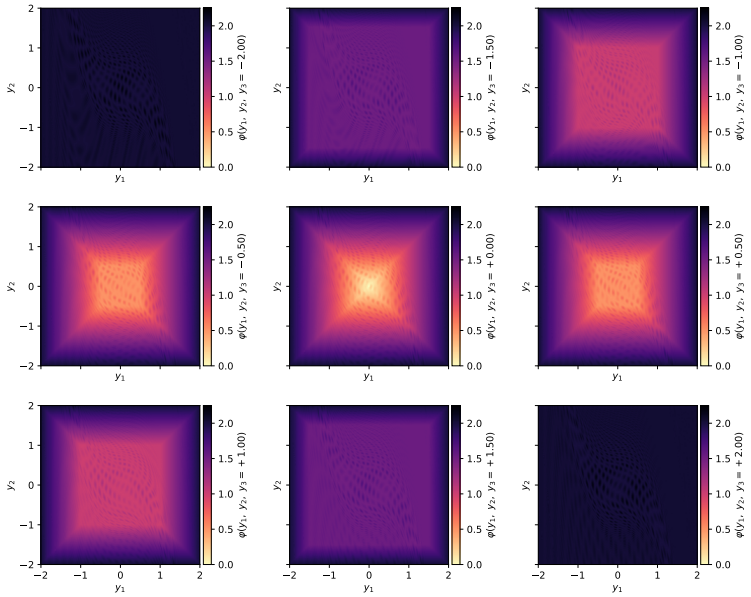


FIGURE 6.8 Partition de $\mathbb{Y}$ et image de γ au chapitre 6.1.4.

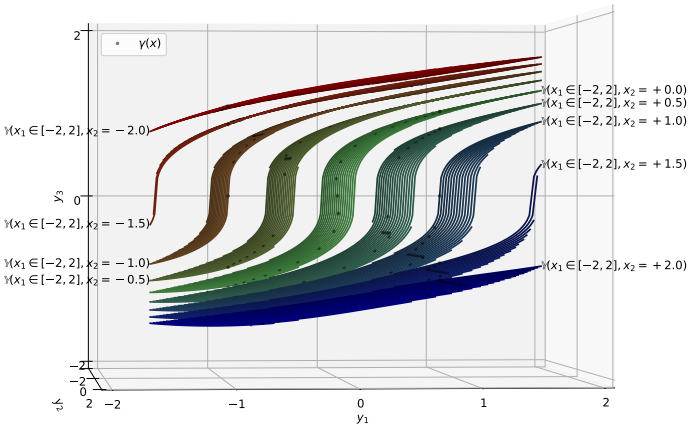
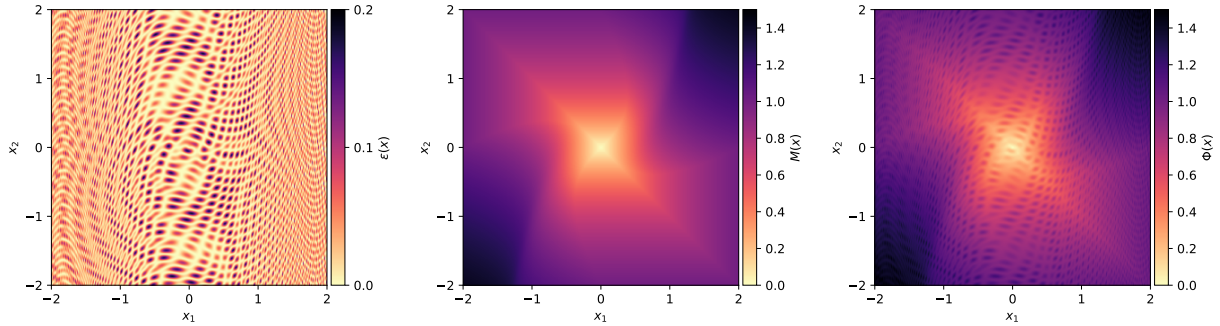


FIGURE 6.9 Fonctions ε, M et Φ au chapitre 6.1.4.



6.2 Étude numérique du gain apporté à DSM par l'optimisation partitionnée

Ce chapitre démontre par l'exemple que lorsque le cadre de la PO est applicable, il améliore la performance des algorithmes sans dérivées. Nous altérons les problèmes du chapitre 6.1 pour qu'ils aient environ 100 variables. Pour chacun, nous résolvons le problème initial avec deux solveurs de DFO (**NOMAD** [19] et **PRIMA** [123]) et le problème reformulé par le cadre de la PO avec une implémentation simple de DSM. On s'attend à ce que ni **NOMAD** et **PRIMA** ne soit performant ici, car les documentations associées déconseillent une utilisation pour optimiser plus de 50 variables. Dans ce qui suit, il serait inutile de comparer l'un à l'autre.

Les exemples étant analytiques, les calculs sont tous quasi-instantanés. Ceci n'est pas représentatif de la réalité. Pour ne pas biaiser l'étude, nous procédons comme suit. Pour chaque problème, nous considérons que le coût de calcul de $\varphi(y)$ est une unité arbitraire, pour tout $y \in \mathbb{Y}$, et que le coût de calcul de $\Phi(x)$ est de $\tau > 1$ unités, pour tout $x \in X$. Ici, τ représente le coût relatif du calcul de γ rapporté à celui de φ, et le coût de calcul pour $\Phi(x)$ modélise le sur-coût requis pour d'abord calculer $\gamma(x)$ et ensuite évaluer $\varphi(\gamma(x))$.

Nous utilisons **NOMAD** avec ses paramètres par défaut, et **PRIMA** avec l'algorithme BObyQA (*Bound Optimzation by Quadratic Approximations*) pour gérer les contraintes de boîte. En présence d'autres contraintes, les deux solveurs suivent la stratégie de *barrière extrême* [14]. Les solveurs sont arrêtés dès que le pas de sonde δ^k tombe sous la valeur 10^{-8}.

Pour notre DSM, nous reprenons un choix proche de celui fait au chapitre 6.1. C'est-à-dire, une cDSM à décroissance suffisante de fonction de forçage $\rho \triangleq (\nu \mapsto (\frac{\nu}{8\delta^0})^3)$, avec $\delta^0 \triangleq 1$ et $r \triangleq \frac{1}{10}$, et avec $\Lambda \triangleq \{\frac{1}{2}\}$ et $\Upsilon \triangleq \{2\}$. L'étape de `search` est négligée, l'étape de `poll` est générée comme dans [1] et l'étape de `covering` tire une direction au hasard à chaque itération. L'algorithme est arrêté au premier $k \in \mathbb{N}$ tel que $\delta^k \leq 10^{-10}$.

Nous choisissons six points de départ $(y_\ell^0)_{\ell \in [\![1,6]\!]}$ pour **NOMAD** et **PRIMA**, pris aléatoirement dans $\mathbb{Y}$, et à chacun nous associons le point de départ $\chi(y_\ell^0)$ fourni à une instance simple de DSM. Pour chaque algorithme, chaque problème et chaque point de départ, nous traçons le graphe représentant la meilleure valeur objectif trouvée en fonction du budget de calcul utilisé. Il ressort que si τ n'est pas excessif (c'est-à-dire, si le calcul de γ n'est pas trop coûteux comparé à celui de φ), alors DSM résolvant le problème reformulé domine **NOMAD** et **PRIMA** résolvant le problème initial. La convergence asymptotique de DSM sur le problème reformulé est observable (même en choisissant une instance peu évoluée), alors que même sous un budget d'évaluation déraisonnable en pratique, **NOMAD** et **PRIMA** (deux solveurs performants) échouent à initier leur convergence vers une solution satisfaisante. L'apport de la PO sur la vitesse et la qualité de résolution est donc sensible numériquement.

6.2.1 Bruit mono-variable en dimension 101

On considère ici la fonction

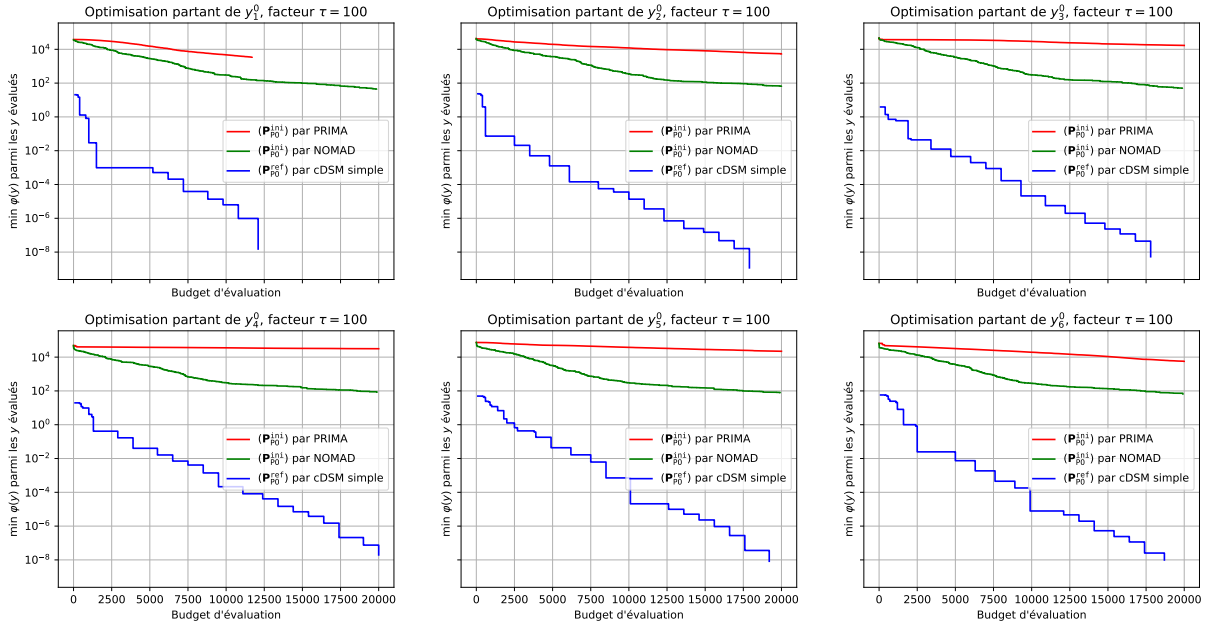
$$\varphi : \begin{cases} \mathbb{Y} = Y \triangleq \mathbb{R}^{101} & \to \quad \mathbb{R} \\ \quad y = (y_i)_{i=0}^{100} & \mapsto \quad \varepsilon(y_0) + \sum_{i=0}^{100} \left(y_i - \sigma_i(y_0)\right)^2 , \end{cases}$$

où $\varepsilon(x) \triangleq |x| \left(1 + \sin(\frac{2\pi}{x})^2\right)^{\frac{1}{2}} + |\lfloor x \rceil|$ si $x \in \mathbb{R}^*$ et $\varepsilon(0) \triangleq 0$, et où

$$\sigma : \begin{cases} \mathbb{R} & \to \quad \mathbb{R}^{101} \\ x & \mapsto \quad \left[x \ , \ \left[2\left(1 + \frac{i-1}{5}\right)\left\lfloor \frac{x}{i} \right\rceil\right]_{i=1}^{25} \ , \ \left[25 \cos\left(\frac{i-25}{5}\pi x\right)\right]_{i=26}^{50} \ , \ \left[x - \frac{10}{i}\right]_{i=51}^{75} \ , \ \left[\frac{i}{10}\right]_{i=76}^{100} \right] \end{cases}$$

L'unique minimiseur global de φ est $y^* \triangleq \sigma(0)$, avec $\varphi(y^*) = 0$. Pour tout $x \in \mathbb{X} \triangleq \mathbb{R}$, on pose $\mathbb{Y}(x) \triangleq \{x\} \times \mathbb{R}^{100}$, car alors $y_0 = x$ est fixé et $\varphi_{|\mathbb{Y}(x)}$ est une quadratique dépendant de $(y_i)_{i=1}^{100}$. Son unique minimiseur global est $\gamma(x) \triangleq \sigma(x)$, avec $\varphi(\gamma(x)) = \varepsilon(x)$. Ainsi, $\Phi(x) \triangleq \varepsilon(x)$ pour tout $x \in \mathbb{R}$ est inchangé par rapport au chapitre 6.1.1. La comparaison entre les stratégies est donnée à la figure 6.10. Les six points de départ de **NOMAD** et **PRIMA**, $(y_\ell^0))_{\ell \in [\![1,6]\!]}$, sont tirés aléatoirement dans $[\![-25, 25]\!]^{101}$, et les points de départ de `cDSM` sont alors $(\chi(y_\ell^0))_{\ell \in [\![1,6]\!]}$. Même en considérant $\tau = 100$, le gain de performance apporté par la `PO` est sensible.

FIGURE 6.10 Gains apportés par l'optimisation partitionnée au chapitre 6.2.1.



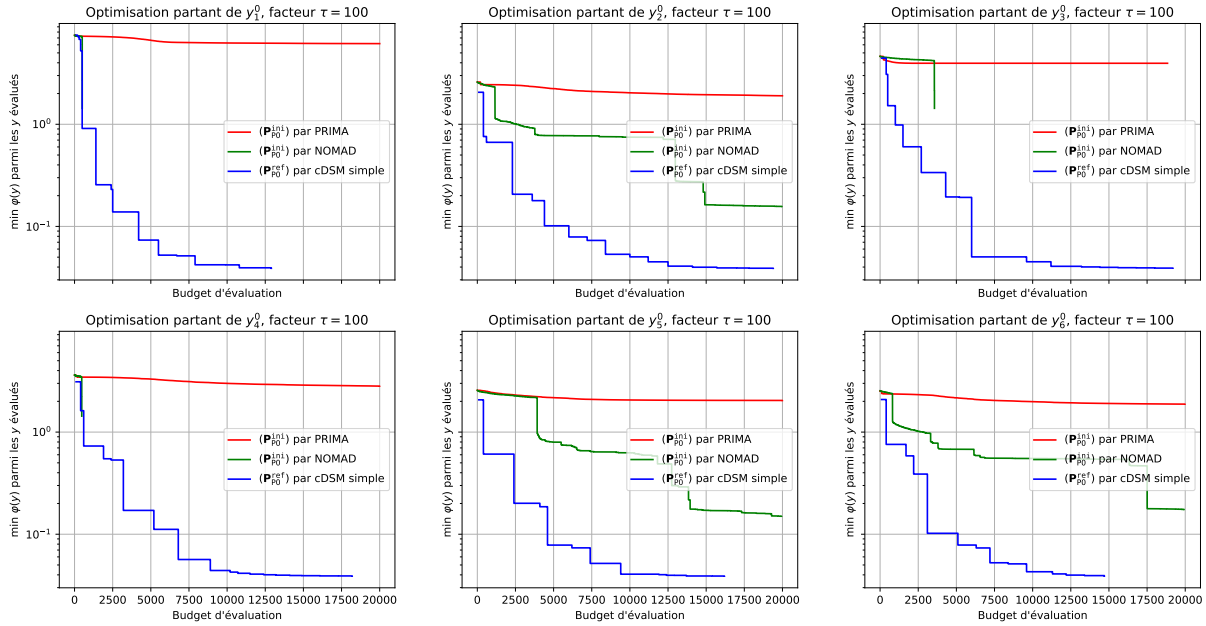
6.2.2 Bruit radial en dimension 101

On considère ici la fonction, donnée en coordonnées polaires,

$$\varphi : \begin{cases} \mathbb{Y} = Y \triangleq \mathbb{R}_+ \times [0, 2\pi[^{100} & \to \quad \mathbb{R} \\ \qquad y = (r, \theta) & \mapsto \quad \begin{cases} \varepsilon(r) + \dfrac{\sqrt{r}}{100} \sum\limits_{i=1}^{100} \sin\left(\dfrac{\theta_i - 2\pi \log_{i+1}(r)}{2}\right)^2 & \text{si } r > 0, \\ \varepsilon(0) & \text{sinon,} \end{cases} \end{cases}$$

où $\varepsilon(x)$ est définie comme au chapitre 6.1.2. L'unique minimiseur global de φ est $y^* \triangleq (r^*, \theta^*)$, où $r^* \triangleq \sqrt{2}$ et $\theta^* \triangleq (2\pi \log_{i+1}(\sqrt{2}))_{i=1}^{100}$, avec $\varphi(y^*) = 0$. Pour tout $x \in \mathbb{X} \triangleq \mathbb{R}$, on pose $\mathbb{Y}(x) \triangleq \{x\} \times [0, 2\pi[^{100}$ si $x \in \mathbb{R}_+$ et $\mathbb{Y}(x) \triangleq \emptyset$ si $x \in \mathbb{R}_-^*$. Alors, pour tout $x \in X \triangleq \mathbb{R}_+$, $\varphi_{|\mathbb{Y}(x)}$ est une somme de sinusoïdes 2π-périodiques indépendantes tandis que $r = x$ est fixé. Son unique minimiseur global est $\gamma(x) \triangleq (x, ([2\pi \log_{i+1}(x)]_{2\pi})_{i=1}^{100})$, avec $\varphi(\gamma(x)) = \varepsilon(x)$. Ainsi, $\Phi(x) \triangleq \varepsilon(x)$ pour tout $x \in X$ est inchangée par rapport au chapitre 6.1.2. La comparaison entre les stratégies est donnée à la figure 6.11. Les six points de départ de **NOMAD** et **PRIMA**, $(y_\ell^0))_{\ell \in [\![1,6]\!]}$, sont tirés aléatoirement dans $[0, 5] \times [0, 2\pi[^{100}$, et les points de départ de `cDSM` sont alors $(\chi(y_\ell^0))_{\ell \in [\![1,6]\!]}$. Même en considérant $\tau = 100$, le gain de performance apporté par la `PO` est sensible.

FIGURE 6.11 Gains apportés par l'optimisation partitionnée au chapitre 6.2.2.



