



HAL
open science

Optimisation multicritère sous incertitudes : un algorithme de descente stochastique

Quentin Mercier

► **To cite this version:**

Quentin Mercier. Optimisation multicritère sous incertitudes : un algorithme de descente stochastique. Equations aux dérivées partielles [math.AP]. COMUE Université Côte d'Azur (2015 - 2019), 2018. Français. NNT : 2018AZUR4076 . tel-02063322v3

HAL Id: tel-02063322

<https://theses.hal.science/tel-02063322v3>

Submitted on 11 Mar 2019

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

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

THÈSE DE DOCTORAT

Mention : Mathématiques

Présentée par :

Quentin Mercier

Soutenue le : 10 Octobre 2018

Optimisation multicritère sous incertitudes : un algorithme de descente stochastique

Jury :

M.	J.-M. Bourinet	Maître de conférences, Institut Pascal CNRS-SIGMA	Rapporteur
M.	E. Vazquez	Maître de conférences, L2S Centrale Supélec	Rapporteur
M.	J.-A. Désidéri	Directeur de recherche, Inria Sophia Antipolis Méditerranée et UCA	Codirecteur
M.	F. Poirion	Maître de recherche, DMAS Onera Université Paris-Saclay	Codirecteur
M.	C. Bès	Professeur, Institut Clément Ader Université Toulouse 3	Examineur
M.	O. Le Maître	Directeur de recherche CNRS, LIMSI Orsay	Examineur
M.	R. Le Riche	Directeur de recherche CNRS, LIMOS École des Mines de St Étienne	Examineur
M.	P. Carpentier	Enseignant-chercheur, ENSTA ParisTech	Examineur

Remerciements

J'aimerais tout d'abord remercier sincèrement mes deux directeurs de thèse Fabrice et Jean-Antoine. Merci de m'avoir encadré au cours de ces trois années de recherche et d'avoir participé activement à la réussite de ce projet. Je remercie tout particulièrement Fabrice pour sa présence quotidienne et sa patience presque sûrement sans borne lorsque je venais écrire des inepties sur son tableau blanc.

Je tiens également à remercier l'Onera dans son ensemble qui a financé ce projet de recherche. L'accueil qui m'a été réservé dans les deux laboratoires MS2 et MSAE que j'ai pu côtoyer m'a permis de travailler dans un cadre tout à fait idéal à la production scientifique. Je remercie donc l'ensemble des chercheurs travaillant dans ces deux unités à la fois pour leurs conseils, mais aussi pour les moments de vie commune passés ensemble. Je remercie aussi l'ensemble des doctorants et stagiaires avec qui j'ai pu partager durant ces trois ans et qui ont apporté de la légèreté à mon quotidien d'apprenti chercheur. Notamment l'armée d'Italiens de l'Onera, et plus particulièrement Rocco mon fidèle compagnon de café. Ainsi que Mikel avec qui j'ai partagé mon bureau pendant deux années.

L'École Normale Supérieure de Cachan mérite elle aussi tout ma gratitude puisqu'elle m'a poussé doucement vers le monde de la recherche sans que je m'en aperçoive et m'a donné les clés nécessaires à la production de contenus scientifiques de qualité. J'en profite également pour remercier l'ensemble des amis de l'école avec qui j'ai gardé contact, et qui pour la plus part se sont eux aussi orientés dans la recherche, pour leurs retours d'expérience sur la vie de doctorant et les galères administratives qu'elle peut impliquer. J'accompagne aussi ces remerciements d'une pensée toute particulière pour mon Albanais qui, depuis de nombreuses années maintenant, a toujours été un compagnon de travail incroyable et qui a grandement influencé certains choix dans la rédaction de ce manuscrit.

Enfin, je tiens à remercier mes parents pour leur soutien dans ma démarche professionnelle et leurs encouragements, mes frères pour le temps passé les mains dans le cambouis à bricoler loin des considérations mathématiques. Je tiens aussi à remercier mes colocataires avec qui j'ai partagé mon quotidien extra laboratoire durant ces trois dernières années ; Arnaud, Denis, Jules et Pierre.

Finalement, j'aimerais remercier l'ensemble des membres du jury qui ont validé et mis en avant la recherche que j'ai pu produire durant ces trois années.

Résumés

Français :

Cette thèse s'intéresse à l'optimisation multiobjectif sans contrainte lorsque les objectifs sont exprimés comme des espérances de fonctions aléatoires. L'aléa est modélisé par l'intermédiaire de variables aléatoires et on considère qu'il n'impacte pas les variables d'optimisation du problème. La thèse consiste à proposer un algorithme de descente qui permet l'obtention des solutions de Pareto du problème d'optimisation ainsi écrit. En utilisant des résultats d'analyse convexe, il est possible de construire un vecteur de descente commun à l'ensemble des objectifs du problème d'optimisation pour un tirage des variables aléatoires donné. Une suite itérative consistant à descendre dans la direction du vecteur de descente commun calculé au point courant et pour un tirage aléatoire unique et indépendant des variables aléatoires est alors construite. De cette manière, l'estimation coûteuse des espérances à chaque étape du processus d'optimisation n'est pas nécessaire. Il est possible de prouver les convergences en norme et presque sûre de cette suite vers les solutions de Pareto du problème d'optimisation en espérance et d'obtenir un résultat de vitesse de convergence lorsque la suite de pas de descente est bien choisie. Après avoir proposé diverses méthodes numériques d'amélioration de l'algorithme, un ensemble d'essais numériques est mené et les résultats de l'algorithme proposé sont comparés à ceux obtenus par deux autres algorithmes classiques de la littérature. Les résultats obtenus sont comparés par l'intermédiaire de mesures adaptées à l'optimisation multiobjectif et sont analysés par le biais de profils de performance. Des méthodes sont alors proposées pour prendre en compte deux types de contrainte et sont illustrées sur des problèmes d'optimisation de structures mécaniques. Une première méthode consiste à pénaliser les objectifs par l'intermédiaire de fonctions de pénalisation exacte lorsque la contrainte est décrite par une fonction déterministe. Pour les contraintes probabilistes, on propose de remplacer les contraintes par des objectifs supplémentaires, ces contraintes probabilistes étant alors reformulées comme des espérances de fonctions indicatrices, le problème étant résolu à l'aide de l'algorithme proposé dans la thèse sans avoir à estimer les probabilités des contraintes à chaque itération.

Mots clés : Optimisation ◦ Multiobjectif ◦ Stochastique ◦ Vecteur de descente ◦ Gradient ◦ Sous-différentielle ◦ Pareto

English :

This thesis deals with unconstrained multiobjective optimization when the objectives are written as expectations of random functions. The randomness is modelled through random variables and we consider that this does not impact the problem optimization variables. A descent algorithm is proposed which gives the Pareto solutions without having to estimate the expectancies. Using convex analysis results, it is possible to construct a common descent vector that is a descent vector for all the objectives simultaneously, for a given draw of the random variables. An iterative sequence is then built and consists in descending following this common descent vector calculated at the current point and for a single independent draw of the random variables. This construction avoids the costly estimation of the expectancies at each step of the algorithm. It is then possible to prove the mean square and almost sure convergence of the sequence towards Pareto solutions of the problem and at the same time, it is possible to obtain a speed rate result when the step size sequence is well chosen. After having proposed some numerical enhancements of the algorithm, it is tested on multiple test cases against two classical algorithms of the literature. The results for the three algorithms are then compared using two measures that have been devised for multiobjective optimization and analysed through performance profiles. Methods are then proposed to handle two types of constraint and are illustrated on mechanical structure optimization problems. The first method consists in penalising the objective functions using exact penalty functions when the constraint is deterministic. When the constraint is expressed as a probability, the constraint is replaced by an additional objective. The probability is then reformulated as an expectation of an indicator function and this new problem is solved using the algorithm proposed in the thesis without having to estimate the probability during the optimization process.

Keywords : Optimization ◦ Multiobjective ◦ Stochastic ◦ Descent vector ◦ Gradient ◦ Subdifferential ◦ Pareto

Table des matières

Table des figures	ix
Liste des tableaux	xi
1 Introduction	1
2 Rappels mathématiques	9
2.1 Probabilités et Processus stochastiques	9
2.1.1 Espace probabilisé, variables aléatoires et espérance	9
2.1.2 Espérance Conditionnelle	11
2.1.3 Processus et Martingales	13
2.2 Analyse convexe	14
2.2.1 Ensemble convexe et Enveloppe convexe	14
2.2.2 Sous-différentielles et dérivées directionnelles	14
3 Optimisation multiobjectif et vecteur de descente commun	21
3.1 Relations binaires	21
3.2 Optimisation multiobjectif	23
3.3 Vecteur de descente commun	30
4 Algorithmes d'optimisation multiobjectifs	35
4.1 Multiple Gradient Descent Algorithm (<i>MGDA</i>)	35
4.2 Stochastic Multiple Gradient Descent Algorithm (<i>SMGDA</i>)	38
4.2.1 Formulation du problème	38
4.2.2 <i>SMGDA</i>	39
5 Essais numériques de <i>SMGDA</i> et comparaisons	51
5.1 Améliorations de <i>SMGDA</i> et stratégies numériques d'optimisation . . .	51
5.1.1 Stratégie pour la génération de points de départ	52

5.1.2	Construction d'un estimateur et modification du vecteur de descente.	54
5.2	Choix des algorithmes de comparaison	59
5.3	Mesures de performance et profils de performance	60
5.4	Benchmarks	63
5.4.1	Réglage des algorithmes	64
5.4.2	Résultats	65
6	Prise en compte de contraintes déterministes et probabilistes, deux méthodes	69
6.1	Contraintes déterministes par pénalisation	69
6.2	Contraintes en probabilité par une approche multiobjectif	80
7	Conclusion	87
	Bibliographie	91
	Annexe A Tableau récapitulatif des essais numériques du Chapitre 5	97

Table des figures

2.1	Exemple de calcul d'une sous-différentielle d'une fonction convexe	16
2.2	Exemple de calcul d'une sous-différentielle d'une fonction non-convexe .	19
3.1	Exemple de la domination selon Pareto pour un problème biobjectif $\min_{\mathbf{x}}\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$	27
4.1	Exemple de calcul de pas ϵ optimal par approximation quadratique dans la direction ξ	37
5.1	Phénomène de glissement sur le front de Pareto	56
5.2	Objectifs du problème traité pour un tirage aléatoire unique et leur moyenne sur 5000 tirages	57
5.3	Résultats obtenus en utilisant le nouveau vecteur basé sur l'estimateur	58
5.4	Normes des vecteurs de descente calculées aux points finaux pour 15000 itérations : en orange, en utilisant l'estimateur et en vert, en utilisant le vecteur de descente commun initialement proposé	59
5.5	Exemple de calcul des mesures pour les solutions du problème JOS1 . .	62
5.6	Profil de performance pour la mesure de pureté	65
5.7	Solutions des algorithmes pour le cas de test MOP6	66
5.8	Profil de performance pour la mesure d'hypervolume	67
6.1	Matériau sandwich à optimiser	70
6.2	Deux échantillons de la cellule nids d'abeille	71
6.3	Indentation du cœur sous une charge critique F_{c1}	72
6.4	Dépassement de la résistance élastique de la peau inférieure sous une charge critique F_{c2}	72
6.5	Génération des points initiaux sur les 4 premières salves	75
6.6	Solutions des algorithmes dans l'espace des variables d'optimisation . .	76
6.7	Solutions des algorithmes dans l'espace des objectifs	76

6.8	Solutions <i>SMGDA</i> pour le problème avec et sans contrainte	77
6.9	Densité de probabilité pour le point solution \mathbf{y}_1	78
6.10	Densité de probabilité pour le point solution \mathbf{y}_2	79
6.11	Poutre étudiée en compression	81
6.12	Loi normale autour de 5% sur les tirages aléatoires lors de la génération de points initiaux	85
6.13	Résultats finaux de <i>SMGDA</i> pour le problème de fiabilité	86

Liste des tableaux

1.1	Liste des publications et communications	7
5.1	Exemple du déroulement de l'Algorithme 3	55
5.2	Fonctions test du benchmark	63
6.1	Propriétés matériau du cœur en fonction de \mathbf{R}	71
6.2	Contraintes de boîte \mathcal{X} sur les variables d'optimisation \mathbf{x}	73
6.3	Variables aléatoires du problème de fiabilité	81
6.4	Exemple de fonctions pour $h(x/r)$, où $\mathbf{1}_1$ désigne l'indicatrice de l'intervalle $[-1, 1[$	83
A.1	Paramètres des algorithmes <i>NSGA-II</i> et <i>DMS</i>	98
A.2	Description des objectifs des problèmes MOP	99
A.3	Description des objectifs des problèmes ZDT	100
A.4	Description des objectifs des problèmes restants	101

Chapitre 1

Introduction

Les constructeurs industriels recherchent à développer des systèmes ayant les meilleures performances, garantissant le meilleur niveau de fiabilité et de sécurité, tout en réduisant au mieux les coûts de production. Un moyen de répondre simultanément à ces objectifs multiples et antagonistes est d'intégrer au niveau du processus de conception une ou plusieurs étapes d'optimisation multiobjectif.

L'aéronautique fournit de nombreux exemples illustrant ce type de problématiques. En aérodynamique, on optimise la géométrie des surfaces portantes, et plus généralement des configurations entières plus ou moins simplifiées, de manière à atteindre un bon compromis entre performances pures, stabilité et manœuvrabilité en extrémalisant un ou plusieurs critères qui intègrent ces notions via les coefficients aérodynamiques (forces et moments). Le bon compromis dépend du point considéré sur l'enveloppe de vol de l'engin ("problème multipoint"). Typiquement, en matière d'aérodynamique des avions civils, domaine industriel très concurrentiel, étant donné les enjeux économiques et les contraintes légales environnementales, il est primordial de minimiser la traînée en régime de croisière de vol transsonique sous contrainte de portance, pour maximiser le rayon d'action, et minimiser la consommation et l'impact de pollution atmosphérique. Pour la phase de décollage (et d'atterrissage), on cherche à maximiser la portance de l'engin dans une configuration différente (vol subsonique à grande incidence, volets hyper-sustentateurs déployés) afin de garantir la sécurité de cette phase de vol. En aérodynamique des avions militaires, d'autres critères sont pris en considération. En particulier, un intercepteur peut être amené à déclencher volontairement un décrochage dans une manœuvre d'évitement. Pour cette raison la stabilité des avions militaires est optimisée différemment.

De manière générale, les formes sont optimisées vis-à-vis de disciplines concourantes. En particulier, les voilures dont la structuration interne est complexe, sont analysées

en matière de résistance des matériaux. La rigidité est habituellement maximisée par minimisation de la compliance. Notons dès à présent, que cette optimisation est habituellement très conflictuelle à celle de l'optimisation aérodynamique, ce qui renforce le besoin de méthodes performantes en optimisation multiobjectif.

Au stade de l'avant projet, le constructeur intègre l'ensemble de ces différentes disciplines dans des "boîtes noires" pour lesquelles, un jeu de paramètres de conception de l'avion résulte sur un ensemble de performances réalisées (masse, coefficients aérodynamiques, rigidité de l'avion, etc. . .) et ceci permet d'aborder globalement les questions de mécanique du vol.

Chercher le ou les meilleurs jeux de paramètres de conception revient mathématiquement à résoudre un problème d'optimisation multiobjectif dont on voit qu'il revêt de nombreux aspects pour optimiser un système aussi complexe qu'un avion.

La présence importante de problématiques d'optimisation multiobjectif chez les industriels a favorisé un développement important de méthodes numériques en optimisation multiobjectif et en recherche opérationnelle. On pourra notamment citer le livre de Collette et Siarry [15] qui recense un grand nombre de méthodes d'optimisation multiobjectif récentes et leur contexte d'utilisation, ainsi que le livre de Miettinen [55] qui propose des méthodes étendues à un cadre où les objectifs sont non-linéaires. La communauté scientifique autour des problèmes d'optimisation multiobjectif est aujourd'hui très riche et active. On note un nombre important de projets de recherche européens et nationaux dont le projet ANR *Optimisation MultiDisciplinaire (OMD)* coordonné par le CNRS et l'École des Mines de Saint-Étienne qui a découlé sur un grand nombre de publications portant sur la thématique multiobjectif dans le livre de Coelho et Breitkopf [14] et de Breitkopf et Coelho [10]. On pourra de plus distinguer deux approches dans l'optimisation multiobjectif, une première appelée l'optimisation multidisciplinaire, où les différents objectifs font appels à des disciplines variées pour lesquelles les objectifs peuvent être interdépendants. C'est le cas de l'optimisation aéro-structure dont une application fait l'objet de la thèse de Achard [1], qui en plus de proposer une méthode innovante pour le calcul des gradients des objectifs par rapport aux paramètres aéro-structures, réalise un exemple d'optimisation d'une aile sous des critères à la fois structuraux et aérodynamiques pour lesquels un couplage entre ces deux disciplines est pris en compte. Le second type de problème est le problème multipoint, où cette fois un seul objectif est considéré, mais dans des configurations différentes.

Les méthodes d'optimisation multiobjectif s'inspirent souvent des méthodes développées en optimisation mono-objectif pour lesquelles une adaptation au cadre multicritère a été effectuée. Les premières méthodes à apparaître consistent entre autres à ramener les problèmes multiobjectifs à des formulations mono-objectifs pour lesquelles un grand nombre d'algorithmes de résolution efficaces existent. Mais ce type de méthode a l'inconvénient de perdre l'aspect multiobjectif du problème, et a tendance à ne proposer qu'une unique solution parmi l'ensemble des solutions possibles et privilégie les objectifs entre eux, on parle d'optimisation *a priori*. Plus récemment, les adaptations des algorithmes heuristiques ont connu un important essor. Ces méthodes ont pour avantage d'être relativement simples à mettre en œuvre, leur adaptation nécessite seulement un changement dans la manière de noter ou de sélectionner les individus performants. On peut notamment citer parmi les algorithmes génétiques les plus réputés ceux décrits dans les papiers suivants : *NSGA-II* [19], *SPEA* [77], *PAES* [43]. Bien que largement utilisés lorsque les objectifs sont simples à calculer, le nombre de calculs nécessaires à leur convergence est très élevé, ce qui les rend difficilement applicables à des problèmes requérant des calculs haute performance ou numériquement coûteux. L'extension des méthodes de gradient n'arrive que plus tard en optimisation multiobjectif, mais vient combler le manque de rapidité des algorithmes précédemment cités. En optimisation multiobjectif convexe différentiable, l'Inria a développé la méthode *Multiple Gradient Descent Algorithm (MGDA)* [20] qui, lorsqu'on connaît en un point les gradients des différents critères du problème, permet de construire une direction de descente commune à l'ensemble des objectifs et ce quel que soit leur nombre. Quand les gradients sont linéairement indépendants, la direction de descente fournie est telle que les dérivées directionnelles des critères dans cette direction sont égales. Au second ordre près, les objectifs diminuent donc d'autant à chaque itération. Lorsque la direction de descente fournie est nulle, on atteint un point Pareto stationnaire du problème multiobjectif. Cette méthode continue à être développée à l'Inria suivant différentes voies d'améliorations et a donné lieu à une plateforme logicielle disponible en ligne <https://mgda.inria.fr>. Un exemple d'application de la méthode *MGDA* à un cas où le nombre de gradients est grand est fourni dans Désidéri et Duvigneau [22], où les auteurs s'intéressent à la minimisation de la traînée d'une plaque plane par l'intermédiaire de jets pulsés périodiques dans un cadre d'écoulement compressible visqueux laminaire. L'optimisation est menée sur 6 paramètres définissant le fonctionnement de ces jets de manière à réduire la traînée à tous les instants de la période considérée. Les développements récents portent sur le traitement des contraintes [21], ainsi que sur le traitement de problèmes multiobjectifs priorisés par partage de territoire et jeux de Nash [23]. Cette

méthode *MGDA-Nash*, à partir d'un point Pareto optimal vis-à-vis des seuls critères prioritaires, permet de construire un continuum de solutions, équilibres de Nash, qui améliorent d'autres critères initialement considérés comme secondaires tout en préservant la Pareto-optimalité des critères prioritaires. La méthode *MGDA* a aussi été étendue au cadre non-différentiable dans l'article de Mäkelä *et al.* [50].

Parallèlement est apparu le besoin de prendre en compte les incertitudes inhérentes à tout problème réel où les phénomènes obtenus sont affectés par des méconnaissances ou des fluctuations aléatoires (défauts géométriques, matériaux, etc...). Les travaux incluant les incertitudes lors du processus d'optimisation ont connu d'importants développements au cours du 20^{ième} siècle avec notamment les travaux de Dantzig [17], Bellman et Zadeh [6], Klamroth *et al.* [42] et de nombreuses approches ont été proposées parmi lesquelles on retrouve l'optimisation dite robuste. L'optimisation robuste inclut un nombre important de concepts de robustesse comme l'optimisation du "pire cas" ou encore l'optimisation sur la moyenne et la variance [36, 41] ainsi que l'optimisation stochastique où les paramètres incertains sont modélisés par des variables aléatoires de distribution connue et où l'information probabiliste est directement incluse dans les approches numériques. Dans ce contexte, les objectifs du problème d'optimisation sont écrits comme des espérances de fonctions aléatoires. Dans cette thèse, on s'intéresse tout particulièrement à cette dernière formulation. Concernant l'optimisation stochastique mono-objectif, la littérature est bien plus riche et comporte un grand nombre d'approches numériques [68, 67]. Deux approches distinctes existent, une première basée sur une approximation stochastique comme l'algorithme proposé initialement par Robbins et Monro [66] et qui a fait l'objet de différentes améliorations [31, 32]. La seconde approche est une approche dite par scénarios [71, 58], cette dernière étant plus fréquemment utilisée pour les problèmes d'optimisation avec contraintes en probabilité. L'optimisation multiobjectif stochastique étant quant à elle moins développée dans la littérature, la majorité des approches proposées sont basées sur l'utilisation d'un algorithme classique d'optimisation déterministe couplé à une formulation robuste où les moyennes stochastiques ou les variances sont soit obtenues de manière analytique sur des objectifs simples [34, 18, 12] ou estimées par le biais d'approches de type scénarios (*Sample Average Approximation*) [35, 8, 51, 47, 24, 74, 44, 64, 72]. L'inconvénient majeur des méthodes couplées est le coût numérique engendré par l'approximation des espérances à chaque itération de l'algorithme d'optimisation.

Cette thèse s'intéresse donc au développement d'un algorithme d'optimisation multiobjectif qui s'abstient du calcul coûteux des espérances. À cette fin, on cherche à étendre l'algorithme du gradient stochastique [66] au cadre multiobjectif en s'appuyant sur les résultats d'analyse convexe qui ont été mis en avant pour l'algorithme multiobjectif *MGDA* dans le cadre de fonctions différentiables et non-différentiables. On cherche de plus à démontrer des résultats théoriques de convergence de l'algorithme vers les solutions du problème d'optimisation multiobjectif en espérance et à comparer, sur un ensemble de tests classiques de la littérature, les performances de ce nouvel algorithme par rapport aux approches présentées ci-dessus.

Après avoir rappelé dans le Chapitre 2 l'ensemble des résultats d'analyse convexe et de probabilité nécessaires aux développements menés dans cette thèse, on présente formellement dans le Chapitre 3 la formulation mathématique d'un problème d'optimisation et ce qu'implique le passage au cadre multiobjectif. Dans ce même chapitre, on donne aussi les résultats théoriques qui concernent la construction d'un vecteur de descente commun pour un ensemble de classes de fonctions objectifs (convexes différentiables, non-différentiables, pseudo-convexes). Une fois la construction du vecteur de descente commun effectuée, on présente alors, dans le Chapitre 4, l'algorithme *MGDA* qui permet de résoudre les problèmes déterministes d'optimisation multiobjectif. Le passage à un problème stochastique donne alors l'opportunité de proposer un nouvel algorithme *Stochastic Multiple Gradient Descent Algorithm (SMGDA)* qui correspond à l'extension de *MGDA* lorsque les objectifs du problème sont écrits comme des espérances de fonctions aléatoires. Des preuves de convergence de ce nouvel algorithme vers les solutions du problème stochastique sont alors proposées et permettent justifier la proposition de ce nouvel algorithme de résolution. Dans le chapitre 5, après avoir introduit des améliorations numériques de *SMGDA*, on compare les performances de ce nouvel algorithme aux performances des algorithmes classiques de la littérature, ceux-ci étant couplés avec des approches *Sample Average Approximation*. Ces essais sont effectués sur un ensemble de tests classiques de la littérature auxquels des incertitudes ont été rajoutées, et les résultats sont analysés par l'intermédiaire de deux mesures de performance. L'analyse finale des résultats s'appuie en partie sur l'utilisation de profils de performances qui notent les algorithmes testés sur l'ensemble des tests numériques sans avoir à les analyser tous un à un. La question des problèmes d'optimisation contraints est traitée au Chapitre 6 par l'intermédiaire d'applications à des problèmes d'optimisation de structures. Deux types de contraintes sont analysés dans ce chapitre, dans un premier temps les contraintes déterministes pour lesquelles on utilise des techniques de pénalisation des objectifs. Puis dans un second temps, on s'intéresse

aux problèmes d'optimisation avec contraintes probabilistes qu'on réécrit comme des problèmes multiobjectifs stochastiques non contraints.

Les travaux de cette thèse ont donné lieu à des publications et communications décrites dans le tableau 1.1 qui suit.

TABLE 1.1 Liste des publications et communications

Journaux	Titre	Journal
[54] QM,FP,JAD	A Stochastic Multiple Gradient Descent Algorithm	Journal of Operational research
[62] FP,QM,JAD	Descent algorithm for nonsmooth stochastic multiobjective optimization	Computational Optimization and Applications
[53] QM,FP,JAD	Nonconvex multiobjective design optimization under uncertainty : a descent algorithm. Application to sandwich plate design and reliability	Engineering Optimization
[63] FP,QM,JAD	Descent methods for design optimization under uncertainty	Aerospacelab Journal
Proceedings	Titre	Recueil
[52] QM,FP,JAD	SMGDA : an uncertainty based multiobjective optimization approach. Illustration to an airplane composite material	Procedia engineering
[61] FP,QM,JAD	Uncertainty-based multidisciplinary design optimization. A descent algorithm	Proceedings of the joint ICVRAM ISUMA UNCERTAIN TIES conference
Conférences	Titre	Lieu
Optimization 2017	Stochastic Multiple Gradient Descent Algorithm (SMGDA). Illustration on a Sandwich Material Optimization Problem	Lisbonne, Portugal
ParaOptXI	A gradient-based method for multiobjective optimization problem under uncertainties (<i>SMGDA</i>)	Prague, République Tchèque
CSM8	RBDO problem solving through multiobjective formulation using a stochastic multiple gradient decent algorithm (<i>SMGDA</i>)	Paros, Grèce
ISMP2018	A descent algorithm for stochastic multiobjective optimization problems	Bordeaux, France

Chapitre 2

Rappels mathématiques

Dans l'ensemble du document, la notation $\langle \cdot, \cdot \rangle$ sera utilisé pour désigner le produit scalaire euclidien de \mathbb{R}^n . De même, on utilise la notation $\| \cdot \|$ pour la norme associée au produit scalaire de \mathbb{R}^n .

2.1 Probabilités et Processus stochastiques

L'algorithme proposé dans la thèse définit une suite récursive de variables aléatoires. Nous rappelons ici, sans les développer, les quelques notions de probabilité nécessaires à la construction de l'algorithme ainsi qu'à la démonstration des résultats de convergence. Les détails relatifs aux résultats rappelés ici peuvent être trouvés dans l'ouvrage de Durrett [28]. Pour les résultats sur les martingales à temps discret, le lecteur pourra se rapprocher de Neveu [59].

2.1.1 Espace probabilisé, variables aléatoires et espérance

Définition 1. Soit Ω un ensemble, on appelle tribu sur Ω un ensemble \mathcal{T} de parties de Ω qui vérifie

— \mathcal{T} n'est pas vide

$$\mathcal{T} \neq \emptyset$$

— \mathcal{T} est stable par complémentaire

$$\forall \mathcal{A} \in \mathcal{T}, \mathcal{A}^c \in \mathcal{T}$$

— \mathcal{T} est stable par union dénombrable

$$\forall n \in \mathbb{N}, \mathcal{A}_n \in \mathcal{T} : \bigcup_{n \in \mathbb{N}} \mathcal{A}_n \in \mathcal{T}$$

Les parties de Ω qui appartiennent à la tribu \mathcal{T} sont appelés les ensembles mesurables ou encore évènements dans un contexte probabiliste.

Définition 2. Soient Ω un ensemble et \mathcal{T} une tribu sur Ω , on appelle ensemble mesurable tout élément de \mathcal{T} . On dit du couple (Ω, \mathcal{T}) qu'il forme un espace mesurable.

Définition 3. Soit (Ω, \mathcal{T}) un espace mesurable. Une application μ définie sur \mathcal{T} à valeurs dans $[0, +\infty]$ est appelée mesure lorsque les deux propriétés suivantes sont satisfaites :

— L'ensemble vide est de mesure nulle

$$\mu(\emptyset) = 0$$

— μ est σ -additive : si E_1, E_2, \dots est une famille dénombrable de parties de Ω appartenant à \mathcal{T} et si ces parties sont deux à deux disjointes, alors la mesure de l'union des parties est égale à la somme des mesures des parties

$$\mu\left(\bigcup_{k=1}^{\infty} E_k\right) = \sum_{k=1}^{\infty} \mu(E_k)$$

Définition 4. Soit μ une mesure sur l'espace mesurable (Ω, \mathcal{T}) , on dit que μ est une mesure de probabilité si μ a la particularité d'avoir une masse égale à 1

$$\mu(\Omega) = 1.$$

De manière générale, on utilise la notation \mathbb{P} pour désigner une mesure de probabilité.

Définition 5. Soit \mathbb{P} une mesure de probabilité sur (Ω, \mathcal{T}) . On dit que le triplet $(\Omega, \mathcal{T}, \mathbb{P})$ est un espace probabilisé.

Définition 6. Soient (E, \mathcal{E}) et (F, \mathcal{F}) deux ensembles mesurables. Une fonction $f : E \rightarrow F$ est dite mesurable si la tribu image réciproque par f de la tribu \mathcal{F} est incluse dans \mathcal{E}

$$\forall B \in \mathcal{F}, f^{-1}(B) \in \mathcal{E}.$$

Définition 7. Soient $(\Omega, \mathcal{T}, \mathbb{P})$ un espace probabilisé et (F, \mathcal{F}) un espace mesurable. On appelle variable aléatoire de Ω vers F toute fonction mesurable X de Ω vers F . La mesure de probabilité \mathbb{P} sur (Ω, \mathcal{T}) permet de définir une mesure de probabilité sur (F, \mathcal{F}) image par la réciproque de la variable aléatoire X de la mesure de probabilité définie sur (Ω, \mathcal{T}) et notée \mathbb{P}_X par

$$\forall B \in \mathcal{F}, \mathbb{P}_X(B) = \mathbb{P}(X^{-1}(B)) = \mathbb{P}(X \in B),$$

\mathbb{P}_X est appelé loi de la variable aléatoire X . À partir d'une famille de variables aléatoires, il est possible d'engendrer une tribu qui est la plus petite tribu qui rend la famille de variables aléatoires mesurables.

Définition 8. Soit $\{X_i\}_{i \in \mathbb{N}}$ une famille de variables aléatoires sur $(\Omega, \mathcal{T}, \mathbb{P})$ et à valeurs dans les espaces mesurables $(F_i, \mathcal{F}_i)_{i \in \mathbb{N}}$. On appelle tribu engendrée par $\{X_i\}_{i \in \mathbb{N}}$ la plus petite tribu contenant la réunion des tribus images réciproques $X_i^{-1}(\mathcal{F}_i)$ et on la note $\sigma(X_i, i \in \mathbb{N})$.

Définition 9. Soit X une variable aléatoire définie sur $(\Omega, \mathcal{T}, \mathbb{P})$ et à valeurs dans \mathbb{R} muni de la tribu des boréliens notée $\mathcal{B}(\mathbb{R})$ (engendrée par les ouverts de \mathbb{R}). Si X est intégrable selon la mesure \mathbb{P} , l'espérance est définie par

$$\mathbb{E}[X] = \int_{\Omega} X(\omega) d\mathbb{P}(\omega) = \int_{\mathbb{R}} x d\mathbb{P}_X(x).$$

2.1.2 Espérance Conditionnelle

Soit $\mathcal{A} \subseteq \mathcal{T}$ une sous-tribu.

Définition 10. Soit X une variable aléatoire réelle intégrable définie sur (Ω, \mathcal{T}) . Il existe une unique variable aléatoire Z \mathcal{A} -mesurable et intégrable vérifiant pour toute variable Y \mathcal{A} -mesurable et intégrable la relation

$$\mathbb{E}[XY] = \mathbb{E}[ZY].$$

On note Z sous la forme $\mathbb{E}[X|\mathcal{A}]$, et on la désigne comme l'espérance conditionnelle de X sachant \mathcal{A} .

