



**HAL**  
open science

# Développement des méthodes AK pour l'analyse de fiabilité. Focus sur les événements rares et la grande dimension

Nicolas Lelièvre

► **To cite this version:**

Nicolas Lelièvre. Développement des méthodes AK pour l'analyse de fiabilité. Focus sur les événements rares et la grande dimension. Génie mécanique [physics.class-ph]. Université Clermont Auvergne [2017-2020], 2018. Français. NNT : 2018CLFAC045 . tel-02060337

**HAL Id: tel-02060337**

**<https://theses.hal.science/tel-02060337v1>**

Submitted on 7 Mar 2019

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

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

Université Clermont Auvergne  
École doctorale n° 70 : Sciences pour l'Ingénieur

## THÈSE

*présentée et soutenue publiquement le 13 Décembre 2018 par*

**Nicolas Lelièvre**

**Ingénieur IFMA**

afin d'obtenir le grade de

**Docteur d'Université**

**Spécialité doctorale "Génie mécanique"**

# Développement des méthodes AK pour l'analyse de fiabilité Focus sur les événements rares et la grande dimension

### Jury

<b>M. Christian Gogu,</b>	Maître de Conférences HDR, Institut Clément Ader, Toulouse	Rapporteur
<b>M. Jérôme Morio,</b>	Maître de Recherche ONERA, Toulouse	Rapporteur
<b>Mme. Céline Helbert,</b>	Maître de Conférences, École Centrale de Lyon	Examinatrice
<b>M. Rodolphe Le Riche,</b>	Directeur de Recherche CNRS, École des Mines de Saint Étienne	Examinateur
<b>M. Bruno Sudret,</b>	Professeur des Universités, ETH Zurich	Examinateur
<b>M. Nicolas Gayton,</b>	Professeur des Universités, SIGMA Clermont	Directeur
<b>M. Pierre Beaurepaire,</b>	Maître de Conférences, SIGMA Clermont	Encadrant
<b>Mme. Cécile Mattrand,</b>	Maître de Conférences, SIGMA Clermont	Encadrante
<b>M. Jean-Marc Bourinet,</b>	Maître de Conférences HDR, SIGMA Clermont	Invité



*À mes parents, ma chérie Flora et mon fils Nathanaël*

*C'est dans son incertitude que réside largement la valeur de la philosophie. Celui qui ne s'y est pas frotté traverse l'existence comme un prisonnier : prisonnier des préjugés du sens commun, des croyances de son pays ou de son temps, de convictions qui ont grandi en lui sans la coopération ni le consentement de la raison.*

Bertrand Russell

---





# Résumé

Les ingénieurs utilisent de plus en plus de modèles numériques leur permettant de diminuer les expérimentations physiques nécessaires à la conception de nouveaux produits. Avec l'augmentation des performances informatiques et numériques, ces modèles sont de plus en plus complexes et coûteux en temps de calcul pour une meilleure représentation de la réalité. Les problèmes réels de mécanique sont sujets en pratique à des incertitudes qui peuvent impliquer des difficultés lorsque des solutions de conception admissibles et/ou optimales sont recherchées. La fiabilité est une mesure intéressante des risques de défaillance du produit conçu dus aux incertitudes. L'estimation de la mesure de fiabilité, la probabilité de défaillance, nécessite un grand nombre d'appels aux modèles coûteux et devient donc inutilisable en pratique. Pour pallier ce problème, la métamodélisation est utilisée ici, et plus particulièrement les méthodes AK (*Active learning and Kriging based methods*) qui permettent la construction d'un modèle mathématique représentatif du modèle coûteux avec un temps d'évaluation beaucoup plus faible.

Le premier objectif de ces travaux de thèse est de discuter des formulations mathématiques des problèmes de conception sous incertitudes. Cette formulation est un point crucial de la conception de nouveaux produits puisqu'elle permet de comprendre les résultats obtenus. Une définition des deux concepts de fiabilité et de robustesse est aussi proposée. Ces travaux ont abouti à une publication dans la revue internationale *Structural and Multidisciplinary Optimization* (Lelièvre et al., 2016a).

Le second objectif est de proposer une nouvelle méthode AK pour l'estimation de probabilités de défaillance associées à des événements rares. Cette nouvelle méthode, nommée AK-MCSi, présente trois améliorations de la méthode AK-MCS : des simulations séquentielles de Monte-Carlo pour diminuer le temps d'évaluation du métamodèle, un nouveau critère d'arrêt sur l'apprentissage plus strict permettant d'assurer le bon classement de la population de Monte-Carlo et un enrichissement multi-points permettant la parallélisation des calculs du modèle coûteux. Ce travail a été publié dans la revue *Structural Safety* (Lelièvre et al., 2018a).

Le dernier objectif est de proposer de nouvelles méthodes pour l'estimation de probabilités de défaillance en grande dimension, c'est-à-dire un problème défini à la fois par un modèle coûteux et un très grand nombre de variables aléatoires d'entrée. Deux nouvelles méthodes, AK-HDMR1 et AK-PCA, sont proposées pour faire face à ce problème et sont basées respectivement sur une décomposition fonctionnelle et une technique de réduction

de dimension. La méthode AK-HDMR1 fait l'objet d'une publication soumise à la revue *Reliability Engineering and Structural Safety* le 1er octobre 2018.

**Mots-clés** : Incertitudes, Fiabilité, Robustesse, Probabilité de défaillance, Évènements rares, Méta-modélisation, Krigeage, Méthodes AK, Grande dimension.

---

# Abstract

Engineers increasingly use numerical model to replace the experimentations during the design of new products. With the increase of computer performance and numerical power, these models are more and more complex and time-consuming for a better representation of reality. In practice, optimization is very challenging when considering real mechanical problems since they exhibit uncertainties. Reliability is an interesting metric of the failure risks of design products due to uncertainties. The estimation of this metric, the failure probability, requires a high number of evaluations of the time-consuming model and thus becomes intractable in practice. To deal with this problem, surrogate modeling is used here and more specifically AK-based methods (Active learning and Kriging based methods) to enable the approximation of the physical model with much fewer time-consuming evaluations.

The first objective of this thesis work is to discuss the mathematical formulations of design problems under uncertainties. This formulation has a considerable impact on the solution identified by the optimization during design process of new products. A definition of both concepts of reliability and robustness is also proposed. These works are presented in a publication in the international journal : *Structural and Multidisciplinary Optimization* (Lelièvre et al., 2016a).

The second objective of this thesis is to propose a new AK-based method to estimate failure probabilities associated with rare events. This new method, named AK-MCSi, presents three enhancements of AK-MCS : (i) sequential Monte-Carlo simulations to reduce the time associated with the evaluation of the surrogate model, (ii) a new stricter stopping criterion on learning evaluations to ensure the good classification of the Monte-Carlo population and (iii) a multipoints enrichment permitting the parallelization of the evaluation of the time-consuming model. This work has been published in *Structural Safety* (Lelièvre et al., 2018a).

The last objective of this thesis is to propose new AK-based methods to estimate the failure probability of a high-dimensional reliability problem, i.e. a problem defined by both a time-consuming model and a high number of input random variables. Two new methods, AK-HDMR1 and AK-PCA, are proposed to deal with this problem based on respectively a functional decomposition and a dimensional reduction technique. AK-HDMR1 has been

submitted to *Reliability Engineering and Structural Safety* on 1st October 2018.

**Keywords** : Uncertainties, Reliability, Robustness, Surrogate modeling, Kriging, AK-based methods, Failure probability, Rare events, High Dimension.

---

# Remerciements

Je tiens tout d'abord à remercier mes encadrants Nicolas Gayton (directeur de thèse), Pierre Beaurepaire (co-encadrant) et Cécile Mattrand (co-encadrante). Nicolas, tu m'as énormément apporté depuis le début de mes études à l'IFMA en m'aidant à trouver mes deux stages et surtout en m'ayant soutenu pour rester aux côtés de mon fils au lieu de partir à l'étranger. Tu m'as aussi proposé d'effectuer ces travaux de thèse pendant lesquelles tu as pu m'apporter ta vision et tes conseils avisés. Je te remercie aussi de ta confiance pour m'avoir envoyé parler de mes travaux en France comme à l'étranger. Encore une fois, merci pour tout. Pierre et Cécile, je tiens à vous remercier aussi beaucoup pour tout le travail que vous avez effectué au jour le jour pour que je mène à bien ces travaux de thèse. Même dans des moments difficiles, et surtout un en particulier qui a duré plusieurs mois pendant lesquels aucune solution n'a été trouvée pour ce fameux point d'ancrage, vous avez été toujours là pour m'apporter vos idées et m'encourager à ne pas lâcher l'affaire. Merci à tous les deux.

Je remercie maintenant mes deux rapporteurs : Jérôme Morio et Christian Gogu. J'ai toujours trouvé nos échanges intéressants et constructifs, que ce soit pendant ma soutenance de thèse, pendant les comités de thèse ou pendant la Journée Fiabilité des Matériaux et des Structures à Bordeaux. Je remercie aussi les autres membres du jury en commençant par Bruno Sudret qui m'a fait l'honneur d'accepter de le présider. Encore une fois, merci aussi de nous avoir accueilli comme tu l'as fait à Zurich pour la conférence IFIP en juin dernier. J'ai passé de très bons moments que ce soit dans le cadre scientifique ou en dehors. Merci à Rodolphe Le Riche et Cécile Helbert qui ont accepté d'examiner mes travaux de thèse. Les échanges qui ont suivi ma présentation étaient très intéressants et enrichissants. Enfin, je tiens à remercier Jean-Marc Bourinet qui a lui aussi participé à ma soutenance de thèse. Je te remercie aussi pour toutes les autres discussions que nous avons pu avoir pendant mes trois ans de thèse et pour tes conseils scientifiques très avisés.

Je tiens aussi à remercier deux communautés scientifiques, le groupe Mécanique et Incertain de l'Association Française de Mécanique et l'Institut pour la Maîtrise des Risques, qui m'ont permis de présenter mes travaux à plusieurs reprises à des industriels et à des universitaires de toute la France. Je remercie donc pour cela les animateurs actuels et anciens de ces deux groupes, Nicolas Gayton et Thierry Yalamas pour le GST MI et André Lannoy et Emmanuel Ardillon pour l'IMdR. Merci aussi à Benoit Blaysat pour l'organi-

sation de ses réunions sandwichs qui permettent de parler des problèmes réels rencontrés par les doctorants et non pas uniquement de leurs réussites.

Je remercie aussi Maurice Lemaire qui a été le directeur de thèse de mon directeur de thèse. Les discussions que nous avons pu avoir que ce soit concernant ma thèse ou concernant tout autre sujet m'ont beaucoup apporté. Ta vision des incertitudes et le recul que tu as sur les sciences sont très intéressants et m'ont permis de mieux comprendre les aboutissants de mes travaux de thèse.

Merci aussi à Rodrigue Décatoire qui m'a encadré pendant mon projet de fin d'étude de l'IFMA chez Phimeca Engineering. Du moment où je t'ai fait part de mes intentions de continuer mes études en thèse, tu m'as encouragé et formé pour que j'arrive le plus préparé possible en thèse. Merci beaucoup pour tout. Merci aussi d'être venu voir ma soutenance de thèse, cela m'a particulièrement touché.

Je tiens aussi à remercier mes co-bureaux Van Dong, Dana et Chahrazed pour leur sympathie. Je remercie aussi Jacqueline pour toute son aide au quotidien concernant les problèmes administratifs.

Merci à certains membres du personnel de SIGMA Clermont : Alexis, Alex, Mylène, Arthur, Nicolas, Clément, Adrien, Marie, Romain, Roumar, Gabin, Pierre. Nous avons pu passer de bons moments que ce soit pour nous aider dans nos projets de cadeaux de thèse, pour boire une bière au FizZ ou au Nota ou pour des discussions plus ou moins mouvementées. C'était vraiment sympa d'être en contact avec des personnes en dehors du monde de la recherche.

Je tiens tout particulièrement à remercier la ZAD et ses occupants permanents Héloïse, Puput, Moustache, Lolo, Carlito, Bibi, tacos, JB, Damien, Mathieu, Matthieu, Irmella, et aussi à ses occupants de passage Chabri, Quche, William et Mathias. Merci à tous les anciens pour nous avoir montré le chemin à suivre ou à ne pas suivre. Merci à Héloïse et Puput d'avoir débiter nos deux semaines de soutenance et à Moustache de l'avoir conclue. Enfin, merci à tous de m'avoir supporter moi et mes discussions souvent bien trop longues. La très bonne ambiance de ce bureau a permis de trouver ces trois années beaucoup moins longues. Merci encore pour tous ces moments de rigolade que ce soit à la ZAD, ou ailleurs. Merci beaucoup.

Merci aussi à tous mes amis d'enfance que j'ai pu garder jusque là et à leurs familles qui sont venues s'ajouter à ce petit groupe : Julie, Niag', Cyp', Vinc' et Aurore, Laurie et Quentin, Vaness' et Anthony et leur petite Coline, Léo et Gwenaëlle et leur petite Maïann, Médé et Julie, Typhanie, Romain, Marion et Olivier. Nous ne nous voyons pas autant que je l'aimerais mais c'est toujours avec grand plaisir que nous nous retrouvons. Ça fait vraiment plaisir de vous avoir encore autour de moi depuis si longtemps, jusqu'à près de 24 ans pour certains.

---

Je tiens à remercier toute ma famille. Tout d'abord, merci à mes parents qui m'ont toujours permis de faire ce que je voulais que ce soit financièrement ou moralement. Il faut tout de même savoir que je leur avais vendu l'idée de faire coiffeur parce qu'il n'y avait pas d'étude... dur dur. Merci à mon frère qui m'a montré la voie un an avant moi. Merci aussi à ma belle famille Sylvia et Éric, Franck et Christine, et leurs parents pour tout leur soutien.

Je remercie fortement ma chérie Flora qui m'a subit durant toute ma thèse que ce soit dans les bons moments ou dans les pires pendant lesquels je pouvais me lever en pleine nuit pour corriger mon code. Merci de m'avoir encourager pour tenir bon jusqu'au bout. Merci d'être là à mes côtés.

Enfin, mes derniers remerciements se doivent d'aller à mon fils Nathanaël qui lui aussi a souvent subit mon impatience alors qu'il était incapable de comprendre le pourquoi du comment. Merci beaucoup mon petit bonhomme d'avoir été plutôt exemplaire pendant la période d'écriture de mon manuscrit et de ma préparation de ma soutenance de thèse. Merci beaucoup.

---





# Table des matières

<b>Introduction</b>	<b>1</b>
<b>1 Incertitudes en conception</b>	<b>5</b>
1.1 Introduction . . . . .	6
1.2 Généralités sur les incertitudes . . . . .	6
1.2.1 Sources et classifications des incertitudes . . . . .	6
1.2.2 Prise en compte des incertitudes en conception . . . . .	7
1.3 Éléments de la théorie des probabilités . . . . .	8
1.3.1 Variables aléatoires . . . . .	8
1.3.2 Vecteurs aléatoires . . . . .	9
1.3.3 Processus aléatoires . . . . .	10
1.3.4 Fonctions d'autocorrélation . . . . .	11
1.4 Fiabilité et Robustesse . . . . .	14
1.4.1 Classification des problèmes de conception . . . . .	14
1.4.2 Formulation mathématique et outils de la fiabilité . . . . .	16
1.4.3 Formulations mathématiques et outils de la robustesse . . . . .	17
1.4.4 Formulations mathématiques des problèmes de conception sous incertitudes . . . . .	18
1.5 Revue des méthodes d'analyse de fiabilité . . . . .	21
1.5.1 Simulations de Monte-Carlo . . . . .	21
1.5.2 Tirages d'importance . . . . .	22
1.5.3 Tirages directionnels . . . . .	24
1.5.4 <i>Line Sampling</i> . . . . .	25
1.5.5 Simulations par <i>Subset</i> . . . . .	26
1.5.6 <i>First and Second Order Reliability Methods</i> . . . . .	26
1.6 Conclusions . . . . .	28
<b>2 Méthodes AK pour l'analyse de fiabilité</b>	<b>31</b>
2.1 Introduction . . . . .	32
2.2 Méta-modélisation . . . . .	32
2.2.1 Présentation de quelques métamodèles . . . . .	33
2.2.2 Métamodèles adaptatifs . . . . .	36
2.2.3 Challenges de la méta-modélisation . . . . .	37

2.3	Méta-modélisation par krigeage . . . . .	38
2.3.1	Théorie du krigeage . . . . .	38
2.3.2	Krigeage adaptatif pour l'analyse de fiabilité . . . . .	41
2.4	Méthodes AK . . . . .	45
2.4.1	Méthode AK-MCS . . . . .	46
2.4.2	Méthodes AK pour l'analyse de fiabilité . . . . .	47
2.4.3	Méthodes AK pour l'optimisation sous incertitudes . . . . .	51
2.4.4	Méthodes AK pour d'autres problèmes . . . . .	52
2.5	Conclusions . . . . .	52
<b>3</b>	<b>AK-MCSi – faibles probabilités</b>	<b>55</b>
3.1	Introduction . . . . .	56
3.2	Revue bibliographique des méthodes d'estimation de faibles probabilités de défaillance . . . . .	56
3.2.1	Méthodes de simulation de faibles probabilités . . . . .	56
3.2.2	Méthodes basées sur la métamodélisation pour l'estimation de faibles probabilités . . . . .	58
3.3	Méthode AK-MCSi . . . . .	60
3.3.1	Simulations séquentielles de Monte-Carlo . . . . .	61
3.3.2	Enrichissement multi-points . . . . .	62
3.3.3	Nouveau critère d'arrêt sur l'enrichissement . . . . .	66
3.4	Validation de la méthode AK-MCSi . . . . .	69
3.4.1	Fonction de Rastrigin . . . . .	69
3.4.2	Fonction de performance avec deux domaines de défaillance distincts	74
3.4.3	Fiabilité d'une structure en treillis . . . . .	76
3.4.4	Fiabilité d'un problème de fluage et de fatigue . . . . .	77
3.4.5	Réponse dynamique d'un oscillateur non-linéaire . . . . .	80
3.5	Conclusions . . . . .	83
<b>4</b>	<b>Analyse de fiabilité en grande dimension</b>	<b>85</b>
4.1	Introduction . . . . .	86
4.2	Méthodes de fiabilité en grande dimension . . . . .	86
4.2.1	Challenge de la grande dimension . . . . .	86
4.2.2	Méthodes de simulation en grande dimension . . . . .	87
4.3	Utilisation des métamodèles en grande dimension . . . . .	87
4.3.1	Optimisation directe des hyper-paramètres . . . . .	88
4.3.2	Techniques de réduction de dimension . . . . .	89
4.3.3	Techniques de décompositions fonctionnelles . . . . .	91
4.4	Méthodes AK-HDMR . . . . .	93
4.4.1	Méthode AK-HDMR1 . . . . .	94
4.4.2	Méthode AK-HDMR2 . . . . .	97
4.4.3	Validation et étude des paramètres de la méthode AK-HDMR1 . . .	100
4.5	Méthode AK-PCA . . . . .	104
4.5.1	Présentation de la méthode AK-PCA . . . . .	104
4.5.2	Validation et étude des paramètres de la méthode AK-PCA . . . . .	105

---

---

4.6	Comparaison des méthodes proposées . . . . .	107
4.6.1	Fonction de Valdebenito . . . . .	107
4.6.2	Fonction modifiée de Griewank – Limitations des méthodes proposées	109
4.7	Conclusions . . . . .	110
	<b>Conclusions et perspectives</b>	<b>113</b>
<b>A</b>	<b>Exemple d’illustration</b>	<b>117</b>
<b>B</b>	<b>Illustrations des différents problèmes de conception</b>	<b>119</b>
B.1	Exemple mathématique . . . . .	119
B.2	Application à un container . . . . .	122
<b>C</b>	<b>Algorithmes des méthodes AK pour l’atteinte des faibles probabilités de défaillance</b>	<b>123</b>
C.1	AK-IS . . . . .	123
C.2	metaAK-IS <sup>2</sup> . . . . .	124
C.3	AK-SS . . . . .	125
C.4	AK-SSIS . . . . .	126
C.5	AK-LS . . . . .	127
C.6	AK-RBIS . . . . .	128
<b>D</b>	<b>Démonstration de la reformulation de la décomposition HDMR pour AK-HDMR2</b>	<b>129</b>

---



# Table des figures

1.1	Fonction d'autocorrélation Dirac. . . . .	11
1.2	Fonction d'autocorrélation linéaire. . . . .	12
1.3	Fonction d'autocorrélation exponentielle. . . . .	12
1.4	Fonction d'autocorrélation exponentielle carrée. . . . .	13
1.5	Fonction d'autocorrélation de Matérn. . . . .	13
1.6	Classification des problèmes de conception proposée par Gang et al., (2015). . . . .	15
1.7	Illustration des notions de l'analyse de fiabilité. . . . .	17
1.8	Illustration des simulations de Monte-Carlo. . . . .	23
1.9	Illustration des tirages d'importance. . . . .	24
1.10	Illustration des tirages directionnels. . . . .	25
1.11	Illustration de la technique de <i>Line Sampling</i> avec $\alpha = \overrightarrow{OP^*}$ . . . . .	26
1.12	Illustration des <i>subset simulations</i> . . . . .	27
1.13	Illustration de l'analyse FORM. . . . .	28
1.14	Illustration de l'analyse SORM. . . . .	29
2.1	Exemple d'une architecture simple d'un réseau de neurones artificiels. . . . .	35
2.2	Illustration des fonctions d'apprentissage pour le krigeage adaptatif. . . . .	44
2.3	Famille des méthodes AK. . . . .	45
2.4	Algorithme de la méthode AK-MCS. . . . .	47
2.5	Illustration de la méthode AK-MCS sur la fonction de Rastrigin. . . . .	48
2.6	Exemple de stabilisation de la méthode AK-MCS (Gaspar et al., 2015). . . . .	51
3.1	Procédure de la méthode AK-MCSi. . . . .	60
3.2	Illustration des algorithmes de clustering. . . . .	65
3.3	Fonction de Rastrigin – Représentation 3D. . . . .	69
3.4	Fonction de Rastrigin – État limite. . . . .	70
3.5	Fonction de Rastrigin – Influence de la taille des sous-populations. . . . .	72
3.6	Fonction de Rastrigin – Influence de la probabilité de défaillance. . . . .	74
3.7	Fonction avec deux domaines de défaillance – État limite. . . . .	75
3.8	Structure en treillis – Schéma du problème. . . . .	76
3.9	Structure en treillis – État-limite. . . . .	78
3.10	Fiabilité fluage et fatigue – État-limite. . . . .	79
3.11	Oscillateur non-linéaire – Schéma du problème (Tong et al., 2015). . . . .	80

---

3.12	Oscillateur non-linéaire – État-limite du Cas 1. . . . .	81
3.13	Oscillateur non-linéaire – État-limite du Cas 2. . . . .	82
4.1	Illustration de la projection de la technique <i>cut-HDMR</i> . . . . .	93
4.2	Illustration de la projection du premier plan d’expériences en dimension 2. . . . .	95
4.3	Illustration du schéma d’enrichissement de AK-HDMR1. . . . .	97
4.4	Procédure de AK-HDMR1. . . . .	98
4.5	Projection du premier plan d’expériences de AK-HDMR2 en dimension 3. . . . .	99
4.6	Fonction <i>oblate spheroid</i> – Étude de l’influence des interactions. . . . .	102
4.7	Fonction <i>oblate spheroid</i> – Étude de la sélection du point d’ancrage. . . . .	103
4.8	Structure en treillis – Schéma du problème. . . . .	106
4.9	Fonction de Valdebenito – Comparaison de AK-HDMR1 et AK-PCA. . . . .	108
B.1	Exemple mathématique – Résultats de l’optimisation du Cas 1. . . . .	120
B.2	Exemple mathématique – Résultats de l’optimisation du Cas 2. . . . .	121
B.3	Exemple container – Résultats de l’optimisation. . . . .	122
C.1	Procédure de AK-IS. . . . .	123
C.2	Procédure de metaAK-IS <sup>2</sup> . . . . .	124
C.3	Procédure de AK-SS. . . . .	125
C.4	Procédure de AK-SSIS. . . . .	126
C.5	Procédure de AK-LS. . . . .	127
C.6	Procédure de AK-RBIS. . . . .	128

---

# Liste des tableaux

1.1	Proposition d'une classification des problèmes de conception. . . . .	15
1.2	Classification des problèmes de conception proposée par Braydi et al., (2018). . . . .	15
3.1	Fonction de Rastrigin – Influence du nombre de <i>clusters</i> . . . . .	71
3.2	Fonction de Rastrigin – Influence de la probabilité cible $P_{cible}$ . . . . .	73
3.3	Fonction de Rastrigin – Efficacité de AK-MCS . . . . .	74
3.4	Fonction avec deux domaines de défaillance – Analyse de fiabilité. . . . .	76
3.5	Structure treillis – Variables aléatoires du problème. . . . .	77
3.6	Structure en treillis – Résultats de l'analyse de fiabilité. . . . .	77
3.7	Fiabilité fluage et fatigue – Variables aléatoires du problème. . . . .	79
3.8	Fiabilité fluage et fatigue – Résultats de l'analyse de fiabilité. . . . .	80
3.9	Oscillateur non-linéaire – Variables aléatoires du problème. . . . .	81
3.10	Oscillateur non-linéaire – Résultats de l'analyse de fiabilité du Cas 1. . . . .	82
3.11	Oscillateur non-linéaire – Résultats de l'analyse de fiabilité du Cas 2. . . . .	82
4.1	Fonction <i>oblate spheroid</i> – Validation de AK-HDMR1. . . . .	101
4.2	Fonction <i>oblate spheroid</i> – Validation de AK-HDMR1 avec interactions. . . . .	102
4.3	Structure en treillis – Variables aléatoires du problème. . . . .	106
4.4	Structure en treillis – Nombre de composantes principales conservées. . . . .	106
4.5	Structure en treillis – Étude de AK-PCA en fonction de la probabilité. . . . .	107
4.6	Fonction de Valdebenito – Très grande dimension. . . . .	108
4.7	Fonction de Griewank – Résultats de l'analyse de fiabilité. . . . .	110
A.1	Exemple Illustration – Variables aléatoires du problème. . . . .	117
B.1	Exemple Mathématique – Résultats de l'optimisation du Cas 1. . . . .	120
B.2	Exemple Mathématique – Résultats de l'optimisation du Cas 2. . . . .	121





# Introduction

## Contexte

L'incertain est un concept abstrait et difficile à appréhender même si inhérent à tout phénomène réel. L'acceptation par les Hommes d'un certain risque lié à cet incertain par rapport à leurs travaux a été longue. Dans le domaine de la construction, le premier code régissant l'élaboration de maisons et la responsabilité du bâtisseur en cas d'écroulement est le code d'Hammourabi qui date d'environ 1755 (Lemaire, 2009). Le bâtisseur risquait alors la mort si la maison s'effondrait et tuait le propriétaire. Depuis ce premier code, des réflexions et divers travaux ont été menés concernant la question du risque, proposant des outils pour la quantification desdits risques par le concepteur. Le risque d'incident est défini par le couple de paramètres suivants : gravité de l'incident et probabilité d'occurrence de cet incident. Un événement peut donc être risqué s'il est peu grave mais récurrent ou au contraire si ses conséquences sont catastrophiques même s'il est rare.

Cette problématique de l'incertain reste toujours d'actualité. Bien que les modèles soient toujours plus précis et représentatifs de la réalité, les paramètres d'entrée de tels modèles sont entachés d'incertitudes qui se propagent inévitablement vers la sortie des modèles. L'ingénieur doit donc prendre en compte cet incertain pour s'assurer du bon fonctionnement du produit, de la structure ou des systèmes qu'il conçoit, quelles que soient les conditions réelles d'utilisation. Pour cela, l'indicateur retenu permettant de quantifier cette garantie est la fiabilité, et plus particulièrement la probabilité de défaillance.

La défaillance est caractérisée par un état du produit conçu pour lequel ce dernier n'est plus fonctionnel. Les modèles physiques ou numériques permettent de conclure sur la fonctionnalité ou non de la conception en fonction des valeurs prises par un certain jeu de paramètres. La recherche de la probabilité de défaillance en phase de conception est appelée analyse de fiabilité et requiert la résolution numérique d'une intégrale. Différents algorithmes ont été proposés pour cette résolution, basés sur un nombre, souvent très grand, de valeurs possibles des paramètres d'entrée du modèle considéré. Cependant, comme évoqué précédemment, les ingénieurs utilisent des modèles physiques ou numériques de plus en plus complexes et donc de plus en plus longs à évaluer. En conséquence, l'utilisation des algorithmes mentionnés précédemment pour évaluer la fiabilité est souvent impossible pour des problèmes industriels dans le temps imparti pour la conception.

De nouveaux outils, dénommés métamodèles, ont reçu une attention particulière depuis plusieurs décennies puisqu'ils permettent de remplacer le modèle coûteux par une fonction

mathématique moins coûteuse à évaluer (c'est-à-dire moins longue) à partir de quelques observations seulement dudit modèle. Ces travaux de thèse s'inscrivent dans la continuité des travaux de thèse de Benjamin Échard 2012, intitulée « Evaluation par krigeage de la fiabilité des structures sollicitées en fatigue », financée par le projet ANR APPROFI (Approche mécano-Probabiliste pour la conception Robuste en Fatigue) qui réunissait plusieurs partenaires industriels dont Safran Aircraft Engines. Un des résultats majeurs de ces travaux est la proposition des premières méthodes AK (*Active learning and Kriging based methods*). Ces méthodes majoritairement conçues pour l'analyse de fiabilité se basent sur un schéma adaptatif du métamodèle de krigeage et se montrent efficaces pour la résolution de plusieurs challenges dans le domaine de la fiabilité. Deux types de problèmes restent encore aujourd'hui des défis : la détermination de la probabilité d'occurrence d'un événement ou incident rare, voire très rare, et la considération d'un modèle comptant un très grand nombre de paramètres d'entrée.

## Objectifs des travaux de thèse

Ces travaux de thèse vont donc tout d'abord s'appliquer à un premier objectif de **clarification des formulations du problème de conception** pour l'ingénieur en se basant sur des objectifs parmi les suivants : performance, fiabilité et/ou robustesse du produit conçu. Ces notions s'appuient sur la prise en compte des incertitudes lors de la conception. Cependant, les méthodes traditionnelles permettant de mesurer la fiabilité d'un produit requièrent souvent de nombreux appels au modèle coûteux. L'utilisation du métamodèle de krigeage est donc une bonne alternative mais fait face à deux verrous scientifiques majeurs : (i) l'estimation de faibles probabilités de défaillance et (ii) l'évaluation de probabilités de défaillance liées à des problèmes de fiabilité en grande dimension.

Le second objectif concerne l'estimation de faibles probabilités de défaillance qui impliquent le classement de populations très grandes. Bien que l'évaluation d'un métamodèle est rapide comparée à celle du modèle coûteux, elle requiert tout de même un temps non négligeable lorsque la population est très grande. De plus, la réduction du nombre d'appels au modèle coûteux est toujours un objectif pour l'estimation de probabilités de défaillance, surtout lorsqu'elles sont faibles. Ainsi, le second objectif est le **développement d'une nouvelle méthode AK permettant d'estimer de faibles probabilités de défaillance**.

Le dernier objectif de ces travaux de thèse concerne l'évaluation de probabilités de défaillance liées à des problèmes de fiabilité en grande dimension. Un grand nombre de variables aléatoires d'entrée pose différents problèmes lors de la calibration du métamodèle de krigeage. Celui-ci ne permet donc pas de réaliser une analyse de fiabilité dans un tel contexte. Le troisième objectif est donc **d'adapter les méthodes AK pour des problèmes comptant un grand nombre de variables d'entrée**.

## Structuration du mémoire

Ce mémoire de thèse est organisé en quatre chapitres. Le Chapitre 1 a pour objectif de poser les bases de ces travaux de thèse. Tout d'abord, comme expliqué précédemment, les ingénieurs doivent prendre en compte l'incertain dans la conception pour garantir le

---

---

bon fonctionnement du produit conçu ce qui n'est pas encore entré dans les mœurs. C'est pourquoi, ce chapitre présente le concept d'incertitudes ainsi que leurs sources pour ensuite clarifier le problème de conception en contexte incertain. Les concepts fondamentaux de fiabilité et de robustesse sont ainsi définis par la façon dont les incertitudes sont prises en compte. L'utilisation de l'approche probabiliste est choisie ici pour caractériser les incertitudes dans le problème de conception. Des éléments de la théorie des probabilités sont donc présentés. Enfin, des méthodes traditionnelles pour la résolution de problèmes liés à l'analyse de fiabilité sont aussi brièvement décrites dans ce chapitre.

L'une des difficultés majeures de l'analyse de fiabilité est le temps de calcul nécessaire à l'évaluation de modèles physiques ou numériques. Le Chapitre 2 se focalise ainsi sur une alternative prometteuse aux méthodes traditionnelles : l'utilisation de la méta-modélisation couplée avec les méthodes traditionnelles. Quelques métamodèles couramment utilisés pour l'analyse de fiabilité sont introduits. Deux principaux challenges restent encore aujourd'hui à résoudre : l'estimation de la probabilité d'occurrence d'un événement rare et l'évaluation de probabilités de défaillance pour des problèmes de fiabilité en grande dimension, c'est-à-dire qui sont définis par un grand nombre de variables d'entrée. Ces challenges sont présentés plus en détail dans ce chapitre. Ces travaux de thèse se concentrent sur les méthodes AK basées sur le métamodèle de krigeage. La théorie du krigeage est donc détaillée. Les méthodes AK sont ensuite introduites et se déclinent en trois branches en fonction du domaine d'utilisation : l'analyse de fiabilité, l'optimisation sous incertitudes et la métrologie.

Le troisième chapitre propose d'aborder le premier verrou scientifique de l'analyse de fiabilité mentionné précédemment : la détermination de la probabilité d'occurrence d'événements rares. Un état de l'art des adaptations des méthodes traditionnelles existantes pour résoudre de tels problèmes est tout d'abord présenté. Une amélioration de la méthode à l'origine de la famille des méthodes AK, intitulée AK-MCS (*Active learning and Kriging based Monte Carlo simulations.*), est proposée ici. Elle est nommée AK-MCSi (*improved Active learning and Kriging based Monte Carlo simulations*) et propose trois améliorations. La première concerne la gestion des temps de calcul annexes de la méthode, c'est-à-dire hors temps d'évaluation du modèle physique ou numérique coûteux. La réduction de ces temps de pré et post-traitement est réalisée en modifiant la méthode de simulation inhérente à AK-MCS par une méthode plus efficace. La seconde amélioration a trait à l'utilisation de machines de calculs permettant la parallélisation de ceux-ci ce qui impose de réfléchir au critère initial d'enrichissement. Un nouveau schéma d'enrichissement est ainsi proposé pour intégrer la possibilité de recourir aux calculs distribués. Enfin, le dernier point concerne la convergence de l'algorithme. Un nouveau critère d'arrêt sur l'apprentissage est ainsi introduit pour obtenir une meilleure précision d'estimation de la probabilité de défaillance. La méthode est ensuite étudiée, validée et testée à partir de cinq exemples académiques couramment utilisés dans le domaine de l'analyse de fiabilité.

Le dernier chapitre adresse le second challenge de l'analyse de fiabilité : la grande dimension. Ce challenge est tout d'abord décrit d'un point de vue géométrique permettant d'expliquer l'inefficacité de certaines méthodes traditionnelles d'analyse de fiabilité. Leurs adaptations proposées dans la littérature sont ensuite présentées. L'utilisation des métamodèles dans ce contexte est aussi un défi. Trois orientations peuvent être prises pour permettre la construction d'un métamodèle dans ce contexte ; elles sont présentées dans ce

---

chapitre. Ensuite, deux nouvelles méthodes AK sont introduites. La première méthode est basée sur la décomposition fonctionnelle du modèle physique ou numérique. La seconde méthode cherche à réduire la dimension du problème considéré. Après validation de ces deux méthodes sur des exemples classiques, elles sont comparées sur deux exemples plus complexes d'analyse de fiabilité. Le premier montre l'efficacité des méthodes pour estimer la probabilité de défaillance d'un problème en grande dimension. Le second exemple souligne les limitations des deux méthodes proposées.

---

# Chapitre 1

## Incertitudes en conception

### Sommaire

---

<b>1.1</b>	<b>Introduction</b>	<b>6</b>
<b>1.2</b>	<b>Généralités sur les incertitudes</b>	<b>6</b>
1.2.1	Sources et classifications des incertitudes	6
1.2.2	Prise en compte des incertitudes en conception	7
<b>1.3</b>	<b>Éléments de la théorie des probabilités</b>	<b>8</b>
1.3.1	Variables aléatoires	8
1.3.2	Vecteurs aléatoires	9
1.3.3	Processus aléatoires	10
1.3.4	Fonctions d'autocorrélation	11
<b>1.4</b>	<b>Fiabilité et Robustesse</b>	<b>14</b>
1.4.1	Classification des problèmes de conception	14
1.4.2	Formulation mathématique et outils de la fiabilité	16
1.4.3	Formulations mathématiques et outils de la robustesse	17
1.4.4	Formulations mathématiques des problèmes de conception sous incertitudes	18
<b>1.5</b>	<b>Revue des méthodes d'analyse de fiabilité</b>	<b>21</b>
1.5.1	Simulations de Monte-Carlo	21
1.5.2	Tirages d'importance	22
1.5.3	Tirages directionnels	24
1.5.4	<i>Line Sampling</i>	25
1.5.5	Simulations par <i>Subset</i>	26
1.5.6	<i>First and Second Order Reliability Methods</i>	26
<b>1.6</b>	<b>Conclusions</b>	<b>28</b>

---

## 1.1 Introduction

Concevoir un système ou une structure est une tâche complexe lors de laquelle une bonne représentation des phénomènes réels est recherchée. Pour cela, des modèles sont construits dans lesquels les paramètres d'entrée sont ajustables par le concepteur. Une difficulté majeure pour représenter de tels phénomènes réels est liée à l'incertitude qui entache chacun des paramètres d'entrée du modèle. Ces incertitudes peuvent modifier fortement le comportement du système ou de la structure. Pour formaliser mathématiquement le problème de conception, deux types de fonctions sont introduits : les fonctions objectifs et les fonctions contraintes. Toutes ces fonctions sont représentées par des modèles paramétrés par des variables d'entrée entachées d'incertitudes. La formulation de tels problèmes de conception sous incertitudes est donc fondamentale pour l'obtention de systèmes ou de structures fiables et/ou robustes. L'analyse de fiabilité est le versant des problèmes de conception sous incertitudes particulièrement étudié ici. Après avoir présenté les sources d'incertitudes et leur prise en compte en conception en section 1.2, les concepts de fiabilité et robustesse sont définis en section 1.4. Dans le cadre de cette thèse, une classification des problèmes de conception ainsi que la formulation mathématique de ces problèmes ont été proposées et sont présentées dans cette même section. Des éléments de la théorie des probabilités sont ensuite présentés en section 1.3. Des techniques classiques pour la résolution de problèmes d'analyse de fiabilité sont décrits en section 1.5 avant de conclure ce chapitre.

## 1.2 Généralités sur les incertitudes

### 1.2.1 Sources et classifications des incertitudes

Lors du processus de conception, les ingénieurs se retrouvent face à la présence d'incertitudes affectant toute quantité réelle. Dans la communauté de l'ingénierie mécanique, il est commun de séparer les incertitudes en deux catégories, les incertitudes dites aléatoires et les incertitudes épistémiques (Beyer et Sendhoff, 2007 ; Der Kiureghian et Ditlevsen, 2009). Ces incertitudes qui ne sont dues qu'à un manque de connaissances, sont dites **épistémiques**. Ces incertitudes peuvent être supprimées ou réduites en étudiant plus précisément le phénomène incertain. Les incertitudes de modèles (modèles mathématiques représentant un phénomène physique, modèles probabilistes,...) et de mesures peuvent être considérées comme épistémiques. Les autres incertitudes sont considérées comme **aléatoires**, c'est-à-dire qu'aucune connaissance supplémentaire ne peut diminuer l'incertitude sur ces variables. Elles sont donc inhérentes aux phénomènes étudiés. Le concepteur doit donc "vivre avec" et continuer le processus de conception avec cette réalité (Beyer et Sendhoff, 2007). Typiquement, les phénomènes physiques comme la vitesse du vent, l'humidité ou certaines propriétés de matériaux sont considérés comme naturels et donc aléatoires. D'un point de vue philosophique, toutes les incertitudes dites aléatoires peuvent être considérées comme épistémiques si le phénomène peut être étudié davantage et qu'un modèle très précis peut être proposé. Der Kiureghian et Ditlevsen, (2009) discutent de cette difficulté et insistent sur le fait que selon le "modèle d'univers" choisi par le concepteur, les incertitudes peuvent être soit épistémiques soit aléatoires. L'exemple classique est la propriété d'un matériau qui peut être vu comme aléatoire puisque c'est une propriété physique, mais aussi comme

---

épistémique puisqu'avec des observations avancées, il est possible d'améliorer sa connaissance et donc de diminuer l'incertitude. Cette classification est donc délicate et dépend fortement du concepteur.

Une autre classification a été proposée par plusieurs auteurs (Chen et al., 1996 ; Allen et al., 2006 ; Rolander et al., 2006 ; Beyer et Sendhoff, 2007 ; Baudoui et al., 2012). Cette classification permet de trier les incertitudes en fonction de la possibilité pour le concepteur d'agir ou non sur ces incertitudes. Les incertitudes de **type I** sont celles sur lesquelles le concepteur ne peut pas agir. Elles sont principalement liées aux variables d'environnement et de conditions d'utilisation du produit à concevoir. Les variables liées aux incertitudes de modèles sont aussi de type I. Dans la suite de ce document, ces variables incertaines avec lesquelles le concepteur "doit vivre" seront notées  $P(\omega)$  où  $\omega$  désigne la dépendance de ces paramètres vis à vis de l'aléa. Le **type II** regroupe les incertitudes sur lesquelles le concepteur peut agir (processus de fabrication, matériaux,...). Les variables de type II sont nommées les variables de conception et sont notées  $X(\bar{X}, \omega)$ , où  $\bar{X}$  désigne le paramètre sur lequel le concepteur peut agir (valeur moyenne, nominale,...). Les variables géométriques en sont de bons exemples. Ce type de classification est à privilégier pour une meilleure compréhension des différentes incertitudes en présence lors du processus de conception.

### 1.2.2 Prise en compte des incertitudes en conception

En conception, deux approches peuvent être envisagées pour prendre en compte les incertitudes qui affectent les différentes quantités du problème : l'approche **implicite** et l'approche **explicite**.

La première vise à remplacer chaque paramètre incertain par une valeur de référence appropriée. Dans ce but, plusieurs techniques sont disponibles dans la littérature. L'approche au "pire des cas" est une technique couramment utilisée en ingénierie, et particulièrement dans le domaine de l'analyse des tolérances (voir par exemple Greenwood et Chase, 1987 ; Scholtz, 1995). L'hypothèse majeure et restrictive de cette technique est l'obligation de borner les variables. L'approche au "pire des cas" requiert de sélectionner la combinaison de bornes qui entraîne le pire scénario. En analyse des tolérances, les variables sont bornées dans un intervalle de tolérance, ainsi cette approche est pertinente. Cependant, les variables incertaines sont rarement bornées. La seconde technique ne nécessite pas une telle hypothèse puisqu'elle utilise des facteurs de sécurité. Cette technique consiste à diminuer ou augmenter l'impact de chacune des variables incertaines selon leur caractère résistant ou sollicitant. L'effet des variables à caractère résistant doit être réduit par le facteur de sécurité alors que les sollicitations doivent elles être renforcées. Ainsi, le cas le plus extrême peut être envisagé et garanti par le concepteur (voir par exemple Faber et Sørensen, 2002). Des méthodes alternatives de prise en compte implicite des incertitudes sont aussi disponibles dans la littérature comme la stratégie *Corner space evaluation* ou la *variation patterns formulation* (Yao et al., 2011).

La prise en compte **explicite** des incertitudes dans un problème de conception consiste à caractériser ces incertitudes par une représentation mathématique. Chaque variable aléatoire doit être complètement caractérisée. Ainsi, les approches explicites peuvent donc requérir beaucoup de travaux préliminaires, nécessaire à l'identification des représentations



mathématiques. Cependant, elles permettent de prendre pleinement en considération la variation des variables incertaines et de les propager dans la réponse du système ou de la structure. Plusieurs techniques peuvent être utilisées pour caractériser mathématiquement les variables incertaines. La plus couramment utilisée est l'approche probabiliste (voir par exemple Blatman et Sudret, 2008 ; Dubourg et al., 2013 ; Meng et al., 2015b ; Bourinet, 2016) qui définit la loi de probabilité jointe des variables aléatoires. Des éléments de la théorie des probabilités sont fournis en section 1.3. Cependant, d'autres approches peuvent être utilisées pour caractériser les variables incertaines et sont référencées sous le nom de probabilités imprécises. L'approche possibiliste définit les variables incertaines par une loi de possibilité (Zadeh, 1978). Inuiguchi et Ramík, (2000) et Mousazadeh et al., (2015) en font par exemple usage dans la littérature sur des problèmes d'optimisation. Une autre approche est l'utilisation d'ensembles flous (*fuzzy sets*) (Dubois et Prade, 1980). Les différentes approches explicites pourraient être couplées en utilisant par exemple l'approche probabiliste pour les variables affectées par des incertitudes aléatoires et l'approche possibiliste pour celles affectées par des incertitudes épistémiques (Pedroni et al., 2013). Cependant, seule la théorie des probabilités est utilisée pour prendre en compte les incertitudes dans la suite de ce document.

## 1.3 Éléments de la théorie des probabilités

### 1.3.1 Variables aléatoires

Soit  $(\Omega, \mathcal{F}, \mathcal{P})$  un espace probabilisé avec  $\Omega$  l'espace des possibilités,  $\mathcal{F}$  une tribu<sup>1</sup> sur  $\Omega$  et  $\mathcal{P}$  une mesure de probabilité.

**Définition 1.3.1.** Une *variable aléatoire*  $X$  est une application définie sur  $(\Omega, \mathcal{F}, \mathcal{P})$  et à valeur dans  $\mathbb{S}$ .

$$X : \begin{cases} (\Omega, \mathcal{F}, \mathcal{P}) & \mapsto \mathbb{S} \\ \omega & \mapsto x \end{cases} \quad (1.1)$$

où  $x = X(\omega^{(0)})$  est appelée *réalisation* de la variable aléatoire  $X$ .

Une variable aléatoire est caractérisée par sa fonction de répartition :

$$F_X(x) = \mathcal{P}_X [X(\omega) \leq x] \quad (1.2)$$

Une variable aléatoire est continue si elle admet une densité de probabilité  $f_X(x)$  telle que :

$$f_X(x) = \lim_{h \rightarrow 0; h > 0} \frac{\mathcal{P}_X [x \leq X \leq x + h]}{h} \quad (1.3)$$

Soit :

$$f_X(x) = \frac{dF_X(x)}{dx} \quad (1.4)$$

Dans la suite de ce document, toutes les variables aléatoires sont continues et admettent donc une densité de probabilité.

---

1. Définition : On appelle tribu de  $\Omega$  toute famille  $\mathcal{F}$  de parties de  $\Omega$  satisfaisant les trois conditions suivantes. (i)  $\emptyset \in \mathcal{F}$ . (ii)  $\mathcal{A} \in \mathcal{F} \Rightarrow \mathcal{A}^c \in \mathcal{F}$ . (iii) Si  $(\mathcal{A}_n)_{n \geq 1}$  est une suite de  $\mathcal{F}$  alors  $\cup_{n \geq 1} \mathcal{A}_n \in \mathcal{F}$ .

---

Une variable aléatoire est complètement définie par la donnée de son support de loi et de sa loi de probabilité. Elle peut être caractérisée par ses moments :

**Propriété 1.3.1.** *L'espérance (moment d'ordre 1), notée  $\mathbb{E}[\bullet]$ , caractérise la valeur centrale de la variable. La moyenne d'une variable aléatoire est donc définie par :*

$$\mu_X = \mathbb{E}[X] = \int_x x f_X(x) dx \quad (1.5)$$

**Propriété 1.3.2.** *La variance (moment centré d'ordre 2), caractérise la dispersion de la variable aléatoire autour de sa moyenne et est définie par l'équation suivante :*

$$\sigma_X^2 = \text{Var}[X] = \mathbb{E}[(X - \mu_X)^2] = \int_x (x - \mu_X)^2 f_X(x) dx \quad (1.6)$$

La racine carrée de la variance  $\sigma_X$  est nommée écart-type. Le coefficient de variation d'une variable aléatoire est défini par  $\delta_X = \sigma_X / |\mu_X|$ ,  $\mu_X \neq 0$ .

**Propriété 1.3.3.** *Les moments centrés d'ordre  $k$  permettent de caractériser la forme de la loi de probabilité de la variable aléatoire. Par exemple, le moment centré d'ordre 3 (skewness) caractérise l'asymétrie de la loi de probabilité tandis que le moment centré d'ordre 4 (kurtosis) caractérise l'aplatissement. D'un point de vue général, le moment centré d'ordre  $k$  est défini par :*

$$\mathbb{E}[(X - \mathbb{E}[X])^k] = \int_x (x - \mu_X)^k f_X(x) dx \quad (1.7)$$

La covariance entre deux variables aléatoires  $X$  et  $Y$  est définie par :

$$\text{Cov}[X, Y] = \mathbb{E}[(X - \mu_X)(Y - \mu_Y)] \quad (1.8)$$

Le coefficient de corrélation entre ces deux variables s'écrit :

$$\rho_{X,Y} = \frac{\text{Cov}[X, Y]}{\sigma_X \sigma_Y} \quad (1.9)$$

### 1.3.2 Vecteurs aléatoires

Soit  $(\Omega, \mathcal{F}, \mathcal{P})$  un espace probabilisé avec  $\Omega$  l'espace des possibilités,  $\mathcal{F}$  une tribu sur  $\Omega$  et  $\mathcal{P}$  une mesure de probabilité.

**Définition 1.3.2.** *Un vecteur aléatoire  $\mathbf{X}$  est une application définie sur  $(\Omega, \mathcal{F}, \mathcal{P})$  et à valeur dans un espace  $\mathbb{S}$  de dimension  $d$ .*

$$\mathbf{X} : \begin{cases} (\Omega, \mathcal{F}, \mathcal{P}) & \mapsto \mathbb{S} \\ \omega & \mapsto \mathbf{x} \end{cases} \quad (1.10)$$

où  $\mathbf{x} = \mathbf{X}(\omega^{(0)})$  est appelée réalisation du vecteur aléatoire  $\mathbf{X}$ .

Chaque composante  $\{X_1, X_2, \dots, X_d\}$  du vecteur aléatoire  $\mathbf{X}$  est une variable aléatoire comme définie en section 1.3.1. Les vecteurs aléatoires continus sont définis par leur densité de probabilité jointe  $f_{\mathbf{X}}(\mathbf{x})$  ou par leur fonction de répartition jointe  $F_{\mathbf{X}}(\mathbf{x})$ .

Un vecteur aléatoire  $\mathbf{X}$  est caractérisé par sa moyenne  $\boldsymbol{\mu}_{\mathbf{X}}$  qui est le vecteur des moyennes de chaque composante :

$$\boldsymbol{\mu}_{\mathbf{X}} = \{\mu_{X_1}, \mu_{X_2}, \dots, \mu_{X_d}\} \quad (1.11)$$

et par les matrices de covariance  $\mathbf{C}$  et de corrélation  $\mathbf{R}$  dont les composantes sont définies pour tout  $i, j = 1, \dots, d$  par :

$$C_{ij} = \text{Cov}[X_i, X_j] \quad (1.12)$$

$$R_{ij} = \frac{\text{Cov}[X_i, X_j]}{\sigma_{X_i} \sigma_{X_j}} \quad (1.13)$$

### 1.3.3 Processus aléatoires

Soit  $(\Omega, \mathcal{F}, \mathcal{P})$  un espace probabilisé avec  $\Omega$  l'espace des possibilités,  $\mathcal{F}$  une tribu sur  $\Omega$  et  $\mathcal{P}$  une mesure de probabilité.

**Définition 1.3.3.** *Un processus aléatoire  $H$  est une application définie sur  $(\Omega, \mathcal{F}, \mathcal{P}) \times \mathbb{X}$  où  $\mathbb{X}$  est un espace d'indices spatiaux ou temporels et à valeur dans un espace  $\mathbb{S}$ .*

$$H(\mathbf{x}) : \begin{cases} (\Omega, \mathcal{F}, \mathcal{P}) \times \mathbb{X} & \longmapsto & \mathbb{S} \\ (\mathbf{x}, \omega) & \longmapsto & h(\mathbf{x}) \end{cases} \quad (1.14)$$

Pour  $\omega^{(0)} \in \Omega$  fixé,  $H(\mathbf{x}, \omega^{(0)})$  est une fonction  $h(\mathbf{x})$  de  $\mathbf{x} \in \mathbb{X}$ , appelée réalisation ou trajectoire du processus aléatoire. Pour un  $\mathbf{x}^{(0)} \in \mathbb{X}$  fixé,  $H(\mathbf{x}^{(0)}, \omega)$  est une variable aléatoire sur  $(\Omega, \mathcal{F}, \mathcal{P})$ .

Les moments statistiques du premier et second ordre sont définis ci-dessous.

**Propriété 1.3.4.** *La moyenne  $\mu(\mathbf{x})$  dépend du vecteur d'indices  $\mathbf{x}$ . Cette moyenne peut donc varier sur l'espace  $\mathbb{X}$ .*

$$\mu(\mathbf{x}) = \mathbb{E}[H(\mathbf{x})] \quad (1.15)$$

**Propriété 1.3.5.** *La fonction d'autocovariance s'écrit :*

$$C_{HH}(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(H(\mathbf{x}) - \mu(\mathbf{x}))(H(\mathbf{x}') - \mu(\mathbf{x}'))] = \sigma(\mathbf{x})\sigma(\mathbf{x}')R(\mathbf{x}, \mathbf{x}') \quad (1.16)$$

où  $R$  est une fonction d'autocorrélation et  $\sigma(\mathbf{x})$  (respectivement  $\sigma(\mathbf{x}')$ ) est l'écart-type du processus exprimé en fonction des indices  $\mathbf{x}$  (respectivement  $\mathbf{x}'$ ).

Des exemples de fonctions d'autocorrélation sont donnés en section 1.3.4. Les processus aléatoires possèdent parfois deux autres propriétés : la stationnarité et l'ergodicité.

**Définition 1.3.4.** *Un processus aléatoire est stationnaire si sa moyenne est constante sur l'espace  $\mathbb{X}$  et sa fonction de autocovariance ne dépend que de la différence entre deux points  $\mathbf{x}$  et  $\mathbf{x}'$  :*

$$\mu(\mathbf{x}) = \mu(\mathbf{x}') = \mu \quad (1.17)$$

$$C_{HH}(\mathbf{x}, \mathbf{x}') = C_{HH}(\mathbf{x} - \mathbf{x}') \quad (1.18)$$

**Définition 1.3.5.** *Un processus aléatoire est ergodique si sa moyenne et sa variance peuvent être évaluées à partir d'une unique trajectoire.*

Lorsqu'un processus n'est pas ergodique, il est nécessaire d'observer plusieurs trajectoires du processus pour pouvoir estimer sa moyenne, son écart-type et son autocovariance.

Le processus gaussien stationnaire est complètement caractérisé par la donnée de sa moyenne  $\mu(\mathbf{x})$  et de sa fonction d'autocovariance  $C_{HH}(\mathbf{x}-\mathbf{x}')$ . Pour un  $\mathbf{x}^{(0)}$  fixé, la variable aléatoire suit une loi gaussienne.

### 1.3.4 Fonctions d'autocorrélation

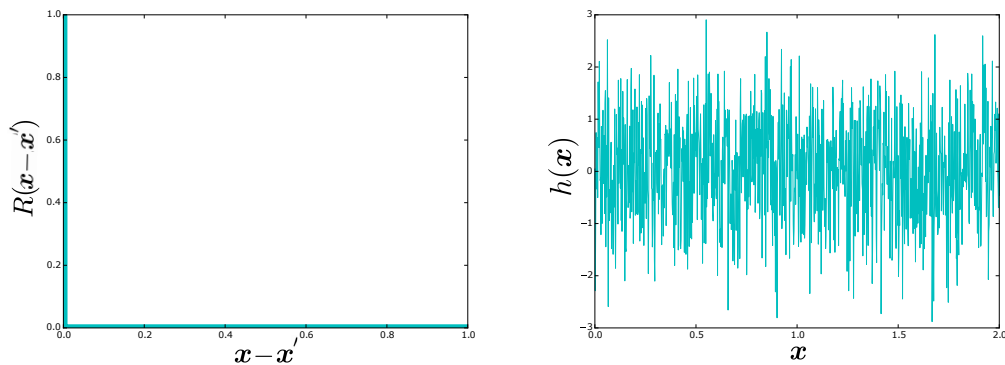
Dans la suite de ce document, les processus aléatoires vont être utilisés. Différentes fonctions d'autocorrélation peuvent être choisies pour la définition de ces processus aléatoires et doivent donc faire l'objet d'une réflexion. Cette section présente une liste non-exhaustive des fonctions d'autocorrélation les plus couramment utilisées dans la littérature.

#### Fonction d'autocorrélation Dirac

La fonction d'autocorrélation Dirac (nugget) est définie par :

$$R(\mathbf{x} - \mathbf{x}') = \delta(\mathbf{x} - \mathbf{x}') \quad (1.19)$$

où  $\delta$  est la fonction Dirac qui est égale à 1 si  $\mathbf{x} - \mathbf{x}' = \mathbf{0}$  et égale à 0 partout ailleurs. Chaque point d'observation est donc indépendant des autres. Cette fonction d'autocorrélation permet donc d'obtenir un processus très bruité, comme un bruit blanc par exemple (voir Figure 1.1).



(a) Représentation de la fonction d'autocorrélation Dirac.

(b) Une trajectoire correspondante associée au processus gaussien de moyenne nulle et de variance unitaire.

