



HAL
open science

Étude d'un problème d'optimisation en aéroélasticité avec incertitudes

Rémi Arnaud

► **To cite this version:**

Rémi Arnaud. Étude d'un problème d'optimisation en aéroélasticité avec incertitudes. Mécanique des structures [physics.class-ph]. Ecole Centrale Paris, 2014. Français. NNT : . tel-00985049

HAL Id: tel-00985049

<https://theses.hal.science/tel-00985049>

Submitted on 29 Apr 2014

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.



École Doctorale des Sciences pour l'Ingénieur
École Centrale de Paris

THÈSE DE DOCTORAT

Spécialité : Mécanique, dynamique des structures

présentée par

Rémi ARNAUD

Étude d'un problème d'optimisation en aéroélasticité avec incertitudes

dirigée par Fabrice POIRION et Didier CLOUTEAU

Soutenue le 10 avril 2014 devant le jury composé de :

M. Christian BÈS	Professeur de l'Université Paul Sabatier	rapporteur
M. Jean-Camille CHASSAING	MdC-HdR de l'Université Pierre et Marie Curie	rapporteur
M ^{lle} Lætitia ANDRIEU	Docteur EDF R & D	examineur
M. Fabien MANGEANT	Ingénieur de Recherche EADS IW	examineur
M. Fabrice POIRION	Docteur Habilité à diriger les Recherches, ONERA	directeur
M. Didier CLOUTEAU	Professeur de l'École Centrale Paris	co-directeur

ONERA
29, ave du Général Leclerc
92329 Chatillon

ECP
École Doctorale des Sciences pour l'In-
génieur
Grande Voie des Vignes
F-92 295 CHÂTENAY-MALABRY Ce-
dex

Remerciements

Pour commencer, je tiens à remercier chaleureusement mon principal encadrant, M. Fabrice Poirion, pour avoir accepté de me confier ce travail de thèse et m'avoir accompagné tout au long de ces trois années. Il m'a également forté soutenu pour trouver des articles et des sources à étudier, des pistes à explorer, ainsi que pour la rédaction des articles que nous avons publiés et mon mémoire de thèse. Sa présence quotidienne m'a permis de ne pas me disperser et de me concentrer sur les points les plus pertinents.

Je remercie également M. Didier Clouteau, mon second encadrant, pour ses remarques pertinentes, en particulier celle qui m'a permis de justifier de manière claire et précise la principale considération à l'origine de mes travaux. Mes remerciements vont aussi à M. Jean-Pierre Grisval, qui m'a permis de faire ma thèse au sein du département DADS, ainsi qu'à M. Nicolas Piet, qui a dirigé l'unité au sein de laquelle j'ai effectué ma thèse.

Je souhaite remercier tous les membres du jury qui ont assisté à ma thèse, les rapporteurs MM. Christian Bès et Jean-Camille Chassaing, et les examinateurs M. Fabien Mangeant et M^{lle} Lætitia Andrieu. J'ai tout particulièrement apprécié les travaux de M^{lle} Andrieu que j'ai eu l'occasion d'étudier au cours de mes recherches et qui m'ont permis de faire des essais numériques sur un tout nouveau genre de problème. Cela a constitué une avancée notable au cours de mes travaux.

Pour leur soutien au niveau informatique, je remercie MM. Cédric Liauzun et Jean-Sébastien Schotte, qui m'ont aidé à me débloquer à plusieurs reprises et expliqué le fonctionnement de quelques logiciels. Je remercie également M. Morchelevitch, qui a mis au point la chaîne de calculs CAPRI dont je me suis servi tout au long de ma thèse, et qui m'a aidé à optimiser son fonctionnement dans mes calculs numériques, permettant ainsi de réaliser des gains substantiels en temps de calculs. Et aussi, je remercie M. Pascal Naudin pour m'avoir expliqué quelques points de détails très intéressants à propos de la maquette que représente le modèle numérique sur lequel j'ai réalisé mes tests numériques.

Plus globalement, je salue chaleureusement l'intégralité des membres de l'unité MSDS, avec qui j'ai passé la majeure partie de mon temps et qui m'ont permis de travailler dans une ambiance détendue.

En dernier, je tiens tout particulièrement à remercier les membres de ma famille, à qui je suis profondément attaché et qui m'ont continuellement soutenu tout au long de mes études.

Résumé

Résumé

La recherche en optimisation est un secteur crucial pour les constructeurs aéronautiques. La performance des appareils est un élément déterminant dans la compétition commerciale qui oppose les principaux manufacturiers du marché. L'incorporation de plus en plus massive des matériaux composites dans les avions de ligne dans les années 2000 illustre le désir des constructeurs de réduire la masse de leurs appareils pour en diminuer la consommation de kéroène. Parallèlement, la sécurité est devenue au fil des années une préoccupation majeure pour l'ensemble des acteurs. Cependant, l'emploi massif de matériaux composites, dont les propriétés physiques sont très intéressantes pour les constructeurs mais qui sont conçus avec une marge de tolérance pour des raisons de coût, induit des variations indésirables dans la structure, des incertitudes. Outre ces matériaux, d'autres éléments non prévisibles sont susceptibles de perturber la structure de l'appareil. Le modèle d'un avion en avant-projet est toujours amené à évoluer pour répondre aux évolutions des exigences du constructeur, mais des études de faisabilité doivent être menées avant que la structure ne soit totalement définie, afin de s'assurer de la viabilité du modèle. Des éléments non pris en compte dans la structure, comme les câbles, peuvent également avoir une influence non négligeable sur le comportement global de l'appareil. Ces incertitudes ont un impact non négligeable sur la stabilité de la structure en vol. Des études ont commencé à incorporer cet aspect incertain dans les processus d'optimisation, mais généralement en adaptant les algorithmes existants et sans exploiter la nature incertaine des problèmes. Afin de tenir compte de l'aspect incertain, on se propose de représenter ces incertitudes par des variables aléatoires et d'exploiter des outils théoriques développés dans d'autres domaines, notamment les outils des mathématiques financières.

Mots-clefs

Optimisation, Aéroélasticité, Incertitudes, Contraintes probabilistes, Recuit simulé stochastique, Arrow-Hurwicz stochastique

Optimization of an aeroelastic system with uncertainties

Abstract

Research in optimization is a fundamental field for aircraft manufacturers. The performance of airplanes is a crucial element in the commercial competition that pits main aircraft manufacturers against each other. Since 2000, composite materials have been more and more used in aircraft design. This shows the manufacturers' desire to reduce the airplane weight to diminish kerosene consumption. At the same time, safety has become a major concern for all the parties involved. But the use of composite materials, which are designed with a margin of tolerance on the physical properties to cut cost, causes unwanted variations in the structure. Other factors in the airplane design could also disrupt the overall structure of the final product. An airplane model is intended to be modified according to the manufacturer's wishes and their evolution, but feasibility studies must be carried out before the structural design is complete, in order to make sure that the model is viable. Elements which are not modeled in the structure, e.g. cables, can affect the overall behavior of the airplane. These uncertainties have a non-negligible influence on the stability of the structure during flights. Some studies have started to take into account these uncertainties in the optimization process, but they usually consist in adapting existing deterministic algorithms, without regard for the inherent uncertainty of the problem. In order to take uncertainty into account, we propose to represent these uncertainties by random variables and to use theoretical tools that are used in other domains, such as financial mathematics.

Keywords

Optimization, Aeroelasticity, Uncertainties, Probabilistic constraints, Stochastic annealing, Stochastic Arrow-Hurwicz

Table des matières

Introduction	9
Notations	13
1 Introduction et état de l'art	15
1.1 État de l'art	19
1.2 Plan de la thèse	20
2 Les équations de l'aéroélasticité	23
2.1 Modélisation de la partie structure	23
2.2 Introduction d'incertitudes dans le modèle	32
3 Algorithmes d'optimisation	35
3.1 Poser un problème d'optimisation	35
3.2 Le cadre déterministe	38
3.3 Le recuit simulé	44
3.4 Le cadre stochastique	47
3.5 Algorithme de Arrow-Hurwicz stochastique	50
3.6 Le recuit simulé stochastique	53
4 Optimisation d'une maquette d'aile d'avion	61
4.1 Le modèle structure	61
4.2 Le modèle aérodynamique	64
4.3 Optimisation de la répartition de masse	69
4.4 Minimisation de la masse de la maquette	81
4.5 Cas pathologiques	86
5 Conclusions et discussions	89
5.1 Apport de la thèse	89
5.2 Difficultés rencontrées non résolues	91
5.3 Perspectives	92
A Projection sur un espace convexe non trivial	93
B Rappels en probabilités	97
B.1 Mesure et espérance	97
B.2 Chaînes de Markov	98

C Suites finies à discrédance faible	101
C.1 Propriétés de ces suites	101
C.2 Construction de suites à discrédance faible	104
D Convergence du gradient stochastique en norme L^2	105
E Calcul de la probabilité approchée pour Arrow-Hurwicz stochastique	109
E.1 Formules des contraintes approchées	109
E.2 Choix des paramètres	110
F Chaîne CAPRI	113
F.1 Description	113
F.2 Programmes utilisés	113
Bibliographie	115

Chapitre 1

Introduction et état de l'art

L'optimisation multi-critères est un axe de recherche fondamental depuis les débuts de l'aviation civile. Les avions de ligne sont des appareils extrêmement complexes et nécessitent de nombreux compromis entre plusieurs paramètres pour fonctionner avec efficacité. Au cours des dernières années, de nouveaux enjeux sont entrés en ligne de compte dans la conception globale des avions de ligne, notamment depuis le protocole de Kyoto qui a fait du changement climatique l'un des principaux enjeux du XXI^e siècle. Tous les secteurs industriels, y compris le secteur aéronautique, sont donc contraints de réduire leur rejet d'émissions polluantes dans l'atmosphère. Les acteurs de ce secteur s'accordent également sur le fait que le trafic aérien et donc les émissions polluantes vont croître à long terme. À ces nouvelles exigences environnementales, s'ajoutent celles des compagnies aériennes qui demandent des produits plus compétitifs tant au niveau de la fiabilité que du coût de fonctionnement (maintenance, consommation de carburant, etc.). La production de nouveaux avions plus performants et efficaces passe donc par l'amélioration de l'aérodynamisme de l'appareil, le rendement de ses moteurs et surtout l'allègement de sa structure afin de réduire la consommation de kérozène, et donc l'émission de gaz polluants. La compétition entre les matériaux métalliques mieux maîtrisés et les matériaux composites plus innovants a donc été relancée, et l'utilisation de ces derniers ne cesse de s'amplifier dans ce domaine.

Au départ, l'utilisation de ces matériaux était cantonnée à des structures secondaires faiblement chargées mécaniquement. Une meilleure compréhension du comportement matériau et une meilleure maîtrise des procédés de fabrication a permis leur utilisation pour la fabrication de structures primaires assurant l'intégrité structural de l'avion. Par exemple, pour l'Airbus A380, le caisson central de voilure a été entièrement réalisé en matériaux composites et a ainsi permis une diminution de la masse de plus de 1.5 tonne en comparaison d'une solution métallique.

Les derniers avions en date des constructeurs Boeing et Airbus comportent une part prédominante de matériaux composites. Le 787 est composé à hauteur de 50% de matériaux composites, tandis que dans l'A350, ceux-ci représentent 53% de la masse de l'appareil. L'avantage de ces matériaux est qu'ils possèdent un rapport masse/rigidité/résistance important par rapport aux matériaux métalliques et une plus grande durée de vie, réduisant ainsi les coûts de maintenance et les intervalles d'inspection.

Outre ces nouveaux enjeux, la conception d'un nouvel appareil est un processus très long et complexe, qui met en œuvre différents champs de compétences très éloignés les uns des autres. Au cours des années, ses différentes étapes ont été améliorées au fur et à mesure des progrès réalisés dans les technologies associées. Les souffleries ont permis de vérifier au

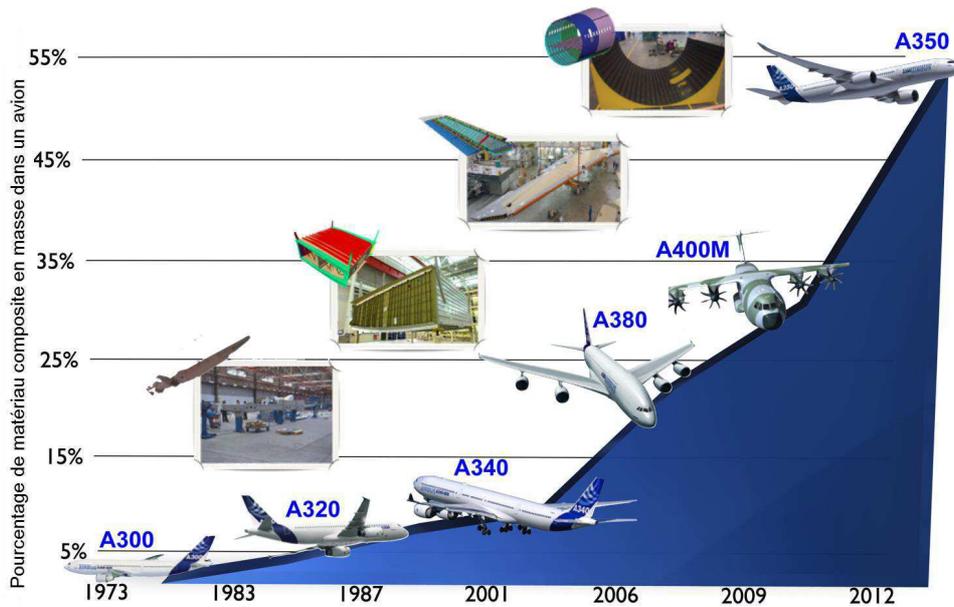


FIG. 1.1: Évolution du pourcentage de matériaux composites en masse dans la gamme d'avions Airbus au cours des 40 dernières années

sol le comportement de la structure sans faire d'essais en vol. Grâce aux équations d'Euler et de Navier-Stokes, il a été possible de définir de façon rigoureuse un cadre physique permettant de caractériser le comportement de l'avion en fonction de différents paramètres, comme la vitesse ou le nombre de Mach. Le développement des outils informatiques et de la modélisation numérique a permis aux ingénieurs de créer des modèles numériques de plus en plus réalistes, permettant de se passer des feuilles de dessin, de réaliser de nombreux tests numériques, d'obtenir une grande souplesse de manipulation de ces modèles, et surtout d'accélérer drastiquement la phase de conception d'un nouvel appareil. Les modèles à éléments finis ont permis une grande simplification de la représentation d'un modèle d'avion et de sa structure, ainsi que la manière dont il réagit à l'effort.

À l'image des autres domaines de l'industrie, la recherche en optimisation est donc intense dans le secteur aéronautique [85, 96]. Une évolution importante de la fin des années 80 est l'introduction de paramètres incertains dans les modèles numériques [15, 18, 45, 52, 60, 61, 68, 69, 70, 71, 73, 74, 75]. L'étude de techniques d'optimisation en présence d'incertitudes en aéroélasticité est plus récente et n'a commencé qu'au début des années 2000, bien qu'il y ait déjà eu des recherches en optimisation stochastique de design dans d'autres domaines industriels bien avant cette époque [12, 13, 19, 84, 88]. Cette incorporation a été effectuée pour tenir compte de certaines situations pouvant rendre difficile une évaluation précise du comportement de l'avion. Par exemple, au cours de la phase de conception d'un avion, afin de répondre à différents critères ou dans le but d'éliminer certains problèmes que rencontre le modèle, ce dernier est amené à évoluer pour répondre efficacement aux demandes et aux besoins qu'il est censé satisfaire. Le modèle n'est pas figé dès les premières ébauches, mais pour des raisons de sécurité, il est nécessaire de vérifier, tout au long du processus de développement, que la structure est capable de résister aux contraintes auxquelles elle est susceptible d'être soumise durant son exploitation. Pour tenir compte de ces possibles modifications, une procédure envisageable est d'introduire des incertitudes dans le modèle.

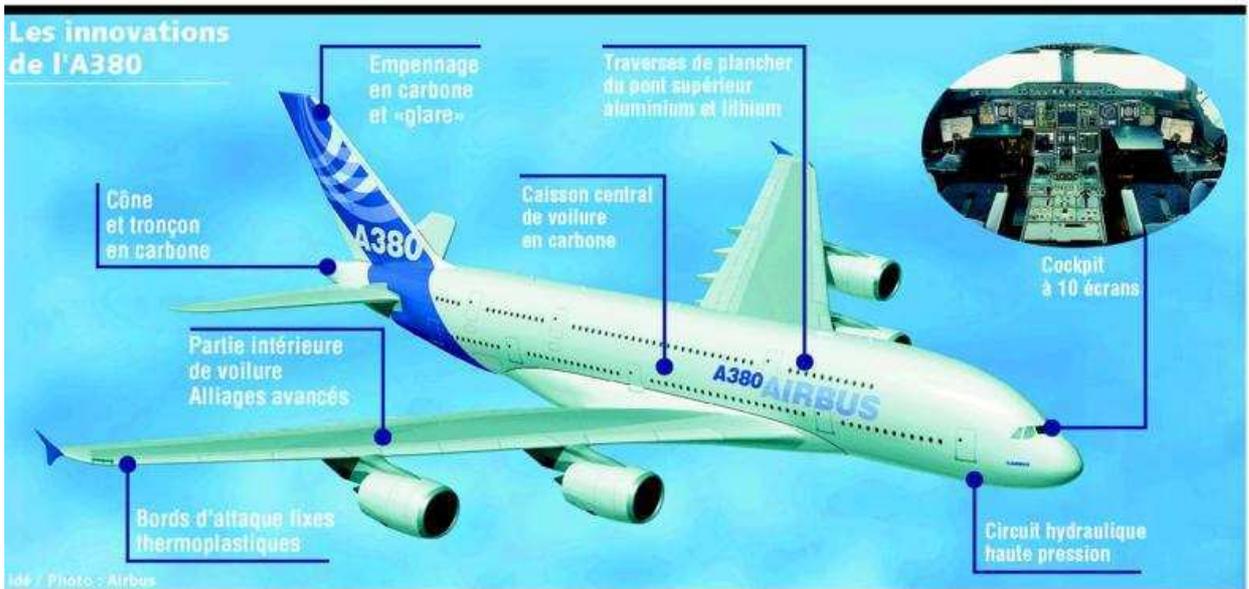


FIG. 1.2: Principales innovations de l'A380, dont l'incorporation de matériaux en fibres de carbone

Une autre source d'incertitudes potentielle est l'emploi précédemment évoqué de matériaux composites. Un matériau composite est en général un matériau hybride, le plus souvent composé de plusieurs couches de matériaux non miscibles, afin d'octroyer à l'ensemble ainsi fabriqué des propriétés que les matériaux naturels ne possèdent pas séparément. Néanmoins, contrairement au cas des matériaux dits classiques, les propriétés physiques des matériaux composites sont parfois difficiles à prévoir et à quantifier de par leur variété. Elles dépendent de la nature des matériaux qui la composent, du nombre de ces matériaux, de la manière dont ils sont empilés, et même de l'orientation des fibres à l'intérieur des couches successives. Une légère erreur au niveau du processus de fabrication, notamment, peut suffire à provoquer une variation non négligeable sur les propriétés physiques. Les constructeurs peuvent affiner autant qu'ils le souhaitent les valeurs des paramètres des matériaux, comme par exemple le module d'Young qui représente la rigidité du matériau. Cependant, pour obtenir un matériau avec les caractéristiques souhaitées, il faut notamment du matériel de grande précision et du personnel hautement qualifié, des éléments qui représentent un coût et un investissement très importants pour les industriels. Ainsi, plus la précision voulue est fine, plus le coût de production du matériau est élevé. Le plus souvent, les constructeurs produisent donc des matériaux composites avec un pourcentage de précision par rapport aux valeurs nominales, afin de garder dans une fenêtre raisonnable le coût de production et des pièces de rechange des avions.

De plus, lors de la conception d'un avion, les constructeurs souhaitent naturellement optimiser les performances du véhicules : réduire les émissions polluantes, le bruit, la traînée, étendre le rayon d'action, maximiser sa stabilité, etc. La minimisation de la masse de la structure est un objectif important pour les constructeurs, qui aura pour impact de satisfaire d'autres critères d'optimisation, comme la réduction des pollutions ou l'allongement du rayon d'action. Mais cela aura également un effet négatif sur d'autres critères, notamment la stabilité de l'appareil en vol, pouvant entraîner le phénomène de "flottement".

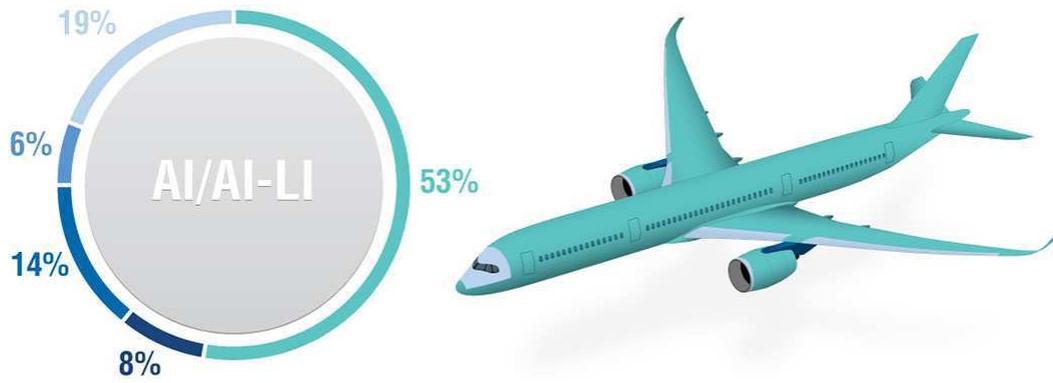


FIG. 1.3: Répartition des types de matériaux dans la structure de l'A350

Il s'agit d'un état instable lié au couplage aéroélastique entre la structure et les forces aérodynamiques, et qui provoque la destruction de l'appareil. Ce phénomène se produit généralement au niveau des ailes et correspond au rapprochement de deux modes structuraux de l'aile. En général, plus le poids d'une aile est faible, plus celle-ci est sensible au flottement. Le constructeur doit donc réaliser une optimisation sous contraintes : minimiser le poids de l'aile tout en s'assurant que le phénomène de flottement ne pourra pas affecter l'avion dans son domaine de vol. Les problèmes d'optimisation en aéroélasticité sont des problèmes multidisciplinaires car mettant en jeu à la fois les comportements structurel et aérodynamique.

Nous souhaitons à présent inclure les aspects incertains précédemment évoqués dans des processus d'optimisation. Mais pour ce faire, il faut en premier lieu identifier la nature de ces incertitudes et comment les représenter dans un modèle d'avion. Plusieurs types d'incertitudes ont été identifiés et classés selon leur nature. Les incertitudes sont généralement classifiées selon 3 grandes catégories : les incertitudes aléatoires, les incertitudes épistémiques et les erreurs liées à l'imprécision des moyens de calculs (numériques comme matériels).

Ici, nous ne considérerons que la première catégorie d'incertitudes. Le problème qu'on se propose d'étudier dans ce travail est de résoudre numériquement un problème d'optimisation probabiliste, où la fonction objectif est issue de l'analyse aéroélastique. Les difficultés du problème découleront, comme nous l'avons noté plus haut, du caractère multidisciplinaire du problème. Nous nous intéresserons donc au problème global suivant :

$$\underset{x}{\operatorname{Argmin}} \mathbb{E}[f(x, \xi)], \quad x \in \mathbb{R}^n, \quad \xi \in \mathbb{R}^p, \quad (1.1)$$

qui pourra être agrémenté de contraintes déterministes ou probabilistes.

Notre but étant d'adapter les algorithmes d'optimisation stochastique, tels que ceux utilisés par exemple en ingénierie financière, à des problèmes d'optimisation aéroélastique fortement non linéaire, et leur apporter un cadre théorique permettant de généraliser ces méthodes à d'autres cas.

1.1 État de l'art

Les ingénieurs aéronautiques ont développé plusieurs types de techniques d'optimisation afin de répondre aux différents enjeux d'un aéroplane. Cela va du choix de l'orientation des pales dans les moteurs d'avion à la répartition du carburant dans les ailes au cours du vol. Des techniques employant des outils déterministes ont été mises en place et appliquées à de nombreux niveaux, mais celles-ci ne tiennent pas compte d'éventuels paramètres incertains dans la structure. La prise en compte des incertitudes est un sujet d'étude depuis les années 50, et dans de nombreux domaines de recherche [84, 88], mais pendant longtemps dans la recherche aéronautique, il n'a pas été possible de prendre en compte, voire de quantifier, ces incertitudes structurelles dans les procédures d'optimisation à cause de la limitation des outils numériques et de la connaissance théorique de leur impact dans les études de fiabilité. Les ingénieurs ont donc dû mettre en place des procédures alternatives permettant de simplifier l'incorporation des incertitudes structurelles dans les modèles développés. Ce n'est que depuis les années 90 que des études sont menées sur ce sujet dans le secteur aérien et que ce domaine de recherche commence à produire des résultats tangibles. Une revue des principales manières de modéliser l'incertain et de l'incorporer dans les modèles physiques dans les problèmes aéronautiques est présentée dans l'article [99].

Parmi ces approches, la plus développée et la plus utilisée chez les industriels est l'optimisation robuste [7]. Elle consiste à approcher l'ensemble de solutions admissibles défini par les contraintes stochastiques par une famille de contraintes linéaires. L'idée est d'exclure les cas les plus improbables statistiquement et physiquement non réalistes, afin de rendre convexe l'ensemble des solutions admissibles. Ainsi, il devient possible d'employer des méthodes tirées de l'optimisation déterministe en utilisant le caractère convexe de l'ensemble de définition restreint. Cependant, cette méthode est rapidement très limitée si les contraintes stochastiques sont trop complexes. De plus, elle s'adresse généralement à des problèmes d'optimisation linéaire et n'est pas aussi efficace en optimisation non linéaire. Il existe dans la littérature de nombreuses études d'optimisation robuste appliquées à l'aéronautique, dont on peut en trouver une revue dans [88].

Un autre type d'algorithmes classiquement employés est la méthode par scénario [27, 93]. Ces techniques s'appliquent lorsque le problème requiert la vérification de contraintes en probabilité. Le principe est d'effectuer un certain nombre de réalisations des variables aléatoires du système et de remplacer les contraintes en probabilité du problème par un sous-ensemble des contraintes déterministes ainsi obtenues, le cardinal du sous-ensemble dépendant de la valeur en probabilité de la contrainte. Ensuite, il s'agit d'effectuer une optimisation déterministe à l'aide d'outils classiques.

En ce qui concerne les problèmes d'optimisation sous contraintes probabilistes, l'optimisation fiabiliste, très répandue dans le milieu industriel, substitue à la contrainte probabiliste un second problème d'optimisation déterministe par le biais de techniques d'approximation FORM-SORM [23, 36, 54, 57]. Ces approches, en se plaçant dans un espace de variables gaussiennes, permettent d'exprimer la contrainte sous la forme d'un quantile de la loi gaussienne. Cependant, comme pour l'optimisation robuste et la méthode par scénario, ce type d'optimisation nécessite généralement de réécrire le problème sous forme linéaire ou de façon à pouvoir employer des techniques essentiellement déterministes [90]. Ces dernières années, des méthodes d'optimisation moins conventionnelles commencent à voir le

jour. On peut citer par exemple l'emploi d'un algorithme hybride gradient-génétique [17]. Néanmoins, le nombre d'études utilisant des outils stochastiques en optimisation dans l'aéronautique reste encore faible.

Le but de cette thèse est de travailler directement avec des outils probabilistes, comme ceux utilisés en mathématiques financières [8, 34, 50]. Parmi les approches basées sur l'estimation stochastique les plus employées, on peut citer l'algorithme de Robbins-Monro [28, 29, 77], historiquement le premier algorithme faisant usage des propriétés stochastiques des problèmes de nature aléatoire. Ce fut le point de départ du travail réalisé dans cette thèse.

1.2 Plan de la thèse

Cette thèse est organisée en 3 parties principales. La première partie pose le contexte physique, la seconde partie développe les outils mathématiques étudiés, et la troisième partie montre les résultats numériques ainsi obtenus. Le tout sera suivi d'une discussion sur les apports de la thèse, les limitations rencontrées et les perspectives envisagées pour des études futures.

Dans un premier chapitre, nous ferons un rappel général sur la théorie aéroélastique et du comportement des solides soumis à l'écoulement viscoélastique d'un fluide. Nous y faisons le rappel de notions permettant de poser l'équation aéroélastique standard qui constituera le socle de notre étude. Ensuite, nous ferons une étude de stabilité et caractériserons le flottement, phénomène instable destructeur pour la structure. Puis, nous mettrons en évidence la notion de paramètre critique, la valeur minimale d'un paramètre physique donné à laquelle se produit le phénomène de flottement. Nous poursuivrons ce chapitre avec l'établissement d'une formule analytique du gradient de ce paramètre critique. Enfin, nous concluerons cette partie en introduisant de manière formelle des incertitudes dans le modèle physique, rendant ainsi le modèle aléatoire ainsi que les grandeurs associées.

Dans le chapitre suivant, nous présenterons un état de l'art des algorithmes d'optimisation classiques. Après avoir fait une liste non exhaustive mais assez complète sur les principaux algorithmes mis au point en optimisation déterministe, nous ferons de brefs rappels sur les métaheuristiques, des méthodes développées pour résoudre des problèmes déterministes mais reposant sur des outils stochastiques en s'inspirant du monde du vivant, principalement pour parcourir l'espace des solutions admissibles. Puis, nous exposerons la nature d'un problème d'optimisation stochastique, ainsi que les principaux algorithmes d'optimisation stochastique existants. Nous avons porté une attention toute particulière sur trois de ces méthodes : le gradient stochastique, la méthode d'Arrow-Hurwicz stochastique et le recuit simulé stochastique.

Dans un dernier chapitre, nous reviendrons sur les tests numériques que nous avons effectués pour mettre en évidence l'efficacité des algorithmes étudiés au chapitre précédent dans un contexte aéroélastique. Nous nous sommes servis du modèle numérique d'une maquette d'avion, l'aile Dyvas. Après avoir exposé les caractéristiques du modèle structure et du modèle aéro associé, et exposé la façon dont nous avons modélisé les incertitudes, nous avons posé deux problèmes d'optimisation. Nous avons ensuite testé les algorithmes du gradient stochastique et du recuit simulé stochastique sur le premier problème posé,

qui consiste à maximiser la pression critique de l'aile Dyvas. Dans un deuxième temps, nous nous sommes penchés sur le second problème ayant pour but de minimiser la masse du modèle sous une contrainte en probabilité sur la pression critique, et avons employé l'algorithme de Arrow-Hurwicz stochastique. Finalement, nous avons conclu ce chapitre en faisant une revue des cas de divergence numérique auxquels nous avons été confrontés, afin de mettre en garde le lecteur contre ces éventuels problèmes.

Après ces trois parties principales, nous avons fait un bilan de l'apport de cette thèse, ainsi que les obstacles que nous avons rencontrés. Nous avons terminé la revue d'ensemble par les perspectives possibles et envisageables dans la continuité de cette thèse, afin de poursuivre le développement de l'optimisation stochastique en aéroélasticité, de l'intégrer à de l'optimisation multi-objectifs et de permettre son incorporation à un processus industriel de conception aéronautique.

Dans les annexes, nous avons explicité quelques détails techniques, dont un algorithme de projection sur un convexe non trivial, et les principales définitions des éléments stochastiques employés dans cette thèse.

Chapitre 2

Les équations de l'aéroélasticité

De manière schématique, les forces exercées sur une structure par l'écoulement d'un fluide autour de celle-ci vont la déformer, ce qui aura pour effet de modifier ces mêmes forces, car elles dépendent de la géométrie de la surface portante. Leurs actions sur la structure vont s'en trouver alors modifiées, entraînant une nouvelle déformation de son profil aérodynamique. Cela induit un couplage entre le fluide et la structure qui, dans certains cas, peut entraîner des mouvements oscillatoires explosifs.

L'étude du comportement de ce système couplé et plus particulièrement de sa stabilité passe par la construction d'un modèle réduit de la structure grâce à la méthode des éléments finis et par l'utilisation de codes aérodynamiques permettant d'évaluer les forces aérodynamiques s'exerçant sur la structure. Afin de réduire la dimension du problème qui peut être très grande dans un cas industriel (un million de degrés de liberté), le mouvement de la structure en basse fréquence est décrit par un modèle réduit de quelques dizaines de modes propres.

2.1 Modélisation de la partie structure

Dans ce paragraphe, nous allons établir les équations décrivant la structure.

2.1.1 Partie structure

Décrivons à présent la partie structurelle de notre solide. On considère D un domaine borné de \mathbb{R}^3 occupé par le solide. On note ∂D sa frontière et Γ une partie fixe de celle-ci. Cette partie fixe représentera, dans le cadre d'une aile d'avion, le raccord avec le fuselage de l'appareil ou, dans le cadre d'essais en soufflerie, la partie de la maquette fixée au mur.

Considérons $v \in \mathbb{R}^3$ un champ de déplacement vérifiant les équations de l'élastodynamique dans D . Les forces surfaciques sur $\partial D \setminus \Gamma$ sont dues à l'écoulement du fluide et s'expriment localement en fonction de la pression qu'il exerce sur l'élément de surface : $\vec{d}f = p\vec{n}dS$. Le vecteur \vec{n} est ici le vecteur unitaire extérieur normal à ∂D . Si ρ_M est la densité volumique du solide et σ le tenseur des contraintes, le problème aux limites pour l'élastodynamique s'écrit pour $i, j \in \{1, 2, 3\}$:

$$\begin{aligned} \rho_M \partial_t^2 v_i - \sum_j \partial_j \sigma_{i,j} &= 0 && \text{dans } D \\ v(x, t) &= 0 && \text{sur } \Gamma \\ \sum_j \sigma_{i,j}(v) n_j &= -p(v) n_i && \text{sur } \partial D \setminus \Gamma \end{aligned} \tag{2.1}$$

Ces équations sont complétées par les conditions initiales $v(x, 0) = v_0$ et $\partial_t v(x, 0) = \dot{v}_0$. Le tenseur linéarisé des déformations s'écrit :

$$\epsilon_{kh} = \frac{1}{2}(\partial_h v_k + \partial_k v_h). \quad (2.2)$$

En outre, pour un matériau viscoélastique à mémoire instantanée, le tenseur symétrique des contraintes s'écrit :

$$\sigma_{ij} = \sum_{k,h} a_{ijkh} \epsilon_{kh}(v) + b_{ijkh} \epsilon_{kh}(\dot{v}), \quad (2.3)$$

où les tenseurs des coefficients réels élastiques a_{ijkh} et des coefficients réels d'amortissement b_{ijkh} vérifient les propriétés classiques de symétrie et de positivité.

Formulation variationnelle et projection de Ritz-Galerkin

On introduit l'ensemble $H^1(D)$ des fonctions définies sur D , à valeurs dans \mathbb{R} , de carré intégrable et dont les dérivées partielles premières sont de carré intégrable. On se restreint à l'espace vectoriel des solutions admissibles $\mathcal{V}(D) = \{\nu \in \{H^1(D)\}^3 \mid \nu = 0 \text{ sur } \Gamma\}$.

Soient $v, \mu : P \in D \mapsto v(P), \mu(P)$ deux champs cinématiquement admissibles vérifiant les équations de l'élastodynamique. On munit \mathbb{R}^3 du produit scalaire usuel. Pour le champ de déplacement $\nu \in \mathcal{V}$, la formulation variationnelle associée au problème (2.1) s'écrit, quel que soit le champ $v \in \mathcal{V}$:

$$\begin{aligned} \sum_i \int_D \rho \partial_t^2 v_i \nu_i dP + \sum_{i,j,k,h} \int_D b_{ijkh} \epsilon_{kh}(\partial_t v) \epsilon_{ij}(\nu) dP + \sum_{i,j,k,h} \int_D a_{ijkh} \epsilon_{kh}(v) \epsilon_{ij}(\nu) dP \\ = \sum_i \int_{\partial D \setminus \Gamma} -p(v) n_i \nu_i dS. \end{aligned} \quad (2.4)$$

On définit alors les formes bilinéaires de masse, d'amortissement et de raideur :

$$\begin{aligned} m(v, \nu) &= \int_D \rho_M(x) v \nu dx, \\ c(v, \nu) &= \sum_{i,j,k,h} \int_D b_{ijkh}(v) \epsilon_{kh}(v) \epsilon_{ij}(\nu) dx, \\ k(v, \nu) &= \sum_{i,j,k,h} \int_D a_{ijkh} \epsilon_{kh}(v) \epsilon_{ij}(\nu) dx. \end{aligned}$$

De plus, à v donné, nous introduisons la forme bilinéaire $\nu \rightarrow f(v, \nu)$ donnée par la formule suivante :

$$f(v, \nu) = \sum_i \int_D -p(v) \vec{n}_i \nu_i d\vec{S}. \quad (2.5)$$

Nous pouvons alors réécrire le problème (2.4) de manière équivalente à l'aide des formes bilinéaires introduites ci-dessus :

$$m(\partial_t^2 v, \nu) + c(\partial_t v, \nu) + k(v, \nu) = f(v, \nu). \quad (2.6)$$

La formulation variationnelle du problème s'exprime alors :

trouver une fonction $v \in \mathcal{V}$ telle que l'équation (2.6) soit vérifiée pour tout $\nu \in \mathcal{V}$.

Les formes bilinéaires définissent des opérateurs linéaires auto-adjoints continus positifs dans l'espace dual \mathcal{V}^* , tel l'opérateur de masse $\mathcal{M} : \mathcal{V} \rightarrow \mathcal{V}^*$:

$$\langle \mathcal{M}v, \nu \rangle = m(v, \nu), \quad (2.7)$$

où \mathcal{V}^* est l'espace dual de \mathcal{V} . De même, on définit les opérateurs d'amortissement \mathcal{C} et de raideur \mathcal{K} :

$$\langle \mathcal{C}v, \nu \rangle = c(v, \nu), \quad (2.8)$$

$$\langle \mathcal{K}v, \nu \rangle = k(v, \nu), \quad (2.9)$$

ainsi que la forme linéaire :

$$\langle \mathcal{F}(v), \nu \rangle = f(v, \nu). \quad (2.10)$$

Le problème peut alors s'écrire sous la forme d'opérateurs :

$$\mathcal{M}\partial_t^2 v + \mathcal{C}\partial_t v + \mathcal{K}v = \mathcal{F}(v). \quad (2.11)$$

Dans la suite, par bijection et par souci de simplification d'écriture, nous confondrons les ensembles \mathcal{V} et \mathcal{V}^* .

2.1.2 Construction d'une solution approchée

Il est possible de construire une solution approchée du problème (2.11) dans un sous-espace de dimension finie N , noté $\mathcal{V}^N \subset \mathcal{V}$, en utilisant la méthode de projection de Ritz-Galerkin. Dans ce but, on choisit une base $\kappa = \{\kappa_1, \dots, \kappa_N\}$ de \mathcal{V}^N , telle que tout élément de \mathcal{V}^N puisse s'écrire sous la forme suivante :

$$v^N(x, t) = \sum_{i=1}^N Q_i(t) \kappa_i(x). \quad (2.12)$$

On approche alors la fonction v dans \mathcal{V}^N par la fonction v^N :

$$v(x, t) \approx v^N(x, t) = \sum_{i=1}^N Q_i(t) \kappa_i(x). \quad (2.13)$$

En introduisant cette expression dans l'équation (2.11), on obtient la formulation variationnelle restreinte à l'espace \mathcal{V}^N .

