



# Couplage optimisation à convergence partielle et stratégie multiparamétrique en calcul de structures

Nicolas Courrier

► **To cite this version:**

Nicolas Courrier. Couplage optimisation à convergence partielle et stratégie multiparamétrique en calcul de structures. Mécanique des solides [physics.class-ph]. Université Paris-Saclay, 2015. Français. <NNT : 2015SACLN026>. <tel-01264667>

**HAL Id: tel-01264667**

**<https://tel.archives-ouvertes.fr/tel-01264667>**

Submitted on 29 Jan 2016

**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.

**THESE DE DOCTORAT  
DE L'UNIVERSITE PARIS-SACLAY  
préparée à l'Ecole Normale Supérieure de CACHAN**

**ECOLE DOCTORALE N° 579**  
Sciences mécaniques et énergétiques, matériaux et géosciences - SMEMAG

**Spécialité de doctorat : Mécanique des solides**

Par

**M. Nicolas COURRIER**

**Couplage optimisation à convergence partielle et stratégie  
multiparamétrique en calcul de structures**

Thèse présentée à Cachan le 8 décembre 2015 devant le jury composé de :

Bruno Sudret	Professeur, École polytechnique fédérale de Zurich	Président de jury
Rodolphe Le Riche	DR CNRS, HDR, EMSE	Rapporteur
Nicolas Gayton	Maître de conférences, HDR, IFMA	Rapporteur
Sébastien Da Veiga	Docteur, Ingénieur de Recherche, Safran Tech	Examineur
Pierre-Alain Boucard	Professeur, ENS de Cachan	Directeur de thèse
Bruno Soulier	Maître de conférences, ENS de Cachan	Co-encadrant

**LMT-Cachan**

ENS Cachan / CNRS / Université Paris-Saclay  
61 avenue du Président Wilson, F-94235 Cachan cedex, France



Bien ces lignes soient les dernières que j'écris, elles sont importantes pour moi afin de remercier toutes les personnes sur qui j'ai pu compter durant ces trois années de thèse.

Je tiens avant toute chose à remercier M. Sudret Bruno, qui a accepté d'être président de mon jury de thèse ainsi que M. Da Veiga Sébastien d'avoir été l'un des examinateurs. Je remercie également chaleureusement MM. Leriche Rodolphe et Gayton Nicolas d'avoir accepté d'être les rapporteurs de ce manuscrit, de toute l'attention qu'ils ont pu y porter et de leurs conseils. Je tiens bien sûr à remercier Pierre-Alain Boucard et Bruno Soulier qui m'ont permis de réaliser cette thèse et de m'avoir laissé une réelle liberté de recherche ce qui m'a permis de passer trois années extrêmement plaisantes au sein du laboratoire. Je les remercie de leur soutien et de leurs nombreux conseils.

Je voudrais remercier dans un second temps le laboratoire du LMT Cachan qui m'a accueilli. J'ai pu y rencontrer, au fur et à mesure de mes années d'études, des gens tout à fait intéressants et intéressés par leurs travaux, leurs recherches et leurs enseignements. Ces personnes m'ont largement inspiré dans mes choix. Je tiens donc à remercier Laurent C., Jean-Luc N., Bruno S., David N., Ludovic C., Peewee, Michel et Jean-Jacques, Jean-Marie V. et bien sûr PAB. Je suis sûr d'en oublier mais ceux-ci me pardonneront largement car ils savent parfaitement ce que je leur dois.

Je remercie également l'ensemble des doctorants que j'ai eu l'occasion de côtoyer : Werty, Elena, Pierre N, Amaury C., Hadrien B., H, MF, Guillaume, Pallier, Paul, Maxime, Luc (à qui je dois d'avoir récupéré un code robuste et simple d'utilisation qui a donc simplifié mon travail), Jérémy M., Christophe H., Claire, Alexis. Je ne peux pas (comment le pourrais-t-on ?) oublier Valentine avec qui j'ai partagé mon bureau pendant trois ans et avec qui j'ai passé de vrais moments de rire et de travail. Je la remercie pour sa relecture quasi complète de ce manuscrit. Je remercie également l'ensemble des gens travaillant au laboratoire et nous aidant avec gentillesse et plaisir dans l'utilisation des différents outils disponibles : Philippe, Pierre, Jean-Christophe, Samy, Hugo, Frisou, Boubou, Rémy.

Je remercie de tout cœur, Danielle, Christine V., Christine B., Lydia M., et Stéphanie H. pour l'ensemble des conseils qu'elles ont pu me fournir durant ces sept années à l'ENS. Je les remercie surtout pour leur bienveillance et leur gentillesse et de leur précieuse aide lorsque que j'ai pu me perdre dans quelques papiers administratifs. Je les remercie aussi pour l'ensemble des bons moments qu'on a pu avoir tous les midis en se retrouvant pour manger. Je sais déjà que ces moments passer avec elles, Cédric, Laëticia, Karine, PAG, Peewee et Emeline vont me manquer.

Finalement, je remercie ma famille. Tout d'abord mes parents qui m'ont toujours soutenu et aidé dans mes études. Je tiens à remercier ma mère pour sa relecture complète du manuscrit et de la correction que nous avons réalisée ensemble. Je remercie également mes beaux-parents pour leur soutien dans les moments difficiles de cette thèse et de leurs conseils tout au long de mes années d'étude (et notamment au moment de passer le concours de l'agrégation). Il ne me reste plus qu'à remercier une personne. Celle-ci est l'une des plus importantes, il s'agit tout simplement de Noëllie qui m'accompagne depuis plusieurs années et qui a accepté de me rejoindre sur Paris, de me voir rentrer tard le soir, de me voir râler quand mes calculs ne tournaient pas, de me voir repartir au laboratoire les week-end. Je sais que je lui dois aussi beaucoup dans l'accomplissement de ce travail.



# Table des matières

<b>Table des matières</b>	<b>i</b>
<b>Table des figures</b>	<b>vii</b>
<b>Liste des tableaux</b>	<b>xi</b>
<b>Introduction</b>	<b>1</b>
<b>1 Contexte et outils numériques</b>	<b>7</b>
1 Introduction aux problèmes d'optimisation et contexte scientifique de l'étude . . .	9
1.1 Introduction . . . . .	9
1.2 Définition du problème d'optimisation . . . . .	9
1.3 Différentes optimisations . . . . .	11
1.3.1 Optimisation de forme . . . . .	11
1.3.2 Optimisation paramétrique . . . . .	12
1.4 Contexte général de l'étude . . . . .	13
1.4.1 Contexte industriel . . . . .	13
1.4.2 Difficulté de l'optimisation . . . . .	14
1.4.3 Optima locaux et globaux . . . . .	14
1.4.4 Quelle stratégie ? . . . . .	16
1.4.5 L'utilisation d'un métamodèle . . . . .	17
1.4.6 Présentation de la méthode utilisée au LMT . . . . .	19
2 Algorithmes d'optimisation . . . . .	22
2.1 Algorithmes d'optimisation locale sans contrainte . . . . .	22
2.1.1 Conditions d'optimalité . . . . .	22
2.1.2 Quelques algorithmes . . . . .	22
2.1.3 Méthode à région de confiance . . . . .	23
2.2 Algorithmes d'optimisation locale sous contraintes . . . . .	27
2.2.1 Méthodes de pénalisation . . . . .	27
2.2.2 Méthodes Lagrangiennes . . . . .	27
2.2.3 Conditions d'optimalité . . . . .	27
2.2.4 Méthode sequential quadratic programming (SQP) . . . . .	28
2.2.5 Méthode des points intérieurs . . . . .	30

2.3	Algorithmes d'optimisation globale . . . . .	30
2.3.1	Introduction . . . . .	30
2.3.2	Algorithmes génétiques . . . . .	31
2.3.3	Optimisation par essaims de particules . . . . .	33
3	Utilisation de métamodèle . . . . .	35
3.1	Construction de métamodèle par krigeage . . . . .	36
3.1.1	Définitions . . . . .	36
3.1.2	Formulation du krigeage . . . . .	38
3.1.3	Précision sur la recherche des paramètres du krigeage . . . . .	41
3.2	Qualité du métamodèle . . . . .	43
3.2.1	EMC, REMC . . . . .	43
3.2.2	$r^2$ . . . . .	43
3.2.3	$r_{ccc}$ . . . . .	44
3.2.4	Erreur relative moyenne absolue (RAAE) . . . . .	44
3.2.5	Erreur relative maximale absolue (RMAE) . . . . .	45
3.2.6	Quelques erreurs normées . . . . .	45
3.3	Optimisation et métamodèle : Stratégie EGO . . . . .	45
3.4	Bilan . . . . .	47
4	La méthode LaTIn . . . . .	48
4.1	Présentation du problème mécanique . . . . .	50
4.1.1	Problème mécanique de référence . . . . .	50
4.1.2	Problème mécanique pour les assemblages . . . . .	51
4.2	Résolution par la méthode LaTIn . . . . .	52
4.2.1	Présentation générale de la méthode . . . . .	52
4.2.2	Problème dans les sous-domaines . . . . .	54
4.2.3	Problème aux interfaces . . . . .	54
4.2.4	Algorithme LaTIn . . . . .	55
4.2.5	Quelques précisions sur le solveur . . . . .	57
4.2.6	Indicateur d'erreur globale . . . . .	58
4.3	Approche multiparamétrique . . . . .	59
4.3.1	L'approche multiparamétrique dans le cadre des assemblages . . . . .	59
4.3.2	Mise en place de la stratégie . . . . .	59
5	Approches multi-fidélité . . . . .	60
5.1	Introduction . . . . .	60
5.2	Précisions sur le vocabulaire utilisé . . . . .	61
5.3	Optimisation par approches multi-niveaux . . . . .	63
5.3.1	Méthode progressive . . . . .	64
5.3.2	Méthode réduction d'espace . . . . .	64
5.4	Modélisations multi-fidélité . . . . .	65
5.4.1	Méthode de correction de surface . . . . .	65
5.4.2	Cokrigeage et krigeage hiérarchique . . . . .	66
5.5	Assimilation aux données bruitées . . . . .	67
6	Bilan du chapitre . . . . .	69

<b>2</b>	<b>Méthode Evofusion pour l'approximation globale et l'optimisation</b>	<b>71</b>
1	Présentation de la problématique . . . . .	72
1.1	Introduction . . . . .	72
1.2	Contexte de l'étude de faisabilité . . . . .	73
1.2.1	Approche avec des données partiellement convergées . . . . .	73
1.2.2	Stratégie d'initialisation . . . . .	75
1.2.3	Description de l'étude effectuée . . . . .	75
1.3	Présentation des cas test traités . . . . .	78
1.3.1	Cas test <i>cubes</i> . . . . .	78
1.3.2	Cas test <i>frette</i> . . . . .	79
2	Données et quantités d'intérêt de l'étude de faisabilité menée . . . . .	80
2.1	Approche multiparamétrique appliquée aux données partiellement convergées . . . . .	81
2.2	Présentation des quantités d'intérêt étudiées pour la stratégie de recalage	83
2.3	Présentation des quantités d'intérêt étudiées pour la stratégie de recherche du minimum . . . . .	85
2.3.1	Modification de la stratégie EGO par <i>weighted expected improvement</i> . . . . .	86
2.3.2	Définitions des gains . . . . .	87
2.3.3	Remarque . . . . .	89
3	Traitement des cas test . . . . .	89
3.1	Etude du cas test <i>cubes</i> . . . . .	89
3.1.1	Stratégie de recalage : cas test <i>cubes</i> . . . . .	90
3.1.2	Stratégie de recherche du minimum : cas test <i>cubes</i> . . . . .	94
3.2	Etude du cas test <i>frette</i> . . . . .	99
3.2.1	Stratégie de recalage : cas test <i>frette</i> . . . . .	99
3.2.2	Stratégie de recherche du minimum : cas test <i>frette</i> . . . . .	100
4	Bilan de l'étude de faisabilité . . . . .	102
<b>3</b>	<b>Comparaison de métamodèles multi-fidélité pour l'approximation globale</b>	<b>103</b>
1	Différentes stratégies de modélisation multi-fidélité . . . . .	104
1.1	Equations générales du cokrigeage . . . . .	105
1.1.1	Remarques . . . . .	108
1.1.2	Preuve de l'interpolation du métamodèle . . . . .	109
1.2	Méthodes de cokrigeage . . . . .	110
1.2.1	Méthode de cokrigeage multi-fidélité, notée méthode « Inter-corrélation » . . . . .	110
1.2.2	Détermination des paramètres du cokrigeage Intercorrélation . . . . .	111
1.2.3	Méthode de cokrigeage multi-fidélité, notée « $s_{fcv}/s_{pcv}$ » . . . . .	112
1.2.4	Détermination des paramètres du cokrigeage $s_{fcv}/s_{pcv}$ . . . . .	114
1.2.5	Méthode de cokrigeage multi-fidélité de type auto-régressif . . . . .	115
1.2.6	Détermination des paramètres du métamodèle de cokrigeage de type auto-régressif . . . . .	117



1.3	Krigeage hiérarchique . . . . .	118
1.3.1	Détermination des paramètres . . . . .	119
1.4	Illustration des stratégies multi-fidélité . . . . .	119
1.4.1	Cokrigeage : approche par la méthode de cokrigeage Inter- corrélation . . . . .	119
1.4.2	Cokrigeage : approche par la méthode de cokrigeage $s_{fcv}/s_{pcv}$ et auto-régressif . . . . .	120
1.4.3	Krigeage hiérarchique . . . . .	122
1.4.4	Bilan intermédiaire . . . . .	123
2	Etudes comparatives sur des exemples mécaniques . . . . .	123
2.1	Introduction . . . . .	123
2.2	Choix de points totalement convergés . . . . .	124
2.3	Stratégie de recalage avec cokrigeage et krigeage hiérarchique . . . . .	126
2.3.1	Présentation de l'algorithme . . . . .	126
2.3.2	Critères d'arrêt d'enrichissement . . . . .	127
2.4	Etude du cas test <i>cubes</i> . . . . .	128
2.4.1	Qualité du métamodèle construit avec des données unique- ment partiellement convergées . . . . .	129
2.4.2	Qualité du métamodèle multi-fidélité initial . . . . .	130
2.4.3	Qualité des métamodèles multi-fidélité lors de la phase d'enri- chissement . . . . .	131
2.4.4	Comparaison des méthodes multi-fidélité pour atteindre $r_{ccc} >$ 0.95 . . . . .	135
2.4.5	Comparaison des méthodes multi-fidélité pour atteindre $\bar{r}_{ccc}^{v^3}$ . . . . .	137
2.4.6	Conclusion de l'étude . . . . .	139
2.5	Etude du cas test <i>1_rainure</i> . . . . .	139
2.5.1	Qualité du métamodèle construit avec des données unique- ment partiellement convergées . . . . .	142
2.5.2	Qualité du métamodèle multi-fidélité initial . . . . .	142
2.5.3	Qualité du métamodèle multi-fidélité en fonction du coût de calcul . . . . .	143
2.5.4	Comparaison des méthodes multi-fidélité pour atteindre $r_{ccc} >$ 0.95 . . . . .	144
2.5.5	Comparaison des méthodes multi-fidélité pour atteindre $\bar{r}_{ccc}^{v^3}$ . . . . .	146
3	Bilan de l'étude comparative des méthodes de construction de métamodèles . . . . .	147
<b>4</b>	<b>Utilisation des approches multi-niveaux et multi-fidélité pour l'optimisation globale</b>	<b>151</b>
1	Critère d'arrêt de la stratégie EGO . . . . .	152
1.1	Introduction . . . . .	152
1.2	Différents critères d'arrêt . . . . .	153
1.2.1	Critère d'arrêt de référence . . . . .	153
1.2.2	Un deuxième critère d'arrêt . . . . .	154
1.2.3	Un dernier critère d'arrêt . . . . .	154

1.3	Comparaison sur des fonctions tests analytiques . . . . .	155
1.3.1	Comparaison sur des fonctions Rosenbrock . . . . .	155
1.3.2	Cas test sur les fonctions Hartmann & Goldstein . . . . .	158
2	Amélioration du tirage initial pour la construction de métamodèle . . . . .	161
2.1	Méthode d'échantillonnage adaptatif avec un métamodèle de krigeage . . . . .	162
2.2	Influence du tirage pour un métamodèle de cokrigeage . . . . .	164
3	Optimisation d'un cas test mécanique : premier niveau d'optimisation . . . . .	168
3.1	Présentation du cas test <b>2_rainures</b> . . . . .	168
3.2	Traitement du cas test <b>2_rainures</b> . . . . .	170
3.2.1	Méthodes de métamodélisation utilisées sur le cas test . . . . .	170
3.2.2	Formulation de l'expected improvement pour les méthodes multi-fidélité . . . . .	171
3.2.3	Résultats de l'étude . . . . .	171
3.2.4	Bilan de l'étude d'optimisation à l'aide de métamodèles multi-fidélité . . . . .	178
3.2.5	Comparatif des résultats entre la méthode de référence et la méthode de cokrigeage . . . . .	178
4	Optimisation d'un cas test mécanique : second niveau d'optimisation . . . . .	181
4.1	Stratégie d'optimisation locale basée sur les méthodes à région de confiance . . . . .	181
4.1.1	Méthodes à région de confiance avec fonctions modèles générales . . . . .	181
4.1.2	Application sur des fonctions tests analytiques . . . . .	182
4.2	Mise en place du second niveau d'optimisation sur le cas test . . . . .	184
4.2.1	Introduction . . . . .	184
4.2.2	Critères d'arrêt de l'algorithme d'optimisation locale . . . . .	184
4.2.3	Résultats de l'optimisation locale sur le cas test <b>2_rainures</b> . . . . .	185
4.3	Bilan de la stratégie d'optimisation multi-niveaux sur le cas test <b>2_rainures</b> . . . . .	186
5	Bilan . . . . .	187
<b>5</b>	<b>Applications sur des cas test industriels et perspectives de développement</b> . . . . .	<b>189</b>
1	Optimisation sous contrainte d'un assemblage boulonné . . . . .	190
1.1	Présentation du cas test . . . . .	190
1.2	Présentation de la problématique traitée . . . . .	192
1.2.1	Présentation du problème d'optimisation . . . . .	192
1.2.2	Précisions sur le calcul numérique . . . . .	192
1.2.3	Surfaces de réponse associées au problème d'optimisation traité . . . . .	193
1.3	Enrichissement avec prise en compte des contraintes . . . . .	195
1.4	Réalisation de l'optimisation du cas test . . . . .	196
1.4.1	Descriptif des stratégies comparées . . . . .	196
1.4.2	Résultats . . . . .	197
1.4.3	Bilan de l'étude . . . . .	201

2	Création d'un abaque virtuel . . . . .	201
2.1	Introduction . . . . .	201
2.2	Présentation du cas test . . . . .	201
2.3	Adaptations des outils disponibles pour le calcul . . . . .	204
2.3.1	Parallélisation des moyens de calcul . . . . .	204
2.3.2	Toolbox d'optimisation multi-optima . . . . .	204
2.3.3	Stratégie multiparamétrique et parallélisme . . . . .	206
2.4	Construction de l'abaque virtuel . . . . .	206
2.4.1	Définition du critère d'arrêt . . . . .	207
2.4.2	Résultats obtenus . . . . .	207
2.5	Bilan . . . . .	208
3	Développements supplémentaires et possibilités offertes par la toolbox GRENAT	209
3.1	Construction de métamodèle multi-fidélité avec prise en compte des gra- dients . . . . .	209
3.1.1	Cokrigeage à gradients . . . . .	209
3.1.2	Cokrigeage multi-fidélité avec prise en compte des gradients . . . . .	213
3.1.3	Krigeage hiérarchique à gradients . . . . .	214
3.1.4	Evofusion avec prise en compte des gradients . . . . .	215
3.2	Illustrations des performances atteintes sur deux cas test mécaniques . . . . .	217
	<b>Conclusions et perspectives</b>	<b>219</b>
	<b>Bibliographie</b>	<b>223</b>

# Table des figures

1.1	Différents types d'optimisation de forme [Duysinx, 1996] . . . . .	12
1.2	Illustration minima globaux et locaux . . . . .	15
1.3	Fonction $y(x) = \exp(-x/10) \cos(x) + 0.1x$ sur $[-15, 15]$ . . . . .	16
1.4	Exemple de choix d'algorithmes en fonction des caractéristiques du problème [Barton et Meckesheimer, 2006] . . . . .	16
1.5	Description optimisation par métamodèle . . . . .	18
1.6	Présentation générale de la méthode d'optimisation . . . . .	19
1.7	Illustration de l'algorithme à région de confiance . . . . .	26
1.8	Présentation générale de la méthode d'optimisation par algorithmes génétiques . . . . .	32
1.9	Croisement uniforme . . . . .	33
1.10	Schéma d'évolution d'une particule . . . . .	34
1.11	Exemples de différents tirages possibles sur un espace à 2 paramètres . . . . .	36
1.12	Comparaison des deux coefficients utilisés . . . . .	44
1.13	Différentes itérations de l'algorithme EGO, enrichissement par EI . . . . .	48
1.14	Représentation schématique de l'approche des méthodes incrémentales et de la méthode LaTIn . . . . .	50
1.15	Problème de référence . . . . .	51
1.16	Problème de référence d'un assemblage . . . . .	52
1.17	Décomposition de domaine de la méthode LaTIn et grandeurs prises en compte au niveau des interfaces . . . . .	53
1.18	Illustration de l'algorithme itératif LaTIn . . . . .	56
1.19	Illustration de l'indicateur global . . . . .	58
1.20	Illustration de l'approche multiparamétrique . . . . .	60
1.21	Problème d'assemblage n°1 . . . . .	61
1.22	Surfaces de réponses obtenues pour différents niveaux de convergence de l'algorithme LaTIn . . . . .	62
1.23	Fonction $\Phi^{-1}(\zeta)$ pour $\zeta = [0.5, 1[$ . . . . .	68
2.1	Fonction objectif représentée pour les deux niveaux de fidélité . . . . .	73
2.2	Principe illustré de l'EvoFusion . . . . .	74
2.3	Algorithme de l'EvoFusion . . . . .	76
2.4	Stratégie recalage : . . . . .	77
2.5	Premier exemple traité : Cas test <i>cubes</i> . . . . .	78

2.6	Second exemple traité : Cas test <i>frette</i> . . . . .	79
2.7	Illustrations de l'approche multiparamétrique LaTin pour le calcul des données partiellement convergées . . . . .	81
2.8	Comparaison de stratégies multiparamétriques de calcul de données partiellement convergées . . . . .	82
2.9	Surfaces de réponse pour le cas test <i>cubes</i> avec $\mu_{Str2/Str4} = 0.1$ . . . . .	86
2.10	Illustration du WEI montrant l'impact du choix de $\omega$ . . . . .	87
2.11	Algorithme associé à la méthode de référence de recherche de minimum de type EGO . . . . .	88
2.12	Comparaison entre la surface de réponse de référence et la surface de réponse construite avec des données partiellement convergées pour le cas test <i>cubes</i> . . . . .	89
2.13	Corrélation $r_{ccc}$ initiale entre le métamodèle construit avec des données partiellement convergées et le métamodèle de référence pour le cas test <i>cubes</i> . . . . .	91
2.14	Temps de calcul moyen pour le cas test <i>cubes</i> . . . . .	91
2.15	Gain temps de calcul pour le cas test <i>cubes</i> . . . . .	92
2.16	Ecart-type relatif des temps de calcul pour le cas test <i>cubes</i> . . . . .	92
2.17	Corrélation $r^2$ initiale . . . . .	95
2.18	Temps de calcul moyen avec la stratégie d'Evofusion . . . . .	96
2.19	Ecart-type relatif des temps de calcul pour le cas test <i>cubes</i> . . . . .	96
2.20	$Gain_1$ de calcul . . . . .	97
2.21	$Gain_2$ de calcul . . . . .	98
2.22	Ecart-type relatif associé au $Gain_2$ pour les approches de calcul multiparamétrique et sans multiparamétrique . . . . .	98
2.23	Corrélation $r_{ccc}$ entre le métamodèle initial et le métamodèle de référence . . . . .	99
2.24	Résultats de la stratégie de recalage appliquée au cas test <i>frette</i> . . . . .	100
2.25	Evolution de la corrélation $r_{ccc}$ pour 50 points initiaux partiellement convergés . . . . .	101
2.26	Gains de calcul . . . . .	101
3.1	Illustration du Cokrigeage Intercorrélation avec $A = 0.5$ . . . . .	120
3.2	Illustration du Cokrigeage Intercorrélation sur le second exemple . . . . .	121
3.3	Résultats obtenus à l'aide du cokrigeage $\sigma_{fcv}/\sigma_{pcv}$ . . . . .	121
3.4	Illustration du Cokrigeage auto-régressif . . . . .	122
3.5	Illustration du krigeage hiérarchique . . . . .	122
3.6	Illustration d'un tirage LHS de 30 points et d'une sélection de 9 points par l'algorithme d'échange . . . . .	126
3.7	Algorithme associé aux stratégies de cokrigeage et de krigeage hiérarchique . . . . .	126
3.8	Qualité moyenne des métamodèles construits avec les données partiellement convergées pour le cas test <i>cubes</i> . . . . .	129
3.9	Qualité des métamodèles multi-fidélité pour le 1 <sup>er</sup> cas test . . . . .	130
3.10	Comparaison de la qualité des métamodèles avec l'approche d'Evofusion (rouge) par rapport à celle obtenue avec les métamodèles de krigeage construits uniquement avec les données totalement convergées (bleu) . . . . .	132

3.11	Comparaison de la qualité des métamodèles avec l'approche de Cokrigage Intecorrélation (en rouge) par rapport à celle obtenue avec les métamodèles de krigeage construits uniquement avec les données totalement convergées (en bleu)	132
3.12	Comparaison de la qualité des métamodèles avec l'approche de Cokrigage $s_{fcv}/s_{pcv}$ (en rouge) par rapport à celle obtenue avec les métamodèles de krigeage construits uniquement avec les données totalement convergées (en bleu)	133
3.13	Comparaison de la qualité des métamodèles avec l'approche de cokrigage auto-régressif (en rouge) par rapport à celle obtenue avec les métamodèles de krigeage construits uniquement avec les données totalement convergées (en bleu)	133
3.14	Comparaison de la qualité des métamodèles avec l'approche de krigeage hiérarchique (en rouge) par rapport à celle obtenue avec les métamodèles de krigeage construits uniquement avec les données totalement convergées (en bleu)	134
3.15	Comparaison $r_{ccc}$ pour les différentes méthodes	134
3.16	Comparaison des méthodes pour atteindre le critère de corrélation $r_{ccc} \geq 0.95$ pour le cas test <i>cubes</i>	135
3.17	Comparaison des méthodes pour atteindre le critère d'arrêt $\bar{r}_{ccc}^3$ pour le cas test <i>cubes</i>	137
3.18	Corrélation $r_{ccc}$ obtenue entre les métamodèles multi-fidélité finaux et le métamodèle de référence	138
3.19	Schéma cas test 1_rainure	140
3.20	Eclaté et maillage du cas test 1_rainure	141
3.21	Surface de réponse en fonction des paramètres $X$ et $L$	141
3.22	Qualité moyenne des métamodèles construits avec les données partiellement convergées pour le cas test 1_rainure	142
3.23	Qualité du métamodèle multi-fidélité et du métamodèle de référence pour le cas test 1_rainure	143
3.24	Comparaison $r_{ccc}$ pour les différentes méthodes	144
3.25	Qualité des métamodèles multi-fidélité (en rouge) par rapport à celle obtenue avec les métamodèles construits uniquement avec les données totalement convergées (en bleu)	144
3.26	Comparaison des méthodes pour atteindre le critère de corrélation $r_{ccc} \geq 0.95$ pour le cas test 1_rainure	145
3.27	Comparaison des méthodes pour atteindre le critère d'arrêt $\bar{r}_{ccc}^3$ pour le cas test 1_rainure	146
3.28	Corrélation $r_{ccc}$ obtenue entre les métamodèles multi-fidélité finaux et le métamodèle de référence	147
4.1	Représentation du cas test 2_rainures	169
4.2	Valeur du paramètre $L$ en fonction de $X_1$ et $X_2$	169
4.3	Surface de réponse en fonction des paramètres $X_1$ et $X_2$	170
4.4	Corrélation $r^2$ pour les différentes méthodes et tirages	172
4.5	Nombre d'essais numériques aboutissant à localiser la zone du minimum global	173
4.6	Qualité du minimum trouvé par les métamodèles multi-fidélité	174

4.7	Nombre de points totalement convergés calculés lorsque les différents critères d'arrêt ont été atteints . . . . .	175
4.8	Temps de calcul CPU nécessaire pour atteindre les critères d'arrêt . . . . .	176
4.9	Précision moyenne sur le minimum trouvé par les métamodèles en fonction du coût de calcul . . . . .	177
4.10	Algorithme à région de confiance . . . . .	183
5.1	Joint d'accouplement (figures issus de [Laurent, 2013]) . . . . .	191
5.2	Convergence du calcul de la contrainte maximale des vis en fonction du niveau d'indicateur $\eta_{fcv}^{(2)}$ . . . . .	193
5.3	Surface de réponse représentant le couple transmissible $M_t$ (en N.m) . . . . .	194
5.4	Surface de réponse représentant le maximum de la contrainte de Tresca dans les vis (en MPa) . . . . .	194
5.5	Couple transmissible dans la zone d'admissibilité . . . . .	195
5.6	Illustration du constrained expected improvement . . . . .	197
5.7	Historique de convergence des deux méthodes comparées . . . . .	198
5.8	Maillage 1/4 de la bride aéronautique . . . . .	202
5.9	Algorithme avec utilisation de solveurs parallèles . . . . .	205
5.10	Métamodèle de cokrigeage multi-fidélité à gradients . . . . .	214
5.11	Comparaison des estimateurs et de cokrigeage avec des gradients choisis . . . . .	216
5.12	Métamodèle évofusé avec ou sans prise en compte des gradients . . . . .	216

# Liste des tableaux

1.1	Développements apportés au cours de cette thèse . . . . .	21
1.2	Récapitulatif de certaines fonctions de corrélation usuelles pour le krigeage . . . . .	40
2.1	Bornes de conception pour le cas test <i>cubes</i> . . . . .	79
2.2	Bornes de conception pour le cas test <i>frette</i> . . . . .	80
2.3	Tableau récapitulatif des indicateurs de qualité de réponse . . . . .	93
2.4	Nombre de points totalement convergés ajoutés en fonction de la précision des données partiellement convergées . . . . .	93
3.1	Bornes de conception pour le cas test 1_ <i>rainure</i> . . . . .	140
4.1	Réapitulatif des résultats obtenus par [Laurent, 2013] . . . . .	153
4.2	Tableau des résultats de la première étape et de la seconde étape d'optimisation pour les fonctions Rosenbrock . . . . .	156
4.3	Comparaison récapitulative des résultats obtenus pour les fonctions de Rosenbrock	158
4.4	Tableau des résultats de la première étape et de la seconde étape d'optimisation pour les fonctions de Goldstein et Hartmann . . . . .	160
4.5	Comparaison récapitulative des résultats obtenus pour les fonctions de Goldstein et Hartmann . . . . .	160
4.6	Résultats de l'étude comparative de la qualité de métamodèle obtenue pour les différentes fonctions tests . . . . .	163
4.7	Stratégies de tirage testées au cours de ce paragraphe et de cette étude . . . . .	165
4.8	Résultats de la stratégie de recalage en fonction des tirages T1, T2, et T3 . . . . .	166
4.9	Comparaison des résultats de la stratégie de recalage en fonction des tirages T1, T2, T4 et T5 . . . . .	166
4.10	Bornes des variables de conception du cas test 2_ <i>rainures</i> . . . . .	168
4.11	Comparatif des performances atteintes pour les deux méthodes lors du premier niveau d'optimisation . . . . .	179
4.12	Comparaison des coûts de calcul de chacune des méthodes . . . . .	184
4.13	Comparatif des performances atteintes pour les deux méthodes lors du second niveau d'optimisation . . . . .	185
4.14	Comparatif des performances atteintes pour les deux méthodes testées pour l'optimisation multi-niveaux complètement réalisée . . . . .	186



---

5.1	Bornes de conception pour le cas test du joint d'accouplement . . . . .	192
5.2	Tableau des résultats obtenus pour les deux approches de construction de métamodèle . . . . .	199
5.3	Bornes de conception pour le cas test de la bride . . . . .	203
5.4	Résultats obtenus avec les deux méthodes de construction de métamodèle . . . .	207
5.5	Résultats de la qualité des méthodes multi-fidélité avec prise en compte ou non des gradients pour le cas test <i>cubes</i> . . . . .	217
5.6	Résultats de la qualité des méthodes multi-fidélité avec prise en compte ou non des gradients pour le cas test <i>2_rainures</i> . . . . .	217

# Introduction

Les nombreux outils industriels de simulation développés au cours de ces dernières années ont encore un impact limité sur la conception des systèmes complexes. Alors que les outils de modélisation de chaque discipline ou sous-systèmes se raffinent graduellement, l'optimisation globale de ces systèmes se heurte à des difficultés d'ordre à la fois méthodologiques, organisationnelles, informatiques et numériques. On peut citer le volume des calculs trop élevé, la collaboration difficile entre les services qui maîtrisent la simulation de chaque discipline, dont ces mêmes simulations peuvent être entachées d'erreurs numériques plus ou moins maîtrisées...

Ces différentes difficultés sont bien présentes dans le cadre des assemblages, où des incertitudes sur le comportement et les interfaces des pièces continuent de subsister et posent des problèmes lors de la phase de conception en bureau d'étude. Ces problèmes sont variés selon les domaines et les échelles de conception. En aéronautique, par exemple, la problématique de liaison boulonnée entre deux plaques composites est de plus en plus répandue par l'usage de ces matériaux. Les différents mécanismes mis en jeu, ceux du matériau en lui-même et ceux d'une liaison boulonnée adaptée, sont mal connus. Cet exemple en est un parmi d'autres, la gestion du nombre de boulons en vue de limiter la masse des assemblages est également une problématique importante. Afin de mieux appréhender les problématiques liées aux assemblages, différents projets de recherche se développent, on citera notamment le projet CARAB (Conception Avancée Robuste pour les Assemblages Boulonnés) lancé en 2012 auquel participe le LMT-Cachan et également des entreprises telles que SAFRAN.

Le développement des outils numériques pour aider à la conception est ainsi de plus en plus développé en vue de réduire les coûts financiers et humains que peut engendrer la mise en place d'essais expérimentaux. La simulation numérique permet d'aider à la conception d'assemblages, mais comme dit précédemment celle-ci est encore limitée. Une des principales raisons de cette limitation est le coût de calcul numérique engendré par le traitement de ces problèmes. Ceux-ci sont fortement non-linéaires à cause de la présence du frottement entre les pièces, de la gestion du contact et du jeu entre elles. En plus du coût de calcul, sa précision est souvent mal connue et maîtrisée. Cependant, le développement de méthodes numériques (grâce à l'augmentation des puissances de calcul) est de plus en plus présent au sein des entreprises et des laboratoires. En effet, dans une logique de gestion et de développement économique, d'investissement ou de conception robuste, ces méthodes sont de plus en plus utilisées malgré les différents verrous décrits précédemment. Les difficultés numériques et les temps de calcul importants peuvent alors expliquer que différents services au sein d'un bureau d'étude se focalisent sur des pièces isolées de l'assemblage, posant alors naturellement des problèmes organisationnels mais également conceptuels. Par exemple, il est souvent compliqué d'appliquer des conditions limites en effort

ou en déplacement, représentatives des conditions de l'assemblage.

Pour un même assemblage, différentes phases de vie existent apportant chacune des exigences différentes, ceci allant de la phase de montage à la phase de rebut de l'assemblage. Le cycle d'exploitation comporte également différentes phases de vie où des charges extrêmes peuvent être envisagées (problématique de tenue statique de l'assemblage), des chocs encaissés, des chargements cycliques à supporter (problématique de fatigue). Au final, les difficultés expérimentales et numériques, ajoutées à toutes ces différentes contraintes expliquent l'utilisation de coefficients de sécurité qui obligent à un sur-dimensionnement des structures d'assemblage afin d'assurer la durée de vie souhaitée. Ce sur-dimensionnement engendre un surcoût inhérent à la production mais également à l'exploitation du produit final.

**Une meilleure prédiction et maîtrise** du comportement numérique de l'assemblage, ainsi que l'usage de **méthodes d'optimisation adaptées**, pourraient fournir une augmentation des différentes performances tout en réduisant de manière non négligeable les masses et les coûts mis en jeu.

En vue de répondre à ce besoin industriel, le LMT-Cachan a développé au cours des dernières années et continue (par l'intermédiaire de projets industriels) de développer des méthodes de calculs performantes (méthode de calcul LaTin, développement de modèles réduits par PGD, etc.). Le laboratoire oriente ses différentes méthodes afin de fournir aux industriels des outils de conception fiable et robuste (méthodes non intrusives de calcul, méthode de calcul multiparamétrique, etc.). C'est d'ailleurs au sein de l'Unité Thématique de Recherche « Ingénierie et Conception Robuste » qu'a été effectuée cette thèse. **Ce travail s'appuie sur ces développements numériques afin de proposer une méthode d'optimisation globale efficace d'assemblages mécaniques.**

Dans le cadre de la mécanique des assemblages, l'optimisation globale de structures industrielles est encore un point difficile. L'optimisation réalisée se fait principalement sur des pièces isolées où les conditions aux limites sont mal maîtrisées. Cette vision de l'optimisation est due aux différents verrous présentés précédemment. Une optimisation par des méthodes classiques, comme par exemple les algorithmes évolutionnaires (type génétiques), sont rendues inenvisageables en raison du coût de calcul associé à un jeu de paramètres et le nombre important d'évaluations à effectuer. Il faut également tenir compte de la difficulté à maîtriser des méthodes de calculs non-linéaires pour traiter le contact et le frottement qui sont souvent mal connues.

Pleinement conscient de ces différents verrous, le LMT-Cachan a participé à différents projets tel que OMD et OMD2 (Optimisation MultiDisciplinaire et Optimisation MultiDisciplinaire Distribuée). Le but de ces projets étaient de proposer des méthodologies et démonstrateurs afin de résoudre des problèmes multidisciplinaires de grande dimension. Le LMT-Cachan a proposé l'emploi de stratégies adaptées à la résolution numérique d'assemblages afin de construire des approximations du modèle mécanique de manière efficace. Une stratégie d'optimisation multi-niveaux a d'ailleurs pu être mise en place :

- Le premier niveau de modélisation a pour but d'effectuer une optimisation globale sur un modèle de substitution au modèle mécanique réel. Ce modèle de substitution est aussi appelé « métamodèle ». Un métamodèle présente l'avantage d'être peu coûteux pour évaluer la fonction objectif que l'on souhaite optimiser. Celui-ci est construit à l'aide d'évaluations d'un échantillonnage de données de l'espace de conception. Une optimisation globale sur

ce modèle est donc parfaitement envisageable, à l'aide d'un algorithme d'optimisation classique comme les algorithmes génétiques ou à essaim de particules. Il est cependant nécessaire que sa précision soit suffisante pour au moins localiser la zone où se situe le minimum global associé au modèle mécanique réel. Afin de s'en assurer, des stratégies d'enrichissement intelligentes du métamodèle existent. De ce fait, la phase de calcul coûteuse est alors la phase d'évaluations des données de l'espace de conception échantillonné, en vue de construire ce métamodèle. Dans notre cas d'étude un point de l'espace de conception est évalué à l'aide de **la méthode de calcul LaTIn multiparamétrique**.

- Dans un second temps, une optimisation locale est menée sur le modèle mécanique réel. Le point de départ de cette optimisation est l'optimum approché trouvé lors de la phase précédente.

Dans la lignée de ces différents travaux, Luc Laurent a développé au cours de sa thèse [Laurent, 2013], la construction de métamodèles robustes dans le cadre de la mécanique des assemblages. Les métamodèles de type krigeage ou RBF avec ou sans prise en compte de gradients ont notamment été développés.

Comme dit précédemment, les phases de calcul les plus coûteuses sont celles exigeant l'évaluation d'un jeu de paramètres par le solveur mécanique (on verra au chapitre 5 qu'un cas test traité exige 12 heures de calcul pour un jeu de paramètres).

Ce travail se concentre essentiellement sur la réduction du coût de calcul lors de la première étape de la démarche d'optimisation multi-niveaux. Un des principaux apports de cette thèse est l'utilisation de données auxiliaires dans la construction du métamodèle, le but étant d'améliorer sa qualité en un temps et un nombre d'appels au solveur réduits. Dans le cadre des travaux de [Laurent, 2013] les données auxiliaires étaient les gradients en un point évalué. On se propose dans ce travail d'utiliser des approches **multi-fidélité** en vue de construire ces métamodèles. Ceux-ci sont construits à partir d'au moins deux sources d'informations : les données dites « basse fidélité » (ce sont les données auxiliaires) et les données dites « haute fidélité » (ce sont les données principales). La hiérarchisation de ces données peut provenir de différentes approches :

- Modèles hiérarchisés selon leur contenu physique. Le modèle basse fidélité étant celui qui a le moins de contenu physique
  - Modèles calculés sur différentes tailles de maillage. Le modèle basse fidélité étant celui calculé sur le maillage le plus grossier
  - Même modèle mais les données sont calculées pour différents niveaux de convergence de calcul. Les données basse fidélité sont celles ayant le plus faible niveau de convergence.
- On parle alors de données partiellement convergées.**

Les données issues du modèle haute fidélité sont les données calculées via le modèle mécanique le plus riche ou le plus précis.

Dans ces travaux, l'ensemble des théories de construction de métamodèle se base sur le krigeage. Ce choix se justifie par la qualité des résultats obtenus dans la littérature ainsi que les travaux de [Laurent, 2013] qui ont montré que le krigeage était bien adapté aux problématiques de la mécanique des assemblages.

Trois grandes stratégies dites **multi-fidélité** pour la prise en compte de données auxiliaires ont été étudiées et développées :

- « Méthode de correction de surface » : on s'intéresse plus particulièrement aux méthodes de correction par fonctions additives. L'Evofusion développée par [Forrester, 2004] est la méthode qui sera privilégiée pour ce type d'approche.
- « Cokrigeage multi-fidélité » : 3 approches différentes permettant de relier les données auxiliaires et principales (intercorrélation ou intercovariance des données) ont été développées.
- « Krigeage hiérarchique » : méthode s'appuyant principalement sur le krigeage mais dont la particularité est d'utiliser un métamodèle auxiliaire comme fonction de régression.

L'intérêt de ces méthodes provient du fait qu'en un temps réduit l'utilisation des données basse fidélité (moins coûteuses à calculer) fournit une première tendance de la fonction objectif que l'on cherche à modéliser. Cette première tendance est corrigée à l'aide des données haute fidélité (plus coûteuses en temps, mais dont le nombre est réduit), fournissant alors un métamodèle pouvant être exploité lors de la première étape de la stratégie d'optimisation décrite précédemment. Afin de qualifier quelles méthodes sont les plus adaptées dans le cadre de la mécanique des assemblages, deux aspects sont principalement étudiés :

- **construire un métamodèle précis de la fonction objectif (abaque virtuel),**
- **rechercher le plus efficacement possible l'optimum de la fonction objectif (avec prise en compte d'éventuelles contraintes).**

On s'intéresse également à la mise en place de critères d'enrichissement de l'échantillonnage de l'espace de conception utilisé pour construire le métamodèle. Des critères d'arrêt de ces enrichissements sont également proposés et se basent sur des considérations statistiques. Ils ont été développés sur les cas tests présentés dans ce travail. Des critères d'arrêt basés sur un budget de calcul ont également été utilisés pour mettre en avant l'apport des stratégies développées au cours de ces travaux.

Cette thèse s'appuie sur des acquis du laboratoire et de ses nombreux développements. Dans cette optique de limitation des coûts de calcul et suite aux travaux réalisés au cours des 15 dernières années, il semblait naturel de continuer à utiliser un solveur LaTIn (Large Time Increment method) multiparamétrique. Le solveur retenu a été COFAST (COntract et Frottement dans les Assemblages et STructures). Ce solveur est basé sur la méthode LaTIn mono-échelle. Celui-ci a montré sa capacité à résoudre des problèmes d'assemblages, ce qui explique son choix dans cette thèse. La méthode LaTIn se base sur une décomposition de domaine mixte. On définit assez naturellement la décomposition de domaine. Les **sous-domaines** ou **sous-structures** sont les pièces de l'assemblage, celles-ci sont liées par les **interfaces** de la décomposition. Chaque sous-structure et interface peut avoir son comportement propre, ceci permet une grande flexibilité de modélisation. L'interface entre chaque pièce peut alors tenir compte du contact avec ou sans frottement et avec ou sans jeu. L'**aspect multiparamétrique** de la méthode consiste à pouvoir utiliser une solution précédemment calculée comme point de départ de l'algorithme LaTIn pour le calcul d'un nouveau jeu de paramètres. Cet aspect est intéressant car la simulation d'assemblages consiste à résoudre des problèmes de même type. Seuls quelques paramètres comme

des coefficients de frottement, de jeux ou de précharges sur des boulons sont modifiés. Les paramètres principalement étudiés se situent donc au niveau des interfaces et sont locaux. Cette approche a montré une bonne efficacité quant à réduire les coûts de calcul.

**Le point clé de ces travaux** est que la méthode LaTIn traite itérativement les équations de la mécanique sur tout l'intervalle de chargement contrairement aux algorithmes classiques de type Newton. En effet, ces derniers sont basés sur un découpage de l'intervalle de chargement en incréments : les itérations étant réalisées à chaque incrément. Le fait que le solveur LaTIn itère globalement permet d'obtenir une approximation de la solution mécanique complète (en tout point de l'espace et en tout instant de l'intervalle de chargement) à chaque itération. Il est alors possible de définir des niveaux de convergence en fonction du nombre d'itérations ou d'un indicateur de convergence de la simulation du calcul d'assemblage. On définira alors des données « partiellement convergées » qui seront les données basse fidélité des différentes stratégies de construction de métamodèle multi-fidélité utilisées. Ces données correspondront à un faible niveau de convergence. Les données haute fidélité seront des données dites « totalement convergées » qui correspondront à un fort niveau de convergence de l'algorithme de calcul LaTIn.

**Coupler la stratégie de calcul LaTIn multiparamétrique avec les stratégies de modélisations multi-fidélité peut s'avérer être extrêmement bénéfique en terme de temps de calcul. L'étude de ce couplage est un des points prépondérant de cette thèse.**

L'utilisation de la méthode de calcul multiparamétrique permet de réduire aussi les coûts de calcul lors de la second phase d'optimisation de la méthode multi-niveaux d'optimisation proposée. Afin de les réduire encore un peu plus le développement d'une méthode d'optimisation locale efficace sera mise en œuvre. Conscient également que les progrès de développement de simulations numériques ne se font pas que sur la puissance brute des processeurs disponibles, mais également sur des outils de parallélisation ; le développement d'une parallélisation du solveur numérique a été mis en place afin de traiter des problèmes de taille industrielle.

D'un point de vue pérennisation des développements, ceux-ci ont été intégrés à la toolbox GRENAT (GRAdient ENhanced Approximation Toolbox) initiée par [Laurent, 2013]. Les développements apportés à cette toolbox seront présentés ainsi que les éléments mathématiques de construction de métamodèles.

Le présent manuscrit est structuré en cinq chapitres afin de présenter les différents développements apportés au cours de cette thèse ainsi que les différents résultats.

- Ces travaux s'appuient sur trois grandes thématiques que sont **le calcul d'assemblage, la construction de modèles de substitution et l'optimisation**. Le **premier chapitre** présente ainsi le contexte dans lequel se situe cette thèse. Une présentation globale des problèmes d'optimisation y sera rappelée ainsi que certaines notions et méthodes de résolutions. Un rappel sur la construction de métamodèles, notamment par krigeage (base sur laquelle s'appuient ces travaux) sera effectué. La méthode de calcul LaTIn dans le cadre des assemblages sera également rappelée. Finalement, différentes approches permettant de prendre en compte des données basse fidélité seront introduites.
- Le **deuxième chapitre** vise à mener une étude de faisabilité de l'utilisation de données partiellement convergées dans le cadre du calcul de structures. Pour cela, on s'intéresse notamment aux nombres de données partiellement convergées à calculer ainsi qu'à leur

niveau de précision en vue d'obtenir un métamodèle pouvant servir d'abaque virtuel ou efficace afin de mener une stratégie d'optimisation globale.

- Le **troisième chapitre** a pour but d'introduire différentes méthodes de modélisations multi-fidélité. Ces méthodes sont notamment comparées sur des cas test en vue d'en déterminer les plus efficaces dans le cadre des assemblages mécaniques. L'efficacité de ces méthodes est comparée à l'aide d'une stratégie dite « de recalage » qui à la suite d'enrichissement de l'échantillonnage de l'espace de conception permet la construction d'un métamodèle de bonne qualité utilisable comme abaque virtuel.
- Le **quatrième chapitre** présente la comparaison, dans le cadre d'optimisation globale de cas test mécaniques, des différentes méthodes multi-fidélité jugées performantes à la suite des résultats obtenus au chapitre précédent. Afin d'améliorer l'efficacité des différentes approches multi-fidélité présentées, différentes méthodes d'échantillonnage de l'espace de conception ainsi que différents critères d'arrêt statistiques d'enrichissement pour la stratégie d'optimisation sont testées.
- Le **cinquième chapitre** a pour but de montrer l'efficacité de la méthode multi-fidélité jugée la meilleure suite aux résultats obtenus au cours des différents chapitres sur des cas test industriels. Une optimisation sous contrainte est réalisée sur un premier cas test et la création d'un abaque virtuel sur un second. Finalement, une perspective à ces travaux est présentée. Il s'agit d'introduire l'information des gradients aux différentes méthodes multi-fidélité présentées.

# Chapitre 1

## Contexte et outils numériques

*Ce premier chapitre vise à présenter le contexte de l'optimisation d'assemblage mécanique. On rappelle le cadre de l'optimisation dans lequel ce travail est effectué. Par la suite, les différents outils utilisés au cours de cette thèse sont présentés. Certains algorithmes d'optimisation sont alors rappelés de même pour les métamodèles et plus spécifiquement ceux basés sur le krigeage. Les grands principes de la méthode LaTIn sur laquelle s'appuie l'ensemble des calculs du solveur mécanique sont également rappelés. Finalement une présentation d'approches multi-fidélité est faite.*

### Sommaire

---

<b>1</b>	<b>Introduction aux problèmes d'optimisation et contexte scientifique de l'étude .</b>	<b>9</b>
1.1	Introduction . . . . .	9
1.2	Définition du problème d'optimisation . . . . .	9
1.3	Différentes optimisations . . . . .	11
1.4	Contexte général de l'étude . . . . .	13
<b>2</b>	<b>Algorithmes d'optimisation . . . . .</b>	<b>22</b>
2.1	Algorithmes d'optimisation locale sans contrainte . . . . .	22
2.2	Algorithmes d'optimisation locale sous contraintes . . . . .	27
2.3	Algorithmes d'optimisation globale . . . . .	30
<b>3</b>	<b>Utilisation de métamodèle . . . . .</b>	<b>35</b>
3.1	Construction de métamodèle par krigeage . . . . .	36
3.2	Qualité du métamodèle . . . . .	43
3.3	Optimisation et métamodèle : Stratégie EGO . . . . .	45
3.4	Bilan . . . . .	47
<b>4</b>	<b>La méthode LaTIn . . . . .</b>	<b>48</b>
4.1	Présentation du problème mécanique . . . . .	50
4.2	Résolution par la méthode LaTIn . . . . .	52
4.3	Approche multiparamétrique . . . . .	59



<b>5</b>	<b>Approches multi-fidélité . . . . .</b>	<b>60</b>
5.1	Introduction . . . . .	60
5.2	Précisions sur le vocabulaire utilisé . . . . .	61
5.3	Optimisation par approches multi-niveaux . . . . .	63
5.4	Modélisations multi-fidélité . . . . .	65
5.5	Assimilation aux données bruitées . . . . .	67
<b>6</b>	<b>Bilan du chapitre . . . . .</b>	<b>69</b>

---

# 1 Introduction aux problèmes d'optimisation et contexte scientifique de l'étude

## 1.1 Introduction

Actuellement, avec l'aide des moyens de calcul de plus en plus conséquents les problèmes d'optimisation sont davantage pris en considération lors de la phase de conception. Cette optimisation d'un produit (soit de ses caractéristiques techniques, de son rendement ou de son prix...) est de plus en plus nécessaire dans des politiques d'économie, d'éco-conception, de valorisation ou d'investissement de différentes entreprises. Par exemple, le groupe Renault Trucks a réduit de manière drastique ses émissions de polluants. Entre 1990 et 2014 (Norme Euro 0 et Norme Euro 6) les rejets d'oxydes d'azote ont été réduits de 98%, les rejets d'hydrocarbures imbrûlés de 95%, le monoxyde de carbone de 89% et les particules de 97%. De plus, la consommation d'un camion longue distance, a été réduite de 30% en 30 ans<sup>1</sup>. Dans le cadre de l'aéronautique, un moteur d'avion comporte 3500 assemblages. En réduisant de 20 grammes chacun des assemblages, une économie globale de 70 kg est réalisée. Cette économie correspond à 95000 litres de kérosène, c'est-à-dire 244 tonnes de CO<sub>2</sub>, ce qui correspond à l'exploitation annuelle du vol Paris-Toulouse (20 allers-retours par jour)<sup>2</sup>.

Cependant, il semble important de noter que ce que beaucoup de professionnels appellent « optimisation », est en réalité une amélioration d'un produit par rapport à une gamme précédente. Les techniques présentées dans la suite de ce document concernent l'optimisation en son sens mathématique (qu'elle soit locale ou globale). Celles qui seront plus largement développées s'appliquent aussi à respecter une certaine robustesse afin de pouvoir fonctionner en boîte noire, et d'être un outil à part entière pour l'ingénieur.

## 1.2 Définition du problème d'optimisation

Comme dit précédemment, les stratégies présentées et/ou développées dans ce document s'appuient sur la définition mathématique de l'optimisation. Ainsi, Allaire [Allaire, 2007] rappelle qu'un problème d'optimisation en mécanique est défini par trois données :

- **un modèle** permettant d'évaluer la ou les structures à optimiser. Ce modèle prend en compte les différentes équations de la mécanique à traiter pour évaluer la réponse de la structure à une sollicitation donnée. Il amène à résoudre différents types d'équations, notamment des équations aux dérivées partielles (EDP), impossibles à résoudre de manière exacte. On adjoint alors au modèle mécanique un modèle éléments finis afin d'estimer la solution des différentes équations de la mécanique. On confondra dans la suite modèle mécanique et modèle éléments finis.

---

1. <http://corporate.renault-trucks.com/fr/actualites/camion-et-environnement.html>

2. eco-calculateur de la DGAC

- **un critère** qui est, soit à maximiser, soit à minimiser. Ce critère est souvent une fonctionnelle appelée « fonction coût », « fonction d'intérêt » ou encore « fonction objectif ». Cette fonctionnelle est notée  $y$  et correspond à la réponse de la structure évaluée. On peut chercher à optimiser plusieurs critères pour une même optimisation (optimisation multiobjectif). Les définitions mathématiques données dans la suite de ce travail sont présentées dans le but de minimiser un (ou des) critère(s). Si on souhaite le (ou les) maximiser, il suffit d'adapter la définition.
- **un ensemble admissible**  $\Omega$  correspondant à l'espace des variables d'optimisation (espace de conception  $\mathcal{D}$ ) et à l'espace d'admissibilité de la fonction objectif  $C_{\mathcal{T}}$ . Ce dernier espace est défini par l'expression de fonctions dites contraintes auxquelles le problème d'optimisation doit répondre. Ces contraintes peuvent être d'inégalités et/ou d'égalités. On appellera également  $\Omega$  ensemble de conception admissible.

Un problème d'optimisation mono-objectif consiste donc à trouver  $\mathbf{x}^*$  sous la forme :

$$\mathbf{x}^* = \arg \min_{\mathbf{x} \in \Omega} y(\mathbf{x}) \quad (1.1)$$

avec  $\mathbf{x}$  : vecteur des variables d'optimisation,  $\Omega$  : espace de conception admissible,  $y$  : fonction objectif à minimiser et évaluée via le modèle mécanique mis en place.

Cependant, cette formulation est peu usitée car peu exploitable. Les algorithmes d'optimisation classiquement employés (dont quelques-uns seront décrits dans la suite) utilisent une formulation différente mais équivalente à l'équation 1.1.

L'espace de conception  $\mathcal{D}$  correspond à un ensemble ou un sous-ensemble mathématique dans lequel le problème est défini. Dans nos cas d'étude, cet ensemble est l'ensemble des réels  $\mathbb{R}^{n_{dim}}$ , avec  $n_{dim}$  dimension de l'espace de conception. On définit alors un sous-ensemble en bornant les valeurs des variables de conception. Ainsi :

$$\mathcal{D} = \left\{ \mathbf{x} = (x_1, \dots, x_{n_{dim}}) \in \mathbb{R}^{n_{dim}} \text{ tel que } \forall i \in \llbracket 1, n_{dim} \rrbracket, x_i^{\min} \leq x_i \leq x_i^{\max} \right. \\ \left. \text{avec } x_i^{\min} \text{ et } x_i^{\max} \text{ bornes mini. et maxi.} \right\} \quad (1.2)$$

On appelle  $\mathbf{G}$  et  $\mathbf{H}$  l'ensemble des fonctions contraintes, respectivement d'inégalité au nombre de  $n_g$  et d'égalité au nombre de  $n_h$  ( $\mathbf{G} = \bigcup_j g_j$  et  $\mathbf{H} = \bigcup_k h_k$ ).

L'espace définissant  $C_{\mathcal{T}}$  peut s'écrire :

$$C_{\mathcal{T}} = \left\{ \mathbf{x} \in \mathcal{D} \text{ tel que } \left| \begin{array}{l} g_j(\mathbf{x}) \geq 0, \quad \forall j \in \llbracket 1, n_g \rrbracket \\ h_k(\mathbf{x}) = 0, \quad \forall k \in \llbracket 1, n_h \rrbracket \end{array} \right. \right\} \quad (1.3)$$

**Remarque :** Bien entendu, les bornes de conception peuvent être définies à l'aide de fonctions contraintes d'inégalité. Si tel est le cas, il suffit de remplacer dans l'équation 1.3,  $\mathcal{D}$  par  $\mathbb{R}^{n_{dim}}$ .

Au final,  $\Omega = \mathcal{D} \cap C_{\mathcal{T}}$ . Le problème d'optimisation peut alors être écrit sous la forme :

$$\mathbf{x}^* = \begin{cases} \arg \min_{\mathbf{x} \in \mathcal{D}} y(\mathbf{x}) \\ \text{tel que :} \\ g_j(\mathbf{x}) \geq 0, \quad \forall j \in \llbracket 1, n_g \rrbracket \\ h_k(\mathbf{x}) = 0, \quad \forall k \in \llbracket 1, n_h \rrbracket \end{cases} \quad (1.4)$$

Les problèmes décrits par les équations 1.1 et 1.4 sont donc équivalents. Les différentes fonctions contraintes peuvent être formulées de manière intrinsèque ou extrinsèque au problème mécanique traité. Par exemple, on peut avoir une relation explicite à vérifier liant différents paramètres à optimiser, ou au contraire avoir une contrainte à évaluer via le modèle mécanique mis en place.

La formulation d'un problème d'optimisation mono-objectif est donc très simple et générale. Cependant, bien que la formulation du problème puisse se faire de manière générale, la méthode de résolution de celui-ci diffère en fonction de la classe de problèmes traités.

Classiquement en mécanique, deux grandes familles de type d'optimisation existent : l'optimisation de forme et l'optimisation paramétrique. Des stratégies spécifiques sont alors développées en fonction du problème étudié.

## 1.3 Différentes optimisations

### 1.3.1 Optimisation de forme

L'optimisation de forme consiste à trouver la meilleure forme géométrique ou topologique d'une structure sous sollicitations mécaniques. L'objectif est de minimiser ou maximiser un (des) critère(s). Classiquement il s'agit d'une minimisation d'un critère de masse. Cette optimisation doit se réaliser dans l'espace des solutions inhérent au problème mécanique posé comme la prise en compte de conditions aux limites mécaniques et des liaisons associées (il s'agit ici en fait du respect du modèle utilisé). Il faut également tenir compte des contraintes fixées par le contexte du problème : respect des critères de tenue mécanique, respect des coûts, respect des contraintes d'ergonomie, etc. Il existe trois grandes classes d'optimisation de forme :

- l'optimisation de forme paramétrique qui consiste à paramétrer dimensionnellement la géométrie des pièces considérées. Cette approche est également appelée dimensionnement automatique des structures [Duisinx, 1996] ou de transformation homéomorphe. Ce type d'optimisation rend nécessaire la bonne définition du problème afin d'éviter des incohérences géométriques. Définir mathématiquement ces contraintes est donc une partie essentielle du travail du concepteur.
- l'optimisation de forme géométrique qui consiste à faire varier une forme initiale en modifiant la position des frontières de la forme géométrique sans en faire varier la topologie. Dans ce contexte d'optimisation, on peut admettre un changement de dimensions ainsi que la configuration de l'objet. Il faut cependant veiller à respecter sa connectivité ou sa nature (ne pas modifier le nombre de ces composants par exemple).

- L'optimisation topologique qui consiste à rechercher la meilleure forme de la structure au sens des critères retenus tout en permettant des modifications de la topologie (apparition de trous par exemple). La topologie désigne le nombre et la position des composantes des domaines. Cette optimisation modifie fondamentalement la nature de l'objet. La géométrie de l'objet est envisagée sans aucun *a priori* sur la connectivité des domaines ou des composants présents dans la solution. [Duysinx, 1996, Bendsøe, 1995, Duysinx et Bendsøe, 1998].

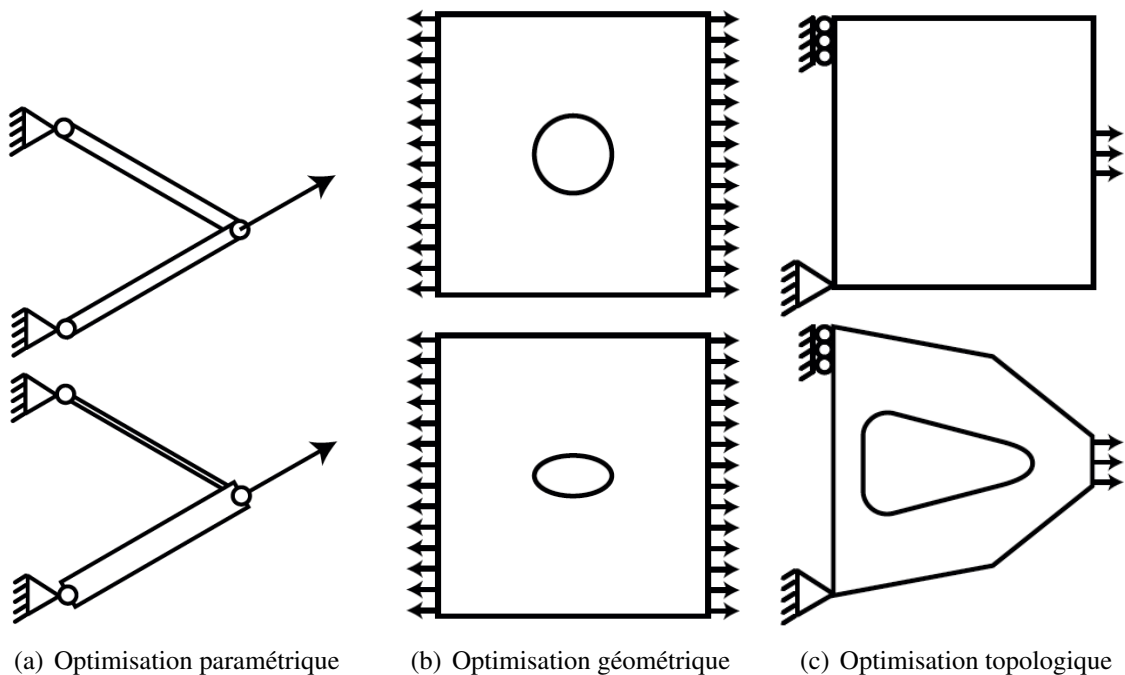


FIGURE 1.1: Différents types d'optimisation de forme [Duysinx, 1996]

**Remarque :** D'autres exemples autres que purement mécaniques peuvent être trouvés dans des champs disciplinaires assez divers, comme notamment l'électronique ou le piézoélectrique ([Khalil, 2009, Sigmund, 2001, Silva et Kikuchi, 1999]).

### 1.3.2 Optimisation paramétrique

L'optimisation paramétrique se place dans un cadre plus large que l'optimisation de forme et peut s'appliquer à plusieurs disciplines. De plus, les variables d'optimisation et les critères à optimiser peuvent être divers. On peut tout autant chercher à optimiser la forme d'une pièce en fonction de paramètres géométriques, tout comme on peut chercher à minimiser sa masse ou son prix en fonction de paramètres matériaux. Les travaux présentés dans cette thèse visent à optimiser des critères dépendant de paramètres inhérents aux problèmes d'assemblages. On s'intéressera notamment aux jeux entre les pièces, à leur coefficient de frottement ou encore aux

précharges de boulons. Quelques paramètres liés à la géométrie locale des pièces seront pris en compte mais ceux-ci seront relativement limités.

## 1.4 Contexte général de l'étude

### 1.4.1 Contexte industriel

La problématique d'optimisation d'assemblage reste peu traitée car le modèle mécanique associé est complexe. En effet, les méthodologies de résolution de ces problèmes sont souvent coûteuses et difficiles à mettre en œuvre. Bien que de nombreux progrès ont été faits dans ce domaine, les temps de calcul nécessaires à la résolution des problèmes d'assemblages sont encore très importants. Le terme assemblage fait référence à différentes catégories de techniques et méthodes visant à réunir différentes pièces afin de créer un produit ou sous-produit fonctionnel. Différents phénomènes physiques (voir même chimiques) sont à prendre en compte lors de l'opération d'assemblage. Ces phénomènes peuvent être des phénomènes de frottement, de contact, de plastification des matériaux en certaines zones, d'endommagement, etc. Des phénomènes liés au procédé d'assemblage peuvent également être à prendre en compte, par exemple, pour la soudure où des phénomènes thermiques locaux engendrent des contraintes locales. Les méthodes d'assemblage les plus utilisées sont celles utilisant le frottement, assemblage par vis ou par frettage.

Dans un souci d'économie, des secteurs tels que l'aéronautique cherchent à limiter les essais expérimentaux souvent coûteux financièrement mais également en temps et en ressources (personnel, matériels, lieux d'expérimentation etc.). Ces essais ont pour but de valider le dimensionnement des structures. Au vu des développements informatiques de ces dernières années, la simulation numérique est actuellement de plus en plus usitée et remplace même parfois l'essai expérimental (on parle alors de « virtual testing »). Comme dit précédemment, ces simulations numériques peuvent s'avérer extrêmement complexes à mettre en œuvre. Des conditions d'adhérence/glisement, de contact avec/sans jeu sont à prendre en compte lors de la phase d'assemblage et d'utilisation. Ces phénomènes sont fortement non-linéaires et nécessitent donc des méthodes adaptées et souvent coûteuses. Ceux-ci ne sont pas négligeables parce qu'ils rendent compte du comportement de la structure ou d'une sous-structure. Par exemple, ces différents aspects sont prépondérants lorsque l'on s'intéresse à la tenue d'un boulon.

On s'intéressera plus spécifiquement dans ce travail aux assemblages utilisant le frottement (vis et boulons). De plus, les pièces à assembler seront considérées comme prêtes à l'assemblage. On ne considérera donc pas d'éventuels défauts qu'ils soient géométriques ou dimensionnels. Si on souhaite prendre en compte ces défauts dans notre approche, ils doivent faire partie de l'espace des paramètres de conception  $\mathcal{D}$ . Dans ce contexte les travaux effectués au laboratoire ([Champaney, 1996, Boucard et Champaney, 2003, Soulier et Boucard, 2013, Soulier et Boucard, 2009, Laurent *et al.*, 2013a]) ont permis de mettre en place des stratégies dédiées à :

- résoudre des problèmes d'assemblages en prenant en compte les non-linéarités dues aux contacts frottants avec ou sans jeu entre les pièces.
- mener une optimisation paramétrique globale sur des problèmes mécaniques.

Une présentation plus détaillée de la méthode utilisée au sein du laboratoire est donnée au paragraphe 1.4.6. Pour résoudre les équations liées au modèle non-linéaire, un solveur a été développé au LMT et a montré son efficacité pour la résolution de ce genre de problèmes. Il s'agit de CO-FAST développé par L.Champaney [Champaney, 1996].

### 1.4.2 Difficulté de l'optimisation

Il est souvent difficile de résoudre analytiquement les problèmes décrits par l'ensemble d'équations 1.1 ou 1.4. Cette approche est envisageable dans les cas où l'on peut connaître de manière assez simple la relation de dépendance entre la fonction objectif, les fonctions contraintes et les paramètres à optimiser. Bien entendu cette approche est impossible dans notre cadre d'étude et le recours à l'optimisation numérique est donc nécessaire. Les problèmes étudiés font partie des problèmes d'optimisation dit « difficiles » [Dréo, 2004, Dréo *et al.*, 2006]. En effet, certaines caractéristiques peuvent être problématiques comme l'absence de convexité stricte (multimodalité de la fonction objectif), la présence de bruit, les discontinuités, etc.

### 1.4.3 Optima locaux et globaux

Un problème d'optimisation peut admettre différents types de solutions. La ou les solutions du problème d'optimisation défini à l'équation 1.4 correspondent à ce qu'on appelle des minima globaux. Il arrive que la solution trouvée corresponde à un minimum local c'est-à-dire que la solution trouvée à l'équation 1.4 n'est valable que dans un sous-domaine de  $\Omega$ .

#### Définition : Minimum global

Soit  $y : \Omega \rightarrow \mathbb{R}$  une fonctionnelle.  $\mathbf{x}^* \in \Omega$  est un point de minimum global si et seulement si,

$$\forall \mathbf{x}^+ \in \Omega, y(\mathbf{x}^*) \leq y(\mathbf{x}^+)$$

On dit alors que  $y(\mathbf{x}^*)$  est un **minimum global** de  $y$  sur  $\Omega$ .

#### Définition : Minimum local

Soit  $y : \Omega \rightarrow \mathbb{R}$  une fonctionnelle.  $\mathbf{x}^* \in \Omega$  est un point de minimum local si et seulement si,

$$\exists \varepsilon \in \mathbb{R}^+ \text{ tels que } \forall \mathbf{x}^+ \in [\mathbf{x}^* - \varepsilon, \mathbf{x}^* + \varepsilon] \subset \Omega, y(\mathbf{x}^*) \leq y(\mathbf{x}^+)$$

On dit alors que  $y(\mathbf{x}^*)$  est un **minimum local** de  $y$  sur  $\Omega$ .

#### Définition : Minimum local strict

Soit  $y : \Omega \rightarrow \mathbb{R}$  une fonctionnelle.  $\mathbf{x}^* \in \Omega$  est un point de minimum local strict si et seulement si,

$$\exists \varepsilon \in \mathbb{R}^+ \text{ tels que } \forall \mathbf{x}^+ \in [\mathbf{x}^* - \varepsilon, \mathbf{x}^* + \varepsilon] \subset \Omega, y(\mathbf{x}^*) < y(\mathbf{x}^+)$$

On dit alors que  $y(\mathbf{x}^*)$  est un **minimum local strict** de  $y$  sur  $\Omega$ .

Ces définitions sont illustrées sur la figure 1.2.

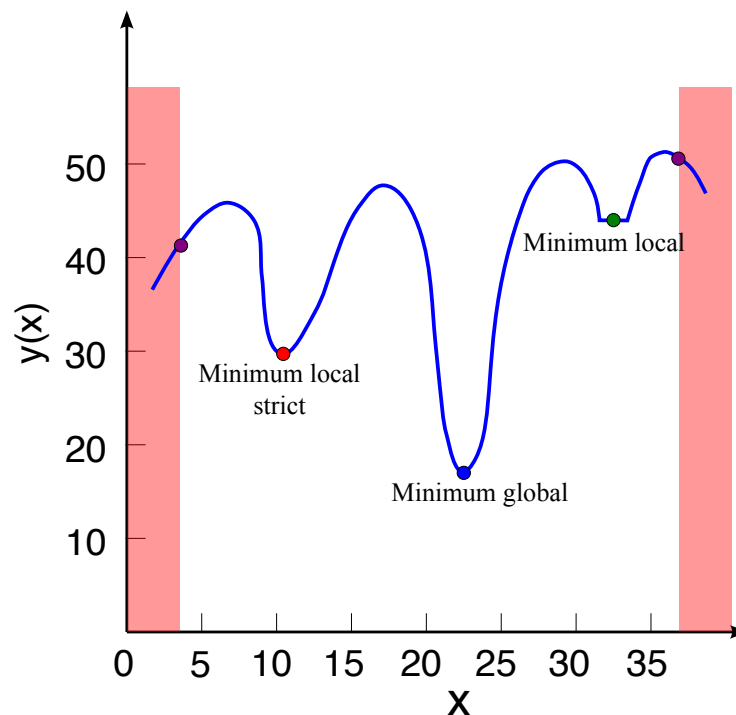


FIGURE 1.2: Illustration minima globaux et locaux

Sur cet exemple, on peut constater qu'un large éventail de possibilités de minimum est représenté. ● est le minimum global de la fonction représentée. ● est un minimum local strict, en effet au voisinage (restreint) de ce point, il n'y en a pas ayant une valeur inférieure ou égale. ● est un minimum local non strict car dans son voisinage des points ont la même évaluation que lui. On constate également que la présence de contraintes (ici de bornes) peut amener d'autres minima locaux ●. La présence de contraintes bien que pouvant sembler complexifier le problème, peut permettre en réalité de le simplifier. Par exemple, la fonction

$$y : \mathbb{R} \longrightarrow \mathbb{R}$$

$$x \longrightarrow \exp(-x/10) \cos(x) + 0.1x$$

n'admet aucun minimum global sur  $\mathbb{R}$  (voir figure 1.3), tandis que des contraintes limitant le domaine d'étude permettent la résolution du problème d'optimisation.

Dans le cadre de l'optimisation numérique, il existe deux grandes familles de stratégies d'optimisation : optimisation locale ou optimisation globale. On rappelle dans la prochaine section, de manière non exhaustive, certaines de ces méthodes qui seront utilisées dans cette thèse. [Nocedal et Wright, 2006] fait une présentation complète et détaillée de beaucoup de méthodes d'optimisation, ainsi que leur contexte mathématique.



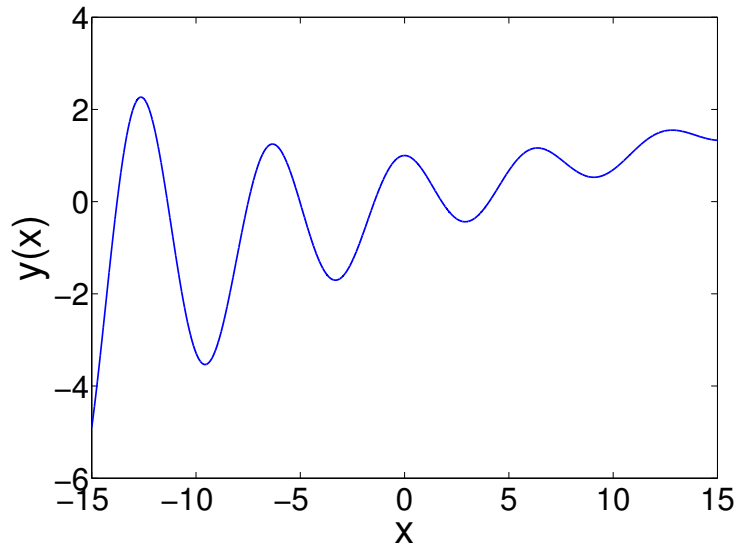


FIGURE 1.3: Fonction  $y(x) = \exp(-x/10) \cos(x) + 0.1x$  sur  $[-15, 15]$

#### 1.4.4 Quelle stratégie ?

Comme rappelé par [Barton et Meckesheimer, 2006], le choix d'une stratégie d'optimisation dépend des caractéristiques de la fonction objectif et de l'espace de conception  $\mathcal{D}$ . On peut distinguer des stratégies adaptées en fonction du fait que l'espace  $\mathcal{D}$  soit discret ou non (voir figure 1.4). Il est ensuite important de regarder le temps de simulation associé à un jeu de paramètres. En effet, le nombre de simulations peut être non négligeable en fonction du coût de calcul et du temps imparti pour résoudre le problème.

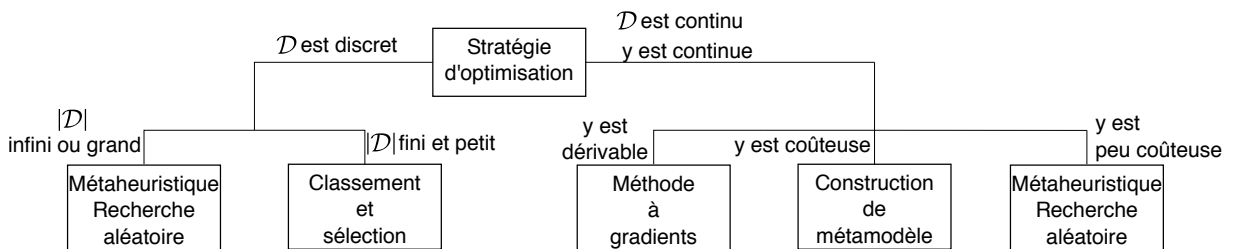


FIGURE 1.4: Exemple de choix d'algorithmes en fonction des caractéristiques du problème [Barton et Meckesheimer, 2006]

Bien entendu, cette figure 1.4 n'est seulement qu'une indication de choix possibles. On peut, par exemple, très bien utiliser un algorithme métaheuristique pour optimiser une fonction objectif  $y$  continue et dérivable.

Notre étude porte sur des assemblages mécaniques, pour lesquels il existe deux grandes difficultés. Celles-ci sont liées à la non-linéarité du problème et à sa taille (grand nombre de degrés de libertés), ce qui peut donc être coûteux en temps de calcul. L'utilisation d'algorithme de type métaheuristique est donc inenvisageable du point de vue du temps de calcul. Les algorithmes

basés sur le gradient ne sont pas non plus envisageables car on cherche à mener une optimisation globale dans cette étude (de plus les fonctions étudiées ne sont pas forcément  $C^1$ ). Le choix d'une optimisation par métamodèle est donc le choix qui semble le plus adapté.

### 1.4.5 L'utilisation d'un métamodèle

Dans le cadre de ces travaux on cherche à développer une stratégie d'optimisation globale permettant de limiter le temps de calcul. Résoudre le problème d'optimisation décrit à l'équation 1.4, via un algorithme d'optimisation classique tel qu'un algorithme génétique, demande un nombre important de simulations du modèle mécanique. Cependant comme déjà dit, une évaluation via le modèle mécanique est souvent coûteuse. L'optimisation serait alors trop chronophage. La stratégie consiste alors à remplacer le modèle mécanique par un modèle de substitution : métamodèle. Le modèle de substitution ainsi créé permet de lier explicitement la fonction objectif à ses paramètres. Une optimisation peut alors être menée sur le métamodèle. Celle-ci est rendue possible par le fait qu'une évaluation de ce modèle est peu coûteuse.

Notre étude ne s'intéressera qu'à la résolution de **problèmes déterministes**, ce qui signifie qu'en un point  $\mathbf{x}$  la réponse mécanique  $y(\mathbf{x})$  est unique. Des travaux sur des problèmes stochastiques peuvent être également abordés, dans ce cas on s'intéresse principalement à l'optimisation de la réponse moyenne [Janusevskis et Le Riche, 2013].

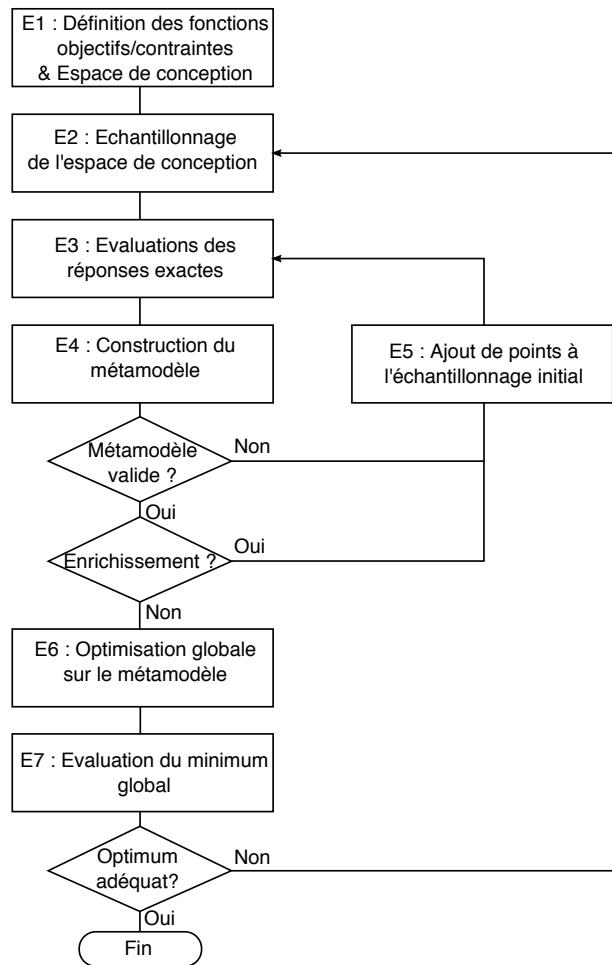
En s'appuyant sur [Barton et Meckesheimer, 2006], il existe quatre points fondamentaux de la métamodélisation :

- 1 Le choix de la forme mathématique du métamodèle  $\hat{y}$
- 2 Le plan d'expérience, le tirage ou l'échantillonnage de points  $\mathbf{x}_i$  à évaluer, choisis dans l'espace de conception  $\mathcal{D}$ .
- 3 La création même du métamodèle, à l'aide des évaluations  $\hat{y}_i$  des points  $\mathbf{x}_i$ .
- 4 L'évaluation de la qualité de la réponse (intervalle de confiance, éventuelle qualité de l'interpolation des données, qualité de réponse).

[Jones, 2001, Barton, 1998] fournissent des exemples de différents métamodèles. En s'appuyant sur les travaux de [Barton et Meckesheimer, 2006, Forrester *et al.*, 2008, Laurent, 2013], on peut présenter un algorithme assez généraliste décrivant le moyen de mener à bien une optimisation sur le modèle de substitution (voir figure 1.5).

Plusieurs étapes constituent cet algorithme

- La première étape E1 consiste à définir la ou les fonctions objectif (ainsi que les fonctions contraintes). Ces fonctions sont dictées par l'analyse fonctionnelle et par l'expertise des ingénieurs de conception travaillant sur le projet. Un autre rôle des ingénieurs est de définir l'espace de conception : choix des paramètres à optimiser (étude de sensibilité des paramètres) et taille de l'espace de conception (valeurs minimales et maximales des paramètres).
- La seconde étape E2 est le choix d'un échantillonnage de l'espace de conception. Différentes techniques sont possibles et notamment le tirage LHS [McKay *et al.*, 1979]



**FIGURE 1.5:** Description optimisation par métamodèle

qui permet d'échantillonner l'espace de manière représentative (ce tirage sera largement utilisé au cours de cette thèse).

- L'étape E3 est l'évaluation des points issus de l'échantillonnage. Cette étape est *a priori* la plus longue de l'algorithme. Le temps de calcul peut être important et dépendant du solveur utilisé et de la complexité du problème.
- L'étape E4 consiste à construire le métamodèle à l'aide des évaluations des points.
- L'étape E5 est **facultative** et correspond à un enrichissement de l'échantillonnage. Cet enrichissement peut se faire sur le plan d'expérience existant (manque de site dans une zone de l'espace de conception, validité du métamodèle) ou en utilisant des informations d'ores-et-déjà fournies par le métamodèle (enrichissement dans les zones où se trouve potentiellement le minimum global par exemple).
- A l'étape E6 un algorithme d'optimisation globale est utilisé sur le métamodèle. Cette optimisation est bien moins coûteuse qu'une optimisation directe sur le modèle mécanique.

- L'étape E7 consiste à évaluer le point d'optimum trouvé sur le métamodèle afin de vérifier qu'il respecte bien les contraintes ou qu'il soit bien le meilleur optimum connu sur le problème traité par les ingénieurs.

#### 1.4.6 Présentation de la méthode utilisée au LMT

La stratégie utilisée au laboratoire et dans les travaux précédents [Laurent *et al.*, 2013a, Laurent *et al.*, 2013b, Soulier et Boucard, 2013], est légèrement différente. Elle se base sur une approche multi-niveaux [Robinson et Keane, 1999, Keane et Petruzzeli, 2000] à deux niveaux de modélisation.

- Le premier niveau consiste à modéliser la réponse mécanique en fonction des différents paramètres qui la composent par un métamodèle. Ce métamodèle est obtenu suite aux évaluations effectuées par un solveur mécanique d'un échantillonnage de l'espace. On peut éventuellement envisager un enrichissement de celui-ci. Ce modèle de substitution ayant l'avantage d'être peu coûteux à évaluer, l'optimum de ce dernier est déterminé à l'aide d'un algorithme d'optimisation globale tel que les algorithmes évolutionnaires ou à essais particuliers. Le travail effectué sur ce niveau de modélisation reprend donc ce qui a été présenté au paragraphe précédent.
- Le second niveau de modélisation est celui du modèle mécanique utilisé. L'optimisation qui est faite alors sur ce modèle est une optimisation locale (pouvant être coûteuse en fonction du modèle mécanique utilisé). Le point de départ de cette optimisation locale est l'optimum trouvé au premier niveau. Cette étape permet d'affiner la recherche du minimum et remplace l'étape E7 de l'algorithme précédemment décrit (voir figure 1.5).

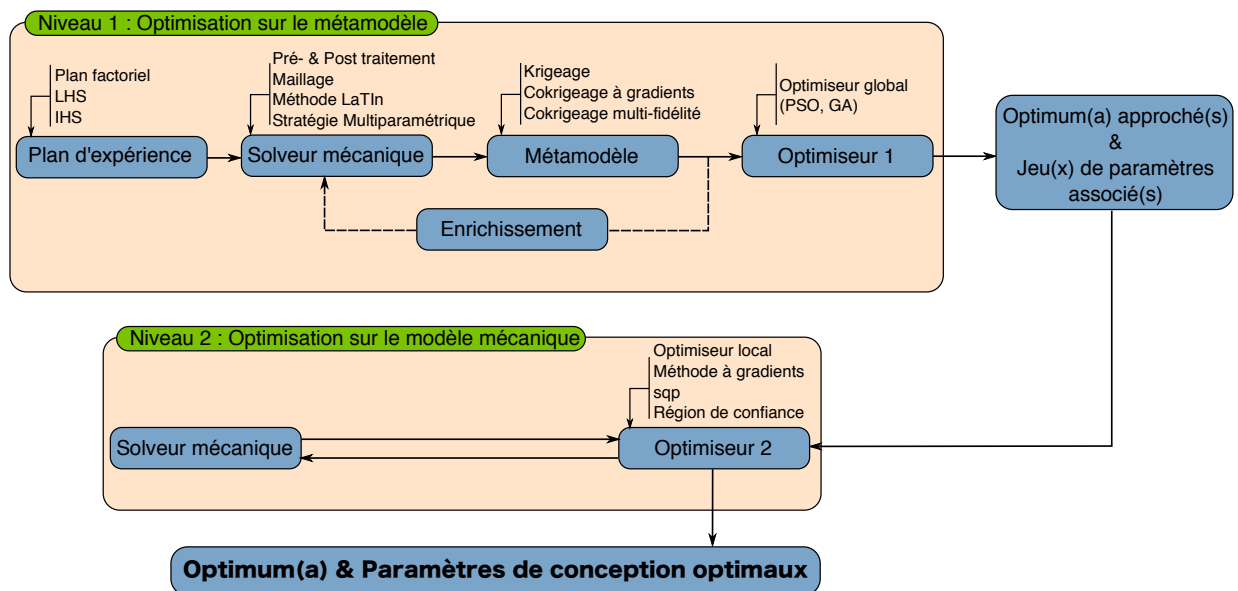


FIGURE 1.6: Présentation générale de la méthode d'optimisation

La figure 1.6 illustre cette approche. L'optimisation structurale est un thème d'étude qui a déjà été abordé au sein du LMT. On peut notamment citer les travaux de B.Soulier [Soulier, 2003] ou encore ceux effectués dans le cadre du projet OMD (Optimisation Multi-Disciplinaire) [Buytet *et al.*, 2008]. Les travaux plus récents ([Soulier et Boucard, 2013]) ont permis la mise en place de l'optimisation multi-niveaux précédemment décrite. Cette optimisation multi-niveaux a été effectuée à l'aide du solveur mécanique COFAST, permettant l'utilisation de la méthode de calcul LaTin multiparamétrique efficace sur les problèmes mécaniques d'assemblages. [Laurent, 2013] a poursuivi dans cette thématique de recherche en proposant notamment l'utilisation de métamodèle à gradients.

**Le but de ces travaux est d'intégrer des approches permettant de tenir compte d'évaluations secondaires provenant de données dites « partiellement convergées » issues du solveur mécanique. Ce travail se centre notamment sur la construction de métamodèles dit multi-fidélité tenant compte d'évaluations de données totalement convergées, calculées de manière précise et de données partiellement convergées, calculées de manière moins précise mais plus rapidement (voir paragraphe 4.2.6 et section 5.4).**

Le tableau 1.1 présente les différents développements apportés au cours de cette thèse pour mener à bien l'optimisation multi-niveaux.

On retrouve donc effectivement l'ensemble des constituants utilisé pour mener une optimisation avec une stratégie par métamodèle illustré en figure 1.5.

Ainsi, on décrira dans les prochaines sous-parties ces différents constituants. On s'attachera dans un premier temps à présenter quelques algorithmes d'optimisation qui seront utilisés lors de ces travaux de thèse. Ces présentations de différents algorithmes permettront de rappeler quelques propriétés mathématiques inhérentes aux problèmes d'optimisation. Le descriptif des constituants nécessaires à la création d'un métamodèle sera ensuite fait. On s'attachera à présenter le processus de construction par krigeage qui s'est avéré particulièrement efficace sur des cas test mécaniques [Laurent, 2013]. On rappellera par la suite la méthode LaTin, méthode sur laquelle s'appuie le solveur mécanique COFAST, utilisé pour évaluer un jeu de paramètres de l'assemblage. La présentation de la méthode LaTin permettra également d'introduire le concept de données partiellement convergées utilisé dans cette thèse. Finalement, une présentation de méthodes permettant l'utilisation de données partiellement convergées sera menée.

**Remarque :** L'utilisation des métamodèles n'est pas réservée au seul contexte de l'optimisation. Par exemple, ces modèles de substitutions sont également utilisés avec succès dans le cadre de la mécanique probabiliste de défaillance [Bichon *et al.*, 2008, Echard *et al.*, 2011, Echard *et al.*, 2013, Dubourg *et al.*, 2011].

Constituants de la méthode	Etats des développements au début de la thèse	Développements apportés
Echantillonnage	Tirage LHS, Plan factoriel complet	Tirage LHS adaptatif (voir paragraphe 2.1 du chapitre 4 )
Solveur mécanique	COFAST. Il s'agit d'un solveur spécifique aux assemblages et basé sur la méthode LaTIn ([Ladevèze, 1999, Champaney, 1996, Boucard et Champaney, 2003])	Parallélisation gros grain du solveur COFAST (voir chapitre 5)
Métamodèle	Des toolbox MATLAB sont disponibles au laboratoire pour effectuer la construction du métamodèle. Les toolbox notamment utilisées sont DACE 2.0 [Lophaven <i>et al.</i> , 2002a, Lophaven <i>et al.</i> , 2002b] et GRENAT [Laurent, 2013]. Des développements sur les métamodèles à gradients ont notamment été effectués.	Méthodes de construction de métamodèles par des approches multi-fidélité (voir chapitre 3) avec prise ou non du gradient (voir chapitre 5).
Optimiseur global	PSO ou algorithme génétique	Optimiseur multi-optima réalisé afin de trouver les minima locaux et globaux d'une fonction (voir paragraphe 2.3.2 du chapitre 5).
Optimiseur local	SQP	Optimiseur local s'appuyant sur une méthode à région de confiance (voir paragraphe 4.1.1 du chapitre 4 ).
Enrichissement	Celui-ci consiste à venir enrichir l'échantillonnage initial par un point répondant à critère spécifique (EI maximum MSE)	Introduction d'un critère d'enrichissement pour mener une optimisation sous contrainte : constrained expected improvement. (voir chapitre 5) et tests de critères d'arrêt (voir paragraphes 2.3.2 au chapitre 3 et 1.2 au chapitre 4).

TABLE 1.1: Développements apportés au cours de cette thèse

## 2 Algorithmes d'optimisation

Comme mentionné précédemment, il existe deux grandes classes d'optimisation : locale ou globale. Ces deux approches sont employées dans la stratégie multi-niveaux décrite au paragraphe 1.4.6. Une brève description des algorithmes utilisés au cours de cette thèse est alors faite.