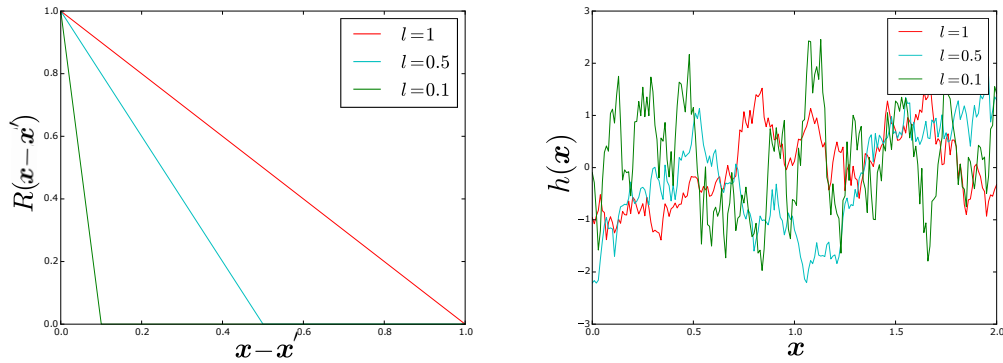
FIGURE 1.1 – Fonction d'autocorrélation Dirac.

#### Fonction d'autocorrélation linéaire

La fonction d'autocorrélation linéaire est définie comme :

$$R(\mathbf{x} - \mathbf{x}') = \prod_{i=1}^n \max\left(0, 1 - \frac{|x_i - x'_i|}{l_i}\right) \quad (1.20)$$

où  $\{l_i > 0, i = 1, \dots, n\}$  est la distance de corrélation (voir Figure 1.2).



(a) Représentation de la fonction d'autocorrélation linéaire.

(b) Trajectoires correspondantes associées au processus gaussien de moyenne nulle et de variance unitaire.

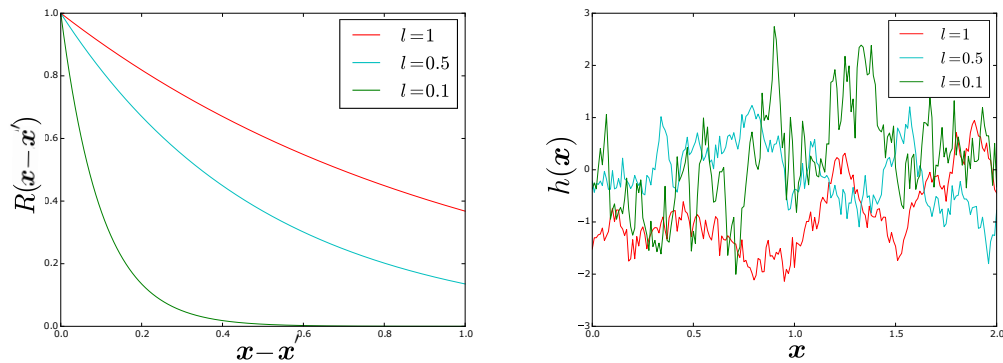
FIGURE 1.2 – Fonction d'autocorrélation linéaire.

### Fonction d'autocorrélation exponentielle

La fonction d'autocorrélation exponentielle est définie par :

$$R(\mathbf{x} - \mathbf{x}') = \exp\left(-\sum_{i=1}^n \frac{|x_i - x'_i|}{l_i}\right) \quad (1.21)$$

où  $\{l_i > 0, i = 1, \dots, n\}$  est l'ensemble des distances de corrélation (voir Figure 1.3).



(a) Représentation de la fonction d'autocorrélation exponentielle.

(b) Trajectoires correspondantes associées au processus gaussien de moyenne nulle et de variance unitaire.

FIGURE 1.3 – Fonction d'autocorrélation exponentielle.

### Fonction d'autocorrélation exponentielle carrée

La fonction d'autocorrélation exponentielle carrée est définie par :

$$R(\mathbf{x} - \mathbf{x}') = \exp\left(-\sum_{i=1}^n \left(\frac{x_i - x'_i}{l_i}\right)^2\right) \quad (1.22)$$

où  $\{l_i > 0, i = 1, \dots, n\}$  est la distance de corrélation (voir Figure 1.4).

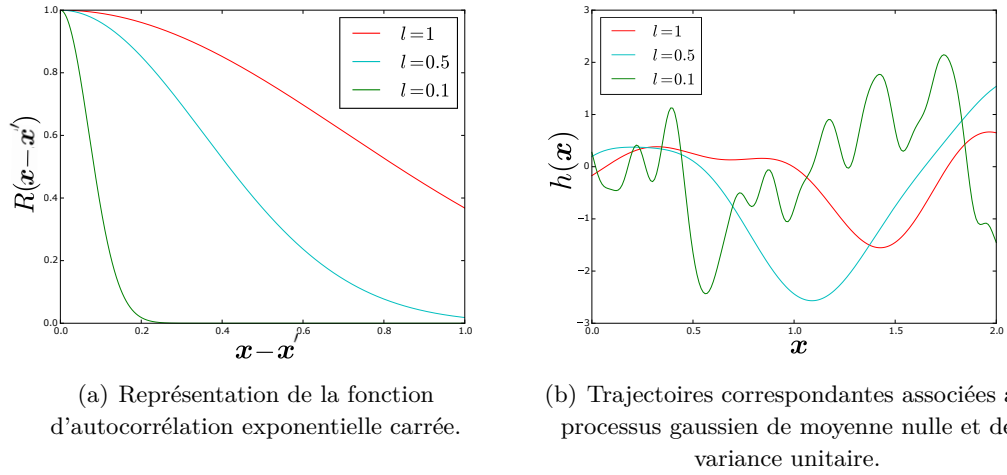


FIGURE 1.4 – Fonction d'autocorrélation exponentielle carrée.

### Fonction d'autocorrélation de Matérn

La fonction d'autocorrélation de Matérn est définie par :

$$R(\mathbf{x} - \mathbf{x}') = \prod_{i=1}^n \frac{1}{2^{\nu-1} \Gamma(\nu)} \left( 2\sqrt{\nu} \frac{|x_i - x'_i|}{l_i} \right)^{\nu} K_{\nu} \left( 2\sqrt{\nu} \frac{|x_i - x'_i|}{l_i} \right) \quad (1.23)$$

où  $\{l_i > 0, i = 1, \dots, n\}$  est la distance de corrélation,  $\nu \geq 1/2$  est le paramètre de forme (voir Figure 1.5),  $\Gamma$  est la fonction Gamma d'Euler et  $K_{\nu}$  est la fonction modifiée de Bessel de seconde espèce :

$$K_{\nu}(z) = \frac{\sqrt{\pi}}{(\nu - \frac{1}{2})!} \left(\frac{1}{2}z\right)^{\nu} \int_1^{\infty} \frac{\cos zt dt}{\sqrt{t^2 + 1}} \quad (1.24)$$

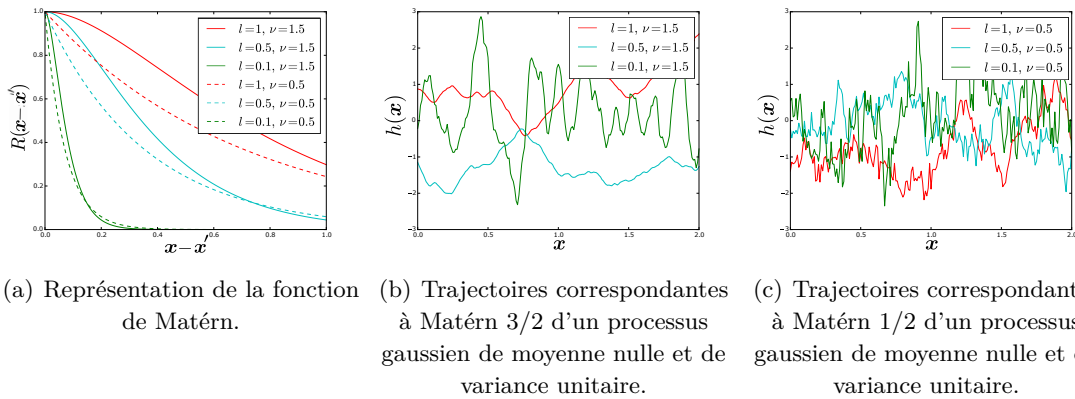


FIGURE 1.5 – Fonction d'autocorrélation de Matérn.

## 1.4 Fiabilité et Robustesse

### 1.4.1 Classification des problèmes de conception

Les problèmes de conception sont caractérisés par deux types de fonction. Tout d'abord, les fonctions de type **objectif** définissent la performance du système ou de la structure étudié. Ces fonctions sont directement associées aux problèmes d'optimisation. En effet, elles sont utilisées pour la recherche de la meilleure solution possible. Ces fonctions sont ici notées  $f(\bullet)$ , fonction des variables de type I et II. Ensuite, les fonctions de type **contrainte** caractérisent l'ensemble des possibles ou l'espace admissible concernant les conditions opérationnelles qui doivent être respectées. Si ces fonctions ne sont pas respectées, alors le système ou la structure ne sera pas fonctionnel. Ces fonctions sont ici notées  $g(\bullet)$  (fonction des variables de type I et II) et l'espace admissible est défini par  $\{\mathbf{X} \in \mathbb{R}^n | g(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) > 0\}$ .

Deux concepts importants sont de plus en plus utilisés pour caractériser les résultats d'un problème de conception : la fiabilité et la robustesse. L'Association Française de Normalisation (AFNOR) définit la fiabilité comme caractérisant la faculté d'un système à assurer ses fonctions dans un contexte donné (voir Lemaire, 2009). Selon la philosophie "Qualité" de Genichi Taguchi, ingénieur et professeur Japonnais qui a eu une grande influence sur l'utilisation des statistiques pour la qualité des produits manufacturés, la robustesse est définie comme caractérisant la faculté de la réponse d'un système à être insensible à de petites variations des paramètres d'entrée du système (voir par exemple Tamhane, 2012). Néanmoins, il est difficile de trouver une unique définition mathématique de ces deux concepts. Gang et al., (2015) proposent une classification des problèmes de conception illustrée par la Figure 1.6. Ils distinguent la prise en compte des incertitudes dans la fonction objectif de la prise en compte de la fiabilité. La robustesse est définie comme étant la considération simultanée des incertitudes dans les deux types de fonctions du problème de conception. Une définition un peu différente de celle-ci a été proposée par Lelièvre et al., (2016a,b). D'après les définitions ci-dessus, la fiabilité correspond au contexte de l'étude et est ainsi représentée par les fonctions contraintes tandis que la robustesse correspond à la réponse du système et est donc liée à la fonction objectif. De ce constat, la fiabilité est définie par la considération des incertitudes dans les contraintes et la robustesse est définie par leur prise en compte dans les objectifs. À partir de ces définitions, le Tableau 1.1 (Lelièvre et al., 2016a,b) présente notre classification des problèmes de conception. Trois états possibles sont définis pour chacun des types de fonction : aucune fonction, fonction à variables déterministes ou fonction à variables incertaines. Huit combinaisons sont donc possibles.

D'autres travaux et réflexions sont disponibles dans la littérature notamment les travaux de Braydi et al., (2018) qui proposent une classification proche mais permettant de sélectionner le type de problème à utiliser. Ils proposent tout d'abord d'étudier l'antagonisme entre la performance (valeur moyenne) et la stabilité de la réponse du système (variance). S'il y a antagonisme, c'est-à-dire s'il existe une différence d'optimum résultant des optimisations de la performance et de la stabilité, alors une optimisation robuste est utile. Si ce n'est pas le cas, alors une optimisation non-robuste donnera le même résultat que l'optimisation robuste. Ces travaux sont résumés dans le Tableau 1.2. La considération des incertitudes pouvant générer un coût considérable, ces travaux permettent de justifier

ou non l'emploi d'optimisation robuste dans les cas étudiés.

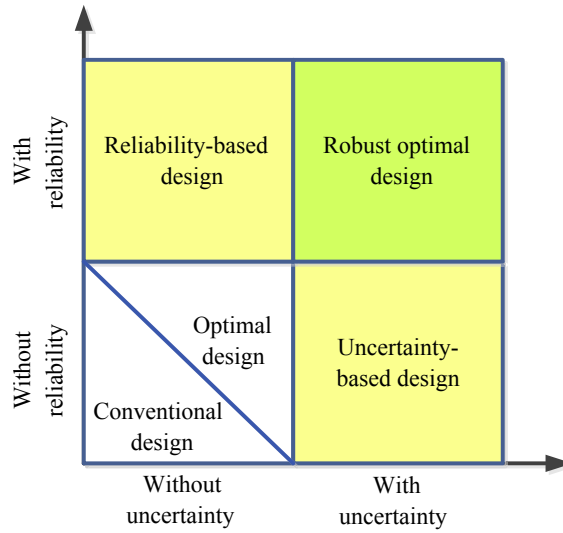


FIGURE 1.6 – Classification des problèmes de conception proposée par Gang et al., (2015).

		Augmentation des efforts numériques →			
		Robustesse →			
Augmentation des efforts numériques ↓	Fiabilité ↓		Aucune fonction objectif	Fonction objectif $X, P$ déterministes	Fonction objectif $X, P$ incertains
		Aucune fonction contrainte		Conception optimale (optimisation sans contrainte)	Conception robuste
		Fonction contrainte $X, P$ déterministes	Conception admissible (dimensionnement)	Conception optimale et admissible (optimisation sous contrainte)	Conception robuste et admissible
		Fonction contrainte $X, P$ incertains	Conception fiable (fiabilité)	Conception optimale et fiable (RBDO)	Conception robuste et fiable (RBRDO)

TABLE 1.1 – Proposition d'une classification des différents problèmes de conception. Les cases grisées correspondent aux problèmes prenant en compte les incertitudes.

Existence des contraintes	Antagonisme Performance-Stabilité	
	N'existe pas	Existe
Non	DO	RDO
Oui	RBDO	RBRDO

TABLE 1.2 – Classification des problèmes de conception proposée par Braydi et al., (2018).



### 1.4.2 Formulation mathématique et outils de la fiabilité

La formulation du problème de fiabilité fait l'objet d'un large consensus dans la communauté scientifique. En effet, lorsque les incertitudes sont prises en compte explicitement par l'approche probabiliste, la fiabilité aboutit à l'estimation d'une probabilité de défaillance  $P_f$  du système étudié et à la garantie que cette probabilité ne dépasse pas un certain seuil  $P_{cible}$  :

$$P_f \leq P_{cible} \quad (1.25)$$

La fiabilité peut aussi être définie comme  $1 - P_f$  mais cette notation n'est pas utilisée dans la suite de ce document.

Dans le cadre de l'optimisation sous contrainte de fiabilité, il est important de conserver la distinction entre variables de type I et de type II puisque le résultat de l'analyse de fiabilité dépend du paramètre  $\bar{\mathbf{X}}$  sur lequel le concepteur peut agir. Cependant, dans un souci de simplification des équations de la fiabilité, les variables de type I et II sont regroupées dans un même vecteur aléatoire  $\mathbf{X} = \{X_1, X_2, \dots, X_n\}$  dont la densité conjointe de probabilité est  $f_{\mathbf{X}}(\mathbf{x})$ . Soit  $g(\mathbf{x})$  la fonction contrainte régissant le problème de conception telle que si  $g(\mathbf{x}) \leq 0$  alors la conception est défaillante.

**Définition 1.4.1.** *La probabilité de défaillance d'une structure est définie par :*

$$P_f = \mathcal{P}[g(\mathbf{X}) \leq 0] = \int_{g(\mathbf{x}) \leq 0} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \quad (1.26)$$

**Définition 1.4.2.** *L'état limite est l'hyper-surface séparant le domaine de sûreté ( $g(\mathbf{x}) > 0$ ) du domaine de défaillance ( $g(\mathbf{x}) \leq 0$ ). L'état limite est donc défini par  $\{\mathbf{x} | g(\mathbf{x}) = 0\}$ .*

Le calcul de l'intégrale présentée en Équation (1.26) est complexe puisqu'il dépend de la dimension du problème, c'est-à-dire du nombre de variables  $X_i$  contenues dans le vecteur aléatoire  $\mathbf{X}$ , ainsi que de la complexité de la fonction contrainte  $g(\mathbf{x})$ . Des méthodes de résolution numérique de cette intégrale sont présentées dans la section 1.5 ; elles utilisent des notions définies ci-dessous.

**Définition 1.4.3.** *L'espace standard est un espace dans lequel toutes les variables aléatoires sont décorréélées et suivent des distributions normales centrées réduites.*

**Définition 1.4.4.** *Différentes méthodes nécessitent de passer de l'espace physique, c'est-à-dire l'espace des variables de conception, à l'espace standard. Une transformation isoprobabiliste  $T$  est définie comme une application bijective de l'espace physique à l'espace standard telle que pour  $\mathbf{X}$  dans l'espace physique,  $\mathbf{U} = T(\mathbf{X})$  est le vecteur correspondant dans l'espace standard. Elle doit garantir que la mesure de la probabilité de défaillance est la même.*

Différentes transformations isoprobabilistes ont été proposées dans la littérature : la transformation de Nataf (Nataf, 1962 ; Liu et Der Kiureghian, 1986) et la transformation de Rosenblatt (Rosenblatt, 1952 ; Hohenbichler et Rackwitz, 1981).

Enfin, certaines méthodes demandent d'identifier un point particulier de l'espace nommé point de défaillance le plus probable (*Most Probable Failure Point* – MPFP) et noté couramment  $P^*$ . L'identification de ce point de défaillance le plus probable est réalisée dans

l'espace standard à partir du problème d'optimisation suivant :

$$\begin{aligned} & \text{Trouver } \mathbf{u}^* \text{ tel que :} \\ & \mathbf{u}^* = \underset{\mathbf{u}}{\text{Arg min}} \|\mathbf{u}\| \\ & \text{Sous } g(\mathbf{u}) = 0 \end{aligned} \quad (1.27)$$

où  $\mathbf{u}^*$  désignent les coordonnées du point de défaillance le plus probable  $P^*$ . Ce point correspond au point de l'état limite le plus proche de l'origine de l'espace standard. La distance séparant le point de défaillance le plus probable de l'origine de l'espace standard est appelé indice de fiabilité et est noté  $\beta$ . Toutes ces notations sont illustrées par la Figure 1.7 sur un exemple défini en Annexe A et qui est utilisé pour toutes les illustrations de ce chapitre.

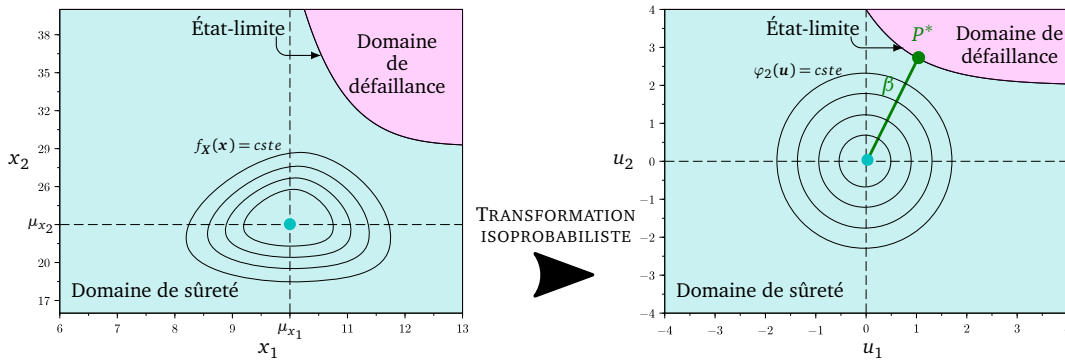


FIGURE 1.7 – Illustration des notions de l'analyse de fiabilité.

### 1.4.3 Formulations mathématiques et outils de la robustesse

Contrairement à la fiabilité, aucun consensus ne peut être dégagé concernant la robustesse (Göhler et al., 2016). Dans la suite, la fonction objectif robuste est notée  $\Psi(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega))$ . Les problèmes de conception robuste cherchent donc à optimiser cette fonction  $\Psi$ . L'idée principale de la robustesse est d'optimiser simultanément l'espérance, c'est-à-dire la valeur moyenne, et la variance, c'est-à-dire la dispersion, de la fonction objectif sujette aux incertitudes (Papadrakakis et al., 2005). Cette idée peut être traduite par une optimisation multi-objectifs (Mourelatos et Liang, 2006 ; Rathod et al., 2013) dans laquelle la fonction à optimiser est définie par :

$$\Psi(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) \equiv \begin{cases} \mathbb{E} \left[ f(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) \right] \\ \text{Var} \left[ f(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) \right] \end{cases} \quad (1.28)$$

La réalisation de cette optimisation multi-objectifs génère un front de Pareto. Aucune suggestion n'est faite sur le choix de la solution. Un critère additionnel est donc nécessaire si une unique solution est souhaitée. Ce problème est l'inconvénient majeur de cette première formulation. Un critère possible pour le choix de la solution est une combinaison linéaire de la valeur moyenne et de la dispersion (Papadrakakis et al., 2005). Cette formulation présentée en Équation (1.29) est facilement implémentable et permet de choisir sur quel

paramètre (valeur moyenne ou dispersion) mettre l'accent par le biais du paramètre  $\lambda \in [0, 1]$ .

$$\Psi(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) \equiv \lambda \times \mathbb{E} [f(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega))] + (1 - \lambda) \times \sqrt{\text{Var} [f(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega))]} \quad (1.29)$$

Quelques alternatives proches sont aussi proposées dans la littérature (Du et al., 2004; Youn et al., 2005; Lee et al., 2008; Yadav et al., 2010; Zaman et al., 2011). Trosset, (1997) et Beyer et Sendhoff, (2007) présentent l'approche de Taguchi. Cette approche quantifie la robustesse par la mesure de la perte de qualité par la fonction d'écart-type moyen de Taguchi notée (MSD). Taguchi ayant longtemps travaillé sur les plans d'expériences, la fonction MSD est définie sur un domaine discret par :

$$MSD(\bar{\mathbf{X}}) = \mathbb{E} [(f(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) - \bar{f})^2] \simeq \frac{1}{p} \sum_{j=1}^p (f(\mathbf{X}(\bar{\mathbf{X}}, \omega_j), \mathbf{P}(\omega_j)) - \bar{f})^2 \quad (1.30)$$

où  $\bar{f}$  est la cible pour la fonction objectif (définie par le cahier des charges). Cette formulation varie selon si l'optimisation est une minimisation, une maximisation ou la recherche d'une cible. Tous les cas sont décrits par Zang et al., (2005). Une alternative à cette expression discrète est définie par l'Équation (1.31).

$$MSD(\bar{\mathbf{X}}) = \text{Var} [f(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega))] + \left( \mathbb{E} [f(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega))] - \bar{f} \right)^2 \quad (1.31)$$

Dans l'objectif de réduire le bruit induit par la fonction  $MSD$ , le ratio Signal/Bruit ( $SNR$ ) est défini par l'Équation (1.32).

$$SNR(\bar{\mathbf{X}}) = -10 \times \log_{10}(MSD(\bar{\mathbf{X}})) \quad (1.32)$$

Contrairement à la combinaison linéaire présentée précédemment (Équation (1.29)), les fonctions de Taguchi ne permettent pas de régler l'importance des paramètres de valeur moyenne et de déviation ce qui représente leur principal inconvénient.

Aucun consensus n'a été trouvé sur la formulation de la robustesse. C'est pourquoi, la notation  $\Psi$  sera utilisée dans la suite de ce document. Il est tout de même important de garder en mémoire que cette fonction  $\Psi$  peut prendre plusieurs formes, certaines ayant été présentées dans cette section.

#### 1.4.4 Formulations mathématiques des problèmes de conception sous incertitudes

Cette section a pour objectif de détailler chacun des problèmes de conception présentés dans le Tableau 1.1. Tous ces problèmes sont illustrés sur deux applications en Annexe B.

##### Conception admissible

La conception admissible, couramment appelée dimensionnement, fournit un ensemble de solutions qui respectent les contraintes. Cette formulation est totalement déterministe et est présentée en Équation (1.33).

$$\text{Trouver } \bar{\mathbf{X}}_{Adm} \text{ tel que : } g(\bar{\mathbf{X}}_{Adm}, \mathbf{P}) > 0 \quad (1.33)$$

Cette formulation peut aussi être utilisée lorsque les incertitudes sont prises en compte de manière implicite avec des facteurs de sécurité ou par l'approche du pire des cas.

### Conception optimale

La conception optimale correspond à une optimisation sans contrainte c'est-à-dire que la solution obtenue est optimale mais potentiellement non-admissible par les contraintes du problème. Cette formulation est donnée en Équation (1.34).

$$\text{Trouver } \bar{\mathbf{X}}_{Opt} \text{ tel que : } \bar{\mathbf{X}}_{Opt} = \underset{\bar{\mathbf{X}}}{\text{Arg max}} f(\bar{\mathbf{X}}, \mathbf{P}) \quad (1.34)$$

Ce type de problème est récurrent dans les applications mathématiques telles que les régressions linéaires et non-linéaires par la méthode des moindres carrés ou le maximum de vraisemblance. De nombreuses méthodes sont proposées dans la littérature (Huang et Lin, 2014; Zhang et Ni, 2015; Andrei, 2016) pour résoudre les problèmes de conception optimale.

### Conception optimale et admissible

La conception optimale et admissible est l'assemblage des deux problèmes de conception précédents sans prise en compte des incertitudes et cherche donc à obtenir l'optimum vis-à-vis de la performance mais admissible pour les contraintes. Cette formulation est présentée par l'Équation (1.35).

$$\begin{aligned} &\text{Trouver } \bar{\mathbf{X}}_{OptAdm} \text{ tel que :} \\ &\bar{\mathbf{X}}_{OptAdm} = \underset{\bar{\mathbf{X}}}{\text{Arg max}} f(\bar{\mathbf{X}}, \mathbf{P}) \\ &\text{Sous } g(\bar{\mathbf{X}}, \mathbf{P}) > 0 \end{aligned} \quad (1.35)$$

Ce problème est un problème d'optimisation classique sous contraintes. Beaucoup de méthodes ont été proposées pour résoudre un tel problème (Rao, 2009) en fonction des différents types d'optimisation : optimisation topologique (Mei et al., 2008), optimisation de forme (Ghabraie et al., 2010) et optimisation multidisciplinaire et de structure (Lee et al., 2014).

### Conception fiable

La conception fiable est une extension de la conception admissible dans laquelle les incertitudes sont prises en compte explicitement. Le résultat d'une telle conception est un espace de solutions respectant la contrainte sur la probabilité de défaillance définie par l'Équation (1.36).

$$\text{Trouver } \bar{\mathbf{X}}_{Fiab} \text{ tel que : } P_f(\bar{\mathbf{X}}) \leq P_{cible} \quad (1.36)$$

La probabilité de défaillance ne doit pas dépasser un certain seuil fixé  $P_{cible}$ . Des méthodes existent pour fixer ce seuil comme les approches basées sur le risque (Rackwitz, 2002). Si l'événement de défaillance est catastrophique, la probabilité cible doit être très faible alors que si l'événement de défaillance est mineur, cette probabilité peut être plus élevée.

### Conception optimale et fiable

La conception optimale et fiable est souvent nommée RBDO pour *Reliability-based Design Optimization*. La formulation est présentée en Équation (1.37) comme le suggèrent Vanmarcke, (1973), Aoues et Chateaufneuf, (2010), Shahraki et Noorossama, (2014) et Meng et al., (2015a,b), c'est-à-dire en considérant une fonction objectif déterministe à maximiser sous contrainte de fiabilité.

$$\begin{aligned} & \text{Trouver } \bar{\mathbf{X}}_{OptFiab} \text{ tel que :} \\ & \bar{\mathbf{X}}_{OptFiab} = \underset{\bar{\mathbf{X}}}{\text{Arg max}} f(\bar{\mathbf{X}}, \mathbf{P}) \\ & \text{sous } P_f(\bar{\mathbf{X}}) \leq P_{cible} \end{aligned} \quad (1.37)$$

La conception optimale et fiable est largement discutée dans la littérature (Acar et Solanki, 2009; Dubourg et al., 2011; Beck et Santana Gomes, 2012). Des formulations alternatives y sont aussi proposées dans lesquelles les incertitudes sont prises en compte dans les deux fonctions objectif et contrainte (Nikolaidis et Burdisso, 1988; Enevoldsen et Sørensen, 1994; Gasser, 1997; Schuëller et Jensen, 2008; Xia et al., 2015).

### Conception robuste

La conception robuste est une optimisation sans contrainte d'une fonction à variables incertaines. Cette formulation présentée en Équation (1.38) n'est que très peu utilisée puisque la majorité des problèmes de conception sont sujets à certaines contraintes (Otto et Antonsson, 1991).

$$\begin{aligned} & \text{Trouver } \bar{\mathbf{X}}_{Rob} \text{ tel que :} \\ & \bar{\mathbf{X}}_{Rob} = \underset{\bar{\mathbf{X}}}{\text{Arg max}} \Psi(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) \end{aligned} \quad (1.38)$$

### Conception robuste et admissible

La conception robuste et admissible présente une fonction objectif robuste soumis à des contraintes déterministes (voir Équation (1.39)). Cette formulation est aussi connue sous le nom de robustesse faisable (Chen et al., 1996; Allen et al., 2006). Cette formulation est peu utilisée dans la littérature.

$$\begin{aligned} & \text{Trouver } \bar{\mathbf{X}}_{RobAdm} \text{ tel que :} \\ & \bar{\mathbf{X}}_{RobAdm} = \underset{\bar{\mathbf{X}}}{\text{Arg max}} \Psi(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) \\ & \text{sous } g(\bar{\mathbf{X}}, \mathbf{P}) > 0 \end{aligned} \quad (1.39)$$

### Conception robuste et fiable

La conception robuste et fiable peut être vue comme la formulation la plus complète puisque les incertitudes sont prises en compte à la fois dans la fonction objectif et dans la fonction contrainte (voir Équation (1.40)). Ce problème est souvent référencé comme RBRDO pour *Reliability-based Robust Design Optimization* dans la littérature (Lee et al., 2008 ; Yadav et al., 2010 ; Rathod et al., 2013 ; Shahraki et Noorossama, 2014). Ce type de conception garantit donc à la fois la faculté du système à assurer ses fonctions dans un contexte donné ainsi que la faculté de sa réponse à être insensible aux petites variations des paramètres d'entrée du système. Cependant, il engendre aussi un effort de calcul pouvant devenir considérable. En effet, la prise en compte des incertitudes que ce soit pour la robustesse ou la fiabilité requiert de nombreux calculs de la fonction considérée qui est souvent coûteuse en temps de calcul dans les problèmes industriels.

$$\begin{aligned} & \text{Trouver } \bar{\mathbf{X}}_{RobFiab} \text{ tel que :} \\ & \bar{\mathbf{X}}_{RobFiab} = \underset{\bar{\mathbf{X}}}{\text{Arg max}} \Psi(\mathbf{X}(\bar{\mathbf{X}}, \omega), \mathbf{P}(\omega)) \\ & \text{sous } P_f \leq P_{cible} \end{aligned} \quad (1.40)$$

## 1.5 Revue des méthodes d'analyse de fiabilité

Dans la suite de ce document, ces travaux de thèse portent uniquement sur l'analyse de fiabilité. Beaucoup de méthodes sont disponibles dans la littérature pour résoudre des problèmes d'analyse de fiabilité. Deux types de méthodes existent : les méthodes de simulation et les méthodes d'approximation. Les premières (Sections 1.5.1-1.5.5) requièrent un tirage aléatoire d'une population sur laquelle la fonction de performance est évaluée. La probabilité de défaillance est ensuite estimée à partir de ces données. Le principal inconvénient de ces méthodes est l'effort numérique nécessaire, particulièrement lorsque la réponse du système est coûteuse à évaluer. Les méthodes d'approximation (section 1.5.6) ont pour objectif d'approximer la fonction d'état limite par une fonction simple. Souvent bien moins coûteuses que les méthodes de simulation, elles peuvent néanmoins causer d'importantes erreurs d'estimation de la probabilité de défaillance puisqu'une hypothèse forte est faite sur la forme de l'état limite, linéaire ou quadratique. Quelques méthodes usuelles d'analyse de fiabilité sont présentées dans la suite de cette section. Le lecteur peut se référer par exemple à (Schuëller et al., 2004 ; El Hami et al., 2017) pour de plus amples informations sur les méthodes d'analyse de fiabilité.

### 1.5.1 Simulations de Monte-Carlo

Les simulations de Monte-Carlo utilisent la relation suivante pour calculer la probabilité de défaillance :

$$P_f = \int_{g(\mathbf{x}) \leq 0} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} = \mathbb{E}[\mathbf{1}(\mathbf{X})] \quad (1.41)$$

où  $\mathbf{1}(\mathbf{X})$  est la fonction indicatrice de défaillance définie par :

$$\mathbf{1}(\mathbf{x}) = \begin{cases} 1 & \text{si } g(\mathbf{x}) \leq 0 \\ 0 & \text{si } g(\mathbf{x}) > 0 \end{cases} \quad (1.42)$$

En pratique,  $N$  réalisations du vecteur  $\mathbf{X}$   $\{\mathbf{x}_i, i = 1, \dots, N\}$  sont tirés aléatoirement suivant la densité conjointe  $f_{\mathbf{X}}$ . La probabilité de défaillance peut ainsi être estimée par l'Équation (1.43). Une illustration est donnée par la Figure 1.8.

$$\hat{P}_f = \frac{1}{N} \sum_{i=1}^N \mathbb{1}(\mathbf{x}_i) \quad (1.43)$$

Soit la variable aléatoire discrète  $Q = \mathbb{1}(\mathbf{X})$ . On note  $q_i = \mathbb{1}(\mathbf{x}_i)$ . L'estimation de la probabilité de défaillance donnée en Équation (1.43) est sans biais, la preuve est donnée en Équation (1.44)

$$\begin{aligned} \mathbb{E}[\hat{P}_f] &= \frac{1}{N} \sum_{i=1}^N \mathbb{E}[q_i] = \frac{1}{N} N \mathbb{E}[Q] \\ &= \mathbb{E}[Q] = P_f \end{aligned} \quad (1.44)$$

La variance de l'estimation de la probabilité de défaillance s'écrit :

$$\begin{aligned} \text{Var}[\hat{P}_f] &= \text{Var}\left[\frac{1}{N} \sum_{i=1}^N q_i\right] = \frac{1}{N^2} \text{Var}\left[\sum_{i=1}^N q_i\right] \\ &= \frac{1}{N^2} \sum_{i=1}^N \text{Var}[q_i] = \frac{1}{N^2} (N \text{Var}[Q]) \\ &= \frac{1}{N} (\mathbb{E}[Q^2] - \mathbb{E}[Q]^2) = \frac{1}{N} (P_f - P_f^2) \\ &= \frac{1}{N} P_f (1 - P_f) \end{aligned} \quad (1.45)$$

Le coefficient de variation de l'estimation de la probabilité de défaillance  $\hat{P}_f$  est donc défini par :

$$\delta_{\hat{P}_f} = \sqrt{\frac{1 - P_f}{N P_f}} \quad (1.46)$$

Ce coefficient de variation caractérise la précision de l'estimation de la probabilité de défaillance. Plus le coefficient de variation est faible, plus l'estimation est précise. Le coefficient de variation dépend de la probabilité de défaillance et du nombre de simulations de Monte-Carlo réalisées. Pour de faibles probabilités de défaillance, il faut donc un nombre de simulations de Monte-Carlo élevé pour obtenir une bonne précision.

### 1.5.2 Tirages d'importance

Dans un souci de réduction de la variance des simulations de Monte-Carlo et donc de la diminution du nombre de simulations, les tirages d'importance (*Importance sampling*) requièrent l'échantillonnage d'une population non pas centrée réduite mais distribuée selon une loi plus pertinente dite instrumentale. La formulation de la probabilité de défaillance présentée en Équation (1.26) devient donc :

$$P_f = \int_{g(\mathbf{x}) \leq 0} \mathbb{1}(\mathbf{x}) \frac{f_{\mathbf{X}}(\mathbf{x})}{h(\mathbf{x})} h(\mathbf{x}) d\mathbf{x} \quad (1.47)$$

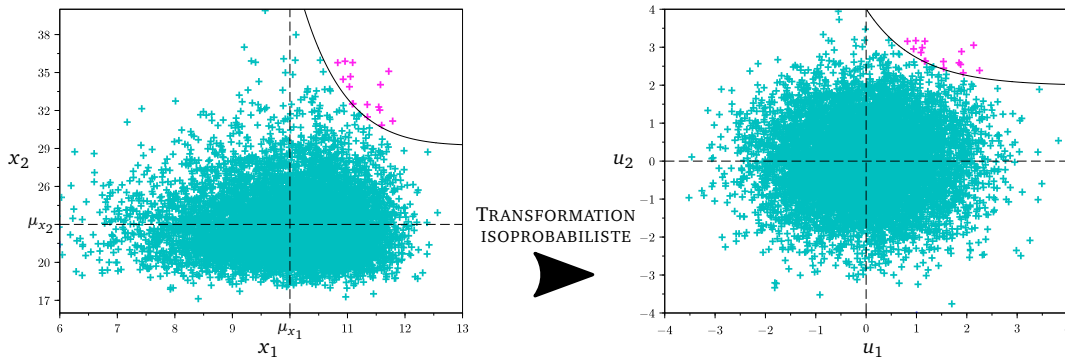


FIGURE 1.8 – Illustration des simulations de Monte-Carlo.

où  $h(\mathbf{x})$  est la fonction de pondération (ou densité instrumentale) induite par le tirage. Cette intégrale peut être estimée par :

$$\hat{P}_f = \mathbb{E} \left[ \mathbf{1}(\mathbf{x}) \frac{f_{\mathbf{X}}(\mathbf{x})}{h(\mathbf{x})} \right] \quad (1.48)$$

L'estimation  $\hat{P}_f$  de la probabilité de défaillance n'est pas biaisée. Soit la variable aléatoire discrète  $Q = \mathbf{1}(\mathbf{X}) \frac{f_{\mathbf{X}}(\mathbf{X})}{h(\mathbf{X})}$ . On note  $q_i = \mathbf{1}(\mathbf{x}_i) \frac{f_{\mathbf{X}}(\mathbf{x}_i)}{h(\mathbf{x}_i)}$ .

$$\begin{aligned} \mathbb{E} [\hat{P}_f] &= \frac{1}{N} \sum_{i=1}^N \mathbb{E} [q_i] = \frac{1}{N} N \mathbb{E} [Q] \\ &= \mathbb{E} [Q] = P_f \end{aligned} \quad (1.49)$$

La variance de la probabilité de défaillance s'écrit :

$$\begin{aligned} \text{Var} [\hat{P}_f] &= \text{Var} \left[ \frac{1}{N} \sum_{i=1}^N q_i \right] = \frac{1}{N^2} \text{Var} \left[ \sum_{i=1}^N q_i \right] \\ &= \frac{1}{N^2} \sum_{i=1}^N \text{Var} [q_i] = \frac{1}{N^2} (N \text{Var} [Q]) \\ &= \frac{1}{N} \left( \mathbb{E} [Q^2] - \mathbb{E} [Q]^2 \right) \\ &= \frac{1}{N} \left( \mathbb{E} \left[ \mathbf{1}(\mathbf{X}) \left( \frac{f_{\mathbf{X}}(\mathbf{X})}{h(\mathbf{X})} \right)^2 \right] - P_f^2 \right) \end{aligned} \quad (1.50)$$

La densité instrumentale optimale dépend de la probabilité de défaillance et est définie telle que :

$$h(\mathbf{x}) = \mathbf{1}(\mathbf{x}) \frac{f_{\mathbf{X}}(\mathbf{x})}{P_f} \quad (1.51)$$

Cependant, la probabilité de défaillance étant inconnue, cette densité instrumentale ne peut pas être identifiée. En pratique, cette fonction de pondération est souvent choisie comme étant la densité de probabilité gaussienne réduite mais non centrée telle que  $h(\mathbf{u}) = \varphi(\mathbf{u} - \mathbf{u}^*)$  où  $\mathbf{u}^*$  sont les coordonnées du point de défaillance le plus probable dans l'espace standard (voir Figure 1.9) (Lemaire, 2009). Le nombre de points appartenant au domaine de défaillance est donc bien supérieur et l'estimation de la probabilité



de défaillance requiert beaucoup moins de simulations. Une hypothèse forte est tout de même formulée : le point de défaillance le plus probable doit être bien identifié (et unique) pour une estimation efficace de la probabilité de défaillance. D'autres méthodes ont été proposées afin d'identifier la fonction de pondération et font l'objet d'une discussion en section 3.2.1.

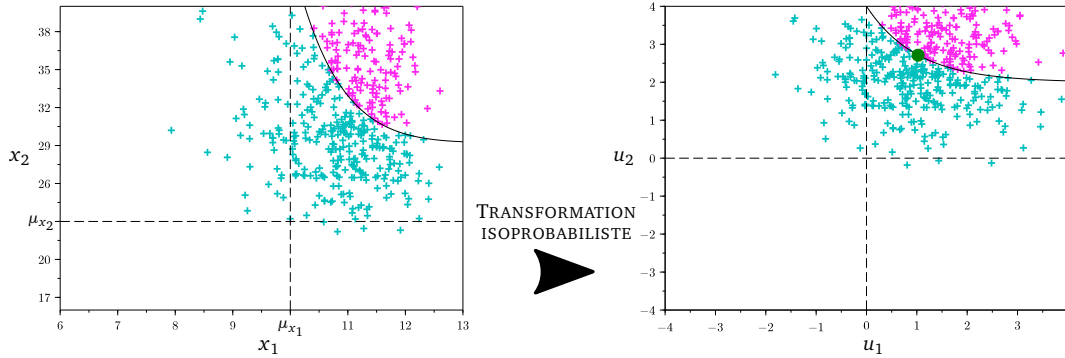


FIGURE 1.9 – Illustration des tirages d'importance.

### 1.5.3 Tirages directionnels

La méthode des tirages directionnels (Bjæger, 1988) est une autre méthode permettant d'estimer la probabilité de défaillance. Elle est basée sur les probabilités conditionnelles et la symétrie de l'espace standard. Des simulations radiales selon différentes directions générées aléatoirement sont réalisées. Soit  $\mathbf{U}$  un vecteur aléatoire dans l'espace standard. Dans le cadre des tirages directionnels, ce vecteur est exprimé tel que :

$$\mathbf{U} = R \times \mathbf{A} \quad (1.52)$$

où  $R^2$  est une variable aléatoire distribuée selon la loi du chi-deux à  $d$  degrés de liberté ( $d$  étant la dimension du problème de fiabilité) et  $\mathbf{A}$  est un vecteur aléatoire unitaire suivant une loi uniforme sur l'hypersphère de rayon unitaire  $\Omega^d$ .  $R$  et  $\mathbf{A}$  sont indépendants. L'intégrale de la probabilité de défaillance (Équation (1.26)) s'écrit comme :

$$P_f = \int_{\mathbf{a} \in \Omega^d} \mathcal{P}[G(R\mathbf{a}) \leq 0] f_{\mathbf{A}}(\mathbf{a}) d\mathbf{a} \quad (1.53)$$

où  $f_{\mathbf{A}}(\mathbf{a})$  est la densité uniforme sur l'hypersphère  $\Omega^d$ . La probabilité de défaillance dans chacune des directions doit donc être calculée. Cette probabilité est exprimée par :

$$\mathcal{P}[G(R\mathbf{a}) \leq 0] = \mathcal{P}[R > r] = \mathcal{P}[R^2 > r^2] = 1 - \chi_d^2(r^2) \quad (1.54)$$

L'estimation d'une telle intégrale est réalisée par les étapes suivantes et est illustrée en Figure 1.10 :

1. Tirage d'une direction  $\mathbf{a}_i$ .
2. Identification du rayon  $r_i$  correspondant à l'intersection de l'état-limite dans la direction  $\mathbf{a}_i$ , c'est-à-dire telle que  $G(r_i \mathbf{a}_i) = 0$ .

3. Après la réalisation des deux premières étapes  $n_D$  fois, estimation de la probabilité de défaillance s'exprimant par :

$$\hat{P}_f = \frac{1}{n_D} \sum_{i=1}^{n_D} [1 - \chi_{n_d}^2(r_i)^2] \quad (1.55)$$

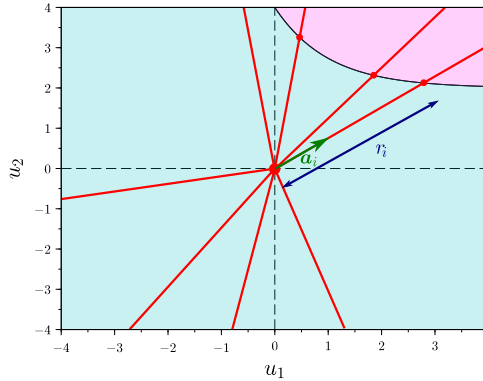


FIGURE 1.10 – Illustration des tirages directionnels.

Le nombre de directions doit être correctement sélectionné pour obtenir une bonne estimation de la probabilité de défaillance. Nie et Ellingwood, (2000) proposent des approches pour l'identification des directions  $\mathbf{a}_i$ . Le nombre de variables aléatoires d'entrée est une limite de la méthode ; elle n'est pas efficace lorsque le problème de fiabilité est en grande dimension.

#### 1.5.4 *Line Sampling*

La technique de *Line Sampling* (Koutsourelakis et al., 2004) s'appuie sur une direction principale notée  $\alpha$ . Une possibilité pour définir cette direction principale est d'utiliser le point de défaillance le plus probable  $P^*$  et ainsi de prendre  $\alpha = \overrightarrow{OP^*}$  où  $O$  est l'origine du repère. La position exacte du point de défaillance le plus probable n'est pas requise pour que cette méthode soit efficace (Schuëller et al., 2004). Les étapes principales de la méthode sont décrites ci-dessous et sont illustrées par la Figure 1.11 :

1. Identification de la direction principale  $\alpha$ .
2. Génération d'un ensemble de  $n_{LS}$  points dans l'espace standard par simulations de Monte-Carlo.
3. Estimation de l'indice de fiabilité  $\beta_i$  de chacune des lignes parallèles à la direction principale  $\alpha$  et passant par un point  $u_i$  de la population de Monte-Carlo.
4. Estimation de la probabilité de défaillance s'exprimant comme :

$$\hat{P}_f = \frac{1}{n_{LS}} \sum_{i=1}^{n_{LS}} \Phi[-\beta_i] \quad (1.56)$$

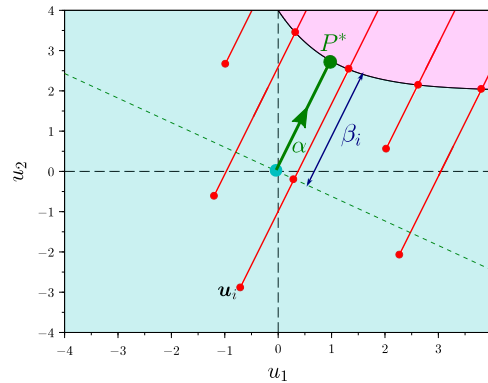


FIGURE 1.11 – Illustration de la technique de *Line Sampling* avec  $\alpha = \overrightarrow{OP^*}$ .

### 1.5.5 Simulations par *Subset*

L'estimation de la probabilité de défaillance est une tâche ardue, particulièrement lorsque celle-ci est supposée faible. Les simulations de Monte-Carlo requièrent alors un très grand nombre de points d'évaluation de la fonction et le temps de calcul augmente fortement. La méthode des *subset simulations* est une méthode permettant de faciliter l'estimation de telles probabilités de défaillance. L'idée principale est de remplacer le problème par une succession de problèmes associés à des événements de défaillance plus probables, c'est-à-dire avec des probabilités de défaillance plus fortes. L'événement de défaillance évolue pendant la méthode de  $E_{f_1} = \{g(\mathbf{x}) \leq y_1\}$  à  $E_f = \{g(\mathbf{x}) \leq 0\}$  en définissant plusieurs seuils entre 0 et  $y_1$ . Une illustration est donnée en Figure 1.12. La probabilité de défaillance est donc un produit de probabilités conditionnelles détaillé en Équation (1.57).

$$P_f = \mathcal{P}[E_f] = \mathcal{P}[E_{f_1}] \times \prod_{s=2}^m \mathcal{P}[E_{f_s} | E_{f_{s-1}}] \quad (1.57)$$

où  $m$  est le nombre d'événements intermédiaires. L'objectif des *subset simulations* est d'estimer chacun des termes du produit par simulations l'un après l'autre. Deux points délicats sont alors à résoudre. Premièrement, le nombre d'événements intermédiaires  $m$  doit être choisis. Pour cela, une simulation de Monte-Carlo classique est réalisée pour estimer la probabilité du premier événement  $E_{f_1}$ . Le seuil  $y_1$  est fixé de telle manière que :

$$\mathcal{P}[E_{f_1}] = \alpha \quad (1.58)$$

où  $\alpha$  est usuellement choisi à 0.1. Ensuite, l'estimation des probabilités conditionnelles  $\mathcal{P}[E_{f_s} | E_{f_{s-1}}]$  est réalisée en utilisant l'algorithme de Metropolis-Hasting modifié par Au et Beck, (2001) pour ce problème.

### 1.5.6 *First and Second Order Reliability Methods*

Les méthodes intitulées *First Order Reliability Method* (FORM) et *Second Order Reliability Method* (SORM) sont des méthodes d'approximation. L'état limite est donc approché par une fonction choisie, respectivement une fonction linéaire et quadratique, afin d'estimer la probabilité de défaillance. Les Figures 1.13 et 1.14 illustrent le principe de ces

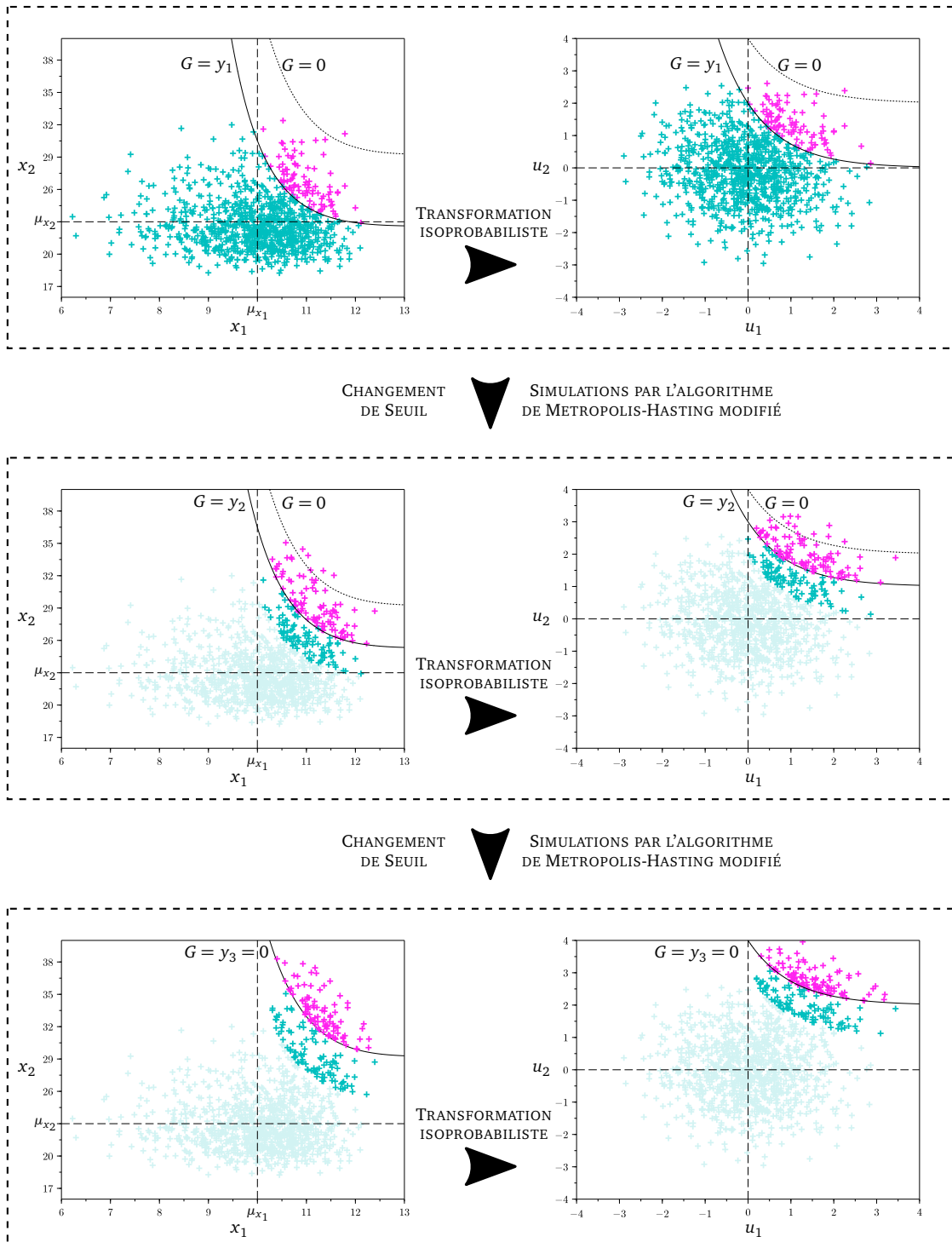


FIGURE 1.12 – Illustration des *subset simulations*.

deux méthodes. Ces deux approximations nécessitent l'identification précise du point de défaillance le plus probable  $P^*$ . Le développement des fonctions linéaire et quadratique se fait à partir de ce point. La probabilité de défaillance est estimée par l'Équation (1.59) pour l'approximation FORM et par l'Équation (1.60) pour l'approximation SORM selon la formulation de Hohenbichler et Rackwitz, (1988).

$$\hat{P}_{f_{FORM}} = \Phi(-\beta) \quad (1.59)$$

où  $\beta$  est l'indice de fiabilité d'Hasofer Lind (voir Figures 1.13 et 1.14) et  $\Phi$  est la fonction de répartition de la variable aléatoire gaussienne centrée réduite.

$$\hat{P}_{f_{SORM}} = \Phi(-\beta) \prod_{i=1}^{n-1} \frac{1}{\sqrt{1 + \psi(\beta)\kappa_i}} \quad (1.60)$$

où  $\psi(\beta) = \varphi(\beta)/\Phi(-\beta)$ ,  $\varphi$  est la densité de probabilité de la variable aléatoire gaussienne centrée réduite et  $\kappa$  est la matrice diagonale des courbures principales de l'état limite en  $p^*$ .

Les méthodes FORM et SORM sont couramment utilisées car elles sont efficaces dans des cas simples et ne demandent pas de coûts de calcul trop importants. Cependant, les limites de telles approximations sont rapidement identifiées. En effet, des hypothèses fortes sont faites sur la forme de l'état limite. Ces méthodes produisent d'importantes erreurs d'estimation lorsque l'on se confronte à des problèmes pour lesquels les états limites sont fortement non-linéaires. Ces méthodes ne sont applicables que si un seul point de défaillance le plus probable est prépondérant. Cependant, de nombreux cas ne correspondent pas à cette dernière hypothèse. Des techniques ont été proposées dans la littérature pour identifier les différents points de défaillance les plus probables (Der Kiureghian et Dakesian, 1998).

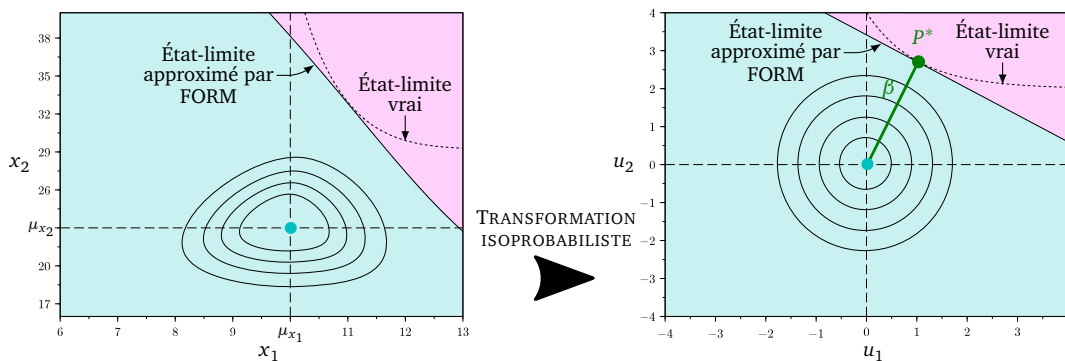


FIGURE 1.13 – Illustration de l'analyse FORM.

## 1.6 Conclusions

Ce chapitre a présenté les bases de la fiabilité, des sources des incertitudes à leur prise en compte dans un problème de conception. Deux classifications de ces incertitudes ont été présentées. La classification aléatoire/épistémique ne va pas être utilisée dans la suite

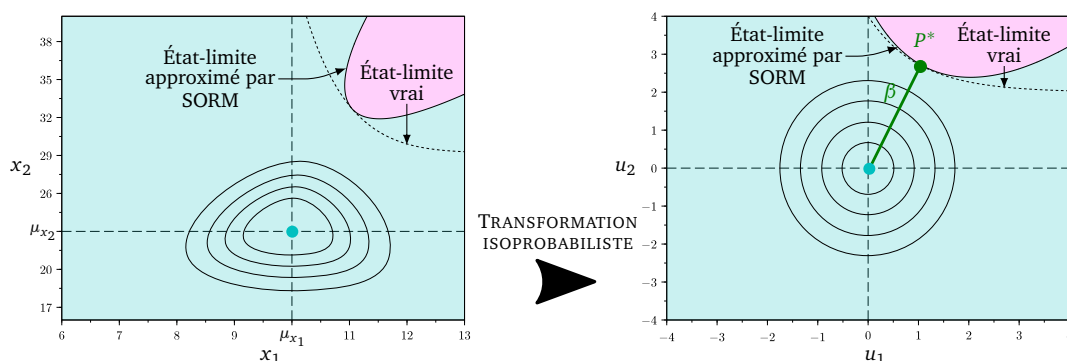


FIGURE 1.14 – Illustration de l'analyse SORM.

due à la part de subjectivité qui est liée. Les incertitudes seront différenciées selon si le concepteur peut ou non agir sur elles.

Une attention particulière a été portée sur la définition de la fiabilité et de la robustesse. Ainsi, la fiabilité est définie par la prise en compte des incertitudes dans la fonction contrainte du problème alors que la robustesse est définie par la considération des incertitudes dans la fonction objectif. Dans le sens de ces deux définitions, l'utilisation de la probabilité de défaillance fait consensus pour la formulation de la fiabilité alors qu'aucune mesure de robustesse n'est encore approuvée par la communauté. Une classification des problèmes de conception a été proposée. Seule l'approche probabiliste, dont des éléments ont été présentés, est utilisée dans la suite du document.