2.1.3 Discrétisation par la méthode des éléments finis

On souhaite maintenant réaliser une discrétisation du domaine D par la méthode des éléments finis. Pour ce faire, on construit une base de \mathcal{V}^N adaptée à la méthode des éléments finis telle que :

$$v^N(x, t) = \sum_{i=1}^N Q_i(t) w_i(x), \quad (2.14)$$

où N est le nombre de degré de libertés et où w_i sont les fonctions polynomiales d'interpolation des éléments finis qui engendrent \mathcal{V}^N . Celles-ci sont choisies de telle sorte que $w_i(P_i^j) = 1$ et $w_i(P_j^j) = 0$, où P_i^j désigne le degré de liberté j du nœud i . Pour plus de détails, le lecteur peut se référer aux ouvrages de référence [11, 100].

2.1.4 Construction d'un modèle réduit du système couplé

Dans la suite, nous supposons que la structure est discrétisée et donc représentée par la méthode des éléments finis. L'équation du mouvement de la structure linéaire soumise à des forces extérieures dues à un écoulement fluide peut alors s'écrire sous la forme suivante :

$$\mathbb{M}\ddot{Q}(t) + \mathbb{C}\dot{Q}(t) + \mathbb{K}Q(t) + (\mathbb{F}(t) - \mathbb{F}_0) = 0. \quad (2.15)$$

$Q(t) \in \mathbb{R}^N$ est le vecteur des degrés de liberté, \mathbb{M} la matrice de masse, \mathbb{C} la matrice d'amortissement et \mathbb{K} la matrice de raideur. $\mathbb{F}(t) - \mathbb{F}_0$ représente le vecteur des forces aérodynamiques qui sont calculées en sommant les pressions appliquées sur chaque élément. Le déplacement total $\tilde{Q}(t)$ est composé d'une partie statique et d'une partie dynamique : $\tilde{Q}(t) = Q_0 + Q(t)$. La déformée statique Q_0 est due à la partie stationnaire des forces aérodynamiques F_0 . Elle est solution de l'équation :

$$\mathbb{K}Q_0 + \mathbb{F}_0 = 0. \quad (2.16)$$

Dans la suite, nous ne nous intéresserons qu'au comportement dynamique en basse fréquence du système couplé.

Il est possible d'obtenir un modèle réduit en projetant l'équation (2.15) par la méthode de Ritz sur le sous-espace C^n engendré par n vecteurs ($n \ll N$) : ce sont les n premiers modes propres, dont nous allons rappeler la définition.

Modes propres du modèle à éléments finis

Le problème spectral conservatif associé à l'équation (2.15) s'écrit de la façon suivante :

$$(\mathbb{K} - \omega_i^2 \mathbb{M})\varphi_i = 0. \quad (2.17)$$

Puisque les matrices \mathbb{K} et \mathbb{M} sont symétriques positives (avec la matrice \mathbb{M} strictement positive), le problème spectral admet N valeurs propres positives notées ω_i^2 :

$$0 \leq \omega_1^2 \leq \dots \leq \omega_N^2. \quad (2.18)$$

À chaque solution ω_j^2 correspond un mode propre $\varphi_j \in \mathbb{R}^N$. La famille $(\varphi_j)_{j=1,N}$ forme alors une base de \mathbb{R}^N , et on peut écrire :

$$Q(t) = \sum_{i=1}^N q_i(t)\varphi_i, \quad (2.19)$$

où les fonctions réelles $(q_j)_{j=1,N}$ sont appelées les coordonnées généralisées. De plus, on a les égalités suivantes :

$$\langle \mathbb{M}\varphi_i \varphi_j \rangle = M_{ij} = \mu_i \delta_{ij} \quad (2.20)$$

$$\langle \mathbb{K}\varphi_i \varphi_j \rangle = K_{ij} = \mu_i \omega_i^2 \delta_{ij} \quad (2.21)$$

où $\mu_i \in \mathbb{R}_+$ est la masse généralisée du mode φ_i et δ désigne le symbole de Kronecker ($\delta_{ij} = 1$ si $i = j$, $\delta_{ij} = 0$ sinon).

Projection sur la base modale tronquée

La solution de l'équation (2.15) est approchée par sa projection sur la base tronquée réduite aux n premiers modes propres $\varphi_1, \dots, \varphi_n$:

$$Q(t) \approx Q^n(t) = \sum_{i=1}^n q_i(t) \varphi_i = \Phi q(t), \quad (2.22)$$

avec $\Phi \in \mathcal{M}_{\mathbb{R}}(N, n)$ la matrice comportant dans ses colonnes les n vecteurs propres de la base modale tronquée. Le mouvement de la structure s'écrit alors $\tilde{Q}^n(t) = Q_0 + Q^n(t)$. En notant $q(t) = (q_1(t), \dots, q_n(t)) \in \mathbb{R}^n$ le vecteur des coordonnées généralisées, nous obtenons :

$$M\ddot{q}(t) + C\dot{q}(t) + Kq(t) + F(t) = 0. \quad (2.23)$$

Ici, $F(t) = (F_1(t), \dots, F_n(t))$ et $F_i(t) = \langle (\mathbb{F}(t) - \mathbb{F}_0), \varphi_i \rangle$. La matrice des masses généralisées M et la matrice des raideurs généralisées K sont des matrices diagonales. La matrice d'amortissement généralisé est donnée par :

$$\langle C\varphi_i, \varphi_j \rangle = C_{ij}. \quad (2.24)$$

En pratique, ce terme est difficile à modéliser, et il est donc souvent rajouté sur des bases phénoménologiques ou expérimentales dans l'analyse modale. Par ailleurs, le modèle à éléments finis est construit à partir de structures réelles testées en soufflerie. À partir de mesures expérimentales permettant d'obtenir la déformée statique de la structure, nous pouvons calculer pour chaque mode propre la déformée, la fréquence, la masse généralisée, etc., et par soustraction, obtenir la composante de l'amortissement. Aussi, bien souvent, cette matrice est remplacée par une matrice diagonale dont les composantes représentent l'amortissement modal calculé expérimentalement.

Les FAG (forces aérodynamiques généralisées) se calculent comme projection des forces de pression sur les modes de la structure. Cela donne :

$$F_i = \int_S p \vec{n} \varphi_i d\vec{S}, \quad (2.25)$$

où l'intégrale \int_S porte sur la surface du profil. Les forces aérodynamiques dépendent de la vitesse moyenne de l'écoulement, qui est une donnée du problème. En présence de turbulences, elles dépendent également des composantes fluctuantes de l'écoulement.

2.1.5 Calcul des forces aérodynamiques instationnaires

Dans le cadre de cette thèse, nous nous sommes intéressés à une aérodynamique linéaire, comme cela est souvent pratiqué dans un contexte industriel. Un mouvement harmonique est imposé à la structure, et seule la première harmonique de chaque réponse est conservée. Ces approximations permettent alors d'exprimer les FAG linéairement dans le domaine fréquentiel. En posant

$$\hat{F}(s) = \int_{-\infty}^{+\infty} e^{-st} F(t) dt \quad (2.26)$$

la transformée de Laplace des FAG, nous pouvons alors écrire :

$$\hat{F}(s) = H(s) \hat{q}(s). \quad (2.27)$$

Les forces aérodynamiques dépendent en outre de la vitesse d'écoulement V et de la densité volumique du fluide ρ . Cette dépendance est décrite par la relation :

$$H(s) = \frac{1}{2}\rho V^2 A(s) \quad (2.28)$$

où l'application $A : \mathbb{C} \rightarrow \mathbb{C}^{n \times n}$ est une fonction matricielle définissant la matrice globale des coefficients aérodynamiques et $\hat{q}(s)$ la transformée de Laplace du vecteur des coordonnées généralisées $q(t)$. Pour calculer la matrice $A(s)$, la méthode des doublets sera utilisée [3]. Il s'agit d'une méthode mise au point dans les années 60 et qui est couramment utilisée par les industriels pour le calcul des FAG dans le domaine subsonique. Sans rentrer dans les détails, il est à signaler que cette méthode est très efficace et robuste, mais elle ne peut prendre en compte des non-linéarités, en particulier les discontinuités dues aux ondes de choc lors d'un écoulement transsonique. Dans le cadre de cette thèse, le calcul des FAG par les doublets sera effectué par l'ensemble de codes de la chaîne CAPRI développée à l'ONERA [59].

Compte tenu de la relation (2.27), l'équation aéroélastique s'écrit alors dans le domaine fréquentiel :

$$(s^2 M + K + \frac{1}{2}\rho V^2 A(s))\hat{q}(s) = 0. \quad (2.29)$$

Méthode d'approximation par fractions rationnelles

Comme nous venons de le voir, les forces aérodynamiques instationnaires sont supposées données dans le domaine de Laplace par une fonction linéaire par rapport à $\hat{q}(s)$. La matrice $A(s)$ ($s = i\omega + \alpha$) peut être calculée pour les fréquences réduites $k = \omega \frac{l}{V}$ par la méthode des doublets, l étant la longueur de corde du profil. La fonction matricielle $s \mapsto \mathcal{A}(s)$ est ainsi connue pour un certain nombre de valeurs complexes :

$$s \in \left\{ \frac{i\omega_1 l}{V}, \dots, \frac{i\omega_M l}{V} \right\} \quad (2.30)$$

Pour effectuer la suite des calculs, et en particulier pour le calcul du gradient des fonctions objectifs, il est nécessaire d'étendre cette fonction à un domaine du plan complexe. À l'aide du théorème de prolongement analytique, il est alors possible d'approcher $A(s)$ par une fonction analytique sur un ouvert du plan complexe contenant l'axe imaginaire. Il existe plusieurs méthodes dans la littérature permettant d'approcher la matrice $A(s)$ par une fonction matricielle rationnelle [42, 43, 72, 78, 95]. En introduisant la variable complexe normalisée $\hat{s} = \frac{s l}{V}$, la matrice $A(s)$ est approchée par une fraction rationnelle :

$$\mathcal{A}(s) \approx A_0 + A_1 \hat{s} + A_2 (\hat{s})^2 + \sum_{p=1}^P \frac{N_p(\hat{s})}{D_p(\hat{s})} \quad (2.31)$$

avec A_0 , A_1 et A_2 des matrices, N_p des fonctions polynomiales matricielles réelles de degré 1 et D_p des fonctions polynomiales réelles de degré 2. Le problème de minimisation à résoudre est alors :

$$\min_{P, A_0, A_1, A_2, N_p, D_p} \sum_{j=1}^M \left\| \mathcal{A}(ik_j) - A_0 - A_1 ik_j - A_2 (ik_j)^2 - \sum_{p=1}^P \frac{N_p(ik_j)}{D_p(ik_j)} \right\|_{Mat_{\mathbb{C}}(n,n)}^2, \quad (2.32)$$

les réels $k_j = \omega_j \frac{l}{V}$ étant les fréquences réduites du système. Cette minimisation est réalisée par un algorithme de type moindres carrés couplé à une méthode de point fixe, la norme $\|\cdot\|_{Mat_{\mathbb{C}}(n,n)}$ étant la norme matricielle de Frobenius.

2.1.6 Vitesse critique

Dans le cadre de cette thèse où les calculs des forces aérodynamiques sont effectués sous une hypothèse de linéarité, l'étude de la stabilité s'effectue dans le domaine fréquentiel et nous allons nous intéresser à l'étude de la stabilité des solutions de l'équation de flottement :

$$(p^2M + K + \frac{1}{2}\rho V^2 A(p, V))X = 0, \quad p \in \mathbb{C}. \quad (2.33)$$

Pour faciliter la résolution de cette équation, nous avons fait l'hypothèse supplémentaire que l'amortissement est nul et que le système est conservatif. Cela est motivé par le fait que rajouter de l'amortissement améliore la stabilité du système, nous permettant ainsi de réaliser nos études de stabilité avec l'assurance que les prévisions resteront exactes même en présence d'amortissement. Dans l'équation, X est assimilé à un pseudo-vecteur propre à droite pour l'équation non linéaire. On le considère comme un "mode aéroélastique". On définit de même un pseudo-vecteur propre à gauche Y tel que :

$$Y(p^2M + K + \frac{1}{2}\rho V^2 A(p, V)) = 0, \quad p \in \mathbb{C}. \quad (2.34)$$

Pour chaque solution p de l'équation (2.33), il existe un pseudo-vecteur propre à gauche Y et un pseudo-vecteur propre à droite X associés à cette solution.

Dans ce paragraphe, nous allons présenter plus en détails le "flottement" qui est un phénomène d'instabilité pour la structure et caractérisé par le passage d'une solution de l'équation aéroélastique standard dans le demi-plan positif de \mathbb{C} .

Description du phénomène de flottement

Le flottement est un phénomène de ruine bien connu des aérodynamiciens, en particulier en aéroélasticité. Il s'agit d'un phénomène destructeur pour l'appareil, car il provoque une vibration explosive des ailes. Mathématiquement, cela correspond à l'existence d'une solution p de l'équation aéroélastique standard donc la partie réelle est positive. Cela signifie que le mouvement de l'aile est exponentiel croissant, et très rapidement, les valeurs de déplacement vont excéder les limites physiques de déformation, provoquant la ruine de la structure.

Cependant, cela ne peut arriver à faible vitesse, en particulier au sol, quand la vitesse est nulle. Les modes structuraux de base n'induisent pas un tel phénomène provoqué par un transfert d'énergie du fluide à la structure. C'est donc à haute vitesse que ce phénomène peut se produire. On appelle vitesse critique la vitesse minimale à laquelle ce phénomène se produit :

$$V_c = \min\{V > 0 \mid \exists p \text{ solution de l'équation aéroélastique, } \operatorname{Re}(p) > 0\}. \quad (2.35)$$

Remarque 2.1. Cas des essais en soufflerie : ce n'est plus le paramètre V qui joue sur la stabilité, car la vitesse du vent est constante, mais la pression $q = \frac{1}{2}\rho V^2$, où de manière équivalente, la densité volumique ρ .

2.1.7 Dérivation de l'équation aéroélastique standard

Par la suite, nous serons amenés à calculer le gradient de fonctions objectifs et en particulier à calculer les dérivées partielles des pseudo-valeurs propres de l'équation aéroélastique standard en fonction des grandeurs du système. Il peut s'agir, dans un premier

temps, de paramètres extérieurs tels que la pression ou la vitesse, ou de paramètres structuraux, comme la masse, les raideurs ou les épaisseurs de plaque.

Cas d'un paramètre structurel

La fonction $A(.,.)$ dépend de deux variables distinctes. On notera $\frac{\partial A}{\partial x_1}$ la dérivée partielle par rapport au premier argument, et $\frac{\partial A}{\partial x_2}$ la dérivée partielle par rapport au second argument. Nous considérons à présent un paramètre structurel α dont dépendent les matrices de l'équation aéroélastique réduite : $M(\alpha)$ et $K(\alpha)$. Alors, en faisant les hypothèses nécessaires de dérivabilité sur les termes de l'équation [35, 44, 83], nous pouvons dériver terme à terme l'équation (2.33) :

$$\begin{aligned} & \left(2\frac{\partial p}{\partial \alpha}pM + p^2\frac{\partial M}{\partial \alpha} + \frac{\partial K}{\partial \alpha} + \frac{1}{2}\rho V^2\frac{\partial p}{\partial \alpha}\frac{\partial A}{\partial x_1}(p, V) \right) X \\ & + \left(p^2M + K + \frac{1}{2}\rho V^2A(p, V) \right) \frac{\partial X}{\partial \alpha} = 0 \end{aligned} \quad (2.36)$$

En multipliant par le pseudo vecteur propre Y à gauche, et en regroupant les termes, on obtient successivement :

$$\left(2\frac{\partial p}{\partial \alpha}pM + p^2\frac{\partial M}{\partial \alpha} + \frac{\partial K}{\partial \alpha} + \frac{1}{2}\rho V^2\frac{\partial p}{\partial \alpha}\frac{\partial A}{\partial x_1}(p, V) \right) X = 0 \quad (2.37)$$

$$\frac{\partial p}{\partial \alpha} = -\frac{Y \left(p^2\frac{\partial M}{\partial \alpha} + \frac{\partial K}{\partial \alpha} \right) X}{Y \left(2pM + \frac{1}{2}\rho V^2\frac{\partial A}{\partial x_1}(p, V) \right) X} \quad (2.38)$$

Cas d'un paramètre aérodynamique

Les paramètres étudiés ici seront la vitesse V ou la densité volumique du fluide ρ . Dans le cas de la vitesse V , on obtient la formule suivante :

$$\frac{\partial p}{\partial V} = \frac{Y \left(\rho V A(p, V) + \frac{1}{2}\rho V^2\frac{\partial A}{\partial x_2}(p, V) \right) X}{Y \left(2pM + \frac{1}{2}\rho V^2\frac{\partial A}{\partial x_1}(p, V) \right) X} \quad (2.39)$$

Toutefois, dans les souffleries, la vitesse du flux est constante, et c'est la pression ou la densité volumique de l'air que l'on fait varier. Les forces aérodynamiques dépendent alors, non pas de la vitesse mais de la densité volumique. Cela aboutit à une formule légèrement différente :

$$\frac{\partial p}{\partial \rho} = \frac{Y \left(\frac{1}{2}V^2A(p, V) + \frac{1}{2}\rho V^2\frac{\partial A}{\partial x_2}(p, V) \right) X}{Y \left(2pM + \frac{1}{2}\rho V^2\frac{\partial A}{\partial x_1}(p, V) \right) X} \quad (2.40)$$

2.1.8 Formule du gradient du paramètre aérodynamique critique

Formulation du problème

On note β le paramètre aérodynamique auquel on s'intéresse : vitesse d'écoulement V ou densité volumique ρ . On suppose être dans une situation où les paramètres structuraux permettant l'optimisation, représentés par le vecteur x , sont associés à une vitesse critique β_c .

On suppose également que le flottement est provoqué par une unique solution p . On suppose que les solutions sont des fonctions continues et continûment dérivables des paramètres structuraux d'optimisation, ce qui fait que le mode associé au flottement sera le même si on conserve des paramètres structuraux très proches. On peut alors écrire, par continuité, que :

$$\beta_{crit} = \min\{\beta > 0 \mid \text{Re}(p(x, \beta)) = 0\} \quad (2.41)$$

au voisinage d'un jeu de paramètres donné (x^0, β_c^0) . En particulier, on peut écrire localement que :

$$\text{Re}(p(x, \beta_{crit})) = 0. \quad (2.42)$$

Théorème des fonctions implicites

On suppose alors que la solution p associée au mode aéroélastique X est une fonction différentielle du vecteur x et du paramètre β , et dont la dérivée partielle en β est non nulle. On peut alors appliquer le théorème des fonctions implicites à la fonction $\text{Re}(p)$.

Proposition 2.2. *Au voisinage de (x^0, β_c^0) , il existe une fonction φ telle que :*

$$\text{Re}(p(x, \beta_c)) = 0 \iff \beta_c = \varphi(x). \quad (2.43)$$

En outre, φ est différentiable au voisinage de β_c^0 , et on a la relation suivante :

$$\nabla_x \varphi(x^0) = -\frac{\nabla_x \text{Re}(p)}{\frac{\partial \text{Re}(p)}{\partial \beta}}(x^0, \beta_c^0), \quad (2.44)$$

Nous pouvons donc en déduire la dérivée partielle du paramètre aérodynamique critique par rapport à un paramètre α quelconque :

$$\frac{\partial \beta_c}{\partial \alpha}(\alpha_0) = -\frac{\frac{\partial \text{Re}(p)}{\partial \alpha}}{\frac{\partial \text{Re}(p)}{\partial \beta}}(\alpha, \beta_c) \quad (2.45)$$

L'expression détaillée de ce gradient dépend du choix du paramètre α . Selon sa nature, il faudra utiliser les expressions (2.38), (2.39), (2.40).

Démonstration. En supposant le premier résultat exact, nous pouvons écrire au voisinage du point (x^0, β_c^0) la relation suivante :

$$\text{Re}(p(x, \varphi(x))) = 0.$$

Toutes les fonctions ici sont différentiables, ce qui nous permet de dériver l'expression par rapport à x . En posant $g = \text{Re}(p)$, on obtient :

$$\nabla_x g(x, \varphi(x)) + \frac{\partial g}{\partial x_2}(x, \varphi(x)) \nabla \varphi(x) = 0. \quad (2.46)$$

Cela est vrai au voisinage de (x^0, β_c^0) , donc en particulier en ce point. Si on écrit la relation précédente en (x^0, β_c^0) , on aboutit à la formule de la dérivée de la vitesse critique en (x^0, β_c^0) , exprimée uniquement en fonction des grandeurs du système en dépit de son caractère implicite. \square

2.2 Introduction d'incertitudes dans le modèle

Nous allons maintenant introduire des incertitudes dans le modèle structurel. Mathématiquement, les effets des diverses méconnaissances présentes dans le modèle de la structure seront représentées par des variables aléatoires dont nous détaillerons plus tard les caractéristiques. Les incertitudes structurelles seront introduites directement dans le modèle à éléments finis, en considérant un certain nombre de paramètres structuraux comme incertains. Par exemple, il est possible d'introduire des incertitudes pour des densités de masse de certains éléments, ou encore des incertitudes sur les coefficients d'Young pour certains matériaux. Les matrices de masse et de raideur du modèle à éléments finis vont devenir des fonctions matricielles de ces paramètres incertains que l'on notera :

$$M(\epsilon) ; K(\mu) ; \epsilon = (\epsilon_1, \dots, \epsilon_{n_M}), \mu = (\mu_1, \dots, \mu_{n_K}). \quad (2.47)$$

Afin d'unifier par la suite les notations, nous noterons :

$$\xi = (\epsilon, \mu) \in \mathbb{R}^{n_M} \times \mathbb{R}^{n_K}, \quad (2.48)$$

le vecteur ξ englobant ainsi l'ensemble des paramètres incertains structuraux. Nous considérerons également les matrices de structure incertaines $M(\xi)$ et $K(\xi)$. Ces matrices vont garder leurs propriétés de symétrie et de positivité par construction.

Exemple 2.3. La matrice \mathbb{M} est construite à partir de l'opérateur de masse \mathcal{M} du système mécanique :

$$\langle \mathcal{M}v, \nu \rangle = \int_D \langle v(x), \nu(x) \rangle d\mu(x). \quad (2.49)$$

Supposons que l'incertitude sur la mesure de masse puisse être décrite par un ensemble d'incertitudes portant sur des sous-structures D_i , la structure D étant divisée en une partition :

$$D = \bigcup_{i=1}^{n_M} D_i. \quad (2.50)$$

L'opérateur de masse s'écrit alors :

$$\langle \mathcal{M}v, \nu \rangle = \sum_{i=1}^{n_M} \int_{D_i} \langle v(x), \nu(x) \rangle d\mu_i(x, \epsilon_i). \quad (2.51)$$

Projetée sur la base modale, cette relation s'écrit :

$$\langle \mathcal{M}\varphi_k, \varphi_l \rangle = \sum_{i=1}^{n_M} \int_{D_i} \langle \varphi_k(x), \varphi_l(x) \rangle d\mu_i(x, \epsilon_i). \quad (2.52)$$

Lorsque chaque sous-structure D_i n'englobe qu'un seul type de matériau, la densité volumique est constante sur chaque sous-structure, et la relation s'écrit :

$$\langle \mathcal{M}\varphi_k, \varphi_l \rangle = \sum_{i=1}^{n_M} \rho_i(\epsilon_i) \int_{D_i} \langle \varphi_k(x), \varphi_l(x) \rangle dx. \quad (2.53)$$

L'incertitude ne porte plus que sur les masses volumiques ρ_i . Cette expression se simplifie quand le modèle décrivant la structure est un type de modèle appelé "en arêtes de poisson" construit uniquement à partir de masses $(m_i)_{i=1, n_M}$ concentrées aux points (P_i) et d'éléments de type poutre les reliant. Dans ce cas,

$$d\mu_i(x, \epsilon_i) = m_i(\epsilon_i)\delta_x(P_i) \quad (2.54)$$

et

$$\langle \mathcal{M}\varphi_k, \varphi_l \rangle = \sum_{i=1}^{n_M} m_i(\epsilon_i) \langle \varphi_k(P_i), \varphi_l(P_i) \rangle. \quad (2.55)$$

L'incertitude porte directement sur la valeur de chaque masse m_i .

Modélisation des incertitudes

On définit l'espace probabilisé $(\Omega, \mathcal{F}, \mathbb{P})$ sur lequel seront construites toutes les variables aléatoires. Les grandeurs ϵ_j et μ_j sont généralement inconnues, à l'exception des bornes des valeurs qu'elles peuvent prendre et parfois de certaines caractéristiques de leur comportement. C'est pourquoi nous allons représenter ces grandeurs par des variables aléatoires. Les hypothèses de construction de ces variables sont les suivantes : pour tous les indices j et l , les variables ϵ_j et μ_l sont indépendantes, centrées et de loi :

$$\mathbb{P}_{\epsilon_j}(dx) = p_{\epsilon_j}(x)dx ; \quad \mathbb{P}_{\mu_l}(dx) = p_{\mu_l}(x)dx, \quad (2.56)$$

les fonctions de densité de probabilité p_{ϵ_j} et p_{μ_l} étant à support compact sur $[-\alpha_j, \alpha_j]$ et $[-\beta_l, \beta_l]$. Des conditions sont à vérifier sur ces densités afin que les matrices de masse et de raideur soient presque sûrement symétriques positives.

Remarque 2.4. Nous avons restreint le cadre d'étude en faisant l'hypothèse d'indépendance des variables aléatoires. Il peut exister des situations où certaines variables sont dépendantes, par exemple quand la masse est localement constante (répartition aléatoire du fret dans la soute d'un avion) ou quand les raideurs sont corrélées (usage du même matériau dans une zone étendue). Néanmoins, dans le cas de relation de dépendance linéaire, il est possible de se ramener par élimination au cas des variables indépendantes.

L'équation aéroélastique qui découle de cette introduction d'incertitudes s'écrit selon la forme suivante :

$$\left(p^2 M(\xi) + K(\xi) + \frac{1}{2} \rho V^2 A(p, V) \right) X = 0. \quad (2.57)$$

Les solutions (p_i, X_i) de cette équation deviennent donc des fonctions de $\xi : (p_i(\xi), X_i(\xi))$, c'est-à-dire qu'elles deviennent des variables aléatoires. Un des problèmes récurrents qui apparaît lorsqu'on modélise les incertitudes par le biais de variables aléatoires est le choix des lois de probabilité de ces variables. Deux cas de figure peuvent se présenter. Ou bien, l'industriel possède suffisamment de données pour se faire une idée du comportement des incertitudes et des lois associées, il suffit alors d'attribuer aux paramètres incertains la

loi établie par les ingénieurs [40]. Ou bien, on ne dispose que d'un nombre limité d'informations et on peut alors s'aider de théories classiques de type information de Fischer et maximum d'entropie [39]. Le but dans ce cas étant d'introduire la loi la plus "incertaine" au sens de la mesure de l'entropie. En règle générale, ce type d'approche est utilisé à partir de la connaissance du support de la loi et de quelques moments statistiques : moyenne, variance, etc.

Remarque 2.5. Les incertitudes portent sur des grandeurs mécaniques ou physiques. Elles sont donc par nature bornées, ce qui exclut l'utilisation de lois gaussiennes comme on le voit souvent dans la pratique.

Finalement, l'équation que nous allons étudier dans le reste de ce travail et qui sera à la base des problèmes d'optimisation rencontrés s'écrit :

$$\left(p^2 M(\xi(\omega)) + K(\xi(\omega)) + \frac{1}{2} \rho V^2 A(p, V) \right) X = 0. \quad (2.58)$$

Les solutions (p, X) de cette équation sont alors des variables aléatoires. Le problème est résolu une fois que la loi du couple (p, X) de l'équation est caractérisée. La complexité de la construction de la matrice A ainsi que la non-linéarité de l'équation (2.58) rendent irréalisable une caractérisation explicite de cette loi. On se tourne alors vers des approches de Monte-Carlo pour construire des estimateurs de grandeurs statistiques des solutions [70, 73, 75].

Chapitre 3

Algorithmes d'optimisation

3.1 Poser un problème d'optimisation

Un problème d'optimisation est un problème mathématique de recherche d'extremums (lorsqu'ils existent) d'une fonctionnelle à valeurs dans \mathbb{R} sur un espace de recherche $U^{ad} \subset \mathbb{R}^d$, contenant l'ensemble des solutions admissibles ou acceptables, et inclus dans l'espace de définition $U \subset \mathbb{R}^d$ de la fonctionnelle. On explicitera par la suite la nature de l'ensemble U selon la situation, qui est la plupart du temps confondu avec \mathbb{R}^d . Comme chercher le maximum d'une fonctionnelle f est équivalent à rechercher le minimum de la fonction opposée $-f$, on se contentera de considérer les problèmes de minimisation dans toute la suite de notre étude. En général, déterminer le point qui réalise cet optimum va de pair avec la détermination de la valeur de cet optimum. Nous nous limiterons aux problèmes statiques, sans influence du temps. Mathématiquement, cela s'exprime de la manière suivante :

$$\min_{x \in U^{ad}} f(x) \quad (3.1)$$

ou, si l'on recherche la valeur x pour laquelle la fonction f est minimale :

$$\operatorname{Argmin}_{x \in U^{ad}} f(x). \quad (3.2)$$

On peut être amené à étudier de tels problèmes auxquels des contraintes ont été ajoutées. Dans de tels cas, la solution doit vérifier ces contraintes, ce qui réduit l'espace des solutions admissibles. Dans certains cas, la formulation du problème sous-entend que l'ensemble U^{ad} prend en compte ces contraintes, généralement quand celles-ci sont linéaires et permettent une construction simple de l'espace restreint. Le plus souvent, U^{ad} est un sous-ensemble de \mathbb{U} défini par des bornes sur les coordonnées. Dans d'autres cas où les contraintes sont non triviales et ne permettent pas de construire analytiquement l'ensemble $U^c \subset U^{ad}$ des points vérifiant ces contraintes, on sépare explicitement l'espace des solutions admissibles et les fonctions contraintes qu'on rajoute par ailleurs. Le problème est alors posé de la façon suivante :

$$\left\{ \begin{array}{l} \min_{x \in U^{ad}} f(x) \\ \text{s.c.} \\ g(x) \leq 0 ; h(x) = 0 \end{array} \right. \quad (3.3)$$

où g et h sont des fonctions contraintes à valeurs de U dans \mathbb{R}^a et \mathbb{R}^b .

On dit que ce problème est bien posé s'il existe une solution, et que celle-ci est unique. Par exemple, si la fonction est non bornée, le minimum est non défini car il n'existe pas. Et même en cas d'existence d'une borne inférieure, celle-ci peut ne pas être un minimum, comme dans le cas de la fonction exponentielle dont la borne inférieure est 0 et n'est jamais atteinte. Dans d'autres cas, il se peut que le minimum soit atteint en plusieurs points, comme par exemple la fonction cosinus qui possède une infinité de points réalisant le minimum de la fonction. On dit alors que le problème est mal posé, car la notion même de solution est impropre.

Réciproquement, si une fonction est continue et que l'espace de définition est compact, on sait que cette fonction possède au moins un minimum et qu'elle l'atteint. Avant de commencer à chercher le minimum, il faut donc vérifier si ce minimum est bien défini, et s'il est unique ou non. Il peut exister plusieurs jeux d'hypothèses permettant de démontrer l'existence et l'unicité de la solution, ou au contraire, sa non-existence. Dans le cas où la solution n'est pas unique, il est parfois possible de caractériser le comportement de la fonction près de la borne inférieure.

Si l'espace de définition est discret, on parle alors de problèmes d'optimisation combinatoire pour lesquels des outils spécifiques ont été développés. Outils qui, on le verra plus tard, pourront être adaptés à l'optimisation continue.

Lorsque l'on s'intéresse à des problèmes d'optimisation issus du milieu industriel, les paramètres du système sont rarement tous connus avec exactitude. Comme cela a été abordé dans le paragraphe précédent, on peut alors introduire des variables aléatoires afin de modéliser cette incertitude. Un problème de définition se pose alors quand on considère des fonctions stochastiques pour la fonction objectif et/ou les fonctions contraintes. Il s'agit de préciser comment le paramètre ω modélisant l'aléatoire doit être pris en compte dans le problème d'optimisation :

$$\min_{x \in U^{ad}} \left\{ f(x, \xi(\omega)) \mid g(x, \xi(\omega)) \geq 0 \right\}. \quad (3.4)$$

Pour donner un sens à ce nouveau problème, il existe plusieurs approches qui correspondent à des techniques et des formulations spécifiques d'optimisation.

Optimisation en espérance :

$$\min_{x \in U^{ad}} \left\{ \mathbb{E} \left[f(x, \xi(\omega)) \right] \mid \mathbb{E} \left[g(x, \xi(\omega)) \right] \geq 0 \right\}. \quad (3.5)$$

Ce problème est le plus simple à étudier mathématiquement. On dispose de nombreux résultats d'existence et d'unicité sur ce type de problème, et sous les bonnes hypothèses, la ou les solutions sont en général simples et rapides à calculer. Cependant, d'un point de vue pratique, il n'est pas optimal, car ses solutions sont peu robustes. Une manière similaire de poser le problème en renforçant la robustesse est de rajouter une part de la variance de la fonction objectif :

$$\min_{x \in U^{ad}} \left\{ \mathbb{E} \left[f(x, \xi(\omega)) \right] + \lambda \text{Var} \left[f(x, \xi(\omega)) \right] \mid \mathbb{E} \left[g(x, \xi(\omega)) \right] \geq 0 \right\}. \quad (3.6)$$

En jouant sur la valeur de λ , on peut ainsi privilégier les solutions avec la variance la plus faible, ce qui donne des solutions plus résistantes aux fluctuations de la variable $\xi(\omega)$.

Optimisation dans le pire des cas :

$$\min_{x \in U^{ad}} \left\{ \mathbb{E} \left[f(x, \xi(\omega)) \right] \mid g(x, \xi(\omega)) \geq 0, \forall \omega \in \Omega \right\}. \quad (3.7)$$

Évidemment, dans ce contexte, les solutions exhibées sont toutes robustes, puisqu'il s'agit d'une optimisation "dans le pire des cas". Cependant, ce problème n'est quasiment jamais étudié. Les solutions sont en général très difficiles à calculer dès que les fonctions f et g deviennent non triviales, et elles n'existent pas forcément, à cause de la nature stochastique des fonctions contraintes.

Optimisation robuste :

$$\min_{x \in U^{ad}} \left\{ \mathbb{E} \left[f(x, \xi(\omega)) \right] \mid g(x, \xi(\omega)) \geq 0, \omega \in \mathbb{H} \right\} ; \mathbb{P}(\mathbb{H}) = p. \quad (3.8)$$

Dans cette situation, on restreint volontairement le domaine délimité par les contraintes en le remplaçant par un ensemble de contraintes linéaires permettant de rendre convexe l'espace de recherche \mathbb{H} . Le but est de parvenir à simplifier les calculs souvent complexes afin d'obtenir une solution réalisable en un temps satisfaisant. Cette méthode est la plus souvent employée dans le milieu industriel, car cela permet de prendre en compte la plupart des situations tout en permettant d'obtenir des solutions réalistes à l'aide de méthodes déterministes et avec des temps de calcul acceptables. Cependant, cette approche est adaptée principalement pour des problèmes linéaires ou avec de fortes hypothèses de convexité/concavité.

Optimisation selon des contraintes en probabilité :

$$\min_{x \in U^{ad}} \left\{ \mathbb{E} \left[f(x, \xi(\omega)) \right] \mid \mathbb{P} \left[g(x, \xi(\omega)) \geq 0 \right] > p \right\}. \quad (3.9)$$

Dans le cas présent, il s'agit d'optimiser une fonction en s'assurant que les contraintes sont vérifiées avec une certaine probabilité. En réglant le niveau de cette probabilité, on peut affiner à volonté la précision de la solution vis-à-vis des contraintes. Le problème majeur est de résoudre le problème théoriquement, car l'espace des solutions admissibles est en général inconnu ou non convexe.

Nous voyons par conséquent qu'il est nécessaire de séparer le traitement des problèmes d'optimisation déterministe, sans incertitude, et les problèmes d'optimisation stochastique, où les incertitudes sont représentées par des variables aléatoires. Pour cette dernière catégorie, les techniques de résolution utilisées sont le plus souvent des extensions au contexte probabiliste des outils employés dans le cadre déterministe. Deux grandes classes d'approches numériques peuvent être identifiées, basées sur la possibilité ou non de calculer le gradient de la fonction objectif.

Nous allons dans un premier temps passer en revue quelques méthodes parmi les plus classiques utilisées dans le cadre déterministe, en détaillant celles qui seront étendues au cadre stochastique.

3.2 Le cadre déterministe

Dans cette section, nous nous intéresserons donc au problème suivant, sans élément stochastique :

$$\min_{x \in U^{ad}} f(x) \quad (3.10)$$

3.2.1 Un aperçu des algorithmes d'optimisation dans le cadre déterministe avec gradient

Dans le cadre des fonctions scalaires définies de \mathbb{R} dans \mathbb{R} , les algorithmes les plus efficaces repose sur la méthode de Newton [20]. Elle a pour objet de rechercher les zéros d'une fonction scalaire f , et a pour prérequis que celle-ci soit de classe \mathcal{C}^1 et sa dérivée non nulle. Il s'agit d'un algorithme à itérations que l'on construit de la façon suivante, en inversant le développement de Taylor au premier degré en x_k :

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad (3.11)$$

Le point construit à chaque étape est en réalité l'abscisse de l'intersection de la tangente de la fonctionnelle en x_k et de l'axe des abscisses. Il a été démontré que la convergence de cette méthode se fait à une vitesse quadratique, c'est-à-dire qu'à partir d'un certain rang, l'erreur entre le point courant et la solution est le carré de l'erreur de l'étape précédente. Le nombre de décimales exactes est donc doublé à chaque étape [48].

Dans le cas où la fonction est deux fois continûment dérivable, des méthodes plus élaborées existent, comme la méthode de Halley, qui est l'application de la méthode de Newton à la fonction f/f' [89] :

$$x_{k+1} = x_k - \frac{2f(x_k)f'(x_k)}{2(f'(x_k))^2 - f(x_k)f''(x_k)}. \quad (3.12)$$

Cette méthode possède une vitesse de convergence cubique, comme la méthode de Schröder qui rajoute un terme à la formule de la méthode de Newton [37, 92] :

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} - \frac{f''(x_k)(f(x_k))^2}{2(f'(x_k))^3}. \quad (3.13)$$

Quand une fonction ne dispose pas de dérivée, il reste possible d'adapter la construction des itérations aux grandeurs disponibles, avec les méthodes de Quasi-Newton, bien que plus lentes que la méthode originale. Citons par exemple la méthode de la sécante, similaire aux différences finies :

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k). \quad (3.14)$$

Cette méthode converge également, mais avec une vitesse de convergence de l'ordre de $O(k^{-1,618})$ environ, contre $O(k^{-2})$ pour la méthode de Newton [9].

Le principe d'intersection entre l'axe des abscisses et de la tangente n'est pas applicable à partir de la dimension 2, c'est pourquoi cette méthode n'est utilisable qu'en dimension 1. En dimension supérieure, il existe de nombreux algorithmes d'optimisation, dont l'efficacité dépend du problème traité.

Parmi ces méthodes, on note une classe importante : les algorithmes de descente, qui construisent des suites itératives diminuant à chaque étape la valeur de la fonctionnelle, assurant que la dernière valeur de la suite ainsi construite sera la meilleure des solutions explorées. Le principe est, à chaque étape, de choisir une direction de descente, puis un coefficient multiplicateur à appliquer à cette direction, puis incrémenter le point courant dans cette direction. Le coefficient multiplicateur étant un scalaire, il est possible de coupler ces méthodes de descente avec la méthode de Newton, afin de choisir le coefficient multiplicateur minimisant la fonction objectif dans la direction de descente. En cas de fonctionnelle quadratique (par exemple, pour résoudre un système linéaire de grande taille), on peut également faire appel à des algorithmes de gradient plus sophistiqués, comme le gradient conjugué qui converge en un nombre fini d'étapes (ce nombre étant égal à la dimension de l'espace), ou le gradient biconjugué [31]. La direction de descente la plus efficace étant dirigée selon le gradient de la fonctionnelle, l'emploi de ces algorithmes est essentiellement conditionné par l'existence de cette quantité.