### 2.1 Algorithmes d'optimisation locale sans contrainte

Les algorithmes d'optimisation sont initialisés par un jeu de paramètres et évoluent autour de cette position initiale. Ils tirent au maximum profit des informations sur l'évolution de la fonction objectif. Certains d'entre eux sont très populaires et permettent une optimisation locale robuste. Des résultats d'existence sont démontrables (à condition de connaître certaines propriétés de la fonction objectif et des fonctions contraintes). Ces résultats s'expriment sous forme de conditions d'optimalité. En fonction du problème traité, (optimisation non contrainte, optimisation avec contraintes d'égalité, optimisation avec contraintes d'inégalité, optimisation avec contraintes d'égalité et d'inégalité) ces conditions s'expriment de manières différentes. Bien évidemment, ces conditions sont valables dans des cas restreints de problèmes (continuité, dérivabilité, convexité, etc.)

#### 2.1.1 Conditions d'optimalité

La définition du minimum local peut s'écrire sous forme de conditions d'optimalité : conditions du premier ordre et du second ordre. Ces conditions dans le cas d'un problème non contraint s'écrivent sous la forme suivante :

##### **Théorème : Condition nécessaire au premier ordre**

Si  $\mathbf{x}^*$  est un minimum local et si  $y$  est continuellement dérivable dans un voisinage ouvert de  $\mathbf{x}^*$ , alors  $\nabla y(\mathbf{x}^*) = 0$

##### **Théorème : Condition nécessaire au second ordre**

Si  $\mathbf{x}^*$  est un minimum local et si  $y$  est continuellement dérivable dans un voisinage ouvert de  $\mathbf{x}^*$ , alors  $\nabla y(\mathbf{x}^*) = 0$  et  $\nabla^2 y(\mathbf{x}^*)$  est défini semi-positif.

##### **Théorème : Condition suffisante au second ordre**

Si  $\nabla^2 y(\mathbf{x}^*)$  est continu dans un voisinage ouvert de  $\mathbf{x}^*$ , que  $\nabla y(\mathbf{x}^*) = 0$  et que  $\nabla^2 y(\mathbf{x}^*)$  est définie positive, alors  $\mathbf{x}^*$  est un minimum local strict  $y$ .

#### 2.1.2 Quelques algorithmes

Ces conditions d'optimalité permettent de prouver la convergence des algorithmes d'optimisation. Parmi les différentes techniques d'optimisation non contrainte, on peut notamment citer :

*Couplage optimisation à convergence partielle et stratégie multiparamétrique en calcul de structures*

- Méthodes de recherche directe [Wright, 1996, Lewis *et al.*, 2000, Kolda *et al.*, 2003]. On peut les classer en trois catégories : méthode de recherche par motifs généralisés [Torczon, 1989], méthodes de directions conjuguées [Powell, 1998], simplexe [Nelder et Mead, 1965]
- Méthodes de descente, à pas constant, à pas optimal, ou par gradients conjugués. Pour utiliser ces algorithmes, il est nécessaire d'avoir accès à l'évaluation de la fonction objectif et à ses dérivées. Ces algorithmes sont itératifs et ont pour principe de trouver à l'itération  $i + 1$  un jeu de paramètres  $\mathbf{x}^{i+1}$  tel que  $y(\mathbf{x}^{i+1}) < y(\mathbf{x}^i)$ . Pour ce faire, on cherche  $\mathbf{x}^{i+1}$  sous la forme  $\mathbf{x}^{i+1} = \mathbf{x}^i + h_i \mathbf{u}_i$  avec  $\nabla y(\mathbf{x}^i)^T \mathbf{u}_i < 0$ .
- Méthodes de Newton, qui cherchent les points critiques de la fonction objectif c'est-à-dire ceux qui vérifient  $\nabla y(\mathbf{x}) = 0$ . On suppose que la fonction est  $\mathcal{C}^2$  et que le hessien de la fonction est inversible. Le principe de base de ces méthodes est d'approximer la fonction objectif autour du point  $\mathbf{x}^i$  par une approximation quadratique.  $y(\mathbf{x}^i + \mathbf{v}) \simeq y(\mathbf{x}^i) + \nabla y(\mathbf{x}^i) \mathbf{v} + \frac{1}{2} \mathbf{v}^T \nabla^2 y(\mathbf{x}^i) \mathbf{v}$ . Le gradient de cette expression s'annule pour  $\mathbf{v} = -[\nabla^2 y(\mathbf{x}^i)]^{-1} \nabla y(\mathbf{x}^i)$ , ainsi  $\mathbf{x}^{i+1} = \mathbf{x}^i - [\nabla^2 y(\mathbf{x}^i)]^{-1} \nabla y(\mathbf{x}^i)$ . Le point dur de cette méthode est bien entendu l'inversion de la matrice hessienne. On peut estimer le terme  $[\nabla^2 y(\mathbf{x}^i)]^{-1} \approx \mathbf{h}_i$  par une méthode DFP [Davidon, 1959, Fletcher et Powell, 1963] ou BFGS [Broyden, 1970, Fletcher, 1970, Goldfarb, 1970, Shanno, 1970].

Estimation de l'hessien par approche DFS :

$$\mathbf{h}_{i+1}^{DFS} = \mathbf{h}_i^{DFS} + \frac{\delta_i \delta_i^T}{\delta_i^T \gamma_i} - \frac{\mathbf{h}_i^{DFS} \gamma_i \gamma_i^T \mathbf{h}_i^{DFS}}{\gamma_i^T \mathbf{h}_i^{DFS} \gamma_i}$$

Estimation de l'hessien par approche BFGS :

$$\mathbf{h}_{i+1}^{BFGS} = \mathbf{h}_i^{BFGS} + \left(1 + \frac{\gamma_i^T \mathbf{h}_i^{BFGS} \gamma_i}{\delta_i^T \gamma_i}\right) \frac{\delta_i \delta_i^T}{\delta_i^T \gamma_i} - \frac{\delta_i \gamma_i^T \mathbf{h}_i^{BFGS} \mathbf{h}_i^{BFGS} \gamma_i \delta_i^T}{\delta_i^T \gamma_i}$$

avec  $\gamma_i = \nabla y(\mathbf{x}^{i+1}) - \nabla y(\mathbf{x}^i)$  et  $\delta_i = \mathbf{x}^{i+1} - \mathbf{x}^i$

### 2.1.3 Méthode à région de confiance

Le principe de ces méthodes à région de confiance est de remplacer le problème initial d'optimisation par une succession de sous-problèmes d'optimisation, ceux-ci étant plus simples à résoudre. A chaque itération de l'algorithme associé à un sous-problème à résoudre, la fonction objectif  $y$  est remplacée par une fonction modèle  $y_m$ . La région de confiance définit la zone dans laquelle il est estimé que la fonction  $y_m$  approxime convenablement la fonction  $y$ . Bien entendu, la taille de cette zone est un point clé de cette méthode. En effet, une zone trop grande peut amener à trouver un minimum sur la fonction modèle  $y_m$  éloigné du minimum de la fonction objectif  $y$ . Au contraire, une zone trop petite peut amener à faire une succession de calculs inutiles et à faire converger l'algorithme vers un point ne correspondant pas au minimum de la fonction



$y$  [Bergmann, 2004]. Cette méthode d'optimisation par région de confiance est donc en soi une optimisation multi-niveaux. En effet, le problème d'optimisation à mener est remplacé par une succession d'optimisations locales effectuées dans une zone plus ou moins réduite (la zone de confiance) sur un modèle se substituant au modèle à optimiser. Celui-ci permet l'adaptation de cette zone de confiance en fonction des résultats obtenus aux optimisations locales.

Une description complète de ces méthodes peut être trouvée dans [Conn *et al.*, 2000, Nocedal et Wright, 2006]. On s'attachera ici à présenter les points importants de la méthode. Classiquement les méthodes à région de confiance consistent à résoudre une suite de sous-problèmes d'optimisation à l'aide d'une fonction modèle  $y_m$  quadratique. Soient  $y$  une fonction dont on cherche le minimum (problème non contraint),  $\mathbf{x}_k$  un point dont on connaît l'évaluation  $y(\mathbf{x}_k)$  et  $y_{m_k}^{quad}$  la fonction modèle approchant  $y$  autour du point  $\mathbf{x}_k$ . On définit  $y_{m_k}^{quad}$  la fonction modèle quadratique comme :

$$y(\mathbf{x}_k + \mathbf{s}) \approx y_{m_k}^{quad}(\mathbf{s}) = y(\mathbf{x}_k) + \nabla y(\mathbf{x}_k)^T \mathbf{s} + \frac{1}{2} \mathbf{s}^T \nabla^2 y(\mathbf{x}_k) \mathbf{s} \quad (1.5)$$

Une formulation similaire de l'algorithme est possible en approximant le gradient et l'hessien de la fonction  $y$ . On notera respectivement dans la suite,  $\mathbf{g}_k = \mathbf{g}(\mathbf{x}_k)$  et  $\mathbf{H}\mathbf{e}_k = \mathbf{H}\mathbf{e}(\mathbf{x}_k)$  les évaluations au point  $\mathbf{x}_k$  du gradient et du hessien de  $y$ . Si les formulations du gradient et de l'hessien sont exactes, on parle alors de région de confiance de Newton ([Bergmann, 2004]).

La philosophie des méthodes à région de confiance est de trouver  $\mathbf{s}$  tel que que l'on vienne minimiser l'expression (1.5). Il existe cependant une restriction :  $\mathbf{s}$  doit se trouver dans une zone définie qui est la région de confiance. Le problème s'écrit alors :

$$\min_{\mathbf{s} \in \mathcal{D}} y_{m_k}^{quad}(\mathbf{x}_k + \mathbf{s}) = y(\mathbf{x}_k) + \mathbf{g}_k^T \cdot \mathbf{s} + \frac{1}{2} \mathbf{s}^T \cdot \mathbf{H}\mathbf{e}_k \cdot \mathbf{s} \quad (1.6)$$

sous la contrainte :

$$\|\mathbf{s}\| \leq \Delta_k$$

$\Delta_k > 0$  est le rayon de la région de confiance. La norme euclidienne est souvent la norme choisie bien que d'autres choix de normes soient possibles.

On note  $\mathbf{s}_k$  la solution du problème de minimisation. Il n'y a évidemment pas *a priori* de preuve comme quoi  $y(\mathbf{x}_k + \mathbf{s}_k) < y(\mathbf{x}_k)$ . Il faut donc vérifier que cette solution  $\mathbf{s}_k$  est acceptable. On définit alors :

— la réduction réelle  $ared_k$  telle que :

$$ared_k(\mathbf{s}_k) = y(\mathbf{x}_k) - y(\mathbf{x}_k + \mathbf{s}_k) \quad (1.7)$$

— la réduction prédite par la fonction modèle  $pred_k$  telle que :

$$pred_k(\mathbf{s}_k) = y_{m_k}^{quad}(\mathbf{x}_k) - y_{m_k}^{quad}(\mathbf{x}_k + \mathbf{s}_k) \quad (1.8)$$

L'efficacité de la fonction modèle peut être déterminée par le rapport suivant :

$$\rho_k = \frac{ared_k}{pred_k} \quad (1.9)$$

Ce rapport sert de paramètre d'actualisation de l'algorithme. En effet, si  $\rho_k$  est négatif alors  $y(\mathbf{x}_k + s_k) > y(\mathbf{x}_k)$  et donc le point  $\mathbf{x}_{k+1} = \mathbf{x}_k + s_k$  et le pas d'itération sont rejetés. Le rayon de la région de confiance est alors diminué et une autre minimisation de la fonction modèle est menée à partir du même point  $\mathbf{x}_k$  que précédemment. Si  $\rho_k$  est positif alors dans ce cas  $y(\mathbf{x}_k + s_k) < y(\mathbf{x}_k)$  est donc le point  $\mathbf{x}_{k+1} = \mathbf{x}_k + s_k$  et l'itération sont acceptés. Le rayon de confiance peut selon les résultats être modifié. Si le critère de convergence n'est pas atteint une nouvelle minimisation est faite sur la fonction modèle mais avec pour point de départ le point  $\mathbf{x}_{k+1}$ .

La méthode globale est décrite à l'algorithme 1 [Bergmann, 2004].

---

**Algorithm 1** Algorithme à région de confiance pour des fonctions modèles quadratiques

---

- 1: Choix de constantes  $\mu_1, \mu_2, \gamma_1, \gamma_2, \gamma_3$  avec :  $0 < \mu_1 < \mu_2 < 1$  et  $0 < \gamma_1 < \gamma_2 < 1 < \gamma_3$   
Choix d'un rayon de confiance  $\Delta_0$  et d'un itéré  $\mathbf{x}_0$ .  
Calcul de  $y(\mathbf{x}_0)$ , initialisation  $k = 0$ .
- 2: Construction de la fonction modèle quadratique  $y_{m_k}^{quad}(\mathbf{x}_k + \mathbf{s})$ , conformément à l'équation 1.5
- 3: Résolution du problème :

$$\min_{\|\mathbf{s}\| \leq \Delta_k} y_{m_k}^{quad}(\mathbf{x}_k + \mathbf{s}) = y(\mathbf{x}_k) + \mathbf{g}_k^T \cdot \mathbf{s} + \frac{1}{2} \mathbf{s}^T \cdot \mathbf{H}_k \cdot \mathbf{s}$$

- 4: Calcul de  $y(\mathbf{x}_k + \mathbf{s}_k)$
  - 5: Evaluation du critère  $\rho_k = \frac{ared_k(\mathbf{s}_k)}{pred_k(\mathbf{s}_k)}$
  - 6: Mise à jour de l'itéré  $\mathbf{x}_k$ 
    - (a) Si  $\rho_k < 0$   $\mathbf{x}_{k+1} = \mathbf{x}_k$
    - (b) Si  $\rho_k \geq 0$   $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
  - 7: Actualisation du rayon de confiance :
    - (a) Si  $\rho_k \geq \mu_2$  :  $\Delta_k \in [\Delta_k, \gamma_3 \Delta_k]$
    - (b) Si  $\mu_1 \leq \rho_k < \mu_2$  :  $\Delta_k \in [\gamma_2 \Delta_k, \Delta_k]$
    - (c) Si  $\rho_k < \mu_1$  :  $\Delta_k \in [\gamma_1 \Delta_k, \gamma_2 \Delta_k]$
  - 8: Si un critère de convergence est atteint, l'algorithme est stoppé. Sinon on passe à l'itération suivante et retour à l'étape 2
- 

Habituellement on choisit  $\mu_1 = 0.25$  et  $\mu_2 = 0.75$ . Ces deux constantes sont appelées critères de performances,  $\gamma_1 = 0.25$ ,  $\gamma_2 = 0.75$  et  $\gamma_3 = 2$  sont appelés critères d'actualisation. Il peut cependant exister quelques petites variantes sur le choix de ces paramètres [Gano et Renaud, 2004, Alexandrov *et al.*, 1998] ou alors sur la mise en place d'une taille maximale pour la région de confiance [Keane et Nair, 2005, Giunta et Eldred, 2000].

La résolution du problème de minimisation de l'étape 3 de l'algorithme peut s'avérer être délicat en elle-même. Différents algorithmes peuvent alors être mis en place selon les caractéristiques de l'hessien (ou de son évaluation)  $\nabla^2 y$  [Nocedal et Wright, 2006, Bergmann, 2004]. On peut notamment citer les algorithmes à pas de Cauchy.

La figure 1.7 illustre la construction du modèle quadratique pour quatre itérations. A la première itération, on choisit  $\mathbf{x}_0 = 11$ . En imposant le rayon de région de confiance à 1, l'optimisation sur le modèle quadratique amène à ce que  $\mathbf{x}_1$  soit égal à 10, le rayon de région de confiance est doublé et ainsi de suite.

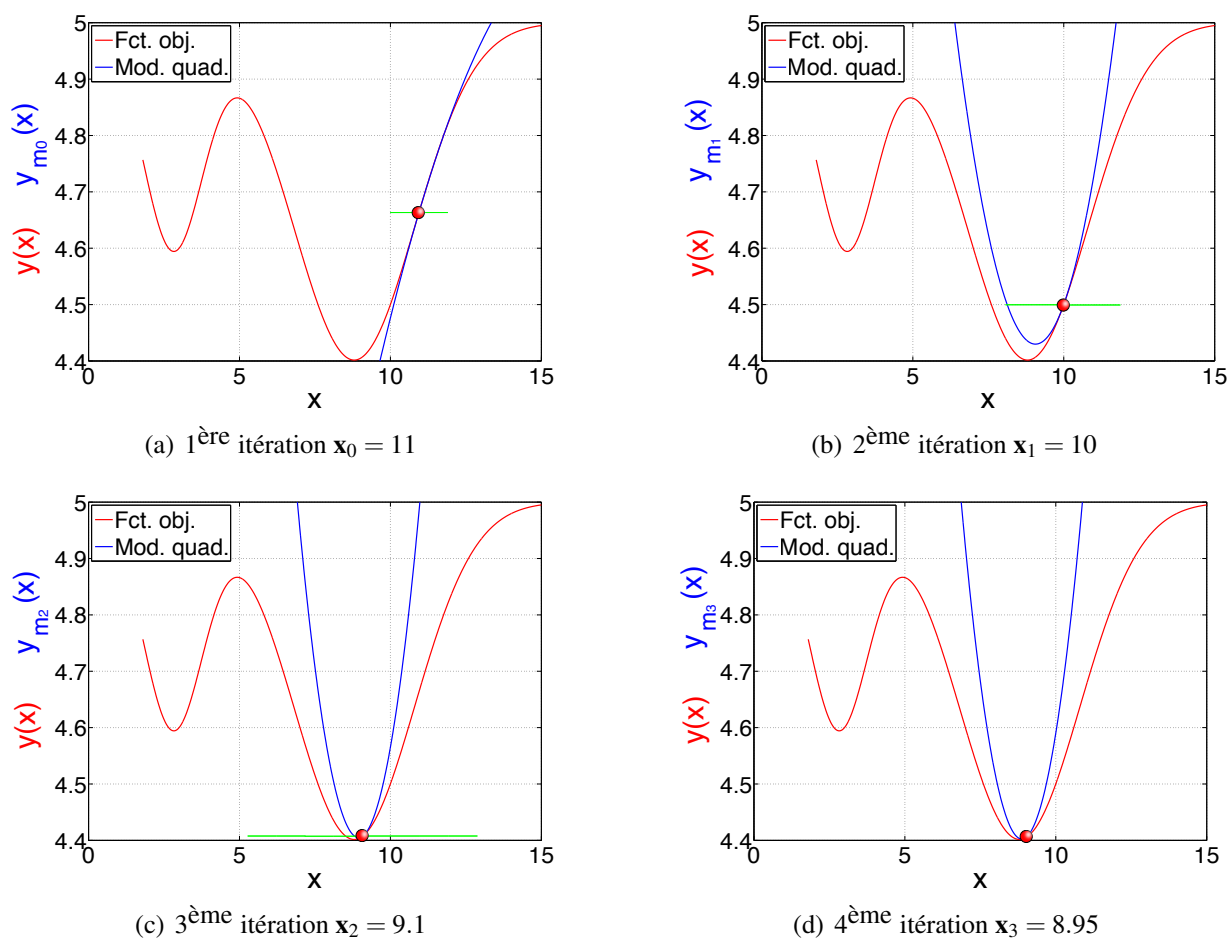


FIGURE 1.7: Illustration de l'algorithme à région de confiance

### Convergence de l'algorithme

Une condition nécessaire pour assurer la convergence des algorithmes à région de confiance est de s'assurer une approximation au premier ordre de la fonction objectif. Ceci se traduit par le fait que :  $y_{m_k} = y(\mathbf{x}_k)$  et  $\nabla y_{m_k} = \mathbf{g}_k \approx \nabla y(\mathbf{x}_k)$ . Plus de précisions sont disponibles dans [Powell, 1975, Carter, 1991, Alexandrov *et al.*, 1998, Bergmann, 2004].

## 2.2 Algorithmes d'optimisation locale sous contraintes

La plupart des problèmes d'optimisation sont contraints (ne serait-ce que par la définition de bornes à l'espace de conception  $\mathcal{D}$ ). Les contraintes sont décrites en deux types : les contraintes d'égalité (notées  $\mathbf{H}(\mathbf{x})$ ) et les contraintes d'inégalités (notées  $\mathbf{G}(\mathbf{x})$ ). Un point important est qu'à chaque itération de l'algorithme d'optimisation, ces contraintes doivent être vérifiées. On définit également les contraintes actives et inactives. Les contraintes actives sont les contraintes qui, à l'itération considérée, sont égales à 0. Par exemple, au point  $\mathbf{x}^i$ , si  $g_1(\mathbf{x}^i) \geq 0$  et  $g_2(\mathbf{x}^i) = 0$  alors  $g_1$  est considérée inactive et  $g_2$  active. Ceci signifie que lors de cette itération de l'algorithme d'optimisation la fonction  $g_1$  n'a pas d'incidence puisqu'elle est par définition respectée.

### 2.2.1 Méthodes de pénalisation

Ces approches ont pour effet de convertir un problème contraint en une séquence de sous-problèmes d'optimisation non contraints. On définit une fonction de mérite  $y_{merit}(\mathbf{x}) = y(\mathbf{x}) + \frac{1}{\mu}P(\mathbf{x})$ .  $P(\mathbf{x})$  est une fonction de pénalité pouvant prendre une valeur importante si les contraintes ne sont pas vérifiées. Le choix du paramètre  $\mu$  est prépondérant dans cette méthode afin d'éviter de trouver un minimum à la fonction  $y_{merit}$  qui ne soit pas admissible. L'avantage de ces approches réside dans le fait que le problème d'optimisation contraint est remplacé par un problème non contraint. Une fonction de pénalité simple est :

$$P(\mathbf{x}) = \begin{cases} 0 & \text{si contrainte vérifiée} \\ \mu \times v & \text{si contrainte non vérifiée avec } v \text{ très grand} \end{cases} \quad (1.10)$$

Une formulation plus usitée se base sur une expression quadratique telle que :

$$y_{merit}(\mathbf{x}) = y(\mathbf{x}) + \frac{1}{\mu} \sum_{i=1}^{n_h} h_i^2(\mathbf{x}) + \frac{1}{\mu} \sum_{i=1}^{n_g} ([g_i(\mathbf{x})]^-)^2 \quad (1.11)$$

avec  $[g_i(\mathbf{x})]^- = \max(-g_i(\mathbf{x}), 0)$ . On peut également citer les méthodes barrières et notamment les log-barrières [Nocedal et Wright, 2006].

### 2.2.2 Méthodes Lagrangiennes

Ces méthodes consistent à former une fonctionnelle nommée lagrangien associée au problème d'optimisation. Ce lagrangien s'écrit  $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}_h, \boldsymbol{\lambda}_g) = y(\mathbf{x}) - \boldsymbol{\lambda}_h \mathbf{H}(\mathbf{x}) - \boldsymbol{\lambda}_g \mathbf{G}(\mathbf{x})$ . La recherche de la solution peut se faire par la recherche d'un point-selle [Arrow *et al.*, 1958].

### 2.2.3 Conditions d'optimalité

De la même manière que pour les problèmes non contraints, on peut définir des conditions d'optimalité.

**Théorème : Condition nécessaire au premier ordre**

Si  $\mathbf{x}^*$  est un minimum local du problème sous contrainte et que les contraintes actives sont linéairement indépendantes au point  $\mathbf{x}^*$ , alors il existe un vecteur multiplicateur de Lagrange  $\boldsymbol{\lambda}^*$  de composantes  $\lambda_i$  tels que  $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ .

$$\begin{aligned}
\nabla_x \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) &= 0 \\
h_i(\mathbf{x}^*) &= 0 \quad \forall i \in \llbracket 1, n_h \rrbracket \\
g_i(\mathbf{x}^*) &\geq 0 \quad \forall i \in \llbracket 1, n_g \rrbracket \\
\lambda_i^* &\geq 0 \quad \forall i \in \llbracket 1, n_g \rrbracket \\
\lambda_i^* g_i(\mathbf{x}^*) &= 0 \quad \forall i \in \llbracket 1, n_g \rrbracket \\
\lambda_i^* h_i(\mathbf{x}^*) &= 0 \quad \forall i \in \llbracket 1, n_h \rrbracket
\end{aligned} \tag{1.12}$$

Ces conditions nécessaires sont appelées communément les conditions de Karush-Kuhn-Tucker (KKT) [Karush, 1939, Kuhn et Tucker, 1951]. Beaucoup d'algorithmes s'appuient sur l'utilisation de ces conditions nécessaires d'optimalité. Celles-ci peuvent être suffisantes sous certaines conditions mathématiques (convexité des fonctions etc.). On peut également ajouter qu'un point-selle vérifie ses conditions.

**Théorème : Conditions nécessaires et suffisantes au second ordre**

Des conditions au second ordre s'appliquent également aux problèmes sous contraintes. La principale différence se situe au niveau de l'hessien considéré. En effet, on ne s'intéresse pas à l'hessien de la fonction objectif  $\nabla^2 y$ , mais à l'hessien du lagrangien  $\nabla^2 \mathcal{L}$ . Pour de plus amples détails, notamment sur le contexte mathématique, on peut se référer à [Nocedal et Wright, 2006].

**2.2.4 Méthode sequential quadratic programming (SQP)**

Cette approche consiste à remplacer le problème initial pouvant être fortement non-linéaire par une suite de problèmes quadratiques sous contraintes linéaires plus simples à résoudre [Garcia Palomares et Mangasarian, 1976, Nocedal et Wright, 2006]. Une façon d'introduire cette méthode est de poser le problème suivant.

$$\begin{cases} \min y(\mathbf{x}) \\ \mathbf{H}(\mathbf{x}) = 0 \end{cases} \tag{1.13}$$

On peut résoudre ce problème sous une forme non contrainte en introduisant un lagrangien.

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = y(\mathbf{x}) - \boldsymbol{\lambda}^T \mathbf{H}(\mathbf{x}) \tag{1.14}$$

La méthode classiquement utilisée pour résoudre ce problème est de chercher les conditions d'optimalité KKT à l'aide de la méthode de Newton (voir paragraphe 2.1.2).

On cherche alors à résoudre le système :

$$S(\mathbf{x}, \boldsymbol{\lambda}) = \begin{bmatrix} \nabla y(\mathbf{x}) - \nabla \mathbf{H}(\mathbf{x})^T \boldsymbol{\lambda} \\ \mathbf{H}(\mathbf{x}) \end{bmatrix} = \mathbf{0} \quad (1.15)$$

avec  $\nabla \mathbf{H}(\mathbf{x})^T = [\nabla h_1(\mathbf{x}) \dots \nabla h_{n_h}(\mathbf{x})]$ . La méthode de Newton comme précédemment décrite cherche à trouver un pas d'itération fonction du jacobien du précédent système. L'itéré de Newton  $i+1$  s'écrit alors :

$$\begin{bmatrix} \mathbf{x}^{i+1} \\ \boldsymbol{\lambda}^{i+1} \end{bmatrix} = \begin{bmatrix} \mathbf{x}^i \\ \boldsymbol{\lambda}^i \end{bmatrix} + \begin{bmatrix} \mathbf{v}_x^i \\ \mathbf{v}_\lambda^i \end{bmatrix} \quad (1.16)$$

On trouve alors les incréments  $\mathbf{v}_x^i$  et  $\mathbf{v}_\lambda^i$  en résolvant le système :

$$\begin{bmatrix} \nabla^2 \mathcal{L}(\mathbf{x}^i, \boldsymbol{\lambda}^i) & -\nabla \mathbf{H}(\mathbf{x}^i)^T \\ \nabla \mathbf{H}(\mathbf{x}^i) & 0 \end{bmatrix} \begin{bmatrix} \mathbf{v}_x^i \\ \mathbf{v}_\lambda^i \end{bmatrix} = - \begin{bmatrix} \nabla \mathcal{L}(\mathbf{x}^i, \boldsymbol{\lambda}^i) \\ \mathbf{H}(\mathbf{x}^i) \end{bmatrix} \quad (1.17)$$

En soustrayant  $\nabla \mathbf{H}(\mathbf{x}^i)^T \boldsymbol{\lambda}$  on obtient un nouveau système équivalent à 1.17 qui est :

$$\begin{bmatrix} \nabla^2 \mathcal{L}(\mathbf{x}^i, \boldsymbol{\lambda}^i) & -\nabla \mathbf{H}(\mathbf{x}^i)^T \\ \nabla \mathbf{H}(\mathbf{x}^i) & 0 \end{bmatrix} \begin{bmatrix} \mathbf{v}_x^i \\ \boldsymbol{\lambda}^{i+1} \end{bmatrix} = - \begin{bmatrix} \nabla y(\mathbf{x}^i) \\ \mathbf{H}(\mathbf{x}^i) \end{bmatrix} \quad (1.18)$$

Ce système d'équation est également obtenu en essayant de résoudre à l'aide d'une approche par lagrangien le problème d'optimisation sous contraintes écrit au point  $\mathbf{x}^i$  :

$$\min_{\mathbf{v}} \nabla y(\mathbf{x}^i)^T \mathbf{v} + \frac{1}{2} \mathbf{v}^T \nabla^2 y(\mathbf{x}^i) \mathbf{v} \quad (1.19)$$

$$\text{et tel que } \mathbf{H}(\mathbf{x}^i) + \nabla \mathbf{H}(\mathbf{x}^i) \mathbf{v} = 0$$

Là encore le point difficile est le calcul de l'hessien, une méthode telle que la méthode BFGS peut être utilisée. L'algorithme général de la méthode SQP en cas de contraintes d'égalité est donc :

---

#### Algorithm 2 Algorithme SQP

---

- 1: Initialisation d'une paire  $(\mathbf{x}^0, \boldsymbol{\lambda}_0)$ ,  $k = 0$
  - 2: Evaluation de  $y(\mathbf{x}^k)$ ,  $\nabla y(\mathbf{x}^k)$ ,  $\nabla^2 y(\mathbf{x}^k)$ ,  $\mathbf{H}(\mathbf{x}^k)$  et  $\nabla \mathbf{H}(\mathbf{x}^k)$
  - 3: Résolution du problème quadratique pour obtenir  $\mathbf{x}^{k+1} \leftarrow \mathbf{x}^k + \mathbf{v}$  et  $\boldsymbol{\lambda}_{k+1}$
  - 4: Si convergence : ARRÊT avec solution  $(\mathbf{x}^{k+1}, \boldsymbol{\lambda}_{k+1})$ , sinon retour à l'étape 2.
-

Dans le cas de contraintes d'inégalité, le problème s'écrit alors :

$$\left\{ \begin{array}{l} \min_{\mathbf{v}} \nabla y(\mathbf{x}^i)^T \mathbf{v} + \frac{1}{2} \mathbf{v}^T \nabla^2 y(\mathbf{x}^i) \mathbf{v} \\ \text{et tel que } \mathbf{H}(\mathbf{x}^i) + \nabla \mathbf{H}(\mathbf{x}^i)^T \mathbf{v} = 0 \\ \text{et tel que } \mathbf{G}(\mathbf{x}^i) + \nabla \mathbf{G}(\mathbf{x}^i)^T \mathbf{v} \geq 0 \end{array} \right. \quad (1.20)$$

L'algorithme utilisé est du même type que précédemment. Cependant, une partie de l'étape 3 consiste alors à déterminer quelles sont les contraintes actives et inactives. Lorsque ces contraintes sont déterminées on peut alors se ramener à un problème ne faisant intervenir que des contraintes d'égalité [Conn *et al.*, 2000, Fletcher, 1987, Nocedal et Wright, 2006].

### 2.2.5 Méthode des points intérieurs

Cette méthode propose une démarche différente de la précédente. Elle s'appuie sur l'utilisation d'une fonction barrière logarithmique. Le but de cet algorithme est à partir d'une solution initiale  $\mathbf{x}_0$  de construire une suite d'itérés, se situant à l'intérieur de la zone d'admissibilité et convergeant vers la solution. Ce genre d'algorithme a été introduit dans les années 50 par [Frisch, 1955]. [Karmarkar, 1984] a relancé ces méthodes pour traiter de problèmes d'optimisation linéaires.

Dans un contexte d'optimisation sous contraintes d'inégalité, la méthode consiste à rechercher à l'itéré  $i$  le minimum de la fonction :

$$\phi_{\mu_i}(\mathbf{x}) = y(\mathbf{x}) - \mu_i \sum_{j=1}^{n_g} \ln(g_j(\mathbf{x}))$$

Après minimisation de ce problème (par méthode quasi-Newton), on passe à l'itéré  $i+1$  avec  $\mu_i > \mu_{i+1} > 0$ . Le paramètre  $\mu$  décroît et tend vers 0. Il est possible d'étendre cette approche aux contraintes d'égalité.

Des informations complémentaires concernant cette méthode (choix de la décroissante de  $\mu$  par exemple) ou sur ce type d'approche sont notamment disponibles dans [Wright, 1997, Nocedal et Wright, 2006].

## 2.3 Algorithmes d'optimisation globale

### 2.3.1 Introduction

Les algorithmes précédents assurent la convergence vers un minimum local. Afin d'essayer de trouver l'optimum global d'autres stratégies sont à envisager. Il s'agit notamment de développer des algorithmes qui soient à la fois capables d'explorer des zones de l'espace  $\Omega$  et d'exploiter des zones où des optima semblent se situer. Un équilibre entre diversification et intensification doit donc être établi. Il existe différentes façons de classer ces algorithmes (en fonction de leur règle d'évolutions, de l'utilisation ou non de mémoire etc.)

Quelques algorithmes envisageables sont rappelés dans cette liste :

- Méthodes énumératives [Land et Doig, 1960, Jones *et al.*, 1993]
- Méthodes à recherche aléatoire [Anderson, 1953, Rastrigin, 1963]
- Méthodes de regroupement *clustering* [Becker et Lago, 1970]
- Méthodes de descente généralisée [Levy et Montalvo, 1985]
- Recherche tabou [Glover, 1986, Dréo *et al.*, 2006]
- Recuit simulé [Kirkpatrick *et al.*, 1983]
- Méthodes évolutionnaires [Holland, 1975, Goldberg, 1989, Fogel, 1964, Rechenberg, 1973]
- Méthodes à intelligence en essaim (particulaires [Kennedy et Eberhart, 1995] ou colonies de fourmis [Dorigo, 1992, Dréo, 2004])

Dans notre cas d'étude, on se propose de décrire plus précisément les algorithmes génétiques issus des méthodes évolutionnaires et les algorithmes d'optimisation à essaim de particules. Ces méthodes sont des algorithmes métaheuristiques. Là encore, il est possible de classer ce genre d'algorithmes [Dréo, 2004, Dréo *et al.*, 2006]. Deux grandes classes se dégagent :

- Métaheuristiques de parcours. Ces approches utilisent une seule configuration ou solution que l'on fait évoluer. On peut notamment citer la recherche tabou et « Greddy Random Adaptive Search Procedures » [Feo et Resende, 1995] qui utilisent cette approche.
- Métaheuristiques de population. Ces approches consistent à échantillonner l'espace et à faire évoluer cette population suivant différentes règles.

Les algorithmes génétiques et à essaim de particules sont issus des algorithmes métaheuristiques à population et sont largement répandus dans la littérature. Ces deux approches sont présentées dans les sections suivantes.

### 2.3.2 Algorithmes génétiques

#### Description des algorithmes

Les algorithmes génétiques sont issus des algorithmes évolutionnaires. Ces méthodes s'inspirent de la théorie d'évolution des espèces décrite par le naturaliste anglais Charles Robert Darwin. Cette théorie décrit l'évolution des espèces comme étant une sélection naturelle des individus les plus adaptés à leur environnement. Ceux-ci en se reproduisant engendrent une progéniture plus résistante et mieux adaptée à son milieu.

Les premiers travaux sur ces algorithmes ont commencé dans les années 50. Cependant, au vu des faibles performances de calculs disponibles et aussi d'un certain manque de connaissances sur la génétique, ces premiers travaux ne furent pas convaincants. Ils reprirent durant les années 60 et 70. [Bäck *et al.*, 1997, Dréo *et al.*, 2006] distinguent trois variantes parmi ces algorithmes évolutionnaires : les *stratégies d'évolution* [Rechenberg, 1973], la *programmation évolutive* [Fogel, 1964] et les *algorithmes génétiques* [Goldberg, 1989].

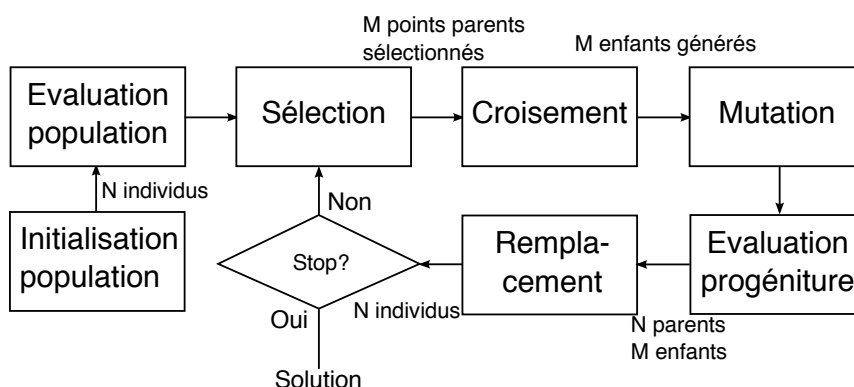


Dans ces algorithmes, le mimétisme de ces méthodes vient du fait que les individus de la population sont considérés comme des jeux de paramètres  $\mathbf{x}^{(k)}$  et que leurs performances sont évaluées via une fonction d'aptitude (*fitness function*). Cette fonction d'aptitude dans notre cas d'étude est la fonction objectif.

Le principe de ces algorithmes est :

- 1 Sélection probabiliste de certains individus en fonction de leur performance (valeur associée à la fonction objectif)
- 2 Croisement des individus pour en créer de nouveaux
- 3 Mutation spontanée de certains d'entre eux (perturbation aléatoire)
- 4 Disparition éventuelle de certains individus (Remplacement)

Les algorithmes génétiques se caractérisent par l'existence d'un génotype et d'un phénotype. Le phénotype est l'expression naturelle d'une solution  $\mathbf{x}^{(k)}$  qui est dans notre cas un vecteur. Le génotype est l'encodage en binaire de ce vecteur. Ce génotype est important lors de la phase de croisement entre individus puisque l'on vient créer un « enfant » à partir du croisement entre deux génotypes « parents ».



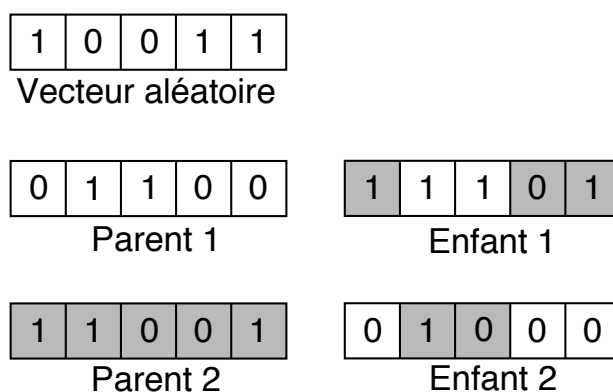
**FIGURE 1.8:** Présentation générale de la méthode d'optimisation par algorithmes génétiques

### Quelques précisions

Dans chacune des étapes décrites précédemment, différents choix doivent être effectués afin de mener à bien l'optimisation. Concernant l'étape de sélection des individus pour le croisement, il est nécessaire de choisir en plus forte probabilité les meilleurs individus (au sens de l'évaluation de la fonction objectif). Cette nécessité provient du fait que l'on souhaite éviter une dérive génétique, c'est-à-dire la sélection d'individus mal adaptés. Il faut cependant conserver une diversité de la population, afin de laisser une partie exploratoire à l'algorithme et éviter une optimisation purement locale. [Dréo *et al.*, 2006] établit un récapitulatif de différentes méthodes de choix. On peut notamment citer les méthodes basées sur le rang des individus. Cette méthode classe les différents individus de la population en fonction de leur estimation de la fonction objectif et leur attribue une valeur calculée grâce à une fonction de mise à l'échelle (*scaling function*).

Par exemple, par défaut sous Matlab cette fonction est  $f_m = \frac{1}{\sqrt{r}}$  avec  $r$  le rang de l'individu. La probabilité de sélection des individus est donc proportionnelle à cette évaluation sur la fonction de mise à l'échelle<sup>3</sup>. Le choix des individus suit là encore certaines techniques (roues biaisées, stochastic universal sampling method, tournoi).

Concernant le croisement, là encore, différents choix sont possibles [Dréo *et al.*, 2006]. Une méthode couramment employée est la méthode de croisement uniforme, représentée sur la figure 1.9. Un vecteur binaire aléatoire ( $V_{ect1}$ ) est créé. En fonction de la valeur d'un bit de ce vecteur, le premier ou le second parent est choisi. Sur la figure 1.9 lorsque le bit du vecteur aléatoire  $V_{ect1}$  vaut 0, le bit du premier parent est choisi, s'il vaut 1 le second parent est choisi. Un deuxième enfant est créé en prenant le vecteur complémentaire de  $V_{ect1}$ .



**FIGURE 1.9:** Croisement uniforme

Pour la mutation, le code binaire est modifié localement suivant une certaine probabilité. Finalement, une fois le croisement et la mutation terminés, une nouvelle population est créée en sélectionnant un certain nombre d'individus parmi ceux disponibles. Un choix possible est de conserver dans une certaine proportion enfants et parents, une autre possibilité est de conserver uniquement les meilleurs individus (stratégie élitiste).

### 2.3.3 Optimisation par essais de particules

Originellement, ces algorithmes ont été proposés par [Kennedy et Eberhart, 1995] en 1995. Ces algorithmes s'inspirent du monde vivant et plus particulièrement du comportement social des animaux évoluant en essaim. Des particules sont réparties de manière aléatoire sur l'espace de recherche : l'essaim. Chaque particule est une évaluation de la fonction objectif. Ces particules possèdent une mémoire leur permettant de se souvenir de leur meilleure position (au sens de l'évaluation de la fonction objectif) ainsi que de celle atteinte par des particules voisines appelées informatrices. Chaque particule  $j$  possède donc une position  $\mathbf{x}_i^j$  à l'itéré  $i$  mais également une vitesse  $\mathbf{v}_i^j$ . L'essaim évolue au fur et à mesure des itérations selon trois composantes [Kennedy et Eberhart, 1995, Clerc et Siarry, 2004, Clerc, 2005] :

- Une composante *d'inertie* : la particule tend à suivre sa trajectoire.

3. <http://fr.mathworks.com/help/gads/genetic-algorithm-options.html>

- Une composante *conservatrice* ou *cognitive* : la particule tend à se diriger vers la meilleure position qu'elle a trouvée.
- Une composante *panurgienne* ou *sociale* : la particule tend à se diriger vers la meilleure position atteinte par ses informatrices.

Une itération de l'algorithme consiste donc à modifier les positions des particules en suivant ces trois règles selon les équations 1.21 :

$$\mathbf{v}_{i+1}^j = c_1 \mathbf{v}_i^j + c_2 \times r_2 (\mathbf{x}_p^j - \mathbf{x}_i^j) + c_3 \times r_3 (\mathbf{x}^{best} - \mathbf{x}_i^j) \quad (1.21)$$

$$\mathbf{x}_{i+1}^j = \mathbf{x}_i^j + \mathbf{v}_{i+1}^j$$

$\mathbf{x}_p^j$  est la meilleure position atteinte par la particule  $j$ ,  $\mathbf{x}^{best}$  est la meilleure position atteinte par les informatrices.  $r_2$  et  $r_3$  sont tirés aléatoirement dans l'intervalle  $[0, 1]$ , de plus, usuellement  $c_1 \approx 1$ .

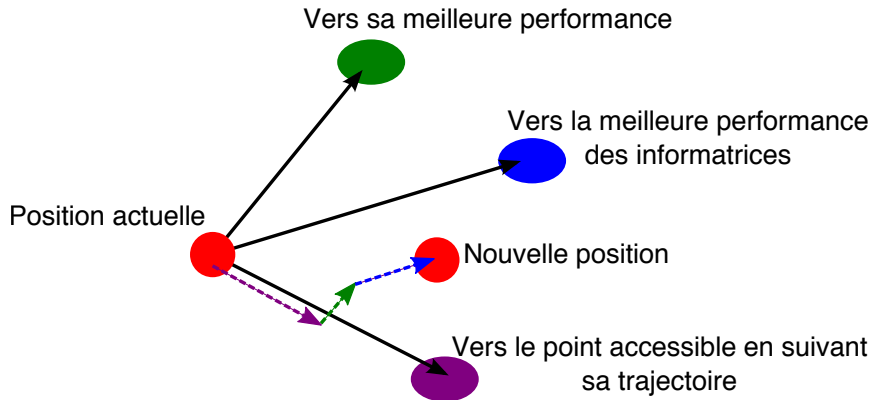


FIGURE 1.10: Schéma d'évolution d'une particule

Il existe des variantes de cet algorithme afin d'en assurer une bonne convergence et éviter qu'il ne diverge (définition d'une vitesse  $V_{max}$  [Eberhart *et al.*, 1996], rebond d'une particule sur la frontière du domaine pour éviter qu'elle n'en sorte, d'autres possibilités sont présentées dans [Kennedy, 1998, Ozcan et Mohan, 1999]). Il existe une version dite de « constriction » [Clerc et Kennedy, 2002] qui limite la vitesse des particules. L'équation de vitesse est modifiée par :

$$\mathbf{v}_{i+1}^j = \chi (c_1 \mathbf{v}_i^j + c_2 \times r_2 (\mathbf{x}_p^j - \mathbf{x}_i^j) + c_3 \times r_3 (\mathbf{x}^{best} - \mathbf{x}_i^j)) \quad (1.22)$$

avec :

$$\chi = \frac{2}{c_2 + c_3 - 2 + \sqrt{(c_2 + c_3)^2 - 4(c_2 + c_3)}} \quad (1.23)$$

En général on fixe  $c_2 = c_3$  et  $c_2 + c_3 = 4.1$ , ce qui donne  $\chi \approx 0.73$ .

Dans cette section, différents algorithmes d'optimisation ont été présentés. Cependant, comme précisé précédemment, ces algorithmes ne sont pas adaptés à la résolution directe de problèmes d'assemblages mécaniques. En effet, les coûts de calcul des algorithmes génétiques ou à essais particuliers seraient trop importants pour être utilisés comme moyen direct d'optimisation. Les algorithmes à optimisation locale sont certes moins coûteux, mais ne répondent pas nécessairement correctement au problème d'optimisation globale que l'on souhaite résoudre. Afin d'y remédier, il est proposé d'utiliser un modèle de substitution moins coûteux à évaluer et supposé être représentatif du modèle mécanique réel. On se propose dans la prochaine section d'introduire les métamodèles et notamment plus précisément ceux s'appuyant sur le krigeage ainsi que certains outils utilisés dans la suite de cette thèse.

### 3 Utilisation de métamodèle

Un métamodèle est un **modèle de substitution**, il permet l'évaluation en tout point de l'espace  $\mathcal{D}$  d'une fonction *a priori* coûteuse à estimer par un solveur numérique. Un métamodèle, dans le cadre de ce travail est une surface de réponse. Dans la littérature ([Robinson *et al.*, 2008]), l'appellation « modèle de substitution » peut être plus générale et peut s'apparenter aux méthodes de surface de réponse ou encore aux modèles réduits type POD, PGD ou analyse modale. Dans cette section, on ne s'intéressera donc qu'aux méthodes de surface de réponse consistant à élaborer une fonction mathématique s'appuyant sur les données évaluées dans l'espace de conception  $\mathcal{D}$ .

Les modèles de type réduit POD (Proper Orthogonal Decomposition) ou PGD (Proper Generalized Decomposition) ne seront pas abordés dans ce travail, cependant le lecteur intéressé pourra trouver des renseignements dans [Chinesta *et al.*, 2011, Relun, 2011].

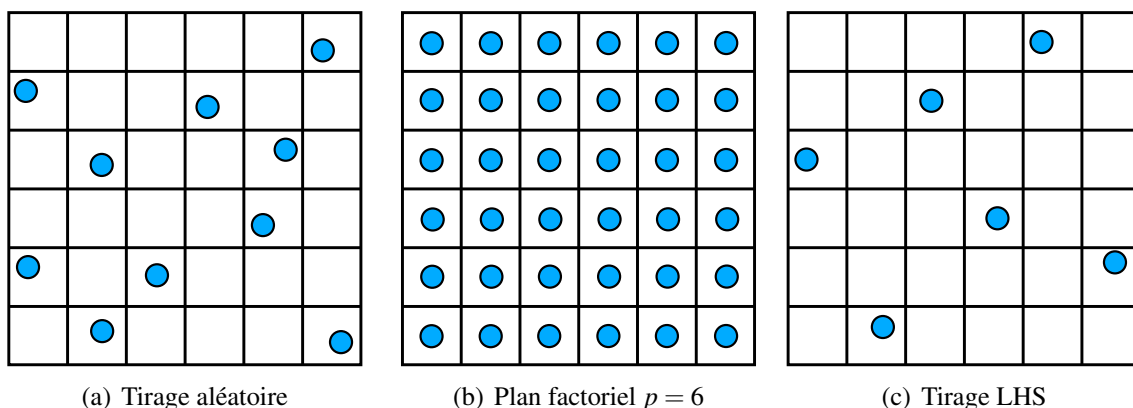
Parmi les métamodèles, on distingue deux grandes catégories : les métamodèles non-paramétriques et les métamodèles paramétriques. Ces derniers nécessitent de choisir des paramètres internes au métamodèle afin d'en définir son comportement. Parmi les métamodèles paramétriques couramment utilisés on trouve les méthodes RBFs (Radial basis Function) [Orr, 1996, Gibbs, 1997] et de krigeage [Krige, 1951, Matheron, 1969, Sacks *et al.*, 1989, Jones, 2001]. Parmi les métamodèles non-paramétriques, on pourra notamment citer les méthodes de régression [Box et Wilson, 1951], les moindres carrés mobiles [Lancaster et Šalkauskas, 1981] ou encore les méthodes par interpolation par spline cubique.

Comme rappelé précédemment il existe quatre points pour la création du métamodèle :

- Le choix mathématique du métamodèle
- L'échantillonnage de l'espace de conception  $\mathcal{D}$ .
- La construction du métamodèle en elle-même
- La validation du métamodèle.

Dans cette section, on ne présente que les métamodèles de type krigeage. En effet, ce type de métamodèle s'est avéré être performant dans le cadre de la mécanique des assemblages [Laurent, 2013].

Il est cependant à remarquer qu'un thème comme l'échantillonnage de l'espace est une problématique de recherche à elle toute seule [Myšáková *et al.*, 2014]. N'étant pas au cœur de cette thèse, les méthodes d'échantillonnage ne font pas l'objet d'une étude complète au cours de ce travail. Celles-ci sont indispensables à l'obtention d'un métamodèle représentatif de la fonction que l'on souhaite estimer (voir chapitre 4). Un échantillonnage de l'espace très couramment utilisé est le tirage LHS [McKay *et al.*, 1979]. Il consiste à diviser l'espace de conception en plusieurs parties et à ne placer qu'un point à évaluer par ligne et colonne. **C'est ce type de tirage qui sera à la base des échantillonnages de l'espace effectués au cours de cette thèse.** D'autres tirages sont possibles. On peut notamment citer deux méthodes répandues que sont les tirages aléatoires (de type Monte-Carlo par exemple) ou de plan factoriel. Ce dernier peut amener à évaluer un nombre important de points :  $n_s = p^{n_{dim}}$  avec  $p$  le nombre de points souhaités par dimension.



**FIGURE 1.11:** Exemples de différents tirages possibles sur un espace à 2 paramètres

### 3.1 Construction de métamodèle par krigeage

Comme dit précédemment, on peut trouver différents types de construction de métamodèles. On présente ici, l'approche de krigeage [Krigé, 1951, Matheron, 1969, Jones, 2001] qui a montré une réelle efficacité de création de métamodèles dans le cadre de la mécanique des assemblages [Laurent, 2013].

#### 3.1.1 Définitions

Le krigeage a pour but de fournir un métamodèle basé sur une estimation probabiliste en fonction des données évaluées. On considère pour cela que la fonction objectif  $y$  est la réalisation d'un processus gaussien stationnaire  $Y$ . Il fut notamment développé dans le cadre de la géostatistique

[Krige, 1951, Matheron, 1963, Matheron, 1970]. Il est finalement utilisé dans le cadre de création de métamodèles par simulations numériques dans [Sacks *et al.*, 1989]. On pourra notamment se référer à [Rasmussen et Williams, 2006] pour plus de précisions sur le formalisme du krigeage et des processus gaussiens. Il est répandu dans la création de métamodèles notamment pour sa robustesse. De plus, il fournit une estimation de l'erreur des moindres carrés du processus qu'il estime. On dispose donc d'un indicateur intrinsèque de qualité du métamodèle.

Le processus gaussien  $Y$  est formulé de la façon suivante :

$$Y(\mathbf{x}) = \mu(\mathbf{x}) + Z(\mathbf{x}) \quad (1.24)$$

$\mu(\mathbf{x})$  est une quantité déterministe et  $Z(\mathbf{x})$  une quantité aléatoire suivant un processus gaussien stationnaire :

$$Z(\mathbf{x}) \sim \mathcal{N}(0, \sigma^2) \quad (1.25)$$

ce qui est équivalent à :

$$\begin{aligned} E[z(\mathbf{x})] &= 0 \text{ Moyenne du processus nulle} \\ E[Z(\mathbf{x})^2] &= \sigma^2 \text{ Variance constante du processus} \end{aligned} \quad (1.26)$$

Ainsi l'espérance et la variance de  $Y$  s'écrivent :

$$E[Y(\mathbf{x})] = E[\mu(\mathbf{x}) + Z(\mathbf{x})] = \mu(\mathbf{x}) \quad (1.27)$$

$$E[Y(\mathbf{x})^2] = \text{Var}[Y(\mathbf{x})] = \text{Var}[\mu(\mathbf{x}) + Z(\mathbf{x})] = \text{Var}[Z(\mathbf{x})] = \sigma^2 \quad (1.28)$$

On formule également la fonction de covariance du processus gaussien par :

$$k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \text{cov}[Y(\mathbf{x}^{(i)}), Y(\mathbf{x}^{(j)})] = \text{cov}[Z(\mathbf{x}^{(i)}), Z(\mathbf{x}^{(j)})] \quad (1.29)$$

De plus, selon la nature de  $\mu(\mathbf{x})$  on distingue classiquement trois types de krigeage.

- Krigeage simple :  $\mu(\mathbf{x}) = m$  avec  $m$  une constante donnée.
- Krigeage ordinaire :  $\mu(\mathbf{x}) = \beta$  avec  $\beta$  une constante inconnue.
- Krigeage universel :  $\mu(\mathbf{x}) = \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta} = \sum_{i=1}^p \beta_i f_i(\mathbf{x})$  avec  $\mathbf{f}(\mathbf{x})$  une fonction de régression polynomiale et  $\boldsymbol{\beta}$  le vecteur des coefficients de régression. Si on suppose une régression d'ordre 0 à savoir  $\mathbf{f}(\mathbf{x}) = 1$ , alors on se retrouve dans le cas du krigeage ordinaire. On pose ainsi  $\mathbf{f}(\mathbf{x}) = [ f_1(\mathbf{x}) \quad \dots \quad f_p(\mathbf{x}) ]^T$  et  $\boldsymbol{\beta} = [ \beta_1 \quad \dots \quad \beta_p ]^T$

Dans la suite, on ne considérera les équations que sous la forme du krigeage universel.

### 3.1.2 Formulation du krigage

On pose  $\mathbf{X}$  le vecteur des  $n_s$  points évalués issus d'un échantillonnage de l'espace de conception :

$$\mathbf{X} = [ \mathbf{x}^{(1)} \quad \dots \quad \mathbf{x}^{(n_s)} ]^T \quad (1.30)$$

$\mathbf{Y}_{n_s}$  est le vecteur des observations du processus gaussien et  $\mathbf{y}_{n_s}$ , celui des évaluations de la fonction  $y$ . Ainsi :

$$\begin{aligned} \mathbf{Y}_{n_s} &= [ Y(\mathbf{x}^{(1)}) \quad \dots \quad Y(\mathbf{x}^{(n_s)}) ]^T \\ \mathbf{y}_{n_s} &= [ y(\mathbf{x}^{(1)}) \quad \dots \quad y(\mathbf{x}^{(n_s)}) ]^T \end{aligned} \quad (1.31)$$

Par la suite, on posera  $y(\mathbf{x}^{(i)}) = y^{(i)}$  et  $Y(\mathbf{x}^{(i)}) = Y^{(i)}$ .

Le but est de construire un estimateur  $\hat{Y}(\mathbf{x})$  de la valeur en un point ou site quelconque  $\mathbf{x} \in \mathcal{D}$  du processus aléatoire  $Y$ . Cet estimateur est recherché sous une combinaison linéaire du type :

$$\hat{Y}(\mathbf{x}) = \mathbf{w}^T(\mathbf{x})\mathbf{Y}_{n_s} \quad (1.32)$$

Le vecteur  $\mathbf{w}^T(\mathbf{x}) = [ w_1(\mathbf{x}) \quad \dots \quad w_{n_s}(\mathbf{x}) ] \in \mathbb{R}^{n_s}$  est constitué des coefficients du krigage qui sont les inconnues du problème. L'estimateur  $\hat{Y}(\mathbf{x})$  est recherché sous la forme du meilleur estimateur linéaire non biaisé (BLUP - *Best Linear Unbiased Predictor*). On cherche donc à trouver  $\hat{Y}(\mathbf{x})$  tel que :

$$\left\{ \begin{array}{l} \mathbb{E}[\hat{Y}(\mathbf{x}) - Y(\mathbf{x})] = 0 \\ \hat{Y}(\mathbf{x}) \text{ minimise l'erreur quadratique moyenne } \implies \min \left( \mathbb{E} \left[ \left( \hat{Y}(\mathbf{x}) - Y(\mathbf{x}) \right)^2 \right] \right) \end{array} \right. \quad (1.33)$$

On peut calculer alors :

$$\begin{aligned} \hat{Y}(\mathbf{x}) - Y(\mathbf{x}) &= \mathbf{w}^T(\mathbf{x})\mathbf{Y}_{n_s} - Y(\mathbf{x}) \\ \hat{Y}(\mathbf{x}) - Y(\mathbf{x}) &= \mathbf{w}^T(\mathbf{x})(\mathbf{F}\boldsymbol{\beta} + \mathbf{Z}_{n_s}) - (\mathbf{f}(\mathbf{x})^T\boldsymbol{\beta} + Z(\mathbf{x})) \\ \hat{Y}(\mathbf{x}) - Y(\mathbf{x}) &= \underbrace{\mathbf{w}^T(\mathbf{x})\mathbf{Z}_{n_s} - Z(\mathbf{x})}_{\text{Processus gaussien}} + \underbrace{(\mathbf{F}^T\mathbf{w}(\mathbf{x}) - \mathbf{f}(\mathbf{x}))^T\boldsymbol{\beta}}_{\text{Déterministe}} \end{aligned} \quad (1.34)$$

avec :

$$\mathbf{F} = \begin{bmatrix} f_1(\mathbf{x}^{(1)}) & f_2(\mathbf{x}^{(1)}) & \dots & f_p(\mathbf{x}^{(1)}) \\ f_1(\mathbf{x}^{(2)}) & \ddots & & \vdots \\ \vdots & & & \vdots \\ f_1(\mathbf{x}^{(n_s)}) & \dots & \dots & f_p(\mathbf{x}^{(n_s)}) \end{bmatrix}$$

et :

$$\mathbf{Z}_{n_s} = [ Z(\mathbf{x}^{(1)}) \quad \dots \quad Z(\mathbf{x}^{(n_s)}) ]^T$$

La condition de non-biais s'exprime alors comme étant :

$$E[\widehat{Y}(\mathbf{x}) - Y(\mathbf{x})] = 0 \iff \mathbf{F}^T \mathbf{w}(\mathbf{x}) - \mathbf{f}(\mathbf{x}) = \mathbf{0} \quad (1.35)$$

Enfin, on cherche à minimiser l'erreur quadratique moyenne  $MSE$ .

$$\begin{aligned} MSE &= E \left[ \left( \widehat{Y}(\mathbf{x}) - Y(\mathbf{x}) \right)^2 \right] \\ &= E \left[ \left( \mathbf{w}^T \mathbf{Z}_{n_s} - Z(\mathbf{x}) \right)^2 \right] \\ &= E[Z(\mathbf{x})^2] + E[\mathbf{w}^T(\mathbf{x}) \mathbf{Z}_{n_s} \mathbf{Z}_{n_s}^T \mathbf{w}(\mathbf{x})] - 2E[\mathbf{w}^T(\mathbf{x}) \mathbf{Z}_{n_s} Z(\mathbf{x})] \\ &= \sigma^2 + \mathbf{w}^T(\mathbf{x}) E[\mathbf{Z}_{n_s} \mathbf{Z}_{n_s}^T] \mathbf{w}(\mathbf{x}) - 2\mathbf{w}^T(\mathbf{x}) E[\mathbf{Z}_{n_s} Z(\mathbf{x})] \\ &= \sigma^2 + \mathbf{w}^T(\mathbf{x}) \mathbf{C} \mathbf{w}(\mathbf{x}) - 2\mathbf{w}^T E[\mathbf{Z}_{n_s} Z(\mathbf{x})] \end{aligned} \quad (1.36)$$

La quantité  $E[\mathbf{Z}_{n_s} \mathbf{Z}_{n_s}^T]$  est égale à  $\mathbf{C}$ . Cette matrice est la matrice de covariance des données  $\mathbf{Z}_{n_s}$ . Elle est définie par :

$$\mathbf{C} = \begin{bmatrix} \text{Cov} \left( Z(\mathbf{x}^{(1)}), Z(\mathbf{x}^{(1)}) \right) & \cdots & \text{Cov} \left( Z(\mathbf{x}^{(1)}), Z(\mathbf{x}^{(n_s)}) \right) \\ \vdots & \ddots & \vdots \\ \text{Cov} \left( Z(\mathbf{x}^{(1)}), Z(\mathbf{x}^{(n_s)}) \right) & \cdots & \text{Cov} \left( Z(\mathbf{x}^{(n_s)}), Z(\mathbf{x}^{(n_s)}) \right) \end{bmatrix} \quad (1.37)$$

Cette matrice de covariance est donc à déterminer. Il s'agit du cœur du problème de construction du métamodèle. Afin de calculer cette matrice, [Sacks *et al.*, 1989] propose l'utilisation d'une fonction de corrélation. On définit alors :  $\text{Cov} \left( Z(\mathbf{x}^{(i)}), Z(\mathbf{x}^{(j)}) \right) = \sigma^2 R(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}; \mathbf{l})$ .  $R$  est donc une fonction de corrélation ne tenant compte que de la distance entre deux points et d'un paramètre de corrélation  $\mathbf{l}$ . Cette fonction a notamment pour propriétés d'être égale à 1 lorsque que  $\mathbf{x}^{(i)} = \mathbf{x}^{(j)}$ , afin de vérifier la relation de l'équation 1.28. Cette fonction est également définie comme une fonction paire.

$$R(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}; \mathbf{l}) = \prod_{k=1}^{n_{dim}} h \left( x_k^{(i)} - x_k^{(j)}; l_k \right) \quad (1.38)$$

On présente au tableau 1.2, un récapitulatif de certaines fonction  $h(r; l_k)$  avec  $r = x_k^{(i)} - x_k^{(j)}$ . On peut donc reformuler à l'aide de cette fonction de corrélation  $R$ , la matrice  $\mathbf{C}$  :

$$\mathbf{C} = \sigma^2 \begin{bmatrix} R(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}; \mathbf{l}) & \cdots & R(\mathbf{x}^{(1)}, \mathbf{x}^{(n_s)}; \mathbf{l}) \\ \vdots & \ddots & \vdots \\ R(\mathbf{x}^{(n_s)}, \mathbf{x}^{(1)}; \mathbf{l}) & \cdots & R(\mathbf{x}^{(n_s)}, \mathbf{x}^{(n_s)}; \mathbf{l}) \end{bmatrix} = \sigma^2 \mathbf{R} \quad (1.39)$$

La matrice  $\mathbf{C}$  est donc une matrice symétrique définie positive de par les propriétés de la fonction  $R$ , ce qui assure son inversibilité.



Nom	$h(r; l_k)$	Paramètres
Exponentielle	$\exp(-l_k  r )$	$l_k > 0$
Exponentielle généralisée	$\exp(-l_k  r ^p)$	$l_k > 0$ et $0 < p \leq 2$
Gaussienne	$\exp(-l_k  r ^2)$	$l_k > 0$
Linéaire	$\max(0, 1 - l_k  r )$	$l_k > 0$
Spline	$\zeta(\xi)$ avec $\xi = l_k  r $ et $\zeta(\xi) = \begin{cases} 1 - 15\xi^2 + 30\xi^3 & \text{pour } 0 \leq \xi \leq 0.2 \\ 1.25(1 - \xi)^3 & \text{pour } 0.2 \leq \xi \leq 1 \\ 0 & \text{pour } \xi > 1 \end{cases}$	$l_k > 0$
Matérn $\nu = 3/2$	$\left(1 + \frac{\sqrt{3} r }{l_k}\right) \exp\left(-\frac{\sqrt{3} r }{l_k}\right)$	$l_k > 0$
Matérn $\nu = 5/2$	$\left(1 + \frac{\sqrt{5} r }{l_k} + \frac{5r^2}{3l_k^2}\right) \exp\left(-\frac{\sqrt{5} r }{l_k}\right)$	$l_k > 0$

TABLE 1.2: Récapitulatif de certaines fonctions de corrélation usuelles pour le krigeage

On peut réécrire l'erreur quadratique moyenne sous la forme :

$$MSE = \sigma^2 + \sigma^2 \mathbf{w}^T(\mathbf{x}) \mathbf{R} \mathbf{w}(\mathbf{x}) - 2\sigma^2 \mathbf{w}^T(\mathbf{x}) \mathbf{r}(\mathbf{x}) \quad (1.40)$$

Le vecteur  $\mathbf{r}(\mathbf{x}) = [R(\mathbf{x}^{(1)}, \mathbf{x}; \mathbf{l}) \quad \dots \quad R(\mathbf{x}^{(n_s)}, \mathbf{x}; \mathbf{l})]^T$  correspond à la corrélation entre un point  $\mathbf{x}$  quelconque et les points  $\mathbf{x}^{(i)}$  du plan d'expérience.

Dans le but de trouver le meilleur estimateur linéaire non biaisé  $\hat{Y}$ , un lagrangien est défini à l'équation 1.41 afin de prendre en compte la condition de non-biais définie par l'équation 1.35 :

$$\mathcal{L}(\mathbf{w}(\mathbf{x}), \boldsymbol{\lambda}) = \sigma^2 (1 + \mathbf{w}^T(\mathbf{x}) \mathbf{R} \mathbf{w}(\mathbf{x}) - 2\mathbf{w}^T(\mathbf{x}) \mathbf{r}(\mathbf{x})) + \boldsymbol{\lambda}^T (\mathbf{F}^T \mathbf{w}(\mathbf{x}) - \mathbf{f}(\mathbf{x})) \quad (1.41)$$

On trouve alors :

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{w}(\mathbf{x})} &= 2\sigma^2 (\mathbf{R} \mathbf{w}(\mathbf{x}) - \mathbf{r}(\mathbf{x})) + \mathbf{F} \boldsymbol{\lambda} = 0 \\ \frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} &= \mathbf{F}^T \mathbf{w}(\mathbf{x}) - \mathbf{f}(\mathbf{x}) = 0 \end{aligned} \quad (1.42)$$

Ce qui amène au système :

$$\begin{bmatrix} \mathbf{R} & \mathbf{F} \\ \mathbf{F}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{r} \\ \mathbf{f} \end{bmatrix} \quad (1.43)$$

La solution à ce système est donnée sous la forme :

$$\begin{aligned}\frac{\lambda}{2\sigma^2} &= (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{r}(\mathbf{x}) - \mathbf{f}(\mathbf{x})) \\ \mathbf{w}(\mathbf{x}) &= \mathbf{R}^{-1} (\mathbf{r}(\mathbf{x}) - \mathbf{F} \frac{\lambda}{2\sigma^2})\end{aligned}\quad (1.44)$$

Au final l'estimateur  $\hat{Y}$  s'écrit sous la forme :

$$\begin{aligned}\hat{Y}(\mathbf{x}) &= \mathbf{w}(\mathbf{x})^T \mathbf{Y}_{n_s} \\ \hat{Y}(\mathbf{x}) &= \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta} + \mathbf{r}(\mathbf{x})^T \mathbf{R}^{-1} (\mathbf{Y}_{n_s} - \mathbf{F}\boldsymbol{\beta}) \\ \text{avec } \boldsymbol{\beta} &= (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{Y}_{n_s})\end{aligned}\quad (1.45)$$

On peut également déterminer l'erreur quadratique moyenne :

$$\begin{aligned}MSE &= \sigma^2 (1 + \mathbf{w}^T(\mathbf{x}) (\mathbf{R}\mathbf{w}(\mathbf{x}) - 2\mathbf{r}(\mathbf{x}))) \\ &= \sigma^2 (1 + \mathbf{u}^T(\mathbf{x}) (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{u}(\mathbf{x}) - \mathbf{r}^T(\mathbf{x}) \mathbf{R}^{-1} \mathbf{r}(\mathbf{x})) \\ \text{avec } \mathbf{u}(\mathbf{x}) &= \mathbf{F}^T \mathbf{R}^{-1} \mathbf{r}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\end{aligned}\quad (1.46)$$

Ainsi, lorsque l'on cherche à déterminer la distribution  $Y(\mathbf{x})$  conditionnellement à l'événement  $\tilde{\mathcal{E}} = \{Y^{(i)} = y^{(i)}, \forall 1 \leq i \leq n_s\}$  on obtient :

$$Y(\mathbf{x}) \mid \tilde{\mathcal{E}} = \mathcal{N}(\hat{y}(\mathbf{x}), s_{\hat{y}}^2) \quad (1.47)$$

avec :

$$\begin{aligned}\hat{y}(\mathbf{x}) &= \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta} + \mathbf{r}(\mathbf{x})^T \mathbf{R}^{-1} (\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta}) \\ s_{\hat{y}}^2(\mathbf{x}) &= \sigma^2 (1 + \mathbf{u}^T(\mathbf{x}) (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{u}(\mathbf{x}) - \mathbf{r}^T(\mathbf{x}) \mathbf{R}^{-1} \mathbf{r}(\mathbf{x}))\end{aligned}\quad (1.48)$$

$\hat{y}(\mathbf{x})$  est donc le métamodèle construit par krigeage.  $s_{\hat{y}}^2(\mathbf{x})$  est quant à lui un estimateur de la variance de  $\hat{y}(\mathbf{x})$  issu du processus gaussien.

### 3.1.3 Précision sur la recherche des paramètres du krigeage

La section précédente a mis en avant différents paramètres pour la construction du krigeage ( $\boldsymbol{\beta}$ ,  $\sigma$  et  $\mathbf{I}$ ). Seul  $\boldsymbol{\beta}$  est pour l'instant déterminé. Il reste donc les quantités  $\sigma^2$  et  $l_k$  (paramètres de la fonction de corrélation utilisée) à définir. Pour cela, il est proposé de chercher ces quantités comme le maximum d'une fonction de vraisemblance  $L$  [Kitanidis, 1983]. Cette fonction est définie comme suit :

$$\begin{aligned}L(\boldsymbol{\beta}, \sigma^2, \mathbf{I}) &= \frac{1}{\sqrt{(2\pi)^{n_s} |\mathbf{C}|}} e^{-\frac{(\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta})^T \mathbf{C}^{-1} (\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta})}{2\sigma^2}} \\ L(\boldsymbol{\beta}, \sigma^2, \mathbf{I}) &= \frac{1}{\sqrt{(2\pi)^{n_s} (\sigma^2)^{n_s} |\mathbf{R}|}} e^{-\frac{(\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta})^T \mathbf{R}^{-1} (\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta})}{2\sigma^2}}\end{aligned}\quad (1.49)$$

Afin de faciliter la recherche du maximum on utilise le logarithme de la fonction qui, en levant les constantes, devient :

$$\ln L(\boldsymbol{\beta}, \sigma^2, \mathbf{I}) = -\frac{n}{2} \ln \sigma^2 - \frac{1}{2} \ln |\mathbf{R}| - \frac{(\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta})^T \mathbf{R}^{-1} (\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta})}{2\sigma^2} \quad (1.50)$$

On peut retrouver la valeur de  $\boldsymbol{\beta}$  trouvée précédemment en cherchant le vecteur  $\boldsymbol{\beta}$  maximisant la fonction de vraisemblance.

$$\frac{\partial \ln L}{\partial \boldsymbol{\beta}} = 0 \implies \boldsymbol{\beta} = (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{y}_{n_s}) \quad (1.51)$$

On fait de même pour le scalaire  $\sigma^2$  :

$$\frac{\partial \ln L}{\partial \sigma^2} = 0 \implies \sigma^2 = \frac{(\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta})^T \mathbf{R}^{-1} (\mathbf{y}_{n_s} - \mathbf{F}\boldsymbol{\beta})}{n_s} \quad (1.52)$$

Les paramètres  $l_k$  influant sur la matrice de corrélation  $\mathbf{R}$  ne sont quant à eux pas déterminables de manière analytique. Ils sont donc déterminés à l'aide d'un algorithme d'optimisation numérique de la fonction de vraisemblance logarithmique concentrée :

$$\ln L(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2, \mathbf{I}) = -\frac{n_s}{2} \ln \hat{\sigma}^2 - \frac{1}{2} \ln |\mathbf{R}|$$

C'est d'ailleurs cette étape de construction du métamodèle qui est la plus difficile numériquement. En effet, le conditionnement de la matrice  $\mathbf{R}$  pouvant être mauvais (voir même très mauvais) cette étape est rendue complexe par le calcul de  $\mathbf{R}^{-1}$  et peut amener à trouver des paramètres  $l_k$  peu adaptés à la construction du métamodèle. De plus, la fonction  $L$  peut être multimodale. Un algorithme d'optimisation globale est donc préférable. C'est d'ailleurs ce qui est proposé dans la toolbox GREMAT.

**Remarque :** Cette méthode du maximum de vraisemblance est actuellement la méthode implantée dans la toolbox GREMAT. Il en existe cependant d'autres comme notamment la méthode de validation croisée. On utilise dans ce cas la méthode connue sous le nom de *Leave One-Out*. En notant  $\hat{y}_{-i}$  le métamodèle construit en l'absence de la réponse  $y(\mathbf{x}^{(i)})$ , on calcule la validation de la manière suivante :

$$\text{LOO}(\mathbf{I}) = \sum_{i=1}^{n_s} \left( \hat{y}_{-i}(\mathbf{x}^{(i)}) - y(\mathbf{x}^{(i)}) \right)^2 \quad (1.53)$$

On cherche alors à déterminer les paramètres  $\hat{\mathbf{I}}$  tels que :

$$\hat{\mathbf{I}} = \arg \min \text{LOO}(\mathbf{I}) \quad (1.54)$$

Cette méthode peut s'avérer coûteuse à mettre en place car il y a  $n_s$  métamodèles à mettre en place. Le lecteur intéressé peut notamment se référer aux travaux de [Rippa, 1999] proposant une méthode efficace de calcul.

## 3.2 Qualité du métamodèle

Après avoir élaboré un métamodèle, on présente dans les prochains paragraphes quelques moyens de quantifier sa qualité. Certaines techniques ne s'appuient que sur les données disponibles (telles que les méthodes de validation croisée), d'autres nécessitent des évaluations supplémentaires. On introduit dans cette section ces dernières approches car ce sont ces indicateurs qui seront principalement utilisés pour quantifier la qualité des métamodèles établis.

### 3.2.1 EMC, REMC

L'erreur des moindres carrées (*Mean Square Error*) est un indicateur classiquement utilisé pour évaluer l'écart entre deux fonctions. On l'utilise spécialement dans le cas de régression polynomiale où il permet de trouver les meilleurs coefficients de la régression après un processus de minimisation de cet indicateur. On peut également définir un indicateur basé sur la racine carrée du précédent (REMC). Ces deux critères permettent d'évaluer l'écart entre la fonction réelle  $y$  et la fonction approchée  $\hat{y}$  pour un ensemble de points échantillonnés  $n_v$ . Ces critères sont définis comme :

$$\text{EMC} = \frac{1}{n_v} \sum_{i=1}^{n_v} \left( y(\mathbf{x}^{(i)}) - \hat{y}(\mathbf{x}^{(i)}) \right)^2 \quad \text{EMC} = \sqrt{\text{EMC}} \quad (1.55)$$

Cependant, dans le cadre des modèles déterministes avec méthode d'interpolation (tel que le krigeage) les points échantillonnés ne doivent pas être (du moins uniquement) ceux sur lesquels s'appuie le modèle d'interpolation.

*Plus ces indicateurs sont faibles, plus le métamodèle est précis.*

### 3.2.2 $r^2$

Le coefficient de corrélation linéaire de Bravais-Pearson est également couramment utilisé. Il traduit une dépendance **linéaire** entre deux variables en un nombre de points échantillonnés  $n_v$ . Il fournit une indication sur la qualité des variations d'une réponse par rapport à l'autre.

Ce coefficient s'écrit :

$$r^2 = \left( \frac{\sigma_{xy}}{\sigma_x \sigma_y} \right)^2 \quad (1.56)$$

avec :

$$\begin{aligned} \sigma_{xy} &= \frac{1}{n_v} \sum_{i=1}^{n_v} \left( y(\mathbf{x}^{(i)}) - \bar{y} \right) \left( \hat{y}(\mathbf{x}^{(i)}) - \bar{\hat{y}} \right) \\ \sigma_x &= \sqrt{\frac{1}{n_v} \sum_{i=1}^{n_v} \left( y(\mathbf{x}^{(i)}) - \bar{y} \right)^2} \\ \sigma_y &= \sqrt{\frac{1}{n_v} \sum_{i=1}^{n_v} \left( \hat{y}(\mathbf{x}^{(i)}) - \bar{\hat{y}} \right)^2} \end{aligned} \quad (1.57)$$

$\bar{y}$  et  $\hat{\bar{y}}$  représentent respectivement les moyennes des évaluations des réponses exactes et approchées aux  $n_v$  points échantillonnés.

$0 \leq r^2 \leq 1$  plus le coefficient est proche de 1, plus le métamodèle est précis.

### 3.2.3 $r_{ccc}$

Le critère de Bravais-Pearson traduit une dépendance **linéaire** entre deux variables. Il ne traduit pas les différences d'amplitude d'une réponse par rapport à une autre ou encore d'un offset. Il peut s'avérer nécessaire d'avoir un indicateur capable de tenir compte de ces différences. Le coefficient de concordance  $r_{ccc}$  tient compte de ces aspects [Lin, 1989, Nickerson, 1997]. Ce coefficient s'écrit de la manière suivante :

$$r_{ccc} = \frac{2 \cdot \sigma_{xy}}{\sigma_x^2 + \sigma_y^2 + (\bar{y} - \hat{\bar{y}})^2} \quad (1.58)$$

Ce coefficient est équivalent à certains coefficients d'interclass (Interclass correlation coefficients), le lecteur intéressé trouvera des précisions dans [Chen et Barnhart, 2008].

La figure 1.12 illustre la différence entre  $r^2$  et  $r_{ccc}$ .

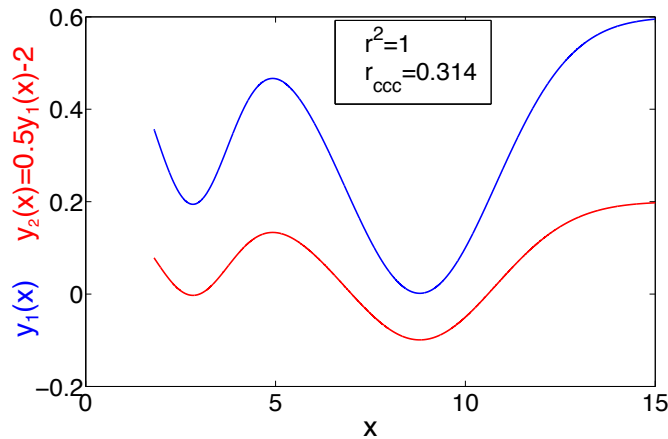


FIGURE 1.12: Comparaison des deux coefficients utilisés

$-1 \leq r_{ccc} \leq 1$ . Plus le coefficient est proche de 1, plus le métamodèle est précis.

### 3.2.4 Erreur relative moyenne absolue (RAAE)

$$\text{RAAE} = \frac{\sum_{i=1}^{n_v} |y(\mathbf{x}^{(i)}) - \hat{y}(\mathbf{x}^{(i)})|}{n_v \sigma} \quad (1.59)$$

avec  $\sigma$  l'écart type des données  $y(\mathbf{x}^{(i)})$

Plus ce critère est faible plus le métamodèle est de qualité.

### 3.2.5 Erreur relative maximale absolue (RMAE)

$$\text{RMAE} = \frac{\max |y(\mathbf{x}^{(i)}) - \hat{y}(\mathbf{x}^{(i)})|}{\sigma} \quad (1.60)$$

avec  $\sigma$  l'écart type des données  $y(\mathbf{x}^{(i)})$

Ce critère est intéressant car c'est un critère d'indication locale. On peut donc avoir des indicateurs comme  $r^2$  ou  $r_{ccc}$  ayant un excellent niveau tandis que celui-ci peut être mauvais. Il permet donc de localiser la zone où le métamodèle est le plus faux. Cependant, des critères globaux sont quand même à préférer pour évaluer la qualité d'un métamodèle. *Au sens de cette métrique, plus elle est faible, plus le métamodèle est bien sûr précis.*

### 3.2.6 Quelques erreurs normées

On ajoute 3 indicateurs aux précédents, ceux-ci ont pour particularité d'être normés. On définit une erreur en chaque point échantillonné par la relation suivante :

$$\forall i \in \llbracket 1, n_v \rrbracket e_i = \frac{\left( (y(\mathbf{x}^{(i)}) - \hat{y}(\mathbf{x}^{(i)}))^2 \right)}{\sup_{j \in \llbracket 1, n_v \rrbracket} y(\mathbf{x}^{(j)})^2} \quad (1.61)$$

3 indicateurs sont alors définis :

$$Q_1 = \sup e_i \quad Q_2 = \sum_{i=1}^{n_v} e_i \quad Q_3 = \frac{Q_2}{n_v} \quad (1.62)$$

*Plus ces indicateurs sont faibles, plus le métamodèle est de qualité.*

## 3.3 Optimisation et métamodèle : Stratégie EGO

L'optimisation par métamodèle est peu coûteuse à réaliser par rapport au coût de calcul que nécessite l'évaluation d'un jeu de paramètres. Cependant, cette optimisation peut conduire à converger vers un minimum local existant sur le métamodèle mais inexistant sur la fonction mécanique réelle. Ce phénomène peut être dû à :

- Un échantillonnage insuffisant ou non pertinent amenant à construire un métamodèle n'approximant pas correctement la zone du minimum
- Le fait que la fonction mécanique réelle est un minimum se trouvant dans une « vallée » (une zone où le gradient de la fonction est très faible)

Afin de trouver l'optimum de la fonction mécanique traitée, il est nécessaire que le métamodèle ait un certain niveau de qualité afin de trouver un optimum proche de l'optimum réel. Une phase d'enrichissement de l'échantillonnage de l'espace  $\mathcal{D}$  est donc souvent nécessaire.

Différentes méthodes d'enrichissement peuvent exister. L'idée de base est d'ajouter un nouveau jeu de paramètres parmi les échantillons déjà connus. Dans le cadre de l'optimisation, différentes stratégies peuvent être mises en place. [Jones, 2001] rappelle que deux grandes catégories se distinguent :

- Les méthodes à deux étapes pour lesquelles un métamodèle est construit à partir de l'échantillonnage initial. Puis s'appuyant sur ce métamodèle et ses propriétés (variance par exemple dans le cadre du krigeage), un critère d'enrichissement est calculé. Un nouveau point est alors ajouté à l'échantillonnage initial par optimisation du dit critère.
- Les méthodes à une étape où le métamodèle n'est pas directement calculé mais où l'on dispose de toutes les informations pour éventuellement le faire. Le but de ces approches est d'enrichir au point maximisant une mesure de crédibilité visant à atteindre une certaine valeur de la fonction objectif. Cette mesure de crédibilité s'appuie sur une fonction de vraisemblance.

Les approches à une étape ne seront pas étudiées dans ce travail. Dans le cadre des approches à deux étapes, on peut rencontrer différentes philosophies d'enrichissement. Les premières sont les méthodes visant à mener l'enrichissement où potentiellement le minimum se situe ([Jones *et al.*, 1998]). Les secondes consistent à enrichir l'échantillonnage afin de rendre plus précis le métamodèle sur l'ensemble de l'espace de conception, ceci, dans le but de localiser par la suite l'optimum global.

Dans le cadre du krigeage, on pourra notamment chercher à enrichir le métamodèle au point maximisant la variance de l'estimateur ([Martin et Simpson, 2002]). D'autres critères d'enrichissement sont tout à fait envisageables comme dans [Sacks *et al.*, 1989] s'appuyant sur l'IMSE (*Integrated Mean Square Error*) ou encore des critères basés sur la corrélation croisée [Laurenceau et Sagaut, 2008]. Dans le prochain paragraphe, on présente une approche mise en place par [Jones *et al.*, 1998] consistant à venir enrichir l'échantillonnage de l'espace en un point qui potentiellement permet de localiser le minimum global. Cette approche appelée *Efficient Global Optimization* vise donc à enrichir l'échantillonnage dans des zones où potentiellement se trouve le minimum global.

### Stratégie EGO

La stratégie EGO (*Efficient Global Optimization*) est une stratégie d'optimisation par métamodèle proposée par [Jones *et al.*, 1998]. Elle consiste notamment à enrichir l'échantillonnage au point du métamodèle vérifiant le maximum d'*expected improvement* (EI). Ce critère a été introduit par [Schonlau *et al.*, 1997].

Pour établir ce critère, on suppose qu'en un point  $\mathbf{x}$ , la valeur de la fonction objectif est la réalisation d'une variable aléatoire  $Y_e(\mathbf{x})$  normalement distribuée, de moyenne égale à  $\hat{y}(\mathbf{x})$  et de variance  $s_y^2(\mathbf{x})$  (qui sont respectivement la valeur du métamodèle et sa variance). On pose  $y_{min} = \min_{i \in \llbracket 1, n_s \rrbracket} \{y(\mathbf{x}^{(i)})\}$ . On pose  $I$  l'amélioration définie par :

$$I = \begin{cases} y_{min} - Y_e(\mathbf{x}) & \text{si } Y_e(\mathbf{x}) < y_{min} \\ 0 & \text{sinon} \end{cases} \quad (1.63)$$

On peut alors calculer la probabilité d'amélioration par :

$$PI(\mathbf{x}) = \int_{I=0}^{I=\infty} \frac{1}{\sqrt{2\pi}s_{\hat{y}}(\mathbf{x})} e^{-(y_{min}-I-\hat{y}(\mathbf{x}))/2s_{\hat{y}}^2(\mathbf{x})} dI = \Phi\left(\frac{y_{min}-\hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})}\right) \quad (1.64)$$

avec  $\Phi$  la fonction de répartition de loi normale centrée réduite.

Pour calculer l'EI, il faut calculer la quantité d'amélioration suivant la loi de probabilité, c'est-à-dire :

$$EI(\mathbf{x}) = \int_{I=0}^{I=\infty} I \frac{1}{\sqrt{2\pi}s_{\hat{y}}(\mathbf{x})} e^{-(y_{min}-I-\hat{y}(\mathbf{x}))/2s_{\hat{y}}^2(\mathbf{x})} dI \quad (1.65)$$

Ce qui en développant les calculs amène à :

$$EI(\mathbf{x}) = (y_{min} - \hat{y}(\mathbf{x})) \Phi\left(\frac{y_{min} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})}\right) + s_{\hat{y}}(\mathbf{x}) \phi\left(\frac{y_{min} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})}\right) \quad (1.66)$$

avec  $\phi$  la fonction de densité de probabilité de la loi normale centrée réduite.

La stratégie EGO consiste donc à : (1) Construire un métamodèle. (2) Rechercher le point maximisant EI. (3) Enrichir l'échantillonnage en ce point. Cette stratégie a montré son efficacité pour la localisation du minimum global d'une fonction objectif [Jones *et al.*, 1998].

Le critère d'expected improvement est composé de deux termes. Le premier favorise l'exploitation des données connues, c'est-à-dire que le minimum de la fonction objectif est recherché dans la zone où se situe le minimum connu par le métamodèle. Le second terme favorise l'exploration, c'est-à-dire les zones où il y a peu d'informations. Sur la figure 1.13(a), on constate que l'enrichissement se fait près du minimum connu. Lors de la deuxième itération (figure 1.13(b)), le maximum d'*expected improvement* est recherché dans une zone peu explorée qui s'avère être la zone du minimum global recherché. Sur les figures 1.13, en bleu est tracée la fonction objectif, en noir un métamodèle de krigeage, et en vert la fonction EI.

### 3.4 Bilan

Il a été vu dans les paragraphes précédents différentes stratégies de construction de métamodèles. Une d'entre elles a été plus largement développée : la méthode s'appuyant sur le krigeage. Celle-ci est d'ailleurs l'une des plus répandues dans la littérature. Elle permet de fournir des métamodèles de qualité. Cependant, il est nécessaire de trouver des moyens d'enrichir le tirage initial de points afin d'affiner la qualité du métamodèle ou de chercher à localiser la zone du minimum global. La stratégie d'enrichissement par EI est actuellement largement employée pour mener à bien une optimisation de fonction objectif par métamodèle. **Cette technique d'enrichissement sera donc utilisée dans le but de rechercher le minimum global. Afin de créer des métamodèles de qualité (création d'abaque virtuel), l'enrichissement sera mené en ajoutant à l'échantillonnage le point maximisant la variance du krigeage ([Martin et Simpson, 2002, Rosenbaum et Schulz, 2012]).** Ces stratégies seront développées au cours des prochains chapitres.



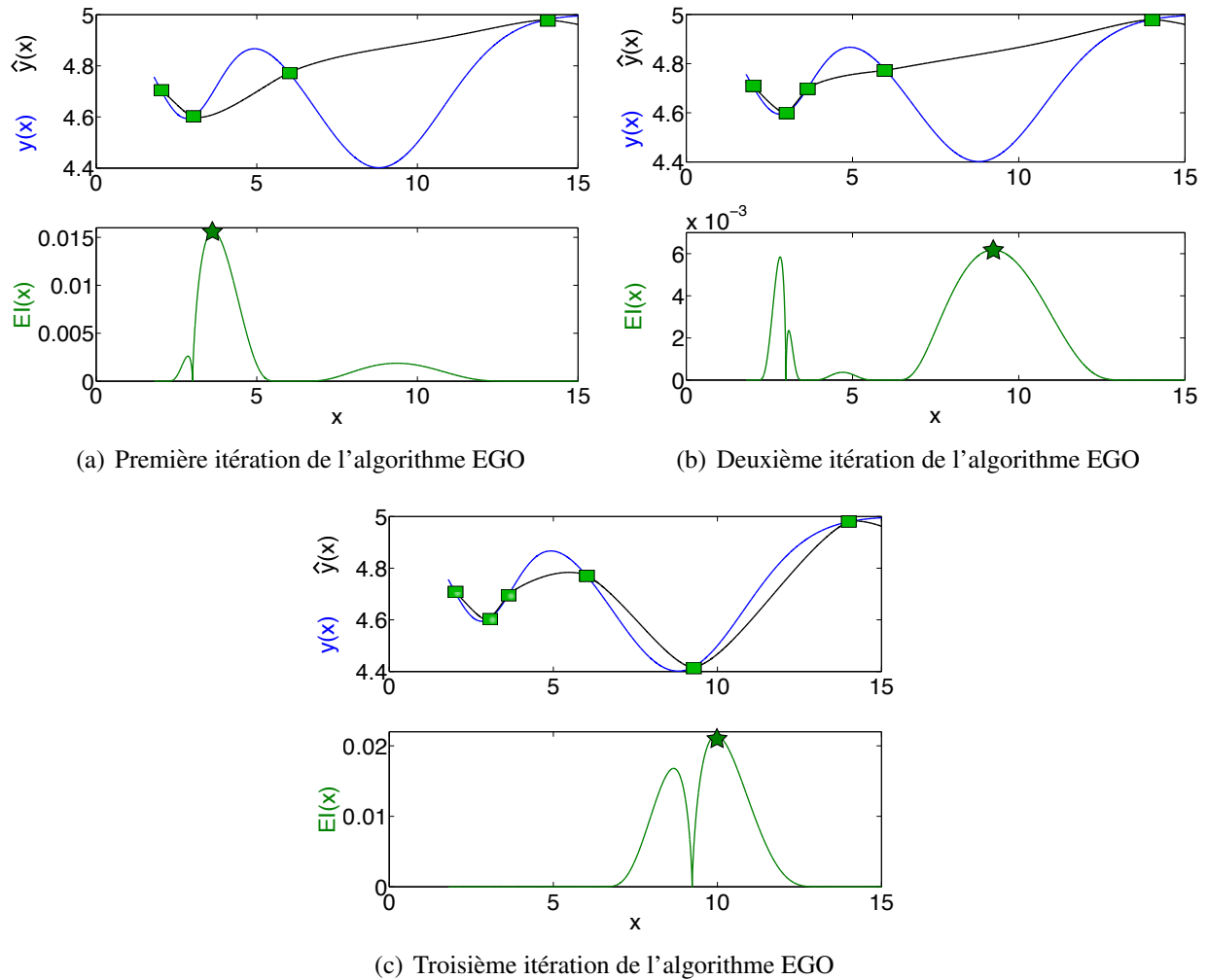


FIGURE 1.13: Différentes itérations de l'algorithme EGO, enrichissement par EI

## 4 La méthode LaTin

Comme vu précédemment, un élément incontournable dans le processus d'optimisation est la définition du **modèle** mécanique utilisé pour simuler la structure. Dans le cadre des assemblages, on se heurte à la fois à des difficultés méthodologiques et numériques. Le modèle mécanique doit tenir compte de la présence de différentes pièces (pouvant être présentes en grand nombre) et de leurs caractéristiques matériaux, des lois de comportement de chacune d'entre elles et du comportement aux interfaces. Dans notre cas d'étude, le comportement des structures se limitera au **comportement linéaire élastique isotherme sous l'hypothèse des petites perturbations**. Concernant les interfaces, on sera particulièrement attentif à la prise en compte des frottements et des contacts avec ou sans jeux. La modélisation du frottement est un sujet d'étude en soi. En effet, celui-ci n'est pas nécessairement uniforme sur une même interface. Pour avoir un comportement

proche du réel, des études topologiques sont nécessaires. Le modèle utilisé par la suite sera un modèle de frottement de Coulomb, modèle couramment utilisé pour ce genre de comportement.