Enfin, ce document se focalise majoritairement sur la résolution de problèmes de fiabilité. Ce premier chapitre présente quelques méthodes classiques pour résoudre de tels problèmes. L'inconvénient majeur des méthodes de simulations (Monte-Carlo, tirages d'importance, *subset simulations*) est le coût de calculs numériques, d'autant plus que la fonction de performance peut être coûteuse, voire très coûteuse (calculs éléments finis par exemple). Les simulations de Monte-Carlo ont un coût qui dépend de la probabilité de défaillance mais elles sont indépendantes de la dimension. Au contraire, les méthodes de simulations basées sur la point de défaillance le plus probable perdent leur efficacité lorsque la dimension augmente. Une discussion plus détaillée de ces assertions est réalisée au Chapitre 4. Concernant les méthodes d'approximation (FORM et SORM), le manque de précision peut souvent être réhibitoire mais leur coût est peu sensible à la probabilité de défaillance. L'augmentation de la dimension peut par contre poser des problèmes d'estimation de la probabilité de défaillance.

La résolution de problèmes d'analyse de fiabilité concernant des fonctions de performance coûteuses est l'objectif de ce document. L'utilisation de métamodèle permet de proposer des méthodes hybrides approximations/simulations permettant d'améliorer la précision concernant les méthodes d'approximations en réduisant fortement les efforts numériques impliqués par les méthodes de simulations.

## Synthèse du chapitre

### Objectifs.

- Établir les bases de la considération des incertitudes en conception : notions de fiabilité, robustesse, probabilité.
- Clarifier les problèmes de conception sous incertitudes et proposer une classification de ces problèmes.
- Introduire les méthodes traditionnelles de fiabilité.

**Proposition.** Une classification des problèmes de conception a été développée en fonction de la prise en compte des incertitudes dans les fonctions objectif et/ou contrainte.

### Publications scientifiques.

- N. Lelièvre et al. (2016a). « On the consideration of uncertainty in design : optimization - reliability - robustness ». In : *Structural and Multidisciplinary Optimization*, p. 1–15
  - N. Lelièvre et al. (2016b). « On the consideration of uncertainty in design : optimization - reliability - robustness ». In : *VII European Congress on Computational Methods in Applied Sciences and Engineering*
-

# Chapitre 2

## Méthodes AK pour l'analyse de fiabilité

### Sommaire

---

<b>2.1</b>	<b>Introduction</b>	<b>32</b>
<b>2.2</b>	<b>Méta-modélisation</b>	<b>32</b>
2.2.1	Présentation de quelques métamodèles	33
2.2.2	Métamodèles adaptatifs	36
2.2.3	Challenges de la méta-modélisation	37
<b>2.3</b>	<b>Méta-modélisation par krigeage</b>	<b>38</b>
2.3.1	Théorie du krigeage	38
2.3.2	Krigeage adaptatif pour l'analyse de fiabilité	41
<b>2.4</b>	<b>Méthodes AK</b>	<b>45</b>
2.4.1	Méthode AK-MCS	46
2.4.2	Méthodes AK pour l'analyse de fiabilité	47
2.4.3	Méthodes AK pour l'optimisation sous incertitudes	51
2.4.4	Méthodes AK pour d'autres problèmes	52
<b>2.5</b>	<b>Conclusions</b>	<b>52</b>

---



## 2.1 Introduction

Avec l'augmentation des capacités de calcul, les modèles numériques des phénomènes et des systèmes étudiés peuvent être de plus en plus complexes, ce qui engendre des temps d'évaluation importants, voire très importants. La plupart des méthodes présentées en Chapitre 1 requiert de très nombreuses évaluations de la fonction de performance, c'est-à-dire du modèle numérique coûteux. Bien que des méthodes de simulation avancées aient été proposées, elles requièrent encore souvent des efforts numériques qui rendent l'analyse de fiabilité souvent inapplicable industriellement. Depuis plusieurs années, la méta-modélisation semble être une approche prometteuse permettant de proposer des méthodes hybrides approximation/simulation. Les méthodes basées sur les métamodèles proposent tout d'abord une approximation de la fonction de performance (à partir de quelques appels au modèle numérique) sur laquelle les méthodes de simulation traditionnelles peuvent être utilisées, réduisant ainsi fortement le nombre d'appels au modèle numérique coûteux.

Ce chapitre se focalise donc sur les méthodes d'estimation de probabilités de défaillance utilisant les métamodèles. Pour cela, des généralités sur la méta-modélisation ainsi que les principes de différents métamodèles sont présentés en section 2.2. Ces méthodes utilisant les métamodèles sont régulièrement confrontées à des difficultés freinant leur efficacité. Ces challenges sont aussi présentés dans cette première section. Ensuite, le métamodèle de krigeage est plus particulièrement abordé dans ce document de part ses propriétés intéressantes pour l'analyse de fiabilité. Ce métamodèle ainsi que des schémas d'enrichissement adaptatifs basés sur celui-ci sont donc plus particulièrement décrits en section 2.3. Parmi ces méthodes adaptatives, les méthodes AK font l'objet d'une attention particulière dans la communauté de la fiabilité; l'article originel (Echard et al., 2011) a été cité 262 fois par la communauté (*source : Scopus Octobre 2018*). En effet, ces méthodes se montrent très avantageuses grâce à une fonction d'apprentissage (c'est-à-dire permettant l'enrichissement) simple et efficace, et grâce à une population à classer générée initialement. Ceci permet d'éviter une optimisation supplémentaire pour l'apprentissage. Cependant, cette population générée initialement impose de devoir recommencer totalement l'analyse de fiabilité si le moindre paramètre des lois de probabilités est modifié. Un état de l'art est donc réalisé sur ces méthodes dans la section 2.4.

## 2.2 Méta-modélisation

Le principe de la méta-modélisation consiste à proposer une fonction mathématique, appelée métamodèle, qui remplace le modèle numérique souvent coûteux représentant le comportement du système ou de la structure étudié. La méta-modélisation est non-intrusive, c'est-à-dire que le modèle numérique coûteux est considéré être une boîte noire dans laquelle il n'est pas possible d'agir. Une telle approche permet de pouvoir évaluer ce métamodèle un grand nombre de fois pour un coût très inférieur à celui inhérent aux modèles numériques. Un métamodèle est calibré à partir d'un ensemble de points, appelé plan d'expériences, pour lequel le modèle numérique a effectivement été évalué. Des méthodes comme les simulations de Monte-Carlo peuvent donc être réalisées sur le métamodèle et non sur le modèle numérique réel permettant un gain de temps très important. Il existe deux approches dans la méta-modélisation : la **classification** et la **régression**.

---

Premièrement, l'approche de **classification** en fiabilité s'appuie sur un ensemble fini, c'est-à-dire une population de points de taille donnée, généré au début de l'analyse qu'il faut séparer en deux classes différentes souvent définies par le signe du métamodèle. Les points correspondant à des valeurs négatives sont considérés défectueux et les points positifs appartiennent au domaine de sûreté. Le principe de la classification est donc de créer une frontière entre les points défectueux et sûrs de la population générée. Seul le signe de la prédiction du métamodèle est étudié, une valeur précise de cette prédiction n'est donc pas nécessaire à la bonne classification de la population. Deuxièmement, l'approche de **régression** cherche à obtenir un métamodèle précis sur tout le domaine de définition de la fonction de performance. Aucune population n'est générée au préalable ; cette approche calibre le métamodèle sur un ensemble continu représentant une zone large du domaine de définition de la fonction à modéliser. Les valeurs exactes de la fonction de performance sont donc étudiées.

L'approche de classification est souvent plus simple à implémenter et à optimiser que l'approche de régression grâce à cette population générée préalablement. Cependant, en régression, le métamodèle possède beaucoup plus d'information qu'un métamodèle en classification et représente de manière plus complète l'espace de définition de la fonction à modéliser. Ainsi, la régression peut être réutilisée si quelques paramètres des lois des variables d'entrée sont modifiés alors que la classification, s'appuyant sur une population générée à partir de ces lois, requiert de recommencer complètement l'analyse de fiabilité.

De nombreux métamodèles ont été présentés dans la littérature (voir par exemple Razavi et al., (2012) et Kajero et al., (2016)). Certains d'entre eux sont présentés dans cette section : les surfaces de réponse polynomiales, la régression par *Moving Least Square*, le chaos polynomial, les réseaux de neurones artificiels et les *Support Vector Machines*. Le krigeage, autre méta-modèle largement diffusé, est l'objet de ce document ; ainsi la section suivante en présente en détail les fondements. La méta-modélisation doit faire face à plusieurs challenges qui freinent son utilisation lorsque des cas complexes se présentent. Ces challenges sont présentés et discutés dans cette section.

### 2.2.1 Présentation de quelques métamodèles

#### Surfaces de réponse polynomiales

La méta-modélisation par surfaces de réponse polynomiales consiste à calibrer une fonction analytique polynomiale permettant le remplacement du modèle numérique. La première apparition de ce métamodèle dans le domaine de la fiabilité est due à Faravelli, (1989) puis il a inspiré de nombreux travaux (voir par exemple Bucher et Bourgund, 1990 ; Enevoldsen et al., 1993 ; Kim et Na, 1997 ; Bourinet et al., 2000 ; Kaymaz et McMahon, 2005, etc).

Une base polynomiale quadratique est le plus souvent utilisée dans la littérature, ainsi le métamodèle s'exprime comme :

$$\hat{G}(\mathbf{x}) = \mathbf{P}(\mathbf{x})^T \times \mathbf{c} \quad (2.1)$$

où  $\mathbf{P}(\mathbf{x}) = \{P_1(\mathbf{x}), \dots, P_p(\mathbf{x})\}$  est la base de fonctions polynomiales utilisées,  $\mathbf{c}$  sont les coefficients à déterminer à partir du plan d'expériences  $\{\mathbf{x}_i, G_i\}$  en minimisant la fonction

de perte quadratique :

$$\mathbf{c}^* = \text{Arg min}_{\mathbf{c}} \frac{1}{N} \sum_{i=1}^N \left( G_i - \hat{G}(\mathbf{x}_i) \right)^2 \quad (2.2)$$

En fiabilité, la construction d'une telle approximation est réalisée dans le voisinage du point de défaillance le plus probable, supposé unique. Pour cela, Gayton et al., (2003) cherchent à calculer des intervalles de confiance autour de ce point de défaillance le plus probable. Selon la dimension du problème, le nombre de coefficients à déterminer peut augmenter rapidement. Ainsi Roussouly et al., (2013) proposent une technique pour sélectionner les termes les plus significatifs de la décomposition. Cependant, de par le choix d'une base de fonctions polynomiales, ce métamodèle est trop « rigide » et ne permet pas d'atteindre des probabilités de défaillance suffisamment précises (Gayton, 2012).

### *Moving Least Square regression*

Le second métamodèle présenté ici est une approximation basée sur les *Moving Least Squares* (Lancaster et Salkauskas, 1981). L'expression *Moving* vient du fait que les coefficients de la régression dépendent de la localisation  $\mathbf{x}$  du point à prédire. Proche des surfaces de réponse polynomiales, il est défini par l'équation suivante :

$$\hat{G}(\mathbf{x}) = F(\mathbf{x})^T \times \mathbf{c}(\mathbf{x}) \quad (2.3)$$

où  $F$  est une base de fonctions ; elle peut être basée sur des polynômes, des splines, etc.  $\mathbf{c}$  sont les coefficients à identifier à l'aide du plan d'expériences en minimisant l'erreur des moindres carrés pondérés :

$$\mathbf{c}^* = \text{Arg min}_{\mathbf{c}} \sum_{i=1}^N \left( G_i - F(\mathbf{x}_i)^T \times \mathbf{c} \right)^2 \theta(\|\mathbf{x}_i - \mathbf{x}\|) \quad (2.4)$$

où  $\theta$  est une fonction de pondération non-négative et  $\|\bullet\|$  désigne la norme euclidienne.

Ce type de métamodèle peut lui aussi être trop « rigide » selon la base de fonctions choisie. L'utilisation de ce métamodèle peut être observée dans la littérature, voir par exemple (Bucher et Most, 2008 ; Proppe, 2008 ; Chowdhury et Rao, 2009b,a ; Kang et al., 2010).

### **Chaos polynomial**

Le chaos polynomial consiste en une décomposition de la réponse du modèle numérique sur une base polynomiale et orthogonale. La représentation d'un tel modèle par un chaos polynomial nécessite en théorie une infinité de termes dans la décomposition. En pratique, une troncature à un nombre fini de  $p$  termes est réalisée, ainsi la décomposition par chaos polynomial est définie par :

$$\hat{G}(\mathbf{x}) = \sum_{i=1}^p c_i \phi_i(\mathbf{x}) \quad (2.5)$$

où  $\{\phi_i\}_{i=1}^p$  est une base orthogonale et polynomiale de l'espace de Hilbert,  $c_i$  sont les coefficients de la décomposition à déterminer. Les coefficients  $c_i$  peuvent être obtenus en

---

minimisant la fonction de perte quadratique, de la même façon que pour les coefficients des réponses de surface polynomiales (voir Équation (2.2)) ou par intégration directe.

L'Équation (2.5)) est la forme généralisée du chaos polynomial (Xiu et Karniadakis, 2002). En effet, le chaos polynomial est basé sur les travaux préliminaires de Wiener, (1938) (voir Ghanem et Spanos, 1991) dans lesquels des variables d'entrée gaussiennes centrées réduites et indépendantes sont considérées et les polynômes d'Hermite sont utilisés.

La méta-modélisation par chaos polynomial dans le cadre de l'analyse de fiabilité est d'abord introduite par Sudret et Der Kiureghian, (2000) puis utilisée plus couramment dans la littérature (Sudret et Der Kiureghian, 2002; Li et Xiu, 2010; Wang et al., 2013; Torii et al., 2017; Kaintura et al., 2018). Ce modèle se montre particulièrement efficace pour l'analyse de sensibilité (voir par exemple Crestaux et al., 2009).

### Réseaux de Neurones Artificiels

Un réseau de neurones artificiels (ANN) est un métamodèle bien connu dans la communauté du *machine learning*. Un réseau de neurones est un ensemble de neurones reliés les uns aux autres par une architecture. Un exemple est donné en Figure 2.1 où un neurone est appelé  $m$ . Chaque neurone est défini par :

$$y_m = f(b_m + c_m^T \mathbf{x}_m) \quad (2.6)$$

où  $\mathbf{x}_m$  est le vecteur de variables d'entrée du neurone  $m$ ,  $y_m$  est sa sortie,  $f$  est une fonction d'activation à fixer,  $b_m$  et  $c_m$  sont respectivement le biais et la pondération correspondant au neurone  $m$  et à identifier à partir de points d'entraînement.

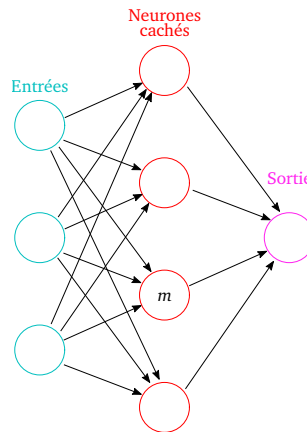


FIGURE 2.1 – Exemple d'une architecture simple d'un réseau de neurones artificiels.

Différents paramètres sont donc à définir au préalable : l'architecture du réseau de neurones et le type de fonction d'activation. Le choix de tels paramètres est complexe et est souvent réalisé par jugement d'expert. Plus de détails concernant l'analyse de fiabilité à partir de réseaux de neurones artificiels peuvent être trouvés dans la littérature (voir par exemple Hurtado et Alvarez, 2001; Hosni Elhewy et al., 2006; Cheng et Li, 2008; Chojaczyk et al., 2015, etc)

### *Support Vector Machines*

Les *Support Vector Machines* (SVM) sont très utilisées dans le cadre de l'analyse de fiabilité. Cette méthode proposée par Vapnik et Lerner, (1963) et améliorée par Boser et al., (1992) peut être utilisée selon deux approches différentes : la classification avec les SVC pour *Support Vector Classification* ou la régression avec les SVR pour *Support Vector Regression*. Bourinet (2018) donne davantage de détails concernant les SVM.

**La classification.** L'idée originelle des SVM était de classer un ensemble de points de l'espace en différentes classes. Dans le cadre de l'analyse de fiabilité, les SVC peuvent donc être directement implémentables pour permettre de classer une population dans les domaines de sûreté et de défaillance. Un hyperplan et une marge sont construits en maximisant la distance entre les points d'apprentissage les plus proches de la marge et de l'hyperplan. Lorsque le séparateur de classes n'est pas linéaire alors une fonction à noyau doit être utilisée permettant de transformer l'espace de travail en un espace dans lequel le séparateur est un hyperplan. Différentes fonctions à noyau sont disponibles dans la littérature (voir Deheeger et Lemaire, 2006). Les méthodes de fiabilité basées sur les SVC sont idéales pour l'utilisation des méthodes de simulation (Monte-Carlo) (voir par exemple Li et al., 2006 ; Most, 2007 ; Basudhar et Missoum, 2010 ; Bourinet et al., 2011, etc). Pour obtenir une bonne classification en utilisant les SVC de manière adaptative, un point défaillant doit être identifié et présent dans le premier plan d'expériences (Li et al., 2006).

**La régression.** Les SVC peuvent être généralisées dans le cadre de la régression (Vapnik, 1995). Le principe est de remplacer la marge des SVC par un paramètre d'erreur sur la déviation du métamodèle par rapport au modèle numérique. Les SVR sont moins utilisées que les SVC puisque beaucoup de problèmes peuvent être facilement transformés en problèmes de classification. Néanmoins des applications ont tout de même été proposées dans le cadre de l'analyse de fiabilité (Smola et Schölkopf, 2004 ; Pai, 2006 ; Yuan et al., 2009 ; Moura et al., 2011 ; Bourinet, 2016).

### 2.2.2 Métamodèles adaptatifs

L'inconvénient majeur associé à l'utilisation d'un métamodèle classique est le besoin de connaître la réponse de la structure assez précisément sur la totalité de l'espace étudié. Le plan d'expériences doit donc avoir une taille suffisamment grande et l'évaluation de la fonction de performance sur ce plan d'expériences peut être coûteuse en temps de calcul. L'objectif majeur des schémas adaptatifs basés sur les métamodèles est donc de commencer l'étude souhaitée avec un plan d'expériences réduit puis de l'enrichir dans les zones d'intérêt. Ces zones à enrichir sont choisies par une fonction d'apprentissage. Dans le domaine de l'analyse de fiabilité, la zone qui doit être la mieux connue est le voisinage de l'état limite, c'est-à-dire la frontière en les domaines de défaillance et de sûreté, qui est donc inconnu au début de l'étude. La procédure classique d'un schéma adaptatif est la suivante :

---

1. Calibration d'un premier métamodèle basée sur le premier plan d'expériences de taille réduite.
2. Sélection d'un ou plusieurs points à enrichir à partir d'un critère d'enrichissement basée sur une fonction d'apprentissage.
3. Évaluation de la vraie fonction de performance coûteuse sur ces points d'apprentissages.
4. Mise à jour du plan d'expériences.
5. Calibration d'un nouveau métamodèle basée ce nouveau plan d'expériences.
6. Évaluation du critère d'arrêt sur l'apprentissage. S'il est respecté, le plan d'expériences est suffisant et le dernier métamodèle est jugé convenable. Sinon, la phase d'apprentissage continue.

Ce type de schéma permet d'optimiser le nombre d'appels à la fonction de performance coûteuse selon le problème considéré.

Différents schémas basés sur les métamodèles, présentés dans la section précédente, ont été proposés dans la littérature : les surfaces de réponse polynomiale adaptative (voir par exemple Roussouly et al., 2013), le chaos polynomial adaptatif (par exemple Cheng et Lu, 2018), les réseaux de neurones artificiels adaptatifs (voir par exemple Al-Masri et al., 2015) ou les SVM adaptatives (Deheeger, 2008 ; Bourinet et al., 2011 ; Bourinet, 2018).

### 2.2.3 Challenges de la méta-modélisation

Comme défini précédemment, l'objectif de la méta-modélisation est de remplacer un modèle numérique coûteux par un modèle mathématique permettant de réduire fortement le temps des diverses analyses souhaitées. Le temps de calcul lié à l'évaluation du modèle numérique doit donc être réduit à son minimum en réduisant le plus possible la taille du plan d'expériences permettant la calibration du métamodèle. Un modèle coûteux est toujours considéré dans la suite. Deux difficultés majeures viennent entacher cet objectif : (1) l'estimation de faibles probabilités de défaillance et (2) la grande dimension en termes de nombre de variables d'entrée du modèle numérique. Une troisième difficulté peut aussi être envisagée dans un contexte différent de celui considéré ici. En contradiction avec la recherche d'un plan d'expériences toujours plus petit : (3) un grand nombre de points d'apprentissage, connu sous le nom de *Big Data*, est aussi un véritable challenge.

Ces trois challenges sont discutés dans la suite de cette section. Seuls les challenges (1) et (2) font l'objet de plus amples discussions et de propositions de nouvelles méthodes dans les chapitres suivants.

#### Challenge (1) : l'estimation de faibles probabilités de défaillance

L'atteinte de faibles probabilités de défaillance est un challenge identifié depuis plusieurs années. De nombreuses méthodes ont été proposées que ce soit par des approches de simulation directes ou basées sur des métamodèles. Les méthodes de simulation directes, c'est-à-dire basées sur le modèle numérique coûteux, requièrent l'évaluation de ce dernier en un nombre très important de points et deviennent donc irréalisables industriellement.

---

Bien que les métamodèles réduisent fortement le nombre de points à évaluer sur le modèle coûteux, lorsque de faibles probabilités de défaillance sont à estimer, la taille du plan d'expériences augmente inévitablement pour explorer des zones très restreintes de l'espace. C'est pourquoi, c'est encore aujourd'hui un challenge qui fait l'objet de nombreuses recherches récentes (voir par exemple Bourinet, 2016 ; Huang et al., 2016 ; Xiong et Tan, 2018, etc). Ce challenge est discuté au Chapitre 3.

### Challenge (2) : La grande dimension

La dimension de l'espace des variables d'entrée est un élément limitant pour l'analyse de fiabilité et plus particulièrement lorsque des métamodèles sont utilisés. Un grand nombre de variables aléatoires en entrée d'un modèle numérique cause d'importantes difficultés pour la calibration d'un métamodèle à partir d'un nombre acceptable de points dans le plan d'expériences. La majorité des métamodèles souffre donc de l'augmentation de la dimension et ces derniers deviennent impossible à utiliser, particulièrement lorsque la fonction de performance est coûteuse. Ce challenge, bien connu sous le terme anglais de *curse of the dimensionality* (Bellman, 1961), est un frein à l'analyse de fiabilité qui est donc restreinte aux problèmes avec peu de variables aléatoires ou à l'utilisation de métamodèles à fortes hypothèses comme par exemple la linéarité. Le problème de la grande dimension est discuté en Chapitre 4.

### Challenge (3) : Le *Big Data*

Avec l'augmentation des capacités de stockage et l'utilisation croissante de capteurs, le nombre de données disponibles concernant les systèmes, structures ou phénomènes physiques étudiés peut devenir massif. Le *Big Data* n'est pas souvent observé dans le domaine de l'analyse de fiabilité puisque classiquement la fonction de performance est coûteuse et peu d'évaluations sont possibles. Les problèmes d'analyse de fiabilité sont souvent confrontés au problème inverse, celui du *Small Data*. Cependant, lorsque la dimension du problème de fiabilité augmente, le plan d'expériences peut devenir très grand et le métamodèle souffrir du *Big Data*. D'autres domaines se servant de la méta-modélisation peuvent aussi souffrir du *Big Data*. Ce grand nombre de données cause des difficultés pour certains métamodèles lors de la calibration de ces derniers notamment s'il est souhaité que le métamodèle soit interpolant. La calibration de tels métamodèles requiert souvent une optimisation qui devient donc délicate. Ce challenge a été étudié concernant par exemple le krigeage (voir par exemple Sakata et al., 2004 ; Hensman et al., 2013) ou les SVM (voir par exemple Cervantes et al., 2008 ; Rizzi, 2016).

## 2.3 Méta-modélisation par krigeage

### 2.3.1 Théorie du krigeage

Le krigeage, développé par Krige, (1951) et théorisé par Matheron, (1973), permet d'approximer la fonction de performance  $G$  par une réalisation d'un processus gaussien noté

---



$\mathcal{G}(\mathbf{x})$ . L'objectif d'un tel métamodèle est de prédire la valeur de la fonction de performance en des points non-observés  $\mathbf{x}$  à partir d'une base de points d'apprentissage  $\mathbf{x}_{obs}$ .

### Principe

Le modèle de processus gaussien est défini par :

$$\mathcal{G}(\mathbf{x}, \omega) = \mathbf{F}^T(\mathbf{x})\boldsymbol{\beta} + Z(\mathbf{x}, \omega) \quad (2.7)$$

où

- $\mathbf{F}(\mathbf{x})\boldsymbol{\beta}$  est la partie déterministe et représente la tendance moyenne du processus.  $\mathbf{F}(\mathbf{x})$  est la matrice des fonctions d'une base donnée.  $\boldsymbol{\beta}$  est le vecteur de coefficients de la régression. Dans la suite de ce document, le krigeage ordinaire est utilisé, c'est-à-dire que le vecteur de coefficients de la régression  $\boldsymbol{\beta}$  est réduit à un unique scalaire  $\beta$  et la matrice  $\mathbf{F}(\mathbf{x})$  est réduite au vecteur  $\mathbf{F} = \{1, 1, \dots, 1\}^T$ . Le modèle de régression ordinaire est souvent suffisant même sur des fonctions fortement non-linéaires (Bichon et al., 2011), l'information étant alors exclusivement portée par la fonction d'autocorrélation. L'identification du paramètre  $\beta$  est présentée dans la section suivante.
- $Z(\mathbf{x}, \omega)$  est un processus gaussien stationnaire de moyenne nulle (voir Équations (2.8) et (2.9)).

$$\mathbb{E}[Z(\mathbf{x}, \omega)] = 0, \quad \forall \mathbf{x} \in \mathbb{X} \quad (2.8)$$

$$C_{ZZ}(\mathbf{x}, \mathbf{x}') = \sigma^2 R(\mathbf{x} - \mathbf{x}', \boldsymbol{\theta}) \quad (2.9)$$

où  $\sigma^2$  correspond à la variance du processus. Le processus étant stationnaire, la fonction d'autocorrélation  $R$  ne dépend que de la différence  $\mathbf{x} - \mathbf{x}'$  et d'un ensemble d'hyper-paramètres  $\boldsymbol{\theta}$ . Des exemples de fonctions d'autocorrélation ont été présentés en section 1.3.4. Dans la suite de ce document, un noyau gaussien est utilisé pour la fonction d'autocorrélation. L'identification des hyper-paramètres est présentée ci-dessous.

### Identification des hyperparamètres de krigeage

Soit un plan d'expériences défini par  $m$  points  $\{\mathbf{x}_{obs}^{(1)}, \mathbf{x}_{obs}^{(2)}, \dots, \mathbf{x}_{obs}^{(m)}\}$  et leurs observations obtenues par la fonction de performance  $\mathbf{G}_{obs} = \{G(\mathbf{x}_{obs}^{(1)}), G(\mathbf{x}_{obs}^{(2)}), \dots, G(\mathbf{x}_{obs}^{(m)})\}$ . L'identification des hyper-paramètres  $\boldsymbol{\beta}, \sigma^2$  et  $\boldsymbol{\theta}$  est ici réalisée par la méthode du maximum de vraisemblance décrite ci-dessous. D'autres techniques permettent d'identifier les hyper-paramètres du krigeage comme la validation croisée par exemple mais ne seront pas présentées dans ce document car non utilisées.

La vraisemblance des observations  $\mathbf{G}_{obs}$  s'écrit :

$$\mathcal{L}(\mathbf{G}_{obs} | \boldsymbol{\beta}, \sigma^2, \boldsymbol{\theta}) = \frac{1}{((2\pi\sigma^2)^m [\det \mathbf{R}(\boldsymbol{\theta})])^{\frac{1}{2}}} \exp \left[ -\frac{1}{2\sigma^2} (\mathbf{G}_{obs} - \mathbf{F}^T \boldsymbol{\beta})^T \mathbf{R}(\boldsymbol{\theta})^{-1} (\mathbf{G}_{obs} - \mathbf{F}^T \boldsymbol{\beta}) \right] \quad (2.10)$$

Les hyper-paramètres  $\boldsymbol{\beta}$  et  $\sigma^2$  peuvent être estimés par les Équations (2.11) et (2.12). Ces estimateurs ne dépendent que des hyper-paramètres  $\boldsymbol{\theta}$ . Le krigeage isotrope est défini



par la considération d'un unique hyper-paramètre  $\theta$  qui est donc réduit à un scalaire. En krigeage complètement anisotrope, le nombre de composantes de  $\theta$  est égal à la dimension. Dans la suite de ce document, le krigeage anisotrope est toujours choisi pour obtenir une approximation précise.

$$\hat{\beta} = \left( \mathbf{F}^T \mathbf{R}(\theta)^{-1} \mathbf{F} \right)^{-1} \mathbf{F}^T \mathbf{R}(\theta)^{-1} \mathbf{G}_{obs} \quad (2.11)$$

$$\hat{\sigma}^2 = \frac{1}{m} (\mathbf{G}_{obs} - \mathbf{F}^T \hat{\beta})^T \mathbf{R}(\theta)^{-1} (\mathbf{G}_{obs} - \mathbf{F}^T \hat{\beta}) \quad (2.12)$$

L'identification des hyper-paramètres  $\theta$  est réalisée par la maximisation de l'Équation (2.10). Les méthodes classiques d'optimisation ont pour objectif principal de minimiser une fonction. Ainsi, le problème de maximisation est transformé en minimisation en considérant la fonction suivante :

$$-\log \mathcal{L}(\mathbf{G}_{obs} | \beta, \sigma^2, \theta) = \frac{m}{2} \log \left( \hat{\sigma}^2(\theta) [\det \mathbf{R}(\theta)]^{\frac{1}{m}} \right) + \frac{m}{2} (\log(2\pi) + 1) \quad (2.13)$$

La minimisation de l'Équation (2.13) peut donc être réduite au problème d'optimisation suivant :

$$\hat{\theta} = \text{Arg min}_{\theta} \hat{\sigma}^2(\theta) [\det \mathbf{R}(\theta)]^{\frac{1}{m}} \quad (2.14)$$

Ce problème d'optimisation est complexe et ne peut pas être résolu analytiquement. Des méthodes de résolution numérique doivent donc être utilisées comme par exemple la méthode de Hooke et Jeeves (Lophaven et al., 2002) qui est utilisée par la *toolbox* DACE ou la méthode CMA-ES (Hansen et al., 2003).

### Prédiction par krigeage

L'objectif principal de la méta-modélisation par krigeage est de prédire la fonction de performance  $G$  en des points non-observés  $\mathbf{x}$ . Dans la suite de cette section, les hyper-paramètres sont supposés identifiés.

L'estimateur de krigeage  $\hat{G}(\mathbf{x})$  pour un  $\mathbf{x}$  non-observé (BLUP *Best Linear Unbiased Predictor*) est la variable aléatoire définie comme :

— linéaire :

$$\hat{G}(\mathbf{x}) = \sum_{i=1}^m a_i(\mathbf{x}) G(\mathbf{x}_{obs}^{(i)}) = \mathbf{a}^T \mathbf{G}_{obs} \quad (2.15)$$

— sans biais :

$$\mathbb{E} \left[ \hat{G}(\mathbf{x}) - G(\mathbf{x}) \right] = 0 \quad (2.16)$$

— meilleur au sens de l'erreur quadratique moyenne :

$$\hat{G}(\mathbf{x}) = \text{Arg min}_{G^*} \mathbb{E} \left[ (G^*(\mathbf{x}) - G(\mathbf{x}))^2 \right] \quad (2.17)$$

La recherche du meilleur estimateur linéaire sans biais aboutit à la résolution d'un problème d'optimisation. Cet estimateur est une variable aléatoire gaussienne de moyenne et de variance conditionnées par les données du plan d'expériences et présentées respectivement en Équations (2.18) et (2.19).

$$\mu_{\hat{G}}(\mathbf{x}) = \mathbf{f}_x^T \hat{\boldsymbol{\beta}} + \mathbf{r}_x^T \mathbf{R}^{-1} (\mathbf{G}_{obs} - \mathbf{F} \hat{\boldsymbol{\beta}}) \quad (2.18)$$

où  $\mathbf{f}_x$  est le vecteur de la tendance évaluée en  $\mathbf{x}$  et  $\mathbf{r}_x$  est le vecteur des corrélations entre les points à prédire et les points de la base d'apprentissage.

$$\sigma_{\hat{G}}^2(\mathbf{x}) = \hat{\sigma}^2 \left( 1 - \mathbf{r}_x^T \mathbf{R}^{-1} \mathbf{r}_x + \mathbf{u}_x^T (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{u}_x \right) \quad (2.19)$$

où  $\mathbf{u}_x = \mathbf{F}^T \mathbf{R}^{-1} \mathbf{r}_x - \mathbf{f}_x$ .

La moyenne de cette variable aléatoire est utilisée comme estimation du métamodèle de krigeage. La variance fournit une mesure de l'incertitude de prédiction à chaque point non-observé. Cette mesure est l'avantage majeur du métamodèle de krigeage puisqu'aucun calcul supplémentaire n'est nécessaire pour obtenir cette information, qui joue le rôle d'estimation a priori de la qualité de la prédiction.

Cette estimation est interpolante, c'est-à-dire exacte aux points d'apprentissage  $\mu_{\hat{G}}(\mathbf{x}_{obs}) = G(\mathbf{x}_{obs})$ . La variance est donc nulle aux points d'apprentissage  $\sigma_{\hat{G}}^2(\mathbf{x}_{obs}) = 0$ .

### 2.3.2 Krigeage adaptatif pour l'analyse de fiabilité

Comme présenté en section 2.2.2, des schémas adaptatifs basés sur le krigeage ont aussi été proposés dans la littérature (Bichon et al., 2008 ; Dubourg, 2011, p. ...). Ces schémas adaptatifs sont basés sur une fonction d'apprentissage permettant d'identifier le meilleur point à enrichir. Une liste non-exhaustive de différentes fonctions d'apprentissage est présentée dans la suite de cette section.

#### Fonction *EFF*

La méthode EGRA (*Efficient Global Reliability Analysis* – Bichon et al., 2008) est basée sur un apprentissage géré par la fonction *EFF* (*Expected Feasibility Function*). Cette fonction exprime l'espérance que la fonction de performance en un point  $\mathbf{u}$  non-évalué respecte la contrainte  $G(\mathbf{u}) \in [-\epsilon(\mathbf{u}); +\epsilon(\mathbf{u})]$ . Bichon et al., (2008) proposent que  $\epsilon$  soit proportionnel à la variance de krigeage ainsi la fonction d'apprentissage s'écrit :

$$\begin{aligned} EFF(\mathbf{u}) = & \mu_{\hat{G}}(\mathbf{u}) \left[ 2\Phi \left( \frac{-\mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) - \Phi \left( \frac{-\epsilon(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) - \Phi \left( \frac{\epsilon(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) \right] \\ & - \sigma_{\hat{G}}(\mathbf{u}) \left[ 2\varphi \left( \frac{-\mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) - \varphi \left( \frac{-\epsilon(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) - \varphi \left( \frac{\epsilon(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) \right] \\ & + \epsilon(\mathbf{u}) \left[ \Phi \left( \frac{\epsilon(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) - \Phi \left( \frac{-\epsilon(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) \right] \end{aligned} \quad (2.20)$$

Lorsqu'un point a une prédiction de krigeage proche de 0 ou une variance de krigeage importante, une valeur forte de *EFF* sera obtenue. L'apprentissage devra donc se faire sur un point avec une valeur importante de *EFF*.

### Fonction $U$

La fonction  $U$  (Echard et al., 2011) est la fonction d'apprentissage commune des méthodes AK qui seront présentées en section 2.4. Ce critère indique la distance en écart-type de la prédiction du krigeage en un point  $\mathbf{u}$  à l'état limite prédit. Il est défini par :

$$U(\mathbf{u}) = \frac{|\mu_{\hat{G}}(\mathbf{u})|}{\sigma_{\hat{G}}(\mathbf{u})} \quad (2.21)$$

Le nouveau point d'enrichissement doit posséder la valeur la plus faible de la fonction  $U$  permettant de trouver un point qui a une prédiction de krigeage proche de 0 et/ou une variance importante.

### Fonction $IMSE_T$

La fonction *targeter Integrated Mean-Squared Error* proposée par Picheny et al., (2010) est définie en Équation (2.22) et propose d'observer la variance du krigeage, particulièrement dans la zone d'intérêt caractérisée par un niveau moyen  $T$  et un paramètre de bande  $\epsilon$ . Ainsi, la zone d'intérêt est définie comme  $[T - \epsilon, T + \epsilon]$ . Le point minimisant cette fonction est le nouveau point à enrichir.

$$IMSE_T(\mathbf{u}) = \int_D \sigma_{\hat{G}}^2(\mathbf{u}) W(\mathbf{u}) d\mathbf{u} \quad (2.22)$$

où  $W(\mathbf{u}) = \mathcal{P}[G(\mathbf{u}) \in [T - \epsilon, T + \epsilon]]$  est une fonction de pondération. La procédure numérique pour estimer ce critère est présentée par Picheny et al., (2010).

### Fonction $SUR$

La fonction *Stepwise Uncertainty Reduction* considère un point de vue bayésien. En effet, le cadre bayésien permet ici d'obtenir une mesure de l'erreur d'estimation. A partir d'une séquence d'estimateur de probabilité de défaillance  $(\hat{\alpha}_n)_{n \leq 1}$  et de l'utilisation de la fonction de perte quadratique, l'Équation (2.23) est définie et doit être minimisée.

$$J_n(\mathbf{u}) = \mathbb{E}_n \left[ (\alpha - \hat{\alpha}_{n+1})^2 | \mathcal{U}_{n+1} = \mathbf{u} \right] \quad (2.23)$$

Aucune procédure numérique efficace n'ayant été trouvée pour approximer ce critère, quatre stratégies de type  $SUR$  ont été proposées considérant les bornes du critère précédent. Ces stratégies ainsi que les procédures numériques pour les estimer sont détaillées par Bect et al., (2012).

### Fonction $H$

Lv et al., (2015) proposent une autre fonction d'apprentissage basée sur l'information d'entropie. Cette fonction définie en Équation (2.24) fournit une mesure de l'incertitude de l'estimateur  $\hat{G}(\mathbf{u})$  au point  $\mathbf{u}$  et permet de garantir une bonne précision au voisinage de l'état limite. L'apprentissage est réalisé en choisissant le point avec la valeur de  $H$  la plus grande.

---

$$H(\mathbf{u}) = \left| \begin{array}{l} \ln \left( \sqrt{2\pi} \sigma_{\hat{G}}(\mathbf{u}) + \frac{1}{2} \right) \left[ \Phi \left( \frac{2\sigma_{\hat{G}}(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) - \Phi \left( \frac{-2\sigma_{\hat{G}}(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) \right] \\ - \left[ \frac{2\sigma_{\hat{G}}(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{2} \varphi \left( \frac{2\sigma_{\hat{G}}(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) + \frac{2\sigma_{\hat{G}}(\mathbf{u}) + \mu_{\hat{G}}(\mathbf{u})}{2} \varphi \left( \frac{-2\sigma_{\hat{G}}(\mathbf{u}) - \mu_{\hat{G}}(\mathbf{u})}{\sigma_{\hat{G}}(\mathbf{u})} \right) \right] \end{array} \right| \quad (2.24)$$

### Fonction *OBJ*

Le schéma adaptatif proposé par Wen et al., (2016) requiert l'optimisation suivante :

$$\text{Maximiser : } OBJ(\mathbf{u}) = \begin{cases} \mathbb{E}[F(\mathbf{u})], & \rho_U(\mathbf{u}) > d_{thr} \\ 0, & \rho_U(\mathbf{u}) < d_{thr} \end{cases} \quad (2.25)$$

où  $\rho_U(\mathbf{u})$  représente la densité de probabilité jointe des variables d'entrée,  $d_{thr}$  est un seuil et  $\mathbb{E}[F(\mathbf{u})]$  est la fonction *EFF* définie par l'Équation (2.20).

### Fonction *L*

Ma et al., (2018) proposent un schéma d'apprentissage se basant sur la fonction *L* définie en Équation (2.26). Ce schéma identifie plusieurs points d'enrichissement à chaque itération. Tous les points ne respectant pas  $L(\mathbf{u}) \leq 10^{-t}$ , où  $t$  est le numéro de l'itération courante, sont les nouveaux points d'enrichissement.

$$L(\mathbf{u}) = \left| \frac{\sigma_{\hat{G}}^2(\mathbf{u})}{\mu_{\hat{G}}(\mathbf{u})} \right| \quad (2.26)$$

### Discussion et illustration

Les fonctions d'apprentissage présentées précédemment sont illustrées par la Figure 2.2 sur le problème de fiabilité défini en Annexe A. Les quinze premiers points choisis par chacune des fonctions sont représentés sur les graphes. Le premier plan d'expériences, la première calibration du métamodèle de krigeage ainsi que la population de Monte-Carlo à classer sont identiques pour chaque fonction. Ainsi, seul l'enrichissement est différent d'un graphique à l'autre.

Tout d'abord, comme expliqué ci-dessus, la fonction *L* propose d'enrichir plusieurs points à chaque itération de la méthode. Cette fonction d'apprentissage peut donc provoquer un trop grand nombre d'appels inutiles à la fonction de performance. Les deux méthodes basées sur l'estimation d'une intégrale (*IMSE<sub>T</sub>* et *SUR*) requièrent des temps de calcul importants, uniquement pour évaluer le critère. Ce temps de calcul augmente d'autant plus que la population de Monte-Carlo à classer est grande. La programmation de tels schémas adaptatifs est aussi assez complexe. L'objectif principal des méthodes de simulation est de classer la population selon les domaines sûr ou défaillant, séparés par la fonction d'état limite. Ainsi, cette fonction d'état limite doit être modélisée précisément par le métamodèle pour espérer obtenir une estimation précise de la probabilité de défaillance. Les points d'enrichissement doivent donc être sélectionnés proche de la fonction d'état limite. Les fonctions *EFF*, *U* et *OBJ* semblent donc être plus efficaces pour approcher la fonction d'état-limite.

Les méthodes AK présentées dans la suite de ce document se basent sur la fonction  $U$  pour sa simplicité, son efficacité et sa rapidité. En effet, aucun calcul supplémentaire n'est requis pour estimer sa valeur, seuls la moyenne et l'écart-type de krigeage sont nécessaires.

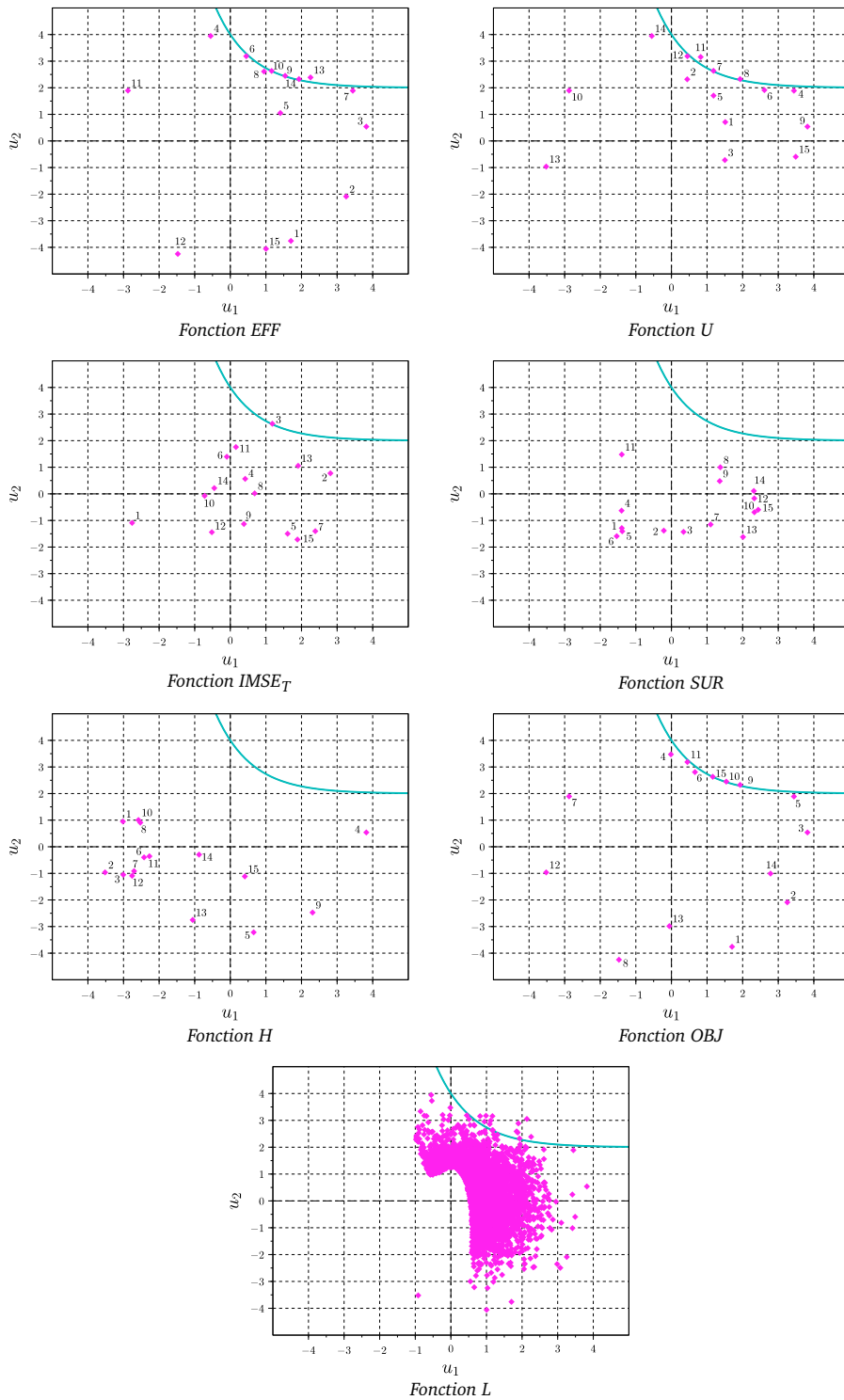


FIGURE 2.2 – Illustration des fonctions d'apprentissage pour le krigeage adaptatif. La fonction d'état limite est tracée en bleu et les points enrichis en rose.

## 2.4 Méthodes AK

Bien que beaucoup de méthodes adaptatives basées sur le krigeage ont été proposées dans la littérature (Bichon et al., 2008; Dubourg, 2011, p. ...), seules les méthodes AK vont être étudiées et présentées dans cette section. Les méthodes AK forment une famille de méthodes adaptatives basées sur le krigeage. La pionnière est la méthode AK-MCS (*Active learning reliability method combining Kriging and Monte Carlo Simulation* – Echard et al., 2011), développée pour la résolution de problèmes d’analyse de fiabilité. Elle est présentée en section 2.4.1. L’utilisation de la fonction d’apprentissage  $U$  (Echard et al., 2011) est un des points communs de toutes les méthodes de la famille AK. Le second point commun repose sur la génération d’une population à classer au début de l’analyse de fiabilité. Ceci permet de rechercher le point d’enrichissement parmi cette population ce qui ne requiert pas d’optimisation supplémentaire. Comme la première méthode AK, de nombreuses améliorations ont été proposées dans la littérature pour résoudre des problèmes d’analyse de fiabilité. Cependant, d’autres domaines ont aussi été explorés comme l’optimisation sous incertitudes et la métrologie. La famille des méthodes AK est présentée par le Figure 2.3. Une revue bibliographique des méthodes AK est présentée dans les sections suivantes (Lelièvre et al., 2017).

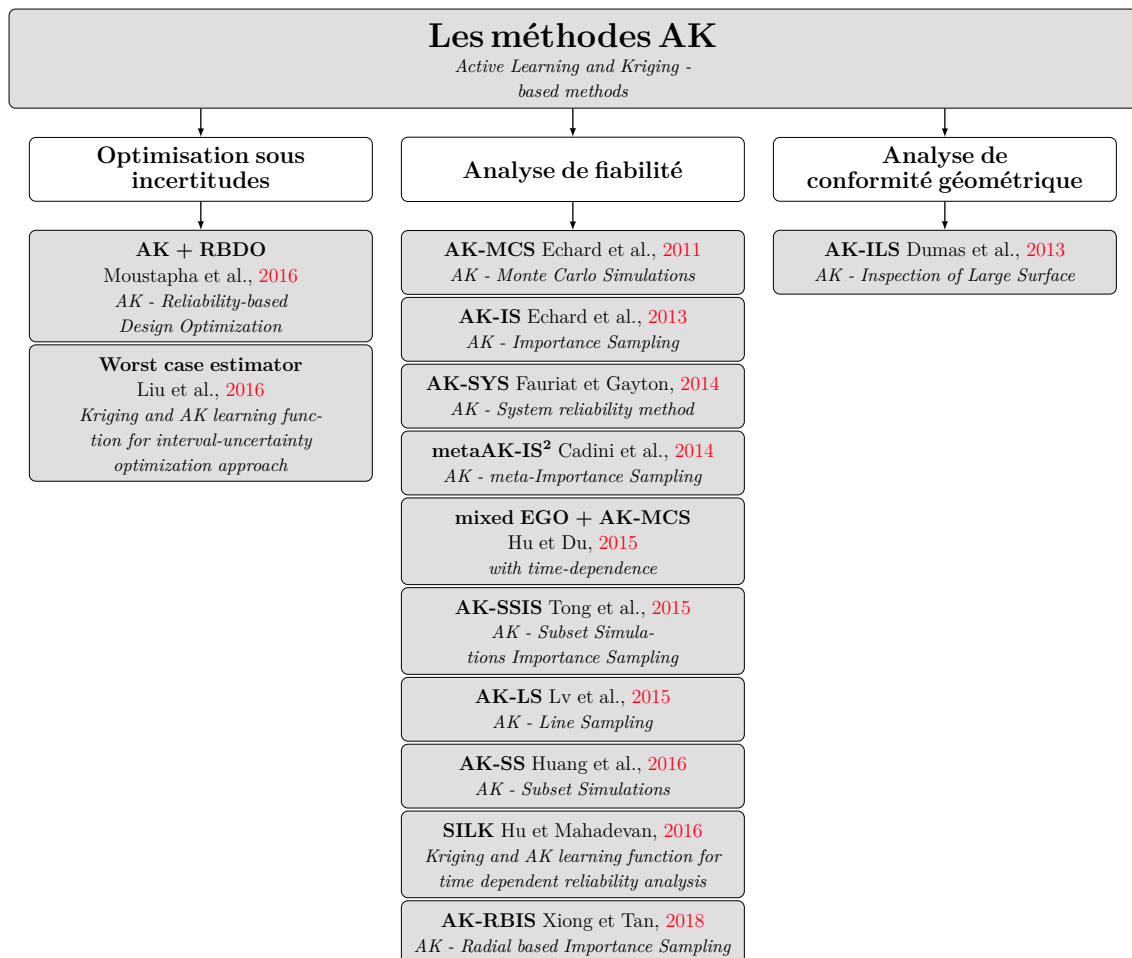


FIGURE 2.3 – Famille des méthodes AK.

### 2.4.1 Méthode AK-MCS

AK-MCS est une méthode permettant de résoudre des problèmes d'analyse de fiabilité de manière hybride. En effet, un métamodèle de krigeage est d'abord calibré (approximation) puis sa réponse est évaluée sur une population de Monte-Carlo (classification) pour estimer la probabilité de défaillance. La procédure générale est présentée ci-dessous.

Soit  $G$  une fonction de performance définie dans l'espace standard.

1. **Génération d'une population de Monte-Carlo  $P$  à classer de taille  $N$ .**
2. **Sélection du premier plan d'expériences de taille  $N_E$ .**
3. **Boucle d'enrichissement du métamodèle de krigeage**
  - (a) **Construction du métamodèle de krigeage noté  $\hat{G}$ .** Le métamodèle est construit à partir du plan d'expériences.
  - (b) **Calcul de la probabilité de défaillance** par l'Équation (2.27) :

$$P_f = \frac{\sum_{j=1}^N \mathbb{1}_{\mu_{\hat{G}}}(\mathbf{u}^{(j)})}{N} \quad (2.27)$$

où  $\mathbb{1}_{\mu_{\hat{G}}}(\mathbf{u}^{(j)})$  est la fonction indicatrice définie par :

$$\mathbb{1}_{\mu_{\hat{G}}}(\mathbf{u}^{(j)}) = \begin{cases} 1 & \text{si } \mu_{\hat{G}}(\mathbf{u}^{(j)}) \leq 0 \\ 0 & \text{sinon} \end{cases} \quad (2.28)$$

- (c) **Évaluation de la fonction d'apprentissage  $U$**  par l'Équation (2.29) :

$$U(\mathbf{u}) = \frac{|\mu_{\hat{G}}(\mathbf{u})|}{\sigma_{\hat{G}}(\mathbf{u})} \quad (2.29)$$

où  $\mu_{\hat{G}}(\mathbf{u})$  et  $\sigma_{\hat{G}}(\mathbf{u})$  sont respectivement la moyenne et l'écart type de krigeage au point  $\mathbf{u}$ .

- (d) **Évaluation du critère d'arrêt d'apprentissage** par l'Équation (2.30) :

$$\min_{j=1, \dots, N} (U(\mathbf{u}^{(j)})) \leq 2 \quad (2.30)$$

Si ce critère d'arrêt est respecté, l'enrichissement est terminé et la méthode passe à l'étape 4, sinon elle continue à l'étape 3e.

- (e) **Enrichissement du plan d'expériences.** Le point possédant la valeur minimale de la fonction d'apprentissage est ajouté au plan d'expériences. La méthode reprend ensuite au début de l'étape 3.
4. **Fin.** L'estimation de la probabilité de défaillance du problème étudié est donnée par la probabilité de défaillance évaluée lors de la dernière boucle d'apprentissage.

La méthode AK-MCS est récapitulée par l'algorithme présenté en Figure 2.4 et est illustrée en Figure 2.5 sur la fonction de Rastrigin définie par :

$$G(x_1, x_2) = 15 - \sum_{i=1}^2 [x_i^2 - 5 \cos 2\pi x_i] \quad (2.31)$$

Les deux variables aléatoires sont distribuées suivant des lois gaussiennes indépendantes de moyennes nulles et de variances unitaires. L'une des conclusions majeures à tirer concerne la zone explorée par les points d'enrichissement. Dans les premières itérations, les points d'enrichissement sont situés dans des zones proches des points du plan d'expériences précédent. Cette première conclusion met en avant l'importance du premier plan d'expériences. En effet, si ce dernier n'est pas suffisamment exploratoire, alors il y a un grand risque d'une perte d'efficacité de la méthode. La simulation par hypercube latin semble être une bonne solution pour débiter avec un plan d'expériences convenable. Dans la suite des itérations, les points sont majoritairement ajoutés au voisinage de l'état limite tout en conservant une partie exploratoire ce qui démontre l'efficacité de la fonction d'apprentissage  $U$  présentée en Équation (2.29). L'itération finale montre un état limite approximé très proche de l'état limite vrai dans les zones où des points sont présents.

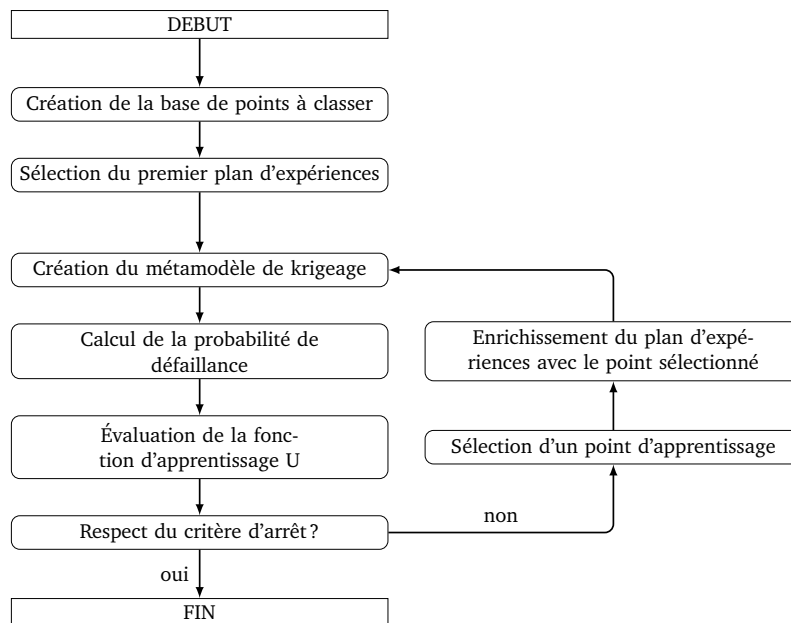


FIGURE 2.4 – Algorithme de la méthode AK-MCS.

### 2.4.2 Méthodes AK pour l'analyse de fiabilité

La méthode AK-MCS permet de réaliser des analyses de fiabilité efficacement par la classification d'une population de Monte-Carlo initialement générée. Cette méthode présente tout de même des inconvénients que de récents développements cherchent à pallier. Le premier, le plus documenté dans la littérature, est l'atteinte des faibles probabilités de défaillance. Le second est l'analyse de fiabilité système et le dernier est l'analyse de fiabilité dépendante du temps. Ces trois challenges et les méthodes proposées dans la littérature sont présentés dans la suite de cette section. Enfin, la question de la convergence et plus particulièrement celle du temps de convergence, c'est-à-dire le nombre d'itérations nécessaire pour que l'algorithme s'arrête, sont souvent posées autour des méthodes AK. Ces problèmes sont donc discutés à la fin de cette section.