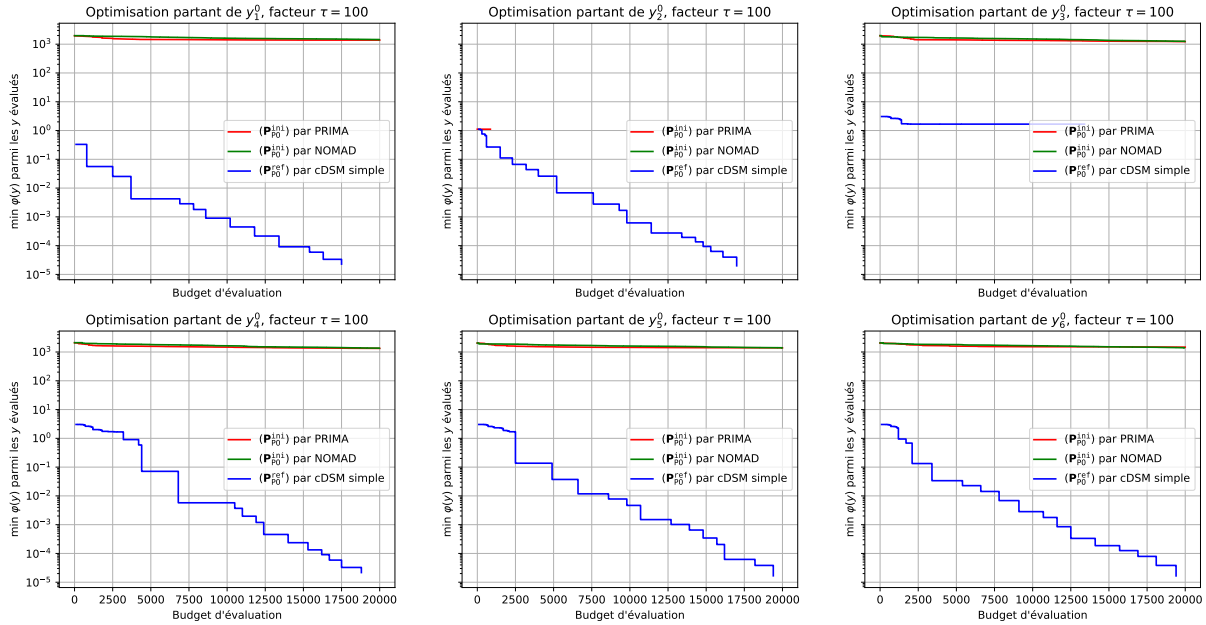
6.2.3 Bruit affecté par une combinaison non-linéaire de 101 variables

On considère ici la fonction

$$\varphi : \begin{cases} \mathbb{Y} \triangleq \mathbb{R}^{100} & \to \quad \mathbb{R} \\ y = (y_i)_{i=1}^{100} & \mapsto \quad \varepsilon\left(\sum_{\ell=0}^{19} \pi_\ell(y)\right) + \sum_{(\ell,p)\in[\![0,19]\!]^2} \ln\left(1 + \left(\frac{\pi_\ell(y)}{\pi_p(y)} - 1\right)^2\right) \end{cases}$$

où $\varepsilon(x)$ est définie comme au chapitre 6.1.3, où $\pi_\ell(y) \triangleq \prod_{i=5\ell+1}^{5(\ell+1)} y_i$ pour tout $\ell \in [\![0, 19]\!]$, et on pose $Y \triangleq \{y \in (\mathbb{R}_+^*)^{100} : \forall i \in [\![1, 100]\!],\ y_{i+1} \geq y_i\}$. Ici, φ n'a pas de minimiseur global, mais son infimum est 0. Approcher $y^* \triangleq \mathbb{1}$ mène à $\varphi(y) \to 0$ par certaines directions, et à $\varphi(y) \to +\infty$ par d'autres. Pour tout $x \in \mathbb{X} = X \triangleq \mathbb{R}$, on pose $\mathbb{Y}(x) \triangleq \{y \in \mathbb{Y} : \Sigma_{\ell=0}^{19}\pi_\ell(y) = x\}$, car alors $\varphi_{|\mathbb{Y}(x)}$ est une fonction lisse puisque $\varepsilon(\Sigma_{\ell=0}^{19}\pi_\ell(y))$ est constant. Son unique minimiseur global vaut $\gamma(x) \triangleq \sqrt[5]{\frac{x}{20}}\mathbb{1}$.[3] Ainsi, $\Phi_{|X}(x) \triangleq \varepsilon(x)$ pour tout $x \in X$ est inchangée par rapport au chapitre 6.1.3. La comparaison entre les stratégies est donnée à la figure 6.12. Les six points de départ de **NOMAD** et **PRIMA**, $(y_\ell^0))_{\ell\in[\![1,6]\!]}$, sont choisis pour soumettre plus ou moins de difficultés aux algorithmes.[4] Même en considérant $\tau = 100$, le gain de performance apporté par la `PO` est sensible.

FIGURE 6.12 Gains apportés par l'optimisation partitionnée au chapitre 6.2.3.



3. À $x \in \mathbb{X}$ fixé, $\varphi_{|Y(x)} \geq 0$ et vaut 0 lorsque les $(\pi_\ell)_{\ell\in[\![0,19]\!]}$ sont tous égaux. Sous la contrainte $\gamma(x) \in Y$, ceci implique $\gamma(x) = \alpha\mathbb{1}$ pour un certain $\alpha \in \mathbb{R}$. De là, $\gamma(x) \in Y(x)$ mène à $20\alpha^5 = x$. D'où $\gamma(x) = \sqrt[5]{\frac{x}{20}}\mathbb{1}$.

4. Pour **NOMAD** et **PRIMA**, on prend $y_1^0 \triangleq (i/100)_{i=1}^{100}$ et $y_2^0 \triangleq 0.5\mathbb{1}$ et $y_3^0 \triangleq (\frac{2i}{100})_{i=1}^{100}$, et les trois points restants y_i^0, $i \in [\![4, 6]\!]$, au hasard dans $\Pi_{j=1}^{100}[\frac{j-1}{100}, \frac{j}{100}]$. Pour `cDSM`, on prend alors les points $(\chi(y_\ell^0))_{\ell\in[\![1,6]\!]}$.

6.2.4 Bruit de dimension 10 dans un problème de dimension 100

On considère ici la fonction

$$\varphi : \begin{cases} \mathbb{Y} = Y \triangleq \mathbb{R}^{100} & \to \quad \mathbb{R} \\ y = (y_i)_{i=1}^{100} & \mapsto \quad \varepsilon\left(g_1(y_{10}) - \sum\limits_{i=1}^{9} y_i,\ \ldots,\ g_{10}(y_{100}) - \sum\limits_{i=91}^{99} y_i\right) + \|y\|_1 , \end{cases}$$

avec $g_j(z) \triangleq (z + (1 + \frac{j}{10})^z - 1)$ pour tout $z \in \mathbb{R}$ et tout $j \in [\![1, 10]\!]$, et

$$\varepsilon : \begin{cases} \mathbb{X} \triangleq \mathbb{R}^{10} & \to \quad \mathbb{R} \\ x = (x_j)_{j=1}^{10} & \mapsto \quad \left(\frac{\sin 5\pi\left(x_2 - x_1^3\right)}{5} + \frac{\sin 6\pi\left(x_4 - e^{-x_2 - x_3} + 1\right)}{7} + \frac{\sin 7\pi\sqrt{x_5^2 + x_6^2 + x_7^2}}{11} + \frac{\sin 8\pi x_8 x_9 x_{10}}{13}\right)^2 . \end{cases}$$

L'unique minimiseur global de φ est $y^* \triangleq 0\mathbb{1}$, avec $\varphi(y^*) = 0$. Pour tout $x \in \mathbb{X} = X \triangleq \mathbb{R}^{10}$, on pose $\mathbb{Y}(x) \triangleq \{y \in \mathbb{Y} : g_j(y_{10j}) - \Sigma_{i=10(j-1)+1}^{10(j-1)+9} y_i = x_j,\ \forall j \in [\![1, 10]\!]\}$. Ainsi, pour tout $x \in \mathbb{X}$, l'unique minimiseur global de $\varphi_{|\mathbb{Y}(x)}$ est la solution du problème

$$\underset{y\ \in\ \mathbb{Y}(x)}{\text{minimiser}} \quad \varepsilon(x) + \sum_{i \in [\![1,100]\!] \setminus 10\mathbb{N}} |y_i| + \sum_{j=1}^{10} \left| g_j^{-1}\left(x_j + \sum_{i=10(j-1)+1}^{10(j-1)+9} y_i\right)\right| ,$$

qui a pour unique solution [5] le vecteur $\gamma(x) \in \mathbb{R}^{100}$ contenant des 0 en chaque composante $i \notin 10\mathbb{N}$ et $g_j^{-1}(x_j)$ en chaque composante $i = 10j$, $j \in [\![1, 10]\!]$. On a alors Φ défini comme

$$\Phi : \begin{cases} \mathbb{X} = X \triangleq \mathbb{R}^{10} & \to \quad \mathbb{R} \\ x = (x_j)_{j=1}^{10} & \mapsto \quad \varepsilon(x) + \sum\limits_{j=1}^{10} \left| g_j^{-1}(x_j)\right| . \end{cases}$$

Son unique minimiseur global est $x^* \triangleq 0\mathbb{1}$, avec $\Phi(x^*) = 0$. Pour tout $x \in \mathbb{R}^{10}$, nous estimons les valeurs $(g_j^{-1}(x_j))_{j=1}^{10} \approx z \in \mathbb{R}^{10}$ comme une solution approximative de $(g_j(z_j))_{j=1}^{10} = x$.

La comparaison entre les stratégies est donnée à la figure 6.13. Les six points de départ de **NOMAD** et **PRIMA**, $(y_\ell^0))_{\ell \in [\![1,6]\!]}$, sont tirés aléatoirement dans $[\![-10, 10]\!]^{100}$, et les points de départ de `cDSM` sont alors $(\chi(y_\ell^0))_{\ell \in [\![1,6]\!]}$. En considérant $\tau = 10$, le gain de performance apporté par la `PO` est sensible en fin d'optimisation. La valeur $\tau = 10$ est assez faible, mais aurait pu être augmentée avec une meilleure implémentation de `cDSM`. Un test avec **NOMAD** résolvant le problème reformulé a donné des résultats plus satisfaisants (voir figure 6.14). On retrouve des résultats similaires, mais cette fois avec un facteur $\tau = 30$.

5. On a $\frac{d}{dz} g_j(z) = (1 + (1 + \frac{j}{10})^z \ln(1 + \frac{j}{10})) > 1$, et donc $\frac{d}{dz} g_j^{-1}(z) = \frac{1}{\frac{d}{dz} g_j\left(g_j^{-1}(z)\right)} \in]0, 1[$, pour tout $(j, z) \in \mathbb{N}^* \times \mathbb{R}$. Ainsi, pour tout $(j, z) \in \mathbb{N}^* \times \mathbb{R}$ et tout $v \in \mathbb{R}$, on a $|g_j^{-1}(z + v)| + |v| \geq |g_j^{-1}(z)|$. Le résultat suit.

FIGURE 6.13 Gains apportés par l'optimisation partitionnée au chapitre 6.2.4.

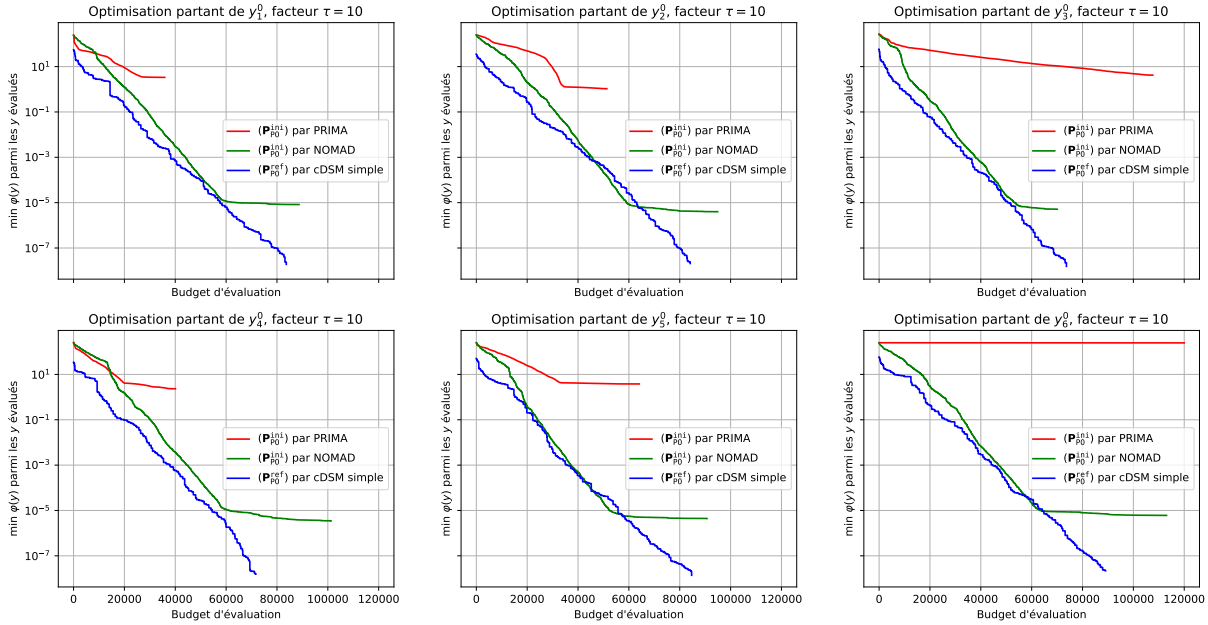
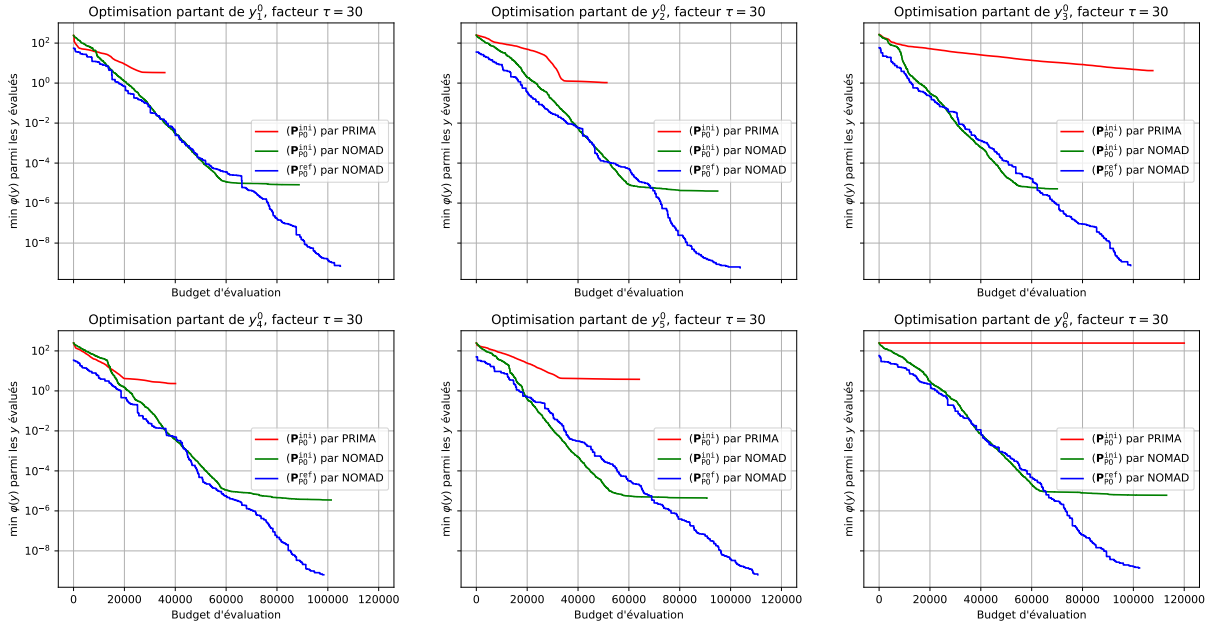


FIGURE 6.14 Surplus de gain apporté par un solveur performant au chapitre 6.2.4.



CHAPITRE 7 APPLICATION AU CONTRÔLE OPTIMAL

Si tu freines, t'es un lâche. Mais si tu freines pas, t'es un con.

Règle fondamentale des sports mécaniques.

Le contrôle optimal est un domaine à priori aux antipodes de l'optimisation sans dérivées. Un problème de contrôle optimal est extrêmement structuré, de dimension infinie, et souvent lissable lorsqu'il est non-lisse. Il présente donc toutes les caractéristiques des problèmes à ne pas résoudre par la DFO, évoquées au chapitre 2.1.1. Toutefois, lorsqu'un problème de contrôle optimal présente des discontinuités, les méthodes de la littérature tombent en échec. Cette remarque a originellement motivé ce chapitre. Nos premiers mois de recherche dans cette direction nous ont menés à esquisser l'idée de reformuler le problème suivant ses variables impliquées dans des discontinuités [10] ; une idée qui a ensuite trouvé un cadre formel dans l'optimisation partitionnée. Nous disposons donc d'un outil pour traiter convenablement des problèmes de contrôle optimal via une approche exploitant les forces de la DFO.

Le chapitre 7.1 discute de la structure d'un problème de contrôle optimal, et localise les discontinuités potentielles. Nous identifions alors un moyen d'appliquer le cadre de la PO à certains de ces problèmes, pour les résoudre via la méthode associée. Nous proposons ensuite trois problèmes de contrôle optimal résolus par cette approche. L'un d'eux devient soluble analytiquement après reformulation via l'optimisation partitionnée, et les deux autres sont bien résolus numériquement. Ceci est discuté au chapitre 7.3. Il ressort de ce chapitre que la DFO a du potentiel en contrôle optimal via la PO, à minima sur la classe de problème que nous traitons ici. Enfin, le chapitre 7.4 identifie des pistes d'améliorations de la mise en application de la DFO dans le domaine du contrôle optimal.

7.1 Problème de contrôle optimal et discontinuités potentielles

Nous commençons par formaliser ce qu'est un problème de contrôle optimal et les discontinuités potentielles qu'il peut contenir, au chapitre 7.1.1. Ensuite, nous passons en revue la littérature pour observer que l'étude des discontinuités est incomplète, au chapitre 7.1.2. Enfin, nous identifions au chapitre 7.1.3 une classe de problèmes que nous pouvons résoudre par la PO. La discussion de cette démarche de résolution est repoussée au chapitre 7.2, et son application fait l'objet du chapitre 7.3.

7.1.1 Problème de contrôle optimal, coût de Mayer discontinu

Les problèmes de contrôle optimal (*Optimal Control*, OC) sont très étudiés, notamment dans plusieurs livres récents [33, 104, 112, 117]. La forme que nous considérons ici est la suivante. Sa spécificité porte sur le *coût de Mayer*, qui peut être discontinu dans notre étude.

Cadre 7.1. **OC à coût de Mayer possiblement discontinu.** Soit le problème

$$\begin{array}{lll} \underset{\substack{x \in \mathcal{AC}([0,T],\mathbb{R}^n) \\ u \in \mathcal{L}^\infty([0,T],\mathbb{R}^m)}}{\text{minimiser}} & g((x(t))_{t\in\zeta}) + \displaystyle\int_0^T \ell(t,x(t),u(t))\ dt & \\ \text{sous contraintes} & x(0) = x^0, & \\ & \dot{x}(t) = f(t,x(t),u(t)), & \text{p.p.t. } t \in [0,T], \\ & c(t,x(t),u(t)) \leq 0, & \text{p.p.t. } t \in [0,T], \\ & c_\zeta((x(t))_{t\in\zeta}) \leq 0. & \end{array} \qquad (\mathbf{P}_{\mathtt{OC}})$$

Les variables sont la *trajectoire* x, dans l'espace $\mathcal{AC}([0,T],\mathbb{R}^n)$ des fonctions absolument continues de $[0,T]$ dans $\mathbb{R}^n$, et le *contrôle* u, dans l'espace $\mathcal{L}^\infty([0,T],\mathbb{R}^m)$ des fonctions essentiellement bornées de $[0,T]$ dans $\mathbb{R}^m$, où $(n,m,T) \in \mathbb{N}^* \times \mathbb{N}^* \times \mathbb{R}_+^*$ sont fixés. La trajectoire est contrainte à partir de l'*état initial* $x^0 \in \mathbb{R}^n$, et elle est dirigée par la *dynamique contrôlée* $\dot{x}(t) = f(t,x(t),u(t))$ à presque tout instant $t \in [0,T]$ (où $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m$ est la *fonction de dynamique*). On souhaite minimiser le *coût de Bolza* défini comme la somme du *coût de Mayer* $g((x(t))_{t\in\zeta})$ et du *coût de Lagrange* $\int_0^t \ell(t,x(t),u(t))\ dt$, où l'ensemble $\emptyset \neq \zeta \subset [0,T]$ est fini et fixé, et $g : (\mathbb{R}^n)^{\#\zeta} \to \mathbb{R}$ est la *fonction de coût de Mayer*, et $\ell : [0,T] \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ est la *fonction de coût de Lagrange*. Enfin, le contrôle doit assurer le respect d'une *contrainte de roulement* $c(t,x(t),u(t)) \leq 0$ à presque tout instant $t \in [0,T]$ et d'une *contrainte d'états* $c_\zeta((x(t))_{t\in\zeta}) \leq 0$.