Le simulateur numérique, employé au cours de cette thèse est COFAST3D (*COntact et Frottement dans les Assemblages de STructure*). Ce solveur s'appuie sur la méthode LaTIn mono-échelle (*Large Time Increment* [Ladevèze, 1999]). Il est donc spécifiquement dédié aux calculs d'assemblages. Cependant, la méthode LaTIn ne se limite pas qu'à ce genre d'études. Elle couvre un large éventail de problèmes non-linéaires, que ce soit au niveau de la loi de comportement utilisée ou de non-linéarités géométriques. [Laurent, 2013] fait un rappel complet des différents cas d'applications développées avec la méthode LaTIn pouvant être couplés avec des réductions de modèle tels que la PGD. On pourra retenir :

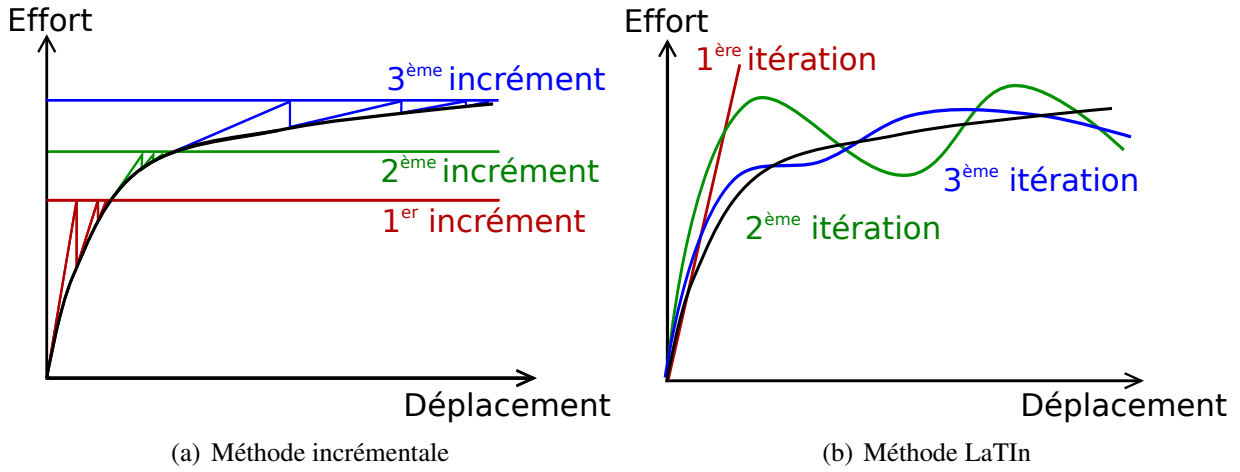
- Grandes déformations/transformations [Vauchez, 1991, Boucard, 1996]
- Elasto-Visco-Plasticité [Relun, 2011, Cognard, 1989]
- Endommagement [Allix, 1992]
- Assemblage de composites et comportement des composites (flambage, délaminage) [Roulet, 2011, Saavedra Redlich, 2012]

Le logiciel COFAST a été développé par Laurent Champaney durant sa thèse [Champaney, 1996]. C'est un logiciel s'appuyant sur le langage GIBIANE et ESOPE, ayant pour base le logiciel CASTEM développé par le CEA. Celui-ci a montré sa robustesse pour la résolution de ce genre de problèmes [Champaney, 1996, Boucard et Champaney, 2003, Laurent *et al.*, 2013a]. On s'attachera dans cette section à rappeler les grands principes de la méthode LaTIn dans le cadre des calculs d'assemblages.

Cette méthode a été introduite au milieu des années 1980 par P. Ladevèze. Un des principaux intérêts de la méthode est que contrairement aux méthodes incrémentales classiquement employées pour résoudre ce type de problèmes, la solution est obtenue de manière globale en temps et en espace. Celle-ci s'appuie sur un algorithme itératif qui, à chaque itération, fournit une approximation complète de la solution (voir représentation figure 1.14). La solution s'enrichit au fur et à mesure des itérations jusqu'à atteindre un critère d'arrêt. Celui-ci peut être un critère de temps (on stoppe l'algorithme une fois un certain temps écoulé) ou lorsqu'un certain niveau d'indicateur d'erreur en énergie (voir paragraphe 4.2.6) a été atteint. Il faut cependant rester conscient que si l'algorithme est stoppé prématurément, la solution est complète en temps et en espace mais ne reste qu'une solution **approchée** du problème mécanique. **Cette propriété est le point central des travaux développés dans cette thèse. L'idée est d'obtenir des évaluations moins précises mais peu coûteuses que l'on utilisera pour construire un métamodèle multi-fidélité.**

La méthode s'appuie sur 3 principes :

- **P1 : Séparation des difficultés.** Les différentes équations du problème sont séparées en deux groupes. Le groupe  $\mathcal{A}_d$  est le groupe des équations linéaires (par exemple dans le cadre des assemblages élastiques, ce groupe d'équations rassemble l'ensemble des équations d'équilibre des sous-structures). Le groupe  $\Gamma$  rassemble l'ensemble des équations non linéaires comme par exemple les équations de contact aux interfaces. La solution du problème est donc à l'intersection de ces deux espaces.



**FIGURE 1.14:** Représentation schématique de l'approche des méthodes incrémentales et de la méthode LaTIn

- **P2 : Résolution.** L'algorithme de résolution LaTIn est itératif contrairement à la plupart des algorithmes permettant de résoudre des problèmes non-linéaires en mécanique du solide qui sont incrémentaux.
- **P3 : Représentation des inconnues.** Choix d'une représentation des inconnues du problème (permettant une formulation PGD par exemple).

## 4.1 Présentation du problème mécanique

Comme dit précédemment, dans notre cas d'étude on se limite à des structures ayant un **comportement linéaire élastique isotherme sous l'hypothèse des petites perturbations**. Le chargement est supposé quasi-statique dans l'intervalle de temps  $[0, T]$ . Les **non-linéarités** considérées dans la suite du travail se situent alors au niveau des interfaces entre les pièces.

### 4.1.1 Problème mécanique de référence

Le problème considéré est défini par un domaine  $\Omega$  soumis à des forces volumiques  $\mathbf{f}_d$  (gravitationnelles, électromagnétiques, etc.), des forces surfaciques  $\mathbf{F}_d$  sur une zone  $\partial_2\Omega$  de la frontière du domaine et à des déplacements imposés  $\mathbf{u}_d$  sur une zone  $\partial_1\Omega$ .

Soit  $M$  un point de  $\Omega$ , un champ de déplacement  $\mathbf{u} = \mathbf{u}(M, t)$  est dit **cinématiquement admissible** (CA) s'il est *suffisamment régulier* et s'il vérifie les conditions aux limites en déplacement imposés  $\mathbf{u}_d$  sur le bord  $\partial_1\Omega$ . L'espace vectoriel  $\mathcal{U}_{ad}^{[0, T]}$  des champs de déplacements cinématiquement admissibles est donc défini tel que :

$$\mathcal{U}_{ad}^{[0, T]} = \{\mathbf{u}(M, t) \mid \mathbf{u} \text{ est suffisamment régulier, } \forall t \in [0, T], \forall M \in \partial_1\Omega, \mathbf{u}(M, t) = \mathbf{u}_d\} \quad (1.67)$$

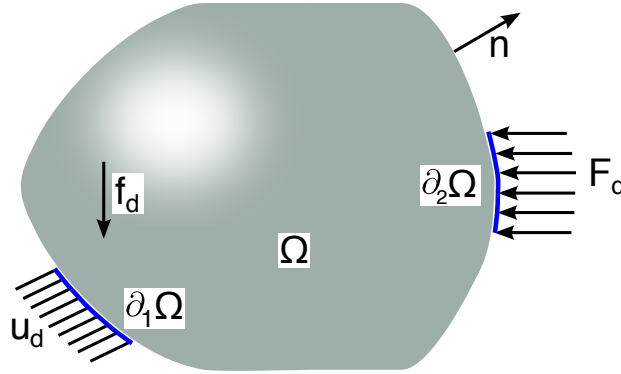


FIGURE 1.15: Problème de référence

Soit  $M$  un point de  $\Omega$ , un champ contraintes  $\boldsymbol{\sigma} = \boldsymbol{\sigma}(M, t)$  est dit **statiquement admissible** (SA) s'il est *suffisamment régulier, symétrique* et s'il vérifie les équations d'équilibre. L'espace vectoriel  $\mathcal{S}_{ad}^{[0, T]}$  des champs de contraintes statiquement admissibles est défini tel que :

$$\begin{aligned} \mathcal{S}_{ad}^{[0, T]} = \{ & \boldsymbol{\sigma}(M, t) \mid \boldsymbol{\sigma} \text{ est suffisamment régulier, symétrique} \\ & \text{et vérifie les équations d'équilibre :} \\ & \forall t \in [0, T], \forall M \in \Omega, \operatorname{div} \boldsymbol{\sigma}(M, t) + \mathbf{f}_d = 0 \\ & \forall t \in [0, T], \forall M \in \partial_2 \Omega, \boldsymbol{\sigma}(M, t) \cdot \mathbf{n} = \mathbf{F}_d \} \end{aligned} \quad (1.68)$$

Cette formulation de l'espace  $\mathcal{S}_{ad}^{[0, T]}$  est appelée **formulation forte** ou **formulation locale**. On peut également l'exprimer sous une formulation faible équivalente à la précédente.

Le problème de référence peut donc s'écrire sous la forme :

#### Problème de référence :

Trouver  $(\mathbf{u}(M, t), \boldsymbol{\sigma}(M, t))$  définis sur  $\Omega \times [0, T]$  tels que :

- $\mathbf{u}(M, t) \in \mathcal{U}_{ad}^{[0, T]}$
- $\boldsymbol{\sigma}(M, t) \in \mathcal{S}_{ad}^{[0, T]}$
- $\boldsymbol{\sigma}(M, t) = \mathbf{K} \boldsymbol{\varepsilon}(\mathbf{u}(M, t))$  avec  $\mathbf{K}$  opérateur de Hooke et  $\boldsymbol{\varepsilon}(\mathbf{u}(M, t))$  champ de déformation associé au champ de déplacement  $\mathbf{u}(M, t)$ . Cette dernière relation est appelée loi en relation de comportement.

#### 4.1.2 Problème mécanique pour les assemblages

Dans le cadre des problèmes d'assemblages, le problème mécanique précédent peut être appliqué à chaque pièce (il n'y a donc plus un seul domaine  $\Omega$  mais plusieurs). Il est cependant nécessaire de modifier les espaces vectoriels  $\mathcal{U}_{ad}^{[0, T]}$  et  $\mathcal{S}_{ad}^{[0, T]}$  en y ajoutant les équations de compatibilité aux interfaces. Ces équations traduisent le comportement d'une pièce vis-à-vis d'une

autre. Comme sur la figure 1.16, on note une pièce (ou un élément de l'assemblage) par le domaine  $\Omega_E$  et l'autre par  $\Omega_{E'}$ , leur interface est notée  $\Gamma^{EE'}$ . Certaines propriétés mécaniques doivent donc être vérifiées au niveau de l'interface entre les pièces. Celles-ci seront développées dans les prochains paragraphes.

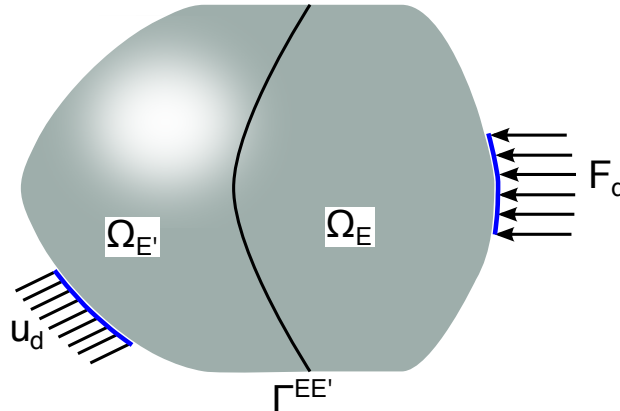


FIGURE 1.16: Problème de référence d'un assemblage

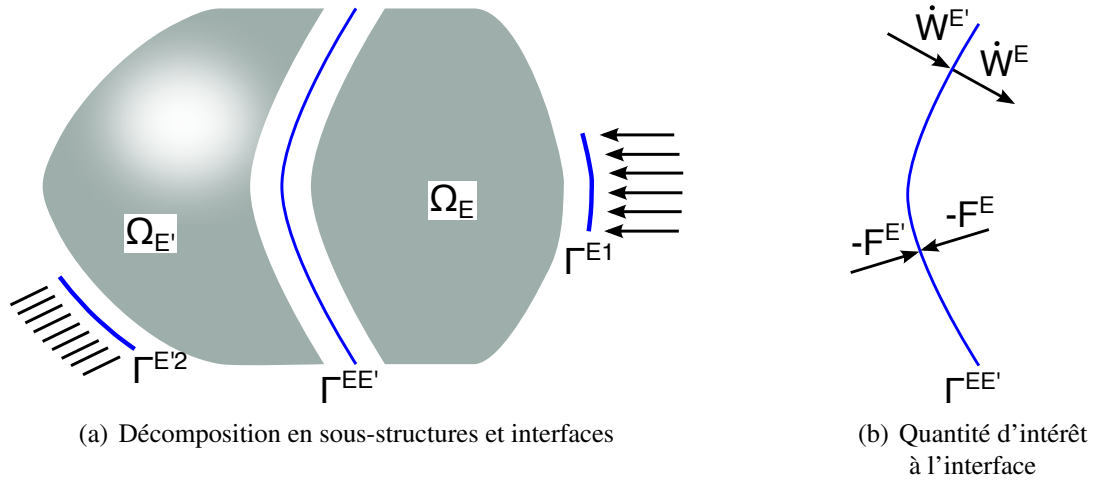
## 4.2 Résolution par la méthode LaTin

### 4.2.1 Présentation générale de la méthode

Afin de résoudre ces problèmes d'assemblages, la méthode LaTin est employée. Cette méthode s'appuie sur une décomposition de domaines mixtes. C'est-à-dire qu'elle cherche à raccorder les efforts et déplacements aux interfaces ([Glowinski et Le Tallec, 1990, Farhat *et al.*, 2000, Series *et al.*, 2003]). On peut également citer comme autres approches, les méthodes primales (cherchant à assurer la continuité des déplacements aux interfaces) et les méthodes duales (cherchant à assurer l'équilibre des efforts aux interfaces). Une taxonomie de ces approches est disponible dans [Gosselet et Rey, 2006]. Les méthodes de décomposition de domaines amènent à définir des **sous-domaines** (ou **sous-ensembles**) et des **interfaces**. Dans le cadre des assemblages, cette décomposition est assez simple à mettre en place, puisque chaque sous-domaine peut être considéré comme une pièce de l'assemblage et donc chaque zone de contact comme une interface.

La méthode LaTin a également pour particularité de considérer l'interface entre deux sous-domaines comme une réelle entité à laquelle on adjoint un comportement particulier. On définit également des interfaces pour prendre en compte les conditions limites entre le sous-domaine et l'extérieur (l'extérieur étant considéré comme une entité mécanique équivalente à un sous-domaine).

Comme décrit sur la figure 1.17, on considère au niveau des interfaces, les champs d'effort  $\mathbf{F}^E$  et  $\mathbf{F}^{E'}$  correspondant aux efforts exercés par l'interface sur les deux sous-domaines. Les variables duales sont les déplacements  $\mathbf{W}^E$  et  $\mathbf{W}^{E'}$  (dans le cas statique) ou les vitesses  $\dot{\mathbf{W}}^E$  et  $\dot{\mathbf{W}}^{E'}$  (dans le cas quasi-statique).



**FIGURE 1.17:** Décomposition de domaine de la méthode LaTIn et grandeurs prises en compte au niveau des interfaces

On précise que dans le cadre quasi-statique, le problème est traité dans l'algorithme de façon incrémentale, ce qui a pour effet de considérer des sauts de déplacements.

Dans le cadre des assemblages étudiés (linéaire élastique isotherme avec hypothèse des petites perturbations), le problème est donc décrit intégralement par le comportement aux interfaces. La solution du problème sous-structuré peut donc se réduire à la solution aux interfaces. En effet, par rapport aux problèmes de référence définis précédemment, les espaces vectoriels  $\mathcal{U}_{ad}^{[0,T]}$  et  $\mathcal{S}_{ad}^{[0,T]}$  doivent tenir compte des déplacements  $\mathbf{W}^E$  et des efforts  $\mathbf{F}^E$  d'interface s'exerçant sur la frontière de chaque sous-structure. Il s'agit donc seulement de prendre en considération des conditions limites dues aux interfaces pour chaque sous-domaine  $\Omega_E$ . Les espaces vectoriels associés à chaque sous-domaine  $\Omega_E$  seront notés  $\mathcal{U}_{E,ad}^{[0,T]}$  et  $\mathcal{S}_{E,ad}^{[0,T]}$ .

L'admissibilité cinématique est définie de la manière suivante :

$$\mathcal{U}_{E,ad}^{[0,T]} = \left\{ \mathbf{u}^E(M,t) \mid \mathbf{u}^E \text{ est suffisamment régulier, } \forall t \in [0, T], \forall M \in \partial\Omega_E, \right. \\ \left. \mathbf{u}^E(M,t) = \mathbf{W}^E(M,t) \right\} \quad (1.69)$$

L'admissibilité statique est définie afin d'assurer l'équilibre du sous-domaine  $\Omega_E$  :

$$\mathcal{S}_{E,ad}^{[0,T]} = \left\{ \boldsymbol{\sigma}^E(M,t) \mid \boldsymbol{\sigma}^E \text{ est suffisamment régulier, symétrique} \right. \\ \left. \text{et vérifie les équations d'équilibre :} \right. \\ \left. \forall t \in [0, T], \forall M \in \Omega_E, \operatorname{div} \boldsymbol{\sigma}^E(M,t) + \mathbf{f}_d = 0 \right. \\ \left. \forall t \in [0, T], \forall M \in \partial\Omega_E, \boldsymbol{\sigma}^E(M,t) \cdot \mathbf{n} = -\mathbf{F}^E \right\} \quad (1.70)$$

Le premier point (point **P1**) de l'approche LaTIn consiste à créer deux groupes d'équations. Deux sous-espaces sont alors créés  $\mathcal{A}_d$  et  $\Gamma$ . L'espace  $\mathcal{A}_d$  contient les solutions aux équations

**linéaires** (éventuellement globales), dans notre contexte elles correspondent aux équations associées aux sous-structures. L'espace  $\Gamma$  contient quant à lui les solutions aux équations **locales** en espace (éventuellement **non-linéaires**). Ces dernières équations sont, dans notre cas d'étude, liées aux interfaces (contact avec ou sans jeux avec ou sans frottement). La solution complète du problème se situe donc à l'intersection de ces deux espaces.

#### 4.2.2 Problème dans les sous-domaines

Pour rappel, on considère un comportement linéaire des matériaux sous l'hypothèse des petites perturbations. Le problème dans un sous-domaine  $\Omega_E$  consiste donc à trouver les solutions  $(\mathbf{u}^E(M, t), \boldsymbol{\sigma}^E(M, t) \in \Omega_E \times [0, T])$  associées à l'espace  $\mathcal{A}_d$  telles que :

- Résolution des équations cinématiques :  
 $\mathbf{u}^E(M, t) \in \mathcal{U}_{E, ad}^{[0, T]}$
- Résolution des équations statiques :  
 $\boldsymbol{\sigma}^E(M, t) \in \mathcal{S}_{E, ad}^{[0, T]}$
- Vérification de la loi en relation de comportement  
 $\boldsymbol{\sigma}^E(M, t) = \mathbf{K}^E \boldsymbol{\varepsilon}(\mathbf{u}^E(M, t))$  avec  $\mathbf{K}^E$  opérateur de Hooke.

#### 4.2.3 Problème aux interfaces

Afin de clarifier le propos, on distingue les grandeurs intervenant lors de la résolution des équations locales et de celles intervenant lors de la résolution des équations linéaires. Dans notre cas, cela revient à distinguer les grandeurs d'interface notées  $\widehat{\mathbf{W}}^E$ ,  $\widehat{\dot{\mathbf{W}}}^E$  et  $\widehat{\mathbf{F}}^E$  des grandeurs se situant aux bords du domaine  $\partial\Omega_E$  ( $\mathbf{W}^E$ ,  $\dot{\mathbf{W}}^E$  et  $\mathbf{F}^E$ ). Bien entendu, quand la solution du problème est trouvée, ces grandeurs sont confondues.

Le problème à une interface  $\Gamma^{EE'}$  revient à déterminer les grandeurs locales  $\widehat{\dot{\mathbf{W}}}^E(M, t)$  (ou  $\widehat{\mathbf{W}}^E(M, t)$ , dans le cas statique) et  $\widehat{\mathbf{F}}^E(M, t)$  avec  $M \in \Gamma^{EE'}$  et  $\forall t \in [0, T]$  solutions du sous-espace des équations associées à  $\Gamma$ . Le comportement de l'interface est défini par la relation  $\mathcal{R}$  telle que :

$$\forall M \in \Gamma^{EE'}, \forall t \in [0, T] \quad \mathcal{R} \left( \widehat{\dot{\mathbf{W}}}^E, \widehat{\dot{\mathbf{W}}}^{E'}, \widehat{\mathbf{F}}^E, \widehat{\mathbf{F}}^{E'} \right) = 0 \quad (1.71)$$

Il est donc possible de considérer plusieurs types de comportement. On rappelle ici certains des comportements possibles et qui seront les principaux utilisés dans ce travail. On pourra se référer à [Blanzé *et al.*, 1996] pour avoir des précisions supplémentaires.

##### Notations :

- $\forall M \in \Gamma^{EE'}$ ,  $\mathbf{N} = \mathbf{N}(M)$  est la normale en tout point de l'interface  $\Gamma^{EE'}$ .
- $\forall M \in \Gamma^{EE'}$ ,  $\Pi = \Pi(M)$  est l'opérateur de projection dans le plan tangent en tout point de l'interface  $\Gamma^{EE'}$ . Il est défini ainsi :  $\Pi \mathbf{v} = \mathbf{v} - \mathbf{N}^T \mathbf{v} \mathbf{N}$  avec  $\mathbf{v} = \mathbf{v}(M)$  un vecteur.

- $\forall M \in \Gamma^{EE'}$  et  $\forall t \in [0, T]$ ,  $\llbracket \widehat{\mathbf{W}} \rrbracket_{EE'} = \dot{\widehat{\mathbf{W}}}^E - \dot{\widehat{\mathbf{W}}}^{E'}$  et  $\llbracket \widehat{\mathbf{W}} \rrbracket_{EE'} = \widehat{\mathbf{W}}^E - \widehat{\mathbf{W}}^{E'}$  représentent les sauts de vitesse et de déplacements à l'interface  $\Gamma^{EE'}$ .

Les différentes interfaces traitées sont de trois types : parfaites, avec contact unilatéral et/ou avec/sans frottement.

- **Interface parfaite** : on considère une interface parfaite lorsque l'on veut que les sous-domaines soient parfaitement connectés. Ainsi, on doit assurer l'équilibre de l'interface et avoir un saut de déplacement nul :

$$\begin{cases} \widehat{\mathbf{F}}^E + \widehat{\mathbf{F}}^{E'} = 0 \\ \llbracket \widehat{\mathbf{W}} \rrbracket_{EE'} = 0 \end{cases} \quad (1.72)$$

- **Interface avec contact** : Deux cas sont à étudier pour ce comportement d'interface, à savoir le décollement ou le contact de l'interface. On considère ici un jeu initial noté  $j$ . Pour connaître l'état de l'interface, les équations suivantes sont à considérer :

- **Décollement** : si  $\mathbf{N}^T \llbracket \widehat{\mathbf{W}} \rrbracket_{EE'} > j$  alors  $\widehat{\mathbf{F}}^E = \widehat{\mathbf{F}}^{E'} = 0$

- **Contact** : si  $\mathbf{N}^T \llbracket \widehat{\mathbf{W}} \rrbracket_{EE'} = j$  alors  $\mathbf{N}^T \widehat{\mathbf{F}}^E + \mathbf{N}^T \widehat{\mathbf{F}}^{E'} = 0$ , de plus  $\Pi \widehat{\mathbf{F}}^E = -\Pi \widehat{\mathbf{F}}^{E'}$ . On peut remarquer que  $\Pi \widehat{\mathbf{F}}^E = 0$  s'il n'y a pas de frottement.

- **Interface avec frottement** : Deux cas sont à nouveau à considérer dans cette situation. Il s'agit de l'adhérence et du glissement. Comme précédemment, un jeu d'équations permet de définir la situation dans laquelle on se situe. La loi de frottement utilisée est celle du Coulomb.

- **Adhérence** : si  $\|\Pi \widehat{\mathbf{F}}^E\| < \mu |\mathbf{N}^T \widehat{\mathbf{F}}^E|$  alors  $\Pi \llbracket \widehat{\mathbf{W}} \rrbracket_{EE'} = 0$

- **Glissement** : si  $\|\Pi \widehat{\mathbf{F}}^E\| = \mu |\mathbf{N}^T \widehat{\mathbf{F}}^E|$  alors  $\exists \lambda > 0$ ,  $\llbracket \widehat{\mathbf{W}} \rrbracket_{EE'} = -\lambda \widehat{\mathbf{F}}^E$

Ces trois types d'interface suffisent à décrire les principaux comportement étudiés dans les assemblages (tout du moins de manière macroscopique). Pour simuler le collage, on pourra également définir une interface de type élastique.

#### 4.2.4 Algorithme LaTIn

Le second point **P2** consiste à venir résoudre successivement l'ensemble des équations associées à l'espace  $\mathcal{A}_d$  et  $\Gamma$  de manière itérative. Dans le cadre des assemblages utilisés dans ce travail, cela est équivalent à résoudre de manière successive les équations associées aux sous-domaines et aux interfaces. Afin de mener ces itérations, deux nouveaux espaces, appelés directions de recherche, et notés  $E^+$  et  $E^-$  relient les espaces  $\mathcal{A}_d$  et  $\Gamma$ .

On définit  $\mathbf{s}_n$  comme étant une solution de l'espace  $\mathcal{A}_d$  et  $\widehat{\mathbf{s}}_n$  comme une solution de l'espace  $\Gamma$ . La figure 1.18 illustre le principe de l'algorithme itératif.

Une itération de l'algorithme correspond alors au passage de la solution  $\mathbf{s}_n$  à la solution  $\mathbf{s}_{n+1}$ . Cette itération correspond à la résolution d'un problème associé aux sous-domaines et d'un autre associé aux interfaces. Ces deux résolutions sont appelées dans la méthode LaTIn sous les termes **étape locale** et **étape linéaire** (ou globale).



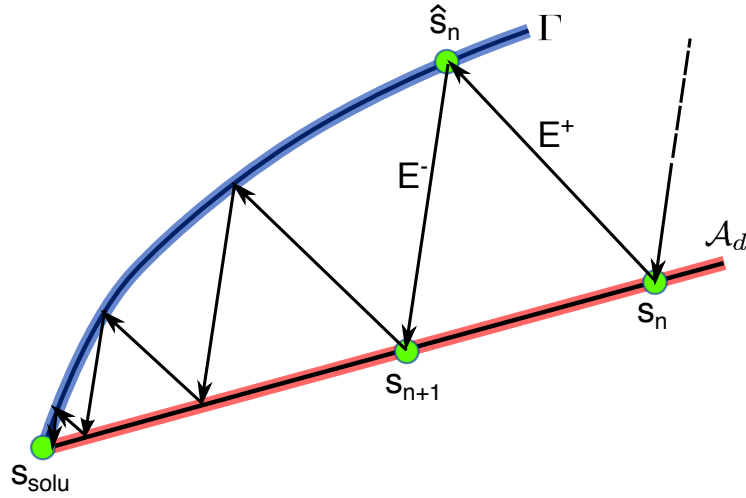


FIGURE 1.18: Illustration de l'algorithme itératif LaTin

- **Etape locale** : On cherche une solution  $\hat{\mathbf{s}}_n$  connaissant  $\mathbf{s}_n$  et vérifiant :  $\hat{\mathbf{s}}_n \in \Gamma$  et  $\hat{\mathbf{s}}_n - \mathbf{s}_n \in E^+$
- **Etape linéaire** : On cherche une solution  $\mathbf{s}_{n+1}$  connaissant  $\hat{\mathbf{s}}_n$  et vérifiant :  $\mathbf{s}_{n+1} \in \mathcal{A}_d$  et  $\mathbf{s}_{n+1} - \hat{\mathbf{s}}_n \in E^-$

La solution du problème est notée  $\mathbf{s}_{solu}$  et se situe à l'intersection des espace  $\Gamma$  et  $\mathcal{A}_d$ .

### Directions de recherche

Les directions de recherche sont fondamentales pour la vitesse de convergence de l'algorithme. Dans notre cadre d'étude lors de l'étape locale, seules les grandeurs d'interface doivent être mise à jour.

- Lors de l'étape locale on définit la direction de recherche comme :

$$\hat{\mathbf{s}}_n - \mathbf{s}_n \in E^+ \iff \hat{\mathbf{F}}_n^E - \mathbf{F}_n^E = \mathbf{k}^+ \left( \hat{\mathbf{W}}_n^E - \mathbf{W}_n^E \right) \quad (1.73)$$

L'étape linéaire met en jeu les quantités sur le bord du sous-domaine. On utilise donc une formulation conjuguée de la précédente.

- Lors de l'étape linéaire, la direction de recherche est donc définie comme :

$$\mathbf{s}_{n+1} - \hat{\mathbf{s}}_n \in E^- \iff \mathbf{F}_{n+1}^E - \hat{\mathbf{F}}_n^E = \mathbf{k}^- \left( \mathbf{W}_{n+1}^E - \hat{\mathbf{W}}_n^E \right) \quad (1.74)$$

$\mathbf{k}^+$  et  $\mathbf{k}^-$  sont deux opérateurs symétriques définis positifs qui sont des paramètres à définir par l'utilisateur. Classiquement, dans le calcul d'assemblages, on choisit des opérateurs simples tel que :  $\mathbf{k}^+ = \mathbf{k}^- = k_0 \mathbf{I}_d$  avec  $k_0$  un scalaire et  $\mathbf{I}_d$  la matrice identité  $[3 \times 3]$ . On peut comparer cet opérateur à un ressort reliant l'interface et le sous-domaine et représentatif de la raideur du système.

Celui-ci est donc le paramètre essentiel de la stratégie de la LaTIn mono-échelle. [Ladevèze, 1999] a démontré la convergence de l'algorithme pour les problèmes traités. Le choix du paramètre n'est donc influent que sur la vitesse de convergence de l'algorithme. Cependant, il est extrêmement important car, mal ajusté, l'algorithme peut mettre un temps très important pour converger. Dans le cadre des problèmes quasi-statiques, on choisit de manière empirique  $k_0$  tel que :

$$k_0 = \frac{En_T}{L_c} \quad (1.75)$$

$E$  étant le module d'Young,  $n_T$  le nombre de pas de temps (dans le cadre statique  $n_T = 1$ ) et  $L_c$  une longueur caractéristique des sous-domaines.

#### 4.2.5 Quelques précisions sur le solveur

Pour le lecteur intéressé, plus de précisions sont disponibles dans le manuscrit de thèse de L.Champaney [Champaney, 1996].

#### Résolution à l'étape locale

Les équations à résoudre à l'étape locale sont :

$$\left\{ \begin{array}{l} \hat{\mathbf{s}}_n \in \Gamma \iff \mathcal{R} \left( \hat{\mathbf{W}}_n^E, \hat{\mathbf{W}}_n^{E'}, \hat{\mathbf{F}}_n^E, \hat{\mathbf{F}}_n^{E'} \right) = 0 \\ \hat{\mathbf{s}}_n - \mathbf{s}_n \in E^+ \iff \begin{array}{l} \hat{\mathbf{F}}_n^E - \mathbf{F}_n^E = \mathbf{k}^+ \left( \hat{\mathbf{W}}_n^E - \mathbf{W}_n^E \right) \\ \hat{\mathbf{F}}_n^{E'} - \mathbf{F}_n^{E'} = \mathbf{k}^+ \left( \hat{\mathbf{W}}_n^{E'} - \mathbf{W}_n^{E'} \right) \end{array} \end{array} \right. \quad (1.76)$$

La principale difficulté à l'étape locale est la résolution des conditions de décollement, de contact, d'adhérence ou de glissement. Afin de faciliter la détermination de l'état de contact ou de décollement, la fonction  $C_N$  est définie comme :

$$C_N = \frac{1}{2} \mathbf{N}^T \llbracket \hat{\mathbf{W}} \rrbracket_{EE'} - \frac{j}{2} - \frac{1}{2k_0} \mathbf{N}^T (\hat{\mathbf{F}}^{E'}, \hat{\mathbf{F}}^E) \quad (1.77)$$

L'état de décollement ou de contact est alors définie par :

- **Décollement** : si  $C_N > 0$  ;
- **Contact** : si  $C_N \leq 0$

De même pour l'adhérence ou le glissement, on introduit la fonction  $Gl_N$  :

$$Gl_N = \frac{k_0}{2} \Pi \llbracket \hat{\mathbf{W}} \rrbracket_{EE'} - \frac{1}{2k_0} \mathbf{N}^T (\hat{\mathbf{F}}^{E'}, \hat{\mathbf{F}}^E) \quad (1.78)$$

L'état d'adhérence ou de glissement se traduit par :

- **Adhérence** : si  $Gl_N \leq \mu | \mathbf{N}^T \hat{\mathbf{F}}^E |$  ;
- **Glissement** : si  $Gl_N > | \mathbf{N}^T \hat{\mathbf{F}}^E |$

### Résolution à l'étape globale

La résolution à l'étape globale ne pose pas de problème particulier. Il est cependant à noter que les opérateurs de rigidité (matrices des sous-domaines et rigidité  $k_0$ ) sont constants à chaque itération. Cette étape reste néanmoins la plus coûteuse du calcul. Des stratégies PGD peuvent alors être appliquées lors de cette étape afin d'accélérer la résolution.

#### 4.2.6 Indicateur d'erreur globale

Puisque la méthode LaTin est itérative, un indicateur de convergence peut être défini. Celui-ci est une mesure énergétique de la distance entre  $\mathbf{s}_n$  et  $\hat{\mathbf{s}}_n$ . Dans le cas de l'élasticité, un indicateur simple d'erreur en énergie est construit à partir des champs surfaciques. Il est défini par l'équation 1.79

$$\eta = \frac{\sum \|s^E - \hat{s}^E\|^2}{\sum \|s^E\|^2 + \|\hat{s}^E\|^2} \text{ avec :} \quad (1.79)$$

$$\|s\|^2 = \int_{\partial\Omega^E} \mathbf{F}^T k_0 \mathbf{F} dS + \int_{\partial\Omega^E} \dot{\mathbf{W}}^T k_0 \dot{\mathbf{W}} dS$$

Cet indicateur tend vers 0 à convergence de l'algorithme. En effet, à convergence  $\mathbf{s}^E$  et  $\hat{\mathbf{s}}^E$  (c'est-à-dire  $(\mathbf{F}^E, \mathbf{W}^E$  et  $(\hat{\mathbf{F}}^E, \hat{\mathbf{W}}^E)$ ) sont confondus.

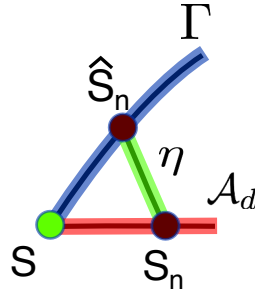


FIGURE 1.19: Illustration de l'indicateur global

**Remarque :** Dans le cadre d'utilisation de données partiellement convergées, **cet indicateur d'erreur sera utilisé pour définir les niveaux de convergence des données.** Une donnée dite totalement convergée correspondra à un niveau d'indicateur d'erreur  $\eta$  de l'algorithme LaTin de  $10^{-4}$  ou  $10^{-5}$  selon les cas d'études. **On confondra dans la suite du document indicateur d'erreur  $\eta$  et niveau de convergence des données.** Déterminer quel niveau de convergence associé aux données partiellement convergées en vue de construire un métamodèle efficace est un des enjeux de ce travail.

### 4.3 Approche multiparamétrique

Comme précisé précédemment, la stratégie de résolution LaTIn est différente des stratégies de résolution classiques. La stratégie employée et la décomposition en sous-espace permettent d’obtenir à chaque itération de l’algorithme itératif une approximation de la solution du problème sur l’ensemble du temps et de l’espace. On peut également remarquer que la solution complète du problème peut être réduite aux valeurs d’efforts et de déplacements calculés aux interfaces  $\mathbf{s}=(\mathbf{F}, \mathbf{W})$ .

Le principe de l’approche multiparamétrique est d’utiliser une solution précédemment calculée comme solution initiale d’un nouveau problème. Bien entendu les problèmes à résoudre sont de même nature. Elle a été développée afin de construire des surfaces de réponse [Boucard, 2001]. Elle permet de mener des analyses de sensibilités et des études paramétriques [Boucard et Champaney, 2003, Boucard et Champaney, 2004, Champaney *et al.*, 2008]. Elle a également permis de mener à bien une optimisation paramétrique [Buytet *et al.*, 2008, Boucard *et al.*, 2009, Soulier et Boucard, 2013, Laurent, 2013]. Elle autorise l’étude de prises en compte d’incertitudes dans la modélisation des assemblages [Blanzé *et al.*, 2004, Champaney *et al.*, 2005, Champaney et Boucard, 2007]. Plus récemment elle a été développée pour d’autres types de problèmes, à savoir : l’étude paramétrique de structures composites endommageables [Roulet, 2011] et de pièces élasto-viscoplastique [Relun, 2011]. Cette stratégie peut également être couplée avec une approche *Proper Generalized Decomposition* (PGD) [Relun, 2011], le calcul PGD ne s’effectuant que lors de l’étape linéaire. Dans le cadre de la mécanique élasto-viscoplastique, l’approche multiparamétrique a été combinée avec l’utilisation de métamodèles pour une étude de probabilité de défaillance et a montré des résultats convaincants [Echard *et al.*, 2013].

#### 4.3.1 L’approche multiparamétrique dans le cadre des assemblages

Dans nos problèmes d’assemblages, les paramètres à optimiser sont du type **précharge** de boulons ou de vis (modélisée par un jeu négatif entre la tête et le corps de vis), **jeu** et **coefficient de frottement**. Ces grandeurs ne sont que des grandeurs liées aux interfaces. Les modifications apportées ne se situent donc qu’au niveau de l’espace  $\Gamma$ . Pour rappel  $n_{dim}$ , correspond à la dimension de l’espace d’optimisation, c’est-à-dire au nombre de paramètres à optimiser. Un vecteur des paramètres est donc :  $\mathbf{x} = [x_1 \dots x_{n_{dim}}]^T$ . On cherche alors à résoudre chaque problème mécanique  $k$  associé au jeu de paramètres  $\mathbf{x}^{(k)}$ . Chaque solution sera notée  $\mathbf{s}_k = (\mathbf{F}^{(k)}, \mathbf{W}^{(k)})$ . Cette solution se trouve à l’intersection des sous-espaces  $\mathcal{A}_d$  et  $\Gamma^k$ .

$$\mathbf{s}_k \in \mathcal{A}_d \cap \Gamma^k \quad (1.80)$$

#### 4.3.2 Mise en place de la stratégie

Le premier calcul associé au jeu de paramètres  $\mathbf{x}^{(1)}$  est résolu par l’algorithme itératif LaTIn. Les grandeurs d’interfaces sont initialisées avec des valeurs nulles au démarrage de l’algorithme : solution  $\mathbf{s}_0$ .

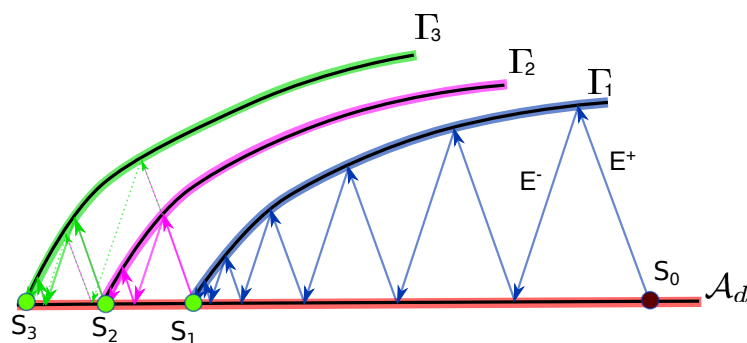


FIGURE 1.20: Illustration de l'approche multiparamétrique

Pour les calculs suivants, on peut choisir de calculer la solution du problème associé au jeu de paramètres  $\mathbf{x}^{(k+1)}$  en initialisant les grandeurs d'interfaces par des valeurs de solutions issues des  $s_j$  précédentes ( $j \in \llbracket 0, k \rrbracket$ ).

La figure 1.20 illustre le principe de la méthode LaTin multiparamétrique où trois problèmes sont à résoudre pour trois différents jeux de paramètres. Lorsque le premier problème est résolu et que la solution  $s_1$  a été trouvée, pour résoudre le problème 2 associé au jeu de paramètres  $\mathbf{x}_2$  on peut alors initialiser l'algorithme LaTin à l'aide des valeurs aux interfaces trouvées à la solution  $s_1$ .

Pour le troisième problème associé à  $\mathbf{x}_3$ , différents choix sont possibles. On peut réinitialiser l'algorithme à l'aide de  $s_0, s_1$  ou  $s_2$ . Toute la difficulté pour optimiser le temps de calcul est de repartir depuis la solution amenant à faire le moins d'itérations possibles.

[Soulie et Boucard, 2009] montre que réinitialiser l'algorithme à partir d'une solution associée au jeu de paramètres le plus proche dans l'espace euclidien est en moyenne la meilleure stratégie à adopter.

## 5 Approches multi-fidélité

### 5.1 Introduction

Comme vu dans la section précédente, il est possible d'obtenir une solution complète en espace et en temps à chaque itération de l'algorithme LaTin. Cette propriété est l'une des bases de ces travaux. En effet, un gain de temps important peut alors espérer être réalisé.

Sur cet exemple (figure 1.21) présenté dans [Boucard et Champaney, 2003], on cherche à déterminer l'effort de réaction de la structure Str2 sur le mur. Le chargement s'effectue en deux temps, la structure Str1 est préchargée puis l'on vient appliquer un effort sur la structure Str2 afin de la pousser contre le mur. Initialement, ce problème se compose de 3 paramètres : le jeu J1 et les coefficients de frottement  $\mu_1$  et  $\mu_2$ . En fixant J1 à  $24 \mu\text{m}$ , le problème est ramené à deux paramètres. On choisit comme fonction objectif l'effort de réaction  $F$  du mur sur la structure Str2.

La figure 1.22 illustre le fait qu'un calcul arrêté de manière prématurée peut fournir des informations quant à la tendance de la fonction objectif que l'on cherche à modéliser. Ce qui est

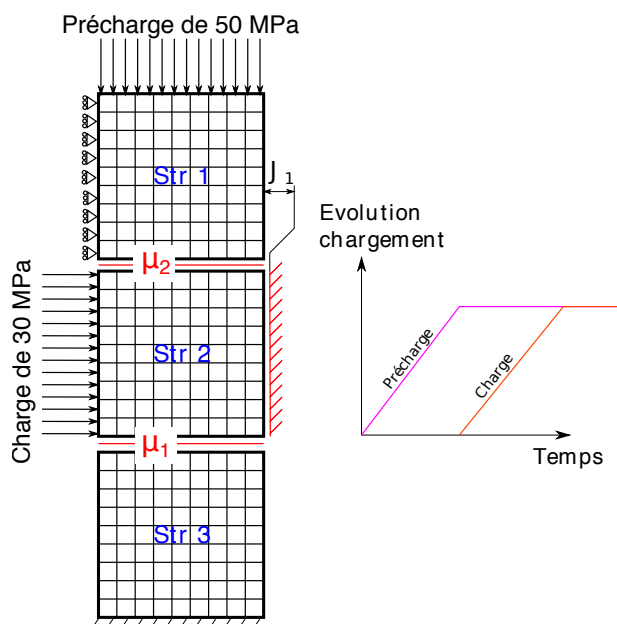


FIGURE 1.21: Problème d'assemblage n°1

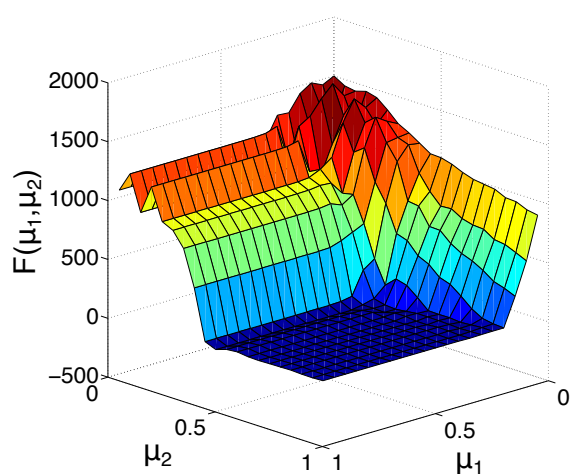
intéressant est que cette tendance peut être obtenue assez rapidement, ce qui permet d'envisager un gain de temps de calcul d'un jeu de paramètres. En effet, pour atteindre le niveau de convergence  $\eta = 10^{-2}$  il a fallu 7 fois moins d'itérations de l'algorithme que pour atteindre le niveau  $\eta = 10^{-4}$ . Un gain de temps de l'ordre de 7 est donc à envisager sur cet exemple entre un point calculé au niveau d'erreur  $\eta = 10^{-2}$  et un point calculé au niveau de référence  $\eta = 10^{-4}$ .

Cette observation est le fondement de cette thèse. Elle vise à utiliser des niveaux de convergence plus faibles de calcul afin notamment de s'insérer dans le processus d'optimisation globale décrit au début de ce chapitre.

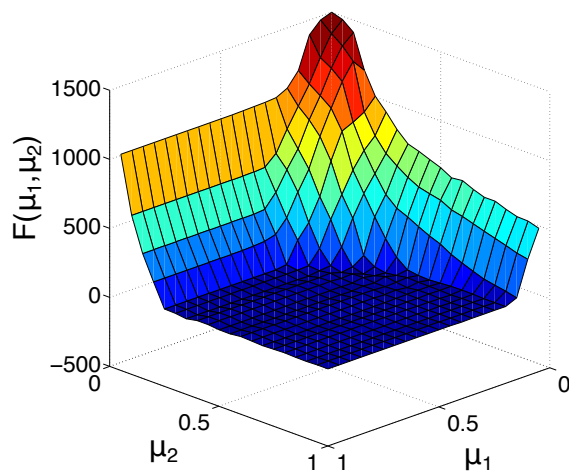
## 5.2 Précisions sur le vocabulaire utilisé

Au cours des travaux réalisés dans cette thèse, on fera référence à différentes qualités de modélisation. Le cadre principal de ce travail est la modélisation **multi-fidélité** (ou à fidélité variable ou à complexité variable). Cette modélisation a pour but d'intégrer des modèles à différents niveaux de fidélité afin de créer un unique métamodèle tirant parti d'informations provenant des différents niveaux (les termes anglais pour cette approche sont principalement dans la littérature *variable fidelity modeling* ou *multifidelity modeling*). Le terme fidélité fait référence au niveau de qualité d'un modèle à représenter le comportement réel d'un système. Différentes approches de modélisation multi-fidélité seront abordées dans la suite de ce document au paragraphe 5.4 et feront l'objet de comparaison en terme d'efficacité tout au long de ce document.

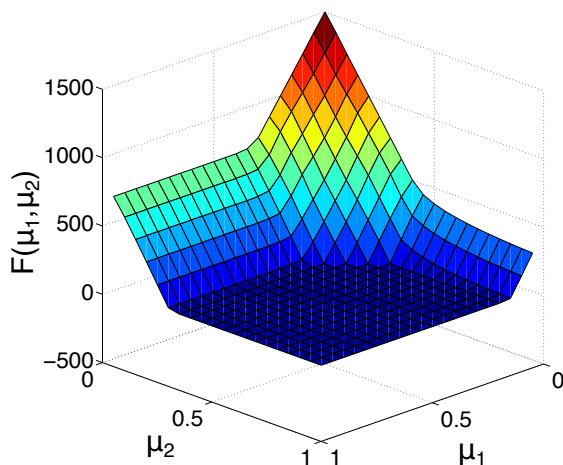
On distinguera alors modèle « haute fidélité » et « basse fidélité ». Le modèle haute fidélité est le modèle sur lequel le calcul sera considéré comme numériquement exact. Le modèle basse fidélité est un modèle dégénéré du précédent plus rapide à calculer mais moins précis. Pour information, les évaluations ou les données obtenues sur le modèle haute fidélité et le modèle basse



(a)  $\eta = 10^{-2}$  - 40 itérations en moyenne par point  $\sim$  7mins



(b)  $\eta = 10^{-3}$  - 80 itérations en moyenne par point  $\sim$  14mins



(c)  $\eta = 10^{-4}$  - 280 itérations en moyenne par point  $\sim$  50mins

**FIGURE 1.22:** Surfaces de réponses obtenues pour différents niveaux de convergence de l'algorithme LaTIn

fidélité peuvent respectivement être appelées « données principales » et « données auxiliaires ». L'utilisation de ces différents modèles ou données peut se classer en plusieurs catégories :

- Modèles hiérarchisés selon leur contenu physique. [Robinson *et al.*, 2008, Choi *et al.*, 2005, Forrester *et al.*, 2007]
- Modèles calculés sur différentes tailles de maillage. [Huang *et al.*, 2006a,

Dadone et Grossman, 2000]

- Même modèle mais calculé pour différents niveaux de convergence de calcul. [Forrester *et al.*, 2006b, Courrier *et al.*, 2014]

Dans notre cas d'étude, le travail effectué s'appuie sur ce dernier point. Afin d'alléger les notations dans la suite de ce document les données haute fidélité seront appelées données « totalement convergées » (*fcv* : fully converged) et les données basse fidélité seront appelées données « partiellement convergées » (*pcv* : partially converged).

Le fait de disposer de différents niveaux de fidélité de modèle peut amener à différentes stratégies d'optimisation et notamment de stratégies multi-niveaux. On présentera alors succinctement la philosophie de ces approches ainsi que deux exemples de stratégies d'optimisation mises en place.

### 5.3 Optimisation par approches multi-niveaux

La stratégie utilisée au sein du laboratoire, ([Buytet *et al.*, 2006, Soulier et Boucard, 2013, Laurent, 2013]) afin de mener à bien le problème d'optimisation est une approche d'optimisation multi-niveaux de modèle (voir figure 1.6) [Robinson et Keane, 1999, Keane et Petruzzeli, 2000, Engels *et al.*, 2004]. D'autres approches multi-niveaux sont envisageables comme notamment l'optimisation multi-niveaux de paramètres qui consistent à remplacer un problème d'optimisation par des sous-problèmes sur un nombre réduit de paramètres [Liu *et al.*, 2004, Kravanja *et al.*, 2003, Chen et Yang, 2005, Bendsøe, 1995].

Les approches multi-niveaux de modèles peuvent elles-mêmes être répertoriées de différentes façons :

- Une première approche consiste à exploiter le résultat obtenu sur le modèle  $i$  au niveau de modélisation  $i + 1$  afin d'affiner la recherche de l'optimum (par exemple [Robinson et Keane, 1999] ou la stratégie décrite sur la figure 1.6).
- Une deuxième approche est de faire communiquer les modèles entre eux. Ils peuvent ainsi s'enrichir mutuellement. On peut citer les travaux de [Engels *et al.*, 2004] consistant à minimiser la masse d'une aile de fuselage sous la contrainte d'inégalité basée sur une limite de contraintes de Von Mises à ne pas dépasser. Deux modèles sont considérés : un modèle éléments finis de l'ensemble de l'aile et un super-élément assurant la description d'une porte. Des optimisations sont menées sur ces deux modèles, chacun prenant compte de contraintes issues de l'autre (conditions aux limites, déformations, etc.).

Pour plus d'informations, on pourra se référer à [Buytet *et al.*, 2006] faisant une bibliographie des méthodes d'optimisations multi-niveaux.

Dans le cadre de l'utilisation de calculs partiellement convergés (ou de calculs basse fidélité), différentes méthodes d'optimisation multi-niveaux peuvent être utilisées. Deux exemples sont données via les méthodes progressives et de réduction d'espace.



### 5.3.1 Méthode progressive

Dadone et al. [Dadone et Grossman, 2000],[Dadone et Grossman, 2003] utilisent une méthode nommée « optimisation progressive ». Cette méthode s'appuie sur des optimisations locales et un raffinement séquentiel du maillage du problème traité. L'idée principale est d'utiliser un problème adjoint dont la solution est le gradient de la fonction d'intérêt. Deux solveurs sont utilisés : un pour le problème direct et un pour le problème adjoint. Les solutions de ces problèmes sont calculées sur le niveau de maillage traité. L'algorithme suit alors la procédure suivante : (1) Effectuer quelques itérations, (2) Evaluer le gradient, (3) Mettre à jour les variables d'optimisation suivant la direction du gradient, (4) Répéter jusqu'à convergence, (5) Réitérer les précédentes étapes à un niveau de maillage plus fin. L'algorithme est stoppé lorsque que la procédure a convergé au dernier niveau de maillage.

Cette méthode s'inscrit donc clairement dans une approche d'optimisation multi-niveaux hiérarchique. Elle utilise différents niveaux de fidélité associés à différentes tailles de maillage. Ainsi, dans [Dadone et Grossman, 2000] trois niveaux de maillages ont été mis en place. Ce type de stratégie pourrait donc être envisageable mais pour différents niveaux de convergence des données. Il s'agit cependant d'une stratégie d'optimisation locale et non globale.

### 5.3.2 Méthode réduction d'espace

Ces méthodes consistent à déterminer la zone de l'optimum global par une suite de constructions de surfaces de réponse de modèles hiérarchisés. L'idée est de détecter la zone où se situe le minimum global à l'aide de nombreuses (mais peu coûteuses) simulations sur le modèle basse fidélité. Une fois cette zone déterminée, la taille de l'espace de conception est réduite (les bornes inférieures et supérieures sont modifiées, le nombre de paramètres à optimiser est conservé). [Giunta *et al.*, 1995] propose d'optimiser sous contraintes un profil d'aile à 4 paramètres. Un premier métamodèle à base de régression polynomiale sur une grille  $6^4$  (1296 points) est ainsi créé. Au final, seulement 157 points sont retenus, un tirage  $D$ -optimal [Fedorov, 1972, Nguyen et Miller, 1992] de 50 points parmi ces 157 est effectué. Puis des simulations de ces points sont réalisés au niveau de modélisation haute fidélité en vue de construire une surface de réponse sur laquelle l'optimisation sera menée. [Forrester *et al.*, 2003] propose le même genre de méthode mais appliqué sur une évolution du modèle basse fidélité. Un solveur de mécanique des fluides commercial *Fluent* est utilisé. Ce solveur utilise un algorithme itératif afin de résoudre le problème soumis. Un modèle basse-fidélité correspond donc au résultat du calcul non convergé. Une série de calculs est menée pour différents niveaux de convergence de l'algorithme (plusieurs modèles basse fidélité sont donc élaborés). L'espace de conception est réduit lorsqu'il est jugé que le métamodèle obtenu à partir des dernières simulations partiellement convergées est suffisamment correct. Les données restant dans l'espace réduit sont calculées complètement convergées et l'optimisation est menée sur un dernier métamodèle construit à partir de ces mêmes données.

Les approches multi-niveaux peuvent s'avérer intéressantes. Elles ont néanmoins quelques limites. Elles peuvent être locales comme la méthode progressive et surtout il est nécessaire de

s'assurer une qualité suffisante du modèle basse fidélité afin de localiser la zone où se situe l'optimum global. Nous verrons plus loin dans le document (voir figure 2.12) que l'optimisation menée sur le modèle basse fidélité peut amener à localiser un jeu de paramètres fortement éloigné du jeu de paramètres optimal recherché. **L'utilisation des données partiellement convergées peut néanmoins s'intégrer dans une démarche d'optimisation globale à l'aide de métamodèles. On parle alors de métamodélisation ou modélisation multi-fidélité.**

## 5.4 Modélisations multi-fidélité

Quelques-une parmi les plus populaires sont notamment issues de la mécanique des fluides et quelques exemples sont présentés dans les paragraphes suivants. On s'intéresse dans cette section aux méthodes permettant la création d'une surface de réponse associée au modèle haute fidélité construite à l'aide de données issues de modèles basse fidélité.

### 5.4.1 Méthode de correction de surface

Les méthodes de correction consistent à créer un métamodèle se substituant au modèle au haute-fidélité à l'aide de données basse-fidélité et d'une fonction de correction. En notant  $\hat{y}_{hf}$  le métamodèle associé au modèle haute-fidélité et  $\hat{y}_{bf}$  le métamodèle associé au modèle basse-fidélité, on peut exprimer la relation entre  $\hat{y}_{hf}$  et  $\hat{y}_{bf}$  par :

$$\hat{y}_{hf}(\mathbf{x}) = f_c(\hat{y}_{bf}(\mathbf{x})) \quad (1.81)$$

Avec  $f_c$  la fonction de correction entre les deux métamodèles. On peut distinguer trois grands types de corrections, *correction additive*, *correction multiplicative* et *correction hybride*.

- *Correction additive* :  $\hat{y}_{hf}(\mathbf{x}) = \gamma_{add}(\mathbf{x}) + \hat{y}_{bf}(\mathbf{x})$
- *Correction multiplicative* :  $\hat{y}_{hf}(\mathbf{x}) = \gamma_{mul}(\mathbf{x}) \cdot \hat{y}_{bf}(\mathbf{x})$
- *Correction hybride* :  $\hat{y}_{hf}(\mathbf{x}) = (1 - \omega) \cdot [\gamma_{add}(\mathbf{x}) + \hat{y}_{bf}(\mathbf{x})] + \omega \cdot \gamma_{mul}(\mathbf{x}) \cdot \hat{y}_{bf}(\mathbf{x})$

On peut trouver dans la littérature différentes approches pour employer ces méthodes. Elles sont relativement similaires ; seule la fonction de correction choisies est différente. On peut cependant distinguer deux approches : l'une s'appuyant sur des corrections locales et l'autre s'appuyant sur des corrections globales. L'approche par correction locale est notamment utilisée par [Alexandrov *et al.*, 2000, Robinson *et al.*, 2008] pour venir assurer les conditions de convergence d'un l'algorithme à région de confiance. En effet, l'optimisation s'effectue sur une fonction modèle définie par  $y_m = \hat{y}_{bf} + \gamma_{add}$  telles que les conditions au premier ordre soient satisfaites afin d'assurer la convergence de l'algorithme. L'approche par correction globale ([Gano et Renaud, 2004, Gano *et al.*, 2006, Vitali *et al.*, 2002, Choi *et al.*, 2005]) consiste à venir estimer une fonction de correction sur l'ensemble du domaine d'étude.

Le lecteur intéressé pourra trouver des références supplémentaires dans [Han *et al.*, 2013]. Cette stratégie de correction globale s'avère d'ailleurs plus efficace avec une correction additive qu'avec une correction multiplicative ([Han *et al.*, 2013]).

De manière plus générale, les approches de corrections de surface s'inscrivent dans des stratégies dites de **space mapping**. Les corrections de surface sont dites stratégies *Output Space mapping*. Les stratégies de space mapping consistent à créer une relation entre les modèles ou les données haute-fidélité et basse-fidélité. Ces stratégies ont été premièrement développées par [Bandler *et al.*, 1994] dans le domaine de l'électronique. On distingue alors deux approches de *space mapping* comme précisé dans [Hassan, 2014] : l'*input space mapping* et l'*output space mapping*.

L'*input space mapping* consiste à créer une relation entre les jeux de paramètres associés aux modèles haute fidélité et basse fidélité. Cette approche est différente de la correction de surface ou *Output space mapping* par le fait que la relation de correction se fait sur les jeux de paramètres et non pas sur les surfaces de réponse.

En notant  $\mathbf{x}^{(hf)} \in \Omega^{(hf)}$  un jeu de paramètres associé à l'espace de conception du modèle haute fidélité et  $\mathbf{x}^{(bf)} \in \Omega^{(bf)}$  un jeu de paramètre associé à l'espace de conception du modèle, on cherche à estimer le modèle haute fidélité via la relation :

$$F_{hf}(\mathbf{x}_{hf}) \simeq F_{bf}(sp(\mathbf{x}_{bf})) \quad (1.82)$$

où  $sp$  est la fonction de *mapping* reliant les données haute fidélité et basse fidélité par la résolution du sous-problème :

$$\mathbf{x}_{bf}^* = sp(\mathbf{x}_{hf}) = \arg \min_{\mathbf{x}_{bf} \in \Omega^{(bf)}} \|F_{bf}(\mathbf{x}_{bf}) - F_{hf}(\mathbf{x}_{hf})\|_2^2 \quad (1.83)$$

Ainsi, on peut résoudre un problème d'optimisation de  $F_{hf}$  par :

$$\min_{\mathbf{x}_{hf} \in \Omega^{(hf)}} F_{hf}(\mathbf{x}_{hf}) \longrightarrow \min_{\mathbf{x}_{hf} \in \Omega^{(hf)}} F_{bf}(sp(\mathbf{x}_{hf})) \quad (1.84)$$

Un intérêt supplémentaire est que la taille des espaces  $\Omega^{(bf)}$  et  $\Omega^{(hf)}$  peut être différent. Le lecteur intéressé pourra trouver des détails supplémentaires dans [Robinson *et al.*, 2008, Choi *et al.*, 2005].

#### 5.4.2 Cokrigage et krigeage hiérarchique

Le cokrigage est une extension multivariable du krigeage, il a été introduit en géostatistique [Matheron, 1970]. Tout comme le krigeage cherche à évaluer une fonction, le cokrigage le fait à l'aide de différentes sources d'informations ([Wachernagel, 2003]). Le point central de la méthode consiste donc à caractériser la corrélation spatiale (ou la covariance, voir paragraphe 3.1.2) entre les différentes sources d'informations. On pourra se référer aux travaux de [Myers, 1982, Isaaks et Srivastana, 1989, Goovaerts, 1997, Wachernagel, 2003] pour une description de l'approche en géostatistique.

Dans le cadre de simulations numériques, on trouve notamment deux grandes familles. La première s'appuie sur l'utilisation des gradients pour construire le métamodèle. On parle alors de cokrigage à gradients [Morris *et al.*, 1993, Koehler et Owen, 1996, Chung et Alonso, 2002, Laurent, 2013]. La seconde s'appuie sur des données issues de modèles numériques de moins

bonne qualité (maillage grossier, calculs partiellement convergés etc. Voir paragraphe 5.2), on parle alors de cokrigeage multi-fidélité [Kennedy et O’Hagan, 2000, Forrester *et al.*, 2007, Zimmermann et Han, 2010, Le Gratiet, 2013]. En expérimentations numériques, celui-ci s’est principalement développé au cours des années 2000, notamment par les travaux menés par [Kennedy et O’Hagan, 2000] proposant un moyen de calculer la corrélation spatiale entre les différentes sources d’informations.

Dans ce travail, les données auxiliaires seront issues du modèle basse fidélité, c’est-à-dire que ce seront des données dites partiellement convergées. On présentera notamment au chapitre 3 différentes façons de calculer la matrice de covariance du cokrigeage ([Kennedy et O’Hagan, 2000, Zimmermann et Han, 2010, Han *et al.*, 2010]) qui est le point central de la méthode. Ces approches seront alors comparées sur des cas test mécaniques pour déterminer laquelle ou lesquelles sont les plus performantes.

Une autre moyen d’intégrer les données basse-fidélité à un modèle mathématique associé à un processus gaussien est le **krigeage hiérarchique** [Han et Görtz, 2012]. Celui-ci consiste à remplacer la fonction polynomiale du krigeage universel (voir paragraphe 3.1.2) par un métamodèle construit uniquement avec des données basse-fidélité. Par rapport au krigeage, son implantation numérique est donc du même ordre de difficulté. Cette approche sera plus précisément présentée au chapitre 3.

## 5.5 Assimilation aux données bruitées

Cette approche consiste à considérer les données issus du modèle basse fidélité comme des données bruitées du modèle haute-fidélité [Picheny *et al.*, 2013, Huang *et al.*, 2006b]. L’idée est donc de créer un métamodèle en considérant la fonction à évaluer comme :

$$y(\mathbf{x}) = \mu(\mathbf{x}) + z(\mathbf{x}) + \varepsilon(\mathbf{x}) \quad (1.85)$$

On note  $\varepsilon_i = \varepsilon(\mathbf{x}_i)$ . De plus :

$$\varepsilon_i \sim \mathcal{N}(0, \tau_i^2) \quad (1.86)$$

Il est important de signaler qu’*a priori* cette erreur est indépendante d’une simulation à une autre,  $\text{Cov}(\varepsilon_i, \varepsilon_j) = 0$  pour tous  $i \neq j$ . On peut alors construire un métamodèle de type de krigeage régressif (le métamodèle n’est plus interpolant aux points évalués à cause de la présence du bruit). La seule modification par rapport à ce qui a été écrit précédemment se situe dans la définition de la matrice de covariance qui s’écrit alors :

$$\mathbf{C} = \begin{bmatrix} \text{Cov}\left(Z(\mathbf{x}^{(1)}), Z(\mathbf{x}^{(1)})\right) + \tau_1^2 & \cdots & \text{Cov}\left(Z(\mathbf{x}^{(1)}), Z(\mathbf{x}^{(n_s)})\right) \\ \vdots & \ddots & \vdots \\ \text{Cov}\left(Z(\mathbf{x}^{(1)}), Z(\mathbf{x}^{(n_s)})\right) & \cdots & \text{Cov}\left(Z(\mathbf{x}^{(n_s)}), Z(\mathbf{x}^{(n_s)})\right) + \tau_{n_s}^2 \end{bmatrix} \quad (1.87)$$

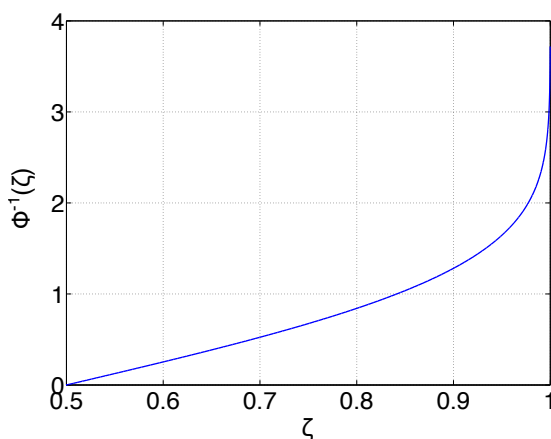
[Forrester *et al.*, 2006b] propose de créer un métamodèle régressif à partir de données bruitées. Le but est d'optimiser un profil d'aile en fonction de variables décrivant la géométrie de celles-ci. Cependant une variation de ces variables engendre nécessairement des modifications de maillage, sources du bruit pour la construction du métamodèle. Il est alors supposé que ce bruit est constant ( $\tau_i^2 = \tau$ ) et est recherché comme un paramètre supplémentaire pour la construction du krigeage.

Cette approche peut être intéressante si l'on souhaite assimiler les données basse fidélité à des données bruitées. Cependant, il est essentiel que les données haute fidélité et basse fidélité soient dans des ordres de grandeurs comparables.

D'autres approches sont alors proposés et s'appuyant sur une stratégie de EGO, en modifiant le critère d'enrichissement. En effet, on cherche à optimiser un quantile. [Picheny *et al.*, 2013] définit le quantile comme :

$$q(\mathbf{x}) = \hat{y}(\mathbf{x}) + \Phi^{-1}(\zeta)s_{\hat{y}}(\mathbf{x}) \quad (1.88)$$

avec  $\zeta = [0.5, 1[$ . La fonction  $\Phi^{-1}(\zeta)$  est illustrée sur la figure 1.23



**FIGURE 1.23:** Fonction  $\Phi^{-1}(\zeta)$  pour  $\zeta = [0.5, 1[$

Une modification de la stratégie EGO est ensuite appliquée. Dans le cas de [Picheny *et al.*, 2013], un calcul d'*expected quantile improvement* est mis en place. Cette approche est assez similaire dans sa formulation à l'*expected improvement* classique (équation 1.66) excepté que la quantité considérée n'est pas la moyenne du krigeage mais son quantile. Cette approche est très intéressante car elle permet la construction d'une stratégie d'optimisation basée sur une infinité de niveaux de précision (contrairement à ce qui peut se faire plus classiquement avec seulement quelques niveaux de précision, deux en général). Le choix de la précision peut également se faire en fonction du coût de calcul que l'on souhaite allouer à l'évaluation. La faiblesse de ces méthodes réside dans le fait qu'il est nécessaire de connaître la variance de l'erreur  $\varepsilon(\mathbf{x})$  pour les niveaux de précision souhaités. Une étude préalable peut donc être à mener.

Dans le cadre où le bruit est supposé avoir une variance constante ( $\tau_i^2 = \tau$ ) [Huang *et al.*, 2006b] utilise également un quantile avec  $\Phi^{-1}(\zeta) = 1$ . Le critère développé dans

[Huang *et al.*, 2006b] est appelé *augmented expected improvement* (AEI) :

$$AEI = (q_{min} - \hat{y}(\mathbf{x})) \Phi \left( \frac{q_{min} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})} \right) + s_{\hat{y}}(\mathbf{x}) \phi \left( \frac{y_{min} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})} \right) \cdot \left( 1 - \frac{\tau}{\sqrt{s_{\hat{y}}(\mathbf{x})} + \tau^2} \right) \quad (1.89)$$

avec  $q_{min} = \min_{i \in \llbracket 1, n_s \rrbracket} q(\mathbf{x}^{(i)})$ . Le terme  $\left( 1 - \frac{\tau}{\sqrt{s_{\hat{y}}(\mathbf{x})} + \tau^2} \right)$  est un terme limitant les réplifications lorsque la prédiction devient plus précise.

Ces méthodes basées sur des quantiles peuvent se montrer tout à fait prometteuses mais il est nécessaire de connaître les caractéristiques du bruit. Si on souhaite adapter ces méthodes aux calculs partiellement convergés, il est donc nécessaire d'évaluer les caractéristiques des différentes évaluations, comme par exemple un écart-type par rapport au calcul totalement convergé. Ceci n'est *a priori* pas possible dans notre cas de calcul avec la méthode LaTin.

Dans le cadre d'étude de fiabilité, ce type d'approche par métamodèle et par quantile est également utilisé afin de localiser les zones où potentiellement peuvent se situer une défaillance [Dubourg *et al.*, 2011]. Un critère d'enrichissement du tirage initial peut alors être le *weighted integrated mean squared error* [Picheny *et al.*, 2010].

## 6 Bilan du chapitre

Le coût de calcul lié aux résolutions non-linéaires ne rend pas viable l'utilisation d'algorithmes d'optimisation classiques. Les travaux précédemment effectués au sein du laboratoire ont abouti à la mise en place d'une stratégie multi-niveaux, consistant à mener une première optimisation globale à l'aide d'un métamodèle puis une seconde locale sur le modèle mécanique réel. Il est cependant nécessaire pour construire un métamodèle d'évaluer un échantillonnage de points qui peuvent être également coûteux. La stratégie de calcul multiparamétrique permet de réduire les coûts de calculs. Afin d'améliorer encore plus les performances atteintes, il est proposé d'utiliser pour la création de métamodèles des stratégies multi-fidélité. Ces stratégies vont s'appuyer dans cette thèse sur le calcul de données partiellement convergées et totalement convergées.

Une problématique associée à ces stratégies est le nombre de points partiellement convergés à calculer lors d'un échantillonnage de l'espace de conception  $\mathcal{D}$  ainsi que le niveau de convergence associé. Afin d'apporter une première réponse à cette problématique une étude de faisabilité est menée au chapitre 2. Une comparaison de l'efficacité de méthodes multi-fidélité quant à l'élaboration de métamodèles sur des cas test sera ensuite abordée au chapitre 3. Cette comparaison se basera essentiellement sur la création de métamodèles ayant un certain niveau de qualité par rapport à un métamodèle de référence en vue de création d'abaque virtuel. Au chapitre 4, l'approche multi-niveaux décrite précédemment sera ensuite mise en place afin de mener à bien un problème d'optimisation et de comparer de nouveau des méthodes multi-fidélité et de juger de leurs performances. Une étude sur des critères d'arrêt de la stratégie EGO sera également

faite. Le dernier chapitre fera l'objet d'optimisation et de création d'abaque virtuel sur des cas test industriels, ainsi que de développements futurs envisageables. Afin de montrer l'apport des méthodes utilisées, chacun des chapitres fera l'objet de comparaisons avec des constructions de métamodèles s'appuyant sur le krigeage de données totalement convergées. Cette méthode sera celle de référence.

## Chapitre 2

# Méthode Evofusion pour l'approximation globale et l'optimisation

*Dans ce deuxième chapitre, on se propose de mener une étude mettant en avant l'intérêt d'utiliser des données partiellement convergées. Pour cela deux points de vue sont adoptés, on se propose de construire un métamodèle pouvant servir d'abaque virtuel ou utilisé en vue de mener une stratégie d'optimisation globale (de type EGO) sur des cas test mécaniques.*

### Sommaire

---

<b>1</b>	<b>Présentation de la problématique . . . . .</b>	<b>72</b>
1.1	Introduction . . . . .	72
1.2	Contexte de l'étude de faisabilité . . . . .	73
1.3	Présentation des cas test traités . . . . .	78
<b>2</b>	<b>Données et quantités d'intérêt de l'étude de faisabilité menée . . . . .</b>	<b>80</b>
2.1	Approche multiparamétrique appliquée aux données partiellement convergées	81
2.2	Présentation des quantités d'intérêt étudiées pour la stratégie de recalage .	83
2.3	Présentation des quantités d'intérêt étudiées pour la stratégie de recherche du minimum . . . . .	85
<b>3</b>	<b>Traitement des cas test . . . . .</b>	<b>89</b>
3.1	Etude du cas test <i>cubes</i> . . . . .	89
3.2	Etude du cas test <i>frette</i> . . . . .	99
<b>4</b>	<b>Bilan de l'étude de faisabilité . . . . .</b>	<b>102</b>

---



# 1 Présentation de la problématique

## 1.1 Introduction

Un point important pour la construction d'un métamodèle est le plan d'expérience ou le tirage utilisé avec lequel le métamodèle est construit. Une approche répandue dans la littérature est l'utilisation d'un tirage LHS [McKay *et al.*, 1979]. Cependant, il est important d'avoir **un nombre suffisant de points** pour fournir un métamodèle de suffisamment bonne qualité. [Jones *et al.*, 1998] propose de calculer  $n_s = 10 \times n_{dim}$  points avant de mener une stratégie EGO (avec  $n_{dim}$  nombre de paramètres du problème mécanique, équivalent à la dimension de la surface de réponse). Ajoutée à ce problème de détermination du nombre de points à utiliser initialement, l'utilisation de données partiellement convergées (données issues d'un calcul itératif non convergé) pose une difficulté supplémentaire qu'est le **niveau de convergence ou de précision** de ces mêmes données. En effet, il est nécessaire de trouver un **équilibre entre coût de calcul et qualité** de la réponse pour construire par la suite un métamodèle le plus **précis** possible en un **temps réduit**.

Pour rappel, dans le cadre de notre travail par approche multi-fidélité, le modèle basse fidélité est mécaniquement et physiquement le même que le modèle haute fidélité, la différence réside dans le niveau de convergence du calcul. **On parle alors de données totalement convergées (haute fidélité) et partiellement convergées (basse fidélité)**.

On s'intéresse dans ce chapitre à déterminer l'existence d'un ou de couples (nombre initial de données partiellement convergées à calculer ; niveau de convergence des données partiellement convergées) afin de construire un métamodèle multi-fidélité précis en un temps réduit (création d'un abaque virtuel) ou utilisable pour mener une stratégie de type EGO dans le but de localiser le minimum d'une fonction objectif. On s'intéresse également à la robustesse des couples envisagés, c'est-à-dire à la stabilité des résultats obtenus pour différents tirages initiaux.

La littérature ne fournit pas de règles spécifiques quant au tirage de points basse fidélité pour la construction de métamodèles. Ce qui peut être proposé est d'effectuer un grand nombre d'évaluations du modèle basse fidélité ([Forrester *et al.*, 2007, Zimmermann et Han, 2010]) et d'utiliser un nombre réduit d'évaluations du modèle haute fidélité. [Huang *et al.*, 2006a] utilise un nombre initial de points partiellement convergés au nombre de  $10 \times n_{dim}$ , ce qui est similaire à la règle de tirages proposée par [Jones *et al.*, 1998]. Il n'y a donc pas de réels consensus quant à un nombre donné de points à évaluer. Des pistes sont cependant proposées telles que [Picheny, 2009] qui propose de répartir le plus possible les points dans l'espace pour minimiser l'IMSE (*Integrated Mean Square Error*) **Le but de ce chapitre est de proposer, dans le cadre de la mécanique des assemblages, un couple (nombre initial de points partiellement convergés ; niveau de convergence des points partiellement convergés) amenant à terme à construire un métamodèle performant.**

## 1.2 Contexte de l'étude de faisabilité

### 1.2.1 Approche avec des données partiellement convergées

Afin de fournir des premiers éléments de réponse quant à la quantité et la qualité des données basse fidélité à calculer, une étude préliminaire a été mise en place. Cette étude se base sur l'utilisation de l'Evofusion [Forrester, 2004, Forrester *et al.*, 2006a]. Il s'agit d'une méthode développée et mise en place pour la prise en compte de données partiellement convergées. **Cette technique peut être classée dans la catégorie des méthodes à correction de surface.** L'idée est de créer un métamodèle représentant le modèle haute fidélité à l'aide d'évaluations de données calculées complètement convergées  $\mathbf{X}_{fcv}$  (*fully converged*) et de données  $\mathbf{X}_{pcv}$  (*partially converged*), la valeur de ces dernières ayant été réévaluée. Cette réévaluation s'effectue grâce à la construction d'un métamodèle d'erreur visant à estimer la différence entre le modèle haute fidélité (calculs complètement convergés) et le modèle basse fidélité (calculs partiellement convergés). Une « fusion de données » est donc effectuée entre les données  $\mathbf{X}_{fcv}$  et les données  $\mathbf{X}_{pcv}$  corrigées.

Quelques propriétés sont à préciser. Dans ce premier cas d'étude, on suppose  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$ , ce qui signifie que les données calculées totalement convergées l'ont également été partiellement. On a alors :

$$\mathbf{X}_{fcv} = [\mathbf{x}^{(1)} \dots \mathbf{x}^{(n_{fcv})}]^T \quad (2.1)$$

$$\mathbf{X}_{pcv} = [\mathbf{x}^{(1)} \dots \mathbf{x}^{(n_{pcv})}]^T \quad (2.2)$$

$n_{fcv}$  et  $n_{pcv}$  (avec  $n_{fcv} > n_{pcv}$ ) font respectivement référence aux nombres de points calculés totalement et partiellement convergés. L'évaluation des données servant à la construction du métamodèle dit « évofusé » se fait par la relation décrite à l'équation 2.3.

$$\hat{y}_{evo}(\mathbf{X}_{pcv}) = y_{pcv}(\mathbf{X}_{pcv}) + \hat{y}_{err}(y_{fcv}(\mathbf{X}_{fcv}), y_{pcv}(\mathbf{X}_{fcv})) \quad (2.3)$$

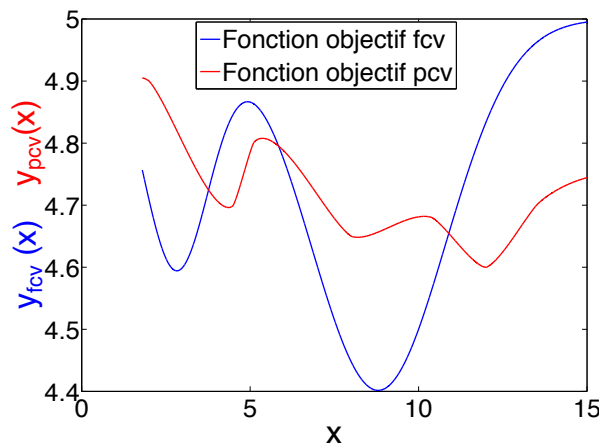
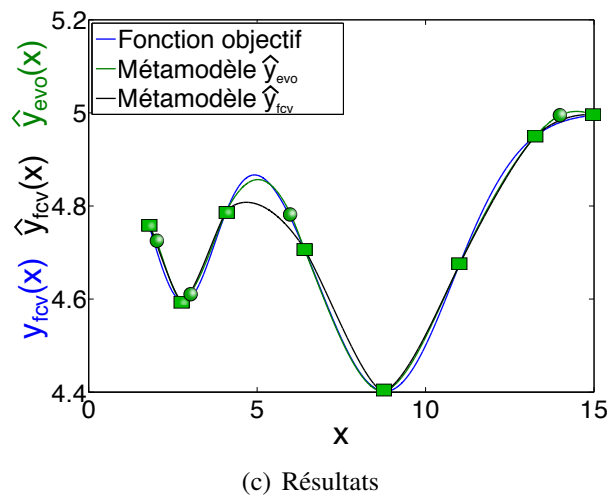
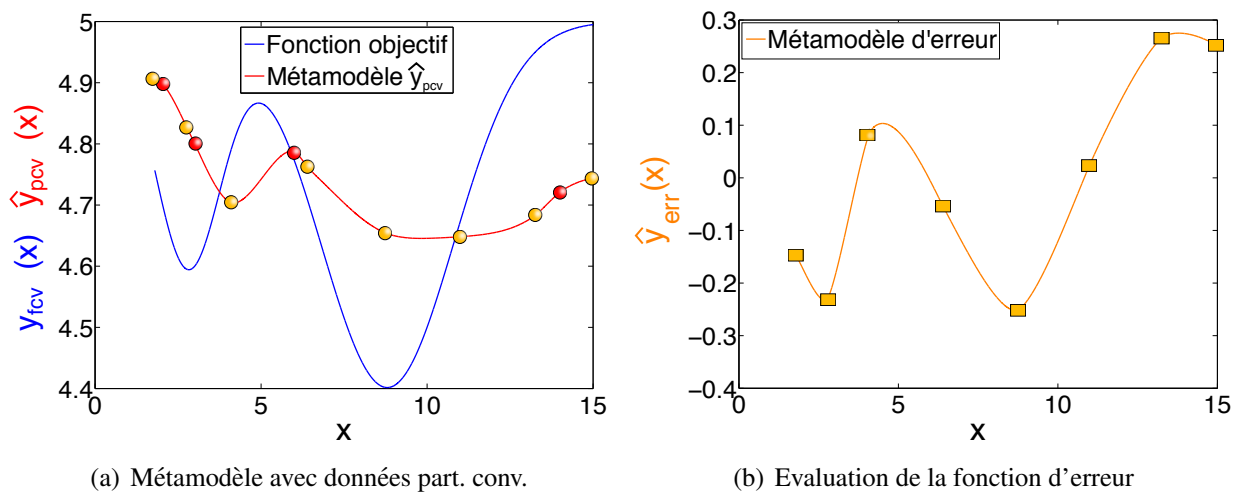


FIGURE 2.1: Fonction objectif représentée pour les deux niveaux de fidélité

$y_{fcv}$  et  $y_{pcv}$  sont respectivement les fonctions objectif associées au modèle haute fidélité (totalement convergé) et au modèle basse fidélité (partiellement convergé). Pour illustrer ce principe, on présente à la figure 2.1 deux fonctions  $y_{fcv}$  et  $y_{pcv}$ .  $\hat{y}_{err}$  est le métamodèle d'erreur ayant pour but d'estimer l'écart entre les fonctions  $y_{fcv}$  et  $y_{pcv}$ . Cette estimation est faite en se basant sur la différence entre les données haute et basse fidélité.  $\hat{y}_{evo}(\mathbf{X}_{pcv})$  est l'évaluation des données servant à la construction du métamodèle évofusé. **Le métamodèle évofusé est alors un métamodèle construit avec le procédé de krigeage aux points  $\mathbf{X}_{pcv}$  ayant pour évaluation  $\hat{y}_{evo}(\mathbf{X}_{pcv})$ .**



**FIGURE 2.2:** Principe illustré de l'Evofusion

● point calculé partiellement convergé, ● point calculé partiellement convergé utilisé pour la construction du métamodèle d'erreur, ■ point calculé de manière totalement convergée et utilisé pour estimer le métamodèle d'erreur, ■ point étant calculé comme la différence entre un point totalement et partiellement convergé (■ = ■ - ●), ● point calculé initialement partiellement convergé puis corrigé par le métamodèle d'erreur

A la figure 2.2(a), on présente l'évaluation de la fonction  $y_{pcv}$  par un métamodèle. Celui-ci est obtenu à l'aide de 12 points dont 8 sont utilisés pour estimer le métamodèle d'erreur. La figure 2.2(b) présente le métamodèle d'erreur entre  $y_{fcv}$  et  $y_{pcv}$  estimé en 8 points. Finalement, la figure 2.2(c) présente le métamodèle évofusé  $\hat{y}_{evo}$  construit à l'aide de 12 points (8 calculés de manière totalement convergée et 4 provenant de la correction des données partiellement convergées par le métamodèle d'erreur).

Ce petit exemple illustre le fait que l'on espère que le métamodèle  $\hat{y}_{evo}$  soit de meilleure qualité que le métamodèle  $\hat{y}_{fcv}$  construit avec uniquement les données totalement convergées ■, ce qui est bien le cas sur cette illustration voir figure 2.2(c). Cet exemple n'est fourni qu'à titre indicatif mais illustre la nécessité d'utiliser un couple (nombre de points partiellement convergés ; niveau de convergence) adéquat afin de construire un métamodèle  $\hat{y}_{evo}$  de qualité.

### 1.2.2 Stratégie d'initialisation

Afin d'essayer d'avoir une certaine qualité de corrélation entre les données provenant du modèle partiellement convergé et celles du modèle totalement convergé, [Forrester, 2004, Forrester *et al.*, 2006a] propose une étude préliminaire. Cette étude vise à déterminer combien de points initiaux partiellement convergés sont calculés et avec quel niveau de convergence. Le but est d'estimer les grandes tendances du modèle haute fidélité à l'aide de ces données partiellement convergées. Un premier tirage de points est évalué avec un niveau de convergence faible. Cet échantillonnage est ensuite enrichi et calculé avec un niveau de convergence légèrement plus avancé. On vient ensuite comparer les métamodèles ainsi créés. Lorsque l'évolution entre le dernier métamodèle créé et quelques précédents est suffisamment faible, l'étude est arrêtée. Le niveau de convergence (ou nombre d'itérations du solveur) nécessaire aux données calculées partiellement convergées est alors fixé au dernier niveau de convergence (ou d'itérations) calculé. Dans notre travail cette recherche préliminaire n'est pas effectuée. En effet, on souhaite essayer de trouver une règle heuristique pour la construction du métamodèle multi-fidélité.

Afin de trouver quel nombre de points initiaux partiellement convergés et avec quel niveau de précision les calculer, on se propose de mener deux études : l'une visant à **déterminer le minimum** d'une fonction objectif et l'autre visant à **construire un métamodèle de bonne qualité (abaque virtuel) sur l'ensemble du domaine d'étude**. On s'intéressera au temps de simulation nécessaire pour venir calculer les points formant le métamodèle et à la robustesse des résultats obtenus pour différents tirages initiaux. On remarquera que le temps de construction ou d'évaluation du métamodèle n'est pas pris en compte. En effet, le but de ce genre de travail est de s'intéresser à des problèmes de grandes tailles où le temps de simulation est prépondérant devant les autres temps de calcul. Par exemple, comme dit au paragraphe 2.1 du chapitre 5, le temps de simulation d'un jeu de paramètres sur le cas traité de taille industrielle est de 12h.

### 1.2.3 Description de l'étude effectuée

Afin de rechercher le minimum de la fonction objectif, une stratégie de type EGO est envisagée (voir paragraphe 3.3). Cette stratégie se nommera par la suite **stratégie de recherche du minimum**. La stratégie visant à créer un métamodèle de qualité sur  $\mathcal{D}$ , dite **stratégie de recalage**,

se base sur un enrichissement de l'échantillonnage initial au point maximum d'erreur quadratique moyenne du **métamodèle d'erreur**. En effet, le métamodèle  $\hat{y}_{evo}$  est construit avec les données partiellement convergées corrigées (●) et totalement convergées (■). Ainsi, en tous ces points, l'erreur quadratique moyenne du métamodèle évofusé est nulle. Or, puisque les données partiellement convergées sont corrigées et ne sont qu'une approximation de la valeur réelle en ce point, l'erreur d'évaluation n'est pas nécessairement nulle. C'est la raison pour laquelle l'enrichissement se fait en se basant sur le métamodèle d'erreur. Ce choix est viable puisqu'il s'effectue là où la prédiction du métamodèle d'erreur (et donc de correction) est probablement la moins juste.

Cependant, concernant la stratégie de recherche de minimum par approche EGO, le calcul de l'*expected improvement* est quand même effectué en supposant que l'erreur quadratique moyenne aux points partiellement convergés corrigés (●) est nulle, tout comme cela est fait dans l'approche originale ([Forrester, 2004, Forrester *et al.*, 2006a]). La raison principale étant qu'un autre choix est difficilement possible pour cette approche puisque qu'une évaluation de l'erreur en ces points est difficilement envisageable. Néanmoins, si le métamodèle  $\hat{y}_{evo}$  est proche du modèle réel, cette hypothèse est valable. Cette difficulté d'estimation de l'erreur quadratique moyenne n'est pas spécifique à l'Evofusion mais aux méthodes de correction de surface en général [Han *et al.*, 2013]. Ce point technique montre une difficulté à laquelle ce travail se heurte ; il s'agit de la combinatoire que l'on peut obtenir avec l'ensemble des déclinaisons d'une technique.