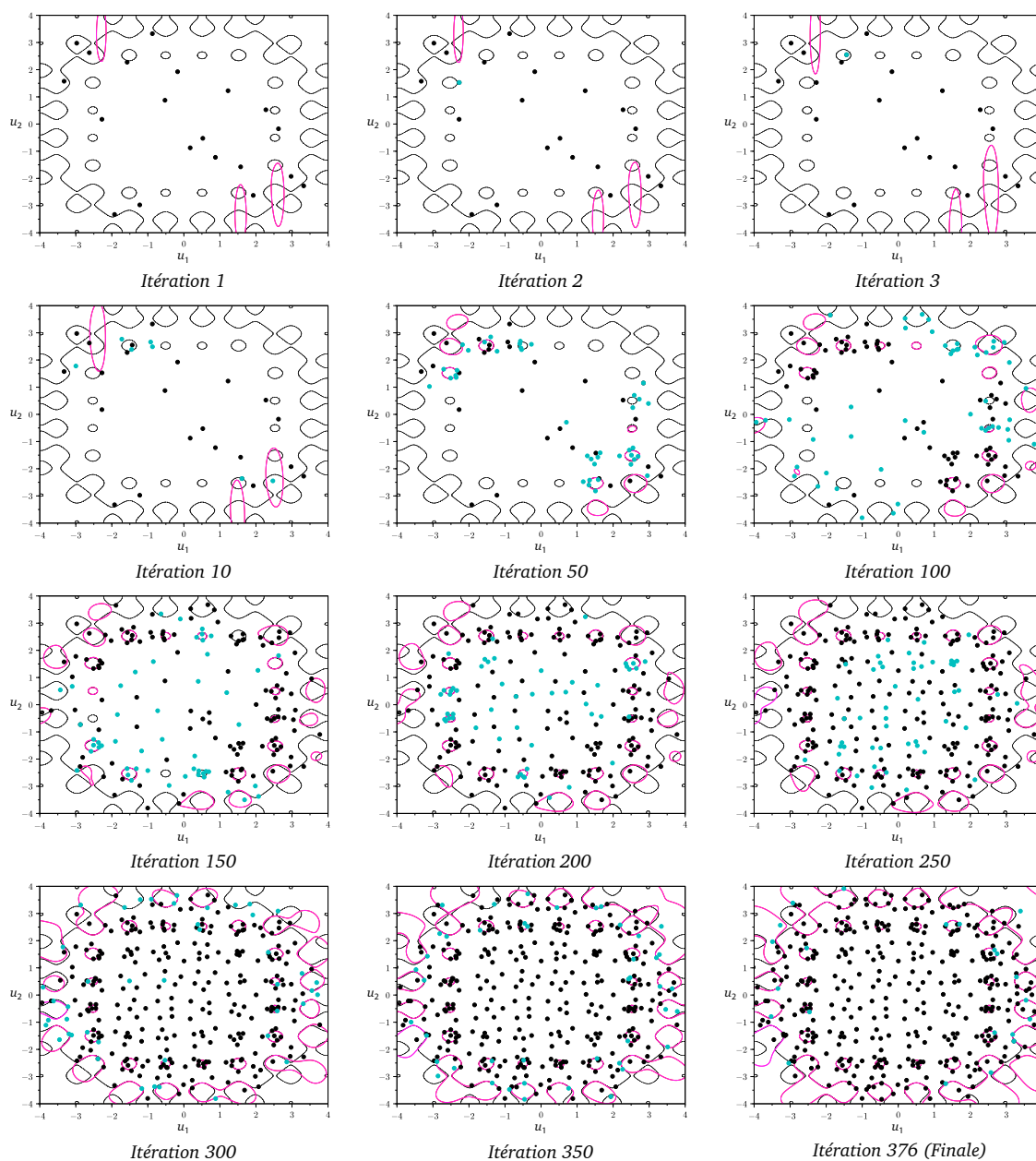


FIGURE 2.5 – Illustration de la méthode AK-MCS sur la fonction de Rastrigin. Les états limites vrai et approximé par le métamodèle sont respectivement représentés en noir et rose. Les points bleus représentent les points ajoutés entre l'itération courante et l'itération précédente.  $P_f \sim 6.3 \times 10^{-3}$ .

### Atteinte des faibles probabilités de défaillance

La méthode AK-MCS génère initialement une population de Monte-Carlo et souffre de devoir utiliser une très grande population lorsque la probabilité de défaillance est faible. En effet, bien qu'un métamodèle de krigeage soit calibré pour remplacer la vraie fonction de performance, la considération d'une large population pour atteindre une faible probabilité de défaillance peut provoquer des temps de calcul importants pour le post-traitement. L'at-

teinte de faibles probabilités de défaillance peut donc rester un véritable challenge. Toutes les méthodes présentées ci-dessous cherchent donc à remplacer la méthode de Monte-Carlo par d'autres méthodes traditionnelles de simulation dans le but de résoudre de tels problèmes. Les algorithmes de chacune des méthodes présentées ci-dessous sont disponibles en Annexe C.

La méthode AK-IS (Echard et al., 2013) propose donc d'utiliser les tirages d'importance au lieu des simulations de Monte-Carlo. La recherche du point de défaillance le plus probable doit être réalisée préalablement afin d'obtenir la fonction de pondération des tirages d'importance (voir section 1.5.2). La probabilité de défaillance sera donc estimée par la même technique que la méthode classique de tirages d'importance. Tout comme cette dernière, l'utilisation du point de défaillance le plus probable permet de cibler la population autour de l'état limite. Cependant, s'il est mal identifié ou s'il existe plusieurs points de défaillance les plus probables, la méthode AK-IS peut engendrer de fortes erreurs d'estimations de la probabilité de défaillance. Deux techniques alternatives ont donc été proposées pour régler ce problème majeur : les méthodes metaAK-IS<sup>2</sup> (Cadini et al., 2014) et AK-RBIS (Xiong et Tan, 2018). Cadini et al., (2014) reprochent à AK-IS l'unicité du point de défaillance le plus probable trouvé par l'approche FORM ainsi que la loi gaussienne pour le tirage de la population. metaAK-IS<sup>2</sup> propose d'utiliser la méthode meta-IS (Dubourg et al., 2013) permettant de trouver une loi de pondération efficace pour les tirages d'importance. Pour la même raison, Xiong et Tan, (2018) remplacent l'approche FORM par l'utilisation de la méthode *Radial-based Importance Sampling* et plus spécifiquement de sa version adaptative : *Adaptive Radial-based Importance Sampling* (Grooteman, 2008). Ces deux méthodes permettent de réduire les erreurs dues à l'utilisation des tirages d'importance combinés à FORM pour la recherche du point de défaillance le plus probable.

Des méthodes n'utilisant pas les tirages d'importance ont aussi été développées dans la littérature. Lv et al., (2015) proposent de coupler la méthode AK aux simulations par *line sampling* (Koutsourelakis et al., 2004). Cette méthode itérative sonde l'état limite pour obtenir l'estimation de la probabilité de défaillance. D'abord, présentée dans les travaux de thèse de Echard, (2012) puis repris dans le mémoire d'habilitation à diriger des recherches de Gayton, (2012), la méthode AK-SS a été proposée pour la première fois dans une revue internationale par Huang et al., (2016). Cette méthode combine le schéma adaptatif des méthodes AK aux *subset simulations* présentées en section 1.5.5. AK-SS couple donc la réduction du temps de calcul des méthodes AK à l'efficacité des *subset simulations* pour atteindre de faibles probabilités de défaillance. Enfin, Tong et al., (2015) proposent la méthode AK-SSIS pour *subset simulations importance sampling*. Cette dernière technique est une approche hybride entre les *subset simulations* et les tirages d'importance. Le tirage de chaque sous population des *subset simulations* est réalisé en utilisant la densité de probabilité des tirages d'importance obtenue par FORM (voir Song et al., 2009).

### Analyse de fiabilité système

L'analyse de fiabilité système consiste à estimer la probabilité de défaillance d'un système de plusieurs composants. Chaque composant a donc une probabilité de défaillance propre, la recherche de la probabilité de défaillance globale du système est un problème complexe

---

à résoudre. Chaque composant possède donc une fonction de performance. Les méthodes classiques d'analyse de fiabilité peuvent être utilisées mais leurs inconvénients sont toujours présents. La méthode AK-SYS a été proposée par Fauriat et Gayton, (2014) pour résoudre de tels problèmes. Dans ce papier, trois techniques différentes ont été proposées. La plus efficace, finalement intitulée AK-SYS, propose de calibrer un métamodèle par composant et d'enrichir uniquement le métamodèle de krigeage des fonctions de performance qui influent réellement sur le domaine de défaillance du système complet. Pour réaliser cet apprentissage, un critère composite est proposé dans le papier :

$$U_s(\mathbf{u}^{(i)}) = \frac{|\mu_{\hat{G}_s}(\mathbf{u}^{(i)})|}{\sigma_{\hat{G}_s}(\mathbf{u}^{(i)})} \quad (2.32)$$

où  $s$  est l'indice du métamodèle qui possède la plus petite prédiction au point  $\mathbf{u}^{(i)}$ . Si  $\mathbf{u}^{(i)}$  est le point à enrichir alors Seul le métamodèle  $\hat{G}_s$  est enrichi. Les appels aux fonctions de performance inutiles ne sont donc pas effectués ; le temps de calcul diminue fortement.

### Analyse de fiabilité dépendante du temps

L'analyse de fiabilité classique est dite instantanée, c'est-à-dire que la probabilité de défaillance est estimée à un instant précis et ne prend donc pas en compte l'influence du temps sur la fonction de performance. Or, la dépendance du temps en analyse de fiabilité peut être majeure. Deux méthodes ont été proposées dans la littérature pour résoudre de tels problèmes. La première combine la méthode AK-MCS et l'algorithme *mixed Efficient Global Optimization* (EGO). L'objectif est de calibrer un métamodèle de krigeage de la fonction des valeurs extrêmes de la réponse du système ou de la structure étudiée. Cette fonction est souvent fortement non-linéaire, ainsi les approches d'approximation FORM/SORM engendrent des fortes erreurs. L'algorithme *mixed*-EGO est une amélioration de la méthode EGO (Jones et al., 1998), qui ne cherchait qu'à identifier l'optimum global, en ajoutant la dépendance au temps. La méthode AK-MCS est ensuite utilisée pour estimer la probabilité de défaillance sur cette fonction de valeurs extrêmes ainsi approximée. Deux boucles sont donc utilisées l'une pour calibrer le métamodèle de la fonction de valeurs extrêmes et la seconde pour y ajouter la dépendance au temps. L'effet couplé des incertitudes sur les variables de conception et du temps est ainsi pris en compte de manière efficace. La seconde méthode a été proposée par Hu et Mahadevan, (2016) et est intitulée SILK. Cette méthode cherche à améliorer la méthode précédente en n'utilisant qu'une simple boucle calibrant le métamodèle directement avec la dépendance du temps. Un algorithme complexe d'apprentissage est proposé prenant en compte la fonction d'apprentissage des méthodes AK et la dépendance du temps. Ces deux méthodes apportent des solutions intéressante pour la résolution de problèmes d'analyse de fiabilité dépendante du temps.

### Discussion autour de la convergence des méthodes AK

La convergence d'un algorithme est un point crucial qui est souvent longuement discuté. La règle s'applique aussi pour les méthodes AK. L'amélioration de la convergence est discutée suivant deux axes dans la littérature. Le premier est l'amélioration de la fonction

d'apprentissage. Certains auteurs (voir par exemple Lv et al., 2015) présentent donc de nouvelles fonctions d'apprentissage qui semblent plus efficaces mais qui sont souvent bien plus complexes à implémenter et optimiser. Le deuxième axe concerne le critère d'arrêt de l'apprentissage. Différents auteurs (Gaspar et al., 2015; Tong et al., 2015) affirment que le critère d'arrêt présenté par l'algorithme AK-MCS (voir Équation (2.30) en section 2.4.1) est trop conservatif. Gaspar et al., (2015) observent une phase de stabilisation de la probabilité de défaillance en fin de convergence de la méthode AK-MCS (voir Figure 2.6). Cette phase peut coûter cher en temps de calcul. Ainsi, ils proposent un nouveau critère d'arrêt basé sur une probabilité de défaillance de référence. Ce critère permet de réduire fortement la phase de stabilisation de la méthode AK-MCS, tant qu'une probabilité de défaillance de référence convenable peut être identifiée a priori. Du même avis, Tong et al., (2015) choisissent de comparer la plus petite probabilité qu'un point soit bien classé à une probabilité seuil. Ce nouveau critère est moins conservatif et permet une convergence plus rapide de la méthode. Cependant, le conservatisme du critère d'arrêt est toujours très important dans des cas complexes. À l'opposé, un nouveau critère sera présenté dans le Chapitre 3 qui se montre encore plus conservatif que le critère classique des méthodes AK compte-tenu des faibles probabilités à estimer.

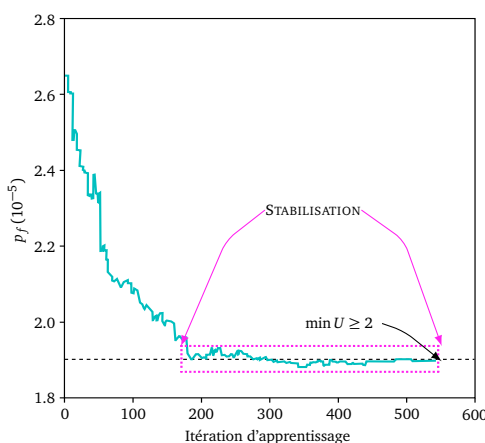


FIGURE 2.6 – Exemple de stabilisation de la méthode AK-MCS (Gaspar et al., 2015).

### 2.4.3 Méthodes AK pour l'optimisation sous incertitudes

L'optimisation et plus particulièrement l'optimisation sous incertitudes est un domaine dans lequel les métamodèles sont aussi couramment utilisés. Les méthodes AK ont donc été adaptées pour résoudre de tels problèmes. Moustapha et al., (2015) proposent un schéma d'optimisation dans lequel deux phases d'enrichissement se succèdent permettant de résoudre des problèmes de RBDO. Premièrement, un enrichissement global est réalisé pour améliorer la précision du métamodèle sur l'espace de travail complet. Ensuite, le métamodèle est enrichi plus localement suivant les régions explorées par l'algorithme d'optimisation CMA-ES qui est utilisé. Les incertitudes sont prises en compte de manière explicite dans ce premier cas, alors que Liu et al., (2016) décident de résoudre un problème d'optimisation sous contraintes de fiabilité comprenant des intervalles d'incertitudes. Un métamodèle de krigeage enrichi en utilisant la fonction d'apprentissage  $U$  est donc utilisé avec l'approche du pire des cas pour l'optimisation.

#### 2.4.4 Méthodes AK pour d'autres problèmes

Les méthodes AK sont des techniques de classification de populations en deux domaines distincts. Concernant l'analyse de fiabilité, ces deux domaines sont caractérisés par la défaillance ou la sûreté. D'autres problèmes peuvent se réduire à la classification d'une population entre deux domaines. Ainsi, des méthodes AK pourraient donc être adaptées à ces problèmes.

Une méthode a été proposée dans la littérature permettant de résoudre un problème de métrologie. En effet, la métrologie est fortement sujette aux incertitudes : les incertitudes de corrélation, de spécification et de mesure (Heping et Xiangqian, 2009). Maîtriser ces incertitudes impliquerait de connaître parfaitement la surface étudiée. En pratique, cette connaissance est complètement inaccessible et la surface est connue uniquement en quelques points. La méthode AK-ILS (*Active learning method based on Kriging for inspection of large surface* Dumas et al., 2013) permet de vérifier la conformité géométrique de grandes surfaces libres. La vérification d'une conformité est donc bien un problème de classification puisque le point d'une surface est soit conforme soit non-conforme.

Tout problème pouvant être réduit à la classification d'une population en deux classes peut donc faire l'objet d'une méthode de la famille AK.

## 2.5 Conclusions

Ce chapitre présente en préambule l'intérêt de la méta-modélisation dans le cadre de l'analyse de fiabilité. Différents métamodèles sont présentés pour traiter des problèmes de classification ou de régression. La classification ne s'intéresse qu'au signe de la fonction de performance permettant de classer les points d'une population dans les domaines de défaillance ou de sûreté, respectivement lorsque le signe de cette fonction est négatif ou positif. Bien que la méta-modélisation semble très efficace, elle fait face à trois grands challenges à résoudre : les faibles probabilités de défaillance, le *Big Data* et la grande dimension. Ce document propose de nouvelles méthodes permettant de résoudre des problèmes de faibles probabilités de défaillance (voir Chapitre 3) et des problèmes en grande dimension (voir Chapitre 4).

Le krigeage est un métamodèle qui a reçu beaucoup d'attention depuis plusieurs décennies puisqu'il apporte deux informations : une prédiction de la fonction de performance et une variance de prédiction permettant d'identifier les zones de l'espace peu explorées par le plan d'expériences. Le caractère « statique » du plan d'expériences des méthodes basées exclusivement sur un métamodèle de krigeage rend ces méthodes peu efficaces concernant le choix du plan d'expériences. De nombreux schémas adaptatifs ont été proposés. Parmi eux, les méthodes AK se montrent particulièrement efficaces en raison de leur simplicité et de leur approche basée sur la classification qui simplifie la recherche du meilleur point suivant. Ces méthodes ont été adaptées pour résoudre de nombreux problèmes d'analyse de fiabilité comme la fiabilité système ou la fiabilité dépendante du temps.

---

### Synthèse du chapitre

**Objectif.** Présenter l'utilisation et les challenges des métamodèles et plus particulièrement des méthodes AK pour l'analyse de fiabilité.

**Propositions.** Une revue bibliographique des méthodes AK a été proposée en identifiant les défis relevés par chacune des méthodes de la littérature.

**Publications scientifiques.**

- N. Lelièvre et al. (2017). « La famille des méthodes AK pour l'analyse de fiabilité : synthèse des récentes avancées ». In : *23<sup>ème</sup> Congrès Français de Mécanique, Lille*



# AK-MC*S*i – Amélioration de la méthode AK-MC*S* pour l’atteinte de faibles probabilités et l’enrichissement multiple

## Sommaire

---

<b>3.1</b>	<b>Introduction</b>	<b>56</b>
<b>3.2</b>	<b>Revue bibliographique des méthodes d’estimation de faibles probabilités de défaillance</b>	<b>56</b>
3.2.1	Méthodes de simulation de faibles probabilités	56
3.2.2	Méthodes basées sur la métamodélisation pour l’estimation de faibles probabilités	58
<b>3.3</b>	<b>Méthode AK-MC<i>S</i>i</b>	<b>60</b>
3.3.1	Simulations séquentielles de Monte-Carlo	61
3.3.2	Enrichissement multi-points	62
3.3.3	Nouveau critère d’arrêt sur l’enrichissement	66
<b>3.4</b>	<b>Validation de la méthode AK-MC<i>S</i>i</b>	<b>69</b>
3.4.1	Fonction de Rastrigin	69
3.4.2	Fonction de performance avec deux domaines de défaillance distincts	74
3.4.3	Fiabilité d’une structure en treillis	76
3.4.4	Fiabilité d’un problème de fluage et de fatigue	77
3.4.5	Réponse dynamique d’un oscillateur non-linéaire	80
<b>3.5</b>	<b>Conclusions</b>	<b>83</b>

---



### 3.1 Introduction

Bien que très efficace dans différents cas, les méthodes AK présentent des inconvénients, dont deux sont davantage détaillés dans ce chapitre. Le premier concerne le tirage préalable d'une population à classer. En effet, cette population doit être stockée et des opérations doivent être réalisées sur celle-ci. Une grande population peut donc provoquer des pertes de temps de calcul mais aussi des difficultés vis-à-vis de la mémoire vive utilisée pour le stockage. Lorsque la probabilité de défaillance devient faible,  $10^{-4} - 10^{-9}$ , la taille de la population à classer devient donc très grande ( $10^6 - 10^{11}$ ) et les méthodes AK perdent de leur efficacité. Les méthodes traditionnelles d'analyse de fiabilité ont aussi des difficultés pour estimer de telles probabilités de défaillance. Ce challenge a été identifié depuis plusieurs années et plusieurs méthodes peuvent être trouvées dans la littérature pour traiter de tels problèmes d'analyse de fiabilité. La section 3.2 présente les difficultés rencontrées par les différentes méthodes classiques d'analyse de fiabilité. Les adaptations proposées dans la littérature sont décrites. Une attention particulière est aussi portée sur l'utilisation de métamodèles.

Le second inconvénient des méthodes AK présenté ici correspond à l'enrichissement multiple. En effet, les méthodes AK ne proposent qu'un unique point d'apprentissage à chaque itération et ne permettent donc de recourir à la parallélisation des calculs.

Ce chapitre présente une nouvelle méthode AK, intitulée AK-MC*Si* (Lelièvre et al., 2018a), dans laquelle trois améliorations de AK-MCS sont proposées afin de limiter ces deux inconvénients. Cette méthode est présentée en section 3.3 et est illustrée en section 3.4. Ces applications montrent l'efficacité de la méthode proposée comparée aux méthodes AK provenant de la littérature.

### 3.2 Revue bibliographique des méthodes d'estimation de faibles probabilités de défaillance

Cette section se concentre sur les difficultés des différentes méthodes traditionnelles d'analyse de fiabilité pour l'estimation de faibles probabilités de défaillance. Les méthodes d'approximation ne souffrent pas particulièrement des probabilités faibles. Le manque de précision reste un inconvénient fort de ces méthodes. Une revue bibliographie est réalisée concernant les méthodes de simulation (section 3.2.1) et les méthodes basées sur les métamodèles (section 3.2.2).

#### 3.2.1 Méthodes de simulation de faibles probabilités

L'estimation de faibles probabilités de défaillance est un réel challenge pour les méthodes de simulation, et particulièrement lorsque la fonction de performance est chère à évaluer. En effet, la méthode classique de simulations de Monte-Carlo requiert un très grand nombre d'appels à la fonction de performance. Considérant des modèles coûteux, l'évaluation de la fonction de performance sur une telle population peut devenir infaisable. Les méthodes *Importance Sampling* (IS) et *Subset simulations* (SS) sont deux méthodes couramment utilisées pour traiter des problèmes d'estimation de faibles probabilités de défaillance. Cependant, elles présentent aussi certaines difficultés qui ont été discutées dans

---

la littérature.

D'une part, la méthode de simulations IS demande la définition d'une densité d'échantillonnage (voir section 1.5.2). La technique la plus courante est de tirer une population suivant une loi gaussienne centrée sur le MPFP et de variance unitaire. Cependant, d'autres densités d'importance ont été proposées dans la littérature permettant d'améliorer les résultats de l'analyse de fiabilité (Bucklew, 2004 ; Dupuis et al., 2007 ; L'Ecuyer et Tuffin, 2008 ; L'Ecuyer et al., 2010 ; Asmussen et al., 2011). L'inconvénient majeur de telles densités d'importance est qu'elles sont fortement dépendantes du problème considéré (Botev et al., 2013). Des schémas adaptatifs basés sur les tirages d'importances ont aussi été proposés (Rubinstein et Kroese, 2004 ; Asmussen et Glynn, 2007). Ils se basent sur le choix a priori d'une famille paramétrique (Botev et al., 2013). Les principaux travaux sont donc effectués sur la recherche d'une densité d'échantillonnage efficace. Botev et al., (2013) proposent alors une méthode avancée basée sur les tirages d'importance couplés aux chaînes de Markov. Des méthodes non-paramétriques ont aussi été proposées dans la littérature afin d'obtenir une approximation de la densité d'échantillonnage optimale (voir par exemple Zhang, 1996 ; Neddermeyer, 2009 ; Morio, 2011). La technique d'entropie croisée (*Cross-Entropy*) permet aussi de déterminer itérativement une densité d'importance intéressante pour traiter des problèmes d'estimation de faibles probabilités de défaillance (Rubinstein et Melamed, 1997 ; Rubinstein et Kroese, 2004 ; Rubinstein et Kroese, 2007). Gouda et Szántai, (2008) montre l'efficacité de l'utilisation de l'entropie croisée pour estimer de faibles probabilités de défaillance. Plusieurs méthodes ont été proposées dans la littérature couplant la technique d'entropie croisée avec d'autres techniques de simulation comme par exemple la méthode SPLITCE (Garvels, 2011).

D'autre part, la méthode de simulations par *subset* (SS) est efficace pour estimer de faibles probabilités de défaillance. Cependant, cette méthode requiert toujours trop d'appels à la fonction de performance lorsque celle-ci est coûteuse. Différentes améliorations ont été proposées dans la littérature pour diminuer ce nombre d'appels à la fonction de performance et ainsi rendre encore plus efficace la méthode de simulations par *subset*. Chen et Li, (2017) proposent de combiner les *subset simulations* et les simulations de Monte-Carlo multicanonique (MMC). L'idée principale des simulations de Monte-Carlo multicanonique (Berg et Neuhaus, 1991 ; Berg et Neuhaus, 1992) est d'estimer la faible probabilité de défaillance en construisant la distribution de la réponse de la fonction de performance. Comme la méthode SS, SMMC définit une séquence de différents seuils puis MMC est utilisée pour le tirage des sous-populations permettant un meilleur tirage dans le domaine de défaillance. La deuxième méthode présentée dans cette section est celle proposée par Rashki, (2018) qui se compose de deux étapes. La première consiste à reformuler l'intégrale de calcul de la probabilité de défaillance en la divisant en deux parties par l'introduction de variables de contrôle sur la densité de probabilité des variables d'entrée. La seconde étape consiste à estimer l'intégrale en utilisant les simulations par *subset*. Cette méthode est intitulée PDF-CSS et se montre très efficace pour estimer de faibles probabilités.

D'autres méthodes de simulation ont été améliorées dans la littérature pour parvenir à estimer précisément une faible probabilité de défaillance. Munoz Zuniga et al., (2011) proposent de coupler les tirages directionnels (Bjerager, 1988 ; Madsen et Ditlevsen, 1996) et les *stratified Sampling* (Cochran, 1977 ; Rubinstein et Kroese, 2007 ; Cannamela et al., 2008 ; Étoré et Jourdain, 2010). La méthode proposée est, selon les auteurs, efficace lorsque

la dimension n'excède pas quatre variables aléatoires d'entrée. Le nombre d'appels à la fonction de performance augmente fortement avec le nombre de variables. Elle se montre tout de même plus efficace que la méthode classique de tirages directionnels. Alban et al., (2017) présentent une comparaison de diverses combinaisons des méthodes suivantes : *Stratified sampling*, *Latin Hypercube Sampling* et simulations de Monte-Carlo conditionnelles.

Beaucoup de méthodes basées sur la simulation sont proposées dans la littérature pour résoudre des problèmes d'analyse de fiabilité avec de faibles probabilités de défaillance. Ce type de problème est toujours un challenge aujourd'hui et de nombreuses recherches continuent à être réalisées sur le sujet. En effet, toutes ces méthodes basées sur la simulation requièrent un grand nombre d'appels à la fonction de performance. Bien que les dernières méthodes proposées cherchent à réduire fortement ce nombre d'appels, il reste toujours trop grand lorsque des modèles coûteux sont utilisés.

### 3.2.2 Méthodes basées sur la métamodélisation pour l'estimation de faibles probabilités

Comme les méthodes d'approximation et de simulation, celles basées sur l'utilisation de métamodèles peuvent souffrir du nombre trop important d'appels à la fonction de performance, logiquement coûteuse, lorsque de faibles probabilités de défaillance doivent être estimées. Une courte revue bibliographique des méthodes proposées dans la littérature est donc présentée ici pour chacun des métamodèles détaillés au Chapitre 2.

Tout d'abord, des variantes de la technique de surfaces de réponse de Faravelli, (1989) ont été proposées dans la littérature et permettent d'atteindre de faibles probabilités de défaillance. Les adaptations de Kim et Na, (1997), Kaymaz et McMahon, (2005) et Nguyen et al., (2009) sont des variantes dans le choix de l'approximation polynomiale et de la technique de sélection du plan d'expériences. Selon les exemples présentés dans ces papiers, elles permettent d'atteindre des probabilités de défaillance de l'ordre de  $10^{-4} - 10^{-5}$ . Roussouly et al., (2013) sélectionnent les termes de la base polynomiale les plus importants vis-à-vis de la fonction de performance afin de diminuer le nombre des termes. Dans le but de rendre cette base polynomiale plus flexible, Bucher et Most, (2008) et Kang et al., (2010) montrent l'efficacité d'utiliser une approximation polynomiale par *Moving Least-Square* permettant aussi d'atteindre de faibles, voire très faibles, probabilités de défaillance.

Les événements rares (liés à de faibles probabilités de défaillance) n'apparaissent que dans les queues de la distribution jointe des variables d'entrée ce qui représente une difficulté majeure. Paffrath et Wever, (2007) proposent d'utiliser un chaos polynomial basé sur les polynômes d'Hermite localement autour du point de défaillance le plus probable déterminé par l'approche FORM. Des méthodes couplant le chaos polynomial et le Krigage (par exemple Schöbi et al., 2015, 2017) montrent aussi une bonne efficacité pour estimer de faibles probabilités de défaillance. Davantage de détails peuvent être trouvés dans le manuscrit d'Habilitation à Diriger des Recherches de Sudret, (2007) concernant le chaos polynomial.

Les réseaux de neurones artificiels utilisés dans le cadre de l'analyse de fiabilité ont aussi fait l'objet d'adaptations pour traiter des problèmes avec une probabilité de défaillance faible. Par exemple, Papadopoulos et al., (2012) choisissent de les coupler avec les *subset*

*simulations* pour augmenter l'efficacité de cette dernière méthode.

Les *Support Vector Machines* (SVM) sont aussi beaucoup utilisées dans le cadre de l'analyse de fiabilité (voir le manuscrit d'Habilitation à Diriger des Recherches de Bourinet, 2018) et font donc l'objet de discussions autour de l'atteinte des faibles probabilités de défaillance. Comme expliqué en section 2.2, deux approches basées sur les *Support Vector Machines* sont utilisées en analyse de fiabilité : la classification et la régression. D'une part concernant la classification, Bourinet et al., (2011) proposent de coupler les SVM aux *subset simulations* pour estimer de faibles probabilités de défaillance. D'autre part, plusieurs méthodes ont été proposées concernant l'utilisation des SVM en régression pour l'atteinte de probabilités de défaillance faibles, voire très faibles. Tout d'abord, Dai et al., (2012) proposent d'approximer la fonction d'état-limite de manière locale et réussissent à estimer des probabilités de défaillance de l'ordre de  $10^{-5}$ . Différents travaux de Bourinet discutent de l'utilisation des SVM pour résoudre de tels problèmes. Il propose la méthode ASVR (Bourinet, 2015, 2016) puis une amélioration de celle-ci (Bourinet, 2017). Elle est très efficace pour atteindre de faibles probabilités de défaillance, voire très faibles ( $10^{-7}$ ).

Enfin, le Krigeage est un des modèles les plus utilisés dans le cadre de l'analyse de fiabilité. De nombreuses méthodes ont déjà été proposées dans la littérature pour parvenir à estimer de faibles probabilités de défaillance. Ce document se focalise sur les méthodes AK qui regroupent plusieurs techniques pour résoudre de tels problèmes et qui sont décrites en section 2.4 (voir par exemple Echard et al., 2013; Cadini et al., 2014; Huang et al., 2016, etc). D'autres méthodes ont aussi été proposées sans être basées sur le schéma adaptatif AK. Tout d'abord, plusieurs auteurs choisissent d'utiliser le krigeage pour réduire le temps de calcul lié aux tirages d'importance (Schueremans et Van Gemert, 2005). Dubourg et al., (2011, 2013) proposent la méthode metaIS dans laquelle le krigeage est utilisé pour construire une densité quasi-optimale des tirages d'importance. Balesdent et al., (2013) couplent les tirages d'importance adaptatifs à un métamodèle de krigeage pour améliorer l'efficacité de la méthode de simulation, puis proposent une adaptation de cette première méthode pouvant considérer des incertitudes épistémiques dans les distributions des variables d'entrée (Balesdent et al., 2016). Enfin, Tyagi et al., (2017) couplent la technique de *mean shift Importance Sampling* à un métamodèle de krigeage lui permettant d'estimer précisément une probabilité de l'ordre de  $10^{-11}$ . Les *subset simulations* ont, elles aussi, été couplées à un métamodèle de krigeage pour augmenter leur efficacité et donc diminuer le temps de calcul nécessaire (Li et al., 2012; Echard, 2012; Huang et al., 2016). Comme exprimé plus tôt, le krigeage a aussi été couplé à d'autres métamodèles comme le chaos polynomial (Schöbi et al., 2015, 2017).

Bien que différentes méthodes aient été proposées, que ce soit en approximation, en simulation ou utilisant des métamodèles, l'atteinte des faibles probabilités de défaillance reste un challenge majeur des analyses de fiabilité à cause des efforts numériques nécessaires qui peuvent devenir colossaux si une fonction de performance coûteuse est considérée. La section suivante présente une nouvelle méthode basée sur AK-MCS permettant d'estimer précisément une probabilité de défaillance faible avec un nombre raisonnable d'appels à la fonction de performance.

### 3.3 Méthode AK-MCSI

L'utilisation de la méthode AK-MCS pour estimer de faibles probabilités de défaillance implique la résolution de différents challenges. Le premier concerne l'espace mémoire requis pour conserver la population de Monte-Carlo tout au long de la méthode. Le second challenge concerne le temps de calcul qui reste la principale motivation pour l'utilisation d'un métamodèle en lieu et place de la vraie fonction de performance. Deux améliorations sont proposées pour résoudre ces problèmes : les simulations séquentielles de Monte-Carlo et l'enrichissement multi-points à chaque itération. La première amélioration requiert de modifier le critère d'arrêt sur l'apprentissage pour qu'il soit plus strict et assure le bon classement des points. La procédure globale de AK-MCSI (*improved Active learning and Kriging based Mont Carlo Simulations*) est illustrée en Figure 3.1. Ces trois techniques ont été publiées dans une revue internationale (Lelièvre et al., 2018a).

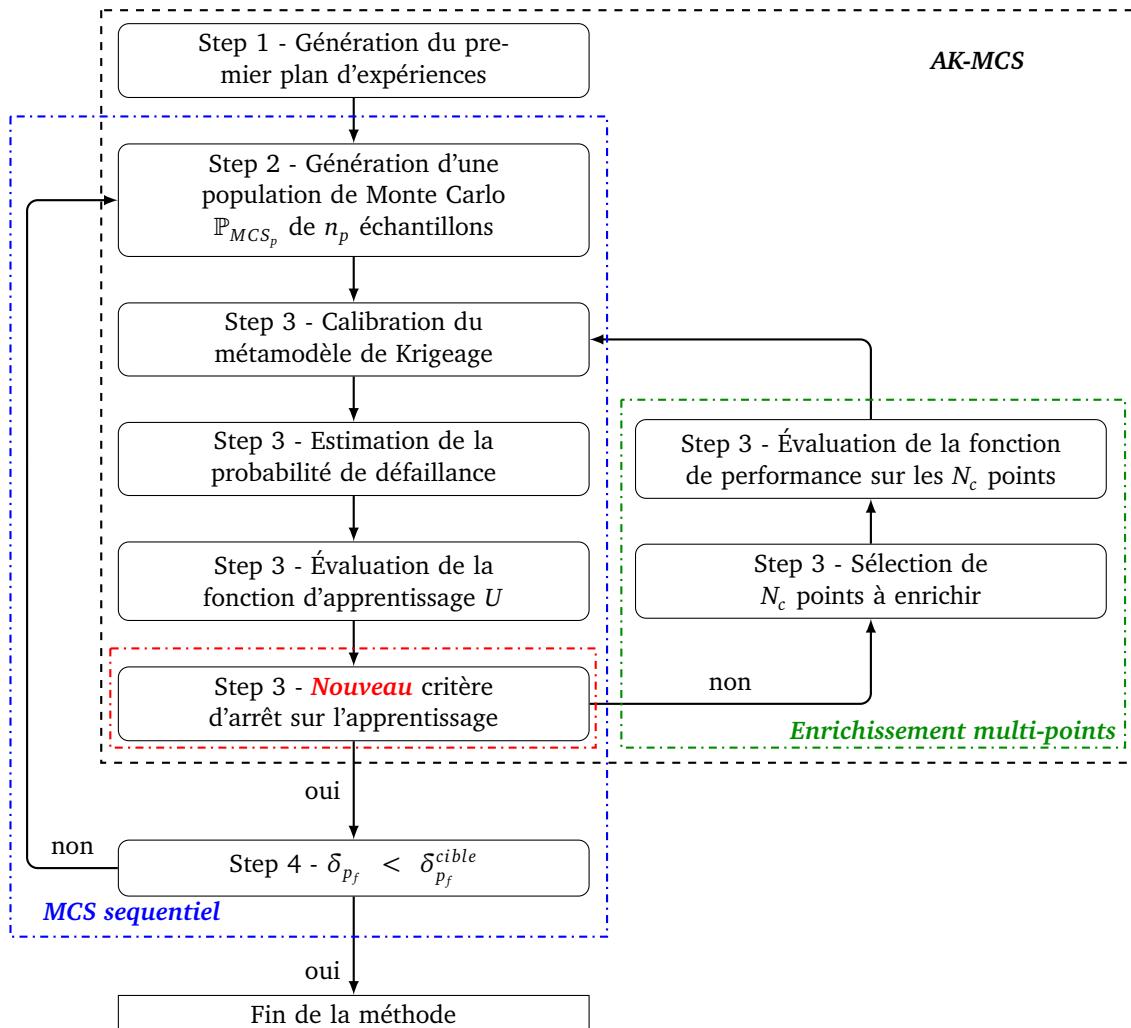


FIGURE 3.1 – Procédure de la méthode AK-MCSI.

### 3.3.1 Simulations séquentielles de Monte-Carlo

La taille de la population de Monte-Carlo augmente fortement lorsque la probabilité de défaillance à estimer diminue. En effet, pour une probabilité de défaillance de l'ordre de  $P_f \approx 10^{-\lambda}$ , une population de Monte-Carlo contenant  $10^{\lambda+2}$  est approximativement requise pour obtenir une précision suffisante. AK-MCS conserve en mémoire (RAM) la même population de Monte-Carlo jusqu'à sa convergence. Lorsqu'une faible probabilité de défaillance doit être estimée, une large population de Monte-Carlo doit être utilisée et gardée en mémoire. Durant la méthode, le métamodèle doit être évalué sur toute la population à chaque itération. Même si l'évaluation de ce métamodèle est beaucoup moins coûteuse que celle de la fonction de performance, l'évaluation d'une telle population peut prendre un temps non négligeable. Deux challenges se posent donc pour l'estimation de faibles probabilités de défaillance : la taille mémoire utilisée qui peut mener au crash des calculs et le temps de calcul pour l'évaluation du métamodèle. Les simulations séquentielles de Monte-Carlo sont donc proposées pour résoudre ces deux problèmes. L'idée principale est de découper la grande population de Monte-Carlo en sous-populations de tailles réduites considérées une par une. La taille de la population de Monte-Carlo n'est donc pas supposée au départ. La procédure est décrite ci-dessous.

Soit  $P_{f\text{ vraie}} = 10^{-\lambda}$  la probabilité de défaillance à estimer, inconnue au début de l'analyse.

1. Génération d'un plan d'expériences initial
2. Génération d'une population  $\mathbb{P}$  contenant  $n_p$  points,  $n_p \ll 10^{\lambda+2}$ .
3. Utilisation de AK-MCS sur la population. Lorsqu'une nouvelle population est utilisée, le dernier métamodèle calibré est directement réutilisé. Lorsqu'une population satisfait le critère d'arrêt sur l'enrichissement, le nombre de points défaillants identifiés est conservé en mémoire. La population n'est quant à elle pas conservée en mémoire.
4. Évaluation du coefficient de variation de la probabilité de défaillance par l'Équation (3.1).

$$\delta_{P_f} = \sqrt{\frac{1 - P_f}{N_p \times n_p \times P_f}} \quad (3.1)$$

où  $N_p$  est le nombre de populations considérées depuis le début de la méthode.

5. Respect du critère d'arrêt sur le coefficient de variation suivant ?

$$\delta_{P_f} \leq \delta_{P_f}^{(cible)} \quad (3.2)$$

Le seuil  $\delta_{P_f}^{(cible)}$  est défini au préalable par l'utilisateur de la méthode. Si ce critère n'est pas respecté, une nouvelle population de taille  $n_p$  est générée et la méthode retourne à l'étape 2. Sinon, la méthode prend fin. La probabilité de défaillance finale est calculée par l'Équation (3.3).

$$P_f = \frac{\sum_{j=1}^{N_p} \left[ \sum_{i=1}^{n_p} \mathbb{1}_{\hat{G}_j}(\mathbf{u}^{(i)}) \right]}{N_p \times n_p} \quad (3.3)$$

où  $\hat{G}_j$  est le métamodèle de krigeage calibré lorsque la  $j^{\text{ème}}$  population était considérée et  $\mathbf{u}^{(i)}$  est le  $i^{\text{ème}}$  point de la  $j^{\text{ème}}$  population.



La taille  $n_p$  des populations a une grande influence sur la rapidité de convergence de la méthode. En effet, de trop petites populations impliquent un grand nombre de générations de populations et provoquent donc une hausse du temps de calcul. À l’opposé, considérer de trop grandes populations peut provoquer des problèmes mémoires et ainsi une augmentation du temps d’évaluation du métamodèle à chaque itération. La taille optimale  $n_p$  des populations dépend de la probabilité de défaillance à estimer. En effet, un compromis doit être formulé entre le gain de temps d’évaluation du métamodèle obtenu à partir d’une petite population et le nombre d’itérations nécessaire pour l’obtention d’un coefficient de variation de la probabilité de défaillance assez faible. Plus la probabilité attendue est faible, plus la taille des populations doit être grande. Cependant, la probabilité de défaillance n’étant pas connue à priori, la définition de la taille optimale des populations n’est pas évidente. L’utilisation de  $n_p = 10^4$  est souvent un bon compromis.

L’utilisation des simulations séquentielles de Monte-Carlo est un outil pour l’estimation de faibles probabilités de défaillance qui permet donc de réduire fortement la taille mémoire (RAM) et qui donc diminue le temps de calcul lors de l’évaluation de la population sur le métamodèle de krigeage. Cette technique est illustrée en section 3.4.

### 3.3.2 Enrichissement multi-points

La seconde amélioration proposée par la méthode AK-MCSi concerne l’enrichissement multi-points. Des installations informatiques sont de plus en plus souvent disponibles et utilisées dans le but de paralléliser et ainsi accélérer les calculs nécessaires pour réaliser des analyses de fiabilité. Des enrichissements multi-points ont déjà été proposés dans la littérature (voir par exemple Dubourg et al., 2011). Cependant, la fonction d’apprentissage  $U$  proposée dans AK-MCS ne permet la sélection que d’un unique point d’enrichissement à chaque itération. L’utilisation d’installations informatiques distribuées implique de pouvoir enrichir le métamodèle en plusieurs points à chaque itération. Dans le but de sélectionner ces points, deux propriétés majeures doivent être considérées. Premièrement, les points doivent être sélectionnés selon la fonction d’apprentissage, c’est-à-dire qu’ils doivent correspondre à de faibles valeurs de  $U$ . Deuxièmement, les points d’enrichissement sélectionnés ne doivent pas être trop proches les uns des autres pour augmenter l’efficacité de la méthode. En effet, choisir deux points trop proches impliquerait une redondance et ainsi demanderait un appel inutile à la fonction de performance.

La sélection de tels points doit donc se baser sur un algorithme de *clustering*, c’est-à-dire d’un partitionnement des données en différents *clusters* ou classes. Les algorithmes de *clustering* sont divisés en deux principaux types de partitionnement : les **partitionnements hiérarchiques** et **non-hiérarchiques**.

Le premier type d’algorithmes cherche à construire un arbre **hiérarchique** des données à partitionner. Les deux principales approches hiérarchiques sont les approches agglomérative et divisive. L’approche agglomérative considère initialement que chaque donnée représente un *cluster*. Ces *clusters* sont ensuite fusionnés deux à deux selon leur similarité. Les algorithmes UPGMA (Sokal et Michener, 1958), WPGMA (McQuitty, 1966) ou BIRCH (Zhang et al., 1997) en sont des exemples. L’approche divisive procède dans le sens inverse de l’approche agglomérative, c’est-à-dire qu’elle considère initialement un unique

*cluster* et cherche donc ensuite à le diviser jusqu'atteindre le nombre de *clusters* souhaité. Cette approche est très peu utilisée parce que le choix du critère de division est complexe. L'algorithme DIVCLUS-T (Chavent et al., 2007) en est un exemple.

Le second type d'algorithmes de *clustering*, les algorithmes **non-hiérarchiques**, se divisent en différents sous-types. Le premier correspond aux méthodes de partition basées sur des centroïdes. L'idée principale est de construire un partitionnement des données en un nombre fixé de classes. La complexité de ces méthodes est linéaire et permet donc de traiter de grandes bases de données. L'avantage majeur de ces méthodes est leur simplicité. L'algorithme K-means est le plus connu de ce type de *clustering*, mais d'autres algorithmes sont disponibles dans la littérature tels que l'algorithme K-medoid, CLARA, CLARANS,... Ces algorithmes sont présentés par Uppada, (2014). Le second type d'algorithmes non-hiérarchiques se base sur la densité de points, c'est-à-dire que les *clusters* sont définis par des régions denses séparées par des régions moins denses. Ces densités peuvent être définies par le voisinage d'objets, les objets noyaux, l'accessibilité ou la connexion entre objets. Les algorithmes les plus connus sont DBSCAN (Ester et al., 1996), OPTICS (Ankerst et al., 1999) ou encore DENCLUE (Hinneburg et Keim, 1998; Hinneburg et Gabriel, 2007). Les méthodes par grilles sont aussi un des types d'algorithmes non-hiérarchiques. Elles sont définies par la division de l'ensemble des données selon un nombre fini de cellules. Ces cellules peuvent être des cubes, des régions ou des hyper-rectangles et définissent la grille. Ces méthodes sont souvent utilisées pour partitionner des données spatiales. Les algorithmes STING (Wang et al., 1997), Wavecluster (Sheikholeslami et al., 1998) et CLIQUE (Agrawal et al., 1998) en sont des exemples. D'autres types d'algorithmes non-hiérarchiques existent dans la littérature comme les méthodes conceptuelles, les méthodes basées sur la théorie des graphes ou sur les réseaux de neurones. Plus de détails sur toutes ces méthodes sont disponibles dans les travaux de thèse de Fouchal, (2011).

Par sa simplicité et son efficacité, l'algorithme K-means est plus particulièrement étudié. En effet, l'algorithme de *clustering* proposé dans AK-MCSi se base sur le *K-means clustering* dans lequel la moyenne arithmétique utilisée est remplacée par une moyenne pondérée par la fonction U. Les algorithmes *K-means clustering* et *K-weighted-means clustering* sont présentés en détail ci-dessous. La méthode a pour but de regrouper un ensemble de points  $S$  en  $K$  sous-ensembles nommés *clusters*  $\{s_1, \dots, s_k, \dots, s_K\}$ . Un nombre de points égal à  $5 \times K$  pour l'ensemble  $S$  est choisi ici et semble être un bon compromis. Ces points sont les  $5 \times K$  points avec les valeurs de U les plus faibles.

### ***K-means clustering***

L'algorithme *K-means clustering* cherche à minimiser la distance entre les points à l'intérieur de chaque *cluster*.

$$\text{Arg min}_s \sum_{j=1}^K \sum_{\mathbf{u} \in s_j} \|\mathbf{u} - \mathbf{c}_j\| \quad (3.4)$$

où  $\mathbf{c}_j$  est le centroïde du *cluster*  $s_j$ .

Les étapes de la procédure de l'algorithme *K-means clustering* sont :

1. Sélection aléatoire d'un ensemble initial de  $K$  centroïdes notés  $\{\mathbf{c}_1^{(1)}, \dots, \mathbf{c}_k^{(1)}, \dots, \mathbf{c}_K^{(1)}\}$ .



2. Regroupement des points  $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$  selon le diagramme de Voronoï (Aurenhammer, 1991) qui classe les points selon le centroïde le plus proche.
3. Évaluation de la moyenne arithmétique de chaque *cluster* pour l'itération suivant  $i + 1$  :

$$\mathbf{c}_k^{(i+1)} = \sum_{j=1}^{n_k} \mathbf{u}_j \quad (3.5)$$

où  $n_k$  est le nombre de points dans le *cluster*  $s_k$  et  $\{\mathbf{u}_1, \dots, \mathbf{u}_j, \dots, \mathbf{u}_{n_k}\}$  sont les points appartenant au *cluster*  $s_k$ .

4. Évaluation du critère d'arrêt sur la différence entre les centroïdes de deux itérations successives définie par l'Équation (3.6). Si le critère d'arrêt, l'algorithme est fini ; sinon l'algorithme retourne à l'étape 2.

$$\sum_{k=1}^K \left| \mathbf{c}_k^{(i)} - \mathbf{c}_k^{(i+1)} \right| < \epsilon \quad (3.6)$$

### *K-weighted-means clustering*

L'utilisation de l'algorithme classique *K-means clustering* dans le cadre des méthodes AK ne prend pas en compte l'information de la fonction d'apprentissage U. Les points d'enrichissement optimaux ne sont donc que rarement sélectionnés. L'algorithme *K-weighted-means clustering* est donc proposé pour prendre en compte la fonction d'apprentissage U et ainsi sélectionner de meilleurs points d'enrichissement. Cet algorithme requiert le remplacement de la moyenne arithmétique utilisée par l'algorithme *K-means clustering* par une moyenne pondérée prenant en compte la fonction d'apprentissage U. Ainsi, l'Équation (3.5) est remplacée par l'Équation (3.7).

$$\mathbf{c}_k^{(i+1)} = \frac{\sum_{j=1}^{n_k} \left(\frac{1}{U_j}\right)^2 \mathbf{u}_j}{\left(\frac{1}{U_j}\right)^2} \quad (3.7)$$

où  $U_j$  est la fonction U évaluée au point  $\mathbf{u}_j$ . La seconde modification proposée par l'algorithme *K-weighted-means clustering* correspond au choix final des centroïdes. Les centroïdes sélectionnés pendant l'algorithme n'appartiennent pas à la population de Monte-Carlo à classer. Ainsi, les points d'apprentissages choisis sont les points les plus proches des centroïdes.

### Illustration

La Figure 3.2 illustre les différences dans le choix des points soit sans algorithme de *clustering*, soit avec l'algorithme K-means ou enfin avec l'algorithme K-w-means. L'exemple utilisé est présenté en Annexe A. Lorsque le *clustering* est utilisé, l'enrichissement est réalisé avec 5 points par itération. Les points roses sont ceux appartenant au plan d'expérience défini à l'itération précédente, les points bleus sont les points sélectionnés lors de l'itération courante. Les courbes noire et rouge représentent respectivement la fonction

d'état-limite vraie et approximée par le métamodèle de Krigeage. L'enrichissement classique (sans *clustering*) sélectionne le meilleur point possible selon la fonction d'apprentissage  $U$ . L'utilisation d'algorithme de *clustering* permet d'obtenir une approximation de l'état limite plus rapidement en terme de nombre d'itérations. L'algorithme proposé K-w-means semble sélectionner les points plus efficacement, c'est-à-dire plus proches de la fonction d'état-limite, que l'algorithme classique K-means. En effet, l'utilisation de la somme pondérée par la fonction d'apprentissage  $U$  permet aux centroïdes de se retrouver plus proches des valeurs de  $U$  les plus faibles.

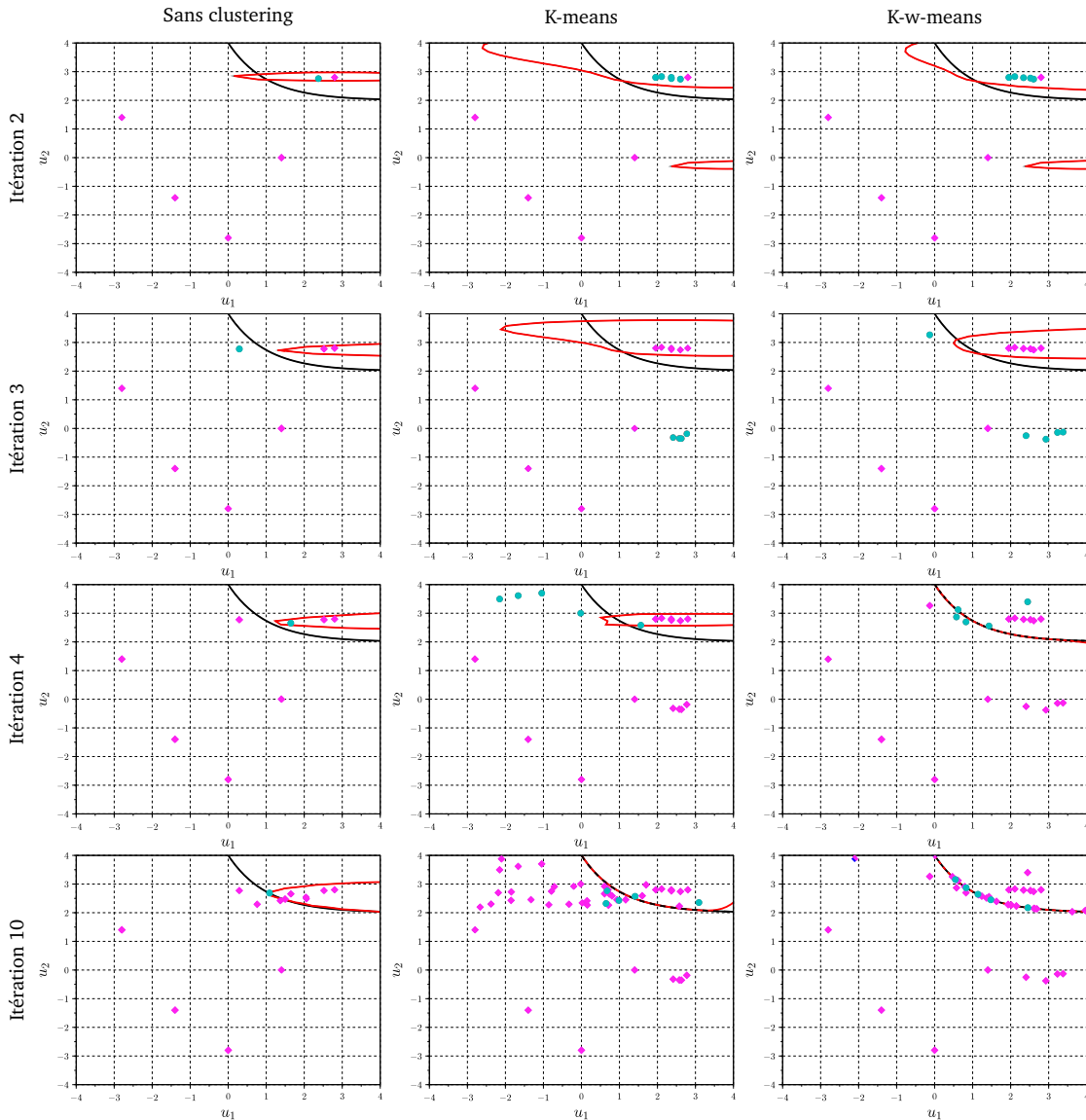


FIGURE 3.2 – Illustration des algorithmes de clustering.

### Discussion sur le nombre optimal de *clusters*

L'algorithme de *clustering* K-w-means requiert le choix d'un nombre fixe de *clusters* de la même façon que l'algorithme K-means. L'optimalité des points d'enrichissement sélectionnés par l'algorithme dépend fortement du nombre de *clusters* choisi initialement. Plusieurs approches peuvent être utilisées pour déterminer le nombre optimal de *clusters*. La première approche est la plus simple et la plus rapide. Le nombre de *clusters* est égal au maximum de calculs possibles en parallèle. Ainsi, si dix calculs peuvent être réalisés en parallèle, alors dix points d'enrichissement seront sélectionnés à chaque itération. La seconde approche cherche à connaître le nombre optimal de *clusters* qui est obligatoirement inférieur au nombre maximal de calculs réalisables en parallèle. Cette approche se base sur des indices de qualité des *clusters*. Le nombre optimal de *clusters* peut être sélectionné suivant la procédure :

1. Définition du nombre maximal  $N_{cmax}$  de calculs réalisables en parallèle.
2. Réalisation de l'algorithme de *clustering* pour tous les nombres de *clusters* compris dans  $[2; N_{cmax}]$ .
3. Évaluation de l'indice de qualité choisi pour chacun des cas.
4. Sélection du nombre de *clusters* optimisant l'indice de qualité.

Différents indices de qualité sont disponibles dans la littérature. L'indice de Dunn (Dunn, 1973) permet d'identifier des *clusters* compacts et bien séparés. Pour cela, l'indice de Dunn est égal au ratio entre la distance minimale entre les *clusters* et la distance maximale à l'intérieur d'un *clusters*. Le nombre optimal de *clusters* est défini comme celui correspondant à la valeur maximale de l'indice. Le second indice présenté dans cette section est l'indice PBM proposé par Pakhira et al., (2004) qui semble particulièrement robuste lorsqu'un grand nombre de données est à partitionner. Beaucoup d'autres indices sont proposés dans la littérature comme par exemple les indices de Ball-Hall (Ball et Hubert, 1965), de Calinski-Harabasz (Caliński et Harabasz, 1974), de Hartigan, (1975), de Xu, (1997), de Davies-Bouldin (Davies et Bouldin, 1979), de Xie-Beni (Xie et Beni, 1991), le *Bayesian Information Criterion* (Fraley et Raftery, 1998) ou le *Silhouette Coefficient* (Kaufman et Rousseeuw, 1990). L'utilisation de tels algorithmes peut toutefois engendrer des temps de calcul longs. En effet, l'algorithme de *clustering* doit être réalisé plusieurs fois. Si la quantité de données à partitionner est importante, alors le temps de calcul va augmenter de manière drastique. Un nombre fixe de *clusters* sera choisi dans la suite et aucun indice de qualité ne sera utilisé.

### 3.3.3 Nouveau critère d'arrêt sur l'enrichissement

L'utilisation des simulations séquentielles de Monte-Carlo présentées en section 3.3.1 implique la considération de plusieurs populations de Monte-Carlo de tailles réduites. L'idée majeure est donc de pouvoir identifier une probabilité de défaillance faible avec des populations de tailles réduites, par exemple estimer  $P_f = 10^{-6}$  à partir de plusieurs populations contenant chacune  $n_p = 10^4$  points. Dans cet exemple, chaque population a peu de chances de contenir un point défaillant. Le challenge d'une telle technique consiste à être sûr de la classification des points dans les populations de tailles réduites. Une fois celle-ci considérée

comme classée correctement, seule la probabilité de défaillance est conservée, la population n'est pas conservée pour ne pas surcharger la mémoire RAM. Bien que beaucoup dans la littérature cherchent à rendre le critère d'arrêt sur l'apprentissage moins strict pour accélérer la convergence de la méthode (voir section 2.4.2), il faut ici s'assurer du bon classement des points selon les domaines de défaillance et de sûreté. Le critère d'arrêt doit donc être plus sévère pour garantir la convergence de la méthode proposée. Un nouveau critère d'arrêt sur l'apprentissage est donc présenté dans cette section.