Nous allons maintenant nous intéresser plus en détails à l'algorithme du gradient, que nous étendrons ultérieurement au cadre stochastique.

3.2.2 Algorithme du gradient déterministe

La propriété de différentiabilité est une propriété fondamentale en optimisation. Cela ouvre les portes à plusieurs outils très puissants en mathématiques. Cela permet de savoir comment se comporte une fonction au voisinage d'un point, et même d'établir une expression approchée simplifiée sur ce même voisinage.

Rappelons tout d'abord quelques définitions et résultats essentiels et nécessaires pour décrire l'algorithme du gradient.

Quelques rappels d'analyse

Définition 3.2.1. Soit E un sous-espace vectoriel de \mathbb{R}^m , et soit f une fonction de E dans \mathbb{R}^n . Soit $x_0 \in E$. On dit que f est différentiable en x_0 si et seulement si, au voisinage de x_0 , il existe une matrice $A \in \mathcal{M}_{m,n}(\mathbb{R})$ telle que :

$$f(x_0 + x) = f(x_0) + Ax + o(\|x\|). \quad (3.15)$$

On dit qu'une fonction est différentiable sur E si et seulement si elle est différentiable en tout point de cet espace. On appelle fonction différentielle de f par rapport à x la fonction qui à x associe la matrice A .

Proposition 3.1. Soit une fonction f différentiable en $x \in E$, avec f_1, \dots, f_n ses fonctions-coordonnées. Alors, les coefficients de la matrice différentielle en x sont :

$$A_{i,j} = \frac{\partial f_i(x_1, \dots, x_{j-1}, y, x_{j+1}, \dots, x_n)}{\partial y}(x_j). \quad (3.16)$$

Dans le cas où $n = 1$ (f est à valeurs dans \mathbb{R}), on appelle ce vecteur le gradient de la fonction f .

Si la fonction différentielle associée à la fonction f est continue, on dit que la fonction f est de classe C^1 .

Définition 3.2.2. Une fonction $f : \mathbb{R}^m \rightarrow \mathbb{R}$ est dite convexe si et seulement si, pour tout points x, y et tout scalaire $\lambda \in [0, 1]$, on a :

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y). \quad (3.17)$$

Cela signifie que la courbe est en dessous de toutes ses cordes. On dit qu'une fonction f est strictement convexe si l'inégalité ci-dessus est stricte dès que $x \neq y$ et $\lambda \notin \{0, 1\}$.

Définition 3.2.3. Une fonction f est dite coercive si

$$\lim_{\substack{\|x\| \rightarrow +\infty \\ x \in E}} f(x) = +\infty. \quad (3.18)$$

Définition 3.2.4. Soit $K > 0$. Une fonction f est dite K -lipschitzienne si, pour tout $(x, y) \in E$, on a l'inégalité suivante :

$$|f(x) - f(y)| \leq K \|x - y\|. \quad (3.19)$$

Proposition 3.2. *En dimension finie, une fonction continue et coercive possède un minimum. De plus, celui-ci est unique si la fonction est strictement convexe.*

Formulation de l'algorithme du gradient

On suppose que la fonction $f : \mathbb{R}^m \rightarrow \mathbb{R}$ est différentiable, coercive et strictement convexe. Soit la suite de réels positifs $(\epsilon_k)_{k \in \mathbb{N}}$ vérifiant les propriétés suivantes :

$$\sum_{i=0}^{+\infty} \epsilon_k = +\infty ; \quad \sum_{i=0}^{+\infty} \epsilon_k^2 < +\infty. \quad (3.20)$$

À x_0 donné, on construit la suite (x_k) :

$$x_{k+1} = x_k - \epsilon_k \nabla_x f(x_k), \forall k \geq 0. \quad (3.21)$$

Alors, la suite (x_k) converge vers le minimum de la fonction f .

Il existe de nombreuses variantes optimisées de cet algorithme qui, pour certaines, peuvent s'étendre au cas stochastique. Comme nous l'avons signalé précédemment, on dispose de versions très performantes dans le cadre linéaire, par exemple l'algorithme du gradient conjugué qui converge vers la solution en un nombre fini d'étapes (ce nombre étant majoré par la dimension de l'espace), ou encore l'algorithme du gradient biconjugué. Une méthode classique est de préconditionner la direction de descente en multipliant le gradient par une matrice bien choisie. Dans le cas où cette matrice est l'inverse de la matrice hessienne, il s'agit de l'algorithme de Newton-Raphson en dimension quelconque [64] :

$$x_{k+1} = x_k - \epsilon_k \left(\nabla_x^2 f(x_k) \right)^{-1} \nabla_x f(x_k). \quad (3.22)$$

Un autre type de problème consiste à considérer une classe de fonctions et de réaliser une régression non linéaire des moindres carrés. Le problème consiste à ajuster la relation $y = f(p, x)$ au plus près et se pose sous la forme suivante :

$$\min_p h(p) = \sum_{i=1}^m (Y_i - f(p, X_i))^2 \quad (3.23)$$

avec les familles $f(\cdot, \cdot)$, (X_1, \dots, X_m) et (Y_1, \dots, Y_m) données. On voit donc qu'on souhaite minimiser la somme des carrés des erreurs entre les données du couple (X, Y) et la famille de fonctions $f(p, \cdot)$. Plusieurs techniques reposant sur le principe de minimisation de la somme des carrés de l'erreur ont été développées, telles que Gauss-Newton, Levenberg-Marquardt, quasi-Newton ou encore Gauss-Seidel. L'algorithme de Levenberg-Marquardt est une amélioration de la méthode classique de Gauss-Newton :

$$p_{k+1} = p_k - (\nabla_p f(p_k, X) \nabla_p f(p_k, X)^t - \lambda I_m) \nabla_p f(p_k, X)^t (Y - f(p_k, X)), \quad (3.24)$$

avec λ un paramètre adaptable pour ajuster les facteurs d'échelle. Toutes ces approches numériques sont développées dans le livre de [58]. Davantage de détails sont donnés dans les références [10, 21, 22, 32, 33, 51, 55, 86].

Un autre critère souvent considéré est la convexité de la fonctionnelle, ainsi que celle des fonctions contraintes délimitant l'espace de solutions admissibles. Dans le cas de fonctions linéaires, voire convexes/concaves, la méthode du point intérieur consiste à approcher la solution par l'intérieur de l'espace admissible. De cette technique découle la méthode de l'ellipsoïde et l'algorithme de Karmarkar [2, 41]. Le principe est de construire une suite telle qu'à chaque étape, on minimise la fonction objectif sur une sphère inscrite dans le domaine admissible et dont le centre est le point courant, en faisant une transformation linéaire de cette sphère pour simplifier les calculs de minimum local.

Nous achevons ici notre récapitulatif rapide des méthodes d'optimisation déterministe sans contraintes afin de considérer maintenant le cas des problèmes avec contraintes.

3.2.3 Formulation des problèmes d'optimisation déterministe sous contraintes

Dans certains cas, les contraintes ne sont pas des fonctions triviales, et le domaine des solutions admissibles est alors impossible à expliciter à l'aide de fonctions simples. A fortiori, on ne peut projeter sur l'espace des solutions admissibles, puisqu'on ne connaît pas son bord analytiquement. On sépare alors clairement les contraintes triviales qui sont représentées par l'espace admissible, et les contraintes non triviales qui sont traitées à part :

$$\begin{cases} \min_{x \in U^{ad}} f(x) \\ \text{s.c.} \\ g_i(x) \leq 0, h_j(x) = 0; 1 \leq i \leq m, 1 \leq j \leq l. \end{cases} \quad (3.25)$$

Pour résoudre un tel problème d'optimisation sous contraintes, on fait généralement appel à la théorie duale en introduisant les coefficients multiplicateurs de Kuhn-Tucker associés à la fonction :

$$L(x, \lambda) = f(x) + \sum_{i=1}^m \lambda_i g_i(x) + \sum_{j=1}^l \mu_j h_j(x), \quad (3.26)$$

avec $\lambda = (\lambda_1, \dots, \lambda_m) \in \mathbb{R}_+^m$ et $\mu = (\mu_1, \dots, \mu_l) \in \mathbb{R}_+^l$. Si la fonctionnelle et les contraintes sont toutes convexes, les conditions d'optimalité de Karush-Kuhn-Tucker assurent que la solution du problème est un point-selle de cette fonction [30, 49]. x^* est alors solution du problème d'optimisation si et seulement s'il existe deux vecteurs $\lambda^* \in \mathbb{R}_+^m$ et $\mu^* \in \mathbb{R}_+^l$ tels que :

$$\begin{cases} \nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) + \sum_{j=1}^l \mu_j^* \nabla h_j(x^*) = 0 \\ \lambda_i^* g_i(x^*) = 0, \forall i = 1, m. \end{cases} \quad (3.27)$$

On remarque que si $l = 0$, il n'y a pas de contraintes d'égalité, les conditions de KKT deviennent les conditions classiques de Lagrange, les coefficients de KKT sont alors les coefficients de Lagrange et la fonction associée s'appelle le lagrangien. Pour obtenir la solution de ce problème, des algorithmes itératifs convergeant vers le point-selle ont été développés, comme l'algorithme de Arrow-Hurwicz, qui sera étendu au cadre stochastique pour prendre en compte des contraintes en probabilité.

Algorithme de Arrow-Hurwicz

L'algorithme de Arrow-Hurwicz est un algorithme déterministe qui permet de prendre en compte des contraintes non linéaires et dont il existe une version stochastique qu'on détaillera dans le prochain chapitre [5].

Soit le problème d'optimisation déterministe sous contraintes suivant (sans contrainte en égalité et ne comportant qu'une seule contrainte d'inégalité) :

$$\begin{cases} \min_{x \in U^{ad}} f(x) \\ \text{s.c.} \\ g(x) \leq 0. \end{cases} \quad (3.28)$$

Le lagrangien d'un tel problème s'écrit sous la forme :

$$L(x, \lambda) = f(x) + \lambda g(x). \quad (3.29)$$

Sous les conditions de Kuhn-Tucker, on sait que la solution du problème d'optimisation est un point-selle (x^*, λ^*) du lagrangien associé. L'algorithme de Arrow-Hurwicz permet de construire une suite convergeant vers ce point-selle. Celui-ci se construit de la façon suivante :

$$x_{k+1} = \Pi_{U^{ad}} \left(x_k - \epsilon_k (\nabla_x f(x_k) + \nabla_x g(x_k) \lambda_k) \right) \quad (3.30)$$

$$\lambda_{k+1} = \Pi_{\mathbb{R}^+} \left(\lambda_k + \rho_k g(x_{k+1}) \right). \quad (3.31)$$

Théorème 3.3. *Sous les conditions de Kuhn-Tucker, le point x^* est la solution du problème (3.28) si et seulement s'il existe un réel λ^* tel que le point (x^*, λ^*) est un point-selle du lagrangien $L(x, \lambda)$.*

De plus, si les suites (ϵ_k) et (ρ_k) respectent les propriétés suivantes :

$$\sum_{k=0}^{+\infty} \rho_k = +\infty ; \sum_{k=0}^{+\infty} \epsilon_k = +\infty ; \sum_{k=0}^{+\infty} \rho_k^2 < +\infty ; \sum_{k=0}^{+\infty} \epsilon_k^2 < +\infty, \quad (3.32)$$

alors la suite (x_k) de l'algorithme de Arrow-Hurwicz converge vers le point x^ .*

Démonstration. Voir l'article [76]. □

Méthodes avec calcul numérique du gradient

Quand aucune formule analytique ne permet d'obtenir le gradient d'une fonction objectif ou contrainte, les logiciels employés dans l'industrie peuvent permettre d'obtenir une valeur numérique, par une approche similaire aux différences finies par exemple. Nous n'avons pas étudié ces stratégies numériques, souvent délicates à mettre en œuvre.

3.2.4 Autres algorithmes : les métaheuristiques

Nous venons de voir que la connaissance du gradient de la fonction objectif permettait d'établir des algorithmes convergeant vers la solution du problème d'optimisation, son existence découlant de propriétés de convexité de la fonction objectif et des contraintes. En l'absence d'expression analytique du gradient, et/ou de propriétés de convexité, on peut néanmoins chercher à obtenir une solution réalisable, qui ne sera pas la solution optimale ou exacte dans la plupart des cas. Il existe alors des méthodes approximatives globales, permettant d'obtenir rapidement une telle solution : les métaheuristiques. Ces méthodes sont généralement basées sur des approches aléatoires issues de l'observation du monde du vivant. Elles sont surtout employées en optimisation combinatoire, qui nécessite de calculer en tout point la fonction objectif pour s'assurer de l'optimalité d'une solution, ce qui s'avère généralement très coûteux dès que le nombre de points est important. Ces algorithmes ont été conçus, non pas pour déterminer la solution optimale car sans calculer point par point la fonction objectif, c'est en général impossible, mais pour apporter une solution acceptable au problème posé.

Les métaheuristiques sont généralement des algorithmes à population, en ce sens qu'ils font évoluer un ensemble d'individus qui interagissent entre eux pour produire la génération suivante, et sont plus particulièrement adaptés à l'optimisation discrète. Les algorithmes génétiques font partie de la famille des algorithmes évolutionnaires et sont fréquemment utilisés. Le principe repose sur la reproduction et la mutation des espèces dans la nature avec plusieurs étapes : sélection des parents, croisement, mutation et sélection naturelle des meilleurs éléments. À chaque itération, il y a 4 étapes :

- sélection : sélectionner les parents par couple dans la population ;
- croisement : produire des enfants en échangeant certaines coordonnées des parents de façon aléatoire ;
- mutation : choisir aléatoirement des coordonnées à modifier sur les enfants ;
- sélection : ne conserver qu'un certain nombre d'individus dans la population enfants+parents.

Les règles de sélection, de croisement et de mutation peuvent être adaptées selon la nature du problème, ce qui permet d'adapter l'algorithme génétique au problème étudié, qu'il soit binaire, combinatoire ou continu. Par exemple, la sélection peut se faire en fonction de l'évaluation de la fonction objectif ou être aléatoire. Ou bien encore, pour le croisement, on tire aléatoirement un nombre, et on échange un nombre équivalent de coordonnées consécutives ou non des deux parents.

Plus récemment, des algorithmes basés sur l'observation du monde animal ont vu le jour à la fin des années 90 : les algorithmes de colonies de fourmis, plutôt dédiés à l'optimisation combinatoire, et l'optimisation par essais particuliers, basée sur les vols d'oiseaux ou d'insectes qui échangent des informations localement. Ces approches sont souvent très

efficaces pour certains problèmes, mais il n'existe pas d'extension au cadre probabiliste.

Parmi les différentes métaheuristiques, il en est une qui se distingue des autres : le recuit simulé, que nous allons décrire plus en détails dans la suite, avant d'étudier une extension au cadre stochastique.

3.3 Le recuit simulé

Parmi les méthodes d'optimisation globale ne reposant pas sur la connaissance du gradient de la fonction objectif, le recuit simulé est l'une des plus connues. Elle présente plusieurs avantages comparée aux autres méthodes basées sur des métaheuristiques. Premièrement, cette méthode ne manipule qu'une suite numérique et pas tout un groupe de points comme le font la plupart des autres métaheuristiques qui sont, quant à elles, généralement basées sur l'évolution d'une famille de solutions. Deuxièmement, le recuit simulé possède une preuve de convergence vers la solution du problème, contrairement à la majorité des autres métaheuristiques pour lesquelles aucun résultat de convergence n'existe¹. Sous les hypothèses appropriées, on peut même obtenir un ordre de grandeur de la vitesse de convergence. La troisième raison, et non des moindres, est que cet algorithme a été étudié et adapté pour un problème de nature stochastique, chose qui n'a pas pu être faite pour la plupart des autres approches. Toutefois, cette adaptation est plus compliquée à mettre en place que la version déterministe, et nécessite de nombreux ajustements. Nous verrons par la suite quelles différences cela implique.

La méthode du recuit simulé est basée sur les propriétés de chaînes de Markov qui seront utilisées pour identifier les ensembles $\text{Argmin } f$. On sait qu'une chaîne de Markov stationnaire va converger vers une variable aléatoire dont la loi est la loi invariante de la chaîne. Si on est capable de construire une telle chaîne dont la loi stationnaire est supportée par l'ensemble $\text{Argmin } f$, alors il suffit de simuler cette chaîne pour converger vers un point de l'ensemble $\text{Argmin } f$. Ceci est le principe de la méthode que nous allons décrire ici. Nous donnons en annexe les définitions et résultats nécessaires sur les chaînes de Markov utilisées ici.

3.3.1 Méthode de Hasting-Metropolis

Le premier ingrédient de la méthode du recuit simulé est la construction d'une chaîne de Markov particulière. Étant donné une mesure π définie sur un ensemble fini \mathcal{S} , nous allons exhiber une chaîne de Markov dont la mesure stationnaire est π . Pour la construire, il suffira de préciser une matrice de transition P adéquate.

Construisons à présent la matrice P sur l'espace de points \mathcal{S} . Tout d'abord, pour chaque point $i \in \mathcal{S}$, on définit un voisinage $\mathcal{N}(i) \subset \mathcal{S}$, c'est-à-dire l'ensemble des points qui seront considérés comme voisins du point i . Ces voisinages peuvent être choisis librement. La seule contrainte est de rendre l'ensemble connexe, c'est-à-dire que pour tout couple de points (i, j) dans \mathcal{S} , il existe une suite finie de points allant de i à j et dont chaque point est dans le voisinage du précédent. On pose alors

$$\lambda(i) = \frac{\pi(i)}{d(i)}, \quad (3.33)$$

1. Des résultats de convergence existent néanmoins pour les algorithmes génétiques sous certaines conditions [16].

où $d(i)$ est le cardinal de l'ensemble $\mathcal{N}(i)$, et on construit la matrice de transition P :

$$P(i, j) = \begin{cases} \frac{1}{d(i)} \min \left\{ 1, \frac{\lambda(i)}{\lambda(j)} \right\} & \text{si } j \in \mathcal{N}(i), \\ 1 - \sum_{k \in \mathcal{N}(i)} P(i, k) & \text{si } i = j, \\ 0 & \text{sinon.} \end{cases} \quad (3.34)$$

On démontre facilement que la mesure π est invariante pour cette matrice de transition : $\pi = \pi P$. Il est alors simple de simuler cette chaîne de Markov. À chaque étape, on effectue un tirage aléatoire uniforme sur l'ensemble des points j du voisinage du point courant i . Une fois ce tirage effectué, à la différence de la marche aléatoire standard, on peut soit se déplacer sur le point candidat j , soit rester sur le point courant i . On accepte alors le point j avec une probabilité de $\min \left\{ 1, \frac{\lambda(i)}{\lambda(j)} \right\}$, et on reste au point i avec une probabilité de $1 - \min \left\{ 1, \frac{\lambda(i)}{\lambda(j)} \right\}$.

Pour un point candidat j donné, les valeurs $\lambda(i)$ et $\lambda(j)$ sont comparées. Si $\lambda(j) \geq \lambda(i)$, alors le nouveau point j est accepté. Si $\lambda(j) < \lambda(i)$, le nouveau point j est accepté avec une probabilité de $\frac{\lambda(j)}{\lambda(i)}$.

Le point fort de cet algorithme est qu'il ne nécessite que de connaître la distribution π (ou, de façon équivalente, la fonction λ) à une constante multiplicative près, ainsi que les voisinages de chaque point de l'ensemble.

3.3.2 Mesure de Boltzmann-Gibbs

Le second ingrédient de la méthode du recuit simulé est la mesure de Boltzmann construite à partir de la fonction objectif qu'on souhaite minimiser. Cette mesure jouera le rôle de mesure stationnaire de la chaîne de Markov. On définit donc la mesure de Boltzmann-Gibbs associée à la fonction objectif f , parfois appelée fonction énergie :

$$\mu_T(x) = \frac{1}{Z_T} \exp \left(-\frac{f(x)}{k_B T} \right). \quad (3.35)$$

Ici, T est un paramètre positif appelé communément "température", k_B une constante positive et Z_T une constante de normalisation. L'un des principaux intérêts de cette mesure est que son comportement global varie selon que la température T est proche de 0 ou non. Dans les hautes températures, la mesure est très proche de la mesure uniforme sur l'espace \mathcal{S} tout entier et ne favorise aucun point de l'espace. Réciproquement, quand T est proche de 0, la mesure de Boltzmann-Gibbs se comporte de manière similaire à la mesure uniforme sur les états d'énergie minimale, c'est-à-dire comme la mesure uniforme sur l'ensemble $\text{Argmin } f$.

On peut donc appliquer la méthode de Hasting-Metropolis en prenant comme mesure stationnaire π une mesure de Boltzmann-Gibbs associée à la fonction f .

3.3.3 Le recuit simulé dans le cas discret

L'idée du principe du recuit simulé tire son origine de la métallurgie et des techniques de refroidissement des métaux. Dans certains cas, on a pu observer qu'après la période de refroidissement, le métal ne se solidifie pas dans sa configuration atomique la plus stable. Sa vitesse de refroidissement naturelle est trop importante, et le métal se fige dans une configuration intermédiaire avant d'avoir atteint sa configuration optimale. Pour empêcher ce phénomène, les métallurgistes ont appris à contrôler la vitesse de refroidissement afin de laisser le métal atteindre son état atomique d'énergie minimale.

À une température donnée, Boltzmann a démontré que la répartition des atomes de métal se comporte selon un certain type de distributions, qu'on appelle à présent distributions de Boltzmann. Les mesures de Boltzmann que nous avons vues plus haut sont les versions mathématiques de ces distributions physiques [56].

Le principe du recuit simulé est d'appliquer simultanément la méthode de Hastings-Metropolis à la mesure de Boltzmann associée à la fonction que l'on souhaite minimiser et de réduire le paramètre température au cours des itérations, par analogie avec le refroidissement des métaux. Le but de l'algorithme est d'explorer l'espace des solutions E afin d'exhiber une solution satisfaisante du problème suivant :

$$\min_{x \in S} f(x) \quad (3.36)$$

L'algorithme du recuit simulé se présente alors de la façon condensée suivante :

1. Choisir un point de départ x_0 et une température de départ T_0 .
2. Effectuer la boucle suivante.
 - (a) À x_k fixé, choisir un point x_t dans le voisinage $\mathcal{N}(x_k)$.
 - (b) Effectuer un tirage d'une loi uniforme U sur $[0,1]$, indépendamment des précédents tirages.
 - (c) Si $U < \exp(\frac{f(x_k) - f(x_t)}{k_B T_m})$, alors le point est accepté : $x_{k+1} = x_t$.
 - (d) Sinon, $x_{k+1} = x_k$.
 - (e) Si un critère de convergence est vérifiée, alors passer à l'étape suivante. Sinon, reprendre à la deuxième étape.
3. Diminuer la température ($T_m \rightarrow T_{m+1}$) et reprendre à la deuxième étape.

On note qu'à l'étape (c), si $f(x_t) \leq f(x_k)$, l'exponentielle est plus grande que 1, et le nouveau point est automatiquement accepté.

Le principe de l'algorithme consiste, à une température fixée, à laisser la chaîne de Markov se stabiliser dans une configuration proche de la mesure de Boltzmann de la température associée, puis de diminuer la température et recommencer l'étape précédente. Ainsi, de proche en proche, la chaîne de Markov va favoriser les états de faible énergie, permettant une convergence vers la mesure uniforme sur les états d'énergie minimale, c'est-à-dire l'ensemble $\text{Argmin } f$. On peut prouver que selon le choix du schéma de température (par exemple, le schéma de Hajek : $T_i \geq \frac{K}{\ln i}$), cet algorithme converge presque sûrement vers un point de l'ensemble $\text{Argmin } f$ [1].

Généralisation à un espace continu

Cet algorithme a été développé historiquement pour des problèmes discrets. Il est toutefois possible de l'adapter en version continue, en modifiant la manière de choisir le nouveau point à tester pour l'adapter à un espace continu. Pour ce faire, il suffit de définir le voisinage de tout point par un espace continu autour de ce point. Le choix des points de la chaîne s'effectue alors selon un tirage uniforme sur ce voisinage. Le reste de l'algorithme est identique au cas discret.

Il existe plusieurs variantes de cet algorithme appliqué à un espace continu, dans le but d'accélérer la recherche de l'optimum. On peut citer par exemple le Fast Simulated Annealing [94] ou encore le Very Fast Simulated Reannealing [38]. Ces méthodes ne possèdent pas de démonstration de convergence, mais en pratique, elles possèdent de bons comportements et une vitesse de convergence accrue par rapport à la méthode suivant les paramètres permettant une convergence théorique.

3.4 Le cadre stochastique

Nous abordons maintenant le cœur du sujet en considérant l'optimisation de fonctions objectifs aléatoires. Dans un premier temps, nous allons nous intéresser à la formulation en espérance du problème suivant :

$$\underset{x \in U^{ad}}{\text{Argmin}} \mathbb{E}[f(x, \xi(\omega))]. \quad (3.37)$$

Nous supposons dans ce paragraphe que les contraintes puissent être incluses dans l'espace admissible U^{ad} . Plus précisément, soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace de probabilité sur lequel sont définies les variables aléatoires. On suppose que $\xi(\omega)$ est une variable aléatoire de loi μ , à valeurs dans \mathbb{R}^p , et que $f : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}$ est une fonction mesurable en son deuxième argument. Une des approches numériques permettant de résoudre le problème (3.37) est basée sur l'algorithme de Robbins-Monro que nous allons détailler dans le prochain paragraphe.

3.4.1 Schéma de Robbins-Monro

Robbins et Monro [77] ont été les premiers à proposer au début des années 50 un algorithme d'approximation stochastique [46, 98]. Plus précisément, il permet d'approcher la solution d'une équation de la forme :

$$\mathbb{E}[f(x, \xi(\omega))] = a. \quad (3.38)$$

Cet algorithme s'écrit :

1. Étape 0 : choisir $x_0 \in \mathbb{R}^n$.
2. Étape $k+1$: effectuer un tirage $\xi(\omega_{k+1})$ selon la loi μ indépendamment des précédents tirages et calculer :

$$x_{k+1} = x_k + \rho_k (f(x_k, \xi(\omega_{k+1})) - a),$$

avec $(\rho_k)_k$ une suite de réels positifs. La démonstration de la convergence de l'algorithme de Robbins-Monro repose sur des considérations techniques liées aux propriétés de martingale du processus discret $(\xi_k(\omega))_k$ construit à partir des réalisations de la variable aléatoire $\xi(\omega)$ modélisant l'incertitude. Elle est détaillée dans l'ouvrage [26].

Théorème 3.4. On note, $\forall k \in \mathbb{N}$, la filtration associée $\mathcal{F}_k = \sigma(\xi_1, \dots, \xi_k)$. Sous les hypothèses suivantes :

– il existe $x^* \in \mathbb{R}^n$ tel que

$$\mathbb{E}[f(x^*, \xi(\omega))] = a \quad (3.39)$$

et

$$\langle x^* - x, \mathbb{E}[f(x, \xi(\omega))] \rangle > 0, \forall x \in \mathbb{R}^n \setminus \{x^*\}, \quad (3.40)$$

– il existe une constante $K \geq 0$ telle que :

$$\forall k \in \mathbb{N}, \mathbb{E} \left[\|f(x_k, \xi_{k+1})\|^2 | \mathcal{F}_k \right] \leq K(1 + \|x_k\|^2),$$

– la série $\sum \rho_k$ est divergente mais la série de son carré est convergente :

$$\sum_{k \in \mathbb{N}} \rho_k = +\infty ; \quad \sum_{k \in \mathbb{N}} (\rho_k)^2 < +\infty,$$

la suite (x_k) engendrée par l'algorithme de Robbins-Monro converge presque sûrement vers x^* .

Exemple 1. Un cadre d'application classique où les hypothèses du théorème précédent sont vérifiées est le cas où la fonction aléatoire f est une fonction déterministe perturbée par un bruit gaussien :

$$f(x, \xi(\omega)) = \beta(x) + \xi(\omega), \quad (3.41)$$

β étant une fonction strictement monotone. On a alors $\xi(\omega) \sim \mathcal{N}(0, \sigma)$. De plus, on a les propriétés suivantes :

$$\mathbb{E}[\xi_k | \mathcal{F}_k] = 0 \quad (3.42)$$

$$\mathbb{E}[\xi_k^2 | \mathcal{F}_k] = \sigma^2. \quad (3.43)$$

On vérifie facilement que la troisième hypothèse du théorème est vérifiée :

$$\mathbb{E}[f(x_k, \xi(\omega_{k+1}))^2 | \mathcal{F}_k] = \alpha(x_k)^2 + 2\mathbb{E}[\alpha(x_k)\xi(\omega_{k+1})] + \mathbb{E}[\xi(\omega_{k+1})^2] = \alpha(x_k)^2 + 2\sigma^2. \quad (3.44)$$

L'algorithme converge alors dès que la suite (ρ_k) vérifie la dernière hypothèse du théorème.

3.4.2 Algorithme du gradient stochastique

Nous nous intéressons maintenant au cas stochastique, où l'on cherche à minimiser l'espérance d'une fonction aléatoire :

$$\underset{x \in U^{ad}}{\text{Argmin}} \mathbb{E}[f(x, \xi(\omega))]. \quad (3.45)$$

Pour cela, on se place toujours sur l'espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ et on considère une variable aléatoire $\xi(\cdot)$ définie sur Ω à valeurs dans \mathbb{R}^p , et μ la loi de probabilité de $\xi(\omega)$.

On se place sur \mathbb{R}^n , dont on note le produit scalaire et la norme associés respectivement $\langle \cdot, \cdot \rangle$ et $\|\cdot\|$. On considère un espace admissible convexe fermé non vide $U^{ad} \subset \mathbb{R}^n$ et une fonction $f : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \overline{\mathbb{R}}$. On souhaite étudier le problème suivant :

$$\underset{x \in U^{ad}}{\text{Argmin}} F(x) \quad (3.46)$$

avec :

$$F(x) = \mathbb{E}[f(x, \xi(\omega))] \quad (3.47)$$

$$= \int_{\Omega} f(x, \xi(\omega)) d\mathbb{P}(\omega) \quad (3.48)$$

$$= \int_{\mathbb{R}^p} f(x, w) d\mu(w). \quad (3.49)$$

On suppose dans notre situation que l'on n'a accès qu'à la fonction $f(x, \cdot)$ et à son gradient $\nabla_x f$ par rapport à la variable x . La solution "naïve" serait de calculer l'espérance à chaque étape de calcul, ce qui reviendrait à éliminer la partie stochastique du problème. Cependant, on ne connaît généralement pas la loi de la variable aléatoire $f(x, \xi(\omega))$. On peut certes approcher cette quantité par simulation des paramètres aléatoires. Cela conduit cependant à des temps de calculs très coûteux. On verra par la suite que cette approche doit cependant être retenue dans le cas où le gradient de la fonction objectif ne peut pas être calculé.

L'algorithme du gradient stochastique ne nécessite pas de faire une estimation de Monte-Carlo à chaque itération en construisant un estimateur du gradient à l'aide des itérations de l'algorithme. Le problème peut se résoudre par l'algorithme de Robbins-Monro si la fonction est différentiable. En effet :

$$x^* = \underset{x \in U^{ad}}{\text{Argmin}} \mathbb{E} [f(x, \xi(\omega))] \Rightarrow \mathbb{E} [\nabla_x f(x^*, \xi(\omega))] = 0. \quad (3.50)$$

L'algorithme de Robbins-Monro s'écrit dans ce cas :

Algorithme du gradient stochastique

1. On choisit une valeur initiale $x_0 \in U^{ad}$, ainsi qu'une suite de réels (ϵ_k) (que l'on explicitera par la suite).
2. À chaque itération, on effectue un tirage $\xi(\omega_{k+1})$ de la variable aléatoire ξ suivant sa loi, indépendamment des précédents tirages $(\xi(\omega_1), \dots, \xi(\omega_k))$.
3. On calcule le gradient de la fonction $x \rightarrow f(x, \xi(\omega_{k+1}))$ au point x_k , et on construit le point suivant de la suite :

$$x_{k+1} = \text{proj}_{U^{ad}} (x_k - \epsilon_k \nabla_x f(x_k, \xi(\omega_{k+1}))). \quad (3.51)$$

4. On vérifie un critère de convergence, et s'il n'est pas vérifié, on retourne à l'étape 2 en incrémentant k .

Pour qu'un tel algorithme soit applicable, il faut nécessairement que la fonction $f(\cdot, \xi(\omega))$ soit différentiable pour presque tout $\omega \in \Omega$. Le but de l'algorithme est d'atteindre ou d'approcher le plus possible l'ensemble $U^\# = \text{Argmin} \{ \mathbb{E} [f(x, \xi(\omega))] \mid x \in U^{ad} \}$.

Théorème 3.5. *Sous les hypothèses suivantes :*

1. *La variable aléatoire $f(x, \xi(\omega)) : \Omega \rightarrow \mathbb{R}$ est mesurable et d'espérance finie pour tout $x \in U^{ad}$.*
2. *La fonction $f(\cdot, \xi(\omega)) : \mathbb{R}^n \rightarrow \mathbb{R}$ est convexe, semi-continue inférieurement, propre et différentiable pour presque tout $\omega \in \Omega$.*

3. Le gradient de la fonction f par rapport à x est borné uniformément en (x, ω) .
4. Le problème (3.45) possède un ensemble $U^\#$ non vide de solutions, qui vérifie la relation :

$$\mathbb{E}[f(x, \xi(\omega))] - \min_{U^{ad}} \mathbb{E}[f] \geq c(\text{dist}_{U^\#}(x))^2. \quad (3.52)$$

5. La suite de réels positifs (ϵ_k) vérifie les propriétés suivantes :

$$\sum_{k=1}^{+\infty} \epsilon_k = +\infty ; \quad \sum_{k=1}^{+\infty} (\epsilon_k)^2 < +\infty.$$

Alors, la suite (x_k) construite par l'algorithme du gradient stochastique converge en norme \mathcal{L}^2 vers l'ensemble $U^\#$.

On pourra consulter [87] pour la démonstration.

3.5 Algorithme de Arrow-Hurwicz stochastique

Dans le cadre d'étude de fiabilité ou de risque, il est courant de considérer un problème d'optimisation associé à des contraintes exprimées en termes de probabilité de ruine : c'est le cas en mécanique lorsqu'on s'intéresse à la fiabilité d'une structure, c'est-à-dire à la probabilité qu'une structure avec paramètres incertains résiste à un chargement. Dans un processus d'optimisation de la structure, il sera alors nécessaire de prendre en compte cette contrainte. On pourrait certes contourner cette difficulté en imposant que cette contrainte soit toujours vérifiée, mais cela pourrait conduire à des solutions irréalisables ou trop coûteuses. Cette contrainte en probabilité décrit la part de risque que le constructeur est prêt à accepter dans le cahier des charges définissant la structure.

Le cadre probabiliste de l'utilisation de l'algorithme d'Arrow-Hurwicz correspond au problème suivant :

$$\begin{cases} \min_{x \in U^{ad}} \mathbb{E}[f(x, \xi(\omega))] \\ \text{s.c.} \\ \mathbb{E}[g(x, \xi(\omega))] \leq 0. \end{cases} \quad (3.53)$$

Comme dans le cas de l'algorithme du gradient stochastique, l'algorithme de Arrow-Hurwicz va utiliser des tirages indépendants de la variable aléatoire $\xi(\omega)$ pour approcher les espérances des quantités aléatoires. L'algorithme de Arrow-Hurwicz s'écrit alors sous la forme suivante :

$$x_{k+1} = \Pi_{U^{ad}} \left(x_k - \epsilon_k \left(\nabla_x f(x_k, \xi(\omega_{k+1})) + \lambda_k \nabla_x g(x_k, \xi(\omega_{k+1})) \right) \right) \quad (3.54)$$

$$\lambda_{k+1} = \Pi_{\mathbb{R}^+} \left(\lambda_k + \rho_k g(x_{k+1}, \xi(\omega_{k+1})) \right). \quad (3.55)$$

Sous les hypothèses décrites dans [5], on prouve que les suites (x_k) et (λ_k) sont bornées presque sûrement et que la suite (x_k) converge vers la solution x^* du problème.

Cependant, cet algorithme ne peut pas être utilisé dans le cas où une contrainte de risque

est introduite dans le problème d'optimisation. En effet, une telle contrainte s'écrit souvent sous la forme :

$$\mathbb{P}[\theta(x, \xi(\omega)) \geq \theta_0] \geq p_0, \quad (3.56)$$

ou encore,

$$\mathbb{E}[g(x, \xi(\omega))] \leq 0, \quad (3.57)$$

avec :

$$g(x, \xi(\omega)) = p_0 - \mathbf{1}_{\mathbb{R}_+}(\theta(x, \xi(\omega)) - \theta_0). \quad (3.58)$$

La fonction g n'est pas dérivable en x , rendant impossible la classique interversion dérivée-intégrale.

Là encore, il serait envisageable de se tourner vers des approches de type scénarios où la contrainte probabiliste est remplacée par un ensemble N de contraintes déterministes construites à partir de réalisations de la variable $\xi(\omega)$ satisfaisant la contrainte. Ainsi, le problème (3.53) est remplacé par le problème d'optimisation déterministe suivant :

$$\begin{aligned} & \min_{x \in U^{ad}} \mathbb{E}[f(x, \xi(\omega))] \\ & \text{s.c.} \begin{cases} \theta(x, \xi(\omega_1)) \geq \theta_0 \\ \vdots \\ \theta(x, \xi(\omega_N)) \geq \theta_0. \end{cases} \end{aligned} \quad (3.59)$$

Dans une autre approche similaire [53], les auteurs remplacent la contrainte en risque (3.56) par la contrainte :

$$\frac{1}{N} \sum_{i=1}^N \mathbf{1}(\theta(x, \xi(\omega_i)) \geq \theta_0) \geq \alpha, \quad (3.60)$$

en introduisant un estimateur de la contrainte probabiliste. Cela rajoute une boucle de calculs interne pour le calcul de cet estimateur. Mais de telles approches ont un cadre d'utilisation très limité et ne s'appliquent efficacement que pour des problèmes linéaires [62, 63]. Nous nous sommes donc tournés dans le cadre de cette thèse vers une approche récente permettant de contourner cette difficulté, développée par Mlle Andrieu dans sa thèse [4] et dans ses articles [5, 6]. Le principe de cette méthode est d'approcher la valeur de cette contrainte non linéaire à l'aide d'une fonction régulière obtenue par convolution. Plus précisément, cette approche considère une fonction de classe \mathcal{C}^1 , $\psi : \mathbb{R} \rightarrow \mathbb{R}$ ayant un unique maximum en 0 telle que :

$$\forall x \in \mathbb{R}, \psi(x) \geq 0, \psi(-x) = \psi(x) \text{ et } \int_{-\infty}^{+\infty} \psi(y) dy = 1. \quad (3.61)$$

La fonction ψ est alors la densité d'une variable aléatoire réelle. Cette fonction est symétrique, continue et possède son unique maximum en 0.