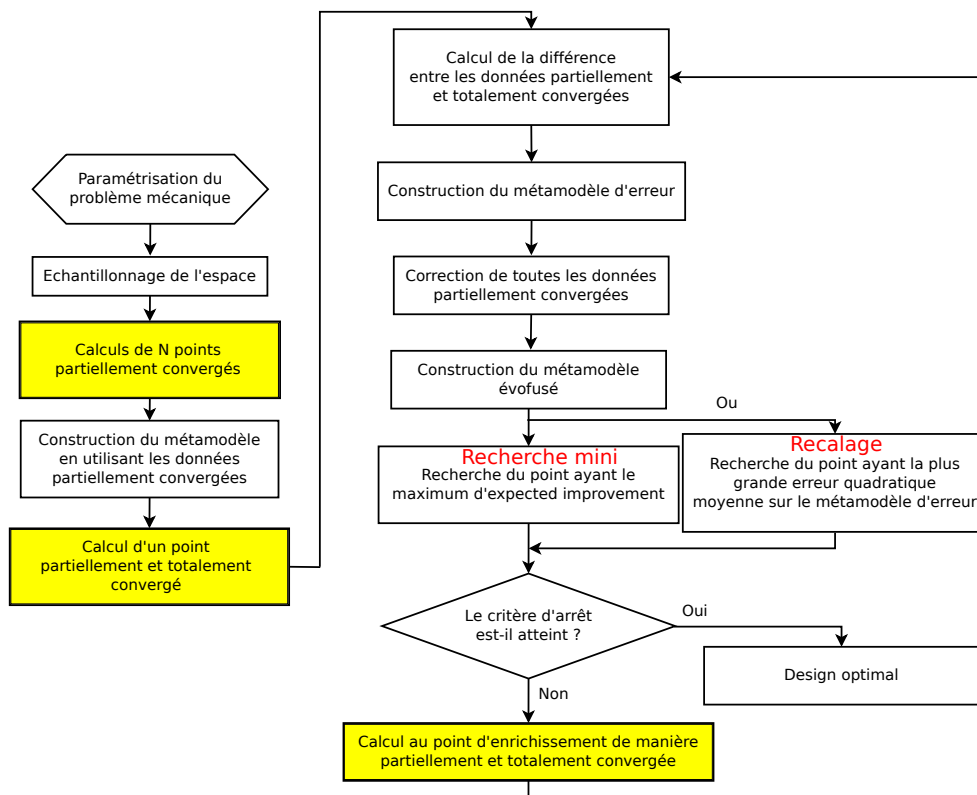
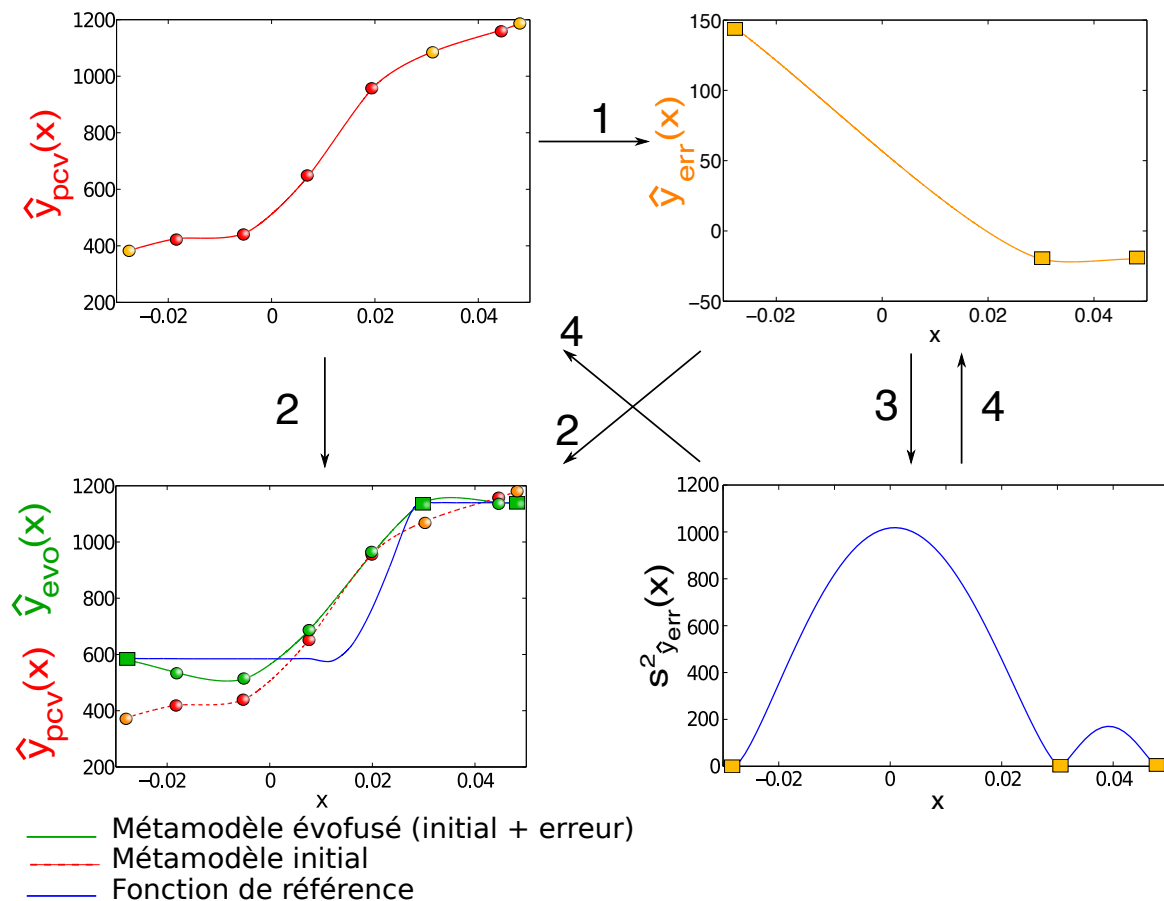


FIGURE 2.3: Algorithme de l'Evofusion

On présente à la figure 2.3 l'algorithme des stratégies mises en place. Les étapes les plus coûteuses en temps de calcul sur un cas test industriel sont représentées en jaune. Ce sont celles dont le temps de calcul est évalué dans la suite de ce travail. La figure 2.4 illustre sur un simple exemple la stratégie de recalage par Evofusion. La stratégie de recherche du minimum diffère uniquement à l'enrichissement.



**FIGURE 2.4:** Stratégie recalage :

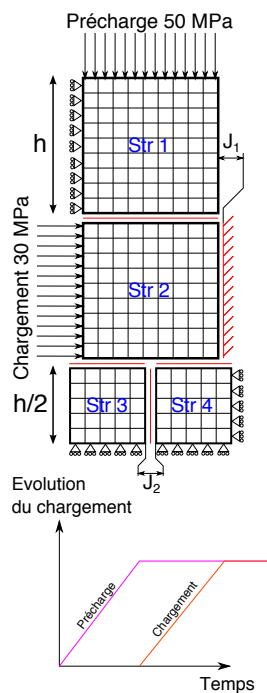
- (0) Calcul de données partiellement convergées ● ou ●. (1) Construction du métamodèle d'erreur. (2) Construction du métamodèle évofusé  $\hat{y}_{evo}$  à l'aide des données ■ et ●. (3) Recherche du maximum d'erreur quadratique moyenne (4) Enrichissement au point du maximum d'erreur quadratique moyenne du métamodèle d'erreur.

### 1.3 Présentation des cas test traités

Afin de déterminer quels couples (nombre initial de points partiellement convergés ; niveau de convergence des points partiellement convergés), utiliser pour une application en mécanique des solides de l'Evofusion, deux cas test sont introduits. Le premier est un cas test multi-contacts avec jeu aux interfaces, frottement et glissement. Il est donc représentatif des problématiques rencontrées en assemblage mécanique. Le second s'approche d'un cas test industriel et a pour intérêt de comporter 5 paramètres.

#### 1.3.1 Cas test *cubes*

Le premier cas test est académique et introduit dans [Laurent, 2013, Courrier *et al.*, 2014] (voir figure 2.5). Il s'agit d'un cas assez représentatif des problématiques rencontrées en simulation numérique d'assemblage. En effet, il s'agit d'un cas multi-contacts avec prise en compte de jeu et de frottement aux interfaces. Il comporte 3 paramètres ( $n_{dim} = 3$ ) : 2 coefficients de frottement et 1 jeu. Le maillage est constitué de 2200 degrés de liberté.



\* Coefficient de frottement entre les structures Str2 et Str3 :  $\mu_{Str2/Str3}$

\* Coefficient de frottement entre les structures Str2 et Str4 :  $\mu_{Str2/Str4}$

\* Coefficient de frottement entre les structures Str1 and Str2 : 0.1

\* Jeu entre les structures Str3 et Str4 :  $j2$

\* Jeu entre la structure Str2 et le mur :  $j1=0.024$  mm

FIGURE 2.5: Premier exemple traité : Cas test *cubes*

La plage de variations des paramètres est définie au tableau 2.1. On note  $F$  l'effort de réaction du mur sur la structure. Cette quantité d'intérêt est l'objet de l'étude sur ce cas test.

Paramètres	min	max
$\mu_{Str2/Str3}$	0.0	0.95
$\mu_{Str2/Str4}$	0.0	0.95
$j2$ (en mm)	-0.028	0.048

TABLE 2.1: Bornes de conception pour le cas test *cubes*

### 1.3.2 Cas test *frette*

Le second exemple a été introduit dans [Soulier et Boucard, 2013]. Il s'agit d'étudier le comportement d'une frette. Une frette est un élément technologique de transmission d'efforts entre un arbre et un moyeu (ou roue dans le cas présent). Une série de vis de la frette viennent mettre en charge le montage par l'intermédiaire de bagues coniques et biconiques. La liaison encastrement entre la roue et l'arbre est donc réalisée par adhérence via cette frette. Les différents paramètres étudiés sont au nombre de cinq (3 coefficients de frottements et 2 jeux). L'exemple traité est un cas 2D axi-symétrique représentant ce montage industriel. Le maillage est composé de 7600 degrés de liberté. On note  $M_a$  le moment transmissible à l'arbre.

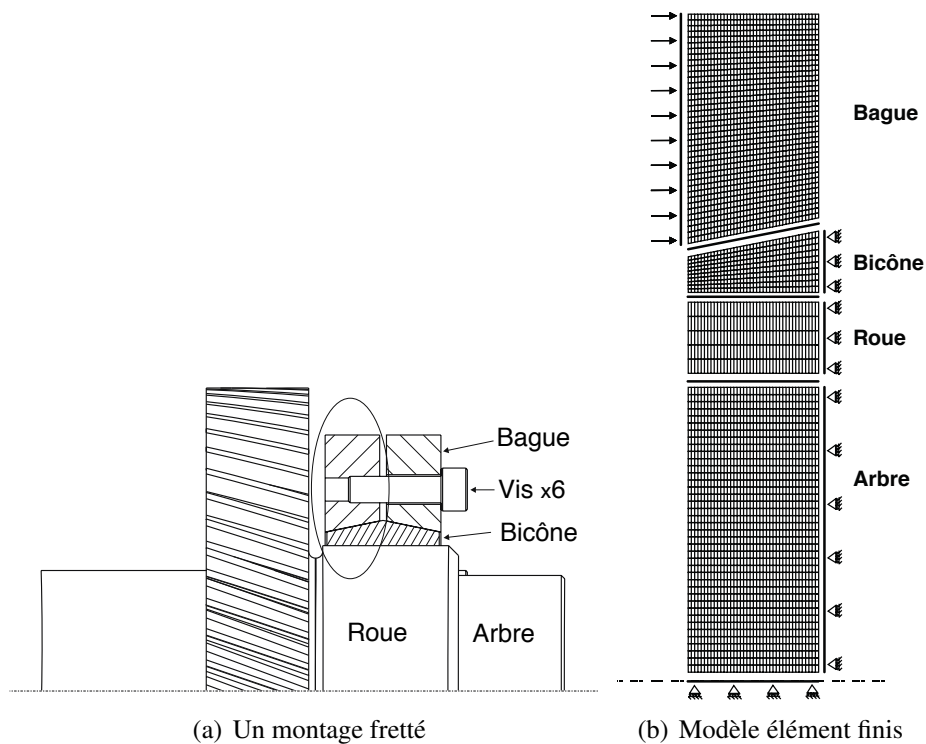


FIGURE 2.6: Second exemple traité : Cas test *frette*



Les 5 variables de conception sont :

- 3 coefficients de frottement :
  - $\mu_{A-R}$  à l'interface Arbre-Roue
  - $\mu_{R-B}$  à l'interface Roue-Bicône
  - $\mu_{B-B}$  à l'interface Bicône-Bague
- 2 jeux :
  - $j_1$  à l'interface Arbre-Roue
  - $j_2$  à l'interface Roue-Bicône

La plage de variation des paramètres est définie au tableau 2.2.

Paramètres	min	max
$\mu_{A-R}$	0.05	0.5
$\mu_{R-B}$	0.05	0.5
$\mu_{B-B}$	0.05	0.5
$j_1$ (en mm)	0	0.049
$j_2$ (en mm)	0.036	0.106

**TABLE 2.2:** Bornes de conception pour le cas test *frette*

Les paramètres mini et maxi des jeux  $j_1$  et  $j_2$  sont issus des ajustements proposés par le constructeur STUWE<sup>1</sup> dont le cas test traité est une libre adaptation.

## 2 Données et quantités d'intérêt de l'étude de faisabilité menée

On rappelle que l'on cherche à estimer quel couple (nombre initial de points ; niveau convergence) est performant dans le cadre de l'utilisation de données partiellement convergées. Pour cela, deux études sont menées : une visant à construire un métamodèle de bonne qualité, l'autre visant à chercher le minimum global.

On s'intéressera dans un premier temps à construire un métamodèle de bonne qualité ou abaque virtuel. Pour ce faire, un métamodèle de référence est créé, l'enrichissement est stoppé lorsque le métamodèle  $\hat{y}_{evo}$  a atteint une corrélation  $r_{ccc} > 0.95$  par rapport à ce métamodèle de référence.

La seconde approche met en place une stratégie de type EGO qui devra permettre de localiser la zone où le minimum global de la fonction objectif se situe.

Les deux approches ont été réalisées à l'aide de la toolbox DACE 2.0 [Lophaven *et al.*, 2002b, Lophaven *et al.*, 2002a]. (La toolbox GRENAT disponible au sein du laboratoire n'était pas encore aboutie au moment où l'ensemble de cette étude de faisabilité a été menée).

1. [www.stuwe.de](http://www.stuwe.de)

On s'intéressera également à employer l'approche d'Evofusion couplée à la stratégie multiparamétrique de l'algorithme LaTIn. Par cette approche, le calcul d'un nouveau jeu de paramètres fournit à convergence le même résultat indépendamment de la solution utilisée pour initialiser celui-ci. Cependant, qu'en est-t-il pour une donnée dite partiellement convergée ? On s'intéressera avant toute étude à la possibilité ou non d'utiliser le calcul multiparamétrique pour une donnée partiellement convergée.

## 2.1 Approche multiparamétrique appliquée aux données partiellement convergées

L'utilisation de l'approche multiparamétrique pour mener à bien les calculs des données partiellement convergées peut être problématique. Sur la figure 2.7(a), on illustre le calcul d'un premier jeu de paramètre. Le calcul est initialisé par une solution  $S_0$ , dite nulle, c'est-à-dire sans contenu mécanique. Le calcul de la donnée partiellement convergée aboutit à la solution  $S_{1\_pcv}$ . Le calcul d'un second jeu de paramètre peut alors s'effectuer de deux manières distinctes : soit, en initialisant l'algorithme LaTIn par  $S_0$ , soit par  $S_{1\_pcv}$ , chacun des calculs aboutissant respectivement à la solution  $S_{2\_pcv}$  et  $S'_{2\_pcv}$  (voir figure 2.7(b)).

Ces deux solutions sont distinctes et peuvent même alors être très « éloignées » l'une de l'autre. Leur contenu mécanique peut donc être potentiellement très différent. Ceci peut poser problème puisque le métamodèle basse fidélité pourrait être construit avec des points n'ayant pas le même contenu physique. Lorsque le calcul s'effectue sur une donnée dite totalement convergée, cette problématique du point de départ n'est pas présente puisque l'indicateur d'erreur  $\eta$  à atteindre pour stopper l'algorithme de calcul LaTIn est suffisamment faible. Les solutions finalement calculées sont donc tout à fait semblables.

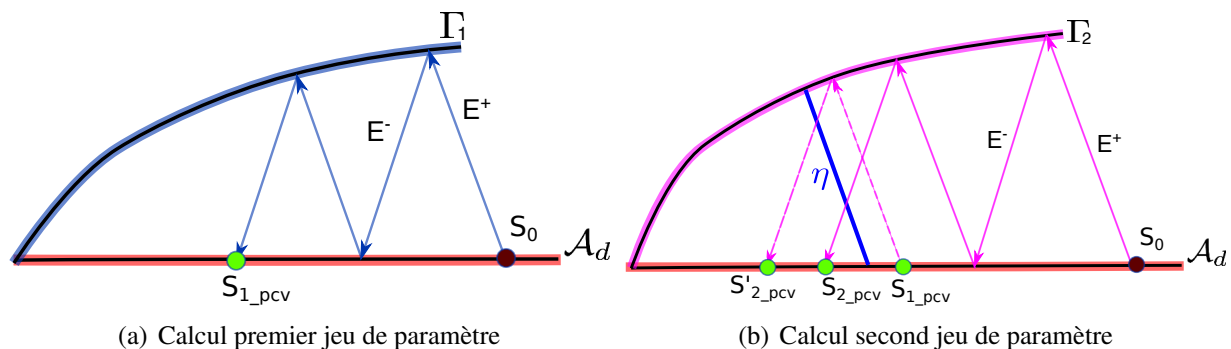


FIGURE 2.7: Illustrations de l'approche multiparamétrique LaTIn pour le calcul des données partiellement convergées

Afin de tester l'impact de l'approche multiparamétrique sur les résultats de calcul de données partiellement convergées, une série d'évaluations de la réponse obtenue est effectuée à l'aide du cas test *cubes*. Sur ce cas test, on impose :

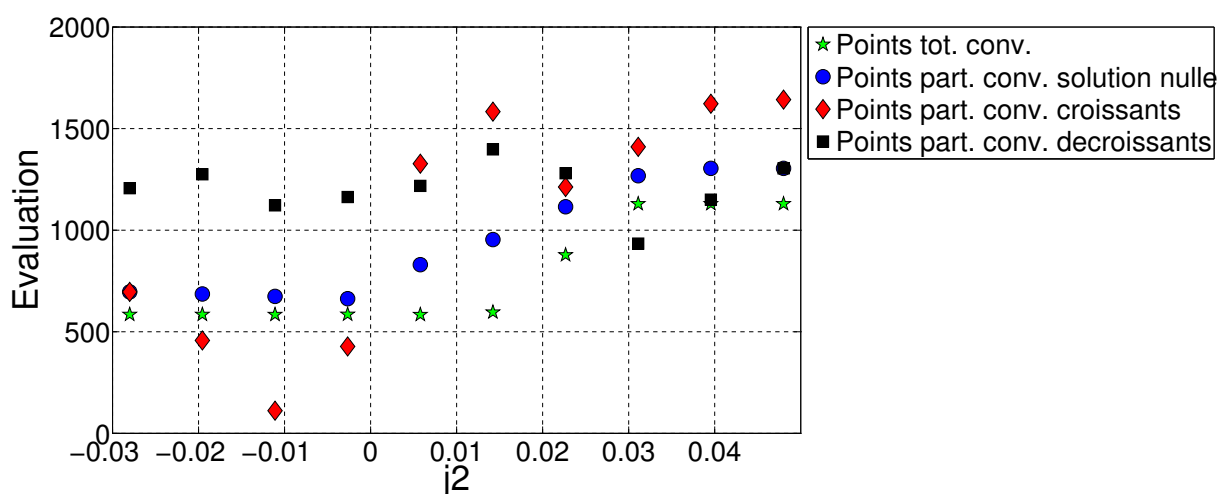
$$\text{— } \mu_{Str2/Str3}=0.5$$

$$\text{— } \mu_{Str2/Str4}=0.1$$

On se ramène donc à un cas d'étude à 1 paramètre. Pour illustrer les différentes évaluations en fonction d'une certaine stratégie multiparamétrique choisie, on propose 3 façons d'appréhender le calcul de données. 10 points sont à calculer à intervalle régulier. Les points à calculer sont rangés dans un ordre croissant ou décroissant suivant  $j_2$ . Les données partiellement calculées sont évaluées pour un niveau d'indicateur d'erreur LaTin  $\eta = 0.05$ .

- On calcule chaque donnée en réinitialisant l'algorithme avec une solution nulle (c'est-à-dire sans aucun contenu mécanique)  $S_0$ .
- On calcule chaque donnée en réinitialisant l'algorithme à l'aide de la solution précédemment calculée, les données étant évaluées de manière croissante suivant le paramètre  $j_2$ .
- On calcule chaque donnée en réinitialisant l'algorithme à l'aide de la solution précédemment calculée, les données étant évaluées de manière décroissante suivant le paramètre  $j_2$ .
- On calcule de manière totalement convergée les données afin d'avoir une mesure de comparaison.

Pour illustrer les différences d'évaluation entre ces différentes approches, on présente les résultats sur la figure 2.8.



**FIGURE 2.8:** Comparaison de stratégies multiparamétriques de calcul de données partiellement convergées

On constate clairement sur cette figure, les écarts d'évaluation entre les différentes stratégies. De plus, on remarque que réinitialiser l'algorithme par une solution nulle fournit des résultats qui suivent la tendance des résultats calculés totalement convergés. Ces résultats sont également moins erratiques que pour les deux autres stratégies d'initialisation de calcul. Le point d'initialisation de la stratégie LaTin a donc un réel impact sur la solution calculée partiellement convergée. Celle-ci peut alors varier fortement en fonction de ce point d'initialisation.

**Bilan :** Dans le but d'obtenir des données partiellement convergées cohérentes entre elles, on choisit de calculer celles-ci en réinitialisant l'algorithme LaTIn à l'aide d'une solution sans contenu mécanique. Certes, on perd l'intérêt de calculer un jeu de paramètres à l'aide d'une solution précédemment obtenue mais le coût de calcul associé au niveau de fidélité partiellement convergé est faible par rapport au niveau de fidélité totalement convergé. Le fait de ne pas utiliser pleinement la stratégie multiparamétrique pour le calcul de données partiellement convergées n'est donc pas un réel handicap.

## 2.2 Présentation des quantités d'intérêt étudiées pour la stratégie de recalage

Le but de la stratégie de recalage est de créer un métamodèle multi-fidélité de bonne qualité (pouvant faire office d'abaque virtuel). Afin de trouver le couple (nombre initial de points ; niveau convergence) répondant à cette problématique le plus rapidement et le plus efficacement possible, différentes quantités d'intérêt sont étudiées pour cette stratégie. Le but de ce paragraphe est de les présenter et de justifier de leur utilisation. Certaines quantités d'intérêt comme notamment les temps de calcul sont présentés **avec utilisation ou non de la stratégie multiparamétrique**. Cette distinction est principalement faite pour vérifier si des couples (nombre initial de points ; niveau convergence) potentiellement intéressants dans une approche de calcul le sont aussi dans l'autre.

Afin de trouver le ou les couples (nombre initial de points ; niveau convergence) intéressants pour mener des études avec des points partiellement convergés,  $n_{b_{test}}$  différents tirages LHS initiaux sont effectués pour un même couple étudié. On s'attachera à regarder quels sont les couples (nombre initial de points ; niveau convergence) fournissant les temps de calcul moyens les plus rapides et les plus robustes en vue de construire un métamodèle multi-fidélité de bonne qualité. Pour rappel, sur le premier cas test la fonction objectif est définie par :

$$y(\mathbf{x}) = F(\mathbf{x})$$

Sur le second cas test, on cherche à construire un métamodèle de la fonction objectif

$$y(\mathbf{x}) = M_a(\mathbf{x})$$

La stratégie d'Evofusion couplée avec la méthode multiparamétrique est appliquée de la façon suivante : les données partiellement convergées sont calculées en partant d'une solution nulle (voir paragraphe 2.1). Les données totalement convergées sont calculées en partant d'une solution totalement convergée la plus proche dans l'espace euclidien (sauf bien sûr le premier calcul), voir chapitre 1.

Ainsi, le temps de calcul est défini par :

$$T_{evo\_multi} = \underbrace{T_{pcv}^{k \times n_{dim} + n_{enri}, err}}_{\text{Temps de calcul des données partiellement convergées}} + \underbrace{T_{fcv\_multi}^{n_{enri}, ref}}_{\text{Temps de calcul des données totalement convergées avec calcul multiparamétrique}} \quad (2.4)$$

$n_s = k \times n_{dim}$  correspond au nombre de points calculés initialement faisant l'objet de cette étude.  $n_{enri}$  est le nombre de points ajoutés lors de l'algorithme de recalage (voir figure 2.3).  $err$  est le niveau de convergence de l'algorithme LaTIn à atteindre pour le calcul des données partiellement convergées, second élément étudié dans cette étude.  $ref$  correspond au niveau de convergence de référence de l'algorithme LaTIn (niveau utilisé pour calculer les données totalement convergées).  $T_{pcv}^{k \times n_{dim} + n_{enri}, err}$  est donc le temps de calcul total associé aux données partiellement convergées. De même,  $T_{fcv\_multi}^{n_{enri}, ref}$  est le temps de calcul total associé aux données totalement convergées.

L'Evofusion sans utilisation de l'approche multiparamétrique n'est pas directement effectuée. Le temps de simulation est une évaluation en fonction du nombre de points calculés. Ceci a été fait également dans [Laurent, 2013] afin de limiter le nombre de données à gérer et les temps de calcul. Le temps estimé pour calculer une donnée totalement convergée est supposé être égal à celui nécessaire pour calculer la première donnée totalement convergée de la méthode. Lors de la phase d'enrichissement, puisqu'une même donnée est calculée partiellement et totalement convergée, le temps de calcul au niveau partiellement convergé est un temps masqué par rapport au temps de calcul du niveau totalement convergé. Ainsi, on définit le temps de calcul de la stratégie de recalage sans approche multiparamétrique par :

$$T_{evo\_sans\_multi} = \underbrace{T_{pcv}^{n \times dim, err}}_{\text{Temps de calcul des données partiellement convergées}} + \underbrace{T_{fcv\_sans\_multi}^{n_{enri}, ref}}_{\text{Temps de calcul des données totalement convergées sans calcul multiparamétrique}} \quad (2.5)$$

$$\text{avec : } T_{fcv\_sans\_multi}^{n_{enri}, ref} = T_{fcv\_sans\_multi}^{1, ref} \times n_{enri}$$

Le temps moyen de calcul avec l'approche multiparamétrique est donc :

$$\langle T_{evo\_multi} \rangle = \frac{1}{n_{b_{test}}} \sum_1^{n_{b_{test}}} T_{evo\_multi} \quad (2.6)$$

et celui estimé sans l'approche multiparamétrique

$$\langle T_{evo\_sans\_multi} \rangle = \frac{1}{n_{b_{test}}} \sum_1^{n_{b_{test}}} T_{evo\_sans\_multi} \quad (2.7)$$

Le gain moyen est également défini par :

$$\begin{aligned} Gain_{reca\_sans\_multi} &= \frac{T_{meta\_ref\_sans\_multi}}{\langle T_{evo\_sans\_multi} \rangle} \\ Gain_{reca\_multi} &= \frac{T_{meta\_ref\_multi}}{\langle T_{evo\_multi} \rangle} \end{aligned} \quad (2.8)$$

Avec  $T_{meta\_ref\_sans\_multi}$ , le temps évalué pour calculer les données formant le métamodèle de référence sans l'approche multiparamétrique.  $T_{meta\_ref\_multi}$  est le temps nécessaire pour calculer ces mêmes données avec l'approche multiparamétrique. Ce gain est un moyen supplémentaire de mettre en valeur les zones où les couples (nombre initial de points ; niveau convergence) sont intéressants.

On s'est également intéressé à la robustesse des temps de calcul obtenus pour les couples (nombre initial de points ; niveau convergence) étudiés. En effet, on cherche des couples fournissant non seulement des temps de calcul moyens intéressants, mais ayant également une certaine robustesse, c'est-à-dire un écart-type des temps de calcul le plus faible possible. Pour cela, on s'appuie sur le calcul de l'écart type relatif permettant de comparer des séries de données n'ayant pas le même ordre de grandeur. L'écart type relatif est défini comme le quotient de l'écart type d'une série et de sa moyenne. Ainsi, en notant  $\epsilon$  une série de données, telle que :

$$\epsilon = \{ \epsilon_1 \quad \dots \quad \epsilon_n \} \quad (2.9)$$

on définit  $\langle \epsilon \rangle$  la moyenne de cette série et  $\sigma_\epsilon$  son écart type. L'écart-type relatif est alors fourni par l'équation 2.10 :

$$\sigma_{\epsilon \text{ norm}} = \frac{\sigma_\epsilon}{\langle \epsilon \rangle} \quad (2.10)$$

## 2.3 Présentation des quantités d'intérêt étudiées pour la stratégie de recherche du minimum

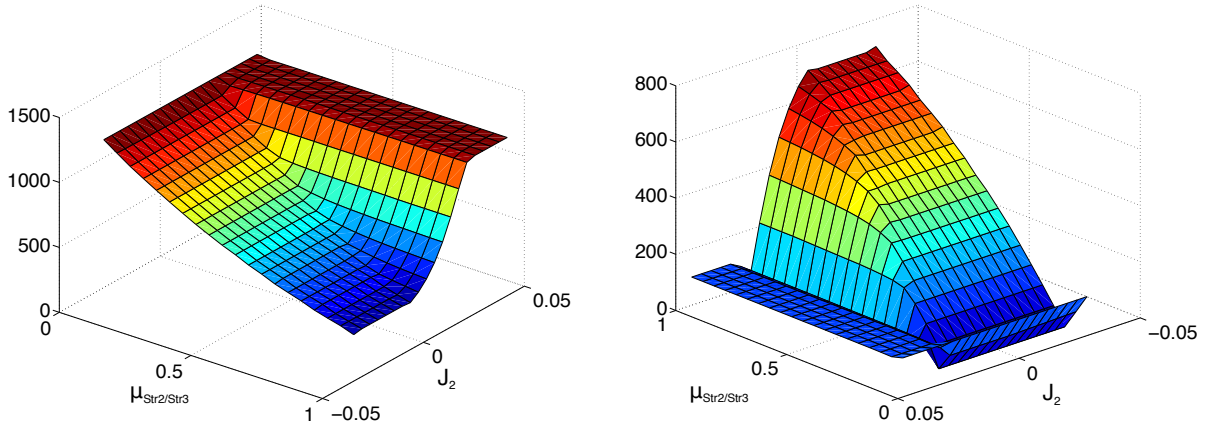
Tout comme pour la stratégie de recalage, on présente dans ce paragraphe les quantités d'intérêt étudiées pour trouver les couples (nombre initial de points ; niveau convergence) efficace dans une stratégie de recherche d'optimum de fonction objectif effectuée par une approche EGO. On s'intéresse plus spécifiquement aux gains de temps de calcul obtenus à l'aide de l'EvoFusion (voir figure 2.3) par rapport à une stratégie EGO classique (c'est-à-dire ne s'appuyant que sur un métamodèle de krigeage de données totalement convergées). Cependant, des aménagements ont été nécessaires pour réaliser une stratégie de recherche de minimum viable. Ces aménagements sont tout d'abord présentés.

Le minimum ou le maximum des fonctions objectif précédemment définies étaient trop simples à situer car se trouvaient dans un coin de la frontière du domaine d'étude. Ils étaient alors localisés en quelques itérations par la stratégie de recherche de minimum (algorithme en figure 2.3). Afin de rendre cette étude de recherche d'optimum plus intéressante pour obtenir des temps de comparaison valables, on s'est intéressé pour le cas test *cubes* à rechercher un minimum de la fonction objectif :

$$y(\mathbf{x}) = \| F(\mathbf{x}) - 1000N \|$$

Pour le second problème on s'intéresse à trouver un jeu de paramètres minimisant la fonction objectif :

$$y(\mathbf{x}) = \| M_a(\mathbf{x}) - 15000N.m \|$$



(a) Surface de réponse pour  $y(\mathbf{x}) = F(\mathbf{x})$

(b) Surface de réponse de  $y(\mathbf{x}) = \| F(\mathbf{x}) - 1000N \|$

**FIGURE 2.9:** Surfaces de réponse pour le cas test *cubes* avec  $\mu_{Str2/Str4} = 0.1$

Un second aménagement dû à la définition de ces fonctions objectif a été effectué. Comme expliqué précédemment (paragraphe 3.3 du chapitre 1), pour rechercher le minimum, l'enrichissement de l'échantillonnage se fait au point maximisant l'*expected improvement* (EI). Or, comme illustré sur la figure 2.9(b), la présence de plateau situé proche de la zone du minimum peut amener à une convergence lente de la stratégie EGO.

Afin d'éviter trop de points enrichis sur ces zones de plateau, une modification du critère d'enrichissement de la stratégie EGO a été effectuée et s'appuie sur le critère du *weighted expected improvement* (WEI) développé par [Sóbester *et al.*, 2005].

### 2.3.1 Modification de la stratégie EGO par *weighted expected improvement*

L'équation de l'*expected improvement* se compose de deux termes comme vu à l'équation 1.65. On trouve un terme servant à l'exploitation de la zone où se situe un minimum connu à savoir :

$$(y_{min} - \hat{y}(\mathbf{x}))\Phi\left(\frac{y_{min} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})}\right) \quad (2.11)$$

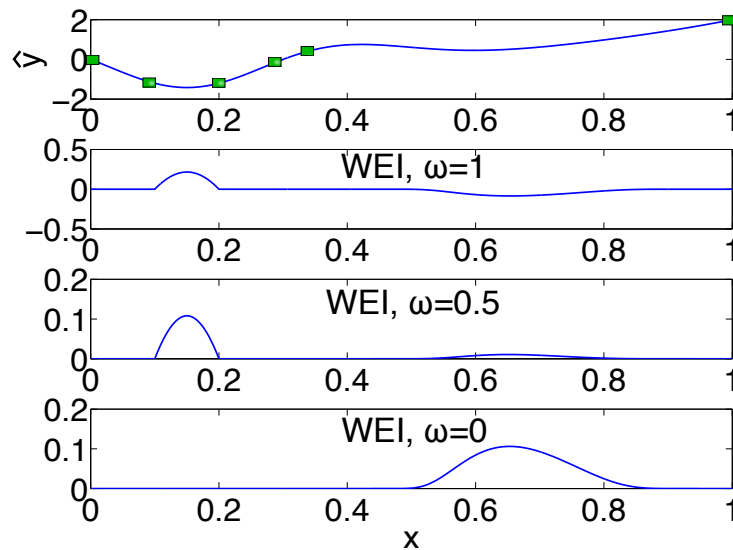
Le second terme permet quant à lui une exploration de l'espace de conception :

$$s_{\hat{y}}(\mathbf{x})\phi\left(\frac{y_{min} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})}\right) \quad (2.12)$$

Ainsi une formulation pondérée de l'*expected improvement* peut être écrite sous la forme de *weighted expected improvement* :

$$WEI(\mathbf{x}) = \omega(y_{min} - \hat{y}(\mathbf{x})) \Phi\left(\frac{y_{min} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})}\right) + (1 - \omega)s_{\hat{y}}(\mathbf{x})\phi\left(\frac{y_{min} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})}\right) \quad (2.13)$$

$\omega \in [0, 1]$  et l'on remarque que si  $\omega = 0.5$  alors  $WEI(\mathbf{x}) = 0.5EI(\mathbf{x})$ . En s'inspirant de l'exemple proposé par [Sóbester *et al.*, 2005], différentes valeurs de  $\omega$  sont testées pour illustrer le principe du *weighted expected improvement* (voir figure 2.10).



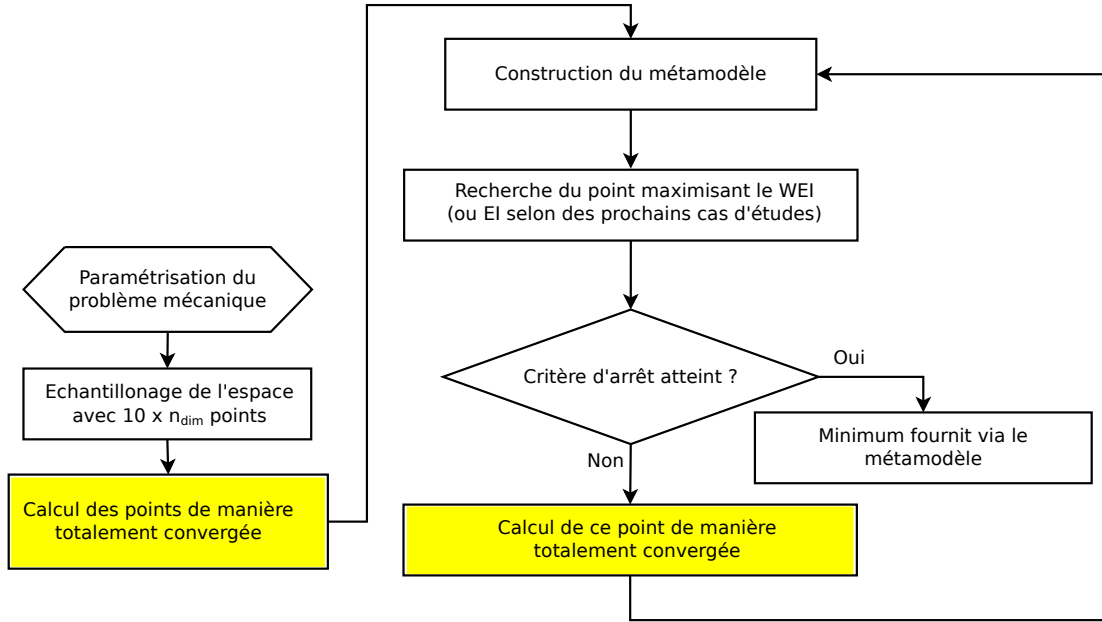
**FIGURE 2.10:** Illustration du WEI montrant l'impact du choix de  $\omega$

La stratégie peut alors être menée pour différents choix de valeurs  $\omega$  (choix cyclique par exemple pour alterner enrichissement par exploitation et exploration). [Forrester, 2004] propose un algorithme d'apprentissage par renforcement (des informations complémentaires peuvent être trouvées dans [Kaelbling *et al.*, 1996]) qui mène à un choix approprié de  $\omega$  à chaque itération de l'algorithme EGO. Dans notre étude, il a été choisi de prendre  $\omega = 0.2$ .

### 2.3.2 Définitions des gains

Tout comme pour la méthode de recalage, on s'intéresse au temps moyen mis pour effectuer les calculs ainsi qu'à la robustesse de la stratégie vis-à-vis de ces temps de calcul. La stratégie de recherche de minimum à l'aide de l'Evofusion pour différents couples (nombre initial de points ; niveau de convergence) de données partiellement convergées sera comparée à une méthode EGO dite de référence qui ne s'appuie uniquement que sur des données totalement convergées. Un métamodèle de krigeage est donc utilisé. Le tirage initial de points utilisé pour cette méthode de référence est un tirage LHS de  $10 \times n_{dim}$  comme préconisé par [Jones *et al.*, 1998] (voir figure 2.11). Le critère d'enrichissement est le maximum de WEI.





**FIGURE 2.11:** Algorithme associé à la méthode de référence de recherche de minimum de type EGO

Le temps de calcul avec utilisation de l'approche LaTin multiparamétrique associée à cette méthode de référence est défini par :

$$T_{EGO.ref.multi} = T_{fcv.multi}^{10 \times dim.ref} + T_{fcv.multi}^{n_{enri.ref}} \quad (2.14)$$

Le temps de calcul sans utilisation de l'approche laTin multiparamétrique est défini quant à lui par :

$$T_{EGO.ref.sans.multi} = T_{fcv.sans.multi}^{10 \times dim.ref} + T_{fcv.sans.multi}^{n_{enri.ref}} \quad (2.15)$$

Un premier gain est alors défini. Celui-ci **met en avant le rapport des temps moyens** obtenu par la stratégie EGO de référence et la stratégie de recherche par Evofusion.

$$Gain_{1.sans.multi} = \frac{\langle T_{EGO.ref.sans.multi} \rangle}{\langle T_{evo.sans.multi} \rangle} \quad (2.16)$$

$$Gain_{1.multi} = \frac{\langle T_{EGO.ref.multi} \rangle}{\langle T_{evo.multi} \rangle}$$

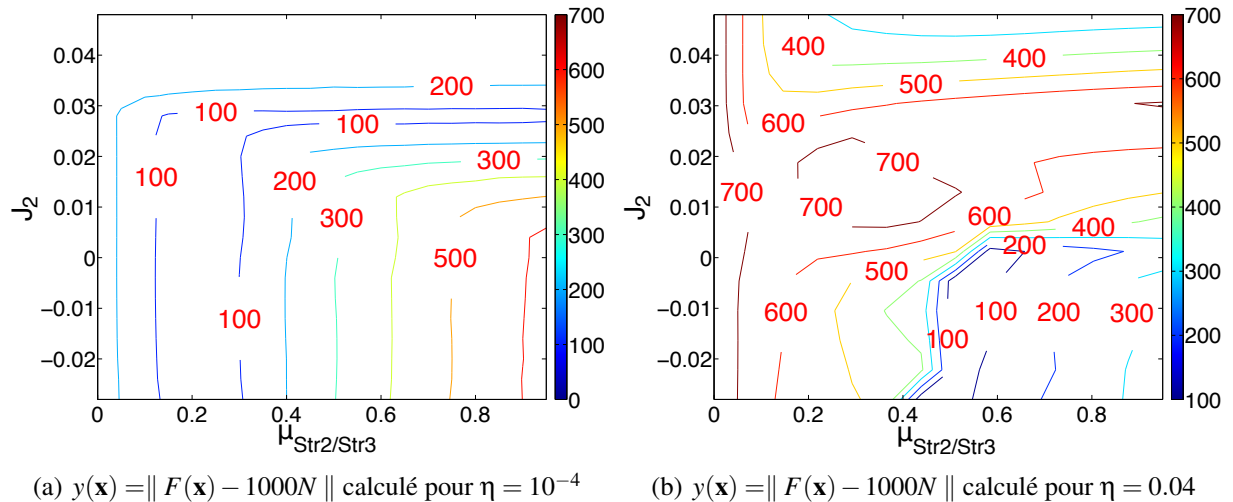
Le second gain présenté **met en avant la moyenne des gains** obtenus pour chaque test mené avec l'Evofusion par rapport au temps moyen de la stratégie de référence.

$$Gain_{2.sans.multi} = \left\langle \frac{\langle T_{EGO.ref.sans.multi} \rangle}{T_{evo.sans.multi}} \right\rangle \quad (2.17)$$

$$Gain_{2.multi} = \left\langle \frac{\langle T_{EGO.ref.multi} \rangle}{T_{evo.multi}} \right\rangle$$

**Critère d'arrêt :** L'enrichissement est stoppé quand le minimum du métamodèle construit se situe dans l'intervalle  $[0-0.01 \times \text{valeur cible}; 0+0.01 \times \text{valeur cible}]$ . La valeur cible pour le cas test *cubes* est 1000, pour le cas test *frette* est 15000.

### 2.3.3 Remarque



**FIGURE 2.12:** Comparaison entre la surface de réponse de référence et la surface de réponse construite avec des données partiellement convergées pour le cast test *cubes*

On précise que l'utilisation directe des données partiellement convergées sans stratégie de correction n'est pas envisagée. En effet, le modèle basse-fidélité peut être de trop mauvaise qualité et amener à un minimum erroné de la fonction objectif estimé. On illustre cette idée sur les figures 2.12, où, sur le cas test *cubes*, on impose  $\mu_{Str2/Str4} = 0.1$ . En figure 2.12(a), la fonction objectif  $y(\mathbf{x}) = \|F(\mathbf{x}) - 1000N\|$  est représentée pour le niveau de convergence de référence ( $\eta = 10^{-4}$ ) tandis qu'en figure 2.12(b), elle l'est pour le niveau partiellement convergé ( $\eta = 0.04$ ). On constate clairement que les zones situant le minimum sont différentes.

## 3 Traitement des cas test

### 3.1 Etude du cas test *cubes*

Dans cette section, on s'intéresse au traitement du premier cas test. Sur cet exemple, le nombre de paramètres définissant l'espace de conception s'élève à 3, on a donc  $n_{dim} = 3$ . Le nombre initial de points des données partiellement convergées varie de  $8 \times n_{dim}$  à  $16 \times n_{dim}$  (24 à 48 points) et le niveau de convergence de l'algorithme LaTin varie quant à lui de  $\eta = 0.09$  à  $\eta = 0.01$  par pas de 0.01 (temps de calcul CPU variant de 0.2s à 1.7s). Le niveau de référence est fixé à  $10^{-4}$  (temps calcul CPU 6.5s). De plus 20 tirages initiaux différents sont utilisés pour

chaque couple (nombre initial de points ; niveau de convergence), ce qui amène à effectuer au total 1620 études de cas (81 couples (nombre initial de points ; niveau de convergence)  $\times$  20 tirages LHS initiaux différents pour chaque couple testé). Un métamodèle de référence est construit à l'aide d'un plan factoriel complet de  $14^3 = 2744$  points.

### 3.1.1 Stratégie de recalage : cas test *cubes*

L'étude de la stratégie de recalage appliquée au cas test *cubes* est présentée au cours des prochains paragraphes en plusieurs étapes. On présente tout d'abord différents résultats.

- On commence à s'intéresser à la corrélation moyenne entre le métamodèle obtenu initialement avec les données uniquement **partiellement** convergées pour différents couples (nombre de points initiaux ; niveau de convergence) et le métamodèle de référence. Ceci a pour but d'évaluer la qualité des données partiellement convergées.
- Ensuite, les résultats de la stratégie de recalage sont présentés ; à savoir, les temps de calcul moyens et de robustesse indiquée par l'écart type relatif obtenu pour chaque couple (nombre de initial points ; niveau de convergence). **On rappelle que la stratégie de recalage est stoppée lorsque le coefficient de corrélation  $r_{ccc}$  est supérieur à 0.95 entre le métamodèle multi-fidélité obtenu par Evofusion et le métamodèle de référence.** On constatera notamment que les zones où la qualité du métamodèle initial de données partiellement convergées étaient globalement bonnes ou meilleures que d'autres, ne sont pas nécessairement les zones où les temps de calcul pour obtenir un métamodèle multi-fidélité de bonne qualité sont les plus intéressantes.
- Une interprétation de ces résultats sera alors proposée.

#### Qualité du métamodèle initial de données partiellement convergées

Un premier métamodèle est créé avec uniquement les données partiellement convergées. On présente sur la figure 2.13 la corrélation  $r_{ccc}$  moyenne entre ce métamodèle initial et le métamodèle de référence pour différents niveaux d'indicateur  $\eta$  et nombre de points. On peut alors constater clairement trois zones distinctes. La zone située entre les niveaux 0.09 et 0.08 est de faible corrélation. La zone située entre 0.08 à 0.05 est de meilleure corrélation. La dernière zone se situant entre 0.05 et 0.01 apporte une « bonne » corrélation  $r_{ccc}$  (environ égale à 0.8 pour les niveaux d'erreur 0.02 et 0.01). Ces résultats peuvent effectivement être attendus, l'indicateur global  $\eta$  décroissant, il peut sembler normal que la qualité du métamodèle augmente (ce n'est pas nécessairement toujours le cas, voir [Forrester *et al.*, 2006a], ou au paragraphe 3.1.2 à la figure 2.17)

#### Résultats de la stratégie de recalage

Les figures 2.14(a) et (b) montrent clairement que les zones où les temps de calcul moyens sont les plus faibles et donc les gains les plus importants (voir figure 2.15) se situent pour les zones où  $\eta = 0.08 - 0.05$  et  $\eta = 0.01$ .

Il est intéressant de constater que ce premier résultat est valable que l'approche soit menée avec la stratégie de calcul multiparamétrique ou non. On remarque également que d'autres zones

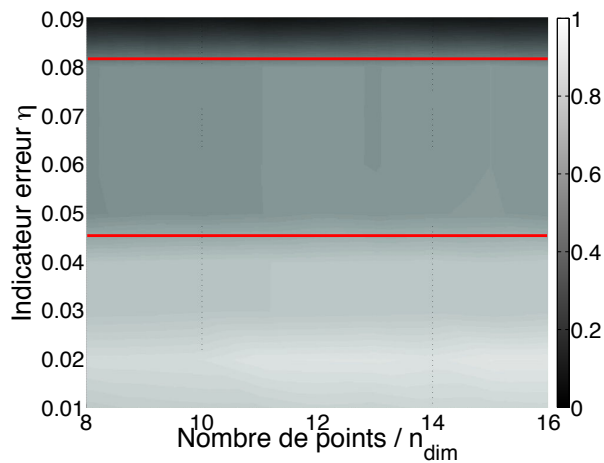


FIGURE 2.13: Corrélation  $r_{ccc}$  initiale entre le métamodèle construit avec des données partiellement convergées et le métamodèle de référence pour le cas test *cubes*

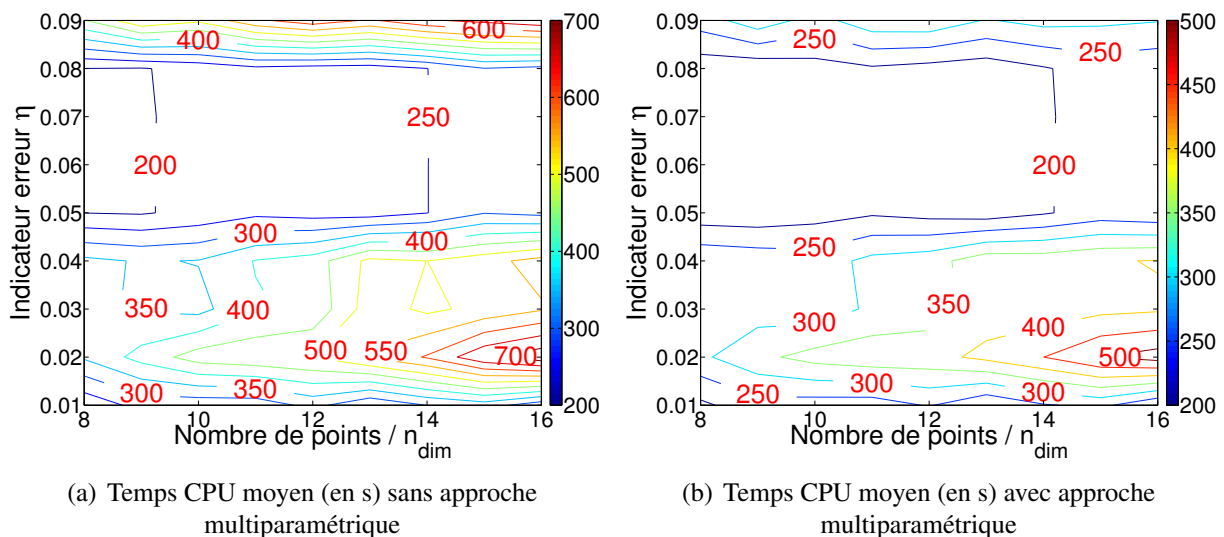
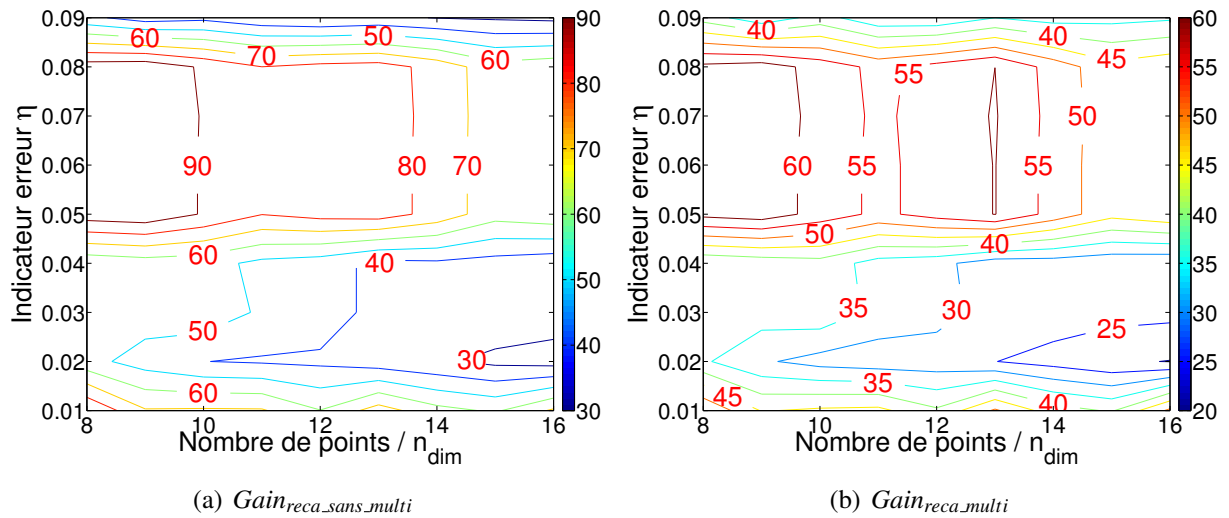
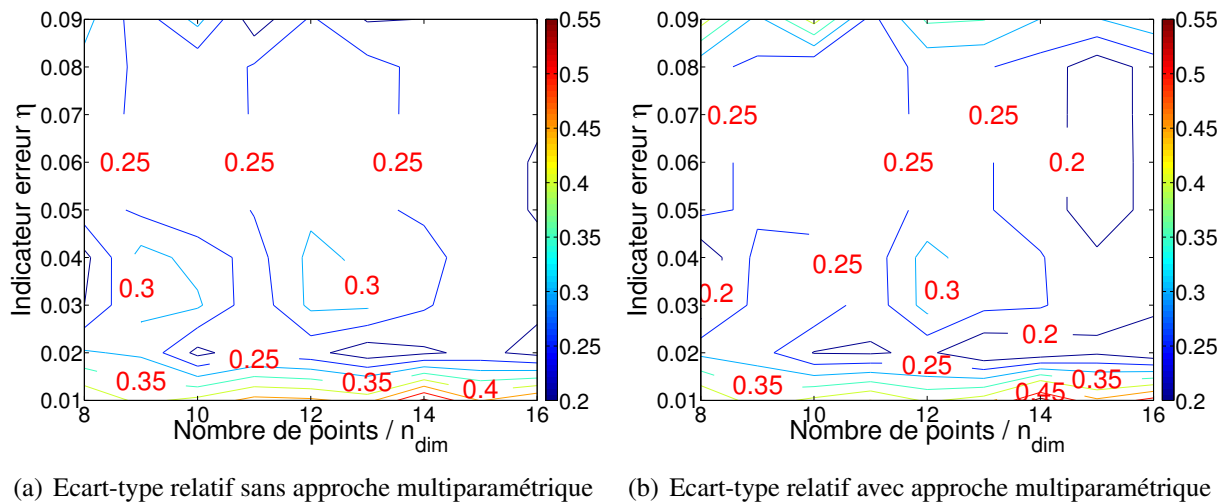


FIGURE 2.14: Temps de calcul moyen pour le cas test *cubes*

potentiellement intéressantes sur la figure 2.13, à savoir, les zones variant pour un indicateur d'erreur  $\eta$  allant de 0.04 à 0.02 ne sont en réalité pas si efficaces. Concernant le choix du nombre de points initiaux partiellement convergés à utiliser pour obtenir le plus rapidement possible un métamodèle multi-fidélité de qualité, celui-ci est optimal pour  $8 - 12 \times n_{dim}$  points.

Le second intérêt de l'étude est de vérifier si les couples (nombre initial de points ; niveau de convergence) fournissant les meilleurs résultats en termes de temps de calcul sont robustes ; c'est-à-dire qu'il y ait la dispersion des temps de calcul obtenus pour les 20 essais numériques de chaque couple la plus faible possible.

Les figures 2.16 montrent une certaine stabilité des résultats obtenus sur l'ensemble de la zone d'étude. En effet, l'écart-type relatif est relativement faible et spécialement dans les zones où les

FIGURE 2.15: Gain temps de calcul pour le cas test *cubes*FIGURE 2.16: Ecart-type relatif des temps de calcul pour le cas test *cubes*

gains de temps de calcul sont les plus intéressants, à savoir :  $8 - 12 \times n_{dim}$  points et indicateur  $\eta$  variant de 0.08 à 0.05.

### Interprétations des résultats

*A priori* certains résultats sont assez surprenants. En effet, la corrélation  $r_{ccc}$  entre le métamodèle initial (construit avec uniquement les premières données partiellement convergées) et le métamodèle de référence varie globalement de manière croissante en fonction de l'indicateur d'erreur  $\eta$ . On pourrait légitimement s'attendre à ce que par exemple pour le niveau d'indicateur d'erreur  $\eta = 0.02$  les temps de calcul soient meilleurs. Or, comme constaté sur les figures 2.14 et 2.15, ce n'est pas le cas.

L'explication vient du fait qu'il y a une plus grande dispersion des écarts entre le métamodèle de référence et certains métamodèles construits avec des données partiellement convergées pour différents niveaux de convergence  $\eta$ . Afin de quantifier cette dispersion, un nouveau critère,  $Q4$ , est introduit :

$$Q4 = \frac{\sigma_e}{\langle e \rangle} = \frac{\sigma_e}{Q3} \tag{2.18}$$

$\mathbf{e} = \{ e_1 \dots e_{n_v} \}$  avec  $n_v$  nombre de points de comparaison (sur ce cas test en l'occurrence 2744) et  $e_i$  défini à l'équation 1.61.

$Q4$  est donc l'écart-type relatif des erreurs normées  $e_i$ , c'est donc un indicateur de la dispersion des erreurs normées.

Le tableau récapitulatif 2.3 présente pour différents niveaux d'indicateur d'erreur  $\eta$  des indicateurs de qualité du métamodèle créé avec uniquement les données partiellement convergées par rapport au métamodèle de référence. Le nombre de points utilisé est de  $10 \times n_{dim}$ .

$\eta$	$r_{ccc}$	$r^2$	$Q1$	$Q3$	$Q4$
0.09	0.133	0.177	2.602	0.762	<b>0.778</b>
0.08 – 0.05	0.535	0.711	<b>0.302</b>	0.0793	0.905
0.04 – 0.03	0.716	<b>0.850</b>	0.639	0.0970	1.179
0.02	<b>0.813</b>	0.725	0.369	<b>0.0355</b>	1.425
0.01	0.759	0.813	0.363	0.0495	1

**TABLE 2.3:** Tableau récapitulatif des indicateurs de qualité de réponse

Le tableau 2.4 présente ensuite le nombre de points totalement convergés qu'il a fallu ajouter pour atteindre le critère d'arrêt.

$\eta$	0.09	0.08 – 0.05	0.04 – 0.03	0.02	0.01
Nbre pts ajoutés	78	27	38	50	27

**TABLE 2.4:** Nombre de points totalement convergés ajoutés en fonction de la précision des données partiellement convergées

Les résultats présentés au tableau 2.3 montrent clairement qu'on ne peut pas définir un niveau de convergence comme étant meilleur qu'un autre. En effet, les meilleurs indicateurs (en gras dans le tableau) ne sont pas situés pour les mêmes niveaux de convergence. Il n'y a donc pas de consensus selon ces indicateurs à définir un niveau de convergence comme étant le meilleur. Cependant, on peut expliquer la grande disparité des résultats décrits au tableau 2.4 suivant deux tendances :

- **La qualité initiale du métamodèle.** Un métamodèle construit avec des données partiellement convergées de très mauvaise qualité comme pour celles calculées avec  $\eta = 0.09$

(la qualité est environ 10 fois inférieure à celle obtenue pour les autres niveaux de convergence selon l'indicateur  $Q3$ ) amène à un enrichissement plus important pour obtenir un métamodèle évofusé de qualité.

- **La dispersion des écarts  $Q4$  entre le métamodèle construit avec les données partiellement convergées et le métamodèle de référence.** La méthode d'Evofusion étant une méthode de correction de surface, elle s'appuie sur un métamodèle d'erreur entre les données totalement et partiellement convergées. Le but de ce métamodèle d'erreur est de fournir une surface de réponse estimant les écarts entre le modèle basse fidélité et le modèle haute fidélité. Or, si la dispersion de ceux-ci est importante entre le métamodèle de référence (que l'on peut assimiler au modèle haute fidélité) et le métamodèle de données partiellement convergées, le métamodèle d'erreur devra contenir plus d'informations locales (c'est-à-dire plus de points) pour corriger cette dispersion. Si la dispersion des écarts est plus faible, un nombre moins important de points sera alors nécessaire. Un cas extrême étant par exemple un offset entre le modèle haute et basse fidélité.  $Q4$  est alors nul et l'évaluation d'un seul point sera nécessaire pour corriger le modèle basse fidélité. Sur notre cas test, le métamodèle initial calculé pour un niveau d'erreur  $\eta = 0.05$  a une qualité globale moindre que celui calculé pour  $\eta = 0.02$  (voir tableau 2.3). Cependant, comme l'indique  $Q4$ , la dispersion des écarts est plus faible, ce qui amène par la suite à un enrichissement moindre (voir tableau 2.4). Il faut cependant garder à l'esprit que l'ensemble de ces indicateurs sont fournis grâce au calcul d'un métamodèle de référence qui n'est *a priori* pas disponible dans un cadre industriel.

## Bilan

On distingue beaucoup d'informations et de résultats obtenus sur ce premier cas test. On a notamment pu remarquer qu'il existait bien des couples (nombre initial de points ; niveau de convergence) appropriés pour construire un métamodèle évofusé  $\hat{y}_{evo}$  de bonne qualité. On constate également que le paramètre le plus influent sur les performances atteintes est le niveau de convergence des données partiellement convergées. On peut néanmoins, sur ce cas test, pour la stratégie de recalage, initialiser la procédure avec un nombre de points de  $10 \times n_{dim}$  points partiellement convergés au niveau de convergence  $\eta = 0.05$ . La prochaine étude vise de nouveau à trouver des couples (nombre initial de points ; niveau de convergence) appropriés, mais dans le cadre d'une stratégie de recherche du minimum par EGO. On va ainsi vérifier si le couple précédemment proposé est également efficace dans ce contexte d'étude.

### 3.1.2 Stratégie de recherche du minimum : cas test *cubes*

Comme pour la stratégie de recalage, l'étude de la stratégie de recherche de minimum met en avant la qualité initiale du métamodèle des données partiellement convergées. Le critère de qualité choisit est  $r^2$ . **En effet, ce critère de corrélation permet de quantifier la qualité de forme d'une surface de réponse par rapport à une autre, ce qui est l'élément prépondérant pour localiser la zone du minimum.** Dans un second temps, les gains de calculs obtenus par la stratégie EGO réalisée à l'aide de l'Evofusion par rapport à la stratégie EGO de référence

(voir figure 2.11) sont présentés. Ces résultats mettront en exergue les couples (nombre initial de points : niveau de convergence) les plus performants à construire un métamodèle  $\hat{y}_{evo}$  dans le cadre de la stratégie de recherche du minimum. Pour chacun des couples étudiés, 20 tirages LHS initiaux sont effectués. La stratégie EGO servant de référence (effectuée avec uniquement des données totalement convergées) est également effectuée pour 20 tirages LHS de  $10 \times n_{dim} = 30$  points. Le critère d'arrêt des deux stratégies est atteint lorsque le minimum du métamodèle construit se situe entre  $[0 \pm 1\% \text{ valeur cible}]$  avec la valeur cible qui est ici de  $1000N$ .

### Qualité du métamodèle initial de données partiellement convergées

La figure 2.17 représente la corrélation moyenne  $r^2$  pour les 20 tirages initiaux différents effectués pour chaque couple (nombre de points ; niveau de convergence).

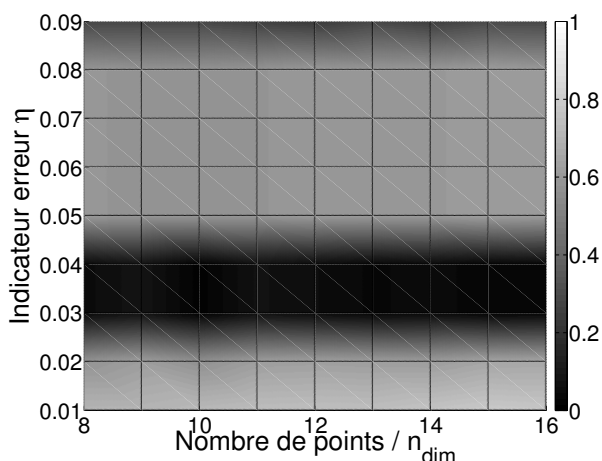


FIGURE 2.17: Corrélation  $r^2$  initiale

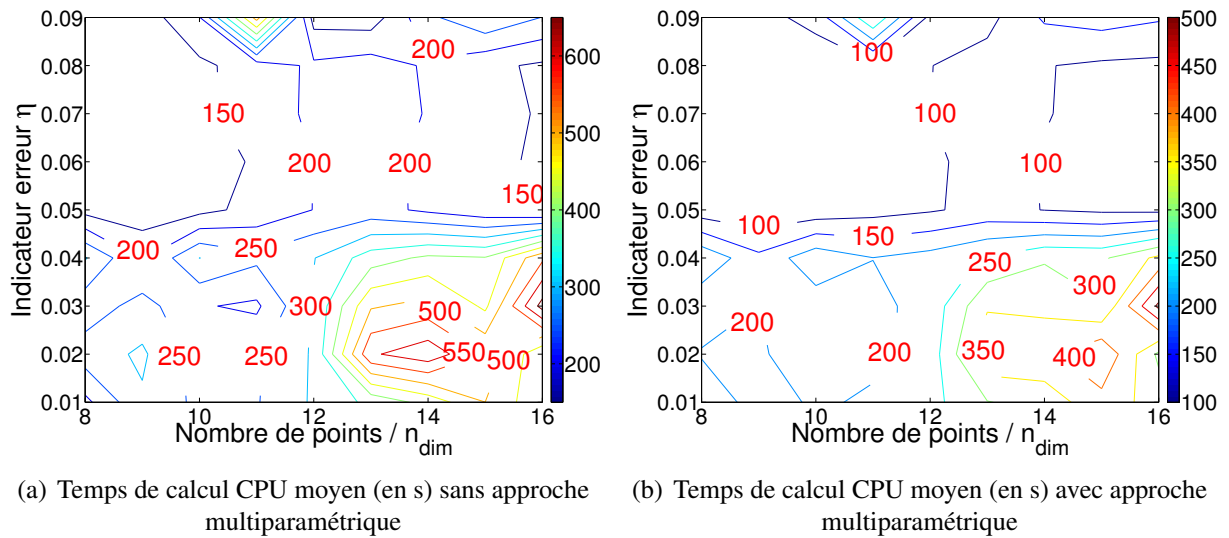
On distingue deux zones où l'indicateur  $r^2$  est élevé : la zone variant de  $\eta = 0.08$  à  $0.05$  et la zone variant de  $\eta = 0.02$  à  $0.01$ . On retrouve des zones de qualité communes avec ce qui a été fait lors de la stratégie de recalage. On constate ce qui a été mis en remarque précédemment sur le fait qu'augmenter la précision des calculs n'amène pas nécessairement à un métamodèle de meilleure qualité. En effet, la zone  $\eta = 0.04$  à  $0.03$  est de plus mauvaise qualité que la zone  $\eta = 0.08$  à  $0.05$ .

### Résultats de la stratégie de recherche de minimum

Les figures 2.18 confirment qu'effectivement pour les niveaux de convergence des données partiellement convergées  $\eta = 0.08$  à  $0.05$ , les temps de calcul moyens pour atteindre le critère d'arrêt sont les plus faibles. L'efficacité de la méthode n'est pas aussi bonne pour les autres zones.

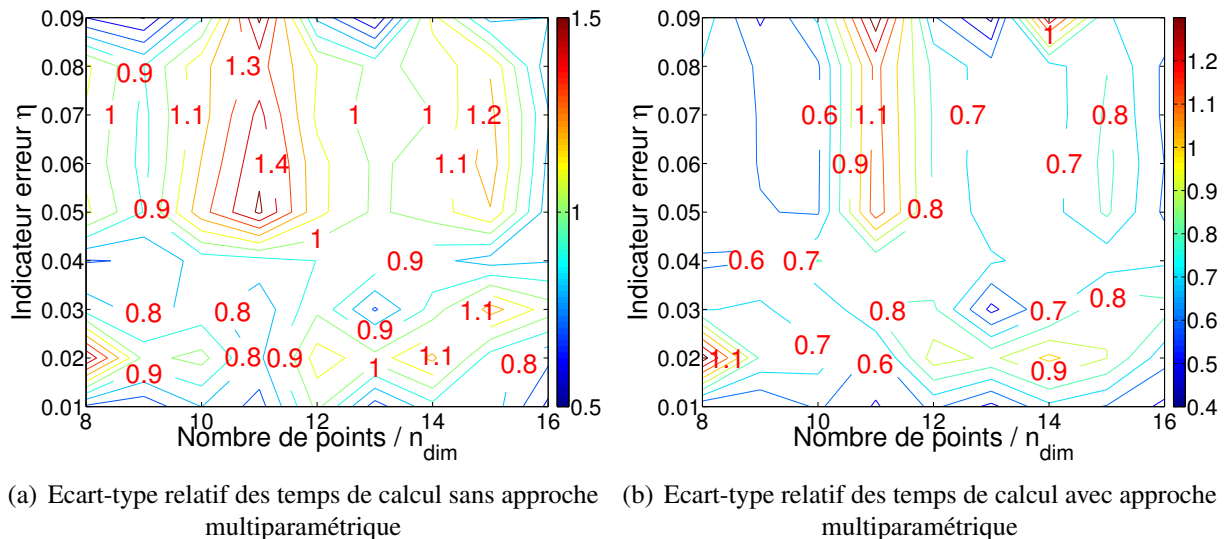
On constate aux figures 2.19 que la dispersion des temps de calcul obtenus sur les 20 essais numériques menés est assez grande. En effet, l'écart-type relatif est aux alentours de 1 pour une approche sans utilisation de la stratégie multiparamétrique. Les résultats sont légèrement plus





**FIGURE 2.18:** Temps de calcul moyen avec la stratégie d'Evofusion

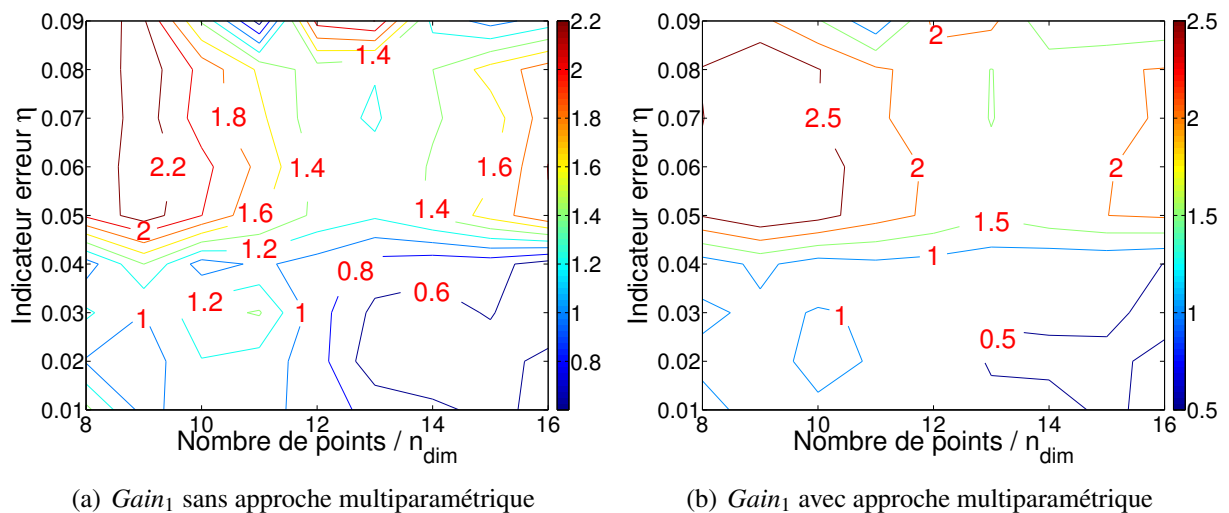
robustes avec cette utilisation. Ceci s'explique par le fait qu'il n'est pas rare qu'un point enrichi soit relativement proche d'un autre point précédemment calculé lors d'une stratégie de type EGO. L'approche multiparamétrique permet donc de calculer rapidement ce nouveau point enrichi, ce qui a pour effet de réduire le temps de calcul et d'impacter de manière moins significative le temps de calcul global moyen. Néanmoins, du point de vue de la robustesse, aucune zone d'étude ne s'avère être réellement meilleure qu'une autre.



**FIGURE 2.19:** Ecart-type relatif des temps de calcul pour le cas test *cubes*

Après avoir illustré les temps de calcul moyens (voir figure 2.18), il est intéressant de présenter les gains de temps de calcul obtenus entre l'approche de recherche du minimum par

Evofusion et l'approche EGO de référence. La figure 2.20 met en avant le  $Gain_1$  obtenu par la stratégie basée sur l'Evofusion (pour rappel, gain obtenu par le rapport des temps de calcul moyen des deux stratégies, voir équation 2.16). Les zones où les gains de temps sont importants sont situées au niveau de la zone d'indicateur d'erreur  $\eta$  variant de 0.08 à 0.05 pour un nombre de points de  $8 - 12 \times n_{dim}$ . La valeur du gain atteinte est alors 2.5 avec l'approche de calcul multiparamétrique. On constate cependant que l'Evofusion n'est pas nécessairement une stratégie gagnante. En effet, on constate que globalement la zone située pour  $\eta = 0.04 - 0.01$  avec un nombre de points initiaux de  $12 - 16 \times n_{dim}$  est une stratégie perdante par rapport à la stratégie EGO de référence.



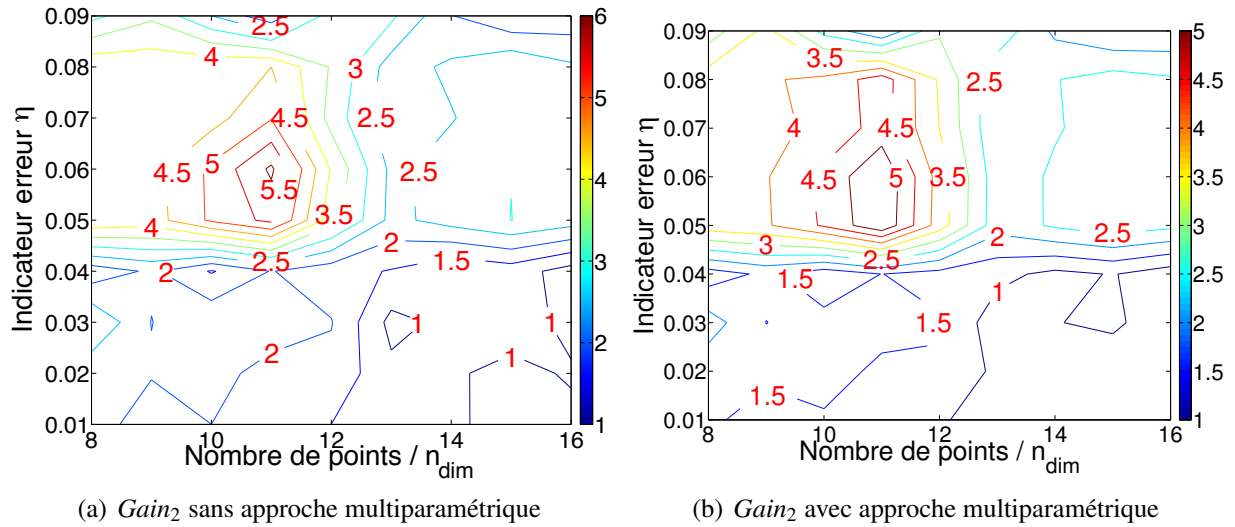
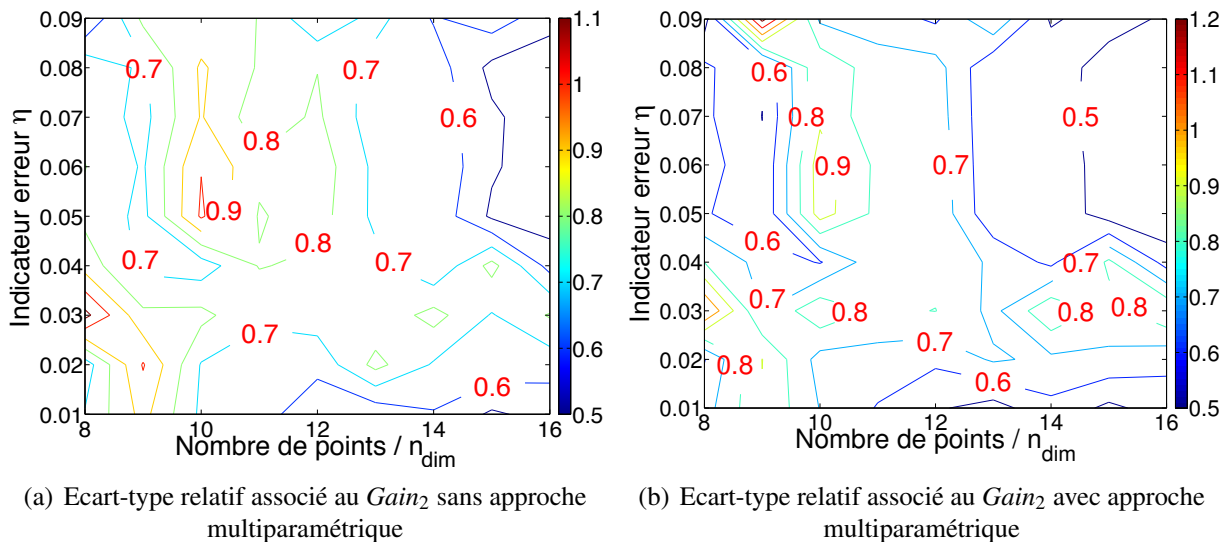
**FIGURE 2.20:**  $Gain_1$  de calcul

Le second gain défini comme  $Gain_2$  (voir équation 2.17) permet d'établir encore plus clairement ce résultat (voir figure 2.21). On constate que la moyenne des gains que l'on peut obtenir est assez élevée, que ce soit avec ou sans approche multiparamétrique. On remarque d'ailleurs que son emploi n'influence que faiblement les gains atteignables. L'Evofusion apporte donc des gains semblables que les calculs soient faits avec la stratégie multiparamétrique ou non.

L'écart-type relatif (et donc la robustesse) des résultats obtenus pour le  $Gain_2$  (présenté à la figure 2.22) montre de nouveau que la dispersion des résultats peut être assez importante. Cependant, les mesures de cet écart-type relatif montrent que globalement la robustesse de la méthode est sensiblement la même pour les différents couples (nombre initial de points : niveau de convergence) testés. (Excepté pour un nombre de points de  $14 - 16 \times dim$  avec un niveau de l'indicateur d'erreur  $\eta$  allant de 0.08 à 0.05, mais cette zone n'est pas intéressante en terme de temps de calcul).

### Bilan des études sur le cas test *cubes*

On a clairement pu constater lors des mises en place des stratégies de recalage et de recherche du minimum qu'un couple (nombre initial de points ; niveau de convergence) de données partiellement convergées apparaissait de façon récurrente sur ces deux études. Ainsi, le couple

FIGURE 2.21:  $Gain_2$  de calculFIGURE 2.22: Ecart-type relatif associé au  $Gain_2$  pour les approches de calcul multiparamétrique et sans multiparamétrique

$(10 \times n_{dim} ; \eta = 0.05)$  est efficace quant à fournir des données partiellement convergées de suffisamment bonne qualité, peu coûteuses à évaluer et permettant de mener à bien rapidement les deux stratégies proposées. De plus, on remarque que la stratégie d'Evofusion peut être une stratégie intéressante au niveau des gains de temps envisageables si ce couple (nombre initial de points ; niveau de convergence) a effectivement été correctement choisi. On a pu constater que le paramètre le plus influent sur les performances de calcul atteintes est le niveau de convergence  $\eta$  des données partiellement convergées.

Les études menées sur le second cas test visent à confirmer ces résultats.

## 3.2 Etude du cas test *frette*

Les études de la stratégie de recalage et de recherche du minimum menées sur cet exemple industriel ont pour but de valider le fait que les zones (nombre initial de points ; niveau de convergence) des données partiellement convergées qui ont été précédemment trouvées sont également efficaces sur ce cas test. Ce cas test de type industriel se compose de 5 paramètres ( $n_{dim} = 5$ ). Le nombre initial de points varie de  $8 \times n_{dim}$  à  $16 \times n_{dim}$  (40 à 80 points) et le niveau de convergence de l'algorithme LaTIn varie quant à lui de 0.07 à 0.01 par pas de 0.01 (temps de calcul CPU variant de 0.25s à 1.4s). Le niveau de référence pour ce cas test est fixé à  $10^{-5}$  (86s de temps de calcul CPU). Pour limiter les temps d'étude, 10 tirages initiaux différents sont utilisés pour chaque couple (nombre de points ; niveau de convergence). Ainsi, au total 630 essais numériques sont effectués, aussi bien pour la stratégie de recalage que de recherche du minimum. Dans le but d'alléger la présentation, seuls les résultats clés dans le cadre de calcul avec approche multiparamétrique seront présentés. Le métamodèle de référence est calculé avec 2000 points issus d'un tirage LHS. En effet, la complexité de la surface de réponse s'avère être faible et nécessite peu de points pour être construite (elle est composée de quelques hyperplans).

### 3.2.1 Stratégie de recalage : cas test *frette*

#### Qualité du métamodèle initial de données partiellement convergées

Comme pour le cas précédent, on s'intéresse à la qualité du métamodèle initial, c'est-à-dire celui construit avec uniquement les données partiellement convergées. La figure 2.23 montre que quel que soit le nombre de points et le niveau de convergence de l'algorithme laTIn pour des niveaux allant de 0.07 à 0.02, la corrélation moyenne  $r_{ccc}$  est extrêmement faible. Cependant, dans le prochain paragraphe on démontre que ces zones sont néanmoins efficaces pour construire rapidement un métamodèle  $\hat{y}_{evo}$  de bonne qualité.

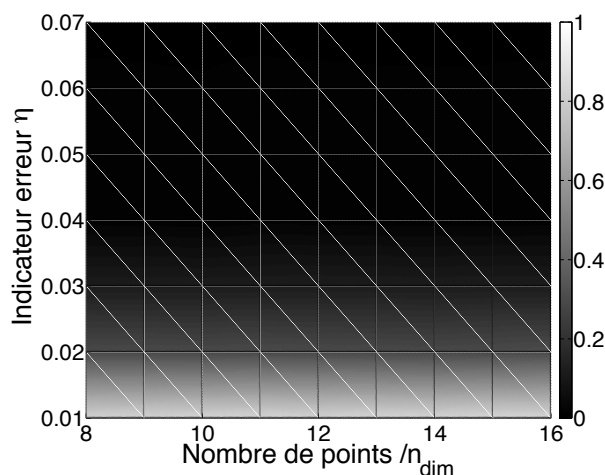


FIGURE 2.23: Corrélation  $r_{ccc}$  entre le métamodèle initial et le métamodèle de référence

### Résultats de la stratégie de recalage

Sur les figures 2.24, on montre que le couple (nombre initial de points ; niveau de convergence) =  $(10 \times n_{dim} ; \eta = 0.05)$  est un couple efficace quant à construire après enrichissement un métamodèle multi-fidélité de bonne qualité. On constate qu'il s'agit même d'un des couples les plus performants à utiliser en terme de temps de calcul mais également en terme de robustesse.

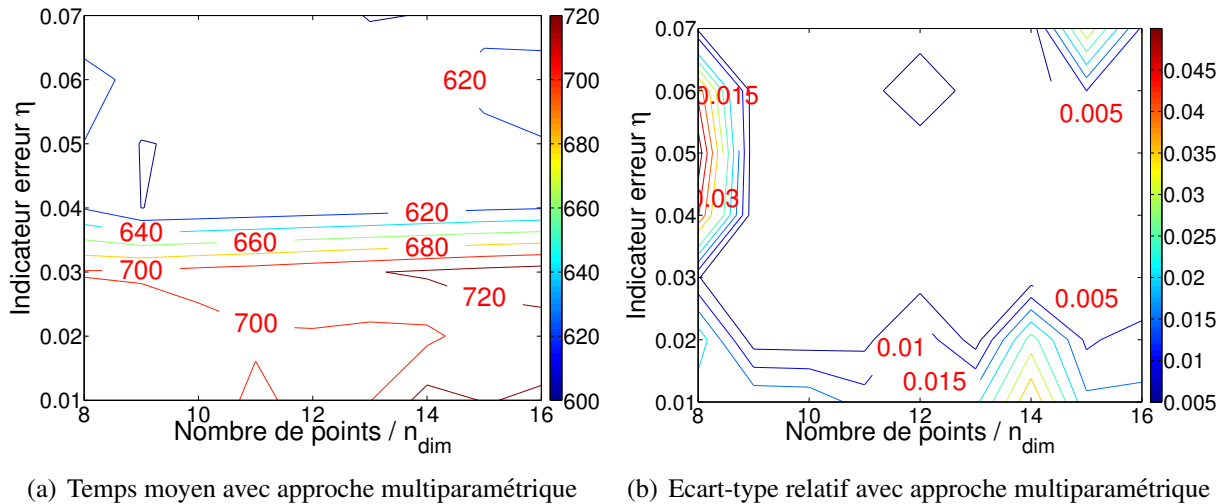


FIGURE 2.24: Résultats de la stratégie de recalage appliquée au cas test *frette*

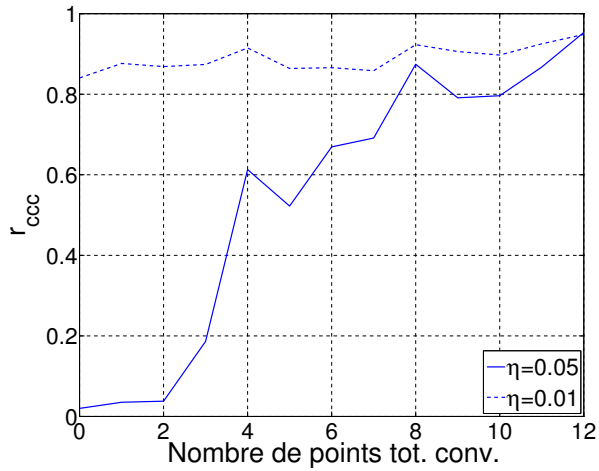
On remarque sur la figure 2.24(b) que la dispersion des résultats est extrêmement faible. Ceci s'explique par le fait que peu de points totalement convergés sont nécessaires pour obtenir rapidement un métamodèle de qualité. Ce nombre de points nécessaires pour atteindre le critère d'arrêt varie entre 12 et 14 et ce, pour tous les niveaux de convergence des données partiellement convergées. L'explication tient au fait que le métamodèle obtenu à l'aide des données partiellement convergées est globalement un hyperplan de valeur nulle. Aucune information n'est donc réellement fournie par celui-ci.

Le fait que la complexité de la surface de réponse associée à ce problème soit faible explique que peu de points suffit à la description de la fonction étudiée. On peut alors expliquer la croissance du temps de calcul de la stratégie de recalage en fonction de  $\eta$ , globalement observée à la figure 2.24(a), par l'unique fait de la croissance du temps de calcul des données partiellement convergées en fonction de  $\eta$ .

Cependant, l'utilisation de données partiellement convergées de qualité (comme pour  $\eta = 0.01$  ici) n'est pas nécessairement vaine. La figure 2.25 montre que la qualité moyenne du métamodèle  $\hat{y}_{evo}$ , comparée pour deux niveaux de convergence  $\eta = 0.05$  et  $\eta = 0.01$  est supérieure avec des données partiellement convergées au niveau  $\eta = 0.01$ .

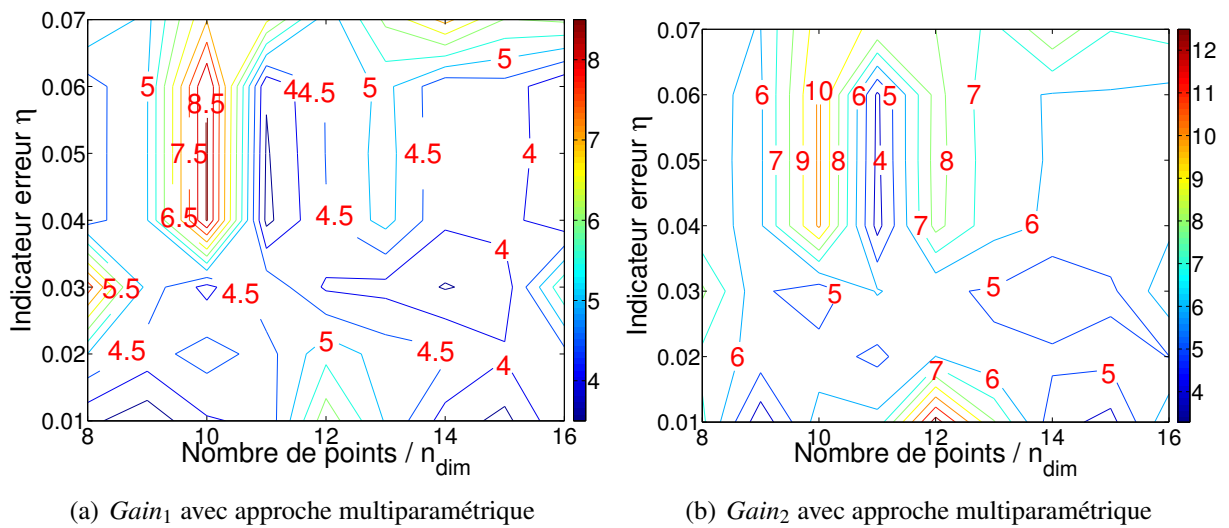
### 3.2.2 Stratégie de recherche du minimum : cas test *frette*

Les résultats de la stratégie de recherche du minimum par approche EGO appliquée à l'Evofusion comparée à la méthode de référence montrent clairement que l'approche par Evofusion



**FIGURE 2.25:** Evolution de la corrélation  $r_{ccc}$  pour 50 points initiaux partiellement convergés

amène à des gains de temps de calcul relativement importants sur ce cas test industriel (8.5 pour  $Gain_1$  et 10 pour  $Gain_2$ ). Les figures 2.26 représentent les  $Gain_1$  et  $Gain_2$  obtenus sur cet exemple (les calculs ont été effectués à l’aide de l’approche multiparamétrique mais la même forme de résultats est obtenue sans son utilisation).



**FIGURE 2.26:** Gains de calcul

On remarque de nouveau que l’on atteint des gains de temps importants pour l’approche par Evofusion avec des données partiellement convergées au niveau  $\eta = 0.05$  et pour un nombre de points initial de  $10 \times n_{dim}$ .

## 4 Bilan de l'étude de faisabilité

Cette étude de faisabilité a mis en avant que l'on peut déterminer des couples (nombre initial de points ; niveau de convergence) de données partiellement convergées amenant à construire un métamodèle multi-fidélité capable de répondre aux deux problématiques étudiées. Celles-ci étaient la construction d'un métamodèle de qualité pouvant servir d'abaque virtuel (obtenu à l'aide de la stratégie de recalage) et la recherche de minimum de la fonction objectif (obtenu à l'aide de l'approche EGO). La méthode d'Evofusion utilisée pour répondre à ces deux problématiques s'est avérée particulièrement efficace pour un nombre initial de points partiellement convergés de  $10 \times n_{dim}$  et pour un niveau de convergence de l'algorithme LaTIn  $\eta = 0.05$ . On a cependant pu constater, notamment sur le cas test *cubes*, que le paramètre le plus influent quant à l'efficacité des résultats obtenus est le niveau de convergence des données partiellement convergées. Les temps de calcul nécessaires pour atteindre les critères d'arrêt des algorithmes peuvent alors être assez disparates d'un niveau de convergence à l'autre. Une autre conclusion essentielle de cette étude est que l'Evofusion est une approche pouvant permettre des gains de calcul intéressants par rapport à l'approche de référence ne s'appuyant que sur des données totalement convergées.

Le nombre de points initiaux de  $10 \times n_{dim}$  est un résultat intéressant puisqu'il correspond à la règle heuristique proposée dans [Jones *et al.*, 1998] lors de la mise en place de la stratégie EGO. Il s'agit également du nombre initial de points utilisé dans les travaux de [Huang *et al.*, 2006a] qui s'appuient sur le cokrigeage multi-fidélité. Dans le cadre de l'Evofusion [Forrester *et al.*, 2006a] a utilisé sur un cas à deux paramètres 26 points partiellement convergés. Les résultats obtenus lors de cette étude sont donc relativement cohérents avec ce que l'on peut trouver dans la littérature.

Le prochain chapitre visera à trouver quelles méthodes de construction de métamodèle multi-fidélité sont les plus adaptées à nos problèmes d'assemblages mécaniques. On comparera alors la méthode d'Evofusion (méthode de correction de surface), trois méthodes de cokrigeage et le krigage hiérarchique. Le nombre de points partiellement convergés initialement calculés pour la construction de ces métamodèles sera donc fixé à  $10 \times n_{dim}$ . On continuera cependant à s'intéresser à l'influence du niveau de convergence de ces données partiellement convergées pour les différentes méthodes mises en place. En effet, ayant constaté la disparité, pouvant être importante, des résultats en fonction de ce paramètre, il est intéressant d'aborder son impact pour les différentes méthodes multi-fidélité utilisées.

## Chapitre 3

# Comparaison de métamodèles multi-fidélité pour l'approximation globale

*Dans ce troisième chapitre, on se propose de comparer différentes méthodes de construction de métamodèle. Ces méthodes de construction sont notamment comparées à l'aide de la stratégie dite de recalage (voir Chapitre 2) utilisée pour construire un abaque virtuel de qualité. 5 méthodes sont comparées : Evofusion, 3 approches de cokrigeage, krigeage hiérarchique.*

### Sommaire

---

<b>1</b>	<b>Différentes stratégies de modélisation multi-fidélité . . . . .</b>	<b>104</b>
1.1	Equations générales du cokrigeage . . . . .	105
1.2	Méthodes de cokrigeage . . . . .	110
1.3	Krigeage hiérarchique . . . . .	118
1.4	Illustration des stratégies multi-fidélité . . . . .	119
<b>2</b>	<b>Etudes comparatives sur des exemples mécaniques . . . . .</b>	<b>123</b>
2.1	Introduction . . . . .	123
2.2	Choix de points totalement convergés . . . . .	124
2.3	Stratégie de recalage avec cokrigeage et krigeage hiérarchique . . . . .	126
2.4	Etude du cas test <i>cubes</i> . . . . .	128
2.5	Etude du cas test <i>1_rainure</i> . . . . .	139
<b>3</b>	<b>Bilan de l'étude comparative des méthodes de construction de métamodèles . .</b>	<b>147</b>

---



Dans ce chapitre, on s'attache à comparer différentes approches multi-fidélité afin de répertorier les plus efficaces dans le cadre des assemblages mécaniques. Ces approches sont testées sur la stratégie de recalage en vue de construire un métamodèle pouvant servir d'abaque virtuel. Celui-ci doit donc être précis tout en étant établi pour un temps de calcul limité. Au cours du chapitre 1, différentes approches multi-fidélité ont été brièvement introduites. On s'attache ici à utiliser les méthodes suivantes :

- l'Evofusion (méthode de correction de surface)
- Cokrigeage multi-fidélité (3 approches sont utilisées dans ce chapitre)
- Le krigeage hiérarchique

Ces différentes approches ont alors été implémentées au sein de la toolbox GREMAT [Laurent, 2013], dans un but de pérennisation des développements. Afin de déterminer quelles stratégies sont les plus performantes, celles-ci sont testées sur deux cas test mécaniques. **On s'attache à comparer des méthodes proposées dans la littérature.** Ainsi certaines variantes ne sont pas testées dans ce chapitre (par exemple, on ne s'intéressera pas au cas où les données totalement convergées ne sont pas calculées partiellement convergées). En effet, la combinatoire des possibilités à tester serait alors trop grande.