Soit  $C$  l'évènement suivant « tous les points de la population de Monte-Carlo sont bien classés » et  $C_i$  l'évènement « le point  $\mathbf{u}^{(i)}$  est bien classé ». L'évènement  $C$  se produit si et seulement si tous les évènements  $C_i$  se produisent aussi. La probabilité d'occurrence de l'évènement  $C$  est exprimée par l'Équation (3.8).

$$\mathcal{P}[C] = \mathcal{P}\left[C_1 \cap C_2 \cap \dots \cap C_{n_p}\right] \quad (3.8)$$

Bien que la taille des populations de Monte-Carlo soit réduite, elle reste trop grande pour évaluer exactement la probabilité de l'évènement  $C$ . L'évaluation d'une telle probabilité nécessite l'obtention de la matrice de corrélation entre tous les points. Cependant, cette matrice est composée d'un très grand nombre d'éléments ( $n_p \times n_p$ ) ce qui empêche de calculer directement l'Équation (3.8). Deux hypothèses fortes peuvent être formulées. La première est de considérer que tous les évènements  $C_i$  sont indépendants, c'est-à-dire que la matrice de corrélation est la matrice identité. Cette hypothèse est celle utilisée dans (Lelièvre et al., 2018a). L'Équation (3.8) devient donc :

$$\mathcal{P}[C] \approx \mathcal{P}[C]_{ind} = \prod_{i=1}^{n_p} \mathcal{P}[C_i] \quad (3.9)$$

La seconde hypothèse possible consiste à considérer les points comme totalement corrélés. Pour estimer la probabilité de l'évènement  $C$  selon cette hypothèse, il faut donc identifier les deux points des domaines de sûreté et de défaillance les plus proches de l'état limite. En effet, si tous les points sont parfaitement corrélés, les deux points les plus proches de l'état limite suffisent pour représenter complètement la population de Monte-Carlo. Soit  $C_{\max_{|G| \leq 0}}$  l'évènement suivant « le point de la population de Monte-Carlo correspondant à la plus grande valeur négative de la prédiction est bien classé » et  $C_{\min_{|G| > 0}}$  l'évènement suivant « le point de la population de Monte-Carlo correspondant à la plus petite valeur positive de la prédiction est bien classé ». Ainsi, après quelques manipulations mathématiques, l'estimation de la probabilité de l'évènement  $C$  de l'Équation (3.8) devient donc :

$$\mathcal{P}[C] \approx \mathcal{P}[C]_{cor} = \mathcal{P}\left[C_{\max_{|G| \leq 0}}\right] + \mathcal{P}\left[C_{\min_{|G| > 0}}\right] - 1 \quad (3.10)$$

Les deux hypothèses formulées ci-dessus sont les deux cas extrêmes dans la prise en compte des corrélations entre les différents points. La vraie probabilité de l'évènement  $C$  doit donc être comprise entre ces deux estimations présentée par les Équations (3.9) et (3.10). L'estimation des deux bornes est faite à partir du métamodèle de krigeage qui fournit la moyenne et la variance en chacun des points de la population de Monte-Carlo. Le calcul de ces bornes passe par l'estimation des probabilités des évènements  $C_i$ . Deux

cas différents doivent être étudiés correspondant d'une part aux points classés dans le domaine de défaillance et d'autre part aux points classés dans le domaine de sûreté.

Tout d'abord, considérons  $\mu_{\hat{G}}(\mathbf{u}^{(i)}) > 0$ , correspondant aux points classés dans le domaine de sûreté :

$$\begin{aligned} \mathcal{P}[C_i] &= \mathcal{P}[\hat{G}(\mathbf{u}^{(i)}) > 0] = 1 - \mathcal{P}[\hat{G}(\mathbf{u}^{(i)}) \leq 0] \\ &= 1 - \mathcal{P}\left[\frac{\hat{G}(\mathbf{u}^{(i)}) - \mu_{\hat{G}}(\mathbf{u}^{(i)})}{\sigma_{\hat{G}}(\mathbf{u}^{(i)})} \leq -\frac{\mu_{\hat{G}}(\mathbf{u}^{(i)})}{\sigma_{\hat{G}}(\mathbf{u}^{(i)})}\right] \\ &= 1 - \Phi[-U(\mathbf{u}^{(i)})] = \Phi[U(\mathbf{u}^{(i)})] \end{aligned} \quad (3.11)$$

où  $\mathbf{u}^{(i)}$  est une variable aléatoire gaussienne de moyenne  $\mu_{\hat{G}}(\mathbf{u}^{(i)})$  et de variance  $\sigma_{\hat{G}}^2(\mathbf{u}^{(i)})$ ,  $\Phi$  est la fonction de répartition gaussienne et  $U$  est la fonction d'apprentissage. De la même façon, considérons  $\mu_{\hat{G}}(\mathbf{u}^{(i)}) \leq 0$ , correspondant aux points classés dans le domaine de défaillance :

$$\begin{aligned} \mathcal{P}[C_i] &= \mathcal{P}[\hat{G}(\mathbf{u}^{(i)}) \leq 0] = \mathcal{P}\left[\frac{\hat{G}(\mathbf{u}^{(i)}) - \mu_{\hat{G}}(\mathbf{u}^{(i)})}{\sigma_{\hat{G}}(\mathbf{u}^{(i)})} \leq -\frac{\mu_{\hat{G}}(\mathbf{u}^{(i)})}{\sigma_{\hat{G}}(\mathbf{u}^{(i)})}\right] \\ &= \mathcal{P}\left[\frac{\hat{G}(\mathbf{u}^{(i)}) - \mu_{\hat{G}}(\mathbf{u}^{(i)})}{\sigma_{\hat{G}}(\mathbf{u}^{(i)})} \leq \frac{|\mu_{\hat{G}}(\mathbf{u}^{(i)})|}{\sigma_{\hat{G}}(\mathbf{u}^{(i)})}\right] \\ &= \Phi[U(\mathbf{u}^{(i)})] \end{aligned} \quad (3.12)$$

Ainsi, les Équations (3.9) et (3.10) deviennent respectivement :

$$\mathcal{P}[C]_{ind} = \prod_{i=1}^{n_p} \Phi[U(\mathbf{u}^{(i)})] \quad (3.13)$$

$$\mathcal{P}[C]_{cor} = \Phi[U(\mathbf{u}^{(\max G|G \leq 0)})] + \Phi[U(\mathbf{u}^{(\min G|G > 0)})] - 1 \quad (3.14)$$

Enfin, l'apprentissage doit être arrêté lorsque la probabilité de l'évènement  $C$  est supérieure à une probabilité cible  $P_{cible}$  :

$$\min\{\mathcal{P}[C]_{ind}, \mathcal{P}[C]_{cor}\} \geq P_{cible} \quad (3.15)$$

Ce nouveau critère d'arrêt permet d'assurer le bon classement de chacune des populations de Monte-Carlo. Contrairement au critère d'arrêt classique de AK-MCS, ce nouveau critère est plus strict et prend en compte la totalité des incertitudes sur chaque point. Une étude d'influence de la probabilité cible est réalisée en section 3.4.

## 3.4 Validation de la méthode AK-MCSI

### 3.4.1 Fonction de Rastrigin

Le premier exemple est la fonction de Rastrigin (Törn et Žilinskas, 1989) modifiée pour obtenir une fonction d'état limite non-linéaire et plusieurs domaines de défaillance disjoints. Cette fonction est souvent utilisée dans la littérature dans le but de valider l'efficacité de méthodes d'analyse de fiabilité (Echard et al., 2011; Cadini et al., 2014; Huang et al., 2016). La fonction de performance est décrite par l'Équation (3.16) et illustrée par la Figure 3.3.

$$G(x_1, x_2) = \beta - \sum_{i=1}^2 \left[ x_i^2 - 5 \cos 2\pi x_i \right] \quad (3.16)$$

Les deux variables aléatoires sont distribuées suivant des gaussiennes indépendantes de moyennes nulles et de variances unitaires. Le paramètre  $\beta$  contrôle l'ordre de grandeur de la probabilité de défaillance. L'état limite est illustrée par la Figure 3.4 en fonction du paramètre  $\beta$ . Plus  $\beta$  est grand, plus la probabilité de défaillance est faible.

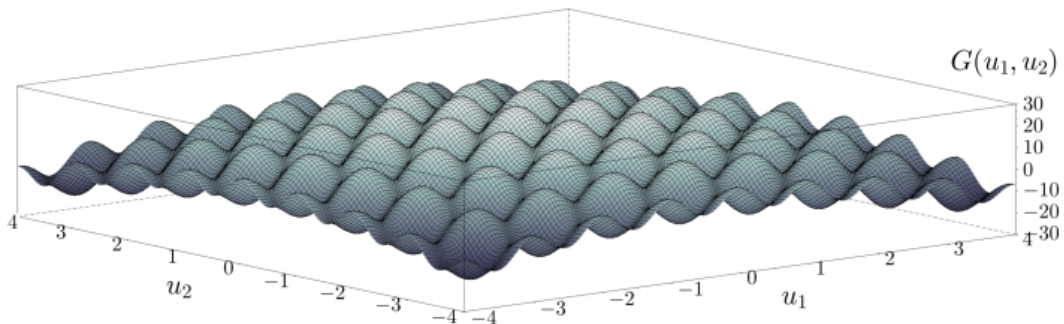


FIGURE 3.3 – Fonction de Rastrigin – Représentation 3D.

La première étude réalisée sur cet exemple concerne l'influence des paramètres de la méthode : le nombre de *clusters* utilisés, la taille des sous-populations  $n_p$  et la probabilité cible du nouveau critère d'arrêt  $P_{cible}$ . Dans l'objectif de comparer les efforts numériques en termes de temps de calcul des méthodes utilisées, toutes les analyses ont été réalisées à partir du même ordinateur avec un Inter(R) Core(TM) i7-4910MQ CPU processor de 2.90GHz avec 32Go RAM. Tout d'abord, le Tableau 3.1 montre l'influence du nombre de *clusters* sur le temps effectif nécessaire pour la convergence de AK-MCSI. L'analyse a été réalisée avec  $\beta = 15$ ,  $P_{cible} = 0.999$ ,  $\delta_{P_f}^{(cible)} = 5\%$  et  $n_p = 10^4$ . Les méthodes débutent toutes avec le même plan d'expériences initial. La taille de la population de Monte-Carlo utilisée pour AK-MCS est de  $7 \times 10^4$  points. Il est important de noter que la fonction de Rastrigin ne demande que très peu de temps pour être évaluée. Le temps CPU correspond donc au temps de post-traitement et non au temps d'évaluation de la fonction de performance. Le temps de post-traitement comprend majoritairement le temps d'évaluation du métamodèle de krigeage sur les sous-populations.

Les deux premières lignes du Tableau 3.1 correspondent aux résultats obtenus par AK-

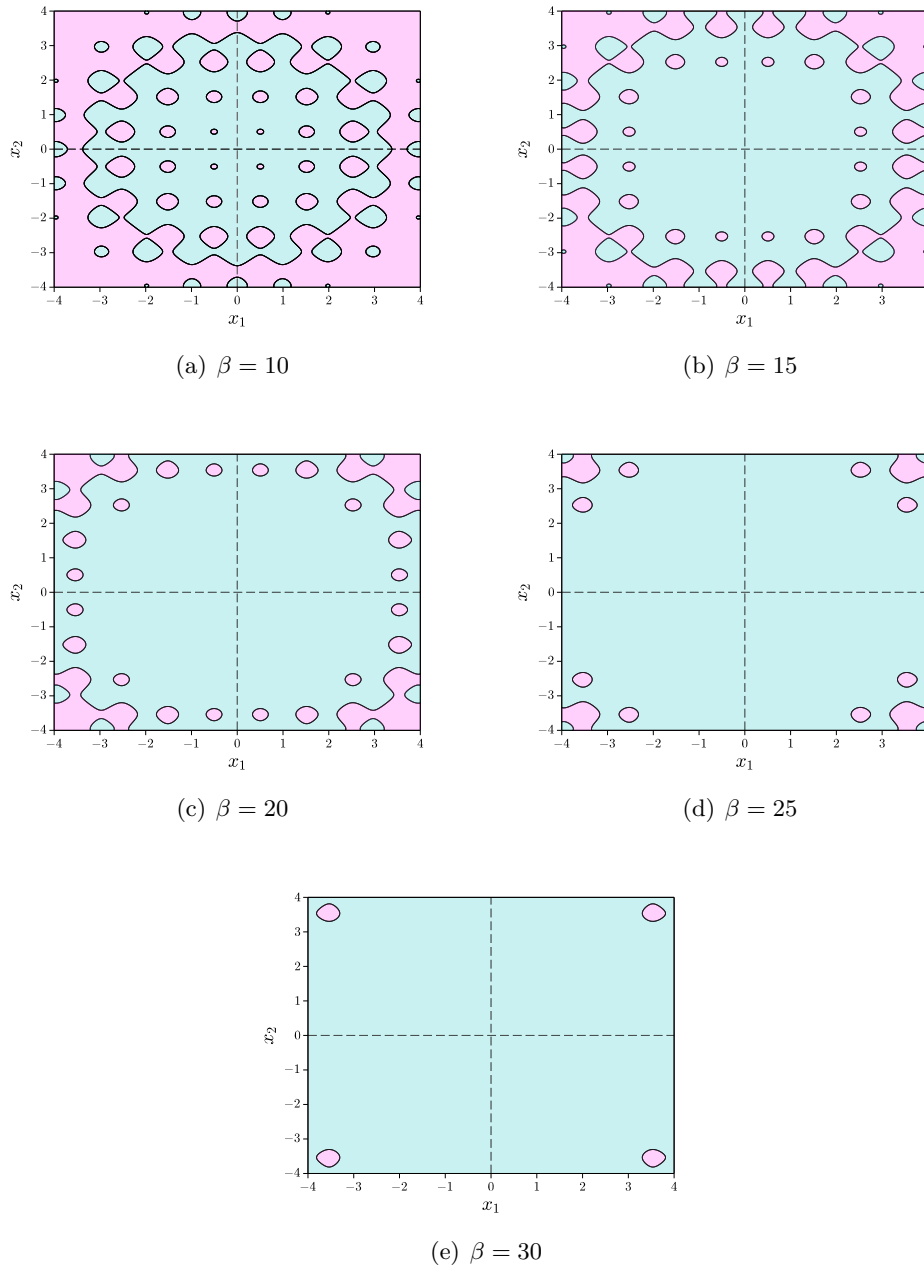


FIGURE 3.4 – Exemple – Fonction de Rastrigin – État limite – En rose : domaine de défaillance. En bleu : domaine de sûreté.

MCS et AK-MCSi sans clustering. Ces résultats montrent une réduction du temps CPU nécessaire uniquement grâce à l'utilisation des simulations séquentielles de Monte-Carlo. Cependant, l'utilisation du nouveau critère d'arrêt révèle une hausse du nombre d'itérations et d'appels à la fonction de performance. En effet, ce nouveau critère d'arrêt est conçu pour être plus conservatif que le critère original de AK-MCS ce qui implique une convergence plus lente de l'algorithme. L'influence de la technique de *clustering* est observée dans la suite du Tableau 3.1. Le nombre d'itérations est une grandeur plus appropriée

Méthode	Nombre de <i>clusters</i>	Nombre d'appels à la fonction de performance	Nombre d'itérations	Temps CPU	
AK-MCS	1	339	320	3006	
	1	394	381	902	
	2	400	197	486	
	5	430	89	246	
	7	454	69	224	
	10	490	54	177	
	15	515	40	149	
	AK-MC <i>S</i> i	20	520	32	128
		25	570	29	143
		30	620	27	161
35		685	26	163	
40		700	24	162	
45		740	23	160	
50		820	23	351	

TABLE 3.1 – Exemple – Fonction de Rastrigin – Influence du nombre de *clusters* – Comparaison des méthodes AK-MCS et AK-MC*S*i –  $P_f \sim 6.3 \times 10^{-3}$  –  $\beta = 15$ ,  $P_{cible} = 0.999$ ,  $\delta_{P_f}^{(cible)} = 5\%$ ,  $n_p = 10000$ ,  $n_{MCS} = 70000$ .

que le nombre d'appels à la fonction de performance pour évaluer la performance de la méthode. En effet, les évaluations de la fonction de performance pouvant être parallélisées, leur nombre ne peut pas décrire correctement la performance de AK-MC*S*i. Le nombre d'itérations de AK-MC*S*i diminue fortement en utilisant l'algorithme de *clustering*. Cependant, l'utilisation d'un trop grand nombre de *clusters* risque d'augmenter le temps CPU nécessaire à la convergence de AK-MC*S*i. Ici, le nombre optimal de *clusters* est égal à 20 en ce qui concerne le nombre d'itérations et le temps CPU.

L'influence de la taille  $n_p$  des sous-populations de Monte-Carlo est ensuite observée en fixant les différents autres paramètres comme suit :  $P_{cible} = 0.999$ ,  $N_{clusters} = 10$  et  $\delta_{P_f}^{(cible)} = 5\%$ . La Figure 3.5 montre que la taille optimale  $n_p$  dépend fortement de l'ordre de grandeur de la probabilité de défaillance à estimer pour un coefficient de variation identique. Plus la probabilité de défaillance à estimer est faible, plus les populations de Monte-Carlo doivent être grandes. En effet, si elles sont trop petites, le nombre d'itérations va augmenter significativement ce qui implique aussi une augmentation du temps CPU.

L'influence de la probabilité cible  $P_{cible}$  du nouveau critère d'arrêt est étudiée en fixant les paramètres de AK-MC*S*i :  $N_{clusters} = 10$  et  $\delta_{P_f}^{(cible)} = 5\%$ . Le Tableau 3.2 présente les résultats de cette étude. Une simulation de Monte-Carlo de référence est effectuée et la fonction de performance est donc évaluée sur la totalité de cette population pour connaître exactement le nombre de points mal classés par AK-MC*S*i. La valeur de la probabilité cible n'a que peu d'influence (dans la limite du raisonnable) sur le nombre de points mal classés si la probabilité de défaillance est assez haute (cas  $\beta = \{15; 20\}$ ). Lorsque la probabilité de défaillance est faible (cas  $\beta = \{25; 30\}$ ), le nombre de points mal classés est réduit par



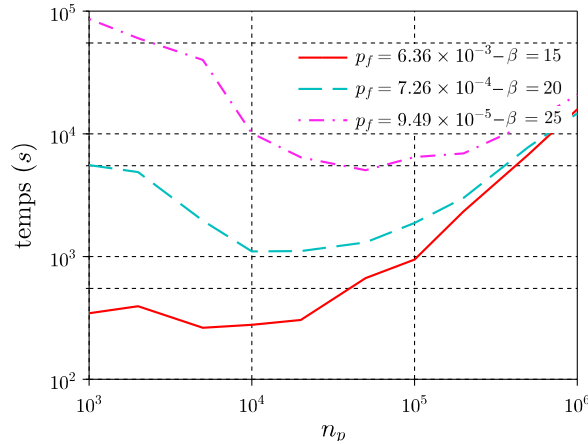


FIGURE 3.5 – Fonction de Rastrigin – Influence de la taille  $n_p$  des sous-populations de Monte-Carlo –  $P_{cible} = 0.999$ ,  $N_{clusters} = 10$ ,  $\delta_{P_f}^{(cible)} = 5\%$ .

l’augmentation de la probabilité cible. Le nombre d’appels à la fonction de performance augmente logiquement avec la probabilité cible puisque le critère d’arrêt devient de plus en plus conservatif. Un compromis doit donc être trouvé entre une bonne précision et un nombre correct d’appels à la fonction de performance.  $P_{cible} = 0.99$  semble être un bon compromis pour cet exemple. Enfin, la méthode AK-MCSi a été exécutée en utilisant le critère original de AK-MCS sur le cas  $\beta = 30$ . Après l’estimation de trois milliards de points, la méthode n’est parvenue à identifier aucun point défaillant alors qu’il en existe 10425. Le critère original de AK-MCS n’est donc pas efficace pour gérer les simulations séquentielles de Monte-Carlo.

Enfin, l’efficacité de AK-MCSi pour estimer une probabilité de défaillance est présentée par la Figure 3.6. Pour chaque valeur de  $\beta$ , la méthode AK-MCSi est répétée indépendamment dix fois pour observer sa robustesse. Les paramètres utilisés sont  $P_{cible} = 0.999$  et  $\delta_{P_f}^{(cible)} = 5\%$ . La méthode AK-MCSi montre une très bonne efficacité même dans le cas de faible probabilité de défaillance ( $P_f = 3 \times 10^{-6}$ ). Le cercle de la Figure 3.6 correspond au cas étudié dans la littérature par Huang et al., (2016) (AK-SS) et Cadini et al., (2014) (meta-AK-IS<sup>2</sup>). Le cas étudié correspond aux paramètres  $\beta = 10$  et  $N_{clusters} = 10$ . Le Tableau 3.3 montre les résultats de cette comparaison. Le germe aléatoire n’est pas identique entre les différentes méthodes utilisées et comparées ; les probabilités de défaillance à estimer ne sont donc pas parfaitement identiques. Ces résultats montrent que la méthode AK-MCSi estime la probabilité de défaillance précisément avec moins d’itérations que les méthodes AK-SS et metaAK-IS<sup>2</sup> mais avec plus d’appels à la fonctions de performance.

$\beta$	$P_{cible}$	$P_f$	Nombre de points mal classés	Nombre d'appels à la fonction de performance
15	0.5	$6.26 \times 10^{-3}$	0	410
	0.6	$6.41 \times 10^{-3}$	0	440
	0.7	$6.3 \times 10^{-3}$	0	420
	0.8	$6.43 \times 10^{-3}$	0	430
	0.9	$6.76 \times 10^{-3}$	0	450
	0.99	$6.55 \times 10^{-3}$	0	480
	0.999	$6.41 \times 10^{-3}$	0	700
20	0.5	$5.35 \times 10^{-4}$	1	540
	0.6	$5.35 \times 10^{-4}$	1	540
	0.7	$5.43 \times 10^{-4}$	0	540
	0.8	$5.42 \times 10^{-4}$	0	550
	0.9	$5.49 \times 10^{-4}$	0	670
	0.99	$5.36 \times 10^{-4}$	0	760
	0.999	$5.53 \times 10^{-4}$	0	1180
25	0.5	$3.16 \times 10^{-5}$	15	640
	0.6	$3.21 \times 10^{-5}$	6	720
	0.7	$3.21 \times 10^{-5}$	4	710
	0.8	$3.23 \times 10^{-5}$	4	670
	0.9	$3.13 \times 10^{-5}$	3	870
	0.99	$3.21 \times 10^{-5}$	0	920
	0.999	$3.12 \times 10^{-5}$	0	1380
30	0.5	$3.64 \times 10^{-6}$	16	740
	0.6	$3.54 \times 10^{-6}$	16	830
	0.7	$3.40 \times 10^{-6}$	4	900
	0.8	$3.48 \times 10^{-6}$	2	960
	0.9	$3.65 \times 10^{-6}$	2	980
	0.99	$3.60 \times 10^{-6}$	0	1160
	0.999	$3.63 \times 10^{-6}$	0	1900

TABLE 3.2 – Fonction de Rastrigin – Influence de la probabilité cible  $P_{cible}$  du nouveau critère d'arrêt –  $N_{clusters} = 10$ ,  $\delta_{P_f}^{(cible)} = 5\%$ .

Pour conclure, la méthode AK-MCSI semble être plus efficace que les méthodes AK-MCS, AK-SS et metaAK-IS<sup>2</sup> pour estimer la probabilité de défaillance de cet exemple. L'utilisation des simulations séquentielles de Monte-Carlo et de l'algorithme de *clustering* permet de diminuer fortement le nombre d'itérations ainsi que le temps CPU nécessaire à la convergence de l'analyse de fiabilité. L'influence des différents paramètres de la méthode a été étudiée. Le nombre de *clusters* utilisés est directement lié au nombre de calculs qu'ils est possible d'exécuter en parallèle. Les autres paramètres dépendent majoritairement de l'ordre de grandeur de la probabilité de défaillance à estimer. Cet ordre de grandeur n'est que rarement connu *a priori*. L'expérience montre que  $n_p = 10^4$  et  $P_{cible} \leq 0.99$  semble être un bon compromis.

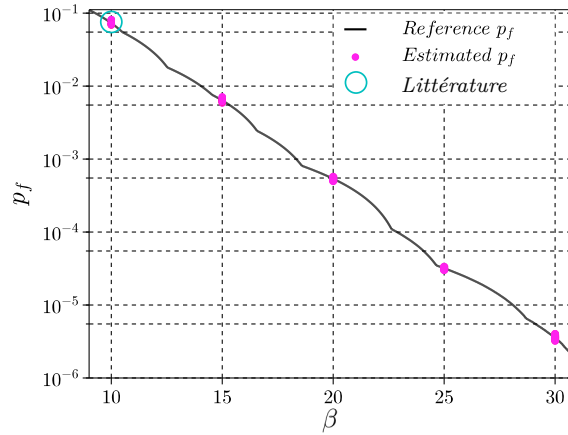


FIGURE 3.6 – Fonction de Rastrigin – Influence de l’ordre de grandeur de la probabilité de défaillance –  $P_{cible} = 0.999$ ,  $N_{clusters} = 10$ ,  $\delta_{P_f}^{(cible)} = 5\%$ .

Méthode	$P_f$	$\delta_{P_f}$ (%)	Nombre d’appels à la fonction de performance	Nombre d’itérations
MCS	$7.34 \times 10^{-2}$	1.45	$6.10^4$	
AK-MCS	$7.13 \times 10^{-2}$	4.66	295	$295 - n_{DoE}^*$
AK-SS	$7.34 \times 10^{-2}$	4.71	288	$288 - n_{DoE}^*$
AK-MCSi	$7.18 \times 10^{-2}$	4.64	330	33
AK-MCS	$7.43 \times 10^{-2}$	2.23	391	$391 - n_{DoE}^*$
metaAK-IS <sup>2</sup>	$7.35 \times 10^{-2}$	2.5	48+432	$48+432 - n_{DoE}^*$
AK-MCSi	$7.12 \times 10^{-2}$	2.28	440	47

TABLE 3.3 – Fonction de Rastrigin – Efficacité de AK-MCSi – Comparaison avec la littérature Huang et al., 2016 ; Cadini et al., 2014 –  $P_{cible} = 0.999$ ,  $N_{clusters} = 10$ ,  $\delta_{P_f}^{(cible)} = 5\%$  – \* Taille du premier plan d’expériences.

### 3.4.2 Fonction de performance avec deux domaines de défaillance distincts

Le second exemple est basé sur une fonction de performance analytique en deux dimensions qui définit deux domaines de défaillance distincts. La fonction de performance est définie en Équation (3.17) et est utilisée dans la littérature par Au et Beck, (1999), Dubourg et al., (2013) et Cadini et al., (2014).

$$G(x_1, x_2) = \min \left\{ \begin{array}{l} c - 1 - x_2 + \exp(-x_1^2/10) + (x_1/5)^4 \\ c^2/2 - x_1 \times x_2 \end{array} \right\} \quad (3.17)$$

où le paramètre  $c$  permet de contrôler l’ordre de grandeur de la probabilité de défaillance à estimer. Les variables aléatoires  $x_i$  sont distribuées suivant des gaussiennes de moyenne nulle et de variance unitaire. Trois cas sont étudiés avec  $c = \{3, 4, 5\}$ ; les états-limites correspondants sont tracés en Figure 3.7. Les coordonnées des trois points de défaillance

les plus probables sont :

$$\begin{cases} P_1^* = \left( \frac{c}{\sqrt{2}}; \frac{c}{\sqrt{2}} \right) \\ P_2^* = \left( -\frac{c}{\sqrt{2}}; -\frac{c}{\sqrt{2}} \right) \\ P_3^* = (0; c) \end{cases}$$

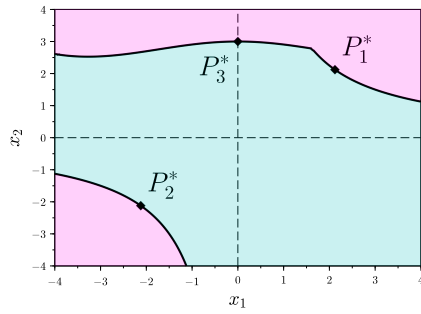
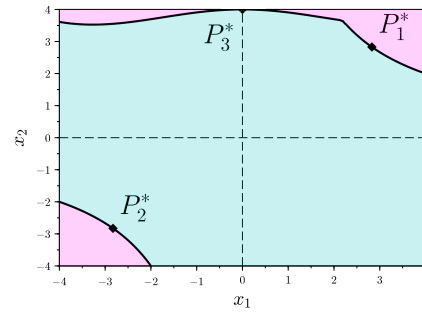
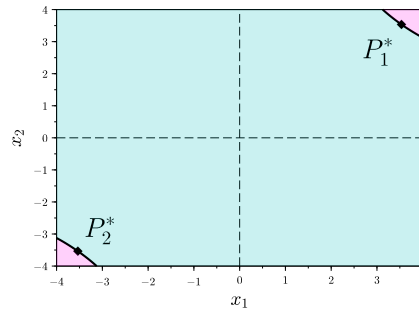
(a)  $c = 3$ (b)  $c = 4$ (c)  $c = 5$ 

FIGURE 3.7 – Fonction avec deux domaines de défaillance – État limite – En rose : domaine de défaillance. En bleu : domaine de sûreté.

Les résultats obtenus par la méthode AK-MCSI sont comparés à différentes méthodes : simulations de Monte-Carlo (MCS), *Subset simulations* (SS) et metaAK-IS<sup>2</sup>. Tous ces résultats sont fournis par Cadini et al., (2014). Le Tableau 3.4 présente les résultats de l'analyse de fiabilité réalisée à partir de chacune des méthodes. La méthode AK-MCSI a été exécutée avec les paramètres suivants :  $N_{clusters} = 5$ ,  $n_p \in \{10^4, 10^5, 10^6\}$  selon le cas étudié et  $P_{cible} = 0.99$ . Les résultats montrent l'efficacité de la méthode AK-MCSI par rapport aux méthodes d'analyse de fiabilité classiques et à la méthode metaAK-IS<sup>2</sup>. La méthode metaAK-IS<sup>2</sup> n'a pas de version d'enrichissement multipoints, ainsi le nombre d'itération est égal au nombre d'appels à la fonction de performance auquel le premier plan d'expériences est soustrait. Le nombre d'itérations de la méthode AK-MCSI semble bien inférieur aux nombres d'itérations de toutes les autres méthodes pour les deux premiers cas ( $c \in \{3; 4\}$ ). Cet exemple montre que AK-MCSI réussit à estimer de très faibles probabilités de défaillance ( $P_f \sim 10^{-7}$ ). Pour ce dernier cas, la méthode metaAK-IS<sup>2</sup> converge plus rapidement en termes d'itération. Cependant, si le nombre d'appels à la

fonction de performance de AK-MCSI est observé, seules  $N_{appels}/N_{clusters} = 255/5 = 51$  itérations servent à enrichir le métamodèle. Les autres itérations servent toutes à générer des populations et à les évaluer sur le métamodèle. C'est pourquoi, le nombre d'itérations obtenu est égal à 484. Cet exemple montre l'efficacité de AK-MCSI pour l'estimation de faibles probabilités de défaillance.

$c$	Méthode	$P_f$	$\delta_{P_f}$	Nombre d'appels à la fonction de performance	Nombre d'itérations
3	MCS	$3.35 \times 10^{-3}$	< 5%	120000	120000
	SS	$3.48 \times 10^{-3}$	< 3%	300000	300000
	metaAK-IS <sup>2</sup>	$3.47 \times 10^{-3}$	< 5%	48 + 69	$48 + 69 - n_{DoE}^*$
	AK-MCSI	$3.63 \times 10^{-3}$	4.8%	140	38
4	MCS	$8.68 \times 10^{-5}$	< 5%	$4.62 \times 10^6$	$4.62 \times 10^6$
	SS	$8.34 \times 10^{-5}$	< 4%	500000	500000
	metaAK-IS <sup>2</sup>	$8.49 \times 10^{-5}$	< 5%	50 + 68	$50 + 68 - n_{DoE}^*$
	AK-MCSI	$9.34 \times 10^{-5}$	4.99%	195	80
5	MCS	$9.48 \times 10^{-7}$	< 5%	$422.11 \times 10^6$	$422.11 \times 10^6$
	SS	$6.55 \times 10^{-7}$	< 5%	700000	700000
	metaAK-IS <sup>2</sup>	$8.16 \times 10^{-7}$	< 5%	117+119	$117 + 119 - n_{DoE}^*$
	AK-MCSI	$9.22 \times 10^{-7}$	4.99%	255	484

TABLE 3.4 – Fonction avec deux domaines de défaillance – Résultats de l'analyse de fiabilité – \* Taille du premier plan d'expériences.

### 3.4.3 Fiabilité d'une structure en treillis

Le troisième exemple concerne la fiabilité d'une structure en treillis illustrée en Figure 3.8. Cette application proposée par Lee et Kwak, (2006) a souvent été utilisée dans le domaine de l'analyse de fiabilité (Blatman et Sudret, 2008, 2011).

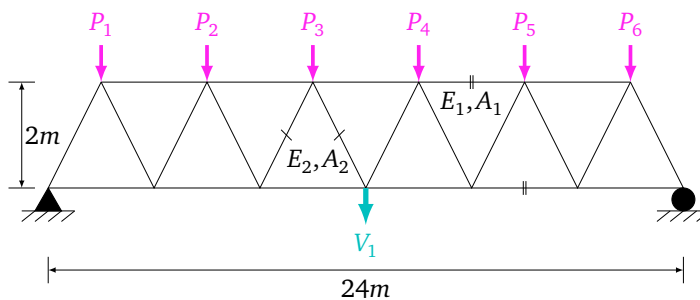


FIGURE 3.8 – Structure en treillis – Schéma du problème.

La défaillance de la structure est définie par une flèche  $V_1$  dépassant un seuil égal à 11 cm. Dix variables aléatoires caractérisent le problème : deux modules d'Young ( $E_1, E_2$ ), deux sections des barres horizontales et obliques ( $A_1, A_2$ ) et six charges appliquées à la structure ( $P_i, i \in \{1, \dots, 6\}$ ). Leurs propriétés sont décrites par le Tableau 3.5. La flèche  $V_1$  est obtenue à partir d'une analyse éléments finis de la structure. L'état limite est illustré

par la Figure 3.9 où sont présentées des coupes des variables deux à deux dans le plan passant par le point de défaillance le plus probable. Cette figure montre la non-linéarité de l'état limite. Le premier plan d'expériences pour AK-MC*Si* contient 80 points générés par LHS et les paramètres de la méthode sont fixés de la manière suivante :

$$\begin{cases} N_{clusters} = 5 \\ n_p \sim 10^4 \\ P_{cible} = 0.99 \end{cases}$$

Variable	Distribution	Moyenne	Écart-type
$E_1, E_2$ (Pa)	Log-normale	$2.1 \times 10^{11}$	$2.1 \times 10^{10}$
$A_1$ (m <sup>2</sup> )	Log-normale	$2.0 \times 10^{-3}$	$2.0 \times 10^{-4}$
$A_2$ (m <sup>2</sup> )	Log-normale	$1.0 \times 10^{-3}$	$1.0 \times 10^{-4}$
$P_1$ - $P_6$ (N)	Gumbel	$5.0 \times 10^4$	$7.5 \times 10^3$

TABLE 3.5 – Structure treillis – Variables aléatoires du problème.

Les résultats de l'analyse de fiabilité sont donnés dans le Tableau 3.6. Ces résultats montrent l'efficacité de la méthode AK-MC*Si* pour estimer la probabilité de défaillance de la structure en treillis. La méthode AK-MC*Si* est ensuite comparée à la méthode AK-MCS. Cet exemple n'est pas traité dans les articles des autres méthodes AK. Les deux méthodes réussissent à estimer la probabilité de défaillance avec une bonne précision. Cependant, AK-MC*Si* requiert beaucoup plus d'appels à la fonction de performance mais bien moins d'itérations avant convergence que AK-MCS. De plus, AK-MC*Si* requiert cinq fois moins de temps CPU pour converger que AK-MCS.

Méthode	Nombre d'appels à la fonction de performance	Nombre d'itérations	$P_f$	$\delta_{p_f}$ (%)	Temps CPU (s)
MCS	$10^6$	$10^6$	$8.70 \times 10^{-3}$	1.07	N/A
AK-MCS	361	341	$8.59 \times 10^{-3}$	1.07	119557
AK-MC <i>Si</i>	985	240	$8.76 \times 10^{-3}$	1.07	22962

TABLE 3.6 – Structure en treillis – Résultats de l'analyse de fiabilité.

Cet exemple montre encore une fois l'efficacité de la méthode AK-MC*Si*. L'utilisation des techniques de simulations séquentielles de Monte-Carlo et de *clustering* permettent de réduire significativement le temps global nécessaire pour que la méthode converge.

### 3.4.4 Fiabilité d'un problème de fluage et de fatigue

Le quatrième exemple est un problème de fluage et de fatigue provenant de l'ingénierie, présenté par Mao et Mahadevan, (2000). Xiong et Tan, (2018) présentent les résultats des méthodes AK-MCS, AK-IS et AK-RBIS sur cet exemple; ces résultats seront comparés ici à la méthode proposée AK-MC*Si*. La fonction de performance est définie en Équation (3.18).

$$G(N_c, N_f, n_c, n_f, \theta_1, \theta_2) = 2 - \exp\left[\frac{\theta_1 n_c}{N_c}\right] + \frac{(\exp[\theta_1] - 2) \left(\exp\left[-\frac{\theta_1 n_c}{N_c}\right] - 1\right)}{\exp[-\theta_2] - 1} - \frac{n_f}{N_f} \quad (3.18)$$

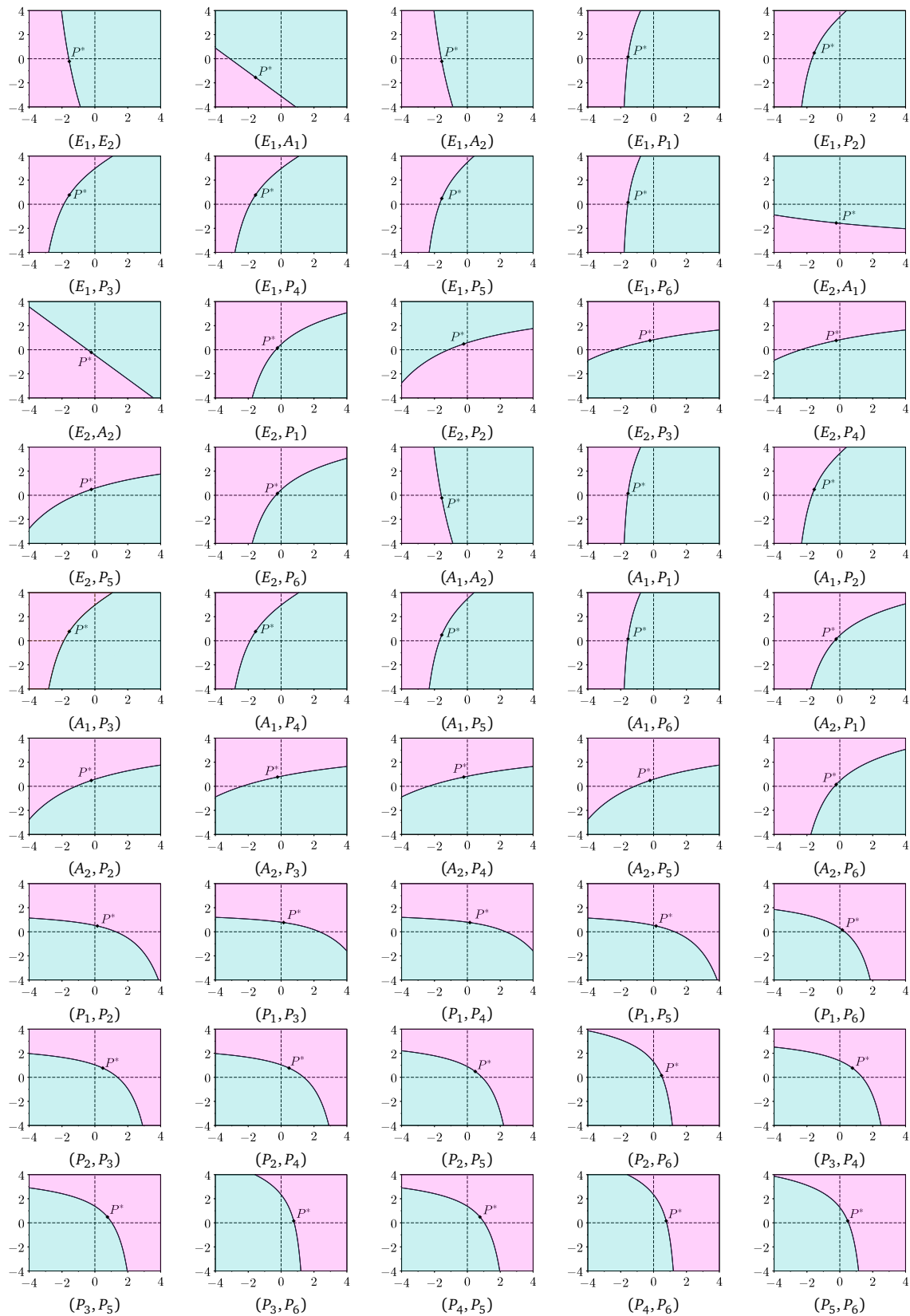


FIGURE 3.9 – Structure en treillis – État-limite du problème de fiabilité dans le plan du point de défaillance le plus probable.

où  $n_c$  et  $n_f$  sont respectivement les nombres de cycles réalisés en fluage et en fatigue,  $N_c$  et  $N_f$  sont respectivement les nombres de cycles de fluage et de fatigue à défaillance.  $\theta_1$  et  $\theta_2$  sont des paramètres empiriques permettant de construire un modèle reliant le dommage dû au fluage et le dommage dû à la fatigue (voir Mao et Mahadevan, 2000). Les caractéristiques de ces variables aléatoires sont présentées dans le Tableau 3.7. Le premier plan d'expériences est construit à partir de 10 points générés par LHS. La forme de l'état-limite est illustrée en Figure 3.10 où des coupes de variables deux à deux dans le plan du point de défaillance le plus probable sont tracées.

Variable	Distribution	Moyenne	Écart-type
$N_c$	Log-Normale	5490	1098
$N_f$	Log-Normale	17100	3420
$n_c$	Log-Normale	549	109.8
$n_f$	Log-Normale	4000	800
$\theta_1$	Normale	0.42	0.084
$\theta_2$	Normale	6	1.2

TABLE 3.7 – Fiabilité fluage et fatigue – Variables aléatoires du problème.

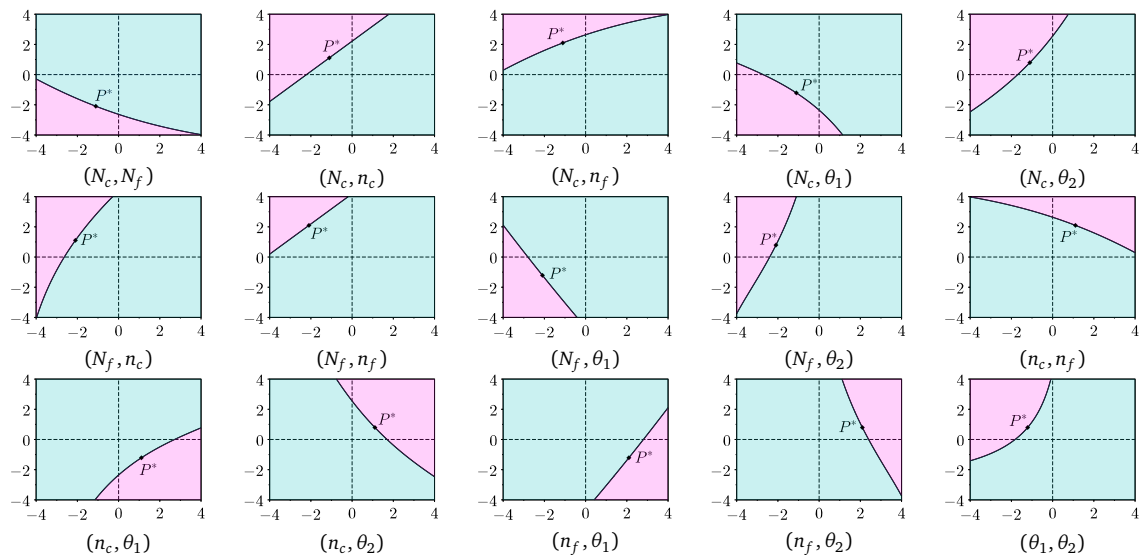


FIGURE 3.10 – Fiabilité fluage et fatigue – État-limite dans le plan du point de défaillance le plus probable.

Les résultats de l'analyse de fiabilité sont présentés dans le Tableau 3.8. Exceptés ceux de la méthode AK-MCSi, tous les autres résultats proviennent de Xiong et Tan, (2018). Ces résultats montrent l'efficacité de la méthode proposée pour estimer une faible probabilité de défaillance. Le nombre d'itérations avant convergence de la méthode semble inférieur ou équivalent aux nombres d'itérations de toutes les autres méthodes présentées. Les utilisations des simulations séquentielles de Monte-Carlo et de l'algorithme de *clustering* semblent rendre la méthode AK-MCSi très efficace pour estimer une probabilité de défaillance avec peu d'itérations et donc peu de temps de calcul.



Méthode	$P_f$	$\delta_{P_f}$ (%)	Nombre d'appels à la fonction de performance	Nombre d'itérations
MCS	$1.42 \times 10^{-4}$	0.84	$10^8$	$10^8$
AK-MCS	$1.49 \times 10^{-4}$	4.9	90	$90 - n_{DoE}^*$
AK-IS	$1.38 \times 10^{-4}$	2.8	42 + 38	$42 + 38 - n_{DoE}^*$
AK-RBIS	$1.35 \times 10^{-4}$	4.9	26 + 35	$26 + 35 - n_{DoE}^*$
AK-MCSi	$1.45 \times 10^{-4}$	4.96	135	53

TABLE 3.8 – Fiabilité fluage et fatigue – Résultats de l'analyse de fiabilité (Xiong et Tan, 2018) – \* Taille du premier plan d'expériences.

### 3.4.5 Réponse dynamique d'un oscillateur non-linéaire

Le dernier exemple est la réponse dynamique d'un système non-linéaire sans amortissement à un degré de liberté illustré en Figure 3.11. Cette application a été utilisée plusieurs fois dans la littérature (Echard et al., 2011, 2013; Tong et al., 2015; Huang et al., 2016). La fonction de performance est définie à l'Équation (3.19).

$$G(C_1, C_2, M, R, T_1, F_1) = 3R - \left| \frac{2F_1}{M\omega_0^2} \sin\left(\frac{\omega_0 T_1}{2}\right) \right| \quad (3.19)$$

avec  $\omega_0 = \sqrt{(C_1 + C_2)/M}$ .  $R$  est le déplacement correspondant à la défaillance d'un ressort,  $C_1$  et  $C_2$  sont les caractéristiques des ressorts,  $M$  est la masse de l'oscillateur et  $F(t)$  est un charge rectangulaire de durée  $T_1$  et d'amplitude  $F_1$ . La fonction de performance dépend de ces six variables aléatoires dont les lois de probabilités sont détaillées dans le Tableau 3.9. Deux cas sont étudiés : seule la définition de la force appliquée  $F_1$  (voir Tableau 3.9) est différente et engendre deux probabilités de défaillance avec des ordres de grandeur très différents. Les états-limites des deux cas sont illustrés en Figures 3.12 et 3.13. Des coupes des variables deux à deux sont présentées dans le plan passant par le point de défaillance le plus probable. L'état-limite de ce problème de fiabilité est peu non-linéaire.

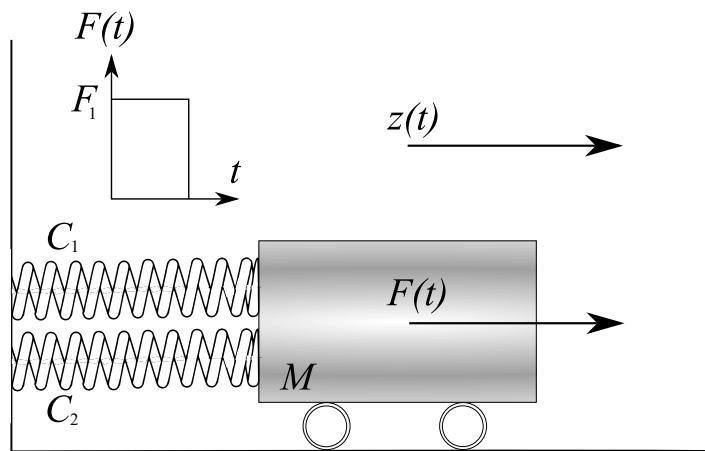


FIGURE 3.11 – Oscillateur non-linéaire – Schéma du problème (Tong et al., 2015).

Variable	Distribution	Moyenne	Écart-type
$C_1$	Normale	1	0.1
$C_2$	Normale	0.1	0.01
$M$	Normale	1	0.05
$R$	Normale	0.5	0.05
$T_1$	Normale	1	0.2
$F_1$ Cas 1	Normale	1	0.2
$F_1$ Cas 2	Normale	0.45	0.075

TABLE 3.9 – Oscillateur non-linéaire – Variables aléatoires du problème.

Les résultats du Cas 1 sont présentés par le Tableau 3.10. La probabilité de défaillance à estimer n'est pas faible, ainsi les deux méthodes AK-MCS et AK-MCSI parviennent à fournir une estimation précise de la probabilité de défaillance. Cinq *clusters* sont utilisés pour AK-MCSI ce qui encore une fois provoque un plus grand nombre d'appels à la fonction de performance mais un plus petit nombre d'itérations comparé à la méthode AK-MCS. Le temps CPU requis par AK-MCSI est aussi plus faible que celui nécessité par AK-MCS.

Le Tableau 3.11 présente les résultats de l'analyse de fiabilité du Cas 2. La probabilité de défaillance à estimer est très faible ( $P_f \sim 10^{-8}$ ). Ainsi, un très grand nombre de sous-populations doivent être considérées une par une. Il est choisi ici d'observer uniquement l'efficacité des simulations séquentielles de Monte-Carlo ; l'algorithme de *clustering* n'est donc pas utilisé. Les résultats montrent que AK-MCSI réussit à estimer une probabilité de défaillance très faible avec une bonne précision. La très grande taille de la population de Monte-Carlo considérée implique que AK-MCS ne peut pas être appliquée. En effet, l'espace mémoire nécessaire devient trop grand.

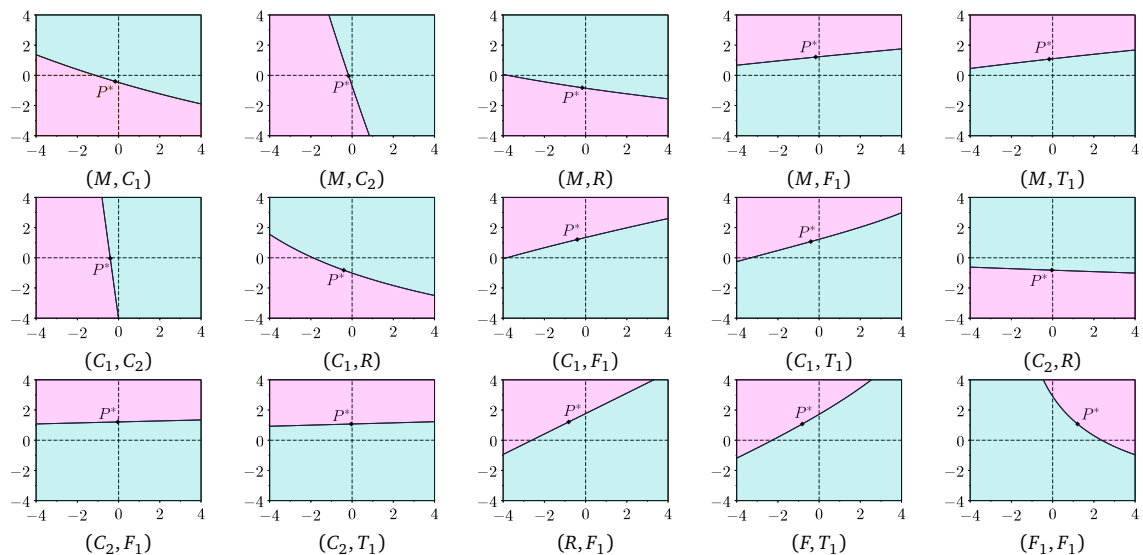


FIGURE 3.12 – Oscillateur non-linéaire – État-limite du Cas 1 dans le plan du point de défaillance le plus probable.

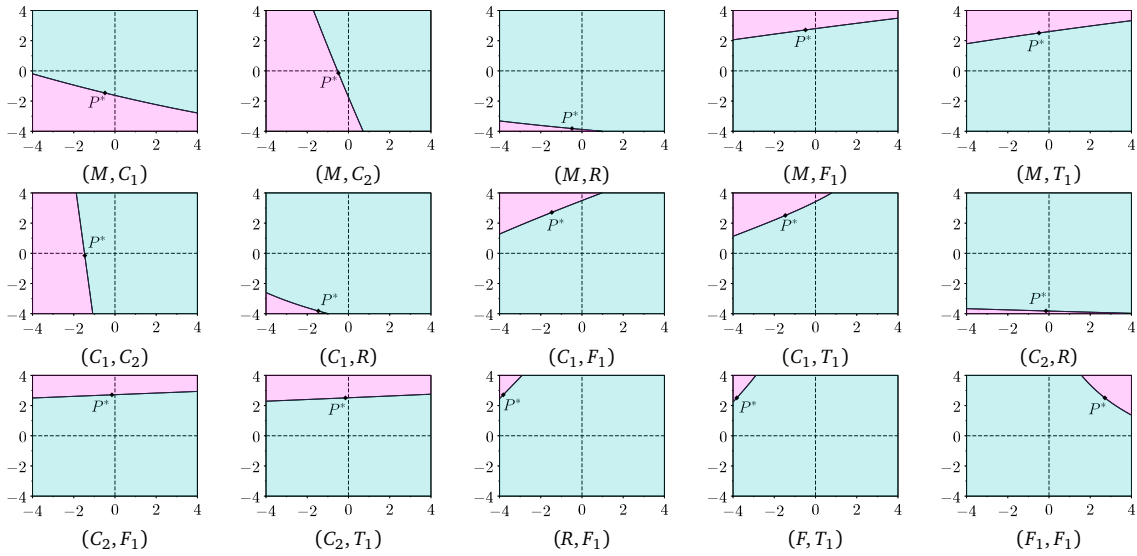


FIGURE 3.13 – Oscillateur non-linéaire – État-limite du Cas 2 dans le plan du point de défaillance le plus probable.

Méthode	Nombre d'appels à la fonction de performance	Nombre d'itérations	$P_f$	$\delta_{P_f}(\%)$	Temps CPU (s)
MCS	$7 \times 10^4$	$7 \times 10^4$	$2.834 \times 10^{-2}$	2.2	N/A
AK-MCS	62	50	$2.834 \times 10^{-2}$	2.2	435
AK-MCSi	85	28	$2.832 \times 10^{-2}$	2.14	55

TABLE 3.10 – Oscillateur non-linéaire – Résultats de l'analyse de fiabilité du Cas 1.

Méthode	Nombre d'appels à la fonction de performance	$P_f$	$\delta_{P_f}(\%)$
MCS	$5.73 \times 10^{10}$	$1.51 \times 10^{-8}$	3.4
AK-MCSi	77	$1.44 \times 10^{-8}$	3.4

TABLE 3.11 – Oscillateur non-linéaire – Résultats de l'analyse de fiabilité du Cas 2.

Pour conclure, cet exemple montre l'efficacité de la méthode proposée même lorsque de très faibles probabilités de défaillance sont à estimer. Il confirme aussi que l'utilisation des simulations séquentielles de Monte-Carlo et de l'algorithme de *clustering* augmente le nombre d'appels à la fonction de performance mais diminue le nombre d'itérations et le temps CPU. La possibilité de paralléliser les évaluations de la fonction de performance permet donc de fortement diminuer le temps de calcul nécessaire à l'analyse de fiabilité.

### 3.5 Conclusions

Une revue bibliographique des méthodes d'analyse de fiabilité permettant d'estimer les faibles probabilités de défaillance a été présentée considérant les trois grands types d'approches telles que l'approximation, la simulation et celle basée sur les métamodèles. Bien que de nombreuses méthodes aient été proposées dans la littérature pour traiter un tel problème, il reste un challenge vis-à-vis du temps de calcul nécessaire ainsi que de la difficulté à identifier une zone restreinte de l'espace.

Une nouvelle méthode a été proposée dans ce chapitre permettant d'obtenir une bonne précision sur l'estimation de faibles probabilités de défaillance. Cette méthode est basée sur le schéma des méthodes AK, c'est-à-dire sur l'utilisation du Krigeage et plus particulièrement sur son schéma adaptatif basé sur la fonction d'apprentissage U. Ce chapitre présente trois grandes améliorations à la méthode AK-MCS. La première considère l'utilisation de simulations séquentielles de Monte-Carlo. Son principe est de diviser la population de Monte-Carlo initialement grande en sous-populations de tailles réduites qui permettent une évaluation rapide du métamodèle. Un gain de temps est donc observé lorsque ces simulations sont utilisées. La deuxième amélioration concerne l'enrichissement multi-points, c'est-à-dire l'enrichissement du métamodèle en plusieurs points à chaque itération de la méthode. Le nombre d'itérations nécessaires à la convergence de la méthode est fortement réduit même si le nombre d'appels à la fonction de performance augmente. Enfin, un nouveau critère d'arrêt sur l'enrichissement est proposé pour s'assurer du bon classement de chacune des sous-populations (potentiellement sans point défaillant). Enfin, les cinq applications présentées montrent l'efficacité de la méthode proposée comparée aux méthodes AK de la littérature telles que AK-MCS, AK-IS, AK-RBIS, AK-SS et metaAK-IS<sup>2</sup>.

La méthode AK-MCSi a ainsi été proposée pour estimer de faibles probabilités de défaillance. Cependant, comme beaucoup de méthodes basées sur les métamodèles, elle est victime de la dimension du problème à traiter. Le chapitre suivant propose de se concentrer sur les problèmes en dimension élevée, considérant un nombre important de variables aléatoires.

### Synthèse du chapitre

**Objectif.** Proposer une nouvelle méthode afin de pallier à deux inconvénients majeurs des méthodes AK : l'estimation de faibles probabilités de défaillance liées à des événements rares et l'enrichissement multiple du métamodèle.

**Propositions.** La méthode AK-MC*S*i a été développée et est une amélioration de la méthode AK-MC*S* composée de trois nouvelles techniques :

- Les simulations séquentielles de Monte-Carlo.
- Un nouveau critère d'arrêt sur l'enrichissement.
- Une technique d'enrichissement multipoints.