Remarque 1. Lorsque la tribu \mathcal{A} est engendrée par une variable aléatoire Y , l'espérance conditionnelle de X sachant Y est définie comme l'espérance conditionnelle de X sachant \mathcal{A} . De plus il existe une fonction f telle que

$$\mathbb{E}[X|Y] = f(Y), \text{ p.s.}$$

Proposition 1. *L'espérance conditionnelle vérifie les propriétés suivantes*

1. *Linéarité :*

$$\mathbb{E}[\lambda X + \mu Y | \mathcal{A}] = \lambda \mathbb{E}[X | \mathcal{A}] + \mu \mathbb{E}[Y | \mathcal{A}],$$

2. *Positivité :*

$$X > Y \Rightarrow \mathbb{E}[X | \mathcal{A}] > \mathbb{E}[Y | \mathcal{A}],$$

3. *Si X est \mathcal{A} -mesurable et intégrable :*

$$\mathbb{E}[X | \mathcal{A}] = X,$$

4. *Si Z est \mathcal{A} -mesurable et intégrable :*

$$\mathbb{E}[ZX | \mathcal{A}] = Z \mathbb{E}[X | \mathcal{A}],$$

5. *Soit \mathcal{B} une sous tribu de \mathcal{A} :*

$$\mathbb{E}[\mathbb{E}[X | \mathcal{A}]] = \mathbb{E}[X]$$

$$\mathbb{E}[\mathbb{E}[X | \mathcal{A}] | \mathcal{B}] = \mathbb{E}[X | \mathcal{B}],$$

6. *Si X est indépendante de \mathcal{A} , c'est à dire indépendante de toutes les variables aléatoires \mathcal{A} -mesurable, alors :*

$$\mathbb{E}[X | \mathcal{A}] = \mathbb{E}[X],$$

7. *Contraction par la norme L^2 :*

$$\mathbb{E}[|\mathbb{E}[X | \mathcal{A}]|^2] \leq \mathbb{E}[|X|^2].$$

Lemme 1. *Soit \mathcal{A} une sous-tribu de \mathcal{T} , X et Y des variables aléatoires telles que X est indépendante de \mathcal{A} et que Y est \mathcal{A} -mesurable. Alors pour toute fonction f mesurable bornée de \mathbb{R}^2 dans \mathbb{R}*

$$\begin{cases} \mathbb{E}[f(X, Y) | \mathcal{A}] = \psi(Y) \\ \psi(y) = \mathbb{E}[f(X, y)]. \end{cases}$$

2.1.3 Processus et Martingales

Nous serons amenés au cours de ce travail à étudier les propriétés d'une suite de variables aléatoires. Une telle suite définit un processus stochastique (à temps discret). Nous rappelons les quelques notions sur les processus qui seront utilisées par la suite.

Définition 11. *Un processus aléatoire à temps discret est une suite de variables aléatoires $(X_k)_{k \in \mathbb{N}}$ définies sur $(\Omega, \mathcal{T}, \mathbb{P})$ à valeurs dans \mathbb{R}^n .*

Définition 12. *On dit qu'un processus aléatoire $(X_k)_{k \in \mathbb{N}}$ converge en norme L^2 vers une variable aléatoire X_∞ si, pour tout temps k , la variable aléatoire X_k est de carré intégrable et*

$$\lim_{k \rightarrow +\infty} \mathbb{E} [\|X_k - X_\infty\|^2] = 0.$$

Définition 13. *On dit qu'un processus aléatoire $(X_k)_{k \in \mathbb{N}}$ converge presque sûrement vers une variable aléatoire X_∞ si*

$$\mathbb{P} \left(\left\{ \lim_{k \rightarrow +\infty} X_k = X_\infty \right\} \right) = 1.$$

Définition 14. *Une filtration est une suite de sous-tribus $(\mathcal{F}_k)_{k \in \mathbb{N}}$ croissante pour l'inclusion*

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \dots \subset \mathcal{F}_k.$$

La tribu \mathcal{F}_k dans ce cas contient l'information acquise au temps k .

Définition 15. *Un processus aléatoire $(X_k)_{k \in \mathbb{N}}$ est dit adapté à la filtration $(\mathcal{F}_k)_{k \in \mathbb{N}}$ si pour tout k , la variable aléatoire X_k est \mathcal{F}_k -mesurable*

$$\mathbb{E} [X_k | \mathcal{F}_k] = X_k.$$

Définition 16. *Soit $(X_k)_{k \in \mathbb{N}}$ un processus à valeurs réelles adapté à la filtration $(\mathcal{F}_k)_{k \in \mathbb{N}}$ tel que X_k est intégrable pour tout k . On dit que le processus est*

— *une martingale si pour tout k*

$$\mathbb{E} [X_{k+1} | \mathcal{F}_k] = X_k,$$

— *une surmartingale si pour tout k*

$$\mathbb{E} [X_{k+1} | \mathcal{F}_k] \leq X_k,$$

— une sousmartingale si pour tout k

$$\mathbb{E}[X_{k+1}|\mathcal{F}_k] \geq X_k.$$

Théorème 1 (Neveu [59]). *Soit $(X_k)_{k \in \mathbb{N}}$ une surmartingale positive ($X_k \geq 0$ p.s., $\forall k$). Il existe une variable aléatoire X_∞ telle que $(X_k)_{k \in \mathbb{N}}$ converge vers X_∞ presque sûrement.*

2.2 Analyse convexe

Cette partie du chapitre donne les quelques notions d'analyse convexe qui seront utilisées lors de la construction de l'algorithme d'optimisation présenté dans la suite. Les notions et résultats rappelés ici sont tirés des ouvrages de Bonnans *et al.* [7], Clarke [13], Miettinen [55].

2.2.1 Ensemble convexe et Enveloppe convexe

Définition 17. *Un ensemble $C \subseteq \mathbb{R}^n$ est dit convexe si pour tout élément \mathbf{x} et \mathbf{y} de C , le segment $[\mathbf{x}, \mathbf{y}]$ est entièrement contenu dans C*

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

Définition 18. *Soit A une partie de $E \subseteq \mathbb{R}^n$. L'enveloppe convexe de A est l'intersection de toutes les parties convexes de E contenant A . On notera l'enveloppe convexe de A par $\text{Conv}(A)$.*

Proposition 2. *Soit A une partie de $E \subseteq \mathbb{R}^n$. L'enveloppe convexe de A est l'ensemble des combinaisons convexes de familles de points de A*

$$\text{Conv}(A) = \left\{ \mathbf{x} \in E : \exists \{\mathbf{a}_i\}_{i \in [1, p]}, \mathbf{a}_i \in A; \mathbf{x} = \sum_{i=1}^p \lambda_i \mathbf{a}_i, \lambda_i \geq 0, \sum_{i=1}^p \lambda_i = 1 \right\}.$$

2.2.2 Sous-différentielles et dérivées directionnelles

Définition 19. *La fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite lipschitzienne si il existe K strictement positif tel que pour tout (\mathbf{x}, \mathbf{y}) dans $(\mathbb{R}^n)^2$ on a l'inégalité suivante*

$$|f(\mathbf{y}) - f(\mathbf{x})| \leq K|\mathbf{y} - \mathbf{x}|.$$

Définition 20. La fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite localement lipschitzienne au point \mathbf{x} si il existe une boule de centre \mathbf{x} et de rayon ρ positif telle que pour tout (\mathbf{y}, \mathbf{z}) dans $\mathcal{B}(\mathbf{x}, \rho)$ on a l'inégalité suivante

$$|f(\mathbf{z}) - f(\mathbf{y})| \leq K|\mathbf{z} - \mathbf{y}|, \quad K > 0.$$

Définition 21. La dérivée directionnelle au point \mathbf{x} suivant la direction \mathbf{v} dans \mathbb{R}^n d'une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est définie par la limite

$$f'(\mathbf{x}, \mathbf{v}) = \lim_{t \downarrow 0} \frac{f(\mathbf{x} + t\mathbf{v}) - f(\mathbf{x})}{t}.$$

Lorsque f est différentiable, on a

$$f'(\mathbf{x}, \mathbf{v}) = \langle \nabla f(\mathbf{x}), \mathbf{v} \rangle,$$

où $\nabla f(\mathbf{x}) = (\frac{\partial}{\partial x_1}(f(\mathbf{x})), \dots, \frac{\partial}{\partial x_n}(f(\mathbf{x})))$ désigne le gradient de la fonction f au point \mathbf{x}

Définition 22. Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite convexe si pour tout \mathbf{x} et \mathbf{y} dans \mathbb{R}^n et λ dans $[0, 1]$ l'inégalité suivante est vérifiée

$$f(\lambda\mathbf{x} + (1 - \lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1 - \lambda)f(\mathbf{y}).$$

Définition 23. La sous-différentielle d'une fonction convexe $f : \mathbb{R}^n \rightarrow \mathbb{R}$ au point \mathbf{x} est définie par l'ensemble

$$\partial f(\mathbf{x}) = \{\mathbf{s} \in \mathbb{R}^n : f'(\mathbf{x}, \mathbf{d}) \geq \langle \mathbf{s}, \mathbf{d} \rangle, \forall \mathbf{d} \in \mathbb{R}^n\}.$$

Cet ensemble est non-vide, convexe et fermé. Chaque vecteur de $\partial f(\mathbf{x})$ est appelé sous-gradient de f au point \mathbf{x} , et la sous-différentielle est réduite au gradient $\nabla f(\mathbf{x})$ lorsque la fonction f est différentiable.

La Figure 2.1 illustre ce qu'est la sous-différentielle d'une fonction convexe. Dans cet exemple, la fonction $f(x)$ est continue et convexe mais elle est non-différentiable aux points x_1 et x_2 . En ces points la sous-différentielle correspond à l'ensemble des vecteurs directeurs des droites qui passent par le point $(x_i)_{i \in \llbracket 1, 2 \rrbracket}$ en restant sous la courbe représentant la fonction $f(x)$.

La dérivée directionnelle définie précédemment n'existe pas forcément quand la fonction n'est plus convexe mais seulement localement lipschitzienne. Nous rappelons

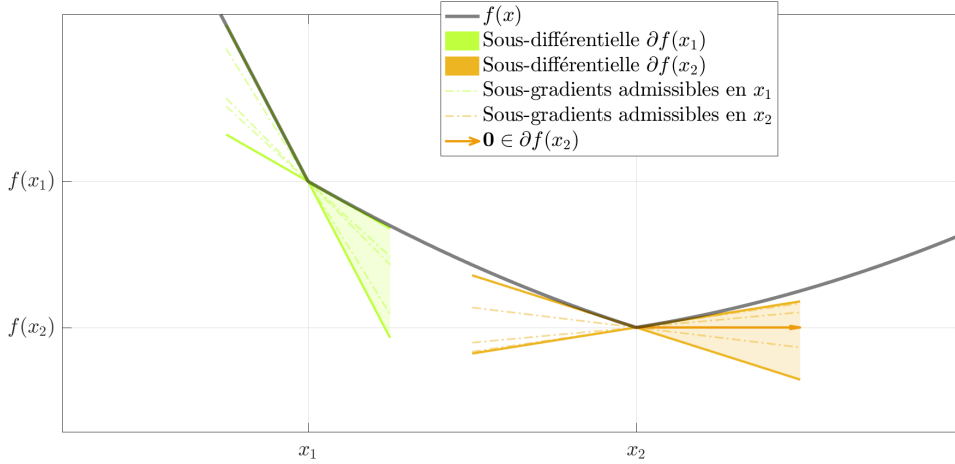


FIGURE 2.1 Exemple de calcul d'une sous-différentielle d'une fonction convexe

l'approche suivie par Clarke [13] pour généraliser cette notion ainsi que la notion de sous-différentielle au cas des fonctions localement lipschitziennes.

Définition 24. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction continue localement lipschitzienne. La dérivée directionnelle généralisée de la fonction f au point \mathbf{x} dans la direction \mathbf{v} de \mathbb{R}^n est définie par

$$f^\circ(\mathbf{x}, \mathbf{v}) = \limsup_{\mathbf{y} \rightarrow \mathbf{x}} \sup_{t \downarrow 0} \frac{f(\mathbf{y} + t\mathbf{v}) - f(\mathbf{y})}{t}.$$

Définition 25. Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ continue localement lipschitzienne est dite régulière pour la sous-différentiabilité en un point $\mathbf{x} \in \mathbb{R}^n$ si pour tout $\mathbf{d} \in \mathbb{R}^n$ la dérivée directionnelle $f'(\mathbf{x}, \mathbf{d})$ existe et si on a l'égalité

$$f'(\mathbf{x}, \mathbf{d}) = f^\circ(\mathbf{x}, \mathbf{d}).$$

Définition 26. Une fonction continue localement lipschitzienne $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est f° -quasi-convexe si

$$\forall (\mathbf{x}, \mathbf{y}) \in (\mathbb{R}^n)^2, f(\mathbf{y}) \leq f(\mathbf{x}) \Rightarrow f^\circ(\mathbf{x}, \mathbf{y} - \mathbf{x}) \leq 0.$$

Définition 27. Une fonction continue localement lipschitzienne $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est f° -pseudo-convexe si

$$\forall (\mathbf{x}, \mathbf{y}) \in (\mathbb{R}^n)^2, f(\mathbf{y}) < f(\mathbf{x}) \Rightarrow f^\circ(\mathbf{x}, \mathbf{y} - \mathbf{x}) < 0.$$

Théorème 2. (Mäkelä et al. [49])

Une fonction f° -pseudo-convexe est f° -quasi-convexe.

Définition 28. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction continue localement lipschitz. La sous-différentielle de Clarke de f au point \mathbf{x} est l'ensemble $\partial^{\mathcal{C}} f(\mathbf{x})$ de vecteurs défini par

$$\partial^{\mathcal{C}} f(\mathbf{x}) = \{\mathbf{s} \in \mathbb{R}^n : f^\circ(\mathbf{x}, \mathbf{v}) \geq \langle \mathbf{s}, \mathbf{v} \rangle \forall \mathbf{v} \in \mathbb{R}^n\}.$$

Proposition 3. (Mäkelä et al. [49])

Les fonctions convexes sont régulières pour la sous-différentiabilité. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe, en un point $\mathbf{x} \in \mathbb{R}^n$, on a

$$\partial f(\mathbf{x}) = \partial^{\mathcal{C}} f(\mathbf{x})$$

Théorème 3. (Mäkelä et al. [49])

Soit m fonctions $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ localement lipschitziennes et continue en $\mathbf{x} \in \mathbb{R}^n$. Et soit $\{\lambda_i\}_{i \in \llbracket 1, m \rrbracket}$ une famille de réels. La fonction définie par

$$h(\mathbf{x}) = \sum_{i=1}^m \lambda_i f_i(\mathbf{x}),$$

est localement lipschitzienne et continue en \mathbf{x} . De plus sa sous-différentielle a la propriété suivante

$$\partial h(\mathbf{x}) \subseteq \sum_{i=1}^m \lambda_i \partial f_i(\mathbf{x}).$$

De plus, lorsque les fonctions f_i sont régulières pour la sous-différentielle et que la famille des $\lambda_i, i = 1, \dots, m$ est uniquement constituée de réels positifs alors h est aussi régulière pour la sous-différentielle, et on a

$$\partial h(\mathbf{x}) = \sum_{i=1}^m \lambda_i \partial f_i(\mathbf{x}).$$

Clarke donne une définition équivalente de la sous-différentielle, définition qui sera utile pour en construire une approximation : elle est basée sur le théorème de Rademacher [33] qui dit qu'une fonction lipschitzienne sur un ensemble $E \subseteq \mathbb{R}^n$ est différentiable presque partout sur E .

Définition 29. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction continue localement lipschitz et Q un sous-ensemble de \mathbb{R}^n contenu dans un voisinage de $\mathbf{x} \in \mathbb{R}^n$ où f est différentiable.

Alors

$$\partial^{\mathcal{C}} f(\mathbf{x}) = \text{Conv} \left(\left\{ \lim_{\mathbf{x}_r \rightarrow \mathbf{x}} \nabla f(\mathbf{x}_r) : \mathbf{x}_r \in Q \right\} \right).$$

Construction numérique de la sous-différentielle de Clarke :

À partir de cette définition de la sous-différentielle de Clarke, Burke [11] a proposé une approche numérique permettant de construire une approximation de cette sous-différentielle. Tout d'abord, la sous-différentielle de Clarke peut-être définie de la manière équivalente suivante ; soit $\mathcal{B} = \{\mathbf{x} : \|\mathbf{x}\| \leq 1\}$ la boule unité, et $\epsilon > 0$. On considère l'ensemble convexe

$$G_\epsilon(\mathbf{x}) = \text{Conv} (\{\nabla f(\mathbf{z}), \mathbf{z} \in (\mathbf{x} + \epsilon\mathcal{B}) \cap Q\}),$$

alors la sous-différentielle de Clarke peut s'écrire encore

$$\partial^{\mathcal{C}} f(\mathbf{x}) = \bigcap_{\epsilon > 0} G_\epsilon(\mathbf{x}).$$

Pour $\delta > 0$ fixé, la démarche proposée consiste alors à effectuer des tirages x_i , $i = 1, \dots, k$ uniformément et indépendamment dans l'ensemble $(\mathbf{x} + \delta\mathcal{B}) \cap Q$, puis de construire la suite des gradients $G_i = \nabla f(\mathbf{x}_i)$, ainsi que la suite croissante d'ensembles convexes fermés

$$\mathcal{C}_k = \text{Conv} (\{G_1, \dots, G_k\}).$$

Théorème 4. *Si f est une fonction localement lipschitzienne au voisinage de \mathbf{x} alors pour δ suffisamment petit et pour tout $\epsilon > 0$ on a*

$$\lim_{k \rightarrow +\infty} \mathbb{P} \left(\{\partial^{\mathcal{C}} f(\mathbf{x}) \subset \mathcal{C}_k + \epsilon\mathcal{B}\} \right) = 1$$

La Figure 2.2 illustre la construction d'une approximation de la sous-différentielle de Clarke pour une fonction non-différentiable au point x_0 . Un voisinage de rayon ϵ est choisi autour de x_0 , et le gradient de la fonction est calculé pour des points x_i du voisinage tirés aléatoirement. L'enveloppe convexe de ces échantillons constitue l'approximation utilisée de la sous-différentielle. Sur cet exemple simple on peut observer une différence entre la vraie sous-différentielle tracée en vert, et son approximation en orange.

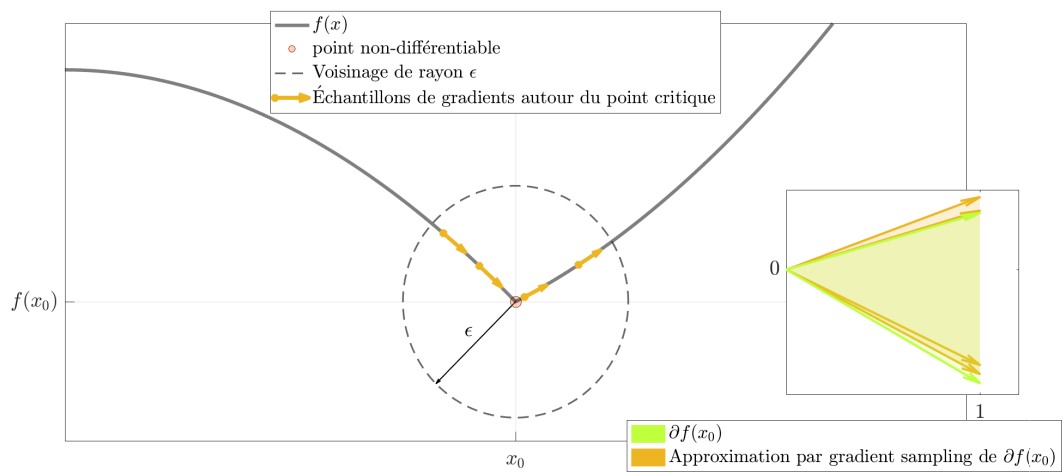


FIGURE 2.2 Exemple de calcul d'une sous-différentielle d'une fonction non-convexe

Chapitre 3

Optimisation multiobjectif et vecteur de descente commun

Dans un premier temps, on rappelle la formulation mathématique d'un problème d'optimisation de manière générale, ainsi que la relation de domination selon Pareto qui est utilisée dans le cadre de cette thèse en vue de faire de l'optimisation multiobjectif sans *a priori*. Les résultats et notions sur la relation de dominance de Pareto sont tirés de l'ouvrage d'Ehrgott [30].

3.1 Relations binaires

L'optimisation correspond à la recherche du ou des meilleurs éléments d'un ensemble, le terme meilleur devant être précisé. Afin de pouvoir identifier les éléments considérés comme les plus performants, il faut pouvoir comparer chacun des éléments présents dans l'ensemble de recherche. Cette partie rappelle les outils mathématiques qui permettent de mettre en place les méthodes de comparaisons entre les éléments d'un ensemble. Dans un premier temps il est nécessaire de définir la notion de relation entre éléments d'un ensemble ou de plusieurs ensembles.

Définition 30. Une relation binaire \mathcal{R} d'un ensemble \mathcal{X} entre \mathcal{Y} est définie par une partie $\mathcal{G} \subseteq \mathcal{X} \times \mathcal{Y}$. On dit qu'un élément $x \in \mathcal{X}$ est en relation avec $y \in \mathcal{Y}$ si le couple $(x, y) \in \mathcal{G}$ et on note

$$x\mathcal{R}y.$$

Lorsque $\mathcal{X} = \mathcal{Y}$ on dit que \mathcal{R} est une relation sur \mathcal{X}

Définition 31. Soit \mathcal{R} une relation sur un ensemble \mathcal{X} , on dit que \mathcal{R} est

1. réflexive (resp. irreflexive) si : $\forall x \in \mathcal{X}, x\mathcal{R}x$ (resp. $x\not\mathcal{R}x$),
2. symétrique si : $\forall(x, y) \in \mathcal{X}^2, x\mathcal{R}y \Rightarrow y\mathcal{R}x$,
3. faiblement antisymétrique si : $\forall(x, y) \in \mathcal{X}^2, x\mathcal{R}y \wedge y\mathcal{R}x \Rightarrow x = y$,
4. fortement antisymétrique si : $\forall(x, y) \in \mathcal{X}^2, x\mathcal{R}y \Rightarrow y\not\mathcal{R}x$,
5. transitive si : $\forall(x, y, z) \in \mathcal{X}^3, x\mathcal{R}y \wedge y\mathcal{R}z \Rightarrow x\mathcal{R}z$,
6. totale si : $\forall(x, y) \in \mathcal{X}^2, x\mathcal{R}y \vee y\mathcal{R}x$,

où \vee (resp. \wedge) désigne le "ou" (resp. le "et") logique.

À partir des relations binaires, il est possible de construire des relations de comparaison entre différents éléments d'un ensemble donné.

Définition 32. On dit qu'une relation binaire \mathcal{R} sur \mathcal{X} définit :

1. un préordre sur \mathcal{X} lorsque la relation \mathcal{R} est réflexive et transitive. On dit du couple $(\mathcal{X}, \mathcal{R})$ qu'il constitue un espace préordonné. On utilise la notation $\preceq_{\mathcal{X}}$ pour désigner un préordre sur \mathcal{X} ,
2. une équivalence sur \mathcal{X} lorsque la relation \mathcal{R} est réflexive, transitive et symétrique et sera notée $\equiv_{\mathcal{X}}$,
3. un ordre sur \mathcal{X} lorsque la relation \mathcal{R} est réflexive, transitive et antisymétrique. On dit du couple $(\mathcal{X}, \mathcal{R})$ qu'il constitue un espace ordonné. La notation $\leq_{\mathcal{X}}$ sera utilisée,
4. un ordre strict sur \mathcal{X} lorsque la relation \mathcal{R} est irreflexive, transitive et antisymétrique et la notation $<_{\mathcal{X}}$ sera utilisée.

L'optimisation utilise naturellement la notion d'ordre et de préordre afin de comparer différents éléments d'un ensemble et d'en déduire quels sont les éléments les plus performants. Il est clair que les éléments d'un ensemble considérés comme les plus performants dépendront fortement des relations binaires utilisées : différentes relations de comparaison donneront différents éléments les plus performants.

Dans la partie suivante, on introduit formellement le principe de l'optimisation et la notion de minimum d'un ensemble ordonné et préordonné. En particulier, on présente le préordre de Pareto qui est une des relations binaires utilisées dans le cadre de l'optimisation multiobjectif.

3.2 Optimisation multiobjectif, préordre de Pareto

Un problème d'optimisation peut toujours être représenté mathématiquement comme un problème de minimisation d'une fonction appelée *fonction coût* f définie sur un ensemble \mathcal{X} appelé *espace des variables* et à valeur dans \mathcal{Y} appelé *espace des objectifs*

$$\begin{aligned} f &: \mathcal{X} \rightarrow \mathcal{Y} \\ \mathbf{x} &\mapsto f(\mathbf{x}). \end{aligned}$$

L'espace $\mathcal{X} \subseteq \mathcal{X}$ sur lequel la minimisation est réalisée correspond aux *solutions admissibles* du problème; les solutions prenant en compte les contraintes éventuelles. On note $\mathcal{Y} \subseteq \mathcal{Y}$ l'image par la *fonction coût* f de l'ensemble \mathcal{X} et on appelle cet espace l'*ensemble réalisable*. Minimiser f sur \mathcal{X} revient en fait à trouver l'ensemble des minima de \mathcal{Y} muni d'un ordre $\leq_{\mathcal{Y}}$

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) = \min(\mathcal{Y}, \leq_{\mathcal{Y}}), \quad (3.1)$$

et à identifier les éléments de \mathcal{X} dont les images sont des minima de $(\mathcal{Y}, \leq_{\mathcal{Y}})$

$$\operatorname{argmin}_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) = \{\mathbf{x} \in \mathcal{X}; f(\mathbf{x}) \in \min(\mathcal{Y}, \leq_{\mathcal{Y}})\}.$$

Définition 33. Soit f une fonction de $\mathcal{X} \subseteq \mathcal{X}$ à valeurs dans $\mathcal{Y} \subseteq \mathcal{Y}$ muni d'un ordre $\leq_{\mathcal{Y}}$ total. L'ensemble des éléments de \mathcal{Y} qui minimisent l'espace ordonné $(\mathcal{Y}, \leq_{\mathcal{Y}})$ est donné par

$$\min(\mathcal{Y}, \leq_{\mathcal{Y}}) = \{\mathbf{y}^* \in \mathcal{Y}; \forall \mathbf{y} \in \mathcal{Y}, \mathbf{y}^* \leq_{\mathcal{Y}} \mathbf{y}\}.$$

Dans le cas où la *fonction coût* f est à valeur dans \mathbb{R} , on lui associe généralement l'ordre usuel \leq des nombres réels. On parle alors d'optimisation mono-objectif.

Lorsque la *fonction coût* f est à valeurs dans \mathbb{R}^m , on associe à chaque composante f_i de f une fonction à valeur dans \mathbb{R} que l'on appelle *fonction objectif*.

$$\begin{aligned} \forall i \in \llbracket 1, m \rrbracket, f_i &: \mathbb{R}^n \rightarrow \mathbb{R} \\ \mathbf{x} &\mapsto f_i(\mathbf{x}). \end{aligned}$$

On parle alors d'optimisation multiobjectif et on peut interpréter le problème (3.1) sous la forme

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) = \min_{\mathbf{x} \in \mathcal{X}} \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})\}. \quad (3.2)$$

L'extension directe de la Définition (33) dans le cas d'une fonction $f = (f_1, \dots, f_m)$ à valeurs dans \mathbb{R}^m avec m un entier strictement supérieur à 1 n'est pas possible puisque

l'ordre produit résultant $\prod^m(\mathbb{R}, \leq)$ n'est pas un ordre total. Une première approche en optimisation multiobjectif est de se ramener dans un espace \mathcal{Z} par l'intermédiaire d'une fonction

$$\begin{aligned} \theta &: \mathcal{Y} \rightarrow \mathcal{Z} \\ f(\mathbf{x}) &\mapsto \theta \circ f(\mathbf{x}), \end{aligned}$$

pour lequel un ordre total $\leq_{\mathcal{Z}}$ existe. La manière la plus classique étant d'utiliser une somme pondérée des objectifs pour se ramener à une *fonction coût* à valeurs réelles. Cela permet de conserver l'ordre \leq des nombres réels. Le problème (3.2) peut alors se réécrire

$$\min_{\mathbf{x} \in \mathcal{X}} (\theta \circ f(\mathbf{x})) = \min_{\mathbf{x} \in \mathcal{X}} \left(\sum_{i=1}^m p_i f_i(\mathbf{x}) \right), \quad (3.3)$$

où $p = (p_1, \dots, p_m) \in [0, 1]^m$ est le vecteur des pondérations tel que $\sum_{i=1}^m p_i = 1$. Cependant, la réduction du problème multiobjectif à un problème mono-objectif muni de l'ordre usuel \leq implique que des préférences entre les objectifs aient été ajoutées par le choix des poids p_i . Lorsque le problème multiobjectif est modifié par une fonction θ afin de se ramener à un nouvel espace totalement ordonné, on parle d'optimisation *a priori*.

Si on cherche à conserver complètement la nature multiobjectif du problème initial (3.2), un nouvel ordre permettant de comparer les *solutions réalisables* doit être construit. On peut dans un premier temps considérer qu'une solution est meilleure qu'une autre si elle est aussi bonne sur l'ensemble des objectifs et est strictement meilleure pour au moins un objectif du problème : c'est la dominance de Pareto.

Définition 34. *Pour un problème d'optimisation de la forme (3.2), on dit qu'un point de l'espace des variables d'optimisation \mathbf{x}_1 domine au sens de Pareto un point \mathbf{x}_2 si le point \mathbf{x}_1 est au moins aussi bon que \mathbf{x}_2 sur l'ensemble des objectifs et si \mathbf{x}_1 est meilleur que \mathbf{x}_2 pour au moins un objectif :*

$$\begin{cases} \forall i \in \llbracket 1, m \rrbracket, f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2) \\ \exists \ell \in \llbracket 1, m \rrbracket, f_\ell(\mathbf{x}_1) < f_\ell(\mathbf{x}_2) \end{cases} \quad (3.4)$$

Soient $(\mathbf{x}_1, \mathbf{x}_2)$ deux *solutions admissibles* et $(\mathbf{y}_1, \mathbf{y}_2)$ leurs images respectives par la fonction coût f à valeurs dans \mathbb{R}^m . Si \mathbf{x}_1 domine au sens de Pareto le point \mathbf{x}_2 , on note

$$\mathbf{x}_1 \stackrel{f}{\prec}_{\mathcal{P}} \mathbf{x}_2,$$

et dans l'espace des objectifs, on note de manière équivalente à l'aide du symbole $\prec_{\mathcal{P}}$

$$f(\mathbf{x}_1) \prec_{\mathcal{P}} f(\mathbf{x}_2).$$

On obtient alors directement

$$\mathbf{y}_1 \prec_{\mathcal{P}} \mathbf{y}_2.$$

Proposition 4. *La relation binaire $\prec_{\mathcal{P}}$ de la dominance selon Pareto définit un ordre strict dans l'espace \mathbb{R}^m pour le problème d'optimisation (3.2).*

Démonstration. Soit \mathbf{x} un point de \mathbb{R}^n , et \mathbf{y} son image par la fonction coût f

$$f_i(\mathbf{x}) = f_i(\mathbf{x}), \forall i \in \llbracket 1, m \rrbracket.$$

Alors, il n'existe pas de $\ell \in \llbracket 1, m \rrbracket$ tel que

$$f_{\ell}(\mathbf{x}) < f_{\ell}(\mathbf{x}),$$

donc

$$\mathbf{y} \not\prec_{\mathcal{P}} \mathbf{y}$$

La relation binaire $\prec_{\mathcal{P}}$ est donc irréflexive.