Dans ce qui suit, g et c_ζ sont possiblement discontinues. Aussi, nous supposons informellement que f, ℓ et c ne soulèvent pas de difficultés. Nous ne donnons pas de sens précis à cette hypothèse car la structure de f, ℓ et c n'importe pas dans ce chapitre. On suppose enfin que $\{0,T\} \subseteq \zeta$, pour alléger la présentation.

Les paramètres passés à g et c_ζ ne dépendent pas de l'intégralité du couple (x,u), mais uniquement des valeurs prises par x aux instants indexés par l'ensemble fini $\zeta \subset [0,T]$. On peut donc partitionner l'espace des variables suivant les valeurs prises par la trajectoire aux instants indexés par ζ, et appliquer le cadre de l'optimisation partitionnée sur ce partitionnement. L'idée est alors d'optimiser l'état atteint par la trajectoire aux instants indexés par ζ, en cherchant une trajectoire de coût minimal passant par ces états lorsqu'ils sont fixés.

7.1.2 État de l'art sur le contrôle optimal en présence de discontinuités

Les cas où f, g, ℓ et c sont Lipschitz-continues sont discutés dans les livres [33, 117]. Des conditions nécessaires d'optimalité sont également connues [49]. Le cadre historique du contrôle optimal suppose que f, g, ℓ et c_ζ sont lisses et c est nulle, et ($\mathbf{P}_{\mathtt{OC}}$) est alors résolu par une *méthode indirecte* calculatoire découlant du *Principe du maximum de Pontryagin* [89]. Donnons-en ici une formalisation dans un cas restreint, qui sert plus loin dans ce chapitre. En présence de contraintes de roulement, la théorie est plus complexe [28, 64].

Définition 7.1 (Principe du maximum de Pontryagin (PMP)). Avec $\zeta \triangleq \{T\}$, c_ζ une contrainte d'égalité $(x(0), x(T)) = (x^0, x^T)$, f et ℓ lisses et c nulle, une condition nécessaire d'optimalité locale au problème ($\mathbf{P}_{\mathtt{OC}}$) pour un couple (x, u) est

$$\begin{cases} x(0) = x^0, \quad x(T) = x^T, \quad \dot{x}(t) = f(t, x(t), u(t)), \\ p(0) \text{ libre}, \quad p(T) \text{ libre}, \quad \dot{p}(t) = p(t) f_x(t, x(t), u(t)) + p_0 \ell_x(t, x(t), u(t)), \\ \text{p.p.t. } t \in [0, T], \ u(t) \in \operatorname{argmax}_u p(t) f(t, x(t), u) + p_0 \ell(t, x(t), u), \end{cases} \tag{PMP}$$

où $f_x \triangleq \frac{\partial f}{\partial x}$ et $\ell_x \triangleq \frac{\partial \ell}{\partial x}$, où $p \in \mathcal{AC}([0, T], \mathbb{R}^n)$ est représentée pour tout $t \in [0, T]$ comme un vecteur ligne, et où $p_0 \in \{-1, 0\}$. On interdit le cas où p est la fonction nulle.

Peu de références étendent le (PMP) aux problèmes discontinus. À notre connaissance, au moment de la rédaction de cette thèse, les seules références existantes sont [57], gérant des discontinuités sur f, et [25], traitant les cas où ℓ représente une indicatrice.

Une autre approche consiste en des *méthodes directes*, où on discrétise ($\mathbf{P}_{\mathtt{OC}}$) dans le temps pour en faire un problème de dimension finie. De nombreuses stratégies numériques existent [30, 94]. Cependant, le problème discrétisé est de grande dimension, et ici discontinu. Une méthode de `DFO` est donc requise ; mais les deux approches proposées au chapitre 2 pour gérer la grande dimension échouent ici. L'approche cherchant à lisser convenablement le problème est difficile à appliquer à cause des discontinuités. L'approche visant à détecter si un petit nombre de variables indépendantes a une bien plus grande influence sur le problème que les autres échoue également, car les variables d'état et de contrôle instantané sont toutes liées par la contrainte de dynamique et l'influence individuelle de chacune est faible.

La stratégie d'optimiser les états aux instants $t \in \zeta$ est parfois nommée *optimisation de trajectoire* [47, 88]. Cette approche exploite le fait qu'entre deux instants $(t_1, t_2) \in [0, T]$ donnés et pour deux états $(x(t_1), x(t_2)) \in (\mathbb{R}^n)^2$ donnés, le problème consistant à rejoindre $x(t_2)$ depuis $x(t_1)$ entre les instants t_1 et t_2 à coût minimal est une instance de ($\mathbf{P}_{\mathtt{OC}}$). Cependant, [47, 88] requièrent des fonctions g et c_ζ lisses, et sont donc inapplicables ici.

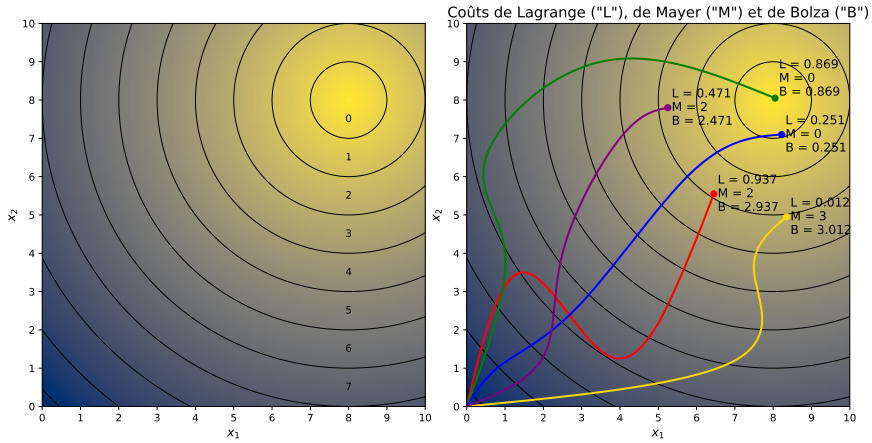
7.1.3 Classe simple de problèmes discontinus : problèmes à plateaux

Une façon simple de mettre en échec la littérature consiste à prendre le coût de Mayer

$$\left\lfloor \left\| x(T) - x^T \right\| \right\rfloor .$$

Ici, on a posé $\zeta \triangleq \{T\}$ et $g : \mathbb{R}^n \to \mathbb{R}$ définie par $g(\mathfrak{X}) \triangleq \lfloor \|\mathfrak{X} - x^T\| \rfloor$ pour tout $\mathfrak{X} \in \mathbb{R}^n$. Un tel coût de Mayer peut représenter, par exemple, une cible comme on en rencontre dans des sports de précision (tir, biathlon, archerie, curling, parachutisme, etc.). La figure 7.1 illustre une telle cible dans le cas $n = 2$, ainsi que quelques propriétés résultantes. D'abord, deux trajectoires finissant dans le même anneau de la cible ne sont départagées que par leur coût de Lagrange. Ensuite, une trajectoire avec un fort coût de Lagrange atteignant un anneau donné peut avoir un coût de Bolza plus faible qu'une autre trajectoire ayant un faible coût de Lagrange mais finissant dans un anneau de plus grand coût. Enfin, atteindre exactement le centre de la cible peut ne pas être optimal.

FIGURE 7.1 Cible circulaire et illustration de trajectoires approchant son centre.



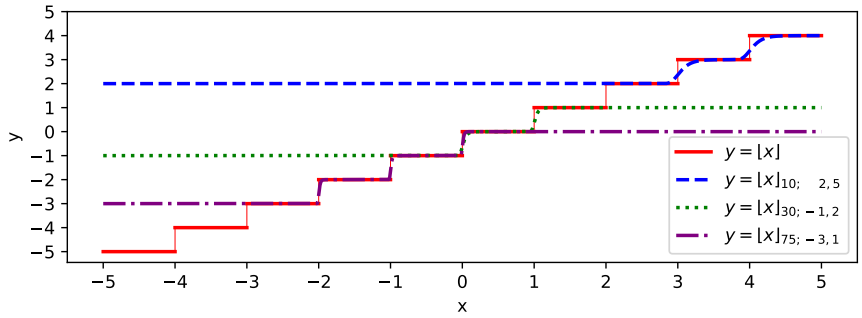
Ce problème est difficile à résoudre car le gradient du coût de Mayer est nul partout où il est défini. On peut localement perturber une trajectoire pour affaiblir son coût de Lagrange, mais on ne peut pas anticiper si cette perturbation fait passer l'état final dans un nouvel anneau de la cible. L'optimisation souffre de cette imprévisibilité due à l'aspect constant par morceaux du coût de Mayer. De plus, on ne peut pas négliger la troncature apportée par la partie entière, car ceci pourrait altérer sensiblement les solutions. Pour lisser convenablement le problème, on doit donc lisser la fonction partie entière. Nous formalisons ci-dessous le lissage que nous proposons dans [10, remarque 3.1], et nous le représentons en figure 7.2. Ce lissage a une dérivée proche de 0 en tout $x \in \mathbb{R}$ suffisamment loin de $\mathbb{Z}$. Ainsi, la partie entière lissée admet bien un gradient, mais celui-ci est très souvent inexploitable car quasi-nul.

Partons de la fonction indicatrice $\mathcal{I}_{\mathbb{R}_+}$, qui est approximée par $\mathcal{I}_{\mathbb{R}_+}(x) \approx \exp(-e^{-x})$ pour tout $x \in \mathbb{R}$. En paramétrant l'approximation d'un facteur $\tau \in \mathbb{R}_+^*$ et en déplaçant la discontinuité en n'importe quel $s \in \mathbb{R}$, nous pouvons donc lisser $\mathcal{I}_{[s,+\infty[}(x) \approx \exp(-e^{-\tau(x-s)})$ pour tout $x \in \mathbb{R}$. De là, on construit une approximation de $\lfloor \cdot \rfloor$ sur n'importe quel intervalle $[a, b]$, avec $(a, b) \in \mathbb{Z}^2$, $a < b$, et n'importe quel facteur $\tau \in \mathbb{R}_+^*$, comme

$$\forall x \in [a, b], \quad \lfloor x \rfloor = a + \sum_{s=a+1}^{b-1} \mathcal{I}_{[s,+\infty[}(x) \approx \lfloor x \rfloor_{\tau;a,b} \triangleq a + \sum_{s=a+1}^{b-1} \exp\left(-e^{-\tau(x-s)}\right). \qquad (\lfloor \cdot \rfloor_{\tau;a.b})$$

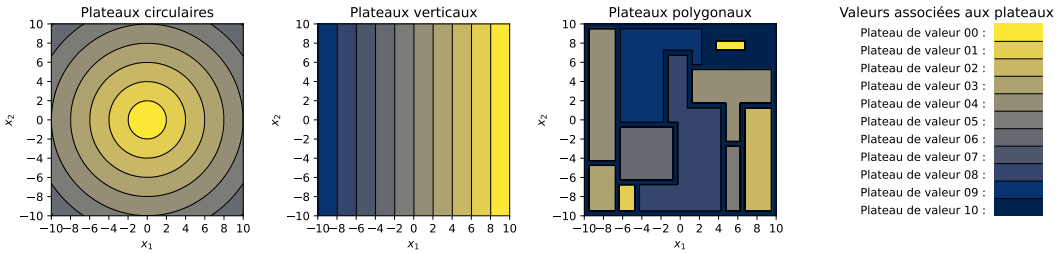
Cette approximation est infiniment dérivable, pour tout triplet $(\tau; a, b) \in \mathbb{R}_+^* \times \mathbb{Z} \times \mathbb{Z}$.

FIGURE 7.2 Fonction *partie entière* et ses lissages.



Dans les illustrations clôturant ce chapitre, nous portons une attention particulière à des problèmes ayant une fonction de coût de Mayer $g : (\mathbb{R}^n)^{\#\zeta} \to \mathbb{R}$ constante par morceaux en chacune de ses $\#\zeta$ variables. On parle alors de *plateaux* pour désigner les ensembles de continuité de g. La figure 7.3 illustre des plateaux pouvant résulter de fonctions $g : \mathbb{R}^2 \to \mathbb{R}$. Cette classe de problèmes présente toutes les difficultés évoquées pour le cas de la cible discrète. Ainsi, elle met en échec les méthodes de la littérature en contrôle optimal.

FIGURE 7.3 Exemples de plateaux de différents coûts.



7.2 Utilisation de l'optimisation partitionnée en contrôle optimal

Nous commençons par exprimer le problème ($\mathbf{P}_{\mathtt{OC}}$) sous la forme du problème ($\mathbf{P}_{\mathtt{PO}}^{\mathtt{ini}}$) pour le résoudre via l'optimisation partitionnée, au chapitre 7.2.1. Ensuite, au chapitre 7.2.2, nous proposons une méthode numérique reposant sur ce principe pour résoudre ($\mathbf{P}_{\mathtt{OC}}$) en pratique.

7.2.1 Validité du cadre de l'optimisation partitionnée

Pour explicitement écrire ($\mathbf{P}_{\mathtt{OC}}$) comme une instance du problème ($\mathbf{P}_{\mathtt{PO}}^{\mathtt{ini}}$), on définit en premier lieu $\mathbb{Y} \triangleq \mathcal{AC}([0,T],\mathbb{R}^n) \times \mathcal{L}^\infty([0,T],\mathbb{R}^m)$ muni de la norme

$$\forall (x,u) \in \mathbb{Y}, \quad \|(x,u)\|_{\mathbb{Y}} \triangleq \sup_{t\in[0,T]} \|x(t)\|_{\mathbb{R}^n} + \int_0^T \|u(t)\|_{\mathbb{R}^m}^2 \; dt.$$

Ensuite, pour partitionner $\mathbb{Y}$, on définit $\mathbb{X} \triangleq (\mathbb{R}^n)^{\#\zeta}$ et on pose

$$\forall \mathfrak{X} \in \mathbb{X}, \quad \mathbb{Y}(\mathfrak{X}) \triangleq \{(x,u) \in \mathbb{Y} : (x(t))_{t\in\zeta} = \mathfrak{X}\}.$$

Ceci revient à partitionner $\mathbb{Y}$ suivant l'état du système aux instants indexés par ζ. On a alors directement la fonction χ, donnée pour tout $(x,u) \in \mathbb{Y}$ par $\chi(x,u) \triangleq (x(t))_{t\in\zeta}$.

On définit également $\varphi(x,u) \triangleq g((x(t))_{t\in\zeta}) + \int_0^T \ell(t,x(t),u(t))\,dt$ pour tout $(x,u) \in \mathbb{Y}$, et Ω comme l'ensemble des éléments $(x,u) \in \mathbb{Y}$ vérifiant toutes les contraintes du problème ($\mathbf{P}_{\mathtt{OC}}$). De plus, pour tout $\mathfrak{X} \in \mathbb{X}$, le sous-problème consistant en la restriction de ($\mathbf{P}_{\mathtt{OC}}$) à $\mathbb{Y}(\mathfrak{X})$ est

$$\begin{array}{lll} \underset{\substack{x\in\mathcal{AC}([0,T],\mathbb{R}^n)\\ u\in\mathcal{L}^\infty([0,T],\mathbb{R}^m)}}{\text{minimiser}} & g(\mathfrak{X}) + \displaystyle\int_0^T \ell(t,x(t),u(t))\;dt & \\ \text{sous contraintes} & (x(t))_{t\in\zeta} = \mathfrak{X}, & \\ & \dot{x}(t) = f(t,x(t),u(t)), & \text{p.p.t. } t\in[0,T], \\ & c(t,x(t),u(t)) \le 0, & \text{p.p.t. } t\in[0,T], \\ & c_\zeta(\mathfrak{X}) \le 0. & \end{array} \qquad (\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X}))$$

Puisque f, ℓ et c sont lisses et $\mathfrak{X}$ est fixé, le sous-problème ($\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X})$) est un problème de contrôle optimal lisse à état initial et final fixés, rentrant totalement dans le cadre usuel de la littérature. Par conséquent, il peut être résolu par l'une des nombreuses méthodes dédiées, à déterminer au cas par cas selon la structure offerte par f, ℓ et c. Comme nous n'imposons pas de conditions sur cette structure, notre partitionnement est compatible avec l'intégralité de la littérature en contrôle optimal. De plus, ce problème est séparé puisque le calcul de

la trajectoire et du contrôle optimal entre deux valeurs successives de ζ est indépendant du calcul entre deux autres valeurs. Ainsi, on peut optimiser en parallèle le couple (trajectoire, contrôle) sur chacun des sous-intervalles de $[0, T]$ délimités par deux valeurs successives de ζ (rappelons qu'on a considéré $0 \in \zeta$ et $T \in \zeta$ pour simplifier les notations). Rappelons toutefois que nous devons supposer que $(\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X}))$ admet un unique couple $\gamma(x, u) \in \mathbb{Y}(\mathfrak{X})$ globalement optimal, pour tout $\mathfrak{X} \in X \triangleq \{\mathfrak{X} \in \mathbb{X} : \mathbb{Y}(\mathfrak{X}) \cap \Omega \neq \emptyset\}$.

Ainsi, le problème $(\mathbf{P}_{\mathtt{OC}})$ entre dans le cadre d'étude 5.1. Ceci permet l'application de la PO discutée au chapitre 5. Pour résoudre $(\mathbf{P}_{\mathtt{OC}})$, on résout donc le problème reformulé

$$\underset{\mathfrak{X} \in \mathbb{X}}{\text{minimiser}} \quad \Phi(x), \qquad \text{avec} \qquad \Phi : \begin{cases} \mathbb{X} & \to & \overline{\mathbb{R}} \\ x & \mapsto & \begin{cases} \varphi(\gamma(\mathfrak{X})) & \text{si } x \in X, \\ +\infty & \text{sinon,} \end{cases} \end{cases} \tag{$\mathbf{P}_{\mathtt{OC}}^{\mathtt{ref}}$}$$

et via les propositions 5.1 et 5.2, on sait garantir qu'une solution locale $\mathfrak{X}^*$ de ce problème mène à $\gamma(\mathfrak{X}^*)$ solution locale du problème $(\mathbf{P}_{\mathtt{OC}})$. Ces propositions requièrent la continuité locale de χ, qui est toujours avérée ici, et la continuité locale de γ, qui doit être analysée au cas par cas.