**Publications scientifiques.**

- N. Lelièvre et al. (2018a). « AK-MC*S*i : A Kriging-based method to deal with small failure probabilities and time-consuming models ». In : *Structural Safety* 73, p. 1–11
-

# Chapitre 4

## Analyse de fiabilité en grande dimension

### Sommaire

---

<b>4.1</b>	<b>Introduction</b>	<b>86</b>
<b>4.2</b>	<b>Méthodes de fiabilité en grande dimension</b>	<b>86</b>
4.2.1	Challenge de la grande dimension	86
4.2.2	Méthodes de simulation en grande dimension	87
<b>4.3</b>	<b>Utilisation des métamodèles en grande dimension</b>	<b>87</b>
4.3.1	Optimisation directe des hyper-paramètres	88
4.3.2	Techniques de réduction de dimension	89
4.3.3	Techniques de décompositions fonctionnelles	91
<b>4.4</b>	<b>Méthodes AK-HDMR</b>	<b>93</b>
4.4.1	Méthode AK-HDMR1	94
4.4.2	Méthode AK-HDMR2	97
4.4.3	Validation et étude des paramètres de la méthode AK-HDMR1	100
<b>4.5</b>	<b>Méthode AK-PCA</b>	<b>104</b>
4.5.1	Présentation de la méthode AK-PCA	104
4.5.2	Validation et étude des paramètres de la méthode AK-PCA	105
<b>4.6</b>	<b>Comparaison des méthodes proposées</b>	<b>107</b>
4.6.1	Fonction de Valdebenito	107
4.6.2	Fonction modifiée de Griewank – Limitations des méthodes proposées	109
<b>4.7</b>	<b>Conclusions</b>	<b>110</b>

---

## 4.1 Introduction

La résolution de problèmes de fiabilité comptant un grand nombre de variables d'entrée ( $> 20$ ) est un véritable challenge bien connu dans la communauté sous le nom anglais : « *the curse of dimensionality* ». Les méthodes de simulation permettent souvent d'estimer des probabilités de défaillance en présence d'un grand nombre de variables aléatoires mais requièrent un grand nombre d'appels au modèle numérique, qui est ici considéré coûteux. Ces méthodes ne sont donc souvent pas applicables. D'autre part, les méthodes d'approximation FORM/SORM souffrent de la dimension puisque le point de défaillance le plus probable perd sa signification lorsque la dimension du problème augmente. Ce challenge a bien été identifié dans la littérature et est présenté plus en détail en section 4.2. Des adaptations des méthodes traditionnelles y sont aussi présentées. L'utilisation de métamodèles est ensuite discutée en section 4.3 par la présentation des différentes options disponibles permettant de résoudre des problèmes en grande dimension. Deux méthodes sont ensuite proposées en sections 4.4 et 4.5 puis comparées sur des applications complexes en section 4.6. Les différentes applications montrent l'efficacité ainsi que les limitations des méthodes proposées.

## 4.2 Méthodes de fiabilité en grande dimension

### 4.2.1 Challenge de la grande dimension

La grande dimension est un véritable challenge pour l'analyse de fiabilité. Katafygiotis et Zuev, (2008) propose une explication géométrique des difficultés rencontrées lorsque la dimension du problème devient grande. Dans l'espace standard, le carré de la norme euclidienne  $R$  d'un point  $\mathbf{u}$  suit une loi du chi-deux à  $d$  degrés de liberté,  $d$  étant la dimension du problème :

$$R^2 = \sum_{i=1}^d u_i^2 \sim \chi_d^2 \quad (4.1)$$

Ainsi, Katafygiotis et Zuev, (2008) montrent que la norme euclidienne est approximativement une variable gaussienne telle que :

$$R \sim \mathcal{N}(\sqrt{d-1/2}, 1/2) \approx \mathcal{N}(\sqrt{d}, 1/2) \quad (4.2)$$

Cette approximation montre que la majeure partie de la contribution à la probabilité de défaillance provient de zones éloignées de l'origine de l'espace standard et du point de défaillance le plus probable. Pour illustrer cette assertion, prenons l'exemple d'un problème de fiabilité contenant une centaine de variables aléatoires. La fonction d'état limite est supposée linéaire et la probabilité de défaillance proche de  $10^{-5}$ . L'analyse FORM fournit un indice de fiabilité  $\beta$  approximativement égal à 4 alors que la masse de probabilité est centrée autour de  $\sqrt{d} = 10$ . La masse de probabilité n'est donc pas dans le voisinage du point de défaillance le plus probable. Ainsi, les méthodes basées sur le point de défaillance le plus probable (FORM, SORM, IS basés sur le MPFP,...) ne sont donc plus efficaces lorsque la dimension augmente (Valdebenito et Schuëller, 2010).

Les autres méthodes de simulation peuvent aussi souffrir de l'augmentation de la dimension même si le point de défaillance le plus probable n'est pas utilisé. Les *Subset*

*simulations* peuvent rencontrer des difficultés dans la génération des sous-populations. En effet, l'algorithme modifié de Metropolis-Hasting (Au et Beck, 2001) utilisé traditionnellement montre des taux de rejets très importants lorsque la dimension augmente. L'efficacité de la méthode diminue donc fortement.

#### 4.2.2 Méthodes de simulation en grande dimension

Certaines méthodes d'analyse de fiabilité ont donc été adaptées permettant de passer outre leurs difficultés à traiter des problèmes en grande dimension.

Les *subset simulations* ont reçus une attention particulière dans cet objectif. Katafygiotis et Cheung, (2003) proposent de remplacer l'algorithme de Metropolis-Hasting (MCMC) par un algorithme basé sur des tirages par partie qui offre une bonne efficacité jusqu'à 4000 variables aléatoires. Plus tard, les mêmes auteurs (Katafygiotis et Cheung, 2007) étudient la puissance de deux de leurs méthodes basées sur les *subset simulations* : *Spherical Subset Simulation* ( $S^3$ ) et *Auxiliary Domain Method* (ADM). De nouveaux algorithmes MCMC sont proposés dans l'objectif de diminuer le taux de rejet de l'algorithme traditionnel. La méthode  $S^3$  propose de diviser le domaine de défaillance en plusieurs sous-domaines appropriés alors que la méthode ADM propose d'identifier un domaine de défaillance auxiliaire. Une quatrième méthode, intitulée S++, a été proposée par Zuev et al., (2012) dans le cadre bayésien et permet de quantifier les incertitudes sur l'estimation de la probabilité de défaillance. Enfin, Papaioannou et al., (2015) présentent une comparaison de plusieurs algorithmes de génération MCMC pour les *subset simulations*. Une nouvelle méthode ainsi qu'une variante à celle-ci y sont aussi proposées.

Bien que plusieurs adaptations aient été proposées dans la littérature, les méthodes de simulation requièrent de nombreux appels à la fonction de performance, qui est ici supposée coûteuse. L'utilisation de métamodèles peut donc être une alternative. Cependant, la grande dimension représente aussi un challenge complexe pour les méthodes sur ces derniers.

### 4.3 Utilisation des métamodèles pour l'analyse de fiabilité en grande dimension

L'utilisation de métamodèles pour résoudre des problèmes de fiabilité en grande dimension peut se confronter à deux types de problématiques selon le métamodèle considéré. Pour certains métamodèles, comme le chaos polynomial, la grande dimension influe sur le nombre de coefficients à déterminer qui peut devenir très grand. Des techniques ont été proposées permettant de réduire ce nombre de coefficients (voir la méthode de Chaos Polynomial Creux (Sudret, 2012)). D'autres métamodèles (SVM, Krigeage,...) nécessitent l'identification d'hyper-paramètres à partir d'un plan d'expériences. L'optimisation induite par cette identification est souvent de dimension identique à celle du problème de fiabilité. Ces travaux de thèse traitent plus particulièrement la problématique suivante : comment calibrer le métamodèle de krigeage lorsque le problème de fiabilité est de dimension élevée ? Pour cela, la littérature regroupe trois grandes approches : l'optimisation directe des hyper-paramètres, la réduction de la dimension et la décomposition fonctionnelle. Des techniques pour ces trois approches sont présentées dans la suite de cette section.

---



### 4.3.1 Optimisation directe des hyper-paramètres

La première approche présentée ici est l'optimisation directe des hyper-paramètres de krigeage. Pour cela, deux types de techniques sont discutés dans la suite : une méthode d'optimisation itérative des hyper-paramètres un par un et des méthodes d'optimisation des hyper-paramètres de manière « globale ».

#### Algorithme d'optimisation itératif

La première technique (Welch et al., 1992) propose d'optimiser les hyper-paramètres un par un et de manière itérative. Pour rappel, les hyper-paramètres  $\theta$  doivent être identifiés de telle manière que la fonction de vraisemblance soit maximisée. L'algorithme basique proposé par Welch et al., (1992) est le suivant :

1. Maximiser la fonction de vraisemblance considérant  $\theta_1 = \theta_2 = \dots = \theta_d = \theta$ . Le maximum trouvé est noté  $l_0$ .
2. Soit  $\mathcal{C} = \{1, 2, \dots, d\}$
3. Optimisation des hyper-paramètres un par un :
  - (a) Pour chaque indice  $j \in \mathcal{C}$ , maximiser la fonction de vraisemblance avec  $\theta_i = \theta$  avec  $i \in \{\mathcal{C} - j\}$ . Le maximum trouvé est noté  $l_j$ .
  - (b) L'indice  $j$  produisant la plus grande différence  $l_j - l_0$  est noté  $j^*$ .
  - (c) Si  $l_{j^*} - l_0$  est assez large, l'indice  $j^*$  est supprimé de  $\mathcal{C}$ , c'est-à-dire  $\mathcal{C} = \mathcal{C} - j^*$ , et  $l_0 = l_{j^*}$ . Sinon, l'algorithme s'arrête et les hyper-paramètres associés à  $l_0$  calculé à l'étape précédente sont sélectionnés.

Welch et al., (1992) remarquent que cet algorithme peut engendrer des coûts excessifs et proposent donc des petites alternatives pour diminuer le temps de calcul nécessaire.

Des tests ont montré que les résultats d'une telle optimisation ne sont pas assez précis pour la calibration d'un métamodèle de krigeage dans le cadre de l'analyse de fiabilité. De plus, le temps requis par une telle optimisation reste grand et ne permet pas son utilisation dans une approche itérative.

#### Algorithmes évolutionnaires

Les algorithmes évolutionnaires sont une famille d'algorithmes s'inspirant de la théorie de l'évolution. Les algorithmes génétiques font partie de ces algorithmes et ont été introduits par Holland (1992). Le principe de tels algorithmes est de simuler l'évolution d'une population à partir de croisements et de mutations des gènes des individus. Une sélection est ensuite réalisée permettant naturellement d'améliorer les individus itérativement en conservant les individus les plus adaptés. Ces algorithmes permettent d'obtenir des résultats satisfaisants à des problèmes d'optimisation souvent très complexes et multi-critères. Différents algorithmes génétiques sont présents dans la littérature comme par exemple l'algorithme génétique à évolution vectorielle (Schaffer, 1985) ou le *Nondominated Sorting Genetic Algorithm II* (NSGA-II) (Deb et al., 2002). Une autre approche est la stratégie d'évolution dont l'un des algorithmes les plus connus est *Covariance Matrix Adaptation Evolution Strategies* (CMA-ES) (Hansen et al., 2003). Cet algorithme est très

efficace et permet d'obtenir des résultats d'optimisation de bonne qualité. Les algorithmes évolutionnaires ont été utilisés dans la littérature dans le cadre de l'identification des hyper-paramètres (voir par exemple Pai, 2006 ; Chen, 2007 ; Cheng et Li, 2008 ; Bourinet, 2018).

Ces différents algorithmes ne souffrent pas de la dimension dans l'atteinte d'un optimum assez précis. Cependant, ces algorithmes sont souvent difficiles à paramétrer et requièrent donc des travaux préliminaires fastidieux. D'autre part, leur convergence peut être longue. L'utilisation de tels algorithmes dans un schéma adaptatif, c'est-à-dire une optimisation nécessaire à chaque itération, peut devenir fortement coûteuse.

### 4.3.2 Techniques de réduction de dimension

La réduction de la dimension du problème de fiabilité nécessite d'identifier des variables qui permettent de représenter globalement le phénomène étudié. L'approche la plus classique en analyse de fiabilité est d'identifier l'influence de la variation des variables d'entrée sur la réponse à partir d'une analyse préalable de sensibilité. Cette analyse permet donc de simplifier le modèle en considérant uniquement les variables les plus influentes. Plusieurs types de méthodes ont été proposés dans la littérature dans cet objectif. Le premier correspond aux méthodes de criblage qui permettent d'explorer efficacement et rapidement l'espace des entrées et d'appréhender le comportement de la réponse du modèle. Différentes méthodes existent pour résoudre ce problème, la méthode de Morris en est un exemple (Morris, 1991). Le second type de méthodes correspond à la recherche de mesures d'importance des variables d'entrée et se basent sur la simulation d'échantillons représentatifs. Différentes méthodes ont été proposées. La plus connue se base sur une décomposition de la variance du modèle et le calcul des indices de Sobol' (Sobol, 1993). Lorsque le modèle est coûteux, l'obtention d'un échantillon assez grand est alors coûteux et ce type de méthodes n'est plus envisageable. Davantage de détails sur l'analyse de sensibilité et le coût qui est lié sont présentés par Iooss, (2011).

Une autre approche de réduction de dimension consiste à identifier des composantes décorrélées, combinaison linéaire des variables d'entrée, expliquant la relation entrées/sortie du modèle étudié. Deux techniques sont présentées dans cette section : l'analyse en composantes principales et la méthode *Partial Least Square*. D'autres techniques existent dans la littérature comme la décomposition en valeurs singulières (voir Routh, 2016) ou l'analyse en composantes indépendantes (voir Herault et al., 1985 ; Jutten et Herault, 1991) mais ne sont pas présentées ici.

#### Analyse en composantes principales

L'analyse en composantes principales (PCA pour *Principal Component Analysis*) est une méthode d'analyse de données permettant de trouver les liens entre les différentes variables d'entrée d'un problème. L'analyse en composantes principales permet de transformer l'espace des variables d'entrée en un espace où toutes les variables sont décorrélées les unes des autres. Le principe de l'analyse en composantes principales est simple. Soit un ensemble de  $m$  observations de  $d$  variables d'entrée  $\mathbf{u}$  d'un modèle quelconque.

---

1. Calculer la matrice de covariance de taille  $(d \times d)$  de l'ensemble des observations.
2. Déterminer les valeurs propres  $\lambda_i$  et vecteurs propres  $\phi_i$  de la matrice de covariance.
3. Conserver uniquement les vecteurs propres correspondants aux valeurs propres les plus élevées telles que par exemple :

$$\sum \lambda_i^{(\text{conservées})} = 0.8 \times \sum \lambda_i^{(\text{totales})} \quad (4.3)$$

4. Transformer l'espace  $U$  des variables d'entrée dans l'espace  $\Upsilon$  réduit des vecteurs propres conservés. Ainsi, les variables  $v$  dans l'espace réduit  $\Upsilon$  sont déterminées par :

$$\mathbf{v} = \boldsymbol{\phi}^T \mathbf{u} \quad (4.4)$$

où  $\boldsymbol{\phi}$  est la matrice des vecteurs propres conservés.

Cette analyse en composantes principales prend en compte uniquement les variables d'entrée et non la sortie d'un modèle quelconque. Elle n'a donc souvent que peu d'intérêt en analyse de fiabilité mais elle est donc très efficace lorsqu'un phénomène physique est observé. Cette technique a toutefois été déjà utilisée dans la littérature dans le cadre de l'analyse de fiabilité (Aversano et Parente, 2016 ; Aversano et al., 2017).

### *Partial Least Square*

Très proche de l'analyse en composantes principales, la méthode *Partial Least Square* propose cependant de prendre en compte la sortie du modèle permettant à la fois d'observer les interactions entre les variables d'entrée elles-mêmes mais aussi d'identifier les relations entre les variables d'entrée et la sortie d'un modèle. Soit  $\mathbf{X}^{(1)}$  une matrice contenant des points centrés des variables d'entrée et  $\mathbf{y}^{(1)}$  le vecteur des réponses centrées de la fonction de performance. La recherche des composantes principales  $\mathbf{t}$  est ici itérative selon le schéma présenté ci-dessous.

1. **Recherche d'une composante principale  $\mathbf{t}^{(i)}$ .** La composante principale  $\mathbf{t}^{(i)} = \mathbf{X}^{(i)} \mathbf{w}^{(i)}$  correspond à la meilleure direction  $\mathbf{w}^{(i)}$  maximisant la covariance carrée entre la composante principale et la sortie  $\mathbf{y}^{(i)}$ .  $\mathbf{w}^{(i)}$  est le vecteur propre de la matrice  $\mathbf{X}^{(i)T} \mathbf{y}^{(i)} \mathbf{y}^{(i)T} \mathbf{X}^{(i)}$  correspondant à la plus grande valeur propre  $\lambda^{(i)}$  en valeur absolue. Soit,  $\boldsymbol{\lambda}$  le vecteur des valeurs propres déterminés à partir de l'équation suivante :

$$\det \left( \mathbf{X}^{(i)T} \mathbf{y}^{(i)} \mathbf{y}^{(i)T} \mathbf{X}^{(i)} - \boldsymbol{\lambda} \mathbf{I} \right) = 0 \quad (4.5)$$

où  $\mathbf{I}$  est la matrice identité.

Le vecteur propre  $\mathbf{w}^{(i)}$  correspondant à la valeur propre  $\lambda^{(i)}$  est obtenu en résolvant le problème suivant :

$$\left( \mathbf{X}^{(i)T} \mathbf{y}^{(i)} \mathbf{y}^{(i)T} \mathbf{X}^{(i)} - \lambda^{(i)} \mathbf{I} \right) \mathbf{w}^{(i)} = 0 \quad (4.6)$$

2. **Calcul des coefficients de régression  $\mathbf{p}^{(i)}$  de la matrice  $\mathbf{X}^{(i)}$  sur la composante principale  $\mathbf{t}^{(i)}$  :**

$$\mathbf{p}^{(i)} = \frac{\mathbf{X}^{(i)} \mathbf{t}^{(i)}}{\mathbf{t}^{(i)T} \mathbf{t}^{(i)}} \quad (4.7)$$

3. **Calcul du coefficient de régression  $c^{(i)}$  du vecteur  $\mathbf{y}^{(i)}$  sur la composante principale  $\mathbf{t}^{(i)}$  :**

$$c^{(i)} = \frac{\mathbf{y}^{(i)\top} \mathbf{t}^{(i)}}{\mathbf{t}^{(i)\top} \mathbf{t}^{(i)}} \quad (4.8)$$

4. **Calcul des résidus  $\mathbf{X}^{(i+1)}$  et  $\mathbf{y}^{(i+1)}$  issus de la régression.**

$$\mathbf{X}^{(i+1)} = \mathbf{X}^{(i)} - \mathbf{t}^{(i)} \mathbf{p}^{(i)} \quad (4.9)$$

$$\mathbf{y}^{(i+1)} = \mathbf{y}^{(i)} - c^{(i)} \mathbf{t}^{(i)} \quad (4.10)$$

La réduction de la dimension est réalisée en sélectionnant les premières composantes principales déterminées par l'algorithme. Son atout est d'identifier ces composantes à partir des variables d'entrée et de la sortie du modèle. L'inconvénient concerne le nombre de composantes principales à sélectionner. Aucun indicateur ne fournit l'importance des différentes composantes. Ainsi, le nombre de composantes à sélectionner est arbitraire. Cette méthode a déjà été couplée avec des métamodèles de krigeage par Bouhlel et al., (2016) et Bouhlel et Martins, (2018) dans le cadre de l'optimisation.

### 4.3.3 Techniques de décompositions fonctionnelles

Le principe fondamental des techniques de décompositions fonctionnelles est de proposer une représentation d'une fonction complexe en une combinaison de fonctions plus simples. La décomposition la plus connue est le développement en série de Taylor permettant de construire une approximation de la fonction complexe au voisinage d'un point à partir de cette fonction et de ses dérivées évaluées en ce point. D'autres développements du même type ont été développés pour s'adapter à des problèmes particuliers comme le développement en série de Volterra (voir par exemple Schetzen, 1980) ou en série de Laurent (voir par exemple Knopp, 1996). Une autre technique intitulée High Dimensional Model Representations (Rabitz et al., 1999; Rabitz et Aliş, 1999; Li et al., 2001) propose de décomposer la fonction complexe en une somme de fonctions définies par l'utilisateur. Cette dernière technique est plus amplement décrite dans la suite de cette section.

#### *High Dimensional Model Representations*

La technique de *High Dimensional Model Representations* (Rabitz et al., 1999; Rabitz et Aliş, 1999; Li et al., 2001) a pour objectif d'exprimer une fonction  $G(\mathbf{u})$  comme une somme de fonctions de dimensions inférieures :

$$G(\mathbf{u}) = g_0 + \sum_{i=1}^d g_i(u_i) + \sum_{1 \leq i < j \leq d} g_{ij}(u_i, u_j) + \dots + g_{12\dots d}(u_1, \dots, u_d) \quad (4.11)$$

où  $g_0$  est une constante. La fonction  $g_i(u_i)$  est la composante à l'ordre 1 de la décomposition et correspond à la seule influence de la variable  $u_i$ . La fonction  $g_{ij}(u_i, u_j)$  est la composante d'ordre 2 de  $G(\mathbf{u})$  et représente l'influence couplée des variables  $u_i$  et  $u_j$ ; et ainsi de suite pour les ordres plus élevés. Si toutes les composantes sont identifiées, la décomposition décrit exactement la vraie fonction  $G(\mathbf{u})$ . Cependant, il est souvent très

difficile et très coûteux d'identifier toutes les composantes, d'autant plus que les composantes d'ordre élevé sont souvent négligeables (Chowdhury et Rao, 2009b ; Rabitz et Aliş, 1999). Une troncature peut donc être nécessaire. Deux techniques sont principalement disponibles dans la littérature permettant l'identification des composantes de la décomposition : ANOVA-HDMR et Cut-HDMR.

La décomposition Cut-HDMR a été utilisée pour résoudre des problèmes de fiabilité en grande dimension à l'aide de métamodèles. En effet, l'utilisation d'une décomposition tronquée permet d'avoir à calibrer uniquement des métamodèles de faible dimension. Chowdhury et Rao, (2009a,b) proposent d'identifier les composantes de la décomposition par la méthode d'interpolation *Moving Least Squares*. Dans l'objectif de diminuer le nombre d'appels à la fonction de performance, Wang et al., (2011) présentent un schéma adaptatif combinant la décomposition HDMR et l'approximation par *Moving Least Squares*. D'autres ont opté pour différents métamodèles dans le but d'obtenir une meilleure précision : Shan et Wang, (2010) utilisent les *Radial Basis Functions* alors que Xiong et al., (2012) proposent d'identifier les composantes de la décomposition avec des approximations de type *Support Vector Regression*.

**ANOVA-HDMR.** La première technique d'identification des composantes de la décomposition HDMR présentée ici est la technique *ANOVA-HDMR* aussi nommée *RS-HDMR* pour *Random Sampling*. Elle est appelée ANOVA-HDMR puisqu'elle permet d'analyser et de quantifier l'importance relative des variables d'entrée et de leur combinaison. Les composantes de la décomposition sont déterminées par une suite d'intégrales sur un ensemble de points tirés aléatoirement dans tout l'espace de définition. Premièrement, l'espace des variables doit être transformé en un hyper-cube unitaire  $K^d = \{(u_1, u_2, \dots, u_d) | 0 \leq u_i \leq 1, i = 1, 2, \dots, d\}$ . Les composantes de la décomposition sont ensuite identifiées par le schéma suivant :

$$g_0 = \int_{K^d} G(\mathbf{u}) d\mathbf{u} \quad (4.12)$$

$$g_i(u_i) = \int_{K^{d-1}} G(\mathbf{u}) \prod_{k \neq i} du_k - g_0 \quad (4.13)$$

$$g_{ij}(u_i, u_j) = \int_{K^{d-2}} G(\mathbf{u}) \prod_{k \notin \{i,j\}} du_k - g_i(u_i) - g_j(u_j) - g_0 \quad (4.14)$$

Ces intégrales peuvent être calculées à partir de simulations de Monte-Carlo afin d'obtenir une bonne estimation. Cependant, l'inconvénient principal d'une telle technique est le nombre nécessaire d'évaluations de la fonction de performance. Dans le cadre de l'analyse de fiabilité, la fonction à estimer est souvent coûteuse. Il peut donc être impossible de calculer les intégrales et ainsi d'identifier les composantes de la décomposition HDMR.

**Cut-HDMR.** La technique d'identification *cut-HDMR* des composantes de la décomposition s'appuie sur une projection de l'espace sur des hyper-plans passant par un unique point dénoté point d'ancrage  $\bar{\mathbf{u}}$ . Une illustration de cette projection est donnée en Figure 4.1. Le processus d'identification des composantes est le suivant :

$$g_0 = G(\bar{\mathbf{u}}) \quad (4.15)$$

$$g_i(u_i) = G(u_i, \bar{\mathbf{u}}) - g_0 \quad (4.16)$$

$$g_{ij}(u_i, u_j) = G(u_i, u_j, \bar{\mathbf{u}}) - g_i(u_i) - g_j(u_j) - g_0 \quad (4.17)$$

$$\dots \quad (4.18)$$

où  $G(u_i, \bar{\mathbf{u}}) = G(\bar{u}_1, \bar{u}_2, \dots, u_i, \bar{u}_{i+1}, \dots, \bar{u}_d)$ .

Le nombre de fonction à identifier est égale à :

$$1 + d + \frac{d(d-1)}{2} + \dots \quad (4.19)$$

Les composantes des sous-fonctions de la décomposition *cut-HDMR* est souvent données numériquement à certaines valeurs discrètes des différentes variables  $u_j$ . La valeur de la fonction  $G(\mathbf{u})$  en un point  $\mathbf{u}$  non-observé peut être obtenu à partir seulement de plusieurs interpolations de faible dimension.

Cette méthode comparée à la précédente requiert beaucoup moins d'appels à la fonction de performance potentiellement coûteuse. Elle est donc préférable dans le cadre de l'analyse de fiabilité. Une méthode est proposée dans la suite couplant la méthode AK-MCS à la technique de *Cut-HDMR* (voir section 4.4).

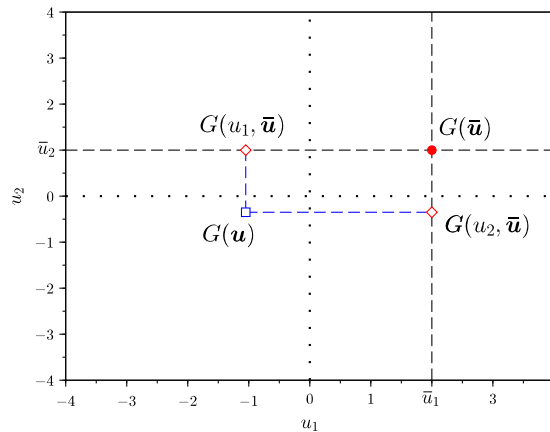


FIGURE 4.1 – Illustration de la projection de la technique *cut-HDMR*.

## 4.4 Méthodes AK-HDMR

Le principe général des méthodes proposées, intitulées AK-HDMR, est de coupler la méthode AK-MCS (voir section 2.4.1) avec la méthode de décomposition fonctionnelle *cut-HDMR* présentée précédemment. Les sous-fonctions induites par la décomposition sont

modélisées par krigeage. Ainsi, la calibration d'un métamodèle en grande dimension est remplacée par la construction de plusieurs métamodèles de krigeage de faible dimension. Deux méthodes sont présentées ici et diffèrent par l'ordre de troncature de la décomposition cut-HDMR : troncature à l'ordre 1 pour AK-HDMR1 et à l'ordre 2 pour AK-HDMR2.

#### 4.4.1 Méthode AK-HDMR1

La méthode AK-HDMR est ici tronquée au premier ordre. La procédure de la méthode proposée, nommée AK-HDMR1, est présentée ci-dessous et est résumée en Figure 4.4.

1. **Génération d'une population de Monte-Carlo contenant  $n_p$  points  $\mathbf{u}^{(k)}$ .**
2. **Sélection du point d'ancrage pour la décomposition HDMR.** Un point d'ancrage approprié doit être choisi pour obtenir un métamodèle convenable. Deux points particuliers sont souvent utilisés en analyse de fiabilité : l'origine de l'espace standard et le point de défaillance le plus probable. L'utilisation de l'origine de l'espace standard ne coûte aucun calcul préliminaire alors que la recherche du MPFP peut être coûteuse selon le problème considéré. Cependant, comme discuté précédemment, la masse de probabilité devient de plus en plus éloignée de ces deux points lorsque la dimension du problème de fiabilité augmente. À partir d'observations empiriques réalisées durant cette thèse, un point d'ancrage approprié semble devoir appartenir à la population de Monte-Carlo et d'être le plus proche possible de l'état-limite, voire de correspondre à un point défaillant (ce dernier point n'étant pas évidemment déterminable facilement). Pour cela, la technique proposée ici est la suivante :
  - (a) Tirage de  $N_s$  points de la population de Monte-Carlo.  $N_s = 20$  semble un bon compromis entre efforts numériques et efficacité de la méthode.
  - (b) Évaluation de la fonction de performance en ces points.
  - (c) Sélection du point correspondant à la plus faible valeur de la fonction de performance. Ce point est le point d'ancrage.
3. **Définition du premier plan d'expériences (DoE).** Un plan d'expériences est nécessaire pour chacun des métamodèles de Krigeage unidimensionnels. Les deux étapes suivantes sont proposées pour obtenir un DoE convenable :
  - (a) Génération par *Latin Hypercube Sampling* d'un ensemble de points dans l'espace standard.
  - (b) Ces points sont ensuite projetés sur les axes passant par le point d'ancrage. Cette projection est illustrée par la Figure 4.2.
  - (c) Évaluation de la fonction de performance sur ces points projetés.
4. **Calibration des  $d$  métamodèles de krigeage basés sur les  $d$  plans d'expériences.** Les prédictions  $\mu_{\hat{G}_i}$  et variances  $\sigma_{\hat{G}_i}^2$  sont estimées sur la population de Monte-Carlo pour chaque métamodèle.
5. **Évaluation de la réponse  $\hat{G}$  de la décomposition HDMR utilisant les métamodèles calibrés.** Elle s'écrit :

$$G(\mathbf{u}) \approx \hat{G}(\mathbf{u}) = g_0 + \sum_{i=1}^d \hat{g}_i(u_i) \quad (4.20)$$

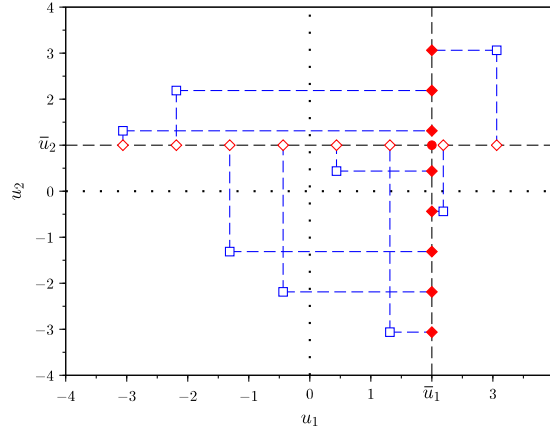


FIGURE 4.2 – Illustration de la projection du premier plan d'expériences en dimension 2 – Point rouge : point d'ancrage. Carrés bleus : points générés par LHS. Losanges rouges vides : DoE pour le métamodèle correspondant à la variable  $u_1$ . Losanges rouges pleins : DoE pour le métamodèle correspondant à la variable  $u_2$ .

avec

$$g_0 = G(\bar{\mathbf{u}}) \quad (4.21)$$

$$\hat{g}_i(u_i) = \hat{G}_i(u_i, \bar{\mathbf{u}}) - g_0 \quad (4.22)$$

où  $\hat{G}_i(u_i, \bar{\mathbf{u}})$  est le  $i^{\text{ème}}$  métamodèle de krigeage correspondant à la  $i^{\text{ème}}$  variable d'entrée. La prédiction  $\mu_{\hat{G}}$ , c'est-à-dire la moyenne globale, et la variance  $\sigma_{\hat{G}}^2$  sont calculées à partir de la propriété suivante : une combinaison linéaire de variables aléatoires gaussiennes est une variable aléatoire gaussienne. L'Équation (4.20) doit être réécrite telle que :

$$\begin{aligned} \hat{G}(\mathbf{u}) &= g_0 + \sum_{i=1}^n g_i(u_i) \\ &= g_0 + \sum_{i=1}^n [\hat{G}_i(u_i, \bar{\mathbf{u}}) - g_0] \\ &= (1 - n)g_0 + \sum_{i=1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) \end{aligned} \quad (4.23)$$

Tous les métamodèles  $\hat{G}_i(u_i, \bar{\mathbf{u}})$  sont indépendants par construction ce qui conduit aux expressions suivantes de la prédiction et de la variance de krigeage.

$$\begin{aligned} \mu_{\hat{G}}(\mathbf{u}^{(k)}) &= \mathbb{E} \left[ (1 - d)g_0 + \sum_{i=1}^d \hat{G}_i(u_i^{(k)}, \bar{\mathbf{u}}) \right] \\ &= (1 - d)g_0 + \sum_{i=1}^d \mu_{\hat{G}_i}(u_i^{(k)}, \bar{\mathbf{u}}) \end{aligned} \quad (4.24)$$



$$\begin{aligned}\sigma_{\hat{G}}^2(\mathbf{u}^{(k)}) &= \text{Var} \left[ (1-d)g_0 + \sum_{i=1}^d \hat{G}_i(u_i^{(k)}, \bar{\mathbf{u}}) \right] \\ &= \sum_{i=1}^d \sigma_{\hat{G}_i}^2(u_i^{(k)}, \bar{\mathbf{u}})\end{aligned}\tag{4.25}$$

La prédiction et la variance de krigeage de toutes les composantes  $\hat{G}_i(u_i, \bar{\mathbf{u}})$  doivent donc être calculées en tous les points  $\mathbf{u}^{(k)}$  de la population de Monte-Carlo.

6. **Estimation de la probabilité de défaillance** par l'équation suivante :

$$p_f \simeq \frac{1}{n_p} \sum_{k=1}^{n_p} \mathbb{1}_{\mu_{\hat{G}}}(\mathbf{u}^{(k)})\tag{4.26}$$

où  $\mathbb{1}_{\mu_{\hat{G}}}(\mathbf{u}^{(k)})$  est égale à 1 si la prédiction de krigeage est négative pour le point  $\mathbf{u}^{(k)}$ , et égale à 0 sinon.

7. **Apprentissage.** Il est nécessaire d'identifier quel point mais aussi quel métamodèle doit être enrichi. En effet, un point peut enrichir la totalité des métamodèles mais aux frais d'efforts numériques potentiellement inutiles. La démarche d'apprentissage des méthodes AK est donc adaptée en deux étapes présentées ci-dessous puis illustrée en Figure 4.3.

(a) Identification du point de la population de Monte-Carlo à enrichir. La fonction d'apprentissage  $U$  est utilisée avec la prédiction et la variance définies précédemment (Équations (4.24) et (4.25)). Le point à enrichir est donc :

$$\mathbf{u}^* = \underset{\mathbf{u}^{(k)}}{\text{Arg min}} U(\mathbf{u}^{(k)}), \quad k = 1, \dots, n_p\tag{4.27}$$

où

$$U(\mathbf{u}^{(k)}) = \frac{(1-d)g_0 + \sum_{i=1}^d \mu_{\hat{G}_i}(u_i^{(k)}, \bar{\mathbf{u}})}{\sqrt{\sum_{i=1}^d \sigma_{\hat{G}_i}^2(u_i^{(k)}, \bar{\mathbf{u}})}}\tag{4.28}$$

(b) Identification du métamodèle unidimensionnel à enrichir. Il correspond à celui possédant la variance la plus grande au point  $\mathbf{u}^*$  :

$$i^* = \underset{i}{\text{Arg max}} \sigma_{\hat{G}_i}^2(u_i^*, \bar{\mathbf{u}}), \quad i = 1, \dots, d\tag{4.29}$$

8. **Arrêt de l'apprentissage.** Le critère d'arrêt sur l'apprentissage est celui présenté en Chapitre 3 pour la méthode AK-MCSi en Équation (3.15) :

$$\min \{ \mathcal{P}[C]_{ind}, \mathcal{P}[C]_{cor} \} \geq P_{cible}\tag{4.30}$$

Si ce critère est respecté, l'apprentissage est terminé (Étape 10). Sinon, la méthode continue avec l'étape 9.

9. **Mise à jour du plan d'expériences et calibration du métamodèle de krigeage.** Seul le DoE correspondant au métamodèle sélectionné pendant l'étape 7 est enrichi. La fonction de performance est donc évaluée pour ce point projeté. L'algorithme retourne à l'étape 5.

10. **Fin de la méthode.** Le résultat de la méthode AK-HDMR1 correspond à la dernière estimation de la probabilité de défaillance.

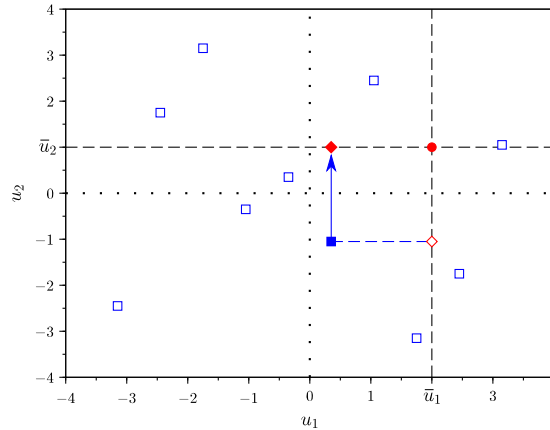


FIGURE 4.3 – Illustration du schéma d’enrichissement de AK-HDMR1 – Point rouge : Point d’ancrage. Carrés bleus : points de la population de Monte-Carlo. Carré bleu plein : point identifié par la fonction d’apprentissage. Losange rouge vide : projection du point  $\mathbf{u}^*$  mais non sélectionné pour l’enrichissement. Losange rouge plein : projection du point  $\mathbf{u}^*$  qui maximise la variance  $\sigma_{G_i}^2(u_i^*, \bar{\mathbf{u}})$ .

#### 4.4.2 Méthode AK-HDMR2

La méthode AK-HDMR est cette fois tronquée à l’ordre 2 et prend donc en compte les interactions deux à deux des variables d’entrée du problème de fiabilité. La procédure de cette méthode, intitulée AK-HDMR2, est globalement identique à celle de AK-HDMR1. Cependant un métamodèle de krigeage doit être calibré pour chacune des variables d’entrée ainsi que pour chacun des couples de variables d’entrée. Le nombre de métamodèles à calibrer s’élève donc à  $\frac{d(d+1)}{2}$ . La construction des plans d’expériences est identique concernant la considération des variables seules. Pour le cas des couples de variables, les points du premier plan d’expériences générés par *Latin Hypercube Sampling* (voir Étape 3 de la méthode AK-HDMR1) sont projetés sur les hyper-plans passant par le point d’ancrage. Cette projection est illustrée en Figure 4.5.

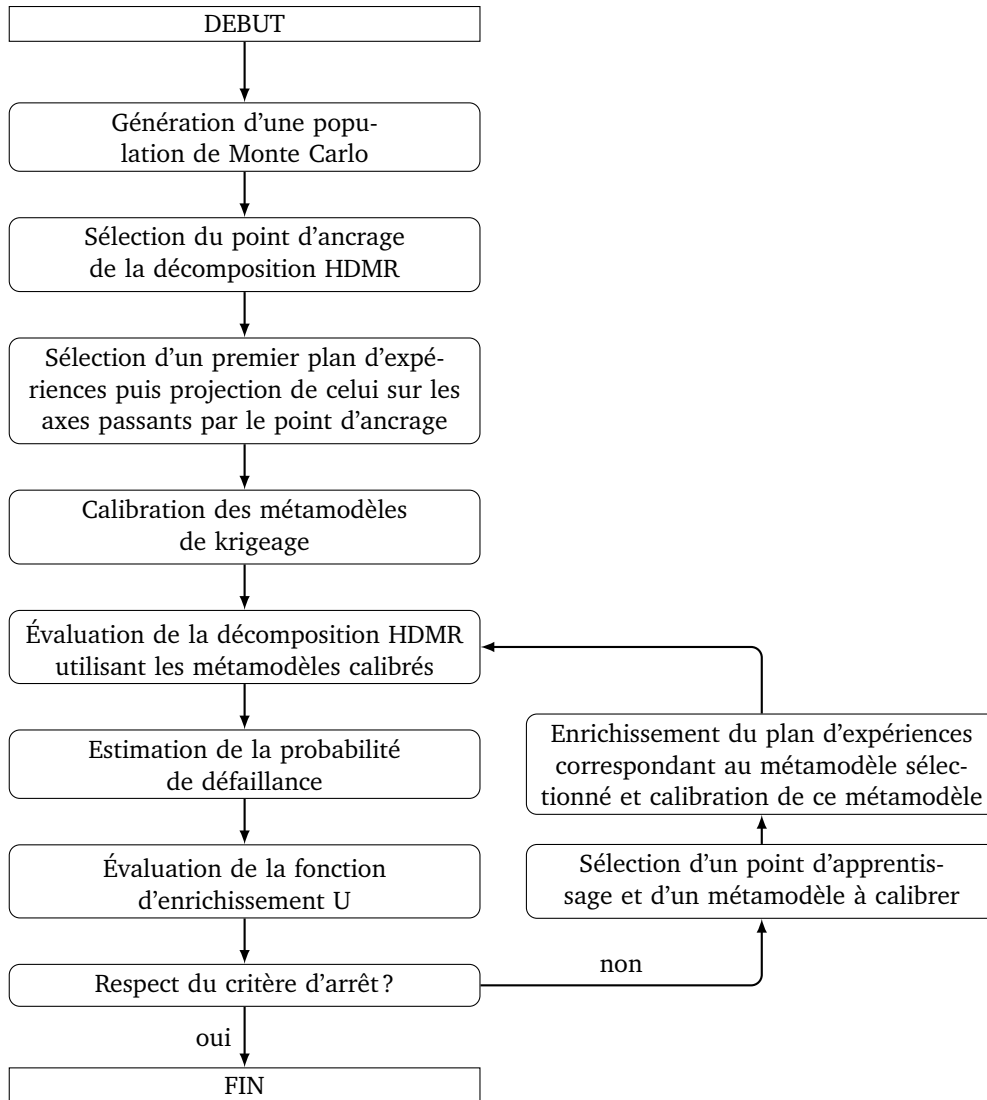


FIGURE 4.4 – Procédure de AK-HDMR1.

Les calculs de la prédiction et de la variance de krigeage (Étape 5 de la méthode AK-HDMR1) doivent aussi être adaptés à la décomposition HDMR à l'ordre 2 utilisant ces métamodèles de krigeage. Cette décomposition s'écrit :

$$G(\mathbf{u}) \approx \hat{G}(\mathbf{u}) = g_0 + \sum_{i=1}^d \hat{g}_i(u_i) \quad (4.31)$$

avec

$$g_0 = G(\bar{\mathbf{u}}) \quad (4.32)$$

$$\hat{g}_i(u_i) = \hat{G}_i(u_i, \bar{\mathbf{u}}) - g_0 \quad (4.33)$$

$$\hat{g}_{ij}(u_i, u_j) = \hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}}) - \hat{g}_i(u_i) - \hat{g}_j(u_j) - g_0 \quad (4.34)$$

où  $\hat{G}_i(u_i, \bar{\mathbf{u}})$  est le  $i^{\text{ème}}$  métamodèle de krigeage correspondant à la  $i^{\text{ème}}$  variable d'entrée et  $\hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}})$  est le métamodèle de krigeage correspondant au couple de variables

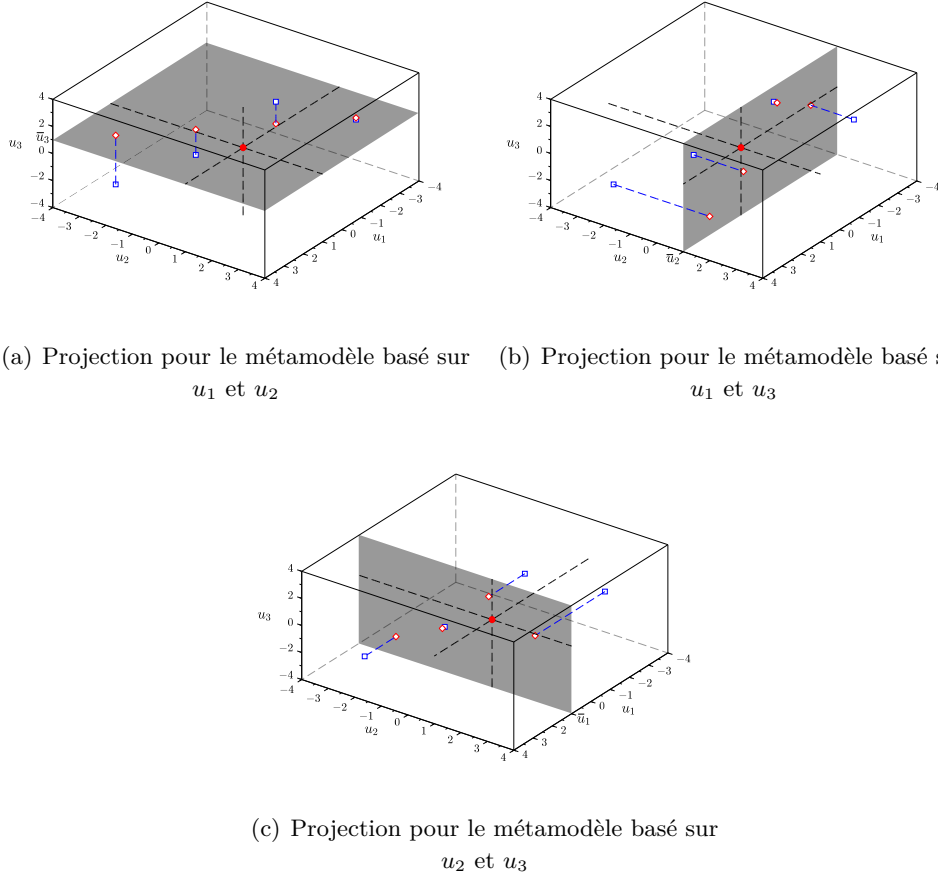


FIGURE 4.5 – Projection du premier plan d'expériences de AK-HDMR2 en dimension 3.

d'entrée  $\{i, j\}$ . Pour permettre le calcul de la prédiction et de la variance de krigeage, l'Équation (4.31) doit être reformulée. Il a été montré (voir démonstration en Annexe D) qu'elle peut s'écrire :

$$\hat{G}(\mathbf{u}) = \frac{2 + n(n-3)}{2}g_0 + (2-n) \sum_{i=1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) + \sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}}) \quad (4.35)$$

Les expressions de la prédiction  $\mu_{\hat{G}}$  et la variance  $\sigma_{\hat{G}}^2$  de krigeage sont donc données ci-dessous en considérant les deux propriétés suivantes : une combinaison linéaire de variables gaussiennes indépendantes est une variable gaussienne ; tous les métamodèles  $\hat{G}_i(u_i, \bar{\mathbf{u}})$  et  $\hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}})$  sont indépendants par construction.

$$\begin{aligned} \mu_{\hat{G}}(\mathbf{u}^{(k)}) &= \mathbb{E} [\hat{G}(\mathbf{u})] \\ &= \frac{2 + n(n-3)}{2}g_0 + (2-n) \sum_{i=1}^n \mu_{\hat{G}_i}(u_i^{(k)}, \bar{\mathbf{u}}) + \sum_{i=1}^{n-1} \sum_{j=i+1}^n \mu_{\hat{G}_{ij}}(u_i^{(k)}, u_j^{(k)}, \bar{\mathbf{u}}) \end{aligned} \quad (4.36)$$

$$\begin{aligned}
\sigma_{\hat{G}}^2(\mathbf{u}^{(k)}) &= \text{Var} \left[ \hat{G}(\mathbf{u}) \right] \\
&= (2 - n) \sum_{i=1}^n \sigma_{\hat{G}_i}^2(u_i^{(k)}, \bar{\mathbf{u}}) + \sum_{i=1}^{n-1} \sum_{j=i+1}^n \sigma_{\hat{G}_{ij}}^2(u_i^{(k)}, u_j^{(k)}, \bar{\mathbf{u}})
\end{aligned} \tag{4.37}$$

La prédiction et la variance de krigeage de toutes les composantes  $\hat{G}_i(u_i, \bar{\mathbf{u}})$  et  $\hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}})$  doivent donc être calculées en tous les points  $\mathbf{u}^{(k)}$  de la population de Monte-Carlo.

Les autres étapes de la méthode AK-HDMR1 restent inchangées. Un point très important a tout de même été soulevé concernant le nombre de métamodèles de krigeage à calibrer devenant très grand lorsque la dimension du problème augmente. Pour rappel,  $\frac{d(d+1)}{2}$  métamodèles doivent être construits dans la procédure de la méthode AK-HDMR2, par exemple si  $d = 100$ , 4950 métamodèles doivent être construits. Certaines méthodes (*forward selection methods*) permettant de diminuer le nombre de couples de variables ont été investiguées comme les algorithmes Lasso, *Forward Stagewise linear regression* ou *Least Angle Regression* (Efron et al., 2004). Le principe général de ces méthodes est de construire un modèle linéaire correct de la fonction de performance à partir d'échantillons des variables d'entrée. Ce modèle permet ensuite de sélectionner un ensemble de variables, ou de couples de variables, influant majoritairement sur la réponse de la fonction de performance. Les expériences ont montré que pour obtenir un bon modèle, cela nécessite un nombre de points assez élevé ce qui représente une difficulté majeure pour le couplage avec AK-HDMR2. En effet, ce plan d'expériences doit être évalué par la fonction de performance coûteuse alors qu'il ne pourra pas être utilisé dans la suite de la méthode. Ces évaluations seront donc perdues. En effet, la méthode AK-HDMR2 requiert une projection avant évaluation de la fonction de performance. Ainsi, le nombre d'appels à la fonction de performance augmenterait rapidement. La méthode AK-HDMR2 n'est plus discuté dans la suite de ce document et est une perspective de ces travaux de thèse.

#### 4.4.3 Validation et étude des paramètres de la méthode AK-HDMR1

La validation des méthodes AK-HDMR est réalisée à partir d'un exemple de fiabilité basée sur la fonction *Oblate Spheroid* modifiée (Waarts, 2000). La fonction de performance est la suivante :

$$g(\mathbf{x}) = R - \sum_{i=1}^d \frac{x_i^2}{1 + i/10} - \frac{p}{d} \times \sum_{i=1}^{d-1} \sum_{j=i+1}^d \frac{x_i x_j}{1 + ij} \tag{4.38}$$

où  $R$  est une constante permettant d'ajuster la probabilité, ici  $P_f \sim 10^{-3}$ , et  $p$  est un paramètre contrôlant le poids des interactions. Toutes les variables  $x_i$  suivent des lois gaussiennes indépendantes de moyennes nulles et de variances unitaires. Différents paramètres de la méthode sont observés ici : la dimension, le poids des interactions et le choix du point d'ancrage.

Tout d'abord, aucune interaction n'est prise en compte, c'est-à-dire  $p = 0$ . Le point d'ancrage est sélectionné selon la démarche proposée en section 4.4.1. Seule l'efficacité de la méthode AK-HDMR1 est observée en fonction de la dimension du problème. La méthode AK-MCS n'est pas applicable pour cet exemple en raison de la dimension élevée du problème. Les résultats sont donnés au Tableau 4.1. Les résultats des analyses FORM et SORM sont conservés ici pour montrer leur inefficacité en grande dimension. Dans la suite, FORM et SORM ne sont plus appliqués puisqu'ils sont inefficaces. Les résultats obtenus à

partir de AK-HDMR1 montrent son efficacité même lorsque la dimension devient vraiment très grande ( $d = 1000$  pour le cas 6). Le nombre d'appels augmente de manière linéaire avec la dimension et reste acceptable compte tenu de la dimension. Ce premier problème ne contient aucune interaction ce qui permet à la méthode AK-HDMR1 d'être efficace puisque la décomposition HDMR1 est exacte.

	$d$	$R$	$p$	Méthode	Nombre d'appels à la fonction de performance	$\hat{p}_f$	$\delta_{\hat{p}_f}(\%)$
Cas 1	20	20	0	MCS	$10^5$	$1.404 \times 10^{-2}$	2.65%
				AK-HDMR1	121	$1.404 \times 10^{-2}$	2.65%
				FORM	252	$1.16 \times 10^{-6}$	-
				SORM	714	$3.06 \times 10^{-5}$	-
Cas 2	50	30	0	MCS	$10^5$	$4.59 \times 10^{-3}$	4.65%
				AK-HDMR1	357	$4.59 \times 10^{-3}$	4.65%
				FORM	612	$3.64 \times 10^{-9}$	-
				SORM	3264	$3.26 \times 10^{-52}$	-
Cas 3	100	35	0	MCS	$10^5$	$8.75 \times 10^{-3}$	3.37%
				AK-HDMR1	600	$8.75 \times 10^{-3}$	3.37%
				FORM	1212	$2.08 \times 10^{-10}$	-
				SORM	11514	-	-
Cas 4	200	42	0	MCS	$10^5$	$7.45 \times 10^{-3}$	3.65%
				AK-HDMR1	1200	$7.45 \times 10^{-3}$	3.65%
				FORM	2613	$6.08 \times 10^{-12}$	-
				SORM	43215	-	-
Cas 5	500	52	0	MCS	$10^5$	$4.80 \times 10^{-3}$	4.55%
				AK-HDMR1	3000	$4.80 \times 10^{-3}$	4.55%
				FORM	6513	$1.41 \times 10^{-14}$	-
				SORM	258015	0	-
Cas 6	1000	58	0	MCS	$10^5$	$7.06 \times 10^{-3}$	3.75%
				AK-HDMR1	6000	$7.06 \times 10^{-3}$	3.75%
				FORM	14014	$5.08 \times 10^{-16}$	-
				SORM	1017016	0	-

TABLE 4.1 – Fonction *oblate spheroid* – Résultat de validation de AK-HDMR1. – Étude de l'influence de la dimension sur l'estimation de la probabilité de défaillance.

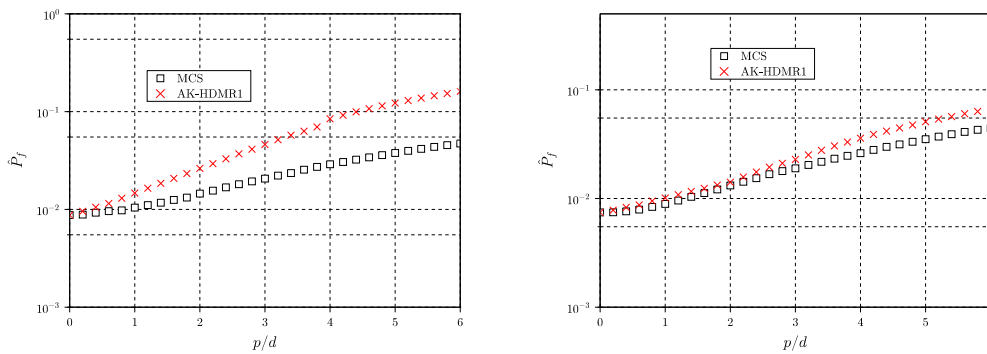
Dans la suite, les interactions sont prises en compte, c'est-à-dire  $p \neq 0$ . La première analyse concerne l'effet de la dimension sur l'efficacité de la méthode AK-HDMR1. Les résultats de cette étude sont présentés en Tableau 4.2. AK-HDMR1 fournit des estimations de la probabilité de défaillance proches de la référence obtenue par simulations de Monte-Carlo. Ces estimations ne sont pas exactes puisque les interactions induites par la fonction de performance ne peuvent pas être prises en compte dans la méthode proposée.

La seconde analyse concerne l'influence du poids des interactions sur l'efficacité de la méthode AK-HDMR1. Les cas 3 et 4 sont étudiés avec un ratio  $p/d$  qui varie de 0 à 6. Le poids des interactions augmente avec le ratio. La Figure 4.6 montre les résultats obtenus lors de cette étude. Lorsque les interactions sont faibles, les résultats de AK-HDMR1 sont

	$d$	$R$	$p$	Méthode	Nombre d'appels à la fonction de performance	$\hat{p}_f$	$\delta_{\hat{p}_f}(\%)$
Cas 1	20	20	10	MCS	$10^5$	$1.457 \times 10^{-2}$	2.60%
				AK-HDMR1	129	$2.12 \times 10^{-2}$	2.15%
Cas 2	50	30	25	MCS	$10^5$	$4.78 \times 10^{-3}$	4.57%
				AK-HDMR1	300	$4.28 \times 10^{-3}$	4.82%
Cas 3	100	35	50	MCS	$10^5$	$9.44 \times 10^{-3}$	3.24%
				AK-HDMR1	600	$11.1 \times 10^{-3}$	2.98%
Cas 4	200	42	100	MCS	$10^5$	$7.77 \times 10^{-3}$	3.57%
				AK-HDMR1	1200	$8.49 \times 10^{-3}$	3.42%
Cas 5	500	52	250	MCS	$10^5$	$5.20 \times 10^{-3}$	4.37%
				AK-HDMR1	3001	$11.2 \times 10^{-3}$	2.97%
Cas 6	1000	58	500	MCS	$10^5$	$7.58 \times 10^{-3}$	3.75%
				AK-HDMR1	6000	$7.97 \times 10^{-3}$	3.53%

TABLE 4.2 – Fonction *oblate spheroid* – Résultat de validation de AK-HDMR1 avec interactions. – Étude de l'influence de la dimension sur l'estimation de la probabilité de défaillance.

satisfaisants. Cependant, lorsque les interactions deviennent responsables d'une partie importante de la probabilité de défaillance, les écarts dus à la méta-modélisation deviennent non-négligeables. AK-HDMR1 semble donc efficace pour l'estimation de la probabilité de défaillance même si le problème présente de faibles interactions avec un nombre acceptable d'appels à la fonction de performance.



(a) Cas 4 –  $d = 100$  –  $R = 35$  – environ 600 appels à la fonction de performance

(b) Cas 4 –  $d = 200$  –  $R = 42$  – environ 1200 appels à la fonction de performance

FIGURE 4.6 – Fonction *oblate spheroid* – Étude de l'influence du poids des interactions sur l'efficacité de AK-HDMR1.