Soient $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \in (\mathbb{R}^n)^3$ et $(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3) \in (\mathbb{R}^m)^3$ leurs images respectives par la fonction coût f tels que $\mathbf{y}_1 \prec_{\mathcal{P}} \mathbf{y}_2$ et $\mathbf{y}_2 \prec_{\mathcal{P}} \mathbf{y}_3$. Alors, il existe (ℓ_{12}, ℓ_{23}) tels que

$$\left\{ \begin{array}{l} \forall i \in \llbracket 1, m \rrbracket; f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2) \\ \exists \ell_{12} \in \llbracket 1, m \rrbracket; f_{\ell_{12}}(\mathbf{x}_1) < f_{\ell_{12}}(\mathbf{x}_2) \end{array} \right. \quad \text{et} \quad \left\{ \begin{array}{l} \forall i \in \llbracket 1, m \rrbracket; f_i(\mathbf{x}_2) \leq f_i(\mathbf{x}_3) \\ \exists \ell_{23} \in \llbracket 1, m \rrbracket; f_{\ell_{23}}(\mathbf{x}_2) < f_{\ell_{23}}(\mathbf{x}_3). \end{array} \right.$$

Par conséquent,

$$\left\{ \begin{array}{l} \forall i \in \llbracket 1, m \rrbracket; f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_3) \\ \exists \ell_{23} \in \llbracket 1, m \rrbracket; f_{\ell_{23}}(\mathbf{x}_1) \leq f_{\ell_{23}}(\mathbf{x}_2) < f_{\ell_{23}}(\mathbf{x}_3), \end{array} \right.$$

donc finalement

$$\mathbf{y}_1 \prec_{\mathcal{P}} \mathbf{y}_3.$$

La relation binaire $\prec_{\mathcal{P}}$ est donc transitive. \square

En utilisant ce principe de dominance il n'est plus possible d'ordonner toutes les solutions les unes par rapport aux autres, et on cherche alors à déterminer les *solutions*

admissibles telles que leurs images par la fonction coût f ne soient pas dominées dans l'espace des objectifs

$$\mathcal{P}_O = \{\mathbf{y}^* \in \mathcal{Y}; \nexists \mathbf{y} \in \mathcal{Y}, \mathbf{y} \prec_P \mathbf{y}^*\}. \quad (3.5)$$

On appelle *front de Pareto* l'ensemble des points réalisables non-dominés. L'image réciproque du *front de Pareto* par la fonction coût f est appelé ensemble des *solutions de Pareto* et est noté \mathcal{P}_D :

$$\mathcal{P}_D = \{\mathbf{x} \in \mathcal{X}; f(\mathbf{x}) \in \mathcal{P}_O\}.$$

L'ensemble des *solutions de Pareto* correspond donc à l'ensemble des *solutions admissibles* tel qu'il n'existe aucune autre *solution admissible* meilleure simultanément sur l'ensemble des objectifs constituant le problème d'optimisation : il est impossible d'améliorer une *solution de Pareto* sur un des objectifs sans en dégrader un autre. Une manière d'interpréter les solutions est de dire que le *front de Pareto* donne accès à l'ensemble des meilleurs compromis entre les différents objectifs du problème d'optimisation. Une fois le front de Pareto déterminé, il devient possible de choisir une solution particulière en privilégiant tel ou tel objectif. On parle alors d'optimisation *a posteriori*.

En guise d'illustration de la notion de dominance de Pareto, on peut observer, sur la Figure 3.1, quelques points appartenant à un *ensemble réalisable* d'un problème d'optimisation biobjectif

$$\min_{\mathbf{x}} \{f_1, f_2\}.$$

En prenant pour référence le point \mathbf{y}_1 , et en utilisant la définition de la dominance de Pareto, il est possible de déterminer les régions pour lesquelles :

- tout point \mathbf{y} est dominé par \mathbf{y}_1 (■),
- tout point \mathbf{y} domine \mathbf{y}_1 (■).

Sur cette figure, il est aussi possible de trouver les points non-dominés ■ pour l'échantillon de l'*ensemble réalisable* représenté.

Il n'est pas possible, à partir de la relation de dominance de Pareto, de créer un ordre compatible avec la Définition (33) d'un minimum. Une seconde définition d'un minimum, plus générale, sur un espace préordonné permet de contourner cette difficulté.

Définition 35. Soit f une fonction de $\mathcal{X} \subseteq \mathcal{X}$ à valeurs dans $\mathcal{Y} \subseteq \mathcal{Y}$ muni d'un préordre $\preceq_{\mathcal{Y}}$. L'ensemble des éléments de \mathcal{Y} qui minimisent l'espace préordonné $(\mathcal{Y}, \preceq_{\mathcal{Y}})$

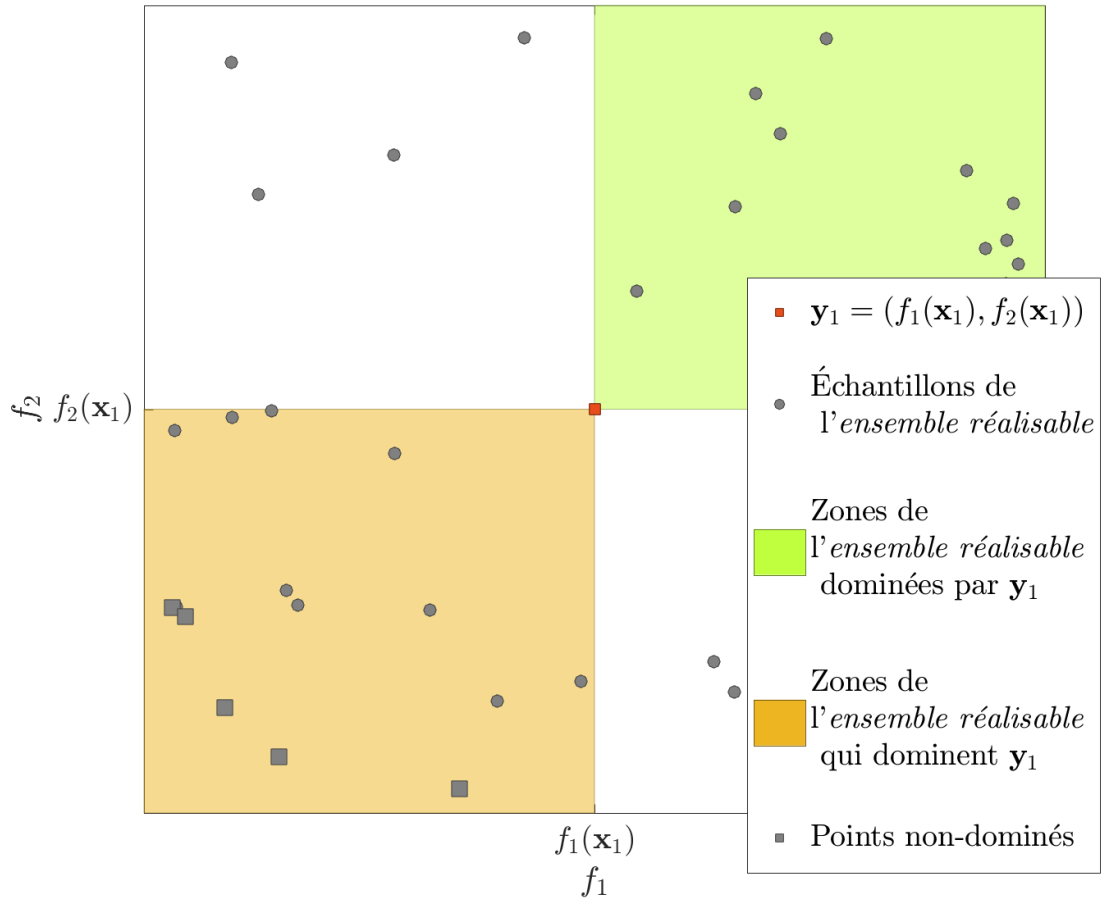


FIGURE 3.1 Exemple de la domination selon Pareto pour un problème biobjectif $\min_{\mathbf{x}} \{f_1(\mathbf{x}), f_2(\mathbf{x})\}$

est donné par

$$\min(\mathcal{Y}, \preceq_{\mathcal{Y}}) = \{\mathbf{y}^* \in \mathcal{Y}; \forall \mathbf{y} \in \mathcal{Y}, \mathbf{y} \preceq_{\mathcal{Y}} \mathbf{y}^* \Rightarrow \mathbf{y} \equiv_{\mathcal{Y}} \mathbf{y}^*\}.$$

Définition 36. Soient f une fonction de \mathcal{X} dans \mathcal{Y} , $\mathcal{X} \subseteq \mathcal{X}$ l'ensemble des solutions admissibles du problème d'optimisation associé, $\mathcal{Y} \subseteq \mathcal{Y}$ l'image de \mathcal{X} par f et $\preceq_{\mathcal{Y}}$ un préordre sur \mathcal{Y} . On dit que \mathbf{x}^* est un minimum local de f sur \mathcal{X} pour le préordre $\preceq_{\mathcal{Y}}$ si et seulement si il existe une boule ouverte $\mathcal{B}(\mathbf{x}^*, \rho)$ de centre \mathbf{x}^* et de rayon $\rho > 0$ telle que

$$\operatorname{argmin}(\mathcal{Y}, \preceq_{\mathcal{Y}}) = \{\mathbf{x}^* \in \mathcal{X}; \forall \mathbf{x} \in \mathcal{X} \cap \mathcal{B}(\mathbf{x}^*, \rho), f(\mathbf{x}) \preceq_{\mathcal{Y}} f(\mathbf{x}^*) \Rightarrow f(\mathbf{x}) \equiv_{\mathcal{Y}} f(\mathbf{x}^*)\},$$

Dans la suite, on cherche à construire un préordre dérivé de la dominance de Pareto compatible avec la Définition (35). On crée une nouvelle relation binaire notée $\preceq_{\mathcal{P}}$ telle que pour tout éléments $(\mathbf{y}_1, \mathbf{y}_2)$, la relation $\mathbf{y}_1 \preceq_{\mathcal{P}} \mathbf{y}_2$ implique que \mathbf{y}_1 domine \mathbf{y}_2 selon Pareto ou alors que $\mathbf{y}_1 = \mathbf{y}_2$

$$\forall (\mathbf{y}_1, \mathbf{y}_2) \in \mathcal{Y}^2, \mathbf{y}_1 \preceq_{\mathcal{P}} \mathbf{y}_2 \implies \begin{cases} \mathbf{y}_1 \prec_{\mathcal{P}} \mathbf{y}_2 \\ \text{ou} \\ \mathbf{y}_1 = \mathbf{y}_2 \end{cases}$$

Proposition 5. *La relation binaire $\preceq_{\mathcal{P}}$ est un préordre sur \mathbb{R}^m*

Démonstration. La relation d'égalité $=$ donne la réflexivité à $\preceq_{\mathcal{P}}$.

Les relations $\prec_{\mathcal{P}}$ et $=$ étant transitives et $=$ étant symétrique, l'union des deux relations binaires conserve la propriété de transitivité. \square

Remarque 2. *Le préordre issue de la dominance de Pareto et de la relation d'égalité correspond en fait à l'ordre produit de \mathbb{R}^m*

Proposition 6. *Soit f une fonction coût à valeur dans \mathbb{R}^m , et soit $\preceq_{\mathcal{P}}$ le préordre de Pareto sur \mathbb{R}^m .*

1. *Tout élément du front de Pareto \mathcal{P}_O défini par la relation (3.5) est un élément minimum de $(\mathbb{R}^m, \preceq_{\mathcal{P}})$,*
2. *tout élément qui minimise $(\mathbb{R}^m, \preceq_{\mathcal{P}})$ appartient au front de Pareto \mathcal{P}_O .*

Démonstration.

1. Soit \mathbf{y}^* un élément du front de Pareto \mathcal{P}_O . Supposons qu'il existe \mathbf{y} tel que $\mathbf{y} \preceq_{\mathcal{P}} \mathbf{y}^*$, c'est à dire que

$$\mathbf{y} \prec_{\mathcal{P}} \mathbf{y}^* \text{ ou } \mathbf{y} = \mathbf{y}^*.$$

Or $\mathbf{y}^* \in \mathcal{P}_O$, donc il n'existe pas de \mathbf{y} tel que $\mathbf{y} \prec_{\mathcal{P}} \mathbf{y}^*$.

Nécessairement, on a $\mathbf{y} = \mathbf{y}^*$ et donc $\mathbf{y}^* \in \min(\mathbb{R}^m, \preceq_{\mathcal{P}})$

2. Soit $\mathbf{y}^* \in \min(\mathbb{R}^m, \preceq_{\mathcal{P}})$. Par définition

$$\forall \mathbf{y} \in \mathbb{R}^m, \mathbf{y} \preceq_{\mathcal{P}} \mathbf{y}^*,$$

c'est à dire

$$\mathbf{y} \prec_{\mathcal{P}} \mathbf{y}^* \text{ ou } \mathbf{y} = \mathbf{y}^*.$$

Mais \mathbf{y}^* est un élément qui minimise $(\mathbb{R}^m, \preceq_{\mathcal{P}})$ donc nécessairement

$$\mathbf{y} = \mathbf{y}^*.$$

Finalement, on peut déduire qu'il n'existe pas de \mathbf{y} de \mathcal{Y} tel que

$$\mathbf{y} \prec_{\mathcal{P}} \mathbf{y}^*.$$

□

Lorsque deux points $(\mathbf{y}_1, \mathbf{y}_2)$ de l'ensemble réalisable sont incomparables, c'est à dire que

$$\mathbf{y}_1 \not\prec_{\mathcal{P}} \mathbf{y}_2 \text{ et } \mathbf{y}_2 \not\prec_{\mathcal{P}} \mathbf{y}_1, \quad (3.6)$$

on dit par abus de langage que les deux points \mathbf{y}_1 et \mathbf{y}_2 sont équivalents au sens de Pareto bien que la relation binaire décrite en (3.6) n'étant pas transitive ne soit pas une relation d'équivalence. On note cette relation

$$\mathbf{y}_1 \equiv_{\mathcal{P}} \mathbf{y}_2.$$

Proposition 7. *Les éléments du front de Pareto \mathcal{P}_O sont incomparables entre eux.*

Démonstration. Soient $(\mathbf{y}_1, \mathbf{y}_2)$ deux éléments du front de Pareto \mathcal{P}_O . D'après la Proposition (6) \mathbf{y}_1 est un élément de $\min(\mathbb{R}^m, \preceq_{\mathcal{P}})$, c'est à dire que

$$\forall \mathbf{y} \in \mathbb{R}^m, \mathbf{y} \preceq_{\mathcal{P}} \mathbf{y}_1 \Rightarrow \mathbf{y} = \mathbf{y}_1.$$

Par conséquent, \mathbf{y} ne domine pas \mathbf{y}_1

$$\mathbf{y} \not\prec_{\mathcal{P}} \mathbf{y}_1.$$

En particulier $\mathbf{y}_2 \not\prec_{\mathcal{P}} \mathbf{y}_1$.

□

Résoudre un problème de minimisation d'une fonction coût à valeurs dans \mathbb{R}^m muni du préordre de Pareto correspond donc à trouver l'ensemble des solutions incomparables telles qu'elles ne puissent pas être dominées par un élément de l'ensemble réalisable.

La partie suivante du chapitre a pour but de définir la notion de vecteur de descente commun. On propose notamment plusieurs méthodes afin de construire un tel vecteur selon la régularité des fonctions objectifs. On considère dans la suite un problème d'optimisation décrit par la relation (3.2), où la fonction coût $f = (f_1, \dots, f_m)$ est une fonction de $\mathcal{X} \subseteq \mathbb{R}^n$ à valeurs dans $\mathcal{Y} \subseteq \mathbb{R}^m$ muni du préordre de Pareto $\preceq_{\mathcal{P}}$ sur \mathcal{Y} .

3.3 Vecteur de descente commun

Dans cette partie du chapitre, on s'attache à donner les définitions et les différentes propriétés des vecteurs de descente commun qui sont des ingrédients essentiels aux algorithmes proposés dans la suite du document.

On suppose que les fonctions objectifs f_i sont au moins f° -pseudo-convexes et localement lipschitziennes.

Définition 37. Soit \mathbf{d} un vecteur de \mathbb{R}^n , on dit que \mathbf{d} est vecteur de descente de la fonction f_i , $i \in \llbracket 1, m \rrbracket$, au point $\mathbf{x} \in \mathcal{X}$ si la fonction f_i décroît de manière monotone sur un segment de direction \mathbf{d}

$$\exists \ell_0 \in \mathbb{R}^+, \forall \ell \in [0, \ell_0]; f_i(\mathbf{x} + \ell \mathbf{d}) \leq f_i(\mathbf{x}).$$

Proposition 8. Soit \mathbf{d} un vecteur de descente pour f_i , $i \in \llbracket 1, m \rrbracket$, au point $\mathbf{x} \in \mathcal{X}$.

Pour une fonction f_i° -pseudoconvexe la dérivée directionnelle généralisée suivant la direction \mathbf{d} est négative

$$f_i^\circ(\mathbf{x}, \mathbf{d}) \leq 0.$$

Démonstration. Le résultat est immédiat de part le Théorème 2. □

Définition 38. Soit \mathbf{d} un vecteur de \mathbb{R}^n , on dit que \mathbf{d} est vecteur de descente commun aux fonctions $\{f_i\}$ avec $i \in \llbracket 1, m \rrbracket$ en $\mathbf{x} \in \mathcal{X}$ si les fonctions f_i décroissent toutes de manière monotone sur un segment de direction \mathbf{d}

$$\exists \ell_0 \in \mathbb{R}^+, \forall \ell \in [0, \ell_0]; f_i(\mathbf{x} + \ell \mathbf{d}) \leq f_i(\mathbf{x}), \forall i \in \llbracket 1, m \rrbracket.$$

On dénote l'ensemble des vecteurs de descente commun au point \mathbf{x} par $\mathcal{D}(\mathbf{x})$

Proposition 9. Soit \mathbf{d} un vecteur de descente commun aux fonctions $\{f_i\}$ avec $i \in \llbracket 1, m \rrbracket$ et au point $\mathbf{x} \in \mathcal{X}$. Les dérivées directionnelles généralisées des $\{f_i\}$ suivant la direction \mathbf{d} sont négatives

$$\forall i \in \llbracket 1, m \rrbracket, f_i^\circ(\mathbf{x}, \mathbf{d}) \leq 0.$$

Démonstration. Voir la preuve de la Proposition 8 □

Dans la suite, on utilise la notation $\mathcal{C}(\mathbf{x})$ pour désigner l'enveloppe convexe de l'union des sous-différentielles calculées au point \mathbf{x}

$$\mathcal{C}(\mathbf{x}) = \text{Conv} \left(\bigcup_{i=1}^m \partial^{\mathcal{C}} f_i(\mathbf{x}) \right). \quad (3.7)$$

Proposition 10. *Soit \mathbf{x} une solution admissible et $\mathcal{C}(\mathbf{x})$ l'enveloppe convexe de l'union des sous-différentielles de Clarke. Tout élément de $\mathcal{C}(\mathbf{x})$ peut s'écrire comme une combinaison convexe d'un élément de chacune des sous-différentielles.*

$$\forall \mathbf{c} \in \mathcal{C}(\mathbf{x}), \exists \{\Lambda_i\}_{i \in [1, m]}, \mathbf{c} = \sum_{i=1}^m \Lambda_i T^i, T^i \in \partial^{\mathcal{C}} f_i(\mathbf{x}),$$

où la somme des Λ_i sont positifs ou nuls et de somme égale à 1, les $\Lambda_i \geq 0$ positifs.

Démonstration. En séparant l'union des sous-différentielles de la manière qui suit,

$$\mathcal{C}(\mathbf{x}) = \text{Conv} \left(\bigcup_{i=1}^m \partial^{\mathcal{C}} f_i(\mathbf{x}) \right) = \text{Conv} \left(\partial^{\mathcal{C}} f_1(\mathbf{x}) \cup \left(\bigcup_{i=2}^m \partial^{\mathcal{C}} f_i(\mathbf{x}) \right) \right),$$

un élément \mathbf{c} de $\mathcal{C}(\mathbf{x})$ peut s'écrire comme

$$\sum_i \alpha_i \tau_i^1 + \sum_i \beta_i \theta_i$$

où $\tau_i^1 \in \partial^{\mathcal{C}} f_1(\mathbf{x})$ et $\theta_i \in \bigcup_{i=2}^m \partial^{\mathcal{C}} f_i(\mathbf{x})$. De plus les α_i et les β_i sont des réels positifs dont la somme vaut 1

$$\sum_i \alpha_i + \sum_i \beta_i = 1.$$

Si la somme $\sum \alpha_i$ n'est pas nulle, on peut écrire

$$\sum_i \alpha_i \tau_i^1 = \sum_i \alpha_i \frac{\sum_i \alpha_i \tau_i^1}{\sum_i \alpha_i}.$$

Or la sous-différentielle $\partial^{\mathcal{C}} f_1(\mathbf{x})$ est un ensemble convexe par définition, donc on sait que

$$T^1 = \frac{\sum_i \alpha_i \tau_i^1}{\sum_i \alpha_i} \in \partial^{\mathcal{C}} f_1(\mathbf{x}).$$

Si la somme des α_i est nulle, on pose $T^1 = \mathbf{0}$. Par conséquent, on obtient

$$\mathbf{c} = \sum_i \alpha_i T^1 + \sum_i \beta_i \theta_i = \Lambda_1 T^1 + \sum_i \beta_i \theta_i$$

Le résultat s'en déduit par récurrence. \square

Définition 39. *On dit qu'un point \mathbf{x}^* de l'espace des variables d'optimisation est un point Pareto stationnaire si le vecteur nul appartient à l'enveloppe convexe des sous-différentielles de Clarke*

$$\mathbf{0} \in \mathcal{C}(\mathbf{x}^*).$$

Remarque 3. *Lorsque les fonctions objectifs sont convexes, alors la définition de la Pareto stationnarité devient*

$$\mathbf{0} \in \text{Conv} \left(\bigcup_{i=1}^m \partial f_i(\mathbf{x}) \right).$$

De plus, lorsque les fonctions sont convexes et différentiables, on obtient

$$\mathbf{0} \in \text{Conv} \left(\{\nabla f_i(\mathbf{x})\}_{i \in \llbracket 1, m \rrbracket} \right).$$

Définition 40. *On dit d'un vecteur \mathbf{x}^* de \mathbb{R}^n qu'il est une solution faible de Pareto pour le problème d'optimisation (3.2), s'il n'existe pas de \mathbf{x} de \mathbb{R}^n tel que*

$$f_i(\mathbf{x}) < f_i(\mathbf{x}^*), \quad \forall i \in \llbracket 1, m \rrbracket.$$

De façon évidente, on peut d'ores et déjà dire que tout point Pareto solution est aussi une solution faible de Pareto.

Théorème 5. (Mäkelä et al. [48])

Si un point \mathbf{x}^ est une solution faible de Pareto locale, alors le vecteur nul appartient à l'enveloppe convexe des sous-différentielles de Clarke*

$$\mathbf{0} \in \mathcal{C}(\mathbf{x}^*). \tag{3.8}$$

Lorsque les fonctions objectifs sont toutes au moins f° -pseudo-convexes, alors la condition (3.8) devient une condition suffisante pour avoir une solution faible de Pareto globale.

Lemme 2. *Soit \mathbf{x} une solution admissible. Il existe un unique élément de norme minimale dans $\mathcal{C}(\mathbf{x})$. De plus, l'élément de norme minimale $\xi = \operatorname{argmin} \{\|\mathbf{v}\|, \mathbf{v} \in \mathcal{C}(\mathbf{x})\}$ vérifie la propriété suivante*

$$\forall \mathbf{c} \in \mathcal{C}(\mathbf{x}), \quad \langle \mathbf{c}, \xi \rangle \geq \|\xi\|^2.$$

Démonstration. L'existence et l'unicité de l'élément de norme minimale ξ est assuré de part le fait que $\mathcal{C}(\mathbf{x})$ est un ensemble convexe fermé.

Soit \mathbf{c} un élément de l'enveloppe $\mathcal{C}(\mathbf{x})$, on pose $\mathbf{h} = \mathbf{c} - \xi$. Puisque $\mathcal{C}(\mathbf{x})$ est convexe

$$\forall \epsilon \in [0, 1], \xi + \epsilon \mathbf{h} \in \mathcal{C}(\mathbf{x}).$$

Nécessairement, par définition de ξ , on a $\|\xi + \epsilon \mathbf{h}\|^2 \geq \|\xi\|^2$, ce qui peut s'écrire

$$\|\xi + \epsilon \mathbf{h}\|^2 - \|\xi\|^2 = 2\epsilon \langle \mathbf{h}, \xi \rangle + \epsilon^2 \|\mathbf{h}\|^2 \geq 0.$$

Le paramètre ϵ pouvant être aussi petit que l'on veut, cela requière

$$\langle \mathbf{h}, \xi \rangle = \langle \mathbf{c} - \xi, \xi \rangle \geq 0.$$

□

Théorème 6. Soient \mathbf{x} une solution admissible et $\mathcal{C}(\mathbf{x})$ l'enveloppe convexe de l'union des sous-différentielles de Clarke. L'opposé de l'élément de norme minimale $-\xi$ dans $\mathcal{C}(\mathbf{x})$, s'il est non nul, est un vecteur de descente commun.

$$\xi = \operatorname{argmin} \{\|\mathbf{c}\|, \mathbf{c} \in \mathcal{C}(\mathbf{x})\}. \quad (3.9)$$

Démonstration. Soit ξ l'élément de norme minimale dans $\mathcal{C}(\mathbf{x})$. D'après la Proposition 10, il existe une famille $\{\tau_i\}_{i \in [1, m]}$ composée d'un élément de chaque sous-différentielle telle que

$$\xi = \sum_{i=1}^m \alpha_i \tau_i, \quad \sum_i \alpha_i = 1, \quad \alpha_i \geq 0.$$

Le Lemme 2 nous donne

$$\forall \mathbf{c} \in \mathcal{C}(\mathbf{x}), \quad \langle \mathbf{c}, \xi \rangle \geq \|\xi\|^2 > 0.$$

Donc en particulier pour tous les éléments des sous-différentielles, on obtient

$$\forall i \in [1, m], \quad \forall \theta_i \in \partial^{\mathcal{C}} f_i(\mathbf{x}), \quad \langle \theta_i, -\xi \rangle < 0.$$

Donc $-\xi$ est une direction de descente commune à l'ensemble des objectifs au point \mathbf{x} . □

À partir de l'enveloppe convexe de l'union des sous-différentielles de Clarke, il est possible de construire un vecteur de descente commun à tous les objectifs. Cette

construction d'une direction de descente permet de généraliser les algorithmes de descente classique à des problèmes où l'image de la fonction coût est de dimension supérieure à 1. Dans la suite, on présente un algorithme de descente compatible avec la Définition 35 d'un minimum sur \mathbb{R}^m muni du préordre de Pareto.

Chapitre 4

Algorithme de descente pour l'optimisation multiobjectif et son extension au cadre stochastique

4.1 Multiple Gradient Descent Algorithm (*MGDA*)

L'algorithme *Multiple Gradient Descent Algorithm (MGDA)* a été introduit pour la première fois en 2009 dans le rapport technique [20] afin de résoudre les problèmes d'optimisation multiobjectif composés de fonctions différentiables

$$\min_{\mathbf{x} \in \mathcal{X}} \{f_1(\mathbf{x}), \dots, f_m(\mathbf{x})\}.$$

MGDA est un algorithme de descente qui se base sur l'existence et la construction d'un vecteur de descente commun à l'ensemble des objectifs à chaque itération. L'algorithme utilise le vecteur de descente commun proposé au Théorème 6 et se place uniquement dans le cadre d'objectifs supposés convexes et différentiable. Par conséquent, l'enveloppe convexe de l'union des sous-différentielles \mathcal{C} en un point \mathbf{x} est réduite à

$$\mathcal{C}(\mathbf{x}) = \text{Conv} \left(\{\nabla f_i(\mathbf{x})\}_{i \in \llbracket 1, m \rrbracket} \right).$$

Le vecteur de descente calculé est alors utilisé en tant que direction de descente pour un pas donné. Les méthodes classiques de sélection de pas de descente peuvent être utilisées pour trouver un pas optimal. Il est possible de prouver la convergence d'un tel algorithme vers des solutions Pareto stationnaires qui, dans le cadre convexe, sont

aussi les solutions de Pareto.

L'algorithme *MGDA* s'écrit :

Algorithm 1: Multiple Gradient Descent Algorithm (*MGDA*)

```

begin
  On se donne :
  — un point de départ  $\mathbf{x}_0$ ,
  — un seuil pour le critère d'arrêt  $\sigma$  et  $\|\xi_0\| = \sigma$ ,
  — le choix d'un pas de descente  $\epsilon$ ,
  — un compteur  $k = 1$ .
  while  $\|\xi_{k-1}\| \geq \sigma$  do
    Évaluer  $\{\nabla f_i(\mathbf{x}_{k-1})\}_{i \in \llbracket 1, m \rrbracket}$ .
    Construire le vecteur de descente commun  $-\xi_k$  en résolvant :

        
$$\xi_k = \operatorname{argmin}\{\|\mathbf{v}\|; \mathbf{v} \in \mathcal{C}(\mathbf{x}_{k-1})\}.$$


    Actualiser le point courant

        
$$\mathbf{x}_k = \mathbf{x}_{k-1} - \epsilon_k \xi_k.$$


     $k = k + 1$ 
  Evaluer  $\{f_i(\mathbf{x}_k)\}_{i \in \llbracket 1, m \rrbracket}$  à la fin de l'algorithme.

```

Les conditions d'arrêt de l'algorithme seront pilotées par la valeur de la norme du vecteur de descente utilisé. En effet, si la norme de ce vecteur de descente est nulle, on sait que le point courant est un point Pareto stationnaire. En pratique, on imposera un seuil sur la norme du vecteur de descente en deçà duquel le point sera considéré comme Pareto stationnaire.

Le choix d'un pas de descente ϵ est une opération délicate à effectuer et plusieurs méthodes ont été proposées dans la littérature. Une première méthode classique consiste à garder un pas fixe sur l'ensemble des itérations. Cette méthode a l'avantage de ne pas nécessiter d'informations supplémentaires lors des itérations mais peut s'avérer inefficace lorsque le pas de descente est mal choisi. En effet un pas trop important pourra amener l'algorithme à osciller autour d'un point Pareto stationnaire sans jamais réussir à l'atteindre, alors qu'un pas trop petit risque d'être très coûteux en terme d'itérations nécessaires pour converger. Un second moyen d'obtenir un pas de descente est de procéder à une étude de variation des objectifs autour du point courant dans

la direction de descente ξ calculée, et d'approcher localement chacun des objectifs par une fonction quadratique dont la localisation du minimum est connue analytiquement. Le pas de descente utilisé sera alors le plus petit des pas de descente optimaux.

Plus précisément, on se donne une *solution admissible* \mathbf{x} dans l'espace des variables d'optimisation, et une direction de descente commune $-\xi$ pour le point courant \mathbf{x} . On se donne un rayon r , et on calcule les valeurs des objectifs aux points $(\mathbf{x} - r\xi)$ et $(\mathbf{x} - 2r\xi)$. À partir de ces valeurs, on construit m fonctions quadratiques

$$\forall i \in \llbracket 1, m \rrbracket, \forall \ell \in \mathbb{R}, f_i(\mathbf{x} - \ell\xi) \approx \tilde{f}_i(\mathbf{x} - \ell\xi) = A_i \ell^2 + B_i \ell + C_i,$$

passant par les valeurs calculées aux points $\mathbf{x} - r\xi$ et $\mathbf{x} - 2r\xi$. Le pas de descente ϵ utilisé est alors de la forme

$$\epsilon = \min_{i \in \llbracket 1, m \rrbracket} \left\{ -\frac{B_i}{2A_i} \right\}$$

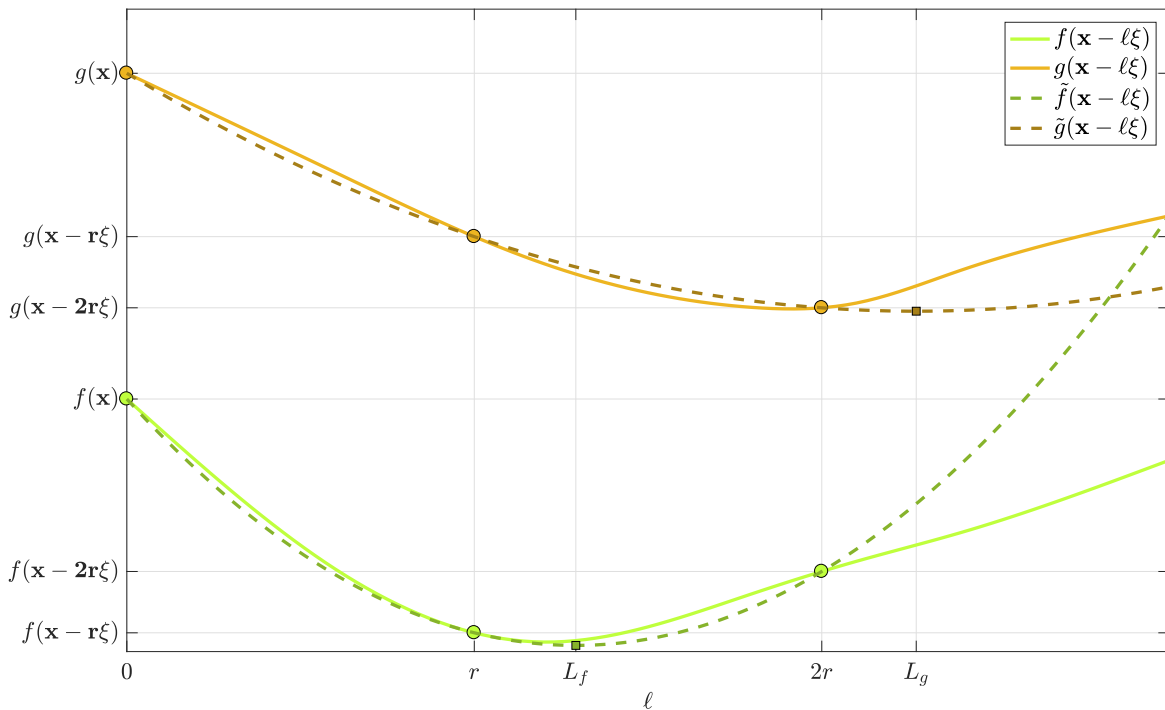


FIGURE 4.1 Exemple de calcul de pas ϵ optimal par approximation quadratique dans la direction ξ

La Figure 4.1 illustre la variation locale de deux objectifs $\{f(\mathbf{x}), g(\mathbf{x})\}$ dans la direction de descente $-\xi$ en fonction du paramètre ℓ . En trait plein, l'évolution des vrais objectifs est représentée, les approximations quadratiques étant quant à elles

représentées en pointillé. Pour chacune des approximations, il est facile de d'obtenir L_i tel qu'il minimise les approximations quadratiques. Le pas de descente utilisé étant pris comme le minimum des L_i ainsi obtenus. Pour cette illustration, le pas de descente choisi sera donc L_f .