Des pistes de recherche existent pour rendre le cadre de la PO moins strict (voir le chapitre 5.3). Toutefois, même lorsqu'on ne parvient pas à garantir l'unicité de la solution optimale du sous-problème pour tout $\mathfrak{X} \in X$, on peut tout de même appliquer le cadre de manière heuristique (c'est-à-dire, sans garantir tous les pré-requis). Pour tout $\mathfrak{X} \in X$, on résout le sous-problème $(\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X}))$ par une méthode du contrôle optimal, et on récupère la solution renvoyée comme substitut à $\gamma(\mathfrak{X})$. Les exemples aux chapitres 7.3.2 et 7.3.3 montrent que même dans cette application heuristique, la résolution de $(\mathbf{P}_{\mathtt{OC}}^{\mathtt{ref}})$ par un algorithme de la DFO donne tout de même des solutions satisfaisantes au problème $(\mathbf{P}_{\mathtt{OC}})$.

De plus, lorsqu'on doit résoudre $(\mathbf{P}_{\mathtt{OC}}^{\mathtt{ref}})$ algorithmiquement, le cadre de la PO requiert en plus les hypothèses 5.1 et 5.2. L'hypothèse 5.1 est vérifiée. Pour tout $(x, u) \in \mathbb{Y}$ et $(y, v) \in \mathbb{Y}$, on a $\|\chi(y, v) - \chi(x, u)\|_{\mathbb{X}} = \|(y(t) - x(t))_{t \in \zeta}\|_{\mathbb{X}} \leq \|(y - x, v - u)\|_{\mathbb{Y}} = \|(y, v) - (x, u)\|_{\mathbb{Y}}$, donc χ est Lipschitz-continue, ce qui garantit aussi que l'image par χ d'un ensemble borné de $\mathbb{Y}$ est un ensemble borné de $\mathbb{X}$. Par contre, l'hypothèse 5.2 liée à la fonction oracle est difficile à garantir en toute généralité. Cependant, en pratique, on peut résoudre le problème $(\mathbf{P}_{\mathtt{OC}}^{\mathtt{ref}})$ via un algorithme de DFO même lorsque l'hypothèse 5.2 n'est pas vérifiée. La solution $\widetilde{\mathfrak{X}}^*$ calculée n'a aucune garantie de donner $\gamma(\widetilde{\mathfrak{X}}^*)$ une solution locale au problème $(\mathbf{P}_{\mathtt{OC}})$, mais cette méthode heuristique est performante sur les exemples des chapitres 7.3.2 et 7.3.3.

7.2.2 Méthode numérique de résolution via l'optimisation partitionnée

Dans les exemples qui suivent, certains problèmes ne peuvent pas être résolus analytiquement après reformulation par la PO. Pour ces problèmes, nous appliquons alors un processus de résolution numérique noté $\mathbb{M}_{\mathtt{PO}}$, qui n'est qu'heuristique mais vise à la performance. La description de ce processus et les choix techniques retenus est donnée en fin de ce chapitre, car nous devons d'abord mentionner les autres méthodes de résolution auxquelles on le compare pour juger de sa performance. Nous devons également mentionner quelques opérations de préconditionnement du problème ($\mathbf{P}_{\mathtt{OC}}$) effectuées à priori. Le contenu de ce chapitre suit notre référence [10, section 3.2].

Préconditionnement du problème par lissage et discrétisation temporelle

Dans nos tests numériques, nous privilégions des méthodes algorithmiques, qui reposent sur une résolution *directe* fondée sur une discrétisation temporelle du problème ($\mathbf{P}_{\mathtt{OC}}$). De fait, nous ne considérons pas ($\mathbf{P}_{\mathtt{OC}}$) mais la discrétisation temporelle suivante :

$$\begin{array}{ll}
\underset{\substack{(x^k)_{k=1}^N \in (\mathbb{R}^n)^N,\\ (u^k)_{k=0}^{N-1} \in (\mathbb{R}^m)^N}}{\text{minimiser}} & g((x^k)_{k\in\zeta_{\mathtt{d}}}) + \sum_{k=0}^{N-1} \ell^k(x^k, x^{k+1}, u^k, u^{k+1}) \\
\text{sous contraintes} & x^{k+1} = f^k(x^k, u^k), \quad \forall k \in [\![0, N-1]\!], \\
& c^k(x^k, u^k) \leq 0, \qquad \forall k \in [\![0, N]\!], \\
& c_{\zeta_{\mathtt{d}}}((x^k)_{k\in\zeta_{\mathtt{d}}}) \leq 0,
\end{array}$$

où $N \in \mathbb{N}^*$ est le nombre de subdivisions (qu'on prendra uniformes) de $[0, T]$, où on considère par simplicité que tous les éléments de ζ sont inclus dans l'ensemble $\{t^k \triangleq \frac{kT}{N}\}_{k=0}^N$ des temps discrétisés et où on note $\zeta_{\mathtt{d}} \triangleq \{k \in [\![0, N]\!] : t^k \in \zeta\}$. La valeur $x^0 \in \mathbb{R}^n$ est fixée. Pour tout $k \in [\![0, N-1]\!]$, ℓ^k est une discrétisation de l'intégrale de la fonction de coût de Lagrange sur $[t^k, t^{k+1}]$ (en pratique, on prend une méthode du trapèze), f^k est une discrétisation de la dynamique sur $[t^k, t^{k+1}]$ (on prend un schéma de Runge-Kutta d'ordre 4 avec 10 subdivisions par unité de temps et les paramètres usuels, mais on pourrait préférer d'autres choix suivant des analyses de l'impact de la modélisation [7, 40, 102, 103]) ; et pour tout $k \in [\![0, N]\!]$, c^k est une discrétisation de la contrainte de roulement à l'instant t^k.

De plus, pour des raisons liées aux solveurs employés, nous devons lisser la fonction g. Dans les exemples qui suivent, g est une fonction constante par morceaux définie via la fonction partie entière. Nous appliquons donc le lissage ($\lfloor\cdot\rfloor_{\tau;a.b}$) (avec $\tau \triangleq 75$ et (a, b) bien choisis en fonction du problème), pour lisser g en $\widetilde{g}$. On considère finalement le problème suivant.

$$\begin{array}{lll} \underset{\substack{(x^k)_{k=1}^N \in (\mathbb{R}^n)^N, \\ (u^k)_{k=0}^{N-1} \in (\mathbb{R}^m)^N}}{\text{minimiser}} & \widetilde{g}((x^k)_{k\in\zeta_{\mathtt{d}}}) + \sum_{k=0}^{N-1} \ell^k(x^k, x^{k+1}, u^k, u^{k+1}) & \\ \text{sous contraintes} & x^{k+1} = f^k(x^k, u^k), \quad \forall k \in [\![0, N-1]\!], & (\mathbf{P}_{\mathtt{dOC}}) \\ & c^k(x^k, u^k) \leq 0, \qquad \forall k \in [\![0, N]\!], & \\ & c_{\zeta_{\mathtt{d}}}((x^k)_{k\in\zeta_{\mathtt{d}}}) \leq 0. & \end{array}$$

Tentative de résolution par IPOPT : méthode $\mathbb{M}_{\mathtt{SO}}$

Puisque le problème ($\mathbf{P}_{\mathtt{dOC}}$) est un problème d'optimisation lisse usuel, nous pouvons envisager de le résoudre via un solveur dédié. Notre première méthode de référence consiste donc à résoudre ($\mathbf{P}_{\mathtt{dOC}}$) via le solveur d'optimisation lisse **IPOPT** [118]. Pour ce faire, nous implémentons les instances de ($\mathbf{P}_{\mathtt{dOC}}$) dans le langage de modélisation **JuMP** [79] et nous appelons l'interface d'**IPOPT** associée. Nous notons cette stratégie $\mathbb{M}_{\mathtt{SO}}$. Puisque g est constante par morceaux et que $\widetilde{g}$ est donc quasi-constante partout, nous nous attendons à ce que cette méthode soit peu fiable en général. Vérifier cette affirmation prouve le fait que les problèmes à plateaux ne sont généralement pas solubles par les stratégies usuelles de la littérature.

Tentative de résolution par NOMAD : méthode $\mathbb{M}_{\mathtt{DFO}}$

Le problème ($\mathbf{P}_{\mathtt{dOC}}$) mettant **IPOPT** en échec, nous pourrions être tenté d'utiliser à la place une méthode sans dérivées. Nous montrons donc que cette stratégie est également inefficace. Le problème ($\mathbf{P}_{\mathtt{dOC}}$) contient beaucoup trop de variables et de contraintes fortes pour que des solveurs comme **NOMAD** [19] ou **PRIMA** [123] puissent le résoudre. Cependant, nous remarquons que nous pouvons atténuer cette difficulté en réduisant la dimension du problème : pour toute valeur fixée de $(u^k)_{k=0}^{N-1} \in (\mathbb{R}^m)^N$, la trajectoire $(x^k)_{k=1}^N \in (\mathbb{R}^n)^N$ est entièrement déterminée (via les contraintes de dynamique $x^{k+1} = f^k(x^k, u^k)$ pour tout $k \in [\![0, N-1]\!]$). Ainsi, nous implémentons dans **JuMP** le problème réduit

$$\begin{array}{lll} \underset{(u^k)_{k=0}^{N-1} \in (\mathbb{R}^m)^N}{\text{minimiser}} & \widetilde{g}((x^k)_{k\in\zeta_{\mathtt{d}}}) + \sum_{k=0}^{N-1} \ell^k(x^k, x^{k+1}, u^k, u^{k+1}) & \\ \text{sous contraintes} & x^{k+1} = f^k(x^k, u^k), \quad \forall k \in [\![0, N-1]\!], & \\ & c^k(x^k, u^k) \leq 0, \qquad \forall k \in [\![0, N]\!], & \\ & c_{\zeta_{\mathtt{d}}}((x^k)_{k\in\zeta_{\mathtt{d}}}) \leq 0, & \end{array}$$

et nous le résolvons via l'interface **JuMP** de **NOMAD** [82]. Nous notons cette stratégie $\mathbb{M}_{\mathtt{DFO}}$. Cependant, ce problème reste de grande dimension, aussi $\mathbb{M}_{\mathtt{DFO}}$ reste inefficace.

Résolution par optimisation partitionnée : méthode $\mathbb{M}_{\mathtt{PO}}$

La troisième méthode que nous proposons consiste conceptuellement à résoudre la reformulation ($\mathbf{P}_{\mathtt{OC}}^{\mathtt{ref}}$) via **NOMAD**, bien que certains détails techniques doivent être altérés. Dans un but de comparaison honnête avec les autres méthodes, le sous-problème doit être lissé et discrétisé dans le temps. On considère donc

$$\begin{aligned}
\underset{\substack{(x^k)_{k=1}^N \in (\mathbb{R}^n)^N,\\ (u^k)_{k=0}^{N-1} \in (\mathbb{R}^m)^N}}{\text{minimiser}} \quad & \widetilde{g}(\mathfrak{X}) + \sum_{k=0}^{N-1} \ell^k(x^k, x^{k+1}, u^k, u^{k+1}) \\
\text{sous contraintes} \quad & (x^k)_{k\in\zeta_{\mathtt{d}}} = \mathfrak{X}, \\
& x^{k+1} = f^k(x^k, u^k), \quad \forall k \in [\![0, N-1]\!], \\
& c^k(x^k, u^k) \leq 0, \qquad \forall k \in [\![0, N]\!], \\
& c_{\zeta_{\mathtt{d}}}(\mathfrak{X}) \leq 0.
\end{aligned} \tag{$\mathbf{P}_{\mathtt{dOC}}^{\mathtt{sub}}(\mathfrak{X})$}$$

Nous notons également la fonction d'infaisabilité liée au sous-problème ($\mathbf{P}_{\mathtt{dOC}}^{\mathtt{sub}}(\mathfrak{X})$) par

$$h(\mathfrak{X}; x, u) \quad \triangleq \quad \left\|(x^k)_{k\in\zeta_{\mathtt{d}}} - \mathfrak{X}\right\|^2 + \sum_{k=0}^{N-1} \left\|x^{k+1} - f^k(x^k, u^k)\right\|^2 + \sum_{k=0}^{N} \max\{c^k(x^k, u^k), 0\}^2$$

pour tout $(x, u) \in (\mathbb{R}^n)^N \times (\mathbb{R}^m)^N$. Notons que la contrainte $c_{\zeta_{\mathtt{d}}}(\mathfrak{X}) \leq 0$ n'est pas intégrée à cette fonction puisqu'elle est transparente pour le sous-problème ($\mathbf{P}_{\mathtt{dOC}}^{\mathtt{sub}}(\mathfrak{X})$).

En pratique, nous résolvons ($\mathbf{P}_{\mathtt{dOC}}^{\mathtt{sub}}(\mathfrak{X})$) par **IPOPT**. Suivant la discussion proposée au chapitre 5.3.3, **IPOPT** renvoie $\widetilde{\gamma}(\mathfrak{X})$ qui est une approximation de solution locale de ($\mathbf{P}_{\mathtt{dOC}}^{\mathtt{sub}}(\mathfrak{X})$) lorsqu'une telle solution existe, et un minimiseur de l'infaisaiblité $h(\mathfrak{X}; \cdot)$ autrement. La reformulation du problème ($\mathbf{P}_{\mathtt{dOC}}$) que nous considérons est alors

$$\underset{\mathfrak{X} \in (\mathbb{R}^n)^{\#\zeta}}{\text{minimiser}} \quad \widetilde{\Phi}(\mathfrak{X}) \quad \text{sous contrainte} \quad \widetilde{h}(\mathfrak{X}) \leq 0 \text{ et } c_{\zeta_{\mathtt{d}}}(\mathfrak{X}) \leq 0, \tag{$\widetilde{\mathbf{P}}_{\mathtt{dOC}}^{\mathtt{ref}}$}$$

où $\widetilde{\Phi}(\mathfrak{X}) \triangleq \varphi(\widetilde{\gamma}(\mathfrak{X}))$ et $\widetilde{h} \triangleq h(\mathfrak{X}, \xi; \widetilde{\gamma}(\mathfrak{X}))$ pour tout $\mathfrak{X} \in (\mathbb{R}^n)^{\#\zeta}$. Nous résolvons alors la reformulation ($\widetilde{\mathbf{P}}_{\mathtt{dOC}}^{\mathtt{ref}}$) par **NOMAD** exploitant la barrière progressive. La solution $\widetilde{\mathfrak{X}}^*$ renvoyée par **NOMAD** mène à $\widetilde{\gamma}(\widetilde{\mathfrak{X}}^*)$ comme une approximation de solution du problème ($\mathbf{P}_{\mathtt{dOC}}$).

Nous notons cette méthode $\mathbb{M}_{\mathtt{PO}}$. Notons que $\mathbb{M}_{\mathtt{PO}}$ est heuristique, car nos choix d'implémentation ne préservent pas le cadre d'étude 5.1 requis par l'analyse formelle de sa convergence. Toutefois, elle devrait bien fonctionner en pratique.

Trajectoire sans contrôle comme initialisation des méthodes

La *trajectoire non contrôlée* partant de x^0 est définie via $u^k \triangleq 0$ et $x^{k+1} \triangleq f^k(x^k, u^k)$ pour tout $k \in [\![0, N-1]\!]$. Cette trajectoire (x, u) est notée $\mathbb{M}_0$. On initialise $\mathbb{M}_{\mathtt{SO}}$ à (x, u) et $\mathbb{M}_{\mathtt{DFO}}$ à u. Dans la méthode $\mathbb{M}_{\mathtt{PO}}$, l'initialisation de **NOMAD** résolvant $(\widetilde{\mathbf{P}}^{\mathtt{ref}}_{\mathtt{dOC}})$ est $(x^k)_{k \in \zeta_{\mathtt{d}}}$, et pour tout $\mathfrak{X} \in \mathbb{X}$, $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{dOC}}(\mathfrak{X}))$ est résolu par **IPOPT** initialisé en (x, u).

7.3 Mise en application sur des problèmes à plateaux

Ce chapitre illustre comment l'optimisation partitionnée permet de résoudre des problèmes de contrôle optimal à plateaux. Le chapitre 7.3.1 montre qu'un tel problème peut devenir soluble analytiquement. Les chapitres 7.3.2 et 7.3.3 montrent que même si on ne vérifie pas tous les pré-requis théoriques liés au cadre de l'optimisation partitionnée, son usage donne tout de même de bonnes solutions aux problèmes considérés.

7.3.1 Solution analytique d'un problème de double intégrateur

Nous considérons ici une instance du problème $(\mathbf{P}_{\mathtt{OC}})$ qui présente un coût de Mayer à plateaux, mais qui devient soluble analytiquement après application du cadre de l'optimisation partitionnée. Le problème consiste à contrôler un *double intégrateur* dont on veut minimiser la *partie plafond* $\lceil \cdot \rceil$ de la position à l'instant final sous la contrainte de vitesse finale fixée à $v^T \in \mathbb{R}$, depuis l'état initial $x^0 \in \mathbb{R}^2$, à accélération minimale :

$$\begin{array}{lll} \underset{\substack{x \in \mathcal{AC}([0,T],\mathbb{R}),\ v \in \mathcal{AC}([0,T],\mathbb{R}) \\ u \in \mathcal{L}^\infty([0,T],\mathbb{R})}}{\text{minimiser}} & \lceil x(T) \rceil + \dfrac{1}{T} \displaystyle\int_0^T u(t)^2 \, dt & \\ \text{sous contraintes} & (x, v)(0) = (x^0, v^0), & \\ & \dot{x}(t) = v(t), & \text{p.p.t. } t \in [0, T], \\ & \dot{v}(t) = u(t), & \text{p.p.t. } t \in [0, T], \\ & v(T) = v^T, & \end{array} \tag{$\mathbf{P}_{\mathtt{OC,DI}}$}$$

On applique la stratégie proposée au chapitre 7.2.1. Ici, on peut même prendre un partitionnement plus subtil pour diminuer encore la difficulté du problème reformulé résultant. En effet, seul la position finale intervient dans la discontinuité du coût de Mayer ; tandis que la vitesse finale n'affecte aucune discontinuité. Ainsi, on peut se contenter de partitionner selon la position finale uniquement, et résoudre le sous-problème consistant en l'ajout d'une contrainte de position finale au problème $(\mathbf{P}_{\mathtt{OC,DI}})$. Pour tout $\mathfrak{X} \in \mathbb{R}$ donné (représentant la

position finale imposée), on s'intéresse donc au sous-problème

$$\begin{array}{lll} \underset{\substack{x \in \mathcal{AC}([0,T],\mathbb{R}),\ v \in \mathcal{AC}([0,T],\mathbb{R}) \\ u \in \mathcal{L}^\infty([0,T],\mathbb{R})}}{\text{minimiser}} & \lceil \mathfrak{X} \rceil + \dfrac{1}{2T}\displaystyle\int_0^T u(t)^2\ dt & \\ \text{sous contraintes} & (x,v)(0) = (x^0, v^0), & \\ & \dot{x}(t) = v(t), & \text{p.p.t. } t \in [0,T], \\ & \dot{v}(t) = u(t), & \text{p.p.t. } t \in [0,T], \\ & (x,v)(T) = (\mathfrak{X}, v^T), & \end{array} \qquad (\mathbf{P}^{\mathtt{sub}}_{\mathtt{OC,DI}}(\mathfrak{X}))$$

dont on peut se convaincre qu'il est globalement résoluble par application du (PMP).