Différentes problématiques se posent également. Les approches de cokrigeage et de krigeage hiérarchique nécessitent des données basse fidélité mais également haute fidélité pour pouvoir construire un premier métamodèle. Celui-ci doit donc contenir ces deux sources d'information. On a vu au cours du chapitre précédent que l'on pouvait choisir comme premier tirage  $10 \times n_{dim}$  points LHS partiellement convergés. Mais qu'en est-il du nombre de points totalement convergés à calculer pour construire un premier métamodèle multi-fidélité ? Il se pose également la question de savoir comment obtenir ces points. Cet enjeu est également abordé au cours de ce chapitre.

Dans un premier temps, les équations des méthodes utilisées sont rappelées. On s'intéresse par la suite à la problématique du choix des points totalement convergés pour construire les métamodèles de cokrigeage et de krigeage hiérarchique. Les différents aspects de l'étude comparative des méthodes sont ensuite présentés. Finalement, les résultats de chacune des méthodes sont présentés sur les cas test envisagés. Certains développements sont également présentés dans [Courrier *et al.*, 2015].

## 1 Différentes stratégies de modélisation multi-fidélité

Dans cette section les équations utiles au cokrigeage et au krigeage hiérarchique sont présentées. Le cokrigeage est un moyen stochastique (comme le krigeage) d'obtenir un métamodèle que l'on souhaite représentatif du modèle haute fidélité en tenant compte d'informations auxiliaires. Ces informations auxiliaires sont, dans ce cadre d'étude, les données basse fidélité. L'enjeu pour les méthodes de cokrigeage est donc de trouver le moyen approprié de quantifier la corrélation entre les données haute fidélité et basse fidélité. De ce fait, différentes approches ou variantes du cokrigeage existent. Trois d'entre elles sont alors testées au cours de ce chapitre.

## 1.1 Equations générales du cokrigeage

On présente dans ce paragraphe les équations du cokrigeage. Ces équations ne sont écrites que pour deux niveaux de fidélité. L'écriture de celles-ci peut cependant se généraliser à plusieurs niveaux de fidélité.

On note  $\mathbf{X}_{pcv} = \left[ \mathbf{x}_{pcv}^{(1)} \dots \mathbf{x}_{pcv}^{(n_{pcv})} \right]^T$  les points du plan d'expérience calculés partiellement convergés. On note  $\mathbf{y}_{pcv} = \left[ y(\mathbf{x}_{pcv}^{(1)}) \dots y(\mathbf{x}_{pcv}^{(n_{pcv})}) \right]^T$  les évaluations de ces différents points. On notera de même  $\mathbf{Y}_{pcv} = \left[ Y(\mathbf{x}_{pcv}^{(1)}) \dots Y(\mathbf{x}_{pcv}^{(n_{pcv})}) \right]^T$  les évaluations de ces points du processus gaussien associé au modèle basse fidélité. De même on note,  $\mathbf{X}_{fcv} = \left[ \mathbf{x}_{fcv}^{(1)} \dots \mathbf{x}_{fcv}^{(n_{fcv})} \right]^T$  les points du plan d'expérience à évaluer de manière totalement convergés. On notera de même  $\mathbf{y}_{fcv}$  et  $\mathbf{Y}_{fcv}$ . On utilise bien sûr le cas où  $n_{pcv} > n_{fcv}$ .

Ayant deux évaluations différentes, deux processus gaussiens sont donc définis (on utilise une approche de type universelle) :

$$\begin{aligned} Y_{fcv}(\mathbf{x}) &= \mathbf{f}_{fcv}^T(\mathbf{x})\boldsymbol{\beta}_{fcv} + Z_{fcv}(\mathbf{x}) \\ Y_{pcv}(\mathbf{x}) &= \mathbf{f}_{pcv}^T(\mathbf{x})\boldsymbol{\beta}_{pcv} + Z_{pcv}(\mathbf{x}) \end{aligned} \quad (3.1)$$

Comme pour le krigeage nous disposons des propriétés suivantes :

$$\begin{aligned} E[Y_{fcv}(\mathbf{x})] &= \mathbf{f}_{fcv}^T(\mathbf{x})\boldsymbol{\beta}_{fcv} \\ E[Y_{pcv}(\mathbf{x})] &= \mathbf{f}_{pcv}^T(\mathbf{x})\boldsymbol{\beta}_{pcv} \end{aligned} \quad (3.2)$$

$$\begin{aligned} E[Y_{fcv}(\mathbf{x})^2] &= \text{Var}[Y_{fcv}(\mathbf{x})] = \text{Var}[Z_{fcv}(\mathbf{x})] = \sigma_{fcv}^2 \\ E[Y_{pcv}(\mathbf{x})^2] &= \text{Var}[Y_{pcv}(\mathbf{x})] = \text{Var}[Z_{pcv}(\mathbf{x})] = \sigma_{pcv}^2 \end{aligned} \quad (3.3)$$

On peut donc également écrire les matrices suivantes :

$$\mathbf{F}_{pcv} = \left[ \mathbf{f}_{pcv}(\mathbf{x}^{(1)}) \dots \mathbf{f}_{pcv}(\mathbf{x}^{(n_{pcv})}) \right]^T \in \mathbb{R}^{n_{pcv} \times p} \quad (3.4)$$

et

$$\mathbf{F}_{fcv} = \left[ \mathbf{f}_{fcv}(\mathbf{x}^{(1)}) \dots \mathbf{f}_{fcv}(\mathbf{x}^{(n_{fcv})}) \right]^T \in \mathbb{R}^{n_{fcv} \times p} \quad (3.5)$$

Pour rappel,  $p$  correspond au nombre de paramètres de régression. Si la régression est d'ordre 0, alors  $p = 1$ , on parle de cokrigeage ordinaire. Dans ce travail, il a été choisi d'utiliser les mêmes fonctions de régression au niveau basse et haute fidélité (comme usuellement fait dans la littérature).

On cherche à construire un estimateur,  $\hat{Y}^m$ , du processus gaussien  $Y_{fcv}$  sous la forme d'un estimateur linéaire tirant parti des deux sources d'information.

$$\hat{Y}^m(\mathbf{x}) = \mathbf{w}_{fcv}^T(\mathbf{x})\mathbf{Y}_{fcv} + \mathbf{w}_{pcv}^T(\mathbf{x})\mathbf{Y}_{pcv} \quad (3.6)$$

**Remarque :** L'exposant  $m$  de  $\hat{Y}^m$  est utilisé pour signifier multi-fidélité.

Les vecteurs inconnus sont  $\mathbf{w}_{fcv}$  et  $\mathbf{w}_{pcv}$  et tout comme pour le krigeage, on recherche cet estimateur sous la forme du meilleur estimateur linéaire non biaisé.

Dans la suite du développement mathématique, on allège les notations en posant  $y_{\bullet}(\mathbf{x}) = y_{\bullet}$ ,  $Y_{\bullet}(\mathbf{x}) = Y_{\bullet}$ ,  $Z_{\bullet}(\mathbf{x}) = Z_{\bullet}$  et  $\mathbf{w}_{\bullet}(\mathbf{x}) = \mathbf{w}_{\bullet}$ .

De la même manière que pour le krigeage, on cherche à exprimer les conditions de non-biais de l'estimateur  $\hat{Y}^m$  à savoir  $E[\hat{Y}^m(\mathbf{x}) - Y_{fcv}(\mathbf{x})] = 0$  :

$$\begin{aligned}
E[\hat{Y}^m(\mathbf{x}) - Y_{fcv}(\mathbf{x})] &= 0 \\
\iff E[\mathbf{w}_{fcv}^T(\mathbf{x})\mathbf{Y}_{fcv} + \mathbf{w}_{pcv}^T(\mathbf{x})\mathbf{Y}_{pcv} - Y_{fcv}(\mathbf{x})] &= 0 \\
\iff E[\mathbf{w}_{fcv}^T(\mathbf{F}_{fcv}\boldsymbol{\beta}_{fcv} + \mathbf{Z}_{fcv}) + \mathbf{w}_{pcv}^T(\mathbf{F}_{pcv}\boldsymbol{\beta}_{pcv} + \mathbf{Z}_{pcv}) - (\mathbf{f}_{fcv}^T\boldsymbol{\beta}_{fcv} + z_{fcv})] &= 0 \quad (3.7) \\
\iff E[\mathbf{w}_{fcv}^T\mathbf{Z}_{fcv} + \mathbf{w}_{pcv}^T\mathbf{Z}_{pcv} - z_{fcv} + (\mathbf{F}_{fcv}^T\mathbf{w}_{fcv} - \mathbf{f}_{fcv}^T)\boldsymbol{\beta}_{fcv} + \mathbf{w}_{pcv}^T\mathbf{F}_{pcv}\boldsymbol{\beta}_{pcv}] &= 0 \\
\iff (\mathbf{F}_{fcv}^T\mathbf{w}_{fcv} - \mathbf{f}_{fcv}^T)\boldsymbol{\beta}_{fcv} + \mathbf{w}_{pcv}^T\mathbf{F}_{pcv}\boldsymbol{\beta}_{pcv} &= 0
\end{aligned}$$

De l'équation 3.7, on peut en tirer les deux équations 3.8 et 3.9. En effet, l'équation 3.7 doit être vérifiée quelles que soient les valeurs de  $\boldsymbol{\beta}_{fcv}$  et de  $\boldsymbol{\beta}_{pcv}$ .

$$\mathbf{F}_{fcv}^T\mathbf{w}_{fcv} - \mathbf{f}_{fcv}^T = \mathbf{0} \quad (3.8)$$

$$\mathbf{w}_{pcv}^T\mathbf{F}_{pcv} = \mathbf{0} \quad (3.9)$$

Comme pour le krigeage, on exprime l'erreur quadratique moyenne (*MSE*) :

$$\begin{aligned}
MSE &= E\left[\left(\hat{Y}^m(\mathbf{x}) - Y_{fcv}(\mathbf{x})\right)^2\right] \\
&= E\left[\left(\mathbf{w}_{fcv}^T\mathbf{Z}_{fcv} + \mathbf{w}_{pcv}^T\mathbf{Z}_{pcv} - z_{fcv}\right)^2\right] \\
&= \sigma_{fcv}^2 + \mathbf{w}_{fcv}^T\mathbf{C}^{fcv}\mathbf{w}_{fcv} + \mathbf{w}_{pcv}^T\mathbf{C}^{pcv}\mathbf{w}_{pcv} + 2\mathbf{w}_{fcv}^T\mathbf{C}^{f-pcv}\mathbf{w}_{pcv} \\
&\quad - 2\mathbf{w}_{fcv}^T\mathbf{c}_{fcv} - 2\mathbf{w}_{pcv}^T\mathbf{c}_{f-pcv}
\end{aligned} \quad (3.10)$$

Les matrices  $\mathbf{C}^{fcv}$  et  $\mathbf{C}^{pcv}$  font respectivement référence aux matrices de covariance des données totalement et partiellement convergées. La matrice  $\mathbf{C}^{f-pcv}$  est quant à elle la matrice d'intercovariance des données totalement et partiellement convergées.

On écrit également  $\mathbf{c}_{fcv}(\mathbf{x}) = \left[ \text{Cov}(\mathbf{x}_{fcv}^{(1)}, \mathbf{x}) \quad \dots \quad \text{Cov}(\mathbf{x}_{fcv}^{(n_{fcv})}, \mathbf{x}) \right]^T$  le vecteur de covariance entre une donnée  $\mathbf{x}$  à évaluer et les données totalement convergées.  $\mathbf{c}_{f-pcv}(\mathbf{x}) = \left[ \text{Cov}(\mathbf{x}_{pcv}^{(1)}, \mathbf{x}) \quad \dots \quad \text{Cov}(\mathbf{x}_{pcv}^{(n_{pcv})}, \mathbf{x}) \right]^T$  est le vecteur calculant la covariance entre une donnée  $\mathbf{x}$  et les données partiellement convergées.

Dans le but de trouver le meilleur estimateur linéaire non biaisé (minimisation de MSE et respect des conditions de non-biais), un lagrangien est alors introduit. Par rapport au krigeage, deux conditions de non-biais sont définies. Ce lagrangien est défini à l'équation 3.11

$$\mathcal{L}(\mathbf{w}_{fcv}, \mathbf{w}_{pcv}, \lambda_1, \lambda_2) = MSE + \lambda_1^T (\mathbf{F}_{fcv}^T \mathbf{w}_{fcv} - \mathbf{f}_{fcv}) + \lambda_2^T (\mathbf{F}_{pcv}^T \mathbf{w}_{pcv}) \quad (3.11)$$

Afin de trouver  $\mathbf{w}_{pcv}$  et  $\mathbf{w}_{fcv}$  permettant d'estimer  $\hat{Y}^m$ , on cherche à résoudre les équations décrites en 3.12

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{w}_{fcv}} &= 2\mathbf{C}^{fcv} \mathbf{w}_{fcv} + 2\mathbf{C}^{f-pcv} \mathbf{w}_{pcv} - 2\mathbf{c}_{fcv} + \mathbf{F}_{fcv} \lambda_1 = 0 \\ \frac{\partial \mathcal{L}}{\partial \mathbf{w}_{pcv}} &= 2\mathbf{C}^{pcv} \mathbf{w}_{pcv} + 2\mathbf{C}^{f-pcvT} \mathbf{w}_{fcv} - 2\mathbf{w}_{pcv}^T \mathbf{c}_{f-pcv} + \mathbf{F}_{pcv} \lambda_2 = 0 \\ \frac{\partial \mathcal{L}}{\partial \lambda_1} &= \mathbf{F}_{fcv}^T \mathbf{w}_{fcv} - \mathbf{f}_{fcv} = 0 \\ \frac{\partial \mathcal{L}}{\partial \lambda_2} &= \mathbf{F}_{pcv}^T \mathbf{w}_{pcv} = 0 \end{aligned} \quad (3.12)$$

On peut également écrire ce système d'équations sous forme matricielle :

$$\begin{bmatrix} \mathbf{C}^{fcv} & \mathbf{C}^{f-pcv} & \mathbf{F}_{fcv} & \mathbf{0} \\ \mathbf{C}^{f-pcvT} & \mathbf{C}^{pcv} & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{fcv} \\ \mathbf{w}_{pcv} \\ \lambda_1/2 \\ \lambda_2/2 \end{bmatrix} = \begin{bmatrix} \mathbf{c}_{fcv} \\ \mathbf{c}_{f-pcv} \\ \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix} \quad (3.13)$$

On pose :

$$\tilde{\mathbf{C}}_m = \begin{bmatrix} \mathbf{C}^{fcv} & \mathbf{C}^{f-pcv} \\ \mathbf{C}^{f-pcvT} & \mathbf{C}^{pcv} \end{bmatrix} = \begin{bmatrix} \mathbf{C}^{fcv}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) & \mathbf{C}^{f-pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) \\ \mathbf{C}^{f-pcvT}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) & \mathbf{C}^{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) \end{bmatrix} \quad (3.14)$$

$$\tilde{\mathbf{F}}_m = \begin{bmatrix} \mathbf{F}_{fcv} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv} \end{bmatrix} \quad (3.15)$$

$$\tilde{\mathbf{w}}_m = \begin{bmatrix} \mathbf{w}_{fcv} \\ \mathbf{w}_{f-pcv} \end{bmatrix} \quad (3.16)$$

$$\tilde{\lambda}_m = \begin{bmatrix} \lambda_1/2 \\ \lambda_2/2 \end{bmatrix} \quad (3.17)$$

$$\tilde{\mathbf{c}}_m = \begin{bmatrix} \mathbf{c}_{fcv} \\ \mathbf{c}_{f-pcv} \end{bmatrix} \quad (3.18)$$

$$\tilde{\mathbf{f}}_m = \begin{bmatrix} \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix} \quad (3.19)$$

Le système décrit à l'équation 3.13 est alors réécrit sous la forme :

$$\begin{bmatrix} \tilde{\mathbf{C}}_m & \tilde{\mathbf{F}}_m \\ \tilde{\mathbf{F}}_m^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{w}}_m \\ \tilde{\boldsymbol{\lambda}}_m \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{c}}_m \\ \tilde{\mathbf{f}}_m \end{bmatrix} \quad (3.20)$$

Le système décrit à l'équation 3.20 est donc tout à fait comparable à celui du krigeage (équation 1.43). L'estimateur qui en résulte a donc la même forme que l'estimateur du krigeage.

$$\hat{Y}^m(\mathbf{x}) = \tilde{\mathbf{f}}_m(\mathbf{x})^T \tilde{\boldsymbol{\beta}}_m + \tilde{\mathbf{c}}_m(\mathbf{x})^T \tilde{\mathbf{C}}_m^{-1} (\tilde{\mathbf{Y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m) \quad (3.21)$$

avec  $\tilde{\boldsymbol{\beta}}_m = (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{C}}_m^{-1} \tilde{\mathbf{F}}_m)^{-1} (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{C}}_m^{-1} \tilde{\mathbf{Y}}_m) = [\beta_{fcv}^T \ \beta_{pcv}^T]^T$ . On note également  $\tilde{\mathbf{Y}}_m = [\mathbf{Y}_{fcv}^T \ \mathbf{Y}_{pcv}^T]^T$

Ainsi, lorsque l'on cherche à déterminer la distribution  $Y_{fcv}(\mathbf{x})$  conditionnellement à l'événement  $\tilde{\mathcal{E}} = \left\{ Y_{fcv}^{(i)} = y_{fcv}^{(i)}, \forall i \in \llbracket 1, n_{fcv} \rrbracket \ \& \ Y_{pcv}^{(j)} = y_{pcv}^{(j)}, \forall j \in \llbracket 1, n_{pcv} \rrbracket \right\}$ , on obtient :

$$Y_{fcv}(\mathbf{x}) \mid \tilde{\mathcal{E}} = \mathcal{N}(\hat{y}^m(\mathbf{x}), s_{\hat{y}^m}^2(\mathbf{x})) \quad (3.22)$$

$\hat{y}^m(\mathbf{x})$  s'exprime donc sous la forme :

$$\hat{y}^m(\mathbf{x}) = \tilde{\mathbf{f}}_m(\mathbf{x})^T \tilde{\boldsymbol{\beta}}_m + \tilde{\mathbf{c}}_m(\mathbf{x})^T \tilde{\mathbf{C}}_m^{-1} (\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m) \quad (3.23)$$

$s_{\hat{y}^m}^2(\mathbf{x})$  s'écrit alors comme l'erreur quadratique moyenne de l'estimateur, à savoir :

$$\begin{aligned} s_{\hat{y}^m}^2(\mathbf{x}) &= \left( \sigma_{fcv}^2 + \tilde{\mathbf{u}}_m(\mathbf{x})^T (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{C}}_m^{-1} \tilde{\mathbf{F}}_m)^{-1} \tilde{\mathbf{u}}_m(\mathbf{x}) - \tilde{\mathbf{c}}_m(\mathbf{x})^T \tilde{\mathbf{C}}_m^{-1} \tilde{\mathbf{c}}_m(\mathbf{x}) \right) \\ &\text{avec : } \tilde{\mathbf{u}}_m(\mathbf{x}) = \tilde{\mathbf{F}}_m^T \tilde{\mathbf{C}}_m^{-1} \tilde{\mathbf{c}}_m(\mathbf{x}) - \tilde{\mathbf{f}}_m(\mathbf{x}) \end{aligned} \quad (3.24)$$

### 1.1.1 Remarques

On peut bien sûr associer à chaque matrice de covariance  $\mathbf{C}^{fcv}$ ,  $\mathbf{C}^{pcv}$  et  $\mathbf{C}^{f-pcv}$ , une matrice de corrélation comme cela est fait pour le krigeage. Ainsi :

$$\mathbf{C}^{fcv}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) = \sigma_{fcv}^2 \mathbf{R}^{fcv}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) \quad (3.25)$$

$$\mathbf{C}^{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) = \sigma_{pcv}^2 \mathbf{R}^{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) \quad (3.26)$$

$$\mathbf{C}^{f-pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) = \sigma_{fcv} \sigma_{pcv} \mathbf{R}^{f-pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) \quad (3.27)$$

Chaque élément des matrices  $\mathbf{R}^{fcv}$ ,  $\mathbf{R}^{pcv}$  et  $\mathbf{R}^{f-pcv}$  s'exprime alors comme suit :

$$\mathbf{R}_{ij}^{fcv} = R^{fcv}(\mathbf{x}_{fcv}^{(i)}, \mathbf{x}_{fcv}^{(j)}; \mathbf{l}_{fcv})_{i,j=1,\dots,n_{fcv}} \quad (3.28)$$

$$\mathbf{R}_{ij}^{pcv} = R^{pcv}(\mathbf{x}_{pcv}^{(i)}, \mathbf{x}_{pcv}^{(j)}; \mathbf{l}_{pcv})_{i,j=1,\dots,n_{pcv}} \quad (3.29)$$

$$\mathbf{R}_{ij}^{f-pcv} = R^{f-pcv} \left( \mathbf{x}_{fcv}^{(i)}, \mathbf{x}_{pcv}^{(j)}; \mathbf{l}_{f-pcv} \right)_{i=1, \dots, n_{fcv}; j=1, \dots, n_{pcv}} \quad (3.30)$$

On constate donc de manière générale que ces matrices peuvent s'exprimer suivant différentes fonctions de corrélation  $R^{fcv}$ ,  $R^{pcv}$  et  $R^{f-pcv}$ .

De la même manière, on peut exprimer les vecteurs  $\mathbf{c}_{fcv}(\mathbf{x})$  et  $\mathbf{c}_{f-pcv}(\mathbf{x})$  sous la forme :

$$\begin{aligned} \mathbf{c}_{fcv}(\mathbf{x}) &= \sigma_{fcv}^2 \mathbf{r}_{fcv}(\mathbf{x}) \\ \mathbf{c}_{f-pcv}(\mathbf{x}) &= \sigma_{fcv} \sigma_{pcv} \mathbf{r}_{f-pcv}(\mathbf{x}) \end{aligned} \quad (3.31)$$

avec :

$$\begin{aligned} \mathbf{r}_{fcv}(\mathbf{x}) &= \left[ R_{fcv}(\mathbf{x}, \mathbf{x}_{fcv}^{(1)}; \mathbf{l}_{fcv}) \quad \dots \quad R_{fcv}(\mathbf{x}, \mathbf{x}_{fcv}^{(n_{fcv})}; \mathbf{l}_{fcv}) \right]^T \\ \mathbf{r}_{f-pcv}(\mathbf{x}) &= \left[ R_{f-pcv}(\mathbf{x}, \mathbf{x}_{pcv}^{(1)}; \mathbf{l}_{f-pcv}) \quad \dots \quad R_{f-pcv}(\mathbf{x}, \mathbf{x}_{pcv}^{(n_{pcv})}; \mathbf{l}_{f-pcv}) \right]^T \end{aligned} \quad (3.32)$$

On peut donc réécrire le système matriciel défini à l'équation 3.13 sous la forme :

$$\begin{aligned} \begin{bmatrix} \sigma_{fcv}^2 \mathbf{R}^{fcv} & \sigma_{fcv} \sigma_{pcv} \mathbf{R}^{f-pcv} & \mathbf{F}_{fcv} & \mathbf{0} \\ \sigma_{fcv} \sigma_{pcv} \mathbf{R}^{f-pcvT} & \sigma_{pcv}^2 \mathbf{R}^{pcv} & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{fcv} \\ \mathbf{w}_{pcv} \\ \lambda_1/2 \\ \lambda_2/2 \end{bmatrix} \\ = \begin{bmatrix} \sigma_{fcv}^2 \mathbf{r}_{fcv} \\ \sigma_{fcv} \sigma_{pcv} \mathbf{r}_{f-pcv} \\ \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix} \end{aligned} \quad (3.33)$$

### 1.1.2 Preuve de l'interpolation du métamodèle

On peut aisément vérifier que l'estimateur trouvé  $\hat{y}^m$  interpole bien les points  $\mathbf{x}^{(i)}$  calculés totalement convergés et qu'en ces mêmes points  $s_{\hat{y}^m}^2$  est nul.

On considère l'estimation faite au point  $\mathbf{x}^{(i)}$ , cette estimation est égale à  $y^{(i)}$ . Le vecteur  $\tilde{\mathbf{c}}_m(\mathbf{x}^{(i)})$  est égal à la  $i^{\text{ème}}$  ligne et colonne de la matrice de corrélation  $\tilde{\mathbf{C}}$  avec  $i \leq n_{fcv}$ . On peut alors écrire :  $\tilde{\mathbf{c}}_m(\mathbf{x}^{(i)}) = \tilde{\mathbf{C}}_m \cdot \delta_i$  et  $\delta_i$  un vecteur de taille  $(n_{fcv} + n_{pcv}) \times 1$  égal à un vecteur uniquement non nul en sa  $i^{\text{ème}}$  composante. Ainsi

$$\delta_i = \left[ \underbrace{0}_{1} \quad \dots \quad \underbrace{0}_{i-1} \quad \underbrace{1}_i \quad \underbrace{0}_{i+1} \quad \dots \quad \underbrace{0}_{n_{fcv}+n_{pcv}} \right]^T. \text{ On obtient alors } \tilde{\mathbf{c}}_m(\mathbf{x}^{(i)})^T \tilde{\mathbf{C}}_m^{-1} = \delta_i^T \cdot \tilde{\mathbf{C}}_m \cdot \tilde{\mathbf{C}}_m^{-1} = \delta_i^T \cdot \mathbf{I}_d = \delta_i^T.$$

Pour rappel  $\tilde{\mathbf{F}}_m = \begin{bmatrix} \mathbf{F}_{fcv} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv} \end{bmatrix}$  et, de ce fait par définition  $\delta_i^T \cdot \tilde{\mathbf{F}}_m = \tilde{\mathbf{f}}_m(\mathbf{x}^{(i)})^T$ .

Ainsi,

$$\tilde{y}^m(\mathbf{x}^{(i)}) = \tilde{\mathbf{f}}_m(\mathbf{x}^{(i)})^T \tilde{\boldsymbol{\beta}}_m + \tilde{\mathbf{c}}_m(\mathbf{x})^T \tilde{\mathbf{C}}_m^{-1} (\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m) = \tilde{\mathbf{f}}_m(\mathbf{x}^{(i)})^T \tilde{\boldsymbol{\beta}}_m + y^{(i)} - \tilde{\mathbf{f}}_m(\mathbf{x}^{(i)})^T \tilde{\boldsymbol{\beta}}_m = y^{(i)} \quad (3.34)$$

De la même façon, on montre que  $s_{\tilde{y}^m}^2(\mathbf{x}^{(i)}) = 0$ . En effet

$$\tilde{\mathbf{u}}_m(\mathbf{x}^{(i)}) = \underbrace{\tilde{\mathbf{F}}_m^T \tilde{\mathbf{C}}_m^{-1} \tilde{\mathbf{c}}_m(\mathbf{x}^{(i)})}_{\tilde{\mathbf{f}}_m(\mathbf{x}^{(i)})} - \tilde{\mathbf{f}}_m(\mathbf{x}^{(i)}) = 0$$

## 1.2 Méthodes de cokrigage

Dans la section précédente, on a introduit les équations permettant de construire un métamodèle grâce au cokrigage multi-fidélité. L'élément déterminant dans la construction du métamodèle est la définition de la matrice de covariance. Cette matrice de covariance est composée de quatre blocs  $\mathbf{C}^{f_{cv}}$ ,  $\mathbf{C}^{pcv}$ ,  $\mathbf{C}^{f-pcv}$  et  $\mathbf{C}^{f-pcv^T}$ . **La difficulté principale** se situe dans le calcul de la matrice d'intercovariance  $\mathbf{C}^{f-pcv}$ . En effet, cette matrice quantifie le lien entre les données totalement et partiellement convergées. Dans le but d'aider à déterminer ces matrices de covariances, il peut être introduit des fonctions de corrélation notées  $R^{f_{cv}}$ ,  $R^{pcv}$  et  $R^{f-pcv}$ . On présente alors trois approches de construction des matrices de covariance (ou de corrélation).

**Remarque :** Dans la suite des équations, certains indices  $m$  sont remplacés par  $c_1$ ,  $c_2$  et  $c_3$  afin de mieux mettre en avant les spécificités des trois méthodes proposées (notamment les matrices de corrélation ou de covariance).

### 1.2.1 Méthode de cokrigage multi-fidélité, notée méthode « Intercorrélation »

Cette première approche est proposée par [Zimmermann et Han, 2010]. Dans ce document, elle est appelée méthode « Intercorrélation ». Elle est la plus simple des trois méthodes de cokrigage multi-fidélité qui sont présentées, que ce soit dans son écriture mathématique ou dans son implémentation numérique. La principale hypothèse de cette méthode est de supposer :

$$\text{Cov}(Z_{f_{cv}}(\mathbf{x}^+), Z_{f_{cv}}(\mathbf{x}^*)) = \text{Cov}(Z_{pcv}(\mathbf{x}^+), Z_{pcv}(\mathbf{x}^*)) \quad \forall(\mathbf{x}^+, \mathbf{x}^*) \quad (3.35)$$

**On fait donc l'hypothèse que la covariance de deux points  $\mathbf{x}^+$  et  $\mathbf{x}^*$  est identique, que ces points soient évalués totalement ou partiellement convergés. Cette hypothèse est assez forte, elle suppose que le comportement des données totalement et partiellement convergées est semblable pour assurer un métamodèle multi-fidélité de qualité.**

Cette équation se traduit par le fait que :

$$\sigma_{f_{cv}}^2 R^{f_{cv}}(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}_{f_{cv}}) = \sigma_{pcv}^2 R^{pcv}(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}_{pcv})$$

Cette relation étant valable pour tout  $\mathbf{x}^+$  et  $\mathbf{x}^*$ , elle l'est également pour  $\mathbf{x}^+ = \mathbf{x}^*$ . Ce qui amène à :

$$\sigma_{f_{cv}}^2 R^{f_{cv}}(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}_{f_{cv}}) = \sigma_{p_{cv}}^2 R^{p_{cv}}(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}_{p_{cv}}) \implies \boxed{\sigma_{f_{cv}}^2 = \sigma_{p_{cv}}^2 = \sigma^2} \quad \forall \mathbf{x}^+ = \mathbf{x}^* \quad (3.36)$$

En effet, si  $\mathbf{x}^+ = \mathbf{x}^*$  alors  $R^{f_{cv}}(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}_{f_{cv}}) = R^{p_{cv}}(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}_{p_{cv}}) = 1$ . On en déduit donc la relation suivante :

$$R^{f_{cv}}(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}_{f_{cv}}) = R^{p_{cv}}(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}_{p_{cv}}) = R(\mathbf{x}^+, \mathbf{x}^*; \mathbf{l}) \quad \forall \mathbf{x}^+, \mathbf{x}^* \quad (3.37)$$

**Conclusion :** Dans cette approche, on impose d'utiliser la même fonction de corrélation  $R$  pour les données totalement et partiellement convergées. La variance des processus gaussien  $Y_{f_{cv}}$  et  $Y_{p_{cv}}$  est également identique.

Il reste alors à définir la fonction d'intercorrélation  $R^{f-p_{cv}}$ . Pour ce faire on pose :

$$R^{f-p_{cv}} = \gamma R \text{ avec } \gamma \in [0, 1[ \quad (3.38)$$

On peut alors réécrire la matrice  $\tilde{\mathbf{C}}_m$  définie par l'équation 3.14 sous la forme :

$$\tilde{\mathbf{C}}_m = \begin{bmatrix} \mathbf{C}^{f_{cv}} & \mathbf{C}^{f-p_{cv}} \\ \mathbf{C}^{f-p_{cv}T} & \mathbf{C}^{p_{cv}} \end{bmatrix} = \sigma^2 \begin{bmatrix} \mathbf{R}(\mathbf{x}_{f_{cv}}, \mathbf{x}_{f_{cv}}) & \gamma \mathbf{R}(\mathbf{x}_{f_{cv}}, \mathbf{x}_{p_{cv}}) \\ \gamma \mathbf{R}(\mathbf{x}_{f_{cv}}, \mathbf{x}_{p_{cv}})^T & \mathbf{R}(\mathbf{x}_{p_{cv}}, \mathbf{x}_{p_{cv}}) \end{bmatrix} = \sigma^2 \tilde{\mathbf{R}}_{c1} \quad (3.39)$$

L'estimateur  $\hat{y}^m$  s'écrit alors dans le cadre de cette méthode :

$$\hat{y}^m(\mathbf{x}) = \tilde{\mathbf{f}}_m(\mathbf{x})^T \tilde{\boldsymbol{\beta}}_m + \tilde{\mathbf{r}}_{c1}(\mathbf{x})^T \tilde{\mathbf{R}}_{c1}^{-1} (\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m) \quad (3.40)$$

avec :

$$\tilde{\mathbf{r}}_{c1}(\mathbf{x}) = \begin{bmatrix} R(\mathbf{x}, \mathbf{x}_{f_{cv}}^{(1)}; \mathbf{l}) & \dots & R(\mathbf{x}, \mathbf{x}_{f_{cv}}^{(n_{f_{cv}})}; \mathbf{l}) & \gamma R(\mathbf{x}, \mathbf{x}_{p_{cv}}^{(1)}; \mathbf{l}) & \dots & \gamma R(\mathbf{x}, \mathbf{x}_{p_{cv}}^{(n_{p_{cv}})}; \mathbf{l}) \end{bmatrix}^T$$

On peut également exprimer l'erreur quadratique moyenne de cet estimateur sous la forme :

$$s_{\hat{y}^m}^2(\mathbf{x}) = \sigma^2 \left( 1 + \tilde{\mathbf{u}}_m(\mathbf{x})^T (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{R}}_{c1}^{-1} \tilde{\mathbf{F}}_m)^{-1} \tilde{\mathbf{u}}_m(\mathbf{x}) - \tilde{\mathbf{r}}_{c1}(\mathbf{x})^T \tilde{\mathbf{R}}_{c1}^{-1} \tilde{\mathbf{r}}_{c1}(\mathbf{x}) \right) \quad (3.41)$$

### 1.2.2 Détermination des paramètres du cokrigeage Intercorrélation

De la même manière que pour le krigage, les inconnus restant à déterminer sont  $\sigma_{f_{cv}}^2 = \sigma_{p_{cv}}^2 = \sigma^2$  et  $\mathbf{l}$ . Ceux-ci sont également déterminés comme étant les paramètres assurant le maximum d'une fonction de vraisemblance  $L_{c1}$ . On définit  $L_{c1}$  comme :

$$L_{c1}(\sigma^2, \mathbf{l}) = \frac{1}{\sqrt{(2\pi)^{n_{f_{cv}}+n_{p_{cv}}} (\sigma^2)^{n_{f_{cv}}+n_{p_{cv}}} |\tilde{\mathbf{R}}_{c1}|}} e^{-\frac{(\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m)^T \tilde{\mathbf{R}}_{c1}^{-1} (\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m)}{2\sigma^2}} \quad (3.42)$$



De manière pratique, on préfère utiliser une formulation logarithmique :

$$\ln L_{c1}(\sigma^2, \mathbf{l}) = -\frac{n_{fcv} + n_{pcv}}{2} \ln \sigma^2 - \frac{1}{2} \ln |\tilde{\mathbf{R}}_{c1}| - \frac{(\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m)^T \tilde{\mathbf{R}}_{c1}^{-1} (\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m)}{2\sigma^2} \quad (3.43)$$

On détermine alors la valeur de  $\sigma^2$  avec l'équation 3.44 :

$$\frac{\partial L_{c1}}{\partial \sigma^2} = 0 \implies \sigma^2 = \frac{(\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m)^T \tilde{\mathbf{R}}_{c1}^{-1} (\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_m)}{n_{fcv} + n_{pcv}} \quad (3.44)$$

Les paramètres  $l_k$  du vecteur  $\mathbf{l}$  sont ensuite déterminés à l'aide d'un algorithme d'optimisation de la fonction  $L_{c1}$ . Dans le toolbox GRENAT, cela se fait au choix en utilisant un algorithme génétique ou un algorithme à essais de particules.

### 1.2.3 Méthode de cokrigage multi-fidélité, notée « $s_{fcv}/s_{pcv}$ »

Cette deuxième approche est proposée par [Han *et al.*, 2010, Han *et al.*, 2012]. Cette méthode est notée, dans ce document méthode de cokrigage «  $s_{fcv}/s_{pcv}$  ». **Elle s'attache à déterminer un facteur multiplicatif entre les données totalement et partiellement convergées. Ce facteur multiplicatif agit comme un indicateur de l'amplitude entre données totalement et partiellement via la corrélation de ces données.** Pour rappel, le système d'équation décrit en 3.33 est :

$$\begin{bmatrix} \sigma_{fcv}^2 \mathbf{R}^{fcv} & \sigma_{fcv} \sigma_{pcv} \mathbf{R}^{f-pcv} & \mathbf{F}_{fcv} & \mathbf{0} \\ \sigma_{fcv} \sigma_{pcv} \mathbf{R}^{f-pcvT} & \sigma_{pcv}^2 \mathbf{R}^{pcv} & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{fcv} \\ \mathbf{w}_{pcv} \\ \lambda_1/2 \\ \lambda_2/2 \end{bmatrix} = \begin{bmatrix} \sigma_{fcv}^2 \mathbf{r}_{fcv} \\ \sigma_{fcv} \sigma_{pcv} \mathbf{r}_{f-pcv} \\ \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix}$$

Ce qui peut également s'écrire sous la forme :

$$\begin{bmatrix} \mathbf{R}^{fcv} & \mathbf{R}^{f-pcv} & \mathbf{F}_{fcv} & \mathbf{0} \\ \mathbf{R}^{f-pcvT} & \mathbf{R}^{pcv} & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{fcv} \\ \frac{\sigma_{pcv}}{\sigma_{fcv}} \mathbf{w}_{pcv} \\ \lambda_1/(2\sigma_{fcv}^2) \\ \lambda_2/(2\sigma_{fcv}\sigma_{pcv}) \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{fcv} \\ \mathbf{r}_{f-pcv} \\ \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix} \quad (3.45)$$

On pose alors :

$$\begin{aligned}
\tilde{\mathbf{R}}_{c2} &= \begin{bmatrix} \mathbf{R}^{fcv} & \mathbf{R}^{f-pcv} \\ \mathbf{R}^{f-pcvT} & \mathbf{R}^{pcv} \end{bmatrix} \\
\tilde{\mathbf{w}}_{c2} &= \begin{bmatrix} \mathbf{w}_{fcv} \\ \frac{\sigma_{pcv}}{\sigma_{fcv}} \mathbf{w}_{pcv} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{w}}_{fcv} \\ \tilde{\mathbf{w}}_{pcv} \end{bmatrix} \\
\tilde{\boldsymbol{\lambda}}_{c2} &= \begin{bmatrix} \lambda_1 / (2\sigma_{fcv}^2) \\ \lambda_2 / (2\sigma_{fcv}\sigma_{pcv}) \end{bmatrix} \\
\tilde{\mathbf{r}}_{c2} &= \begin{bmatrix} \mathbf{r}_{fcv} \\ \mathbf{r}_{f-pcv} \end{bmatrix}
\end{aligned} \tag{3.46}$$

ce qui amène à écrire le système matriciel :

$$\begin{bmatrix} \tilde{\mathbf{R}}_{c2} & \tilde{\mathbf{F}}_m \\ \tilde{\mathbf{F}}_m^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{w}}_{c2} \\ \tilde{\boldsymbol{\lambda}}_{c2} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{r}}_{c2} \\ \tilde{\mathbf{f}}_m \end{bmatrix} \tag{3.47}$$

Pour rappel, on cherche l'estimateur  $\hat{Y}^m$  sous la forme :

$$\hat{Y}^m(\mathbf{x}) = \mathbf{w}_{fcv}^T(\mathbf{x})\mathbf{Y}_{fcv} + \mathbf{w}_{pcv}^T(\mathbf{x})\mathbf{Y}_{pcv}$$

ce qui est équivalent à :

$$\hat{Y}^m(\mathbf{x}) = \tilde{\mathbf{w}}_{fcv}\mathbf{Y}_{fcv} + \tilde{\mathbf{w}}_{pcv}\frac{\sigma_{fcv}}{\sigma_{pcv}}\mathbf{Y}_{pcv} \tag{3.48}$$

On pose alors :

$$\tilde{\mathbf{Y}}_{c2} = \begin{bmatrix} \mathbf{Y}_{fcv} \\ \frac{\sigma_{fcv}}{\sigma_{pcv}}\mathbf{Y}_{pcv} \end{bmatrix}$$

et

$$\tilde{\mathbf{y}}_{c2} = \begin{bmatrix} \mathbf{y}_{fcv} \\ \frac{\sigma_{fcv}}{\sigma_{pcv}}\mathbf{y}_{pcv} \end{bmatrix}$$

La résolution du système défini à l'équation (3.47) amène à écrire l'estimateur  $\hat{y}^m$  sous la forme :

$$\hat{y}^m(\mathbf{x}) = \tilde{\mathbf{f}}_m(\mathbf{x})^T \tilde{\boldsymbol{\beta}}_{c2} + \tilde{\mathbf{r}}_{c2}(\mathbf{x})^T \tilde{\mathbf{R}}_{c2}^{-1} (\tilde{\mathbf{y}}_{c2} - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_{c2}) \tag{3.49}$$

avec :

$$\tilde{\boldsymbol{\beta}}_{c2} = (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{R}}_{c2}^{-1} \tilde{\mathbf{F}}_m)^{-1} (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{R}}_{c2}^{-1} \tilde{\mathbf{y}}_{c2})$$

L'erreur quadratique moyenne  $s_{\hat{y}^m}^2$  de cet estimateur  $\hat{y}^m$  peut être définie par l'équation 3.24. Cependant, une écriture équivalente est ici présentée afin de disposer plus aisément de cette quantité. Pour cela, il est nécessaire de repartir de la définition faite de l'erreur quadratique moyenne écrite à l'équation 3.10. On en déduit :

$$\begin{aligned}
s_{\tilde{y}_{fcv}}^2 &= \sigma_{fcv}^2 + \mathbf{w}_{fcv}^T \mathbf{C}^{fcv} \mathbf{w}_{fcv} + \mathbf{w}_{pcv}^T \mathbf{C}^{pcv} \mathbf{w}_{pcv} + 2\mathbf{w}_{fcv}^T \mathbf{C}^{f-pcv} \mathbf{w}_{pcv} - 2\mathbf{w}_{fcv}^T \mathbf{c}_{fcv} - 2\mathbf{w}_{pcv}^T \mathbf{c}_{f-pcv} \\
&= \sigma_{fcv}^2 + \begin{bmatrix} \mathbf{w}_{fcv}^T & \mathbf{w}_{pcv}^T \end{bmatrix} \begin{bmatrix} \mathbf{C}^{fcv} & \mathbf{C}^{f-pcv} \\ \mathbf{C}^{f-pcvT} & \mathbf{C}^{pcv} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{fcv} \\ \mathbf{w}_{pcv} \end{bmatrix} - 2 \begin{bmatrix} \mathbf{w}_{fcv}^T & \mathbf{w}_{pcv}^T \end{bmatrix} \begin{bmatrix} \mathbf{c}_{fcv} \\ \mathbf{c}_{f-pcv} \end{bmatrix} \\
&= \sigma_{fcv}^2 + \begin{bmatrix} \tilde{\mathbf{w}}_{fcv}^T & \tilde{\mathbf{w}}_{pcv}^T \end{bmatrix} \begin{bmatrix} \mathbf{1}_{fcv} & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{pcv} \frac{\sigma_{fcv}}{\sigma_{pcv}} \end{bmatrix} \begin{bmatrix} \mathbf{C}^{fcv} & \mathbf{C}^{f-pcv} \\ \mathbf{C}^{f-pcvT} & \mathbf{C}^{pcv} \end{bmatrix} \begin{bmatrix} \mathbf{1}_{fcv}^T & \mathbf{0}^T \\ \mathbf{0}^T & \mathbf{1}_{pcv}^T \frac{\sigma_{fcv}}{\sigma_{pcv}} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{w}}_{fcv} \\ \tilde{\mathbf{w}}_{pcv} \end{bmatrix} \\
&\quad - 2 \begin{bmatrix} \tilde{\mathbf{w}}_{fcv}^T & \tilde{\mathbf{w}}_{pcv}^T \end{bmatrix} \begin{bmatrix} \mathbf{1}_{fcv} & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{pcv} \frac{\sigma_{fcv}}{\sigma_{pcv}} \end{bmatrix} \begin{bmatrix} \mathbf{c}_{fcv} \\ \mathbf{c}_{f-pcv} \end{bmatrix} \\
&= \sigma_{fcv}^2 \left( 1 + \begin{bmatrix} \tilde{\mathbf{w}}_{fcv}^T & \tilde{\mathbf{w}}_{pcv}^T \end{bmatrix} \begin{bmatrix} \mathbf{R}^{fcv} & \mathbf{R}^{f-pcv} \\ \mathbf{R}^{f-pcvT} & \mathbf{R}^{pcv} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{w}}_{fcv} \\ \tilde{\mathbf{w}}_{pcv} \end{bmatrix} - 2 \begin{bmatrix} \tilde{\mathbf{w}}_{fcv}^T & \tilde{\mathbf{w}}_{pcv}^T \end{bmatrix} \begin{bmatrix} \mathbf{r}_{fcv} \\ \mathbf{r}_{f-pcv} \end{bmatrix} \right) \\
&= \sigma_{fcv}^2 (1 + \tilde{\mathbf{w}}_{c2}^T (\tilde{\mathbf{R}}_{c2} \tilde{\mathbf{w}}_{c2} - 2\tilde{\mathbf{r}}_{c2}))
\end{aligned} \tag{3.50}$$

On en déduit alors

$$\begin{aligned}
s_{\tilde{y}^m}^2(\mathbf{x}) &= \sigma_{fcv}^2 \left( 1 + \tilde{\mathbf{u}}_{c2}(\mathbf{x})^T (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{R}}_{c2}^{-1} \tilde{\mathbf{F}}_m)^{-1} \tilde{\mathbf{u}}_{c2}(\mathbf{x}) - \tilde{\mathbf{r}}_{c2}(\mathbf{x})^T \tilde{\mathbf{R}}_{c2}^{-1} \tilde{\mathbf{r}}_{c2}(\mathbf{x}) \right) \\
&\quad \text{avec } \tilde{\mathbf{u}}_{c2}(\mathbf{x}) = \tilde{\mathbf{F}}_m^T \tilde{\mathbf{R}}_{c2}^{-1} \tilde{\mathbf{r}}_{c2}(\mathbf{x}) - \tilde{\mathbf{f}}_m(\mathbf{x})
\end{aligned} \tag{3.51}$$

Cette méthode offre donc une grande souplesse de construction de la matrice de corrélation de cokrigeage  $\tilde{\mathbf{R}}_{c2}$ . Cependant, comme précisé dans [Han *et al.*, 2010, Yamazaki et Mavriplis, 2011], cette souplesse de choix de fonction de corrélation  $R^{fcv}$ ,  $R^{pcv}$  et  $R^{f-pcv}$  et des paramètres  $\mathbf{l}_{fcv}$ ,  $\mathbf{l}_{pcv}$  et  $\mathbf{l}_{f-pcv}$  associés à ces fonctions pose des problèmes de conditionnement de la matrice  $\tilde{\mathbf{R}}_{c2}$  et donc d'inversion de cette matrice. Afin d'y remédier, il est choisi d'imposer  $R^{fcv} = R^{pcv} = R$  et  $R^{f-pcv} = 0.9999R$  ainsi que  $\mathbf{l}_{fcv} = \mathbf{l}_{pcv} = \mathbf{l}_{f-pcv} = \mathbf{l}$ , comme préconisé dans [Han *et al.*, 2010, Yamazaki et Mavriplis, 2011]. De ce fait, on définit également  $\tilde{\mathbf{r}}_{c2}$  par :

$$\tilde{\mathbf{r}}_{c2}(\mathbf{x}) = \begin{bmatrix} R(\mathbf{x}, \mathbf{x}_{fcv}^{(1)}; \mathbf{l}) & \dots & R(\mathbf{x}, \mathbf{x}_{fcv}^{(n_{fcv})}; \mathbf{l}) & 0.9999R(\mathbf{x}, \mathbf{x}_{pcv}^{(1)}; \mathbf{l}) & \dots & 0.9999R(\mathbf{x}, \mathbf{x}_{pcv}^{(n_{pcv})}; \mathbf{l}) \end{bmatrix}^T$$

#### 1.2.4 Détermination des paramètres du cokrigeage $s_{fcv}/s_{pcv}$

Les paramètres de cokrigeage restant à déterminer sont  $\sigma_{fcv}^2$  et  $\frac{\sigma_{fcv}}{\sigma_{pcv}}$  et  $\mathbf{l}$ . Ceux-ci sont calculés comme maximisant une fonction de vraisemblance  $L_{c2}$ .

$$L_{c2}(\boldsymbol{\beta}, \sigma_{fcv}/\sigma_{pcv}, \sigma_{fcv}^2, \mathbf{l}) = \frac{1}{\sqrt{(2\pi\sigma_{fcv}^2)^{n_{fcv}+n_{pcv}} |\tilde{\mathbf{R}}_{c2}|}} e^{-\frac{(\tilde{\mathbf{y}}_{c2} - \tilde{\mathbf{F}}_m \boldsymbol{\beta})^T \tilde{\mathbf{R}}_{c2}^{-1} (\tilde{\mathbf{y}}_{c2} - \tilde{\mathbf{F}}_m \boldsymbol{\beta})}{2\sigma_{fcv}^2}} \tag{3.52}$$

En utilisant la formulation logarithmique, on obtient :

$$\ln L_{c2} = -\frac{n_{fcv} + n_{pcv}}{2} \ln \sigma_{fcv}^2 - \frac{1}{2} \ln |\tilde{\mathbf{R}}_{c2}| - \frac{(\tilde{\mathbf{y}}_{c2} - \tilde{\mathbf{F}}_m \boldsymbol{\beta})^T \tilde{\mathbf{R}}_{c2}^{-1} (\tilde{\mathbf{y}}_{c2} - \tilde{\mathbf{F}}_m \boldsymbol{\beta})}{2\sigma_{fcv}^2} \quad (3.53)$$

On retrouve  $\tilde{\boldsymbol{\beta}}_{c2}$  précédemment défini, par l'équation :

$$\frac{\partial \ln L_{c2}}{\partial \boldsymbol{\beta}} = 0 \implies \boldsymbol{\beta} = (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{R}}_{c2}^{-1} \tilde{\mathbf{F}}_m)^{-1} (\tilde{\mathbf{F}}_m^T \tilde{\mathbf{R}}_{c2}^{-1} \tilde{\mathbf{y}}_{c2}) = \tilde{\boldsymbol{\beta}}_{c2} \quad (3.54)$$

On trouve  $\sigma_{fcv}^2$  par l'équation suivante :

$$\frac{\partial L_{c2}}{\partial \sigma_{fcv}^2} = 0 \implies \sigma_{fcv}^2 = \frac{(\tilde{\mathbf{y}}_{c2} - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_{c2})^T \tilde{\mathbf{R}}_{c2}^{-1} (\tilde{\mathbf{y}}_{c2} - \tilde{\mathbf{F}}_m \tilde{\boldsymbol{\beta}}_{c2})}{n_{fcv} + n_{pcv}} \quad (3.55)$$

De même pour la quantité  $\sigma_{fcv}/\sigma_{pcv}$

$$\begin{aligned} \frac{\partial L_{c2}}{\partial \sigma_{fcv}/\sigma_{pcv}} = 0 \implies \\ \frac{\sigma_{fcv}}{\sigma_{pcv}} = \left( \begin{bmatrix} 0 \\ \mathbf{y}_{pcv} \end{bmatrix}^T \tilde{\mathbf{R}}_{c2}^{-1} \begin{bmatrix} 0 \\ \mathbf{y}_{pcv} \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ \mathbf{y}_{pcv} \end{bmatrix}^T \tilde{\mathbf{R}}_{c2}^{-1} \begin{bmatrix} \mathbf{F}_{fcv} \boldsymbol{\beta}_{fcv} - \mathbf{y}_{fcv} \\ \mathbf{F}_{pcv} \boldsymbol{\beta}_{pcv} \end{bmatrix} \end{aligned} \quad (3.56)$$

Cependant, on ne peut pas déterminer directement la valeur de ces paramètres. En effet, au vu des équations 3.54, 3.55 et 3.56, on constate que celles-ci ne sont pas indépendantes. La détermination directe des paramètres  $\tilde{\boldsymbol{\beta}}_{c2}$ ,  $\sigma_{fcv}^2$  et  $\sigma_{fcv}/\sigma_{pcv}$  n'est donc pas envisageable. Il est donc nécessaire de résoudre les équations 3.54 et 3.56. Cette résolution ne pose pas de problèmes particuliers. Une fois que les valeurs  $\tilde{\boldsymbol{\beta}}_{c2}$  et  $\sigma_{fcv}/\sigma_{pcv}$  sont déterminées, on peut à l'aide de l'équation 3.55 trouver la valeur de  $\sigma_{fcv}^2$ .

On peut alors déterminer les paramètres  $\mathbf{l}$ , comme pour les autres approches, à l'aide d'un processus d'optimisation numérique.

### 1.2.5 Méthode de cokrigage multi-fidélité de type auto-régressif

Cette dernière approche fût développée dans [Kennedy et O'Hagan, 2000] et est l'une des plus répandues dans la littérature [Kennedy et O'Hagan, 2000, Forrester *et al.*, 2007, Ulaganathan *et al.*, 2014, Miller *et al.*, 2012]. Dans le cadre des travaux de cette thèse, le formalisme utilisé s'appuie plus particulièrement sur les développements réalisés par [Forrester *et al.*, 2007].

On suppose tout d'abord  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$ . **La méthode de construction du cokrigage s'appuie sur une approche de type auto-régressive**, à savoir :

$$Y_{fcv}(\mathbf{x}) = \rho Y_{pcv}(\mathbf{x}) + Y_{cor}(\mathbf{x}) \quad (3.57)$$

où  $\rho$  est un paramètre de régression de  $Y_{pcv}$  et  $Y_{cor}$  un processus indépendant de  $Y_{fcv}$  et  $Y_{pcv}$ . Ce modèle s'appuie sur une propriété de Markov [Kennedy et O'Hagan, 2000] à savoir :

$$\text{Cov}[Y_{fcv}(\mathbf{x}^{(i)}), Y_{pcv}(\mathbf{x}^+) | Y_{pcv}(\mathbf{x}^{(i)})] = 0, \quad \forall \mathbf{x}^{(i)} \neq \mathbf{x}^+ \quad (3.58)$$

Cette propriété signifie que si l'évaluation  $Y_{pcv}(\mathbf{x}^{(i)})$  est connue, alors rien de plus ne peut être appris sur  $Y_{fcv}(\mathbf{x}^{(i)})$  avec une simulation  $Y_{pcv}(\mathbf{x}^+)$  pour  $\mathbf{x}^{(i)} \neq \mathbf{x}^+$ .

Déterminer les termes de la matrice  $\tilde{\mathbf{C}}_m$  définie par l'équation 3.14 est alors rendu possible :

$$\begin{aligned} \mathbf{C}^{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) &= \text{Cov}[Y_{pcv}(\mathbf{x}_{pcv}), Y_{pcv}(\mathbf{x}_{pcv})] \\ &= \text{Cov}[Z_{pcv}(\mathbf{x}_{pcv}), Z_{pcv}(\mathbf{x}_{pcv})] \\ &= \sigma_{pcv}^2 \mathbf{R}^{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) \end{aligned} \quad (3.59)$$

$$\begin{aligned} \mathbf{C}^{f-pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) &= \text{Cov}[Y_{fcv}(\mathbf{x}_{fcv}), Y_{pcv}(\mathbf{x}_{pcv})] \\ &= \text{Cov}[Z_{fcv}(\mathbf{x}_{fcv}), Z_{pcv}(\mathbf{x}_{pcv})] \\ &= \text{Cov}[\rho Z_{pcv}(\mathbf{x}_{fcv}) + Z_{cor}(\mathbf{x}_{fcv}), Z_{pcv}(\mathbf{x}_{pcv})] \\ &= \text{Cov}[\rho Z_{pcv}(\mathbf{x}_{fcv}), Z_{pcv}(\mathbf{x}_{pcv})] + \underbrace{\text{Cov}[Z_{cor}(\mathbf{x}_{fcv}), Z_{pcv}(\mathbf{x}_{pcv})]}_0 \\ &= \rho \sigma_{pcv}^2 \mathbf{R}^{pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) \end{aligned} \quad (3.60)$$

$$\begin{aligned} \mathbf{C}^{fcv}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) &= \text{Cov}[Y_{fcv}(\mathbf{x}_{fcv}), Y_{fcv}(\mathbf{x}_{fcv})] \\ &= \text{Cov}[Z_{fcv}(\mathbf{x}_{fcv}), Z_{fcv}(\mathbf{x}_{fcv})] \\ &= \text{Cov}[\rho Z_{pcv}(\mathbf{x}_{fcv}) + Z_{cor}(\mathbf{x}_{fcv}), \rho Z_{pcv}(\mathbf{x}_{fcv}) + Z_{cor}(\mathbf{x}_{fcv})] \\ &= \rho^2 \text{Cov}[\rho Z_{pcv}(\mathbf{x}_{fcv}), \rho Z_{pcv}(\mathbf{x}_{fcv})] + \text{Cov}[Z_{cor}(\mathbf{x}_{fcv}), Z_{cor}(\mathbf{x}_{fcv})] \\ &= \rho^2 \sigma_{pcv}^2 \mathbf{R}^{pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) + \sigma_{cor}^2 \mathbf{R}^{cor}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) \end{aligned} \quad (3.61)$$

Au final (en renommant  $\tilde{\mathbf{C}}_m$  par  $\tilde{\mathbf{C}}_{c3}$  pour éviter d'éventuelles confusions) :

$$\tilde{\mathbf{C}}_{c3} = \begin{bmatrix} \rho^2 \sigma_{pcv}^2 \mathbf{R}^{pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) + \sigma_{cor}^2 \mathbf{R}^{cor}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) & \rho \sigma_{pcv}^2 \mathbf{R}^{pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) \\ \rho \sigma_{pcv}^2 \mathbf{R}^{pcvT}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) & \sigma_{pcv}^2 \mathbf{R}^{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) \end{bmatrix} \quad (3.62)$$

De la même manière on peut définir,  $\tilde{\mathbf{c}}_{c3} = [ \mathbf{c}_{fcv}^T \quad \mathbf{c}_{f-pcv}^T ]^T$ .

$$\tilde{\mathbf{c}}_{c3} = \begin{bmatrix} \rho^2 \sigma_{pcv}^2 \mathbf{r}_{pcv}(\mathbf{x}_{fcv}, \mathbf{x}) + \sigma_{pcv}^2 \mathbf{r}_{cor}(\mathbf{x}_{fcv}, \mathbf{x}) \\ \rho \sigma_{pcv}^2 \mathbf{r}_{pcv}(\mathbf{x}_{pcv}, \mathbf{x}) \end{bmatrix} \quad (3.63)$$

Par rapport à l'approche originale (voir [Kennedy et O'Hagan, 2000]), on simplifie le calcul des termes déterministes du cokrigeage. On suppose notamment dans le cadre dans lequel la méthode a été implantée que ([Forrester *et al.*, 2007]) :

$$\mathbf{f}_{fcv}^T(\mathbf{x}) \boldsymbol{\beta}_{fcv} = \mathbf{f}_{pcv}^T(\mathbf{x}) \boldsymbol{\beta}_{pcv} \quad \& \quad \boldsymbol{\beta}_{fcv} = \boldsymbol{\beta}_{pcv} \quad \& \quad \mathbf{f}_{fcv} = \mathbf{f}_{pcv} \quad (3.64)$$

Cette hypothèse a pour effet de ramener le nombre de conditions de non-biais à une au lieu de deux. La nouvelle condition de non-biais devient alors (voir équation 3.7) :

$$\mathbf{F}_{fcv}^T \mathbf{w}_{fcv} + \mathbf{F}_{pcv}^T \mathbf{w}_{pcv} - \mathbf{f}_{fcv} = 0 \quad (3.65)$$

Le nouveau système à résoudre devient en adaptant celui décrit par l'équation 3.13.

$$\begin{bmatrix} \mathbf{C}^{fcv} & \mathbf{C}^{f-pcv} & \mathbf{F}_{fcv} \\ \mathbf{C}^{f-pcvT} & \mathbf{C}^{pcv} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{F}_{pcv}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{fcv} \\ \mathbf{w}_{pcv} \\ \lambda_1/2 \end{bmatrix} = \begin{bmatrix} \mathbf{c}_{fcv} \\ \mathbf{c}_{f-pcv} \\ \mathbf{f}_{fcv} \end{bmatrix} \quad (3.66)$$

On pose alors :

$$\tilde{\mathbf{F}}_{c3} = \begin{bmatrix} \mathbf{F}_{fcv} \\ \mathbf{F}_{pcv} \end{bmatrix}$$

Au final, d'après l'équation 3.23, on déduit l'estimateur  $\hat{\mathbf{y}}^m(\mathbf{x})$  comme :

$$\hat{\mathbf{y}}^m(\mathbf{x}) = \mathbf{f}_{fcv}(\mathbf{x})^T \tilde{\boldsymbol{\beta}}_{c3} + \tilde{\mathbf{c}}_{c3}(\mathbf{x})^T \tilde{\mathbf{C}}_{c3}^{-1} (\tilde{\mathbf{y}}_m - \tilde{\mathbf{F}}_{c3} \tilde{\boldsymbol{\beta}}_{c3}) \quad (3.67)$$

avec  $\tilde{\boldsymbol{\beta}}_{c3} = (\tilde{\mathbf{F}}_{c3}^T \tilde{\mathbf{C}}_{c3}^{-1} \tilde{\mathbf{F}}_{c3})^{-1} (\tilde{\mathbf{F}}_{c3}^T \tilde{\mathbf{C}}_{c3}^{-1} \tilde{\mathbf{y}}_m)$

$s_{\hat{\mathbf{y}}^m}^2$  s'exprime donc comme l'erreur quadratique moyenne de l'estimateur, à savoir :

$$s_{\hat{\mathbf{y}}^m}^2(\mathbf{x}) = \left( \sigma_{pcv}^2 + \rho \sigma_{cor}^2 + \tilde{\mathbf{u}}_{c3}(\mathbf{x})^T (\tilde{\mathbf{F}}_{c3}^T \tilde{\mathbf{C}}_{c3}^{-1} \tilde{\mathbf{F}}_{c3})^{-1} \tilde{\mathbf{u}}_{c3}(\mathbf{x}) - \tilde{\mathbf{c}}_{c3}(\mathbf{x})^T \tilde{\mathbf{C}}_{c3}^{-1} \tilde{\mathbf{c}}_{c3}(\mathbf{x}) \right) \quad (3.68)$$

$$\text{avec : } \tilde{\mathbf{u}}_{c3}(\mathbf{x}) = \tilde{\mathbf{F}}_{c3}^T \tilde{\mathbf{C}}_{c3}^{-1} \tilde{\mathbf{c}}_{c3}(\mathbf{x}) - \mathbf{f}_{fcv}(\mathbf{x})$$

En effet, de par la relation 3.57, on en déduit :  $\sigma_{fcv}^2 = \sigma_{pcv}^2 + \rho \sigma_{cor}^2$ . Si on se place dans le cadre du cokrigeage ordinaire, on retrouve bien les mêmes estimateurs que [Forrester *et al.*, 2007].

### 1.2.6 Détermination des paramètres du métamodèle de cokrigeage de type auto-régressif

Comme pour les autres méthodes de cokrigeage présentées précédemment, certains paramètres restent à déterminer, à savoir :  $\rho$ ,  $\mathbf{l}_{cor}$ ,  $\mathbf{l}_{pcv}$ ,  $\sigma_{cor}^2$  et  $\sigma_{pcv}^2$ . Leurs estimations s'effectuent en deux temps. On peut, tout d'abord estimer les paramètres  $\mathbf{l}_{pcv}$  et  $\sigma_{pcv}^2$  via le calcul du maximum de la fonction de vraisemblance utilisée dans le cadre du krigeage (équation 1.49).

Pour déterminer les paramètres,  $\rho$ ,  $\mathbf{l}_{cor}$  et  $\sigma_{cor}^2$ , on pose tout d'abord :

$$\mathbf{d} = \mathbf{y}_{fcv} - \rho \mathbf{y}_{pcv}(\mathbf{X}_{fcv}) \quad (3.69)$$

avec  $\mathbf{y}_{pcv}(\mathbf{X}_{fcv})$  le vecteur des valeurs de  $\mathbf{y}_{pcv}$  aux points  $\mathbf{X}_{fcv}$ . Ceci justifie donc le fait que  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$ . Si cette propriété n'est pas vérifiée, une extrapolation doit être faite. On peut alors choisir comme valeur de  $\mathbf{y}_{pcv}(\mathbf{X}_{fcv})$ , les valeurs fournies par le métamodèle de krigeage  $\hat{\mathbf{y}}_{pcv}$  aux points  $\mathbf{X}_{fcv}$ ; comme proposé dans [Forrester *et al.*, 2007] (si celui-ci est suffisamment précis, cette hypothèse est tout à fait viable). On cherche alors à optimiser la fonction de vraisemblance suivante :

$$L_{c3\_cor}(\boldsymbol{\beta}_{cor}, \sigma_{cor}^2, \mathbf{l}_{cor}) = \frac{1}{\sqrt{(2\pi)^{n_{fcv}} |\mathbf{C}^{cor}|}} e^{-\frac{1}{2}(\mathbf{d} - \mathbf{F}_{fcv} \boldsymbol{\beta}_{cor})^T \mathbf{C}^{cor-1} (\mathbf{d} - \mathbf{F}_{fcv} \boldsymbol{\beta}_{cor})} \quad (3.70)$$

On en déduit ainsi :

$$\begin{aligned}\sigma_{cor}^2 &= \frac{(\mathbf{d} - \mathbf{F}_{fcv}\boldsymbol{\beta}_{cor})^T \mathbf{R}^{cor-1} (\mathbf{d} - \mathbf{F}_{fcv}\boldsymbol{\beta}_{cor})}{n_{fcv}} \\ \boldsymbol{\beta}_{cor} &= (\mathbf{F}_{fcv}^T \mathbf{R}^{cor-1} \mathbf{F}_{fcv})^{-1} (\mathbf{F}_{fcv}^T \mathbf{R}^{cor-1} \mathbf{d})\end{aligned}\quad (3.71)$$

Les paramètres  $\mathbf{l}_{cor}$  et  $\rho$  sont alors déterminés par optimisation numérique.

### 1.3 Krigeage hiérarchique

Le krigeage hiérarchique a été introduit par [Han et Görtz, 2012]. Celui-ci a montré une bonne qualité de réponse et de robustesse. Il est également intéressant d'un point de vue implémentation numérique puisqu'il est équivalent en terme de taille de matrice à du krigeage ordinaire. **L'idée du krigeage hiérarchique est de remplacer, dans les équations du krigeage, la fonction déterministe par le métamodèle des données partiellement convergées, celui-ci étant pondéré par un facteur  $\beta$ . Ce facteur  $\beta$  fait ainsi le lien entre les données totalement et partiellement convergées.** On suppose que le processus gaussien utilisé pour déterminer le métamodèle est construit par :

$$Y_{fcv}(\mathbf{x}) = \hat{y}_{pcv}(\mathbf{x})\beta + Z(\mathbf{x}) \quad (3.72)$$

avec  $\hat{y}_{pcv}(\mathbf{x})$  le métamodèle créé à partir des données  $\mathbf{y}_{pcv}$  (ce métamodèle de données partiellement convergées n'est pas nécessairement issu du krigeage). On constate donc que la seule modification du krigeage hiérarchique par rapport au krigeage est le remplacement du terme de régression qui devient  $\hat{y}_{pcv}(\mathbf{x})\beta$ . De la même manière que pour le krigeage, on cherche le meilleur estimateur linéaire non-biaisé, ce qui nous amène en adaptant les notations à :

$$\begin{bmatrix} \mathbf{R}_{fcv} & \hat{\mathbf{Y}}_{pcv} \\ \hat{\mathbf{Y}}_{pcv}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{fcv} \\ \lambda \\ \frac{\lambda}{2\sigma_{fcv}^2} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{fcv} \\ \hat{y}_{pcv} \end{bmatrix} \quad (3.73)$$

avec  $\hat{\mathbf{Y}}_{pcv} = [\hat{y}_{pcv}(\mathbf{x}^1) \dots \hat{y}_{pcv}(\mathbf{x}^{n_{fcv}})]^T$ . On en déduit, en s'appuyant sur ce qui a été développé sur le krigeage au premier chapitre, les estimateurs  $\hat{y}^m$  et  $s_{\hat{y}^m}^2$  :

$$\begin{aligned}\hat{y}^m(\mathbf{x}) &= \hat{y}_{pcv}(\mathbf{x})\beta + \mathbf{r}_{fcv}(\mathbf{x})^T \mathbf{R}_{fcv}^{-1} (\mathbf{y}_{fcv} - \hat{\mathbf{Y}}_{pcv}\beta) \\ \text{avec : } \beta &= (\hat{\mathbf{Y}}_{pcv}^T \mathbf{R}_{fcv}^{-1} \hat{\mathbf{Y}}_{pcv})^{-1} (\hat{\mathbf{Y}}_{pcv}^T \mathbf{R}_{fcv}^{-1} \mathbf{y}_{fcv})\end{aligned}\quad (3.74)$$

$$\begin{aligned}s_{\hat{y}^m}^2(\mathbf{x}) &= \sigma_{fcv}^2 \left( 1 + \mathbf{u}(\mathbf{x})^T (\hat{\mathbf{Y}}_{pcv}^T \mathbf{R}_{fcv}^{-1} \hat{\mathbf{Y}}_{pcv})^{-1} \mathbf{u}(\mathbf{x}) - \mathbf{r}_{fcv}(\mathbf{x})^T \mathbf{R}_{fcv}^{-1} \mathbf{r}_{fcv}(\mathbf{x}) \right) \\ \text{avec : } \mathbf{u}(\mathbf{x}) &= \hat{\mathbf{Y}}_{pcv}^T \mathbf{R}_{fcv}^{-1} \mathbf{r}_{fcv}(\mathbf{x}) - \hat{y}_{pcv}(\mathbf{x})\end{aligned}\quad (3.75)$$

$\mathbf{r}_{fcv}$  est le vecteur de corrélation entre les données totalement convergées  $fcv$  et la donnée  $\mathbf{x}$  à évaluer.  $\mathbf{R}_{fcv}$  est la matrice de corrélation des données totalement convergées.

### 1.3.1 Détermination des paramètres

Tout comme pour le krigeage, il est nécessaire de déterminer les paramètres  $\sigma_{fcv}^2$  et  $\mathbf{I}_{fcv}$ . De même que pour les méthodes précédentes, une fonction de vraisemblance est utilisée pour trouver la valeur adéquate de ces paramètres. La fonction de vraisemblance définie est semblable à celle utilisée pour déterminer les paramètres du krigeage :

$$L(\boldsymbol{\beta}, \sigma_{fcv}^2, \mathbf{I}_{fcv}) = \frac{1}{\sqrt{(2\pi)^{n_{fcv}} (\sigma_{fcv}^2)^{n_{fcv}} |\mathbf{R}_{fcv}|}} e^{-\frac{(\mathbf{y}_{fcv} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta})^T \mathbf{R}_{fcv}^{-1} (\mathbf{y}_{fcv} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta})}{2\sigma_{fcv}^2}} \quad (3.76)$$

Ce qui nous amène en maximisant cette expression à :

$$\sigma_{fcv}^2 = \frac{(\mathbf{y}_{fcv} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta})^T \mathbf{R}_{fcv}^{-1} (\mathbf{y}_{fcv} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta})}{n_{fcv}} \quad (3.77)$$

## 1.4 Illustration des stratégies multi-fidélité

Dans cette section, on illustre sur un exemple analytique 1D les performances atteintes par les méthodes de cokrigeage et de krigeage hiérarchique. L'exemple utilisé a été introduit par Forrester ([Forrester *et al.*, 2007]). Il est répandu dans la littérature pour illustrer les performances envisageables. Il est défini sur l'intervalle [0 1] par :

$$\begin{aligned} \text{Modèle haute fidélité : } x &\mapsto y_{hf}(x) = (6x - 2)^2 \times \sin(12x - 4) \\ \text{Modèle basse fidélité : } x &\mapsto y_{lf}(x) = A \times (6x - 2)^2 \times \sin(12x - 4) + 10(x - 0.5) - 5 \end{aligned} \quad (3.78)$$

Le facteur  $A$  peut prendre différentes valeurs, on s'intéressera plus particulièrement aux valeurs  $A = 0.5$ ,  $A = 0.08$  or  $A = 100$ . Le modèle basse fidélité est évalué avec 11 points (●) calculés à intervalle régulier. On comparera également le métamodèle multi-fidélité construit avec le métamodèle de krigeage obtenu uniquement avec les données haute fidélité (■). Les objectifs simples de ce paragraphe sont :

- Illustrer sur un exemple analytique les méthodes de cokrigeage et de krigeage hiérarchique
- Valider l'implémentation de ces méthodes dans la toolbox GREMAT
- Proposer une première comparaison des méthodes

### 1.4.1 Cokrigeage : approche par la méthode de cokrigeage Intercorrélation

On présente sur la figure 3.1 le résultat obtenu pour le cokrigeage Intercorrélation ordinaire et universel (degré de régression égal à l'ordre 1). On constate clairement que l'approche sur cet exemple est peu intéressante par rapport au résultat obtenu avec le métamodèle de krigeage construit avec uniquement les données haute fidélité.

Le modèle de cokrigeage obtenu ne permet pas de représenter avec précision le modèle haute fidélité. Ceci s'explique notamment par le fait que les données totalement et partiellement convergées ne sont pas hautement corrélées. En effet, les coefficients de corrélation  $r_{ccc}$



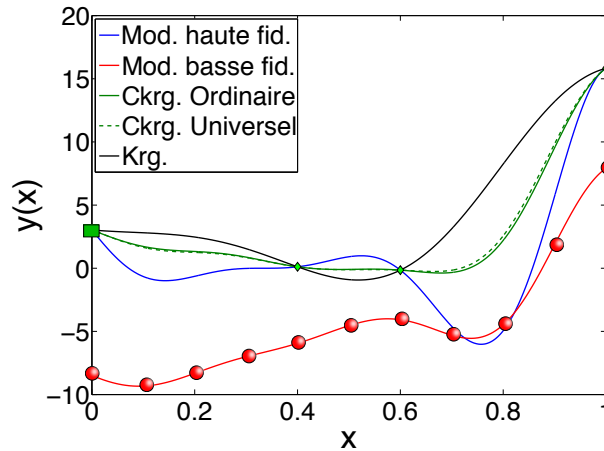


FIGURE 3.1: Illustration du Cokrigage Intercorrélation avec  $A = 0.5$

et  $r^2$  entre le modèle haute fidélité et basse fidélité sont respectivement de 0.4 et 0.5, ce qui est assez faible.

Afin d'illustrer sous un angle plus flatteur les performances pouvant être atteintes par cette approche [Zimmermann et Han, 2010] propose d'utiliser un autre exemple :

$$\begin{aligned} \text{Modèle haute fidélité : } x &\longmapsto y_{hf}(x) = \sin\left(\frac{2\pi x}{5}\right) - 0.5x + 5 \\ \text{Modèle basse fidélité : } x &\longmapsto y_{lf}(x) = 0.9 \sin\left(\frac{2\pi x}{5}\right) - 0.3x + 5 \end{aligned} \quad (3.79)$$

Sur cet exemple (figure 3.2), 15 points du modèle basse fidélité et trois du modèle haute fidélité sont utilisés. Une fonction de corrélation gaussienne est également employée (pour plus de détails voir [Zimmermann et Han, 2010]). On constate clairement que l'approche de cokrigage est une stratégie gagnante quant à la qualité du métamodèle obtenue par rapport au métamodèle de krigage obtenu avec uniquement les données haute fidélité.

La méthode de cokrigage Intercorrélation peut donc être une stratégie très efficace. Pour cela, il est nécessaire que les données basse et haute fidélité soient suffisamment corrélées.

#### 1.4.2 Cokrigage : approche par la méthode de cokrigage $s_{fcv}/s_{pcv}$ et auto-régressif

Les deux autres méthodes de cokrigage sont plus appropriées pour traiter l'exemple défini à l'équation 3.78. En effet, elles estiment chacune un rapport multiplicatif entre les données partiellement et totalement convergées. L'approche de cokrigage  $s_{fcv}/s_{pcv}$  est testée sur l'exemple analytique avec une fonction de corrélation spline comme proposé dans [Han *et al.*, 2010]. Elle est testée pour différentes valeurs de  $A$  (0.08, 0.5 et 100 sur la figure 3.3(a)). L'estimation faite de la quantité  $\sigma_{fcv}/\sigma_{pcv}$  est comparée à celle obtenue en construisant un métamodèle de krigage des modèles haute et basse fidélité à l'aide de 1000 points. Ceci, afin de vérifier que notre estimation est correctement faite. (Ceci a été également étudié dans [Han *et al.*, 2010]).

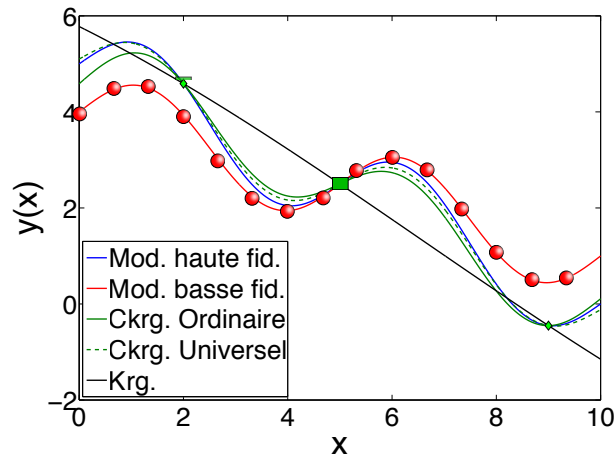


FIGURE 3.2: Illustration du Cokrigage Intercorrélation sur le second exemple

Les résultats obtenus à la figure 3.3 montrent clairement une efficacité de la méthode à fournir un métamodèle multi-fidélité de qualité. La qualité obtenue provient du fait que l'estimation faite de  $\sigma_{fcv}/\sigma_{pcv}$  est inversement proportionnelle à  $A$ . Ce qui est corroboré par le fait que l'évaluation de celle-ci est correctement effectuée par rapport à l'estimation réelle (voir figure 3.3(b)). On retrouve donc bien les mêmes résultats qui ont été obtenus originellement dans [Han *et al.*, 2010].

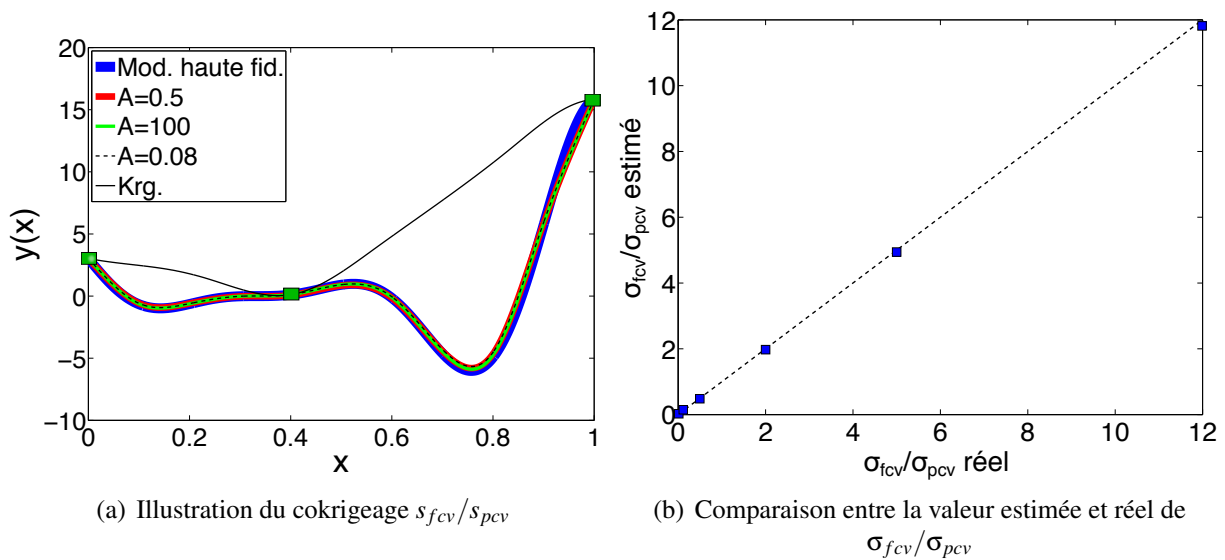


FIGURE 3.3: Résultats obtenus à l'aide du cokrigage  $\sigma_{fcv}/\sigma_{pcv}$

Pour l'approche par cokrigage de type auto-régressif, on construit le métamodèle multi-fidélité également pour différentes valeurs de  $A$  (voir figure 3.4).

Les fonctions de corrélation utilisées sont de type exponentielle généralisée comme dans

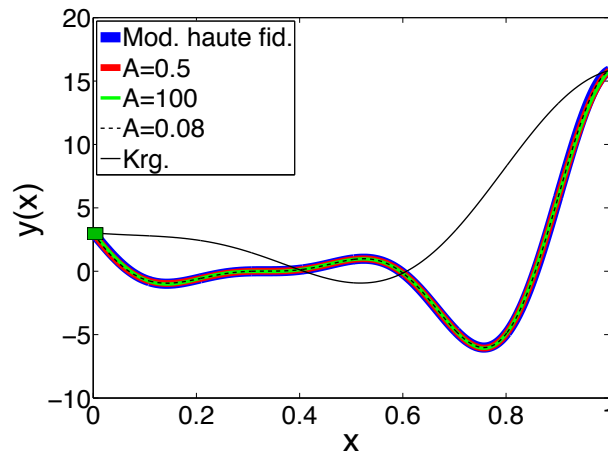


FIGURE 3.4: Illustration du Cokrigage auto-régressif

[Forrester *et al.*, 2007]. On constate là encore une bonne efficacité quant à évaluer le modèle haute fidélité. Le paramètre de cokrigage  $\rho$  est également inversement proportionnel à  $A$  et permet donc une mise à l'échelle efficace des matrices de covariances.

### 1.4.3 Krigeage hiérarchique

De la même façon que précédemment on illustre sur cet exemple le krigeage hiérarchique (voir figure 3.5). On constate également que cette méthode est appropriée pour créer un métamodèle de qualité représentatif du modèle haute fidélité que l'on souhaite décrire. Le paramètre  $\beta$  se trouve alors égal au paramètre  $A$ , expliquant la précision du résultat obtenu. Le métamodèle de données partiellement convergées est obtenu par krigeage. Les fonctions de corrélation utilisées sont de type exponentielle généralisée.

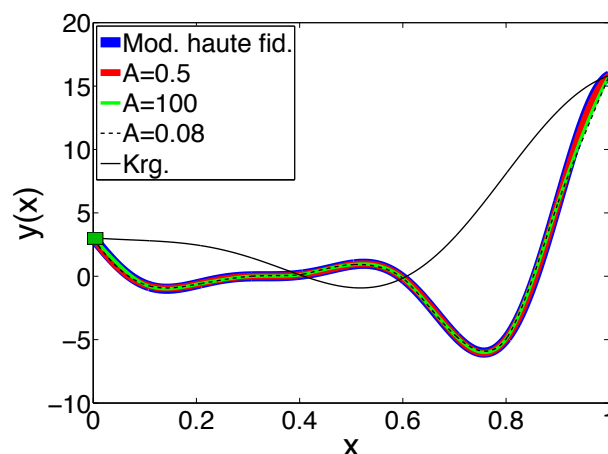


FIGURE 3.5: Illustration du krigeage hiérarchique

#### 1.4.4 Bilan intermédiaire

De manière théorique et au vu des équations, la méthode de cokrigage Intercorrélation est appropriée lorsque les données haute fidélité et basse fidélité sont suffisamment corrélées. Ceci a été mis en avant sur les deux exemples analytiques traités précédemment. La présence du paramètre  $\gamma$  rend l'approche plus flexible et potentiellement plus performante que les méthodes de correction de surface. Les deux autres méthodes de cokrigage s'appuient sur une mise à l'échelle des données basse fidélité ou de leurs covariances, ce qui permet de calculer leur influence sur le métamodèle multi-fidélité finalement construit. Le krigeage hiérarchique, quant à lui, tient compte de l'influence des données basse fidélité sans avoir à calculer une matrice d'intercorrélation. C'est ce dernier point qui rend cette approche robuste (voir [Han et Görtz, 2012]).

On a pu constater ces différentes propriétés sur le précédent exemple analytique. Ainsi, certaines méthodes multi-fidélité ont montré leur efficacité quant à décrire efficacement le modèle haute fidélité. Cependant, il est important de garder en mémoire que sur cet exemple, le modèle basse fidélité n'est qu'une transformation linéaire du modèle haute fidélité. La relation entre les deux modèles n'est donc pas complexe et se trouve bien adaptée pour les approches proposées utilisant un coefficient multiplicatif entre données haute fidélité et basse fidélité. Néanmoins, la détermination de ces coefficients peut s'avérer complexe sur des cas test réels. Ainsi, dans le cadre de la mécanique des fluides, [Zimmermann et Han, 2010] conclut que la méthode de cokrigage Intercorrélation peut finalement s'avérer être plus efficace que la méthode de cokrigage de type auto-régressive. En effet, déterminer le paramètre  $\rho$  de cette méthode peut s'avérer ardu lorsque la relation entre le modèle basse fidélité et haute fidélité n'est pas aussi simple que sur ce cas test analytique. Une étude sur des cas test de mécanique du solide est donc nécessaire afin de déterminer quelles sont les méthodes de cokrigage les plus efficaces.

## 2 Etudes comparatives sur des exemples mécaniques

### 2.1 Introduction

On s'intéresse à comparer les méthodes d'Evofusion, de cokrigage et de krigeage hiérarchique sur des cas test mécanique d'assemblage. Comme dit précédemment, cette étude s'avère nécessaire d'être réalisée car le lien entre modèle haute et basse fidélité n'est pas clairement établi et est *a priori* inconnu. Le but de cette étude est de trouver quelles sont les méthodes de modélisation multi-fidélité les plus adaptées aux cas test mécaniques. Pour mener à bien ce travail, différents éléments de comparaison sont étudiés **dans le cadre de la stratégie de recalage**. Ces aspects peuvent être découpés en quatre sous-études :

- Qualité du métamodèle pour un nombre initial de points partiellement convergés ( $10 \times n_{dim}$ ) et totalement convergés.
- Qualité du métamodèle en fonction du temps de calcul et du nombre de points utilisés lors de la phase d'enrichissement de la stratégie de recalage.
- Nombre de points calculés et temps de calcul nécessaire pour atteindre une corrélation  $r_{ccc}$  de 0.95 par rapport à un métamodèle de référence.

- Temps de calcul nécessaire pour atteindre un critère d'arrêt d'enrichissement (défini par la suite à l'équation 3.82) et qualité du métamodèle multi-fidélité obtenu.

Ces critères sont étudiés pour différents niveaux de convergence des données partiellement convergées. Les niveaux de convergence varient dans la suite de l'étude pour des niveaux d'indicateurs d'erreurs  $\eta$  de l'algorithme LaTIn de 0.05 à 0.01. En effet, l'étude préliminaire du chapitre 2 a montré une sensibilité à cet indicateur d'erreur concernant les temps de calcul nécessaires à la conception d'un métamodèle représentatif de la fonction objectif.

Comme expliqué au cours de ce chapitre, il est nécessaire pour construire les métamodèles de cokrigage et de krigeage hiérarchique de disposer de données totalement convergées. Le paragraphe suivant vise à expliquer l'algorithme utilisé permettant de choisir un échantillonnage de points à calculer totalement convergés parmi ceux à calculer partiellement convergés (afin de vérifier  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$ ). Par la suite, on définira la stratégie de recalage appliquée au métamodèle de cokrigage et de krigeage hiérarchique (qui est tout à fait semblable à celle utilisée précédemment pour l'EvoFusion).

## 2.2 Choix de points totalement convergés

Comme vu dans la présentation des différentes méthodes, il est nécessaire de calculer des données partiellement et totalement convergées. Le choix  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$  n'est pas rare dans la littérature mais n'est pas toujours une prérogative pour toutes les méthodes de modélisation multi-fidélité (comme notamment pour le cokrigage Intercorrélation ou  $s_{fcv}/s_{pcv}$  ainsi que le krigeage hiérarchique). Dans ce chapitre, ce choix est effectué afin d'assurer une comparaison viable des cinq méthodes proposées (EvoFusion, 3 approches de cokrigage et krigeage hiérarchique). Pour les méthodes de cokrigage et de krigeage hiérarchique, un choix initial de points  $\mathbf{X}_{fcv}$  parmi  $\mathbf{X}_{pcv}$  est à effectuer. Ce choix doit être fait intelligemment, ces points devant en effet être représentatifs de l'espace de conception.

**Remarque :** Pour rappel, concernant l'EvoFusion, les points totalement convergés sont issus directement de l'enrichissement de la stratégie de recalage. On pourrait également construire un premier métamodèle évofusé comportant des points  $\mathbf{X}_{fcv}$  choisis parmi  $\mathbf{X}_{pcv}$ , mais ce n'est pas la philosophie proposée par Forrester. Hors, comme dit précédemment, **on s'attache dans ce chapitre à comparer des méthodes proposées dans la littérature**. Ce choix n'a d'ailleurs pas d'impact pour l'EvoFusion sur **le résultat de la stratégie de recalage** comme il a été vérifié et plus précisément mis en avant au paragraphe 2.2 du chapitre 4. Il n'en est pas de même pour la stratégie de recherche de minimum (voir chapitre 4). Cette remarque illustre que beaucoup d'adaptations de méthodes sont possibles et que certaines ne pourront être qu'abordées.

On se propose alors de choisir ces points  $\mathbf{X}_{fcv}$  à l'aide du critère de Morris-Mitchell [Morris et Mitchell, 1995]. Ce critère est défini par :

$$\phi_p(D) = \left[ \sum_j d_j^{-p} \right]^{1/p} \quad (3.80)$$

avec  $D$  un choix de tirages  $\mathbf{X}_{fcv}$ ,  $j$  une paire de points,  $d_j$  la distance entre ces points. Minimiser ce critère a pour but de trouver le jeu de points ayant la plus grande distance minimale possible entre deux points<sup>1</sup>.

Trouver le meilleur tirage au sens de  $\phi_p$  n'est pas dur en soi mais demande un grand nombre de combinaisons de choix de tirages  $\mathbf{X}_{fcv}$  possibles. Le nombre de tirages à envisager est  $n_{pcv}C_{n_{fcv}} = n_{pcv}!/n_{fcv}!(n_{pcv}-n_{fcv})!$ . Pour un nombre de points partiellement convergés,  $n_{pcv} = 30$  et un nombre de points totalement convergés souhaités  $n_{fcv} = 9$ , le nombre possible de choix est d'environ 14 millions ! Tester toutes les combinatoires est alors trop coûteux.

Un algorithme d'échange (*exchange algorithm*) ([Cook et Nachtsheim, 1980]) est alors utilisé comme dans [Forrester *et al.*, 2007] pour faire cette sélection. Une description complète de cette technique est présentée par l'algorithme 3. Un premier choix de points  $\mathbf{X}_{fcv}$  est effectué. Le critère  $\phi_p(D)$  est alors évalué. Le premier point  $\mathbf{x}_{fcv}^{(1)}$  est échangé avec les points restants  $\mathbf{X}_{pcv} \setminus \mathbf{X}_{fcv}$ . Le point échangé amenant au critère de Morris-Mitchell minimum est alors retenu. Le processus est alors répété pour les points  $\mathbf{x}_{fcv}^{(2)} \dots \mathbf{x}_{fcv}^{(n_{fcv})}$ .

---

### Algorithm 3 Algorithme d'échange

---

- 1: Choix aléatoire d'un tirage  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$
  - 2: Evaluation du critère  $\phi_p$
  - 3: **for**  $i=1 : n_{fcv}$  **do**
  - 4:   **for**  $j=1 : n_{pcv} - n_{fcv}$  **do**
  - 5:      $\mathbf{x}_{fcv}^{(j)} \in \mathbf{X}_{pcv} \setminus \mathbf{X}_{fcv} \rightarrow \mathbf{x}_{fcv}^{(i)}$
  - 6:     Evaluation du critère  $\phi_p$
  - 7:   **end for**
  - 8:   Remplacer  $\mathbf{x}_{fcv}^{(i)}$  par le point  $\mathbf{x}_{fcv}^{(j)}$  minimisant  $\phi_p$
  - 9: **end for**
- 

Avec cet algorithme, le nombre de possibilités testées est  $n_{fcv} \times (n_{pcv} - n_{fcv}) + 1$ , ce qui donne pour  $n_{pcv} = 30$  et  $n_{fcv} = 9$ , un nombre de possibilités de 190. Le nombre de combinaisons testées est donc drastiquement réduit. Cependant, le choix des points  $\mathbf{X}_{fcv}$  n'est pas nécessairement le plus optimal du point de vue du critère  $\phi_p(D)$ , puisque toutes les possibilités n'ont pas été explorées. Cependant, cet algorithme fournit un échantillonnage représentatif de l'espace de conception (voir figure 3.6).