Théorème 7. (Désidéri [20]) *Lorsque les objectifs du problème d'optimisation (3.2) sont convexes, une sous-suite de la suite itérative des $\{\mathbf{x}_k\}_{k \in \mathbb{N}}$ décrite dans l'Algorithme 1 converge vers un point \mathbf{x}^* solution de Pareto du problème d'optimisation (3.2).*

MGDA est donc un algorithme de descente compatible avec la minimisation dans \mathbb{R}^m muni du préordre de Pareto. L'algorithme repose sur l'existence d'un vecteur de descente commun à l'ensemble des objectifs et offre des résultats de convergences vers les solutions du problème (3.2) lorsque les objectifs du problèmes sont convexes et différentiables en tout point de l'ensemble des *solutions admissibles*. Dans la suite du document nous nous intéresserons au cas où les fonctions objectifs dépendent de paramètres incertains qui seront modélisés par des variables aléatoires. Dans le cadre de ce travail, nous considérons que l'aléa ne porte pas sur les variables d'optimisation. Nous montrerons comment alors adapter l'algorithme *MGDA*. Dans un premier temps, nous allons tout d'abord formuler le problème multicritère aléatoire avant de présenter l'algorithme.

4.2 Stochastic Multiple Gradient Descent Algorithm (*SMGDA*)

Dans cette partie du chapitre on se propose de modifier le problème d'optimisation (3.2) initialement traité. On considère désormais dans la suite que la fonction coût f du problème est une fonction aléatoire à valeurs dans \mathbb{R}^m .

Tout d'abord, le sigle choisi pour désigner l'algorithme peut sembler à priori erroné puisqu'il permet de traiter le cas des fonctions objectifs non-différentiables. Il sera néanmoins retenu car d'une part il permet de marquer sa filiation avec *MGDA* et d'autre part on a pu voir que numériquement, il utilisera des gradients pour approcher les sous-différentielles aux points non-réguliers.

4.2.1 Problème d'optimisation stochastique multiobjectif

Soit $(\Omega, \mathcal{A}, \mathbb{P})$ un espace probabilisé, et $W : \Omega \rightarrow \mathbb{R}^d$, $\omega \mapsto W(\omega)$ un vecteur aléatoire supposé donné. On note μ la distribution du vecteur aléatoire W et \mathcal{W} son image

$W(\Omega) \subseteq \mathbb{R}^d$. Soit W_1, \dots, W_p, \dots des copies indépendantes de la variable aléatoire W qui seront utilisées pour générer des tirages aléatoires indépendants avec la distribution μ .

On considère m fonctions $f_i : \mathbb{R}^n \times \mathcal{W} \rightarrow \mathbb{R}$, $i = 1, \dots, m$. Le problème proposé dans ce paragraphe est de résoudre le problème multiobjectif moyen qui peut s'écrire de la façon suivante :

$$\min_{\mathbf{x} \in \mathbb{R}^n} \{\mathbb{E}[f_1(\mathbf{x}, W)], \mathbb{E}[f_2(\mathbf{x}, W)], \dots, \mathbb{E}[f_m(\mathbf{x}, W)]\}. \quad (4.1)$$

Plus précisément, on cherche à construire un ensemble de points Pareto solutions. De la façon dont il est présenté, le problème (4.1) est un problème déterministe. Cependant, les espérances des objectifs ne sont généralement pas connues. Une approche classique consiste à remplacer chaque espérance par une approximation construite par le biais d'un estimateur d'espérance utilisant des tirages aléatoires indépendants w_k de la variable aléatoire W [8, 35]. Cette approche est numériquement coûteuse puisqu'elle fait apparaître deux boucles imbriquées dans les algorithmes de résolution. Comme pour l'algorithme du gradient stochastique, l'algorithme qui est proposé dans cette partie du chapitre ne nécessite pas le calcul des espérances, il est uniquement basé sur le calcul des sous-différentielles de Clarke pour des tirages de la variable aléatoire.

On introduit la filtration $(\mathcal{F}_k)_{k \in \mathbb{N}}$ où les tribus sont générées par les k premières variables aléatoires de la suite $(W_i)_{i \in \mathbb{N}}$

$$\forall k \in \mathbb{N}, \mathcal{F}_k = \sigma(W_1, \dots, W_k).$$

4.2.2 Algorithme et résultats théoriques

On se donne un problème d'optimisation (4.1)

$$\min_{\mathbf{x} \in \mathbb{R}^n} \{\mathbb{E}[f_1(\mathbf{x}, W)], \mathbb{E}[f_2(\mathbf{x}, W)], \dots, \mathbb{E}[f_m(\mathbf{x}, W)]\}.$$

Trois classes de fonctions objectifs localement lipschitziennes seront considérées dans cette partie de chapitre :

\mathcal{G}^1 la classe des fonctions convexes et différentiables : $\partial^\mathcal{C} f_i(\mathbf{x}, W(\omega)) = \nabla f_i(\mathbf{x}, W(\omega))$,

\mathcal{G}^2 la classe des fonctions convexes non-différentiables : $\partial^\mathcal{C} f_i(\mathbf{x}, W(\omega)) = \partial f_i(\mathbf{x}, W(\omega))$,

\mathcal{G}^3 la classe des fonctions f° -pseudo-convexes.

On notera $\xi(\mathbf{x}, W(\omega))$ l'élément de norme minimale de l'enveloppe convexe $\mathcal{C}(\mathbf{x}, W(\omega))$:

$$\exists \{\alpha_i(\mathbf{x}, W(\omega))\}_{i \in \llbracket 1, m \rrbracket}, \quad \xi(\mathbf{x}, W(\omega)) = \sum_{i=1}^m \alpha_i(\mathbf{x}, W(\omega)) \mathbf{s}_i(\mathbf{x}, W(\omega)),$$

où les $\mathbf{s}_i(\mathbf{x}, W(\omega)) \in \partial^{\mathcal{C}} f_i(\mathbf{x}, W(\omega))$, les $\alpha_i(\mathbf{x}, W(\omega))$ sont positifs et $\sum_i \alpha_i(\mathbf{x}, W(\omega)) = 1$ presque sûrement.

L'algorithme proposé dans la suite du chapitre consiste à effectuer un tirage aléatoire w_k d'une copie W_k de la variable aléatoire W et de calculer l'élément de norme minimale dans l'enveloppe $\mathcal{C}(\mathbf{x}, w_k)$ ainsi généré. Son opposé est un vecteur de descente commun aux objectifs du problème pour le tirage particulier w_k . On utilise comme pas de descente les éléments d'une σ -suite définie par

Définition 41. On dit qu'une suite $(\epsilon_i)_{i \in \mathbb{N}}$ est une σ -suite si sa série d'ordre 1 diverge, et que sa série d'ordre 2 converge

$$\begin{cases} \sum_{i=1}^{+\infty} \epsilon_i = \infty \\ \sum_{i=1}^{+\infty} \epsilon_i^2 < \infty \end{cases}$$

Exemple :

Pour (A, B, C, D) des réels positifs tels que $C \in]\frac{1}{2}, 1]$, les suites de type

$$\epsilon_k = \frac{A}{B \cdot k^C + D}, \quad \forall k \in \mathbb{N},$$

sont des σ -suites.

Algorithm 2: Stochastic Multiple Gradient Descent Algorithm (SMGDA)

```

begin
  — Un point de départ  $\mathbf{X}_0$ .
  — Un nombre d'itérations  $N$ .
  — Une  $\sigma$ -suite  $(\epsilon_i)_{i \in \mathbb{N}}$ .
  — Un compteur  $k = 1$ .

  for  $k = 1, \dots, N$  do
    Effectuer un tirage aléatoire  $w_{k+1}$  de la variable aléatoire  $W_k$ .
    Évaluer  $\mathcal{C}_k(\mathbf{X}_k, w_{k+1})$ .
    Construire le vecteur de descente commun  $-\xi_k(\mathbf{X}_k, w_{k+1})$  en résolvant :

        
$$\xi_k = \operatorname{argmin}\{\|\mathbf{v}\|; \mathbf{v} \in \mathcal{C}(\mathbf{X}_k, w_{k+1})\}.$$


    Faire l'itération

        
$$\mathbf{X}_{k+1} = \mathbf{X}_k - \epsilon_k \xi_k. \tag{4.2}$$


  Evaluer  $\{\mathbb{E}[f_i(\mathbf{X}_N, W)]\}_{i \in \llbracket 1, m \rrbracket}$ 

```

Remarque 4. *L'algorithme SMGDA ne nécessite jamais l'estimation des espérances des objectifs durant le processus d'optimisation. Les espérances sont évaluées uniquement à la fin pour le point \mathbf{X}_N afin de construire le front de Pareto dans l'espace des objectifs.*

Remarque 5. *À chaque itération k , la variable aléatoire \mathbf{X}_k est \mathcal{F}_k -mesurable, la suite réursive décrite par l'équation*

$$\mathbf{X}_{k+1}(\omega) = \mathbf{X}_k(\omega) - \epsilon_k \xi(\mathbf{X}_k(\omega), W_{k+1}(\omega))$$

est un processus stochastique adapté à la filtration $\mathcal{F} = (\sigma(W_1, \dots, W_k))_{k \in \mathbb{N}}$.

Remarque 6. *Partant d'un unique point de départ \mathbf{x}_0 , l'algorithme SMGDA donne une seule solution appartenant au front de Pareto en fin de processus d'optimisation. Par conséquent, afin d'avoir une description complète du front de Pareto, il est nécessaire de lancer l'algorithme depuis plusieurs points de départ. Ce qui peut se faire par parallélisation en effectuant une optimisation pour un point de départ différent par cœur de calcul disponible.*

Remarque 7. *Dans cet algorithme, aucun critère d'arrêt n'est proposé. L'existence d'indicateur de convergence sera discuté plus loin dans le chapitre après avoir donné certains résultats.*

On introduit la notation \mathbf{x}^* qui désigne un des points solution de Pareto parmi les plus proches d'un point courant \mathbf{x}

$$\forall k \in \mathbb{N}, \mathbf{x}^* \in \operatorname{argmin}_{\mathbf{v} \in \mathcal{P}_D} \|\mathbf{x} - \mathbf{v}\|.$$

On utilisera la notation \mathcal{L}_k^* pour désigner la distance au carré entre \mathbf{X}_k et un des éléments de \mathcal{P}_D les plus proches

$$\mathcal{L}_k^* = \|\mathbf{X}_k - \mathbf{X}_k^*\|^2, \forall k \in \mathbb{N}.$$

Afin de prouver la convergence de l'algorithme, nous supposerons vérifiées l'ensemble d'hypothèses noté \mathcal{H} suivant :

Hypothèses \mathcal{H} :

- H1. L'ensemble des solutions de Pareto est non vide $\mathcal{P}_D \neq \emptyset$.
- H2. Les fonctions $\mathbf{x} \mapsto f_i(\mathbf{x}, W)$ appartiennent à une des trois classes $(\mathcal{G}^1, \mathcal{G}^2, \mathcal{G}^3)$ de fonctions pour tout $i \in \llbracket 1, m \rrbracket$.
- H3. Les variables aléatoires $\omega \mapsto f_i(\mathbf{x}, W(\omega))$ sont intégrables pour tout $i \in \llbracket 1, m \rrbracket$.
- H4. La norme de tout vecteur $\mathbf{s}_i \in \partial^{\mathcal{C}} f_i(\mathbf{x}, W(\omega))$ est presque sûrement majorée par un nombre réel M_i

$$\|\mathbf{s}_i\| \leq M_i, \text{ p.s. } \forall \mathbf{x} \in \mathcal{X}.$$

- H5. Il existe un objectif f_ℓ convexe pour lequel la contribution $\alpha_\ell(\mathbf{x}, W(\omega))$ dans la construction du vecteur de descente commun n'est pas nulle presque sûrement. De plus :

$$\begin{cases} \forall i \in \llbracket 1, m \rrbracket, f_i(\mathbf{x}, W) - f_i(\mathbf{x}^*, W) \geq 0, \text{ p.s.} \\ \exists \ell \in \llbracket 1, m \rrbracket, f_\ell(\mathbf{x}, W) - f_\ell(\mathbf{x}^*, W) \geq \frac{C}{2} \|\mathbf{x} - \mathbf{x}^*\|^2, \text{ p.s..} \end{cases}$$

- H6. La suite $(\epsilon_i)_{i \in \mathbb{N}}$ est une σ -suite.

Remarque 8. L'hypothèse H5. induit que le point \mathbf{x}^* domine presque sûrement le point \mathbf{x} .

Avant de proposer des résultats de convergence, certaines propriétés du vecteur de descente commun dans ce cadre d'hypothèses devront être démontrées. Il sont présentés ci-dessous.

Proposition 11. *La norme du vecteur de descente commun est presque sûrement majoré par un réel M :*

$$\|\xi(\mathbf{x}, W(\omega))\| \leq M, \text{ p.s., } \mathbf{x} \in \mathcal{X}.$$

Démonstration. ξ est l'élément de norme minimal dans l'enveloppe convexe \mathcal{C}_k au point $\mathbf{x} \in \mathcal{X}$ pour un tirage fixé $\omega \in \Omega$, il existe donc une famille $\{\alpha_i(\mathbf{x}, W(\omega))\}_{i \in \llbracket 1, m \rrbracket}$ de réels positifs telle que

$$\|\xi(\mathbf{x}, W(\omega))\| = \left\| \sum_{i=1}^m \alpha_i(\mathbf{x}, W(\omega)) \mathbf{s}_i(\mathbf{x}, W(\omega)) \right\|,$$

où $\alpha_i \in [0, 1]$, $\sum \alpha_i = 1$ et où \mathbf{s}_i est un élément de la sous-différentielle $\partial^{\mathcal{C}} f_i$. On peut donc majorer la norme de ξ par la somme des éléments \mathbf{s}_i ce qui implique

$$\|\xi(\mathbf{x}, W(\omega))\| \leq \sum_{i=1}^m \|\mathbf{s}_i(\mathbf{x}, W(\omega))\|, \text{ p.s.}$$

L'utilisation de l'hypothèse H4. permet d'avoir la relation suivante et de conclure

$$\|\xi(\mathbf{x}, W(\omega))\| \leq \sum_{i=1}^m M_i, \text{ p.s.}$$

□

Définition 42. *On définit le vecteur de descente commun moyen au point $\mathbf{x} \in \mathcal{X}$ par la relation*

$$\Xi(\mathbf{x}) = \mathbb{E} [\xi(\mathbf{x}, W(\omega))] = \sum_{i=1}^m \mathbb{E} [\alpha_i(\mathbf{x}, W(\omega)) \mathbf{s}_i(\mathbf{x}, W(\omega))], \quad \mathbf{s}_i \in \partial^{\mathcal{C}} f_i, i = 1, \dots, m.$$

Proposition 12. *En un point \mathbf{x} de \mathcal{X} , le vecteur de descente commun moyen vérifie la relation*

$$\Xi(\mathbf{x})(\mathbf{x} - \mathbf{x}^*) \geq \frac{C}{2} \|\mathbf{x} - \mathbf{x}^*\|^2$$

Démonstration. L'hypothèse H5. nous dit qu'il existe ℓ dans $\llbracket 1, m \rrbracket$ tel que

$$\langle \mathbf{s}_\ell(\mathbf{x}, W(\omega)), \mathbf{x} - \mathbf{x}^* \rangle \geq f_\ell(\mathbf{x}, W(\omega)) - f_\ell(\mathbf{x}^*, W(\omega)) \geq \frac{C}{2} \|\mathbf{x} - \mathbf{x}^*\|^2, \text{ p.s.} \quad (4.3)$$

De plus, on a supposé que \mathbf{x}^* domine le point \mathbf{x} presque sûrement ce qui donne

$$\forall i \in \llbracket 1, m \rrbracket, f_i(\mathbf{x}^*, W(\omega)) \leq f_i(\mathbf{x}, W(\omega)), \text{ p.s.}$$

En utilisant les caractéristiques des différentes classes de fonctions traitées, on peut obtenir facilement que

$$\langle \mathbf{s}_i(\mathbf{x}, W(\omega)), \mathbf{x} - \mathbf{x}^* \rangle \geq 0, \forall i \in \llbracket 1, m \rrbracket, \text{ p.s.} \quad (4.4)$$

En multipliant cette relation par α_i positif, et en sommant les expressions (4.3) et (4.4) pour tout $i = 1, \dots, m$, on obtient la relation

$$\sum_{i=1}^m \alpha_i \langle \mathbf{s}_i(\mathbf{x}, W(\omega)), \mathbf{x} - \mathbf{x}^* \rangle \geq \alpha_\ell(\mathbf{x}, W(\omega)) \frac{C}{2} \|\mathbf{x} - \mathbf{x}^*\|^2, \text{ p.s.} \quad (4.5)$$

En prenant l'espérance de l'expression (4.5), on aboutit au résultat

$$\Xi(\mathbf{x}) \cdot (\mathbf{x} - \mathbf{x}^*) = \sum_{i=1}^m \langle \mathbb{E} [\alpha_i(\mathbf{x}, W(\omega)) \mathbf{s}_i(\mathbf{x}, W(\omega))], \mathbf{x} - \mathbf{x}^* \rangle \geq \mathbb{E} [\alpha_\ell(\mathbf{x}, W(\omega))] \frac{C}{2} \|\mathbf{x} - \mathbf{x}^*\|^2.$$

L'hypothèse H5. permet de conclure car $\mathbb{E} [\alpha_\ell(\mathbf{x}, W(\omega))] > 0$.

□

Un dernier résultat sur les vecteurs de descente commun est nécessaire, il s'agit cette fois d'une propriété probabiliste lié au processus $(W_k)_{k \in \mathbb{N}}$. On désigne par ξ_k le vecteur de descente commun au point \mathbf{X}_k

$$\xi_k(\omega) = \xi(\mathbf{X}_k, W_{k+1}(\omega)),$$

et on utilise Ξ_k pour désigner son espérance $\Xi_k = \mathbb{E} [\xi_k(\omega)]$.

Lemme 3. *L'espérance conditionnelle de $\langle \mathbf{X}_k - \mathbf{X}_k^*, \xi_k - \Xi_k \rangle$ sachant la tribu \mathcal{F}_k est nulle*

$$\mathbb{E} [\langle \mathbf{X}_k - \mathbf{X}_k^*, \xi_k - \Xi_k \rangle | \mathcal{F}_k] = 0, \text{ p.s.}$$

Démonstration. La preuve résulte directement du Lemme 1 puisque la variable aléatoire W_{k+1} est indépendante de \mathcal{F}_k , de même que ξ_k

$$\mathbb{E} [\langle \mathbf{X}_k - \mathbf{X}_k^*, \xi_k - \Xi_k \rangle | \mathcal{F}_k] = \psi(\mathbf{X}_k), \text{ p.s.,}$$

où ψ est la fonction définie par

$$\psi(\mathbf{x}) = \langle \mathbf{x} - \mathbf{x}^*, \mathbb{E} [\xi_k] - \Xi_k \rangle, \text{ p.s., } \forall \mathbf{x} \in \mathcal{X}.$$

La conclusion résulte directement de la définition du vecteur de descente commun moyen pour lequel

$$\Xi_k = \mathbb{E}_{W_{k+1}}[\xi(\mathbf{X}_k, W_{k+1})].$$

□

Un premier théorème prouve la convergence en moyenne quadratique du processus $(\mathbf{X}_k)_{k \in \mathbb{N}}$ vers une solution \mathbf{X}^* de Pareto du problème d'optimisation.

Théorème 8. *Sous l'ensemble d'hypothèses \mathcal{H} , le processus stochastique construit par les itérations de SMGDA converge en norme L^2 vers un point \mathbf{X}^* Pareto solution*

$$\lim_{k \rightarrow +\infty} \mathbb{E} \left[\|\mathbf{X}_k - \mathbf{X}_k^*\|^2 \right] = 0.$$

Démonstration. Soit

$$\mathcal{L}_k^* = \|\mathbf{X}_k - \mathbf{X}_k^*\|^2, \text{ p.s.}$$

Comme \mathbf{X}_{k+1}^* est l'une des solutions de Pareto les plus proches de \mathbf{X}_{k+1} , on a

$$\mathcal{L}_{k+1}^* \leq \|\mathbf{X}_{k+1} - \mathbf{X}_k^*\|^2, \text{ p.s.}$$

En introduisant la relation de récurrence (4.2) induite par SMGDA, on obtient

$$\mathcal{L}_{k+1}^* \leq \|\mathbf{X}_k - \epsilon_k \xi_k - \mathbf{X}_k^*\|^2, \text{ p.s.}$$

En développant le terme de droite :

$$\mathcal{L}_{k+1}^* \leq \mathcal{L}_k^* + \epsilon_k^2 \|\xi_k\|^2 - 2\epsilon_k \langle \mathbf{X}_k - \mathbf{X}_k^*, \xi_k \rangle, \text{ p.s.}$$

En ajoutant le terme nul $\langle \mathbf{X}_k - \mathbf{X}_k^*, \Xi_k - \Xi_k \rangle$ à droite, puis en utilisant simultanément les propositions 11 et 12, on est capable de majorer à nouveau l'expression pour arriver à :

$$\mathcal{L}_{k+1}^* \leq \mathcal{L}_k^* (1 - \epsilon_k C) + \epsilon_k^2 M^2 - 2\epsilon_k \langle \mathbf{X}_k - \mathbf{X}_k^*, \xi_k - \Xi_k \rangle, \text{ p.s.}$$

Par la suite, en prenant l'espérance conditionnelle de l'expression précédente par rapport à la tribu \mathcal{F}_k , \mathbf{X}_k étant \mathcal{F}_k -mesurable, on obtient :

$$\mathbb{E} \left[\mathcal{L}_{k+1}^* | \mathcal{F}_k \right] \leq \mathcal{L}_k^* (1 - \epsilon_k C) + \epsilon_k^2 M^2 - 2\epsilon_k \mathbb{E} \left[\langle \mathbf{X}_k - \mathbf{X}_k^*, \xi_k - \Xi_k \rangle | \mathcal{F}_k \right], \text{ p.s.}$$

En introduisant le résultat du Lemme 3, on aboutit à la relation

$$\mathbb{E} \left[\mathcal{L}_{k+1}^* | \mathcal{F}_k \right] \leq \mathcal{L}_k^* (1 - \epsilon_k C) + \epsilon_k^2 M^2, \text{ p.s.}$$

En prenant l'espérance de cette relation, et en utilisant le fait que $\mathbb{E} \left[\mathbb{E} \left[\mathcal{L}_{k+1}^* | \mathcal{F}_k \right] \right] = \mathbb{E} \left[\mathcal{L}_{k+1}^* \right]$, on obtient finalement

$$\mathbb{E} \left[\mathcal{L}_{k+1}^* \right] \leq \mathbb{E} \left[\mathcal{L}_k^* \right] (1 - \epsilon_k C) + \epsilon_k^2 M^2.$$

En sommant la relation précédente pour N itérations consécutives :

$$\mathbb{E} \left[\mathcal{L}_{k+1+N}^* \right] \leq \mathbb{E} \left[\mathcal{L}_k^* \right] \prod_{j=k}^{k+N} (1 - \epsilon_j C) + \sum_{j=k}^{k+N} \epsilon_j^2 M^2.$$

La preuve de convergence finale résulte du fait que le produit et la somme présentés ci-dessus tendent vers 0. La somme tend vers 0 car la suite $(\epsilon_k)_{k \in \mathbb{N}}$ est une σ -suite. Il est facile de prouver la convergence du terme produit en considérant son logarithme. Finalement, on a prouvé que

$$\lim_{N \rightarrow +\infty} \mathbb{E} \left[\mathcal{L}_{k+1+N}^* \right] = 0,$$

ce qui donne la convergence en norme L^2 . \square

Il est possible d'obtenir une information sur la vitesse de convergence lorsqu'on considère une σ -suite particulière :

Théorème 9. *Soit \mathbf{x}_0 un point de \mathcal{X} , si on utilise la σ -suite*

$$\epsilon_k = \frac{1}{C/2 \cdot k + \frac{M^2}{\mathbb{E}[\mathcal{L}_0^*]C/2}}, \quad \mathbb{E}[\mathcal{L}_0^*] = \mathbb{E} \left[\|\mathbf{x}_0 - \mathbf{x}_0^*\|^2 \right],$$

alors

$$\mathbb{E}[\mathcal{L}_k^*] \leq \frac{1}{\frac{C/2}{M^2} k + \frac{1}{\mathbb{E}[\mathcal{L}_0^*]}}.$$

Démonstration. La démonstration est exactement identique à la preuve de vitesse de convergence que l'on peut trouver pour l'algorithme du gradient stochastique [25]. \square

La Remarque 7 met en avant l'absence de critère d'arrêt dans l'algorithme *SMGDA*. Cependant, en s'appuyant sur le résultat de vitesse de convergence, il est possible de proposer des critères d'arrêts comme cela a été fait dans les papiers de Bouleau

et al. [9], Dvoretzky [29]. L'inconvénient majeur des critères ainsi proposés sont qu'ils nécessitent des grandeurs non connues comme $\mathbb{E}[\mathcal{L}_0^*]$ qui impose de connaître l'ensemble des solutions du problème afin de déterminer l'élément \mathbf{x}_0^* . D'autre part, les conditions d'optimalité du Théorème 5 peuvent permettre de donner un critère d'arrêt sur la norme du vecteur de descente associé au problème en espérance de manière équivalente au critère proposé pour l'algorithme *MGDA*. En effet, en un point \mathbf{x}^* Pareto solution du problème d'optimisation

$$\min_{\mathbf{x} \in \mathcal{X}} \{ \mathbb{E}[f_1(\mathbf{x}, W(\omega))], \dots, \mathbb{E}[f_m(\mathbf{x}, W(\omega))] \},$$

il existe une combinaison convexe d'éléments des espérances des sous-différentielles de Clarke en \mathbf{x}^* nulle

$$\exists \{ \alpha_i \}_{i \in [1, m]}, \sum_{i=1}^m \alpha_i \mathbf{s}_i(\mathbf{x}^*) = 0,$$

où $\mathbf{s}_i \in \mathbb{E}[\partial^{\mathcal{C}} f_i(\mathbf{x}^*, W(\omega))]$ et la somme des α_i positifs ou nuls vaut un $\sum_i \alpha_i = 1$. L'extension directe du critère proposé dans *MGDA* revient donc à calculer l'élément de norme minimale dans l'espérance de l'enveloppe convexe de l'union des sous-différentielles de Clarke

$$\zeta = \operatorname{argmin} \left\{ \|\mathbf{u}\|, \mathbf{u} \in \mathbb{E} \left[\operatorname{Conv} \left(\bigcup_{i=1}^m \partial^{\mathcal{C}} f_i(\mathbf{x}^*, W(\omega)) \right) \right] \right\}.$$

L'algorithme pouvant alors être stoppé lorsque la norme de ζ atteint un certain seuil. Ce critère d'arrêt est nécessairement coûteux puisqu'il nécessite le calcul d'espérances, ce qui est contraire à la philosophie de *SMGDA*.

Théorème 10. *Le processus stochastique décrit par la séquence de variables aléatoires $(\mathbf{X}_k)_{k \in \mathbb{N}}$ converge presque sûrement vers une solution de Pareto \mathbf{X}^* .*

$$\mathbb{P} \left(\left\{ \lim_{k \rightarrow +\infty} \mathbf{X}_k - \mathbf{X}^* = 0 \right\} \right) = 1.$$

Démonstration. Soit $(Y_k)_{k \in \mathbb{N}}$ le processus stochastique décrit par la relation

$$Y_k = \|\mathbf{X}_k - \mathbf{X}_k^*\|^2 + M^2 \sum_{j \geq k} \epsilon_j^2.$$

Il est possible de majorer Y_{k+1} par la relation suivante en utilisant un raisonnement équivalent à la preuve de convergence en norme L^2

$$Y_{k+1} \leq \|\mathbf{X}_k - \mathbf{X}_k^*\|^2 + \epsilon_k^2 M^2 + M^2 \sum_{j \geq k+1} \epsilon_j^2 - 2\epsilon \langle \mathbf{X}_k - \mathbf{X}_k^*, \xi_k - \Xi_k + \Xi_k \rangle.$$

La Propostion 12 permet de majorer par 0 le dernier terme de la relation ci-dessus

$$-2\epsilon_k \langle \mathbf{X}_k - \mathbf{X}_k^*, \Xi_k \rangle \leq -\epsilon_k C \|\mathbf{X}_k - \mathbf{X}_k^*\|^2 \leq 0,$$

ce qui mène finalement à

$$Y_{k+1} \leq \|\mathbf{X}_k - \mathbf{X}_k^*\|^2 + \epsilon_k^2 M^2 + M^2 \sum_{j \geq k+1} \epsilon_j^2 - 2\epsilon \langle \mathbf{X}_k - \mathbf{X}_k^*, \xi_k - \Xi_k \rangle.$$

En prenant l'espérance conditionnelle de l'expression précédente sachant la tribu \mathcal{F}_k et en utilisant le Lemme 3, on obtient

$$\mathbb{E}[Y_{k+1} | \mathcal{F}_k] \leq \mathbb{E}[Y_k | \mathcal{F}_k].$$

Or la variable aléatoire Y_k est \mathcal{F}_k -mesurable, par conséquent

$$\mathbb{E}[Y_{k+1} | \mathcal{F}_k] \leq Y_k.$$

Le processus $(Y_k)_{k \in \mathbb{N}}$ est donc une surmartingale qui est de manière évidente positive. En utilisant les résultats du Théorème 1, on peut affirmer que le processus décrit par les $(Y_k)_{k \in \mathbb{N}}$ converge presque sûrement vers une variable aléatoire que l'on notera Y_∞ . L'utilisation du Lemme de Fatou permet d'écrire

$$0 \leq \mathbb{E} \left[\lim_{k \rightarrow +\infty} \left(\inf_{j \geq k} Y_j \right) \right] = \mathbb{E}[Y_\infty] \leq \lim_{k \rightarrow +\infty} \left(\inf_{j \geq k} \mathbb{E}[Y_j] \right) \leq \lim_{k \rightarrow +\infty} \left(\mathbb{E}[\mathcal{L}_k^*] + \sum_{j \geq k} \epsilon_j^2 M^2 \right).$$

La convergence en norme et le fait que le second terme est le reste de la série carré sommable des $(\epsilon_k)_{k \in \mathbb{N}}$ nous permet de déduire que

$$\mathbb{E}[Y_\infty] = 0$$

Le processus stochastique $(Y_k)_{k \in \mathbb{N}}$ étant positif, cela implique que nécessairement $Y_\infty = 0$ presque sûrement :

$$\mathbb{P} \left(\lim_{k \rightarrow +\infty} Y_k = 0 \right) = 1.$$

□

Les Théorèmes de convergence précédents donnent une justification théorique à l'algorithme *SMGDA*. Ces résultats constituent le cœur de la thèse. Nous allons maintenant évaluer les capacités numériques de l'algorithme en l'utilisant sur un certain nombre de problèmes tests issus de benchmarks classiques et nous comparerons ses performances à celles obtenues par deux autres algorithmes classiques (*NSGA-II*, *DMS*) utilisant pour ces derniers une approche *Sample Average Approximation*.

Chapitre 5

Essais numériques de *SMGDA* et comparaisons

Nous présentons dans ce chapitre les essais numériques qui ont permis de valider l'algorithme. Ces essais, réalisés sur une famille de problèmes issus de benchmarks classiques en optimisation, ont également permis de comparer les performances de *SMGDA* à celles des algorithmes *DMS* et *NSGA-II*. Nous débuterons en présentant deux stratégies numériques permettant d'améliorer les performances de *SMGDA*. Puis nous présentons deux métriques, ou mesures de performance, utilisées pour la comparaison d'algorithmes d'optimisation multiobjectifs.

5.1 Améliorations de *SMGDA* et stratégies numériques d'optimisation

On présente ici deux stratégies permettant d'améliorer les performances de *SMGDA*. La première, en guidant le choix des points initiaux, permet d'augmenter la dispersion sur le front de Pareto des solutions données par l'algorithme et réduire ainsi le nombre de points initiaux nécessaires pour parvenir à décrire correctement le front. La seconde démarche utilise une modification du vecteur de descente utilisé à chaque itération : il est construit à partir des estimateurs récursifs des gradients ou des sous-gradients. Cette stratégie permet là encore d'améliorer la dispersion des solutions sur le front et offre un test d'arrêt.