L'application du (PMP) au sous-problème $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{dOC}}(\mathfrak{X}))$ mène au système

$$\begin{cases} x(0) = x^0, & \dot{x} = v, & x(T) = \mathfrak{X}, \\ v(0) = v^0, & \dot{v} = u, & v(T) = v^T, \\ & \dot{p}_1 = 0, & \\ & \dot{p}_2 = p_1, & \\ \forall t,\ u(t) \in \operatorname{argmax}_u p_1(t)v(t) + p_2(t)u + p_0 \frac{1}{2T}u^2, & & \end{cases}$$

d'où on déduit que pour presque tout $t \in [0,T]$,

$$\begin{cases} x(t) = x^0 + \int_0^t v(s)\ ds, & x(T) = \mathfrak{X}, \\ v(t) = v^0 + \int_0^t u(s)\ ds, & v(T) = v^T, \\ p_1(t) = a, & \\ p_2(t) = b + at, & \\ u(t) \in \operatorname{argmax}_u u\left(2Tp_2(t) + p_0 u\right). & \end{cases}$$

Il n'y a pas de solution anormale (c'est-à-dire, avec $p_0 = 0$) ici.[1] On a donc $p_0 = -1$, d'où

$$\begin{cases} x(t) = x^0 + \int_0^t v(s)\ ds, & x(T) = \mathfrak{X}, \\ v(t) = v^0 + \int_0^t u(s)\ ds, & v(T) = v^T, \\ p_1(t) = a, & \\ p_2(t) = b + at, & \\ u(t) = Tp_2(t). & \end{cases}$$

1. Si $p_0 = 0$, alors soit p_2 n'est pas la fonction nulle, mais alors le contrôle vaut $\pm\infty$ (selon le signe de $p_2(t)$) à presque tout instant, et ainsi $u \notin \mathcal{L}^\infty([0,T],\mathbb{R})$, soit p_2 est la fonction nulle, mais alors on est dans la solution triviale $p(t) = 0$ pour tout $t \in [0,T]$ (qui est rejetée par le (PMP)).

La résolution s'achève en identifiant a et b vérifiant la contrainte d'état final, c'est-à-dire le couple d'équations $(x(T), v(T)) = (\mathfrak{X}, v^T)$. Après quelques calculs, ceci se traduit par

$$\begin{bmatrix} \frac{T^4}{6} & \frac{T^3}{2} \\ \frac{T^3}{2} & T^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \mathfrak{X} - x^0 - v^0 T \\ v^T - v^0 \end{bmatrix} \quad \text{ou encore} \quad \begin{bmatrix} a \\ b \end{bmatrix} = \frac{-12}{T^6} \begin{bmatrix} T^2 & \frac{-T^3}{2} \\ \frac{-T^3}{2} & \frac{T^4}{6} \end{bmatrix} \begin{bmatrix} \mathfrak{X} - x^0 - v^0 T \\ v^T - v^0 \end{bmatrix},$$

d'où on tire

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \dfrac{12}{T^4}\left(x^0 + v^0 T - \mathfrak{X}\right) + \dfrac{6}{T^3}\left(v^T - v^0\right) \\ \dfrac{6}{T^3}\left(\mathfrak{X} - x^0 - v^0 T\right) + \dfrac{2}{T^2}\left(v^0 - v^T\right) \end{bmatrix}.$$

Ainsi, on en déduit la solution globale du sous-problème $(\mathbf{P}^{\mathtt{sub}}_{\mathtt{OC,DI}}(\mathfrak{X}))$ pour tout $\mathfrak{X} \in \mathbb{R}$. En notant $s \triangleq \frac{t}{T}$, on trouve après quelques simplifications

$$\begin{aligned} x(t) &= x^0 + s^2\left(3 - 2s\right)\left(\mathfrak{X} - x^0\right) + s\left(1 - s\right) T v^0 - \left(1 - s\right) s^2 T \left(v^T + v^0\right), \\ v(t) &= \left(1 - 2s\right) v^0 + s\left(3s - 2\right)\left(v^0 + v^T\right) + 6s^2\left(1 - s\right)\frac{\mathfrak{X} - x^0}{T}, \\ u(t) &= 6\left(1 - 2s\right)\frac{\mathfrak{X} - x^0}{T^2} + 2\left(1 - 3s\right)\frac{v^0 + v^T}{T}. \end{aligned}$$

On note alors $\gamma(\mathfrak{X}) \triangleq (x, v, u)$, avec (x, v, u) donné comme ci-dessus. Sa valeur objectif est

$$\varphi\left(\gamma(\mathfrak{X})\right) = \lceil \mathfrak{X} \rceil + \frac{2}{T^2}\left(3\left(\frac{\mathfrak{X} - x^0}{T^2}\right)^2 + 3\frac{\mathfrak{X} - x^0}{T^2}\frac{v^0 - v^T}{T} + \left(\frac{v^0 - v^T}{T}\right)^2\right).$$

On peut remarquer que γ est continue. Sachant que χ l'est également, les propositions 5.1 et 5.2 affirment que tout $\mathfrak{X}$ minimisant $\varphi \circ \gamma$ donne $\gamma(\mathfrak{X})$ solution du problème $(\mathbf{P}_{\mathtt{OC,DI}})$. De plus, ici la minimisation de $\varphi \circ \gamma$ est possible analytiquement, puisqu'il s'agit d'une quadratique (d'une seule variable réelle) en $\mathfrak{X}$ plus la partie plafond de $\mathfrak{X}$. [2]

Notons que dans cette résolution analytique, nous n'avons pas besoin de vérifier que les hypothèses 5.1 et 5.2 tiennent. En effet, ces hypothèses ne servent que dans l'analyse de convergence de `cDSM` résolvant le problème reformulé. Ici, nous n'avons pas eu besoin de `cDSM` mais seulement des propositions 5.1 et 5.2, qui ne requièrent que la garantie que le cadre de l'optimisation partitionnée est applicable et la continuité locale de γ et χ.

2. Soit $q : z \in \mathbb{R} \mapsto a(z - z_0)^2 + b \in \mathbb{R}$ avec $z_0 \in \mathbb{R}$ et $a > 0$ et $b \in \mathbb{R}$, et $f \triangleq \lceil \cdot \rceil + q(\cdot)$. Tout point $z \notin \mathbb{Z} \cup \{z_0\}$ n'est pas un minimiseur local de f, et de plus on a $f(z) > f(z_0)$ lorsque $z > z_0$ et $f(z) \to +\infty$ lorsque $z \to -\infty$. Ainsi, l'ensemble $\mathcal{Z} \triangleq \{z \in \mathbb{Z} \cup \{z_0\} : f(z) \leq f(z_0)\} \subset \left]-\infty, z_0\right]$ est fini. Pour trouver le minimiseur global de f, il suffit donc de partir de z_0 puis de parcourir tous les points de $\mathcal{Z}$ par valeurs décroissantes tant que la valeur de f associée décroît. Le point d'inflexion est le minimiseur global.

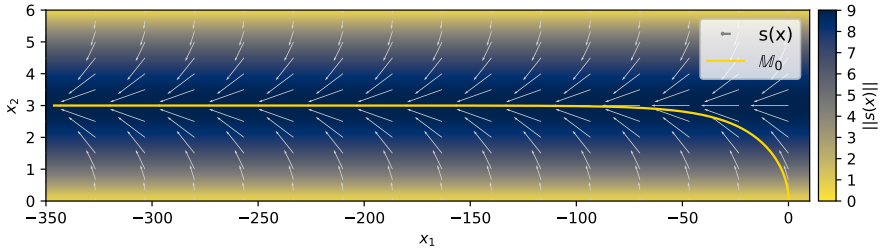
7.3.2 Solution numérique d'un problème de Zermelo à plateaux circulaires

Nous étudions ici un problème dit *de Zermelo*, modélisant le contrôle optimal d'un navire sur une rivière, présenté comme le problème de cible discrète suivant :

$$\begin{array}{lll} \underset{\substack{x \in \mathcal{AC}([0,T],\mathbb{R}^2) \\ (u,\theta) \in \mathcal{L}^\infty([0,T],\mathbb{R}^2)}}{\text{minimiser}} & \left\lfloor \left\| x\left(\frac{T}{2}\right) - A \right\|_a \right\rfloor + \lfloor \|x(T) - B\|_b \rfloor + \dfrac{1}{T}\displaystyle\int_0^T u(t)\,dt & \\ \text{sous contraintes} & x(0) = (0,0), & \\ & \dot{x}(t) = s(x(t)) + u(t)\mathbb{1}_{\theta(t)}, & \text{p.p.t. } t \in [0,T], \\ & 0 \leq x_2(t) \leq h, & \forall t \in [0,T], \\ & 0 \leq u(t) \leq 1, & \text{p.p.t. } t \in [0,T], \end{array} \qquad (\mathbf{P}_{\texttt{OC,Zt}})$$

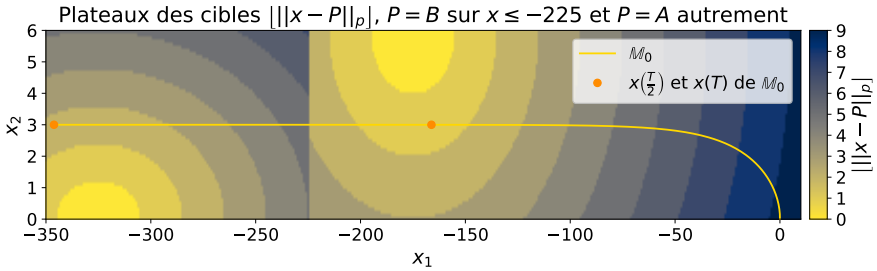
où $A \in \mathcal{R}$ et $B \in \mathcal{R}$ sont deux points fixés de la rivière modélisée par $\mathcal{R} \triangleq \mathbb{R} \times [0,h]$ pour un certain $h > 0$ fixé, où $\|x\|_\alpha \triangleq ((\alpha_1 x_1)^2 + (\alpha_2 x_2)^2)^{\frac{1}{2}}$ pour tout $x \in \mathbb{R}^2$ et $\alpha \in \mathbb{R}^2_+$, où $\mathbb{1}_\theta \triangleq (\cos\theta, \sin\theta)$ pour tout $\theta \in \mathbb{R}$, et où s, défini comme $s(x) \triangleq (x_2(x_2 - h), \frac{8}{10}\cos(\pi\frac{x_2}{h}))$ pour tout $x \in \mathcal{R}$ et illustré dans la figure 7.4, est le courant de la rivière. Ce problème modélise la recherche du contrôle minimal pour passer au plus proche des points A et B, où la distance à ces points est mesurée dans un sens discret et aux instants $\frac{T}{2}$ et T respectivement.

FIGURE 7.4 Modélisation du courant de la rivière dans le problème de Zermelo.



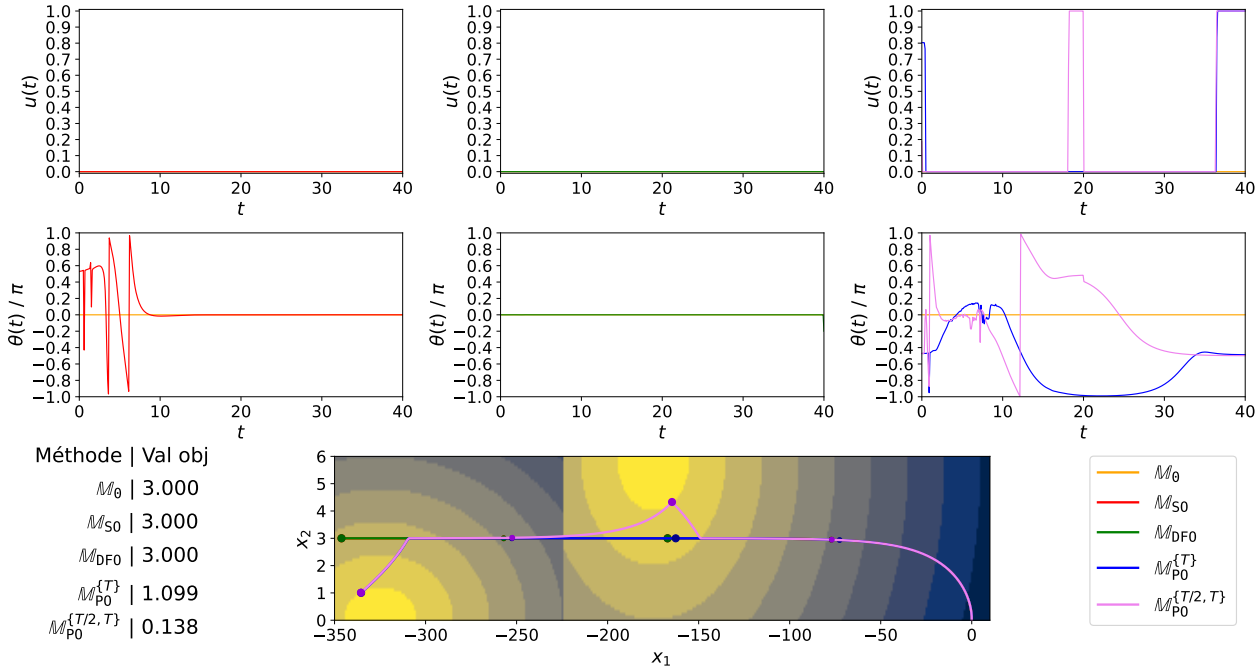
On pose $h \triangleq 6$ et $T \triangleq 40$. On définit les cibles via $A \triangleq (-175, 6)$ et $a \triangleq (\frac{1}{20}, \frac{1}{2})$; et via $B \triangleq (-325, 0)$ et $b \triangleq (\frac{1}{20}, \frac{5}{6})$. Le coût de Mayer de ce problème est à plateaux et dépend de la trajectoire aux instants $\frac{T}{2}$ et T. Pour résoudre efficacement le problème, on partitionne donc l'espace des variables suivant les valeurs prises par x aux instants $\frac{T}{2}$ et T. La figure 7.5 représente les cibles discrètes associées à A et B, et montre que la trajectoire sans contrôle approche grossièrement ces points aux instants désirés. On peut donc présupposer qu'une solution optimale consiste à laisser le navire dériver vers les cibles grâce au courant, et à ne prendre des contrôles (non nuls et) maximaux que sur un court intervalle de temps précédant les instants $\frac{T}{2}$ et T. Cette conjecture trouve écho dans la théorie des contrôles *turnpike* [113].

FIGURE 7.5 Cibles discrètes à atteindre sur la rivière.



Ni le problème ($\mathbf{P}_{\texttt{OC,Zt}}$) ni son sous-problème après partition ne sont solubles par le (PMP). On doit donc appliquer le processus numérique proposé au chapitre 7.2.2. Ici, on peut anticiper le comportement des différentes méthodes qu'on propose. Cet exemple est construit de sorte que la dynamique et le coût de Lagrange incitent localement le navire à ne pas approcher les cibles. Le courant tend à repousser le navire des côtes, donc affronter le courant est pénalisé par le coût de Lagrange. De plus, la nullité du gradient associé au coût de Mayer ne permet pas de détecter l'intérêt de confronter le courant, ni à l'instant final T ni à l'instant moyen $\frac{T}{2}$. Ainsi, seule une méthode faisant une recherche non-infinitésimale aux instants $\frac{T}{2}$ et T pour explicitement tenir compte du coût de Mayer peut détecter l'intérêt d'affronter le courant pour amener le navire à un point donné à un instant donné. Les résultats de chacune des méthodes sont illustrés en figure 7.6.

FIGURE 7.6 Résultats numériques sur le problème de cibles discrètes de Zermelo.



La trajectoire sans contrôle, $\mathbb{M}_0$, est une initialisation décente pour une optimisation mais n'est pas satisfaisante en tant que solution approximative. La méthode $\mathbb{M}_{\mathtt{SO}}$ part de $\mathbb{M}_0$, qui est une trajectoire donnant des gradients du coût de Mayer nuls. Comme, de plus, le coût de Lagrange incite à ne pas contrôler la trajectoire, la méthode $\mathbb{M}_{\mathtt{SO}}$ détecte que la trajectoire sans contrôle est localement optimale et y demeure donc. La méthode $\mathbb{M}_{\mathtt{DFO}}$ part de $\mathbb{M}_0$, et elle pourrait théoriquement altérer convenablement les contrôles instantanés pour détecter un gain résultat d'un changement de plateaux. Ceci est toutefois improbable, aussi la méthode $\mathbb{M}_{\mathtt{DFO}}$ stagne également à la trajectoire sans contrôle. Enfin, nous lançons deux instances de la méthode $\mathbb{M}_{\mathtt{PO}}$. La variante notée $\mathbb{M}_{\mathtt{PO}}^{\{T/2,T\}}$ suit le schéma numérique proposé au chapitre 7.2.2. La variante $\mathbb{M}_{\mathtt{PO}}^{\{T\}}$ consiste en une altération de $\mathbb{M}_{\mathtt{PO}}^{\{T/2,T\}}$ dans laquelle l'état de la trajectoire à l'instant $\frac{T}{2}$ n'est pas une variable de la reformulation mais est laissé comme variable du sous-problème. Dans cette variante, c'est au sous-problème de gérer les plateaux associés à la cible A. Seule la variante $\mathbb{M}_{\mathtt{PO}}^{\{T/2,T\}}$ est satisfaisante. La méthode $\mathbb{M}_{\mathtt{PO}}^{\{T\}}$ n'optimise que l'état final de la trajectoire via un algorithme de `DFO`, et ainsi l'algorithme d'optimisation lisse calculant une trajectoire atteignant cet état final ne tient pas compte de la cible marquée par A. On observe ici l'intérêt d'optimiser tous les états impactant les discontinuités du coût de Mayer par un algorithme `DFO` : cette approche est la seule à laquelle on puisse se fier.