Enfin, l'impact de la sélection du point d'ancrage de la décomposition HDMR est étudié en fixant les différents paramètres de la fonction tels que :  $d = 100$ ,  $R = 5$  et  $p/d = 1$ . Une population de Monte-Carlo  $\mathbb{P}_{MCS}$  contenant  $10^5$  points est tout d'abord générée puis une population  $\mathbb{P}_{sub}$  de  $10^4$  points est tirée aléatoirement dans cette population. La méthode AK-HDMR1 classe ensuite la population  $\mathbb{P}_{MCS}$  avec les différents points de

la population  $\mathbb{P}_{sub}$  comme point d’ancrage de la décomposition. La population  $\mathbb{P}_{MCS}$  a été complètement évaluée par la fonction de performance pour juger la classification réalisée par AK-HDMR1. Les résultats de cette étude sont donnés en Figure 4.7. Cette figure montre le ratio de la probabilité estimée par AK-HDMR1 sur la probabilité de référence fournie par les simulations de Monte-Carlo. Les abscisses correspondent à la valeur de la fonction de performance au point d’ancrage. L’origine de l’espace standard et le point de défaillance le plus probable ont aussi été utilisés comme point d’ancrage. La figure montre la dépendance de l’efficacité de la méthode AK-HDMR1 avec la sélection du point d’ancrage. Ici, l’origine de l’espace standard permet d’obtenir une bonne estimation de la probabilité de défaillance, mais cette conclusion ne peut pas être généralisée à d’autres exemples. Au vue de la Figure 4.7, il semble moins risqué de sélectionner un point d’ancrage avec une faible valeur de fonction de performance, mais ce résultat mériterait d’être confirmé.

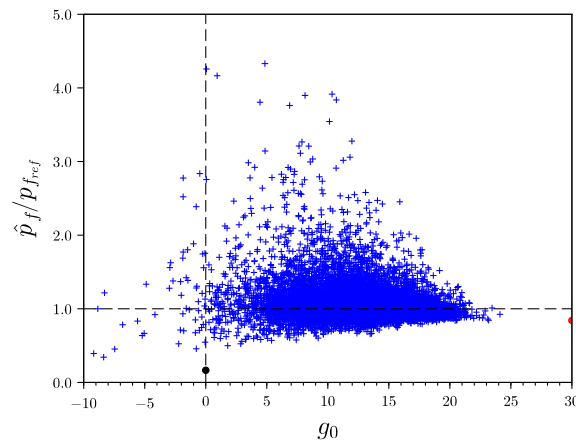


FIGURE 4.7 – Fonction *oblate spheroid* – Étude de l’influence de la sélection du point d’ancrage sur l’efficacité de AK-HDMR1 – Point rouge : utilisation de l’origine de l’espace standard comme point d’ancrage. Point noir : utilisation du MPFP comme point d’ancrage. Croix bleues : utilisation des points de Monte-Carlo comme point d’ancrage.

À partir de ce premier exemple, les différentes études menées montrent que même si les interactions réduisent l’efficacité de la méthode AK-HDMR1, cette dernière semble fournir une probabilité de défaillance proche de celle de référence avec un nombre raisonnable d’appels à la fonction de performance. La dernière étude montre que le choix du point d’ancrage est crucial pour l’estimation de la probabilité de défaillance. Il semble donc moins risqué pour l’utilisateur de sélectionner un point d’ancrage dans la population de Monte-Carlo à classer et proche de l’état-limite, c’est-à-dire avec une valeur de fonction de performance proche de zéro. Cette analyse mériterait cependant d’être confirmée à partir d’autres exemples ou en augmentant le nombre de points de la population  $\mathbb{P}_{sub}$ .



## 4.5 Méthode AK-PCA

La seconde méthode proposée dans le cadre de ces travaux de thèse pour traiter des problèmes de fiabilité en grande dimension se base sur la réduction de la dimension. Ici, le principe n'est pas de réduire la dimension des variables d'entrée du problème mais la dimension de l'espace des hyper-paramètres. Pour cela, les deux techniques présentées en section 4.3.2 peuvent être utilisées. La technique de *Partial Least Square* ne proposant pas d'indice permettant de sélectionner un nombre approprié de composantes, ce dernier doit donc être choisi arbitrairement par l'utilisateur. Cette technique est donc ici écartée. L'analyse en composantes principales (PCA) sert donc ici à identifier les composantes principales dans l'espace des hyper-paramètres.

### 4.5.1 Présentation de la méthode AK-PCA

La procédure de la méthode AK-PCA est la suivante.

1. **Génération d'une population de Monte-Carlo contenant  $n_p$  points  $\mathbf{u}^{(k)}$ .**  
Cette population est générée selon la densité de probabilité jointe des variables aléatoires puis transposée dans l'espace standard par une transformation isoprobabiliste.
2. **Définition du premier plan d'expériences (DoE).** L'utilisation d'un LHS est ici conseillée.
3. **Calibration du métamodèle de Krigeage basé sur le plan d'expériences.**  
L'analyse en composantes principales est ici utilisée pour permettre d'identifier les hyper-paramètres  $\boldsymbol{\theta}$  de krigeage selon la démarche suivante :
  - (a) **Détermination de la fonction de vraisemblance** à partir du premier plan d'expériences.
  - (b) **Tirage uniforme aléatoire de  $n_e$  réalisations du vecteur  $\boldsymbol{\theta}$  dans l'espace des hyper-paramètres.**
  - (c) **Sélection de  $n_{pca}$  points parmi les  $n_e$  tirés aléatoirement.** La fonction de vraisemblance est tout d'abord évaluée sur l'ensemble des  $n_e$  points. Un tirage aléatoire pondéré par la fonction de vraisemblance est ensuite réalisé pour sélectionner  $n_{pca}$  points parmi les  $n_e$  générés initialement. Plus la valeur de vraisemblance est grande en un point, plus le poids qui lui est affecté est grand et plus la probabilité de sélectionner ce point est grande.
  - (d) **Une analyse en composantes principales est réalisée à partir des  $n_{pca}$  points.**
  - (e) **Réduction de l'espace des hyper-paramètres.** Seules les composantes principales correspondantes aux valeurs propres les plus grandes sont conservées. Il semble approprié de conserver les composantes principales telles que la somme des valeurs propres conservées représente  $Pct_{pca} = 80\%$  de la somme totale.
  - (f) **Identification des hyper-paramètres.** L'optimisation est réalisée dans l'espace réduit par l'algorithme CMA-ES puis le résultat est transféré dans l'espace primaire permettant d'aboutir aux hyper-paramètres utilisés pour le métamodèle. Il n'y a aucune difficulté liée à la transformation inverse. Ce métamodèle de krigeage est ainsi calibré.

4. **Estimation de la probabilité de défaillance** par l'équation suivante :

$$p_f \simeq \frac{1}{n_p} \sum_{k=1}^{n_p} \mathbb{1}_{\mu_{\hat{G}}}(\mathbf{u}^{(k)}) \quad (4.39)$$

où  $\mathbb{1}_{\mu_{\hat{G}}}(\mathbf{u}^{(k)})$  est égale à 1 si la prédiction de krigeage est négative pour le point  $\mathbf{u}^{(k)}$ , et égale à 0 sinon.

5. **Apprentissage.** L'apprentissage classique des méthodes AK est ici utilisé. La fonction d'apprentissage  $U$  est donc évaluée pour toute la population de Monte-Carlo.

6. **Arrêt de l'apprentissage.** Le critère d'arrêt sur l'apprentissage est celui présenté en Chapitre 3 pour la méthode AK-MCSi en Équation (3.15) :

$$\min \{ \mathcal{P}[C]_{ind}, \mathcal{P}[C]_{cor} \} \geq P_{cible} \quad (4.40)$$

Si ce critère est respecté, l'apprentissage est terminé (Étape 8). Sinon, la méthode continue avec l'étape 7.

7. **Mise à jour du plan d'expériences.** Le point à enrichir est choisi tel qu'il possède la valeur de fonction d'apprentissage  $U$  la plus faible :

$$\mathbf{u}^* = \underset{\mathbf{u}^{(k)}}{\text{Arg min}} U(\mathbf{u}^{(k)}), \quad k = 1, \dots, n_p \quad (4.41)$$

La fonction de performance  $G$  est ensuite évaluée en ce point. La méthode retourne ensuite à l'étape 3.

8. **Fin de la méthode.** Le résultat de la méthode AK-PCA est obtenu par la dernière estimation de la probabilité de défaillance.

L'utilisation de l'Analyse en Composantes Principales permet donc de diminuer la dimension du problème d'optimisation des hyper-paramètres et ainsi de pouvoir calibrer le métamodèle de krigeage.

#### 4.5.2 Validation et étude des paramètres de la méthode AK-PCA

La validation de la méthode AK-PCA est basée sur l'exemple de la fiabilité d'une structure en treillis (section 3.4.3) pour laquelle toutes les barres sont maintenant indépendantes. Cet exemple, illustré en Figure 4.8, est donc caractérisé par cinquante-deux variables aléatoires : onze sections et modules d'Young des barres horizontales ( $A_i^{(h)}, E_i^{(h)}$ ,  $i = 1, \dots, 11$ ), douze sections et modules d'Young des barres obliques ( $A_i^{(o)}, E_i^{(o)}$ ,  $i = 1, \dots, 12$ ) et six charges appliquées à la structure ( $P_i$ ,  $i = 1, \dots, 6$ ). Leurs propriétés sont décrites par le Tableau 4.3. La défaillance est régie par la flèche  $V_1$  : si elle dépasse un seuil  $V_{1_{seuil}}$  alors la structure est défaillante.

La première étude porte sur le nombre de composantes sélectionnées par le PCA avant l'optimisation des hyper-paramètres. Ce nombre est défini par le paramètre  $Pct_{pca}$  qui caractérise le pourcentage de valeurs propres conservées sur la somme totale de ces valeurs propres. Les résultats de cette étude sont donnés dans le Tableau 4.4, la probabilité de référence a été estimée par simulations de Monte-Carlo et est égale à  $8.7 \times 10^{-4}$ . Pour cet exemple, l'estimation de la probabilité de défaillance est correcte dans chacun des cas. Cependant, lorsque  $Pct_{pca}$  est trop faible le temps à convergence de la méthode augmente,

c'est-à-dire le nombre d'appels à la fonction de performance augmente fortement. C'est pourquoi,  $Pct_{pca} = 80\%$  semble être un bon compromis entre efficacité de l'estimation, nombre d'appels à la fonction de performance et temps nécessaire à la convergence.

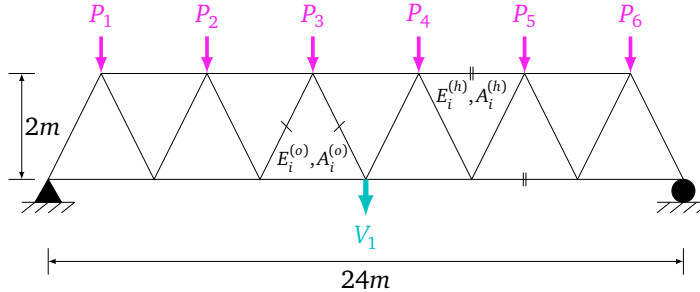


FIGURE 4.8 – Structure en treillis – Schéma du problème.

Variable	Distribution	Moyenne	Écart-type
$E_1^{(h)}-E_{11}^{(h)}$ (Pa)	Log-normale	$2.1 \times 10^{11}$	$2.1 \times 10^{10}$
$A_1^{(h)}-A_{11}^{(h)}$ (m <sup>2</sup> )	Log-normale	$2.0 \times 10^{-3}$	$2.0 \times 10^{-4}$
$E_1^{(o)}-E_{12}^{(o)}$ (Pa)	Log-normale	$2.1 \times 10^{11}$	$2.1 \times 10^{10}$
$A_1^{(o)}-A_{12}^{(o)}$ (m <sup>2</sup> )	Log-normale	$1.0 \times 10^{-3}$	$1.0 \times 10^{-4}$
$P_1-P_6$ (N)	Gumbel	$5.0 \times 10^4$	$7.5 \times 10^3$

TABLE 4.3 – Structure en treillis – Variables aléatoires du problème.

$Pct_{pca}(\%)$	Nombre d'appels à la fonction de performance	$P_f$	Temps CPU (s)
50	206	$7.3 \times 10^{-4}$	9390
60	103	$7.1 \times 10^{-4}$	9683
70	102	$7.0 \times 10^{-4}$	9853
80	109	$7.2 \times 10^{-4}$	10756
90	101	$8.3 \times 10^{-4}$	10930

TABLE 4.4 – Structure en treillis – Étude du nombre de composantes principales conservées (pourcentage des valeurs propres) –  $V_{1_{seuil}} = 105$  mm.

La seconde étude concerne l'efficacité de la méthode AK-PCA par rapport à la probabilité de défaillance à estimer. Les résultats de cette étude sont donnés au Tableau 4.5. Ils montrent que AK-PCA est efficace dans la limite approximative de  $P_f \sim 10^{-4}$ . Lorsque celle-ci devient plus petite, plusieurs solutions peuvent se présenter : soit augmenter le pourcentage de composantes principales conservées mais le temps de convergence va augmenter, soit coupler la méthode AK-PCA avec des méthodes plus efficaces pour estimer de faibles probabilités de défaillance comme AK-MCSi par exemple.

Cet exemple montre l'aptitude de AK-PCA à estimer des probabilités de défaillance liées à des problèmes en grande dimension. Le paramètre ajustable de la méthode peut être fixé à  $Pct_{pca} = 80\%$  pour obtenir un bon compromis. L'estimation de faibles probabilités de défaillance en grande dimension avec AK-PCA nécessite des travaux supplémentaires.

$V_{1_{pca}}$ (mm)	Méthode	Nombre d'appels à la fonction de performance	$P_f$
105	MCS	$10^5$	$1.0 \times 10^{-3}$
	AK-PCA	231	$9.7 \times 10^{-4}$
111	MCS	$5 \times 10^5$	$1.74 \times 10^{-4}$
	AK-PCA	215	$1.62 \times 10^{-4}$
120	MCS	$2 \times 10^6$	$1.0 \times 10^{-5}$
	AK-PCA	92	$3.5 \times 10^{-6}$

TABLE 4.5 – Structure en treillis – Étude de l'efficacité de AK-PCA en fonction de la probabilité à estimer –  $Pct_{pca} = 80\%$ .

## 4.6 Comparaison des méthodes proposées

Les deux méthodes proposées pour traiter des problèmes de fiabilité en grande dimension, AK-HDMR1 et AK-PCA, sont comparées sur deux applications « défis » pour l'analyse de fiabilité. La première permet d'observer l'efficacité des deux méthodes pour estimer une probabilité de défaillance d'un problème en grande dimension. Le second exemple illustre les limitations actuelles des deux méthodes proposées.

### 4.6.1 Fonction de Valdebenito

Le premier exemple se base sur la fonction de Valdebenito (Valdebenito et Schuëller, 2010) définie par :

$$g(\mathbf{x}) = \beta - x_1 + \left(1 - \frac{1}{1 + C \times \sum_{i=2}^d x_i^2}\right) \quad (4.42)$$

où  $\beta$  est un paramètre permettant d'ajuster la probabilité de défaillance, ici  $\beta = 2$ .  $C$  est un paramètre qui fixe l'influence de toutes les variables excepté la première d'entre-elles. Toutes les variables sont gaussiennes, indépendantes, de moyenne nulle et de variance unitaire. Seule l'influence de la dimension est observée ici. Les résultats sont donnés en Figure 4.9 pour les dimensions 50 à 100, et dans le Tableau 4.6 au delà. Ces résultats montrent que jusqu'à la dimension 100, les deux méthodes estiment correctement la probabilité de défaillance. La méthode AK-PCA requiert moins d'appels à la fonction de performance que AK-HDMR1. En effet, AK-HDMR1 nécessite la projection du premier plan d'expériences sur toutes les dimensions du problème ce qui implique un nombre plus grand d'appels préliminaires à la fonction de performance que AK-PCA. Cependant, lorsque la dimension devient plus grande, l'optimisation inhérente à AK-PCA dans l'espace réduit des hyperparamètres ne parvient plus à converger. Le métamodèle de krigeage ne peut donc pas être calibré et la méthode s'arrête prématurément. AK-HDMR1 estime toujours aussi bien la probabilité de défaillance pour un nombre d'appels à la fonction de performance tout à fait acceptable étant donnée la dimension.

Cet exemple montre l'efficacité de la méthode AK-HDMR1 pour des problèmes en grande et très grande dimension. Au contraire, AK-PCA ne parvient pas à converger lorsque la dimension est trop grande. Cependant, AK-PCA surpasse AK-HDMR1 en nombre d'appels à la fonction de performance si la dimension n'est pas trop élevée.

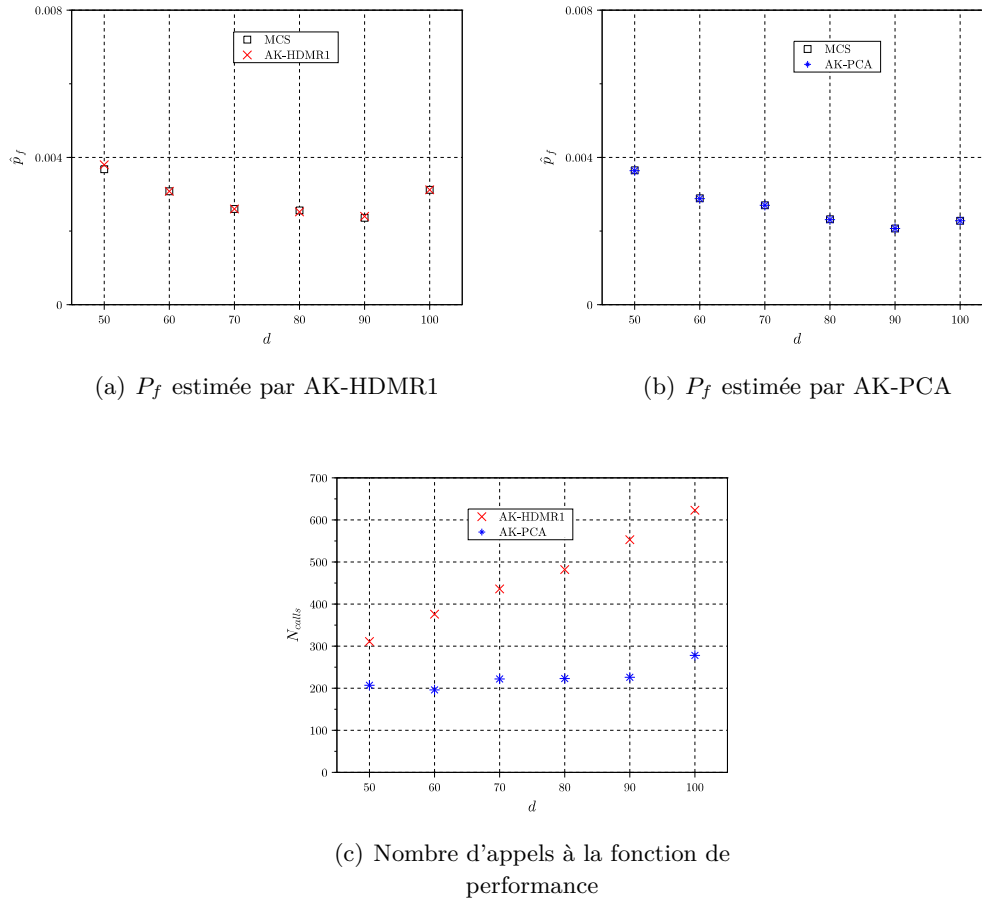


FIGURE 4.9 – Fonction de Valdebenito – Comparaison de AK-HDMR1 et AK-PCA.

Dimension	Méthode	Nombre d'appels à la fonction de performance	$\hat{p}_f$	$\delta_{\hat{p}_f}$ (%)
200	MCS	$10^5$	$1.86 \times 10^{-3}$	7.32%
	AK-HDMR1	1219	$1.86 \times 10^{-3}$	7.32%
	AK-PCA*	-	-	-
500	MCS	$10^5$	$1.59 \times 10^{-3}$	7.92%
	AK-HDMR1	3015	$1.58 \times 10^{-3}$	7.95%
	AK-PCA*	-	-	-
1000	MCS	$10^5$	$1.44 \times 10^{-3}$	8.33%
	AK-HDMR1	5867	$1.41 \times 10^{-3}$	8.42%
	AK-PCA*	-	-	-

TABLE 4.6 – Fonction de Valdebenito – Comparaison de AK-HDMR1 et AK-PCA – Très grande dimension. – \*non convergence de l'optimisation des hyper-paramètres.

### 4.6.2 Fonction modifiée de Griewank – Limitations des méthodes proposées

La seconde application présentée ici est basée sur la fonction de Griewank utilisée pour tester les algorithmes d'optimisation (Yang, 2010). Cette fonction a été adaptée dans ce document afin de devenir un problème de fiabilité. Cette fonction est définie par :

$$G(\mathbf{x}) = \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) - \sum_{i=1}^d \frac{x_i^2}{4000} + h \quad (4.43)$$

Les variables aléatoires sont toutes distribuées selon une loi gaussienne de moyenne nulle et de variance unitaire.

Ce problème de fiabilité est un challenge pour lequel peu, voire aucune, méthode de la littérature ne semble permettre d'estimer précisément la probabilité de défaillance. Les deux méthodes proposées dans ces travaux de thèse montrent leur limitation sur cet exemple. Les résultats sont présentés dans le Tableau 4.7. Ces résultats montrent tout d'abord que la méthode AK-PCA ne parvient à identifier aucun point négatif dans la population de Monte-Carlo et aboutit à une probabilité de défaillance estimée nulle.

Les trois techniques de sélection du point d'ancrage présentées précédemment (section 4.4.1) ne permettent pas à la méthode AK-HDMR1 de converger vers une bonne estimation de la probabilité de défaillance de ce problème. Ici, la même population de Monte-Carlo est utilisée pour les simulations de Monte-Carlo et la méthode AK-HDMR1, ainsi un nouveau point d'ancrage peut être testé correspondant à un point effectivement négatif selon la vraie fonction de performance. À partir de ce point d'ancrage, la méthode AK-HDMR1 réussit à estimer une probabilité de défaillance proche de la probabilité de référence estimée par simulations de Monte-Carlo. Bien que l'algorithme de sélection du point d'ancrage proposé en section 4.4.1 trouve un point acceptable dans beaucoup de cas, lorsque la fonction de performance devient très fortement non-linéaire, ce point n'est plus suffisant et fait aboutir la méthode à une mauvaise estimation de la probabilité de défaillance. Un point effectivement négatif selon la fonction de performance est un meilleur point d'ancrage. Cependant, il est très difficile de trouver un tel point en quelques évaluations de la fonction de performance. En effet, cela requiert souvent un grand nombre d'appels au modèle coûteux et devient donc contre-productif.

Cet exemple montre la limitation des deux méthodes. En effet, la complexité du problème rend la méthode AK-PCA complètement inefficace. Concernant AK-HDMR1, l'estimation de la probabilité de défaillance est correcte uniquement lorsqu'un point effectivement défaillant est choisi comme point d'ancrage de la décomposition. Cependant, la recherche d'un tel point peut provoquer des coûts numériques trop importants.

		$d$					
		50		100		200	
		$h = 0.22$		$h = 0.22$		$h = 0.20$	
		$\hat{P}_f$	$N_{appels}$	$\hat{P}_f$	$N_{appels}$	$\hat{P}_f$	$N_{appels}$
MCS		$5.88 \times 10^{-3}$	$10^5$	$1.92 \times 10^{-3}$	$10^5$	$1.20 \times 10^{-3}$	$10^5$
AK-PCA		0	–	0	–	0	–
AK-HDMR1	Point d'ancrage						
	Point moyen	0.937	293	0.995	510	0.999	1001
	MPFP	$4.09 \times 10^{-5}$	250	$5.6 \times 10^{-6}$	500	$2.56 \times 10^{-2}$	1010
	Schéma proposé	$1.0 \times 10^{-3}$	256	$9.0 \times 10^{-5}$	504	$8.7 \times 10^{-4}$	1028
	Point négatif	$7.75 \times 10^{-3}$	267	$2.6 \times 10^{-3}$	209	$1.22 \times 10^{-3}$	1014

TABLE 4.7 – Fonction de Griewank – Résultats de l'analyse de fiabilité.

## 4.7 Conclusions

Le challenge de la grande dimension a tout d'abord été expliqué d'un point de vue géométrique. Après un état de l'art des adaptations des méthodes traditionnelles d'analyse de fiabilité, les options disponibles pour la méta-modélisation ont été discutées. Un constat peut être formulé : peu de méthodes ont été proposées dans la littérature pour résoudre des problèmes de fiabilité en grande dimension considérant des modèles coûteux en temps de calcul.

Deux nouvelles méthodes AK ont donc été proposées, l'une basée sur une décomposition fonctionnelle de type HDMR et l'autre basée sur la réduction de la dimension du problème d'optimisation inhérent à la calibration du krigeage. Les deux méthodes proposées ont été comparées sur deux exemples « défis » pour aboutir aux conclusions suivantes. AK-PCA est efficace et requiert moins d'appels au modèle coûteux que AK-HDMR1 tant que la dimension du problème n'est pas trop élevée ( $\leq 100$ ). Lorsque la dimension augmente, les risques que AK-PCA ne parvienne pas à converger augmentent. Ainsi, AK-HDMR1 devient plus efficace que AK-PCA. Enfin, la considération de problèmes très complexes limite fortement l'efficacité des deux méthodes proposées. Cependant, si un point effectivement défaillant peut être identifié rapidement, alors la méthode AK-HDMR1 semble pouvoir estimer correctement la probabilité de défaillance de ce problème.

## Synthèse du chapitre

**Objectif.** Proposer de nouvelles méthodes, basées sur l'utilisation du métamodèle de krigeage et du schéma adaptatif AK, pour l'analyse de fiabilité en grande dimension avec un modèle coûteux en temps de calcul.

**Propositions.** Deux méthodes ont été développées :

- Méthode AK-HDMR basée sur une décomposition Cut-HDMR de la fonction de performance où chaque composante de la décomposition est modélisée par krigeage.
- Méthode AK-PCA basée sur une réduction de la dimension de l'optimisation des hyper-paramètres du krigeage en utilisant une Analyse en Composantes Principales.

**Publications scientifiques.**

- N. Lelièvre et al. (2018b). « A Kriging-based method to deal with High Dimensional Reliability problems ». In : *IFIP WG-7.5 – Reliability and Optimization of Structural Systems, Zurich*
- N. Lelièvre et al. (2018c). « AK-HDMR1 : a Kriging-based method for high-dimensional reliability problems ». In : *Reliability Engineering and Structural Safety (soumis le 1<sup>er</sup> octobre 2018)*





# Conclusions et perspectives

## Conclusions

La prise en compte des incertitudes est fondamentale lors de la conception de nouveaux produits pour considérer les cas d'utilisation possibles de ce produit et ainsi assurer son bon fonctionnement. Ces travaux de thèse contribuent au développement des méthodes pour l'analyse de fiabilité. L'objectif est tout d'abord de permettre aux concepteurs de bien comprendre les problèmes de conception sous incertitudes, puis de répondre à deux verrous scientifiques majeurs de l'analyse de fiabilité : le calcul de probabilité d'évènements rares et la prise en compte de la grande dimension.

La première contribution de ces travaux de thèse réside en la clarification des problèmes de conception sous incertitudes (Chapitre 1). La formulation de ces problèmes est une étape cruciale en conception. Les concepts de fiabilité et de robustesse ont été définis aux regards de la formulation du problème de conception. Les formulations de problèmes de conception où les incertitudes concernent les fonctions contraintes sont relatives à des problèmes de fiabilité. L'approche probabiliste fait consensus pour la détermination de la probabilité de défaillance du produit conçu. Lorsque les incertitudes affectent les fonctions objectifs du problème, on parle de robustesse. Une conception dite robuste conduit à des performances du produit peu sensibles aux variations des paramètres d'entrée. Aucun consensus n'a permis la formulation d'un indicateur caractérisant la robustesse. L'obtention d'un produit fiable et robuste requiert la prise en compte des incertitudes à la fois dans l'objectif et dans les contraintes du problème. La formulation du problème de conception est donc fondamentale et doit se baser sur la problématique du concepteur.

La seconde contribution concerne l'estimation de faibles probabilités de défaillance, c'est-à-dire associées à des évènements rares, en présence d'un modèle physique ou numérique coûteux. Ces probabilités peuvent traditionnellement être estimées soit par des méthodes de simulation soit par des méthodes d'approximation. Les premières méthodes requièrent de trop nombreuses évaluations du modèle coûteux et ne sont donc pas utilisables industriellement en raison des temps limités dédiés à la conception de nouveaux produits. La deuxième classe de méthodes requiert moins d'évaluations du modèle mais s'appuient sur des hypothèses fortes concernant la forme de l'état limite, ce qui peut engendrer de fortes erreurs d'estimation de la probabilité de défaillance. Ces travaux de thèse

se focalisent donc sur la méta-modélisation et plus particulièrement sur les méthodes AK (Chapitre 2). Une nouvelle méthode AK est proposée pour estimer de faibles probabilités de défaillance (Chapitre 3). Cette méthode, intitulée AK-MCSi, est une amélioration de la méthode AK-MCS. Trois aspects importants sont adressés. Le premier aspect concerne la réduction du temps de calcul engendré par la très grande population de Monte-Carlo à classer par la méthode AK-MCS pour estimer de faibles probabilités de défaillance. Pour cela, des simulations séquentielles sont utilisées pour scinder la population initiale de Monte-Carlo de grande taille ( $10^{\lambda+2}$  pour l'évaluation d'une probabilité de défaillance de l'ordre de  $10^{-\lambda}$ ) en sous-populations de tailles réduites. La seconde amélioration concerne l'utilisation croissante des machines de calculs permettant la parallélisation des simulations. Ainsi, à chaque itération, plusieurs évaluations du modèle coûteux peuvent être réalisées simultanément sans coût de calcul supplémentaire. L'enrichissement de la méthode AK-MCS a donc été adapté pour permettre l'identification de plusieurs points d'apprentissage à chaque itération de l'algorithme. Pour cela, une nouvelle technique basée sur l'algorithme de partitionnement k-means et la fonction d'apprentissage des méthodes AK a été proposée. Enfin, le dernier point porte sur le critère d'arrêt de l'apprentissage. En effet, lors de l'utilisation des simulations séquentielles de Monte-Carlo, il faut s'assurer du bon classement de chacune des sous-populations. Le critère d'arrêt doit donc être sévère pour ne pas perdre de l'information à chaque changement de sous-population. Un nouveau critère a donc été formulé à partir de la probabilité de bonne classification de chacun des échantillons de la population de Monte-Carlo. L'amélioration de la méthode a été étudiée sur différentes applications qui démontrent son efficacité pour l'estimation de faibles probabilités de défaillance avec un nombre convenable d'itérations.

La dernière contribution de ces travaux de thèse concerne le challenge de l'analyse de fiabilité et la grande dimension. Les modèles coûteux constituent ici encore une contrainte inévitable aux problèmes de conception dans l'industrie. Les méthodes de simulation souffrent toujours du nombre trop élevé d'évaluations nécessaires du modèle coûteux. Selon le métamodèle choisi, des difficultés différentes sont rencontrées lors de la considération de problème de fiabilité en grande dimension. Certains métamodèles se retrouvent face à un très grand nombre de coefficients à déterminer comme le chaos polynomial. La détermination de ces coefficients requiert donc un plan d'expériences très larges ce qui provoque des coûts de calcul considérables. D'autres métamodèles demandent la résolution d'un problème d'optimisation souvent de taille au moins égale à la dimension du problème de fiabilité. Le krigeage en est un parfait exemple. Différentes approches peuvent donc être envisagées : l'optimisation directe à partir d'algorithmes performants en grande dimension, la réduction de la dimension du problème ou la décomposition fonctionnelle du modèle. Deux nouvelles méthodes AK ont été proposées pour résoudre de tels problèmes, AK-HDMR1 et AK-PCA (Chapitre 4). La première méthode (AK-HDMR1) s'appuie sur la décomposition fonctionnelle HDMR (*High Dimensional Model Representation*) et permet de remplacer la calibration d'un métamodèle de krigeage en grande dimension par quelques calibrations de métamodèles unidimensionnels. Le schéma d'enrichissement de la méthode AK-MCS a dû être adapté permettant d'identifier à la fois un échantillon et un métamodèle à enrichir. La seconde méthode (AK-PCA) est basée sur la réduction de dimension du problème d'optimisation. Une analyse en composantes principales est réalisée dans

---

l'espace des paramètres à optimiser permettant de sélectionner les composantes les plus représentatives du problème d'optimisation. Un algorithme d'optimisation classique peut ainsi être appliqué à ce nouveau problème de dimension réduite. Différentes applications montrent l'efficacité et soulignent les limitations des deux méthodes proposées.

## Perspectives

Suivant les conclusions obtenues sur le chapitre dédié à la grande dimension, plusieurs investigations peuvent encore être menées pour améliorer l'estimation de probabilités de défaillance associées à des problèmes comptant un grand nombre de variables d'entrée. Tout d'abord, le dernier exemple du Chapitre 4 montre les limitations des deux méthodes proposées et met en avant une piste d'amélioration pour la méthode AK-HDMR1. En effet, cette méthode nécessite de définir un point d'ancrage. Même si une stratégie d'identification a été proposée, l'exemple montre que cette méthode n'est pas optimale et qu'un point d'ancrage pertinent serait un échantillon de la population de Monte-Carlo effectivement défaillant. Des travaux supplémentaires sont donc à envisager pour permettre de construire une méthode d'identification d'un échantillon défaillant à partir de quelques évaluations du modèle coûteux seulement.

La seconde perspective envisageable concerne la mise en œuvre de la méthode AK-HDMR2 qui pourrait fournir une estimation plus précise de la probabilité de défaillance comparée à la méthode AK-HDMR1 en prenant en compte les interactions entre variables d'entrée. La difficulté majeure de cette méthode réside dans le nombre de métamodèles à calibrer qui devient très grand lorsque la dimension augmente. Les méthodes de sélection de variables et de couple de variables devraient donc être plus profondément investiguées pour permettre, à partir de quelques évaluations du modèle coûteux, de diminuer fortement le nombre de métamodèles à calibrer.

Enfin, une dernière piste de réflexion porte sur le couplage des méthodes proposées (AK-PCA + AK-MCS<sub>i</sub> ou AK-HDMR1 + AK-MCS<sub>i</sub>) pour estimer de faibles probabilités de défaillance lorsque les problèmes de fiabilité sont en grande dimension. Il ne semble pas a priori avoir de frein au couplage de ces méthodes puisqu'elles portent sur des parties distinctes de l'algorithme AK-MCS. Ceci apporterait indéniablement une avancée considérable par rapport aux méthodes actuellement proposées.

---



# Annexe **A**

## Exemple d'illustration

L'exemple utilisé pour l'illustration des notions du Chapitre 1 est défini par la fonction de performance suivante :

$$g(u_1, u_2) = 2 + \frac{2}{\exp u_1} - u_2 \quad (\text{A.1})$$

Les lois de probabilité des deux variables aléatoires  $u_1$  et  $u_2$  sont définies par le Tableau A.1.

Variable	Distribution	Moyenne	Écart-type
$u_1$	Weibull	10	1
$u_2$	Gumbel	23	2.5

TABLE A.1 – Exemple Illustration – Variables aléatoires du problème.

La probabilité de défaillance de cet exemple est  $P_f \sim 1.44 \times 10^{-3}$ .



# Illustrations des différents problèmes de conception

Cet annexe présente deux exemples pour illustrer la classification des problèmes de conception. Les différences entre les résultats de chacun des problèmes sont particulièrement observées.

## B.1 Exemple mathématique

Le premier exemple est une fonction mathématique basée sur un unique paramètre  $X$  définie par :

$$f(X) = 9 + 6 \exp -14 + X \sin 15 - X - \frac{1}{8}(8 - X)^2 \quad (\text{B.1})$$

Deux cas sont étudiés ici. Le premier suppose que cette fonction doit être maximisée pour améliorer la performance du système. Pour le second cas, une valeur cible est fixée à  $\bar{f} = 6$ . Dans les deux cas, les contraintes du problème d'optimisation sont définies par :

$$g(X) = X - 5 \quad (\text{B.2})$$

Cette fonction doit être positive pour que le système soit fonctionnel. Dans le contexte incertain, la variable  $X$  suit une loi gaussienne centrée sur sa valeur nominale  $\bar{X}$  et de variance unitaire. La probabilité de défaillance maximale est fixée à  $P_{cible} = 0.1$ . La fonction SNR de Taguchi est ici utilisée comme fonction objectif robuste.

### Cas 1

Les résultats de l'optimisation du cas 1 sont donnés en Figure B.1 et Tableau B.1. Les conceptions admissibles et fiables sont des zones de l'espace correspondant à celles respectivement la fonction contrainte et la fonction contrainte fiable. Sans prendre en compte les incertitudes, toutes les optimisations aboutissent au même résultat correspondant au pic de la fonction objectif. Les optimisations robustes aboutissent elles aussi toutes au même résultat correspondant à un maximum local de la fonction objectif. La différence entre ces deux optima est la forme de la fonction objectif à ces points, l'optimum global est un pic et ne peut donc pas être robuste dans le sens de la sensibilité aux variations de  $X$ .



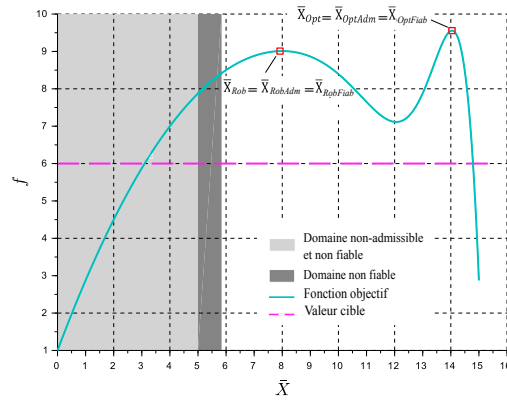


FIGURE B.1 – Exemple mathématique – Résultats de l’optimisation du Cas 1.

		Augmentation des efforts numériques →		
		Robustesse →		
		Aucune fonction objectif	Fonction objectif $X, P$ déterministes	Fonction objectif $X, P$ incertains
Augmentation des efforts numériques ↓	Fiabilité ↓	Aucune fonction contrainte	Conception optimale $\bar{X}_{Opt} = 14.0$ $f = 9.56$	Conception robuste $\bar{X}_{Rob} = 7.9$ $f = 9.00$
		Fonction contrainte $X, P$ déterministes	Conception admissible $\bar{X}_{Adm} \geq 5.0$ Conception optimale et admissible $\bar{X}_{OptAdm} = 14.0$ $f = 9.56$	Conception robuste et admissible $\bar{X}_{RobAdm} = 7.9$ $f = 9.00$
		Fonction contrainte $X, P$ incertains	Conception fiable (fiabilité) $\bar{X}_{Fiab} \geq 5.7$	Conception optimale et fiable (RBDO) $\bar{X}_{OptFiab} = 14.0$ $f = 9.56$

TABLE B.1 – Exemple Mathématique – Résultats de l’optimisation du Cas 1.

### Cas 2

Les résultats de l’optimisation du cas 2 sont donnés en Figure B.2 et Tableau B.2. Une cible  $\bar{f} = 6$  est donc recherchée. Les conceptions admissibles et fiables sont identiques au cas 1. Concernant les optimisations sans considération des incertitudes, les résultats sont identiques pour tous les cas. Il est à noter que selon le point de départ de l’algorithme d’optimisation, l’optimum sans contrainte aurait pu être trouvé dans le voisinage de  $\bar{X} = 3$ . Les optimisations sans incertitudes aboutissent à un résultat exact de  $f = \bar{f} = 6$ . L’optimisation robuste sans contrainte aboutit à un résultat non admissible, et donc non

fiable, proche de la cible. Cependant, lorsque les contraintes sont ajoutées, les optimisations aboutissent à un résultat moins bon par rapport à la cible mais très peu sensible à la variation de  $\bar{X}$ .

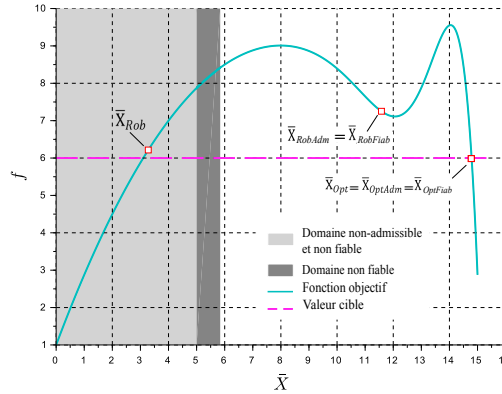


FIGURE B.2 – Exemple mathématique – Résultats de l’optimisation du Cas 2.

		Robustesse >		
		Aucune fonction objectif	Fonction objectif $X, P$ déterministes	Fonction objectif $X, P$ incertains
Augmentation des efforts numériques	Fiabilité	Aucune fonction contrainte	Conception optimale $\bar{X}_{Opt} = 14.8$ $f = 6.00$	Conception robuste $\bar{X}_{Rob} = 3.4$ $f = 6.38$
	↓	Fonction contrainte $X, P$ déterministes	Conception admissible $\bar{X}_{Adm} \geq 5.0$ $\bar{X}_{OptAdm} = 14.8$ $f = 6.00$	Conception robuste et admissible $\bar{X}_{RobAdm} = 11.6$ $f = 7.23$
	↓	Fonction contrainte $X, P$ incertains	Conception fiable (fiabilité) $\bar{X}_{Fiab} \geq 5.7$ $\bar{X}_{OptFiab} = 14.8$ $f = 6.00$	Conception robuste et fiable (RBRDO) $\bar{X}_{RobFiab} = 11.6$ $f = 7.23$

TABLE B.2 – Exemple Mathématique – Résultats de l’optimisation du Cas 2.

### Conclusion

Bien que cet exemple soit basé sur une fonction mathématique simple ne présentant qu’une unique variable de conception, il montre des différences entre les différentes conceptions

possibles. Ici, la différence entre optimum robuste et non robuste est surtout observable par une forte variation de la performance obtenue.

## B.2 Application à un container

Le second exemple est basé sur un container cylindrique défini par sa hauteur  $h$  et son rayon  $R$ . L'objectif de cet exemple est de minimiser la quantité de matériaux utilisée pour la fabrication de ce container. La fonction objectif est donc définie par l'aire de la tôle d'acier utilisée :

$$f(R, h) = 2\pi Rh + 2\pi R^2 \quad (\text{B.3})$$

Ce container doit pouvoir contenir un volume de  $33\text{cm}^3$ . La fonction contrainte est donc définie par :

$$g(R, h) = \pi R^2 h - 33 \quad (\text{B.4})$$

Les variables de conception sont bornées telle que  $1 \leq R \leq 4$  et  $1 \leq h \leq 10$ . La fonction robuste utilisée est une combinaison linéaire de l'espérance et de l'écart-type de la fonction objectif telle que :

$$\Psi(R, h) = 0.25 \times \mathbb{E}[f(R, h)] + 0.75 \times \sigma(f(R, h)) \quad (\text{B.5})$$

Les résultats de l'optimisation sont donnés en Figure B.3. Lorsque les incertitudes sont considérées, les variables suivent des lois gaussiennes centrées sur leur valeur nominale et de variance unitaire. La probabilité de défaillance cible est égale à 0.1. Les conceptions admissible et fiable correspondent à des ensembles de solution définis sur les deux figures. Lorsque la contrainte n'est pas prise en compte, les conceptions optimale et robuste aboutissent à une solution où les variables sont à leur minimum. Si les contraintes sont prises en compte, les quatre optimisations aboutissent à des résultats tous différents. La considération des incertitudes, dans la fonction objectif et/ou contrainte, a donc une très grande influence sur le résultat obtenu lors de la conception.

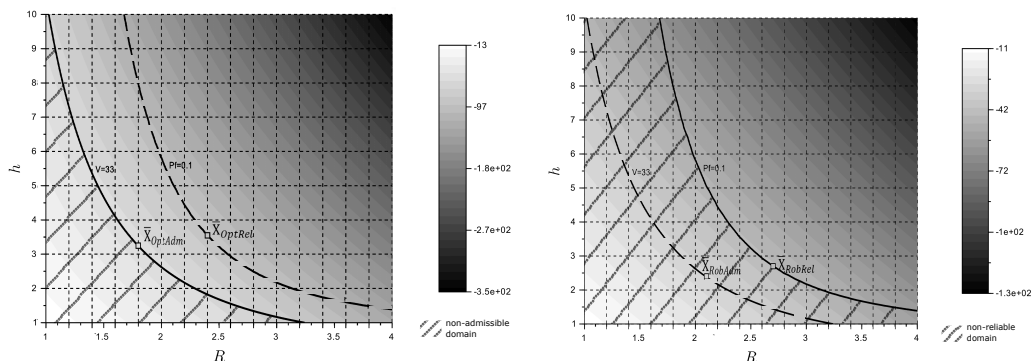


FIGURE B.3 – Exemple container – Résultats de l'optimisation.

# Algorithmes des méthodes AK pour l'atteinte des faibles probabilités de défaillance

## C.1 AK-IS

La méthode AK-IS propose de combiner le schéma adaptatif de méta-modélisation de type AK avec les tirages d'importance. La procédure est représentée en Figure C.1.

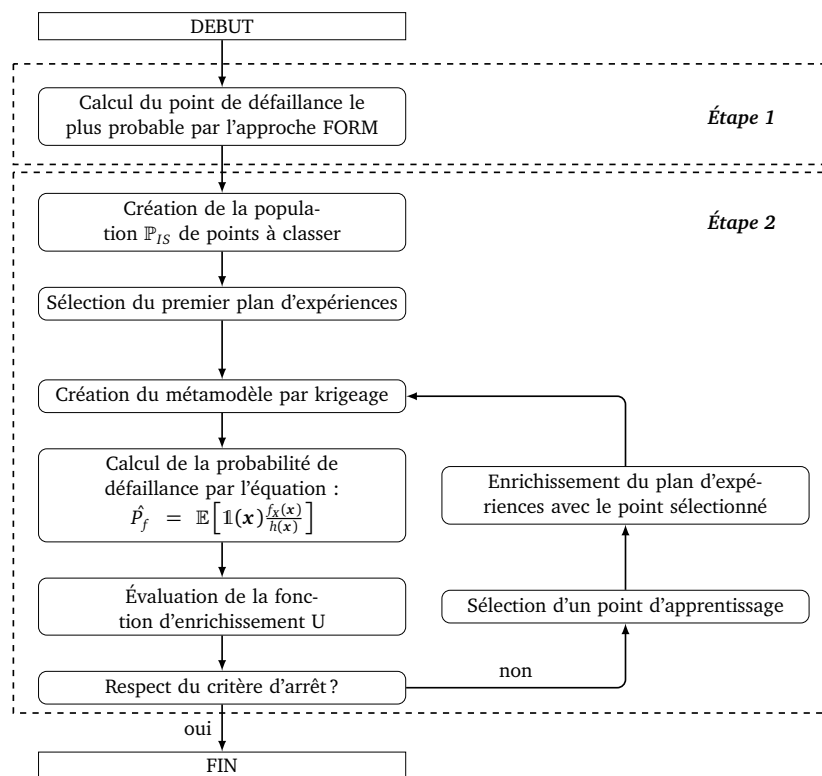


FIGURE C.1 – Procédure de AK-IS.

## C.2 metaAK-IS<sup>2</sup>

La méthode metaAK-IS<sup>2</sup> propose de combiner le schéma adaptatif de métamodélisation de type AK avec les tirages d'importance. La procédure est représentée en Figure C.2. DOE désigne le plan d'expériences.

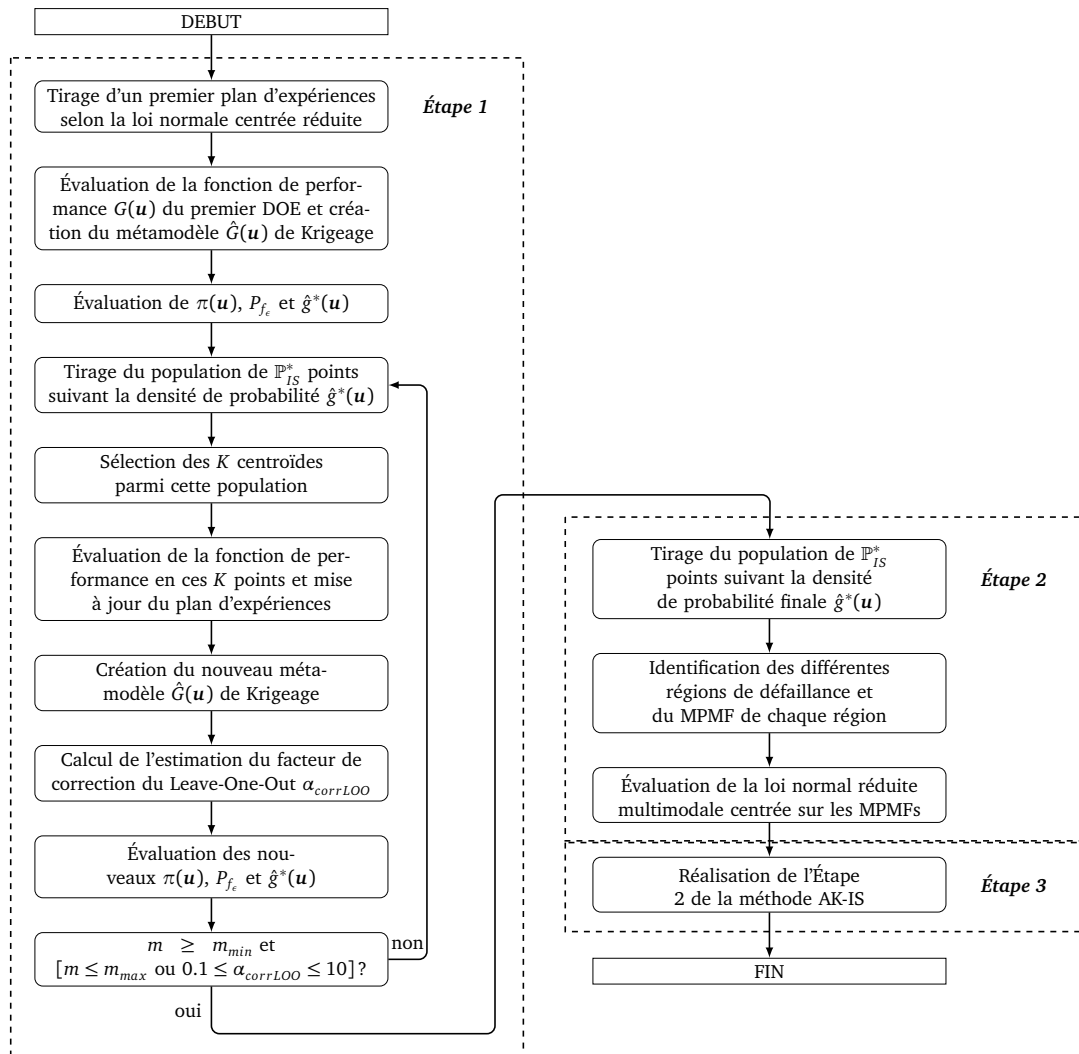


FIGURE C.2 – Procédure de metaAK-IS<sup>2</sup>.

### C.3 AK-SS

La méthode AK-SS propose de combiner le schéma adaptatif de métamodélisation de type AK avec les *Subset Simulations*. La procédure est représentée en Figure C.3.

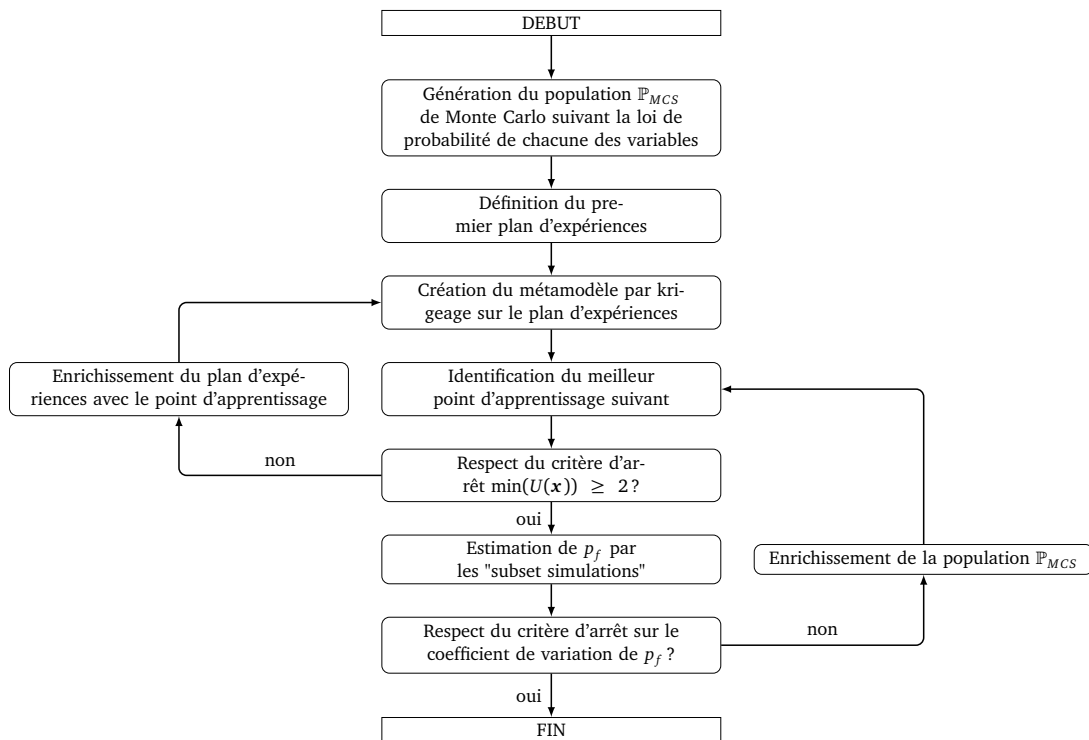


FIGURE C.3 – Procédure de AK-SS.

## C.4 AK-SSIS

La méthode AK-SSIS propose de combiner le schéma adaptatif de métamodélisation de type AK avec les *Subset Simulations Importance Sampling*. La procédure est représentée en Figure C.4.

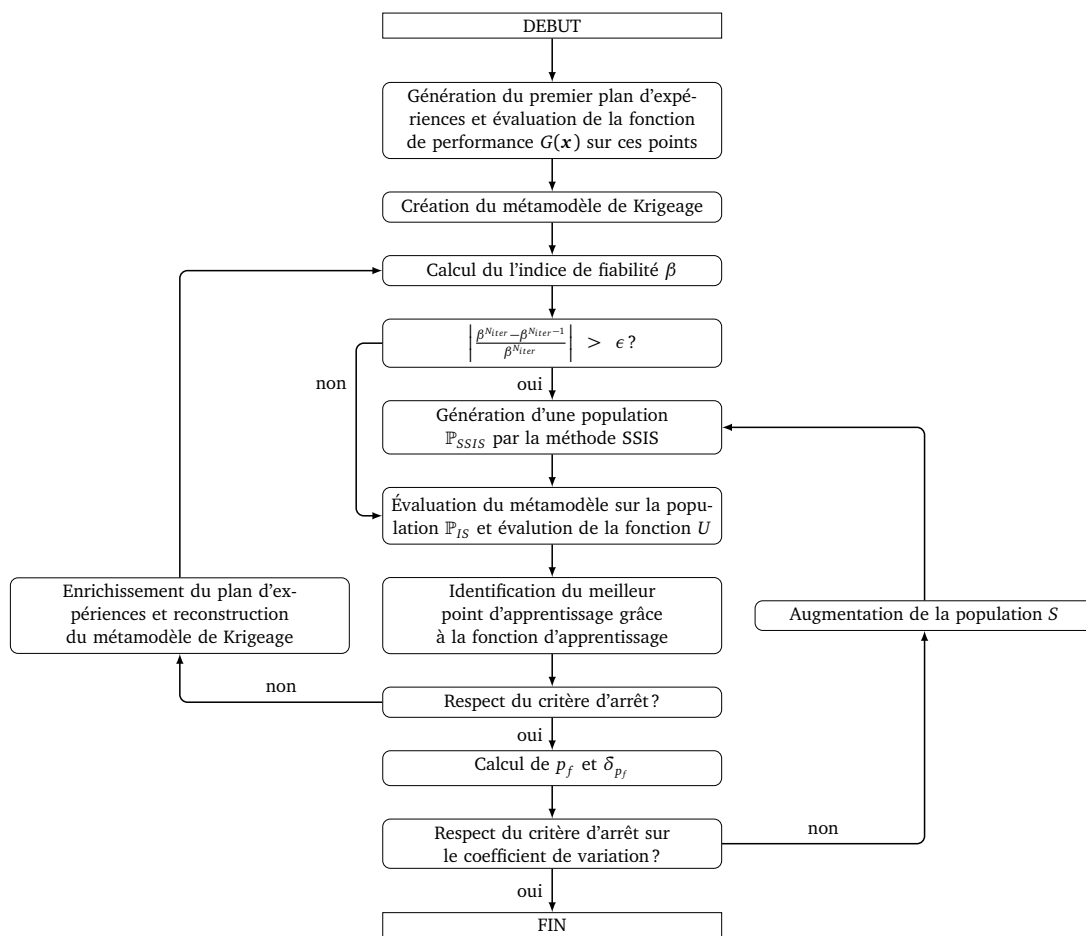


FIGURE C.4 – Procédure de AK-SSIS.

## C.5 AK-LS

La méthode AK-LS propose de combiner le schéma adaptatif de métamodélisation de type AK avec les *Line Sampling*. La procédure est représentée en Figure C.5.