5.1.1 Stratégie pour la génération de points de départ

La Remarque 6 nous permet de lancer *SMGDA* simultanément sur un ensemble de cœurs de calcul noté N_{th} pour des points de départs différents. Lorsqu'on désire un nombre de solutions finales N_{sol} pour décrire le front de telle sorte que $N_{sol} \gg N_{th}$ le processus d'optimisation total requiert plusieurs salves d'optimisation. On considère que le nombre de salves d'optimisation à effectuer est donné par

$$N_{batch} = E\left(\frac{N_{sol}}{N_{th}}\right) + 1,$$

où $E(\cdot)$ désigne la partie entière. On utilisera l'indice $j = 1, \dots, N_{batch}$ pour désigner la salve en cours de traitement. On cherche à profiter des informations obtenues à la fin de l'optimisation de chacune des salves afin d'améliorer la procédure d'optimisation en générant des points initiaux potentiellement plus proches de l'ensemble solution, et dans des zones mal décrites par les solutions déjà acquises.

On se donne un problème d'optimisation associé à une fonction coût $f : \mathcal{X} \subseteq \mathbb{R}^n \rightarrow \mathcal{Y} \subseteq \mathbb{R}^m$

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}).$$

On se place à un instant j pour lequel, $j - 1$ salves d'optimisation ont été effectuées. On note $\mathcal{P}_D^{(j)}$ l'ensemble des solutions de Pareto obtenues sur les $j - 1$ premières salves et $\mathcal{P}_O^{(j)}$ leurs images par la fonction coût f . La procédure consiste à construire une approximation numérique qui permet de passer de l'espace des objectifs à l'espace des variables d'optimisation

$$\begin{aligned} \tilde{f}^{-1} &: \mathbb{R}^m \rightarrow \mathbb{R}^n \\ \mathbf{y} &\mapsto \tilde{f}^{-1}(\mathbf{y}). \end{aligned}$$

Cette approximation est construite à l'aide des résultats obtenus par les salves d'optimisation déjà effectuées $\mathcal{P}_O^{(j)}$ et leurs antécédents dans l'espace des variables $\mathcal{P}_D^{(j)}$ en utilisant un modèle de substitution basé sur une approche de Krigeage.

Parallèlement, à l'aide de $\mathcal{P}_O^{(j)}$, on identifie les zones $\mathcal{P}_O^{(j)}$ de \mathcal{Y} à l'instant j qui peuvent contenir des éléments du front de Pareto. Ces régions sont définies par :

$$\mathcal{P}_O^{(j)} = \mathcal{Y} - \{\mathbf{y} \in \mathcal{Y}, \forall \mathbf{y}^* \in \mathcal{P}_O^{(j)}, (\mathbf{y} \preceq_{\mathcal{P}} \mathbf{y}^*) \cup (\mathbf{y}^* \preceq_{\mathcal{P}} \mathbf{y})\},$$

ce qui peut s'écrire plus simplement en utilisant la relation d'incomparabilité

$$\mathcal{P}_O^{(j)} = \{\mathbf{y} \in \mathcal{Y}, \forall \mathbf{y}^* \in \mathcal{P}_O^{(j)}, \mathbf{y} \equiv_{\mathcal{P}} \mathbf{y}^*\}.$$

Des tirages aléatoires uniformes de points dans $\mathcal{P}_O^{(j)}$ sont ensuite effectués et correspondent à la performance cible des futurs points de départ dans l'espace des objectifs. La fonction \tilde{f}^{-1} permet alors de ramener ces tirages dans l'espace des variables d'optimisation qui vont être utilisés comme points de départs $\{\mathbf{x}_0^{(j)}\}$ pour la salve d'optimisation présente j . L'ensemble de cette procédure peut être représenté sous la forme de l'Algorithme 3.

Algorithm 3: Génération de points de départ

begin

Choisir une σ -suite;

Choisir un nombre d'itérations N_1 ;

Choisir un ensemble de points de départ $\{\mathbf{x}_0^{(1)}\}$ pour la première salve;

Déclarer les variables $\mathcal{P}_O^{(0)} = \mathcal{P}_D^{(0)} = \emptyset$;

for $j = 1, \dots, N_{batch}$ **do**

if $j \neq 1$ **then**

 Entraîner le métamodèle de Krigeage $\mathcal{P}_O^{(j)} \mapsto \mathcal{P}_D^{(j)}$;

 Identifier les zones de solutions possibles dans l'espace objectif $\mathcal{P}_O^{(j)}$;

 Effectuer des tirages aléatoires dans $\mathcal{P}_O^{(j)}$ correspondant à la performance espérée des futurs points de départ;

 Utiliser le métamodèle pour obtenir un ensemble de points de départ $\{\mathbf{x}_0^{(j)}\}$;

 Adapter le nombre d'itérations N_j à effectuer en fonction de l'erreur estimée par le métamodèle;

 Optimiser la salve de points de départ $\{\mathbf{x}_0^{(j)}\}$ avec *SMGDA* en parallèle;

 Ajouter les nouvelles solutions

$$\mathcal{P}_D^{(j+1)} = \mathcal{P}_D^{(j)} \cup \{\mathbf{x}_{N_j}^{(j)}\}$$

 Estimer la performance de l'ensemble $\{\mathbf{x}_{N_j}^{(j)}\}$ et les ajouter à $\mathcal{P}_O^{(j+1)}$;

Afin d'illustrer cette procédure, on se propose d'observer le déroulement de cet algorithme sur un exemple d'optimisation simple, pour lequel on détaille les étapes de la procédure. L'illustration est basée sur un exemple classique biobjectif (JOS1)

déterministe suivant

$$\min_{(x,y) \in [-1,3]^2} \left\{ \begin{array}{l} f_1(x, y) = x^2 + y^2 \\ f_2(x, y) = (x - 2)^2 + (y - 2)^2. \end{array} \right\}$$

Les solutions de Pareto analytiques sont données par $x = y$, $x \in [0, 2]$. Les différentes étapes sont présentées dans le Tableau 5.1 où chaque ligne correspond aux étapes décrites ci-après :

1. Un premier ensemble de points de départ $\{\mathbf{x}_0^{(1)}\}$ est tiré aléatoirement dans l'ensemble réalisable.
2. *SMGDA* est utilisé en parallèle pour optimiser l'ensemble des points de départ $\{\mathbf{x}_0^{(1)}\}$, et le métamodèle de Krigeage \tilde{f}^{-1} est entraîné.
3. Les zones de solutions $\mathcal{P}_O^{(1)}$ non-dominées dans l'espace des objectifs sont identifiées.
4. Des tirages aléatoires sont effectués dans les zones $\mathcal{P}_O^{(1)}$ et le métamodèle \tilde{f}^{-1} est utilisé pour obtenir les nouveaux points de départ $\{\mathbf{x}_0^{(1)}\}$.

Les zones où les solutions sont les moins nombreuses, sont aussi les zones de $\mathcal{P}_O^{(1)}$ les plus vastes. Par conséquent, l'utilisation de tirage uniforme dans l'ensemble $\mathcal{P}_O^{(1)}$ aura tendance à générer des points de départ dans des zones faibles en information, ne contenant pas de solution identifiée à l'issue des salves précédentes. De cette manière, on espère pouvoir venir enrichir le front de Pareto en générant les nouveaux points de départ dans cette zone.

De même, on pourra se baser sur l'indicateur d'erreur donné par le métamodèle en sortie, pour modifier le nombre d'itérations alloué à *SMGDA* pour cette salve d'optimisation. Dans les applications numériques, on limitera la réduction du nombre d'itérations à la moitié du nombre d'itération alloué pour la première salve N_1 .

5.1.2 Construction d'un estimateur et modification du vecteur de descente.

Lors des premiers essais numériques, nous avons été confrontés à un phénomène de glissement des solutions sur le front de Pareto. En effet, *SMGDA* donne rapidement des solutions proches du front de Pareto et ce de façon dispersée, mais en prolongeant l'algorithme sur quelques itérations de plus, on a constaté un phénomène de glissement de ces solutions vers des zones restreintes du front. Ce phénomène de glissement est illustré dans la Figure 5.1 pour le problème d'optimisation biobjectif $\min \{\mathbb{E}[f_1], \mathbb{E}[f_2]\}$,

TABLE 5.1 Exemple du déroulement de l'Algorithme 3

Étape	Espace des variables	Espace des objectifs
1		
2		
3		
4		

où les objectifs sont donnés par les expressions suivantes :

$$\begin{cases} f_1(\mathbf{x}, W) = -\frac{1}{2} + |x + W_1 + 1| + \frac{1}{2}(y + W_2)^2 \\ f_{2,1}(\mathbf{x}, W) = \sqrt{W_3 \cdot (x + y)^2} \\ f_{2,2}(\mathbf{x}, W) = (x + y)^2 \\ f_2(\mathbf{x}, W) = \min(f_{2,1}, f_{2,2}), \end{cases}$$

et dont les composantes du vecteur aléatoire W suivent les lois

$$\begin{cases} W_1 \rightarrow 0.3 \cdot \mathcal{U}(-1, 1) \\ W_2 \rightarrow 0.3 \cdot \mathcal{U}(-1, 1) \\ W_3 \rightarrow (\mathcal{N}(0, 1))^2, \end{cases}$$

où $\mathcal{U}(-1, 1)$ désigne la variable aléatoire de loi uniforme entre $[-1, 1]$ et $\mathcal{N}(0, 1)$ la variable normale centrée réduite.

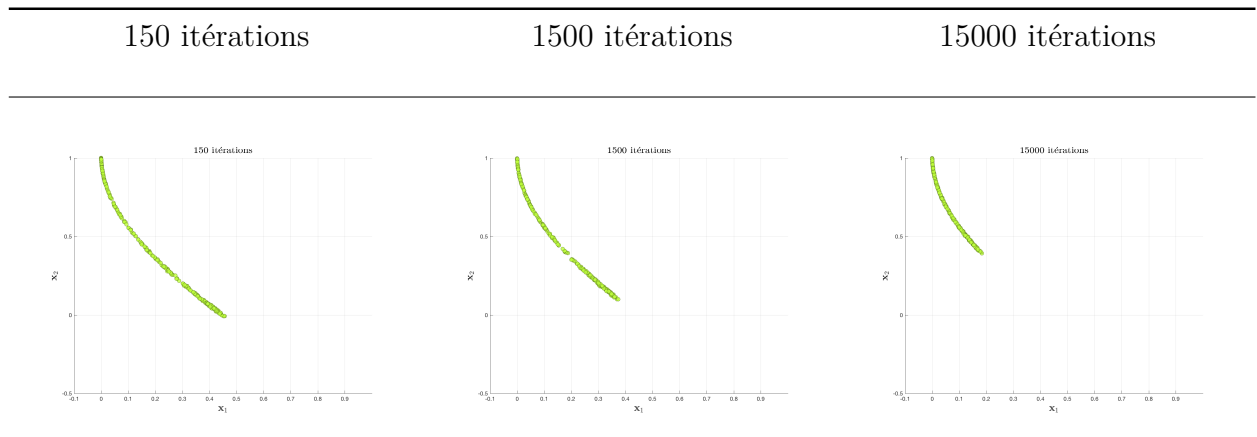


FIGURE 5.1 Phénomène de glissement sur le front de Pareto

Les objectifs constituant ce problème d'optimisation sont tous deux non-différentiables, et sont représentés pour un tirage aléatoire fixé ainsi que leurs moyennes sur 5000 échantillons sur la Figure 5.2. On peut notamment remarquer que la moyenne stochastique de ces objectifs a tendance à lisser les fonctions objectifs.

Le phénomène de glissement observé à la Figure 5.1 peut s'expliquer par le fait que le vecteur de descente choisi ne correspond pas au vecteur de descente du problème moyen : sa norme ne devient donc jamais nulle. Nous allons modifier la construction de ce vecteur de descente de façon à ce qu'il converge au cours des itérations vers le

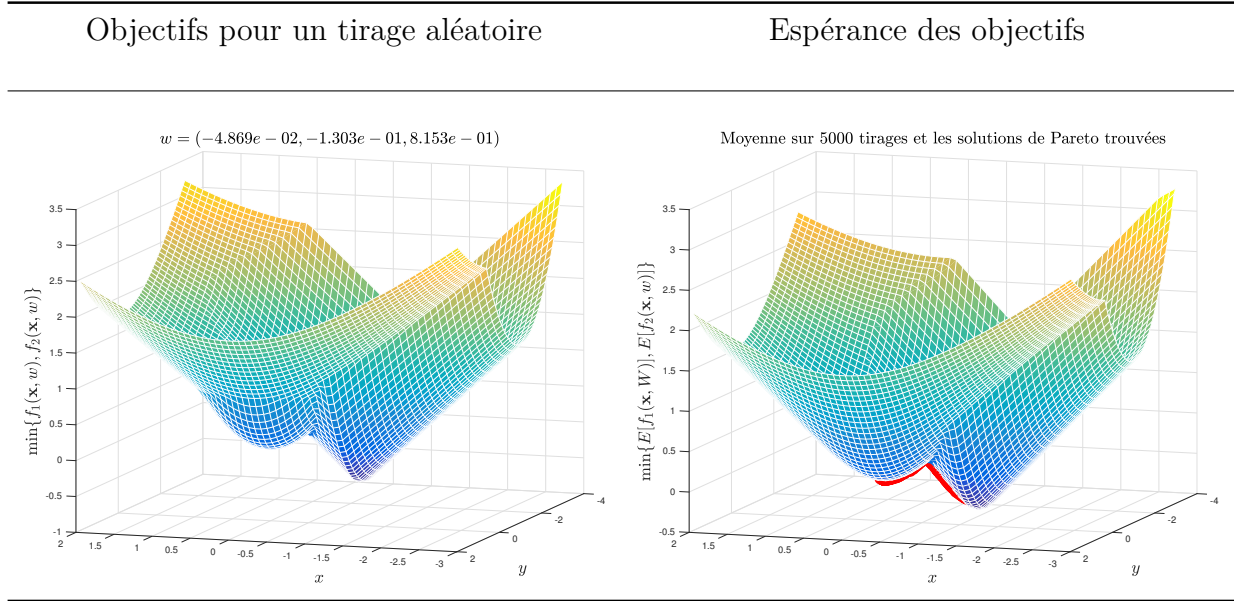


FIGURE 5.2 Objectifs du problème traité pour un tirage aléatoire unique et leur moyenne sur 5000 tirages

vecteur de descente associé au problème en espérance.

$$\min_{\mathbf{x} \in \mathcal{X}} \{ \mathbb{E} [f_1(\mathbf{x}, W(\omega))], \dots, \mathbb{E} [f_m(\mathbf{x}, W(\omega))] \}$$

Plus précisément, on cherche à construire un vecteur de descente associé aux estimateurs des espérances des objectifs sur N_e échantillons

$$\min_{\mathbf{x} \in \mathcal{X}} \left\{ \frac{1}{N_e} \sum_{j=1}^{N_e} f_1(\mathbf{x}, W_j(\omega)), \dots, \frac{1}{N_e} \sum_{j=1}^{N_e} f_m(\mathbf{x}, W_j(\omega)) \right\}. \quad (5.1)$$

Afin d'éviter l'utilisation d'un estimateur basé sur une méthode *Sample Average Approximation*, on cherche à profiter des itérations précédentes pour construire l'estimateur. Pour cela, on considère les estimateurs récursifs, construits au cours des itérations, des espérances des fonctions objectifs et de leur gradients ou sous-gradients

$$\begin{aligned} \hat{f}_{i,k}(\mathbf{x}_k) &= \frac{1}{N_e} \sum_{j=1}^{N_e} f_i(\mathbf{x}_{k-j}, w_{k+1-j}), \quad k \geq N_e \\ \partial^{\mathcal{C}} \hat{f}_{i,k}(\mathbf{x}_k) &= \frac{1}{N_e} \sum_{j=1}^{N_e} \partial^{\mathcal{C}} f_i(\mathbf{x}_{k-j}, w_{k+1-j}), \quad k \geq N_e. \end{aligned} \quad (5.2)$$

La relation (5.2) est trivialement vérifiée dans le cas où les fonctions sont différentiables. Elle l'est également pour des fonctions objectifs régulières pour la sous-différentielle au vue du Théorème 3 énoncé au Chapitre 2. Le vecteur de descente alors choisi est l'élément de norme minimale de l'ensemble convexe engendré par l'union des estimateurs des sous-différentielles

$$\zeta(\mathbf{x}_k) = \operatorname{argmin} \left\{ \|\mathbf{u}\|, \mathbf{u} \in \operatorname{Conv} \left(\bigcup_{i=1}^m \partial^{\mathcal{C}} \hat{f}_i(\mathbf{x}_k) \right) \right\}.$$

Le vecteur $\zeta(\mathbf{x}_k)$ peut être considéré comme une approximation du vecteur de descente commun associé au problème (5.1) lorsque k est suffisamment grand et si les points $\{\mathbf{x}_{k-j}\}_{j \in \llbracket 1, N_e \rrbracket}$ restent suffisamment proches

$$\forall j \in \llbracket 1, N_e \rrbracket, \mathbf{x}_{k-j} \in \mathcal{B}(\mathbf{x}_k, \rho), \rho \ll 1,$$

ce qui est le cas par exemple lorsque le point \mathbf{x}_k est proche des solutions de Pareto.

Sur l'exemple précédent, le front de Pareto obtenu en utilisant ce nouveau vecteur de descente est représenté sur la Figure 5.3. On peut observer une bien meilleure stabilité des solutions et une meilleure dispersion. Le phénomène de glissement a disparu.

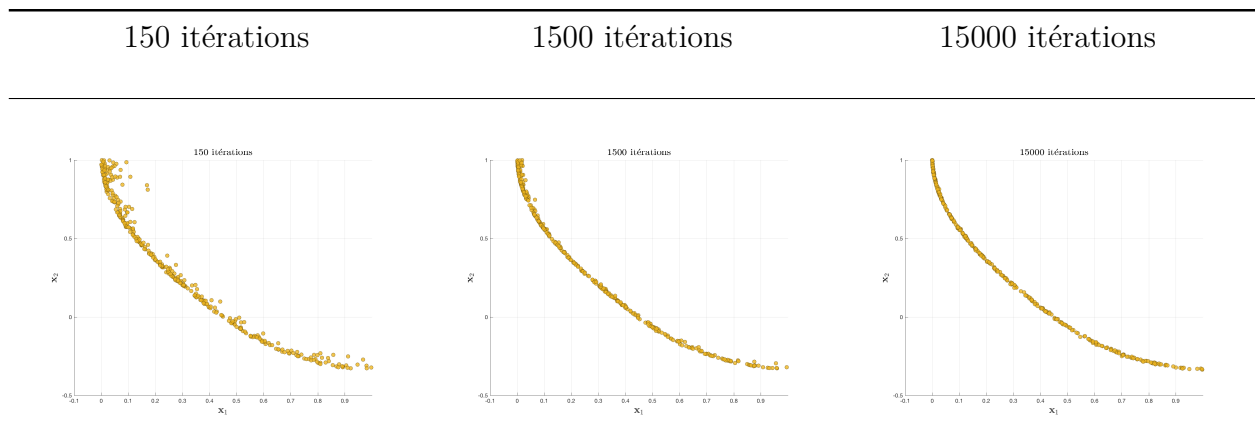


FIGURE 5.3 Résultats obtenus en utilisant le nouveau vecteur basé sur l'estimateur

On a représenté sur la Figure 5.4 l'histogramme des normes des vecteurs de descente commun calculées en chaque point solution et pour la 15000^{ième} itération du processus d'optimisation, lorsque le vecteur est calculé par l'intermédiaire de l'estimateur (orange) ou comme proposé initialement dans l'algorithme 2 (vert). On observe entre un et deux ordres de grandeur d'écart sur la norme calculée, ce qui explique la stabilité et le non glissement des solutions sur le front lorsque l'estimateur est utilisé.

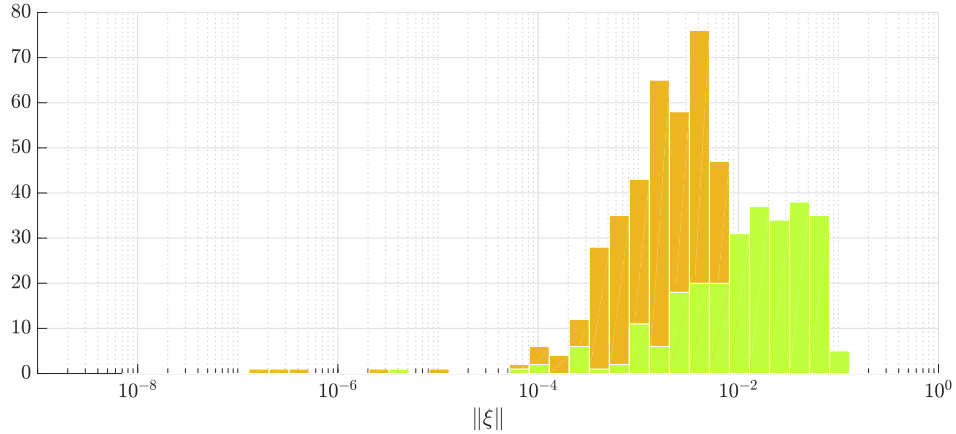


FIGURE 5.4 Normes des vecteurs de descente calculées aux points finaux pour 15000 itérations : en orange, en utilisant l'estimateur et en vert, en utilisant le vecteur de descente commun initialement proposé

5.2 Choix des algorithmes de comparaison

Afin de comparer les performances de *SMGDA* aux algorithmes de la littérature, nous avons sélectionné deux algorithmes : un algorithme génétique et un algorithme de recherche directe. Le choix de l'algorithme génétique s'est porté sur *NSGA-II* développé par Deb *et al.* [19]. Il s'agit d'un algorithme génétique dont le classement des solutions à chaque génération est fait selon un critère de non-dominance. Ainsi, des fronts successifs de points équivalents au sens de Pareto peuvent être tracés à chaque génération et servent à classer les individus. Le second algorithme qui sera testé dans les benchmarks effectués dans la suite du chapitre est un algorithme de recherche dit directe : *DMS*, pour *Direct MultiSearch*. Il apparaît pour la première fois dans l'article de Custódio *et al.* [16]. Il s'agit en fait de la généralisation au cadre multiobjectif des méthodes de recherche directionnelle.

Les deux algorithmes présentés précédemment ne sont pas prévus pour résoudre les problèmes stochastiques multiobjectifs. Par conséquent, on utilisera un estimateur de type *Sample Average Approximation* pour estimer les espérances des fonctions objectives, et on remplace le problème à résoudre initialement par le problème déterministe suivant :

$$\min_{\mathbf{x} \in \mathcal{X}} \left\{ \frac{1}{N_s} \sum_{k=1}^{N_s} f_1(\mathbf{x}, w_k^{(1)}), \dots, \frac{1}{N_s} \sum_{k=1}^{N_s} f_m(\mathbf{x}, w_k^{(m)}) \right\},$$

où les $w_k^{(i)}$, $i = 1, \dots, m$ sont des tirages aléatoires indépendants du vecteur aléatoire $W(\omega)$. Le calcul de l'estimateur des espérances doit être effectué à chaque itération de l'algorithme d'optimisation, ce qui rend l'approche numérique coûteuse. Par ailleurs, dans la suite, on parlera de manière équivalente du temps de calcul et du nombre d'appels aux fonctions objectifs.

Avant de présenter les résultats obtenus, on présente les outils utilisés pour comparer et analyser les solutions obtenues par les trois algorithmes.

5.3 Mesures de performance et profils de performance

Il existe un grand nombre de mesures qui permettent d'analyser les résultats d'une optimisation multiobjectif. La plupart des mesures sont présentées dans l'article de Riquelme *et al.* [65] qui compare leur fréquence d'utilisation dans la littérature ainsi que leurs défauts et avantages. Quand on analyse un front de Pareto, deux facteurs entrent en jeu. Dans un premier temps, la performance pure des solutions trouvées par un algorithme. Ces mesures quantifient l'appartenance des solutions au front de Pareto. C'est le cas par exemple de la mesure de Pureté [16] qui sera utilisée ici.

La procédure pour mesurer la pureté des solutions d'un algorithme consiste à créer, à partir des solutions de tous les algorithmes, un front de Pareto de référence. Puis de faire un comptage pour chaque algorithme du nombre de solutions faisant partie du front de référence ainsi construit. Soit $\mathcal{S} = \{SMGDA, DMS, NSGA-II\}$ l'ensemble de solveurs, et $\mathcal{P} = \{p_1, \dots, p_n\}_{n \in \mathbb{N}}$ un ensemble de problèmes d'optimisation. On désigne le front de Pareto trouvé par le solveur $s \in \mathcal{S}$ pour le problème $p \in \mathcal{P}$ par $F_{p,s}$. Pour un problème $p \in \mathcal{P}$, le front de référence est calculé de la manière suivante

$$F_p = \{\mathbf{y}^* \in \bigcup_{s \in \mathcal{S}} F_{p,s}; \forall \mathbf{y} \in \bigcup_{s \in \mathcal{S}} F_{p,s}, \mathbf{y} \preceq_{\mathcal{P}} \mathbf{y}^* \Rightarrow \mathbf{y} = \mathbf{y}^*\}.$$

Autrement dit, il s'agit de l'ensemble des points non-dominés trouvés par l'union des solutions trouvées par tous les solveurs. La pureté d'un solveur s pour le problème p est alors donnée par

$$\bar{t}_{p,s} = \frac{|F_{p,s} \cap F_p|}{|F_{p,s}|},$$

où $|\cdot|$ désigne le cardinal d'un ensemble. Ainsi, plus un algorithme fournit de solutions appartenant au front de référence, plus sa note est grande.

Un autre type de mesure est aussi très utilisé et correspond à la mesure de la dispersion des solutions sur le front de Pareto. En effet, pour obtenir une description satisfaisante du front de Pareto, il faut que les solutions trouvées par un algorithme soient suffisamment dispersées sur le front c'est à dire que le front de Pareto construit par l'algorithme soit le plus étendu possible. Nous avons choisi d'utiliser la mesure la plus répandue, qui cumule à la fois mesure de performance et de dispersion, c'est la mesure d'hypervolume. Partant d'un point de référence \mathbf{r} bien choisi [5], cette mesure consiste à calculer le volume de l'union des hypervolumes générés entre le point de référence, et les solutions trouvées par un solveur s pour un problème p

$$\bar{t}_{p,s} = V \left(\bigcup_{\mathbf{y}^* \in F_{p,s}} HV(\mathbf{y}^*, \mathbf{r}) \right),$$

où V désigne le volume, et HV correspond à l'hypervolume généré entre le point de référence \mathbf{r} et une solution \mathbf{y}^* .

Afin d'illustrer les deux mesures précédentes, la Figure 5.5 présente les fronts de Pareto des solveurs pour un des problèmes traités. Le front de Pareto de référence est présenté en trait noir plein, il est construit avec l'ensemble des points non dominés de tous les solveurs. La mesure d'hypervolume est aussi présentée pour les solutions du solveur *DMS*. On peut voir sur cette illustration que plus un solveur a de solutions performantes et dispersées, plus le volume mesuré est grand.

Pour éviter d'avoir à présenter, pour chaque test réalisé, les fronts de Pareto de chaque solveur et leurs mesures associées, il existe des méthodes de représentation qui permettent de faire une concaténation des résultats sur l'ensemble des tests et d'exprimer la performance globale d'un algorithme d'optimisation. Ces méthodes sont appelées profils de performance [26] et correspondent à des fonctions de répartition générées par l'intermédiaire des mesures faites sur l'ensemble des résultats des tests. Les profils de performance nécessitent des mesures de performance décroissantes (plus l'algorithme est bon, plus sa note est basse), par conséquent les mesures $t_{p,s}$ considérées seront prises comme les inverses de la mesure de pureté ou de la mesure d'hypervolume

$$t_{p,s} = \frac{1}{\bar{t}_{p,s}}.$$

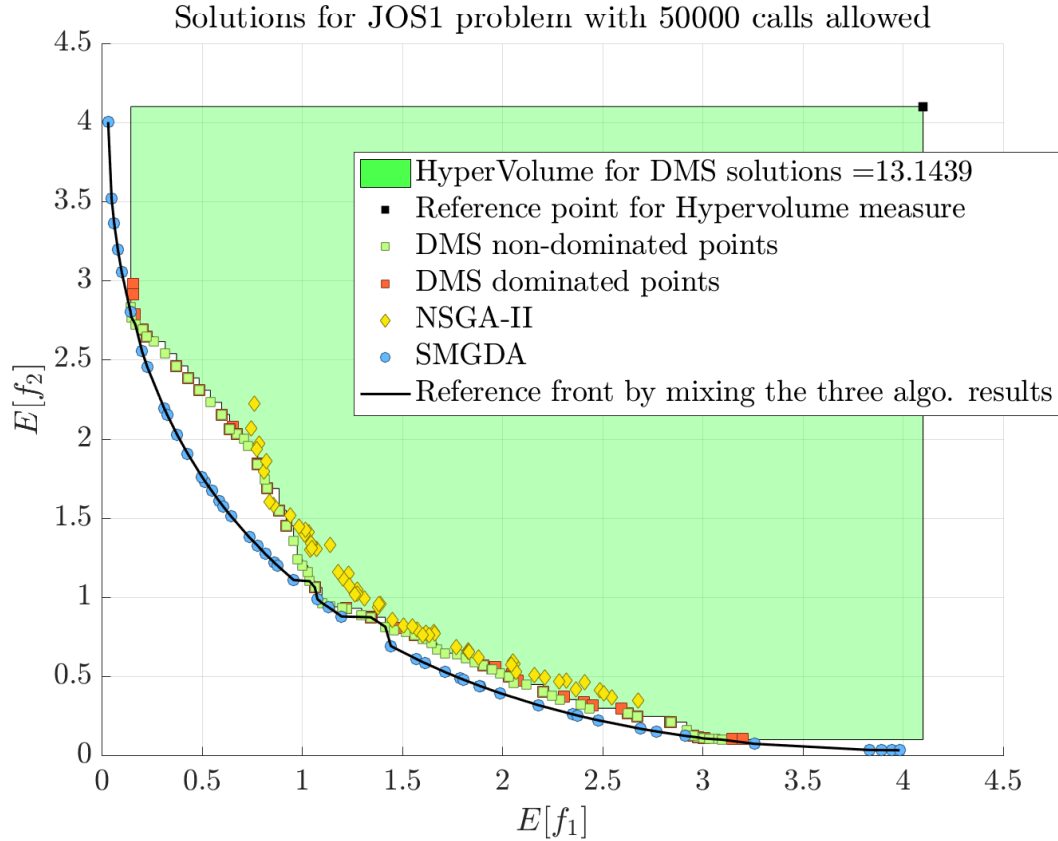


FIGURE 5.5 Exemple de calcul des mesures pour les solutions du problème JOS1

On introduit le ratio $r_{p,s}$ pour un solveur s et un problème p par la relation

$$r_{p,s} = \frac{t_{p,s}}{\min \{t_{p,\bar{s}}, \bar{s} \in \mathcal{S}\}}.$$

Ce qui correspond à la division de la note d'un solveur s pour un problème p par la note du meilleur solveur pour ce même problème p . Puis on introduit la fonction $\rho(\tau)$, qui donne le nombre de problèmes qui peut être considéré comme résolu pour un seuil τ par

$$\rho_s(\tau) = \frac{1}{|\mathcal{P}|} \times |\{p \in \mathcal{P}, r_{p,s} \leq \tau\}|.$$

Ainsi, pour $\tau = 1$, on obtient le nombre de problèmes qui ont été le mieux résolus suivant la mesure t analysée, pour un solveur s fixé. Lorsque $\rho(\tau)$ atteint la valeur 1, tous les problèmes ont été résolus par le solveur s pour un seuil τ . Un défaut majeur de ces profils de performance est qu'ils ne prennent pas en compte le coût en temps de calcul qu'un algorithme met pour avoir les résultats analysés. Le nombre d'appels

autorisés étant à la discrétion de l'utilisateur.

5.4 Benchmarks

Les trois algorithmes considérés dans ce chapitre ont été soumis à un benchmark composé de 10 problèmes déterministes comportant un nombre de paramètres d'optimisation très varié. La majorité des problèmes tests présents dans la littérature sont des problèmes déterministes, nous avons donc construit des problèmes stochastiques en introduisant artificiellement des variables aléatoires dans ces exemples. L'ensemble des tests effectués est décrit dans le Tableau 5.2. Un second tableau situé dans les annexes décrit précisément les fonctions objectifs de chaque test du benchmark (A.2, A.3, A.4). La version déterministe de ces cas peut être retrouvée dans les publications [38, 76, 39].

TABLE 5.2 Fonctions test du benchmark

	Nom	dim (\mathbf{x})	dim ($W(\omega)$)	Appels aux objectifs
[38]	Fonseca-Fleming	15	30	10^5
[38]	MOP3	2	18	10^4
[38]	MOP6	2	3	10^4
[76]	ZDT{1,2,3}	32	30	$5 \cdot 10^5$
[39]	JOS1	30	60	$5 \cdot 10^5$
[39]	JOS2	30	32	$5 \cdot 10^5$
[38]	SCH1	1	4	10^4
[38]	IM1	2	3	10^4

Le nombre total d'appels aux objectifs est donné dans le Tableau 5.2. Il correspond au nombre de fois qu'un algorithme est autorisé à calculer l'ensemble des fonctions objectifs pour un jeu de variables d'optimisation, et un tirage de la variable $W(\omega)$.