Dans le cadre de la construction de métamodèles, il est également essentiel de déterminer le nombre initial de points totalement convergés  $\mathbf{X}_{fcv}$  à utiliser. Le nombre de points partiellement convergés  $\mathbf{X}_{pcv}$  a été fixé grâce à l'étude menée précédemment au chapitre 2 à  $10 \times n_{dim}$ . Dans le cadre de cette étude et de ce chapitre, le nombre initial de points des données totalement convergées à calculer est aussi un élément important pour construire des métamodèles de cokrigage et de krigeage hiérarchique. Une étude de variabilité de ce nombre de points (de  $2 \times n_{dim}$  et  $5 \times n_{dim}$  points) a alors été menée sur les cas test mécaniques.

---

1. cours de Bertrand Iooss [http://www.gdr-mascotnum.fr/media/cours2\\_unice.pdf](http://www.gdr-mascotnum.fr/media/cours2_unice.pdf)

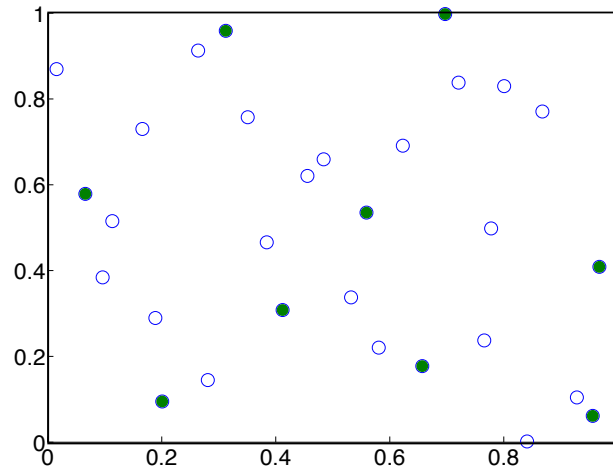
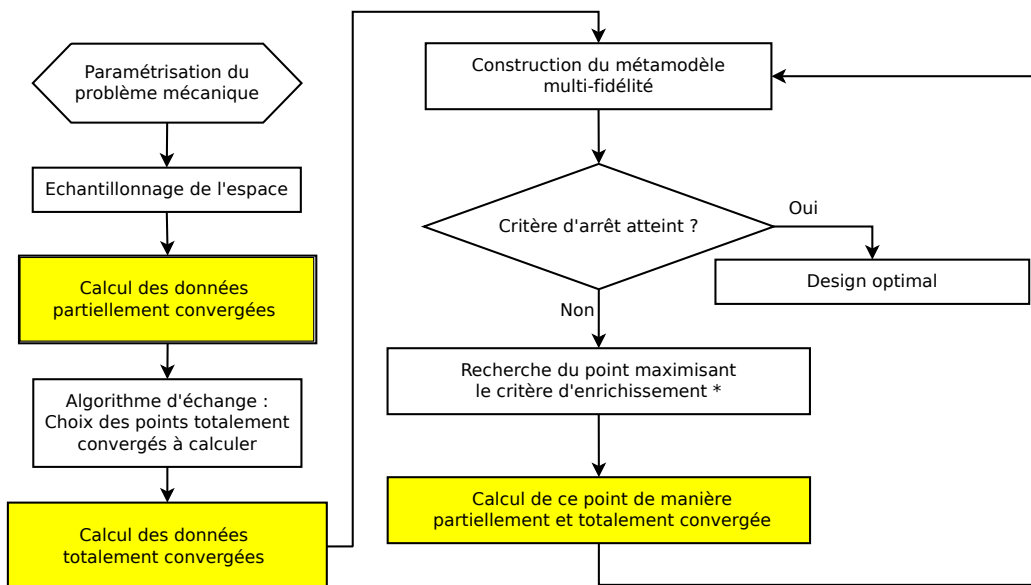


FIGURE 3.6: Illustration d'un tirage LHS de 30 points et d'une sélection de 9 points par l'algorithme d'échange

## 2.3 Stratégie de recalage avec cokrigage et krigeage hiérarchique

### 2.3.1 Présentation de l'algorithme

L'algorithme décrit en figure 3.7 explique comment la stratégie de recalage a été mise en place pour les méthodes de cokrigage et de krigeage hiérarchique. L'enrichissement de l'échantillonnage initial se fait au maximum de l'erreur quadratique  $s_{ym}^2$  des métamodèles.



\* Dans le cadre de la stratégie de recalage, le critère d'enrichissement est le maximum d'erreur quadratique moyenne du métamodèle. Dans le cadre de la stratégie de recherche du minimum, le critère d'enrichissement est le maximum d'expected improvement.

FIGURE 3.7: Algorithme associé aux stratégies de cokrigage et de krigeage hiérarchique

**Remarque :** Une stratégie de recherche de minimum par EGO serait du même type excepté

que le critère d'enrichissement de l'échantillonnage de l'espace se ferait au maximum d'EI (voir chapitre 4).

### 2.3.2 Critères d'arrêt d'enrichissement

Comme dit lors de l'évocation des points de comparaison des méthodes multi-fidélité, deux critères d'arrêt seront utilisés pour stopper la stratégie de recalage. Le premier consiste tout simplement à stopper l'enrichissement lorsque le coefficient de corrélation de concordance entre un métamodèle de référence et le métamodèle multi-fidélité est supérieur à 0.95 ( $r_{ccc} > 0.95$ ). Le second critère d'arrêt ne nécessite pas l'utilisation d'un métamodèle de référence qui n'est *a priori* pas disponible. On se propose de s'inspirer du critère d'arrêt d'enrichissement proposé par [Forrester *et al.*, 2006a, Forrester, 2004]. Celui-ci consiste à venir calculer la moyenne des corrélations entre les  $\mu$  derniers métamodèles construits et à arrêter l'enrichissement lorsque celle-ci a atteint un certain niveau.

On définit alors le critère de corrélation entre le dernier métamodèle construit  $\hat{y}$  et le  $\mu^{\text{ème}}$  précédent  $\hat{y}_\mu$ , par :

$$r_{ccc\mu} = \frac{2 \cdot \sigma_{\hat{y}_\mu \hat{y}}}{\sigma_{\hat{y}_\mu}^2 + \sigma_{\hat{y}}^2 + (\bar{\hat{y}} - \bar{\hat{y}}_\mu)^2} \quad (3.81)$$

**Remarques :** Forrester utilise dans le cadre d'une stratégie de recherche d'optimum le critère  $r^2$  plutôt que  $r_{ccc}$ . De plus, comme écrit dans [Forrester *et al.*, 2006a, Forrester, 2004], ce critère peut être vu comme similaire au critère de validation croisée de type « leave-one-out » excepté que dans ce cas, la validation se fait par ajout de points et non par retrait.

On calcule alors le critère  $\bar{r}_{ccc}^v$  qui est la moyenne des critères  $r_{ccc\mu}$  calculés.

$$\bar{r}_{ccc}^v = \frac{1}{v} \sum_{\mu=1}^v r_{ccc\mu} \quad (3.82)$$

On fixe  $v = 3$  (selon [Forrester *et al.*, 2006a, Forrester, 2004]). L'enrichissement est stoppé lorsque  $\bar{r}_{ccc}^v$  devient supérieur trois fois de suite à 0.99. Le critère d'arrêt est alors noté  $\bar{r}_{ccc}^{v^3}$ . Le fait de s'assurer trois fois de suite  $\bar{r}_{ccc}^v > 0.99$  est juste utilisé comme sécurité, afin de garantir au mieux la qualité finale du métamodèle. Si le métamodèle multi-fidélité construit est réellement stabilisé, le coût de calcul supplémentaire associé à cette sécurité est faible.

L'ensemble des ingrédients nécessaires à l'élaboration de la comparaison de l'efficacité des métamodèles multi-fidélité a donc été présenté. La suite du chapitre se concentre à présenter les résultats de cette étude de comparaison sur la qualité des métamodèles obtenus et sur le temps de calcul nécessaire pour y aboutir. Comme précédemment, le temps de calcul présenté est le temps de calcul associé aux simulations numériques, celui-ci étant prépondérant (dans un cadre industriel) sur les autres temps de calcul. Afin de comparer les différentes méthodes de métamodélisation, celles-ci ont été appliquées sur deux cas test. Le premier d'entre eux est le cas test *cubes* précédemment utilisé dans le chapitre 2.



## 2.4 Etude du cas test *cubes*

Pour chaque méthode multi-fidélité testée, une série de 20 tirages initiaux de points partiellement convergés a été mise en place. Les différentes études ont également été menées pour différents nombres de points initiaux totalement convergés. Pour rappel, on s'intéresse dans cette section aux 4 aspects de qualité et de temps de calcul décrits au paragraphe 2.1. L'étude complète a été réalisée pour un nombre initial de points de données totalement convergées  $\mathbf{X}_{fcv}$  variant de  $2 \times n_{dim}$  à  $5 \times n_{dim}$ .

Pour des raisons d'allégement de la présentation de ce travail, ce comparatif n'est pas présenté. Il montre, en effet, un intérêt relativement limité. On a pu cependant en déduire que le nombre initial de points totalement convergés avait une influence sur la qualité du premier métamodèle multi-fidélité construit. Résultat, somme toute, assez logique. Cependant, il a été également constaté que pour atteindre les différents critères d'arrêt fixés, il n'était pas nécessaire de disposer initialement de beaucoup de points totalement convergés. Ainsi, il a été choisi d'utiliser un nombre initial de points  $\mathbf{X}_{fcv}$  de  $3 \times n_{dim}$ . Ce choix offre un bon compromis entre qualité du métamodèle et temps de simulation des données nécessaires à l'obtention de celui-ci. [Huang *et al.*, 2006a] utilise également le même nombre de points initiaux haute fidélité dans une approche de cokrigage. Le choix de ce nombre de points est donc en accord avec ce qui peut se faire dans la littérature.

On s'intéresse également à l'influence que peut avoir le niveau de l'indicateur d'erreur LaTin  $\eta$  pour le calcul des données partiellement convergées. L'étude est réalisée pour une variation de  $\eta$  allant de 0.05 à 0.01.

L'étude complète est présentée en cinq paragraphes.

- Le premier consiste à fournir la qualité des données partiellement convergées. Pour cela, on présente la corrélation moyenne (sur les 20 tirages initiaux effectués)  $r_{ccc}$  et  $r^2$  entre le métamodèle de référence et le métamodèle de krigage des données partiellement convergées, ainsi que les indicateurs  $Q3$  et  $Q4$ .
- Le deuxième paragraphe présente la qualité moyenne des métamodèles multi-fidélité construits initialement pour  $10 \times n_{dim}$  points  $\mathbf{X}_{pcv}$  LHS et  $3 \times n_{dim}$  points  $\mathbf{X}_{fcv}$  (ces derniers étant, soit issus d'un enrichissement de l'échantillonnage initial pour la stratégie d'Evofusion, soit de l'algorithme d'échange pour les méthodes de cokrigage et de krigage hiérarchique).
- Le troisième paragraphe illustre en fonction du temps de calcul et du nombre de points totalement convergés calculés, la qualité des métamodèles multi-fidélité obtenus.
- Le quatrième paragraphe donne le temps de calcul et le nombre de points moyens nécessaires à la stratégie de recalage pour construire des métamodèles ayant un certain niveau de qualité, à savoir  $r_{ccc} > 0.95$ . **Les calculs sont effectués à l'aide de la stratégie LaTin multiparamétrique.**
- Le dernier paragraphe fournit le même genre d'information que le précédent mais concernant le critère d'arrêt  $\bar{r}_{ccc}^3$ .

**De plus, l'ensemble des résultats proposés est comparé à ceux que fournit un**

**métamodèle de krigeage des points  $X_{fcv}$  utilisés au cours de cette étude.** Le but est d'illustrer l'intérêt des méthodes de modélisation multi-fidélité par rapport au krigeage.

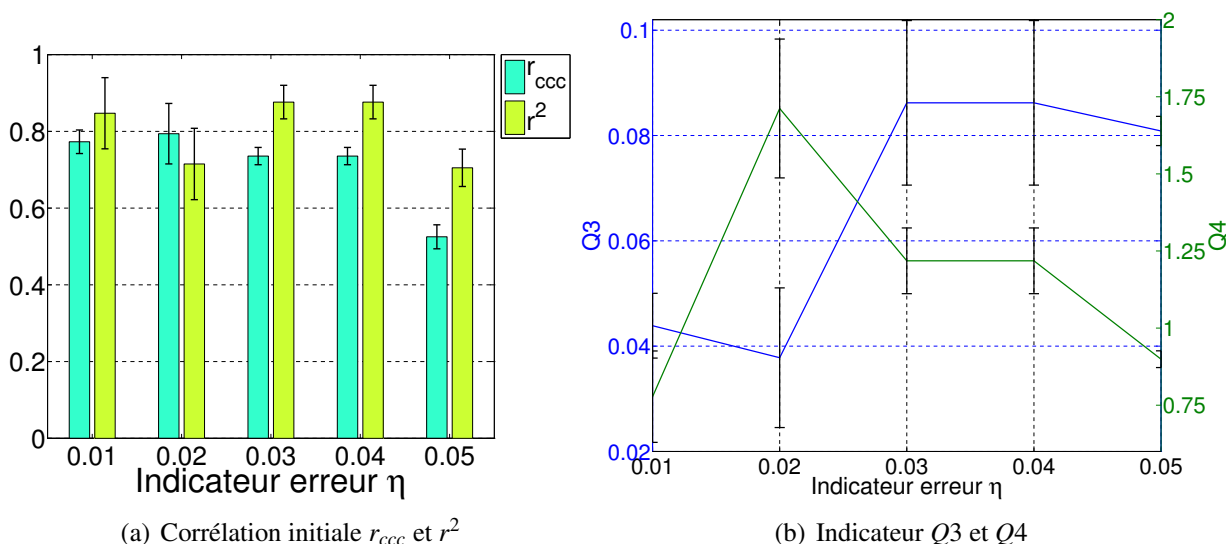
L'ensemble des métamodèles est calculé grâce à la toolbox GRENAT.

**Remarques :** Les différentes abréviations utilisées pour la présentation des résultats sont :

- Ckrg. Inter. pour Cokrigeage Intercorrélation
- Ckrg.  $s_f/s_p$  pour Cokrigeage  $s_{fcv}/s_{pcv}$
- Ckrg. Reg. pour le Cokrigeage s'appuyant sur une approche auto-régressive.
- Krg. Hier. pour krigeage hiérarchique
- fcv pour le métamodèle de krigeage s'appuyant uniquement sur les données totalement convergées

### 2.4.1 Qualité du métamodèle construit avec des données uniquement partiellement convergées

On présente dans ce paragraphe la qualité moyenne (sur 20 essais numériques pour chaque niveau de convergence  $\eta$ ) du métamodèle construit uniquement avec les données partiellement convergées. On présente les niveaux de corrélation moyen  $r_{ccc}$  et  $r^2$  entre ce métamodèle et le métamodèle de référence. Les coefficients  $Q3$  et  $Q4$  sont également calculés. Ce dernier indicateur a en effet montré au chapitre précédent une importance quant aux nombres de points à ajouter afin d'obtenir un métamodèle multi-fidélité d'une bonne qualité.



**FIGURE 3.8:** Qualité moyenne des métamodèles construits avec les données partiellement convergées pour le cas test *cubes*

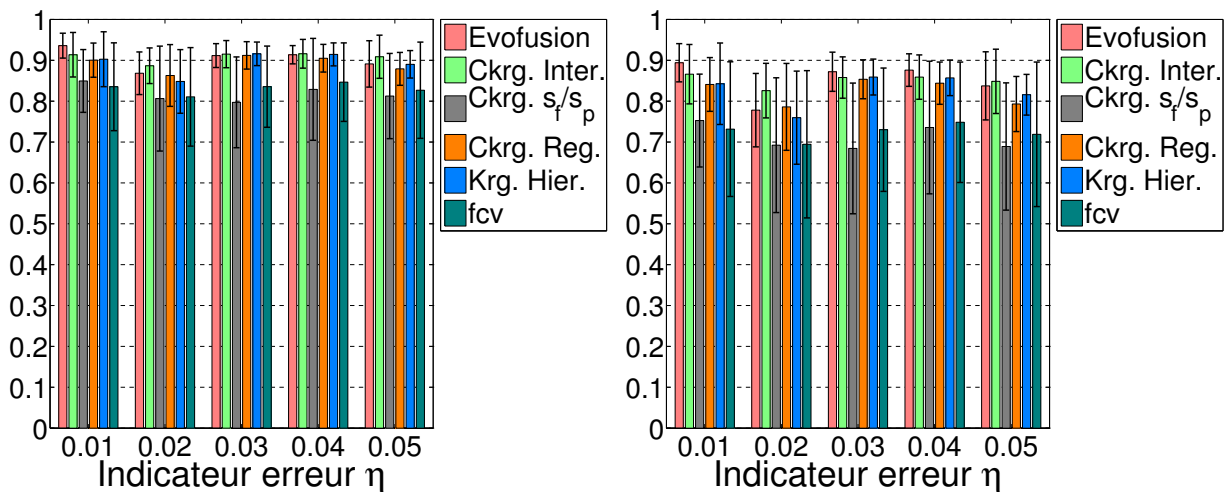
On peut constater à la figure 3.8 une qualité variable en fonction de l'indicateur d'erreur  $\eta$

des données partiellement convergées. On remarque notamment que pour le niveau  $\eta = 0.02$  l'indicateur  $Q4$  est maximal. (Résultat obtenu également avec l'utilisation de la toolbox DACE lors du chapitre précédent voir tableau 2.3). Le niveau de convergence semblant offrir les meilleurs indicateurs de qualité (corrélations  $r_{ccc}$  et  $r^2$  élevées, indicateurs  $Q3$  et  $Q4$  faibles) est le niveau  $\eta = 0.01$ . Le niveau de convergence  $\eta = 0.05$ , *a priori* désigné par l'étude préliminaire du chapitre précédent comme un bon niveau de convergence des données partiellement convergées, semble être un peu en retrait. Mais l'on avait pu remarquer lors de cette même étude préliminaire qu'il s'était avéré au final très efficace. On constatera dans la suite de ce travail qu'il l'est effectivement.

### 2.4.2 Qualité du métamodèle multi-fidélité initial

Dans ce paragraphe, on s'intéresse à la qualité du métamodèle multi-fidélité initial construit : c'est-à-dire le métamodèle construit avec initialement  $10 \times n_{dim}$  points  $\mathbf{X}_{pcv}$  et  $3 \times n_{dim}$  points  $\mathbf{X}_{fcv}$ .

On présente à la figure 3.9 les différentes corrélations moyennes  $r_{ccc}$  et  $r^2$  entre les métamodèles multi-fidélité et le métamodèle de référence. On présente également la qualité de la réponse obtenue par krigeage des données totalement convergées (celles-ci étant celles utilisées pour construire les métamodèles de cokrigeage et de krigeage hiérarchique).



(a) Corrélation  $r_{ccc}$  entre le métamodèle multi-fidélité et le métamodèle de référence

(b) Corrélation  $r^2$  entre le métamodèle multi-fidélité et le métamodèle de référence

**FIGURE 3.9:** Qualité des métamodèles multi-fidélité pour le 1<sup>er</sup> cas test

Les figures 3.9 permettent de constater que :

- une amélioration significative est obtenue par rapport au métamodèle construit uniquement avec les données partiellement convergées
- l'utilisation unique des données totalement convergées par krigeage (méthode fcv) offre une qualité moindre

- une méthode semble offrir moins de garantie quant à la qualité du métamodèle obtenu (méthode de cokrigeage  $s_{fcv}/s_{pcv}$ )
- toutes les autres méthodes multi-fidélité sont des méthodes gagnantes par rapport au krigage.
- le niveau de convergence  $\eta = 0.05$  des données partiellement convergées permet d'atteindre une qualité semblable à celle obtenue avec  $\eta = 0.01$  (celle-ci étant cependant légèrement meilleure)

L'étape suivante est de comparer la qualité des métamodèles en fonction du temps de calcul et du nombre de points totalement calculés lors de la phase d'enrichissement de la stratégie de recalage.

### 2.4.3 Qualité des métamodèles multi-fidélité lors de la phase d'enrichissement

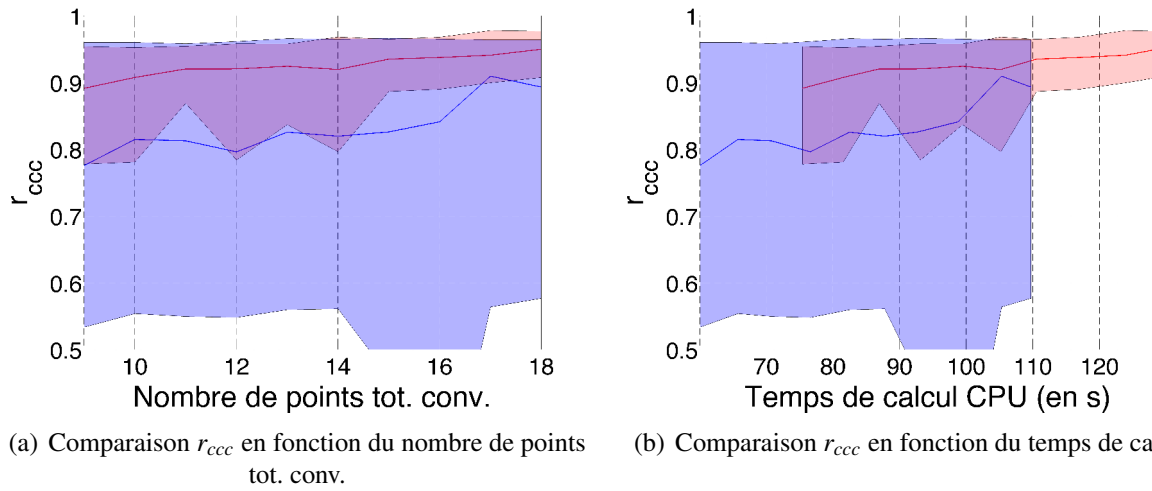
On s'intéresse ici à la qualité des métamodèles obtenus en fonction du nombre de points totalement convergés calculés et du temps de calcul associé lors de la phase d'enrichissement. On compare sur les différentes figures 3.10 à 3.14 la qualité des métamodèles multi-fidélité (en rouge sur la figure) par rapport aux métamodèles de krigage construits uniquement avec les données totalement convergées disponibles (en bleu sur la figure). On compare les corrélations moyennes obtenues entre les deux types de métamodèles, ainsi que l'étendue minimum et maximum des niveaux de corrélation atteints par les deux méthodes lors des 20 essais numériques (zones de couleur rouge et bleue sur les prochaines figures).

La comparaison de qualité s'effectue en fonction du nombre de points totalement convergés calculés (variant de 9 à 18) et du coût de calcul associé. La comparaison par rapport aux nombres de points totalement convergés permet de mettre en valeur l'apport des données basse fidélité des métamodèles multi-fidélité par rapport aux métamodèles de krigage des seules données haute fidélité. La comparaison par rapport au temps de calcul permet de vérifier que celui des données basse fidélité n'est pas trop handicapant pour mettre en place ces stratégies multi-fidélité.

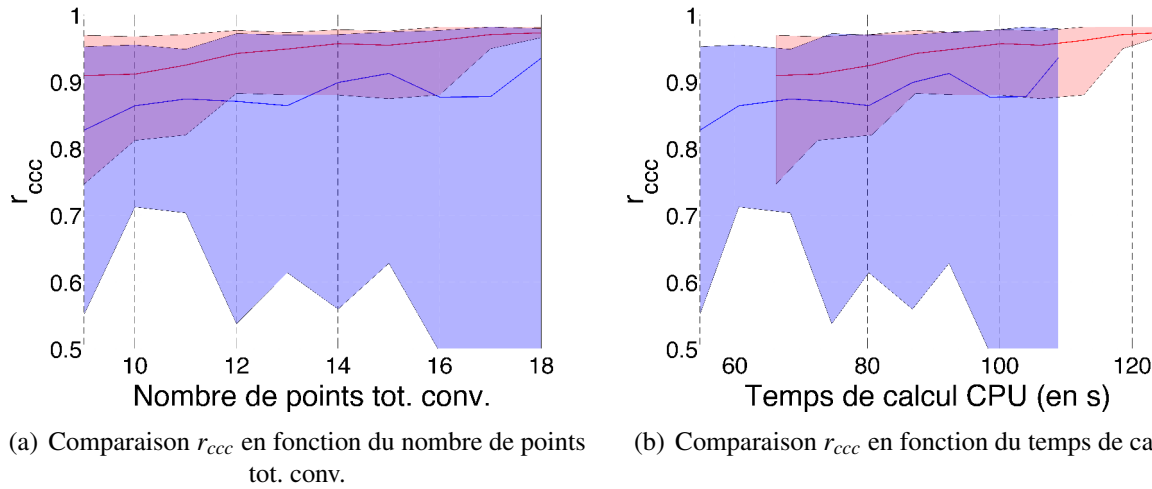
Par la suite, on compare les méthodes multi-fidélité entre elle afin de déterminer lesquelles fournissent les meilleurs métamodèles. Cette étude est menée pour un niveau de convergence  $\eta = 0.05$  des données partiellement convergées.

#### Résultats pour l'Evofusion et pour le Cokrigeage Intercorrélation

Les méthodes d'Evofusion et de cokrigeage Intercorrélation (figures 3.10, 3.11) apportent un vrai bénéfice de qualité par rapport aux métamodèles de krigage calculés avec des données totalement convergées. L'apport de points partiellement convergés est incontestable. Par rapport au temps de calcul, la différence de qualité entre les deux types de métamodèles (multi-fidélité et krigage) se réduit logiquement dû au coût de calcul des données partiellement convergées. On constate cependant largement l'intérêt d'utiliser ces méthodes multi-fidélité. Celles-ci sont également intéressantes du point de vue de la robustesse des résultats obtenus. En effet, la zone d'étendue minimale et maximale des corrélations atteintes (en rouge) est plus réduite que celle des métamodèles de krigage des données totalement convergées (en bleue).



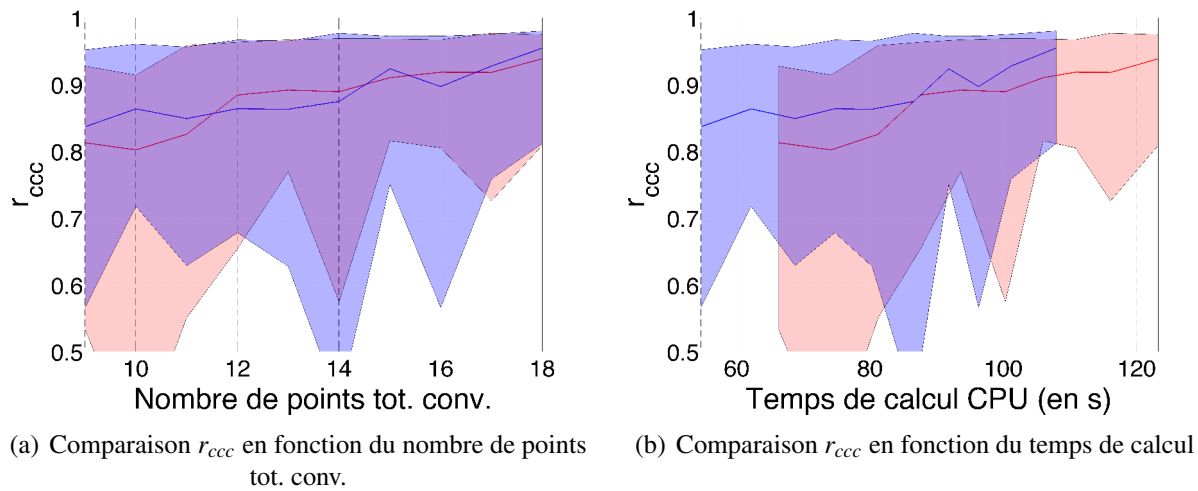
**FIGURE 3.10:** Comparaison de la qualité des métamodèles avec l'approche d'Evofusion (rouge) par rapport à celle obtenue avec les métamodèles de krigeage construits uniquement avec les données totalement convergées (bleu)



**FIGURE 3.11:** Comparaison de la qualité des métamodèles avec l'approche de Cokrigeage Intercorrélation (en rouge) par rapport à celle obtenue avec les métamodèles de krigeage construits uniquement avec les données totalement convergées (en bleu)

### Résultats pour le Cokrigeage $s_{fcv}/s_{pcv}$

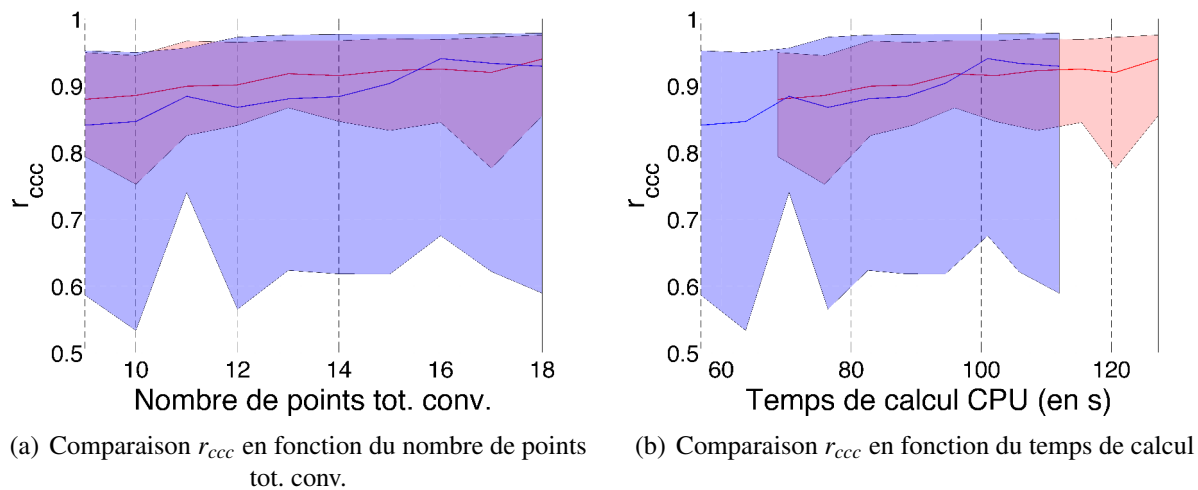
La méthode de cokrigeage  $s_{fcv}/s_{pcv}$  fournit comme précédemment les résultats les moins intéressants. On constate qu'à nombre égal de points totalement convergés, la qualité des métamodèles multi-fidélité et de krigeage sont semblables. Cependant, on remarque qu'en fonction du temps de calcul cette stratégie multi-fidélité est une stratégie perdante. Cette méthode



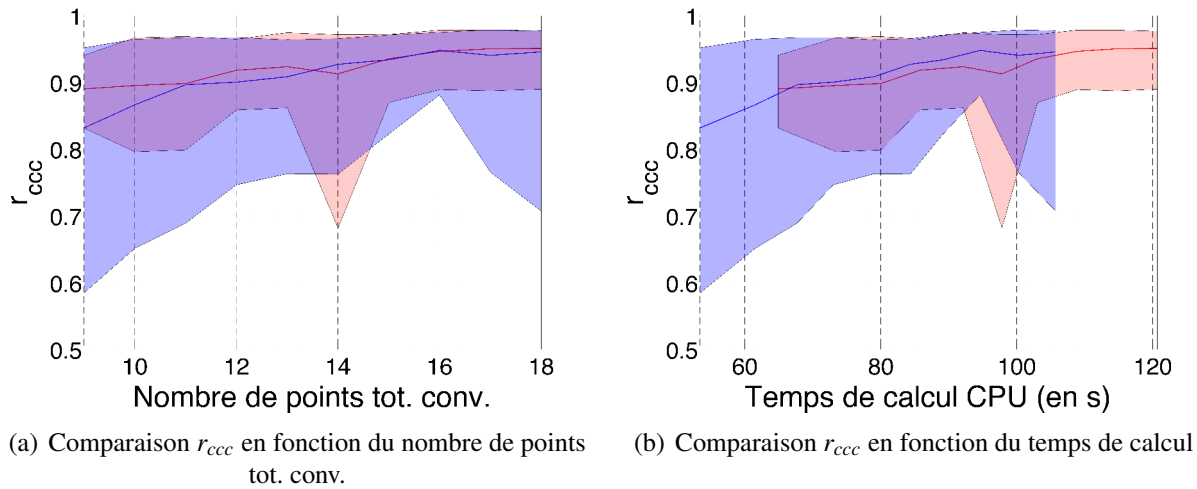
**FIGURE 3.12:** Comparaison de la qualité des métamodèles avec l’approche de Cokrigage  $s_{fcv}/s_{pcv}$  (en rouge) par rapport à celle obtenue avec les métamodèles de krigage construits uniquement avec les données totalement convergées (en bleu)

n’offre pas non plus de meilleure garantie de robustesse. Les zones de corrélation mini-maxi atteintes sont de même amplitude pour le cokrigage ou le krigage. L’apport de cette méthode est donc nul sur ce cas test.

**Résultats pour le cokrigage de type auto-régressif et le krigage hiérarchique**



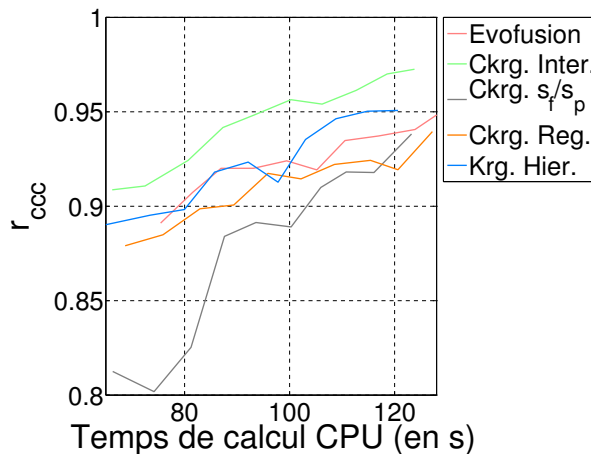
**FIGURE 3.13:** Comparaison de la qualité des métamodèles avec l’approche de cokrigage auto-régressif (en rouge) par rapport à celle obtenue avec les métamodèles de krigage construits uniquement avec les données totalement convergées (en bleu)



**FIGURE 3.14:** Comparaison de la qualité des métamodèles avec l'approche de krigeage hiérarchique (en rouge) par rapport à celle obtenue avec les métamodèles de krigeage construits uniquement avec les données totalement convergées (en bleu)

Le cokrigeage de type auto-régressif et le krigeage hiérarchique sont moins performants que l'Evofusion et que le cokrigeage Intercorrélaton. En effet, la qualité moyenne obtenue par ces méthodes est assez semblable à celle obtenue par le krigeage. Cependant, ce sont des méthodes offrant une plus grande robustesse de résultats, la zone de corrélation mini-maxi atteinte étant moins large que celle obtenue pour des métamodèles de krigeage (voir figures 3.13 et 3.14).

### Comparaison directe des méthodes multi-fidélité



**FIGURE 3.15:** Comparaison  $r_{ccc}$  pour les différentes méthodes

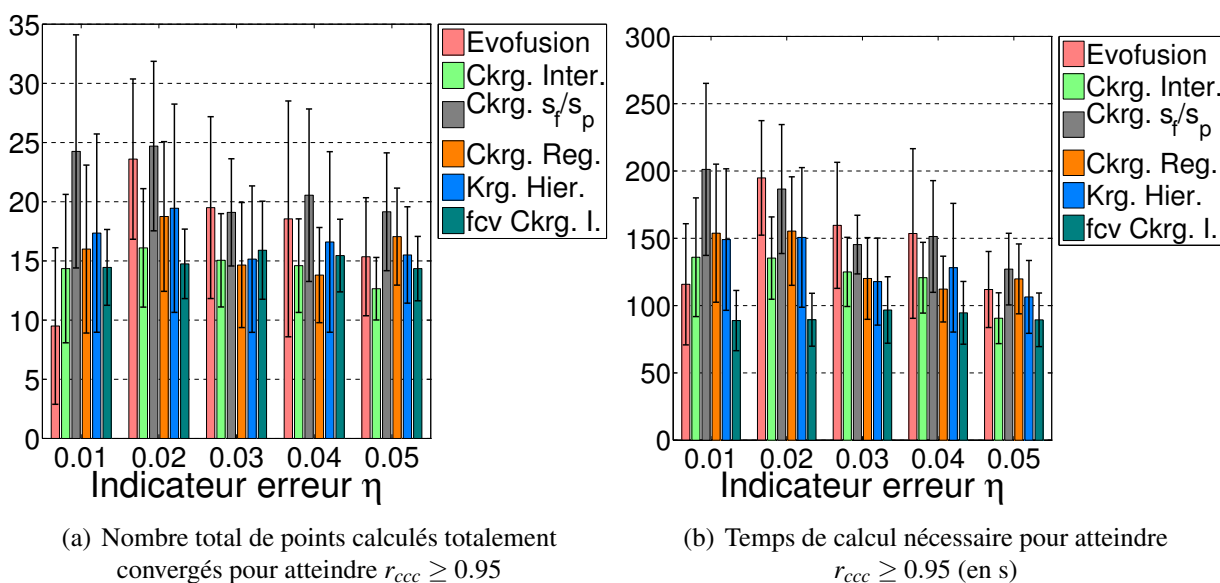
Sur la figure 3.15, on représente la qualité moyenne des différents métamodèles multi-fidélité obtenus en fonction du coût de calcul. On constate clairement que la méthode de cokrigeage

Intercorrélation est la plus performante sur cet exemple. Viennent ensuite la méthode de krigeage hiérarchique et celle d'Evofusion. Les deux autres méthodes de cokrigeage sont les méthodes les moins performantes. Ceci s'explique par la difficulté de déterminer les paramètres multiplicatifs liés aux données partiellement convergées, sur des cas test réels. On retombe sur la conclusion qui a pu être faite dans [Zimmermann et Han, 2010] qui met en avant que la détermination de ces paramètres peut s'avérer complexe sur des cas test où la relation entre modèle basse et haute fidélité n'est pas établie de manière simple.

### 2.4.4 Comparaison des méthodes multi-fidélité pour atteindre $r_{ccc} > 0.95$

Dans le paragraphe précédent, nous avons comparé la qualité des différents métamodèles multi-fidélité en fonction du temps de calcul. Dans ce paragraphe, on se propose de comparer les temps de calcul nécessaires aux méthodes multi-fidélité pour atteindre un certain niveau de qualité par rapport au métamodèle de référence. Ceci dans le but de trouver les méthodes les plus adaptées pour construire un abaque virtuel. On considère ce critère atteint lorsque la corrélation entre le métamodèle multi-fidélité et le métamodèle de référence est supérieur à 0.95. Cette étude permet donc de déterminer quelles sont les méthodes les plus performantes pour obtenir un certain niveau de qualité.

De plus, à titre informatif, on fournit les résultats obtenus si l'on utilisait uniquement par krigeage les données totalement convergées. Les données utilisées pour cette étude sont celles provenant du cokrigeage Intercorrélation (notée fcv Ckrg I. sur les prochaines figures). **Des résultats tout à fait semblables sont obtenus si l'on choisit les données totalement convergées des autres méthodes.**



**FIGURE 3.16:** Comparaison des méthodes pour atteindre le critère de corrélation  $r_{ccc} \geq 0.95$  pour le cas test *cubes*

Plusieurs choses sont à souligner suite aux résultats présentés aux figures 3.16 :



- On peut observer une certaine disparité des résultats obtenus en fonction du niveau de convergence des données partiellement convergées et de l'approche multi-fidélité utilisée. Ainsi, on constate que pour l'EvoFusion, le niveau  $\eta = 0.02$  amène à calculer un plus grand nombre de points pour atteindre le critère d'arrêt. Cette conclusion est en accord avec ce qui a été trouvé au chapitre 2. L'explication tient notamment au fait que pour ce niveau de convergence le métamodèle de données partiellement convergées possède un indicateur  $Q4$  plus élevé que pour les autres niveaux. On constate cependant que par rapport au précédent chapitre, moins de points sont nécessaires pour obtenir la qualité de métamodèle souhaitée. Cette différence de performance tient au fait que l'optimisation de la fonction de vraisemblance a été effectuée à l'aide d'un algorithme d'optimisation locale sous la toolbox DACE, alors que sous GREMAT, cette optimisation a été effectuée par une stratégie d'optimisation globale (algorithme génétique).
- On remarque que pour l'ensemble des méthodes, un nombre plus important de points est nécessaire pour atteindre le critère d'arrêt lorsque le niveau de convergence des données partiellement convergées est  $\eta = 0.02$ . Cependant, l'effet n'en est pas nécessairement aussi marqué que pour l'EvoFusion. L'ensemble des méthodes est donc sensible à ce niveau de convergence.
- Le cokrigage Intercorrélation est sur ce cas test l'une des meilleures stratégies. Elle offre un temps de calcul minimal pour atteindre le critère d'arrêt tout en étant relativement stable suivant les niveaux de convergence des données partiellement convergées.
- Le cokrigage  $s_{fcv}/s_{pcv}$  fournit des résultats relativement erratiques en fonction du niveau de convergence des données partiellement convergées. De façon globale, il s'agit de la méthode la moins performante.
- Un résultat important de cette étude est que la seule utilisation des données totalement convergées permet d'obtenir rapidement un métamodèle de krigeage répondant au critère de qualité exigée. Seule la stratégie de cokrigage Intercorrélation utilisant des données partiellement convergées au niveau  $\eta = 0.05$  est légèrement plus performante. Plusieurs explications peuvent être fournies. Tout d'abord la simplicité de la fonction objectif (elle peut être approximée par des hyperplans). Cette simplicité à représenter efficacement la fonction objectif justifie qu'en un nombre relativement faible de points ( $\approx 15$  en moyenne pour cette étude voir figure 3.16 (a)) un métamodèle de bonne qualité peut être trouvé. De plus, l'enrichissement lors de la phase de recalage s'effectuant au maximum d'erreur quadratique moyenne, il a été constaté que celui-ci s'effectue principalement sur la frontière du domaine d'étude. Or, la taille du problème étant relativement faible ( $n_{dim} = 3$ ) et le nombre d'hyperplans réduit, cet enrichissement suffit à fournir une représentation de suffisamment bonne qualité. On peut également dire que le tirage des données totalement convergées est issu d'un enrichissement intelligemment mené en vue de construire un métamodèle précis. Les points utilisés pour construire le métamodèle de krigeage ont donc un certain sens en vue de la création d'un abaque virtuel. Et bien sûr, il n'y a pas besoin de tenir compte du temps de calcul des données partiellement convergées puisqu'il n'y en a pas.

On peut donc en conclure que la méthode multi-fidélité la plus adaptée pour fournir un métamodèle d'un certain niveau de qualité est la méthode de cokrigage Intercorrélation. Celle-ci s'avère être en moyenne la plus performante et également la plus robuste vis-à-vis des niveaux de convergence des données partiellement convergées. La méthode de cokrigage  $s_{fcv}/s_{pcv}$  est la moins performante de toutes. Ce qui corrobore ce qui a pu être constaté précédemment. On peut également remarquer que le niveau de convergence  $\eta = 0.05$  est effectivement bien approprié pour construire un métamodèle multi-fidélité de bonne qualité (à savoir  $r_{ccc} > 0.95$ ).

On note également que cette fois-ci l'apport des stratégies multi-fidélité n'est pas aussi clair que précédemment. En effet, le krigeage des données totalement convergées permet d'obtenir rapidement un métamodèle répondant à la qualité exigée. Il faut cependant garder en mémoire que le tirage de points effectué l'a été intelligemment.

On a pu voir sur ce cas test, qu'à **coût de calcul fixé**, un métamodèle de krigeage était moins performant (voir les deux paragraphes précédents) que des approches multi-fidélité. Cependant, à **qualité exigée**, ceci n'est pas nécessairement vérifié.

### 2.4.5 Comparaison des méthodes multi-fidélité pour atteindre $\bar{r}_{ccc}^{v^3}$

Afin de créer un abaque virtuel sans disposer de moyens extérieurs pour vérifier la qualité obtenue, on se propose d'effectuer la stratégie de recalage jusqu'à atteindre le critère d'arrêt  $\bar{r}_{ccc}^{v^3}$ . On compare donc le temps de calcul et le nombre de points totalement convergés nécessaires à chacune des méthodes pour parvenir à ce critère (voir figure 3.17). La qualité moyenne obtenue des métamodèles est également présentée pour vérifier si ce critère d'arrêt est bien adapté. On fournit, comme précédemment, les résultats obtenus en ne considérant uniquement que les données totalement convergées (utilisées pour la stratégie de cokrigage Intercorrélation).

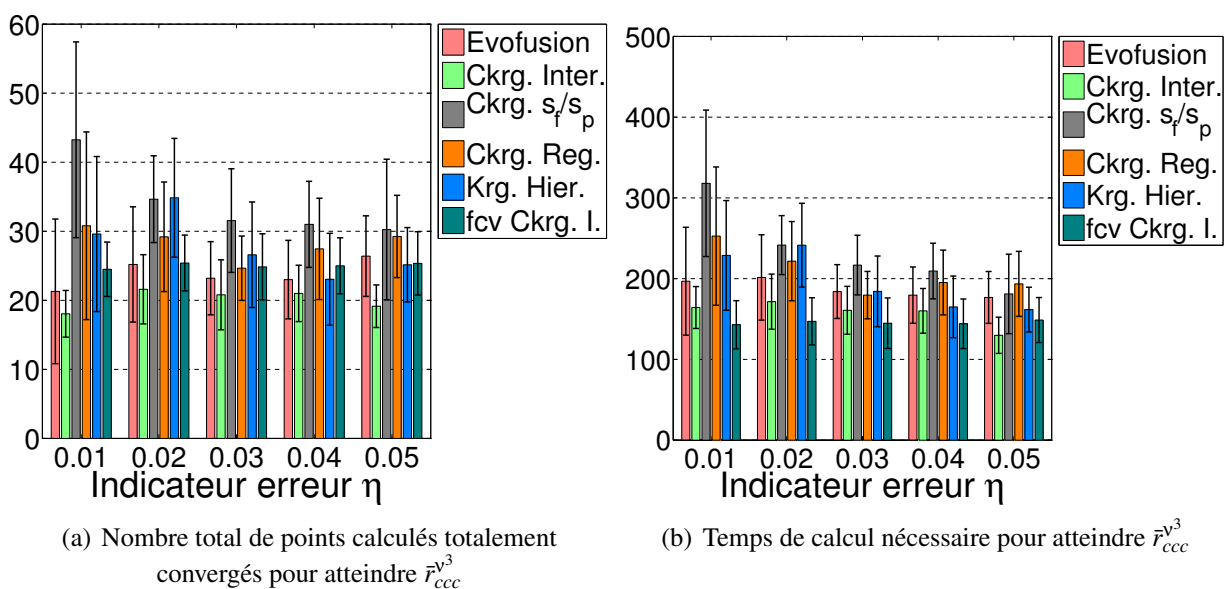
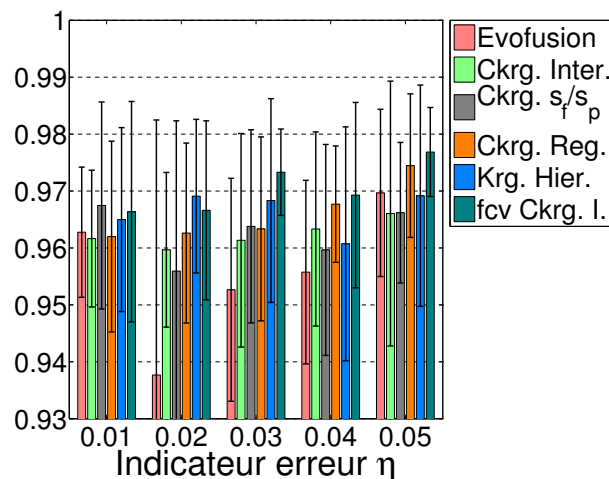


FIGURE 3.17: Comparaison des méthodes pour atteindre le critère d'arrêt  $\bar{r}_{ccc}^{v^3}$  pour le cas test *cubes*

On peut tirer des conclusions tout à fait semblables à celles du paragraphe précédent, à savoir :

- La méthode de cokrigage Intercorrélation est la méthode la plus rapide pour atteindre le critère d'arrêt de l'ensemble des méthodes multi-fidélité. Pour un faible taux de convergence ( $\eta = 0.05$ ), il s'agit même de la stratégie la plus performante.
- La seule utilisation des données totalement convergées permet également d'atteindre rapidement le critère d'arrêt  $\bar{r}_{ccc}^{v^3}$ .
- La méthode de cokrigage  $s_{fcv}/s_{pcv}$  est de nouveau la méthode la moins performante de celles testées.
- On remarque que les quatre autres méthodes multi-fidélité n'ont pas besoin au total et en moyenne de plus de 30 points calculés totalement convergés pour atteindre ce critère d'arrêt (notamment pour le niveau  $\eta = 0.05$ ). Certes, il faut tenir compte du temps de calcul des données partiellement convergées mais ceci tend à montrer que l'on peut efficacement trouver une surface de réponse de qualité en calculant moins de  $10 \times n_{dim}$  points totalement convergés (règle heuristique de référence) et ainsi potentiellement réduire les temps de calcul par rapport à une approche conventionnelle.

A titre informatif, on fournit le niveau de corrélation moyen entre le métamodèle multi-fidélité calculé et le métamodèle de référence à la figure 3.18. On constate une certaine homogénéité des résultats. Les écarts entre les différentes méthodes sont relativement faibles, excepté peut-être pour l'Evofusion utilisant des données partiellement convergées au niveau  $\eta = 0.02$ . Cependant, le niveau de corrélation atteint peut être jugé satisfaisant puisqu'il dépasse généralement 0.96.



**FIGURE 3.18:** Corrélation  $r_{ccc}$  obtenue entre les métamodèles multi-fidélité finaux et le métamodèle de référence

### 2.4.6 Conclusion de l'étude

Le but de cette étude était de trouver quelles étaient les méthodes multi-fidélité les plus adaptées à la création de métamodèles sur un cas test réel de mécanique des assemblages. Sur ce cas test, quatre méthodes se sont montrées relativement probantes : le cokrigeage Inter-corrélation, l'Evofusion, le krigeage hiérarchique et le cokrigeage de type auto-régressif. Elles ont permis l'obtention de métamodèles de qualité pour des temps de calcul réduits. Elles ont montré également leur efficacité quant à construire un métamodèle avec un certain niveau de qualité exigée. La méthode de cokrigeage  $s_{fcv}/s_{pcv}$  s'est montrée être la méthode la moins performante. Elle s'avère être une méthode inefficace sur ce cas test et peut même se montrer handicapante par rapport à une stratégie de krigeage (voir figure 3.12). La principale raison venant du fait que l'estimateur  $\sigma_{fcv}/\sigma_{pcv}$  est difficile à trouver. On constate donc sur un cas test réel que cette méthode n'est pas aussi efficace que sur le cas test analytique précédemment utilisé.

On a également pu remarquer que le niveau de convergence  $\eta = 0.05$  des données partiellement convergées était un niveau de convergence efficace tout au long de cette étude et pour l'ensemble des méthodes multi-fidélité testées.

Ce que l'on peut principalement retenir de cette première étude est qu'**à coût de calcul fixé**, les quatre approches multi-fidélité précédemment citées s'avèrent être en moyenne plus performantes et plus robustes qu'une simple approche par krigeage des données totalement convergées. Par contre à **qualité exigée**, ce n'est pas nécessairement le cas. Il faut cependant garder en mémoire qu'il suffit sur ce cas test de relativement peu de points pour trouver une surface de qualité et que les points ayant servi à la construction du métamodèle de krigeage ont été choisis intelligemment.

Afin d'infirmier ou confirmer les résultats précédemment obtenus, une étude sur un second cas test mécanique est menée. La fonction objectif est notamment multimodale et demande, par rapport aux nombres de paramètres le composant, plus de points pour la représenter correctement par un métamodèle.

## 2.5 Etude du cas test 1 *rainure*

Ce second cas test est un cas test mécanique 2D, constitué de deux paramètres. Bien que réduit en terme de paramètres par rapport au précédent, il offre l'avantage d'être multimodal.

Ce cas test introduit dans [Buytet *et al.*, 2006] et utilisé par [Laurent, 2013] dans le cadre de la construction de métamodèles à gradients, est schématisé en figure 3.19. Il consiste à venir appuyer sur une première structure de manière dissymétrique et à mesurer la quantité :

$$f_{obj} = y(\mathbf{x}) = \frac{P_{max} - P_{min}}{P_{moy}} \quad (3.83)$$

où  $P_{max}$ ,  $P_{min}$  et  $P_{moy}$  sont respectivement la pression maximale, la pression minimale et la pression moyenne mesurées au niveau de la ligne pointillée de la figure 3.19.

La première structure comporte un défaut localisé en position  $X$ , ayant une largeur  $L$ , et une hauteur  $h$  fixée à 0.25mm. Le coefficient de frottement à l'interface des deux structures est noté

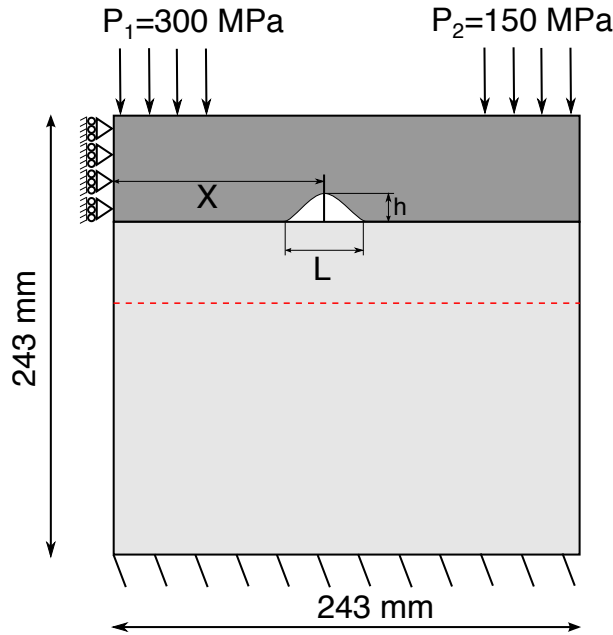


FIGURE 3.19: Schéma cas test 1\_rainure

$\mu_{1-2}$  et est fixé à 0.6. Les deux paramètres du problème mécanique sont donc  $X$  et  $L$ , les bornes minimum et maximum de ces paramètres sont présentées au tableau 3.1.

Paramètres	min	max
$X$ (en mm)	35	208
$L$ (en mm)	1	30

TABLE 3.1: Bornes de conception pour le cas test 1\_rainure

Le cas test est constitué de 15 sous-domaines, comporte 15 836 éléments TRI6 (triangles à 6 nœuds) et 57 036 degrés de liberté. Un éclaté des sous-domaines et des interfaces est présenté en figure 3.20.

A l'exception de l'interface entre les deux structures (interface entre  $\Omega_{10}$  et  $\Omega_{11}$ ) qui comprend du contact et du frottement, toutes les interfaces sont parfaites. Le matériau est supposé avoir un comportement linéaire élastique isotrope avec un module d'Young de 210 GPa et un coefficient de Poisson de 0.3.

La fonction objectif est décrite à la figure 3.21. Il existe deux minimas, un global situé en  $X=89.9$  mm et  $L=22.9$  mm ayant une valeur de 1.49 et un local situé en  $X=207$  mm et  $L=13.1$  mm ayant une valeur 1.50. Sachant que le maximum obtenu est de 1.95, la différence entre ces deux minimas est donc d'environ 2%.

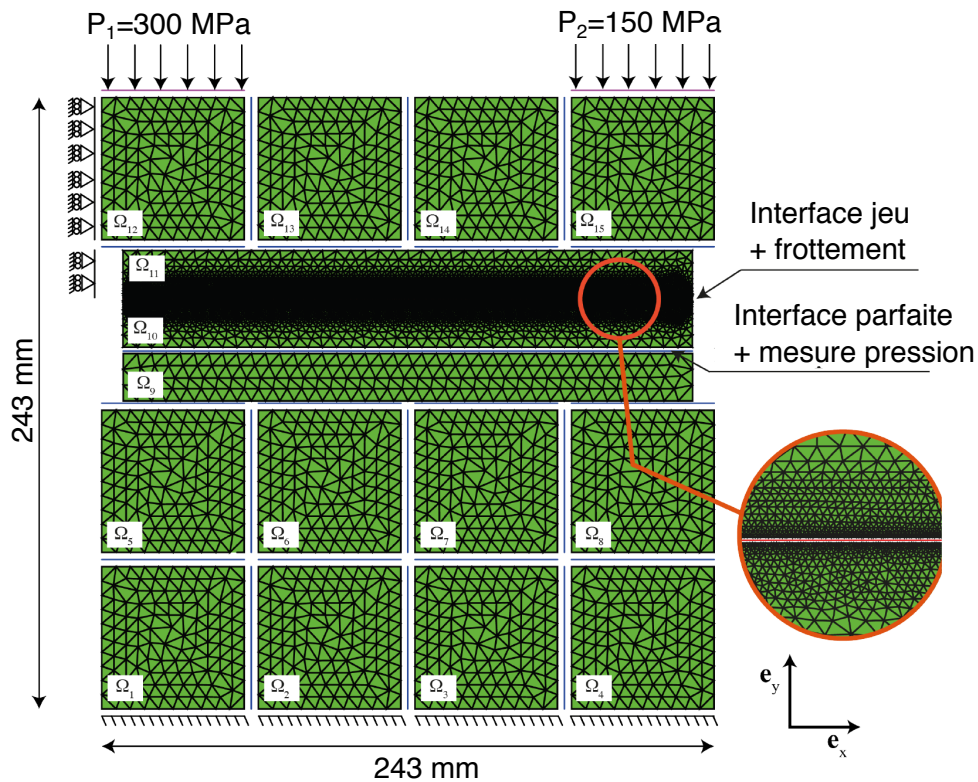
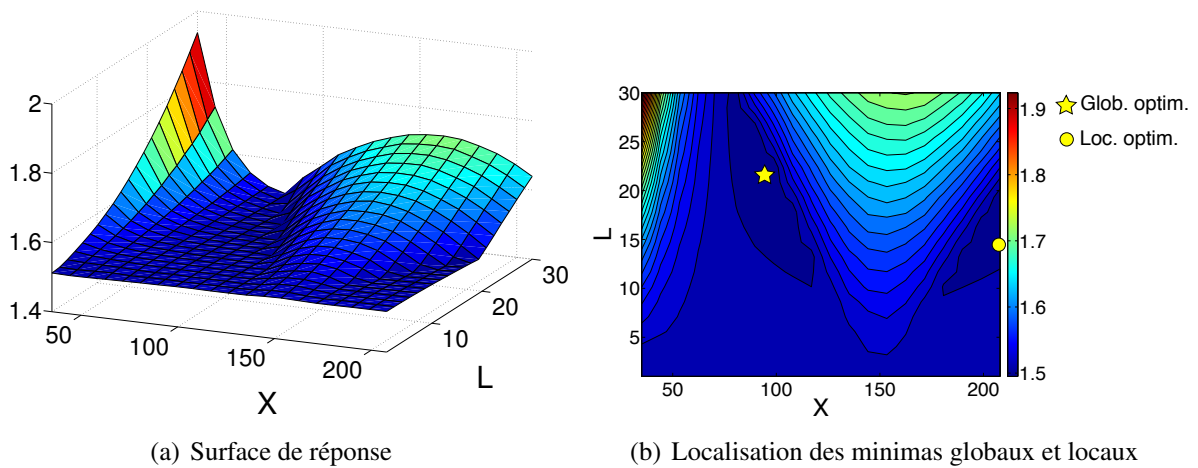


FIGURE 3.20: Eclaté et maillage du cas test 1\_rainure



(a) Surface de réponse

(b) Localisation des minima globaux et locaux

FIGURE 3.21: Surface de réponse en fonction des paramètres  $X$  et  $L$

L'étude précédemment effectuée sur le cas test *cubes* est appliquée sur ce nouvel exemple. De nouveau, on étudie l'influence du paramètre de convergence  $\eta$  des données partiellement convergées. Les résultats présentés ont été effectués pour 20 tirages LHS différents des données partiellement convergées. La structure de présentation des résultats est la même que

précédemment bien qu'allégée. On précise que les résultats sont présentés pour un nombre initial de points totalement convergés  $\mathbf{X}_{fcv}$  de  $3 \times n_{dim}$ , c'est-à-dire 6 sur ce cas test. Le niveau de convergence des données partiellement convergées varie de 0.05 à 0.01 (temps de calcul CPU variant de 4.2s à 9s), le niveau de convergence des données totalement convergées est fixé à  $10^{-4}$  (temps de calcul CPU 130s)

### 2.5.1 Qualité du métamodèle construit avec des données uniquement partiellement convergées

A titre informatif, on présente ici la qualité moyenne (sur 20 essais numériques pour chaque niveau de convergence  $\eta$ ) du métamodèle construit uniquement avec les données partiellement convergées.

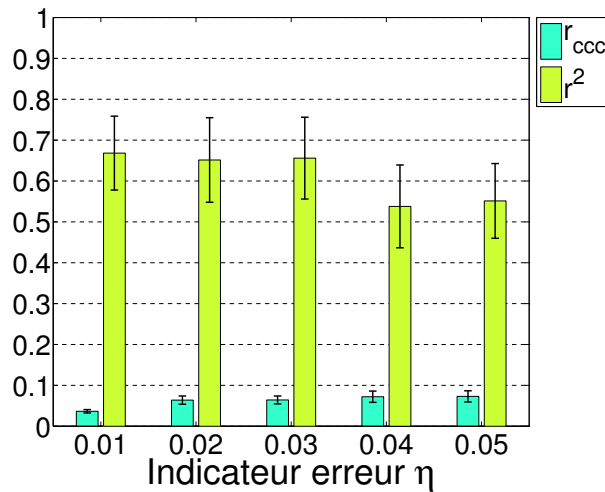


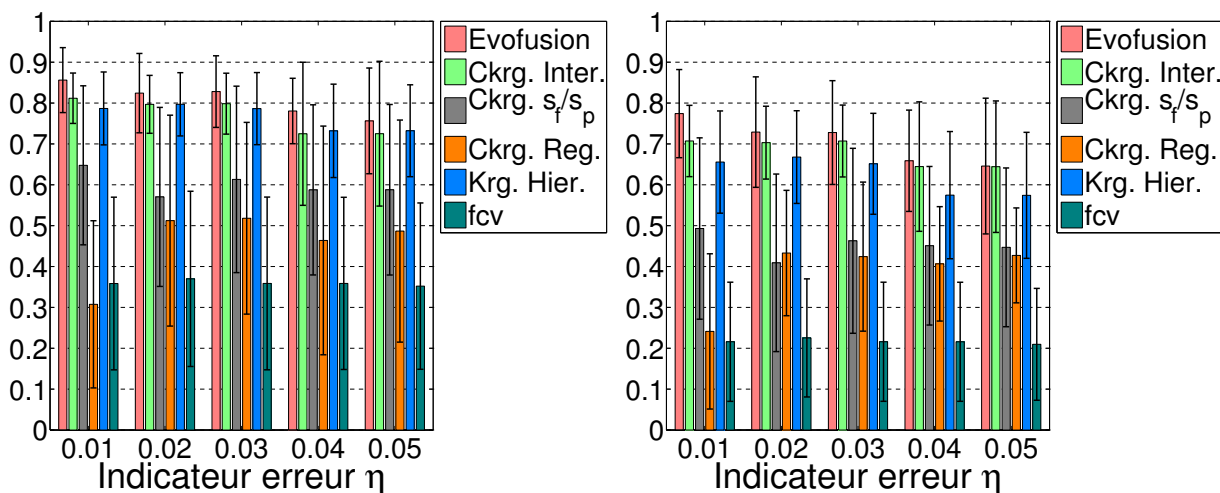
FIGURE 3.22: Qualité moyenne des métamodèles construits avec les données partiellement convergées pour le cas test 1\_rainure

On constate une faible corrélation  $r_{ccc}$  obtenue à l'aide des données partiellement convergées mais une corrélation  $r^2$  plus élevée. Ceci s'explique par le fait que globalement entre les données totalement et partiellement convergées il y a un offset. Cet offset n'est cependant pas constant, la relation entre modèle basse et haute fidélité est donc complexe.

### 2.5.2 Qualité du métamodèle multi-fidélité initial

On se propose à la figure 3.23 de présenter les différentes corrélation  $r_{ccc}$  et  $r^2$  entre les métamodèles multi-fidélité et le métamodèle de référence. On présente également la qualité de la réponse avec krigeage des données totalement convergées.

On constate clairement que l'utilisation de données partiellement convergées afin de construire un métamodèle multi-fidélité apporte une information essentielle quant à la construction de celui-ci pour 3 approches, à savoir l'Evofusion, le cokrigeage Intercorrélation et le



(a) Corrélation  $r_{ccc}$  entre le métamodèle multi-fidélité et le métamodèle de référence (b) Corrélation  $r^2$  entre le métamodèle multi-fidélité et le métamodèle de référence

**FIGURE 3.23:** Qualité du métamodèle multi-fidélité et du métamodèle de référence pour le cas test *1\_rainure*

krigeage hiérarchique. Les deux autres méthodes de cokrigeage sont clairement moins efficaces. Les trois méthodes multi-fidélité citées précédemment offrent également une qualité de réponse supérieure à celle obtenue avec krigeage des données totalement convergées. L’apport des données partiellement convergées est donc indéniable.

On remarque que les méthodes de cokrigeage  $s_{fcv}/s_{pcv}$  et de type auto-regréssif ne sont pas performantes. La détermination respective des paramètres de pondération  $\sigma_{fcv}/\sigma_{pcv}$  et  $\rho$  n’est donc pas efficace sur ce cas test. Cependant, le krigeage hiérarchique, utilisant lui aussi un paramètre de pondération entre données partiellement et totalement convergées, offre une plus grande robustesse de construction du métamodèle ([Han et Görtz, 2012]).

### 2.5.3 Qualité du métamodèle multi-fidélité en fonction du coût de calcul

Comme pour le précédent cas test, on présente dans ce paragraphe l’évolution de la corrélation  $r_{ccc}$  entre les métamodèles multi-fidélité et le métamodèle de référence en fonction du coût de calcul. Les résultats sont présentés pour un niveau de convergence des données partiellement convergées calculées de  $\eta = 0.05$ . La figure 3.24 montre clairement en fonction du nombre de points totalement convergés que les trois approches citées précédemment sont de nouveau les meilleures pour fournir un métamodèle de qualité en un temps de calcul limité.

On illustre également à titre indicatif sur les figures 3.25 la qualité des métamodèles multi-fidélité de cokrigeage Intercorrélation et krigeage hiérarchique comparée à celle obtenue avec le krigeage des données totalement convergées calculées au fil des itérations de la stratégie de recalage. On distingue clairement que pour un faible coût de calcul les méthodes multi-fidélité offrent une qualité de réponse bien supérieure.



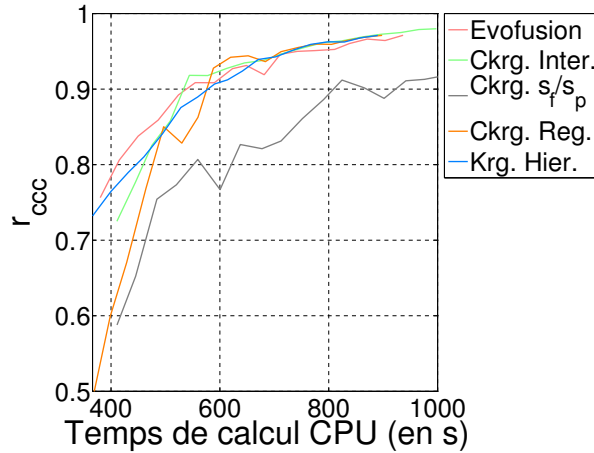
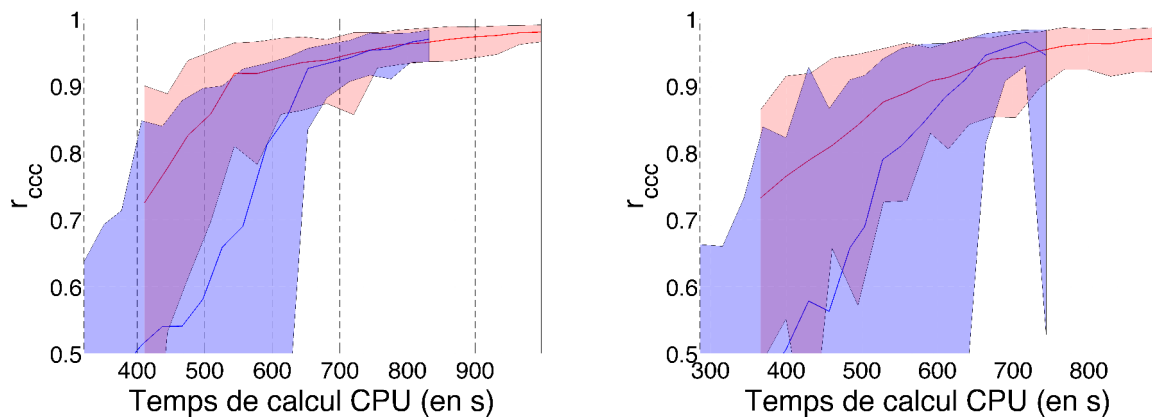


FIGURE 3.24: Comparaison  $r_{ccc}$  pour les différentes méthodes



(a) Comparaison de la qualité de métamodèles entre le cokrigage Intercorrélation et le krigeage des données totalement convergées

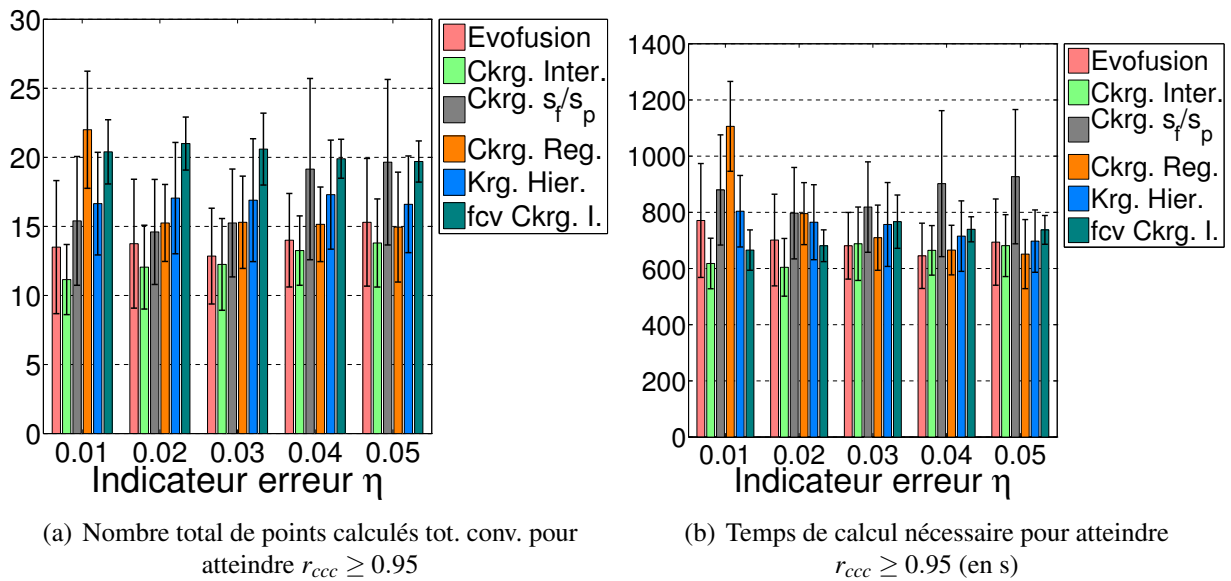
(b) Comparaison de la qualité de métamodèles entre le krigeage hiérarchique et le krigeage des données totalement convergées

FIGURE 3.25: Qualité des métamodèles multi-fidélité (en rouge) par rapport à celle obtenue avec les métamodèles construits uniquement avec les données totalement convergées (en bleu)

Cependant, on a pu constater sur le premier cas test que l'apport des stratégies multi-fidélité par rapport au krigeage pouvait être limité dans le cas où l'on souhaitait atteindre une qualité de métamodèle exigée. Qu'en est-t-il sur ce cas test ?

#### 2.5.4 Comparaison des méthodes multi-fidélité pour atteindre $r_{ccc} > 0.95$

Les figures 3.26 présentent le nombre total de points complètement convergés qu'il a fallu calculer pour atteindre le critère d'arrêt  $r_{ccc} > 0.95$ , ainsi que le temps de calcul CPU associé.



**FIGURE 3.26:** Comparaison des méthodes pour atteindre le critère de corrélation  $r_{ccc} \geq 0.95$  pour le cas test *1\_rainure*

On constate alors que les méthodes de cokrigage  $s_{fcv}/s_{pcv}$  et de type auto-régressif peuvent s'avérer efficaces sur cette étude. Cependant, cette efficacité dépend du niveau de convergence  $\eta$  des données partiellement convergées. Les méthodes d'Evofusion, de cokrigage Intercorrélation et de krigeage hiérarchique offrent des résultats intéressants et réguliers suivant les niveaux de convergence des données partiellement convergées. De nouveau, la méthode la plus performante pour atteindre rapidement le niveau de qualité exigée est la méthode de cokrigage Intercorrélation.

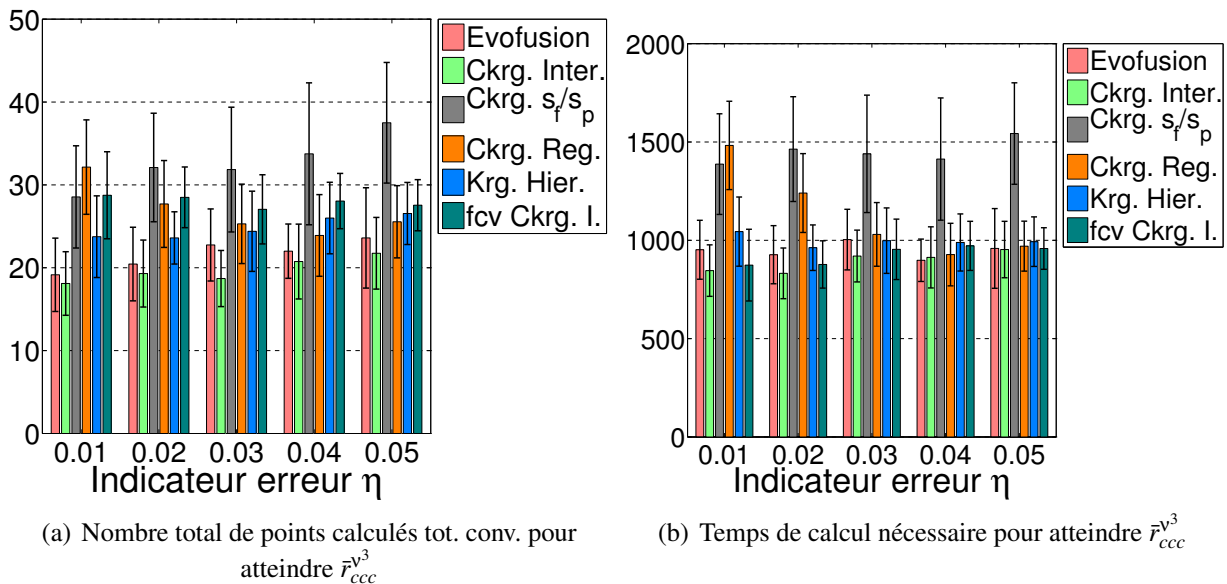
Contrairement au cas test précédent, on constate que la méthode de krigeage des données partiellement convergées exige techniquement plus de points totalement convergés pour atteindre le critère d'arrêt. L'apport des méthodes multi-fidélité de ce point de vue est donc intéressant. Cependant, en terme de temps de calcul, l'économie réalisée n'est pas aussi importante. En effet, le calcul du métamodèle de krigeage ne tient compte que des données totalement convergées (le temps de calcul des données partiellement convergées associé à cette méthode est nul). On peut également ajouter que la stratégie de calcul multiparamétrique tend à réduire les coûts de calcul des nouveaux jeux de paramètres (ayant plus de solutions précédemment calculées, on peut initialiser plus intelligemment l'algorithme LaTIn et ainsi réduire encore les coûts de calculs).

Par exemple, pour le niveau d'erreur  $\eta = 0.05$  le nombre de points pour atteindre le critère d'arrêt avec la méthode de cokrigage Intercorrélation est de 14, celui avec le krigeage (fcv. Ckrg. I) est de 20. En moyenne, un point calculé totalement convergé à l'aide de la stratégie multiparamétrique est d'environ 30s. Pour rappel, un point partiellement convergé a un coût de calcul de 4.2s. Ainsi le coût de calcul supplémentaire pour la méthode de krigeage n'est que de :  $6 \times 30 - 4.2 \times 28 = 68s$  (28 correspond au nombre de points partiellement convergés qu'il a fallu calculer en moyenne avec la méthode de cokrigage).

Finalement, on peut conclure que les meilleures approches multi-fidélité permettant d'atteindre un niveau de qualité exigée à  $r_{ccc} \geq 0.95$  sont : l'EvoFusion, le cokrigeage Intercorrélation et le krigeage hiérarchique. **D'un point de vue technique**, ces méthodes demandent légèrement moins de points totalement convergés pour atteindre ce critère d'arrêt que par krigeage. Cependant, **d'un point de vue réalisation et coût de calcul**, l'apport des stratégies multi-fidélité n'est pas aussi significatif dans le cadre de calcul de la LaTin multiparamétrique.

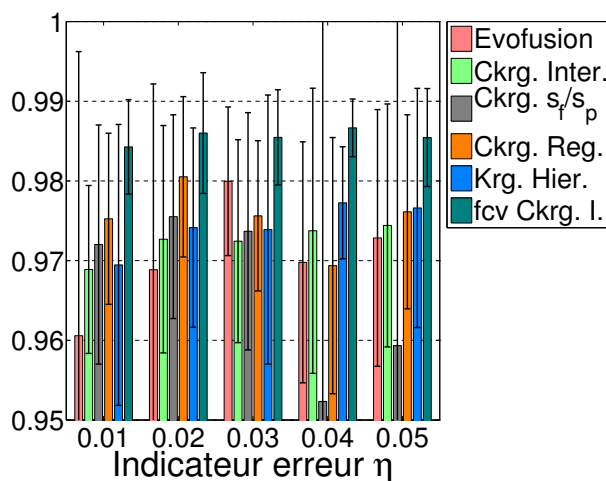
### 2.5.5 Comparaison des méthodes multi-fidélité pour atteindre $\bar{r}_{ccc}^{v^3}$

Afin de compléter l'étude sur ce cas test, on se propose de présenter aux figures 3.27 et 3.28, les résultats obtenus par les différentes stratégies pour atteindre le critère d'arrêt  $\bar{r}_{ccc}^{v^3}$ .



**FIGURE 3.27:** Comparaison des méthodes pour atteindre le critère d'arrêt  $\bar{r}_{ccc}^{v^3}$  pour le cas test  $1\_rainure$

On constate que comme pour le cas test précédent, l'utilisation de ce critère d'arrêt permet d'obtenir au final un métamodèle de bonne qualité. Celui-ci est donc tout à fait intéressant en vue d'obtenir un métamodèle pouvant servir d'abaque virtuel. Concernant les coûts de calcul et l'efficacité des méthodes multi-fidélité, le même genre de conclusions qu'au paragraphe précédent peut être tiré. Les méthodes d'EvoFusion, de cokrigeage Intercorrélation et de krigeage hiérarchique sont les plus performantes. Cependant, concernant le coût de calcul, la méthode de krigeage des données totalement convergées s'avère tout aussi coûteuse (bien que techniquement plus de points soient nécessaires) pour une qualité obtenue légèrement supérieure. L'apport des stratégies multi-fidélité par rapport au krigeage est donc une fois encore limité.



**FIGURE 3.28:** Corrélacion  $r_{ccc}$  obtenue entre les métamodèles multi-fidélité finaux et le métamodèle de référence

Cependant, il est important de garder en mémoire que les données totalement convergées utilisées pour construire les métamodèles de krigeage sont issues principalement d'une stratégie d'enrichissement intelligente. Mener la même stratégie de recalage avec uniquement des données totalement convergées et avec un tirage LHS initial de  $10 \times n_{dim}$  points (donc 20 en l'occurrence sur ce cas test) est également un bon élément de comparaison. La qualité moyenne (pour 20 tirages LHS initiaux différents) du métamodèle de krigeage construit avec ces  $10 \times n_{dim}$  points est alors seulement de  $r_{ccc} = 0.86$ . Le temps nécessaire moyen pour atteindre le critère d'arrêt  $\bar{r}_{ccc}^v$  est de 1100s pour un nombre moyen de points calculés de 32. Dans ce cadre d'étude, cette approche est donc la moins efficace en terme de temps de calcul. Cette perspective tend donc bien à montrer que le tirage des points joue donc un rôle prépondérant dans cette recherche de création d'abaque virtuel. Un choix intelligent des points de l'échantillonnage aide logiquement à réduire les temps de calcul, ce qui prouve de nouveau, que ce champ de recherche peut amener des résultats intéressants dans la création de métamodèles ([Myšáková *et al.*, 2014, Rosenbaum et Schulz, 2012]). **Cette perspective sera d'ailleurs utilisée au cours des prochains chapitres, en vue de construire des métamodèles plus performants.**

### 3 Bilan de l'étude comparative des méthodes de construction de métamodèles

On rappelle que cinq approches ont été testées dans cette étude : L'Evofusion, trois procédés de cokrigeage et le krigeage hiérarchique. Le but principal de cette étude était de répertorier dans le cadre de la mécanique des assemblages, les techniques les plus performantes suivant différents critères. Trois stratégies se sont montrées intéressantes par rapport aux deux autres dans le contexte de création de métamodèles à **coût de calcul fixé** ou à **qualité exigée**. Ces trois

stratégies sont :

- L'Evofusion
- Le cokrigeage Intercorrélation
- Le krigeage hiérarchique

L'apport de ces stratégies multi-fidélité est intéressant en vue d'obtenir un bon métamodèle (c'est-à-dire aussi proche que possible de la fonction objectif recherchée) pour un temps de calcul limité. Ces méthodes se sont montrées en moyenne plus performantes et plus robustes que l'approche consistant à construire un métamodèle de krigeage des données totalement convergées. **A coût de calcul fixé**, les stratégies multi-fidélité se sont donc révélées bien adaptées à nos problèmes de mécanique des assemblages. Cette conclusion est importante car elle illustre le potentiel de ces méthodes dans le cadre de ce travail.

Celles-ci se sont montrées également robustes. En effet, on a pu voir au cours des études sur les deux cas test que la seule utilisation des données partiellement ou totalement convergées initiales ne pouvait pas suffire à créer un métamodèle de « bonne qualité ». C'est réellement la combinaison des deux sources d'information qui a permis d'obtenir des métamodèles multi-fidélité performants. Dans la littérature, certaines de ces méthodes s'avèrent également intéressantes mais une source d'information (données partiellement convergées ou totalement convergées) peut déjà être d'une très bonne qualité. Ainsi dans [Forrester *et al.*, 2007], l'utilisation du seul krigeage des données totalement convergées amène à une qualité de métamodèle évalué par  $r^2$  à 0.949 contre 0.962 pour le métamodèle construit par cokrigeage (de type auto-régressif). Les données provenant du modèle basse fidélité peuvent également être bien corrélées avec le modèle haute fidélité comme dans [Han *et al.*, 2010].

On remarque que les deux autres méthodes de cokrigeage exigeant la détermination d'un paramètre de mise à l'échelle des données partiellement convergées sont moins performantes. Ces approches sont théoriquement plus flexibles mais s'avèrent dans la réalité être plus difficiles à mettre en œuvre. La raison est que le lien entre les modèles basse et haute fidélité est complexe et que ces méthodes se basent principalement sur l'évaluation de ce lien.

Du point de vue de la construction de métamodèle à **qualité exigée**, l'apport des approches multi-fidélité par rapport au krigeage est moins intéressant. En effet, il arrive que le seul krigeage des données totalement convergées fournit un métamodèle de bonne qualité ( $r_{ccc} > 0.95$ ) en un temps moyen équivalent voir moindre à celui nécessaire aux méthodes multi-fidélité. Les principales raisons sont :

- Les données totalement convergées utilisées pour le krigeage sont celles utilisées par les approches multi-fidélité (en l'occurrence celles du cokrigeage Intercorrélation). Ces données sont principalement issues d'un enrichissement visant à construire un métamodèle le plus précis possible. Elles ont donc été choisies intelligemment. Dans le cadre du krigeage, l'application de la stratégie de recalage à l'aide de tirages LHS initiaux de  $10 \times n_{dim}$  points amènent à des temps de calcul pouvant être plus conséquents (voir paragraphe 2.5.5). Le choix du tirage est donc prépondérant et explique en partie ce résultat.
- L'économie du temps de calcul des données partiellement convergées par rapport aux approches multi-fidélité.

- La limitation de l'impact de calcul d'une donnée totalement convergée au fur et à mesure des itérations de la stratégie de recalage par l'approche multiparamétrique (voir paragraphe 2.5.4). De ce fait, même si techniquement plus de points totalement convergés sont nécessaires pour atteindre le critère de qualité souhaitée, la stratégie de calcul LaTin multiparamétrique tend au fur et à mesure des jeux de paramètres à calculer à réduire les temps de calcul ([Néron *et al.*, 2015, Soulier *et al.*, 2014])

On a également pu constater au cours de cette étude que le niveau de convergence des données partiellement convergées fixées à  $\eta = 0.05$  s'avère intéressant. Il permet d'obtenir des métamodèles multi-fidélité de qualité tout en limitant le temps de calcul de ce type de données. Globalement sur l'ensemble de l'étude, il s'agit du niveau amenant aux meilleures performances. Ceci corrobore ce qui a pu être mis en avant lors du chapitre 2, sauf que l'on peut étendre ce résultat non seulement à l'Evofusion mais également au cokrigeage Intercorrélation et krigage hiérarchique. Ce niveau de convergence est donc celui qui nous servira lors du calcul de données partiellement convergées sur des cas test mécaniques 2D.

Le prochain chapitre visera à intégrer les trois approches multi-fidélité qui ont été jugées efficaces dans une démarche de type EGO afin de mener une stratégie de recherche de minimum d'une fonction objectif. On s'intéressera également : à tester différents critères d'arrêt d'enrichissement de la stratégie EGO, à la qualité de résultats obtenus et au temps de calcul nécessaire pour atteindre ces critères. Finalement la démarche d'optimisation multi-niveaux utilisées au sein du laboratoire (voir figure 1.6) sera mise en place sur un cas test mécanique.



# Chapitre 4

## Utilisation des approches multi-niveaux et multi-fidélité pour l'optimisation globale

*Dans ce quatrième chapitre, les méthodes multi-fidélité d'Evofusion, de cokrigeage Intercorrélation et de krigeage hiérarchique sont comparées afin de déterminer laquelle ou lesquelles sont les plus efficaces dans le cadre de la stratégie d'optimisation multi-niveaux. Différents critères d'arrêt de la stratégie d'enrichissement et différentes méthodes de tirage sont également mises en place pour améliorer l'efficacité de l'optimisation globale réalisée à l'aide des métamodèles multi-fidélité.*

### Sommaire

---

<b>1</b>	<b>Critère d'arrêt de la stratégie EGO</b>	<b>152</b>
1.1	Introduction	152
1.2	Différents critères d'arrêt	153
1.3	Comparaison sur des fonctions tests analytiques	155
<b>2</b>	<b>Amélioration du tirage initial pour la construction de métamodèle</b>	<b>161</b>
2.1	Méthode d'échantillonnage adaptatif avec un métamodèle de krigeage	162
2.2	Influence du tirage pour un métamodèle de cokrigeage	164
<b>3</b>	<b>Optimisation d'un cas test mécanique : premier niveau d'optimisation</b>	<b>168</b>
3.1	Présentation du cas test <b>2_rainures</b>	168
3.2	Traitement du cas test <b>2_rainures</b>	170
<b>4</b>	<b>Optimisation d'un cas test mécanique : second niveau d'optimisation</b>	<b>181</b>
4.1	Stratégie d'optimisation locale basée sur les méthodes à région de confiance	181
4.2	Mise en place du second niveau d'optimisation sur le cas test	184
4.3	Bilan de la stratégie d'optimisation multi-niveaux sur le cas test <b>2_rainures</b>	186
<b>5</b>	<b>Bilan</b>	<b>187</b>

---



Dans ce chapitre, on s'intéresse plus spécifiquement aux performances atteintes par les méthodes multi-fidélité dans le cadre d'une optimisation d'une fonction objectif. Les méthodes de construction multi-fidélité jugées performantes à la suite des résultats du chapitre précédent, à savoir, la méthode d'Evofusion, de cokrigeage Intercorrélation **qui dans la suite sera simplement référencée par cokrigeage**, et de krigeage hiérarchique sont utilisées au cours du chapitre.

La stratégie d'optimisation employée est la stratégie multi-niveaux décrite à la figure 1.6 du chapitre 1. Une première optimisation globale est effectuée sur un métamodèle. Le jeu de paramètres correspondant au minimum trouvé est ensuite utilisé comme point de départ à une stratégie d'optimisation locale sur le modèle mécanique réel. Dans la littérature (par exemple [Jones *et al.*, 1998, Forrester, 2004]), le second niveau d'optimisation n'est pas nécessairement utilisé car un nombre d'appels au solveur qui ne justifie pas le gain obtenu en terme de précision sur l'optimum peut être utilisé. Dans le cadre de calcul de la LaTIn multiparamétrique, le temps de calcul peut être drastiquement réduit, ce qui rend viable cette stratégie d'optimisation multi-niveaux [Laurent, 2013]. Il est à noter que pour effectuer l'optimisation sur le premier niveau, une stratégie de type EGO (stratégie de recherche du minimum) est mise en place. L'échantillonnage de l'espace permettant la construction du métamodèle est donc enrichi au point maximisant l'EI.

Dans la première partie de ce chapitre, on s'intéresse à comparer des critères d'arrêt de la stratégie EGO. En effet, tout au long du chapitre, on s'attache à utiliser des critères d'arrêt s'appuyant sur des considérations statistiques (et non des critères d'arrêt en fonction du temps de calcul ou du nombre d'appels au solveur effectué). Leur comparaison se fait notamment sur la qualité du résultat obtenu et du temps de calcul nécessaire pour les atteindre. Le but est de trouver, si possible, un critère d'arrêt amenant à localiser correctement la zone du minimum global en un temps réduit.

Par la suite, une étude sur l'influence du tirage initial des données partiellement convergées est menée. Cette étude vise à améliorer, pour un coût de calcul similaire, la qualité du métamodèle multi-fidélité utilisé pour mener la stratégie EGO.

Finalement, la stratégie d'optimisation multi-niveaux est mise en place sur un cas test mécanique. On verra, sur celui-ci, que le couplage entre la méthode de calcul LaTIn multiparamétrique et la construction de métamodèle multi-fidélité (en l'occurrence le cokrigeage) permet un gain de calcul d'un facteur 4.

## 1 Critère d'arrêt de la stratégie EGO

### 1.1 Introduction

Dans les travaux de [Laurent, 2013], on rappelle que la stratégie d'optimisation multi-niveaux mise en place sur le cas test *1\_rainure* a permis de trouver le minimum global de la fonction d'intérêt (voir tableau 4.1). Pour rappel, les métamodèles furent construits par un procédé de krigeage (pas d'utilisation de données partiellement convergées) et l'optimisation locale a été faite par un algorithme SQP. Un nombre important de points servant à l'enrichissement a été utilisé :  $\approx 50\%$  de l'ensemble des points calculés pour mener la stratégie complète. Cet enrichissement a amené à une localisation précise du minimum. Le deuxième niveau d'optimisation

s'est alors avéré n'être qu'une étape de vérification du minimum précédemment trouvé. On peut donc à juste titre supposer que l'enrichissement effectué lors de la stratégie EGO a été « inutilement » trop coûteux. Le critère d'arrêt utilisé lors de cette étude est celui décrit au paragraphe 1.2.1. Il peut être intéressant de proposer d'autres critères d'arrêt afin d'essayer d'économiser du temps de calcul.

Niv. optimisation	Etapas		
Niveau 1	Métamodèle	Nb. pts. initiaux	20
		Temps CPU (s)	1920
		$\mathbf{x}_{\min}^{meta}$	(94.8mm, 22.7mm)
	EGO	Nb. pts. ajoutés.	34
		Temps CPU (s)	2500
		$\mathbf{x}_{\min}^{EGO}$	(89.8mm, 22.9mm)
Niveau 2	SQP	Nb. pts. calculés.	12
		Temps CPU (s)	450
		$\mathbf{x}_{\min}$	(89.9mm, 22.9mm)
Total		Nb. pts. calculés.	66
		Temps CPU (s)	4542

TABLE 4.1: Récapitulatif des résultats obtenus par [Laurent, 2013]

## 1.2 Différents critères d'arrêt

On introduit ici les trois critères d'arrêt de la stratégie EGO qui sont comparés pour cette étude. L'efficacité de ces critères d'arrêt est comparée sur des fonctions tests de différentes natures puis sur un cas test mécanique.