À présent, considérons une fonction $\Phi : \mathbb{R} \rightarrow \mathbb{R}$, $\Phi \in L^1$ et r un réel strictement positif. La convolution de la suite d'approximation de l'unité

$$\psi_r(\cdot) = \frac{1}{r} \psi\left(\frac{\cdot}{r}\right), \quad (3.62)$$

avec la fonction Φ non nécessairement dérivable, donne :

$$\Phi_r(x) = \frac{1}{r} \int_{-\infty}^{+\infty} \Phi(y) \psi\left(\frac{x-y}{r}\right) dy. \quad (3.63)$$

$\Phi_r(x)$ peut être vu comme une approximation de $\Phi(x)$ quand r tend vers 0, car $\psi_r(\cdot)$ converge au sens des distributions vers la fonction Dirac en 0. La fonction Φ_r est dérivable, et sa fonction dérivée est :

$$\Phi'_r(x) = \frac{1}{r^2} \int_{-\infty}^{+\infty} \Phi(y) \psi'\left(\frac{x-y}{r}\right) dy. \quad (3.64)$$

Dans le cadre de la contrainte en risque, la fonction indicatrice est régularisée par convolution avec la fonction $\psi_r(\cdot)$:

$$\begin{aligned} (\psi_r * \mathbf{1}_{\mathbb{R}_+})(z) &= \frac{1}{r} \int_{-\infty}^{+\infty} \mathbf{1}_{\mathbb{R}_+}(y) \psi\left(\frac{z-y}{r}\right) dy \\ &= \frac{1}{r} \int_0^{+\infty} \psi\left(\frac{y-z}{r}\right) dy. \end{aligned} \quad (3.65)$$

Puis on considère la valeur de cette fonction convolution en $\theta(x, \xi(\omega)) - \theta_0$:

$$(\psi_r * \mathbf{1}_{\mathbb{R}_+})(\theta(x, \xi(\omega)) - \theta_0) = \frac{1}{r} \int_0^{+\infty} \psi\left(\frac{y + \theta_0 - \theta(x, \xi(\omega))}{r}\right) dy. \quad (3.66)$$

On pose alors :

$$g_r(x, \xi(\omega)) = p_0 - (\psi_r * \mathbf{1}_{\mathbb{R}_+})(\theta(x, \xi(\omega)) - \theta_0). \quad (3.67)$$

De par les propriétés de la convolution, on obtient le résultat suivant :

$$\forall x, \forall \omega \in \Omega \text{ fixé, } g_r(x, \xi(\omega)) \xrightarrow{r \rightarrow 0} g(x, \xi(\omega)), \quad (3.68)$$

car $g(x, \xi(\omega)) = p_0 - \mathbf{1}_{\mathbb{R}_+}(\theta(x, \xi(\omega)) - \theta_0)$.

La fonction $g_r(\cdot, \xi(\omega))$, contrairement à $g(\cdot, \xi(\omega))$, est dérivable si la fonction θ l'est aussi, et sa dérivée vaut alors :

$$\nabla_x g_r(x, \xi(\omega)) = \frac{1}{r} \psi\left(\frac{\theta(x, \xi) - \theta_0}{r}\right) \nabla_x \theta(x, \xi(\omega)). \quad (3.69)$$

Si ψ est une fonction à support compact et de classe \mathcal{C}^1 sur son support, alors les conditions nécessaires à l'interversion intégrale-espérance sont vérifiées, et la convergence des fonctions convoluées est conservée. On introduit une suite (r_k) tendant vers 0 et on remplace donc la fonction g par sa version approchée g_{r_k} dans l'algorithme de Arrow-Hurwicz stochastique, qui devient :

$$\begin{aligned} x_{k+1} &= \Pi_{U^{ad}}(x_k - \epsilon_k(\nabla_x f(x_k, \xi(\omega_{k+1})) + \nabla_x g_{r_k}(x_k, \xi(\omega_{k+1}))\lambda_k)) \\ \lambda_{k+1} &= \Pi_{\mathbb{R}_+}(\lambda_k + \rho_k g_{r_k}(x_{k+1}, \xi(\omega_{k+1}))). \end{aligned} \quad (3.70)$$

La preuve de la convergence de cet algorithme vers la solution du problème d'optimisation (3.53) s'appuie là encore sur des considérations très techniques que nous ne détaillerons pas ici. Sous des hypothèses suffisantes de dérivabilité, de convexité, et si les suites (ϵ_k) , (ρ_k) et (r_k) vérifient certaines hypothèses de décroissance vers 0, alors cet algorithme modifié converge bien vers la solution du problème initial. Nous renvoyons le lecteur à l'article [5] pour les détails de cette preuve.

L'utilisation de cet algorithme nécessite de choisir trois suites : (ϵ_k) , (ρ_k) et (r_k) , ainsi qu'une fonction Φ . Ce choix est délicat et va conditionner la qualité de la convergence, comme l'a souligné l'auteur dans [5]. D'un certain point de vue, on peut considérer cet algorithme comme se rapprochant de l'optimisation multi-objectif, en le sens qu'il cherche à optimiser une fonction objectif tout en vérifiant une contrainte d'inégalité en probabilité. Ce sont deux objectifs aux tendances contraires, provoquant une instabilité du comportement de l'algorithme. Ce qui explique que les paramètres permettant à l'algorithme de converger sont parfois difficiles à trouver en pratique et nécessitent plusieurs essais numériques.

Après avoir exposé les méthodes numériques basées sur le gradient que nous allons utiliser pour les applications, nous passons maintenant à la description de l'extension de l'algorithme du recuit simulé au cas stochastique, qui sera utilisé au chapitre suivant pour illustrer son efficacité sur un problème d'optimisation aéroélastique.

3.6 Le recuit simulé stochastique

Le problème d'optimisation que nous considérerons ici est le suivant :

$$\underset{x \in U^{ad}}{\text{Argmin}} f(x) = \mathbb{E}[f(x, \xi(\omega))]. \quad (3.71)$$

Comme nous l'avons vu précédemment, la difficulté réside dans le calcul de l'espérance. Il est en effet très rare de connaître une expression analytique de cette quantité, de même qu'en général, on ne connaît pas l'expression de la loi de la variable $f(x, \xi(\omega))$, rendant le calcul numérique de l'espérance impossible. C'est en particulier le cas lorsque le gradient de la fonction f ne peut pas être calculé (cas de sortie d'un code numérique "boîte noire"). On peut alors chercher à approcher l'espérance par un estimateur construit en utilisant une méthode de Monte-Carlo, la qualité de l'estimation étant contrôlée par la construction d'intervalles de confiance associés à des probabilités données. On remplace alors le problème initial par sa version échantillonnée avec N tirages :

$$\underset{x \in U^{ad}}{\text{Argmin}} \frac{1}{N} \sum_{j=1}^N f(x, \xi(\omega_j)). \quad (3.72)$$

Comme on peut le constater sur la figure (3.1), un algorithme d'optimisation stochastique basé sur la construction d'un estimateur de la fonction objectif nécessite deux boucles imbriquées, l'une pour l'estimation de la fonction objectif, l'autre pour l'évolution des variables d'optimisation. Le temps de calcul induit par l'estimation constitue un goulot d'étranglement qui limite l'utilisation de telles approches [97]. Dans le cadre de ce travail, nous avons voulu étendre une méthodologie développée par Diwekar [67], initialement conçue dans un cadre d'optimisation combinatoire, au cas continu. L'auteur s'est particulièrement penché sur les coûts numériques d'une utilisation du recuit simulé dans le cadre stochastique, et a proposé plusieurs approches susceptibles de réduire efficacement ces coûts.

Le principe de la méthode repose sur une adaptation du nombre d'échantillons utilisés pour construire l'estimateur de la fonction objectif : au début de l'algorithme, c'est la phase exploratrice qui est importante, et il n'est pas nécessaire de connaître la fonction

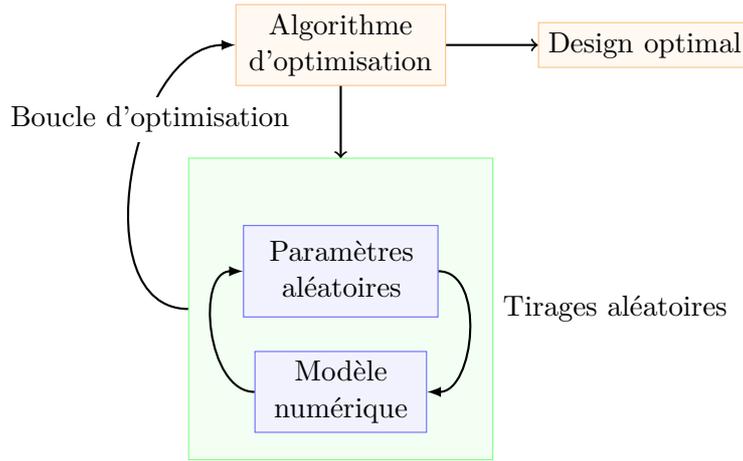


FIG. 3.1: Algorithme stochastique

objectif avec précision, le nombre d'échantillons peut être faible. Puis, au cours de l'évolution de l'algorithme, lorsque l'on s'approche de la solution, la précision de l'estimateur doit être plus grande, donc le nombre de tirages également.

L'adaptation du nombre d'échantillons sera obtenue par le biais d'une fonction objectif pénalisée qui se substituera à la fonction objectif d'origine. Le terme de pénalité rajouté à la fonction objectif va évoluer au cours des itérations de l'algorithme. Cette fonction objectif pénalisée se présente sous la forme :

$$f_{pen}(x, \xi(\omega)) = \frac{1}{N} \sum_{j=1}^N f(x, \xi(\omega_j)) - b(T_i) \frac{\sigma}{\sqrt{N}}, \quad (3.73)$$

où $T \mapsto b(T)$ est une fonction décroissante de \mathbb{R} dans \mathbb{R}_+ et $i \mapsto T_i$ est une fonction décroissante de \mathbb{N} dans \mathbb{R} : c'est la loi de la température. σ^2 est la variance de la variable aléatoire $f(x, \xi(\omega))$ et N est le nombre de tirages effectués.

Afin d'éclairer le rôle du terme de pénalité $b(T_i) \frac{\sigma}{\sqrt{N}}$, nous allons rappeler comment est construit un intervalle de confiance associé à un estimateur de la moyenne.

Si l'on considère une loi $X \in \mathcal{L}^2$ de variance $\sigma^2 < +\infty$, on sait d'après le théorème central limite [14] que si (X_1, \dots, X_N) sont N tirages indépendants de même loi que X , et en posant

$$\bar{X}_N = \frac{1}{N} \sum_{i=1}^N X_i, \quad (3.74)$$

alors on a la convergence en loi :

$$\sqrt{N}(\bar{X}_N - \mathbb{E}[X]) \rightarrow \mathcal{N}(0, \sigma^2). \quad (3.75)$$

De plus, en notant $\bar{\sigma}_N$ l'estimateur non biaisé de la variance σ défini par :

$$\bar{\sigma}_N^2 = \frac{1}{N-1} \sum_{i=1}^N (X_i - \bar{X}_N)^2 \quad (3.76)$$

on sait que $\bar{\sigma}_N^2 \rightarrow \sigma^2$, et le lemme de Slutsky [65] permet d'écrire :

$$\sqrt{\frac{N}{\bar{\sigma}_N^2}} (\bar{X}_N - \mathbb{E}[X]) \rightarrow \mathcal{N}(0, 1). \quad (3.77)$$

Une autre possibilité de représenter cet intervalle de confiance, que nous n'exploiterons pas ici, peut se faire à l'aide de la loi de Student St . En effet, dans le cas de l'estimation de la moyenne d'une variable aléatoire sans en connaître la variance, on dispose du résultat suivant, plus précis que le théorème central limite :

$$\sqrt{N} \left(\frac{\bar{X} - \mathbb{E}[X]}{\bar{\sigma}^2} \right) \approx St(N - 1). \quad (3.78)$$

Dans le cadre du théorème central limite, en supposant que N est assez grand, et que $q_{1-\alpha/2}$ est le quantile d'ordre $1 - \alpha/2$ de la loi normale centrée réduite, on a le résultat d'encadrement suivant :

$$\mathbb{P}[-q_{1-\alpha/2} \leq \sqrt{N} \left(\frac{\bar{X}_N - \mathbb{E}[X]}{\bar{\sigma}_N} \right) \leq q_{1-\alpha/2}] = \alpha. \quad (3.79)$$

Par conséquent, on a l'encadrement suivant avec une probabilité égale à α :

$$\mathbb{E}[X] \in \left[\bar{X}_N - q_{1-\alpha/2} \frac{\bar{\sigma}_N}{\sqrt{N}}, \bar{X}_N + q_{1-\alpha/2} \frac{\bar{\sigma}_N}{\sqrt{N}} \right]. \quad (3.80)$$

On voit alors que le terme $b(T_i)$ va jouer le rôle du quantile $q_{1-\alpha/2}$ de l'intervalle de confiance associé à l'estimateur de $\mathbb{E}[f(x, \xi(\omega))]$. La fonction pénalisée introduite peut être considérée comme la borne inférieure de cet intervalle de confiance. Cependant, l'ajout de ce terme de pénalité entraîne rapidement deux difficultés : tout d'abord, la loi étudiée à chaque évaluation est différente de par la nature du problème étudié : à chaque point x de l'espace, la fonction aléatoire $f(x, \xi(\cdot))$ obéit à une loi différente, et possède en particulier une variance différente. Il faut donc estimer cette variance à nouveau à chaque itération du calcul. Un autre problème est qu'on ne maximise qu'à un certain pourcentage la moyenne de la fonction aléatoire, de par la présence du terme de pénalisation, ce qui fait que les valeurs en dehors de cet intervalle ne sont pas prises en compte par l'algorithme.

L'idée qui a été développée dans le cadre du recuit simulé stochastique permet de s'affranchir de ces deux problèmes, si on admet que les variances de la fonction aléatoire restent dans un domaine borné. Nous considérons alors un majorant σ_0 des écarts-types σ , et nous construisons la fonction de pénalisation suivante :

$$f_{pen}(x) = \frac{1}{N} \sum_{j=1}^N f(x, \xi(\omega_j)) - b(T) \frac{\sigma_0}{\sqrt{N}}, \quad (3.81)$$

où $\sigma_0 = \sup_x \left(\mathbb{E}[f(x, \xi(\omega))^2] - \mathbb{E}[f(x, \xi(\omega))]^2 \right)$.

L'idée sous-jacente consiste à choisir une constante de départ qui sera un majorant de la valeur maximale prise par les variances, et un terme croissant exponentiellement avec le niveau de température i . Par exemple, on peut prendre pour b la fonction :

$$i \mapsto b(T_i) = \frac{b_0}{k^i}; \quad 0 < k < 1, \quad i = 1, 2, \dots \quad (3.82)$$

Ainsi, la fonction pénalisée correspond à la borne inférieure d'un intervalle de confiance dont la probabilité associée croît avec le niveau de température, mettant à profit la nature et les particularités du recuit simulé.

Une telle stratégie a une autre conséquence : le choix du nombre de tirages N devient partie intégrante de la fonction objectif. En effet, afin que la fonction objectif reste la partie la plus importante de la fonction objectif pénalisée, il a été choisi d'imposer que le terme de pénalisation ne soit pas supérieur à un pourcentage donné p de la valeur de la fonction objectif. Pour cela, sachant qu'à une température donnée, le terme de pénalisation est connu au nombre de tirages près, c'est sur ce nombre que l'on peut agir afin de satisfaire cette contrainte. Au cours de l'algorithme, on impose donc que si le nombre de tirages est insuffisant pour vérifier la contrainte du pourcentage $p\%$ sur le terme de pénalité, c'est-à-dire si la relation (3.84) n'est pas vérifiée :

$$b(T) \frac{\sigma_0}{\sqrt{N}} \leq \frac{p}{100N} \sum_{j=1}^N f(x, \xi(\omega_j)), \quad (3.83)$$

$$\sqrt{N} \geq \frac{100 b(T) \sigma_0}{\frac{p}{N} \sum_{j=1}^N f(x, \xi(\omega_j))}, \quad (3.84)$$

ce nombre est alors augmenté de façon à diminuer le terme de pénalisation en dessous du seuil critique. Cela permet aussi de favoriser les estimations réalisées avec le plus de tirages, c'est-à-dire les plus fiables.

Cependant, à niveau de température égal, on constate que le terme de pénalisation est constant. Dans une telle situation, son ajout n'influence pas le test de Metropolis, et donc le comportement de l'algorithme. La solution proposée est de rendre ce nombre aléatoire et variable au cours de l'algorithme, tout en le faisant augmenter au fur et à mesure que l'on se rapproche de la solution du problème d'optimisation. De cette façon, le terme de pénalisation sera différent entre deux évaluations successives et aura un réel intérêt tout au long de l'algorithme.

Le terme $b(T_i)$, croissant en fonction du niveau de température i et inclus dans le terme de pénalisation, a pour but de faire évoluer la probabilité α du quantile $q_{1-\alpha/2}$ vers 0. À partir d'un certain nombre d'itérations de l'algorithme, la probabilité pour que l'intervalle de confiance contienne $\mathbb{E}[f(x, \xi(\omega))]$ sera proche de 1. Le fait qu'il soit croissant force le nombre de tirages à augmenter pour respecter la contrainte des $p\%$. Comme en fin d'algorithme, le point courant est à proximité de la solution du problème d'optimisation, ce nombre important de tirages permet de diminuer la variance de l'estimateur de la solution et d'affiner la précision des calculs.

Stratégie du choix du nombre de tirages N

Dans notre étude, nous avons donc fait évoluer de manière aléatoire le nombre de tirage à chaque évaluation. Au départ, le nombre est fixé à N_0 , puis à chaque itération, on lui additionne un entier choisi selon une loi uniforme sur $[-\nu, \nu]$, indépendante des précédents tirages. Pour éviter des divergences possibles de ce nombre, nous mettons en place des bornes supérieures et inférieures N_{min} et N_{max} . Après incrémentation, le nombre de

tirages est ramené sur l'espace $\llbracket N_{min}, N_{max} \rrbracket$, et c'est cette quantité qui devient le nombre de tirages de l'itération en cours.

Il est toutefois nécessaire que la contrainte sur le pourcentage du terme de pénalité par rapport à l'estimation de la fonction objectif soit toujours vérifiée. On effectue N_{min} tirages, puis on vérifie si le terme de pénalité ne dépasse les $p\%$ de la moyenne des réalisations de la fonction objectif. Si ce n'est pas le cas, on incrémente N_{min} et N_{max} d'une certaine quantité $\eta \geq 1$, on réalise les tirages supplémentaires et on vérifie à nouveau si la contrainte est vérifiée. Tant que ce n'est pas le cas, on continue d'incrémenter N_{min} et N_{max} . Puis, y compris dans le cas où la contrainte est vérifiée sans qu'il y ait à modifier N_{min} , on actualise N , et on effectue le nombre de tirages additionnels nécessaires.

1. Initialiser les grandeurs numériques : N_0, N_{min}, N_{max} .
2. Tant que

$$\sqrt{N_{min}} \geq \left(\frac{100 b(T)\sigma_0}{\frac{p}{N_{min}} \sum_{j=1}^{N_{min}} f(x_t, \xi(\omega_j))} \right) : \quad (3.85)$$

- (a) actualiser : $N_{min} = N_{min} + \eta$;
- (b) actualiser : $N_{max} = N_{max} + \eta$;
- (c) réaliser η tirages de $\xi(\omega)$.

3. Actualiser : $N_{k+1} = \text{proj}_{\llbracket N_{min}, N_{max} \rrbracket}(N_k + U_{[-\nu, \nu]})$.

Algorithme du recuit simulé stochastique

Les points essentiels de cette méthode sont l'actualisation aléatoire et le respect de la règle des $p\%$ de la fonction objectif pour le terme de pénalité. Une fois la méthode de modification du nombre de tirages déterminée, l'algorithme du recuit simulé stochastique se présente sous la forme suivante :

1. Initialiser la solution initiale x_0 et choisir les valeurs numériques des constantes.
2. Initialiser la température T_0 et le nombre de paliers de température.
3. Pour chaque température T_i , faire la boucle suivante jusqu'à vérifier un critère de convergence :
 - (a) actualiser le nombre de tirages selon le principe cité précédemment,
 - (b) choisir aléatoirement un point candidat dans le voisinage du point courant : $x_{cand} = x_k + \delta x$,
 - (c) effectuer N_{tir} tirages de la variable aléatoire ξ ,
 - (d) calculer la fonction échantillonnée pénalisée $f_{pen}(x_{cand})$,
 - (e) mesurer l'écart $\Delta F = f_{pen}(x_{cand}) - f_{pen}(x_k)$,
 - (f) si $\Delta F \geq 0$, accepter le point candidat : $x_{k+1} = x_{cand}$,
 - (g) sinon, accepter le point candidat avec une probabilité de $e^{-\frac{\Delta F}{k_B T}}$,
 - (h) rester en x_k si le point candidat n'est pas accepté : $x_{k+1} = x_k$.
 - (i) faire un test de convergence ; s'il est vérifié, incrémenter le palier de température i , sinon revenir en (a).

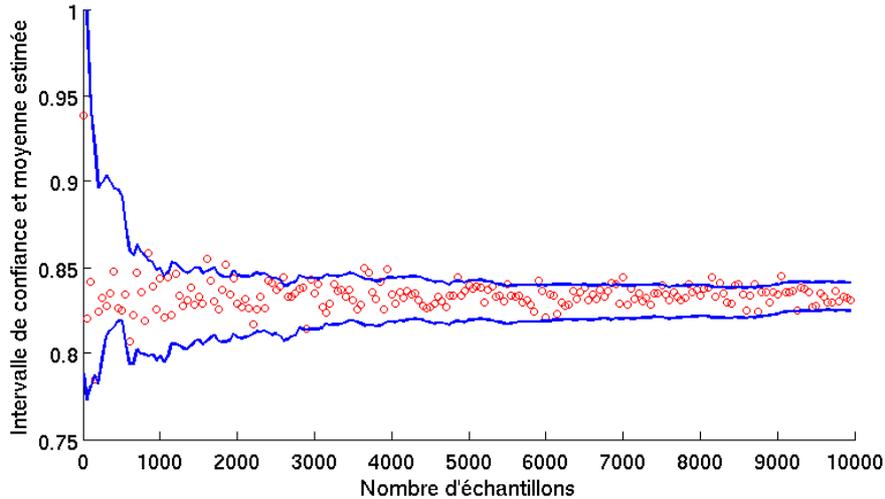


FIG. 3.2: Évolution d'une suite de tirages aléatoires et des bornes de l'intervalle de confiance associé

3.6.1 Utilisation des suites d'Hammersley

Comme on l'a vu au paragraphe précédent, l'efficacité de l'algorithme du recuit simulé stochastique repose sur le terme de pénalisation introduit dans la fonction objectif qui est étroitement lié à l'intervalle de confiance de l'estimateur de la moyenne. De plus, on a vu que les bornes de l'intervalle de confiance associées à l'estimateur de la moyenne de la variable aléatoire étaient proportionnelles au quotient de l'écart-type de la variable par la racine carrée du nombre d'échantillons. Cependant, le théorème central limite n'est valable que pour des échantillons choisis par tirages aléatoires indépendants, et ne s'applique donc pas aux échantillons choisis par plans d'expériences ou par des méthodes d'échantillonnages quasi aléatoires. Pour illustrer ce propos considérons la fonction suivante :

$$f(u_1(\omega), u_2(\omega)) = u_1^2(\omega) + u_2(\omega),$$

où u_1 et u_2 sont deux variables aléatoires indépendantes de lois uniformes sur $[0, 1]$. Dans un premier temps, nous avons réalisé 10 000 tirages de ces deux variables en utilisant le générateur aléatoire de MatLab, et nous avons estimé la variance σ_f de la fonction f . Puis, nous avons tracé sur la figure (3.2) l'intervalle de confiance associé à l'estimateur de la moyenne $\mathbb{E}[f]$ en fonction du nombre d'échantillons. Nous avons également représenté des estimateurs de cette moyenne construite à partir d'autres réalisations de ces mêmes variables u_1 et u_2 . Nous pouvons constater sur cette figure la dispersion des résultats. Dans un deuxième temps, nous avons refait la même simulation numérique, cette fois en utilisant pour u_1 et u_2 un échantillonnage quasi aléatoire à l'aide de suite d'Halton générées par MatLab et nous avons tracé les mêmes résultats sur la figure (3.3). Nous voyons que l'intervalle de confiance construit par le théorème central limite n'est pas pertinent lorsqu'on utilise des suites quasi-aléatoires, les bornes étant largement surestimées.

En réalité, une espérance n'est autre qu'une intégrale, et son calcul approché via une méthode de Monte-Carlo n'utilise pas le côté "aléatoire" des tirages, mais la répartition uniforme des tirages sur le domaine d'intégration. Dans le cadre de calcul d'intégrales, des

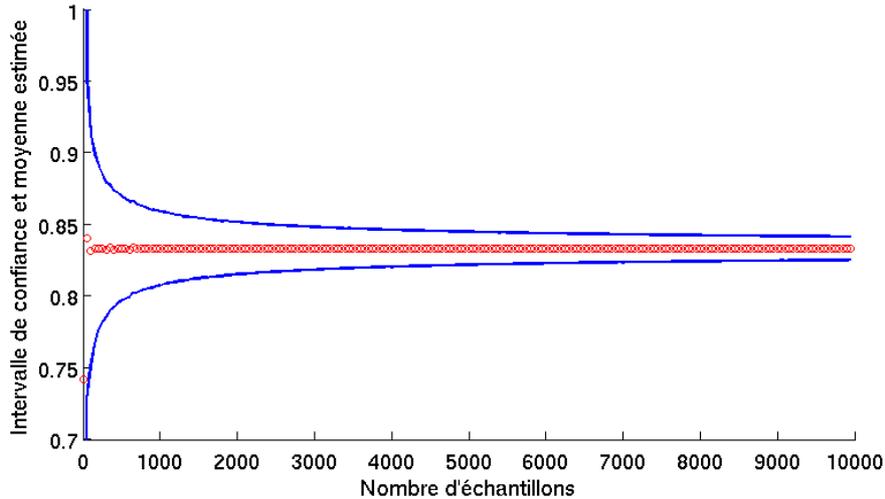


FIG. 3.3: Évolution d'une suite quasi aléatoire et des bornes du théorème central limite

suites à discrédance faible ont été mises au point pour réduire la variance de l'estimateur et accélérer la convergence. On appelle de telles procédures des méthodes de quasi Monte-Carlo. L'utilisation de suites à discrédance faible a été étudiée par Diwekar dans le cadre du recuit simulé stochastique dans [66] afin de réduire les temps de calcul.

Partant de ces considérations, Diwekar s'est penchée dans des articles ultérieurs [24, 25, 47] sur le problème de déterminer un intervalle de confiance adapté à un échantillonnage à l'aide de suites à discrédance faible, et elle s'est intéressée plus particulièrement aux suites d'Hammersley. En s'inspirant du comportement du mouvement brownien fractionnaire, et grâce à des méthodes empiriques basées sur des expériences numériques, Diwekar a identifié une relation d'échelle entre la largeur de l'intervalle de confiance et le nombre d'échantillons, proposant de substituer au coefficient \sqrt{N} donné par le théorème central limite le coefficient $N^{1.8}$.

Si bien que si l'on reprend le dernier exemple, l'intervalle de confiance ainsi construit serait celui illustré sur la figure (3.4). On peut apprécier le gain obtenu sur la largeur de cet intervalle qui, dans le cas particulier de l'exemple traité, converge très rapidement : une vingtaine de points suffit à réduire cet intervalle à zéro. En définitive, si on utilise des échantillons de points choisis à l'aide de suites d'Hammersley, il faut remplacer la fonction pénalisée (3.81) par la fonction :

$$f_{pen}(x) = \frac{1}{N} \sum_{j=1}^N f(x, \xi(\omega_j)) - b(T) \frac{\sigma_0}{N^{1.8}}. \quad (3.86)$$

Une des conséquences immédiates est que le terme de pénalisation décroissant plus vite que dans le cas de Monte-Carlo, le respect de la borne des $p\%$ à ne pas dépasser pour la pénalité va se concrétiser par un nombre plus faible d'échantillons, à intervalle de confiance

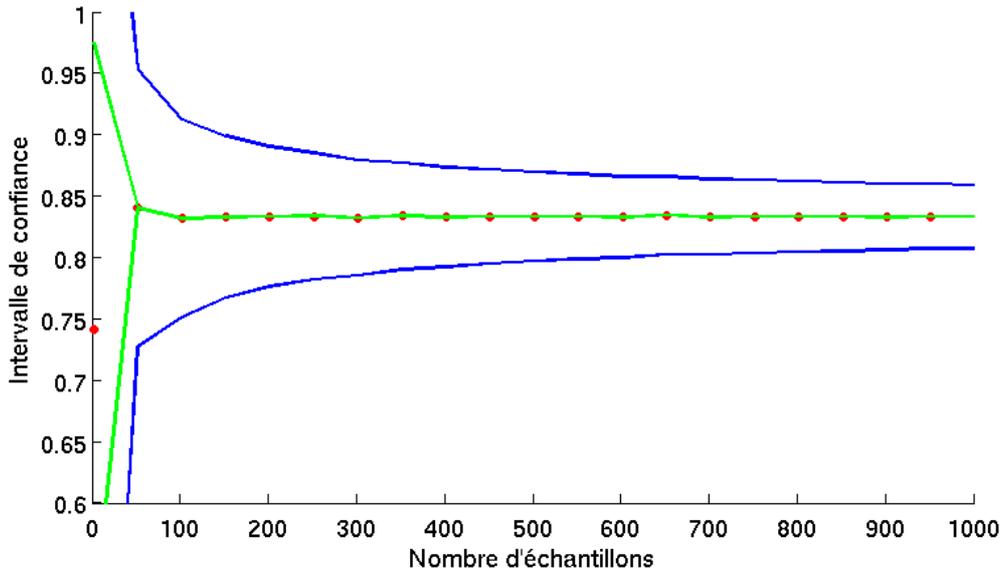


FIG. 3.4: Évolution d'une suite quasi aléatoire et des bornes de l'intervalle de confiance associé

donné :

$$\text{Monte-Carlo : } N \geq \left(\frac{100b(T)\sigma_0}{\frac{p}{N} \sum_{j=1}^N f(x, \xi(\omega_j))} \right)^2, \quad (3.87)$$

$$\text{Hammersley : } N \geq \left(\frac{100b(T)\sigma_0}{\frac{p}{N} \sum_{j=1}^N f(x, \xi(\omega_j))} \right)^{5/9}. \quad (3.88)$$

Remarque 3.6. Contrairement à l'estimateur de Monte-Carlo où on peut ajouter n'importe quel autre tirage aléatoire de la même loi, pourvu qu'il soit indépendant des autres, il est nécessaire de connaître à l'avance le nombre de points que l'on souhaite calculer et la dimension exacte de l'espace dans lequel sont construites les suites à discrédance faible. En effet, la construction de ces suites demande explicitement le nombre de points à construire et le nombre de coordonnées. Néanmoins, dans le cadre du recuit simulé stochastique, cette contrainte peut être levée, car le nombre de tirages évolue au cours de l'algorithme, et ce nombre est donc connu avant d'effectuer les tirages. Il faut toutefois s'assurer que le terme de pénalité respecte la contrainte des $p\%$ de la fonction objectif. Une autre limitation de l'utilisation des suites d'Hammersley et des suites à discrédance faible en général est leur mauvais comportement en grande dimension. La vitesse de convergence de la méthode de Monte-Carlo, elle, est indépendante de la dimension de l'espace. Ainsi, en grande dimension, l'intérêt de ces suites est réduit.

Remarque 3.7. Plus de détails sur les suites à discrédance faible peuvent être trouvés en annexe.

Chapitre 4

Optimisation d'une maquette d'aile d'avion

Nous nous proposons dans ce chapitre de résoudre deux problèmes d'optimisation liés à l'aéroélasticité d'une maquette d'aile d'avion en présence d'incertitudes. Le premier problème posé aura pour but d'étendre le domaine de stabilité de la structure en maximisant la valeur de la pression critique. Deux algorithmes seront testés, l'algorithme du gradient stochastique et l'algorithme du recuit simulé stochastique. Le second problème d'optimisation étudié traitera de la minimisation de la masse de la structure sous une contrainte probabiliste liée à la stabilité de la structure. L'algorithme d'Arrow-Hurwicz stochastique sera alors utilisé.

Le modèle de structure à la base de ces problèmes étudié ici n'est certes pas un modèle complet d'avion ou d'aile comme ceux utilisés par les constructeurs, mais il comporte un nombre déjà conséquent de variables d'optimisation (près d'une centaine). Le but de ce travail est d'évaluer l'applicabilité de ces algorithmes au domaine de l'aéroélasticité. Nous donnerons un critère très simple permettant de juger le coût numérique pour tout modèle donné.

Notons que le modèle que nous considérerons correspond à une maquette en soufflerie. La vitesse du fluide et la température sont constantes, le paramètre variable sera la pression. De plus, la pression et la densité volumique du fluide sont alors linéairement dépendants, c'est pourquoi nous parlerons de pression critique tout en faisant des calculs de gradient de la densité volumique de façon équivalente à une constante près.

4.1 Le modèle structure

Le modèle numérique que nous avons utilisé pour réaliser nos tests numériques est un modèle à éléments finis d'une maquette d'aile d'avion réalisée dans les années 90 : l'aile Dyvas, appelée depuis AMP-Wing. Cette maquette a été conçue pour être testée en soufflerie et sa structure est basée sur l'aile d'un avion de transport de grande capacité. Son poids est de 6.7 kilos et son envergure est de 1.16 mètre. La longueur de corde à l'emplanture est égale à 0.42 mètre, celle à l'extrémité est égale à 0.1 mètre. La flèche de l'aile forme un angle de 32° . Cette aile a été modifiée spécifiquement pour présenter une instabilité dans le domaine d'exploitation de la soufflerie et permettre d'étudier ce phénomène. Elle entre donc parfaitement dans notre cadre d'étude où le flottement est une contrainte importante.

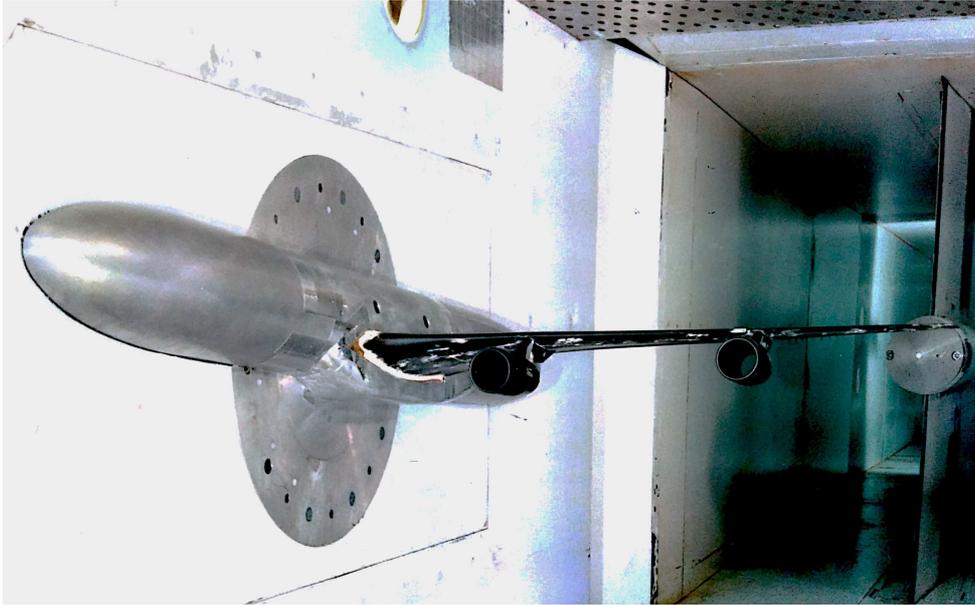


FIG. 4.1: Maquette de l'aile Dyvas

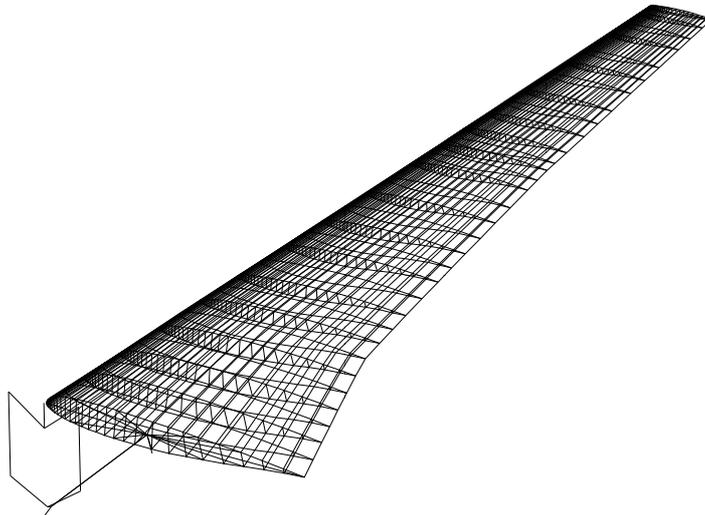


FIG. 4.2: Modèle à éléments finis

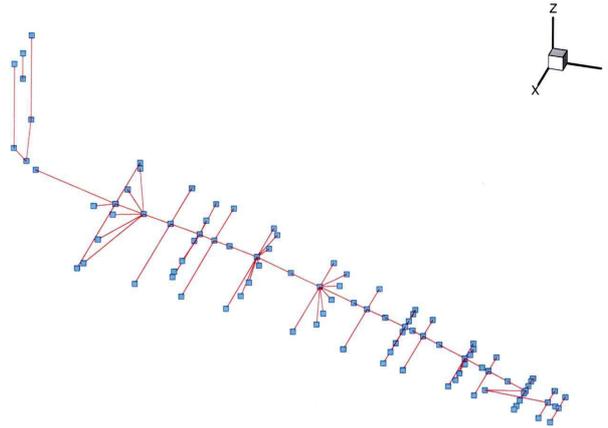


FIG. 4.3: Modèle à masses concentrées

Quelques années plus tard, cette maquette d'aile a été modélisée dans le but de réaliser des simulations numériques et de comparer différents codes aérodynamiques. Un modèle à éléments finis fonctionnant avec le code structure NASTRAN a été produit. La figure (4.2) représente le modèle à éléments finis complet.

Pour les applications, nous avons utilisé un modèle à éléments finis simplifié, de type arête de poisson. Un tel modèle se caractérise par les particularités suivantes : il s'agit d'un modèle à masses concentrées au lieu d'éléments volumiques avec une certaine densité massique. Les masses sont donc représentées par des points, et reliées entre elles par des segments qui représentent des poutres auxquelles sont associées les raideurs. Comme les masses sont ponctuelles, nous allons pouvoir représenter l'ensemble de la masse de l'aile du modèle à l'intérieur d'un unique vecteur masse, dont chaque composante correspondra à un point massique du modèle. De plus, les masses et les raideurs sont décorréélées dans ce modèle numérique, ce qui va aboutir à des simplifications lors des calculs, notamment la formule analytique du gradient de la pression critique. Le modèle comporte 97 masses concentrées et 93 poutres. Ce modèle est représenté sur la figure (4.3).

À première vue, l'utilisation d'un tel modèle peut sembler un peu trop éloigné de la structure réelle, mais il faut savoir que de tels modèles à masses concentrées sont utilisés par les constructeurs dans les phases d'avant-projet, car ils permettent d'avoir une représentation claire et synthétique des propriétés de la structure.

L'étude du comportement modal de ce modèle est réalisé par le biais du code NASTRAN. Nous avons représenté dans la figure (4.4) les deux premiers modes souples de la structure. Il s'agit du premier mode de flexion (à gauche) de fréquence propre 23.4 Hz, et du premier mode de torsion (à droite) de fréquence propre 31.85 Hz. C'est le couplage de ces deux modes qui sera à l'origine de l'instabilité aéroélastique de la maquette.