Le nombre d'appels aux objectifs est considéré représentatif du coût en terme de temps de calcul, puisque la complexité des objectifs est responsable de la majorité du temps à consommer particulièrement quand les objectifs sont composés de modèles à grand nombre de degrés de liberté. À titre d'exemple, pour les cas analytiques qui sont traités dans ce chapitre, un processus complet d'optimisation de *SMGDA* dure en temps réel moins de deux minutes sur un ordinateur composé de sept cœurs de calcul. Pour *SMGDA*, ce nombre d'appels autorisés a été fixé 10 fois moindre que le nombre alloué aux deux autres algorithmes concurrents. La performance finale des solutions obtenues par les algorithmes est obtenue par simulation de Monte Carlo avec 10^5 tirages aléatoires. En effet, les mesures de performance nécessitent la construction du front de Pareto du problème moyen, qu'il est nécessaire d'estimer.

5.4.1 Réglage des algorithmes

Les algorithmes d'optimisation sont très sensibles aux choix des paramètres qui les définissent. Afin de pouvoir offrir une comparaison juste et honnête des algorithmes, nous avons choisi dans un premier temps les paramètres pour lesquels ces algorithmes donnent les meilleurs résultats. Ce choix s'est fait en utilisant une procédure d'optimisation qui maximise la mesure d'hypervolume pour chaque cas de test. Les paramètres optimisés comprennent l'ensemble des réglages possibles sur chacun des algorithmes ainsi que le partage des appels aux objectifs entre la boucle d'approximation des espérances et les itérations de l'algorithme d'optimisation. La maximisation de la mesure d'hypervolume a été effectuée par l'intermédiaire d'un algorithme génétique sur 15 générations et une population de 20 individus. Étant donné le caractère semi-stochastique de *NSGA-II*, et l'aspect incertain des problèmes d'optimisation à résoudre, nous avons choisi comme fonction coût à maximiser la moyenne obtenue sur 5 résolutions successives de la mesure d'hypervolume. Pour *NSGA-II* et un problème p , on peut résumer ce nouveau problème d'optimisation par

$$\max_{\theta} \left(\frac{1}{5} \sum_{i=1}^5 t_{p,i} \left(\min_{\mathbf{x} \in \mathcal{X}}^{NSGA(\theta)} (\mathbb{E}[f(\mathbf{x}, W)]) \right) \right),$$

où θ désigne le jeu de paramètres de l'algorithme utilisé et $t_{p,i}$ la mesure d'hypervolume pour la $i^{\text{ième}}$ résolution du problème d'optimisation. Les jeux de paramètres utilisés pour chaque algorithme et pour chaque cas de test peuvent être vu dans le Tableau A.1 donné en annexe.

5.4.2 Résultats

Une fois le jeu idéal de paramètres déterminé, les algorithmes *NSGA-II* et *DMS* sont appliqués 5 fois pour chaque problème du benchmark. Seule la meilleure solution pour la mesure d'hypervolume a été conservée. Sur la Figure 5.6 est représenté le profil de performance des trois algorithmes pour la mesure de pureté. *SMGDA* réalise un très bon score en comparaison aux deux autres algorithmes et ce malgré une quantité d'appels aux objectifs bien inférieure. En effet, *SMGDA* a été capable de résoudre mieux que ses concurrents huit des dix problèmes du benchmark. De plus la valeur $\rho = 1$ est atteinte pour un seuil τ inférieur à 2, ce qui signifie que pour le cas de test que *SMGDA* a le moins bien réussi, il n'a été que deux fois moins performant que le meilleur parmi *DMS* ou *NSGA-II*. Les deux autres algorithmes n'ont pas été capables d'atteindre la valeur $\rho = 1$, cela signifie que pour certains tests ces algorithmes n'ont pas réussi à trouver un point non-dominé.

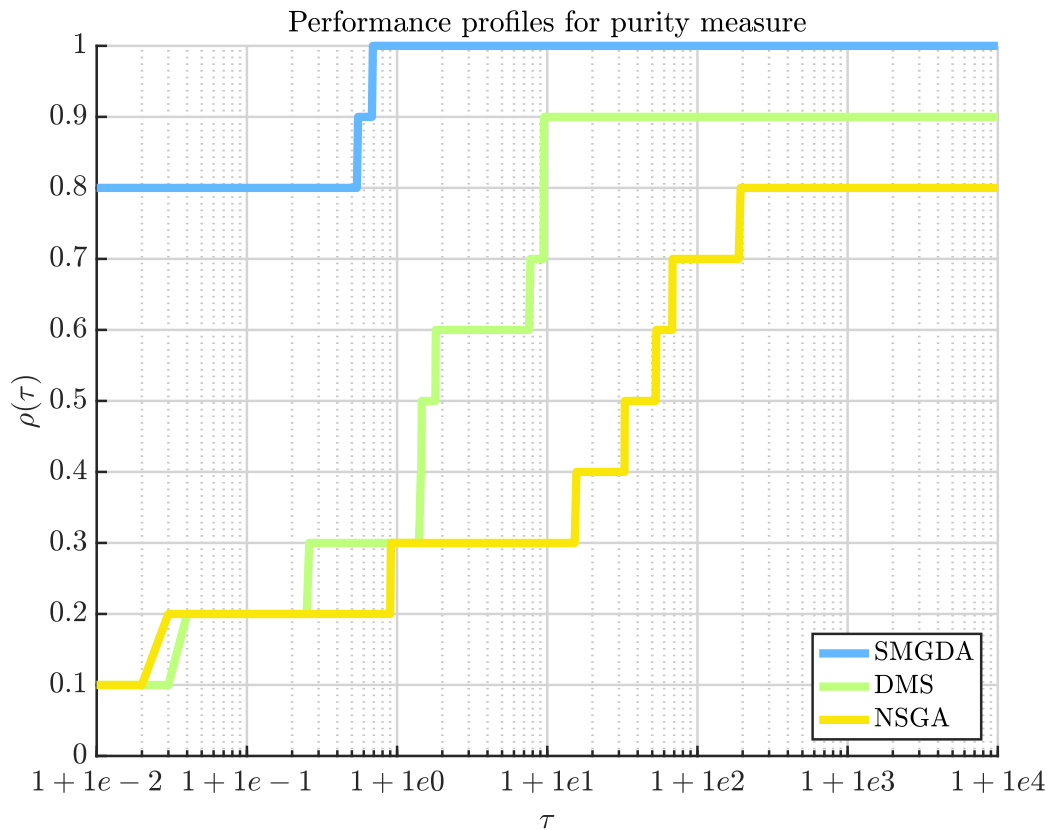


FIGURE 5.6 Profil de performance pour la mesure de pureté

Ces résultats étaient en partie attendus pour la mesure de pureté, puisque seul *SMGDA* parmi les trois algorithmes possède des résultats de convergence vers les solutions de Pareto du problème. Cependant, certains tests ne satisfont pas aux hypothèses nécessaires à la convergence. Notamment le test MOP6 qui est observable sur la Figure 5.7 et pour lequel il existe des points Pareto stationnaires qui ne sont pas des solutions de Pareto du problème d'optimisation.

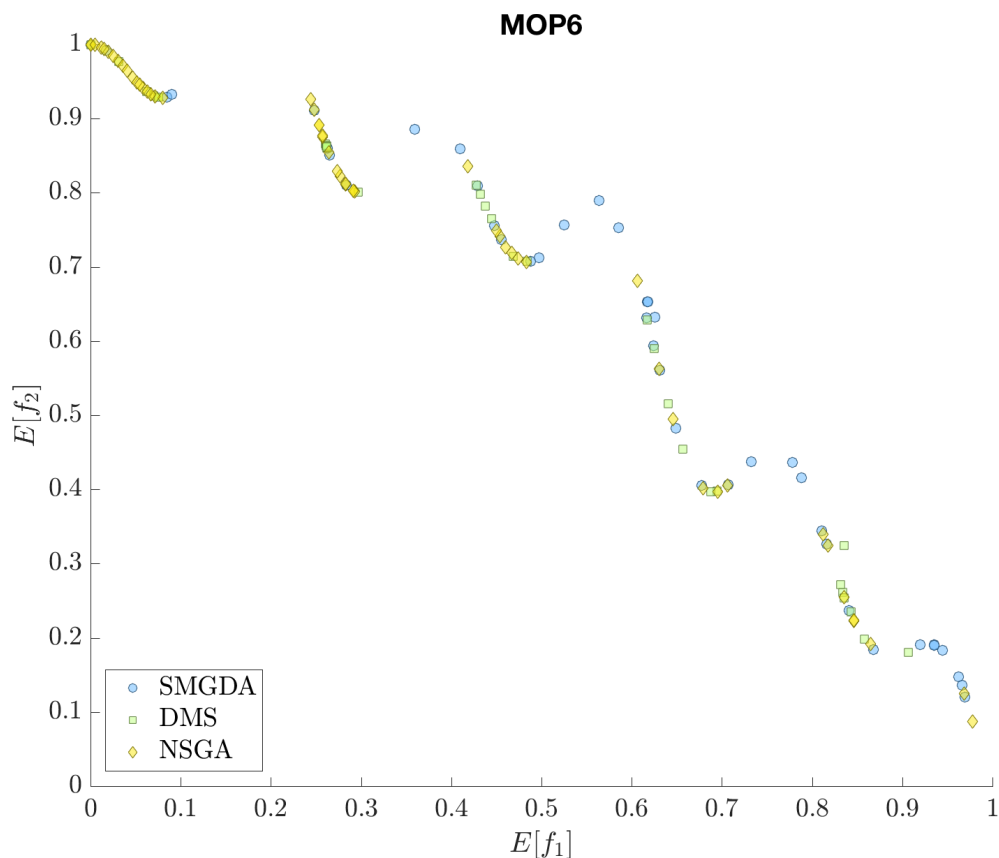


FIGURE 5.7 Solutions des algorithmes pour le cas de test MOP6

La mesure d'hypervolume rend, quant à elle, des résultats plus équilibrés puisque *SMGDA* est désormais le meilleur algorithme pour seulement six cas de test sur dix. Le profil de performance associé à la mesure d'hypervolume peut être vu sur la Figure 5.8. Cette fois, l'ensemble des algorithmes atteignent la valeur $\rho = 1$ pour un seuil τ inférieur à 2. *NSGA-II* et *DMS* ayant été optimisés pour rendre la meilleure note d'hypervolume, il est logique que les deux algorithmes fassent mieux que pour la mesure précédemment analysée. *SMGDA* reste tout de même plus performant que ses

deux concurrents pour le benchmark, puisqu'il est capable de résoudre l'ensemble des problèmes pour un seuil $\tau = 1.02$.

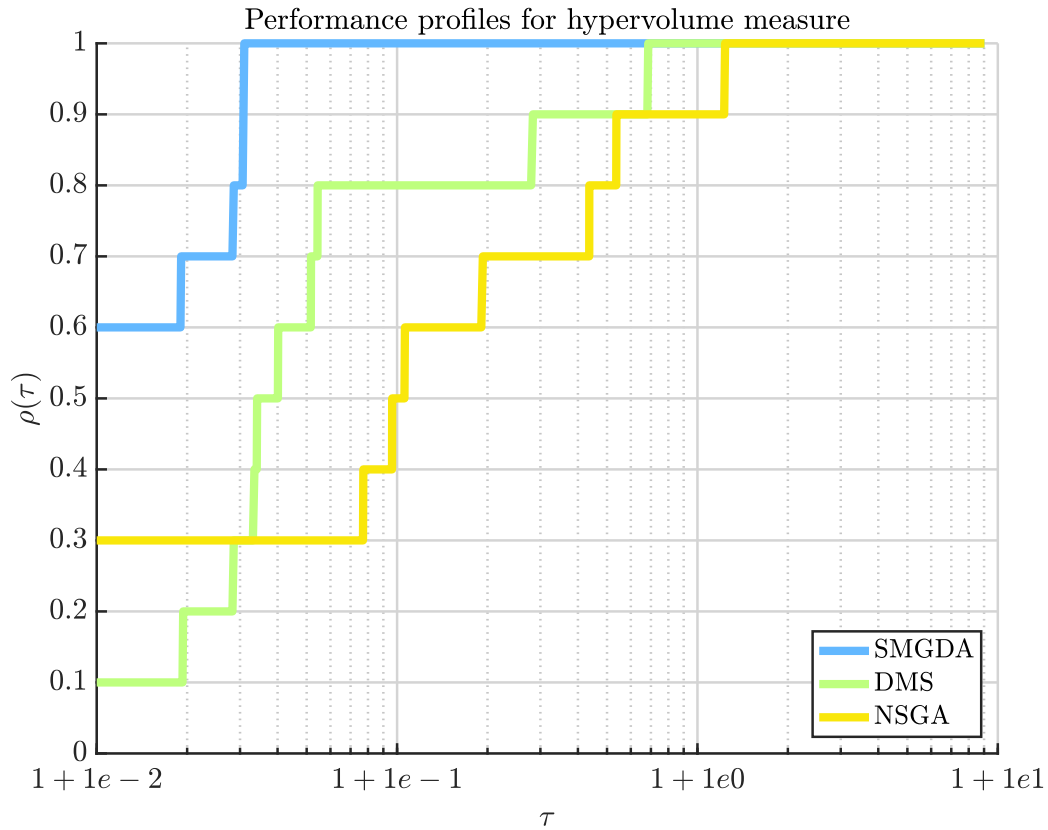


FIGURE 5.8 Profil de performance pour la mesure d'hypervolume

Dans l'ensemble, *SMGDA* semble être un algorithme très performant en considérant le benchmark étudié. De plus, il nécessite moins d'appels aux objectifs que les deux concurrents. Cette rapidité et précision des résultats est grandement due aux résultats de convergence obtenus et au fait que les espérances des objectifs n'ont jamais besoin d'être estimées. On pourra tout de même noter que si le gradient des objectifs n'est pas disponible, une estimation par différence finie pourrait pour certains tests de grande dimension rendre *SMGDA* plus coûteux que ses deux algorithmes concurrents. De plus, lorsque les objectifs ne satisfont pas aux hypothèses de convergence, *SMGDA* peut se retrouver piégé sur des points Pareto stationnaires non nécessairement solutions du problème d'optimisation.

Les problèmes apparaissant dans le benchmark traité dans ce chapitre étaient jusqu'à présent des problèmes d'optimisation sans contraintes. Nous allons voir dans le

chapitre suivant comment des contraintes déterministes ou probabilistes peuvent être résolues à l'aide de l'algorithme *SMGDA*.

Chapitre 6

Prise en compte de contraintes déterministes et probabilistes, deux méthodes

Nous allons dans ce chapitre étendre l'utilisation de l'algorithme *SMGDA* à des problèmes d'optimisation avec contraintes. Lorsque les contraintes sont déterministes nous utiliserons une méthode de pénalisation exacte. Nous montrerons ensuite que la solution d'un problème d'optimisation mono-objectif avec une contrainte probabiliste peut être interprété comme une solution particulière d'un problème multiobjectif stochastique, qu'on pourra alors résoudre par l'algorithme *SMGDA*. Des illustrations viennent éclairer les résultats.

6.1 Contraintes déterministes par pénalisation

Considérons tout d'abord le cas de contraintes déterministes. Notre propos se base sur l'étude d'un problème particulier d'optimisation. Plus précisément, nous reprenons une application traitée à l'origine par Leite [46] dans sa thèse : il s'agit d'optimiser un matériau sandwich trois couches sous une contrainte d'épaisseur totale. On considère une poutre en matériau sandwich (Figure 6.1). Les peaux du sandwich (surfaces extérieures) sont constituées de plaques d'aluminium d'épaisseur \mathbf{t}_u et \mathbf{t}_b considérées comme des variables d'optimisation. Le cœur du sandwich est constitué de nids d'abeilles symétriques en aluminium et d'épaisseur \mathbf{t}_c .

La géométrie des nids d'abeilles est illustrée sur la Figure 6.2 et le ratio épaisseur sur longueur ($\mathbf{R} = t/\ell$) des parois constituant les cellules est une variable. On considère

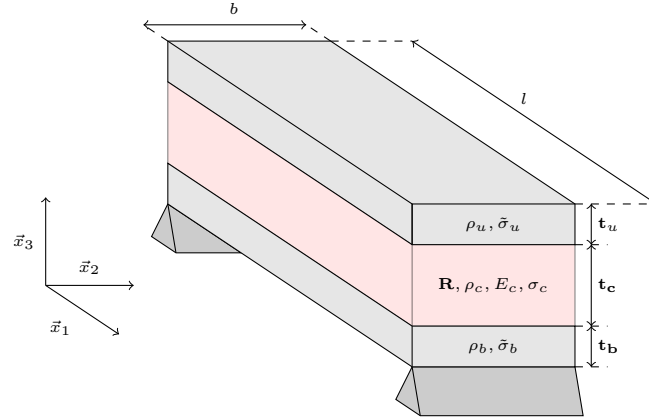


FIGURE 6.1 Matériau sandwich à optimiser

de plus que certains paramètres de ce matériau sont incertains, et on modélise ces incertitudes par le biais de variables aléatoires définies sur un espace probabilisé $(\Omega, \mathcal{T}, \mathbb{P})$. La qualité des aluminiums constituant les peaux est considérée comme aléatoire, et la résistance élastique peut varier en suivant une loi uniforme de la manière suivante

$$\begin{cases} \tilde{\sigma}_u = (1 + \mathcal{U}(\omega)) \cdot \sigma_{alu} \\ \tilde{\sigma}_b = (1 + \mathcal{U}(\omega)) \cdot \sigma_{alu} \end{cases}$$

où $\mathcal{U}(\omega)$ désigne une variable aléatoire de loi uniforme sur $[-0.2, 0.2]$ et σ_{alu} représente la résistance élastique nominale de l'aluminium qui est égale à 350 MPa. On considère de même que la fabrication des cellules nids d'abeille en aluminium peut engendrer des erreurs sur la géométrie des cellules par le biais de l'angle $\tilde{\theta}$, voir Figure 6.2

$$\tilde{\theta} = (1 + \mathcal{U}(\omega)) \cdot \theta,$$

où θ est l'angle nominal fixé à 30° . On utilisera la notation \mathbf{x} pour désigner le vecteur des variables d'optimisation

$$\mathbf{x} = [\mathbf{t}_u, \mathbf{t}_b, \mathbf{t}_c, \mathbf{R}],$$

et la notation W pour désigner le vecteur aléatoire

$$W(\omega) = [\tilde{\sigma}_u(\omega), \tilde{\sigma}_b(\omega), \tilde{\theta}(\omega)].$$

Les relations proposées par Gibson et Ashby [37] sont utilisées afin de déduire, en partant de la description des cellules, les caractéristiques matériau équivalentes. Les relations sont rappelées dans le Tableau 6.1 et montrent que la masse volumique du

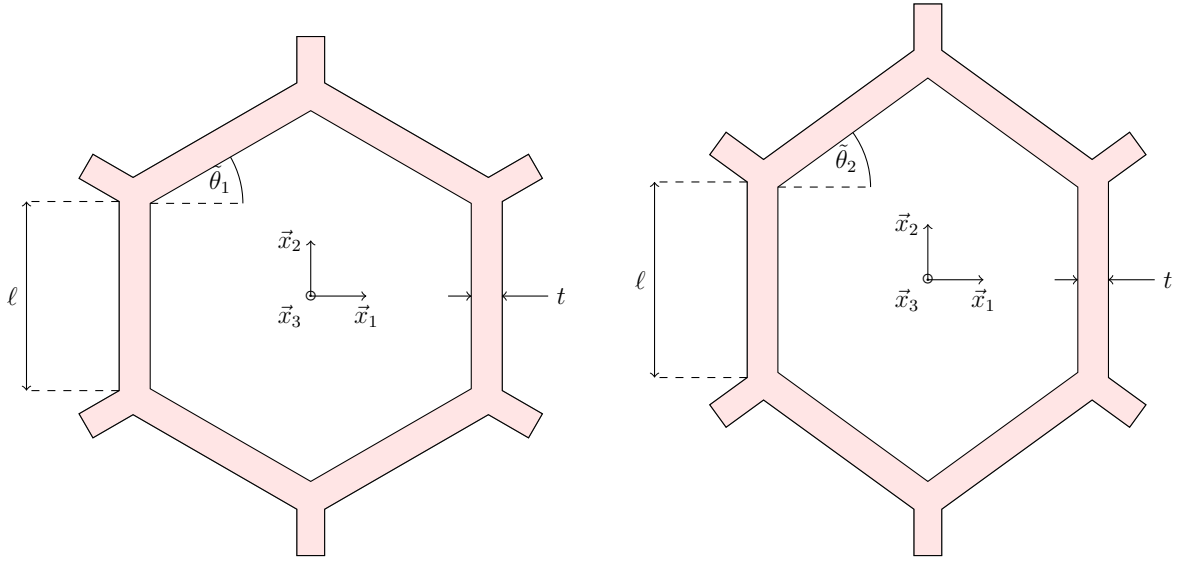


FIGURE 6.2 Deux échantillons de la cellule nids d'abeille

matériau équivalent $\rho_{\mathbf{R}}$ ainsi construit est une variable aléatoire dépendant de l'angle incertain $\tilde{\theta}$ et de la masse volumique du matériau constitutif des cellules ρ_c fixé à 2700 kg/m^3 .

TABLE 6.1 Propriétés matériau du cœur en fonction de \mathbf{R}

\mathbf{R}	t/ℓ
$\rho_{\mathbf{R}}$	$= \frac{3}{2 \cos(\tilde{\theta})(1+\sin(\tilde{\theta}))} \rho_c \mathbf{R}$
$\sigma_{\mathbf{R}}$	$= 5.6 (\mathbf{R})^{\frac{5}{3}} \sigma_c$

Le but de l'optimisation de ce matériau est de minimiser la masse du matériau par unité de surface

$$M(\mathbf{x}, W(\omega)) = \rho_u \mathbf{t}_u + \rho_{\mathbf{R}}(\omega) \mathbf{t}_c + \rho_b \mathbf{t}_b, \quad (6.1)$$

tout en maximisant la force critique qui induit un mode de défaillance en flexion. Suivant la conception du matériau sandwich, les modes de défaillance qui peuvent engendrer la ruine du sandwich ne répondent pas nécessairement aux mêmes mécanismes. On considère dans cet exemple deux modes de défaillance en flexion que l'on suppose représentatifs des mécanismes de ruine pouvant intervenir. Un premier mode de

défaillance correspond à l'indentation du cœur sous une charge en flexion imposée par un indentateur de taille $a = 1$ cm. La force critique qui induit ce mode de défaillance représentée sur la Figure 6.3 est donnée par la relation (6.2)

$$F_{c1}(\mathbf{x}, W(\omega)) = 2bt_u\sqrt{\sigma_{\mathbf{R}}\tilde{\sigma}_u(\omega)} + ab\sigma_{\mathbf{R}}. \quad (6.2)$$

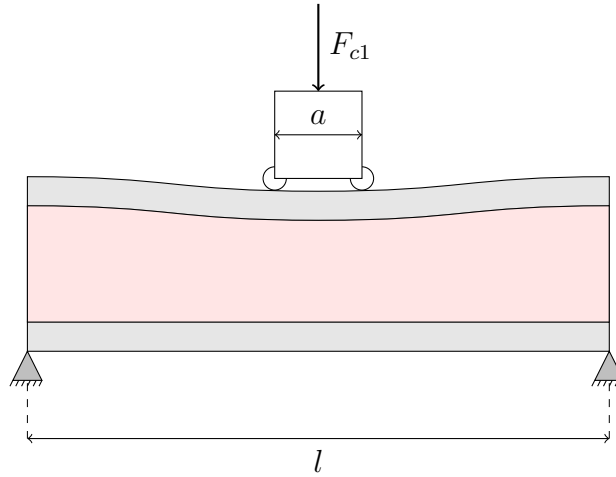


FIGURE 6.3 Indentation du cœur sous une charge critique F_{c1}

Le second mode de défaillance pris en compte est le dépassement de la résistance élastique de la peau inférieure sous une charge critique F_{c2} . Cette charge critique, illustrée sur la Figure 6.4, peut être calculée par l'intermédiaire de la relation (6.3)

$$F_{c2}(\mathbf{x}, W(\omega)) = \frac{4bt_b(\mathbf{t}_c + (\mathbf{t}_u + \mathbf{t}_b)/2)}{l}\tilde{\sigma}_b(\omega). \quad (6.3)$$

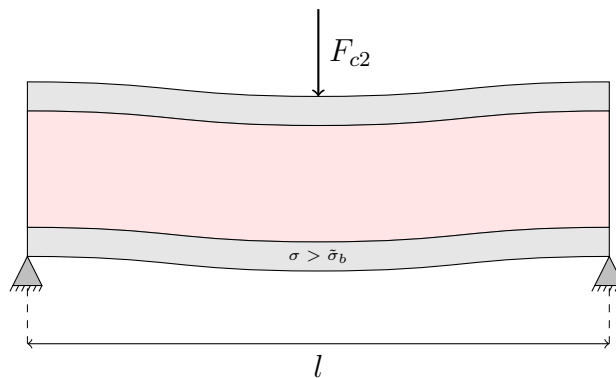


FIGURE 6.4 Dépassement de la résistance élastique de la peau inférieure sous une charge critique F_{c2}

Le but du problème d'optimisation est de retarder au maximum l'apparition des modes de défaillance, ce qui se traduit par la maximisation de la charge critique minimale entre F_{c1} et F_{c2} qui induit un mode de défaillance en flexion de la poutre :

$$F_c(\mathbf{x}, W(\omega)) = \min \{F_{c1}(\mathbf{x}, W(\omega)), F_{c2}(\mathbf{x}, W(\omega))\}.$$

On souhaite contrôler l'épaisseur du matériau sandwich en imposant une épaisseur maximale sous la forme d'une contrainte à ajouter au problème d'optimisation

$$e(\mathbf{x}) = \mathbf{t}_u + \mathbf{t}_c + \mathbf{t}_b \leq 25 \text{ cm.}$$

On peut donc écrire le problème d'optimisation complet sous la forme de l'équation (6.4) où la maximisation de la charge critique a été transformée en la minimisation de son opposé

$$\begin{cases} \min_{\mathbf{x} \in \mathcal{X}} \{ \mathbb{E} [M(\mathbf{x}, W(\omega))] , \mathbb{E} [-\min \{F_{c1}(\mathbf{x}, W(\omega)), F_{c2}(\mathbf{x}, W(\omega))\}] \} \\ \text{sous la contrainte } e(\mathbf{x}) \leq .25 \end{cases} \quad (6.4)$$

Pour ce problème particulier, on impose en plus des bornes sur les variables d'optimisation, contraintes décrites dans le tableau 6.2. Ces contraintes sont traitées par l'intermédiaire d'une projection sur les bornes à chaque itération de *SMGDA* quand les contraintes sont saturées.

TABLE 6.2 Contraintes de boîte \mathcal{X} sur les variables d'optimisation \mathbf{x}

variable		intervalle	unité
\mathbf{t}_u	\in	[0.03, 0.14]	m
\mathbf{t}_c	\in	[0.05, 0.32]	m
\mathbf{t}_b	\in	[0.03, 0.14]	m
\mathbf{R}	\in	[0.01, 0.2]	\emptyset

La contrainte géométrique $e(\mathbf{x})$ définissant un plan dans l'espace des variables, il aurait aussi été possible de la traiter par l'intermédiaire d'une projection. Mais

nous avons choisi, pour illustrer le traitement de contraintes déterministes, d'utiliser une formulation par pénalisation exacte. On définit alors la fonction de pénalisation suivante

$$g(\mathbf{x}) = C_p \min \{0, (1 - e(\mathbf{x})/.25)\},$$

où C_p désigne le coefficient de pénalisation. Ce coefficient, dans l'application, sera fonction du nombre n_c de fois consécutives où la contrainte est violée au cours des itérations de l'optimisation. On posera $C_p = 10^{n_c}$. On peut alors réécrire le problème d'optimisation :

$$\min_{\mathbf{x} \in \mathcal{X}} \{ \mathbb{E} [M(\mathbf{x}) - C_p g(\mathbf{x})], \mathbb{E} [-F_c(\mathbf{x}, W) - C_p g(\mathbf{x})] \} \quad (6.5)$$

On souhaite obtenir 49 points sur le front de Pareto. L'optimisation est réalisée sur un ordinateur possédant sept cœurs, le processus d'optimisation se déroule donc sur sept salves d'optimisation. La Figure 6.5 illustre l'algorithme qui génère les points initiaux sur les quatre premières salves d'optimisation. On remarque bien sur cet ensemble de figures que l'algorithme de génération a tendance à générer des points initiaux dans les zones sans solutions proches. Plus le nombre de solutions disponibles est grand, plus les zones identifiables contenant le front de Pareto se réduisent. Il en va de même pour la précision du métamodèle de Krigeage qui gagne en précision au fur et à mesure des salves d'optimisation déjà effectuées. La génération de points initiaux gagne alors en qualité, et le nombre d'itérations nécessaires à la convergence à chaque salve se réduit considérablement.

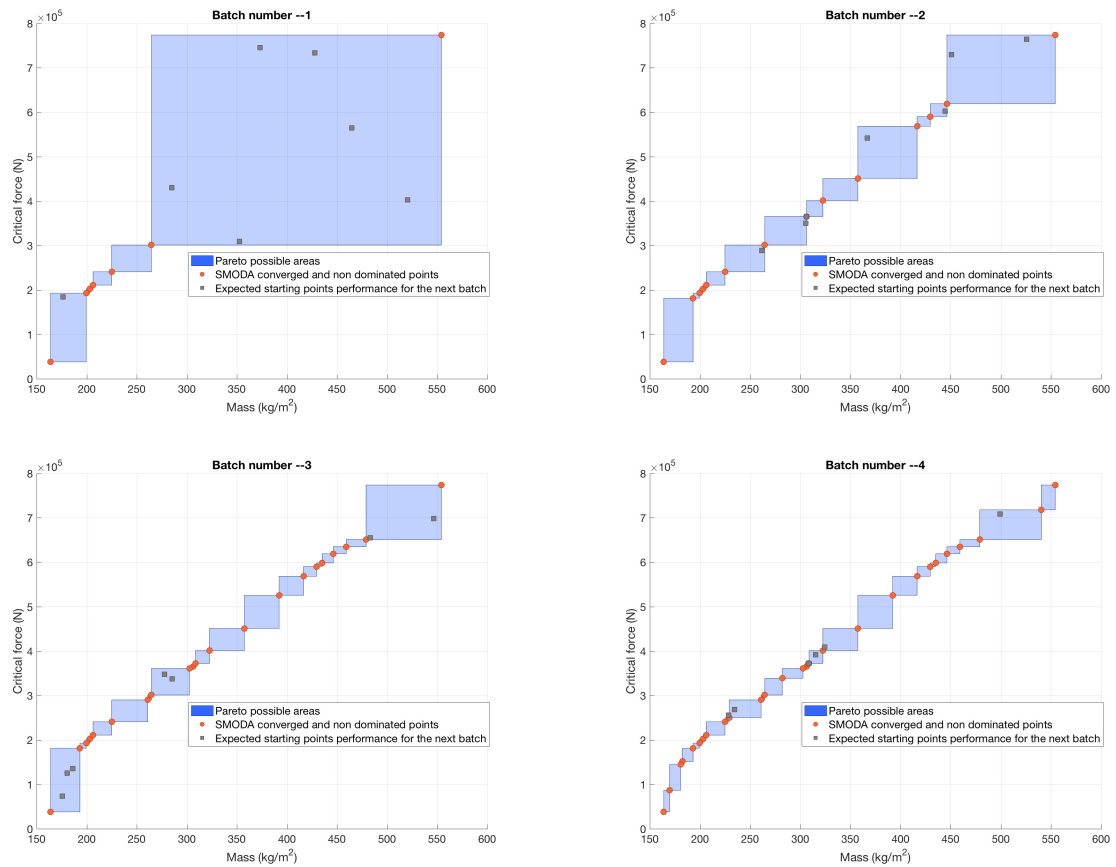


FIGURE 6.5 Génération des points initiaux sur les 4 premières salves

Le problème a été résolu par la suite avec *NSGA-II*, et on s'est donné un nombre de 250 appels aux fonctions objectifs par point à optimiser. Les résultats peuvent être observés sur la Figure 6.6 qui illustre les solutions dans l'espace des variables d'optimisation. La variable \mathbf{R} est représentée par l'intermédiaire de l'échelle de couleurs. La contrainte géométrique d'épaisseur maximale est décrite par un plan dans l'espace $(\mathbf{t}_u, \mathbf{t}_c, \mathbf{t}_b)$. On peut constater que la totalité des points valide la contrainte et ce, quel que soit l'algorithme d'optimisation utilisé.