### 1.2.1 Critère d'arrêt de référence

Le premier critère d'arrêt est celui introduit par [Huang *et al.*, 2006b]. Il s'appuie sur la quantité  $\Delta_h$  définie par :

$$\Delta_h = \max_{\mathbf{x} \in \mathcal{D}} \frac{EI(\mathbf{x})}{y_{\max} - y_{\min}} \quad (4.1)$$

avec  $y_{\max} = \max \{y(\mathbf{x}^{(i)})\}$ ,  $y_{\min} = \min \{y(\mathbf{x}^{(i)})\}$ . Comme préconisé dans [Huang *et al.*, 2006b], le critère d'arrêt final d'enrichissement est atteint lorsque la quan-

tité  $\Delta_h$  est inférieure  $(n_{dim} + 1)$  fois de suite à une valeur  $\delta_h$ . La valeur du paramètre  $\delta_h$  est égale à 0.001. Le critère final est noté  $\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$ . **Ce critère d'arrêt est celui nous servant de référence.**

### 1.2.2 Un deuxième critère d'arrêt

Le deuxième critère étudié est celui proposé dans [Jones *et al.*, 1998]. La quantité étudiée est notée :

$$\Delta_j = \max_{\mathbf{x} \in \mathcal{D}} \frac{\text{EI}(\mathbf{x})}{|y_{\min}|} \quad (4.2)$$

[Jones *et al.*, 1998] propose de stopper l'enrichissement lorsque  $\Delta_j < \delta_j = 0.01$ . Cependant, suite à des observations effectuées lors de différents essais, ce critère peut poser des problèmes de convergence. En effet, si le minimum de la fonction est :  $y_{\min} = 0$  alors ce critère tend vers l'infini. A contrario, si la valeur de  $y_{\min}$  est très grande alors ce critère sera atteint très rapidement. Afin de remédier à ces problèmes on se propose de recentrer les données par l'opération suivante :

$$\tilde{y}(\mathbf{x}^{(i)}) = y(\mathbf{x}^{(i)}) + \frac{y_{\max} + y_{\min}}{2} \quad (4.3)$$

La quantité  $\Delta_j$  est modifiée en :

$$\begin{aligned} \Delta_j &= \max_{\mathbf{x} \in \mathcal{D}} \frac{\text{EI}(\mathbf{x})}{|\tilde{y}_{\min}|} < \delta_j \\ \Delta_j &= \max_{\mathbf{x} \in \mathcal{D}} \frac{\text{EI}(\mathbf{x})}{y_{\max} - y_{\min}} < \frac{\delta_j}{2} \end{aligned} \quad (4.4)$$

Ainsi le second critère devient semblable au premier, excepté que  $\delta_h = 0.005$ . Le critère d'arrêt final est atteint lorsque  $\Delta_j$  est inférieure  $(n_{dim} + 1)$  fois de suite à  $\frac{\delta_j}{2}$ . Celui-ci est noté  $\tilde{\Delta}_{j\ 0.005}^{n_{dim}+1}$ .

### 1.2.3 Un dernier critère d'arrêt

Les deux précédentes méthodes évaluent, pour chaque phase d'enrichissement, un certain critère d'arrêt. L'enrichissement est interrompu lorsque le critère choisi est  $(n_{dim} + 1)$  fois inférieur à une certaine valeur. Plutôt que de considérer de façon isolée chaque phase d'enrichissement, on propose de prendre en compte plusieurs phases successives par l'intermédiaire d'une moyenne. Ainsi, on traite un effet d'histoire et on teste la stabilisation de l'enrichissement. Le critère proposé s'appuie sur la quantité définie par :

$$\Delta_m = \frac{1}{n_{dim} + 1} \sum_{i=0}^{n_{dim}} \Delta_h^{v-i} \quad (4.5)$$

Où  $v$  représente le dernier calcul effectué et  $\Delta_h^{v-i}$  correspond aux valeurs de la quantité  $\Delta_h$  des  $v - i$  calculs précédents (le calcul de cette quantité n'est donc possible qu'après avoir effectué  $n_{dim}$  premiers enrichissements). Cette quantité a pour effet de régulariser les différentes valeurs trouvées pour  $\Delta_h$ . Là encore, le critère d'arrêt est atteint lorsque  $\Delta_m$  est inférieure à une valeur  $\delta_m$ ,  $(n_{dim} + 1)$  fois d'affilé. Après une série de test  $\delta_m = 0.01$  semble un choix pertinent. On note alors  $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$  le critère final.

### 1.3 Comparaison sur des fonctions tests analytiques

Dans les prochains paragraphes, l'optimisation multi-niveaux est mise en place. Pour rappel, le premier niveau d'optimisation se base sur la construction d'un métamodèle sur lequel une stratégie de type EGO est mise en place. On compare alors, l'efficacité et le coût de calcul des trois critères d'arrêt  $\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$ ,  $\tilde{\Delta}_{j\ 0.005}^{n_{dim}+1}$  et  $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$ . Pour cette étude comparative, on s'intéresse lors du premier niveau d'optimisation (mené sur le métamodèle) **aux coûts de calcul nécessaires pour atteindre ces différents critères**, ainsi qu'à **la précision du minimum trouvé**. Cette précision est définie par la distance euclidienne normée suivante :

$$\text{Distance au minimum (en \%)} = \frac{\| \mathbf{x}_{\min}^{EGO} - \mathbf{x}_{\min} \|}{\max_{\mathbf{x} \in \mathcal{D}} \{ \| \mathbf{x} - \mathbf{x}_{\min} \| \}} \times 100 \quad (4.6)$$

**On précise que les espaces sont normés** afin d'avoir une même influence pour chacune des dimensions de l'espace.  $\mathbf{x}_{\min}^{EGO}$  est le minimum estimé par le métamodèle suite à la procédure EGO et  $\mathbf{x}_{\min}$  est le minimum global de la fonction objectif.

L'algorithme utilisé lors du second niveau d'optimisation est l'algorithme SQP, précédemment utilisé dans [Laurent, 2013] et disponible sous MATLAB via la fonction *fmincon*. Là encore, on s'intéresse aux coûts de calcul nécessaires pour amener à convergence cet algorithme selon le point de départ obtenu après avoir atteint les critères  $\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$ ,  $\tilde{\Delta}_{j\ 0.005}^{n_{dim}+1}$  et  $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$ . On fournit également l'information essentielle de savoir si le minimum global a effectivement été bien trouvé lors de cette seconde phase d'optimisation. On peut alors déterminer en fonction des critères d'arrêt EGO testés, non seulement la précision des résultats obtenus (via la quantité définie à l'équation 4.6) mais également leur taux de réussite à localiser correctement la zone du minimum global.

Différentes fonctions tests sont utilisées pour faire ces essais numériques (Rosenbrock, Hartmann et Goldstein). Pour chaque fonction testée, 100 tirages LHS initiaux différents sont utilisés. Le nombre de points initial est fixé à  $10 \times n_{dim}$ . Afin d'éviter un enrichissement trop important si les critères d'arrêt ne sont pas atteints, celui-ci est limité à  $20 \times n_{dim}$ . Ainsi le dernier métamodèle utilisé pour évaluer la fonction objectif est construit avec au maximum  $30 \times n_{dim}$  points. Seules des données haute fidélité sont utilisées dans le cadre de cette étude sur des exemples analytiques, les métamodèles construits s'appuient donc sur le procédé de krigeage.

#### 1.3.1 Comparaison sur des fonctions Rosenbrock

Dans ce paragraphe, on présente les différents résultats obtenus sur des cas test issus des fonctions de Rosenbrock à 2, 3, 4 et 5 dimensions. Ces fonctions sont définies par :

$$\forall \mathbf{x} \in [-2, 2]^{n_{dim}}, y(\mathbf{x}) = \sum_{i=1}^{n_{dim}-1} \left[ 100 \cdot (x_i^2 - x_{i+1})^2 + (x_i - 1)^2 \right] \quad (4.7)$$

avec  $n_{dim} \in \llbracket 2, 5 \rrbracket$ . En 2 et 3 dimensions la fonction de Rosenbrock n'admet qu'un minimum global. Pour les dimensions supérieures, deux minima sont définis (un global et un local). Le minimum global pour les différentes fonctions se situe en (-1,1), (-1,1,1), (-1,1,1,1)

et (-1,1,1,1,1). Des informations complémentaires sur ces fonctions peuvent être trouvées dans [Shang et Qiu, 2006].

On présente dans le tableau 4.2 les différents résultats obtenus. Pour le premier niveau d'optimisation, à la suite de la stratégie EGO et pour chaque critère d'arrêt testé, le nombre de points enrichis est présenté, ainsi que la précision du minimum obtenu. Pour le second niveau, on présente le nombre d'appels à la fonction (donc de points à calculer) pour arriver à convergence de l'algorithme d'optimisation locale. Sur les 100 essais numériques menés, on fournit également le taux de réussite à localiser effectivement le minimum global.

			$\tilde{\Delta}_h^{n_{dim}+1}_{0.001}$	$\tilde{\Delta}_j^{n_{dim}+1}_{0.005}$	$\tilde{\Delta}_m^{n_{dim}+1}_{0.01}$
1 <sup>er</sup> niveau d'optimisation	Rosen. 2D	Distance au minimum (en %)	6.4	13.1	14.1
		Nbre. pts ajoutés	37.7	15	10.4
	Rosen. 3D	Distance au minimum (en %)	10	16.6	18.8
		Nbre. pts ajoutés	48.4	18.9	13.6
	Rosen. 4D	Distance au minimum (en %)	17	22	23.9
		Nbre. pts ajoutés	77.3	33.3	19.2
	Rosen. 5D	Distance au minimum (en %)	24.9	25	25.4
		Nbre. pts ajoutés	100	46	24.6
2 <sup>nd</sup> niveau d'optimisation	Rosen. 2D	SQP : Nbre appels à la fonction	34.3	45.4	47.8
		Localisation de la zone du minimum (en %)	100	100	100
	Rosen. 3D	SQP : Nbre appels à la fonction	67.8	83.1	91.3
		Localisation de la zone du minimum (en %)	100	100	100
	Rosen. 4D	SQP : Nbre appels à la fonction	120.4	137.2	142.7
		Localisation de la zone du minimum (en %)	98	89	85
	Rosen. 5D	SQP : Nbre appels à la fonction	200	200.4	210
		Localisation de la zone du minimum (en %)	84	83	82

**TABLE 4.2:** Tableau des résultats de la première étape et de la seconde étape d'optimisation pour les fonctions Rosenbrock

Plusieurs informations peuvent être tirées des différents résultats précédents.

- On constate que la qualité ainsi que le nombre d'appels au solveur pour résoudre ces problèmes d'optimisation sont variables selon les critères d'arrêt testés. Le critère d'arrêt exigeant un enrichissement plus important pour être atteint est le critère  $\tilde{\Delta}_h^{n_{dim}+1}_{0.001}$ . Il s'agit donc logiquement du critère amenant une meilleure précision de la localisation du minimum lors du premier niveau d'optimisation. A titre informatif, il s'agit également du critère amenant à une dispersion moindre des résultats obtenus.

- Une information essentielle que l'on peut tirer de cette étude est qu'un enrichissement important lors du premier niveau n'amène pas nécessairement à un résultat final de bien meilleure qualité. Ainsi, sur la fonction de Rosenbrock à 5 dimensions l'utilisation du critère d'arrêt  $\tilde{\Delta}_h^{n_{dim}+1}$  lors de la première phase d'optimisation aboutit finalement à localiser dans 84% des cas le minimum global contre 83% et 82% pour l'utilisation des critères  $\tilde{\Delta}_j^{n_{dim}+1}$  et  $\tilde{\Delta}_m^{n_{dim}+1}$ . Cependant, ces derniers offrent une économie pouvant être assez conséquente lors de la première étape d'optimisation (respectivement 50 et 75% pour la fonction de Rosenbrock à 5 dimensions)

**Remarque :** On constate d'ailleurs que le critère d'arrêt  $\tilde{\Delta}_h^{n_{dim}+1}$  n'est pas réellement atteint puisque la limite d'enrichissement à  $20 \times n_{dim} = 100$  points a été atteinte pour chacun des 100 essais numériques menés. Ce critère d'arrêt peut donc s'avérer coûteux. Cela est dû au faible gradient autour de la zone du minimum ce qui aboutit à une diminution lente du maximum d'*expected improvement* et en conséquence de la quantité  $\Delta_h$ .

- Pour la fonction de Rosenbrock à 4 dimensions, les résultats décrits précédemment sont différents. En effet, les trois critères demandent un enrichissement qui est relativement dans les mêmes proportions que pour la fonction à 5 dimensions mais pour des taux de réussite moins comparables à savoir 98%, 89% et 85%. Le coût de calcul associé aux critères  $\tilde{\Delta}_j^{n_{dim}+1}$  et  $\tilde{\Delta}_m^{n_{dim}+1}$  est donc moindre mais amène également un taux de réussite plus faible pour trouver le minimum global.
- Les fonctions de Rosenbrock à 2 et 3 dimensions sont bien sûr différentes car elles ne présentent qu'un seul minimum. Une conclusion que l'on peut tirer pour ces deux fonctions est que globalement sur l'ensemble de la stratégie multi-niveaux l'utilisation des critères  $\tilde{\Delta}_m^{n_{dim}+1}$  et  $\tilde{\Delta}_j^{n_{dim}+1}$  permet une économie de points à évaluer.

On se propose dans le tableau 4.3 de faire un récapitulatif du nombre total d'appels effectué au solveur lors de la première étape d'enrichissement EGO et lors de la seconde étape d'optimisation. On présente alors les gains d'appels au solveur réalisés par l'utilisation des critères d'arrêt  $\tilde{\Delta}_j^{n_{dim}+1}$  et  $\tilde{\Delta}_m^{n_{dim}+1}$  par rapport au critère  $\tilde{\Delta}_h^{n_{dim}+1}$ . Afin de comparer également leur rapport qualité-coût, deux indicateurs sont mis en place (voir équations 4.8 et 4.9).

$$\tau_{EGO} = \frac{\text{Réussite à localiser la zone du minimum global (en \%)}}{\text{Nombre de points pour l'enrichissement EGO}} \quad (4.8)$$

$$\tau_{tot} = \frac{\text{Réussite à localiser la zone du minimum global (en \%)}}{\text{Nombre d'appels au solveur lors des deux phases d'optimisation (EGO+SQP)}} \quad (4.9)$$

Le premier s'intéresse au rapport qualité-coût spécifique à la localisation de la zone du minimum global. Le second s'intéresse au rapport qualité-coût de l'ensemble de la méthode.

	$\tilde{\Delta}_h^{n_{dim}+1} 0.001$			$\tilde{\Delta}_j^{n_{dim}+1} 0.005$				$\tilde{\Delta}_m^{n_{dim}+1} 0.01$			
	Nbre Itér.	$\tau_{EGO}$	$\tau_{tot}$	Nbre Itér.	Gain	$\tau_{EGO}$	$\tau_{tot}$	Nbre Itér.	Gain	$\tau_{EGO}$	$\tau_{tot}$
2D	72	2.7	1.4	60.4	16.1%	6.7	<b>1.7</b>	58.2	<b>19.2%</b>	<b>9.6</b>	<b>1.7</b>
3D	116.2	2.1	0.9	102	<b>12.2%</b>	5.3	<b>1</b>	104.9	9.7%	<b>7.4</b>	<b>1</b>
4D	197.7	1.3	0.5	170.5	13.8%	2.7	<b>0.5</b>	161.9	<b>18.1%</b>	<b>4.4</b>	<b>0.5</b>
5D	300	0.8	0.3	246.4	17.9%	1.8	0.3	234.6	<b>21.8%</b>	<b>3.4</b>	<b>0.4</b>

**TABLE 4.3:** Comparaison récapitulative des résultats obtenus pour les fonctions de Rosenbrock

**Bilan :** On observe que les rapports  $\tau_{EGO}$  et  $\tau_{tot}$  sont, quelles que soient les fonctions de Rosenbrock traitées, favorables au critère  $\tilde{\Delta}_m^{n_{dim}+1} 0.01$ . Ainsi, du point de vue qualité-coût, pour limiter le temps de calcul, on pourra choisir ce critère d'arrêt qui peut s'avérer tout à fait performant. En effet, sur la fonction de Rosenbrock à 5 dimensions pour quasiment le même taux de réussite, le coût total est réduit d'un peu plus de 20%. Il faut cependant être conscient que la probabilité de localisation de la zone du minimum est réduite comme l'illustre la fonction de Rosenbrock à 4 paramètres.

### 1.3.2 Cas test sur les fonctions Hartmann & Goldstein

Les résultats obtenus précédemment sont intéressants concernant le coût de calcul et la réussite à localiser le minimum, cependant le nombre de minima et la dimension des problèmes étaient limités. On se propose d'étudier d'autres fonctions tests : une fonction à 2 paramètres, une autre à 3 et finalement une à 6 avec plusieurs minima.

#### Fonction de Goldstein-Price

La première fonction étudiée est la fonction de Goldstein-Price. Il s'agit d'une fonction à deux dimensions.

$$\forall \mathbf{x} \in [-2, 2]^2 \quad y(\mathbf{x}) = \left( 1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2) \right) \quad (4.10)$$

Cette fonction possède 4 minima. Le global se situe au point  $\mathbf{x}_{\min} = (0, -1)$ . On préférera étudier le logarithme de cette fonction comme préconisé dans [Jones *et al.*, 1998] :  $\ln(y)$ .

#### Fonction d'Hartmann3

La deuxième fonction étudiée est Hartmann3.

$$\forall \mathbf{x} \in [0, 1]^3 \quad y(\mathbf{x}) = - \sum_{i=1}^4 \alpha_i \exp \left( - \sum_{j=1}^3 A_{ij} (x_j - P_{ij})^2 \right) \quad (4.11)$$

avec :

$$\alpha = (11.233.2)^T$$

$$\mathbf{A} = \begin{bmatrix} 3 & 10 & 10 \\ 0.1 & 10 & 35 \\ 3 & 10 & 30 \\ 0.1 & 10 & 35 \end{bmatrix}$$

$$\mathbf{P} = 10^{-4} \begin{bmatrix} 3689 & 1170 & 2673 \\ 4699 & 4387 & 7470 \\ 1091 & 8732 & 5547 \\ 381 & 5743 & 8828 \end{bmatrix}$$

3 minima sont présents sur cette fonction. Le minimum global se situe en  $\mathbf{x}_{\min} = (0.114614, 0.555649, 0.852547)$ .

### Fonction d'Hartmann6

La dernière fonction étudiée est Hartmann6.

$$\forall \mathbf{x} \in [0, 1]^6 \quad y(\mathbf{x}) = - \sum_{i=1}^4 \alpha_i \exp \left( - \sum_{j=1}^6 A_{ij} (x_j - P_{ij})^2 \right) \quad (4.12)$$

avec :

$$\alpha = (11.233.2)^T$$

$$\mathbf{A} = \begin{bmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{bmatrix}$$

$$\mathbf{P} = 10^{-4} \begin{bmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{bmatrix}$$

Cette fonction comporte 2 minima. Le minimum global se situe en  $\mathbf{x}_{\min} = (0.20169, 0.150011, 0.476874, 0.275332, 0.311652, 0.6573)$ . On préférera étudier cette fonction via une transformation logarithmique comme proposé dans [Jones *et al.*, 1998], c'est-à-dire :  $-\ln(-y)$

### Résultats de l'étude des fonctions de Goldstein et Hartmann

Les résultats de la stratégie d'optimisation menée sur ces trois fonctions sont alors présentés au tableau 4.4.

Comme pour les fonctions tests de Rosenbrock on constate une diminution du nombre de points enrichis pour les critères  $\tilde{\Delta}_j^{n_{dim}+1}_{0.005}$  et  $\tilde{\Delta}_m^{n_{dim}+1}_{0.01}$ . Cette diminution est très intéressante pour la



			$\tilde{\Delta}_h^{n_{dim}+1}$ 0.001	$\tilde{\Delta}_j^{n_{dim}+1}$ 0.005	$\tilde{\Delta}_m^{n_{dim}+1}$ 0.01
1 <sup>er</sup> niveau	Goldstein	Distance au minimum (en %)	3	5.2	6.8
		Nbre. pts ajoutés	14.4	10.5	9
	Hartmann3	Distance au minimum (en %)	2.3	4.5	3.8
		Nbre. pts ajoutés	11	6.7	7.2
	Hartmann6	Distance au minimum (en %)	5.5	12.6	17.4
		Nbre. pts ajoutés	70	31	21.3
2 <sup>nd</sup> niveau	Goldstein	SQP : Nbre appels à la fonction	27.6	29.1	29.8
		Localisation de la zone du minimum (en %)	92	89	86
	Hartmann3	SQP : Nbre appels à la fonction	32.5	33.5	33.1
		Localisation de la zone du minimum (en %)	100	100	100
	Hartmann6	SQP : Nbre appels à la fonction	91	95	95.8
		Localisation de la zone du minimum (en %)	94	85	78

**TABLE 4.4:** Tableau des résultats de la première étape et de la seconde étape d'optimisation pour les fonctions de Goldstein et Hartmann

fonction Hartmann6 où l'on divise par plus de 3 le nombre de points enrichis lors de la première étape d'optimisation pour le critère  $\tilde{\Delta}_m^{n_{dim}+1}$ . Cependant, on constate une baisse de la réussite à localiser le minimum global (78% contre 94%).

On remarque pour les deux autres fonctions testées, un nombre limité de points ajoutés quel que soit le critère d'arrêt à atteindre. Les taux de réussite à trouver le minimum global sont du même ordre de grandeur pour tous les critères d'arrêt testés.

Comme pour les cas test sur Rosenbrock, on présente au tableau 4.5 les gains réalisés par les différents critères  $\tilde{\Delta}_j^{n_{dim}+1}$  et  $\tilde{\Delta}_m^{n_{dim}+1}$ . Les indicateurs  $\tau_{EGO}$  et  $\tau_{tot}$  sont également calculés afin de mettre en avant le rapport qualité-coût obtenu.

	$\tilde{\Delta}_h^{n_{dim}+1}$ 0.001			$\tilde{\Delta}_j^{n_{dim}+1}$ 0.005				$\tilde{\Delta}_m^{n_{dim}+1}$ 0.01			
	Nbre Itér.	$\tau_{EGO}$	$\tau_{tot}$	Nbre Itér.	Gain	$\tau_{EGO}$	$\tau_{tot}$	Nbre Itér.	Gain	$\tau_{EGO}$	$\tau_{tot}$
Gol.	42	6.4	2.2	39.6	5.7%	8.5	<b>2.2</b>	38.8	<b>7.6%</b>	<b>9.6</b>	<b>2.2</b>
H3	43.5	9.1	2.3	40.2	<b>7.6%</b>	<b>14.9</b>	<b>2.5</b>	40.3	7.4%	13.9	<b>2.5</b>
H6	161	1.3	0.6	126	21.7%	2.7	<b>0.7</b>	117.1	<b>27.2%</b>	<b>3.7</b>	<b>0.7</b>

**TABLE 4.5:** Comparaison récapitulative des résultats obtenus pour les fonctions de Goldstein et Hartmann

**Bilan :** On constate donc des indicateurs  $\tau_{EGO}$  et  $\tau_{tot}$  favorables aux critères  $\tilde{\Delta}_{j, 0.005}^{n_{dim}+1}$  et  $\tilde{\Delta}_{m, 0.01}^{n_{dim}+1}$ . Là encore, ces deux critères favorisent la limitation du temps de calcul et offrent un rapport qualité-coût plus satisfaisant que le critère d'arrêt  $\tilde{\Delta}_{h, 0.001}^{n_{dim}+1}$ . Comme précédemment, ce dernier permet d'améliorer le taux de réussite de localisation de la zone du minimum global.

L'utilisation du critère  $\tilde{\Delta}_{m, 0.01}^{n_{dim}+1}$  est donc intéressante du point de vue temps de calcul. Cependant, son taux de réussite à localiser correctement le minimum peut être jugé trop faible. Afin de remédier à ce problème d'efficacité, on se propose à la section suivante de modifier la méthode initiale de tirage des données (LHS) afin de fournir un métamodèle de meilleure qualité sur lequel se fait la stratégie EGO. On s'intéresse également à l'impact que cette modification de tirage initial des données peut avoir sur les métamodèles multi-fidélité.

## 2 Amélioration du tirage initial pour la construction de métamodèle

Dans cette partie on s'intéresse à améliorer la qualité des métamodèles à l'aide d'un échantillonnage des données plus adapté. Dans le cadre d'étude des métamodèles multi-fidélité, **le tirage des données partiellement convergées initiales** est un point important de cette section.

L'idée est d'utiliser un premier tirage LHS que l'on viendra enrichir à l'aide d'un métamodèle créé à partir de cet échantillonnage initial. Ce genre de méthode est connue sous le nom d'échantillonnage adaptatif (*adaptive sampling* ou *adaptive design*) [Rosenbaum et Schulz, 2012, Picheny *et al.*, 2010]. Ce type d'approche est d'ailleurs tout à fait assimilable à ce qui a été fait précédemment pour les stratégies de recalage et de recherche du minimum (EGO) aux chapitres 2 et 3.

La technique que l'on se propose d'utiliser est l'une des plus simples à mettre en œuvre. Il s'agit d'effectuer un premier tirage LHS, de construire un métamodèle à partir de ce tirage initial et de venir l'enrichir au point maximisant l'erreur quadratique moyenne ([Martin et Simpson, 2002]). Cette stratégie s'est montrée intéressante dans le cadre du krigage ([Martin et Simpson, 2002, Jin *et al.*, 2002a, Rosenbaum et Schulz, 2012]). Dans le cadre des stratégies de métamodélisation multi-fidélité, on se propose d'effectuer ce type d'approche pour déterminer les  $10 \times n_{dim}$  données partiellement convergées préconisées pour la construction des métamodèles. **Le but recherché est d'améliorer la qualité des métamodèles multi-fidélité pour un coût de calcul similaire, puisque le même nombre de données partiellement convergées est utilisé.**

Dans cette section, il ne s'agit donc pas de comparer différentes méthodes de tirage mais d'en utiliser une et d'enrichir l'échantillonnage initial en fonction du métamodèle construit. Les méthodes de tirage ne sont en effet pas le cœur de cette thèse. Le lecteur intéressé pourra trouver des informations sur les comparatifs de différentes méthodes d'échantillonnage dans [Simpson *et al.*, 2001a, Simpson *et al.*, 2001b, Queipo *et al.*, 2005, Santner *et al.*, 2003, Myšáková *et al.*, 2014]. Concernant les méthodes d'échantillonnage adaptatif, des critères d'enrichissement autres que le maximum d'erreur quadratique moyenne du métamodèle sont pos-

sibles (comme par exemple des critères s'appuyant sur la validation croisée [Jin *et al.*, 2002b, Laurenceau et Sagaut, 2008] ou l'erreur quadratique moyenne intégrée [Sacks *et al.*, 1989]). On trouvera plus d'informations concernant ces méthodes dans [Rosenbaum, 2013] où une revue en est faite ainsi qu'une comparaison de trois d'entre elles.

Dans un premier paragraphe, on illustre la stratégie d'échantillonnage mise en place sur des fonctions tests analytiques et pour des métamodèles construits par krigeage. Dans un second temps, l'influence de l'échantillonnage initial des données partiellement convergées sur la qualité des métamodèles de cokrigeage multi-fidélité est étudiée. Ce travail est complété par les tests de certaines déclinaisons de calculs de données utilisées pour construire le métamodèle de cokrigeage, **notamment**  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$  ou  $\mathbf{X}_{fcv} \not\subset \mathbf{X}_{pcv}$ . La stratégie de recalage (permettant de construire un abaque virtuel) est également mise en place afin de s'intéresser à l'influence de ce tirage initial.

## 2.1 Méthode d'échantillonnage adaptatif avec un métamodèle de krigeage

Dans ce paragraphe, on s'intéresse à comparer sur des fonctions test, la qualité moyenne des métamodèles de krigeage obtenue par deux stratégies de tirage :

- Tirage LHS de  $k \times n_{dim}$  points
- Tirage LHS de  $k/2 \times n_{dim}$  points suivi d'un enrichissement de  $k/2 \times n_{dim}$  obtenus au maximum d'erreur quadratique moyenne du métamodèle : Méthode notée LHS adaptatif.

$k = 10$  ou  $k = 15$  pour les tests réalisés. Cette première étude s'appuie sur les fonctions de Rosenbrock de 2 à 5 dimensions et sur la fonction d'Hartmann à 6 dimensions. On réalise une centaine d'essais pour obtenir une qualité moyenne des stratégies employées.

La stratégie de LHS adaptatif est décrit par l'algorithme suivant :

---

### Algorithm 4 Algorithme LHS adaptatif

---

- 1: Tirage LHS de  $k/2 \times n_{dim}$  points (Remarque : si  $k = 15$  on utilise alors  $8 \times n_{dim}$  points)
  - 2:  $n_s = k/2 \times n_{dim}$  nombre de points utilisés pour construire le métamodèle
  - 3: **while**  $n_s < k \times n_{dim}$  **do**
  - 4:   Construction du métamodèle à partir des  $n_s$  points calculés
  - 5:   Recherche du point maximisant l'erreur quadratique moyenne
  - 6:   Evaluation de ce point
  - 7:    $n_s = n_s + 1$
  - 8: **end while**
  - 9: Construction du métamodèle final
- 

Les résultats de l'étude sont présentés dans le tableau 4.6.

	LHS				LHS adaptatif			
	$r_{ccc}$	$r^2$	$Q_1$	$Q_2$	$r_{ccc}$	$r^2$	$Q_1$	$Q_2$
Rosenbrock 2D k=10	0.999	0.999	0.0005	0.0036	<b>1</b>	<b>1</b>	$5.10^{-6}$	<b>0.0001</b>
Rosenbrock 3D k=10	0.942	0.92	0.16	17.47	<b>0.977</b>	<b>0.962</b>	<b>0.02</b>	<b>8.70</b>
Rosenbrock 4D k=10	0.852	<b>0.812</b>	0.29	<b>90.8</b>	<b>0.874</b>	0.787	<b>0.09</b>	112
Rosenbrock 4D k=15	0.971	0.969	0.09	20.0	<b>0.987</b>	<b>0.978</b>	<b>0.009</b>	<b>11.7</b>
Rosenbrock 5D k=10	0.79	<b>0.756</b>	0.34	<b>505</b>	<b>0.827</b>	0.711	<b>0.12</b>	661
Rosenbrock 5D k=15	0.928	0.927	0.15	187	<b>0.949</b>	<b>0.906</b>	<b>0.04</b>	<b>170</b>
Hartmann 6D k=10	0.79	0.705	0.35	715	<b>0.858</b>	<b>0.749</b>	<b>0.13</b>	<b>620</b>
Hartmann 6D k=15	0.879	0.807	0.21	446	<b>0.91</b>	<b>0.84</b>	<b>0.07</b>	<b>396</b>

**TABLE 4.6:** Résultats de l'étude comparative de la qualité de métamodèle obtenue pour les différentes fonctions tests

On constate donc que l'utilisation d'un tirage adaptatif n'amène pas nécessairement à une meilleure qualité de métamodèle, comme le montrent les indicateurs  $r^2$  et  $Q_2$ . En effet, pour les fonctions de Rosenbrock à 4 et 5 dimensions et avec  $k = 10$ , on remarque que le tirage LHS classique amène une corrélation  $r^2$  et un coefficient  $Q_2$  de meilleure qualité. Cette remarque est également présente dans la conclusion de [Jin *et al.*, 2002a]. Les stratégies d'échantillonnage adaptatif ne sont donc pas nécessairement meilleures que les stratégies de tirage dite *one-stage approach*. Le fait que le coefficient  $r^2$  soit de moins bonne qualité, implique que la forme globale du métamodèle est moins bonne. La localisation de l'optimum peut alors s'en voir plus ardue, ce qui n'est pas le but initialement recherché. Par contre, lorsque  $k = 15$ , la stratégie LHS adaptatif est la meilleure stratégie de tirage. On remarque également que concernant la fonction d'Hartmann, le tirage LHS adaptatif est le meilleur des deux tirages.

Néanmoins quelle que soit la fonction testée ou la valeur de  $k$ , le coefficient de corrélation  $r_{ccc}$  et le coefficient  $Q_1$  sont de meilleure qualité avec un tirage LHS adaptatif. La raison vient du fait que l'enrichissement au point maximisant l'erreur quadratique moyenne se fait souvent sur les bords du domaine de conception. Or, sur les fonctions testées, la frontière du domaine conditionne leur amplitude. L'information sur l'amplitude de la fonction test est donc plus précise et le coefficient  $r_{ccc}$  devient de ce fait plus précis avec la méthode de tirage LHS adaptatif. Il est en de même pour  $Q_1$ .

**Bilan :** On peut donc en déduire une dépendance de la qualité des métamodèles obtenus en

fonction des exemples analytiques étudiés et de la quantité de points calculés ( $k = 10$  ou  $k = 15$ ). Cependant, de manière générale cette méthode de tirage adaptatif se montre intéressante.

On s'intéresse dans le prochain paragraphe à étudier l'influence de différentes méthodes d'échantillonnage des données partiellement convergées pour un métamodèle de cokrigeage. La stratégie de recalage, en vue de construire un abaque virtuel est également testée. Dans le chapitre 3, avec la stratégie de recalage, les points enrichis étaient calculés partiellement et totalement convergés. On s'intéresse maintenant également à l'influence que peut avoir un point enrichi calculé uniquement totalement convergé lors de cette phase de recalage, sur la qualité de la réponse fournie par le métamodèle.

## 2.2 Influence du tirage pour un métamodèle de cokrigeage

Dans ce paragraphe, on s'applique à comparer les stratégies LHS et LHS adaptatif pour effectuer le tirage initial des données partiellement convergées dans le cadre des métamodèles multi-fidélité. La stratégie multi-fidélité utilisée dans un premier temps est le cokrigeage. Le cas test mécanique utilisé est le cas test 1\_rainure (la stratégie de recalage a été effectuée précédemment dessus et donc plusieurs données peuvent être comparées). On effectue de nouveau sur ce cas **la stratégie de recalage** afin d'étudier certaines déclinaisons de tirages possibles et comparer leur efficacité quant à construire un métamodèle de bonne qualité. Plusieurs paramètres sont donc pris en compte.

- La méthode de tirage des points partiellement convergés initiaux : LHS ou LHS adaptatif.
- La façon de choisir les  $3 \times n_{dim}$  premiers points totalement convergés pour construire le premier métamodèle multi-fidélité :  $\mathbf{X}_{fcv}^{3 \times n_{dim}} \subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$  ou  $\mathbf{X}_{fcv}^{3 \times n_{dim}} \not\subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$ . Avec  $\mathbf{X}_{fcv}^{3 \times n_{dim}}$  représentant les  $3 \times n_{dim}$  premiers points totalement convergés et  $\mathbf{X}_{pcv}^{10 \times n_{dim}}$  les  $10 \times n_{dim}$  premiers points partiellement convergés utilisés pour construire le métamodèle de cokrigeage.
- La façon d'enrichir le métamodèle de cokrigeage lors de la mise en place de la stratégie de recalage : effectuer le calcul du point enrichi partiellement et totalement convergé ou seulement totalement convergé.

Le tableau 4.7 présente l'ensemble des possibilités testées dans ce paragraphe.

	Descriptif de la méthode de tirage des points
T1	Tirage initial $\mathbf{X}_{pcv}^{10 \times n_{dim}}$ : $10 \times n_{dim}$ points LHS Tirage initial $\mathbf{X}_{fcv}^{3 \times n_{dim}}$ : $\mathbf{X}_{fcv}^{3 \times n_{dim}} \subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$ Manière de calculer un point lors de l'enrichissement par stratégie de recalage : partiellement et totalement convergé
T2	Tirage initial $\mathbf{X}_{pcv}^{10 \times n_{dim}}$ : $10 \times n_{dim}$ points LHS Tirage initial $\mathbf{X}_{fcv}^{3 \times n_{dim}}$ : $\mathbf{X}_{fcv}^{3 \times n_{dim}} \subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$ Manière de calculer un point lors de l'enrichissement par stratégie de recalage : seulement totalement convergé
T3	Tirage initial $\mathbf{X}_{pcv}^{10 \times n_{dim}}$ : $10 \times n_{dim}$ points LHS Tirage initial $\mathbf{X}_{fcv}^{3 \times n_{dim}}$ : $\mathbf{X}_{fcv}^{3 \times n_{dim}} \not\subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$ Manière de calculer un point lors de l'enrichissement par stratégie de recalage : seulement totalement convergé
T4	Tirage initial $\mathbf{X}_{pcv}^{10 \times n_{dim}}$ : $10 \times n_{dim}$ points LHS adaptatif Tirage initial $\mathbf{X}_{fcv}^{3 \times n_{dim}}$ : $\mathbf{X}_{fcv}^{3 \times n_{dim}} \subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$ Manière de calculer un point lors de l'enrichissement par stratégie de recalage : partiellement et totalement convergé
T5	Tirage initial $\mathbf{X}_{pcv}^{10 \times n_{dim}}$ : $10 \times n_{dim}$ points LHS adaptatif Tirage initial $\mathbf{X}_{fcv}^{3 \times n_{dim}}$ : $\mathbf{X}_{fcv}^{3 \times n_{dim}} \subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$ Manière de calculer un point lors de l'enrichissement par stratégie de recalage : seulement totalement convergé

**TABLE 4.7:** Stratégies de tirage testées au cours de ce paragraphe et de cette étude

Une première étude est menée lors de la mise en place de recalage pour les tirages T1, T2 et T3. L'élément commun entre ces méthodes est le tirage initial des points  $\mathbf{X}_{pcv}^{10 \times n_{dim}}$  qui est effectué par tirage LHS.

**Remarques :** le tirage T1 est le tirage qui a été utilisé précédemment au chapitre 3 pour la construction de métamodèle de cokrigeage multi-fidélité.  
 Pour le tirage T1, on vérifie  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$  tandis que pour T3 on vérifie  $\mathbf{X}_{fcv} \not\subset \mathbf{X}_{pcv}$ .  
 Aux tableaux 4.8 et 4.9, la corrélation initiale fait référence à la corrélation entre le premier métamodèle de cokrigeage (construit avec 20 points partiellement convergés et 6 points totalement convergés) et le métamodèle de référence. La corrélation finale fait référence à la qualité du dernier métamodèle créé lorsque le critère  $\bar{r}_{ccc}^3$  est atteint.

Les résultats de cette première étude sont présentés au tableau 4.8. On constate clairement que la propriété  $\mathbf{X}_{fcv}^{3 \times n_{dim}} \subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$  (tirages T1 et T2) ou  $\mathbf{X}_{fcv}^{3 \times n_{dim}} \not\subset \mathbf{X}_{pcv}^{10 \times n_{dim}}$  (tirage T3) a un impact limité sur la qualité initiale du métamodèle créé (voir colonne « Corrélation Initiale »).

Pour atteindre une corrélation  $r_{ccc} > 0.95$ , on constate que le tirage T1 demande, en moyenne, légèrement moins de points totalement convergés qu'avec le tirage T2 et surtout T3.

	Corr. initiale		Nbre pts pour corr. 0.95	Temps (s)	Nbre pts pour crit. arrêt $\bar{r}_{ccc}^{v^3}$	Corr. finale		Temps (s)
	$r_{ccc}$	$r^2$				$r_{ccc}$	$r^2$	
T1	0.73	0.64	13.8	488	20.5	0.97	0.95	639
T2			14.5	468	19.9	0.97	0.94	579
T3	0.73	0.62	17.6	574	23.4	0.97	0.95	696

**TABLE 4.8:** Résultats de la stratégie de recalage en fonction des tirages T1, T2, et T3

Le complément d'information qu'apporte les données partiellement convergées permet donc de limiter le nombre de points à enrichir pour atteindre un certain niveau de qualité. Cependant, en terme de temps de calcul, il peut être intéressant de ne pas effectuer l'enrichissement lors de la phase de recalage au niveau basse fidélité. En effet, le temps de calcul associé à T2 bien que très proche de celui obtenu par T1 est légèrement inférieur. De plus, le tirage T2 est plus intéressant que le tirage T3. Le fait de vérifier  $\mathbf{X}_{fcv}^{3 \times ndim} \subset \mathbf{X}_{pcv}^{10 \times ndim}$  est donc une source d'information limitant le temps de calcul pour atteindre les différents critères d'arrêt. On peut donc en conclure qu'avoir en commun des points totalement et partiellement convergés est intéressant du point de vue de la construction de métamodèle.

Le bilan des résultats obtenus pour atteindre le critère d'arrêt  $\bar{r}_{ccc}^{v^3}$  est tout à fait semblable à ce qui a été dit précédemment.

**Bilan :** Les tirages de type T1 ou T2 apportent des résultats tout à fait comparables. Le tirage de type T3 est le tirage le moins performant testé. Par rapport aux deux autres il a pour propriété de vérifier  $\mathbf{X}_{fcv} \not\subset \mathbf{X}_{pcv}$ . Ainsi, avoir dans l'échantillonnage servant à construire le métamodèle de cokrigeage des points totalement et partiellement convergés en commun amène à une meilleure qualité de métamodèle. Pour rappel T1 vérifie  $\mathbf{X}_{fcv} \subset \mathbf{X}_{pcv}$  et T2 vérifie  $\mathbf{X}_{fcv}^{3 \times ndim} \subset \mathbf{X}_{pcv}^{10 \times ndim}$ .

Dans la suite de cette étude, on va comparer les méthodes de type T1 et T2 aux méthodes de type T4 et T5 qui ont pour différence d'avoir un tirage initial des données partiellement convergées effectué par un tirage LHS adaptatif. Ce tirage est réalisé comme décrit à l'algorithme 4. Les résultats sont présentés au tableau 4.9.

	Corr. initiale		Nbre pts pour corr. 0.95	Temps (s)	Nbre pts pour crit. arrêt $\bar{r}_{ccc}^{v^3}$	Corr. finale		Temps (s)
	$r_{ccc}$	$r^2$				$r_{ccc}$	$r^2$	
T1	0.73	0.64	13.8	488	20.5	0.97	0.95	639
T2			14.5	468	19.9	0.97	0.94	579
T4	0.88	0.79	13.5	481	22	0.97	0.95	685
T5			14.7	470	19.4	0.96	0.93	572

**TABLE 4.9:** Comparaison des résultats de la stratégie de recalage en fonction des tirages T1, T2, T4 et T5

**Bilan :** On constate que la différence principale entre les méthodes de tirage LHS (T1 et T2) et de tirage LHS adaptatif (T4 et T5) se situe au niveau de la qualité initiale du métamodèle multi-fidélité créé. La qualité est en effet grandement améliorée. La raison de cette amélioration est que les données partiellement convergées obtenues par le tirage LHS adaptatif sont plus représentative du modèle basse fidélité (la corrélation entre le métamodèle des données partiellement convergées et le modèle basse fidélité est  $r_{ccc}=0.99$  et  $r^2=0.99$  avec le tirage LHS adaptatif contre  $r_{ccc}=0.80$  et  $r^2=0.71$  avec le tirage LHS).

Cependant, bien que la corrélation initiale soit augmentée avec les méthodes T4 et T5, le temps de calcul pour atteindre les différents critères d'arrêt est équivalent aux méthodes T1 et T2. De nouveau, ceci tend donc à montrer qu'à **haute qualité exigée** l'enrichissement joue un rôle primordial (voir chapitre 3). Néanmoins, obtenir pour un **coût de calcul similaire** un métamodèle multi-fidélité de **meilleure qualité** peut permettre de rendre plus efficace une stratégie de type EGO. En effet, si le métamodèle sur lequel s'appuie cette approche est de meilleure qualité, la convergence vers le minimum global peut s'en voir accéléré.

A titre informatif, on fournit la qualité moyenne initiale d'un métamodèle de cokrigage construit avec  $20 \times n_{dim}$  points LHS partiellement convergés et  $3 \times n_{dim}$  points totalement convergés. On obtient alors  $r_{ccc} = 0.81$  et  $r^2 = 0.75$ . La qualité obtenue est également inférieure à celle que l'on peut atteindre en utilisant le tirage LHS adaptatif. Cette information montre donc qu'au lieu de privilégier la puissance de calcul, l'utilisation du tirage LHS adaptatif permet en un temps de calcul réduit d'obtenir des métamodèles de meilleure qualité.

Cette perspective est un résultat, qui dans le cadre des métamodèles multi-fidélité, peut s'avérer très intéressant, si le coût de calcul d'une donnée partiellement convergée est jugé non négligeable (voir cas test développé à la section 2.2 du chapitre 5 où le coût de calcul d'une donnée partiellement convergée est d'une 1h20 contre 12h pour une donnée totalement convergée).

Des résultats tout à fait similaires sont obtenus avec la méthode d'Evofusion et de krigage hiérarchique. Originellement, pour l'Evofusion, les points  $\mathbf{X}_{fcv}$  étaient directement issus du maximum d'erreur quadratique du métamodèle d'erreur. Il n'y avait donc pas de tirage des premiers points  $\mathbf{X}_{fcv}^{3 \times n_{dim}}$  comme avec T1, T4 ou T5. Cependant, les résultats obtenus avec cette méthode de tirage des points totalement convergés et la méthode T1 sont tout à fait identiques en terme de temps de calcul pour atteindre les critères d'arrêt et en terme de qualité pour cette **méthode de recalage**. En effet, pour la recherche de minimum par stratégie de type EGO, les résultats présentés sont différents (voir la section suivante).

Dans les deux prochaines parties, on se propose d'effectuer l'optimisation d'un cas test mécanique, dans le cadre de la stratégie d'optimisation multi-niveaux décrite au chapitre 1 et plus précisément au paragraphe 1.4.6. Pour rappel, l'optimisation se fait sur deux niveaux.

Une première optimisation globale s'effectue sur un métamodèle sur lequel une stratégie d'enrichissement de type EGO visant à localiser la zone du minimum est mise en place.

Le second niveau correspond à une optimisation locale sur le modèle mécanique réel. Le point de départ de cette optimisation locale étant le minimum du métamodèle précédemment localisé.



### 3 Optimisation d'un cas test mécanique : premier niveau d'optimisation

On se propose dans cette section d'effectuer une étude sur le premier niveau d'optimisation. Différentes méthodes multi-fidélité (Evofusion, cokrigeage et krigeage hiérarchique), différents critères d'arrêt de la stratégie EGO ( $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$  et  $\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$ ) ainsi que différents tirages de points (T1, T4 et T5) sont comparés sur un cas test mécanique. Cette étude vise à déterminer quelle est la méthode multi-fidélité permettant de répondre le plus efficacement au problème d'optimisation proposé.

Les différentes méthodes sont comparées entre elles mais également avec une méthode de référence. Cette méthode de référence, précédemment utilisée au chapitre 2, est décrite en figure 2.11. Pour rappel, celle-ci consiste à construire un premier métamodèle de krigeage à l'aide de  $10 \times n_{dim}$  points totalement convergés qui est par la suite enrichi au point maximisant l'EI. Celle-ci ne s'appuie que sur l'utilisation de données totalement convergées.

On présente dans un premier temps le cas test mécanique sur lequel l'optimisation est effectuée.

#### 3.1 Présentation du cas test 2\_rainures

Un cas test inédit est ici présenté. Celui-ci a été développé afin d'avoir une fonction objectif multimodale avec plusieurs minima (un global et deux locaux).

En effet, le cas test 1\_rainure a été jugé trop peu discriminant car les deux minima sont relativement proches l'un de l'autre (voir paragraphe 2.5 du chapitre 3). Il était donc difficile de juger des performances atteintes par les métamodèles multi-fidélité dans un contexte de recherche du minimum global.

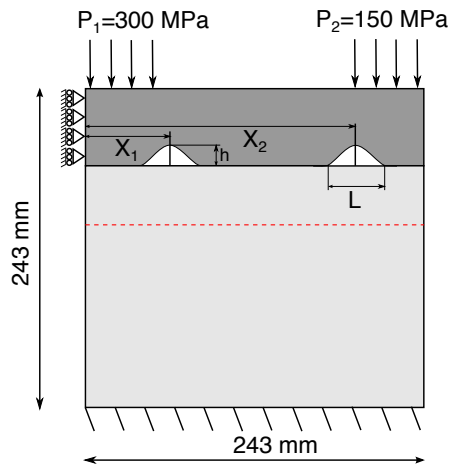
Cependant, dans sa conception le cas test 2\_rainures s'appuie sur le précédent. Deux rainures sont introduites au lieu d'une. Celles-ci sont définies par leur position  $X_1$  et  $X_2$ , comme indiqué sur la figure 4.1. La fonction objectif étudiée est la même que précédemment (voir équation 3.83).

Les variables du problème  $X_1$  et  $X_2$  sont définies dans le tableau 4.10

Paramètres	min	max
$X_1$ (en mm)	35	100
$X_2$ (en mm)	135	208

**TABLE 4.10:** Bornes des variables de conception du cas test 2\_rainures

Une relation de dépendance interne entre  $L$  et les variables  $X_1$  et  $X_2$  est mise en place (voir figure 4.2). Ceci a été effectué afin de renforcer le caractère multimodal du problème et permet d'obtenir une surface de réponse représentative de ce qui peut se faire en mécanique des assemblages comme par exemple la discontinuité des gradients (voir figure 4.3).



- \* Hauteur des rainures :  $h=0.25\text{mm}$
- \* Position de la rainure 1 :  $X_1$
- \* Position de la rainure 2 :  $X_2$
- \* Largeur des rainures :  $L$  (dépend de  $X_1$  et  $X_2$ )

FIGURE 4.1: Représentation du cas test 2\_rainures

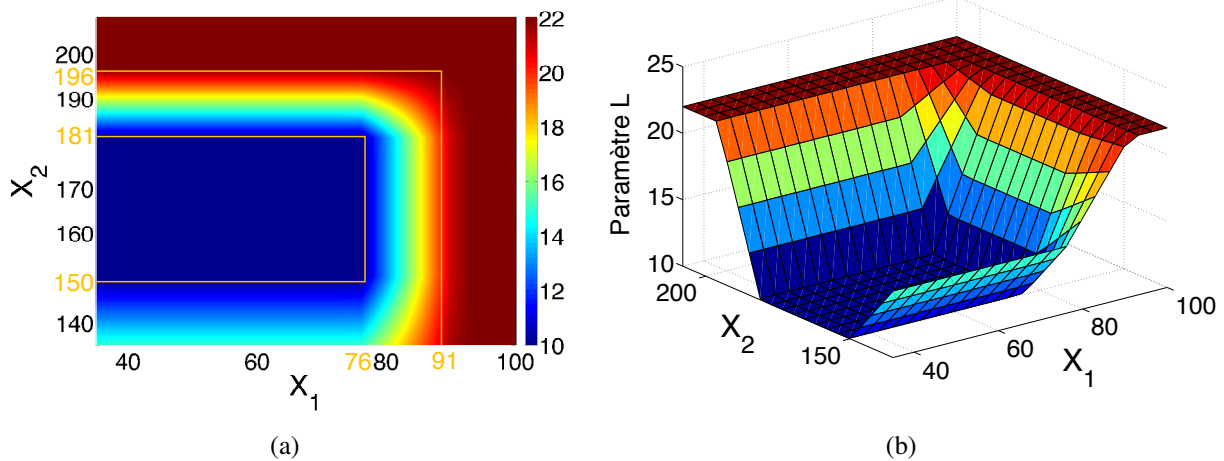


FIGURE 4.2: Valeur du paramètre L en fonction de  $X_1$  et  $X_2$

Le minimum global est défini en  $\mathbf{x}_{\min} = [76 \ 181]$  et  $y(\mathbf{x}_{\min}) = 1.505$ . Un premier minimum local est situé en  $\mathbf{x}_{\text{loc}1} = [100 \ 208]$  et  $y(\mathbf{x}_{\text{loc}1}) = 1.53$ . Un second minimum local est situé en  $\mathbf{x}_{\text{loc}2} = [100 \ 135]$  et  $y(\mathbf{x}_{\text{loc}2}) = 1.605$ .

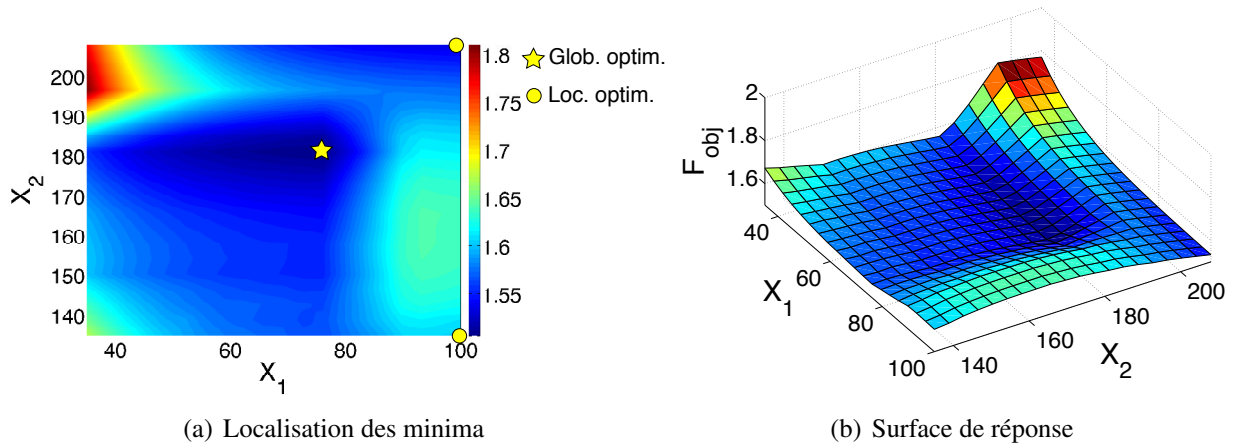


FIGURE 4.3: Surface de réponse en fonction des paramètres  $X_1$  et  $X_2$

## 3.2 Traitement du cas test 2\_rainures

A partir des résultats obtenus au chapitre 2 et 3, le niveau de convergence des données partiellement convergées est fixé à  $\eta = 0.05$  (temps de calcul CPU : 3.1s). Le niveau de référence des données totalement convergées est fixé pour  $\eta = 10^{-4}$  (temps de calcul CPU : 92s). Chaque méthode testée est répétée pour 20 tirages initiaux différents.

### 3.2.1 Méthodes de métamodélisation utilisées sur le cas test

Cinq approches sont comparées lors de cette étude :

- La stratégie d'Evofusion originelle ne s'appuyant au départ que sur un premier métamodèle construit avec  $10 \times n_{dim}$  points partiellement convergés. L'algorithme lié à cette méthode est le même que celui utilisé au chapitre 2 et décrit à la figure 2.3. Sur ce cas test l'enrichissement de la stratégie EGO se fait au maximum d'EI. Sur les prochaines figures, cette approche est notée Evof.
- La stratégie de cokrigeage dont les métamodèles sont construits à l'aide des approches de tirage T1 (méthode de tirage de référence), T4 ou T5. L'enrichissement lors de la mise en place de la stratégie EGO se fait alors au maximum d'EI (voir paragraphe 3.2.2). L'algorithme précédemment décrit à la figure 3.7 présente la stratégie utilisée lors de l'utilisation du type de tirage T1. Pour les tirages T4 et T5, il suffit d'adapter le tirage LHS des données partiellement convergées en tirage LHS adaptatif et pour le tirage de type T5 de calculer seulement de manière totalement convergée le point maximisant l'EI.
- La stratégie de krigeage hiérarchique est effectuée de la même manière que pour le cokrigeage (voir point précédent).
- La stratégie d'Evofusion dont le premier métamodèle évofusé est construit à l'aide de  $10 \times n_{dim}$  données partiellement convergées et  $3 \times n_{dim}$  points totalement convergés. Les

méthodes de tirage testées sont alors T1, T4 et T5. Cette méthode est alors dans sa construction tout à fait similaire à ce qui est fait pour le cokrigeage et le krigeage hiérarchique (voir figure 3.7).

Par rapport à l'approche d'Evofusion originelle, la différence est que la mise en place de la stratégie EGO se fait sur un métamodèle évofusé ayant déjà une source d'information provenant de données totalement convergées. On verra que cette source d'information préalable par des données totalement convergées a une influence certaine sur les résultats. Cette approche est notée Evof. enri. (Evofusion enrichie)

- L'approche de référence, s'appuyant sur un métamodèle de krigeage construit initialement avec  $10 \times n_{dim}$  points LHS totalement convergés. L'enrichissement se fait au point maximisant l'EI (voir figure 2.11). Cette approche est notée dans la suite Ref.

### 3.2.2 Formulation de l'expected improvement pour les méthodes multi-fidélité

Pour rappel, le calcul de l'*expected improvement* (EI) dans le cadre d'un métamodèle construit avec du krigeage se fait via la formule décrite à l'équation 1.66.

Pour l'Evofusion, on a vu au chapitre 2 que la formulation en était similaire en s'appuyant sur les données totalement convergées et partiellement convergées corrigées par le métamodèle d'erreur.

Dans le cadre des stratégies de cokrigeage et de krigeage hiérarchique l'équation de l'*expected improvement* est donnée par : [Forrester *et al.*, 2007]

$$EI(\mathbf{x}) = \begin{cases} (\min \{ \mathbf{y}_{fcv} \} - \hat{y}^m(\mathbf{x})) \Phi \left( \frac{\min \{ \mathbf{y}_{fcv} \} - \hat{y}^m(\mathbf{x})}{s_{\hat{y}^m}(\mathbf{x})} \right) \\ \quad + s_{\hat{y}^m}^2(\mathbf{x}) \phi \left( \frac{\min \{ \mathbf{Y}_{fcv} \} - \hat{y}^m(\mathbf{x})}{s_{\hat{y}^m}(\mathbf{x})} \right) & \text{si } s_{\hat{y}^m}(\mathbf{x}) > 0 \\ 0 & \text{si } s_{\hat{y}^m}(\mathbf{x}) = 0 \end{cases} \quad (4.13)$$

$\mathbf{y}_{fcv}$  correspond aux données calculées totalement convergées,  $\hat{y}^m(\mathbf{x})$  correspond à l'évaluation du métamodèle multi-fidélité et  $s_{\hat{y}^m}^2(\mathbf{x})$  à l'erreur quadratique moyenne de celui-ci.

### 3.2.3 Résultats de l'étude

Dans ce paragraphe, on présente les résultats de l'étude menée. Ceux-ci sont exposés en plusieurs parties :

- On introduit tout d'abord la corrélation moyenne  $r^2$  entre les différents métamodèles construits **avant la mise en place de l'enrichissement par approche EGO** et un métamodèle de référence (construit avec un plan factoriel de  $20 \times 20$  points). Cette information va permettre notamment de comparer les qualités obtenues suivant les différentes métamodélisations testées mais également suivant les types de tirages effectués.

- Ensuite, on présente la qualité du minimum obtenu après enrichissement par l'approche EGO. Ces résultats sont exposés selon les différentes métamodélisations testées ainsi que selon les tirages et les critères d'arrêt comparés.
- Par la suite, on présente les temps de calcul nécessaires pour atteindre les critères d'arrêt  $\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$  et  $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$ , en fonction de l'utilisation ou non de la stratégie de calcul LaTin multi-paramétrique (les résultats obtenus par le critère d'arrêt  $\tilde{\Delta}_{j\ 0.005}^{n_{dim}+1}$  sont tout à fait semblables à ceux obtenus avec  $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$  et ne sont donc pas présentés).
- Afin de compléter l'étude, on présente également la précision du minimum obtenu à l'aide du métamodèle en fonction du temps de calcul. Ceci permet de comparer l'efficacité des différentes méthodes de métamodélisation en fonction du coût de calcul.

### Qualité initiale des métamodèles avant enrichissement par approche EGO

Le métamodèle obtenu par les données partiellement convergées est de très faible qualité (voir Evof. sur la figure 4.4) par rapport au métamodèle de référence. La mise en place de la stratégie EGO sur la méthode d'Evofusion se fait donc sur un métamodèle ayant seulement une corrélation  $r^2 = 0.3$  par rapport au métamodèle de référence.

Les métamodèles multi-fidélité initiaux d'Evofusion enrichie, de cokrigage et krigeage hiérarchique (construits respectivement avec 20 et 6 points partiellement et totalement convergés) sont de meilleure qualité avec des tirages de type T4 ou T5. On remarque d'ailleurs que cette amélioration est plus significative pour les méthodes d'Evofusion enrichie et de cokrigage. Il est également intéressant de remarquer que le métamodèle obtenu par des approches multi-fidélité est d'aussi bonne facture qu'un métamodèle de krigeage construit avec 20 points LHS (voir Ref. sur la figure 4.4).

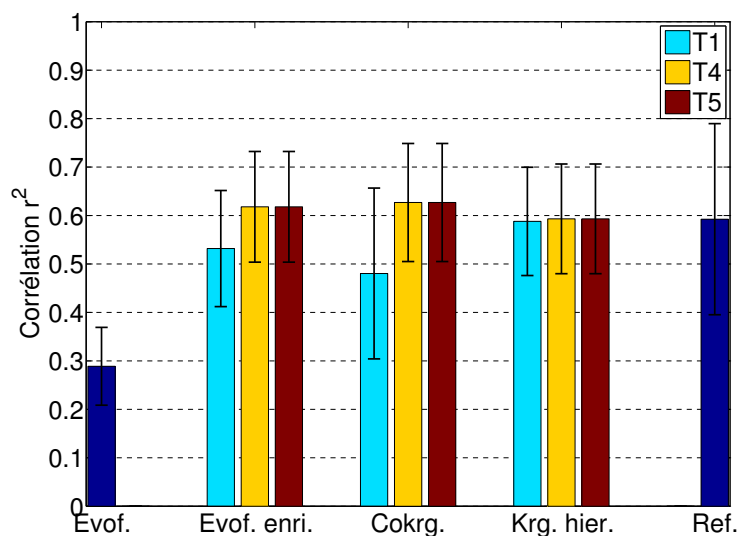


FIGURE 4.4: Corrélation  $r^2$  pour les différentes méthodes et tirages

On s'intéresse alors à la qualité de minimum obtenu lorsque les différents critères d'arrêt  $\tilde{\Delta}_h^{n_{dim}+1}$  et  $\tilde{\Delta}_m^{n_{dim}+1}$  ont été atteints.

### Qualité du minimum trouvé à la suite de la stratégie EGO

Pour rappel, l'étude a été menée pour 20 essais numériques différents. On présente ici la qualité des minima obtenus pour ces 20 essais numériques. Tout d'abord à la figure 4.5 on présente sur les 20 essais menés pour chacune des métamodélisations et pour chacun des tirages, le nombre d'essais numériques ayant abouti à la localisation de la zone où se situe le minimum global.

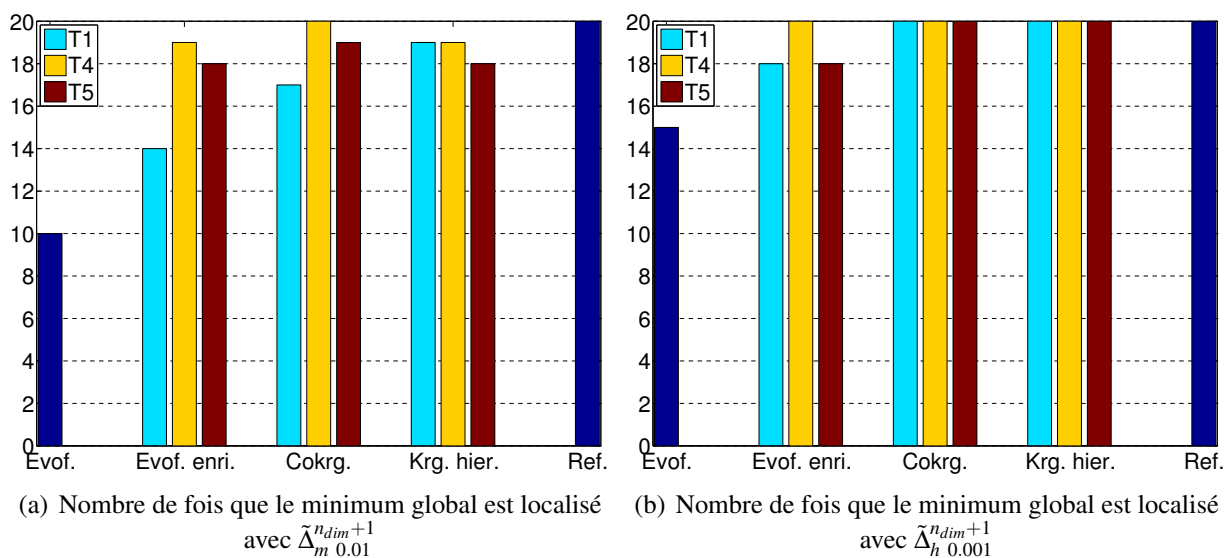


FIGURE 4.5: Nombre d'essais numériques aboutissant à localiser la zone du minimum global

On constate que comme pour l'étude menée au paragraphe 1.3, l'utilisation du critère d'arrêt  $\tilde{\Delta}_m^{n_{dim}+1}$  tend à fournir des résultats de moins bonne qualité qu'avec l'utilisation du critère  $\tilde{\Delta}_h^{n_{dim}+1}$ .

Pour les méthodes d'Evofusion enrichie et de cokrigeage, l'utilisation des tirages T4 et T5 apporte une amélioration significative des résultats obtenus par rapport au tirage T1. L'apport du tirage LHS adaptatif des données partiellement convergées est donc intéressant d'un point de vue de la qualité de métamodèle obtenu et de la qualité du minimum trouvé suite à la procédure EGO.

Cependant, au niveau de la qualité des résultats obtenus, le tirage T4 semble plus adapté si l'on choisit d'utiliser  $\tilde{\Delta}_m^{n_{dim}+1}$  comme critère d'arrêt. Il permet d'atteindre de manière plus efficace la zone du minimum global.

On constate également que l'Evofusion (Evof. sur les figures) est peu adapté sur ce cas test. En effet, le minimum global est localisé 10 fois sur 20 essais par utilisation du critère d'arrêt  $\tilde{\Delta}_m^{n_{dim}+1}$  et 15 fois sur 20 avec le critère d'arrêt  $\tilde{\Delta}_h^{n_{dim}+1}$ . Il s'agit de la méthode la moins performante du point de vue de la qualité du minimum trouvé.

La figure 4.6 montre la distance moyenne des minima obtenus suite à la procédure EGO.

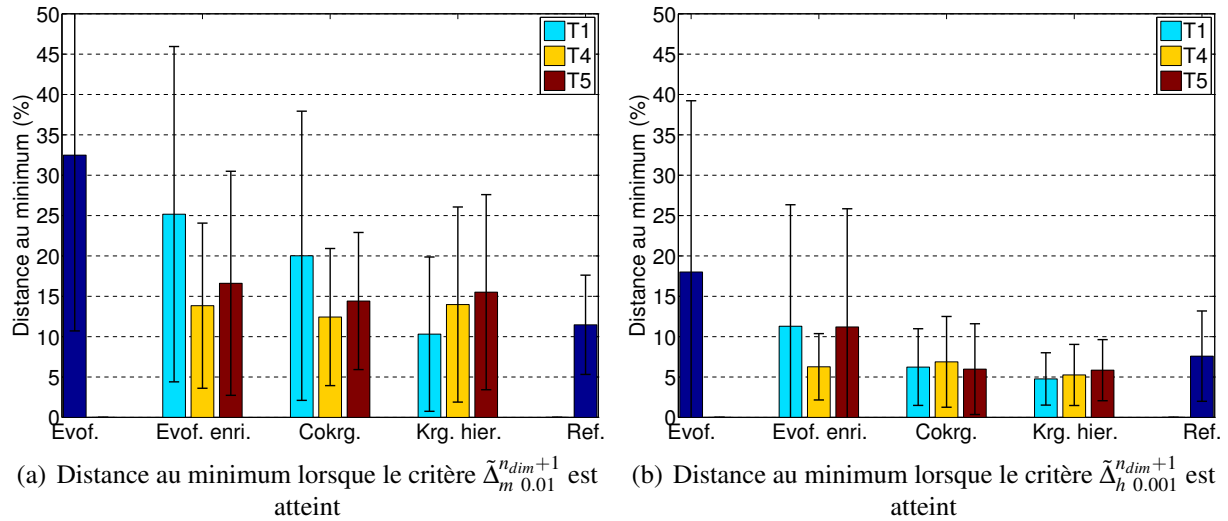


FIGURE 4.6: Qualité du minimum trouvé par les métamodèles multi-fidélité

A la figure 4.6(b), on constate que pour les méthodes de cokrigage, de krigeage hiérarchique et d'Evofusion enrichie avec utilisation du tirage T4, le minimum localisé est plus précis qu'avec la méthode de référence. La distance moyenne au minimum avec la méthode de référence est de 7.6% contre 4.8 à 6.9% pour ces méthodes multi-fidélité.

- Ce résultat est important puisqu'il met en avant la qualité des résultats obtenus par les approches multi-fidélité lorsque le critère d'arrêt  $\tilde{\Delta}_h^{n_{dim}+1}$  est utilisé. Pour un ingénieur ne se basant que sur une stratégie EGO pour effectuer l'optimisation du problème mécanique d'assemblage, l'utilisation des approches multi-fidélité permet alors une meilleure précision.
- Par contre, si l'on souhaite utiliser le critère d'arrêt  $\tilde{\Delta}_m^{n_{dim}+1}$  dans le but de localiser la zone du minimum, les résultats obtenus sont légèrement moins précis (12.4% avec la méthode de cokrigage et tirage T4 contre 11.5% pour l'approche de référence).

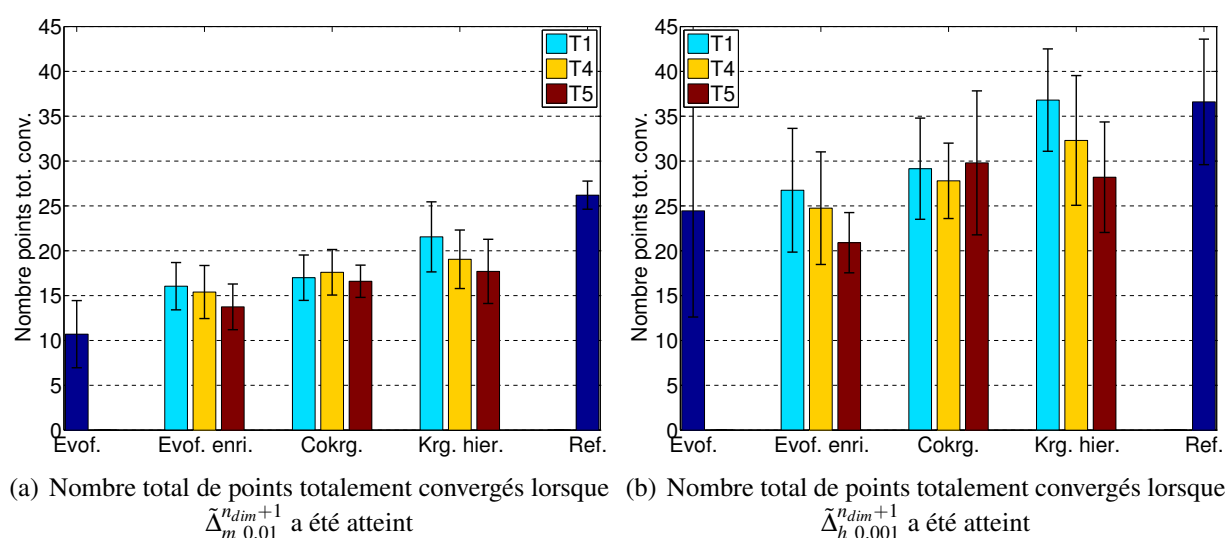
Les résultats obtenus par les méthodes d'Evofusion enrichie et de krigeage hiérarchique sont également de bonne qualité. Mais, sur tous les 20 tests menés, aucune de ces méthodes quel que soit le tirage utilisé n'a permis de localiser à 100% le minimum global (voir figure 4.5(a)).

**Bilan :** Les méthodes multi-fidélité peuvent être très bien adaptées pour mener à bien une stratégie de type EGO afin de localiser le minimum global. L'utilisation du tirage T4 augmente le taux de réussite à localiser ce minimum global. Le critère d'arrêt  $\tilde{\Delta}_h^{n_{dim}+1}$  amène d'ailleurs à une meilleure localisation du minimum global qu'avec la stratégie de référence. Cependant, afin de compléter l'étude, il est nécessaire de présenter les coûts de calcul nécessaires pour atteindre les deux critères d'arrêt  $\tilde{\Delta}_h^{n_{dim}+1}$  et  $\tilde{\Delta}_m^{n_{dim}+1}$ . On constatera notamment que la méthode de référence est une méthode coûteuse par rapport au cokrigage qui a permis (tout comme cette dernière) de localiser efficacement le minimum global.

### Temps de calcul des différentes approches

Les résultats présentés dans ce paragraphe sont principalement utilisés à titre de comparaison. Ils mettent en avant les coûts de calcul nécessaires pour atteindre les critères d'arrêt. Ils complètent les résultats décrits au paragraphe précédent afin d'identifier les méthodes multi-fidélité les plus performantes.

On présente à la figure 4.7, le nombre **total** de points totalement convergés qui ont été calculés pour l'ensemble des méthodes et des critères d'arrêt testés. Ce nombre de points tient compte des  $3 \times n_{dim} = 6$  premiers points utilisés pour créer le premier métamodèle multi-fidélité des méthodes Evofusion enrichie, cokrigage et krigeage hiérarchique.



**FIGURE 4.7:** Nombre de points totalement convergés calculés lorsque les différents critères d'arrêt ont été atteints

Plusieurs constatations peuvent être tirées de ces figures.

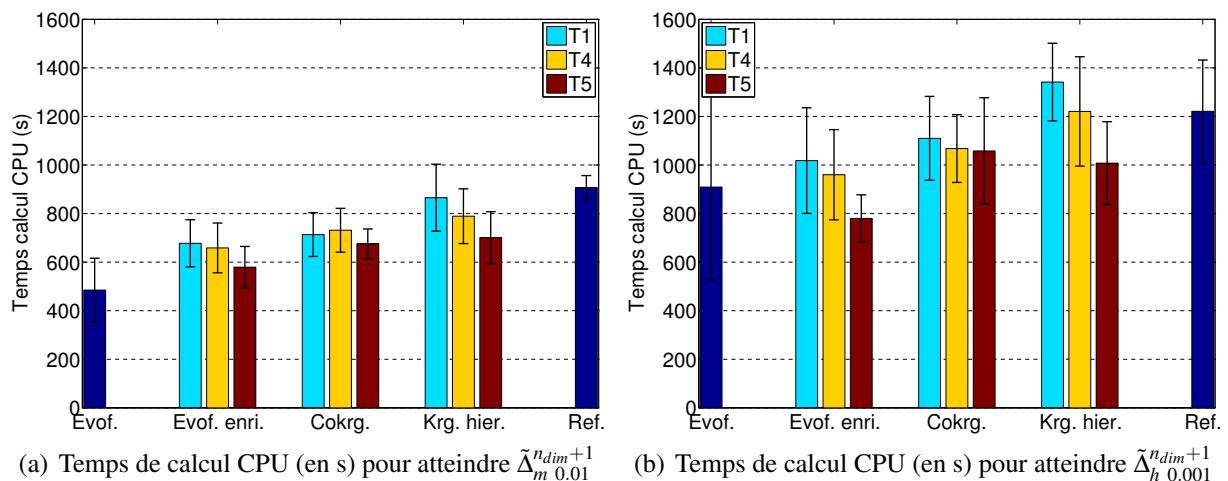
- La méthode de référence est clairement l'une des plus exigeantes en terme de points totalement convergés. Seule la méthode de krigeage hiérarchique pour atteindre  $\tilde{\Delta}_h^{n_{dim}+1}$  et utilisant un tirage de type T1 est légèrement plus coûteuse.
- Une forte économie peut être envisagée en terme de points totalement convergés à calculer grâce à l'utilisation du critère d'arrêt  $\tilde{\Delta}_m^{n_{dim}+1}$  par rapport à  $\tilde{\Delta}_h^{n_{dim}+1}$ . Globalement sur l'ensemble des méthodes et pour chaque type de tirage, on peut diminuer ce nombre de points totalement convergés de 30 à 40%. L'économie réalisée n'est donc pas négligeable. De plus, on a vu que le cokrigage avec le tirage T4 a permis de localiser le minimum global pour les 20 essais numériques. Une économie de temps de calcul peut donc être envisagée sans nuire à la localisation de la zone du minimum global.
- Le faible coût associé à la méthode d'Evofusion (voir Evof. sur les différentes figures) couplé à la qualité relativement mauvaise des données partiellement convergées utilisées



initialement pour mener la stratégie EGO (voir figure 4.4), explique le faible taux de réussite à localiser le minimum global. La raison expliquant ce faible coût de calcul pour atteindre les critères d'arrêt est qu'il est supposé que les données partiellement convergées corrigées par le métamodèle d'erreur ont une variance nulle et donc un EI nul. La valeur maximale de l'EI est ainsi plus faible et amène à atteindre les critères d'arrêt plus rapidement que pour les autres méthodes. L'utilisation initiale de données totalement convergées pour l'approche d'Evofusion (voir Evof. enri.) permet en un temps de calcul relativement semblable d'améliorer fortement la qualité des résultats obtenus (voir figures 4.5 et 4.6).

- On peut remarquer que la méthode de cokrigage demande un nombre de points totalement convergés relativement identique suivant les méthodes de tirage T1, T4 ou T5. Pour les méthodes d'Evofusion et de krigeage hiérarchique on constate une décroissance de points totalement convergés à calculer suivant ces méthodes de tirage.

A titre informatif, on présente à la figure 4.8, les temps de calcul (avec approche multiparamétrique) des différentes méthodes testées. Les conclusions que l'on peut tirer de cette figure sont tout à fait semblables à celles décrites précédemment (il en va de même pour les résultats sans le calcul des données par approche multiparamétrique).



**FIGURE 4.8:** Temps de calcul CPU nécessaire pour atteindre les critères d'arrêt

**Remarque :** Pour la méthode de cokrigage, une description plus précise des gains atteints selon le critère d'arrêt utilisé et l'utilisation ou non de la stratégie multiparamétrique sera fournie au paragraphe 3.2.5.

**Bilan :** Au vu des résultats sur la qualité du minimum trouvé et aux temps de calculs nécessaires pour atteindre les différents critères d'arrêt, la méthode de cokrigage avec tirage T4 semble être la méthode la plus adaptée pour répondre à notre problématique d'optimisation d'assemblage mécanique. En effet, quel que soit le critère d'arrêt utilisé, la localisation de la zone

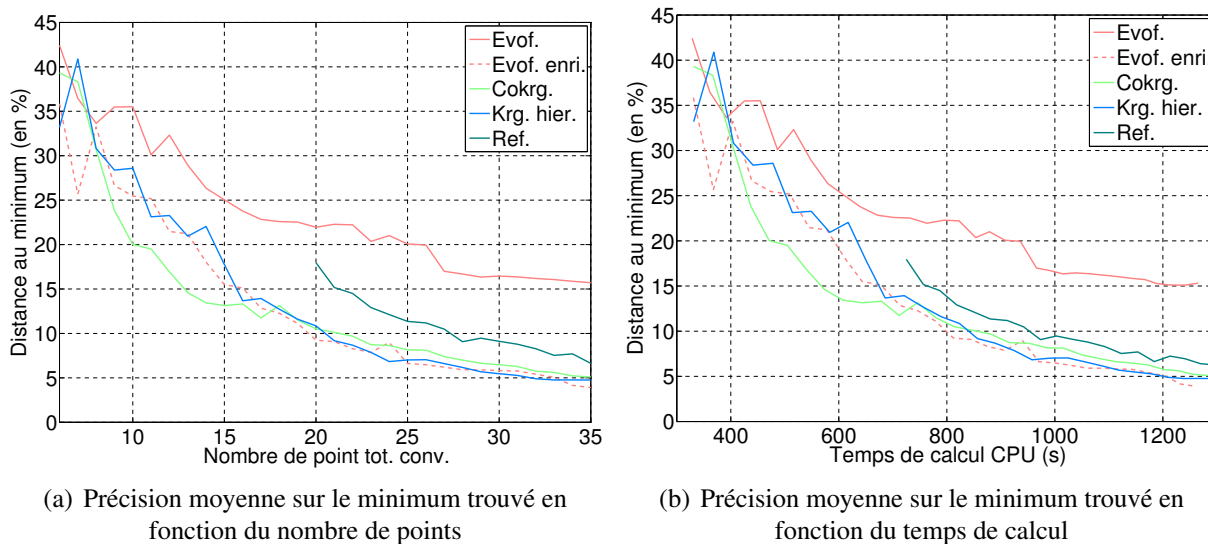
du minimum global a été une réussite. De plus, cette technique est moins coûteuse en terme de temps de calcul que la méthode de référence. Elle est également moins coûteuse que la méthode de krigeage hiérarchique (qui s'est également avérée assez performante pour localiser le minimum global de la fonction objectif, voir figures 4.5 et 4.6). Le temps de calcul de la méthode d'Evofusion enrichie est légèrement plus faible que le cokrigeage mais la qualité des résultats obtenus est inférieure.

**Le procédé de cokrigeage multi-fidélité avec un tirage de type T4** est donc le meilleur compromis entre temps de calcul et qualité du résultat obtenu.

L'actuelle étude comparative est effectuée par rapport à des critères d'arrêt de la stratégie EGO. Un autre élément de comparaison peut être la précision sur le minimum recherché en fonction du temps de calcul. Les résultats présentés au paragraphe suivant vont permettre de confirmer que le cokrigeage est bien adapté à notre problématique d'optimisation.

### Comparaison de la précision du minimum obtenu par le métamodèle en fonction du temps de calcul

La figure 4.9 illustre la qualité moyenne du minimum trouvé à l'aide des différents métamodèles en fonction du coût de calcul. On précise que le tirage utilisé pour les méthodes d'Evofusion enrichie, de cokrigeage et de krigeage hiérarchique est de type T4.



**FIGURE 4.9:** Précision moyenne sur le minimum trouvé par les métamodèles en fonction du coût de calcul

On constate clairement que l'usage des métamodèles multi-fidélité d'Evofusion enrichie, de cokrigeage et de krigeage hiérarchique amène une meilleure précision de résultat en fonction du temps de calcul que la méthode de référence. Cette figure illustre de nouveau l'intérêt des méthodes multi-fidélité dans le cadre d'optimisation de structures mécaniques d'assemblage.

Les trois méthodes multi-fidélité offrent, pour des coûts de calcul allant de 800 à 1300 secondes, une qualité de réponse tout à fait comparable. Le cokrigeage a l'intérêt supplémentaire de fournir une qualité de réponse meilleure que les deux autres méthodes pour des coûts de calcul plus faibles.

L'Evofusion originelle (Evof.) montre son incapacité à localiser rapidement la zone du minimum global. La raison de cet échec est que les données partiellement convergées sont faiblement corrélées par rapport au modèle haute fidélité (voir figure 4.4). Dans [Forrester, 2004, Forrester *et al.*, 2006a], les résultats obtenus par l'Evofusion sont de meilleure qualité car les données partiellement convergées le sont également.

### 3.2.4 Bilan de l'étude d'optimisation à l'aide de métamodèles multi-fidélité

On a pu constater sur le cas test *2\_rainures* que les méthodes d'Evofusion enrichie, de cokrigeage et de krigeage hiérarchique pouvaient selon le type de tirage utilisé et le critère d'arrêt utilisé être performantes pour localiser le minimum global de la fonction objectif. La méthode de cokrigeage, avec l'utilisation du tirage T4 et pour les deux critères d'arrêt  $\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$  et  $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$ , est la seule méthode multi-fidélité qui a permis de localiser le minimum global pour chacun des essais numériques menés. Le cokrigeage s'est alors également avéré être moins coûteux que la méthode de référence qui n'utilise que des données totalement convergées. De plus, en fonction du temps de calcul la précision moyenne sur le minimum obtenu par le métamodèle de cokrigeage est de meilleure qualité qu'avec le métamodèle de la méthode de référence (voir figure 4.9).

**A partir des résultats obtenus au chapitre 3 et de cette étude on peut en déduire que la stratégie de cokrigeage semble alors être la méthode multi-fidélité la plus adaptée dans notre cadre d'étude d'assemblage mécanique.** Dans la prochaine section, afin de compléter les résultats présentés au paragraphe 3.2.3, on expose un comparatif plus détaillé entre la méthode de référence et la méthode de cokrigeage utilisé avec le tirage T4.

### 3.2.5 Comparatif des résultats entre la méthode de référence et la méthode de cokrigeage

Le tableau 4.11 fournit un comparatif en terme de coût de calcul et de précision des méthodes de référence et de cokrigeage avec un tirage de type T4.

Les résultats présentés au tableau 4.11 montrent clairement l'intérêt de la méthode de cokrigeage par rapport à la méthode de référence. En effet, pour des résultats ayant une précision tout à fait semblable les temps de calcul sont inférieurs.

On fournit alors le gain obtenu entre la stratégie de référence sans approche multiparamétrique et utilisant le critère d'arrêt  $\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$  par rapport à la stratégie de cokrigeage avec approche multiparamétrique et utilisant le critère d'arrêt  $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$ . Ce gain est noté *Gain\_EGO* et est défini à l'équation 4.14.

	Méthode de référence		Méthode de cokrigage	
	$\tilde{\Delta}_m^{n_{dim}+1}$ 0.01	$\tilde{\Delta}_h^{n_{dim}+1}$ 0.001	$\tilde{\Delta}_m^{n_{dim}+1}$ 0.01	$\tilde{\Delta}_h^{n_{dim}+1}$ 0.001
Nbre de points tot. conv.	26.2	36.6	17.6	27.8
Tps calcul CPU (s) avec approche multiparamétrique	910	1220	730	1070
Tps calcul CPU (s) sans approche multiparamétrique	2385	3330	1660	2600
Distance au minimum	11.5%	7.6%	12.4%	6.9%
Valeur moyenne du minimum trouvé	1.5086	1.5069	1.510	1.5068

**TABLE 4.11:** Comparatif des performances atteintes pour les deux méthodes lors du premier niveau d'optimisation

$$\begin{aligned}
 \text{Tps de calcul pour atteindre } \tilde{\Delta}_h^{n_{dim}+1} \text{ avec la stratégie de} \\
 \text{référence sans approche multiparamétrique} \\
 \text{Tps de calcul pour atteindre } \tilde{\Delta}_m^{n_{dim}+1} \text{ avec le cokrigage} \\
 \text{avec approche multiparamétrique} \\
 \text{Gain}_{EGO} = \frac{\text{Tps de calcul pour atteindre } \tilde{\Delta}_h^{n_{dim}+1} \text{ avec la stratégie de} \\
 \text{référence sans approche multiparamétrique}}{\text{Tps de calcul pour atteindre } \tilde{\Delta}_m^{n_{dim}+1} \text{ avec le cokrigage} \\
 \text{avec approche multiparamétrique}} \\
 = \underbrace{\text{Gain}_{EGO}_{\text{cokrigage}}}_{\text{Gain de calcul} \\ \text{obtenu par} \\ \text{l'utilisation du} \\ \text{cokrigage}} \times \underbrace{\text{Gain}_{EGO}_{\text{crit.arret}}}_{\text{Gain de calcul} \\ \text{obtenu par} \\ \text{l'utilisation du} \\ \text{critère } \tilde{\Delta}_m^{n_{dim}+1}} \times \underbrace{\text{Gain}_{EGO}_{\text{multiparamétrique}}}_{\text{Gain de calcul} \\ \text{obtenu avec} \\ \text{l'approche} \\ \text{multiparamétrique}} \quad (4.14)
 \end{aligned}$$

avec :

$$\begin{aligned}
 \text{Tps de calcul pour atteindre} \\
 \tilde{\Delta}_h^{n_{dim}+1} \text{ avec la stratégie de} \\
 \text{référence sans approche} \\
 \text{multiparamétrique} \\
 \text{Tps de calcul pour atteindre} \\
 \tilde{\Delta}_h^{n_{dim}+1} \text{ avec la stratégie de} \\
 \text{cokrigage sans approche} \\
 \text{multiparamétrique} \\
 \text{Gain}_{EGO}_{\text{cokrigage}} = \frac{\text{Tps de calcul pour atteindre} \\
 \tilde{\Delta}_h^{n_{dim}+1} \text{ avec la stratégie de} \\
 \text{référence sans approche} \\
 \text{multiparamétrique}}{\text{Tps de calcul pour atteindre} \\
 \tilde{\Delta}_h^{n_{dim}+1} \text{ avec la stratégie de} \\
 \text{cokrigage sans approche} \\
 \text{multiparamétrique}} = \frac{3330}{2600} \approx 1.28 \quad (4.15)
 \end{aligned}$$

$$Gain\_EGO_{crit\_arret} = \frac{\text{Tps de calcul pour atteindre } \tilde{\Delta}_{h\ 0.001}^{n_{dim}+1} \text{ avec la stratégie de cokrigeage sans approche multiparamétrique}}{\text{Tps de calcul pour atteindre } \tilde{\Delta}_{m\ 0.01}^{n_{dim}+1} \text{ avec la stratégie de cokrigeage sans approche multiparamétrique}} = \frac{2600}{1660} \approx 1.57 \quad (4.16)$$

$$Gain\_EGO_{multiparametrique} = \frac{\text{Tps de calcul pour atteindre } \tilde{\Delta}_{m\ 0.01}^{n_{dim}+1} \text{ avec la stratégie de cokrigeage sans approche multiparamétrique}}{\text{Tps de calcul pour atteindre } \tilde{\Delta}_{m\ 0.01}^{n_{dim}+1} \text{ avec la stratégie de cokrigeage avec approche multiparamétrique}} = \frac{1660}{731} \approx 2.27 \quad (4.17)$$

De ce fait, on en déduit :

$$Gain\_EGO \approx 4.56 \quad (4.18)$$

La décomposition de ce gain en trois gains distincts permet d'exposer l'apport de chacun des développements et des techniques exposés au cours de ce travail. On constate que chacun des gains est supérieur à 1 et qu'ainsi chacun des développements amenés au cours de cette thèse est bénéfique.

Afin de compléter cette étude, il est également intéressant de donner le gain de calcul en effectuant :

$$Gain\_EGO\_bis = \frac{\text{Tps de calcul pour atteindre } \tilde{\Delta}_{h\ 0.001}^{n_{dim}+1} \text{ avec la stratégie de référence avec approche multiparamétrique}}{\text{Tps de calcul pour atteindre } \tilde{\Delta}_{m\ 0.01}^{n_{dim}+1} \text{ avec la stratégie de cokrigeage avec approche multiparamétrique}} = \frac{1221}{731} \approx 1.67 \quad (4.19)$$

**Ce gain représente le gain de temps calcul apporté par l'ensemble des développements au cours de ces travaux par rapport à ce qui a été effectué précédemment au sein du laboratoire.** La valeur de ce gain est relativement intéressant puisqu'il assure une économie de temps de calcul d'environ 40% lors du premier niveau d'optimisation.

Les différents développements apportés au cours de ce travail ont permis d'apporter un gain de calcul intéressant ( $Gain_{EGO} = 4.56$ ) pour le premier niveau d'optimisation. Concernant le second niveau d'optimisation, la section suivante présente la mise en place d'une stratégie d'optimisation locale efficace s'appuyant sur le principe des régions de confiance.

## 4 Optimisation d'un cas test mécanique : second niveau d'optimisation

Dans cette section on s'intéresse à mener l'optimisation locale sur le modèle mécanique réel. Cette étape correspond au second niveau de la stratégie multi-niveaux mise en place. Dans cette section, on cherchera à mettre en avant une stratégie d'optimisation locale efficace, basée sur les méthodes à région de confiance. Celle-ci sera ensuite testée sur des cas test analytiques et amènera à montrer ses performances. Finalement, cette stratégie d'optimisation locale sera effectuée sur le cas test mécanique *2\_rainures*.

### 4.1 Stratégie d'optimisation locale basée sur les méthodes à région de confiance

Une méthode d'optimisation semble tout à fait prometteuse, il s'agit d'une méthode se basant sur les régions de confiance [Alexandrov *et al.*, 2000, Gano et Renaud, 2004, Gano *et al.*, 2006]. Comme vu dans le chapitre 1, cette méthode s'appuie sur une fonction modèle simple à calculer et qui est optimisée à l'intérieur d'une zone (région de confiance) où cette fonction modèle est censée être représentative de la fonction réelle. Traditionnellement, cette fonction est une approximation quadratique de la fonction mécanique associée au modèle réel. Cependant, il est tout à fait possible de généraliser cette approche [Alexandrov *et al.*, 1998]. Il existe une condition de convergence de l'algorithme, celle-ci impose qu'au point courant le gradient de la fonction modèle soit égal au gradient de la fonction réelle (voir paragraphe 2.1.3 du chapitre 1).

#### 4.1.1 Méthodes à région de confiance avec fonctions modèles générales

[Alexandrov *et al.*, 1998] propose d'utiliser un algorithme à région de confiance avec des fonctions modèles générales. Ces fonctions ont pour propriétés principales :

$$\begin{aligned} y_{m_k}(\mathbf{x}_k) &= y(\mathbf{x}_k) \\ \nabla y_{m_k}(\mathbf{x}_k) &= \nabla y(\mathbf{x}_k) \end{aligned} \quad (4.20)$$

Ces deux propriétés sont nécessaires pour assurer la convergence de l'algorithme ([Keane et Nair, 2005]). Ces fonctions modèles doivent donc vérifier ces conditions au premier ordre de la fonction objectif  $y$ . En effet, les équations 4.20 garantissent que suffisamment proche de  $\mathbf{x}_k$ , la fonction modèle  $y_m$  est une bonne approximation de la fonction objectif  $y$ . L'algorithme à région de confiance décrit au chapitre 1 est donc appliquée à une forme généralisée.

[Alexandrov *et al.*, 1998] propose différents choix possibles de fonctions modèles autres que quadratiques.

Cette généralisation des fonctions modèles peut s'appliquer au métamodèle. Le métamodèle doit vérifier les conditions de l'équation 4.20.

On se propose sur la figure 4.10 d'illustrer l'algorithme à région de confiance appliqué à une fonction modèle qui est un métamodèle ( $\hat{y}$ ) de la fonction objectif ( $y$ ) à optimiser. Le point initial où commence l'algorithme est  $\mathbf{x}_0 = 11$ , on impose également que le gradient du métamodèle en ce point soit celui de la fonction objectif. Celui-ci est imposé par cokrigage de gradient indirect [Chung et Alonso, 2002, Liu, 2003, Laurenceau et Sagaut, 2008].