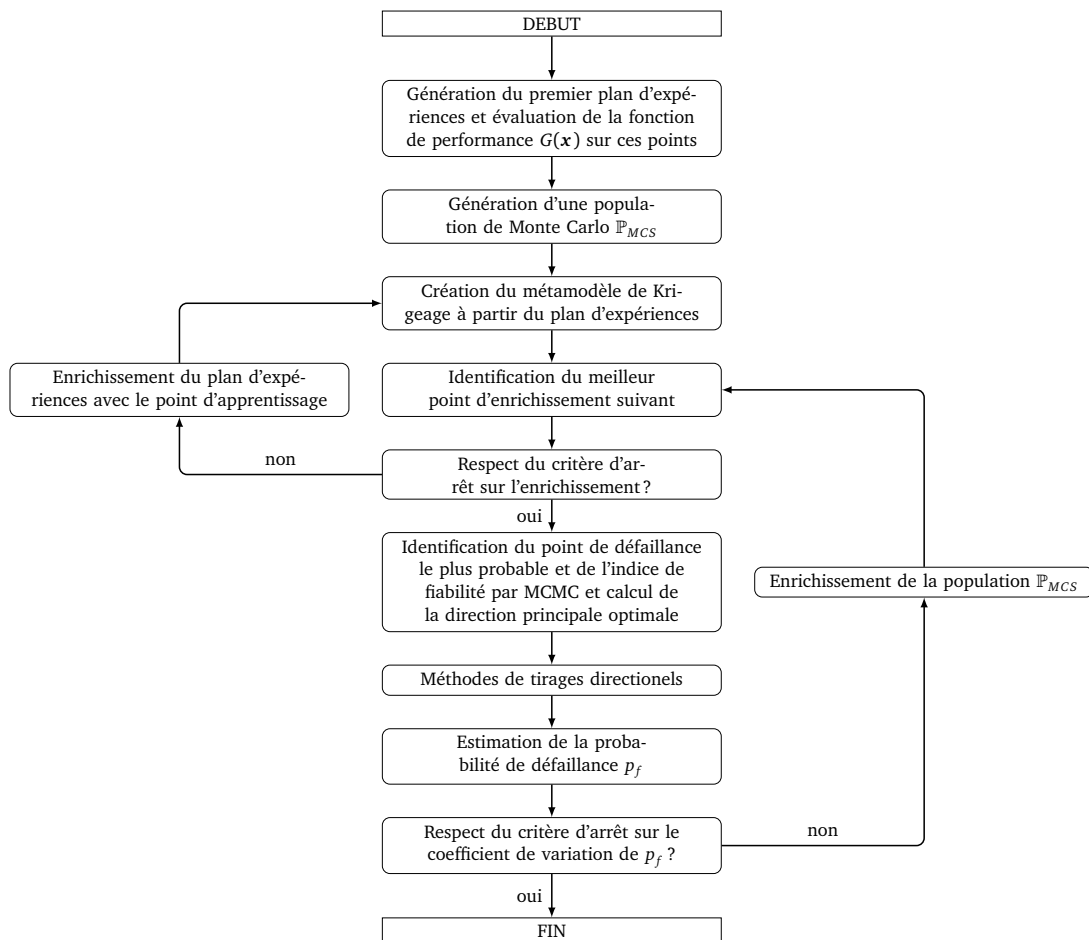


FIGURE C.5 – Procédure de AK-LS.



## C.6 AK-RBIS

La méthode AK-RBIS propose de combiner le schéma adaptatif de métamodélisation de type AK avec les *Radial based Importance Sampling*. La procédure est représentée en Figure C.6.

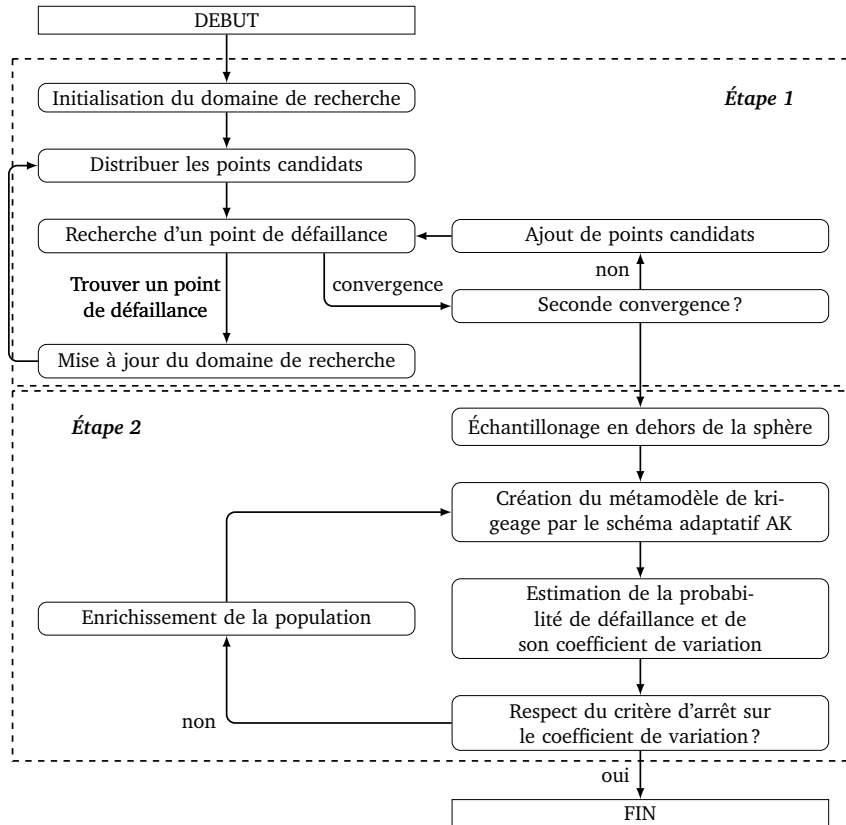


FIGURE C.6 – Procédure de AK-RBIS.

## Démonstration de la reformulation de la décomposition HDMR pour AK-HDMR2

Cette annexe présente la démonstration de la reformulation de la décomposition HDMR permettant de déterminer la prédiction et la variance de krigeage dans le cadre de la méthode AK-HDMR2.

La décomposition originelle s'écrit

$$G(\mathbf{u}) \approx \hat{G}(\mathbf{u}) = g_0 + \sum_{i=1}^d \hat{g}_i(u_i) \quad (\text{D.1})$$

avec

$$g_0 = G(\bar{\mathbf{u}}) \quad (\text{D.2})$$

$$\hat{g}_i(u_i) = \hat{G}_i(u_i, \bar{\mathbf{u}}) - g_0 \quad (\text{D.3})$$

$$\hat{g}_{ij}(u_i, u_j) = \hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}}) - \hat{g}_i(u_i) - \hat{g}_j(u_j) - g_0 \quad (\text{D.4})$$

soit, en condensant les quatre équations précédentes :

$$\hat{G}(\mathbf{u}) = g_0 + \sum_{i=1}^n [\hat{G}_i(u_i, \bar{\mathbf{u}}) - g_0] + \sum_{i=1}^{n-1} \sum_{j=i+1}^n [\hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}}) - \hat{G}_i(u_i, \bar{\mathbf{u}}) - \hat{G}_j(u_j, \bar{\mathbf{u}}) + g_0] \quad (\text{D.5})$$

Tous les termes d'ordres identiques peuvent ensuite être regroupés :

$$\begin{aligned} \hat{G}(\mathbf{u}) &= \left(1 - n + \frac{n(n-1)}{2}\right) g_0 && (\text{ordre 0}) \\ &+ \sum_{i=1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) - \sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) + \hat{G}_j(u_j, \bar{\mathbf{u}}) && (\text{ordre 1}) \\ &+ \sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}}) && (\text{ordre 2}) \end{aligned} \quad (\text{D.6})$$

Le terme d'ordre 2 ne nécessite aucune manipulation. Le terme d'ordre 0 s'écrit :

$$\hat{G}_{\text{ordre 0}}(\mathbf{u}) = \left(\frac{2 + n(n-3)}{2}\right) g_0 \quad (\text{D.7})$$

Le terme d'ordre 1 est transformé de la manière suivante :

$$\begin{aligned}\hat{G}_{\text{ordre 1}}(\mathbf{u}) &= \sum_{i=1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) - \sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) + \hat{G}_j(u_j, \bar{\mathbf{u}}) \\ &= \sum_{i=1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) - \sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) - \sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_j(u_j, \bar{\mathbf{u}})\end{aligned}\tag{D.8}$$

Le second terme de cette expression peut s'écrire :

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) = \sum_{i=1}^{n-1} (n-i) \hat{G}_i(u_i, \bar{\mathbf{u}}) = \sum_{i=1}^n (n-i) \hat{G}_i(u_i, \bar{\mathbf{u}})\tag{D.9}$$

Le troisième terme peut s'écrire :

$$\begin{aligned}\sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_j(u_j, \bar{\mathbf{u}}) &= \hat{G}_2(u_2, \bar{\mathbf{u}}) + 2 * \hat{G}_3(u_3, \bar{\mathbf{u}}) + \dots + (n-1) \hat{G}_n(u_n, \bar{\mathbf{u}}) \\ &= \sum_{i=2}^n (i-1) \hat{G}_i(u_i, \bar{\mathbf{u}}) \\ &= \sum_{i=1}^n (i-1) \hat{G}_i(u_i, \bar{\mathbf{u}})\end{aligned}\tag{D.10}$$

En regroupant ces différentes expressions, la terme d'ordre 1 devient :

$$\hat{G}_{\text{ordre 1}}(\mathbf{u}) = \sum_{i=1}^n (2-n) \hat{G}_i(u_i, \bar{\mathbf{u}})\tag{D.11}$$

Finalement, la décomposition HDMR à l'ordre 2 peut être écrite comme :

$$\boxed{\hat{G}(\mathbf{u}) = \frac{2+n(n-3)}{2} g_0 + (2-n) \sum_{i=1}^n \hat{G}_i(u_i, \bar{\mathbf{u}}) + \sum_{i=1}^{n-1} \sum_{j=i+1}^n \hat{G}_{ij}(u_i, u_j, \bar{\mathbf{u}})}\tag{D.12}$$

# Bibliographie

- Acar, E. et K. Solanki (2009). « System reliability based vehicle design for crashworthiness and effects of various uncertainty reduction measures ». In : *Structural and Multidisciplinary Optimization* 39, p. 311–325.
- Agrawal, R., J. Gehrke, D. Gunopulos et P. Raghavan (1998). « Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications ». In : *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*. SIGMOD '98. Seattle, Washington, USA : ACM, p. 94–105.
- Alban, A., H.A. Darji, A. Imamura et M.K. Nakayama (2017). « Efficient Monte Carlo methods for estimating failure probabilities ». In : *Reliability Engineering & System Safety* 165, p. 376–394.
- Allen, J., C.C. Seepersad et F. Mistree (2006). « Robust Design for Multiscale and Multidisciplinary Applications ». In : *Journal of Mechanical Design* 128, p. 832–843.
- Andrei, N. (2016). « An adaptive conjugate gradient algorithm for large-scale unconstrained optimization ». In : *Journal of Computational and Applied Mathematics* 292, p. 83–91.
- Ankerst, M., M. M. Breunig, H.-P. Kriegel et J. Sander (1999). « OPTICS : Ordering Points to Identify the Clustering Structure ». In : *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*. SIGMOD '99. Philadelphia, Pennsylvania, USA : ACM, p. 49–60.
- Aoues, Y. et A. Chateauneuf (2010). « Benchmark study of numerical methods for reliability-based design optimization ». In : *Structural and Multidisciplinary Optimization* 41, p. 277–294.
- Asmussen, S., J. Blanchet, S. Juneja et L. Rojas-Nandayapa (2011). « Efficient simulation of tail probabilities of sums of correlated lognormals ». In : *Annals of Operations Research* 189.1, p. 5–23.
- Asmussen, S. et P.W. Glynn (2007). *Stochastic Simulation : Algorithms and Analysis*. Springer, New York.
- Au, S.K. et J.L. Beck (1999). « A new adaptive importance sampling scheme for reliability calculations ». In : *Structural Safety* 21.2, p. 135–158.

- Au, S.-K. et J.L. Beck (2001). « Estimation of small failure probabilities in high dimensions by subset simulation ». In : *Probabilistic Engineering Mechanics* 16.4, p. 263–277.
- Aurenhammer, F. (1991). « Voronoi Diagrams - A survey of a fundamental geometric data structure ». In : *ACM Comput. Surv.* 23.3.
- Aversano, G. et A. Parente (2016). *PCA & Kriging for model reduction*. Rapp. tech. The Combustion Institute.
- Aversano, G., A. Parente et O. Gicquel (2017). « PCA & Kriging for Surrogate Models ». In : *Sixteenth International Conference on Numerical Combustion, SIAM, April 3-5*.
- Balesdent, M., J. Morio et L. Brevault (2016). « Rare Event Probability Estimation in the Presence of Epistemic Uncertainty on Input Probability Distribution Parameters ». In : *Methodology and Computing in Applied Probability* 18.1, p. 197–216.
- Balesdent, M., J. Morio et J. Marzat (2013). « Kriging-based adaptive Importance Sampling algorithms for rare event estimation ». In : *Structural Safety* 44, p. 1–10.
- Ball, G.H. et L.J. Hubert (1965). *ISODATA, a novel method of data analysis and pattern classification (Tech. Rep. NTIS No. AD 699616)*. Rapp. tech. Menlo Park, CA : Stanford Research Institute.
- Basudhar, A. et S. Missoum (2010). « An improved adaptive sampling scheme for the construction of explicit boundaries ». In : *Structural and Multidisciplinary Optimization* 42.4, p. 517–529.
- Baudoui, V., P. Klotz, J-B. Hiriart-Urruty, S. Jan et F. Morel (2012). « LOcal Uncertainty Processing (LOUP) method for multidisciplinary robust design optimization ». In : *Structural and Multidisciplinary Optimization* 46, p. 711–726.
- Beck, A.T. et W.J. de Santana Gomes (2012). « A comparison of deterministic, reliability-based and risk-based structural optimization under Uncertainty ». In : *Probabilistic Engineering Mechanics* 28, p. 18–29.
- Bect, J., D. Ginsbourger, L. Li, V. Picheny et E. Vazquez (2012). « Sequential design of computer experiments for the estimation of a probability of failure ». In : *Statistics and Computing* 22 (3), p. 773–793.
- Bellman, R.E. (1961). *Adaptive Control Processes : a Guided Tour*. Princeton University Press.
- Berg, B. A. et T. Neuhaus (1992). « Multicanonical ensemble : A new approach to simulate first-order phase transitions ». In : *Physical Review Letters* 68 (1), p. 9–12.
- Berg, B.A. et T. Neuhaus (1991). « Multicanonical algorithms for first order phase transitions ». In : *Physics Letters B* 267.2, p. 249–253.
- Beyer, H.G. et B. Sendhoff (2007). « Robust optimization - A comprehensive survey ». In : *Computer Methods in Applied Mechanics and Engineering* 196, p. 3190–3218.
- Bichon, B.J., M.S. Eldred, L.P. Swiler, S. Mahadevan et J.M. McFarland (2008). « Efficient Global Reliability Analysis for Nonlinear Implicit Performance Functions ». In : *AIAA Journal* 46.10, p. 2459–2468.
-

- Bichon, B.J., J.M. John McFarland et S. Mahadevan (2011). « Efficient surrogate models for reliability analysis of systems with multiple failure modes ». In : *Reliability Engineering & System Safety* 96.10, p. 1386–1395.
- Bjerager, P. (1988). « Probability Integration by Directional Simulation ». In : *Journal of Engineering Mechanics* 114.8, p. 1285–1302.
- Blatman, G. et B. Sudret (2008). « Sparse polynomial chaos expansion and adaptive Stochastic finite element using a regression approach ». In : *Comptes rendus mécanique*.
- (2011). « Adaptive sparse polynomial chaos expansion based on least angle regression ». In : *Journal of Computational Physics* 230.6, p. 2345–2367.
- Boser, B.E., I.M. Guyon et V.N. Vapnik (1992). « A Training Algorithm for Optimal Margin Classifiers ». In : *Proceedings of the Fifth Annual Workshop on Computational Learning Theory. COLT '92*. Pittsburgh, Pennsylvania, USA : ACM, p. 144–152.
- Botev, Z. I., P. L'Ecuyer et B. Tuffin (2013). « Markov chain importance sampling with applications to rare event probability estimation ». In : *Statistics and Computing* 23.2, p. 271–285.
- Bouhlel, M.A., N. Bartoli, A. Otsmane et J. Morlier (2016). « Improving kriging surrogates of high-dimensional design models by Partial Least Squares dimension reduction ». In : *Struct. Multidiscip. Optim.* 53.5, p. 935–952.
- Bouhlel, M.A. et J. R. R. A. Martins (2018). « Gradient-enhanced kriging for high-dimensional problems ». In : *Engineering with Computers*.
- Bourinet, J.-M. (2015). « Reliability assessment with adaptive surrogates based on support vector machine regression ». In : *1<sup>st</sup> International Conference on Uncertainty Quantification in Computational Sciences and Engineering (UNCECOMP 2015)*, Heraklion, Crète, Grèce, p. 652–663.
- (2016). « Rare-event probability estimation with adaptive support vector regression surrogates ». In : *Reliability Engineering & System Safety* 150, p. 210–221.
- Bourinet, J.-M. (2017). « Anisotropic-kernel-based support vector regression for reliability assessment ». In : *12<sup>th</sup> International Conference on Structural Safety and Reliability*, Vienne, Autriche, p. 257–266.
- (2018). « Reliability analysis and optimal design under uncertainty - Focus on adaptive surrogate-based approaches ». Thèse de doct. (HDR) SIGMA Clermont, Institut Pascal, UMR CNRS 6603.
- Bourinet, J.-M., F. Deheeger et M. Lemaire (2011). « Assessing small failure probabilities by combined subset simulation and Support Vector Machines ». In : *Structural Safety* 33.6, p. 343–353.
- Bourinet, J.-M., N. Gayton, M. Lemaire et A. Combescure (2000). « Reliability analysis of stability of shells based on combined finite element and response surface methods ». In : *Proc. 4<sup>th</sup> International Colloquium on Computation of Shell & Spatial Structures (IASS-IACM 2000)*, Chania, Crète, Grèce.
-

- Braydi, O., P. Lafon et R. Younes (2018). « On the formulation of optimization problems under uncertainty in mechanical design ». In : *International Journal on Interactive Design and Manufacturing (IJIDeM)*.
- Bucher, C. et T. Most (2008). « A comparison of approximate response functions in structural reliability analysis ». In : *Probabilistic Engineering Mechanics* 23.2. 5th International Conference on Computational Stochastic Mechanics, p. 154–163.
- Bucher, C.G. et U. Bourgund (1990). « A fast and efficient response surface approach for structural reliability problems ». In : *Structural Safety* 7.1, p. 57–66.
- Bucklew, J. (2004). *Introduction to Rare-Event Simulation*. Springer, New York.
- Cadini, F., F. Santos et E. Zio (2014). « An improved adaptive kriging-based importance technique for sampling multiple failure regions of low probability ». In : *Reliability Engineering & System Safety* 131, p. 109–117.
- Caliński, T. et J. Harabasz (1974). « A dendrite method for cluster analysis ». In : *Communications in Statistics* 3.1, p. 1–27.
- Cannamela, C., J. Garnier et B. Iooss (2008). « Controlled Stratification for Quantile Estimation ». In : *The Annals of Applied Statistics* 2.4, p. 1554–1580.
- Cervantes, J., X. Li, W. Yu et K. Li (2008). « Support vector machine classification for large data sets via minimum enclosing ball clustering ». In : *Neurocomputing* 71.4, p. 611–619.
- Chavent, M., Y. Lechevallier et O. Briant (2007). « DIVCLUS-T : A monothetic divisive hierarchical clustering method ». In : *Computational Statistics & Data Analysis* 52.2, p. 687–701.
- Chen, K.-Y. (2007). « Forecasting systems reliability based on support vector regression with genetic algorithms ». In : *Reliability Engineering & System Safety* 92.4, p. 423–432.
- Chen, W., J.K. Allen, K-L. Tsui et F. Mistree (1996). « A procedure for robust design : minimizing variations caused by noise factors and control factors ». In : *Journal of Mechanical Design* 118, p. 478–485.
- Chen, X. et J. Li (2017). « A subset multicanonical Monte Carlo method for simulating rare failure events ». In : *Journal of Computational Physics* 344, p. 23–35.
- Cheng, J. et Q.S. Li (2008). « Reliability analysis of structures using artificial neural network based genetic algorithms ». In : *Computer Methods in Applied Mechanics and Engineering* 197.45, p. 3742–3750.
- Cheng, K. et Z. Lu (2018). « Adaptive sparse polynomial chaos expansions for global sensitivity analysis based on support vector regression ». In : *Computers & Structures* 194, p. 86–96.
- Chojaczyk, A.A., A.P. Teixeira, L.C. Neves, J.B. Cardoso et C. Guedes Soares (2015). « Review and application of Artificial Neural Networks models in reliability analysis of steel structures ». In : *Structural Safety* 52, p. 78–89.
-

- Chowdhury, R. et B.N. Rao (2009a). « Assessment of high dimensional model representation techniques for reliability analysis ». In : *Probabilistic Engineering Mechanics* 24.1, p. 100–115.
- (2009b). « Hybrid High Dimensional Model Representation for reliability analysis ». In : *Computer Methods in Applied Mechanics and Engineering* 198.5, p. 753–765.
- Cochran, W.G. (1977). *Sampling technique. 3rd ed.* Wiley.
- Crestaux, T., O. Le Maître et J.-M. Martinez (2009). « Polynomial chaos expansion for sensitivity analysis ». In : *Reliability Engineering & System Safety* 94.7. Special Issue on Sensitivity Analysis, p. 1161–1172.
- Dai, H., H. Zhang, W. Wang et G. Xue (2012). « Structural Reliability Assessment by Local Approximation of Limit State Functions Using Adaptive Markov Chain Simulation and Support Vector Regression ». In : *Computer-Aided Civil and Infrastructure Engineering* 27.9, p. 676–686.
- Davies, D.L. et D.W. Bouldin (1979). « A Cluster Separation Measure ». In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1.2, p. 224–227.
- Deb, K., A. Pratap, S. Agarwal et T. Meyarivan (2002). « A fast and elitist multiobjective genetic algorithm : NSGA-II ». In : *IEEE Transactions on Evolutionary Computation* 6.2, p. 182–197.
- Deheeger, F. (2008). « Couplage mécano-fiabiliste : 2SMART - méthodologie d'apprentissage stochastique en fiabilité ». Thèse de doct. Université BLAISE PASCAL - Clermont II.
- Deheeger, F. et M. Lemaire (2006). « Reliability analysis by support vector machine classification ». In : *3<sup>rd</sup> ASRANet International Colloquium, Glasgow, UK*.
- Der Kiureghian, A. et T. Dakessian (1998). « Multiple design points in first and second-order reliability ». In : *Structural Safety* 20.1, p. 37–49.
- Der Kiureghian, A. et O. Ditlevsen (2009). « Aleatory or epistemic? Does it matter? » In : *Structural Safety* 31, p. 105–112.
- Du, X., A. Sudjianto et W. Chen (2004). « An Integrated Framework for Optimization Under Uncertainty Using Inverse Reliability Strategy ». In : *Journal of Mechanical Design* 126, p. 562–570.
- Dubois, D. et H. Prade (1980). « Fuzzy sets and systems : Theory and Application ». In : *Mathematics in Science and Engineering* 144.
- Dubourg, V. (2011). « Adaptive surrogate models for reliability analysis and reliability-based design optimization ». Thèse de doct. Université Blaise Pascal - Clermont Ferrand.
- Dubourg, V., B. Sudret et J.-M. Bourinet (2011). « Reliability-based design optimization using kriging surrogates and subset simulation ». In : *Structural and Multidisciplinary Optimization* 44, p. 673–690.
-



- Dubourg, V., B. Sudret et F. Deheeger (2013). « Metamodel-based importance sampling for structural reliability analysis ». In : *Probabilistic Engineering Mechanics* 33, p. 47–57.
- Dumas, A., B. Echard, N. Gayton, O. Rochat, J.-Y. Dantan et S. Van Der Veen (2013). « AK-ILS : An Active learning method based on Kriging for the Inspection of Large Surfaces ». In : *Precision Engineering* 37.1, p. 1–9.
- Dunn, J.C. (1973). « A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters ». In : *Journal of Cybernetics* 3.3, p. 32–57.
- Dupuis, P., K. Leder et H. Wang (2007). « Importance Sampling for Sums of Random Variables with Regularly Varying Tails ». In : *ACM Transactions on Modeling and Computer Simulation* 17.3.
- Echard, B. (2012). « Evaluation par krigeage de la fiabilité des structures sollicitées en fatigue ». Thèse de doct. Université Blaise Pascal.
- Echard, B., N. Gayton et M. Lemaire (2011). « AK-MCS : An active learning reliability method combining Kriging and Monte Carlo Simulation ». In : *Structural Safety* 33.2, p. 145–154.
- Echard, B., N. Gayton, M. Lemaire et N. Relun (2013). « A combined Importance Sampling and Kriging reliability method for small failure probabilities with time-demanding numerical models ». In : *Reliability Engineering and System Safety* 111, p. 232–240.
- Efron, B., T. Hastie, I. Johnstone et R. Tibshirani (2004). « Least Angle Regression ». In : *The Annals of Statistics*.
- El Hami, A., B. Radi et C. Huang (2017). « Overview of Structural Reliability Analysis Methods ». In : *Incertitudes et fiabilité des systèmes multiphysiques* 1. Optimisation et Fiabilité.
- Enevoldsen, I., M. H. Faber et J.D. Sørensen (1993). « Adaptive response surface techniques in reliability estimation ». In : *Proc. 6<sup>th</sup> International Conference on Structural Safety and Reliability (ICOSSAR '93), Innsbruck, Austria*, p. 1257–1264.
- Enevoldsen, I. et J.D. Sørensen (1994). « Reliability-based optimization in structural engineering ». In : *Structural Safety* 15, p. 169–196.
- Ester, M., H.-P. Kriegel, J. Sander et X. Xu (1996). « A Density-based Algorithm for Discovering Clusters a Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise ». In : *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. KDD'96*. Portland, Oregon : AAAI Press, p. 226–231.
- Étoré, P. et B. Jourdain (2010). « Adaptive Optimal Allocation in Stratified Sampling Methods ». In : *Methodology and Computing in Applied Probability* 12.3, p. 335–360.
- Faber, M.H. et J.D. Sørensen (2002). « Reliability Based Code Calibration ». In : *Structural Safety (Joint committee)*.
- Faravelli, L. (1989). « Response-Surface Approach for Reliability Analysis ». In : *Journal of Engineering Mechanics* 115.12, p. 2763–2781.
-

- Fauriat, W. et N. Gayton (2014). « AK-SYS : An adaptation of the AK-MCS method for system reliability ». In : *Reliability Engineering & System Safety* 123, p. 137–144.
- Fouchal, S. (2011). « Partitionnement d'objet complexe : méthodes et algorithmes ». Thèse de doct. Université Paris 8.
- Fraley, C. et A. E. Raftery (1998). « How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis ». In : *The Computer Journal* 41.8, p. 578–588.
- Gang, W., S. Wang, F. Xiao et D. Gao (2015). « Robust optimal design of building cooling systems considering cooling load uncertainty and equipment reliability ». In : *Applied Energy* 159, p. 265–275.
- Garvels, M. J. J. (2011). « A combined splitting–cross entropy method for rare-event probability estimation of queueing networks ». In : *Annals of Operations Research* 189.1, p. 167–185.
- Gaspar, B., A.P. Teixeira et C. Guedes Soares (2015). « A study on a stopping criterion for active refinement algorithms in Kriging surrogate models ». In : *Safety and Reliability of Complex Engineered Systems*.
- Gasser, M. (1997). « Reliability-Based Optimization of Structural Systems ». In : *Mathematical Methods of Operations Research* 46, p. 287–307.
- Gayton, N. (2012). « Méthode probabilistes pour la conception mécanique. Application à la fatigue des structures et à l'analyse des tolérances (Habilitation à Diriger des Recherches) ». Thèse de doct. Institut Français de Mécanique Avancée, Institut Pascal.
- Gayton, N., J.-M. Bourinet et M. Lemaire (2003). « CQ2RS : a new statistical approach to the response surface method for reliability analysis ». In : *Structural Safety* 25.1, p. 99–121.
- Ghabraie, K., R. Chan, X. Huang et Y.M. Xie (2010). « Shape optimization of metallic yielding devices for passive mitigation of seismic energy ». In : *Engineering Structures* 32, p. 2258–2267.
- Ghanem, R.G. et P.D. Spanos (1991). *Stochastic finite elements : a spectral approach*. Springer-Verlag New York.
- Göhler, S.M., T. Eifler et T.J. Howard (2016). « Robustness Metrics : Consolidating the multiple approaches to quantify Robustness ». In : *Journal of Mechanical Design* 138.
- Gouda, A. A. et T. Szántai (2008). « Rare event probabilities in stochastic networks ». In : *Central European Journal of Operations Research* 16.4, p. 441–461.
- Greenwood, W.H. et K.W. Chase (1987). « A new tolerance analysis method for designers and manufacturers ». In : *Journal of Mechanical Design* 109, p. 112–116.
- Grooteman, F. (2008). « Adaptive radial-based importance sampling method for structural reliability ». In : *Structural Safety* 30.6, p. 533–542.
-

- Hansen, N., S.D. Müller et P. Koumoutsakos (2003). « Reducing the time complexity of the Derandomized Evolution Strategy with covariance matrix adaptation (CMA-ES) ». In : *Evolutionary Computation* 11.1, p. 1–18.
- Hartigan, J.A. (1975). *Clustering Algorithms*. 99th. New York, NY, USA : John Wiley & Sons, Inc.
- Hensman, J., N. Fusi et N. D. Lawrence (2013). « Gaussian Processes for Big Data ». In : *CoRR*.
- Heping, P. et J. Xiangqian (2009). « Evaluation and management procedure of measurement uncertainty in new generation geometrical product specification (GPS) ». In : *Measurement* 42.5, p. 653–660.
- Herault, J., C. Jutten et B. Ans (1985). « Détection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en Apprentissage non supervisé ». In : *Acte du dixième colloque sur le traitement du signal et ses applications, Nice, France*.
- Hinneburg, A. et H.-H. Gabriel (2007). « DENCLUE 2.0 : Fast Clustering Based on Kernel Density Estimation ». In : *Advances in Intelligent Data Analysis VII*. Sous la dir. de R.M. Berthold, J. Shawe-Taylor et N. Lavrač. Springer Berlin Heidelberg, p. 70–80.
- Hinneburg, A. et D.A. Keim (1998). « An Efficient Approach to Clustering in Large Multimedia Databases with Noise ». In : *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining. KDD'98*. New York, NY : AAAI Press, p. 58–65.
- Hohenbichler, M. et R. Rackwitz (1981). « Non-normal dependent vector in structural safety ». In : *Journal of the Engineering Mechanics Division* 107(6), p. 1227–1238.
- (1988). « Improvement of Second-Order Reliability Estimates by Importance Sampling ». In : *Journal of Engineering Mechanics* 114 (12), p. 2195–2199.
- Holland, J. (1992). *Adaptation in natural and artificial systems*. MIT press edition.
- Hosni Elhewy, A., E. Mesbahi et Y. Pu (2006). « Reliability analysis of structures using neural network method ». In : *Probabilistic Engineering Mechanics* 21.1, p. 44–53.
- Hu, Z. et X. Du (2015). « Mixed Efficient global optimization for time-dependent Reliability analysis ». In : *Journal of Mechanical Design* 137.
- Hu, Z. et S. Mahadevan (2016). « Global sensitivity analysis-enhanced surrogate (GSAS) modeling for reliability analysis ». In : *Structural and Multidisciplinary Optimization* 53.3, p. 501–521.
- Huang, H. et S. Lin (2014). « A modified Wei–Yao–Liu conjugate gradient method for unconstrained optimization ». In : *Applied Mathematics and Computation* 231, p. 179–186.
- Huang, X., J. Chen et H. Zhu (2016). « Assessing small failure probabilities by AK–SS : An active learning method combining Kriging and Subset Simulation ». In : *Structural Safety* 59, p. 86–95.
-

- Hurtado, J.E. et D.A. Alvarez (2001). « Neural-network-based reliability analysis : a comparative study ». In : *Computer Methods in Applied Mechanics and Engineering* 191.1. Micromechanics of Brittle Materials and Stochastic Analysis of Mechanical Systems, p. 113–132.
- Inuiguchi, M. et J. Ramík (2000). « Possibilistic linear programming : a brief review of fuzzy mathematical programming and a comparison with stochastic programming in portfolio selection problem ». In : *Fuzzy Sets and Systems* 111, p. 3–28.
- Iooss, B. (2011). « Revue sur l’analyse de sensibilité globale de modèles numériques ». In : *Journal de la Societe Française de Statistique, Societe Française de Statistique et Societe Mathématique de France* 152 (1), p. 1–23.
- Jones, D.R., M. Schonlau et W.J. Welch (1998). « Efficient Global Optimization of Expensive Black-Box Functions ». In : *Journal of Global Optimization* 13.4, p. 455–492.
- Jutten, C. et J. Herault (1991). « Blind separation of sources, part I : An adaptive algorithm based on neuromimetic architecture ». In : *Signal Processing* 24.1, p. 1–10.
- Kaintura, A., T. Dhaene et D. Spina (2018). « Review of Polynomial Chaos-Based Methods for Uncertainty Quantification in Modern Integrated Circuits ». In : *Electronics* 7.3.
- Kajero, Olumayowa T., Tao Chen, Yuan Yao, Yao-Chen Chuang et David Shan Hill Wong (2016). « Meta-modelling in chemical process system engineering ». In : *J. Taiwan Inst. Chem. Eng.*
- Kang, S.-C., H.-M. Koh et J.F. Choo (2010). « An efficient response surface method using moving least squares approximation for structural reliability analysis ». In : *Probabilistic Engineering Mechanics* 25.4, p. 365–371.
- Katafygiotis, L.S. et S.H. Cheung (2003). « An efficient method for calculation of reliability integrals ». In : *Advances in stochastic structural dynamics*, p. 263–270.
- (2007). « Application of spherical subset simulation method and auxiliary domain method on a benchmark reliability study ». In : *Structural Safety* 29.3, p. 194–207.
- Katafygiotis, L.S. et K.M. Zuev (2008). « Geometric insight into the challenges of solving high-dimensional reliability problems ». In : *Probabilistic Engineering Mechanics* 23.2. 5th International Conference on Computational Stochastic Mechanics, p. 208–218.
- Kaufman, L. et P.J. Rousseeuw (1990). *Finding Groups in data. An introduction to cluster analysis*. New York : Wiley.
- Kaymaz, I. et C.A. McMahon (2005). « A response surface method based on weighted regression for structural reliability analysis ». In : *Probabilistic Engineering Mechanics* 20.1, p. 11–17.
- Kim, S.-H. et S.-W. Na (1997). « Response surface method using vector projected sampling points ». In : *Structural Safety* 19.1. Asian-Pacific Symposium on Structural Reliability and Its Applications, p. 3–19.
- Knopp, K. (1996). « The Laurent Expansion ». In : *Theory of Functions Parts I and II, Two Volumes Bound as One, Part I*. New York : Dover.
-

- Koutsourelakis, P.S., H.J. Pradlwarter et G.I. Schuëller (2004). « Reliability of structures in high dimensions, part I : algorithms and applications ». In : *Probabilistic Engineering Mechanics* 19.4, p. 409–417.
- Krige, D.G. (1951). « A statistical approach to some basic mine valuation problems on the Witwatersrand ». In : *Journal of the Chemical, Metallurgical and Mining Society of South Africa* 52.6, p. 119–139.
- Lancaster, P. et K. Salkauskas (1981). « Surfaces generated by moving least squares Methods ». In : *Mathematics of computation* 37.155, p. 141–158.
- L'Ecuyer, P., J.H. Blanchet, B. Tuffin et P.W. Glynn (2010). « Asymptotic Robustness of Estimators in Rare-event Simulation ». In : *ACM Transactions on Modeling and Computer Simulation* 20.1, 6 :1–6 :41.
- L'Ecuyer, P. et B. Tuffin (2008). « Approximate zero-variance simulation ». In : *2008 Winter Simulation Conference*, p. 170–181.
- Lee, I., K.K. Choi, L. Du et D. Gorsish (2008). « Dimension reduction method for reliability-based robust design optimization ». In : *Computers and Structures* 86, p. 1550–1562.
- Lee, S.H. et B.M. Kwak (2006). « Response surface augmented moment method for efficient reliability analysis ». In : *Structural Safety* 28.3, p. 261–272.
- Lee, Y-S., B-L. Choi, J.H. Lee, S.Y. Kim et S. Han (2014). « Reliability-based design optimization of monopile transition piece for offshore wind turbine system ». In : *Renewable Energy* 71, p. 729–741.
- Lelièvre, N., P. Beaurepaire, C. Mattrand et N. Gayton (2017). « La famille des méthodes AK pour l'analyse de fiabilité : synthèse des récentes avancées ». In : *23<sup>ème</sup> Congrès Français de Mécanique, Lille*.
- (2018a). « AK-MC*S*i : A Kriging-based method to deal with small failure probabilities and time-consuming models ». In : *Structural Safety* 73, p. 1–11.
- Lelièvre, N., P. Beaurepaire, C. Mattrand, N. Gayton et V. Marguin (2018b). « A Kriging-based method to deal with High Dimensional Reliability problems ». In : *IFIP WG-7.5 – Reliability and Optimization of Structural Systems, Zurich*.
- (2018c). « AK-HD*M*R1 : a Kriging-based method for high-dimensional reliability problems ». In : *Reliability Engineering and Structural Safety (soumis le 1<sup>er</sup> octobre 2018)*.
- Lelièvre, N., P. Beaurepaire, C. Mattrand, N. Gayton et A. Otsmane (2016a). « On the consideration of uncertainty in design : optimization - reliability - robustness ». In : *Structural and Multidisciplinary Optimization*, p. 1–15.
- (2016b). « On the consideration of uncertainty in design : optimization - reliability - robustness ». In : *VII European Congress on Computational Methods in Applied Sciences and Engineering*.
- Lemaire, M. (2009). *Structural Reliability*. Sous la dir. de . John Wiley & Sons.
- Li, G., C. Rosenthal et H. Rabitz (2001). « High Dimensional Model Representations ». In : *The Journal of Physical Chemistry A* 105.33, p. 7765–7777.
-

- Li, H.-S., Z.-Z. Lü et Z.-F. Yue (2006). « Support vector machine for structural reliability analysis ». In : *Applied Mathematics and Mechanics* 27.10, p. 1295–1303.
- Li, J. et D. Xiu (2010). « Evaluation of failure probability via surrogate models ». In : *Journal of Computational Physics* 229.23, p. 8966–8980.
- Li, L., J. Bect et E. Vazquez (2012). « Bayesian Subset Simulation : a kriging-based subset simulation algorithm for the estimation of small probabilities of failure ». In : *11<sup>th</sup> International Probabilistic Assessment and Management Conference (PSAM11) and The annual European Safety and Reliability Conference (ESREL2012)*, Helsinki, Finlande.
- Liu, P.-L. et A. Der Kiureghian (1986). « Multivariate distribution models with prescribed marginals and covariance ». In : *Probabilistic Engineering Mechanics* 1(2), p. 105–112.
- Liu, Y., H.K. Jeong et M. Collette (2016). « Efficient optimization of reliability-constrained structural design problems including interval uncertainty ». In : *Computers and Structures* 177, p. 1–11.
- Lophaven, S.N., H.B. Nielsen et J. Søndergaard (2002). *Aspects of the Matlab Toolbox DACE*. Rapp. tech. Technical University of Denmark.
- Lv, A., Z. Lu et P. Wang (2015). « A new learning function for Kriging and its applications to solve reliability problems in engineering ». In : *Computers & Mathematics with Applications* 70.5, p. 1182–1197.
- Ma, J., Z. Ren, G. Zhao, Y. Zhang et C. S. Koh (2018). « A New Reliability Analysis Method Combining Adaptive Kriging With Weight Index Monte Carlo Simulation ». In : *IEEE Transactions on Magnetics* 54.3, p. 1–4.
- Madsen, H.O. et O. Ditlevsen (1996). *Structural reliability methods*. Wiley.
- Mao, H. et S. Mahadevan (2000). « Reliability analysis of creep–fatigue failure ». In : *International Journal of Fatigue* 22.9, p. 789–797.
- Al-Masri, A.N., M.Z.A. Ab Kadir, H. Hizam et N. Mariun (2015). « Simulation of an adaptive artificial neural network for power system security enhancement including control action ». In : *Applied Soft Computing* 29, p. 1–11.
- Matheron, G. (1973). « The Intrinsic Random Functions and Their Applications ». In : *Advances in Applied Probability* 5.3, p. 439–468.
- McQuitty, L.L. (1966). « Similarity Analysis by Reciprocal Pairs for Discrete and Continuous Data ». In : *Educational and Psychological Measurement* 26.4, p. 825–831.
- Mei, Y., X. Wang et G. Cheng (2008). « A feature-based topological optimization for structure design ». In : *Advances in Engineering Software* 39, p. 71–87.
- Meng, Z., P. Hao, G. Li, B. Wang et K. Zhang (2015a). « Non-probabilistic reliability-based design optimization of stiffened shells under buckling constraint ». In : *Thin-Walled Structures* 94, p. 325–333.
-

- Meng, Z., G. Li, B.P. Wang et P. Hao (2015b). « A hybrid chaos control approach of the performance measure function for reliability-based design optimization ». In : *Computers and Structures* 146, p. 32–43.
- Morio, J. (2011). « Non-parametric adaptive importance sampling for the probability estimation of a launcher impact position ». In : *Reliability Engineering & System Safety* 96.1. Special Issue on Safecom 2008, p. 178–183.
- Morris, M.D. (1991). « Factorial Sampling Plans for Preliminary Computational Experiments ». In : *Technometrics* 33.2, p. 161–174.
- Most, T. (2007). « An adaptive response surface approach for structural reliability analyses based on support vector machines ». In : *11<sup>th</sup> International Conference on Civil, Structural and Environmental Engineering Computing, St. Julians, Malta*.
- Moura, M.C., E. Zio, I.D. Lins et E. Drogue (2011). « Failure and reliability prediction by support vector machines regression of time series data ». In : *Reliability Engineering & System Safety* 96.11, p. 1527–1534.
- Mourelatos, Z.P. et J. Liang (2006). « A Methodology for Trading-Off Performance and Robustness Under Uncertainty ». In : *Journal of Mechanical Design* 128.
- Mousazadeh, M., S.A. Torabi et B. Zahiri (2015). « A robust possibilistic programming approach for pharmaceutical supply chain network design ». In : *Computers and Chemical Engineering* 82, p. 115–128.
- Moustapha, M., B. Sudret, J.-M. Bourinet et B. Guillaume (2015). « Adaptive Kriging reliability-based design optimization of an automotive body structure under crashworthiness constraints ». In : *12<sup>th</sup> International Conference on Application of Statistics and Probability in Civil Engineering*.
- (2016). « Quantile-based optimization under uncertainties using adaptive Kriging surrogate models ». In : *Structural and Multidisciplinary Optimization*, p. 1–19.
- Munoz Zuniga, M., J. Garnier, E. Remy et E. de Rocquigny (2011). « Adaptive directional stratification for controlled estimation of the probability of a rare event ». In : *Reliability Engineering and System Safety* 96.12, p. 1691–1712.
- Nataf, A. (1962). « Détermination des distributions dont les marges sont données ». In : *Comptes Rendus de l'Académie des Sciences* 225, p. 42–43.
- Neddermeyer, J. C. (2009). « Computationally Efficient Nonparametric Importance Sampling ». In : *Journal of the American Statistical Association* 104.486, p. 788–802.
- Nguyen, X.S., A. Sellier, F. Duprat et G. Pons (2009). « Adaptive response surface method based on a double weighted regression technique ». In : *Probabilistic Engineering Mechanics* 24.2, p. 135–143.
- Nie, J. et B.R. Ellingwood (2000). « Directional methods for structural reliability analysis ». In : *Structural Safety* 22.3, p. 233–249.
- Nikolaidis, E. et R. Burdisso (1988). « Reliability Based Optimization : a Safety Index Approach ». In : *Computers and Structures* 28.6, p. 781–788.
-

- Otto, K.N. et E.K. Antonsson (1991). « Extensions to the Taguchi Method of Product Design ». In : *Journal of Mechanical Design* 115, p. 5–13.
- Paffrath, M. et U. Wever (2007). « Adapted polynomial chaos expansion for failure detection ». In : *Journal of Computational Physics* 226.1, p. 263–281.
- Pai, P.-F. (2006). « System reliability forecasting by support vector machines with genetic algorithms ». In : *Mathematical and Computer Modelling* 43.3, p. 262–274.
- Pakhira, M.K., S. Bandyopadhyay et U. Maulik (2004). « Validity index for crisp and fuzzy clusters ». In : *Pattern Recognit.* 37.3, p. 487–501.
- Papadopoulos, V., D.G. Giovanis, N.D. Lagaros et M. Papadrakakis (2012). « Accelerated subset simulation with neural networks for reliability analysis ». In : *Computer Methods in Applied Mechanics and Engineering* 223-224, p. 70–80.
- Papadrakakis, M., N.D. Lagaros et V. Plevris (2005). « Design optimization of steel structures considering uncertainties ». In : *Engineering Structures* 27, p. 1408–1418.
- Papaioannou, I., W. Betz, K. Zwirgmaier et D. Straub (2015). « MCMC algorithms for Subset Simulation ». In : *Probabilistic Engineering Mechanics* 41, p. 89–103.
- Pedroni, N., E. Zio, E. Ferrario, A. Pasanisi et M. Couplet (2013). « Hierarchical propagation of probabilistic and non-probabilistic uncertainty in the parameters of a risk model ». In : *Computers and Structures* 126, p. 199–213.
- Picheny, V., D. Ginsbourger, O. Roustant, R. Haftka et N.-H. Kim (2010). « Adaptive design of experiments for accurate approximation of a target region ». In : *Journal of Mechanical Design* 132 (7), p. 071008.
- Proppe, C. (2008). « Estimation of failure probabilities by local approximation of the limit state function ». In : *Structural Safety* 30.4, p. 277–290.
- Rabitz, H. et Ö. F. Aliş (1999). « General foundations of high-dimensional model representations ». In : *Journal of Mathematical Chemistry* 25.2, p. 197–233.
- Rabitz, H., Ö. F. Aliş, J. Shorter et K. Shim (1999). « Efficient input–output model representations ». In : *Computer Physics Communications* 117.1, p. 11–20.
- Rackwitz, R. (2002). « Optimization and risk acceptability based on the Life Quality Index ». In : *Structural Safety* 24, p. 297–331.
- Rao, S.S. (2009). *Engineering Optimization Theory and Practice*. Sous la dir. de . John Wiley & Sons.
- Rashki, M. (2018). « Hybrid control variates-based simulation method for structural reliability analysis of some problems with low failure probability ». In : *Applied Mathematical Modelling* 60, p. 220–234.
- Rathod, V., O.P. Yadav, A. Rathore et R. Jain (2013). « Optimizing reliability-based robust design model using multi-objective genetic algorithm ». In : *Computers & Industrial Engineering* 66, p. 301–310.
- Razavi, Saman, Bryan A. Tolson et Donald H. Burn (2012). « Review of surrogate modeling in water resources ». In : *Water Resour. Res.* 48.7, p. 1–32.
-



- Rizzi, A.M. (2016). « Support vector regression model for BigData systems ». In : *CoRR* abs/1612.01458.
- Rolander, N., J. Rambo, Y. Joshi, K.K. Allen et F. Mistree (2006). « An Approach to Robust Design of Turbulent Convective Systems ». In : *Journal of Mechanical Design* 128, p. 844–855.
- Rosenblatt, M. (1952). « Remarks on a multivariate transformation ». In : *The Annals of Mathematical Statistics* 23(3), p. 470–472.
- Roussouly, N., F. Petitjean et M. Salaun (2013). « A new adaptive response surface method for reliability analysis ». In : *Probabilistic Engineering Mechanics* 32, p. 103–115.
- Routh, O.U. (2016). « Chapter 8 - Algorithms of solution of singular value decomposition ». In : *Matrix Algorithms in MATLAB*. Academic Press, p. 419–444.
- Rubinstein, R.Y. et D.P. Kroese (2004). *The Cross-Entropy method : a unified approach to combinatorial optimization*. Springer, New York.
- Rubinstein, R.Y. et D.P. Kroese (2007). *Simulation and the Monte Carlo Method (2nd ed.)* New York : Wiley.
- Rubinstein, R.Y. et B. Melamed (1997). *Modern simulation and modeling*. New York : Wiley.
- Sakata, S., F. Ashida et M. Zako (2004). « An efficient algorithm for Kriging approximation and optimization with large-scale sampling data ». In : *Computer Methods in Applied Mechanics and Engineering* 193, p. 385–404.
- Schaffer, J.D. (1985). « Multiple Objective Optimization with Vector Evaluated Genetic Algorithms ». In : *Proceedings of the 1<sup>st</sup> International Conference on Genetic Algorithms*. Hillsdale, NJ, USA : L. Erlbaum Associates Inc., p. 93–100.
- Schetzen, M. (1980). *The Volterra and Wiener Theories of Nonlinear systems*. New York : Wiley.
- Schöbi, R., B. Sudret et S. Marelli (2017). « Rare Event Estimation Using Polynomial-Chaos Kriging ». In : *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A : Civil Engineering* 3.2, p. D4016002.
- Schöbi, R., B. Sudret et J. Wiart (2015). « Polynomial-chaos-based Kriging ». In : *International Journal for Uncertainty Quantification* 5.2, p. 171–193.
- Scholtz, F. (1995). *Tolerance Stack Analysis Methods*. Rapp. tech. Research et Technology Boeing Information & Support Services.
- Schuëller, G.I. et H.A. Jensen (2008). « Computational methods in optimization considering uncertainties – An overview ». In : *Computer Methods in Applied Mechanics and Engineering* 198, p. 2–13.
- Schuëller, G.I., H.J. Pradlwarter et P.S. Koutsourelakis (2004). « A critical appraisal of reliability estimation procedures for high dimensions ». In : *Probabilistic Engineering Mechanics* 19.4, p. 463–474.
-

- Schueremans, L. et D. Van Gemert (2005). « Use of Kriging as meta-model in simulation procedure for structural reliability ». In : *9<sup>th</sup> International Conference on Structural Safety and Reliability (ICOSSAR 2005), Rome*.
- Shahraki, A.F. et R. Noorossama (2014). « Reliability-based robust design optimization : A general methodology using genetic algorithm ». In : *Computers & Industrial Engineering* 74, p. 199–207.
- Shan, S. et G.G. Wang (2010). « Metamodeling for High Dimensional Simulation-based design problems ». In : *ASME Journal of Mechanical Design* 132, p. 051009-1–051009-11.
- Sheikholeslami, G., S. Chatterjee et A. Zhang (1998). « WaveCluster : A Multi-Resolution Clustering Approach for Very Large Spatial Databases ». In : *Proceedings of the 24<sup>th</sup> International Conference on Very Large Data Bases. VLDB '98*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., p. 428–439.
- Smola, A.J. et B. Schölkopf (2004). « A tutorial on support vector regression ». In : *Statistics and Computing* 14.3, p. 199–222.
- Sobol, I. M. (1993). « Sensitivity estimates for non linear mathematical models ». In : *Mathematical Modelling and Comptationnal Experiments* 1, p. 407–414.
- Sokal, R.R. et C.D. Michener (1958). « A statistical method for evaluating system relationships ». In : *University of Kasas Science Bulletin* 38, p. 1409–1438.
- Song, S., Z. Lu et H. Qiao (2009). « Subset simulation for structural reliability sensitivity analysis ». In : *Reliability Engineering & System Safety* 94.2, p. 658–665.
- Sudret, B. (2007). « Uncertainty propagation and sensitivity Analysis in mechanical models Contributions to structural reliability and stochastic spectral methods (Habilitation à Diriger des Recherches) ». Thèse de doct. Université Blaise Pascal – Clermont II.
- (2012). « Meta-models for structural reliability and uncertainty quantification ». In : *HAL*.
- Sudret, B. et A. Der Kiureghian (2000). *Stochastic Finite Element Methods and Reliability A State-of-the-Art Report*. Rapp. tech. Department of civil & environmental engineering university of California, Berkeley.
- Sudret, B. et A. Der Kiureghian (2002). « Comparison of finite element reliability methods ». In : *Probabilistic Engineering Mechanics* 17.4, p. 337–348.
- Tamhane, A.C. (2012). « Experiments for Response Optimization ». In : *Statistical Analysis of Designed Experiments*. Wiley-Blackwell. Chap. 10, p. 395–447.
- Tong, C., Z. Sun, Q. Zhao, Q. Wang et S. Wang (2015). « A hybrid algorithm for reliability analysis combining Kriging and subset simulation importance sampling ». In : *Journal of Mechanical Science and Technology* 29, p. 3183–3193.
- Torii, A. J., R. H. Lopez et L. F. F. Miguel (2017). « A gradient-based polynomial chaos approach for risk and reliability-based design optimization ». In : *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 39.7, p. 2905–2915.
-

- Törn, A. et A. Žilinskas (1989). *Global Optimization – Lecture Notes in Computer Science*.
- Trosset, M.W. (1997). *Taguchi and Robust Optimization*. Rapp. tech. Adjunct Associate Professor, Department of Computational & Applied Mathematics, Rice University, Houston, TX.
- Tyagi, A.K., X. Jonsson, T.G.J. Beelen et W.H.A. Schilders (2017). « Speeding up rare event simulations using Kriging models ». In : *2017 IEEE 21<sup>st</sup> Workshop on Signal and Power Integrity (SPI)*, p. 1–4.
- Uppada, S.K. (2014). « Centroid Based Clustering Algorithms - a clarion study ». In : *International Journal of Computer Science and Information Technologies* 5.6, p. 7309–7313.
- Valdebenito, M.A. et G.I. Schuëller (2010). « A survey on approaches for reliabilitybased optimization. » In : *Structural and Multidisciplinary Optimization* 42.5, p. 1–19.
- Vanmarcke, E.H. (1973). « Matrix formulation of reliability analysis and reliability-based design ». In : *Computers and Structures* 3, p. 757–770.
- Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer.
- Vapnik, V. et A. Lerner (1963). « Pattern recognition using generalized portrait method ». In : *Automation and Remote Control* (24).
- Waarts, P.H (2000). *Structural Reliability using Finite Element Methods*. Delft University Press.
- Wang, H., L. Tang et G.Y. Li (2011). « Adaptive MLS-HDMR metamodeling techniques for high dimensional problems ». In : *Expert Systems with Applications* 38.11, p. 14117–14126.
- Wang, W., J. Yang et R.R. Muntz (1997). « STING : A Statistical Information Grid Approach to Spatial Data Mining ». In : *Proceedings of the 23<sup>rd</sup> International Conference on Very Large Data Bases. VLDB '97*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., p. 186–195.
- Wang, Y., X. Zhao et B. Wang (2013). « LS-SVM and Monte Carlo methods based reliability analysis for settlement of soft clayey foundation ». In : *Journal of Rock Mechanics and Geotechnical Engineering* 5.4, p. 312–317.
- Welch, W.J., R.J. Buck, J. Sacks, H.P. Wynn, T.J. Mitchell et M.D. Morris (1992). « Screening, Predicting, and Computer Experiments ». In : *Technometrics* 34.1, p. 15–25.
- Wen, Z., H. Pei, H. Liu et Z. Yue (2016). « A Sequential Kriging reliability analysis method with characteristics of adaptive sampling regions and parallelizability ». In : *Reliability Engineering & System Safety* 153, p. 170–179.
- Wiener, N. (1938). « The Homogeneous Chaos ». In : *American Journal of Mathematics* 60.4.
- Xia, B., H. Lü, D. Yu et C. Jiang (2015). « Reliability-based design optimization of structural systems under hybrid probabilistic and interval model ». In : *Computers and Structures* 160, p. 126–134.
-

- Xie, X. L. et G. Beni (1991). « A validity measure for fuzzy clustering ». In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13.8, p. 841–847.
- Xiong, B. et H.F. Tan (2018). « A robust and efficient structural reliability method combining radial-based importance sampling and Kriging ». In : *Science China Technological Sciences* 61.5, p. 724–734.
- Xiong, H., Z. Chen, H. Qiu, H. Hao et H. Xu (2012). « Adaptive SVR-HDMR Metamodeling Technique for High Dimensional Problems ». In : *AASRI Procedia* 3, p. 95–100.
- Xiu, D. et G. Karniadakis (2002). « The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations ». In : *SIAM Journal on Scientific Computing* 24.2, p. 619–644.
- Xu, L. (1997). « Bayesian Ying–Yang machine, clustering and number of clusters ». In : *Pattern Recognition Letters* 18.11, p. 1167–1178.
- Yadav, O.P., S.S. Bhamare et A. Rathore (2010). « Reliability-based Robust Design Optimization : A Multi-objective Framework Using Hybrid Quality Loss Function ». In : *Quality and Reliability Engineering International* 26, p. 27–41.
- Yang, X. (2010). « Appendix A : Test Problems in Optimization ». In : *Engineering Optimization*. Wiley-Blackwell, p. 261–266.
- Yao, W., X. Chen, W. Luo, M. van Tooren et J. Guo (2011). « Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles ». In : *Progress in Aerospace Sciences* 47.6, p. 450–479.
- Youn, B.D., K.K. Choi et K. Yi (2005). « Performance Moment Integration (PMI) Method for Quality Assessment in Reliability-Based Robust Design Optimization ». In : *Mechanics Based Design of Structures and Machines* 33, p. 185–213.
- Yuan, X., Z. Lu et Y. Lu (2009). « Support Vector Machine response surface method based on fast Markov chain simulation ». In : *2009 IEEE International Conference on Intelligent Computing and Intelligent Systems*. T. 4, p. 279–282.
- Zadeh, L.A. (1978). « Fuzzy Sets as a basis for a theory of possibility ». In : *Fuzzy Sets and Systems* 1, p. 3–28.
- Zaman, K., M. McDonald, S. Mahadevan et L. Green (2011). « Robustness-based design optimization under data uncertainty ». In : *Structural and Multidisciplinary Optimization* (44), p. 183–197.
- Zang, C., M.O. Friswell et J.E. Mottershead (2005). « A review of robust optimal design and its application in dynamics ». In : *Computers and Structures* 83, p. 315–326.
- Zhang, H. et Q. Ni (2015). « A new regularized quasi-Newton algorithm for unconstrained optimization ». In : *Applied Mathematics and Computation* 259, p. 460–469.
- Zhang, P. (1996). « Nonparametric Importance Sampling ». In : *Journal of the American Statistical Association* 91.435, p. 1245–1253.
-

- Zhang, T., R. Ramakrishnan et M. Livny (1997). « BIRCH : A New Data Clustering Algorithm and Its Applications ». In : *Data Mining and Knowledge Discovery* 1.2, p. 141–182.
- Zuev, K.M., J.L. Beck, S.-K. Au et L.S. Katafygiotis (2012). « Bayesian post-processor and other enhancements of Subset Simulation for estimating failure probabilities in high dimensions ». In : *Computers and Structures* 92-93, p. 283–296.
-