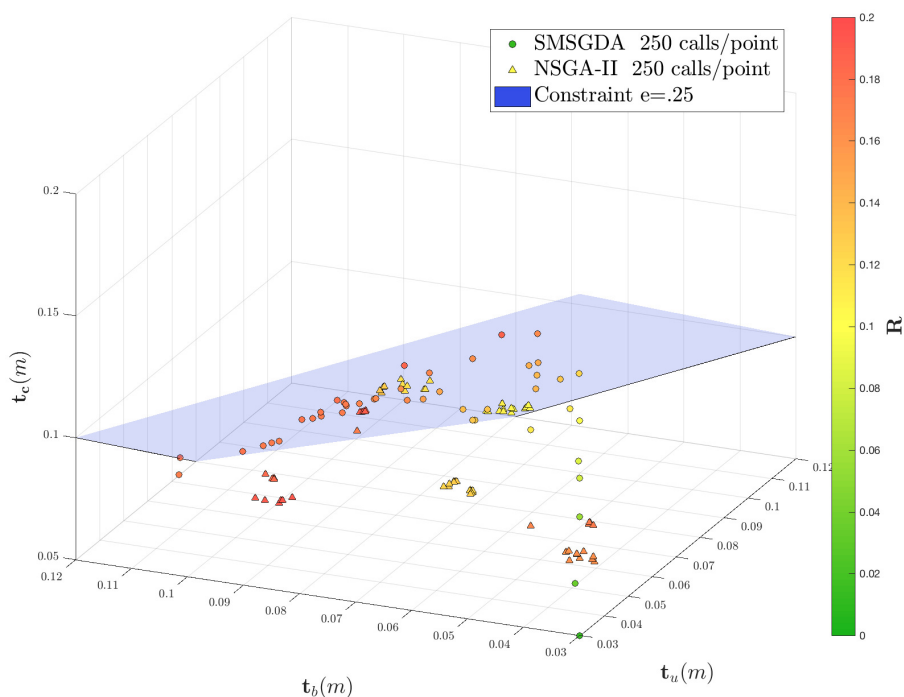


FIGURE 6.6 Solutions des algorithmes dans l'espace des variables d'optimisation

Les solutions obtenues par *SMGDA* dans l'espace des objectifs sont représentées sur la Figure 6.7, où sont également représentées les solutions obtenues par *NSGA-II* pour un budget de 250 et 25000 appels aux objectifs.

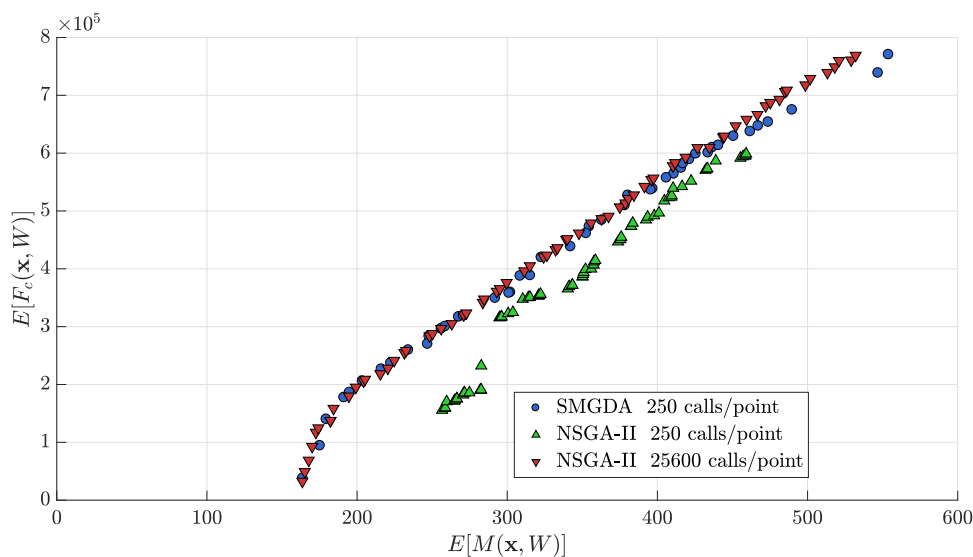


FIGURE 6.7 Solutions des algorithmes dans l'espace des objectifs

À nombre d'appels équivalent, *SMGDA* propose des solutions qui dominent les solutions de *NSGA-II* en plus d'être bien plus représentatives de l'ensemble du front de Pareto. *NSGA-II* nécessite un plus grand nombre d'appels (de l'ordre de 100 fois plus) pour parvenir à une description du front équivalente comme le montre la seconde résolution. La résolution de ce même problème d'optimisation sans la contrainte géométrique a aussi été menée avec *SMGDA*. La Figure 6.8 illustre la différence de front de Pareto obtenue lorsque la contrainte géométrique n'est plus présente dans le problème. On peut voir que la contrainte géométrique commence à devenir effective par rapport au problème sans contrainte à partir d'une masse surfacique de 450 kg/m^2 . Au delà, les deux fronts identifiés se séparent.

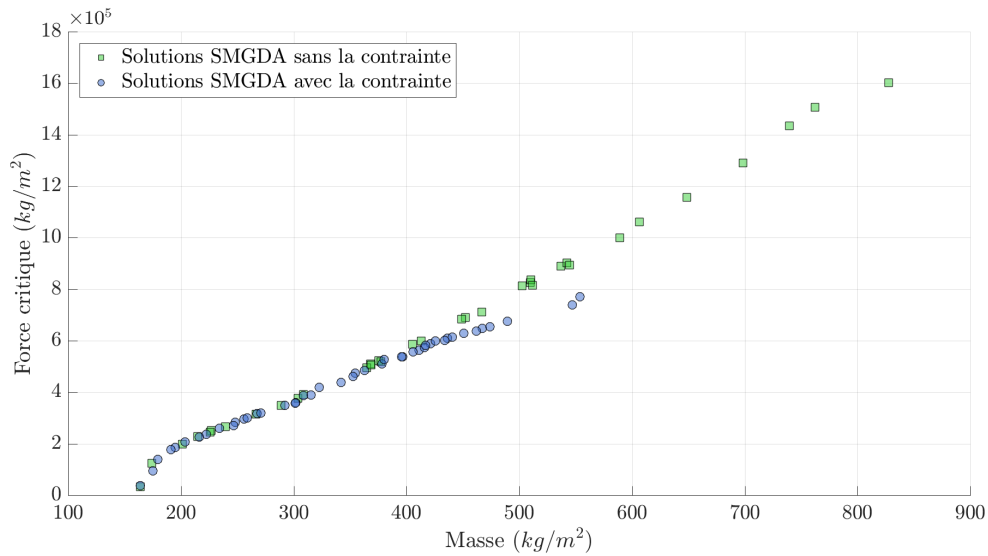
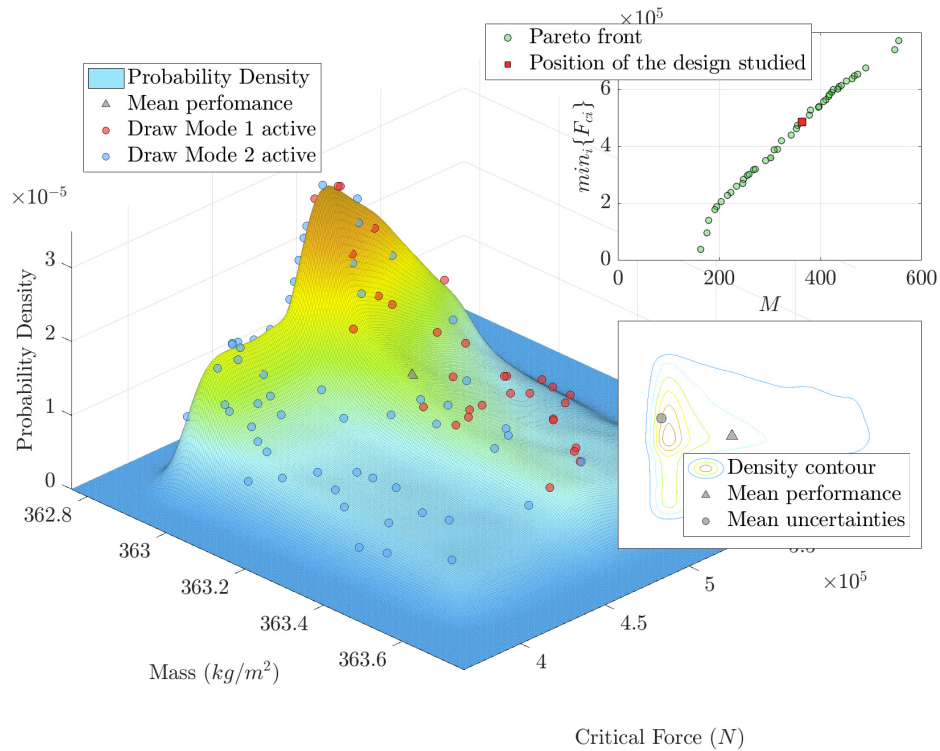


FIGURE 6.8 Solutions *SMGDA* pour le problème avec et sans contrainte

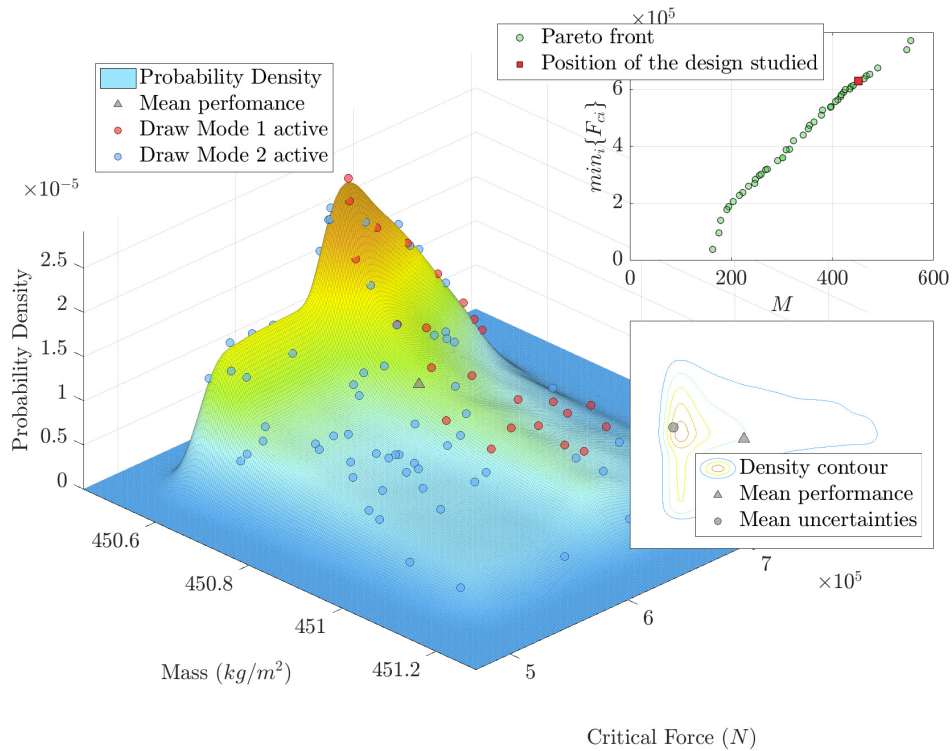
Partant d'une solution de *SMGDA* dans l'espace des variables d'optimisation, il est possible de reconstruire la densité de probabilité résultante dans l'espace des objectifs. Pour cela, on procède à une propagation des incertitudes en simulant 10^5 échantillons de la variable aléatoire W . Puis la densité de probabilité est reconstruite à l'aide d'un estimateur de densité à noyaux gaussiens. La construction de la densité de probabilité pour deux points \mathbf{y}_1 (resp. \mathbf{y}_2) solutions est représentée dans les Figures 6.9 (resp. 6.10). La reconstruction des densités permet en partie de comprendre les effets engendrés par la présence de variables aléatoires sur les objectifs du problème d'optimisation.

Pour chacune des deux densités, est donné en haut à droite le point du front analysé. La sensibilité de la masse aux incertitudes reste faible puisque la masse n'est soumise qu'à l'aléa sur la géométrie des cellules. Cependant, pour le calcul de la charge critique,

FIGURE 6.9 Densité de probabilité pour le point solution y_1

la variation due à la présence d'incertitudes est bien plus importante puisqu'elle peut varier de plus de 30%. De plus, sur la surface représentant la densité de probabilité on a représenté par le symbole \blacktriangle la valeur moyenne des objectifs, et la performance de cette même solution pour la moyenne des incertitudes $(M(\mathbf{x}_i, \mathbb{E}[W(\omega)]), F_c(\mathbf{x}_i, \mathbb{E}[W(\omega)]))$ par le symbole \bullet . On a également porté sur la surface, une centaine de performances calculées pour des réalisations de $W(\omega)$ et nous avons discriminé ces points en fonction du mode de ruine qui apparaît en premier (indentation du cœur ou plastification de la peau inférieure). Les solutions présentes sur le front de Pareto se trouvent toutes à la limite entre les deux modes de ruine considérés.

Cette application a permis de montrer qu'il est possible de résoudre des problèmes d'optimisation multicritères stochastiques en utilisant l'algorithme *SMGDA* et ce de manière efficace. La prise en compte des incertitudes prend tout son intérêt sur ce type d'exemple puisque la propagation de celles-ci à travers les objectifs peut influencer de manière importante la performance finale d'une solution comme cela a été montré sur les densités de probabilité reconstruites. Cependant le coût numérique induit par les incertitudes n'est pas négligeable, il faut compter un nombre d'itérations de

FIGURE 6.10 Densité de probabilité pour le point solution y_2

l'algorithme d'optimisation plus important qu'un cas déterministe. Cette demande d'itérations supplémentaires, qui dépend majoritairement de la sensibilité des objectifs aux incertitudes, pour la résolution peut être limitante dans une phase d'avant projet. Il faut de plus noter que la méthode de pénalisation exacte telle qu'elle est appliquée ici, en rajoutant un terme non-différentiable aux objectifs, rend nécessaire l'utilisation de sous-différentielles pour construire le vecteur de descente commun ce qui induit un nouveau surcoût numérique. Cependant, on reste optimiste sur le fait que des solutions seront trouvées pour repousser les limites dues au coût numérique.

Outre la difficulté numérique induite par l'ajout d'incertitude, la mise en place d'un problème sous incertitudes peut aussi s'avérer complexe. En effet, la difficulté d'un tel problème réside tout particulièrement dans la mise en place de lois de probabilité qui rendent compte de la réalité. Les informations concernant les incertitudes sur les matériaux, ou les géométries n'étant pas évidentes à trouver. Sur cet exemple, on peut notamment regretter l'absence de données industrielles concernant les incertitudes que nous avons ajoutées au problème et qui auraient permis d'obtenir un test plus réaliste.

Dans la suite, on se propose de traiter le cas où la contrainte est exprimée comme une probabilité d'un évènement. L'optimisation fiabiliste rentre dans ce cadre particulier.

6.2 Contraintes en probabilité par une approche multiobjectif

L'optimisation fiabiliste (Reliability based design optimization (RBDO)) est un problème d'optimisation sous une contrainte en probabilité. La fiabilité est exprimée par l'intermédiaire d'une probabilité de ruine d'un système mécanique, et est ajoutée au problème d'optimisation par l'intermédiaire d'une contrainte. La véritable difficulté est l'évaluation de cette probabilité, puisqu'en règle générale, il n'existe pas d'expression analytique qui donne la valeur de la probabilité et que celle-ci est souvent désirée faible. La littérature est très riche sur les méthodes de résolution de ce type de problème, on peut par exemple citer les articles suivants [2, 56, 60, 57, 69, 73]. La formulation générale d'un problème d'optimisation fiabiliste est donné par

$$\begin{cases} \min_{x \in \mathbb{R}^n} \{\mathbb{E}[f(x, W(\omega))]\} \\ \text{sous la contrainte } \mathbb{P}[g(x, W(\omega)) \geq 0] \leq \alpha, \end{cases} \quad (6.6)$$

où $g(x, W(\omega)) \geq 0$ est l'évènement de ruine dont on désire sa probabilité d'apparition inférieure à un certain seuil α .

Le but de cette application est d'illustrer les capacités de *SMGDA* à traiter les problèmes d'optimisation sous contrainte probabiliste. Pour cela, on considère un exemple simple tiré de l'article de Moustapha *et al.* [57]. Le but de cette optimisation est de minimiser la section d'une poutre soumise à une charge de compression. La probabilité de défaillance en flambage de cette poutre devant être inférieure à un seuil α ($\alpha = 5\%$ dans [57]). La Figure 6.11 illustre la poutre qui sera étudiée dans cette application où b et h sont les variables d'optimisation du problème, et on note $\mathbf{x} = [b, h] \in [0.15, 0.35]^2$. La section de la poutre peut être écrite de la manière

$$M(\mathbf{x}) = b \times h.$$

Une première contrainte géométrique déterministe est introduite sur ce problème de telle sorte que $b \geq h$. Cette contrainte sera traitée par projection dans le cadre de cette application.

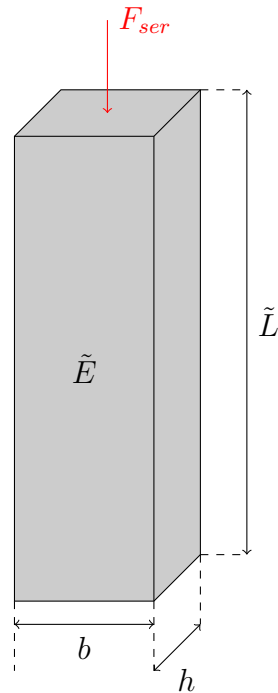


FIGURE 6.11 Poutre étudiée en compression

Des incertitudes sont introduites dans le modèle par l'intermédiaire de variables aléatoires. La première incertitude porte sur un coefficient multiplicatif $\tilde{k}(\omega)$ du seuil de flambage d'Euler. Le module de Young $\tilde{E}(\omega)$ sera aussi considéré incertain ainsi que la longueur de la poutre \tilde{L} . Les incertitudes suivent toutes les trois des lois lognormales décrites dans le tableau 6.3.

TABLE 6.3 Variables aléatoires du problème de fiabilité

Variable	distribution	moyenne (μ)	coefficient de variation (δ)
\tilde{k}	lognormale	0.6	10%
\tilde{E} (MPa)		10^4	5%
\tilde{L} (m)		3	1%

On peut alors écrire le problème d'optimisation complet sous la forme :

$$\left\{ \begin{array}{l} \min_{\mathbf{x} \in [0.15, 0.35]^2} M(\mathbf{x}, W(\omega)) \\ \text{sous les contraintes} \end{array} \right\} \begin{cases} b \geq h \\ \mathbb{P} \left(F_{ser} - \tilde{k}(\omega) \frac{\pi^2 \tilde{E}(\omega) b h^3}{\tilde{L}(\omega)} \geq 0 \right) \leq \alpha. \end{cases} \quad (6.7)$$

Le choix d'incertitude s'est porté sur des lois lognormales puisqu'elles permettent de construire une solution analytique du vecteur de variables d'optimisation pour un seuil de probabilité α donné. Cette solution analytique est donnée par la relation

$$b^*(\alpha) = h^*(\alpha) = \left(\frac{12F_{ser}}{\pi^2 \exp(\lambda_{\tilde{k}} + \lambda_{\tilde{E}} - 2\lambda_{\tilde{L}} + \Phi^{-1}(\alpha) \sqrt{\zeta_{\tilde{k}}^2 + \zeta_{\tilde{E}}^2 + \zeta_{\tilde{L}}^2})} \right)^{\frac{1}{4}}, \quad (6.8)$$

où $\Phi(\alpha)$ est la fonction de répartition de la loi normale standard, $\zeta_j = \sqrt{\ln(1 + \delta_j^2)}$ et $\lambda_j = \ln(\mu_j) - 1/2 \times \zeta_j^2$. Pour un seuil de probabilité α donné, il va de soi que les performances associées aux points $(b^*(\alpha), h^*(\alpha))$ sont situées sur le front de Pareto du problème d'optimisation multiobjectif

$$\left\{ \begin{array}{l} \min_{\mathbf{x} \in [0.15, 0.35]^2} \left\{ M(\mathbf{x}), \mathbb{P} \left(F_{ser} - \tilde{k}(\omega) \frac{\pi^2 \tilde{E}(\omega) b h^3}{\tilde{L}(\omega)} \geq 0 \right) \right\} \\ \text{Sous la contrainte } b \geq h. \end{array} \right\} \quad (6.9)$$

La plus grande difficulté de ce type de problème d'optimisation est le calcul de la probabilité qui peut avoir une valeur très petite. Dans ce cas, les méthodes classiques d'estimation de la probabilité par simulation de Monte Carlo ne sont alors plus envisageables. On propose dans ce paragraphe de modifier le problème de fiabilité initial, en un problème multiobjectif que l'on résoudra avec *SMGDA*. L'intérêt d'une telle approche est qu'elle ne nécessite pas de calculer la probabilité de défaillance à chaque itération de l'algorithme. Elle donne en particulier accès à l'ensemble des meilleurs compromis pour l'ensemble des valeurs de la probabilité de défaillance $\alpha \in [0, 1]$.

Afin de résoudre le problème (6.9) en utilisant *SMGDA*, il est nécessaire d'exprimer les objectifs comme des espérances de fonctions aléatoires. On remarque que la probabilité de défaillance est l'espérance d'une fonction indicatrice de l'évènement de

défaillance :

$$\mathbb{P} \left(F_{ser} - \tilde{k}(\omega) \frac{\pi^2 \tilde{E}(\omega) b h^3}{\tilde{L}(\omega)} \geq 0 \right) = \mathbb{E} \left[\mathbf{1}_{F_{ser} - \tilde{k}(\omega) \frac{\pi^2 \tilde{E}(\omega) b h^3}{\tilde{L}(\omega)} \geq 0} \right],$$

où $\mathbf{1}$ correspond à la fonction indicatrice. On peut alors réécrire le problème d'optimisation (6.9) par sous la forme :

$$\left\{ \begin{array}{l} \min_{\mathbf{x} \in [0.15, 0.35]^2} \left\{ M(\mathbf{x}), \mathbb{E} \left[\mathbf{1}_{F_{ser} - \tilde{k}(\omega) \frac{\pi^2 \tilde{E}(\omega) b h^3}{\tilde{L}(\omega)} \geq 0} \right] \right\} \\ \text{Sous la contrainte } b \geq h. \end{array} \right. \quad (6.10)$$

Le problème ainsi formulé correspond aux formulations adaptées à *SMGDA*. Cependant la fonction indicatrice n'appartient à aucune classe de fonction considérées ici puisque qu'elle n'est pas localement lipschitzienne. Dans sa thèse, Andrieu [3] propose d'approcher la fonction indicatrice par un produit de convolution de l'indicatrice avec une fonction régulière, dépendant d'un paramètre et qui converge vers la fonction de Dirac lorsque le paramètre est grand : c'est la technique des "mollifiers". Cette approche va être rappelée dans le paragraphe suivant.

On introduit une fonction régulière $h : \mathbb{R} \rightarrow \mathbb{R}$, dépendant d'un paramètre r , paire, à valeurs positives ou nulles, d'intégrale égale à 1 avec son maximum en 0 et approchant le Dirac lorsque le paramètre r tend vers 0. Des exemples pour la fonction h sont donnés dans le tableau 6.4.

TABLE 6.4 Exemple de fonctions pour $h(x/r)$, où $\mathbf{1}_1$ désigne l'indicatrice de l'intervalle $[-1, 1[$

$h(x/r)$
$\frac{1}{2} \mathbf{1}_1(x/r)$
$(1 - x/r) \mathbf{1}_1(x/r)$
$\frac{\pi}{4} \cos(\frac{\pi}{2} x/r) \mathbf{1}_1(x/r)$
$\frac{3}{4} (1 - (x/r)^2) \mathbf{1}_1(x/r)$

On construit alors l'approximation de la fonction indicatrice sur \mathbb{R}^+

$$\mathbb{1}_{\mathbb{R}^+}^r(\mathbf{z}) = \frac{1}{r} \int_{-\infty}^{+\infty} \mathbb{1}_{\mathbb{R}^+}(\mathbf{y}) h\left(\frac{\mathbf{z} - \mathbf{y}}{r}\right) d\mathbf{y} = \frac{1}{r} \int_0^{+\infty} h\left(\frac{\mathbf{z} - \mathbf{y}}{r}\right) d\mathbf{y},$$

ce qui peut s'interpréter comme l'espérance de la fonction indicatrice pour une variable aléatoire \mathbf{y} et de distribution $\frac{1}{r}h(\cdot/r)$ centrée autour de \mathbf{z} . Cette distribution sera de plus en plus concentrée autour de \mathbf{z} lorsque r décroît vers 0. En remplaçant \mathbf{z} par l'évènement de défaillance $g(\mathbf{x}, W(\omega)) \geq 0$ pour un tirage ω fixé, on obtient :

$$\mathbb{1}_{\mathbb{R}^+}^r(g(\mathbf{x}, W(\omega))) = \frac{1}{r} \int_0^{+\infty} h\left(\frac{g(\mathbf{x}, W(\omega)) - \mathbf{y}}{r}\right) d\mathbf{y}.$$

Cette approximation est différentiable par rapport à \mathbf{x} (différentielle notée par ∇) :

$$\begin{aligned} \nabla \mathbb{1}_{\mathbb{R}^+}^r(g(\mathbf{x}, W)) &= \frac{1}{r} \int_0^{+\infty} h'\left(\frac{g(\mathbf{x}, W(\omega)) - \mathbf{y}}{r}\right) \frac{\nabla g(\mathbf{x}, W(\omega))}{r} d\mathbf{y} \\ &= \int_0^{+\infty} \frac{1}{r^2} h'\left(\frac{g(\mathbf{x}, W(\omega)) - \mathbf{y}}{r}\right) \nabla g(\mathbf{x}, W(\omega)) d\mathbf{y}. \end{aligned}$$

Ce qui conduit à l'expression finale

$$\nabla \mathbb{1}_{\mathbb{R}^+}^r(g(\mathbf{x}, W(\omega))) = -\frac{1}{r} h\left(\frac{g(\mathbf{x}, W(\omega))}{r}\right) \nabla g(\mathbf{x}, W(\omega)).$$

Des études de cet estimateur ont été menée dans la thèse [3], et l'auteur préconise de faire décroître le paramètre r avec une loi

$$r_k = \frac{1}{k^{1/5}},$$

où k est le numéro de l'itération de l'algorithme d'optimisation. Ce qui permet d'obtenir un compromis satisfaisant entre le biais et la variance de l'estimateur ainsi construit. La suite retenue pour l'application est donnée par :

$$r_k = \frac{7 \cdot 10^5}{k^{1/5}}.$$

Afin de mieux répondre au problème initial posé pour un seuil fixé, l'algorithme de génération de points initiaux a été modifié. Au lieu d'effectuer des tirages uniformes dans les zones pouvant contenir le front de Pareto, on effectue des tirages qui suivent une loi normale centrée en 0.05 dans la direction de la probabilité de défaillance. L'ajout

de cette loi uniforme sur les tirages est illustré sur la Figure 6.12 qui illustre les points obtenus à la fin de la première salve d'optimisation . Le front de Pareto final obtenu

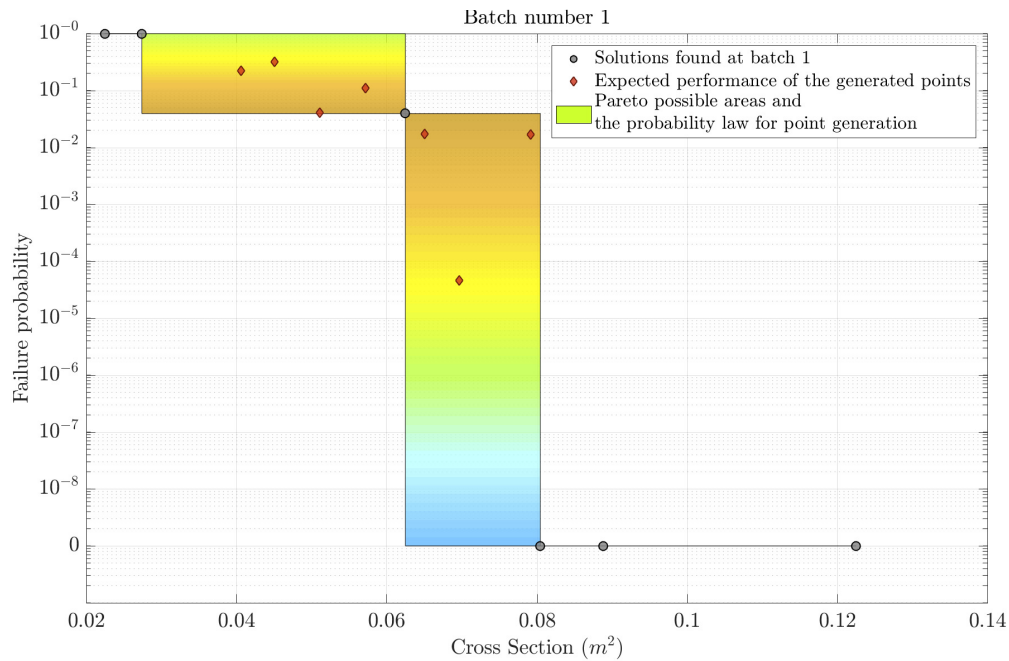


FIGURE 6.12 Loi normale autour de 5% sur les tirages aléatoires lors de la génération de points initiaux

est représenté sur la Figure 6.13. Un budget de 650 appels aux objectifs par point a été attribué et les performances finales des solutions obtenues ont été évalués en utilisant une méthode *Sample Average Approximation* sur 10^9 échantillons ce qui a permis d'estimer des probabilités de l'ordre de 10^{-7} .

Les résultats obtenus par *SMGDA* sont tous proches du front de Pareto analytique donné par l'expression (6.8) ce qui est très satisfaisant vu le nombre réduit d'itérations par point alloué à *SMGDA*. L'enrichissement des solutions autour de la valeur initiale de 5% par le biais de l'algorithme 3 permet d'obtenir des solution proche de la contrainte du problème original, mais aussi d'estimer les compromis possibles autour de ce seuil critique. L'algorithme, avec ce nombre d'itérations faible, a donné des solutions pour lesquels la probabilité de défaillance est très faible 10^{-6} sans avoir à calculer les espérances durant le processus d'optimisation. Le calcul de performance final des objectifs et de la probabilité de défaillance reste cependant à effectuer, avec un coût non négligeable nécessaire pour voir les probabilités faible (10^9 échantillons sur ce dernier test). Cette nouvelle difficulté pour l'estimation des performances finales est

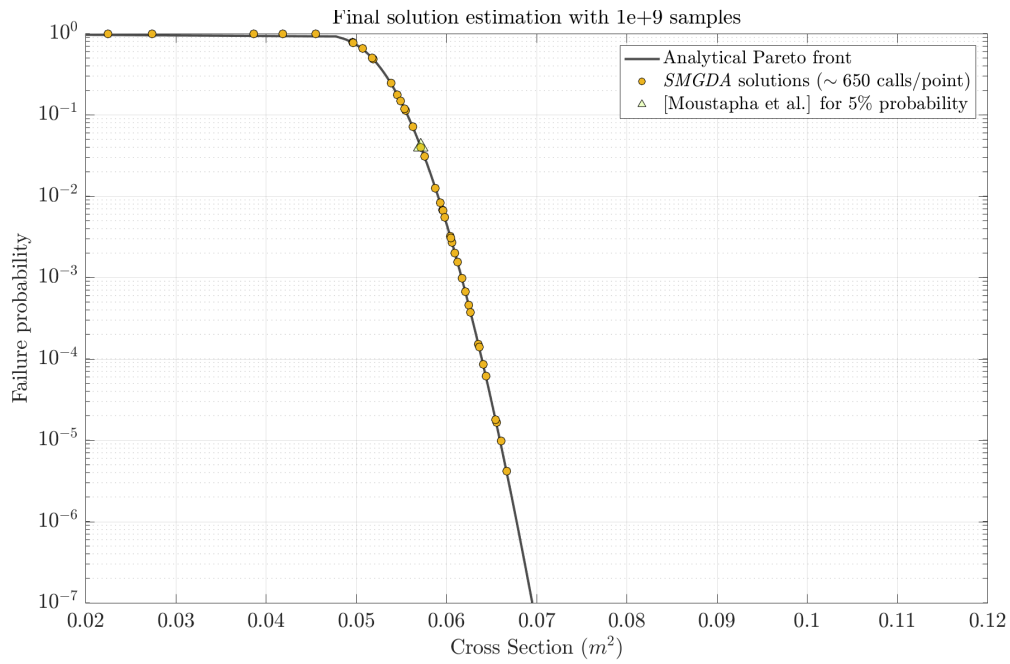


FIGURE 6.13 Résultats finaux de *SMGDA* pour le problème de fiabilité

cependant indépendante du problème d'optimisation et peut être résolue à l'aide de méthodes dédiées (*subset simulation* [4, 27]).