Comme les masses et les raideurs sont décorréélées dans ce modèle, il est normal de penser que ce modèle est simple et ne suffit pas à modéliser avec précision et réalisme une aile d'avion. En règle générale, les paramètres modulables d'une aile d'avion les plus simples

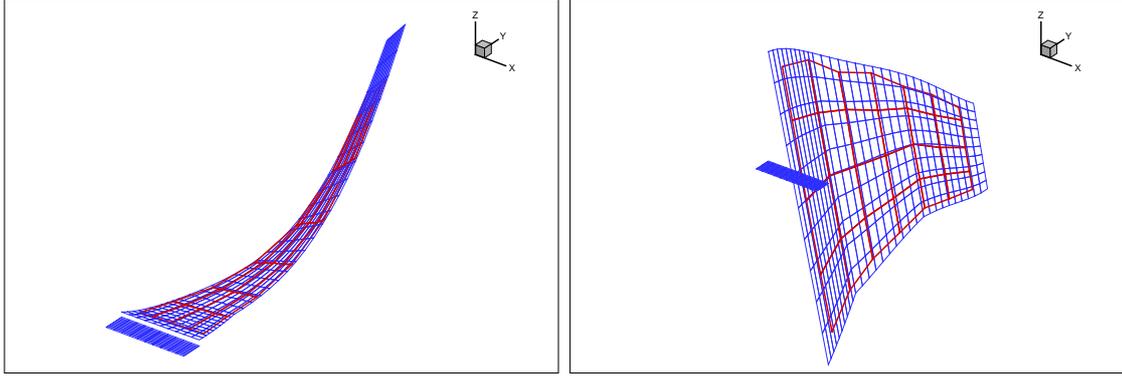


FIG. 4.4: Déformées modales

sont les épaisseurs de plaque, et leurs raideurs dépendent de leur épaisseur, tout comme leur poids. Cependant, un tel modèle permet d'avoir une vue synthétique du comportement de la structure.

4.2 Le modèle aérodynamique

La méthode des doublets utilisée pour le calcul des FAG repose sur un maillage des surfaces portantes de la structure. Ce maillage est plus grossier que le maillage à éléments finis utilisé pour modéliser la structure. Nous avons représenté dans la figure (4.5) le maillage utilisé pour représenter la maquette.

Comme cela a été expliqué dans le chapitre 2, dans cette thèse, la transformée de Laplace des FAG est calculée en faisant une hypothèse de linéarité sur les mouvements q :

$$\hat{F}(s) = \frac{1}{2}\rho V^2 A(s)\hat{q}(s). \quad (4.1)$$

Le terme $A(s, V)$ des FAG est calculé par la méthode des doublets et approché d'après le modèle de Roger par une fraction rationnelle en $\frac{s}{V}$ comportant P pôles $\frac{i\omega_j}{V}$, $j = 1, \dots, N$:

$$A(s) = D_0 + D_1 \frac{s}{V} + D_2 \left(\frac{s}{V}\right)^2 + \sum_{i=1}^P \frac{A_i^1 \frac{s}{V} + A_i^0}{B_i(\frac{s}{V})}. \quad (4.2)$$

Les matrices D_i , A_i^1 , A_i^0 sont à coefficients réels, et les polynômes B_i sont de degré 2 à coefficients réels. Les pôles sont les zéros des polynômes B_i et sont choisis aléatoirement pour interpoler au mieux la valeur des FAG.

Dans nos tests numériques, nous avons choisi 11 fréquences réduites équiréparties, nous avons calculé via la méthode des doublets la valeur des FAG en ces fréquences réduites, nous avons choisi deux pôles dans le modèle de Roger et nous avons réalisé une minimisation des moindres carrés de l'expression (4.2) sur les valeurs calculées des FAG en les fréquences réduites. Par la suite, nous emploierons cette approximation en lieu et place de l'expression théorique des FAG dans nos calculs numériques. On représente dans la figure (4.6) l'interpolation ainsi obtenue des deux premiers éléments de la matrice des FAG $A(s)$.

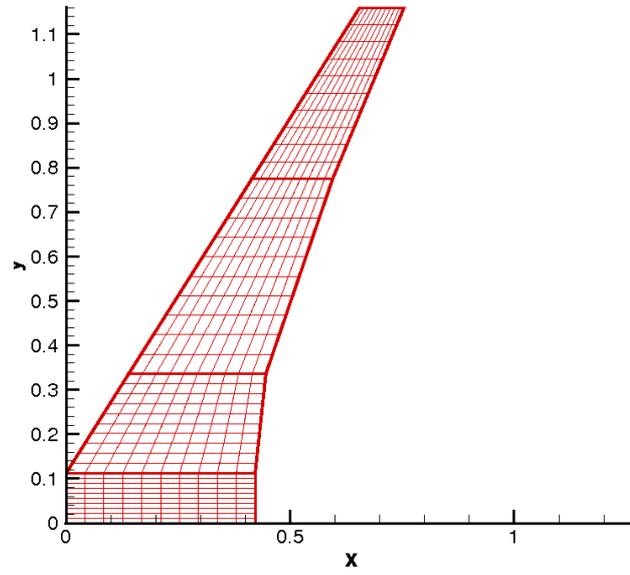
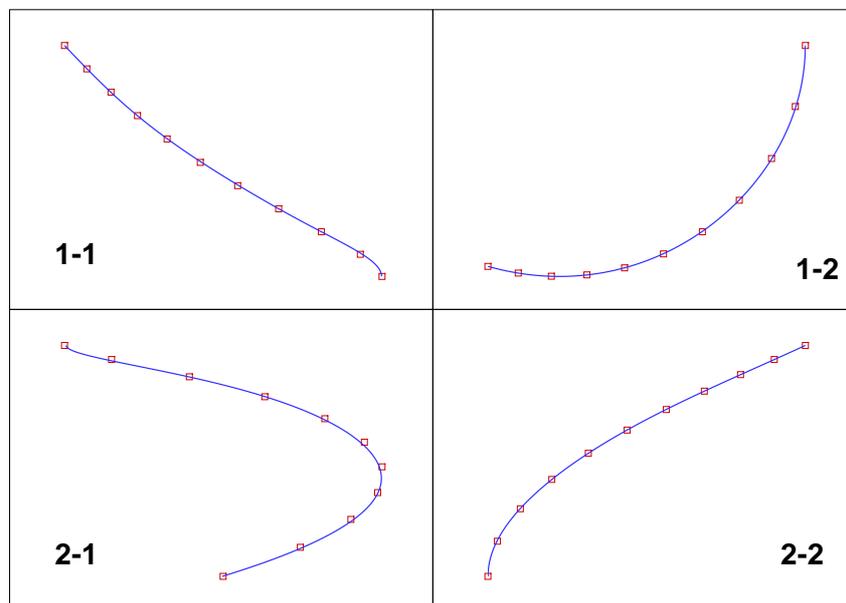


FIG. 4.5: Maillage aérodynamique

FIG. 4.6: Interpolation des termes (i,j) de la fonction $A(s)$, $i,j=1,2$

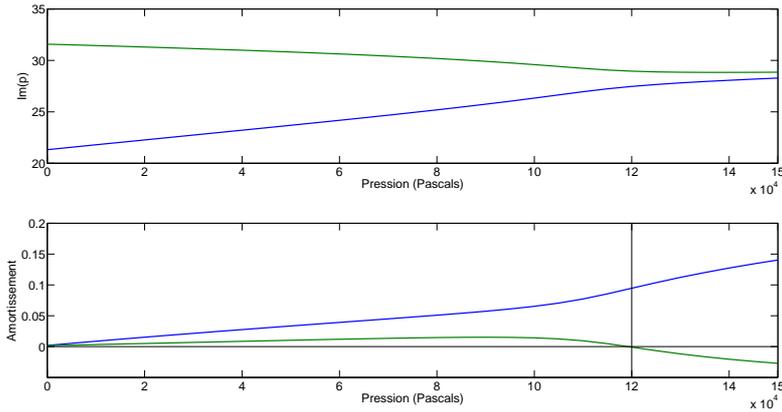


FIG. 4.7: Évolution de la partie imaginaire et de l'amortissement deux premiers modes propres en fonction de la pression

4.2.1 Stabilité de la structure nominale

Il faut remarquer qu'en soufflerie, la vitesse de l'air ambiant est constante, et la pression de l'air est le paramètre ajustable pour simuler une vitesse d'écoulement différente. Comme ce modèle numérique correspond à une maquette en soufflerie, la vitesse sera constante, et ce sera la pression qui variera au cours des différentes simulations numériques. Les forces aérodynamiques seront calculées avec la méthode des doublets et projetées dans le domaine fréquentiel puis sur le modèle structurel, car le modèle à éléments finis des forces aérodynamiques est plus simple que celui de la structure de l'aile, et il faut donc interpoler les forces calculées au moyen de fonctions rationnelles.

Nous résolvons l'équation de flottement sans incertitudes :

$$\left(p^2 M + K + \frac{1}{2} \rho V^2 A(p, V) \right) X = 0, \quad (4.3)$$

pour un ensemble de valeurs de la pression $q = \frac{1}{2} \rho V^2$:

$$q = k \Delta q ; k = 0, \dots, 31 ; \Delta q = 5\,000 \text{ Pascals}, V = 256 \text{ km/h}.$$

Nous traçons à présent sur la figure (4.7) l'évolution de la partie imaginaire et de l'amortissement aéroélastique des deux premiers modes propres de la maquette en fonction de la pression q . L'amortissement aéroélastique est défini par la quantité :

$$\alpha(p) = \frac{\text{Re}(p)}{\text{Im}(p)}. \quad (4.4)$$

On identifie une valeur critique pour la pression au-delà de laquelle le système couplé devient instable, l'amortissement changeant de signe. Dans le cas nominal, la pression critique vaut :

$$q_c \approx 119\,000 \text{ Pascals}. \quad (4.5)$$

4.2.2 Introduction et modélisation d'incertitudes

Discutons à présent de la manière d'inclure des variables aléatoires dans le modèle numérique d'une aile d'avion, afin de reproduire l'effet des méconnaissances diverses qui peuvent affecter la structure. Pour cela, il faut tout d'abord caractériser les phénomènes les plus susceptibles d'être source d'incertitudes dans le modèle.

Les constructeurs définissent un modèle initial de masse réaliste. Ce modèle de masse étant susceptible d'être modifié par rapport aux plans originaux pour une meilleure corrélation du modèle numérique avec ses performances prévues en avant-projet, les masses ajoutées peuvent alors être considérées comme un facteur incertain, et ce facteur doit être pris en compte dans les études de stabilité avant ajout définitif des masses dans le modèle. Une autre source d'incertitudes concernent le carburant, les câbles, les équipements électroniques de l'appareil, le personnel de bord et les passagers. Autant d'éléments non inclus dans le modèle structure et dont la présence peut avoir une influence sur le comportement de l'appareil.

Ces considérations ont le désavantage d'être des incertitudes épistémiques, en cela qu'elles ne sont pas quantifiables par manque d'informations ou à cause de la trop grande dispersion des valeurs empiriques. Il est possible, grâce à l'expérience des ingénieurs, de connaître certaines tendances, par exemple déterminer un domaine borné dans lequel peuvent évoluer les valeurs des grandeurs incertaines. Grâce à l'utilisation de la méthode du maximum d'entropie et de l'information de Fisher [39], on peut alors déterminer les lois de variables aléatoires qui correspondent à ces tendances. Dans le cas de la seule connaissance du support de la "loi" aléatoire, le maximum d'entropie correspond à une loi uniforme.

Cependant, on s'intéressera à une quantité plus facilement identifiable dans notre étude : la variation des modules d'Young des matériaux composites. Comme nous l'avons souligné dans l'introduction, ceux-ci représentent une partie très importante de la structure des avions récents, car ils possèdent des propriétés physiques très intéressantes : une faible densité massique et une grande rigidité. Cependant, il existe de nombreuses manières différentes de réaliser un matériau composite, puisqu'il s'agit généralement d'empilements de différents matériaux avec des orientations de fibres qui varient d'une couche à l'autre. Le choix et l'orientation des matériaux de base étant libres, on peut obtenir une multitude de matériaux composites aux propriétés très diverses. La nature même de ces matériaux est donc très variable et leurs propriétés physiques peuvent être mal connues, en dépit de leurs propriétés physiques très intéressantes.

Les constructeurs peuvent néanmoins produire des matériaux composites dont les caractéristiques ne s'écartent pas plus d'un certain pourcentage des valeurs nominales. Il est même possible de créer des matériaux avec exactement les caractéristiques voulues, mais cela revient en général très cher, car il faut pour cela du personnel hautement qualifié et du matériel de grande précision. C'est pourquoi la plupart du temps, les industriels se contentent d'un "intervalle de confiance" sur les valeurs numériques des propriétés physiques. On peut donc supposer dans notre étude que les paramètres physiques incertains des matériaux composites sont à valeurs dans un intervalle borné. Comme nous l'avons déjà signalé, selon le maximum d'entropie et la loi d'information de Fisher, la loi la plus incertaine d'une variable aléatoire sur un support donné est la loi uniforme, c'est pourquoi nous modéliserons ici les paramètres incertains par des lois uniformes.

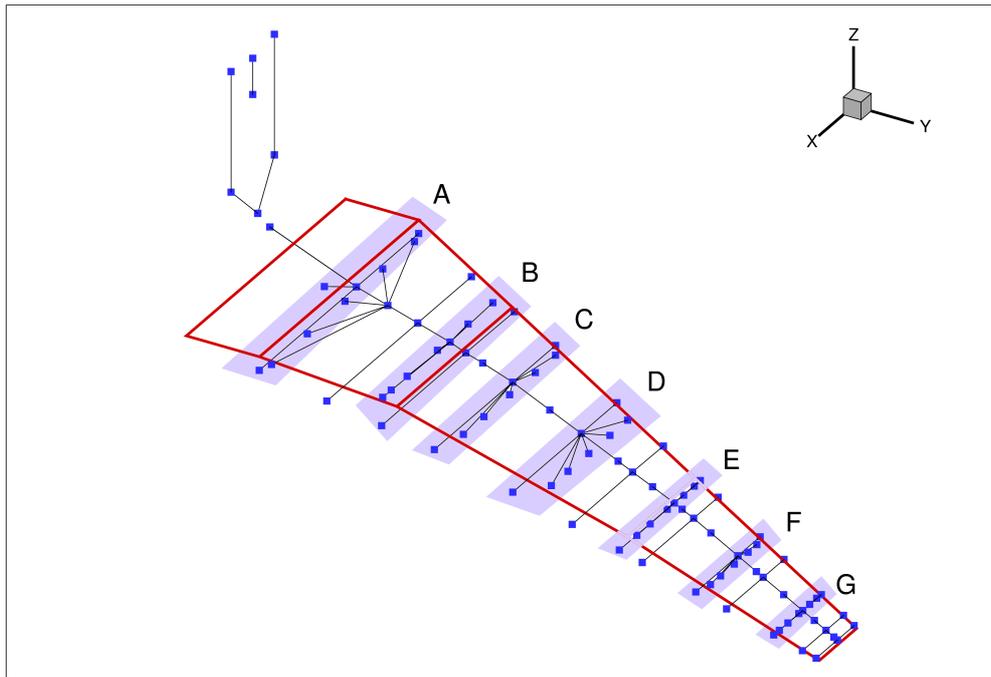


FIG. 4.8: Répartition des zones de raideur aléatoire

Dans le cadre de ce travail, nous définirons certaines zones dans le modèle numérique identifiées par les zones colorées sur la figure (4.8), et nous représenterons le module d'Young de chaque zone par une variable aléatoire uniforme indépendante des autres zones. Les autres raideurs du système seront inchangées.

Remarque 4.1. Tant que les variables aléatoires introduites pour modéliser les incertitudes sont indépendantes, leur nombre n'aura pas de répercussion sur le coût numérique des méthodes d'optimisation.

Effets des incertitudes

Une fois les incertitudes modélisées par des variables aléatoires, la solution de l'équation aéroélastique standard (2.33) devient elle-même une variable aléatoire, et par conséquent la pression critique également. Il est intéressant dans un premier temps d'évaluer l'influence des incertitudes sur la stabilité de la maquette. Pour cela, nous avons effectué dans un premier temps une étude de propagation d'incertitudes. Nous avons effectué \mathcal{N} tirages de variables aléatoires modélisant les incertitudes et réalisé pour chaque tirage une analyse complète de stabilité à partir de laquelle nous avons construit un histogramme de la pression critique. La figure (4.9) représente cet histogramme lorsqu'on a choisi $\mathcal{N} = 1\,000$ et que les variables aléatoires suivent des lois uniformes à support sur $[0.75k_l, 1.25k_l]$, où k_l est la raideur nominale de la zone l . La valeur nominale de la pression critique étant égale à 119 000 Pascals, nous pouvons constater que les paramètres aléatoires choisis pour modéliser l'incertitude induisent une dispersion non négligeable sur la valeur de la pression critique, ce qui justifie notre étude.

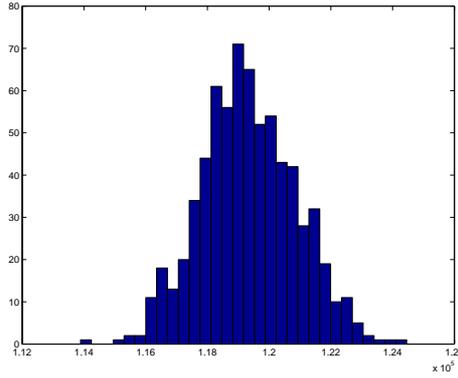


FIG. 4.9: Histogramme de la pression critique du modèle nominal

4.3 Optimisation de la répartition de masse

Ayant constaté une valeur critique pour la pression ainsi qu'une influence des incertitudes de raideur sur la stabilité de la maquette, nous allons chercher à optimiser la répartition de masse sur les points massiques du modèle à éléments finis de la maquette, de manière à augmenter le plus possible la valeur de cette pression critique et à étendre le plus possible le domaine de vol de cette maquette. Mathématiquement, le problème qu'on cherche à résoudre s'écrit :

$$\left\{ \begin{array}{l} \min_{m \in \mathbb{R}_+^N} \mathbb{E}[-q_c(m, \xi(\omega))] ; m = (m_1, \dots, m_N) \\ \text{s.c.} \\ \sum_{i=1}^N m_i = \sum_{i=1}^N m_{i,0} ; \forall i, m_i \in [(1 - \beta)m_{i,0}, (1 + \beta)m_{i,0}]. \end{array} \right. \quad (4.6)$$

Pour la suite, on pose :

$$U^{ad} = \left\{ m \in \mathbb{R}^m \mid \sum_{i=1}^N m_i = \sum_{i=1}^N m_{i,0} ; m_i \in [(1 - \beta)m_{i,0}, (1 + \beta)m_{i,0}] \right\}, \quad (4.7)$$

l'ensemble des solutions admissibles. C'est un ensemble fermé borné convexe, permettant d'utiliser des outils de projection. Dans les applications numériques effectuées, le super-élément qui représente la fixation de l'aile au fuselage de l'avion n'a pas été modifié au cours des phases d'optimisation. Les masses à l'extrémité de l'aile, par rapport auxquelles le comportement de l'aile était très sensible, n'ont pas non plus été incluses dans l'optimisation. Nous avons donc considéré seulement $N = 83$ points massiques du modèle, ceux considérés comme étant non critiques, et avons autorisé une modification maximale de $\beta = 25\%$ par rapport aux valeurs nominales. On représentera donc l'ensemble des paramètres d'optimisation par un unique vecteur massique $m = (m_1, \dots, m_N)$ dont chaque coordonnée correspond à un point massique du modèle.

4.3.1 Enchaînement des calculs

Le modèle numérique de l'aile Dyvas est un fichier NASTRAN. Il contient plusieurs données, dont la position des points massiques dans l'espace, leur masse et moment cinétique associés, ainsi que les barres qui relient les points massiques entre eux et les raideurs

associées. Tous les coefficients d'Young des barres sont égaux. Dans notre étude, nous avons déterminé 7 zones de barres que nous allons rendre aléatoires, comme pour la propagation d'incertitudes. Lors de chaque nouveau calcul numérique sur le modèle, on effectue un tirage de 7 variables aléatoires indépendantes de lois uniformes sur l'intervalle $[1 - \alpha, 1 + \alpha]$. On multiplie ensuite la raideur nominale des zones délimitées par les réalisations de ces lois uniformes :

$$k_l(\omega) = k_{l,0} \times U_l(\omega) \quad (4.8)$$

où $U_l(\omega)$ suit une loi uniforme sur $[1 - \alpha, 1 + \alpha]$. On prendra $\alpha = 0.25$ dans l'expérience numérique. Ainsi, notre modèle numérique est devenu aléatoire. Le modèle étant à masse ponctuelle, on peut représenter la masse de l'ensemble du modèle par un unique vecteur m . Lors de la manipulation du fichier numérique, il faudra modifier les valeurs numériques des masses, mais aussi celles des moments cinétiques de manière proportionnelle, directement dans le fichier de données. Un script Unix a été utilisé pour réaliser automatiquement la mise à jour des paramètres.

Le calcul numérique se fait via le logiciel NASTRAN. Il produit plusieurs fichiers nécessaires au fonctionnement de la chaîne CAPRI qui va permettre de calculer les forces aérodynamiques et la pression de flottement. La chaîne CAPRI est un ensemble de programmes qui a été développé à l'ONERA au département DADS, spécifiquement pour le calcul des FAG.

Le fichier final de la chaîne CAPRI contient l'évolution des premiers modes propres en fonction de la pression. La pression à laquelle le flottement se produit est ainsi simple à déterminer. Afin de réaliser un calcul des pseudo-valeurs propres de l'équation aéroélastique, il nous faut également déterminer les matrices servant à calculer les forces aérodynamiques généralisées. Celles-ci sont modélisées par des fractions rationnelles basées sur le modèle de Roger, et un code de la chaîne Capri permet d'extraire les matrices calculées au cours du fonctionnement de la chaîne.

Une fois toutes ces informations extraites, on peut réaliser les boucles d'optimisation, par exemple dans un programme MatLab. Dans le cas d'un tirage aléatoire comme dans le recuit simulé stochastique, il faut effectuer plusieurs fois le calcul de la pression critique et rassembler les résultats. Une fois ces calculs réalisés, il faut calculer la nouvelle valeur du vecteur massique du modèle selon l'algorithme employé. Puis, il faut remplacer les masses actuelles du modèle par leurs nouvelles valeurs, avant de réitérer une boucle de calcul.

4.3.2 Utilisation de l'algorithme du gradient stochastique

Le premier algorithme dont nous allons tester le comportement numérique est le gradient stochastique. L'algorithme se présente de la façon suivante :

1. poser $m^{(0)} \in \mathbb{R}^N$, les valeurs initiales du modèle,
2. choisir la suite de pas $(\epsilon_k)_{k \in \mathbb{N}}$,
3. à chaque itération, actualiser et projeter la suite :

$$m^{(k+1)} = \text{proj}_{U^{ad}}(m^{(k)} + \epsilon_k \nabla_m q_c(m^{(k)}, \xi(\omega_{k+1}))), \quad (4.9)$$

4. si un test de convergence est vérifié, stopper l'algorithme, sinon revenir en (c).

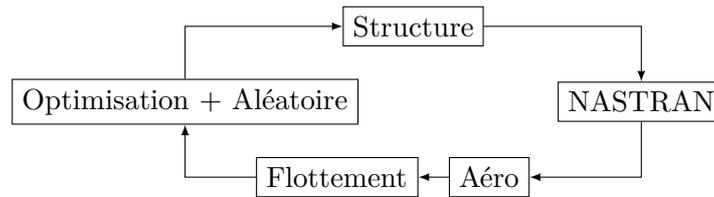


FIG. 4.10: Organigramme schématique des méthodes sans boucles aléatoires

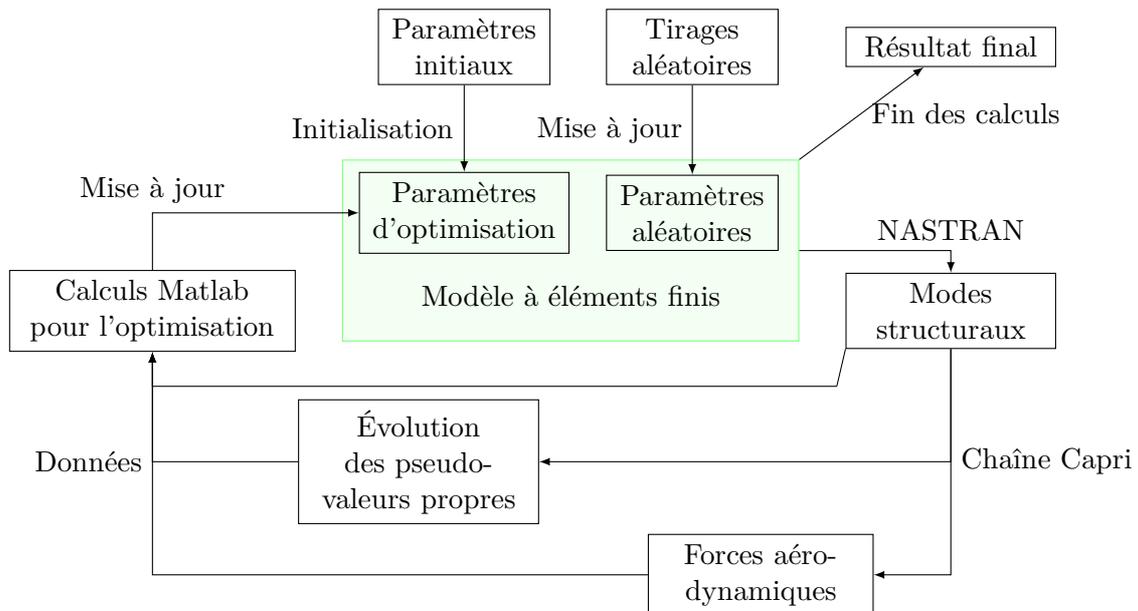


FIG. 4.11: Organigramme des méthodes sans boucles aléatoires

Schématiquement, l'algorithme suit les grandes lignes présentées dans la figure (4.10).

Pour résumer, l'algorithme est constitué de boucles d'optimisation et s'arrête quand un critère d'arrêt ou de convergence est vérifié. À l'intérieur d'une boucle, les programmes s'enchaînent séquentiellement : les paramètres d'optimisation et aléatoires sont définis, le modèle numérique est implémenté, on calcule les forces aérodynamiques, puis le flottement qui en découle, on applique l'algorithme dédié, et on recommence la boucle en faisant un nouveau tirage des paramètres aléatoires et en actualisant les paramètres d'optimisation en accord avec les calculs de l'algorithme. On a donc en simultané la phase d'optimisation et les tirages aléatoires. Pour réaliser ces boucles de calculs, on a toutefois besoin de nombreux programmes, dont la chaîne CAPRI. L'organigramme des calculs numériques et des logiciels employés est décrit schématiquement dans la figure (4.11).

Formules de gradient

Les relations établies dans les chapitres précédents nous permettent d'obtenir la dérivée de la vitesse critique en fonction d'un paramètre quelconque α . Ici, notre modèle garde une vitesse constante au cours des simulations numériques, et la pression est l'élément variable. C'est pourquoi nous avons introduit un problème portant sur la pression critique.

Le gradient de cette grandeur en fonction des masses concentrées du modèle s'écrit sous la forme :

$$\frac{\partial q_c}{\partial m_i}(m_i) = -\frac{\text{Re}(H_1)}{\text{Re}(H_2)}, \quad (4.10)$$

avec

$$H_1 = V^2 \frac{p^2(m_i, \rho)Y(m_i, \rho) \frac{\partial M}{\partial m_i}(m_i)X(m_i, \rho)}{Y(m_i, \rho)(2p(m_i, \rho)M + \frac{1}{2}\rho V^2 \frac{\partial A}{\partial x_1}(p(m_i, \rho), V))X(m_i, \rho)},$$

et

$$H_2 = \frac{2V^2Y(m_i, \rho)A(p(m_i, \rho), V)X(m_i, \rho)}{Y(m_i, \rho)(4p(m_i, \rho)M + \rho V^2 \frac{\partial A}{\partial x_1}(p(m_i, \rho), V))X(m_i, \rho)},$$

Nous avons également besoin de calculer les dérivées partielles de la matrice de masse par rapport aux points massiques. On rappelle que cette matrice est définie grâce à l'opérateur de masse \mathcal{M} qui vérifie la relation suivante sur les modes structure :

$$\langle \mathcal{M}\varphi_k, \varphi_l \rangle = \sum_{i=1}^N m_i \langle \varphi_k(P_i), \varphi_l(P_i) \rangle. \quad (4.11)$$

Grâce à cette formule, on peut écrire le coefficient (k, l) de la matrice de masse M sous la forme suivante :

$$M_{k,l} = \sum_{i=1}^N m_i \langle \varphi_k(P_i), \varphi_l(P_i) \rangle. \quad (4.12)$$

Cela signifie que la matrice de masse dépend linéairement des points massiques. On pose $\Phi(P_i)$ la matrice dont les vecteurs-colonnes sont les modes structure en le point P_i . La dérivée de la matrice de masse par rapport au point massique P_i associé à la masse m_i découle directement de la formule ci-dessus :

$$\frac{\partial M}{\partial m_i} = {}^t\Phi(P_i)\Phi(P_i). \quad (4.13)$$

Le calcul de cette relation nécessite de connaître la valeur des modes structure en l'ensemble des points du modèle à éléments finis, que l'on calcule à l'aide du logiciel structure NASTRAN.

Choix des paramètres

Ici, le choix des constantes est crucial pour le bon comportement de l'algorithme. Celui-ci vise à manipuler des grandeurs telles que la masse, et la masse totale de notre modèle numérique est environ de 6.7 kg. Le vecteur massique, qui contient 83 masses, est modifié par des valeurs de gradient sur la pression, qui vaut 119 000 Pascals, et dont les variations peuvent être de l'ordre de 10%, comme nous l'a montré la propagation d'incertitudes. Si le pas de l'algorithme du gradient stochastique est choisi trop grand, les valeurs vont immédiatement atteindre les bornes de l'espace admissible, et l'algorithme ne pourra pas atteindre en un temps raisonnable la solution du problème d'optimisation si celui-ci est à l'intérieur du domaine admissible.

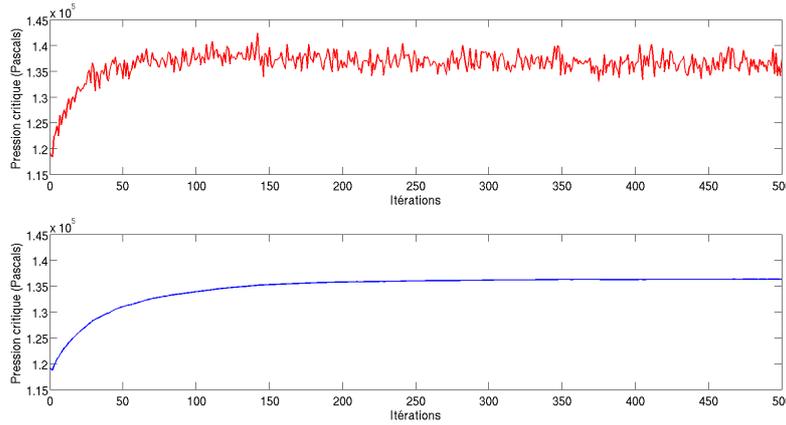


FIG. 4.12: Évolution de la pression critique et de sa moyenne de Cesaro

Pour des raisons de simplicité, on souhaite choisir la suite (ϵ_k) de la forme :

$$\epsilon_k = \frac{K_0}{(K_1 + k)^\gamma}. \quad (4.14)$$

Les conditions de convergence de l'algorithme données dans le théorème (3.5) imposent de prendre le paramètre γ dans l'intervalle $]0.5, 1]$. On choisit alors $\gamma = 1$, pour que les pas diminuent le plus rapidement possible, afin d'accélérer la convergence à proximité de la solution. K_0 est une constante choisie pour que le rapport entre le vecteur $m^{(k)}$ et le gradient $\nabla_m q_c$ soit compris dans une fourchette entre 0.01 et 0.1 au départ et éviter des comportements incohérents (si K_0 est trop petit, la convergence sera trop lente, et s'il est trop important, l'algorithme aura du mal à s'extraire du bord de l'ensemble des solutions admissibles). Il faut également éviter que les pas diminuent trop entre le début et la fin de l'algorithme, sinon la convergence à proximité de la solution est compromise. C'est pourquoi on rajoute une constante K_1 dans l'expression des pas, sa valeur numérique étant de l'ordre de grandeur du nombre d'itérations prévu. Ainsi, le rapport entre la valeur initiale et la valeur finale de la suite (ϵ_k) est choisi de façon à ne pas dépasser la dizaine, surtout quand le nombre total d'itérations dépasse le millier voire davantage.

Dans notre premier test numérique, nous choisissons $K_0 = 1$ et $K_1 = 100$ pour le pas de l'algorithme. Les variables aléatoires suivent des lois uniformes sur $[0.75, 1.25]$ ($\alpha = 0.25$) et les masses pourront être modifiées jusqu'à 25% ($\beta = 0.25$). Nous avons fait fonctionner l'algorithme pendant 500 itérations. La figure (4.12) représente l'évolution de la pression critique au cours des itérations, ainsi que la moyenne de Cesaro des pressions critiques calculées.

On constate sur la courbe de l'évolution de la moyenne de Cesaro que la pression semble avoir atteint une limite asymptotique. Afin de comparer la solution avant et après optimisation, nous avons réalisé une étude de propagation d'incertitudes sur les valeurs initiales et finales du modèle en effectuant 1 000 tirages de variables aléatoires par solution. Nous avons reporté sur la figure (4.13) les histogrammes estimés pour les deux séries de calcul : à gauche l'histogramme de la pression critique avant optimisation, à droite celui obtenu

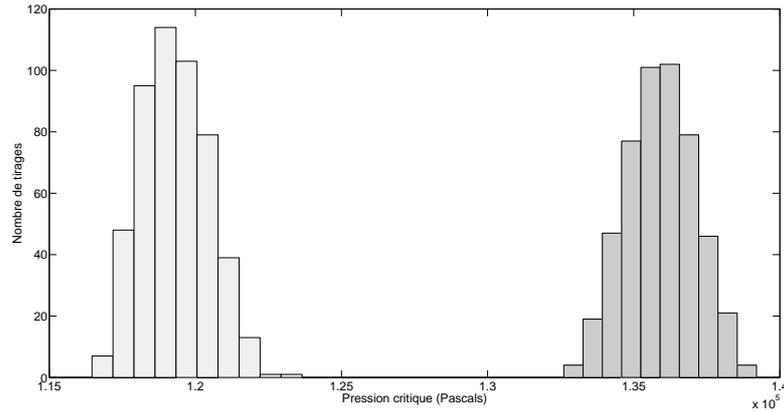


FIG. 4.13: Histogramme de la pression critique des valeurs nominales et optimisées

Masses nominales (kg)	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}
Masses optimisées (kg)	0.6102	0.8218	0.7816	0.5327	0.3214	0.4123	0.6272	0.4435	0.4029	0.2412	0.1597
	0.6832	1.0272	0.8883	0.3995	0.2410	0.3092	0.4704	0.3326	0.3022	0.3015	0.1996
Masses nominales (kg)	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}	m_{17}	m_{18}	m_{19}	m_{20}	m_{21}	m_{22}
Masses optimisées (kg)	0.1645	0.1263	0.0978	0.0959	0.1469	0.1152	0.0819	0.0814	0.0815	0.0734	0.0647
	0.2056	0.1579	0.1222	0.1199	0.1836	0.1440	0.1024	0.1017	0.1019	0.0551	0.0485

TAB. 4.1: Valeurs des points massiques après optimisation

après optimisation.

En jouant uniquement sur la répartition de masse, on a pu nettement décaler vers la droite l'histogramme de la pression critique du modèle. Dans le tableau (4.1), on peut observer l'effet de l'algorithme sur les 22 masses situées sur l'arête centrale de l'aile, qui constituent les masses les plus importantes et les plus sensibles pour la stabilité de l'aile (nous les avons représentées sur la figure (4.14)).

D'autre part, quarante-deux paramètres d'optimisation ont atteint leur borne inférieure du domaine de solution admissible. On constate que ces paramètres sont répartis sur l'ensemble du modèle, aucune région de l'aile n'est privilégiée.

Il faut également vérifier l'effet de l'algorithme sur le centre de gravité de l'aile. En effet, un déplacement trop important peut affecter d'autres pièces de l'appareil. Par exemple, si la masse s'est déplacée en bout d'aile, il faudra une fixation au fuselage plus résistante, donc plus lourde. Les études sur la globalité de la structure pourraient ainsi être remises en cause. Nous avons donc calculé le centre de gravité de l'aile après optimisation et l'avons comparé avec le centre de gravité initial. Comme on peut le voir dans le tableau (4.3.2), celui-ci ne s'est pas déplacé de manière significative au cours de l'algorithme.

Avant optimisation (m)	0.3284	0.4490	-0.0046
Après optimisation (m)	0.3296	0.4509	-0.0053

TAB. 4.2: Coordonnées du centre de gravité des masses avant et après optimisation

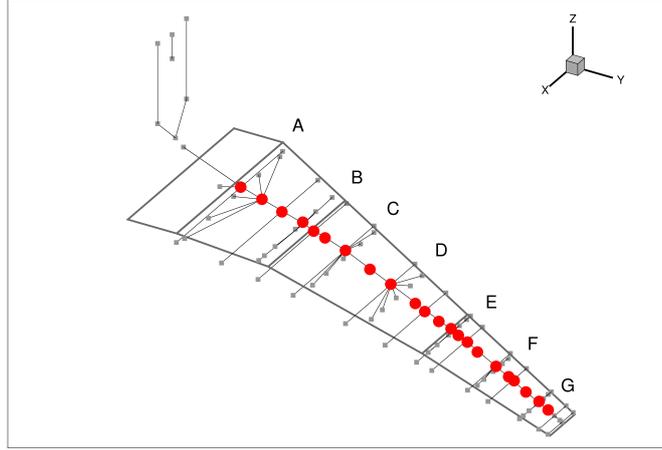


FIG. 4.14: Masses optimisées le long de l'arête centrale

4.3.3 Utilisation du recuit simulé stochastique

Nous souhaitons évaluer dans ce paragraphe les performances de l'algorithme du recuit simulé stochastique sur le problème précédent. Le gradient de la fonction objectif ne sera bien évidemment pas utilisé. Pour rappel, le problème posé est le suivant :

$$\begin{cases} \min_{m \in \mathbb{R}^N} \mathbb{E}[-q_c(m, \xi(\omega))] ; m = (m_1, \dots, m_N) \\ \text{s.c.} \\ \forall i, m_i \in [(1 - \beta)m_{i,0}, (1 + \beta)m_{i,0}]. \end{cases} \quad (4.15)$$

L'algorithme se présente de la façon suivante :

1. poser $m^{(0)} \in \mathbb{R}_+^N$ égal aux valeurs initiales du modèle,
2. choisir les constantes de départ,
3. faire M fois la boucle suivante pour le palier de température i :
 - (a) choisir m^c de manière aléatoire dans le voisinage du point courant $m^{(k)}$,
 - (b) actualiser le nombre de tirages N_{alea} et effectuer les tirages des paramètres aléatoires du système, de façon à vérifier la condition des $p\%$ sur le terme de pénalité,
 - (c) calculer la pression critique pour chaque tirage,
 - (d) calculer $f_{pen}(m^c) = \sum_{j=1}^{N_{alea}} -q_c(m^c, \xi(\omega_j)) + b(T_i) \frac{\sigma_0}{(N_{alea})^\tau}$,
 - (e) calculer $\Delta f = f_{pen}(m^c) - f_{pen}(m^{(k)})$
 - (f) si $\Delta f \geq 0$, poser $m^{(k+1)} = m^c$,
 - (g) sinon, tirer une variable aléatoire U sur $[0, 1]$ indépendamment des autres tirages,
 - (h) si $U < \exp\left(\frac{\Delta f}{k_B T_i}\right)$, poser $m^{(k+1)} = m^c$, sinon poser $m^{(k+1)} = m^{(k)}$.
4. actualiser le palier de température : $i \mapsto i + 1$,
5. si un test de convergence est vérifié, stopper l'algorithme, sinon revenir en (3).