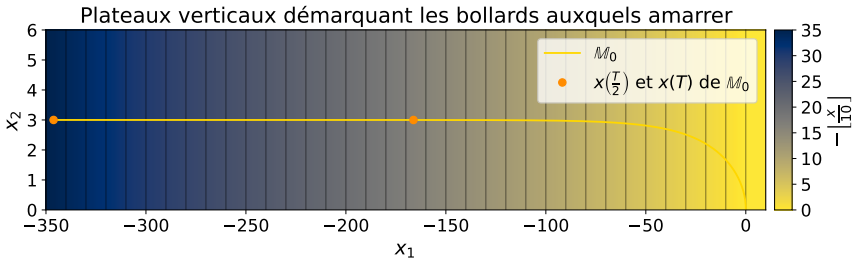
7.3.3 Solution numérique d'un problème de Zermelo à plateaux verticaux

Ce chapitre considère un second problème de Zermelo, proche du problème ($\mathbf{P}_{\mathtt{OC,Zt}}$). Nous étudions ici un problème modélisant une recherche de stationnement optimal après traversée de la rivière, modélisé par

$$
\begin{array}{lll}
\underset{\substack{x \in \mathcal{AC}([0,T],\mathbb{R}^2) \\ (u,\theta) \in \mathcal{L}^\infty([0,T],\mathbb{R}^2)}}{\text{minimiser}} & -\left\lfloor \frac{x_1(T)}{w} \right\rfloor + \dfrac{1}{T}\displaystyle\int_0^T u(t)\,dt & \\
\text{sous contraintes} & x(0) = (0,0), & \\
& \dot{x}(t) = s(x(t)) + u(t)\mathbb{1}_{\theta(t)}, & \text{p.p.t. } t \in [0,T], \\
& 0 \leq x_2(t) \leq h, & \forall t \in [0,T], \\
& 0 \leq u(t) \leq 1, & \text{p.p.t. } t \in [0,T], \\
& x_2(T) = h. &
\end{array}
\qquad (\mathbf{P}_{\mathtt{OC,Zp}})
$$

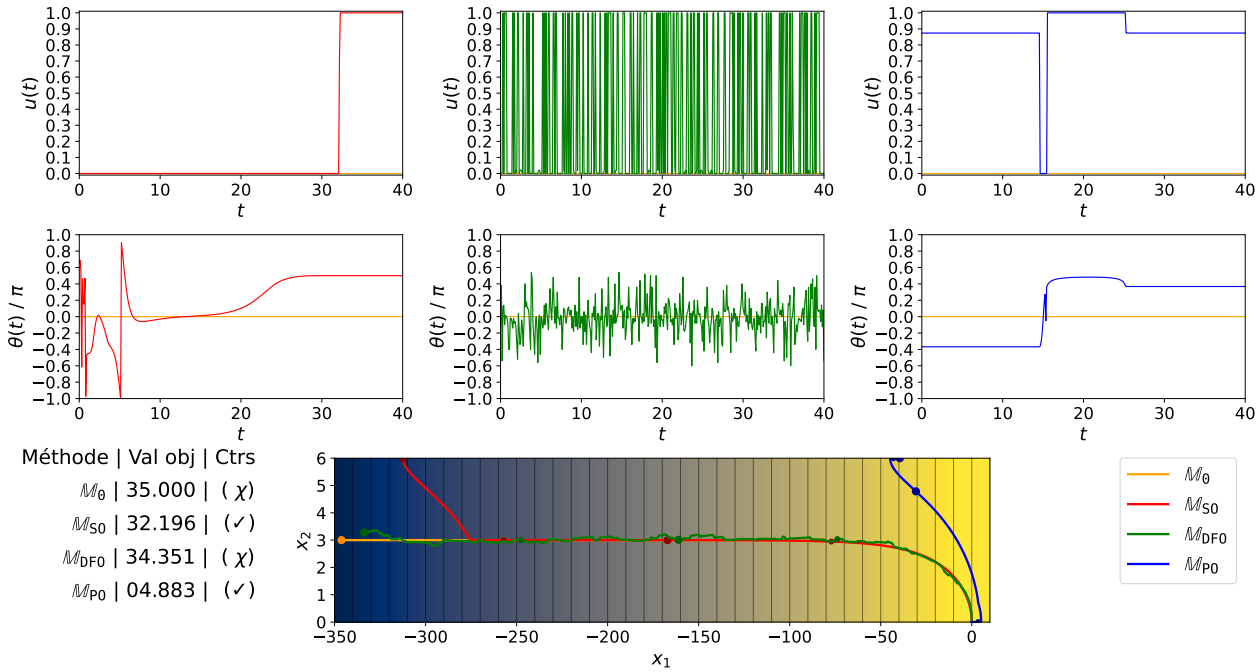
Les paramètres restent définis comme au problème ($\mathbf{P}_{\mathtt{OC,Zt}}$), avec en plus $w \triangleq 10$. Ce problème modélise un navire partant de la rive sud d'une rivière dont le courant pousse vers l'ouest, et cherchant à amarrer à un bollard le plus à l'est possible dans un port situé sur la rive nord, à consommation minimale. La figure 7.7 illustre les plateaux résultants du coût de Mayer.

FIGURE 7.7 Qualité mesurée des bollards sur la rive nord de la rivière.



Nous remarquons ici que seule l'abscisse finale $x_1(T)$ intervient dans la discontinuité du coût de Mayer, alors que l'ordonnée finale $x_2(T)$ est fixée. Par conséquent, nous pouvons partitionner l'espace des variables sur la valeur de l'abscisse de la trajectoire à l'instant final uniquement. Comme dans le problème de Zermelo précédent, le (PMP) ne permet pas de résoudre le problème, et le sous-problème résultant d'un état final fixé non plus. Par conséquent, nous appliquons encore les processus de résolution numérique proposés au chapitre 7.2.2. Les résultats de chacune des méthodes sont donnés dans la figure 7.8.

FIGURE 7.8 Résultats numériques sur le problème de plateaux de Zermelo.



La trajectoire sans contrôle, $\mathbb{M}_0$, ne respecte pas la contrainte d'ordonnée finale. Ainsi, toutes les méthodes sont cette fois initialisées depuis une trajectoire inacceptable. La méthode $\mathbb{M}_{\mathtt{SO}}$ parvient à garantir la faisabilité. On peut remarquer que durant ce processus, elle parvient à franchir quelques plateaux, mais qu'elle semble converger dans un plateau de

mauvaise qualité dès que la faisabilité est garantie. Ceci peut s'expliquer par le fait que le coût de Mayer est constant dans chaque plateau, alors que le coût de Lagrange incite à ne pas lutter contre le courant pour tenter de progresser le long de la rive nord. La méthode $\mathbb{M}_{\texttt{DFO}}$ ne parvient pas à rétablir la réalisabilité. Ceci s'explique par le fait que pour trouver une trajectoire réalisable, il faut que les contrôles travaillent de concert dans la bonne direction et avec une norme importante ; or un algorithme DFO travaille sur chaque contrôle indépendemment des autres. Enfin, la méthode $\mathbb{M}_{\texttt{PO}}$ est efficace et retourne une solution pertinente.

7.4 Pistes d'améliorations des processus de résolution

Les processus analytique et numérique discutés au chapitre 7.2 semblent efficaces au vu des tests du chapitre 7.3. Malgré tout, ces deux processus peuvent être modifiés sur de nombreux aspects pour améliorer encore leur performance. Précisons que toutes les pistes proposées ici brisent les garanties théoriques de convergence, puisque le cadre de l'optimisation partitionnée (le cadre d'étude 5.1) n'est plus respecté. Toutefois, ceci n'est pas problématique dans le processus numérique altéré, car le processus numérique initial 7.2.2 est déjà heuristique. La perte de garanties est plus pénible pour le processus analytique ; mais au moment de la rédaction de cette thèse, nous avons des idées pour dériver une nouvelle analyse de convergence du processus théorique altéré (qui n'est pas discutée dans ce manuscrit car encore inachevée).

Le chapitre 7.4.1 discute de moyens d'accélérer la résolution du sous-problème $(\mathbf{P}_{\texttt{OC}}^{\texttt{sub}}(\mathfrak{X}))$ pour tout $\mathfrak{X} \in \mathbb{X}$ donné. Le chapitre 7.4.2 propose une relaxation de $(\mathbf{P}_{\texttt{OC}}^{\texttt{sub}}(\mathfrak{X}))$ pour affaiblir la contrainte $(x(t))_{t\in\xi} = \mathfrak{X}$ imposant au système de passer exactement par les états proposés par l'algorithme DFO résolvant la reformulation $(\mathbf{P}_{\texttt{OC}}^{\texttt{ref}})$. En effet, cette contrainte peut être source de nombreuses difficultés numériques que nous discutons.

7.4.1 Étude de sensibilité du sous-problème

Une adaptation de la méthode qui n'altère pas ses propriétés théoriques consiste à exploiter la structure du sous-problème $(\mathbf{P}_{\texttt{OC}}^{\texttt{sub}}(\mathfrak{X}))$ pour obtenir rapidement sa solution globale $\gamma(\mathfrak{X})$ pour tout $\mathfrak{X} \in X$. Admettons qu'on dispose d'un point $\mathfrak{X} \in X$ en lequel on a réussi à calculer $\gamma(\mathfrak{X})$, et qu'on cherche à résoudre $(\mathbf{P}_{\texttt{OC}}^{\texttt{sub}}(\mathfrak{X} + \Delta\mathfrak{X}))$ pour $\Delta\mathfrak{X}$ suffisamment petit.

Si le sous-problème est résolu algorithmiquement, on peut initialiser l'algorithme résolvant $(\mathbf{P}_{\texttt{OC}}^{\texttt{sub}}(\mathfrak{X} + \Delta\mathfrak{X}))$ au point $\gamma(\mathfrak{X})$ plutôt qu'à la trajectoire $\mathbb{M}_0$. Il est vraisemblable que ce changement d'initialisation apporte des gains de vitesse la plupart du temps.

On peut également estimer la valeur de $\gamma(\mathfrak{X} + \Delta\mathfrak{X})$. Ceci repose sur le théorème des fonctions implicites, usuel ou généralisé au cas non-lisse [98], ainsi que sur une étude de

sensibilité du sous-problème ($\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X})$) vis-à-vis de sa contrainte $(x(t))_{t\in\xi} = \mathfrak{X}$. Pour simplifier la discussion, considérons que $\xi \triangleq \{0, T\}$ et ainsi que $\mathfrak{X}$ consiste en un état initial et final imposé. Ceci ne restreint pas la généralité de notre raisonnement, puisque pour tout $\mathfrak{X} \in X$, le sous-problème ($\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X})$) est séparable en problèmes tiers se concentrant sur chaque intervalle de temps entre deux instants $t \in \zeta$. Dans ce cas, on sait que $\gamma(\mathfrak{X}) \triangleq (x^*, u^*)$ est une extrémale du (PMP), au sens où

$$\begin{cases} x^*(0) = \mathfrak{X}^0, \quad x^*(T) = \mathfrak{X}^T, \quad \dot{x}^*(t) = f(t, x^*(t), u^*(t)), \\ p^*(0) \text{ libre}, \quad p^*(T) \text{ libre}, \quad \dot{p}^*(t) = p^*(t) f_x(t, x^*(t), u^*(t)) + p_0^* \ell_x(t, x^*(t), u^*(t)), \\ \text{p.p.t. } t \in [0, T], \ u^*(t) \in \operatorname{argmax}_u p^*(t) f(t, x^*(t), u) + p_0^* \ell(t, x^*(t), u). \end{cases}$$

On peut alors exprimer $\gamma(\mathfrak{X}+\Delta\mathfrak{X})$ comme une extrémale du (PMP) associé à ($\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X} + \delta\mathfrak{X})$), c'est-à-dire qu'il vérifie

$$\begin{cases} x(0) = \mathfrak{X}^0 + \Delta\mathfrak{X}^0, \quad x(T) = \mathfrak{X}^T + \Delta\mathfrak{X}^T, \quad \dot{x}(t) = f(t, x(t), u(t)), \\ p(0) \text{ libre}, \quad p(T) \text{ libre}, \quad \dot{p}(t) = p(t) f_x(t, x(t), u(t)) + p_0 \ell_x(t, x(t), u(t)), \\ \text{p.p.t. } t \in [0, T], \ u(t) \in \operatorname{argmax}_u p(t) f(t, x(t), u) + p_0 \ell(t, x(t), u). \end{cases}$$

On pose alors $x \triangleq x^* + \delta x$, et $u \triangleq u^* + \delta u$, et $p \triangleq p^* + \delta p$. Si les fonctions f et ℓ sont assez lisses, on peut alors les linéariser le long de (x^*, u^*, p^*) pour obtenir un système différentiel en $(\delta x,\ \delta u,\ \delta p)$. Ce système peut potentiellement apporter de l'information. Toutefois, cette piste de recherche soulève de nombreuses questions et difficultés théoriques, aussi nous n'avons pas plus d'éléments à apporter au moment de la rédaction de cette thèse. La littérature en contrôle optimal manque de références discutant de la sensibilité du (PMP) vis-à-vis des contraintes d'états initial et final. On peut également penser aux systèmes chaotiques [23] pour percevoir les limitations de cette approche.

7.4.2 Relaxation de la contrainte spatiale sur les états fixés

Pour n'importe quel $\mathfrak{X} \in X$, le sous-problème ($\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X})$) est très fortement impacté par sa contrainte $(x(t))_{t\in\zeta} = \mathfrak{X}$. En effet, si cette contrainte est irréalisable, le comportement d'un algorithme cherchant à résoudre le problème ($\mathbf{P}_{\mathtt{dOC}}^{\mathtt{sub}}(\mathfrak{X})$) devient difficile à prévoir. Prenons l'exemple d'un algorithme qui cherche à minimiser l'infaisabilité totale. Il est possible qu'il converge vers une solution irréalisable (x, u) telle que toutes les contraintes sont vérifiées, sauf $\|(x(t))_{t\in\zeta} - \mathfrak{X}\| \gg 0$; mais à l'inverse, il est aussi possible que (x, u) vérifie $(x(t))_{t\in\zeta} = \mathfrak{X}$ mais néglige fortement les autres contraintes (notamment la plus importante : le respect de la dynamique). Ce comportement erratique est dommageable. Pour l'atténuer, nous proposons de passer la contrainte $(x(t))_{t\in\zeta} = \mathfrak{X}$ comme une pénalisation dans l'objectif. En d'autres

termes, nous proposons d'altérer le sous-problème ($\mathbf{P}_{\mathtt{OC}}^{\mathtt{sub}}(\mathfrak{X})$) comme

$$\begin{aligned}
\underset{\substack{x \in \mathcal{AC}([0,T],\mathbb{R}^n) \\ u \in \mathcal{L}^\infty([0,T],\mathbb{R}^m)}}{\text{minimiser}} \quad & g(\mathfrak{X}) + \int_0^T \ell(t, x(t), u(t))\ dt + M \left\| x(t)_{t\in\zeta} - \mathfrak{X} \right\|^2 \\
\text{sous contraintes} \quad & \dot{x}(t) = f(t, x(t), u(t)), \quad \text{p.p.t. } t \in [0,T], \\
& c(t, x(t), u(t)) \leq 0, \qquad \text{p.p.t. } t \in [0,T], \\
& c_\zeta(\mathfrak{X}) \leq 0,
\end{aligned}$$

où $M \in \mathbb{R}_+$ est un facteur de pénalisation, qui doit être assez grand. Une autre approche serait de conserver la contrainte $(x(t))_{t\in\zeta} = \mathfrak{X}$, mais de pouvoir agencer lexicographiquement les contraintes de sorte à prioriser le respect de la dynamique.

CHAPITRE 8 CONCLUSION

Une Hirondelle, en ses voyages, avait beaucoup appris.
Quiconque a beaucoup vu peut avoir beaucoup retenu.

« L'Hirondelle et les petits oiseaux », Jean de la Fontaine.

Pour clôturer ce manuscrit, mettons en valeur les principaux apports à la DFO proposés par cette thèse. Le chapitre 8.1 survole le contenu du manuscrit pour synthétiser ces apports. Le chapitre 8.2 identifie les freins à leur usage. Enfin, le chapitre 8.3 entrevoit des pistes pour pousser la théorie et améliorer les méthodes numériques construites à partir de ces travaux. Le tableau 8.1 proposé au chapitre 8.4 synthétise tout ce qui est dit ici.

Rappelons que la finalité de ce manuscrit est de proposer une stratégie de résolution du problème d'optimisation (**P**) dans le cadre d'étude 1.1, et d'assurer que cette stratégie est performante en pratique. En réponse à ceci, nos trois contributions principales sont l'algorithme cDSM, le cadre de l'optimisation partitionnée PO, et l'exemple d'application à une classe de problèmes de contrôle optimal permettant de surmonter des difficultés pratiques.

8.1 Synthèse des travaux

Notre première contribution principale est la détection et la correction d'une erreur dans la référence « Analysis of direct searches for discontinuous functions ». Le détail de cette contribution fait l'objet du chapitre 3. « Analysis of direct searches for discontinuous functions » visait à démontrer que les propriétés de DSM bien identifiées dans le cas continu (la convergence vers des points satisfaisant des conditions nécessaires d'optimalité) sont conservées dans le cas discontinu. Après application de notre correctif, nous disposons donc de l'algorithme rDSM, une altération minime de l'algorithme DSM usuel de la littérature, ayant dans le cas discontinu des propriétés similaires à DSM dans le cas continu (théorème 4.2). Cette contribution résorbe donc la distinction que la littérature en DFO fait entre le cas continu et le cas discontinu. Ceci requiert toutefois quelques hypothèses supplémentaires, que nous identifions clairement et discutons en détail.

Le second apport de cette thèse consiste à prouver que rDSM dispose en réalité de propriétés plus fortes que ce que « Analysis of direct searches for discontinuous functions » entrevoyait pour DSM. Nous montrons au chapitre 4 que sous les hypothèses du chapitre 3, rDSM est certifié de générer une solution locale au problème (**P**). Ensuite, nous ré-adaptons rDSM pour alléger au maximum les hypothèses requises par cette propriété. Au sortir du chapitre 4, nous

disposons donc de l'algorithme cDSM qui offre des garanties de générer des solutions locales au problème (**P**) (théorème 4.13). Cette garantie repose sur une unique hypothèse légère sur la fonction objectif du problème, dont nous prouvons qu'elle est la plus générale possible. Il est donc impossible d'envisager des propriétés de convergence plus fortes en toute généralité.

Enfin, le troisième travail effectué durant cette thèse se concentre sur la performance pratique de cDSM. Nous discutons de choix de conception pertinents pour cDSM en fin de chapitre 4. De plus, nous pouvons conjecturer que cDSM est au moins aussi performant que DSM. Aussi, nous nous concentrons sur des problèmes présentant une caractéristique bien identifiée par la littérature en DFO pour mettre les méthodes en échec : le nombre de variables dépasse le seuil critique de 50 [17, chapitre 1.4]. Nous identifions une famille d'instances du problème (**P**), de dimension potentiellement infinie, sur laquelle la structure de PO peut être exploitée pour rendre cDSM viable malgré la grande dimension. L'étude théorique complète de la PO fait l'objet du chapitre 5. La preuve empirique des gains importants qu'elle apporte sur des problèmes de la DFO à 100 variables est donnée au chapitre 6. Ce chapitre montre donc que cDSM permet de repousser certaines des limites de la DFO identifiées par la littérature, et il illustre en quoi cette performance subsiste avec un nombre de variables quelconque. De plus, la structure de PO est également présente dans des problèmes d'optimisation issus d'autres littératures très différentes de la DFO. Nous montrons au chapitre 7 que certains problèmes ouverts en contrôle optimal sont résolus efficacement par application de la PO. Ce chapitre crée donc un lien entre la DFO et le contrôle optimal, alors que ces deux domaines pourraient à priori être considérés comme antinomiques.

8.2 Limites théoriques et pratiques

Ce chapitre se concentre sur les limitations associées à l'algorithme cDSM, au cadre de la PO, et aux difficultés résultants de l'application de la PO à une classe de problèmes donnée. Pour chacun, nous identifions les défauts théoriques et les limitations pratiques d'utilisation.