Chapitre 7

Conclusion

Les travaux menés dans cette thèse ont abouti au développement d'un algorithme de résolution de problèmes d'optimisation pour lesquels la fonction coût s'exprime comme l'espérance d'une fonction aléatoire à valeurs vectorielles. À l'inverse de nombreuses méthodes proposées dans la littérature, l'algorithme s'appuie directement sur le préordre de Pareto associé à l'optimisation multiobjectif et permet donc d'effectuer des optimisations sans *a priori* sur les objectifs du problème. L'algorithme est construit autour de résultats d'analyse convexe prouvant l'existence d'un vecteur de descente commun à l'ensemble des objectifs par le biais d'informations sur les dérivées directionnelles des fonctions à minimiser. À l'instar de l'algorithme du gradient stochastique, nous avons alors construit un nouvel algorithme nommé *SMGDA*, qui propose de construire une suite itérative pour laquelle chaque étape consiste à avancer dans la direction donnée par le vecteur de descente commun, calculé au point courant et pour un tirage unique et indépendant des variables aléatoires modélisant l'aléa dans la fonction coût. Ainsi construit, l'algorithme ne nécessite pas la construction coûteuse des estimateurs des espérances des objectifs à chaque itération.

En s'appuyant sur les différentes preuves de convergence de l'algorithme du gradient stochastique, nous avons pu mettre en place un ensemble d'hypothèses pour lequel des résultats de convergence de ce nouvel algorithme vers les solutions de Pareto ont pu être prouvés. Une première convergence en distance a ainsi pu être démontrée, associée à un résultat de vitesse de convergence pour une suite de pas de descente bien choisie. Une seconde preuve, s'appuyant sur les surmartingales, a permis de mettre en avant la convergence presque sûre de l'algorithme vers les solutions de Pareto. Ces résultats rendent l'utilisation de *SMGDA* tout à fait pertinente pour résoudre les problèmes d'optimisation multicritères stochastiques. On pourra cependant noter que certaines

hypothèses avancées dans ce document sont fortes (gradients et sous-différentielles bornées presque sûrement) et pourraient faire l'objet d'un travail complémentaire pour les affaiblir.

Les essais numériques nous ont permis dans un premier temps de procéder à des améliorations moins formelles de l'algorithme, et en particulier à améliorer à la fois la dispersion des solutions obtenues sur le front de Pareto, mais aussi le nombre d'itérations nécessaires à la convergence. Dans un premier temps, nous avons proposé de remplacer le vecteur de descente commun par un vecteur calculé sur le même schéma mais en utilisant des estimateurs des espérances des gradients et des sous-différentielle estimés en profitant de l'information acquise aux itérations précédentes. Cette méthode nous a permis d'atténuer le phénomène de glissement des solutions sur le front de Pareto, et garantit donc une certaine stabilité et dispersion des solutions sur le front. Une seconde amélioration a consisté, sachant que les points de départ sont déterminants pour les solutions finales, à construire une procédure de génération de points initiaux de manière à atteindre les zones où aucune solution n'a été obtenue préalablement (amélioration de la dispersion) et plus proches des solutions de Pareto (moins d'itérations). Cette méthode s'appuie en partie sur la création d'un métamodèle de krigeage, et l'utilisation de la relation de dominance de Pareto pour identifier les zones où des solutions du front peuvent exister.

L'algorithme *SMGDA* a ensuite été testé sur un benchmark comportant dix tests variés et extraits de la littérature, et ses performances ont été comparées à celles obtenues par deux autres algorithmes classiques du domaine de l'optimisation multiobjectif (*NSGA-II*, *DMS*). Les performances de chaque algorithme ont pu être comparées par l'intermédiaire de mesures (pureté, hypervolume) et analysées par le biais de profils de performances. Sur l'ensemble des tests, *SMGDA*, avec dix fois moins d'appels aux objectifs alloués que ses concurrents, s'est montré très performant puisque sur les deux mesures on a obtenu de meilleurs résultats sur le plus grand nombre de tests. Nous avons pu en particulier voir que les essais pour lesquels *SMGDA* s'est montré moins performant sont en partie des tests pour lesquels les hypothèses de convergence ne sont pas validées (objectifs multimodaux). Cependant, même lorsqu'il n'a pas été le plus performant, *SMGDA* a donné des résultats ayant des notes faiblement éloignés de ses deux concurrents. L'algorithme *SMGDA* s'est donc montré très performant lors des premiers essais numériques et cela face à des algorithmes réputés.

Nous avons ensuite procédé à une série de mise en situation sur des problèmes d'optimisation de structures plus réalistes. Ces nouveaux essais avaient pour objectif d'étendre les illustrations numériques aux problèmes contraints.

Un premier essai a été mené sur l'optimisation d'une poutre en considérant deux objectifs sous une contrainte géométrique déterministe. L'utilisation d'une fonction de pénalisation a permis la gestion de cette contrainte et la résolution de ce problème. Une seconde possibilité qui n'a pas été traitée dans cette thèse aurait été la gestion de ce type de contrainte déterministe par l'utilisation de méthodes de type Lagrangien. L'intégration des méthodes Lagrangiennes dans *MGDA* a déjà en partie été analysée et effectuée avec succès, et leur extension au cadre stochastique devrait être possible. Il faudra néanmoins être attentif à l'ajout des incertitudes qui engendre une difficulté supplémentaire.

Dans un second temps, nous avons voulu étudier comment des contraintes en probabilité peuvent être prises en compte et résolues par l'intermédiaire de *SMGDA*. Nous avons donc considéré une optimisation fiabiliste pour laquelle on cherche à minimiser la masse d'une poutre en conservant une probabilité de ruine inférieure à 5%. La solution d'un tel problème de fiabilité est aussi une solution de Pareto du problème multiobjectif cherchant à minimiser à la fois la masse de la poutre et la probabilité de ruine. Par conséquent, nous avons transformé l'écriture de ce problème contraint en un problème multiobjectif sans contrainte. La probabilité de ruine est alors réécrite comme l'espérance d'une fonction indicatrice valant 1 lorsque l'évènement de ruine est actif. Cette réécriture nous a permis d'obtenir par l'intermédiaire de *SMGDA* le front de Pareto correspondant à ce problème de fiabilité sans avoir à estimer les valeurs des probabilités à chaque itération.

Cependant, quelque soit l'essai numérique réalisé, l'estimation des performances finales des solutions obtenues nécessite le calcul d'espérances de fonctions aléatoires. Dans cette thèse, seule une méthode de type Monte Carlo naïve a été utilisée en fin de processus d'optimisation afin d'estimer les performances. L'ajout de méthodes spécifiques à la propagation d'incertitudes ou à l'estimation de probabilités faibles pourrait permettre de réduire le coût de cette dernière estimation.

Finalement, on a démontré l'efficacité de l'algorithme *SMGDA* grâce aux preuves de convergence vers les solutions du problème et aux illustrations numériques. L'absence du calcul des espérances permet aussi de lui conférer une très bonne rapidité et un coût numérique moindre en regard des approches classiques existantes à ce jour. Ce dernier point pourrait éventuellement permettre l'utilisation de cette méthode d'optimisation

lors de phases d'optimisation d'avant projet où les objectifs à calculer sont bien plus coûteux en terme de temps, et bien plus réalistes. Une éventuelle hybridation de la méthode *SMGDA* avec des méthodes d'approximations (métamodèles, modèles réduits) pourraient notamment permettre de résoudre des problèmes d'optimisation soumis à la discrétisation d'une équation aux dérivées partielles comme les Éléments Finis ou les Volumes Finis et où le nombre de paramètres d'optimisation ou d'objectifs est plus conséquent. Ce type d'hybridation a déjà été menée dans la thèse de Zerbinati [75] qui construit des approximations par métamodèle de krigeage d'objectifs d'un problème d'optimisation aérodynamique haute fidélité afin de réduire le coût des objectifs et d'avoir accès aux gradients approchés. L'algorithme *MGDA* est alors utilisé sur ces approximations pour trouver les solutions du problème d'optimisation multiobjectif. Il existe aussi de nombreuses méthodes d'accélération pour l'algorithme du gradient stochastique dans la littérature et particulièrement dans le domaine du machine learning [45, 40, 70], où la vitesse d'apprentissage d'un algorithme est souvent une caractéristique critique. Ces méthodes pouvant potentiellement être transférées au cadre multiobjectif et à l'algorithme *SMGDA*.

Bibliographie

- [1] ACHARD, T. (2017). *Techniques de calcul de gradient aéro-structure haute-fidélité pour l'optimisation de voilure flexibles*. Thèse de doctorat, CNAM.
- [2] ALLEN, M. et MAUTE, K. (2004). Reliability-based design optimization of aeroelastic structures. *Structural and Multidisciplinary Optimization*, 27:228–242.
- [3] ANDRIEU, L. (2004). *Optimisation sous contrainte en probabilité*. Thèse de doctorat, École des Ponts et Chaussées.
- [4] AU, S.-K. et BECK, J. L. (2001). Estimation of small probabilities in high dimension by subset simulation. *Probabilistic Engineering Mechanics*, 16:263–277.
- [5] AUGER, A., BADER, J., BROCKHOFF, D. et ZITZLER, E. (2009). Theory of hypervolume indicator : Optimal μ -distributions and the choice of the reference point. *Proceedings of the Tenth ACM SIGEVO Workshop Foundations of Genetic Algorithms*, pages 87–102.
- [6] BELLMAN, R. et ZADEH, L. (1970). Decision-making in fuzzy environment. *Management Science*, 17:141–161.
- [7] BONNANS, J.-F., GILBERT, J., LEMARECHAL, C. et SAGASTIZÁBAL, C. (2006). *Numerical Optimization Numerical Optimization, Theoretical and Practical Aspects*. Springer-Verlag Berlin Heidelberg.
- [8] BONNEL, H. et COLLONGE, J. (2014). Stochastic optimization over a pareto set associated with stochastic multiobjective optimization problem. *Journal of Optimization Theory and Applications*, 162(2):405–427.
- [9] BOULEAU, N., LAMBERTON, D. et LAPEYRE, B. (1986). Une méthode numérique adaptée au calcul probabiliste des structures. *Journal de Mécanique Théorique et Appliquée*, 5(5).
- [10] BREITKOPF, P. et COELHO, R. F. (2010). *Multidisciplinary Design Optimization in Computational Mechanics*. Wiley-ISTE.
- [11] BURKE, J. V., LEWIS, A. S. et OVERTON, M. L. (2002). Approximating subdifferentials by random sampling of gradients. *Mathematics of Operations Research*, 27(3):567–584.

- [12] CABALLERO, R., CERDÁ, E., del MAR MUÑOZ et REY, L. (2004). Stochastic approach versus multiobjective approach for obtaining efficient solutions in stochastic multiobjective programming problems. *European Journal of Operational Research*, 158:633–648.
- [13] CLARKE, F. (1990). *Optimization and Nonsmooth Analysis*. Society for Industrial and Applied Mathematics.
- [14] COELHO, R. F. et BREITKOPF, P. (2009). *Optimisation multidisciplinaire en mécanique 1 : démarche de conception, stratégie collaboratives et concourantes, multiniveaux de modèles et de paramètres*. Hermes Science Publications-Lavoisier.
- [15] COLLETTE, Y. et SIARRY, P. (2011). *Optimisation multiobjectif*. Editions Eyrolles.
- [16] CUSTÓDIO, A. L., MADEIRA, F. A., VAZ, A. I. F. et VINCENTE, L. N. (2011). Direct multisearch for multiobjective optimization. *SIAM Journal on Optimization*, 21(3):1109–1140.
- [17] DANTZIG, B. (1955). Linear programming under uncertainty. *Management Science*, 1:197–206.
- [18] DEB, K. et GUPTA, H. (2005). Searching for robust pareto-optimal solutions in multi-objective optimization. In COELLO COELLO, C., HERNÁNDEZ AGUIRRE, A. et ZITZLER, E., éditeurs : *Evolutionary Multi-Criterion Optimization*, volume 3410 de *Lecture Notes in Computer Science*, pages 150–164. Springer Berlin Heidelberg.
- [19] DEB, K., PRATAP, A., AGARWAL, S. et MEYARIVAN, T. (2002). A fast and elitist multiobjective genetic algorithm : Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.
- [20] DÉSIDÉRI, J.-A. (2009). Multiple-Gradient Descent Algorithm (MGDA). Research Report RR-6953, INRIA.
- [21] DÉSIDÉRI, J.-A. (2018). Quasi-riemannian multiple gradient descent algorithm for constrained multiobjective differential optimization. Rapport technique 9159, Inira.
- [22] DÉSIDÉRI, J.-A. et DUVIGNEAU, R. (2017). Parametric optimisation of pulsating jets in unsteady flow by multiple gradient descent algorithm (mgda).
- [23] DÉSIDÉRI, J.-A., DUVIGNEAU, R. et HABBAL, A. (2014). Multiobjective design optimization using nash games. In VASILE, M. et BECERRA, V. M., éditeurs : *Computational intelligence in Aerospace Science, Progress in Astronautics and Aeronautics*, chapitre 16. American Institute of Aeronautics and Astronautics.
- [24] DIAZ, J. E., HANDL, J. et XU, D.-L. (2017). Evolutionary robust optimization in production planning interactions between number of objectives, sample size and choice of robustness measure. *Computers & Operations Research*, 79:266–278.
- [25] DODU, J., GOURSAT, M., HERTZ, A., QUADRAT, J.-P. et VIOT, M. (1981). *Stochastic gradient methods for optimizing investments in electric power system studies*. Bulletin de la Direction des Études et Recherches EDF.

- [26] DOLAN, E. D. et MORÉ, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91(2):201–213.
- [27] DUBOURG, V., SUDRET, B. et BOURINET, J.-M. (2011). Reliability-based design optimization using kriging surrogates and subset simulation. *Structural and Multidisciplinary Optimization*, 44(5):673–690.
- [28] DURRETT, R. (2010). *Probability Theory and Examples*. Cambridge Series in Statistical and Probabilistic Mathematics.
- [29] DVORETZKY, A. (1956). On stochastic approximation. *Proceeding of the Third Berkeley Symposium on Mathematical Statistics and Probability*, 1:39–55.
- [30] EHRGOTT, M. (2006). *Multicriteria optimization*. Springer Science & Business Media.
- [31] ERMOLIEV, Y. (1983). Stochastic quasigradient methods and their application to systems optimization. *Stochastics*, 9:1–36.
- [32] ERMOLIEV, Y. et WETS, R. (1988). *Numerical Technics for Stochastic Optimization*. Springer Verlag.
- [33] FEDERER, H. (1996). *Geometric Measure Theory*. Springer-Verlag Berlin Heidelberg.
- [34] FLIEGE, J. et WERNER, R. (2014). Robust multiobjective optimization & applications in portfolio optimization. *European Journal of Operational Research*, 234:422–433.
- [35] FLIEGE, J. et XU, H. (2011). Stochastic multiobjective optimization : Sample average approximations and applications. *Journal of Optimization Theory and Applications*, 151(1):135–162.
- [36] GABREL, V., MURAT, C. et THIELE, A. (2014). Recent advances in robust optimization : An overview. *European Journal of Operational Research*, 235:471–483.
- [37] GIBSON, L. J. et ASHBY, M. F. (1999). *Cellular Solids : Structure and Properties*. Cambridge university press.
- [38] HUBAND, S., HINGSTON, P., BARONE, L. et WHILE, L. (2006). A review of multiobjective test problems and a scalable problem toolkit. *IEEE Transactions on Evolutionary Computation*, 10:477–506.
- [39] JIN, Y., OLHOFFER, M. et SENDHOFF, B. (2001). Dynamic weighted aggregation for evolutionary multiobjective optimization : Why does it work and how ? *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1042–1049.
- [40] JOHNSON, R. et ZHANG, T. (2013). Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS'13 Proceedings of the 26th International Conference on Neural Information Processing Systems*, volume 1, pages 315–323.
- [41] KANG, J.-S., SUH, M.-H. et LEE, T.-Y. (2004). Robust economic optimization of process design under uncertainty. *Engineering Optimization*, 36:51–75.

- [42] KLAMROTH, K., KBIS, E., SCHBEL, A. et TAMMER, C. (2017). A unified approach to uncertain optimization. *European Journal of Operational Research*, 260:805–819.
- [43] KNOWLES, J. et CORNE, D. (1999). The pareto evolution strategy : A new baseline algorithm for pareto multiobjective optimization. In PISCATAWAY, N., éditeur : *Congres on Evolutionary Computation (CEC99)*, volume 1, pages 98–105.
- [44] KOVACH, J. et CHO, B. R. (2008). Development of a multidisciplinary multiresponse robust design optimization model. *Engineering Optimization*, 40:805–819.
- [45] LE ROUX, N., SCHMIDT, M. et BACH, F. (2012). A stochastic gradient method with exponential convergence rate for strongly-convex optimization with finite training sets. *Advances in Neural Information Processing Systems*.
- [46] LEITE, P. (2013). *Conception architecturale appliquée aux matériaux sandwichs pour propriétés multifonctionnelles*. Thèse de doctorat, I-MEP2.
- [47] LÖHNDORF, N. (2016). An empirical analysis of scenario generation methods for stochastic optimization. *European Journal of Operational Research*, 255:121–132.
- [48] MÄKELÄ, M. M., ERONEN, V.-P. et KARMITSA, N. (2014a). *Optimization in Science and Engineering*, chapitre On Nonsmooth Optimality Conditions with Generalized Convexities, pages 333–357. Springer Verlag.
- [49] MÄKELÄ, M. M., KARMITSA, N. et ERONEN, V.-P. (2010). On generalized pseudo and quasi-convexities for nonsmooth functions. Rapport technique No 989, Turku Center for Computer Science.
- [50] MÄKELÄ, M. M., KARMITSA, N. et WILPPU, O. (2014b). Multiobjective proximal bundle method for nonsmooth optimization. Rapport technique No 1120, Turku Center for Computer Science.
- [51] MATTSON, C. A. et MESSAC, A. (2005). Pareto frontier based concept selection under uncertainty, with visualization. *Optimization and Engineering*, 6:85–115.
- [52] MERCIER, Q., POIRION, F. et DÉSIDÉRI, J.-A. (2017). Smgda : an uncertainty based multiobjective optimization approach. illustration to an airplane composite material. In *Procedia Engineering*, volume 199, pages 1199–1203.
- [53] MERCIER, Q., POIRION, F. et DÉSIDÉRI, J.-A. (2018a). Nonconvex multiobjective design optimization under uncertainty : a descent algorithm. application to sandwich plate design and reliability. *Engineering Optimization*.
- [54] MERCIER, Q., POIRION, F. et DÉSIDÉRI, J.-A. (2018b). A stochastic multiple gradient descent algorithm. *European Journal of Operational Research*, 271(3):808–817.
- [55] MIETTINEN, K. (1999). *Nonlinear Multiobjective Optimization*. Springer Science & Business Media.
- [56] MISSOUM, S., DRIBUSCH, C. et BERAN, P. (2010). Reliability-based design optimization of nonlinear aeroelasticity problems. *Journal of Aircraft*, 47(3):992–998.

- [57] MOUSTAPHA, M., SUDRET, B. et BOURINET, J.-M. (2016). Quantile-based optimization under uncertainties using adaptative kriging surrogate models. *Structural and Multidisciplinary Optimization*, 54(6):1403–1421.
- [58] NEMIROVSKI, A. et SHAPIRO, A. (2006). Scenario approximation of chance constraints. In *Probabilistic and Randomized Methods for Design under Uncertainty*. Springer.
- [59] NEVEU, J. (1972). *Martingales à temps discret*. Masson & CIE.
- [60] NIKBAY, M., FAKKUSOGLU, N. et KURU, M. (2010). Reliability-based aeroelastic optimization of a composite aircraft wing via fluid-structure interaction of high fidelity solvers. In *IOP Conf Series : Materials Science and Engineering*, volume 10.
- [61] POIRION, F. et MERCIER, Q. (2018). Uncertainty-based multidisciplinary design optimization. a descent algorithm. In *Proceedings of the joint ICVRAM ISUMA UNCERTAINTIES conference*.
- [62] POIRION, F., MERCIER, Q. et DÉSIDÉRI, J.-A. (2017). Descent algorithm for nonsmooth stochastic multiobjective optimization. *Computational Optimization and Applications*, 68(2):317–331.
- [63] POIRION, F., MERCIER, Q. et DÉSIDÉRI, J.-A. (2018). Descent methods for design optimization under uncertainty. *Aerospacelab Journal*.
- [64] RAY, T., TAI, K. et CHYESEOW, K. (2007). Multiobjective design optimization by an evolutionary algorithm. *Engineering Optimization*, 33:399–424.
- [65] RIQUELME, N., VON LÜCKEN, C. et BARAN, B. (2015). Performance metrics in multiobjective optimization. *Latin American Computing Conference (CLEI)*.
- [66] ROBBINS, H. et MONRO, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407.
- [67] ROY, R., HINDUJA, S. et TETI, R. (2008). Recent advances in engineering design optimization : challenge and future trends. *Manufacturing Technology*, 57:697–715.
- [68] SAHINIDIS, N. V. (2004). Optimization under uncertainty : state-of-the-art and opportunities. *Computers and Chemical Engineering*, 28(6–7):971 – 983. {FOCAPO} 2003 Special issue.
- [69] SCHÖBI, R. et SUDRET, B. (2017). Structural reliability analysis for p-boxes using multi-level meta-models. *Probabilistic Engineering Mechanics*, 48:27–38.
- [70] SHALEV-SCHWARTZ, S. et ZHANG, T. (2013). Stochastic dual coordinate ascent methods for regularized loss. *The Journal of Machine Learning Research*, 14(1):567–599.
- [71] SHAPIRO, A. (2003). Monte carlo sampling methods. In *Stochastic Programming*, volume 10 de *Handbooks in Operations Research and Management Science*, pages 353 – 425. Elsevier.

- [72] SUN, G., LI, G., ZHOU, S., LI, H., HOU, S. et LI, Q. (2011). Crashworthiness design of vehicle by using multiobjective robust optimization. *Structural and Multidisciplinary Optimization*, 44:99–110.
- [73] SUN, G., ZHANG, H., FANG, J., LI, G. et LIAMIS, Q. (2017). Multi-objective and multi-case reliability-based design optimization for tailor rolled blank (trb) structures. *Structural and Multidisciplinary Optimization*, 55:1899–1916.
- [74] WANG, Z., GUOA, J., ZHENG, M. et WANG, Y. (2015). Uncertain multiobjective traveling salesman problem. *European Journal of Operational Research*, 241:478–489.
- [75] ZERBINATI, A. (2013). *Algorithme à gradients multiples pour l'optimisation multiobjectif en simulation de haute fidélité : application à l'aérodynamique compressible*. Theses, Université Nice Sophia Antipolis.
- [76] ZITZLER, E., DEB, K. et THIELE, L. (2000). Comparison of multiobjective evolutionary algorithms : Empirical results. *Evolutionary Computation*, 8:173–195.
- [77] ZITZLER, E. et THIELE, L. (1999). Multiobjective evolutionary algorithms : a comparative case study and the strength pareto approach. *Evolutionary Computation, IEEE Transactions on*, 3(4):257–271.

Annexe A

Tableau récapitulatif des essais numériques du Chapitre 5

Les valeurs des paramètres des algorithmes d'optimisation utilisées pour chaque cas de test sont contenues dans le tableau A.1. Ces paramètres correspondent aux paramètres déterminés lors du processus d'optimisation en vue de maximiser le résultat d'hypervolume de chaque algorithme pour chacun des tests. Pour l'algorithme *DMS*, le chiffre indiqué dans la colonne correspond au nombre d'échantillons utilisés dans la boucle d'estimation des espérances. Pour *NSGA-II*, le premier chiffre du vecteur de paramètres correspond lui aussi au nombre d'échantillons de la boucle d'échantillonnage. Le second nombre est le ratio du nombre d'individus par le nombre de générations. Le troisième nombre correspond à la valeur de la probabilité de mutation utilisée. Le dernier chiffre correspond quant à lui à un paramètre qui modifie la distribution de probabilité qui gère les croisements entre individus.

Les trois tableaux A.2, A.3 et A.4 correspondent aux descriptions des différents tests qui ont été effectués dans le Chapitre 5.

TABLE A.1 Paramètres des algorithmes *NSGA-II* et *DMS*

Problème	<i>NSGA-II</i>	<i>DMS</i>
MOP2	(2, .89, 92, 1.05)	45
MOP3	(5, 2.8897, 79, 3.3857)	178
MOP6	(46, 2.8573, 84, 1.1424)	149
ZDT1	(5, 0.9558, 96, 1.1556)	10
ZDT2	(5, 0.48758, 75, 1.0165)	19
ZDT3	(5, 0.51436, 79, 4.7875)	47
JOS1	(5, 0.35807, 78, 1.4316)	7
JOS2	(39, 2.7511, 80, 4.839)	48
SCH1	(6, 2.9513, 95, 4.2607)	192
IM1	(5, 2.3811, 92, 1.1904)	182

TABLE A.2 Description des objectifs des problèmes MOP

Nom	\mathbf{x}	$W(\omega)$	Objectifs
MOP2	$[-4, 4]^{15}$	$[\mathcal{U}(-.7, .7)]^{30}$	$\begin{cases} \mathbb{E}[f_1(\mathbf{x}, W)] = \mathbb{E}\left[1 - \exp\left(\sum_{i=1}^{15} \left(\mathbf{x}_i - \frac{1 + W_i(\omega)}{\sqrt{15}}\right)^2\right)\right] \\ \mathbb{E}[f_2(\mathbf{x}, W)] = \mathbb{E}\left[1 - \exp\left(\sum_{i=1}^{15} \left(\mathbf{x}_i + \frac{1 + W_{15+i}(\omega)}{\sqrt{15}}\right)^2\right)\right] \end{cases}$
MOP3	$[-\pi, \pi]^2$	$\begin{cases} [\mathcal{U}(-\frac{\pi}{6}, \frac{\pi}{6})]^{16} \\ [\mathcal{U}(-.5, .5)]^2 \end{cases}$	$\begin{cases} A1 = .5 \sin(1 + W_1) - 2 \cos(1 + W_2) + \sin(2 + W_3) - 1.5 \cos(2 + W_4) \\ A2 = 1.5 \sin(1 + W_5) - \cos(1 + W_6) + 2 \sin(2 + W_7) - .5 \cos(2 + W_8) \\ B1 = .5 \sin(\mathbf{x}_1 + W_9) - 2 \cos(\mathbf{x}_1 + W_{10}) + \sin(\mathbf{x}_2 + W_{11}) - 1.5 \cos(\mathbf{x}_2 + W_{12}) \\ B2 = 1.5 \sin(\mathbf{x}_1 + W_{13}) - \cos(\mathbf{x}_1 + W_{14}) + 2 \sin(\mathbf{x}_2 + W_{15}) - .5 \cos(\mathbf{x}_2 + W_{16}) \\ \mathbb{E}[f_1(\mathbf{x}, W)] = \mathbb{E}\left[1 - (A1 - B1)^2 - (A2 - B2)^2\right] \\ \mathbb{E}[f_2(\mathbf{x}, W)] = \mathbb{E}\left[-(\mathbf{x}_1 + 3 + W_{17})^2 - (\mathbf{x}_2 + 1 + W_{18})^2\right] \end{cases}$
MOP6	$[-1, 1]^2$	$\begin{cases} [\mathcal{U}(-.3, 3)]^2 \\ [\mathcal{U}(-\frac{\pi}{10}, \frac{\pi}{10})] \end{cases}$	$\begin{cases} b = 1 + 10\mathbf{x}_2(1 + W_2) \\ q = (1 + W_1)\mathbf{x}_1/b \\ s = \sin(8\pi\mathbf{x}_1(1 + W_1) + W_3) \\ \mathbb{E}[f_1(\mathbf{x}, W)] = \mathbb{E}[(1 + W_1)\mathbf{x}_1] \\ \mathbb{E}[f_2(\mathbf{x}, W)] = \mathbb{E}[b(1 - q^2 - q \cdot s)] \end{cases}$

TABLE A.3 Description des objectifs des problèmes ZDT

Nom	\mathbf{x}	$W(\omega)$	Objectifs
ZDT1	$[0, 1]^{30}$	$\left\{ \begin{array}{l} [\mathcal{U}(-.3, 3)] \\ [\mathcal{U}(-.05, 0.5)]^{29} \\ [\mathcal{U}(-.1, 1)] \\ [\mathcal{U}(-.2, 2)] \end{array} \right.$	$\left\{ \begin{array}{l} \mathbb{E}[f_1(\mathbf{x}, W)] = \mathbb{E}[(1 + W_1)\mathbf{x}_1] \\ g = 1 + 9(1 + W_{32}) \sum_{i=2}^{30} \mathbf{x}_i + W_i^2/29 \\ h = 1 - \sqrt{((1 + W_{31}) \cdot f_1/g)} \\ \mathbb{E}[f_2(\mathbf{x}, W)] = \mathbb{E}[g \cdot h] \end{array} \right.$
ZDT2			$\left\{ \begin{array}{l} \mathbb{E}[f_1(\mathbf{x}, W)] = \mathbb{E}[(1 + W_1)\mathbf{x}_1] \\ g = 1 + 9(1 + W_{32}) \sum_{i=2}^{30} \mathbf{x}_i + W_i^2/29 \\ h = 1 - (\cdot(1 + W_{31}) \cdot f_1/g)^2 \\ \mathbb{E}[f_2(\mathbf{x}, W)] = \mathbb{E}[g \cdot h] \end{array} \right.$
ZDT3			$\left\{ \begin{array}{l} \mathbb{E}[f_1(\mathbf{x}, W)] = \mathbb{E}[(1 + W_1)\mathbf{x}_1] \\ g = 1 + 9(1 + W_{32}) \sum_{i=2}^{30} \mathbf{x}_i + W_i^2/29 \\ h = 1 - \sqrt{((1 + W_{31}) \cdot f_1/g)} - ((1 + W_{31}) \cdot f_1/g) \cdot \sin(10\pi(1 + W_{31}) \cdot f_1) \\ \mathbb{E}[f_2(\mathbf{x}, W)] = \mathbb{E}[g \cdot h] \end{array} \right.$

TABLE A.4 Description des objectifs des problèmes restants

Nom	\mathbf{x}	$W(\omega)$	Objectifs
JOS1	$[-10, 10]^{30}$	$[\mathcal{U}(-.3, .3)]^{60}$	$\begin{cases} \mathbb{E}[f_1(\mathbf{x}, W(\omega))] = \mathbb{E}\left[\frac{1}{30} \cdot \sum_{i=1}^{30} (\mathbf{x}_i + W_i)^2\right] \\ \mathbb{E}[f_2(\mathbf{x}, W(\omega))] = \mathbb{E}\left[\frac{1}{30} \cdot \sum_{i=1}^{30} (\mathbf{x}_i - 2 + W_{30+i})^2\right] \end{cases}$
JOS2	$[-10, 10]^{30}$	$\begin{cases} \mathcal{U}(-.6, .6) \\ [\mathcal{U}(-.05, .05)]^{29} \\ \mathcal{U}(-.1, .1) \\ \mathcal{U}(-.2, .2) \end{cases}$	$\begin{cases} \mathbb{E}[f_1(\mathbf{x}, W(\omega))] = \mathbb{E}[(1 + W_1) \cdot \mathbf{x}_1] \\ g = 1 + 9(1 + W_{32}) \sum_{i=2}^{30} \frac{1}{29} \cdot (\mathbf{x}_i + W_i^2) \\ h = (1 + W_{31}) \cdot f_1/g \\ \mathbb{E}[f_2(\mathbf{x}, W(\omega))] = \mathbb{E}[g \cdot (1 - (h)^{.25} - (h)^4)] \end{cases}$
SCH1	$[-100, 100]$	$\begin{cases} [\mathcal{U}(-.2, .2)]^3 \\ \mathcal{U}(-.1, .1) \end{cases}$	$\begin{cases} \mathbb{E}[f_1(\mathbf{x}, W(\omega))] = \begin{cases} \mathbb{E}[-\mathbf{x}], \mathbf{x} \leq 1 + W_1 \\ \mathbb{E}[-2 + \mathbf{x}], \mathbf{x} \in]1 + W_1, 3 + W_2] \\ \mathbb{E}[4 - \mathbf{x}], \mathbf{x} \in]3 + W_2, 4 + W_3] \\ \mathbb{E}[-4 + \mathbf{x}], \mathbf{x} \geq 4 + W_3 \end{cases} \\ \mathbb{E}[f_2(\mathbf{x}, W(\omega))] = \mathbb{E}[(\mathbf{x} - 5 + W_4)^2] \end{cases}$
IM1	$\begin{cases} [1, 4] \\ [1, 2] \end{cases}$	$\begin{cases} [\mathcal{U}(-.5, .5)]^2, \mathcal{U}(-3, 3) \end{cases}$	$\begin{cases} \mathbb{E}[f_1(\mathbf{x}, W(\omega))] = \mathbb{E}\left[2 \cdot \sqrt{\mathbf{x}_1(1 + W_1)}\right] \\ \mathbb{E}[f_2(\mathbf{x}, W(\omega))] = \mathbb{E}[(1 + W_1)\mathbf{x}_1(1 - \mathbf{x}_2(1 + W_2)) + 5 + W_2] \end{cases}$