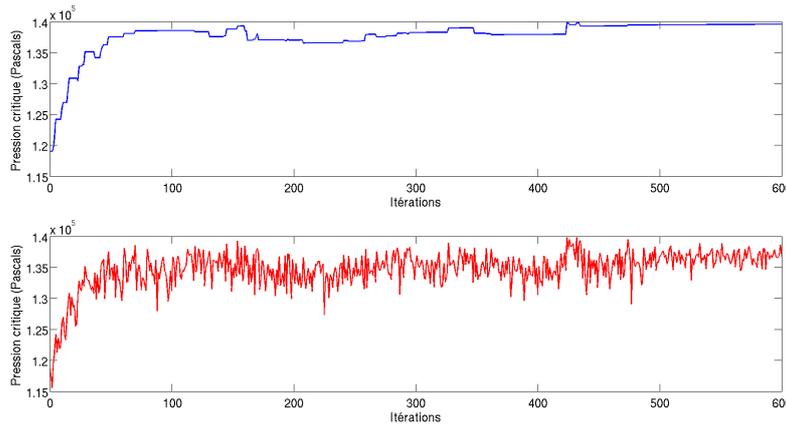


FIG. 4.17: Évolution de la pression critique du point courant et du point candidat

où m^c est le point candidat à tester, $\mathcal{N}(0, D_i) \in \mathbb{R}^N$ est un vecteur gaussien de moyenne nulle, et la matrice D_i est diagonale et a pour coefficients

$$(D_i)_{j,j} = 2^{-i}(m_0^j)^2\gamma^2, \quad j = 1, \dots, N.$$

Ce vecteur massique est ensuite projeté sur l'espace des solutions admissibles, puis les tirages aléatoires et les calculs de stabilité sont effectués pour le vecteur massique candidat.

Notre premier choix de paramètres a été le suivant :

$$\begin{aligned} p = 5\% ; \quad \alpha = 0.25 ; \quad \beta = 0.25 ; \quad T_0 = 1\,000 ; \quad (b_0\sigma_0) = 8\,000 ; \quad t = 0.97, \\ s = \frac{1}{0.97} ; \quad \gamma = 0.05 ; \quad N_{min} = 5 ; \quad N_{max} = 15 ; \quad \nu = 3 ; \quad \eta = 5. \end{aligned} \quad (4.18)$$

Les variables aléatoires suivent donc des lois uniformes sur $[0.75, 1.25]$ ($\alpha = 0.25$) et les masses pourront être modifiées jusqu'à 25% ($\beta = 0.25$).

Le paramètre τ apparaissant dans la fonction pénalisée à l'étape (d) dépend, comme cela a été expliqué dans le chapitre 3, de la nature de l'échantillonnage utilisé pour générer les paramètres aléatoires. Il vaut 0,5 pour un échantillonnage aléatoire de type Monte-Carlo, et il vaut 1,8 pour un échantillonnage utilisant des suites d'Hammersley. Dans ce premier calcul, τ sera égal à 0,5.

On fait tourner l'algorithme sur 60 paliers de température, et on effectue $M = 10$ itérations par palier. Le tirage des paramètres incertains se fait selon la procédure de Monte-Carlo. Sur la figure (4.17), nous avons représenté en haut l'évolution de la pression critique du point courant au cours des itérations, et en bas son évolution pour le point candidat.

Comme pour le gradient stochastique, on constate une convergence vers une limite asymptotique de la pression critique du point courant. Afin de comparer la solution avant et après optimisation, nous avons réalisé à nouveau une étude de propagation d'incertitudes sur les valeurs initiales et optimisées du modèle, en effectuant 2 000 tirages de variables aléatoires par solution, que nous avons représentée dans la figure (4.18).

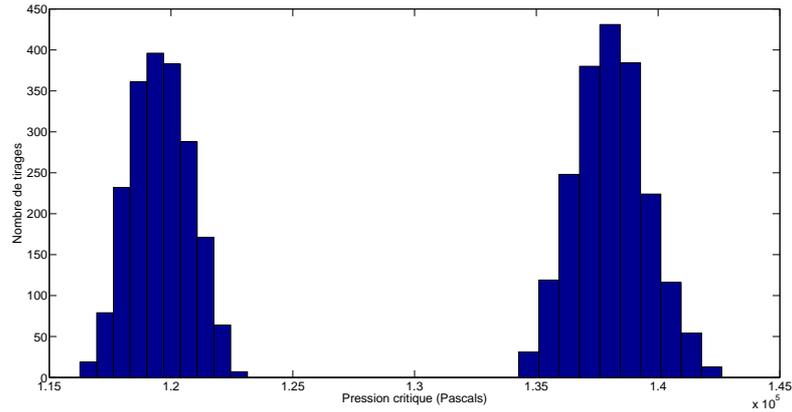


FIG. 4.18: Histogrammes des valeurs initiales et optimisées du modèle

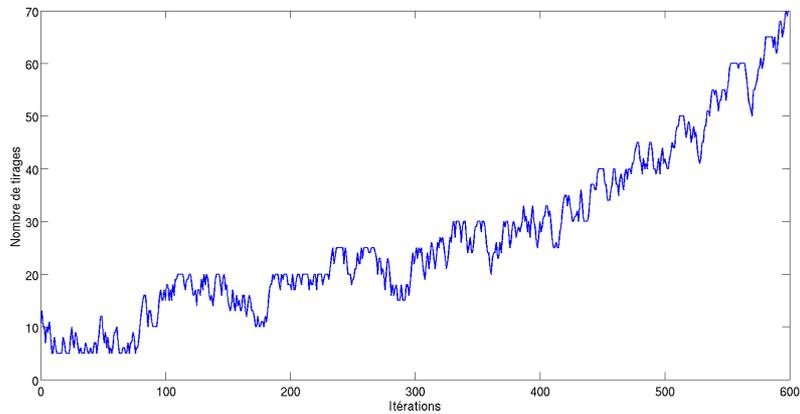


FIG. 4.19: Évolution du nombre de tirages pour l'évaluation de la fonction objectif

À gauche, nous avons tracé l'histogramme de la pression critique associée aux valeurs nominales du modèle, et à droite l'histogramme des valeurs obtenues par l'algorithme du recuit simulé stochastique. On constate ainsi, comme pour le gradient stochastique, un net déplacement sur la droite de l'histogramme. Toutefois, le coût numérique est bien plus important, comme le montre le nombre de tirages effectués à chaque itération de la boucle d'optimisation, représenté dans la figure (4.19).

L'algorithme a nécessité 16 241 fois la mise en œuvre de l'ensemble des calculs structure et aéro, pour calculer seulement 600 estimations de la pression critique. Il reste donc préférable d'employer la méthode du gradient quand celui-ci est disponible, et n'utiliser la méthode du recuit simulé qu'en dernier recours. Cependant, le gain de cet algorithme par rapport à un algorithme où le nombre d'échantillons est constant est sans appel : un tel algorithme aurait nécessité environ 42 000 analyses de flottement.

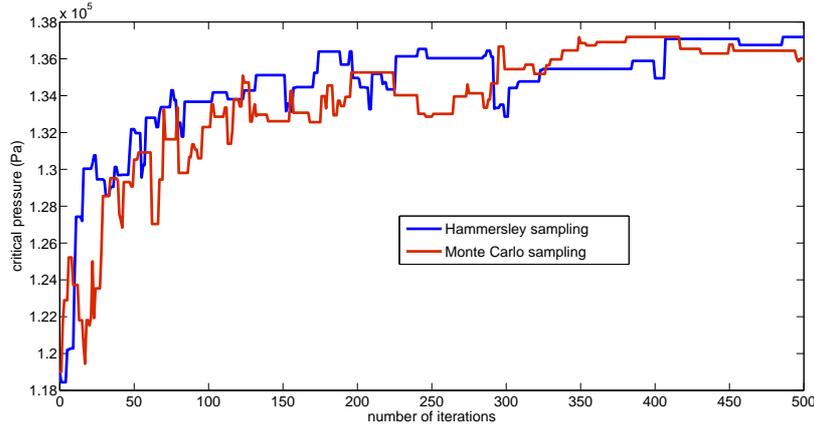


FIG. 4.20: Comparaison de l'évolution de la pression critique avec estimation par Monte-Carlo et par Hammersley

Masses nominales (kg)	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}
Monte-Carlo (kg)	0.6102	0.8218	0.7816	0.5327	0.3214	0.4123	0.6272	0.4435	0.4029	0.2412	0.1597
Hammersley (kg)	0.6015	0.7012	0.6233	0.5744	0.3792	0.3608	0.5235	0.4849	0.4580	0.2642	0.1945
	0.4577	0.6357	0.5862	0.4686	0.2741	0.3888	0.7385	0.4092	0.4715	0.2896	0.1996
Masses nominales (kg)	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}	m_{17}	m_{18}	m_{19}	m_{20}	m_{21}	m_{22}
Monte-Carlo (kg)	0.1645	0.1263	0.0978	0.0959	0.1469	0.1152	0.0819	0.0814	0.0815	0.0734	0.0647
Hammersley (kg)	0.1795	0.1480	0.1044	0.1199	0.1699	0.1440	0.0980	0.0954	0.0930	0.0917	0.0485
	0.2056	0.1579	0.1222	0.1097	0.1641	0.1440	0.1024	0.1017	0.1019	0.0917	0.0485

TAB. 4.3: Valeurs des masses avant et après optimisation

Échantillonnage par des suites d'Hammersley

Nous souhaitons à présent vérifier le gain que peut procurer l'utilisation de suites d'Hammersley pour l'estimation de la fonction objectif. Pour cela, nous avons résolu le même problème d'optimisation en utilisant successivement des suites d'Hammersley et un échantillon aléatoire pour générer les paramètres incertains. Nous avons représenté l'évolution de la pression critique du point courant dans la figure (4.20).

On ne constate pas une flagrante différence de comportement au niveau de l'évolution de la fonction objectif au cours des deux algorithmes. Ce qui est normal, puisque les deux techniques utilisent la même grandeur, la borne inférieure d'un intervalle de confiance de même probabilité associé à la même fonction objectif. Il n'y a pas non plus une différence significative sur les valeurs massiques finales trouvées le long de l'arête centrale, à l'exception de quelques points, comme le montre le tableau (4.3).

Là encore, une étude de propagation d'incertitudes a été conduite, similaire à celle menée précédemment. Nous avons tracé sur la figure (4.21) l'estimateur de la loi de la pression critique avant et après optimisation, afin de comparer plus finement les deux solutions. Nous avons également superposé l'estimateur obtenu précédemment afin de comparer les solutions données par un échantillon aléatoire et quasi aléatoire.

La différence majeure se situe au niveau du nombre de tirages à chaque itération comme on peut le visualiser sur la figure (4.22).

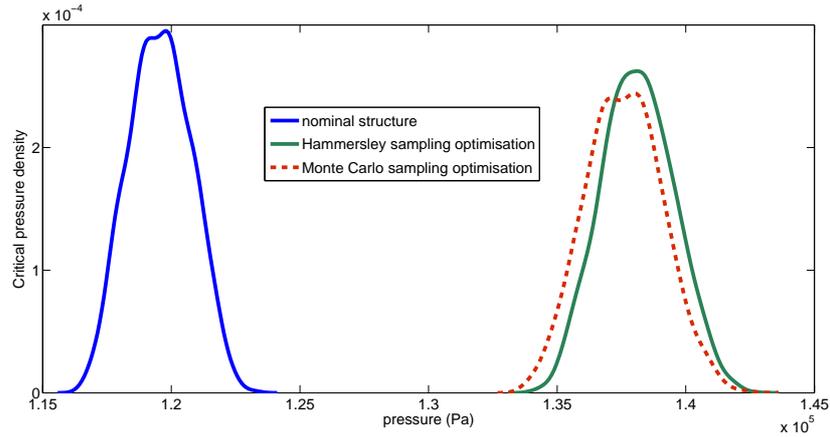


FIG. 4.21: Représentation de la distribution des densités de la pression critique

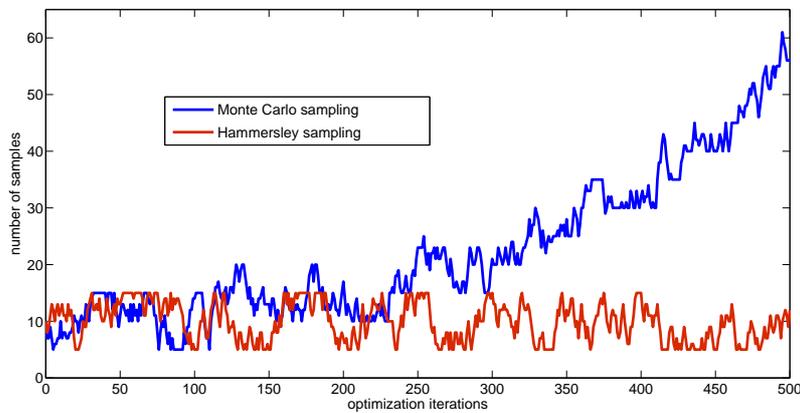


FIG. 4.22: Comparaison du nombre d'échantillons utilisés par Monte-Carlo et par Hammersley

On constate que le nombre de tirages pour la méthode d'estimation de Monte-Carlo est passée de 10 à 60 au cours de l'algorithme, alors que la méthode par suites d'Hammersley n'a pas encore atteint la limite des 5% pour le terme de pénalité associé. Par conséquent, le nombre minimal d'échantillons n'a pas été actualisé au cours de l'optimisation. Le gain ainsi obtenu en utilisant les suites d'Hammersley est de l'ordre de 50 à 70% dans les exemples traités.

Impact du choix du pourcentage pour le terme de pénalité

Dans nos précédents tests numériques, nous avons inclus une contrainte sur le terme de pénalité : celui-ci ne doit pas dépasser $p\%$ de la fonction objectif estimée. Les tests ci-dessus ont été conduits avec la valeur $p = 5$. Nous avons voulu constater quel était l'impact sur le comportement de l'algorithme d'un tel choix. Pour cela, nous avons réalisé trois tests numériques avec des valeurs de : $p = 2$, $p = 5$, $p = 10$. Afin de ne pas pénaliser ou avantager en temps de calcul une solution, nous avons modifié de façon proportionnelle la valeur

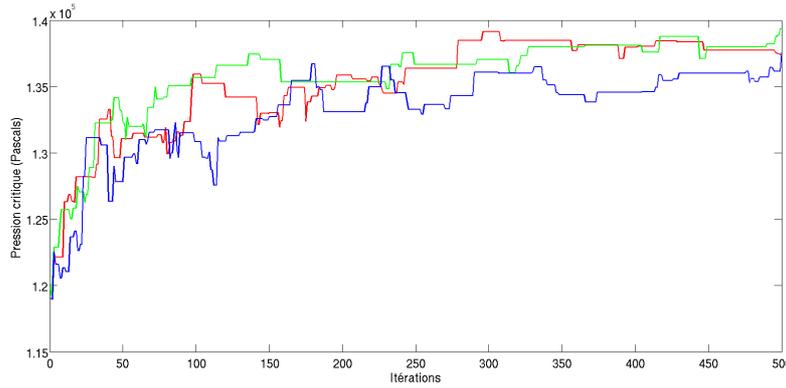


FIG. 4.23: Comparaison entre l'estimation selon le pourcentage maximal de pénalisation

$b_0\sigma_0$, afin que le terme $pb_0\sigma_0$ soit constant et que le nombre de tirages soit similaire pour chaque cas. Les autres constantes sont les mêmes que dans le premier test (4.18). Nous avons tracé sur la figure (4.23) l'évolution de la pression critique de la solution dans chacun des trois cas : en bleu pour la valeur $p = 10\%$, en rouge pour $p = 5\%$ et en vert pour $p = 2\%$.

Nous constatons que l'algorithme adopte globalement le même comportement avec les trois valeurs testées, même si la qualité de la solution semble meilleure lorsque p prend de petites valeurs. Toutefois, nous n'avons pas pu mettre en évidence de différences nettes au cours de nos calculs numériques.

4.4 Minimisation de la masse de la maquette

Nous changeons de problème à présent, nous souhaitons minimiser la masse totale de l'aile avec une contrainte en probabilité sur la pression critique. Il s'agit d'obtenir une solution telle que la pression critique $q_c(m, \xi(\omega))$ reste au-dessus d'un certain seuil avec une probabilité minimale donnée. La nature de cette contrainte est due au fait qu'avec une loi continue de support non borné, celle-ci ne peut être presque sûrement au-dessus d'une certaine borne. On est donc en pratique obligé de relâcher la contrainte idéale qui serait d'avoir la certitude que la solution apportée maintient la pression critique au-dessus d'une certaine borne.

Le problème posé est le suivant :

$$\begin{aligned} \min_{m \in \mathbb{R}^N} \sum_{i=1}^N m_i & \quad (4.19) \\ \text{s.c.} \quad & \begin{cases} \mathbb{P}[q_c(m, \xi(\omega)) > q_0] > p_0 ; 0 \leq p_0 \leq 1 \\ \forall i, m_i \in [(1 - \beta)m_{i,0}, (1 + \beta)m_{i,0}]. \end{cases} \end{aligned}$$

4.4.1 Utilisation de l'algorithme de Arrow-Hurwicz

L'algorithme utilisé ici sera l'algorithme de Arrow-Hurwicz stochastique modifié. On rappelle qu'il se présente sous la forme suivante :

1. initialiser $m^{(0)} \in \mathbb{R}^N$ aux valeurs initiales du modèle et $\lambda_0 = 0$,
2. choisir les constantes de départ, la forme des pas et la fonction d'interpolation h ,
3. à chaque itération, actualiser et projeter :

$$m^{(k+1)} = \text{proj}_{U^{ad}} \left(m^{(k)} - \epsilon_k \left(1_N - \nabla_m g_{r_k}(m^{(k)}, \xi(\omega_{k+1})) \lambda_k \right) \right), \quad (4.20)$$

4. poser $\lambda_{k+1} = \max \left(0, \lambda_k + \rho_k g_{r_k}(m^{(k+1)}, \xi(\omega_{k+1})) \right)$,
5. si un test de convergence est vérifié, stopper l'algorithme, sinon revenir en (3).

La fonction g_r est la fonction interpolée de la fonction h_r vue dans le chapitre 3 et de la contrainte en probabilité :

$$g_r(m, \xi(\omega)) = p_0 - (h_r * \mathbf{1}_{\mathbb{R}_+}) (q_c(m, \xi(\omega)) - q_0). \quad (4.21)$$

Le choix de la fonction h est donc déterminant pour le comportement global de cette méthode. Dans son article [5], l'auteur a comparé plusieurs fonctions candidates en fonction de leurs normes, selon un critère de vitesse de convergence et a conclu que la fonction

$$h(x) = \frac{3(1-x^2)}{4} \mathbf{1}_{[-1,1]}(x) \quad (4.22)$$

offrait la meilleure efficacité. C'est pourquoi nous avons retenu cette fonction dans nos calculs numériques. En posant :

$$K_r = \text{proj}_{[-1,1]} \left(\frac{q_c(m, \xi(\omega)) - q_0}{r} \right), \quad (4.23)$$

nous pouvons alors écrire :

$$g_r(m, \xi(\omega)) = p_0 - \frac{1}{4} (K_r^3 - 3K_r + 2), \quad (4.24)$$

$$\nabla_m g_r(m, \xi(\omega)) = -\frac{3}{4r} (1 - K_r^2) \nabla_m q_c(m, \xi(\omega)). \quad (4.25)$$

Pour la preuve de ces deux formules, nous renvoyons le lecteur aux annexes. Par la suite, nous avons remplacé les valeurs de g_r et $\nabla_m g_r$ par ces formules dans l'algorithme modifié de Arrow-Hurwicz :

$$m^{(k+1)} = \text{proj}_{U^{ad}} \left(m^{(k)} - \epsilon_k \left(1_N + \frac{3\lambda_k}{4r_k} (1 - K_{r_k}^2) \nabla_m q_c(m^{(k)}, \xi(\omega_{k+1})) \right) \right), \quad (4.26)$$

$$\lambda_{k+1} = \max \left(0, \lambda_k + \rho_k \left(p_0 - \frac{1}{4} (K_{r_k}^3 - 3K_{r_k} + 2) \right) \right). \quad (4.27)$$

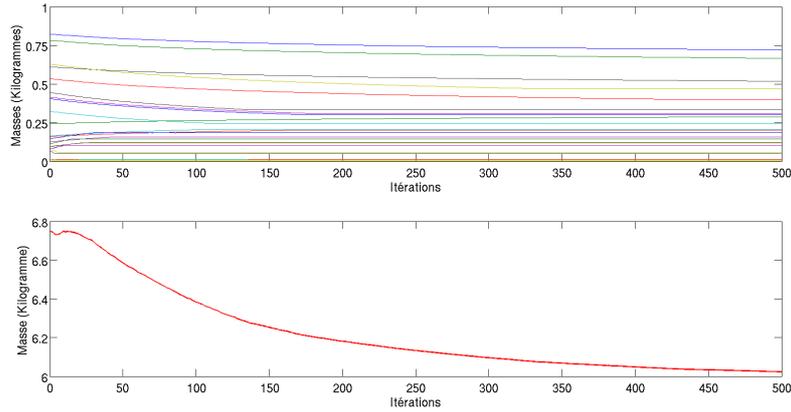


FIG. 4.24: Évolution des masses ponctuelles et de la masse totale au cours de l'algorithme

Choix des pas de calcul

Nous devons à présent choisir les suites de pas numériques de l'algorithme : ρ_k , ϵ_k et r_k . Dans son article [5], Andrieu suggère les formes suivantes :

$$r_k = \frac{R_1}{(R_2 + k)^{1/5}} ; \rho_k = \frac{R_3}{R_4 + k} ; \epsilon_k = \frac{R_5}{R_6 + k}. \quad (4.28)$$

Nous avons donc suivi les recommandations de l'auteur dans la suite des calculs.

Choix des paramètres

Pour cet algorithme, il y a de nombreuses constantes à déterminer. Les constantes des pas numériques sont en particulier délicates à régler, car l'algorithme est très instable numériquement. Dans certains cas, il a fallu effectuer une dizaine de tests numériques pour obtenir un jeu de paramètres efficace.

Dans notre premier exemple, nous avons choisi les valeurs numériques suivantes :

$$q_0 = 130\,000 ; p_0 = 0.99 ; \alpha = 0.25 ; \beta = 0.25 \quad (4.29)$$

$$R_1 = 0.01 ; R_2 = 0 ; R_3 = 20 ; R_4 = 0 ; R_5 = 0.002 ; R_6 = 100. \quad (4.30)$$

Avec ces valeurs numériques, nous avons fait fonctionner l'algorithme de Arrow-Hurwicz stochastique pendant 500 itérations.

La figure (4.24) montre que la fonction objectif décroît au cours de l'algorithme. On constate numériquement une baisse de 13% de la masse totale, en ayant autorisé un écart de 25% pour chaque point massique pour les solutions admissibles, ce qui est un résultat substantiel. Cependant, il nous faut étudier l'évolution de la pression critique pour vérifier que la contrainte en probabilité est bien vérifiée (4.25).

Comme pour le gradient stochastique, on voit ici une amélioration nette de la pression critique, qui passe largement au-dessus de la pression q_0 imposée. Cependant, cela n'est pas totalement suffisant pour nous assurer de la vérification de la contrainte. Encore

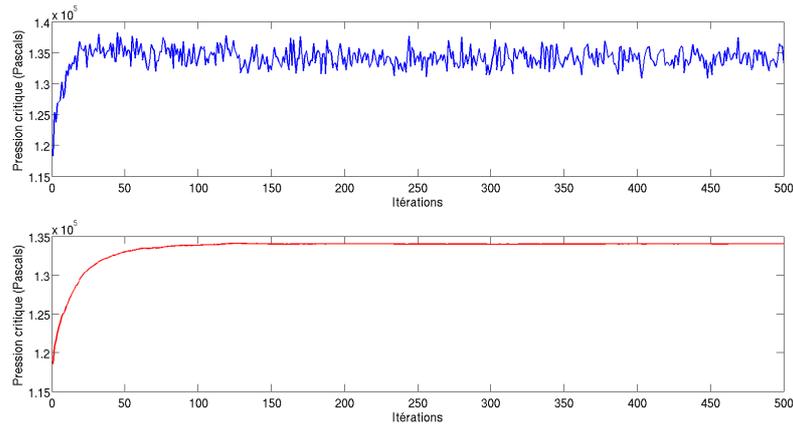


FIG. 4.25: Évolution de la pression critique et de sa moyenne de Cesaro au cours de l'algorithme

Masses nominales (kg)	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}
Masses optimisées (kg)	0.6102	0.8218	0.7816	0.5327	0.3214	0.4123	0.6272	0.4435	0.4029	0.2412	0.1597
	0.4577	0.6163	0.6066	0.4272	0.2435	0.4015	0.7618	0.5527	0.5036	0.3015	0.1996
Masses nominales (kg)	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}	m_{17}	m_{18}	m_{19}	m_{20}	m_{21}	m_{22}
Masses optimisées (kg)	0.1645	0.1263	0.0978	0.0959	0.1469	0.1152	0.0819	0.0814	0.0815	0.0734	0.0647
	0.2056	0.1579	0.1222	0.1199	0.1836	0.1440	0.1024	0.1017	0.1019	0.0917	0.0809

TAB. 4.4: Valeurs des points massiques avant et après optimisation

une fois, nous avons réalisé une étude de propagation d'incertitudes en utilisant successivement les valeurs initiales et optimales trouvées par l'algorithme, avec un total de 10 000 tirages par solution. Nous avons représenté les résultats obtenus dans la figure (4.26), avec à gauche l'histogramme des valeurs initiales, à droite l'histogramme des valeurs optimisées.

Ainsi, nous n'avons obtenu qu'un seul tirage ne vérifiant pas la contrainte sur 10 000 tirages. Nous pouvons donc affirmer que la contrainte à 99% est effectivement vérifiée par la solution donnée par l'algorithme. Nous avons également comparé les valeurs initiales et finales des masses dans le tableau (4.4).

Remarque 4.2. Temps de calcul

Les trois illustrations numériques présentées dans ce chapitre ont montré leur applicabilité au problème considéré. Toutes trois sont basées sur la détermination du paramètre critique, la pression dans la soufflerie dans notre cas, qui nécessite une analyse aérodynamique complète et un calcul de stabilité pour le calculer. Nous voyons que l'unité de mesure adéquate pour quantifier le temps de calcul de ces techniques d'optimisation est le temps nécessaire à une analyse de stabilité, qu'il suffira de multiplier par le nombre d'itérations pour obtenir le temps total.

En ce qui concerne les algorithmes basés sur le gradient, les temps de calcul sont les mêmes que dans le cas déterministe, sans incertitude, comme le montre la lecture de la figure (4.27). Sur un ordinateur personnel, le temps de calcul pour effectuer une analyse de flottement du modèle de l'aile Dyvas est de l'ordre de 4 secondes.

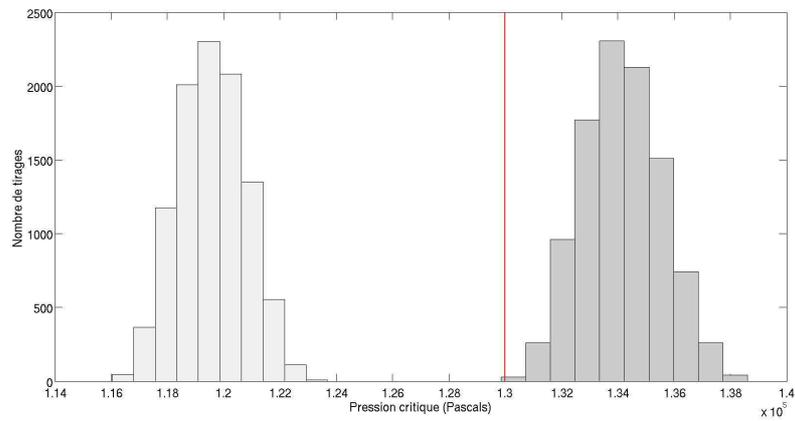


FIG. 4.26: Histogramme de la pression critique pour les valeurs initiales et celles données par l'algorithme

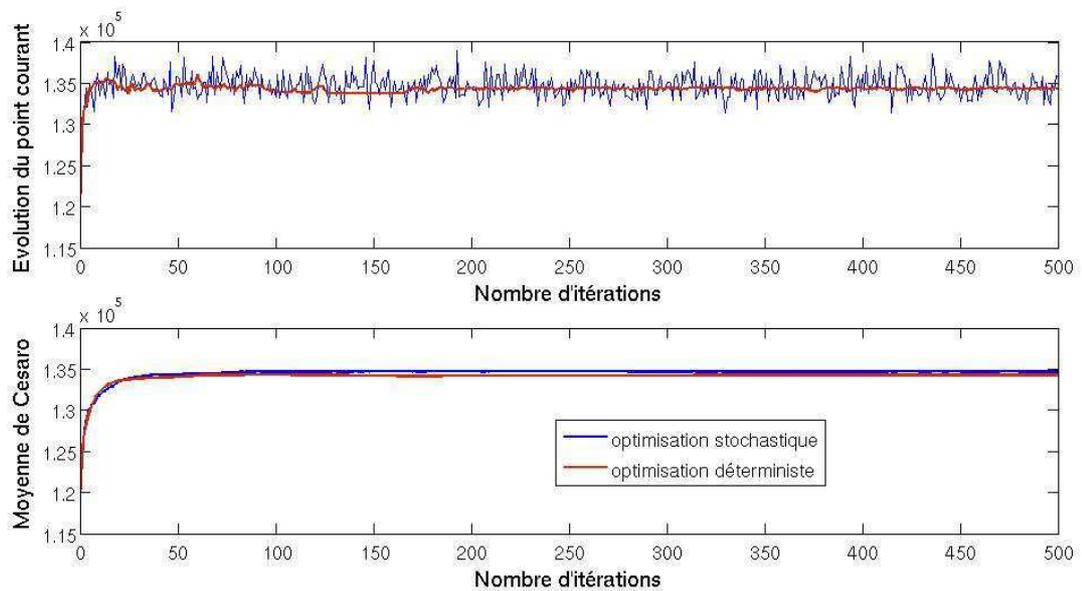


FIG. 4.27: Évolution de la pression critique au cours d'algorithmes du gradient déterministe et stochastique

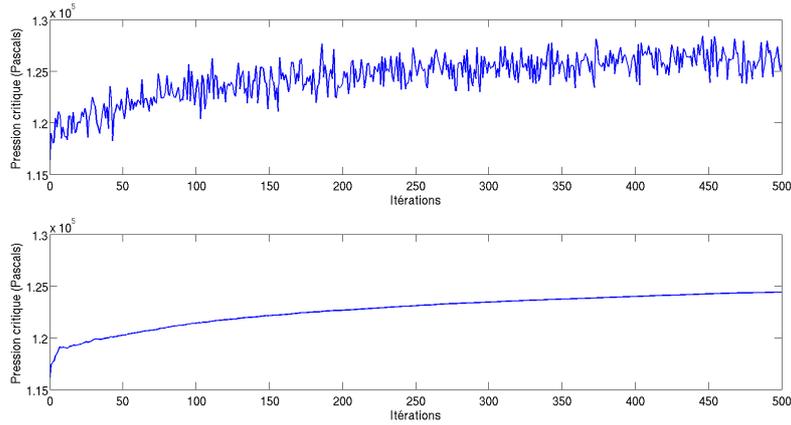


FIG. 4.28: Évolution de la pression critique et de sa moyenne de Cesaro

4.5 Cas pathologiques

Si les exemples que nous avons montrés plus haut donnent des résultats satisfaisants, cela n'a pas toujours aussi bien fonctionné. Il y a parfois une différence entre le comportement théorique d'une méthode et celui de son application numérique, à cause de problèmes informatiques ou de calibrage. En effet, les algorithmes manipulés dans ce chapitre sont itératifs et fonctionnent à l'aide de suites de pas numériques qui modulent l'importance de certaines quantités telles que le gradient. En cas de mauvais choix des constantes qui composent ces suites de pas, nous pouvons obtenir des résultats pathologiques. Nous allons donner quelques exemples de cas pathologiques représentatifs des comportements indésirables que l'on peut obtenir numériquement et que nous avons rencontrés au cours de nos simulations numériques. On souhaite éviter ce genre de résultats, et pour cela, on réalise des tests numériques préalables pour régler les algorithmes et afin d'en obtenir un comportement satisfaisant. Selon la complexité de l'algorithme, ces essais peuvent être nombreux et impacter l'efficacité globale de la méthode numérique.

4.5.1 Convergence trop lente de l'algorithme du gradient stochastique

On rappelle la forme choisie du pas numérique pour le gradient stochastique :

$$\epsilon_k = \frac{K_0}{(K_1 + k)\gamma}. \quad (4.31)$$

Ici, le principal défi est de régler la constante K_0 afin que le gradient de la fonction objectif et le vecteur des paramètres d'optimisation soient d'un ordre de grandeur suffisamment proche. Dans le cas où on choisirait une constante trop faible, on obtiendrait une convergence trop lente pour aboutir à une solution intéressante avant un laps de temps raisonnable, comme cela est montré dans la figure (4.28).

Ici, les paramètres étaient les mêmes que dans la figure (4.12), à l'exception de K_0 que l'on a divisé par 100. Le choix de K_1 peut aboutir à un résultat similaire. Si sa valeur est petite devant le nombre d'itérations total de l'algorithme, il y aura une grande disparité entre les pas numériques de début et de fin d'algorithme. Cela peut provoquer un ralentissement de

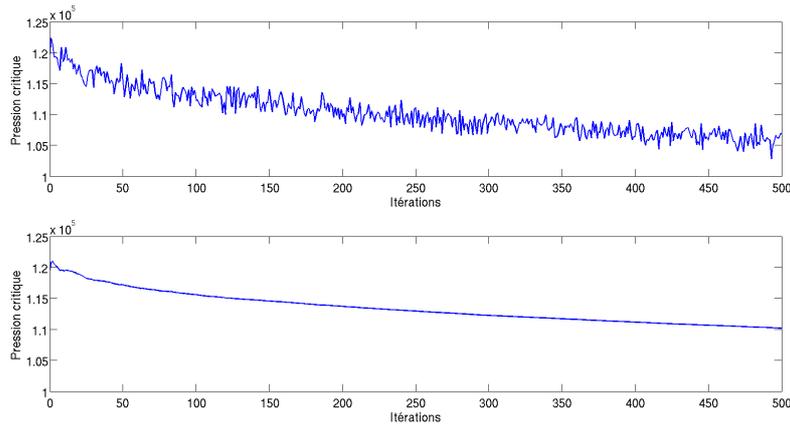


FIG. 4.29: Évolution de la pression critique et de sa moyenne de Cesaro

la convergence et donner un résultat proche de celui de la figure (4.28). Dans notre cas, nous avons trouvé des paramètres satisfaisants pour faire fonctionner l'algorithme après quelques essais.

4.5.2 Baisse de la pression critique dans l'algorithme de Arrow-Hurwicz stochastique

L'algorithme de Arrow-Hurwicz comportent trois suites de pas différentes et a pour objectif de concilier deux objectifs aux tendances opposées, la minimisation de la masse et l'augmentation de la pression critique. Cela rend cet algorithme assez délicat à régler, les résultats numériques peuvent totalement diverger de ce qui est théoriquement prévu. Rappelons que les suites de pas que nous avons utilisées étaient les suivantes :

$$r_k = \frac{R_1}{(R_2 + k)^{1/5}} ; \rho_k = \frac{R_3}{R_4 + k} ; \epsilon_k = \frac{R_5}{R_6 + k}. \quad (4.32)$$

r_k est le paramètre de convolution, ρ_k la suite de pas de λ_k et ϵ_k la suite de pas du gradient. Comme pour le gradient stochastique, les suites de pas ont pour objectif de ramener les éléments manipulés par l'algorithme à des ordres de grandeur similaires. Si la constante R_1 dans la suite de pas r_k est trop petite, l'approximation du gradient de la contrainte sera souvent ramenée à 0, et la masse sera diminuée sans que la pression critique soit optimisée par l'algorithme, puisque seul le gradient de la masse sera pris en compte dans l'actualisation des masses du modèle, comme le montre la figure (4.29).

Dans ce cas de figure, les paramètres d'optimisation choisis sont identiques à ceux de la figure (4.25), à l'exception de la constante R_1 qu'on a divisée par 100 également. La masse totale du modèle a diminué, mais la pression critique a diminué au lieu d'augmenter afin d'atteindre et de dépasser le seuil critique défini à 130 000 Pascals. Ainsi, la solution donnée par l'algorithme ne vérifie pas la contrainte en probabilité, et elle est donc non viable.

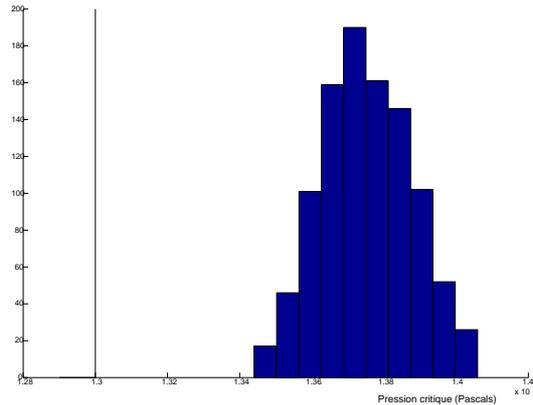


FIG. 4.30: Histogramme de la pression critique après optimisation

4.5.3 Surestimation de la contrainte dans l'algorithme de Arrow-Hurwicz stochastique

Un autre problème que nous avons souvent rencontré dans nos études numériques est la surestimation de la contrainte en probabilité. La seconde suite construite par l'algorithme de Arrow-Hurwicz a pour but d'influencer l'évolution de la première suite pour satisfaire la contrainte en probabilité. Mais pour des probabilités proches de 1, cette suite diminue lentement quand la contrainte est vérifiée, tandis qu'elle augmente rapidement dans le cas contraire. De plus, au début de nos études numériques, la contrainte n'est pas vérifiée et met un certain temps à atteindre le seuil voulu. Une fois ce seuil atteint, même initiée à 0, la suite λ_k prend de grandes valeurs numériques et ne baisse que lentement. Cela fait que même une fois que la contrainte est vérifiée, l'algorithme continue de poursuivre l'évolution en faveur de l'augmentation de la pression critique. Cela aboutit à une surestimation de cette quantité, alors qu'en théorie, la contrainte devrait être active une fois la solution du problème atteinte.

Sur la figure (4.30), on a représenté l'histogramme des valeurs finales de l'algorithme avec une contrainte à 99% sur la pression critique. Le seuil de la contrainte, fixé à 130 000 Pascals, est également représenté sur la figure. Les paramètres numériques utilisés sont les mêmes que précédemment, à l'exception de R_6 qui a été fixé à 0 et R_5 qui a été multiplié par 100 pour partir de la même valeur numérique. On constate grâce à cet histogramme que le support de la distribution de la pression critique s'est décalé nettement au-delà du seuil demandé, induisant une surévaluation de la contrainte. La solution exhibée reste proche de la solution finale trouvée précédemment, mais on est en droit de penser qu'il existe une meilleure solution, puisque la solution théorique doit vérifier la contrainte en probabilité par une égalité.