L'algorithme cDSM requiert quelques hypothèses sur le problème (**P**) pour avoir des garanties de convergence vers une solution locale. L'aspect uniquement local de la solution renvoyée est commun à toutes les méthodes de la DFO. En effet, sauf quelques cas extrêmement spécifiques, il est impossible de garantir qu'une solution donnée est globale. Toutefois, les autres hypothèses sont légères, et potentiellement les plus légères possible. Ensuite, cDSM conserve certains défauts théoriques de DSM, dont il s'inspire. Notamment, sa performance sur un problème de grande dimension (plus de 50 variables) n'est pas garantie en pratique. De plus, il est restreint à reposer soit sur une structure de DSM à treillis adaptatif, soit sur une structure de DSM à décroissance suffisante. Chacune de ces deux structures a des défauts

propres. Enfin, si un problème pratique est ce que la DFO nomme une boîte grise, l'information accessible n'est ni prise en compte ni détectée par cDSM. Il est donc nécessaire d'adapter cDSM au cas par cas, via la conception d'une étape de search ou de covering dédiée, pour tenir compte de cette information.

Le cadre de l'optimisation partitionnée PO, décrit par le cadre 5.1, n'est pas universel. Il existe des problèmes d'optimisation n'entrant pas dans le cadre de la PO. Ainsi, la PO permet à la DFO de gérer de nouvelles tailles de problèmes, mais tous les problèmes d'une taille donnée ne deviennent pas solubles. Ensuite, le cadre actuel de la PO dépend de la fonction oracle γ, dont la construction est très stricte et dont le calcul peut être difficile. De plus, les garanties de convergence de cDSM via la PO reposent sur l'hypothèse 5.2 portant sur γ, qui est riche et possiblement difficile à vérifier en pratique. Nos pistes d'allègements de la définition de γ et des pré-requis clamés par l'hypothèse 5.2 ne sont pas encore achevées ; donc au moment de la rédaction de ce manuscrit, la fonction γ est un gros frein à l'utilisation pratique de la PO. Enfin, en pratique, la PO n'est exploitable que si le coût de calcul de γ n'est pas trop important comparé à celui de la fonction objectif du problème. De plus, il est difficile de proposer un ratio limite entre ces deux quantités (au-delà duquel la PO perd en pertinence), car une telle valeur semble dépendre du problème.

Lorsqu'on applique la PO à une classe de problèmes donnée, d'autres difficultés peuvent apparaître. La principale réside dans la façon d'appliquer la PO. Il faut pour cela bien définir la partition de l'espace des variables qu'on considère. Il est possiblement simple de trouver un partitionnement admissible, mais nos exemples issus du contrôle optimal montrent que le choix le plus évident n'est pas toujours le plus pertinent. Ainsi, il y a un potentiel manque à gagner lorsqu'on choisit un partitionnement pas assez spécifique au problème considéré. Enfin, en pratique, il est nécessaire d'utiliser des méthodes d'optimisation dédiées pour résoudre le sous-problème défini par la PO. Le choix de la méthode est un questionnement propre à la classe de problèmes considérée, et il ne peut donc pas être dicté par des règles générales. De plus, il peut arriver que le solveur choisi pour résoudre le sous-problème a des spécifications techniques. Dans un tel cas, on doit potentiellement altérer le sous-problème (comme, par exemple, en le lissant) pour lui faire passer les spécifications du solveur. Ceci requiert des efforts supplémentaires, et brise les garanties théoriques de convergence de la PO. Ainsi, en pratique, l'utilisation de la PO est possiblement heuristique.

8.3 Extensions possibles

Nous clôturons ce manuscrit en énumérant des perspectives de travaux futurs.

Algorithme cDSM

L'algorithme cDSM pourrait être simplifié à l'extrême en une variante ne conservant que l'étape de covering à chaque itération, sans treillis adaptatif ni décroissance suffisante. Ceci fournirait une référence pour quantifier l'apport réel de chacune des étapes de search, covering et poll à différents avancements dans le processus d'optimisation. Une telle étude numérique pourrait également illustrer à quel point le cadre général de l'étude de cDSM (cadre 4.2) permet une bonne performance pratique. Par exemple, même si la théorie ne requiert que le caractère ample des ensembles de continuité de la fonction objectif, il ne serait pas surprenant que la propriété de cône intérieur mène à une meilleure performance numérique. Il y a donc plusieurs considérations numériques liées à cDSM qu'il serait intéressant de développer. De plus, une analyse numérique pourrait confirmer ou infirmer la qualité des schémas d'oracles de covering que nous proposons au chapitre 4.

Puisque l'étape de covering est autosuffisante, nous pouvons également envisager de l'intégrer à d'autres méthodes de la DFO. N'importe quelle méthode peut virtuellement obtenir des garanties de générer des solutions locales au problème (**P**) après ajout d'une étape de covering, sous les mêmes hypothèses très larges que dans le cadre d'étude 4.2. Nous pouvons notamment penser à des algorithmes bayésiens par exemple, ainsi qu'à toutes les méthodes listées dans la revue de littérature [73].

Il est également envisageable d'enrichir cDSM de nouvelles mécaniques. Par exemple, en intégrant une étape de covering à l'étape de sonde autour du point irréalisable dans la stratégie de *barrière progressive* [15]. Ceci pourrait donner de la flexibilité à l'algorithme sur des problèmes présentant des contraintes quantifiables [74] continues. Il est possible que cette stratégie puisse même traiter des contraintes semi-continues ; mais l'adaptation serait plus complexe à mettre en oeuvre. Un autre exemple serait la mécanique de DiscoMads [8] visant à éviter de renvoyer à l'utilisateur des points trop proches des discontinuités du problème. Une telle stratégie sacrifie la demande de renvoyer une solution locale pour, en contrepartie, garantir une certaine robustesse du point renvoyé par l'algorithme. DiscoMads a le défaut de parfois approcher une discontinuité sans la détecter, mais cDSM évite cet écueil par construction. De fait, cDSM enrichi d'une mécanique de *distanciation aux discontinuités* pourrait être très intéressant d'un point de vue pratique. En considérant l'incertitude entourant toute application pratique, il est bien plus pertinent de considérer une pseudo-solution lointaine des discontinuités qu'une solution proche de celles-ci.

Cadre de l'optimisation partitionnée

Des extensions au cadre de la PO ont déjà été proposées au chapitre 5. Nous redonnons les grandes lignes ici, mais nous référons au chapitre 5.3 pour plus de détails.

Une extension du cadre de la PO, pour l'alléger sans l'altérer en profondeur, consiste en une définition alternative de la fonction γ. Actuellement, pour tout $x \in X$ décidé par l'algorithme résolvant la reformulation $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{ref}})$, $\gamma(x)$ est défini comme l'unique solution du sous-problème $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x))$. Or, en pratique, on résout $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x))$ via un algorithme d'optimisation, dont on interrompt l'exécution en temps fini. Ainsi, le point renvoyé par l'algorithme n'est qu'une approximation d'une solution locale de $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x))$. Nous souhaitons donc alléger le cadre en définissant $\gamma(x)$ pour tout $x \in X$ comme une approximation d'une solution locale de $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x))$. Une telle définition est technique et complexifie l'analyse, mais elle rend le cadre de la PO représentatif du comportement réel des méthodes que nous employons en pratique.

On pourrait aussi adapter le cadre de la PO en proposant un analogue au théorème 5.4 qui dépendrait d'un algorithme non enrichi par une étape de `covering`. En effet, une telle méthode peut être plus rapide, au prix de propriétés de convergence plus faibles. Il faudrait ajouter de nouvelles hypothèses au cadre de la PO pour garantir que les propriétés de la méthode résolvant la reformulation $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{ref}})$ se transmettent au problème initial $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{ini}})$.

L'autre extension forte de la PO mènerait à une altération plus profonde du cadre. Il s'agit du développement de *méthodes hybrides* optimisant en même temps $x \in \mathbb{X}$ et un point $y \in \mathbb{Y}$ contraint à toujours être dans $\mathbb{Y}(x)$. Une telle méthode est dite *hybride* car x est optimisé via un algorithme de DFO, et en même temps y est optimisé via un algorithme adapté à la structure du sous-problème $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x))$. Ceci met en synergie la DFO avec d'autres littératures, en laissant chacune traiter une partie du problème pour laquelle elle est pensée. Nous croyons fortement au potentiel de ces méthodes ; mais leur analyse et leur développement sont tout juste commencés au moment de la rédaction de ce manuscrit.

Application de la PO à des classes de problèmes données

En appliquant la PO à un problème donné, on s'assure que le sous-problème $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x))$ devient simple. Ainsi, on a potentiellement des informations sur des liens entre les solutions de $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x))$ et celles de $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x+\delta x))$ pour x donné et δx petit. Par exemple, une analyse de sensibilité de $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{sub}}(x))$ vis-à-vis de x peut prédire l'évolution de la fonction γ au voisinage de x. Il y a donc un gain de temps potentiel très important. Toutefois, cette analyse est vraisemblablement à mener au cas par cas, selon la structure de l'instance de $(\mathbf{P}_{\mathtt{PO}}^{\mathtt{ini}})$ et du partitionnement retenu.

8.4 Tableau récapitulatif des contributions et des extensions possibles

TABLEAU 8.1 Synthèse des travaux discutés et extensions possibles.

Stratégie.	Usage sur le problème (**P**). Limites.	Extensions.
Algos DFO hors DSM.	Renvoie x^* vérifiant des conditions nécessaires d'optimalité, si f continue. Max 50 variables.	Ajout d'une étape de covering.
DSM.	Renvoie x^* vérifiant des conditions nécessaires d'optimalité, si f continue. Max 50 variables.	cDSM.
rDSM.	Préférer cDSM, qui le renforce.	cDSM.
cDSM.	Renvoie x^* solution locale, si ensembles de continuité de f amples et x^0 réalisable. Néglige la structure de f. Max 50 variables.	Ajout de barrière progressive. Distanciation aux discontinuités. Quantifier l'apport pratique de la search, la covering et la poll. Oracles de covering bien conçus.
PO.	Problème simplifiable par restriction de l'espace des variables. Difficulté transposée au choix de la bonne restriction à prendre. Permet de trouver une solution même en grande dimension. Construction mathématique actuelle rigide.	Assouplissement de la construction. Allègement des hypothèses requises pour résoudre algorithmiquement. Analyse de sensibilité du sous-problème. Garanties théoriques pour méthodes autres que cDSM. Méthodes hybrides avancées.
PO par classe de problèmes.	Résout par DFO un problème normalement inaccessible à celle-ci. Permet une méthode hybride entre DFO et littérature dédiée. Méthode non restrictive, ni sur la partie DFO ni sur l'autre partie. Difficultés potentielles d'implémentation. Peut donc devenir heuristique en pratique.	Utilisation d'information propre à la classe de problèmes. Spécification poussée à la classe de problèmes. Règles de choix de partitionnement. Peut amener d'autres idées, quitte à s'éloigner du cadre.

Pour tirer le meilleur parti des connaissances acquises, pour en extraire toute la richesse, il importe de ne pas s'y habituer trop vite, de se laisser le temps de la surprise et de l'étonnement.

« L'Espace prend la forme de mon regard », Hubert Reeves.

RÉFÉRENCES

[1] M. A. Abramson, C. Audet, J. E. Dennis, Jr. et S. Le Digabel : OrthoMADS : A Deterministic MADS Instance with Orthogonal Directions. *SIAM Journal on Optimization*, 20(2):948–966, 2009. URL https://dx.doi.org/10.1137/080716980.

[2] S. P. Adam, S.-N. N. Alexandropoulos, P. M. Pardalos et M. N. Vrahatis : *No Free Lunch Theorem : A Review*, p. 57–82. Springer International Publishing, Cham, 2019. URL https://doi.org/10.1007/978-3-030-12767-1_5.

[3] S. Alarie, C. Audet, P.-Y. Bouchet et S. Le Digabel : Optimisation of stochastic blackboxes with adaptive precision. *SIAM Journal on Optimization*, 31(4):3127–3156, 2021. URL https://dx.doi.org/10.1137/20M1318894.

[4] S. Alarie, C. Audet, A. E. Gheribi, M. Kokkolaras et S. Le Digabel : Two decades of blackbox optimization applications. *EURO Journal on Computational Optimization*, 9, 2021. URL https://doi.org/10.1016/j.ejco.2021.100011.

[5] A. Y. Aravkin, R. Baraldi et D. Orban : A Proximal Quasi-Newton Trust-Region Method for Nonsmooth Regularized Optimization. *SIAM Journal on Optimization*, 32 (2):900–929, 2022. URL https://doi.org/10.1137/21M1409536.

[6] R. Astudillo et P. I. Frazier : Thinking inside the box : A tutorial on grey-box bayesian optimization. *In 2021 Winter Simulation Conference (WSC)*, p. 1–15, 2021. URL https://doi.org/10.1109/WSC52266.2021.971534.

[7] C. Audet : Tuning Runge-Kutta parameters on a family of ordinary differential equations. *International Journal of Mathematical Modelling and Numerical Optimisation*, 8(3):277–286, 2018. URL https://doi.org/10.1504/IJMMNO.2018.088992.

[8] C. Audet, A. Batailly et S. Kojtych : Escaping unknown discontinuous regions in blackbox optimization. *SIAM Journal on Optimization*, 32(3):1843–1870, 2022. URL https://doi.org/10.1137/21M1420915.

[9] C. Audet, V. Béchard et S. Le Digabel : Nonsmooth optimization through Mesh Adaptive Direct Search and Variable Neighborhood Search. *Journal of Global Optimization*, 41(2):299–318, 2008. URL https://dx.doi.org/10.1007/s10898-007-9234-1.

[10] C. Audet, P.-Y. Bouchet et L. Bourdin : A derivative-free approach to optimal control problems with a piecewise constant Mayer cost function. Les Cahiers du GERAD G-2021-63, Groupe d'études et de recherche en analyse des décisions, GERAD, Montréal QC H3T 2A7, Canada, 2021. URL https://www.gerad.ca/en/papers/G-2021-63.

[11] C. Audet, P.-Y. Bouchet et L. Bourdin : Erratum, counterexample and an additional revealing poll step for a result of "Analysis of direct searches for discontinuous functions". Les Cahiers du GERAD G-2022-56, Groupe d'études et de recherche en analyse des décisions, GERAD, Montréal QC H3T 2A7, Canada, 2022. URL https://www.gerad.ca/fr/papers/G-2022-56.

[12] C. Audet, A. R. Conn, S. Le Digabel et M. Peyrega : A progressive barrier derivative-free trust-region algorithm for constrained optimization. *Computational Optimization and Applications*, 71:307–329, 2018. URL https://doi.org/10.1007/s10589-018-0020-4.

[13] C. Audet et J. E. Dennis, Jr. : Analysis of generalized pattern searches. *SIAM Journal on Optimization*, 13(3):889–903, 2003. URL https://dx.doi.org/Doi:10.1137/S1052623400378742.

[14] C. Audet et J. E. Dennis, Jr. : Mesh Adaptive Direct Search Algorithms for Constrained Optimization. *SIAM Journal on Optimization*, 17(1):188–217, 2006. URL https://dx.doi.org/Doi:10.1137/040603371.

[15] C. Audet et J. E. Dennis, Jr. : A Progressive Barrier for Derivative-Free Nonlinear Programming. *SIAM Journal on Optimization*, 20(1):445–472, 2009. URL https://dx.doi.org/10.1137/070692662.

[16] C. Audet, J. E. Dennis, Jr. et S. Le Digabel : Globalization strategies for Mesh Adaptive Direct Search. *Computational Optimization and Applications*, 46(2):193–215, 2010. URL https://dx.doi.org/10.1007/s10589-009-9266-1.

[17] C. Audet et W. Hare : *Derivative-Free and Blackbox Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, Cham, Switzerland, 2017. URL https://dx.doi.org/10.1007/978-3-319-68913-5.

[18] C. Audet, A. Ihaddadene, S. Le Digabel et C. Tribes : Robust optimization of noisy blackbox problems using the Mesh Adaptive Direct Search algorithm. *Optimization Letters*, 12(4):675–689, 2018. URL https://dx.doi.org/10.1007/s11590-017-1226-6.

[19] C. Audet, S. Le Digabel, V. Rochon Montplaisir et C. Tribes : Algorithm 1027 : NOMAD version 4 : Nonlinear optimization with the MADS algorithm. *ACM Transactions on Mathematical Software*, 48(3):35 :1–35 :22, 2022. URL https://dx.doi.org/10.1145/3544489.

[20] C. Audet et C. Tribes : Mesh-based Nelder-Mead algorithm for inequality constrained optimization. *Computational Optimization and Applications*, 71(2):331–352, 2018. URL https://link.springer.com/article/10.1007/s10589-018-0016-0.

[21] T. Bäck : *Evolutionary Algorithms in Theory and Practice : Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, New York, NY, USA, 1996. URL https://dl.acm.org/doi/book/10.5555/229867.

[22] A. M. Bagirov, M. Gaudioso, N. Karmitsa, M. M. Mäkelä et S. Taheri : *Numerical Nonsmooth Optimization : State of the Art Algorithms*. Springer International Publishing, 2020. URL https://doi.org/10.1007/978-3-030-34910-3.

[23] G. L. Baker, J. P. Gollub, R. et Fox : Chaotic dynamics : An introduction. *Physics Today*, 43(7):67–67, 07 1990. ISSN 0031-9228. URL https://doi.org/10.1063/1.2810630.

[24] B. Bank, J. Guddat, D. Klatte, B. Kummer et K. Tammer : *Nonlinear Parametric Optimization*. Birkhäuser Basel, 1983. URL https://doi.org/10.1137/1026121.

[25] T. Bayen et L. Pfeiffer : Second order conditions for a control problem with discontinuous cost. *2019 IEEE 58th Conference on Decision and Control (CDC)*, p. 1480–1485, 2019. URL https://doi.org/10.1109/CDC40024.2019.9029314.

[26] A. Berahas, L. Cao et K. Scheinberg : Global convergence rate analysis of a generic line search algorithm with noise. *SIAM Journal on Optimization*, 31(2):1489–1518, 2021. URL https://doi.org/10.1137/19M1291832.

[27] N. Boumal : *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023. URL https://www.nicolasboumal.net/book/.

[28] L. Bourdin : Note on pontryagin maximum principle with running state constraints and smooth dynamics – proof based on the ekeland variational principle, 2016. URL https://arxiv.org/abs/1604.04051.

[29] E. Brochu, V. M. Cora et N. de Freitas : A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning, 2010. URL https://arxiv.org/abs/1012.2599.

[30] A. E. Bryson et Y. C. Ho : *Applied Optimal Control : optimization, estimation and control*. Taylor & Francis, 1975. URL https://doi.org/10.1201/9781315137667.

[31] C. Cartis et L. Roberts : Scalable subspace methods for derivative-free nonlinear least-square optimization. *Mathematical Programming*, 199:461–524, 2023. URL https://doi.org/10.1007/s10107-022-01836-1.

[32] F. H. Clarke : *Optimization and Nonsmooth Analysis*. SIAM, 1990. URL https://doi.org/10.1137/1.9781611971309.

[33] F. H. Clarke : *Functional Analysis, Calculus of Variations and Optimal Control*. Springer, 2013. URL https://doi.org/10.1007/978-1-4471-4820-3.

[34] A. R. Conn, N. I. M. Gould et P. L. Toint : *Trust region methods*. SIAM, 2000. URL https://dx.doi.org/10.1137/1.9780898719857.