Chapitre 5

Conclusions et discussions

5.1 Apport de la thèse

Le but principal de cette thèse était d'apporter une première démarche de réflexion sur l'optimisation stochastique dans un contexte aéroélastique. Et d'en extraire quelques applications possibles et quelques résultats. Pour ce faire, nous avons apporté une formulation mathématique d'un problème d'optimisation stochastique dans un problème aéroélastique et une technique d'incorporation formelle des incertitudes dans une structure. Nous avons également pris en considération des contraintes potentielles pouvant intervenir en aéroélasticité quand la structure d'un appareil est trop instable. En l'occurrence, il s'agit du phénomène de flottement conduisant à la destruction de la structure. Les différents problèmes d'optimisation que nous avons étudiés reposaient en grande partie sur la considération de cette contrainte de ruine. La valeur du paramètre à laquelle se produit le phénomène de flottement, la valeur critique, a été soit la fonction objectif soit une contrainte dans les problèmes considérés.

Nous avons considéré l'ensemble des algorithmes stochastiques classiques existant dans la littérature, et nous en avons retenu quelques-uns selon plusieurs critères. Nous avons délibérément favorisé les algorithmes possédant à l'heure actuelle une preuve formelle de convergence vers la solution du problème qu'ils sont censés résoudre. Puisque le but de cette thèse était entre autres de poser un cadre formel, nous n'avons pas voulu explorer trop en détails des algorithmes dont les seules considérations suggérant la convergence finale sont empiriques.

Dans ce cadre, l'algorithme le plus classique et qui a été parfaitement disséqué par les mathématiciens depuis les années 50 est l'algorithme du gradient stochastique. Dans notre cas, nous avons eu à considérer un problème d'optimisation portant sur la vitesse critique. C'est pourquoi il nous a fallu, dans le but d'appliquer l'algorithme du gradient stochastique, exhiber une formule analytique du gradient du paramètre critique. Or, un tel paramètre correspond à une borne inférieure sur des pseudo-valeurs propres d'une équation polynomiale matricielle non linéaire, ce qui rend l'étude théorique de sa dérivation difficile. Toutefois, en faisant quelques hypothèses sur les pseudo-valeurs propres, on peut voir une relation entre les paramètres d'optimisation et la valeur critique. Cela a permis d'utiliser le théorème des fonctions implicites pour obtenir une formule analytique du gradient du paramètre critique en fonction des termes de l'équation aéroélastique standard. Ce résultat nous a été utile à plusieurs reprises par la suite.

Un autre problème que nous avons dû résoudre concerne la projection du gradient stochastique. Ce dernier fonctionne avec une projection sur l'espace des solutions admissibles, et selon les contours d'un tel espace, cette projection est réalisable ou non de manière pratique. Le problème vient du fait qu'on sait parfaitement définir une projection sur un espace compact mathématiquement, mais qu'il est en pratique assez difficile de construire un processus permettant de construire une telle projection. Les seuls cas où cela peut se faire de manière automatique sont la projection sur un intervalle dans \mathbb{R} ou sur un sous-espace affine. Dans notre cas d'étude, l'espace des solutions admissibles est une combinaison de ces deux types d'espaces. À l'aide d'un algorithme et en considérant une norme associée à un produit scalaire vérifiant quelques hypothèses, nous avons pu construire de façon automatique une suite finie dont le dernier terme est la projection du point initial sur l'espace des solutions admissibles.

Ceci ne concerne cependant que les problèmes où la fonction objectif est directement le paramètre critique. En général, les industriels considèrent davantage des problèmes d'optimisation où le paramètre critique apparaît comme une contrainte. À cause de la nature stochastique du problème considéré, nous avons cherché une manière d'intégrer à un problème d'optimisation classique une contrainte en probabilité sur le paramètre critique. Nous avons alors étudié une version stochastique de l'algorithme d'Arrow-Hurwicz qu'on peut voir comme une extension de l'algorithme du gradient. Le problème considéré est l'optimisation d'une fonction objectif soumise à une contrainte non triviale, ne permettant pas de construire l'espace des solutions admissibles. En vertu de Kuhn-Tucker, la solution est une coordonnée d'un point-selle du lagrangien du système. L'algorithme consiste alors à construire deux suites imbriquées convergeant vers le point-selle en question. Cet algorithme nécessite néanmoins de pouvoir calculer le gradient de la contrainte. Or, dans le cas d'une contrainte en probabilité, il est impossible de faire une interversion intégrale-dérivée, car alors, la fonction à dériver comporte une indicatrice de l'espace \mathbb{R}_+ , une fonction non dérivable. L'idée présentée par un papier récent consiste à approcher la valeur de la contrainte, puis de dériver cette quantité et de l'inclure à la place de la contrainte dans la construction de l'algorithme. En faisant converger cette quantité vers la valeur exacte simultanément qu'on construit la paire de suites, et sous certaines hypothèses, cette méthode permet de faire converger la première suite construite vers la solution de notre problème.

Dans notre étude, nous avons également supposé que dans certains cas, le gradient de la fonction objectif du problème n'était pas disponible. Soit parce qu'il n'existe pas de formule analytique du gradient, soit parce que celle-ci est obtenue par un code dit "boîte noire", et qu'il est donc impossible d'en tirer cette information. Dans de telles situations, il faut résoudre le problème en utilisant des méthodes alternatives, plus coûteuses mais implémentables informatiquement. Dans cette optique, nous avons réalisé des études numériques sur le recuit simulé stochastique. Cet algorithme ne requiert pas l'usage du gradient et permet donc de se passer de cette information. Le principe de l'algorithme consiste à explorer de manière aléatoire l'espace des solutions admissibles, en comparant la valeur de la fonction objectif en le point courant et un point candidat. La spécificité de cet algorithme est d'accepter de se déplacer au point candidat avec une probabilité non nulle même quand le point candidat est moins bon que le point courant au sens de la fonction objectif. Cela permet d'éviter à l'algorithme de se retrouver bloqué dans un minimum local et non global. Dans le cas du recuit simulé stochastique, la fonction objectif est aléatoire,

et la comparaison entre les points courant et candidat se fait sur une estimation de la fonction objectif, auquel on a adjoint un terme de pénalité pour tenir compte de la fiabilité des estimations effectuées. De cette façon, l'algorithme permet la construction d'une suite convergant vers la solution du problème posé.

À l'aide des simulations numériques effectuées dans notre étude au chapitre 4, nous avons ainsi pu constater l'efficacité des algorithmes que nous avons implémentés. Nous avons constaté une bonne efficacité de l'algorithme du gradient stochastique appliqué à un problème de maximisation du paramètre critique sur l'aile Dyvas, la suite construite convergant rapidement vers une solution apportant un comportement nettement plus stable que les valeurs nominales du modèle. Sur ce même problème, nous avons également pu faire fonctionner avec une relative efficacité le recuit simulé stochastique, avec ou sans suites d'Hammersley. Cependant, la différence de temps de calcul entre les deux méthodes désavantage grandement le recuit simulé stochastique, d'autant plus quand le calcul de la fonction objectif devient important, et cet algorithme ne doit donc être utilisé qu'en dernier recours. Il reste néanmoins intéressant quand le problème n'apporte pas d'informations sur la nature de la fonction objectif, et donc sur l'existence éventuelle du gradient de la fonction objectif.

La minimisation de la masse est un problème d'optimisation plus intéressant pour les industriels, c'est pourquoi nous avons porté sur lui une grande attention. Au cours de nos tests numériques, à l'aide de l'algorithme d'Arrow-Hurwicz stochastique, nous avons pu nettement diminuer la masse de l'aile tout en améliorant de façon significative le paramètre critique. Le temps de calcul total est en outre équivalent au cas du gradient stochastique, ce qui est normal puisque les deux algorithmes ont une approche similaire.

Notons que ces trois algorithmes fonctionnent à l'aide de suites itératives, un terme additif est ajouté à chaque élément de la suite pour construire l'élément suivant. Cela nécessite d'utiliser des suites de pas numériques permettant de moduler le comportement des algorithmes. Comme nous l'avons vu, de mauvais choix de paramètres pour ces suites de pas peuvent aboutir à des comportements singuliers et ne pas converger vers la solution du problème, voire vers une solution acceptable. À chaque nouveau problème considéré, il est nécessaire de recalibrer les valeurs numériques choisies pour ces suites de pas, ce qui peut prendre parfois jusqu'à une dizaine de tests.

5.2 Difficultés rencontrées non résolues

Malgré quelques avancées significatives, plusieurs écueils sont à déplorer. Le premier est l'absence de preuves de la différentiabilité de la vitesse critique. Cela s'explique par la manipulation de grandeurs telles que des solutions d'équations polynomiales, sur lesquelles il est difficile d'obtenir des hypothèses permettant d'obtenir leur différentiabilité. Nous n'avons pas pu déterminer d'autre part si les hypothèses d'application des algorithmes étudiés étaient vérifiées ou non, à cause de la complexité du problème.

Dans le cadre du recuit simulé stochastique, nous avons préféré ne pas suivre à la lettre les hypothèses de convergence de l'algorithme. En les respectant, la vitesse de convergence est logarithmique, ce qui est trop lent pour un industriel. Nous avons donc choisi d'accélérer cette convergence, en effectuant des tirages non uniformes sur le voisinage des points

courants, mais des tirages gaussiens. Il s'agit du Fast-Simulated Annealing, qui converge plus vite en pratique, mais sans assurance théorique de convergence.

Un autre problème d'ordre pratique a été la pertinence de notre modèle numérique. Il s'agit d'un modèle à éléments finis en arête de poisson et à masses concentrées. Il ne comporte que 97 points massiques et 93 poutres de raccord. De plus, les masses et les raideurs sont décorréélées. Ce n'est pas nécessairement un problème car le modèle est suffisamment réaliste pour de premières études d'optimisation stochastique dans le cadre d'études avant-projet. Cela nous a également permis de simplifier la visualisation du problème, car la masse totale du système pouvait être représentée en un seul vecteur. Cependant, ce n'est généralement pas ce genre de modèle numérique qui est utilisé par les industriels. De plus, ce genre d'optimisation se fait plutôt sur les épaisseurs de plaque, et cela a une incidence sur les raideurs. Par manque de modèle réaliste, nous n'avons pas pu réaliser de tests numériques dans des conditions plus proches de l'industrie.

5.3 Perspectives

Il est apparu au cours de la thèse que plusieurs voies sont possibles pour raffiner les résultats déjà obtenus. Une première avancée significative serait d'étudier plus en profondeur les propriétés des solutions de l'équation aéroélastique standard, et essayer de satisfaire les hypothèses nécessaires à la convergence des algorithmes d'optimisation. Le tout afin de donner une assise mathématique plus solide, avant de commencer à explorer des voies plus "exotiques".

Les méthodes d'optimisation stochastiques exposées dans cette thèse ne sont pas les seules existantes. Une étude plus large et approfondie de ces techniques sera à effectuer dans des études ultérieures. Quelques pistes possibles ont été évoquées. Notons par exemple que dans les algorithmes évoqués mais non étudiés, les algorithmes génétiques semblent être de bons candidats. Ils sont très étudiés depuis quelques années, on peut les décliner sous une forme stochastique, et surtout, il existe également quelques résultats de convergence.

Nous pourrions également essayer de considérer un nouveau type de problème. Comme cela a été évoqué plus tôt, c'est sur la base d'emploi de matériaux composites que notre introduction d'incertitudes dans le modèle s'est principalement reposée. Il s'agit de la conséquence de leur construction, à savoir des empilements de couches de différentes natures. Ces matériaux n'étant pas homogènes, les disparités provoquent des variations plus ou moins importantes dans les propriétés physiques si la technique d'empilement n'est pas précise. Il est possible de réduire ces variations, mais plus la marge d'erreur est faible, plus cela coûte cher. Il serait possible d'étudier un problème multi-critère faisant intervenir le prix des matériaux composites employés et la fiabilité de l'appareil. D'un point de vue plus large, l'objectif à long terme est d'incorporer les méthodes d'optimisation stochastique à des problèmes industriels multi-critères portant sur la globalité de l'appareil.

Dans ces études ultérieures, il sera intéressant de réaliser les mêmes tests numériques sur des modèles plus poussés, tout en prenant garde à l'aspect temporel, les forces aéro-dynamiques de certains modèles numériques mettent plusieurs minutes à être calculées.

Annexe A

Projection sur un espace convexe non trivial

Nous expliciterons ici la méthode utilisée pour réaliser une projection sur l'espace des fonctions admissibles dans le premier problème considéré dans cette thèse (4.6). Théoriquement, la projection sur un espace convexe dans un espace métrique complet est parfaitement définie. Dans la pratique, calculer l'exact position de la projection pour un convexe quelconque est quelque chose de difficile. En effet, il n'existe que très peu de méthodes de projection effectives. Les seuls espaces sur lesquels la projection est connue sont les hyperplans et les demi-plans. Pour la première catégorie, il suffit de calculer un vecteur orthogonal à l'hyperplan et de rajouter au point qu'on cherche à projeter la composante correspondante de telle sorte que le vecteur translaté par le vecteur orthogonal appartienne au plan. Cela peut s'écrire très simplement.

Soit E un espace préhilbertien de dimension finie et H un hyperplan tel que $x = (x_1, \dots, x_n)$ appartient à l'hyperplan si et seulement si $\langle a, x \rangle = c$, avec $a = (a_1, \dots, a_n)$ un vecteur non nul et c un réel quelconque. Un vecteur orthogonal à l'hyperplan H est alors le vecteur $a = (a_1, \dots, a_n)$. En effet, si x et y sont des vecteurs de H , le vecteur $x - y$ vérifie : $\langle a, x - y \rangle = 0$, donc a est orthogonal à tous les vecteurs de l'hyperplan affine H . Or, la projection orthogonale d'un vecteur sur un hyperplan affine vérifie une certaine propriété : $x - p_H(x)$ est orthogonal à $p_H(x) - y$, quel que soit y dans H . Donc, étant donné que H est un hyperplan, par définition, son complémentaire orthogonal est un sous-espace vectoriel de dimension 1, donc il s'agit de la droite vectoriel dirigée par le vecteur a . Ainsi, $p_H(x) = x + \lambda a$. Pour déterminer λ , il suffit de savoir que $\langle a, p_H(x) \rangle = c$. Ainsi, on calcule très facilement que :

$$\lambda = \frac{c - \langle a, x \rangle}{\langle a, a \rangle}. \quad (\text{A.1})$$

Dans le deuxième cas de projection, il s'agit de la généralisation de la projection dans \mathbb{R} sur un intervalle, borné ou non. Dans le cas de la projection sur l'intervalle $[a, b]$, cela peut s'écrire en informatique très simplement, sous la forme : $p(x) = \min(\max(x, a), b)$. Cela se généralise à un espace vectoriel de dimension finie, et ce quelle que soit la norme associée à un produit scalaire diagonal, en appliquant le principe précédent composante par composante.

Si l'on excepte ces 2 types d'espace de projection, on ne sait pas en général résoudre ce problème de manière effective. Cependant, en combinant les propriétés énoncées ci-dessus,

on peut parvenir à mettre en place un procédé de projection sur un espace hybride. Dans notre situation, nous souhaitons construire la projection d'un point massique de notre modèle à éléments finis, de telle sorte que la masse reste constante, et que chaque coordonnée reste dans un intervalle borné autour des coordonnées initiales. Cet ensemble peut s'écrire sous la forme : $K = \{x \in \mathbb{E} \mid \sum x_i = c, x_i \in [a_i, b_i] \forall i\}$. Comme on peut le voir, K est ici l'intersection des deux types d'espaces précédemment évoqués. La solution de la projection passera par une combinaison des techniques de projection déjà étudiées. Avant cela, on a besoin d'une propriété de la projection.

Proposition A.1. *Soit E un espace préhilbertien de dimension n . On appelle sous-espace affine toute translation d'un sous-espace vectoriel de E . On suppose que l'ensemble convexe K de E est contenu dans le sous-espace affine H , qui est en particulier convexe. Alors :*

$$p_K(x) = p_K(p_H(x)). \quad (\text{A.2})$$

Démonstration. Par définition de la projection,

$$\forall y \in K, \|x - y\| \geq \|x - p_K(x)\|,$$

et $p_K(p_H(x))$ appartient à K , d'où une première inégalité. On sait également que la projection sur H étant orthogonale, on a la relation $x - p_H(x) \perp p_H(x) - y$, quel que soit $y \in H$. C'est en particulier pour $y = p_K(p_H(x))$, puisque K est inclus dans H . Cela signifie donc que

$$\|x - p_K(x)\|^2 = \|x - p_H(x)\|^2 + \|p_H(x) - p_K(x)\|^2,$$

par orthogonalité entre les deux vecteurs. Or, toujours par définition,

$$\|p_H(x) - p_K(x)\| \geq \|p_H(x) - p_H(p_K(x))\|.$$

En intégrant cette inégalité dans la formule précédente et en remontant, on obtient la deuxième inégalité. On a donc l'égalité suivante :

$$\|x - p_K(x)\| = \|x - p_H(p_K(x))\|. \quad (\text{A.3})$$

Or, nous savons que le point de K qui minimise cette quantité est unique, donc par unicité, nous avons bien : $p_K(x) = p_K(p_H(x))$. \square

On notera aussi qu'il n'est pas nécessaire que K soit entièrement inclus dans H . Il suffit en effet que la projection de x sur H appartienne également à K . Dans ce cas, il suffit de voir que la projection sur H est équivalente à la projection sur $K \cap H$ dans le cas du vecteur x , puis d'effectuer la projection finale sur $K \cap H$.

Ce résultat nous apprend que sous certaines conditions, nous pouvons effectuer des projections intermédiaires sans affecter le résultat final. Il suffit pour cela que la ou les projections intermédiaires se fassent sur des sous-espaces affines contenant la projection du point x . Toute la démonstration consiste à déterminer quels sont ces espaces affines.

Théorème A.2. *Algorithme de projection :*

Soit E un espace préhilbertien de dimension finie n . On considère l'ensemble non vide

$$K = \{x \in E \mid \sum_{i=1}^n x_i = c, x_i \in [\alpha_i, \beta_i]\} \quad (\text{l'hypothèse non vide est vérifiée si } \sum_{i=1}^n \alpha_i \leq c \leq \sum_{i=1}^n \beta_i),$$

et on souhaite réaliser la projection du point x sur cet ensemble. Appelons x^ la projection du point x . On réalise les étapes suivantes :*

- On pose $x^{(1)} = x + \frac{c - \langle a, x \rangle}{n}$ avec $a_i = 1, \forall i$. On pose $H^{(1)} = \{x \in E \mid \sum_{i=1}^n x_i = c\}$.
- On suppose $x^{(k)}$ construit et que x^* n'est pas encore atteint. On pose alors $K^+ = \{i \mid x_i > \beta_i\}$, $K^- = \{i \mid x_i < \alpha_i\}$ et $K^* = \{i \mid \alpha_i \geq x_i \geq \beta_i\}$. On vérifie alors si $\sum_{K^-} (\alpha_i - x_i^{(k)}) > \sum_{K^+} (x_i^{(k)} - \beta_i)$. Si c'est le cas, on pose $H^{(k+1)} = \{x \in H^{(k)}, \forall i \in K^-, x_i = \alpha_i\}$. Dans le cas contraire, on pose $H^{(k+1)} = \{x \in H^{(k)}, \forall i \in K^+, x_i = \beta_i\}$. On effectue alors la projection sur $H^{(k+1)}$.

Ainsi, cet algorithme construit une suite qui converge vers x^* en au plus n étapes.

Démonstration. Le principe de l'algorithme est de construire une suite de telle sorte que les projections des points de cette suite sur les ensembles $H^{(k)}$ restent le point recherché.

En effet, en application de la précédente propriété, le point $x^{(1)}$ projeté sur $H^{(1)}$ possède la même projection que x sur l'ensemble K .

Par la suite, on suppose le point $x^{(k)}$ construit et qu'il appartient à $H^{(k)}$ qui est inclus dans $H^{(1)}$. On vérifie alors si ses coordonnées violent les contraintes. Si ce n'est pas le cas, $x^{(k)}$ est dans K , donc $p_K(x^{(k)}) = x^{(k)}$, donc en remontant la chaîne, $x^{(k)} = x^*$, et l'algorithme est terminé. Dans le cas contraire, les ensembles K^+ et K^- contiennent au moins un élément. On réalise le test suivant H1 : $\sum_{K^-} (\alpha_i - x_i^{(k)}) > \sum_{K^+} (x_i^{(k)} - \beta_i)$. Si le test est vrai, on démontre que x^* vérifie que, $\forall i \in K^-, x_i^* = \alpha_i$.

En effet, supposons qu'il existe au moins une coordonnée $i_0 \in K^-$ telle que $x_{i_0}^* > \alpha_{i_0}$. Cela implique qu'il existe une coordonnée $j_0 \in K^+ \cup K^*$ telle que $x_{j_0}^* < \beta_{j_0}$ ou $x_{j_0}^* < x_{j_0}^{(k)}$. En effet, sinon, par construction, $x^{(k)}$ appartient à $H^{(k)}$, donc à $H^{(1)}$, donc :

$$\begin{aligned}
0 &= c - c \\
&= \sum x_i^{(k)} - \sum x_i^* \\
&= \sum_{i \in K^-} (x_i^{(k)} - x_i^*) + \sum_{i \in K^*} (x_i^{(k)} - x_i^*) + \sum_{i \in K^+} (x_i^{(k)} - x_i^*) \\
&\geq \sum_{i \in K^-} (x_i^{(k)} - x_i^*) + \sum_{i \in K^+} (x_i^{(k)} - x_i^*) \text{ car } x_i^{(k)} \geq x_i^* \text{ sur } K^* \\
&\geq \sum_{i \in K^-} (x_i^{(k)} - x_i^*) + \sum_{i \in K^+} (x_i^{(k)} - \beta_i) \text{ car } x_i^* = \beta_i \text{ sur } K^+ \\
&\geq \sum_{i \in K^-} (x_i^{(k)} - \alpha_i) + \sum_{i \in K^+} (x_i^{(k)} - \beta_i) \text{ car } \alpha_i \leq x_i^* \text{ sur } K^+ \\
&> 0 \text{ (hypothèse H1) : contradiction.}
\end{aligned}$$

Par l'absurde, on prouve l'existence de la coordonnée j_0 . On a de fait deux quantités strictement positives : $x_{i_0}^* - \alpha_{i_0}$ et $x_{j_0}^* - x_{j_0}$. On pose donc $m = \min(x_{i_0}^* - \alpha_{i_0}, x_{j_0}^* - x_{j_0})$, et on construit le point y^* tel que :

$$\begin{aligned}
y_{i_0}^* &= x_{i_0}^* - m \\
y_{j_0}^* &= x_{j_0}^* + m \\
y_i^* &= x_i^* \text{ sinon.}
\end{aligned}$$

Alors, par les relations précédentes, on peut vérifier que $y^* \in K$. Les composantes ont été modifiées de façon à ce que les contraintes soient vérifiées, et la somme totale des coordonnées reste constante. Pour tous les produits scalaires associés à une matrice diagonale,

on déduit très facilement que $\|y^* - x^{(k)}\| < \|x^* - x^{(k)}\|$, car toutes les coordonnées sont identiques, sauf deux qui sont plus petites en valeur absolue (ce résultat est faux pour un produit scalaire quelconque). Or, ceci aboutit à une contradiction, car x^* est en particulier la projection de $x^{(k)}$ sur K , et on a construit un point dans K plus proche de $x^{(k)}$ que x^* .

Par l'absurde, on a démontré que $\forall i \in K^-, x_i^* = \alpha_i$ (dans le cas où le test H1 est faux, le raisonnement fonctionne de manière symétrique pour démontrer que $\forall i \in K^+, x_i^* = \beta_i$). On pose alors $H^{(k+1)} = \{x \in H^{(k)}, \forall i \in K^-, x_i = \alpha_i\}$. Par la propriété démontrée précédemment, on sait que $x^* = p_K(x) = p_K(x^{(k)}) = p_K(p_{H^{(k+1)}}(x^{(k)}))$. On pose alors $x^{(k+1)} = p_{H^{(k+1)}}(x^{(k)})$. Ainsi, le nouveau point $x^{(k+1)}$ vérifie toujours la propriété suivante : $p_K(x) = p_K(x^{(k+1)})$.

On remarque aussi qu'à chaque étape, le principe est de fixer au moins une des coordonnées libres du point $x^{(k)}$. Ainsi, si les contraintes ne sont pas toutes vérifiées, on fixe au moins l'une des coordonnées aux bornes admises, et on modifie les coordonnées libres en conséquence. Cela signifie qu'à chaque étape, on diminue au moins d'un le nombre de coordonnées "libres" du point à projeter. Comme il n'y a que n coordonnées, l'algorithme ne peut compter que $n-1$ étapes au maximum.

Quand l'algorithme ne peut plus réaliser la construction du sous-espace affine suivant, cela peut signifier deux choses. La première, c'est que toutes les coordonnées vérifient les contraintes, donc que le point construit est dans l'ensemble K , et donc, on a construit la projection du point x . La seconde solution, c'est que les coordonnées "libres" violent toutes les contraintes de la même façon, en étant supérieures ou inférieures aux valeurs extrémales autorisées. Il ne serait donc pas possible de ramener ces valeurs dans les bornes autorisées tout en gardant les autres coordonnées inchangées et en gardant la somme des coordonnées constante. Mais cela impliquerait que certaines projections n'étaient pas correctes au cours de l'algorithme, ou que le point x ne possède pas de projection sur l'ensemble K . Or, tout ceci est impossible, car ces résultats ont été prouvés ci-dessus, car la projection existe, et si les sous-espaces affines sont choisis comme le veut l'algorithme, les projections construisent la bonne suite, comme cela vient d'être démontré. Donc, quand l'algorithme termine, on se retrouve forcément avec la projection du point x . L'algorithme converge vers la bonne valeur en au plus n étapes de projection. \square

Dans le cadre d'une projection orthogonale, on dispose d'outils permettant de faciliter les calculs numériques. Lorsque l'on projette selon le produit scalaire usuel sur un ensemble de type $H = \{x \in E | \langle a, x \rangle = c\}$, un vecteur orthogonal à cet hyperplan est a . Pour des raisons pratiques, on peut être amené à privilégier une certaine direction de descente, c'est pourquoi on construit le produit scalaire $\langle \cdot, \cdot \rangle_A$ avec :

$$A = \begin{pmatrix} \frac{1}{a_1} & & (0) \\ & \ddots & \\ (0) & & \frac{1}{a_n} \end{pmatrix} \quad (\text{A.4})$$

Ainsi, si on pose l'hyperplan $H = \{x \in E | \sum x_i = c\}$, celui-ci peut se réécrire sous la forme $H = \{x \in E | \langle a, x \rangle_A = c\}$. Donc la direction de descente sera le vecteur a lors de la projection orthogonale sur l'hyperplan H . Les projections sur les sous-espaces affines intermédiaires doivent également se faire selon le produit vectoriel associé.

Annexe B

Rappels en probabilités

B.1 Mesure et espérance

Définition B.1.1. Soit Ω un espace quelconque et $\mathcal{A} \subset \mathcal{P}(\omega)$ un sous-ensemble de l'ensemble des partitions de Ω . On dit que \mathcal{A} est une tribu de Ω si :

1. $\emptyset \in \mathcal{A}$,
2. $\forall A \in \mathcal{A}, {}^c A \in \mathcal{A}$,
3. $\forall A_1, A_2, \dots \in \mathcal{A}, \bigcup_{i=1}^{+\infty} A_i \in \mathcal{A}$

Définition B.1.2. On dit qu'un espace (Ω, \mathcal{A}) est un espace mesurable si \mathcal{A} est une tribu de Ω .

Définition B.1.3. On dit qu'une application $\mu : \mathcal{A} \mapsto \mathbb{R}$ est une mesure sur l'espace mesurable (Ω, \mathcal{A}) si :

$$\forall A \in \mathcal{A}, 0 \leq \mu(A) \leq 1, \quad (\text{B.1})$$

$$\forall A_1, A_2, \dots \in \mathcal{A} \text{ deux à deux disjoints, } \mu \left(\bigcup_{i=1}^{+\infty} A_i \right) = \sum_{i=1}^{+\infty} \mu(A_i). \quad (\text{B.2})$$

Si en outre, $\mu(\Omega) = 1$, alors la mesure est appelée une mesure de probabilité.

Définition B.1.4. On dit qu'un espace $(\Omega, \mathcal{A}, \mu)$ est un espace mesuré si (Ω, \mathcal{A}) est un espace mesurable et si μ est une mesure sur (Ω, \mathcal{A}) . Si en outre, la mesure μ est une mesure de probabilité, on dit que l'espace $(\Omega, \mathcal{A}, \mu)$ est un espace probabilisé.

Définition B.1.5. Soient ω et F des espaces mesurables munis de leurs tribus respectives \mathcal{A} et \mathcal{F} . Une fonction $f : E \mapsto F$ est dite $(\mathcal{A}, \mathcal{F})$ -mesurable si la tribu image réciproque par f de la tribu \mathcal{F} est incluse dans \mathcal{A} :

$$\forall B \in \mathcal{F}, f^{-1}(B) \in \mathcal{A}. \quad (\text{B.3})$$

Définition B.1.6. Soient $(\Omega, \mathcal{A}, \mathbb{P})$ 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 E .

Définition B.1.7. On appelle l'ensemble des boréliens $\mathcal{B}(\mathbb{R})$ la plus petite tribu de $\mathcal{P}(\mathbb{R})$ contenant l'ensemble des intervalles de \mathbb{R} . Ainsi, $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$ est un espace mesurable.

Définition B.1.8. Une fonction f à valeurs de Ω dans \mathbb{R} est dite simple si elle est une combinaison linéaire finie de fonctions caractéristiques :

$$f(x) = \sum_{i=1}^N a_i \mathbf{1}_{A_i}(x), \quad (\text{B.4})$$

où $(a_1, \dots, a_N) \in \mathbb{R}^N$ et $\forall i, A_i \subset \mathbb{R}$. De plus, la fonction est dite étagée si elle est définie sur un espace mesurable et qu'elle est mesurable pour la tribu des boréliens.

Définition B.1.9. Une fonction f à valeurs de \mathbb{R} dans \mathbb{R} est dite à escalier si elle est étagée, positive et constante sur des intervalles.

Définition B.1.10. Soit f une fonction étagée à valeurs réelles positives, définie sur un espace mesuré $(\Omega, \mathcal{A}, \mu)$: $f(x) = \sum_{i=1}^N a_i \mathbf{1}_{A_i}(x)$. Alors on définit l'espérance de f pour la mesure μ par :

$$\int f d\mu = \sum_{i=1}^N a_i \mu(A_i). \quad (\text{B.5})$$

Définition B.1.11. Soit f une fonction à valeurs réelles positives, définie sur un espace mesuré $(\Omega, \mathcal{A}, \mu)$. On définit l'espérance de f pour la mesure μ par :

$$\int f d\mu = \sup_{g \text{ étagée, } 0 \leq g \leq f} \int g d\mu. \quad (\text{B.6})$$

On dit que f est intégrable si son intégrale est finie.

Proposition B.1. – *L'intégrale est un opérateur linéaire : $\forall f, g, \forall \lambda, \int (f + \lambda g) = \int f + \lambda \int g$.*
 – *Si $f = f_+ - f_-$ est une fonction quelconque, avec f_+ et f_- des fonctions positives, alors f est intégrable si et seulement si f_+ et f_- le sont, et dans ce cas, $\int f = \int f_+ - \int f_-$.*

Définition B.1.12. Soit X une variable aléatoire à valeurs d'un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ dans $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$. Alors, on définit l'espérance de X par son intégrale associée à la mesure \mathbb{P} :

$$\mathbb{E}[X] = \int X d\mathbb{P}. \quad (\text{B.7})$$

B.2 Chaînes de Markov

Définition B.2.1. Soit X une variable aléatoire de $(\Omega, \mathcal{A}, \mathbb{P})$ dans (F, \mathcal{F}) . On appelle tribu engendrée par X la plus petite sous-tribu de \mathcal{A} au sens de l'inclusion pour laquelle X est mesurable. On note cette sous-tribu $\sigma(X)$. De même, si \mathcal{P} est un sous-ensemble de \mathcal{A} , on note $\sigma(\mathcal{P})$ la plus petite sous-tribu de \mathcal{A} contenant \mathcal{P} .

Proposition B.2. *Soit X une variable aléatoire intégrable pour $(\Omega, \mathcal{A}, \mathbb{P})$, et soit $A, B \in \mathcal{A}$ deux évènements avec $\mathbb{P}(B) \neq 0$. On définit la probabilité conditionnelle de A par rapport à B :*

$$\mathbb{P}[A|B] = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}. \quad (\text{B.8})$$

De la même façon, on définit l'espérance conditionnelle de X par rapport à B :

$$\mathbb{E}[X|B] = \mathbb{E}[X|\sigma(B)] = \frac{1}{\mathbb{P}(B)} \int_B X d\mathbb{P}. \quad (\text{B.9})$$

Définition B.2.2. Soit X une variable aléatoire intégrable pour $(\Omega, \mathcal{A}, \mathbb{P})$, et soit \mathcal{G} une sous-tribu de \mathcal{A} . Alors, il existe une variable aléatoire Z mesurable pour la tribu \mathcal{G} telle que, pour toute variable aléatoire Y bornée et \mathcal{G} -mesurable, on a :

$$\mathbb{E}[XY] = \mathbb{E}[ZY]. \quad (\text{B.10})$$

On dit alors que Z est l'espérance conditionnelle de X relativement à \mathcal{G} , et on note :

$$Z = \mathbb{E}[X|\mathcal{F}]. \quad (\text{B.11})$$

Définition B.2.3. Une filtration sur l'espace mesurable (Ω, \mathcal{A}) est une suite croissante de tribus.

Définition B.2.4. Un processus stochastique à temps discret est une variable aléatoire définie sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ à valeurs dans les suites sur $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$: $M \mapsto (M_i)_{i=0,1,\dots}$.

Définition B.2.5. Une chaîne de Markov (X_n) est un processus stochastique à temps discret défini sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ tel que la loi de X_{n+1} sachant le passé de la chaîne ne dépend que de X_n , dernier élément de la suite (propriété de Markov) :

$$\mathcal{L}(X_{n+1}|X_n, X_{n-1}, \dots) = \mathcal{L}(X_{n+1}|X_n). \quad (\text{B.12})$$

Définition B.2.6. Soit une chaîne de Markov (X_n) sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$. On dit que la chaîne est ponctuelle lorsque l'espace d'arrivée E est discret.

Définition B.2.7. Soit une chaîne de Markov (X_n) sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$. On dit que la chaîne est homogène lorsque la loi $\mathcal{L}(X_{n+1}|X_n)$ ne dépend pas de n :

$$\mathcal{L}(X_{n+1}|X_n) = \mathcal{L}(X_1|X_0). \quad (\text{B.13})$$

Définition B.2.8. Soit une chaîne de Markov (X_n) ponctuelle homogène sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$. On appelle matrice de transition la matrice stochastique (dont les éléments sont positifs et dont la somme des éléments de chaque ligne est égale à 1) telle que, pour tout $x, y \in E$,

$$\mathbb{P}[X_{n+1} = y|X_n = x] = P(x, y). \quad (\text{B.14})$$

Ainsi, la loi de X_n sachant X_0 correspondra à la matrice stochastique P^n .

Définition B.2.9. Soit une chaîne de Markov (X_n) ponctuelle homogène sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$. On dit que X_n est une chaîne de Markov irréductible si, quel que soit $x, y \in E$, il existe un entier naturel n tel que :

$$P^n(x, y) > 0. \quad (\text{B.15})$$

Définition B.2.10. Soit une chaîne de Markov (X_n) homogène sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$. On appelle mesure invariante de X_n toute mesure de probabilité μ qui est inchangée par la chaîne de Markov : si X_0 suit la loi μ , alors X_1 suit la même loi. En particulier, si (X_n) est une chaîne de Markov ponctuelle homogène de matrice de transition P , on a l'égalité suivante :

$$\mu P = \mu. \quad (\text{B.16})$$

Définition B.2.11. Soit une chaîne de Markov (X_n) ponctuelle homogène irréductible sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ à valeurs dans E . On dit que X_n est une chaîne de Markov aperiodique si, quel que soit $x \in E$, le groupe engendré par l'ensemble $\{n \in \mathbb{N}^* | P^n(x, x) > 0\}$ est l'ensemble \mathbb{N} .

Proposition B.3. *Une chaîne de Markov ponctuelle irréductible possède au plus une mesure de probabilité invariante. Si de plus, l'espace d'état E est fini, alors la mesure invariante existe et est unique.*

Théorème B.4. *Soit une chaîne de Markov (X_n) ponctuelle homogène irréductible aperiodique sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ à valeurs dans E . On appelle π sa mesure de probabilité invariante associée. Soit $f : E \mapsto \mathbb{R}$ une fonction à valeur de E dans \mathbb{R} . Alors, quelle que soit la mesure de probabilité de départ μ associée à X_0 , la moyenne des images de la chaîne de Markov X_n par f converge vers l'espérance mathématique de la loi π :*

$$\frac{1}{n} \sum_{i=1}^n f(X_i) = \int_E f(x) d\pi(x). \quad (\text{B.17})$$

Annexe C

Suites finies à discrédance faible

C.1 Propriétés de ces suites

Lors de calculs d'estimation d'espérance de type Monte-Carlo, la vitesse de convergence de l'estimation dépend de la variance de la suite utilisée pour construire l'estimateur. Pour réduire le temps de calcul, une solution est de ne pas effectuer une estimation de type Monte-Carlo, avec des tirages aléatoires indépendants, mais d'utiliser à la place des suites à discrédance faible. Ces suites possèdent la propriété de posséder une faible variance, car si leur cardinal est suffisamment important, la proportion de ses éléments dans un borélien est proche de la mesure du borélien. Via la loi des grands nombres, nous savons qu'en moyenne, un tirage de type Monte-Carlo suit cette règle, mais pas sur un tirage d'un nombre relativement restreint d'échantillons.

Ces suites à discrédance faible sont appelées "quasi-aléatoires", car elles sont en fait déterministes, et cependant fréquemment utilisées pour remplacer des tirages de lois aléatoires indépendantes identiquement distribuées. En effet, elles partagent des propriétés communes avec les suites indépendantes et identiquement distribuées, et plusieurs méthodes d'évaluation basées sur des suites à discrédance faible existent. Les méthodes de quasi Monte-Carlo en font partie. Le principe repose sur le principe suivant :

$$\int_{[0,1]^d} f(x)dx \approx \sum_{i=1}^N f(x_i), \quad (\text{C.1})$$

avec (x_1, \dots, x_N) un ensemble de points suffisamment répartis de $[0, 1]^d$. La différence entre la méthode de Monte-Carlo et les méthodes de quasi Monte-Carlo repose sur le choix des points. Dans le cas de quasi Monte-Carlo, ces points constituent une suite à discrédance faible. Ces méthodes sont très utilisées en mathématiques financières, afin d'obtenir des estimations numériques rapides. Ces suites sont construites pour que la répartition de ces points soit à peu près proportionnelle à la mesure de la loi. Il en existe plusieurs bien connues : le carré latin, le carré latin hypercube, ou la suite de Sobol. L'inégalité de Koksma-Hlawka nous apprend que l'erreur de calcul dépend de deux facteurs multiplicatifs, l'un dépendant uniquement de la fonction, l'autre de la discrédance de la suite de points utilisés pour calculer l'intégrale. Cela explique l'intérêt de suites à faible discrédance dans l'estimation d'intégrale ou d'espérance.

Définition C.1. Soit $S = \{x_1, \dots, x_N\}$ un ensemble de points dans l'espace $[0, 1]^d$. On définit alors la discrédance de l'ensemble P par :

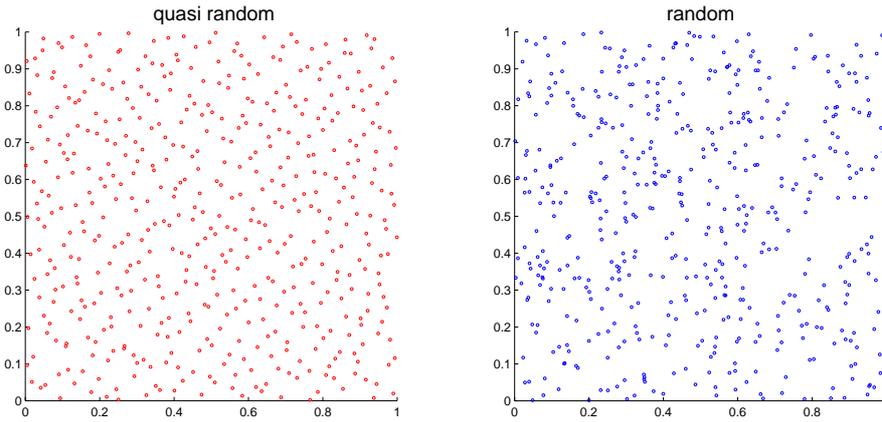


FIG. C.1: Répartition de tirages quasi aléatoires et aléatoires sur un carré du plan

$$D_N(S) = \sup_{B \in \mathcal{B}} \left| \frac{\#\{i | x_i \in B\}}{N} - \lambda(B) \right| \quad (\text{C.2})$$

où \mathcal{B} est l'ensemble des produits d'intervalles de dimension d dans $[0, 1[$. B est donc un ensemble de la forme : $\prod_{i=1}^d [a_i, b_i[$ avec $0 \leq a_i < b_i \leq 1$.

On définit également la discrétance-étoile de l'ensemble P par :

$$D_N^*(S) = \sup_{B \in \mathcal{B}^*} \left| \frac{\#\{i | x_i \in B\}}{N} - \lambda(B) \right| \quad (\text{C.3})$$

où ici, \mathcal{B}^* est l'ensemble des produits d'intervalles de dimension d dans $[0, 1[$ avec les bornes inférieures égales à 0. B est ici un ensemble de la forme : $\prod_{i=1}^d [0, c_i[$ avec $0 \leq c_i < 1$.

C.1.1 Premiers résultats

On peut calculer immédiatement que $D_N^*(S) \geq D_N(S) \geq 2^d D_N^*(S)$.

Si $d = 1$, on a $D_N^*(x_1, \dots, x_N) \leq \frac{1}{2N}$.

En dimension $d = 2$, on sait que $D_N^*(x_1, \dots, x_N) \leq \frac{\ln N}{N}$, avec C le maximum de la fonction $f(x) = \frac{x-2}{16x \ln(x)}$ sur $[3, +\infty[$ ($C \approx 0.023335$) [91].

En dimension $d \geq 2$, on dispose du résultat suivant [79, 80, 81, 82] :

$$D_N^*(x_1, \dots, x_N) \leq \frac{1}{2^{4d}} \frac{1}{(d(-1)\ln 2)^{\frac{d-1}{2}}} \frac{\ln^{\frac{d-1}{2}} N}{N}. \quad (\text{C.4})$$

Ce résultat est généralement utilisé pour la dimension $d = 3$.

Il existe plusieurs conjectures pour encadrer la discrédance de suites. En voici les principales :

Conjecture C.2. *Il existe une constante C_d dépendant de la dimension d telle que, pour toute suite (x_1, \dots, x_N) , on a :*

$$D_N^*(x_1, \dots, x_N) \leq C_d \frac{(\ln N)^{d-1}}{N}. \quad (\text{C.5})$$

Conjecture C.3. *Il existe une constante C'_d dépendant de la dimension d telle que, pour au moins un nombre N de n'importe quelle suite infinie (x_1, x_2, \dots) , on a :*

$$D_N^*(x_1, \dots, x_N) \leq C'_d \frac{(\ln N)^d}{N}. \quad (\text{C.6})$$

Définition C.4. Soit f une fonction à valeurs de $[a, b]$ dans \mathbb{R} . On définit la variation totale de f sur $[a, b]$ par :

$$V_b^a(f) = \sup_{P \in \mathcal{P}} \sum_{i=0}^{n_P-1} |f(x_{i+1}) - f(x_i)|, \quad (\text{C.7})$$

où \mathcal{P} est l'ensemble des partitions $P = (x_0, \dots, x_{n_P})$ de l'intervalle $[a, b]$ telles que $x_0 = a$, $x_i < x_{i+1}$ et $x_{n_P} = b$.

Si f est une fonction dérivable, alors $V_b^a(f) = \int_a^b |f'(x)| dx$. Cette définition peut s'étendre aux dimensions supérieures.

Définition C.5. Soit Ω un ouvert de \mathbb{R}^d et f une fonction à valeurs dans \mathbb{R} définie sur cet ouvert. Alors, on dit que f est à variations finies sur Ω si et seulement s'il existe une mesure de Radon Du telle que :

$$\int_{\Omega} f(x) \operatorname{div} \Phi(x) dx = \int_{\Omega} \langle \Phi, Du(x) \rangle, \quad (\text{C.8})$$

quel que soit $\Phi \in \mathcal{C}^1(\Omega, \mathbb{R}^d)$. Dans ce cas, la variation totale de la fonction f est défini par :

$$V(f, \Omega) = \sup \left\{ \int_{\Omega} f(x) \operatorname{div} \Phi(x) dx, \Phi \in \mathcal{C}^1(\Omega, \mathbb{R}^d), \|\Phi\|_{\mathbb{L}^\infty(\Omega)} \right\} \quad (\text{C.9})$$

C.1.2 Inégalité de Koksma-Hlawka

Soit f une fonction définie sur $[0, 1]^d$, à variation bornée : $V(f) < +\infty$. Alors, pour toute suite d'éléments $(x_n)_{n=1, N}$ dans $[0, 1]^d$, on a :

$$\left| \frac{1}{N} \sum_{i=1}^N f(x_i) - \int_{[0, 1]^d} f(u) du \right| \leq V(f) D_N^*(x_1, \dots, x_N). \quad (\text{C.10})$$

Cette inégalité est optimale. En effet, pour toute suite (x_1, \dots, x_N) dans $[0, 1]^d$ et $\epsilon > 0$, il existe une fonction f telle que $V(f) = 0$ et :

$$\left| \frac{1}{N} \sum_{i=1}^N f(x_i) - \int_{[0, 1]^d} f(u) du \right| > D_N^*(x_1, \dots, x_N) - \epsilon. \quad (\text{C.11})$$

C.2 Construction de suites à discrépanance faible

Suite de Van der Corput :

Il s'agit d'une suite à valeurs dans $]0, 1[$ basée sur la représentation inverse en base n des entiers naturels. Par exemple, en base 2, la suite prendra les valeurs : (0.1, 0.01, 0.11, 0.001, 0.101, 0.011, 0.111, 0.0001...). Si un entier n s'écrit sous la forme $n = \sum_{i=0}^M a_i s^i$ dans la base s , alors le nombre $g_s(n)$ vaut : $g_s(n) = \sum_{i=0}^M a_i s^{-i-1}$.

Une telle suite est dense et équidistribuée dans $[0, 1]$, quelle que soit la base s choisie. Il existe une constante $C > 0$ telle que :

$$D_N^*(g_s(1), \dots, g_s(N)) \leq C \frac{\ln N}{N}. \quad (\text{C.12})$$

C.2.1 Suite de Halton

Il s'agit d'une généralisation des suites de Van der Corput aux dimensions supérieures. Soit d un entier strictement supérieur à 1 et (b_1, \dots, b_d) une suite d'entiers deux à deux premiers entre eux. On pose alors la suite : $x(n) = (g_{b_1}(n), \dots, g_{b_d}(n))$. On a alors la propriété :

$$D_N^*(x(1), \dots, x(N)) \leq C_1 \frac{(\ln N)^d}{N}. \quad (\text{C.13})$$

C.2.2 Suite d'Hammersley

Il s'agit d'un raffinement de la suite de Halton, afin d'en réduire la discrépanance. Le principe consiste à remplacer une des suite $(g_{b_i}(n))$ par la suite n/N : $x(n) = (\frac{n}{N}, g_{b_1}(n), \dots, g_{b_{d-1}}(n))$. On a alors la propriété :

$$D_N^*(x(1), \dots, x(N)) \leq C_1 \frac{(\ln N)^{d-1}}{N}. \quad (\text{C.14})$$

Généralement, les entiers (b_1, \dots, b_d) sont choisis de façon à être des nombres premiers consécutifs.

Annexe D

Convergence du gradient stochastique en norme L^2

Pour rappel, le problème posé est :

$$\underset{x \in U^{ad}}{\operatorname{Argmin}} \mathbb{E}[f(x, \xi(\omega))] \quad (\text{D.1})$$

L'algorithme du gradient stochastique associé à ce problème est le suivant :

1. On choisit une valeur initiale $x_0 \in U^{ad}$, ainsi qu'une suite de réels (ϵ_k) .
2. À chaque itération, on effectue un tirage $\xi(\omega_{k+1})$ de la variable aléatoire ξ suivant sa loi, indépendamment des précédents tirages $(\xi(\omega_1), \dots, \xi(\omega_k))$.
3. On calcule le gradient de la fonction $x \rightarrow f(x, \xi(\omega_{k+1}))$ au point x_k , et on construit le point suivant de la suite :

$$x_{k+1} = \operatorname{proj}_{U^{ad}} (x_k - \epsilon_k f'_x(x_k, \xi(\omega_{k+1}))) . \quad (\text{D.2})$$

4. On vérifie un critère de convergence, et s'il n'est pas vérifié, on retourne à l'étape 2 en incrémentant k .

Théorème D.1. *Soit le jeu d'hypothèses suivant :*

1. U^{ad} est une partie convexe de \mathbb{R}^N .
2. La variable aléatoire $f(x, \cdot) : \Omega \rightarrow \mathbb{R}$ est mesurable et son espérance existe, quel que soit $u \in U^{ad}$.
3. La fonction $f(\cdot, \xi(\omega)) : U \rightarrow \mathbb{R}$ est convexe, semi-continue inférieurement, propre et différentiable pour tout $\omega \in \Omega$.
4. Le gradient de f par rapport à x est borné uniformément :

$$\exists m > 0, \forall x \in U^{ad}, \forall \omega \in \Omega, \|f'_x(x, \xi(\omega))\| \leq m. \quad (\text{D.3})$$

5. L'ensemble $U^\#$ des solutions du problème d'optimisation est non vide et vérifie la relation :

$$\forall x \in U^{\text{ad}}, F(x) - F(x^*) \geq c(\text{dist}_{U^\#}(x))^2,$$

où $\text{dist}_{U^\#}(\cdot)$ est la fonction distance avec l'ensemble $U^\#$.

6. La suite (ϵ_k) est positive décroissante, et la série associée vérifie :

$$\sum_k \epsilon_k = +\infty, \sum_k (\epsilon_k)^2 < +\infty. \quad (\text{D.4})$$

Si ces six hypothèses sont vérifiées, alors la suite x_k converge en norme \mathcal{L}^2 vers une solution du problème :

$$\lim_{k \rightarrow +\infty} \mathbb{E}[\text{dist}_{U^\#}(x^{(k)})^2] = 0. \quad (\text{D.5})$$

De plus, si l'on choisit les pas $\epsilon^{(k)}$ de la forme suivante :

$$\epsilon^{(k)} = \frac{1}{ck + \frac{m^2}{cd^{(0)}}}, \quad (\text{D.6})$$

(avec $d^{(0)} = \text{dist}_{U^\#}(x^{(0)})$), on obtient la borne suivante sur la vitesse de convergence :

$$\mathbb{E}[\text{dist}_{U^\#}(x^{(k)})^2] \leq \frac{1}{\frac{c^2}{m^2}k + \frac{1}{d^{(0)}}}. \quad (\text{D.7})$$

Démonstration. Par semi-continuité inférieure et par convexité de la fonction F , $U^\#$ est convexe fermé, donc la projection sur cet ensemble est bien définie. Soit $\{x^{(k)}\}_{k \in \mathbb{N}}$ une réalisation de l'algorithme du gradient stochastique, et soit $\{\bar{x}^{(k)}\}_{k \in \mathbb{N}}$ la suite des projections sur l'ensemble $U^\#$: $\text{dist}_{U^\#}(x^{(k)})^2 = \|x^{(k)} - \bar{x}^{(k)}\|^2$.

On pose alors $d^{(k)}(\omega) = \text{dist}_{U^\#}(x^{(k)}(\omega))^2$. Par propriété de la projection, $\forall x \in U^\#$, $\|x^{(k+1)} - \bar{x}^{(k+1)}\|^2 \leq \|x^{(k+1)} - x\|^2$. C'est en particulier vrai pour $x = \bar{x}^{(k)}$:

$$\begin{aligned} d^{(k+1)}(\omega) &\leq \|x^{(k+1)} - \bar{x}^{(k)}\|^2 \\ &\leq \left\| \text{proj}_{U^{\text{ad}}}(u^{(k)} - \epsilon^{(k)} f'_x(x^{(k)}, \omega^{(k+1)})) - \bar{x}^{(k)} \right\|^2 \\ &\leq \left\| u^{(k)} - \epsilon^{(k)} f'_x(x^{(k)}, \omega^{(k+1)}) - \bar{x}^{(k)} \right\|^2 \\ &\leq d^{(k)} + \epsilon^{(k)^2} m^2 - 2\epsilon^{(k)} \langle x^{(k)} - \bar{x}^{(k)}, f'_x(x^{(k)}, \omega^{(k+1)}) \rangle. \end{aligned}$$

On prend alors l'espérance conditionnelle de cette expression par rapport à la tribu $\mathcal{F}^{(n)} = \sigma(\omega^{(k)}, k \leq n)$:

$$\begin{aligned} \mathbb{E}[d^{(k+1)} | \mathcal{F}^{(k)}] &\leq d^{(k)} + \epsilon^{(k)^2} m^2 - 2\epsilon^{(k)} \langle x^{(k)} - \bar{x}^{(k)}, F'_x(x^{(k)}, \omega^{(k+1)}) \rangle \\ &\leq d^{(k)} + \epsilon^{(k)^2} m^2 - 2\epsilon^{(k)} (F(x^{(k)}) - F(x^*)) \\ &\leq (1 - 2\epsilon^{(k)} c) d^{(k)} + \epsilon^{(k)^2} m^2. \end{aligned}$$

On prend alors l'espérance de cette inégalité, et on obtient donc :

$$\mathbb{E}[d^{(k+1)}] \leq (1 - 2\epsilon^{(k)}c)\mathbb{E}[d^{(k)}] + \epsilon^{(k)^2}m^2. \quad (\text{D.8})$$

Pour k_0 suffisamment grand, $1 > 1 - 2\epsilon^{(k)}c > 0$, $\forall k \geq k_0$, donc :

$$\mathbb{E}[d^{(k+k_0+1)}] \leq \left(\prod_{i=0}^k (1 - 2\epsilon^{(k_0+i)}c) \right) \mathbb{E}[d^{(k)}] + \left(\sum_{i=0}^k \epsilon^{(k_0+k)^2} \right) m^2. \quad (\text{D.9})$$

Or, la suite $\prod_{i=0}^k (1 - 2\epsilon^{(k_0+i)}c)$ est décroissante et converge vers 0. En effet, tous les termes compris entre 0 et 1, et le logarithme de cette suite vaut $\sum_{i=0}^k \log(1 - 2\epsilon^{(k_0+i)}c) \leq -2c \sum_{i=0}^k \epsilon^{(k_0+i)}$. La série $\sum \epsilon^{(k)}$ étant positive divergente, on en déduit que le logarithme de la suite initiale tend vers $-\infty$, donc que la suite produit converge vers 0. \square

Annexe E

Calcul de la probabilité approchée pour Arrow-Hurwicz stochastique

Dans cette annexe, nous apportons la démonstration des formules de convolution de l'algorithme d'Arrow-Hurwicz.

E.1 Formules des contraintes approchées

Pour rappel, la fonction g vaut :

$$g(x, \xi(\omega)) = p_0 - \mathbf{1}_{\mathbb{R}_+}(\theta(x, \xi(\omega)) - \theta_0). \quad (\text{E.1})$$

Afin de régulariser cette fonction et la rendre dérivable, on convole la fonction indicatrice avec une suite d'approximation de l'unité :

$$\psi_r(\cdot) = \frac{1}{r} \psi\left(\frac{\cdot}{r}\right), \quad (\text{E.2})$$

La fonction ψ doit être positive, symétrique, continue, intégrable sur \mathbb{R} et d'intégrale 1, et posséder son unique maximum en 0. On considère à présent la fonction :

$$\psi(x) = \frac{3}{4}(1 - x^2)\mathbf{1}_{[-1,1]}(x). \quad (\text{E.3})$$

Cette fonction remplit bien les hypothèses demandées. Elle a été choisie parmi une famille de fonctions satisfaisant l'ensemble des hypothèses, car elle minimisait un critère de produit de variance régulant la vitesse de convergence de la fonction convoluée, comme nous le détaillerons davantage dans la suite. La formule de la fonction contrainte approchée g_r vaut alors :

$$g_r(x, \xi(\omega)) = p_0 - (\psi_r * \mathbf{1}_{\mathbb{R}_+})(\theta(x, \xi(\omega)) - \theta_0). \quad (\text{E.4})$$

Pour des raisons de simplifications d'écriture, on pose la quantité suivante :

$$K_r = \Pi_{[-1,1]}\left(\frac{\theta(x, \xi(\omega)) - \theta_0}{r}\right).$$

On calcule facilement que :

$$g_r(x, \xi(\omega)) = p_0 - \frac{1}{4}(K_r^3 - 3K_r + 2), \quad (\text{E.5})$$

$$\nabla_x g_r(x, \xi(\omega)) = -\frac{1}{r}(1 - K_r^2)\nabla_x \theta(x, \xi(\omega)). \quad (\text{E.6})$$

En effet, pour toute quantité $(\theta(x, \xi(\omega)) - \theta_0) \in \mathbb{R}$, on a :

$$\begin{aligned}
 (\psi_r * \mathbf{1}_{\mathbb{R}_+})(\theta(x, \xi(\omega)) - \theta_0) &= \frac{1}{r} \int_{-\infty}^{+\infty} \mathbf{1}_{\mathbb{R}_+}(y) \psi \left(\frac{y + \theta_0 - \theta(x, \xi(\omega))}{r} \right) dy \\
 &= \frac{1}{r} \int_0^{+\infty} \psi \left(\frac{y + \theta_0 - \theta(x, \xi(\omega))}{r} \right) dy \\
 &= \frac{1}{r} \int_{\theta(x, \xi(\omega)) - \theta_0}^{+\infty} \psi \left(\frac{y}{r} \right) dy \\
 &= \int_{\frac{\theta(x, \xi(\omega)) - \theta_0}{r}}^{+\infty} \psi(y) dy \\
 &= \frac{3}{4} \int_{[-1, 1] \cap [\frac{\theta(x, \xi(\omega)) - \theta_0}{r}, +\infty[} (1 - y^2) dy.
 \end{aligned}$$

On remarque que l'ensemble d'intégration ci-dessus est égal à l'intervalle $[K_r, 1]$ à un ensemble de mesure nulle près. On peut donc calculer la fonction g_r en fonction de K_r :

$$\begin{aligned}
 g_r(x, \xi(\omega)) &= p_0 - \frac{3}{4} \int_{K_r}^1 (1 - y^2) dy \\
 &= p_0 - \frac{3}{4} \left[y - \frac{y^3}{3} \right]_{K_r}^1 \\
 &= p_0 - \frac{3}{4} \left(\frac{K_r^3}{3} - K_r + 1 - \frac{1}{3} \right) \\
 &= p_0 - \frac{1}{4} (K_r^3 - 3K_r + 2).
 \end{aligned}$$

De la même façon, on peut calculer le gradient de la fonction g_r en fonction de K_r :

$$\begin{aligned}
 \nabla_x g_r(x, \xi(\omega)) &= \frac{1}{r} \psi \left(\frac{\theta(x, \xi(\omega)) - \theta_0}{r} \right) \nabla_x \theta(x, \xi(\omega)) \\
 &= \frac{3}{4r} \left(1 - \left(\frac{\theta(x, \xi(\omega)) - \theta_0}{r} \right)^2 \right) \mathbf{1}_{[-1, 1]} \left(\frac{\theta(x, \xi(\omega)) - \theta_0}{r} \right) \nabla_x \theta(x, \xi(\omega)) \\
 &= \frac{3}{4r} (1 - K_r^2) \nabla_x \theta(x, \xi(\omega))
 \end{aligned}$$

La dernière égalité se prouve facilement, en supposant que $|K_r| < 1$ ou non (dans le second cas, les deux termes sont nuls, et dans le premier, il s'agit simplement d'une réécriture).

E.2 Choix des paramètres

Cet algorithme, malgré la solution qu'il apporte, n'est pas toujours simple à mettre en application. Son comportement dépend de plusieurs séries de pas, et donc de plusieurs paramètres que doit définir l'utilisateur. Or, il s'agit d'un algorithme similaire à l'optimisation multi-objectif, en le sens qu'il cherche à optimiser une fonction objectif tout en vérifiant une contrainte d'inégalité en probabilité. Ce sont deux objectifs aux tendances contraires, provoquant une instabilité du comportement de l'algorithme. Ce qui fait que les paramètres permettant à l'algorithme de converger sont souvent difficiles à trouver en pratique.

Dans le cadre de la convolution, nous disposons d'un paramètre réglable : la fonction h permettant la construction de la suite d'approximation de l'unité. Puisque les hypothèses de convergence de l'algorithme ne dépendent pas du choix de cette fonction si elle vérifie quelques propriétés, la stratégie préférable serait de la choisir selon un critère de minimisation de la variance, dans le but d'accélérer la vitesse de convergence. Supposons que r_k soit le paramètre utilisé dans le tirage de k variables aléatoires (ξ_i) . Alors, la variance de la variable $\frac{\partial}{\partial x} p_r(x, \xi)$ sera proportionnel à $\|h\|_{L^2}^2 / (kr_k)$. Le biais de cette quantité par rapport à $P'(x)$ est proportionnel à $\sigma_h^4 r_k^4$.

L'erreur minimale du carré de la moyenne est atteinte en $r_k = (\|h\|_{L^2}^2 / (4\sigma_h^4 k))^{1/5}$, et cette erreur minimale vaut alors $(5\sigma_h^{4/5} \|h\|_{L^2}^{8/5}) / (4k)^{4/5}$. Le critère déterminant sera alors $\sigma_h \|h\|_{L^2}^2$.

Parmi un choix de fonctions correspondant aux hypothèses, celle qui minimise le critère ci-dessus et qui a été retenue pour l'application de l'algorithme est :

$$h(x) = \frac{3}{4}(1 - x^2)\mathbf{1}_{[-1,1]}. \quad (\text{E.7})$$

Annexe F

Chaîne CAPRI

F.1 Description

CAPRI est l'acronyme de Chaîne Aéroélastique de Prévion de la Réponse et des Instabilités. La chaîne CAPRI est un ensemble de programmes informatiques permettant de mettre en œuvre le calcul de matrices des FAG selon différentes méthodes linéaires comme les doublets ou le potentiel harmonique.

Ces programmes fonctionnent à partir des fichiers produits par le logiciel NASTRAN. Ils sont conçus de façon à ce que les fichiers de sortie d'un programme soient les fichiers d'entrée du programme suivant, d'où l'appellation de "chaîne".

F.2 Programmes utilisés

Les principaux programmes employés dans nos algorithmes de calcul sont les suivants :

- Le code "capriMode" permet de générer des modes souples sur les surfaces aérodynamiques. Ceux-ci seront ensuite projetés sur le modèle structure.
- Le code "capridoub3d" est utilisé pour calculer les matrices des FAG dans le domaine subsonique.
- Le code "capriFkv4" réalise le calcul de flottement avec les FAG obtenues par les précédents codes de calcul. C'est ce programme qui permet de déterminer la valeur du paramètre critique.
- Le code "capriRoger" permet d'obtenir les valeurs des matrices des FAG. Il sera possible à partir de ces matrices de construire les formules de gradient des paramètres critiques.

Les algorithmes que nous avons employés ont récupéré les résultats calculés par ces programmes, puis ont réalisé des calculs d'optimisation en utilisant ces résultats numériques. À chaque boucle de calcul, nous avons fait tourner le logiciel NASTRAN avec le modèle numérique, puis nous avons fait fonctionner les programmes de la chaîne CAPRI nécessaires à nos calculs d'optimisation. Puis, une fois les données calculées et récupérées, le calcul des nouvelles valeurs du modèle a été faite sous MatLab/SciLab, et celles-ci ont été intégrées dans le fichier numérique, avant d'entamer une nouvelle boucle de calcul aéroélastique.

Bibliographie

- [1] E.H.L. Aarts and P.J.M. van Laarhoven. Statistical cooling : A general approach to combinatorial optimization problems. *Philips J. Res.*, 40(4) :193–226, 1985.
- [2] I. Adler, N. Karmarkar, M. Resende, and G. Veiga. An implementation of karmarkar’s algorithm for linear programming. *Mathematical Programming*, 44 :297–335, 1989.
- [3] E. Albano and W. Rodden. A doublet lattice method for calculating lift distributions on surfaces in subsonic flows. *AIAA Journal*, 7 :279–285, 1969.
- [4] L. Andrieu. *Optimisation sous contrainte en probabilité*. PhD thesis, École Nationale des Ponts et Chaussées, december 2004.
- [5] L. Andrieu, G. Cohen, and F. Vázquez-Abad. Gradient-based simulation optimization under probability constraints. *European Journal of Operational research*, 212 :345–351, 2011.
- [6] Laetitia Andrieu, Guy Cohen, and Felisa Vázquez-Abad. Stochastic programming with probability constraints. hal-00166149, version 1, August 2007.
- [7] R. Apparigliato. *Règles de décision pour la gestion du risque : Application à la gestion hebdomadaire de la production électrique*. PhD thesis, École Polytechnique, 2008.
- [8] B. Arouna. Robbins-Monro algorithms and variance reduction in finance. *The Journal of Computational Finance*, 7(2), Winter 2003/2004., 7(2), Winter 2003/2004.
- [9] N. Bakhvalov. *Méthodes numériques*. Éditions Mir, 1976.
- [10] D.M. Bates and D.G. Watts. *Nonlinear Regression Analysis and its Applications*. Wiley, New York, 1988.
- [11] K.J. Bathe and E.L. Wilson. *Numerical methods in finite element analysis*. Prentice Hall, 1972.
- [12] R.E. Bellman. *Dynamic programming*. Princeton University Press, 1957.
- [13] R.E. Bellman and L.A. Zadeh. Decision-making in a fuzzy environment. *management science*, 17 :141–161, 1970.
- [14] P. Billingsley. *Probability and measure*. Wiley, third edition, 1995.
- [15] S. C. Castravete and R. A. Ibrahim. "effect of stiffness uncertainties on the flutter of a cantilever wing". *AIAA Journal*, 46 :925–935, 2008.
- [16] R. Cerf. *Une Théorie Asymptotique des Algorithmes Génétiques*. PhD thesis, Université Montpellier II, 1994.
- [17] Yang Chao, Xiao ZhiPeng, Wan ZhiQiang, Yan De, and Dai YuTing. Aeroelastic optimization design for wing with maneuver load uncertainties. *Science China, Technological Sciences*, 53 :3102–3109, 2010.

- [18] J-C Chassaing and D. Lucor. Adaptive approaches for uncertainty quantification of airfoil lco. In *15th International Forum on Aeroelasticity and Structural Dynamics, IFASD2011*, 2011.
- [19] B.G. Dantzig. Linear programming under uncertainty. *management science*, 1 :197–206, 1955.
- [20] J.-P. Dedieu. Mathématiques & applications. In *Points fixes, zéros et la méthode de Newton*, 54. Springer, 2006.
- [21] J.E. Dennis, D.M. Gay, and R.E. Welsch. An adaptative nonlinear least-squares algorithm. *ACM Transactions on Mathematical Software*, 7 :348–368, 1981.
- [22] J.E. Dennis Jr. Nonlinear least squares. In Academic Press (New York), editor, *The State of the Art in Numerical Analysis*, pages 269–312. D. Jacobs, 1977.
- [23] der Kiureghian. *First and second order reliability methods*, chapter 14. CRC Press, 2005.
- [24] U. Diwekar. A novel sampling approach to combinatorial optimization under uncertainty. *Computational Optimization and Applications*, 24 :335, 2003.
- [25] U. Diwekar. *Introduction to Applied Optimization, Second Edition*. Springer-Verlag, 2008.
- [26] M. Dufflo. *Random Iterative Models*. Springer-Verlag, 1997.
- [27] J. DupacovĀj. Stochastic programming : Approximation via scenarios. In *Proceedings of 3rd Caribbean Conference on Approximation and Optimization, Puebla*, 1995.
- [28] Y. Ermoliev. Stochastic quasigradient methods and their application to systems optimization. *Stochastics*, 9 :1–36, 1983.
- [29] Y. Ermoliev and R. Wets. *Numerical Techniques for Stochastic Optimization*. Springer verlag, 1988.
- [30] R. Eustaquio, E. Karas, and A. Ribeiro. Constraint qualification for nonlinear programming. Technical report, Federal University of Parana, 2010.
- [31] R. Fletcher. *Numerical Analysis*, chapter Conjugate gradient methods for indefinite systems, pages 73–89. Springer Berlin Heidelberg, 1976.
- [32] C.F. Gauss. *Theoria Motus Corporum Coelestium in Sectionibus Conicus Solem Ambientum*. Dover, New York, 1809.
- [33] P.E. Gill and Murray ; W. Algorithms for the solution of the nonlinear least-squares problem. *SIAM Journal on Numerical Analysis*, 15 :977–992, 1978.
- [34] P. Glasserman. Monte Carlo methods in financial engineering. In *Stochastic Modeling and Applied Probability*. volume 53 of Applications of Mathematics (New York). Springer-Verlag, New York, 2004, 2004.
- [35] R.T. Haftka and Z. G]rdal. *Elements of Structural Optimization*. Springer, 1992.
- [36] A.M. Hasofer and N.C. Lind. *An exact and invariant first-order reliability format*. Waterloo, Ont. : Solid Mechanics Division, University of Waterloo, 1973.
- [37] A. S. Householder. *The Numerical Treatment of a Single Nonlinear Equation*. New York : McGraw-Hill, 1970.
- [38] L. Ingber. Very fast simulated re-annealing. *Mathematical and Computer Modelling*, 12(8) :967–973, 1989.

-
- [39] E.T. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106 :620–630, 1957.
- [40] J.N. Kapur and H.K. Kesavan. *Entropy Optimization Principles with Applications*. Academic Press, 1992.
- [41] N. Karmarkar. A new polynomial time algorithm for linear programming. *Combinatorica*, 4 :373–395, 1984.
- [42] M. Karpel. Design for active flutter suppression and gust alleviation using state-space aeroelastic modeling. *J. of Aircraft*, 19(3) :221–227, 1982.
- [43] M. Karpel. Time-domain aeroservoelastic modeling using weighted unsteady aerodynamic forces. *J. of Guidance, Control, and Dynamics*, 13(1) :30–37, 1990.
- [44] T. Kato. *Perturbation Theory for Linear Operators*. Springer, Februar 1995.
- [45] H.H. Khodaparast, J.E. Mottershead, and K.J. Badcock. Propagation of structural uncertainty to linear aeroelastic stability. *Computers and Structures*, 88 :223–236, 2010.
- [46] J. Kiefer and J. Wolfowitz. Stochastic estimation of the maximum of a regression function. *Ann. Math. Stat.*, 23 :452–466, 1952.
- [47] K.J. Kim and U. Diwekar. Hammersley stochastic annealing : Efficiency improvement for combinatorial optimization under uncertainty. *IIE Transactions*, 34 :761–777, 2002.
- [48] D.R. Kincaid and E.W. Cheney. *Numerical Analysis : Mathematics of Scientific Computing*. American Mathematical Soc., 2002.
- [49] H.W. Kuhn and A.W. Tucker. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, 1950.
- [50] J. Lelong. *Asymptotic properties of stochastic algorithms and pricing of Parisian options*. PhD thesis, Ecole Nationale des Ponts et Chaussées, 2007.
- [51] K. Levenberg. A method for the solution of certain problems in least squares. *Quarterly of Applied Mathematics*, 2 :164–168, 1944.
- [52] R. Lind and M. Brenner. *Robust Aeroservoelastic Stability Analysis*. Springer-Verlag, 1999.
- [53] J. Luedtke and S. Ahmed. A sample approximation approach for optimization with probabilistic constraints. *SIAM Journal on Optimization*, 19 :674–699, 2008.
- [54] H.O. Madsen, S. Krenk, and N.C. Lind. *Methods of structural safety*. Prentice-Hall, 1985.
- [55] D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal of Applied Mathematics*, 11 :431–441, 1963.
- [56] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21 :1087–1092, 1953.
- [57] S. Missoum, C. Dribusch, and P. Beran. Reliability-based design optimization of nonlinear aeroelasticity problems. *Journal of Aircraft*, 47(3) :992–998, 2010.
- [58] J.J. Moré. The levenberg-marquardt algorithm : implementation and theory. *Lecture Notes in Mathematics*, 630 :105–116, 1978.

- [59] G. Mortchéléwicz. Manuel utilisateur simplifié de la chaîne capri. Technical Report 1/18220 DADS, ONERA, Département Aéroélasticité et Dynamique des Structures, Septembre 2011.
- [60] S. Murugan, R. Ganguli, and D. Harursampath. Effects of structural uncertainty on aeroelastic response of composite helicopter rotor. In *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, pages Chapter DOI : 10.2514/6.2007-2298, 2007.
- [61] NATO - RTO. *Uncertainty and Aeroelasticity : A Review of ONERA's Experience*, Athenes, 2007.
- [62] A. Nemirovski and A. Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17(4) :969–996, 2006.
- [63] A. Nemirovski and A. Shapiro. Scenario approximations of chance constraints. In G. Calafiore and F. Dabbene, editors, *Probabilistic and Randomized Methods for Design under Uncertainty*. Springer, 2006.
- [64] J.M. Ortega and W.C. Rheinbold. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, 1970.
- [65] J.-Y. Ouvrard. *Probabilités tome 2, Maîtrise-Agrégation*. Cassini, 2000.
- [66] L. Painton and U. Diwekar. Stochastic annealing for synthesis under uncertainty. *European J. of Operational Research*, 83 :489–502, 1995.
- [67] Laura Painton and Urmila Diwekar. Stochastic annealing for synthesis under uncertainty. *European Journal of Operational Research*, 83(3) :489–502, 1995.
- [68] C. Pettit. Uncertainty quantification in aeroelasticity : recent results and research challenges. *Journal of Aircraft*, 41 :1217–1229, 2004.
- [69] C.L. Pettit and P.S. Beran. Effects of parametric uncertainty on airfoil limit cycle oscillation. *J. Aircraft*, 40 :1004–1006, 2003.
- [70] F. Poirion. Modélisation et prise en compte des incertitudes de masse pour l'analyse des instabilités aéroélastiques dans le domaine transsonique. Rapport Technique 48/3064 RY 097 R, ONERA, 1990.
- [71] F. Poirion. Effect of structural uncertainties on flutter analysis. *La Recherche Aérospatiale*, 6 :31–41, 1992.
- [72] F. Poirion. Multi-mach rational approximation to generalized aerodynamic forces. *Journal of Aircraft*, 33(6), nov-dec. 1996.
- [73] F. Poirion. Flutter analysis of a two-dimensional airfoil with a randomly perturbed geometry. In P. Spanos and G. Deodatis, editors, *Computational Stochastic mechanics*, pages 481–487. Millpress, 2003.
- [74] F. Poirion. Chaos polynomial representation of parametric uncertainties in aeroelasticity. In *International Forum on aeroelasticity and structural dynamics, IFASD 2005*, pages CDrom,, 2005.
- [75] F. Poirion. Effective probabilistic methods for uncertainties in aeroelasticity. In *8th World Congress on Computational Mechanics WCCM8*, volume CDROM, Venice, Italy, 2008.
- [76] W. Queck. The convergence factor of preconditioned algorithms of the arrow-hurwicz type. *SIAM Journal on Numerical Analysis*, 26(4) :1016–1030, August 1989.
- [77] H. Robbins and S. Monro. A stochastic approximation method. *Ann. Math. Statistics*, 22 :400–407, 1951.

- [78] K.L. Roger, G.E. Hodges, and L. Felt. Active flutter suppression—a flight test demonstration. *J. of Aircraft*, 12(6) :551–556, 1975.
- [79] K. F. Roth. On irregularities of distribution. *Mathematika*, 1 :73–79, 1954.
- [80] K. F. Roth. On irregularities of distribution. ii. *Comm. Pure Appl. Math.*, 29 :739–744, 1976.
- [81] K. F. Roth. On irregularities of distribution. iii. *Acta Arith.*, 35 :373–384, 1979.
- [82] K. F. Roth. On irregularities of distribution. iv. *Acta Arith.*, 37 :67–75, 1980.
- [83] B. Rousselet and D. Chenais. Continuité et différentiabilité d'éléments propres : Application à l'optimisation de structures. *Applied Mathematics and Optimization*, 22(1) :27–59, July 1990.
- [84] R. Roy, S. Hinduja, and R. Teti. Recent advances in engineering design optimisation : challenges and future trends. *manufacturing technology*, 57 :697–715, 2008.
- [85] C.S. Rudisill and K Bhatia. Optimization of complex structures to safety flutter requirements. *AIAA Journal*, 9(8) :1487–1491, 1971.
- [86] A. Ruhe. Accelerated gauss-newton algorithms for nonlinear least-squares problems. *SIAM Review*, 22 :318–337, 1980.
- [87] B. Ruppert. *Handbook of Sequential Analysis*. Marcel Dekker, NY, B.K. Ghosh and P.K. Sen eds, 1991.
- [88] N. V. Sahinidis. Optimization under uncertainty : State-of-the-art and opportunities. *Computers & Chemical Engineering*, 28(6-7) :971–983, 2004.
- [89] T. R. Scavo and J. B. Thoo. On the geometry of halley's method. *Amer. Math. Monthly*, 102 :417–426, 1995.
- [90] K Schittkowski. A FORTRAN subroutine for solving constrained nonlinear programming problems. *Ann Operations research*, 5 :485–500, 1985.
- [91] W. M. Schmidt. Irregularities of distribution. vii. *Acta Arith.*, 21 :45–50, 1972.
- [92] E. Schroeder. Uber unendlich viele algorithmen zur auflosung der gleichungen. *Math. Ann.*, 2 :317–365, 1870.
- [93] A. Shapiro. *Handbooks in OR & MS, Vol. 10*, chapter 6. Monte Carlo Sampling Methods, pages 353–425. Elsevier Science B.V., 2003.
- [94] H. Szu and R. Hartley. Fast simulated annealing. *Physics Letters A*, 122(3-4) :157–162, June 1987.
- [95] S.H. Tiffany and M. Karpel. Aeroservoelastic modeling and applications using minimum-state approximations of the unsteady aerodynamics. Technical Report TM-101574, NASA, 1989.
- [96] M. J. Turner. Optimization of structures to satisfy flutter requirements. *AIAA Journal*, 7(5) :945–951, 1969.
- [97] R.J.-B. Wets. *Stochastic programming*, volume Optimization (1). Nemhauser, G.L. and Rinnooy Kan, A.H.G. and. Todd, M.J., 1990. Handbooks in Operations Research and Management Science, North-Holland, Amsterdam, Netherlands.
- [98] Douglass Wilde. *Optimum Seeking Methods*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1964.
- [99] W. Yao, X Chen, W. Luo, M. van Tooren, and J. Guo. Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles. *Progress in Aerospace Sciences*, 47 :450–479, 2011.

- [100] O. Zienkiewicz and R. Taylor. *The finite element method, 4th Edition*, volume 1. McGraw-Hill, 1989.