[35] A. R. Conn, K. Scheinberg et L. N. Vicente : *Introduction to Derivative-Free Optimization*. MOS-SIAM Series on Optimization. SIAM, 2009. URL https://dx.doi.org/10.1137/1.9780898718768.

[36] S. Dempe : Bilevel optimization : theory, algorithms, applications and a bibliography. *Bilevel Optimization : Advances and Next Challenges*, p. 581–672, 2020. URL https://doi.org/10.1007/978-3-030-52119-6_20.

[37] Y. Diouane, S. Gratton et L. N. Vicente : Globally convergent evolution strategies for constrained optimization. *Computational Optimization and Applications*, 62(2):323–346, 2015. URL https://dx.doi.org/10.1007/s10589-015-9747-3.

[38] Y. Diouane, V. Kungurtsev, F. Rinaldi et D. Zeffiro : Inexact direct-search methods for bilevel optimization problems, 2023. URL https://arxiv.org/abs/2307.09924.

[39] Y. Diouane, V. Picheny, R. Le Riche et A. Scotto Di Perrotolo : TREGO : a trust-region framework for efficient global optimization. *Journal of Global Optimization*, p. 1–23, 2022. URL https://dx.doi.org/10.1007/s10898-022-01245-w.

[40] A. L. Dontchev, W. M. Hager et V. M. Veliov : Second-order Runge-Kutta approximations in control constrained optimal control. *SIAM journal on numerical analysis*, 38(1):202–226, 2000. URL https://doi.org/10.1137/S0036142999351765.

[41] J. Dreo, A. Petrowski, P. Siarry et E. Taillard : *Métaheuristiques pour l'optimisation difficile*. Eyrolles, 2003. URL https://hal.science/hal-00843020.

[42] D. Dvinskikh, V. Tominin, I. Tominin et A. Gasnikov : Noisy zeroth-order optimization for non-smooth saddle point problems. *In Mathematical Optimization Theory and Operations Research*, p. 18–33, Cham, 2022. Springer International Publishing. URL https://doi.org/10.1007/978-3-031-09607-5_2.

[43] D. Eriksson, M. Pearce, J. Gardner, R. D. Turner et M. Poloczek : Scalable global optimization via local bayesian optimization. *Advances in Neural Information Processing Systems*, 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/6c990b7aca7bc7058f5e98ea909e924b-Abstract.html.

[44] I. Fajfar, J. Puhan et Á. Bűrmen : The Nelder-Mead simplex algorithm with perturbed centroid for high-dimensional function optimization. *Optimization Letters*, 13(5):1011–1025, 2019. URL https://dx.doi.org/10.1007/s11590-018-1306-2.

[45] E. A. Feinberg, P. O. Kasyanov et D. N. Kraemer : Continuity of parametric optima for possibly discontinuous functions and noncompact decision sets, 2021. URL https://arxiv.org/abs/2109.06299.

[46] E. Fermi et N. Metropolis : Numerical solution of a minimum problem. Los Alamos Unclassified Report LA–1492, Los Alamos National Laboratory, 1952. URL https://doi.org/10.2172/4377177.

[47] F. Fisch : *Development of a Framework for the Solution of High-Fidelity Trajectory Optimization Problems and Bilevel Optimal Control Problems*. Thèse de doctorat, Technische Universität München, 2011. URL https://mediatum.ub.tum.de/?id=1001868.

[48] R. Fletcher et S. Leyffer : Nonlinear programming without a penalty function. *Mathematical Programming*, Series A, 91:239–269, 2002. URL https://dx.doi.org/10.1007/s101070100244.

[49] H. Frankowska : The first order necessary conditions for nonsmooth variational and control problems. *SIAM Journal on Control and Optimization*, 22(1):1–12, 1984. URL https://doi.org/10.1137/0322001.

[50] P. I. Frazier : A tutorial on bayesian optimization, 2018. URL https://arxiv.org/abs/1807.02811.

[51] A. Galántai : Convergence of the Nelder-Mead method. *Numerical Algorithms*, 90 (3):1043–1072, 2022. URL https://doi.org/10.1007/s11075-021-01221-7.

[52] R. Garmanjani, D. Júdice et L. N. Vicente : Trust-region methods without using derivatives : Worst case complexity and the nonsmooth case. *SIAM Journal on Optimization*, 26(4):1987–2011, 2016. URL https://dx.doi.org/10.1137/151005683.

[53] R. Garnett : *Bayesian Optimization*. Cambridge University Press, 2023. URL https://bayesoptbook.com/.

[54] H. Ghanbari et K. Scheinberg : Black-box optimization in machine learning with trust region based derivative free algorithm, 2017. URL http://arxiv.org/abs/1703.06925.

[55] S. Gratton, C. W. Royer, L. N. Vicente et Z. Zhang : Direct search based on probabilistic descent. *SIAM Journal on Optimization*, 25(3):1515–1541, 2015. URL https://dx.doi.org/10.1137/140961602.

[56] S. Gratton, C. W. Royer, L. N. Vicente et Z. Zhang : Direct search based on probabilistic feasible descent for bound and linearly constrained problems. *Computational Optimization and Applications*, 72(3):525–559, 2019. URL https://doi.org/10.1007/s10589-019-00062-4.

[57] T. Haberkorn et E. Trélat : Convergence results for smooth regularizations of hybrid nonlinear optimal control problems. *SIAM Journal on Control and Optimization*, 49 (4):1498–1522, 2011. URL https://doi.org/10.1137/100809209.

[58] J. Halton : On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2(1):84–90, 1960. URL https://dx.doi.org/10.1007/BF01386213.

[59] P. Hansen et N. Mladenović : Variable neighborhood search : Principles and applications. *European Journal of Operational Research*, 130(3):449–467, 2001. URL https://dx.doi.org/10.1016/S0377-2217(00)00100-4.

[60] W. Hare, G. Jarry-Bolduc, S. Kerleau et C. W. Royer : Using orthogonally structured positive bases for constructing positive k-spanning sets with cosine measure guarantees, 2023. URL http://arxiv.org/abs/2306.06383.

[61] W. Hare, G. Jarry-Bolduc et C. Planiden : Hessian approximations, 2020. URL http://arxiv.org/abs/2011.02584.

[62] W. Hare, G. Jarry-Bolduc et C. Planiden : Nicely structured positive bases with maximal cosine measure. *Optimization Letters*, 17:1495–1515, 2023. URL https://doi.org/10.1007/s11590-023-01973-2.

[63] W. E. Hart : A generalized stationary point convergence theory for evolutionary algorithms. *In Proceedings of the International Conference on Genetic Algorithms*, p. 127–134. Morgan Kaufmann, 1997. URL https://www.osti.gov/biblio/486145.

[64] R. F. Hartl, P. Sethi et R. G. Vickson : A survey of the maximum principles for optimal control problems with state constraints. *SIAM Review*, 37(2):181–218, 1995. URL https://doi.org/10.1137/1037043.

[65] J. H. Holland : *Adaptation in Natural and Artificial Systems : An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press, Cambridge, 1992. URL https://doi.org/10.7551/mitpress/1090.001.0001.

[66] R. Horst, P. M. Pardalos et N. V. Thoai : *Introduction to Global Optimization*. Kluwer Academic, 2000. URL https://api.semanticscholar.org/CorpusID:62285710.

[67] J. Hu, X. Liu, Z. W. Wen et Y. X. Yuan : A brief introduction to manifold optimization. *Journal of the Operations Research Society of China*, 8:199–248, 2020. URL https://doi.org/10.1007/s40305-020-00295-9.

[68] D. R. Jones et J. R. R. A. Martins : The DIRECT algorithm : 25 years later. *Journal of Global Optimization*, 79:521–566, 2021. URL https://dx.doi.org/10.1007/s10898-020-00952-6.

[69] D. R. Jones, M. Schonlau et W. J. Welch : Efficient Global Optimization of Expensive Black Box Functions. *Journal of Global Optimization*, 13(4):455–492, 1998. URL https://dx.doi.org/10.1023/A:1008306431147.

[70] S. Katoch, S. S. Chauhan et V. Kumar : A review on genetic algorithm : past, present and future. *Multimedia Tools and Applications*, 80:8091–8126, 2021. URL https://doi.org/10.1007/s11042-020-10139-6.

[71] S. Kojtych : *Contributions à l'optimisation de systèmes mécaniques non réguliers : reconception d'aubes de compresseur*. Thèse de doctorat, Polytechnique Montréal, 2022. URL https://publications.polymtl.ca/10393/.

[72] D. G. Krige : A statistical approach to some mine valuations and allied problems at the Witwatersrand. Mémoire de D.E.A., University of Witwatersrand, 1951. URL https://api.semanticscholar.org/CorpusID:106850054.

[73] J. Larson, M. Menickelly et S. Wild : Derivative-free optimization methods. *Acta Numerica*, 28:287–404, 2019. URL https://dx.doi.org/10.1017/S0962492919000060.

[74] S. Le Digabel et S. Wild : A taxonomy of constraints in black-box simulation-based optimization. *Optimization and Engineering*, 2023. URL https://dx.doi.org/10.1007/s11081-023-09839-3.

[75] R. Le Riche, D. Gaudrie, V. Picheny, Y. Diouane, A. Spagnol et S. da Veiga : Which gaussian process for bayesian optimization ?, 2022. URL https://hal.science/hal-03679142.

[76] N. Levine : Semi-open sets and semi-continuity in topological spaces. *American Mathematical Monthly*, 70:36–41, 1963. URL https://dx.doi.org/10.1080/00029890.1963.11990039.

[77] G. Liuzzi, S. Lucidi et V. Piccialli : A partition-based global optimization algorithm. *Journal of Global Optimization*, 48:113–128, 2010. URL https://dx.doi.org/10.1007/s10898-009-9515-y.

[78] G. Liuzzi, S. Lucidi, F. Rinaldi et L. N. Vicente : Trust-region methods for the derivative-free optimization of nonsmooth black-box functions. *SIAM Journal on Optimization*, 29(4):3012–3035, 2019. URL https://doi.org/10.1137/19M125772X.

[79] M. Lubin, O. Dowson, J. Dias Garcia, J. Huchette, B. Legat et J. P. Vielma : JuMP 1.0 : Recent improvements to a modeling language for mathematical optimization. *Mathematical Programming Computation*, 2023. URL https://doi.org/10.1007/s12532-023-00239-3.

[80] K. I. M. McKinnon : Convergence of the Nelder-Mead simplex method to a nonstationary point. *SIAM Journal on Optimization*, 9(1):148–158, 1998. URL https://dx.doi.org/10.1137/S1052623496303482.

[81] N. Mladenović et P. Hansen : Variable neighborhood search. *Computers and Operations Research*, 24(11):1097–1100, 1997. URL https://dx.doi.org/10.1016/S0305-0548(97)00031-2.

[82] A. Montoison, P. Pascal et L. Salomon : NOMAD.jl : A Julia interface for the constrained blackbox solver NOMAD, 2020. URL https://github.com/bbopt/NOMAD.jl.

[83] B. Mordukhovich : *Bilevel Optimization and Variational Analysis*, p. 197–226. Springer, 2020. URL https://doi.org/10.1007/978-3-030-52119-6_7.

[84] J. J. Moré et D. C. Sorensen : Computing a trust region step. *SIAM Journal on Scientific Computing*, 4(3):553–572, 1983. URL https://dx.doi.org/10.1137/0904038.

[85] J. A. Nelder et R. Mead : A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965. URL http://comjnl.oxfordJournals.org/content/7/4/308.abstract.

[86] J. Nocedal et S. J. Wright : *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, second édn, 2006. URL http://www.springer.com/mathematics/book/978-0-387-30303-1.

[87] P. M. Pardalos et H. E. Romeijn : *Handbook of global optimization : Volume 2*, vol. 62. Springer Science & Business Media, 2013. URL https://doi.org/10.1007/978-1-4757-5362-2.

[88] F. M. Petersen, D. E. Cornick, G. L. Brauer et J. R. Rehder : A two-level trajectory decomposition algorithm featuring optimal intermediate target selection. *Journal of Spacecraft and Rockets*, 14(11):676–682, 1977. URL https://doi.org/10.2514/3.27991.

[89] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze et E. F. Mischenko : *The mathematical theory of optimal processes*. Taylor & Francis Group, 1962. URL https://doi.org/10.1201/9780203749319.

[90] M. J. D. Powell : On search directions for minimization algorithms. *Mathematical Programming*, 4(2):193–201, 1973. URL https://doi.org/10.1007/BF01584660.

[91] M. J. D. Powell : On the convergence of a wide range of trust region methods for unconstrained optimization. *IMA Journal of Numerical Analysis*, 30(1):289–301, 2010. URL https://doi.org/10.1093/imanum/drp021.

[92] T. M. Ragonneau : *Model-Based Derivative-Free Optimization Methods and Software*. Thèse de doctorat, Hong Kong Polytechnic University, 2023. URL http://arxiv.org/abs/2210.12018.

[93] T. M. Ragonneau et Z. Zhang : An optimal interpolation set for model-based derivative-free optimization methods, 2023. URL http://arxiv.org/abs/2302.09992.

[94] A. V. Rao : A survey of numerical methods for optimal control. *In Astrodynamics 2009 : AAS/AIAA Astrodynamics Specialist Conference*, vol. 135, p. 497–528, 2009. URL https://www.tib.eu/de/suchen/id/BLCP%3ACN075980000.

[95] S. S. Ravi, D. J. Rosenkrantz et G. K. Tayi : Facility dispersion problems : Heuristics and special cases. *In Algorithms and Data Structures*, p. 355–366. Springer Berlin Heidelberg, 1991. ISBN 978-3-540-47566-8. URL https://doi.org/10.1007/BFb0028275.

[96] R. G. Regis : The calculus of simplex gradients. *Optimization Letters*, 9(5):845–865, 2015. URL https://dx.doi.org/10.1007/s11590-014-0815-x.

[97] L. Roberts et C. W. Royer : Direct search based on probabilistic descent in reduced spaces, 2023. URL http://arxiv.org/abs/2204.01275.

[98] S. M. Robinson : An implicit-function theorem for a class of nonsmooth functions. *Mathematics of Operations Research*, 16(2):292–309, 1991. URL https://doi.org/10.1287/moor.16.2.292.

[99] R. T. Rockafellar : Generalized directional derivatives and subgradients of nonconvex functions. *Canadian Journal of Mathematics*, 32(2):257–280, 1980. URL https://doi.org/10.4153/CJM-1980-020-7.

[100] S. Ruder : An overview of gradient descent optimization algorithms, 2016. URL http://arxiv.org/abs/1609.04747.

[101] T. J. Santner, B. J. Williams et W. I. Notz : *The Design and Analysis of Computer Experiments*. Springer, 2003. URL 10.1007/978-1-4939-8847-1.

[102] A. L. Schwartz et E. Polak : Consistent approximations for optimal control problems based on Runge–Kutta integration. *SIAM Journal on Control and Optimization*, 34 (4):1235–1269, 1996. URL https://doi.org/10.1137/S0363012994267352.

[103] A. L. Schwartz et E. Polak : Runge-Kutta discretization of optimal control problems. *IFAC Proceedings Volumes*, 29:123–128, 1996. URL https://doi.org/10.1016/S1474-6670(17)43687-1.

[104] S. Sethi et G. Thompson : *Optimal Control Theory : Applications to Management Science and Economics*. Springer New York, 2000. URL https://doi.org/10.1007/978-3-030-91745-6.

[105] Z. Shao : *On random embeddings and their application to optimisation*. Thèse de doctorat, University of Oxford, 2021. URL https://ora.ox.ac.uk/objects/uuid:e2efee00-a69d-49f4-bf32-c85ae3ba10ec.

[106] T. W. Simpson, J. J. Korte, T. M. Mauery et F. Mistree : Kriging models for global approximation in simulation-based multidisciplinary design optimization. *AIAA Journal*, 39(12):2233–2241, 2001. URL https://dx.doi.org/10.2514/2.1234.

[107] G. Still : Lectures on parametric optimization : An introduction, 2018. URL https://optimization-online.org/2018/04/6587.

[108] E.-G. Talbi : Machine learning into metaheuristics : A survey and taxonomy. *ACM Comput. Surv.*, 54(6), 2021. URL https://doi.org/10.1145/3459664.

[109] A. Tamir : Obnoxious facility location on graphs. *SIAM Journal on Discrete Mathematics*, 4(4):550–567, 1991. URL https://doi.org/10.1137/0404048.

[110] A. Tfaily, M. Kokkolaras, N. Bartoli et Y. Diouane : Efficient acquisition functions for bayesian optimization in the presence of hidden constraints, 2023. URL https://arc.aiaa.org/doi/abs/10.2514/6.2023-4261.

[111] V. Torczon : On the convergence of pattern search algorithms. *SIAM Journal on Optimization*, 7(1):1–25, 1997. URL https://dx.doi.org/10.1137/S1052623493250780.

[112] E. Trélat : *Contrôle optimal : théorie et applications*. Vuibert supérieur, 2005. URL https://www.ljll.math.upmc.fr/~trelat/publications.html.

[113] E. Trélat et E. Zuazua : The turnpike property in finite-dimensional nonlinear optimal control. *Journal of Differential Equations*, 258(1):81–114, 2015. URL https://doi.org/10.1016/j.jde.2014.09.005.

[114] B. Van Dyke et T. J. Asaki : Using QR Decomposition to Obtain a New Instance of Mesh Adaptive Direct Search with Uniformly Distributed Polling Directions. *Journal of Optimization Theory and Applications*, 159(3):805–821, 2013. URL https://dx.doi.org/10.1007/s10957-013-0356-y.

[115] A. Verdério et E. W. Karas : On the construction of quadratic models for derivative-free trust-region algorithms. *EURO Journal on Computational Optimization*, 5(4):501–527, 2017. URL http://dx.doi.org/10.1007/s13675-017-0081-7.

[116] L. N. Vicente et A. L. Custódio : Analysis of direct searches for discontinuous functions. *Mathematical Programming*, 133(1-2):299–325, 2012. URL https://dx.doi.org/10.1007/s10107-010-0429-8.

[117] R. Vinter : *Optimal Control*. Birkhäuser Boston, 2010. URL https://doi.org/10.1007/978-0-8176-8086-2.

[118] A. Wächter et L. T. Biegler : On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006. URL https://dx.doi.org/10.1007/s10107-004-0559-y.

[119] G. Watson : Smoothing and interpolation by kriging and with splines. *Mathematical Geology*, 16:601–615, 1984. URL https://doi.org/10.1007/BF01029320.

[120] D. H. Wolpert et W. G. Macready : No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997. URL https://doi.org/10.1109/4235.585893.

[121] M. H. Wright : Nelder, Mead, and the other simplex method. *In Documenta Mathematica Extra Volume : Optimization Stories*, p. 271–276. Journal der Deutschen Mathematiker-Vereinigung Gegründet 1996, 2012. URL http://www.math.uiuc.edu/documenta/vol-ismp/42_wright-margaret.pdf.

[122] Y. Yuan : Recent advances in trust region algorithms. *Mathematical Programming*, 151 (1):249–281, 2015. URL https://dx.doi.org/10.1007/s10107-015-0893-2.

[123] Z. Zhang : PRIMA : Reference Implementation for Powell's methods with Modernization and Amelioration, 2023. URL http://www.libprima.net.