Les points utilisés pour définir le gradient en un point  $\mathbf{x}^{(j)}$  sont définis par :

$$\forall i \in \llbracket 1, n_{dim} \rrbracket, \mathbf{x}^{(j)grad.i} = \mathbf{x}^{(j)} + \Delta x_i \mathbf{e}_i \quad (4.21)$$

où  $\mathbf{e}_i$  est un vecteur de la base orthonormale de l'espace des paramètres avec  $|\Delta x_i| < 1$ . Afin de déterminer quelle valeur du paramètre  $\Delta x_i$  utiliser, on s'appuie sur les travaux de [Laurent, 2013]. En effet, dans le contexte des assemblages mécaniques, lorsque le gradient est déterminé par différence finie, [Laurent, 2013] a montré que le pas de différence finie était judicieusement choisi pour  $\delta = 0.05$  dans la relation définie à l'équation 4.22 :

$$\Delta x_i = \delta \cdot (x_i^{\max} - x_i^{\min}) \quad (4.22)$$

Pour rappel,  $x_i^{\max}$  et  $x_i^{\min}$  sont les bornes supérieures et inférieures de l'espace de conception (voir équation 1.2).

**Remarque :** Ce choix d'utiliser une approche indirecte pour incorporer les gradients au métamodèle vient du fait que l'incorporation du gradient de manière directe pour les approches multi-fidélité est une perspective à ce travail et sera abordée au chapitre 5.

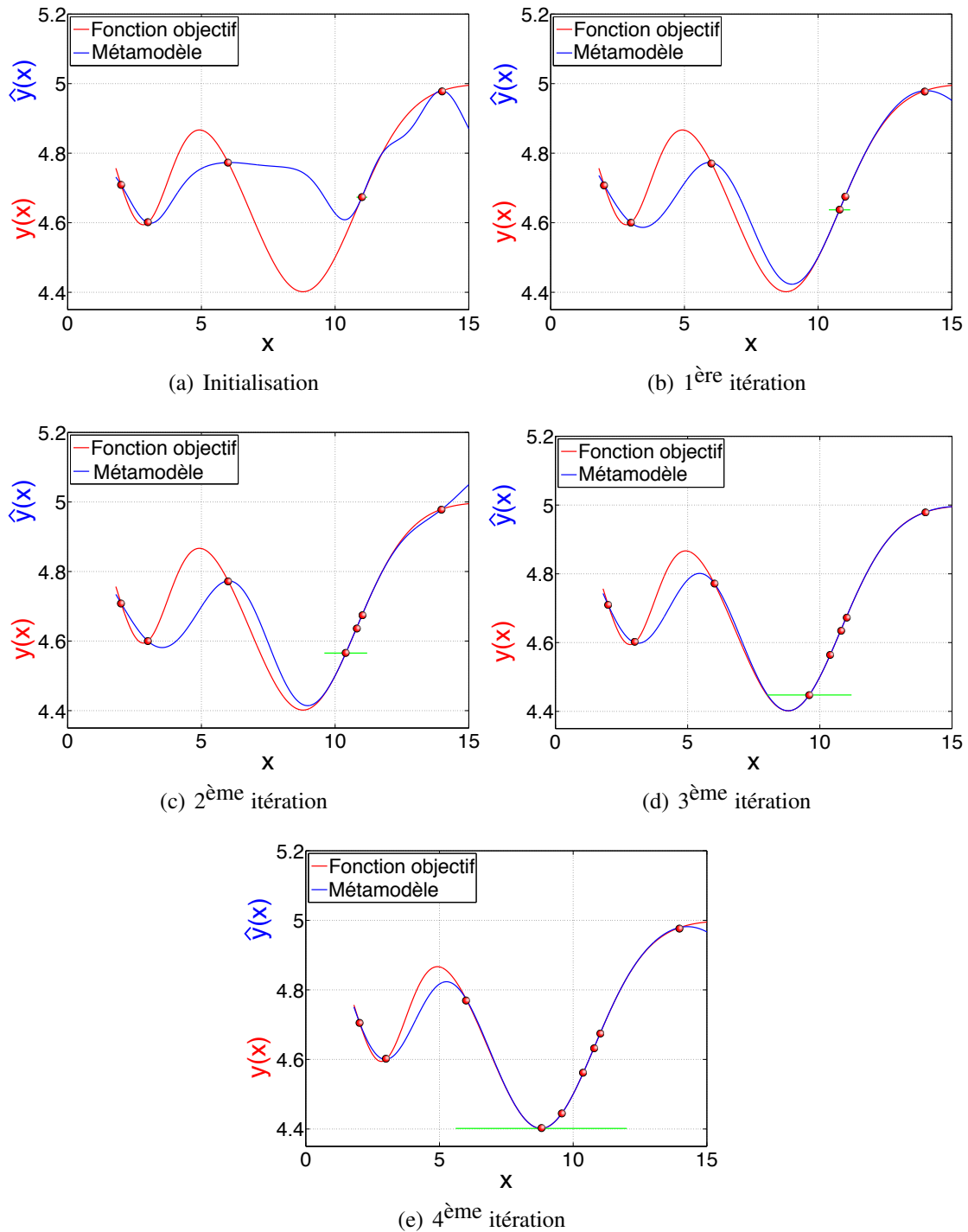
#### 4.1.2 Application sur des fonctions tests analytiques

Dans ce paragraphe, on teste l'algorithme à région de confiance avec une fonction modèle provenant d'un métamodèle. Pour réaliser ces tests, tout comme dans [Keane et Nair, 2005] ou [Gano *et al.*, 2006] l'algorithme à région de confiance est stoppé lorsque l'un des deux critères suivants est satisfait après que le pas ait été accepté ( $\rho > 0$  voir algorithme 1) :

$$y(\mathbf{x}_{k+1}) - y(\mathbf{x}_k) \leq -1.10^{-4} \quad (4.23)$$

$$\text{Dans un espace normé : } \|\mathbf{x}_{k+1} - \mathbf{x}_k\| \leq 1.10^{-4} \quad (4.24)$$

Cet algorithme est testé sur les fonctions d'Hartmann6 et Goldstein. Le métamodèle utilisé pour mener l'optimisation est celui obtenu lorsque le critère d'arrêt  $\tilde{\Delta}_{h, 0.001}^{n_{dim}+1}$  est atteint lors du premier niveau d'optimisation (voir paragraphe 1.3.2). De plus, comme proposé dans [Keane et Nair, 2005] une taille de zone de région de confiance minimale  $d_{min}$  a été instaurée. La taille de cette zone est  $d_{min} = 1.10^{-4}$ . Ceci est mis en place pour limiter les enrichissements de points pouvant amener un mauvais conditionnement de la matrice de covariance du métamodèle.



**FIGURE 4.10:** Algorithme à région de confiance

On constate au tableau 4.12 que la méthode à région de confiance est plus performante que la méthode SQP (environ 20% pour la fonction d'Hartmann6). Cette méthode est donc mise en



	Goldstein		Hartmann6	
	SQP	Région de confiance	SQP	Région de confiance
Nombre d'évaluations pour atteindre le critère d'arrêt	25.1	23.1	70	55

**TABLE 4.12:** Comparaison des coûts de calcul de chacune des méthodes

place pour venir effectuer l'optimisation locale pour le second niveau d'optimisation du problème *2\_rainures*.

## 4.2 Mise en place du second niveau d'optimisation sur le cas test

### 4.2.1 Introduction

On s'intéresse dans cette section à utiliser l'algorithme d'optimisation à région de confiance s'appuyant sur un métamodèle sur le cas test *2\_rainures*.

L'optimisation locale est effectuée à l'aide des métamodèles obtenus par la méthode de référence et de cokrigage (utilisant le tirage de type T4) après avoir atteint les critères d'arrêt  $\tilde{\Delta}_h^{n_{dim}+1}$  et  $\tilde{\Delta}_m^{n_{dim}+1}$  au premier niveau. On compare ainsi ces deux approches.

**Remarque :** Concernant la méthode de cokrigage, lors de cette optimisation locale, les évaluations de la fonction objectif ne sont plus effectuées que de manière totalement convergée, ceci afin d'économiser du temps de calcul.

Cette implantation se heurte à une difficulté numérique qu'est le bruit. En effet, même arrivé au niveau de convergence dit totalement convergé, l'utilisation du simulateur numérique CO-FAST engendre un bruit. Comme précisé dans [Laurent, 2013], ce bruit provient du fait que le solveur utilisé s'appuie sur un algorithme itératif. Ainsi, le calcul de plusieurs jeux de paramètres proches dans l'espace de conception peut entraîner l'obtention de réponses ne correspondant pas exactement au même niveau d'indicateur d'erreur  $\eta$ . Le bruit numérique obtenu peut alors engendrer un impact non négligeable sur la valeur du gradient par exemple.

D'un point de vue technique, ce bruit numérique, non présent pour les fonctions analytiques, a demandé un aménagement des critères d'arrêt de l'algorithme testé.

### 4.2.2 Critères d'arrêt de l'algorithme d'optimisation locale

De manière pratique, il s'est avéré nécessaire de modifier des paramètres de l'algorithme à région de confiance s'appuyant sur l'utilisation d'un métamodèle. Il a été choisi de modifier la taille de la zone de région minimale passant de  $d_{min} = 1.10^{-4}$  à  $d_{min} = 5.10^{-4}$ . Ceci a pour effet

de limiter le bruit de calcul de points trop proches dans l'espace de conception, lorsque la taille de la zone de région de confiance est minimale.

Les critères d'arrêt de l'algorithme sont alors :

- Arrêt si le pas a été accepté ( $\rho > 0$ ) et que  $y(\mathbf{x}_{k+1}) - y(\mathbf{x}_k) \leq -1.10^{-4}$ . Ce critère d'arrêt permet de stopper l'algorithme si le gradient est devenu suffisamment petit.
- Arrêt si :  $\|\mathbf{x}_{k+1} - \mathbf{x}_k\| \leq 2.10^{-4}$  **même si le pas a été rejeté**. On a constaté effectivement que même si le pas a été rejeté, à cause des erreurs numériques, il est préférable de stopper l'algorithme. Ceci pour éviter un mauvais conditionnement de la matrice de corrélation et potentiellement construire un métamodèle qui ne soit plus représentatif de la zone d'optimisation. Dans le cadre du krigeage, on a pu tester et observer numériquement que des points distants de moins de  $1.10^{-4}$  amènent à un mauvais conditionnement de matrice de corrélation. Dans le cadre du cokrigeage multi-fidélité, le minimum de distance entre deux points afin d'éviter des problèmes de conditionnement est plutôt de l'ordre de  $2.10^{-4}$ , ceci expliquant alors le choix du critère d'arrêt qui a été fait.

Ces deux critères d'arrêt ont été également testés sur les fonctions analytiques précédentes, les résultats trouvés sont tout à fait semblables.

#### 4.2.3 Résultats de l'optimisation locale sur le cas test 2 rainures

	Méthode de référence		Méthode de cokrigeage	
	$\tilde{\Delta}_m^{n_{dim}+1}$ 0.01	$\tilde{\Delta}_h^{n_{dim}+1}$ 0.001	$\tilde{\Delta}_m^{n_{dim}+1}$ 0.01	$\tilde{\Delta}_h^{n_{dim}+1}$ 0.001
Nbre de points tot. conv.	16.7	12.9	19.3	12.3
Tps calcul CPU (s) avec approche multiparamétrique	330	200	365	190
Tps calcul CPU (s) sans approche multiparamétrique	1535	1190	1775	1130
Distance au minimum	3.6%	3.9%	3.4%	3%
Valeur moyenne du minimum trouvé	1.5059	1.506	1.5059	1.5057

**TABLE 4.13:** Comparatif des performances atteintes pour les deux méthodes lors du second niveau d'optimisation

La première remarque que l'on peut faire est que l'apport de la stratégie multiparamétrique lors de cette phase d'étude est très intéressant puisqu'elle permet de diviser le temps de calcul par 5 voire 6. On constate de plus que quel que soit le critère précédemment utilisé pour la stratégie EGO, la qualité du résultat obtenu est tout à fait semblable. On remarque même que le cokrigeage amène à une localisation légèrement meilleure que la méthode de référence utilisant un métamodèle de krigeage.

### 4.3 Bilan de la stratégie d'optimisation multi-niveaux sur le cas test *2\_rainures*

Le tableau 4.14 présente un récapitulatif des coûts de calcul associés à chacune des méthodes.

	Méthode de référence		Méthode de cokrigage	
	$\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$	$\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$	$\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$	$\tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}$
Nbre de points tot. conv.	42.9	49.5	36.9	40.1
Tps calcul CPU (s) avec approche multiparamétrique	1240	1420	1095	1260
Proportion temps de calcul (1 <sup>er</sup> niveau - 2 <sup>ème</sup> niveau)	73%-27%	86%-14%	67%-33%	85%-15%
Tps calcul CPU (s) sans approche multiparamétrique	3915	4520	3435	3730
Proportion temps de calcul (1 <sup>er</sup> niveau - 2 <sup>ème</sup> niveau)	61%-39%	74%-26%	48%-52%	70%-30%

**TABLE 4.14:** Comparatif des performances atteintes pour les deux méthodes testées pour l'optimisation multi-niveaux complètement réalisée

On constate que la répartition des temps de calcul est relativement différente en fonction de l'utilisation ou non de l'approche de calcul multiparamétrique. Son utilisation permet de limiter le coût de calcul lors du second niveau d'optimisation. Le fait de ne pas l'utiliser peut engendrer un coût de calcul important sur l'ensemble de la méthode et le second niveau d'optimisation peut même être le plus coûteux (voir la méthode de cokrigage avec utilisation du critère d'arrêt  $\tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}$ ).

En s'appuyant sur les résultats du tableau 4.14, on définit alors le gains de temps de calcul :

$$\begin{aligned}
 \text{Gain}_{optim} &= \frac{\text{Tps de calcul avec la stratégie de référence sans approche multiparamétrique en utilisant le critère } \tilde{\Delta}_{h\ 0.001}^{n_{dim}+1}}{\text{Tps de calcul avec le cokrigage avec approche multiparamétrique en utilisant le critère } \tilde{\Delta}_{m\ 0.01}^{n_{dim}+1}} \\
 &= \text{Gain}_{optim_{cokrigage}} \times \text{Gain}_{optim_{crit\_arret}} \times \text{Gain}_{optim_{multiparametrique}} \\
 &\approx 1.21 \times 1.09 \times 3.14 \approx 4.14
 \end{aligned} \tag{4.25}$$

On peut constater que l'ensemble des développements apportés au cours de ce travail (cokrigage multi-fidélité et critère d'arrêt) et des travaux précédents (stratégie multiparamétrique) amène tous des gains supérieurs à 1. Le résultat des développements apportés est donc tout à

fait positif que ce soit pour le premier niveau d'optimisation ou pour la stratégie multi-niveaux complète. Le gain total de la méthode sur cet exemple mécanique 2D à deux paramètres est légèrement supérieur à 4. **Ainsi, le couplage de calcul multiparamétrique et de méthodes multi-fidélité est rendu tout à fait viable pour traiter des cas mécaniques d'assemblage.**

A titre informatif, on fournit le gain obtenu lorsque les calculs de la méthode de référence sont effectués avec l'approche multiparamétrique. Ce gain mettra plus spécifiquement en avant l'apport des développements réalisés au cours de cette thèse (cokrigeage multi-fidélité et critère d'arrêt  $\tilde{\Delta}_{m,0.01}^{n_{dim}+1}$ ).

$$Gain_{optim\_bis} = \frac{\text{Tps de calcul avec la stratégie de référence avec approche multiparamétrique en utilisant le critère } \tilde{\Delta}_{h,0.001}^{n_{dim}+1}}{\text{Tps de calcul avec le cokrigeage avec approche multiparamétrique en utilisant le critère } \tilde{\Delta}_{m,0.01}^{n_{dim}+1}} \quad (4.26)$$

$$\approx 1.3$$

Ainsi les développements réalisés permettent une économie de temps de calcul d'environ 23% par rapport à la stratégie de référence se basant sur le krigeage et le calcul multiparamétrique qui ont été notamment utilisés dans les travaux de [Laurent, 2013].

## 5 Bilan

Dans la première partie, différents critères d'arrêt de la stratégie EGO ont été testés. Le critère  $\tilde{\Delta}_{m,0.01}^{n_{dim}+1}$  a offert le rapport qualité-coût le plus intéressant. Cependant, la qualité des résultats obtenus étant inférieure à ce qui pouvait être réalisé avec le critère d'arrêt de référence  $\tilde{\Delta}_{h,0.001}^{n_{dim}+1}$ , il a été proposé d'utiliser un tirage initial différent du tirage LHS. La mise en place d'un tirage LHS adaptatif a permis d'améliorer la qualité des métamodèles construits.

Différentes méthodes multi-fidélité (EvoFusion, cokrigeage et krigeage hiérarchique), avec différentes méthodes de tirages (tirage LHS ou tirage LHS adaptatif des données partiellement convergées) et pour différents critères d'arrêt de la stratégie EGO ( $\tilde{\Delta}_{m,0.01}^{n_{dim}+1}$  ou  $\tilde{\Delta}_{h,0.001}^{n_{dim}+1}$ ) ont alors été mises en place. Au vu des résultats obtenus, la méthode de cokrigeage s'est avérée être la méthode la plus efficace pour déterminer le minimum de la fonction objectif associée au cas test mécanique. L'utilisation du tirage LHS adaptatif s'est également avéré être performant. On notera de plus que lors de la phase d'enrichissement de la stratégie EGO (enrichissement au point maximisant l'EI), il est conseillé de calculer le point enrichi de manière partiellement et totalement convergé (comme pour les tirages de type T1 ou T4).

Par rapport à une méthode de référence, s'appuyant sur un métamodèle de krigeage un gain important peut être atteint grâce au couplage de la méthode de calcul multiparamétrique et du cokrigeage multi-fidélité (environ 4.6 pour le premier niveau d'optimisation et 4.1 sur l'ensemble de la stratégie d'optimisation). De plus, même si pour la méthode de référence le calcul des données se fait à l'aide de l'approche multiparamétrique, une économie de gain de temps de 40%

lors du premier niveau d'optimisation a été atteint. Pour l'ensemble de la méthode, l'économie de temps est de 23%.

Lors du second niveau d'optimisation, une optimisation locale basée sur la méthode à région de confiance utilisant comme fonction modèle le métamodèle précédemment obtenu s'est avérée efficace et a été implémentée avec succès.

Ainsi, ce cas test illustre l'intérêt d'utiliser les approches multi-fidélité pour mener l'optimisation sur un cas mécanique. La méthode la plus adaptée sur cet exemple est le cokrigeage. Dans le prochain chapitre on s'attachera à mener une stratégie de recalage (création d'abaque virtuel) et de recherche d'optimum (optimisation avec approche EGO) sur des cas test industriels.

# Chapitre 5

## Applications sur des cas test industriels et perspectives de développement

*Dans ce dernier chapitre, une mise en œuvre du cokrigeage multi-fidélité sur des cas test industriels est effectuée. On propose sur un premier cas test d'effectuer une optimisation sous contrainte. Pour le second cas test, on propose de créer un métamodèle de qualité (abaque virtuel) sur un cas test à un million de degrés de liberté. Finalement, en perspective à l'ensemble des travaux précédemment présentés le développement des approches multi-fidélité avec prise en compte des gradients sera illustré sur deux cas test déjà utilisés.*

### Sommaire

---

<b>1</b>	<b>Optimisation sous contrainte d'un assemblage boulonné . . . . .</b>	<b>190</b>
1.1	Présentation du cas test . . . . .	190
1.2	Présentation de la problématique traitée . . . . .	192
1.3	Enrichissement avec prise en compte des contraintes . . . . .	195
1.4	Réalisation de l'optimisation du cas test . . . . .	196
<b>2</b>	<b>Création d'un abaque virtuel . . . . .</b>	<b>201</b>
2.1	Introduction . . . . .	201
2.2	Présentation du cas test . . . . .	201
2.3	Adaptations des outils disponibles pour le calcul . . . . .	204
2.4	Construction de l'abaque virtuel . . . . .	206
2.5	Bilan . . . . .	208
<b>3</b>	<b>Développements supplémentaires et possibilités offertes par la toolbox GRENAT</b>	<b>209</b>
3.1	Construction de métamodèle multi-fidélité avec prise en compte des gradients	209
3.2	Illustrations des performances atteintes sur deux cas test mécaniques . . . .	217

---

On s'attachera dans ce chapitre à utiliser le cokrigeage multi-fidélité, afin de réaliser l'optimisation d'un assemblage sur un premier cas test industriel et la création d'un abaque virtuel sur un second cas industriel. Ces exemples ont en commun d'être tous deux des assemblages boulonnés. Une attention particulière a été portée sur la contrainte que les vis pouvaient supporter sous chargement (dans nos cas d'étude, ceux-ci étaient constants). En effet, en lien avec le projet CARAB, s'intéressant essentiellement aux problématiques des assemblages boulonnés et ce, de manière relativement exhaustive (topologie, modélisation, expérimentation etc.), on a jugé intéressant de s'intéresser à cette problématique. La rupture des vis (due notamment aux contraintes de cisaillement engendrées par le contact entre la vis et son logement) est une des raisons des défaillances des assemblages boulonnés. Trouver les paramètres amenant à une utilisation optimale de l'assemblage sans le dégrader relève d'un réel souhait industriel.

Il s'agit donc de s'intéresser à une problématique complexe d'un point de vue numérique. En effet, elle demande la prise en compte de la précharge du boulon, du frottement et de l'éventuel contact entre la vis et son logement. Le solveur COFAST a montré comme déjà dit dans cette thèse, son habilité à résoudre ce genre de problème. Certaines précautions sont néanmoins à prendre en compte, dues notamment au fait que la quantité d'intérêt (contrainte dans les vis) est fortement localisée par rapport à la taille du problème (fort niveau de convergence du solveur itératif LaTin pour obtenir une réponse correcte).

L'ensemble des outils utilisés a été introduit dans les chapitres précédents où l'on a pu vérifier leurs capacités quant à la résolution de problèmes d'optimisation ou de construction efficaces de métamodèles. On cherchera donc à mettre en avant l'utilisation de ces outils sur des cas test 3D plus conséquents en terme de temps de calcul, de sous-structures et d'interfaces.

Ainsi, on se propose dans un premier temps d'effectuer une optimisation sous contrainte d'un assemblage mécanique via la méthode de cokrigeage multi-fidélité. Par la suite, sur un second cas test, on se propose de créer un abaque virtuel via cette méthode de cokrigeage. Pour les deux exemples, une approche comparative par utilisation de métamodèle de krigage est également mise en place afin de mettre en avant les gains de calculs pouvant être atteints.

## 1 Optimisation sous contrainte d'un assemblage boulonné

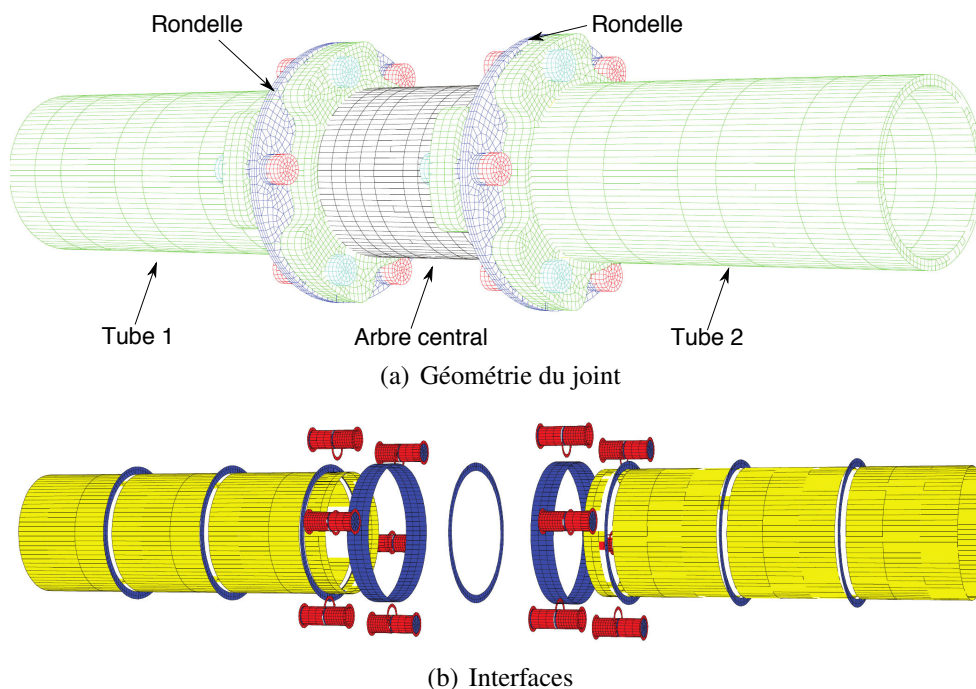
### 1.1 Présentation du cas test

On présente ici, le cas test industriel sur lequel une optimisation sera menée. Ce cas test développé au sein du laboratoire depuis plusieurs années est une adaptation libre d'un joint d'accouplement de type ROBA-DS de la marque mayr<sup>1</sup>. Ce joint d'accouplement à la particularité de supporter un désalignement entre les deux arbres accouplés.

L'assemblage comporte 50 sous-structures, 89 interfaces et 144588 degrés de liberté. L'ensemble des pièces est considéré comme étant constitué du même matériau en acier (Module d'Young = 210 GPa, et coefficient de Poisson = 0.3). Sur la figure 5.1(b), les différentes interfaces sont présentées. En bleu, les interfaces parfaites, en rouge les interfaces unilatérales et en jaune les interfaces où le déplacement est imposé. Ces dernières se situent à l'intérieur des tubes.

---

1. <http://www.mayr.com>



**FIGURE 5.1:** Joint d'accouplement (figures issus de [Laurent, 2013])

On impose un désalignement angulaire entre le tube 1 et 2 (mise en flexion du joint). Dans un second temps, on imposera une torsion via un déplacement angulaire sur l'ensemble du tube 1 (mise en torsion). Sous COFAST le chargement total de la structure s'effectue en 2 temps (2 steps de chargement) :

- phase de précharge : précharge des 12 vis réalisée en un pas de temps (statique)
- phase de chargement : mise en flexion du joint d'accouplement (simulation du désalignement entre les deux tubes) et mise en torsion de celui-ci. Cette flexion et cette torsion sont modélisées par un champ de déplacement piloté par des angles qui sont respectivement  $\theta_{fle}$  et  $\theta_{tor}$ . L'angle de torsion  $\theta_{tor}$  est imposé sur le tube 1. Ce chargement est réalisé en 5 pas de temps (quasi-statique).

On impose  $\theta_{fle} = 2^\circ$  et  $\theta_{tor} = 0.5^\circ$ . Techniquement, le rattrapage du désalignement des arbres se fait à l'aide de rondelles (pièces souples par rapport au reste de l'assemblage) assemblées par l'intermédiaire de  $2 \times 3$  vis avec les tubes 1 et 2 et  $2 \times 3$  vis avec la partie centrale du joint d'accouplement. Le frottement et le contact seront modélisés entre chaque pièce de l'assemblage. La rigidité en torsion de l'assemblage est réalisée par la précharge plus ou moins importante des boulons. Le contrôle de la précharge des boulons assure donc le contrôle du couple transmissible.



## 1.2 Présentation de la problématique traitée

### 1.2.1 Présentation du problème d'optimisation

On se propose sur ce cas test d'étudier deux paramètres d'optimisation. Ces deux paramètres sont :

- Précharge des vis entre le tube 1 et l'arbre central, notée  $P_{t1/a}$
- Précharge des vis entre le tube 2 et l'arbre central, notée  $P_{t2/a}$

La précharge est imposée par un jeu négatif entre la tête et le corps de vis. Le problème d'optimisation que l'on cherche à résoudre est un problème d'optimisation sous contrainte. On souhaite déterminer quels sont les paramètres de précharge permettant de transmettre le couple maximal sans dépasser une contrainte de Tresca de 1000 MPa dans la vis la plus contrainte. L'espace de conception  $\mathcal{D}$  est défini comme suit :

Paramètres	min	max
$P_{t1/a}$ (en mm)	-0.3	-0.01
$P_{t2/a}$ (en mm)	-0.3	-0.01

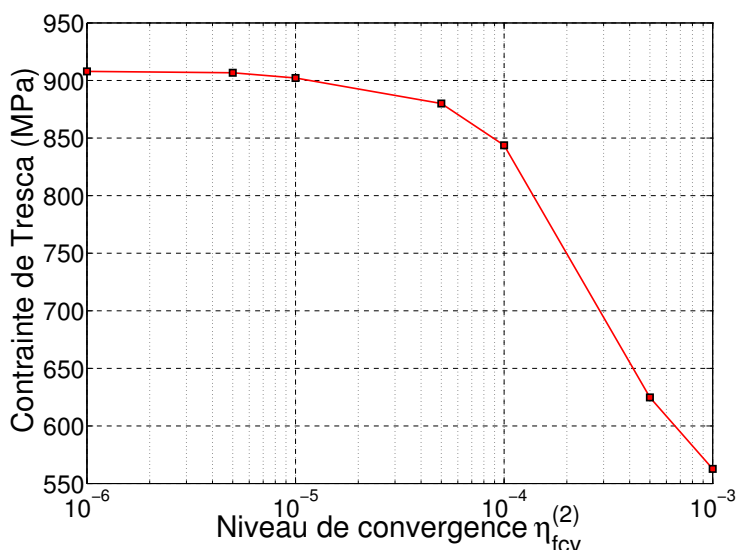
**TABLE 5.1:** Bornes de conception pour le cas test du joint d'accouplement

En posant  $M_t$  le couple transmissible et  $C_{Tr}^i$  la contrainte de Tresca dans la vis  $i$ , le problème d'optimisation se pose sous la forme suivante :

$$\mathbf{x}^* = \left\{ \begin{array}{l} \arg \max_{\mathbf{x} \in \mathcal{D}} M_t(\mathbf{x}) \\ \text{tel que :} \\ \max_{\forall i \in [1,12]} C_{Tr}^i(\mathbf{x}) \leq 1000 \text{MPa} \end{array} \right. \quad (5.1)$$

### 1.2.2 Précisions sur le calcul numérique

Une difficulté importante pour mener à bien la résolution de cas test est la précision nécessaire au calcul pour évaluer avec précision la contrainte de Tresca. Cette quantité d'intérêt relativement locale vis-à-vis de la taille du problème impose un indicateur d'erreur faible. Une difficulté supplémentaire provient du fait que le calcul complet s'effectue en deux steps. On notera respectivement  $\eta_{fcv}^{(1)}$ ,  $\eta_{fcv}^{(2)}$ ,  $\eta_{pcv}^{(1)}$  et  $\eta_{pcv}^{(2)}$ , les indicateurs d'erreur LaTin à atteindre lors du calcul de données totalement convergées lors du step 1 et 2 et lors du calcul de données partiellement convergées lors de ces mêmes step. Le calcul du premier step, représentant la précharge sur les boulons, est celui qui est le moins chronophage (les vis travaillent uniquement en traction et le calcul est réalisé en statique). De ce fait, afin d'assurer une bonne qualité de réponse, il a été choisi  $\eta_{fcv}^{(1)} = 1.10^{-4}$ . Pour le second step, la figure 5.2 met en avant la nécessité d'atteindre un faible indicateur d'erreur  $\eta_{fcv}^{(2)}$ .



**FIGURE 5.2:** Convergence du calcul de la contrainte maximale des vis en fonction du niveau d'indicateur  $\eta_{fcv}^{(2)}$

Au vu des résultats obtenus à la figure 5.2, le niveau de convergence  $\eta_{fcv}^{(2)}$  est fixé à  $1.10^{-5}$ . L'étape de précharge s'effectue en 120 secondes CPU pour un calcul totalement convergé. L'étape de chargement est calculée en 4580 secondes CPU. La phase de précharge est donc 37 fois moins coûteuse que la phase de chargement, il n'est donc pas nécessaire d'essayer d'économiser du temps de calcul lors de cette phase puisqu'elle ne consomme que 2.6% du temps de calcul total. Il a donc été choisi  $\eta_{fcv}^{(1)} = \eta_{pcv}^{(1)} = 1.10^{-4}$ .

Le niveau de convergence des données partiellement convergées pour le second step de calcul est fixé à  $\eta_{pcv}^{(2)} = 1.10^{-3}$ . Le coût de calcul CPU associé à ce niveau de convergence est de 80s. Le niveau  $\eta_{pcv}^{(2)} = 1.10^{-3}$  amène un gain de temps de calcul d'environ 60 par rapport à  $\eta_{fcv}^{(2)} = 1.10^{-5}$ . L'utilisation de ce niveau de convergence vient donc du fait que le coût de calcul d'un jeu de paramètres associé à ce niveau est faible. Une étude complète comme menée dans le chapitre 2 pourrait être à effectuer sur des cas test 3D. Cependant une telle étude serait très coûteuse, et de plus, on a vu au chapitre 3 que la méthode de cokrigeage s'est avérée stable en fonction du niveau de convergence des données partiellement convergées. On verra dans ce chapitre que travailler à ce niveau de convergence amène des résultats de bonne qualité sur les cas test 3D étudiés.

### 1.2.3 Surfaces de réponse associées au problème d'optimisation traité

On présente ici les surfaces de réponses obtenues en fonction des paramètres  $P_{t1/a}$  et  $P_{t2/a}$ . Les surfaces de réponses sont calculées à l'aide d'un plan factoriel complet de  $10 \times 10$  points. Les figures 5.3 et 5.4 représentent respectivement le couple transmissible  $M_t$  et la contrainte maximale dans les vis de l'assemblage en fonction des précharges  $P_{t1/a}$  et  $P_{t2/a}$ .

A partir de ces surfaces de réponses, on peut donc en déduire la zone d'admissibilité des paramètres  $P_{t1/a}$  et  $P_{t2/a}$  dans laquelle on cherchera à déterminer le maximum du couple trans-

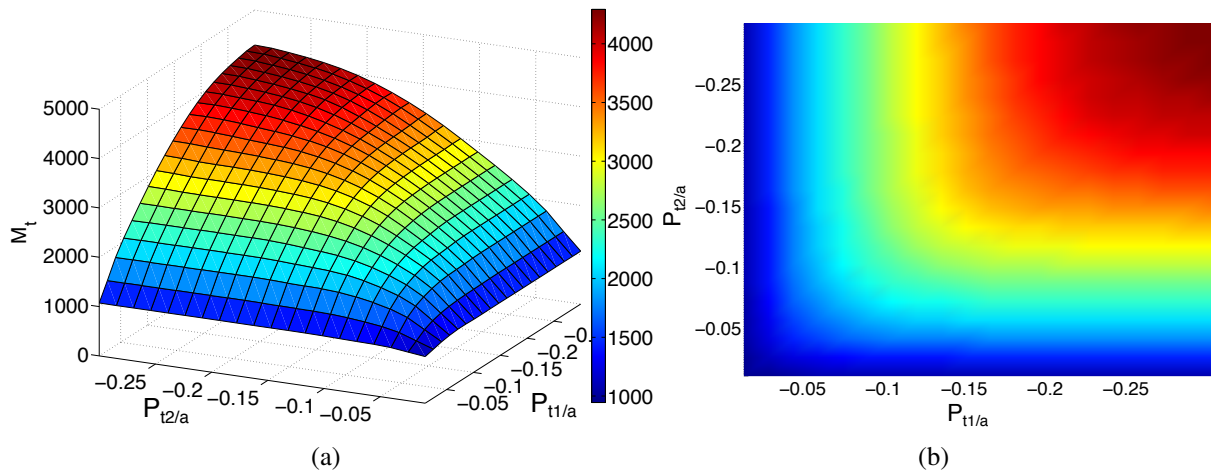


FIGURE 5.3: Surface de réponse représentant le couple transmissible  $M_t$  (en N.m)

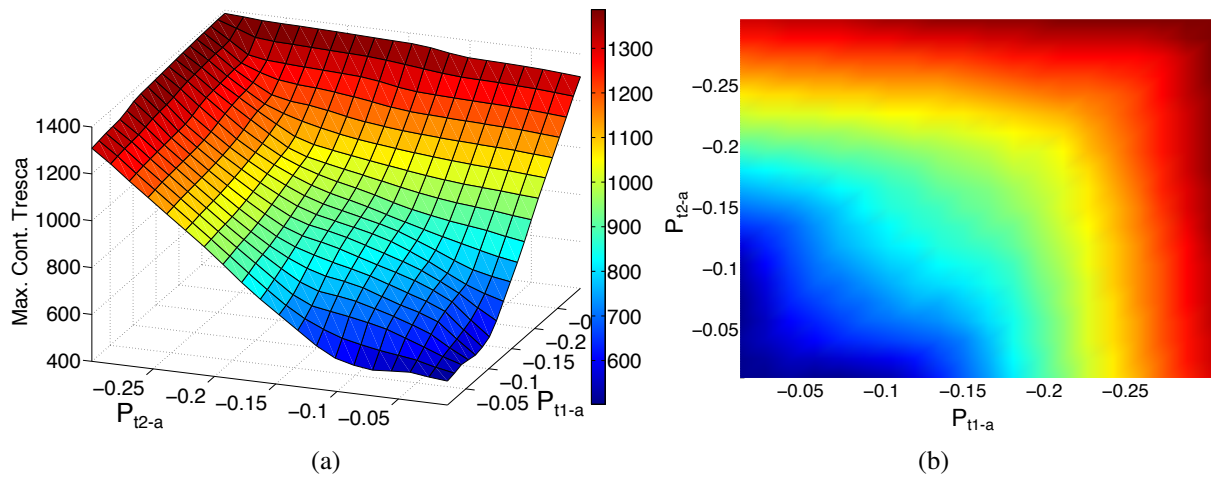


FIGURE 5.4: Surface de réponse représentant le maximum de la contrainte de Tresca dans les vis (en MPa)

missible. Cette zone est présentée en figure 5.5.

En se basant sur cette représentation, on constate que le couple maximal transmissible est d'environ 3400 MPa.

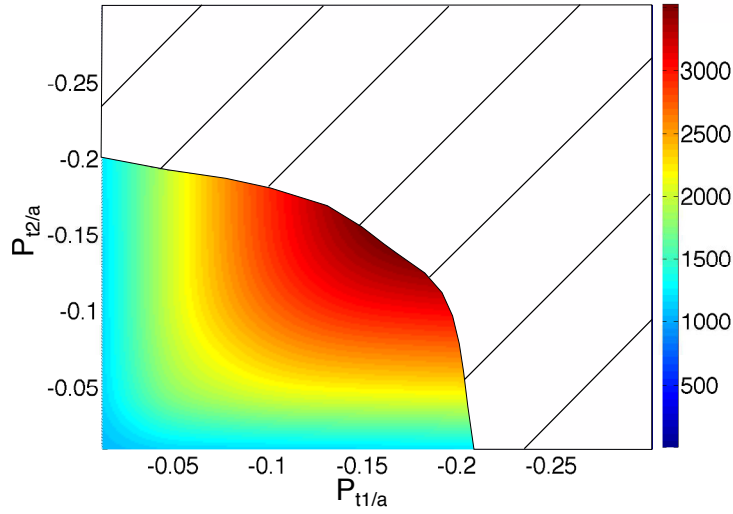


FIGURE 5.5: Couple transmissible dans la zone d'admissibilité

### 1.3 Enrichissement avec prise en compte des contraintes

Afin de mener une stratégie d'EGO avec prise en compte de fonctions contraintes, une modification du critère d'enrichissement est nécessaire. [Keane et Nair, 2005, Forrester *et al.*, 2008, Forrester et Keane, 2009, Parr *et al.*, 2012] montrent que l'utilisation du critère d'enrichissement *constrained expected improvement* (CEI) comme défini par [Forrester *et al.*, 2008] et proposé par [Schonlau, 1997] répond à cette problématique. Des méthodes de prise en compte des contraintes par pénalisation peuvent également être envisagées mais [Parr *et al.*, 2010] montre que la méthode EGO par enrichissement au maximum de CEI est la plus performante.

#### Définition du critère CEI

Soient  $y(\mathbf{x})$  la fonction objectif et  $g_i(\mathbf{x})$  une fonction contrainte. Les fonctions contraintes doivent vérifier :

$$g_i(\mathbf{x}) \leq 0, \quad \forall i \in \llbracket 1, n_g \rrbracket \quad (5.2)$$

Chacune des fonctions contraintes est évaluée à l'aide d'un métamodèle  $\hat{g}_i$  et de variance  $s_{\hat{g}_i}^2$ . On définit alors le processus gaussien  $G_i$  par :

$$G_i \sim \mathcal{N}(\hat{g}_i, s_{\hat{g}_i}^2) \quad (5.3)$$

De même on définit le processus gaussien  $Y_e(\mathbf{x}) \sim \mathcal{N}(\hat{y}, s_{\hat{y}}^2)$  (voir paragraphe 3.3 du chapitre 1).

L'amélioration est définie par ([Keane et Nair, 2005]) :

$$I_c = \begin{cases} y_{min\_admi} - Y_e(\mathbf{x}) & \text{si } Y_e(\mathbf{x}) < y_{min\_admi} \text{ et } G_i(\mathbf{x}) \leq 0 \text{ pour } i = 1 \dots n_g \\ 0 & \text{sinon} \end{cases} \quad (5.4)$$

$y_{min\_admi}$  est le minimum connu de la fonction objectif qui se situe dans la zone d'admissibilité des fonctions contraintes. Les modèles étant indépendants, on peut calculer la fonction CEI comme [Schonlau, 1997, Keane et Nair, 2005, Forrester et Keane, 2009] :

$$CEI(\mathbf{x}) = \left[ (y_{min\_admi} - \hat{y}(\mathbf{x})) \Phi \left( \frac{y_{min\_admi} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})} \right) + s_{\hat{y}}(\mathbf{x}) \phi \left( \frac{y_{min\_admi} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})} \right) \right] \cdot \prod_{i=1}^{n_g} P[G_i \leq 0] \quad (5.5)$$

avec  $P[G_i \leq 0]$  la probabilité de vérifier  $G_i \leq 0$ . Selon l'équation 1.64, on trouve alors :

$$P[G_i \leq 0] = \Phi \left( \frac{y_{min\_admi} - \hat{y}(\mathbf{x})}{s_{\hat{y}}(\mathbf{x})} \right) \quad (5.6)$$

En s'inspirant de l'exemple proposé par [Forrester et Keane, 2009], on illustre le CEI à la figure 5.6.

Les fonctions étudiées sont :

$$\begin{aligned} y(x) &= (6x - 2)^2 \times \sin(12x - 4) \\ g_1(x) &= -y(x) - 2 \leq 0 \end{aligned} \quad (5.7)$$

## 1.4 Réalisation de l'optimisation du cas test

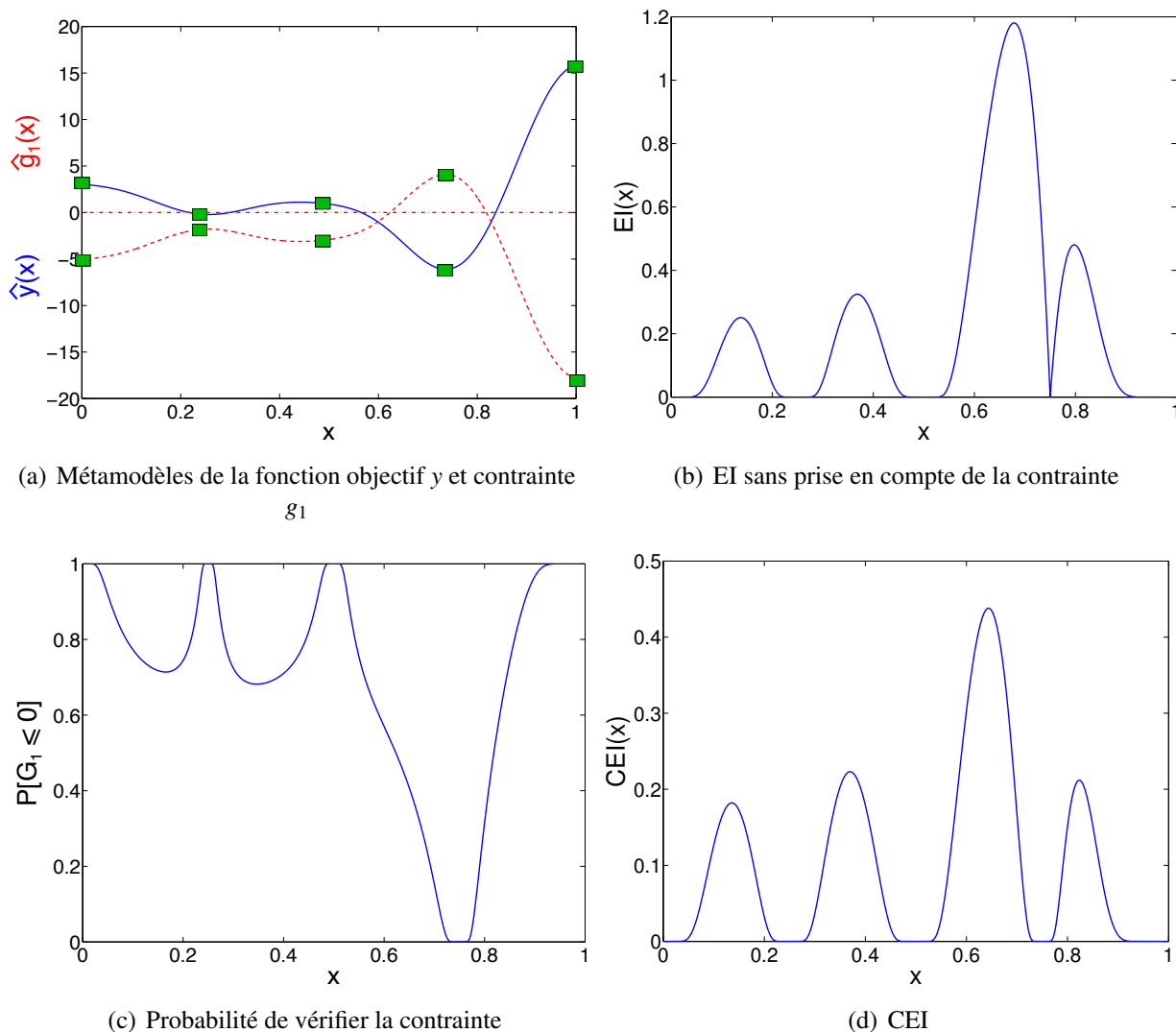
### 1.4.1 Descriptif des stratégies comparées

Deux méthodes de construction de métamodèles sont comparées pour mener l'optimisation de cas test.

- Approche de cokrigeage : Un métamodèle de cokrigeage est créé à l'aide d'un tirage de type T4, c'est-à-dire qu'un tirage LHS adaptatif de  $10 \times n_{dim} = 20$  points des données partiellement convergées est effectué et que le point maximisant le critère CEI est calculé de manière totalement et partiellement convergée. Le nombre initial de points totalement convergés utilisés pour construire le premier métamodèle de cokrigeage est de  $3 \times n_{dim} = 6$  points. On stoppe la stratégie lorsque 20 points totalement convergés (nombre de points initialement calculés pour la méthode de référence) ont été calculés.
- Approche de référence par krigeage :  $10 \times n_{dim} = 20$  points LHS initiaux sont calculés totalement convergés. L'enrichissement s'effectue ensuite au point maximisant le CEI. La stratégie est stoppée après un enrichissement de 30 points.

Les stratégies comparées sont donc tout à fait semblables à ce qui a pu être fait lors du précédent chapitre. La seule modification est le critère d'enrichissement qui est sur ce cas test le point maximisant le *constrained expected improvement* (CEI).

Le but de cette étude est de comparer la qualité des résultats obtenus en fonction du coût de calcul, comme cela a notamment pu être fait dans [Forrester, 2004]. Le critère d'arrêt est défini par un budget de calcul alloué à chacune des méthodes. Pour l'approche par cokrigeage, on stoppe l'algorithme lorsqu'au **total**  $10 \times n_{dim} = 20$  points totalement convergés ont été calculés.



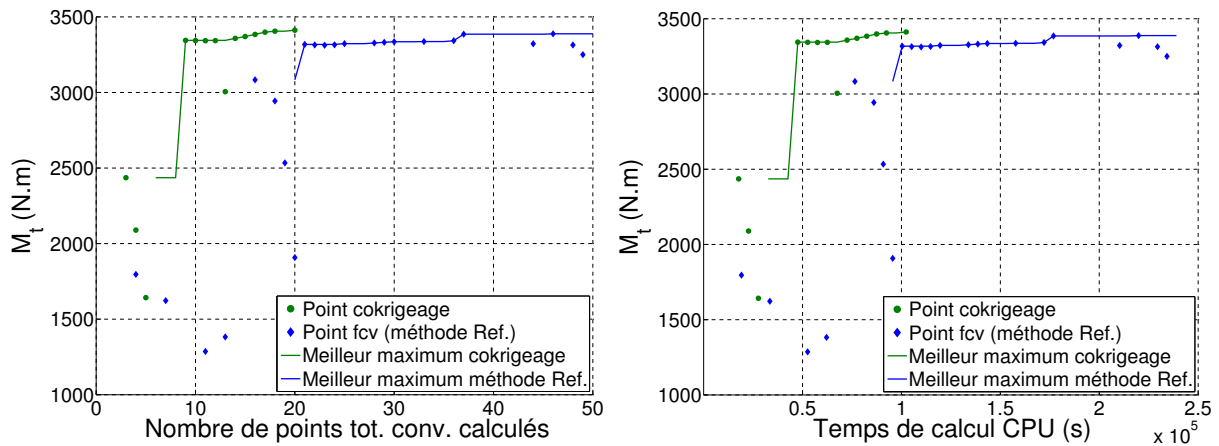
**FIGURE 5.6:** Illustration du constrained expected improvement

Pour l'approche de référence, on stoppe l'algorithme lorsque 50 points totalement convergés ont été calculés. Le but est de montrer que l'on peut rapidement à l'aide du cokrigage obtenir une solution appropriée au problème posé.

### 1.4.2 Résultats

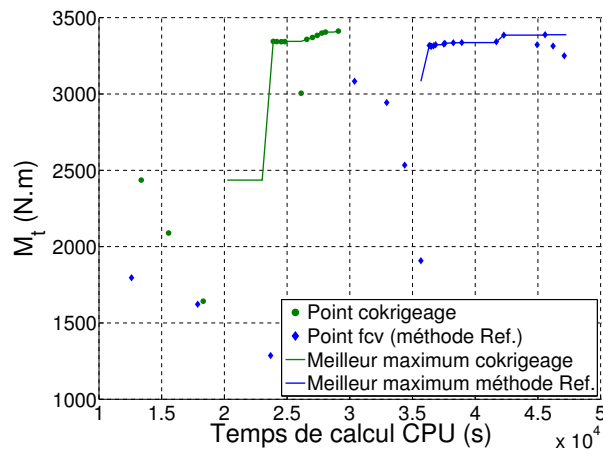
On présente en figure 5.7 les résultats de convergence obtenus à l'aide des deux méthodes, en fonction du temps de calcul CPU (avec ou sans emploi de la stratégie multiparamétrique) et du nombre de points totalement convergés calculés.

Sur les figures 5.7(a) et 5.7(b), on constate que pour un même effort de calcul, à savoir 100.000 secondes (c'est-à-dire pour 20 points calculés totalement convergés pour le cokrigage



(a) Couple maximum transmissible en respectant la contrainte maximale admissible dans les vis en fonction du nombre de points totalement convergés calculés

(b) Couple maximum transmissible en respectant la contrainte maximale admissible dans les vis en fonction du temps calcul CPU **sans** l'approche multiparamétrique



(c) Couple maximum transmissible en respectant la contrainte maximale admissible dans les vis en fonction du temps calcul CPU **avec** l'approche multiparamétrique

FIGURE 5.7: Historique de convergence des deux méthodes comparées

et 21 pour la méthode de référence) l'utilisation de la méthode de cokrigage permet de trouver un jeu de paramètres amenant un couple maximal transmissible de 3412 N.m contre 3318 N.m avec la méthode de référence. Cette approche par cokrigage fournit alors une amélioration d'environ 3%.

Pour le même nombre de points totalement convergés calculés mais cette fois-ci avec l'approche multiparamétrique (figure 5.7(c)), on obtient un gain de temps de calcul de 20% (29100 secondes avec la méthode de cokrigage contre 36340 secondes avec l'approche traditionnelle).

Ce gain de temps s'explique par le fait que très tôt avec la méthode de cokrigeage, on se concentre à rechercher le maximum admissible global et que la zone où il se situe est rapidement localisée et que donc le calcul d'un nouveau point devient moins coûteux grâce à l'approche multiparamétrique et à l'ensemble des points calculés précédemment dans la zone d'intérêt. De ce fait, avec la méthode de cokrigeage, le critère d'arrêt d'enrichissement est atteint alors que dans le même temps, le calcul des points du tirage LHS initial des données totalement convergées n'est pas encore terminé pour la méthode de référence.

Le tableau 5.2 présente les différents résultats obtenus au cours de cette étude. On présente notamment :

- La qualité du métamodèle avant la mise en place de la stratégie d'enrichissement au maximum de CEI (20 points partiellement convergés et 6 totalement convergés pour le cokrigeage et 20 points totalement convergés pour la stratégie de référence).
- Les paramètres optimaux trouvés par les deux méthodes ainsi que le couple transmissible et la contrainte maximale dans les vis.
- Le coût de calcul nécessaire pour atteindre l'optimum trouvé par chacune des méthodes.

	Méthode de cokrigeage		Méthode de référence	
	$r_{ccc}$	$r^2$	$r_{ccc}$	$r^2$
Qualité du métamodèle $M_t$	0.88	0.79	0.98	0.99
Qualité du métamodèle $\max_{\forall i \in [1,12]} C_{Tr}^i$	0.98	0.96	0.96	0.98
$P_{t1/a}$ (mm)	-0.16		-0.157	
$P_{t2/a}$ (mm)	-0.155		-0.15	
$M_t$ (N.m)	3412		3388	
$\max_{\forall i \in [1,12]} C_{Tr}^i$ (MPa)	999.9		997.7	
	Sans multiparamétrique	Avec multiparamétrique	Sans multiparamétrique	Avec multiparamétrique
Temps CPU (h)	28.44	8.08	61.11	12.73
Gain temps CPU Cokrg./Ref.	2.15	1.58	-	-

**TABLE 5.2:** Tableau des résultats obtenus pour les deux approches de construction de métamodèle



On constate que la précision obtenue avec les 20 points LHS de la méthode de référence est meilleure qu'avec l'approche en cokrigage, mais la qualité obtenue avec ce métamodèle de cokrigage est suffisamment bonne pour localiser rapidement la zone d'intérêt, comme vu sur les figures 5.7.

On constate que pour les deux méthodes, la contrainte maximale supportée par les vis est à la limite des 1000 MPa à ne pas dépasser. L'optimisation s'est d'ailleurs révélée être plus efficace avec la méthode de cokrigage qui a permis de trouver un jeu de paramètres  $P_{t1/a}$  et  $P_{t2/a}$  amenant un plus grand couple transmissible qu'avec la méthode de référence.

On remarque non seulement que le résultat obtenu suite au processus d'optimisation est de meilleure qualité ( $\approx 0.7\%$ ) mais qu'il est également obtenu de manière plus rapide. L'optimum localisé par la méthode de cokrigage est trouvé au 20<sup>ème</sup> point calculé totalement convergé tandis que pour la méthode de référence il est trouvé au 46<sup>ème</sup>. Si les calculs ne sont pas effectués avec l'approche multiparamétrique, le gain de calcul obtenu est de 2.15 (un gain légèrement moindre est obtenu par [Forrester, 2004] avec l'Evofusion sur un cas test de profil d'aile). Avec l'approche multiparamétrique le gain est de 1.58. L'utilisation de la méthode de cokrigage est donc tout à fait intéressante pour mener à bien cette optimisation sous contrainte.

L'apport de la stratégie multiparamétrique est indéniable sur cet exemple puisque dans le cas de la stratégie multi-fidélité de cokrigage on divise le temps de calcul par 3.5 et dans l'approche traditionnelle par 4.8. Ainsi, on définit  $Gain_{optim}$  qui est le gain offert par le couplage de la méthode de cokrigage et l'approche multiparamétrique par :

$$Gain_{optim} = \frac{\text{Tps de calcul méthode de référence sans approche multiparamétrique}}{\text{Tps de calcul méthode de cokrigage avec approche multiparamétrique}} \quad (5.8)$$

$$= Gain_{optim_{cokrigage}} \times Gain_{optim_{multiparametrique}}$$

avec :

$$Gain_{optim_{cokrigage}} = \frac{\text{Tps de calcul méthode de référence sans approche multiparamétrique}}{\text{Tps de calcul méthode de cokrigage sans approche multiparamétrique}} \quad (5.9)$$

$$Gain_{optim_{multiparametrique}} = \frac{\text{Tps de calcul méthode de cokrigage sans approche multiparamétrique}}{\text{Tps de calcul méthode de cokrigage avec approche multiparamétrique}} \quad (5.10)$$

De ce fait, on trouve :

$$Gain_{optim} = 2.15 \times 3.52 \approx 7.6 \quad (5.11)$$

Le couplage entre méthode multi-fidélité (cokrigage) et de calcul multiparamétrique amène un gain de calcul de 7.6, ce qui sur ce cas test 3D à deux paramètres est un gain intéressant.

### 1.4.3 Bilan de l'étude

Le couplage de l'approche de calcul multiparamétrique et de la méthode de cokrigage s'avère très efficace sur cet exemple de type industriel. En effet, l'optimum trouvé dans le temps de calcul alloué est de meilleure qualité que celui obtenu avec la méthode de référence par krigage. De plus, le gain de calcul s'élève à 7.6.

De nouveau, on constate que l'utilisation de la méthode de cokrigage est profitable sur des cas test d'optimisation. Les résultats obtenus que ce soit en terme d'amélioration de l'optimum trouvé ou des gains de calculs, sont tout à fait comparables à ceux obtenus dans la littérature par [Forrester, 2004] sur un cas test de mécanique des fluides.

## 2 Création d'un abaque virtuel

### 2.1 Introduction

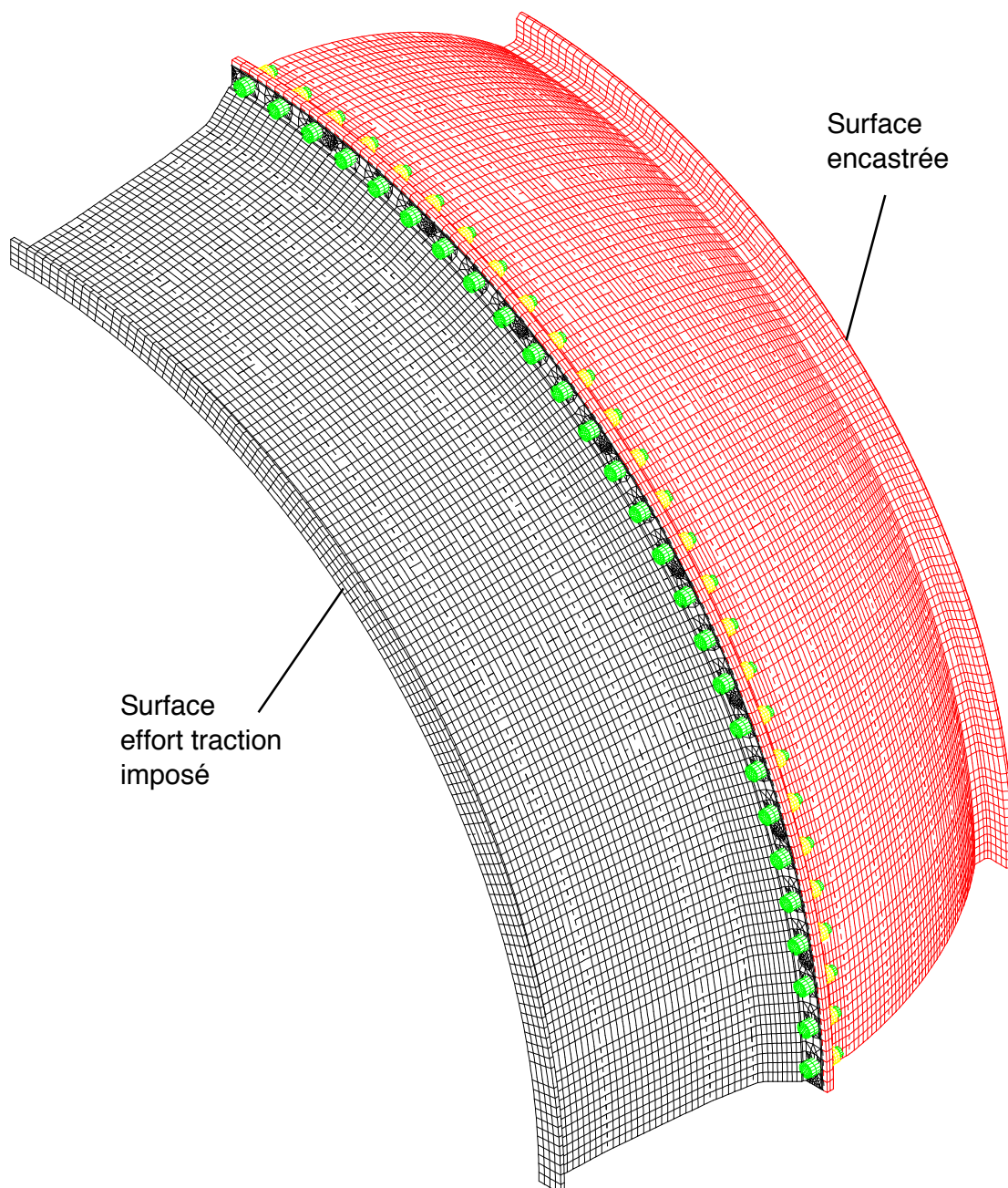
La stratégie de recalage, consistant à enrichir un métamodèle au point maximisant l'erreur quadratique moyenne (pour rappel voir figure 3.7) est mise en place sur un second cas test industriel. Le but est de construire un abaque virtuel afin de décrire le comportement d'un assemblage en fonction de 5 paramètres. Là encore, on compare le temps de calcul nécessaire pour construire cette surface de réponse avec la stratégie multi-fidélité de cokrigage et la stratégie de référence (krigage avec uniquement des données totalement convergées). Pour le problème traité, on ne dispose pas de modèle de référence. En effet, le temps de calcul nécessaire pour construire un modèle de référence serait assez conséquent (un jeu de paramètres à évaluer = 12 heures de calcul). La problématique traitée ici s'approche donc au plus près des préoccupations industrielles.

### 2.2 Présentation du cas test

Ce cas test est une libre adaptation d'une bride aéronautique. Le maillage représentant 1/4 de la structure complète est illustré à la figure 5.8.

La structure complète comporte au total 108 boulons. Pour des raisons de coût de calcul, le problème ne sera cependant qu'étudié sur un quart de la structure (des conditions de symétrie sont mises en place). Le problème mécanique traité consiste à imposer sur une surface de la bride un effort de traction uniforme. La fonction objectif que l'on cherche à représenter est le maximum de la contrainte de Von Mises dans les vis. 3 paramètres de précharge des boulons sont utilisés.

Pour un quart de la structure on dispose donc de 27 boulons, chacun des paramètres de précharge est lié à un groupe de 9 boulons. Ces paramètres sont notés  $P_{g1}$ ,  $P_{g2}$  et  $P_{g3}$ . Le problème comporte au total 5 paramètres dont l'effort de traction imposé noté  $F_{trac}$  et un jeu de fretage entre les deux parties de la bride (maillage noir et rouge sur la figure 5.8), noté  $J_{fret}$ .



**FIGURE 5.8:** Maillage 1/4 de la bride aéronautique

Les paramètres sont définis dans le tableau suivant :

Paramètres	min	max
$P_{g1}$ (en mm)	-0.08	-0.005
$P_{g2}$ (en mm)	-0.08	-0.005
$P_{g3}$ (en mm)	-0.08	-0.005
$J_{fret}$ (en mm)	-0.07	0
$F_{trac}$ (en k.N)	180	725

**TABLE 5.3:** Bornes de conception pour le cas test de la bride

La fonction objectif  $y$  est définie par :

$$y(\mathbf{x}) = \max_{\forall i \in \llbracket 1, 27 \rrbracket} C_{VM}^i(\mathbf{x}) \quad (5.12)$$

avec  $C_{VM}^i$  la contrainte de Von Mises dans la vis  $i$ .

Le cas test étudié est calculé avec des éléments de degré 2. Il comporte 55 000 éléments et 1 million de degrés de liberté. Comme pour le cas test précédent, le calcul s'effectue en deux steps de chargement : le premier concerne la mise en place de la précharge dans les boulons, le second correspond à la mise en place du chargement. Les niveaux de convergence des données totalement et partiellement convergées sont  $\eta_{fcv}^{(1)} = \eta_{pcv}^{(1)} = 1.10^{-4}$ ,  $\eta_{pcv}^{(2)} = 1.10^{-3}$  et  $\eta_{fcv}^{(2)} = 1.10^{-5}$ .

- Le temps de calcul d'un point partiellement et totalement convergé lors du premier step de calcul est de : 1993s
- Le temps de calcul d'un point totalement convergé lors du second step de calcul est de : 41556s
- Le temps de calcul d'un point partiellement convergé lors du second step de calcul est de : 1872s

Un calcul effectué sur le plus puissant des nœuds de calcul du cluster du LMT-Cachan (Bi Xeon(R) E5-2670 0) se fait donc en 12h. Certes, la stratégie multiparamétrique tend à faire diminuer le coût de calcul au fur et à mesure des points à calculer, mais on constate que la construction d'un métamodèle même avec quelques points peut alors prendre un temps relativement conséquent. L'emploi de la stratégie de recalage telle que présentée précédemment n'est pas la plus appropriée. Un aménagement de celle-ci est donc nécessaire pour les problèmes de plus grande taille. Une adaptation des outils numériques disponibles au sein du laboratoire pour mettre la résolution de ce genre de problème a donc été effectuée. Cette adaptation consiste à utiliser une parallélisation des calculs.

## 2.3 Adaptations des outils disponibles pour le calcul

### 2.3.1 Parallélisation des moyens de calcul

Afin de remédier à ce problème de temps de calcul on se propose d'utiliser une parallélisation des calculs. Cette approche a, par exemple, été utilisée dans [Sobester *et al.*, 2004].

On se propose d'utiliser une partie des capacités des moyens de calcul disponibles au laboratoire pour proposer une version parallélisée gros grain du solveur COFAST.

**Remarque :** Développer un solveur COFAST à résolution à seconds membres multiples a été par un temps envisagé. En effet, pour les problèmes d'assemblage avec la méthode LaTin, l'étape globale est numériquement semblable pour tous les paramètres d'assemblage testés dans cette thèse (jeux, coefficients de frottements, etc.). Utiliser une résolution à plusieurs seconds membres pourrait améliorer la vitesse de calcul (les données de calculs étant alors chargées en mémoire). Cependant, les performances atteintes ont été relativement décevantes.

On représente sur la figure 5.9 l'algorithme utilisé lors de l'utilisation de solveurs COFAST en parallèle.

On constate que si l'on dispose de  $n_{\text{solveur}}$ , il est nécessaire lors de la phase d'enrichissement de disposer d'un outil permettant de sélectionner  $n_{\text{solveur}}$  points. Dans le cadre de la stratégie de recalage, les points choisis sont les maxima de la fonction d'erreur quadratique moyenne du métamodèle. Dans le cadre d'une stratégie EGO, les points choisis sont les maxima de la fonction EI ou CEI selon les cas d'études. En disposant de plusieurs solveurs, il peut également être intéressant de mixer plusieurs types d'enrichissement. Ainsi, dans le cadre de cette thèse, il a été développé une toolbox d'optimisation multi-optima.

### 2.3.2 Toolbox d'optimisation multi-optima

La création de ce nouvel outil s'est principalement faite dans le cadre d'un projet de recherche de Master1. Deux étudiants du département de Génie Mécanique de l'ENS Cachan, Stéphane Nachar et Quentin Chanzy, ont travaillé sur ce projet. Le développement de cette toolbox s'appuie sur celle développée par Brian Birge [Birge, 2003], nommé PSOt et disponible sur la page web « MATLAB central »<sup>2</sup>. Cette toolbox permet l'optimisation globale d'une fonction par algorithme PSO.

Ce choix a notamment été motivé par le fait que la toolbox initiale PSOt est ouverte et peut être ainsi facilement modifiée. Cette toolbox s'intégrant dans une démarche de conception de surface ou de recherche d'optimum global par métamodèle, la recherche de gain de temps sur des fonctions peu coûteuses à évaluer n'était pas la priorité. En effet, le temps passé à optimiser une fonction d'enrichissement (maximum d'erreur quadratique moyenne ou maximum EI) reste anecdotique devant le temps de simulation d'un jeu de paramètres. Il a été alors choisi de privilégier une recherche de minima robuste.

Le fonctionnement de la toolbox s'appuie sur une découpe régulière de l'espace d'étude. D'un point de vue utilisateur, il est juste nécessaire de spécifier le nombre de découpes par

2. <http://www.mathworks.com/matlabcentral/fileexchange/7506-particle-swarm-optimization-toolbox>

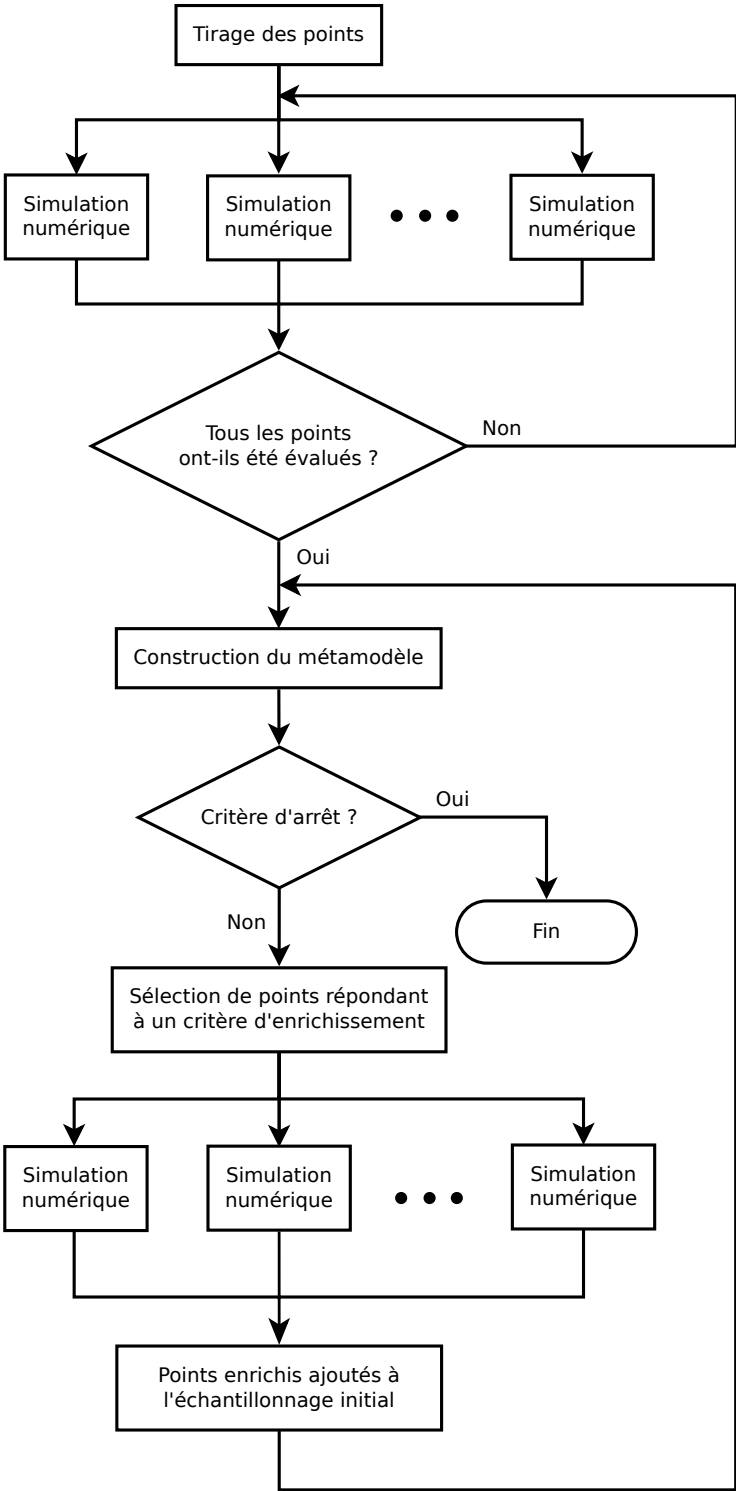


FIGURE 5.9: Algorithme avec utilisation de solveurs parallèles

dimension que l'on souhaite faire de l'espace. On pose  $p_{dec}$  le nombre de découpe souhaité. L'algorithme 5 présente la structure générale de la toolbox développée au sein du laboratoire.

---

**Algorithm 5** Principe de fonctionnement de la toolbox d'optimisation multi-optima

---

- 1: Découpe de l'espace d'étude en hypercube. Nombre d'hypercubes= $p_{dec}^{n_{dim}}$
  - 2: Recherche de l'optimum sur chacun des hypercubes par PSO (cette résolution peut se faire en multi-thread)
  - 3: Stockage de chacun des optima trouvés
  - 4: Post traitement des optima trouvés sur chacun des hypercubes afin de déterminer les optima de la fonction étudiée.
- 

La plus-value du travail réalisé par Stéphane Nachar et Quentin Chanzy se situe lors de la dernière étape qui permet de déterminer pour les  $p_{dec}^{n_{dim}}$  optima trouvés sur les hypercubes, lesquels sont réellement des optima de la fonction étudiée. Avec cette stratégie, il est possible de déterminer l'ensemble des optima globaux et locaux d'une fonction si l'espace est suffisamment découpé.

### 2.3.3 Stratégie multiparamétrique et parallélisme

La mise en place de la stratégie multiparamétrique peut être effectuée de plusieurs manières. La première approche consiste à utiliser comme solution de départ à un calcul d'un nouveau jeu de paramètres sur le processeur  $i$ , le jeu de paramètres le plus proche dans l'espace euclidien précédemment calculé sur **le processeur  $i$** . Une autre approche peut être mise en place. Elle consiste à rassembler toutes les solutions calculées sur chacun des processeurs en les stockant dans un COFAST auxiliaire. Un calcul associé à un nouveau jeu de paramètres peut alors être initialisé en fonction du point le plus proche dans l'espace euclidien calculé sur **tous les processeurs**. Lors des phases de tests, cette approche ne s'est pas avérée être la plus efficace. En effet, pour importer une solution dans un logiciel COFAST du processeur  $i$ , il est pour l'instant nécessaire d'importer également les maillages et de les fusionner avec celui actuellement en mémoire. Or, cette phase prend un temps de calcul non négligeable et demande une plus grande capacité de mémoire. Il a donc été choisi d'utiliser la première approche. Des développements supplémentaires sont donc à apporter pour optimiser au mieux cette gestion parallélisée des solveurs.

## 2.4 Construction de l'abaque virtuel

Pour la création de cet abaque virtuel 8 processeurs ont été utilisés. Comme précédemment, deux approches sont comparées :

- Construction d'un métamodèle initial de cokrigeage avec  $10 \times n_{dim} = 50$  points LHS adaptatif partiellement convergés (24 points LHS + 26 points enrichis) et  $3 \times n_{dim} = 15$  points totalement convergés. Par la suite, l'enrichissement se fait au maximum d'erreur quadratique du métamodèle de cokrigeage. Les points enrichis sont calculés de manière partiellement et totalement convergée.

- Construction d'un métamodèle initial avec  $10 \times n_{dim} = 50$  points LHS totalement convergés suivi d'un enrichissement au maximum de l'erreur quadratique moyenne. Il s'agit de la méthode de référence.

#### 2.4.1 Définition du critère d'arrêt

Le fait de calculer les points par groupe de 8 demande une légère modification du critère d'arrêt  $\bar{r}_{ccc}^v$  (voir équation 3.82 au paragraphe 2.3.2 du chapitre 3). Pour rappel, ce critère s'intéresse à mesurer la variation des métamodèles construits point enrichi par point enrichi. Lorsqu'il est jugé que les variations sont minimales ( $\bar{r}_{ccc}^v > 0.99$ , trois fois de suite) l'enrichissement est stoppé. Ici, puisque l'enrichissement se fait par groupe de 8 points, on fixe  $v = 1$ .

#### 2.4.2 Résultats obtenus

	Méthode de référence	Méthode de cokrigeage
Nbre d'appels au solveur pour calculer les points part. conv.	-	12
Nbre d'appels au solveur pour calculer les points tot. conv.	12	7
Nbre de points tot. conv. pour atteindre le critère arrêt	90 (50 points LHS + 40 enrichissement)	54 (15 initiaux + 39)
Temps de calcul CPU (en h) avec approche multiparamétrique	128.9	89.2
Temps de calcul CPU (en h) sans approche multiparamétrique	145.2	93.7
Gain de temps Ckrg./Ref. avec approche multiparamétrique		1.45
Gain de temps Ckrg./Ref. sans approche multiparamétrique		1.55

**TABLE 5.4:** Résultats obtenus avec les deux méthodes de construction de métamodèle

Le tableau de résultats 5.4 montre clairement l'intérêt d'utiliser l'approche multi-fidélité de cokrigeage. Une autre constatation que l'on peut faire est que la stratégie multiparamétrique montre une efficacité relativement limitée. La raison principale est que peu d'appels au solveur ont été effectués. De plus, beaucoup de points ajoutés l'ont été sur la frontière de l'espace de conception  $\mathcal{D}$  et donc la distance entre les points n'est pas nécessairement suffisamment faible pour rendre la stratégie multiparamétrique réellement efficace. Cependant, la parallélisation des calculs est vraiment intéressante puisqu'en 130h de calcul (temps nécessaire pour mener à bien la



stratégie de référence avec approche multiparamétrique), 12 appels aux solveurs ont été effectués, ce qui correspond sensiblement au nombre de points que l'on aurait évalué sans parallélisation.

Le gain total obtenu en couplant la stratégie de cokrigeage avec l'approche multiparamétrique par rapport à une stratégie classique de krigeage sans approche multiparamétrique est de **1.63**. Ce gain peut sembler faible par rapport à ceux obtenus précédemment mais si l'on considère que le même nombre de points pour atteindre le critère d'arrêt serait nécessaire avec la stratégie de référence sans parallélisation alors le gain total serait de **13**. En effet, le coût de calcul de l'approche de référence sans parallélisation et sans approche multiparamétrique serait de 1168h (48 jours).

Ne possédant pas de métamodèle de référence, il est difficile de quantifier la qualité des métamodèles obtenus par les deux méthodes. Cependant, on peut les comparer entre eux. Ainsi les métamodèles obtenus par la méthode de référence et de cokrigeage ont une corrélation  $r_{ccc} = 0.97$  et  $r^2 = 0.95$ . Les métamodèles construits par les deux approches sont donc de qualité comparable.

**Remarque :** En construisant un métamodèle de krigeage uniquement avec les données totalement convergées utilisées pour la méthode de cokrigeage, 10 appels au solveur (soit 78 points) seraient nécessaires pour atteindre le critère d'arrêt. Le coût de calcul associé serait alors de 110.5h. Ainsi, le procédé de cokrigeage pour la stratégie de recalage permet une réelle limitation du coût de calcul sur ce cas test de taille industrielle par rapport au procédé de krigeage avec les mêmes points totalement convergés, contrairement à ce qui a pu être vu ou dit sur les cas tests *cubes* et *1\_rainure* (voir paragraphes 2.4.5 et 2.5.5 et figures 3.17 et 3.27 du chapitre 3).

## 2.5 Bilan

La création d'abaque virtuel avec l'approche de cokrigeage multi-fidélité s'avère donc efficace sur cet exemple industriel. Couplée avec la stratégie de calcul multiparamétrique, un gain de temps de 1.63 est atteint par rapport à l'approche de référence. La réalisation de calculs parallélisés a également permis d'atteindre un gain de calcul conséquent. Sur ce cas test de taille industrielle, la stratégie de recalage couplée avec l'approche de cokrigeage multi-fidélité a permis une construction efficace de métamodèle.

Différents outils ont dû être adaptés pour la réalisation de cet abaque virtuel. Tout d'abord il a fallu construire l'architecture et la gestion des différents logiciels utilisés en parallèle. Il a également été nécessaire de développer un optimiseur multi-optima. La réalisation de cet optimiseur s'est faite en collaboration avec deux étudiants du département Génie Mécanique de l'ENS Cachan.

Les développements apportés lors de cette thèse ont donc permis de traiter de manière plus efficace la création de métamodèle d'un cas étudié de taille industrielle.

### 3 Développements supplémentaires et possibilités offertes par la toolbox GRENAT

L'ensemble des développements apportés à la toolbox GRENAT au cours de cette thèse et de celle de Luc Laurent offre un grand nombre de possibilités de construction de métamodèles. Les travaux précédemment réalisés [Laurent *et al.*, 2013a] ont mis au sein du laboratoire un outil permettant notamment de construire des métamodèles se basant sur le krigeage, les RBFs, le cokrigeage à gradients et les RBFs à gradients. Les derniers développements portant sur des approches multi-fidélité ont permis d'enrichir cet outil. La construction de métamodèle s'appuyant sur une approche multi-fidélité avec prise en compte des gradients est donc envisageable et facilement implémentable dans la toolbox GRENAT. Par facilement « implémentable » il faut comprendre « codable » car la construction de ce genre de métamodèle peut se heurter à de forts problèmes de conditionnement de matrice de corrélation.

#### 3.1 Construction de métamodèle multi-fidélité avec prise en compte des gradients

Avant de présenter le potentiel que peut offrir l'utilisation d'approches multi-fidélité avec prise en compte des gradients, les équations les constituant seront présentées. L'utilisation de données basse fidélité combinée avec l'information du gradient n'est pas nouvelle dans la construction de métamodèle mais ne semble pas très répandue. On peut cependant citer [Gano et Renaud, 2004, Han *et al.*, 2013, Yamazaki et Mavriplis, 2013, Ulaganathan *et al.*, 2014] qui utilisent des approches multi-fidélité différentes (corrections de surface, cokrigeage  $s_{fcv}/s_{pcv}$  ou cokrigeage de type auto-regressif). Dans un premier temps, on rappelle les équations du cokrigeage à gradients (*Gradient Enhanced Kriging*).

##### 3.1.1 Cokrigeage à gradients

Le cokrigeage à gradients ou le krigeage avec prise en compte de gradients (*Gradient Enhanced Kriging*) se construit sensiblement de la même façon que le cokrigeage multi-fidélité présenté au chapitre 3. Il s'agit de considérer l'information des gradients comme une donnée auxiliaire ([Wachernagel, 2003]).

Le processus gaussien associé à la variable principale est  $Y$ , le processus gaussien associé aux gradients est noté  $W$ . Celui-ci est défini par :

$$\forall i \in \llbracket 1, n_{dim} \rrbracket \quad W^i(\mathbf{x}) = \frac{\partial Y}{\partial x_i}(\mathbf{x}) \quad (5.13)$$

Dans le cadre d'une écriture de cokrigeage de gradients, les processus gaussiens associés sont alors définis par :

$$\begin{aligned} Y(\mathbf{x}) &= \mu_0(\mathbf{x}) + Z_0(\mathbf{x}) \\ \forall i \in \llbracket 1, n_{dim} \rrbracket \quad W^i(\mathbf{x}) &= \mu_i(\mathbf{x}) + Z_i^{grad}(\mathbf{x}) \end{aligned} \quad (5.14)$$

On retrouve les propriétés classiquement énoncées concernant ces processus gaussiens, à savoir :

$$\begin{aligned} E[Y(\mathbf{x})] &= \mu_0(\mathbf{x}) \\ \forall i \in \llbracket 1, n_{dim} \rrbracket \quad E[W^i(\mathbf{x})] &= \mu_i(\mathbf{x}) \end{aligned} \quad (5.15)$$

et :

$$\begin{aligned} E[Y(\mathbf{x})^2] &= \text{Var}[Y(\mathbf{x})] = \sigma^2 \\ \forall i \in \llbracket 1, n_{dim} \rrbracket \quad E[W^i(\mathbf{x})^2] &= \text{Var}[W^i(\mathbf{x})] = \sigma_{Z_i}^2 \end{aligned} \quad (5.16)$$

Dans le cadre du cokrigeage à gradients universel, on a :

$$\mu_0(\mathbf{x}) = \mathbf{f}_0(\mathbf{x})^T \boldsymbol{\beta} = \sum_{i=1}^p \beta_i f_i(\mathbf{x}) \quad (5.17)$$

avec  $\mathbf{f}_0(\mathbf{x}) = [f_1(\mathbf{x}) \ \dots \ f_p(\mathbf{x})]^T$  et  $\boldsymbol{\beta} = [\beta_1 \ \dots \ \beta_p]^T$

Classiquement, il est choisi de calculer les termes  $\mu_i$  comme étant les termes dérivés de  $\mu_0$  ([Morris *et al.*, 1993]). De ce fait :

$$\forall i \in \llbracket 1, n_{dim} \rrbracket \quad \mu_i(\mathbf{x}) = \frac{\partial \mu_0}{\partial x_i}(\mathbf{x}) = \sum_{j=1}^p \beta_j \frac{\partial f_j}{\partial x_i}(\mathbf{x}) = \mathbf{f}_0^i(\mathbf{x})^T \boldsymbol{\beta} \quad (5.18)$$

avec  $\forall i \in \llbracket 1, n_{dim} \rrbracket \quad \mathbf{f}_0^i = \left[ \frac{\partial f_1}{\partial x_i} \ \dots \ \frac{\partial f_p}{\partial x_i} \right]^T$ .

Pour la suite, on pose, comme pour le krigage décrit au chapitre 1,  $\mathbf{X}$ , le vecteur des  $n_s$  évaluations et  $\mathbf{Y}_{n_s}$  le vecteur des observations du processus gaussien et  $\mathbf{y}_{n_s}$  celui des évaluations de la fonction  $y$ . De la même façon, on pose  $\mathbf{dY}_{n_s}$  le vecteur des observations des gradients du processus gaussien et  $\mathbf{dy}_{n_s}$ , celui des gradients de la fonction  $y$ .

$$\mathbf{dY}_{n_s} = \left[ \frac{\partial Y(\mathbf{x}^{(1)})}{\partial x_1} \ \frac{\partial Y(\mathbf{x}^{(1)})}{\partial x_2} \ \dots \ \frac{\partial Y(\mathbf{x}^{(1)})}{\partial x_{n_{dim}}} \ \frac{\partial Y(\mathbf{x}^{(2)})}{\partial x_1} \ \dots \ \frac{\partial Y(\mathbf{x}^{(n_s)})}{\partial x_{n_{dim}}} \right]^T \quad (5.19)$$

$$\mathbf{dy}_{n_s} = \left[ \frac{\partial y(\mathbf{x}^{(1)})}{\partial x_1} \ \frac{\partial y(\mathbf{x}^{(1)})}{\partial x_2} \ \dots \ \frac{\partial y(\mathbf{x}^{(1)})}{\partial x_{n_{dim}}} \ \frac{\partial y(\mathbf{x}^{(2)})}{\partial x_1} \ \dots \ \frac{\partial y(\mathbf{x}^{(n_s)})}{\partial x_{n_{dim}}} \right]^T \quad (5.20)$$

De plus, on pose  $\tilde{\mathbf{y}}_{grad} = \begin{bmatrix} \mathbf{y}_{n_s} \\ \mathbf{dy}_{n_s} \end{bmatrix}$

L'estimateur de cokrigeage à gradients  $\hat{Y}^{grad}$  est recherché en un point quelconque  $\mathbf{x}$  sous la forme d'une combinaison linéaire de l'ensemble des variables primaires et secondaires :

$$\hat{Y}^{grad}(\mathbf{x}) = \mathbf{w}^T(\mathbf{x}) \mathbf{Y}_{n_s} + \mathbf{w}_{grad}^T(\mathbf{x}) \mathbf{dY}_{n_s} \quad (5.21)$$

Comme pour la stratégie de cokrigeage multi-fidélité présentée au paragraphe 1.1 du chapitre 3, on recherche cet estimateur sous la forme du meilleur estimateur linéaire non-biaisé. Ceci nous

amène finalement après développement des différentes équations à l'équation du métamodèle de cokrigage à gradients  $\hat{y}^{grad}$  :

$$\begin{aligned} \hat{y}^{grad}(\mathbf{x}) &= \mathbf{f}_0(\mathbf{x})^T \tilde{\boldsymbol{\beta}}_{grad} + \tilde{\mathbf{c}}_{grad}(\mathbf{x})^T \tilde{\mathbf{C}}_{grad}^{-1} (\tilde{\mathbf{y}}_{grad} - \tilde{\mathbf{F}}_{grad} \tilde{\boldsymbol{\beta}}_{grad}) \\ \text{avec : } \tilde{\boldsymbol{\beta}}_{grad} &= (\tilde{\mathbf{F}}_{grad}^T \tilde{\mathbf{C}}_{grad}^{-1} \tilde{\mathbf{F}}_{grad})^{-1} (\tilde{\mathbf{F}}_{grad}^T \tilde{\mathbf{C}}_{grad}^{-1} \tilde{\mathbf{y}}_{grad}) \end{aligned} \quad (5.22)$$

et à son estimateur de l'erreur quadratique moyenne :

$$\begin{aligned} s_{\hat{y}^{grad}}^2(\mathbf{x}) &= \left( \sigma^2 + \tilde{\mathbf{u}}_{grad}(\mathbf{x})^T (\tilde{\mathbf{F}}_{grad}^T \tilde{\mathbf{C}}_{grad}^{-1} \tilde{\mathbf{F}}_{grad})^{-1} \tilde{\mathbf{u}}_{grad}(\mathbf{x}) - \tilde{\mathbf{c}}_{grad}(\mathbf{x})^T \tilde{\mathbf{C}}_{grad}^{-1} \tilde{\mathbf{c}}_{grad}(\mathbf{x}) \right) \\ \text{avec : } \tilde{\mathbf{u}}_{grad}(\mathbf{x}) &= \tilde{\mathbf{F}}_{grad}^T \tilde{\mathbf{C}}_{grad}^{-1} \tilde{\mathbf{c}}_{grad}(\mathbf{x}) - \mathbf{f}_0(\mathbf{x}) \end{aligned} \quad (5.23)$$

On montre, en effectuant les différents développements pour trouver le meilleur estimateur non-biaisé que :

$$\tilde{\mathbf{F}}_{grad} = \begin{bmatrix} \mathbf{F} \\ \mathbf{dF} \end{bmatrix} \quad (5.24)$$

et  $\mathbf{dF} = [ \mathbf{f}_0^1(\mathbf{x}^{(1)}) \quad \mathbf{f}_0^1(\mathbf{x}^{(2)}) \quad \dots \quad \mathbf{f}_0^1(\mathbf{x}^{(n_s)}) \quad \mathbf{f}_0^2(\mathbf{x}^{(1)}) \quad \dots \quad \mathbf{f}_0^{n_{dim}}(\mathbf{x}^{(n_s)}) ]^T$ , cette matrice est de taille  $n_{dim} n_s \times p$ .

De même que pour le cokrigage multi-fidélité introduit au paragraphe 1.1 du chapitre 3, la matrice  $\tilde{\mathbf{C}}_{grad}^{-1}$  peut se décomposer en une matrice bloc  $2 \times 2$ .

$$\tilde{\mathbf{C}}_{grad} = \begin{bmatrix} \mathbf{C} & \mathbf{C}_{YW} \\ \mathbf{C}_{YW}^T & \mathbf{C}_{WW} \end{bmatrix} \quad (5.25)$$

La matrice  $\tilde{\mathbf{C}}_{grad}$  est constituée de la matrice de covariance  $\mathbf{C}$  employée pour le krigage, de la matrice d'intercovariance entre les réponses et les gradients  $\mathbf{C}_{YW}$  et de la matrice de covariance des gradients  $\mathbf{C}_{WW}$ .

Afin de calculer ces matrices, il est nécessaire d'établir la relation entre les données primaires et secondaires. Dans le cas de la prise en compte des gradients, ces relations sont mises en place comme proposé dans [Morris *et al.*, 1993] :

$$\begin{aligned} \text{Cov}(Z(\mathbf{x}^{(i)}), Z(\mathbf{x}^{(j)})) &= c_{ij} = c_{ji} = \sigma^2 R(\mathbf{x}^i, \mathbf{x}^j; \mathbf{l}) \\ \text{Cov}(Z(\mathbf{x}^{(i)}), \frac{\partial Z(\mathbf{x}^{(j)})}{\partial x_k}) &= c_{ij,k} = \sigma^2 \frac{\partial R}{\partial x_k}(\mathbf{x}^i, \mathbf{x}^j; \mathbf{l}) \\ \text{Cov}(\frac{\partial Z(\mathbf{x}^{(j)})}{\partial x_k}, Z(\mathbf{x}^{(i)})) &= -c_{ji,k} = -\sigma^2 \frac{\partial R}{\partial x_k}(\mathbf{x}^j, \mathbf{x}^i; \mathbf{l}) \\ \text{Cov}(\frac{\partial Z(\mathbf{x}^{(i)})}{\partial x_k}, \frac{\partial Z(\mathbf{x}^{(j)})}{\partial x_l}) &= c_{ij,kl} = -\sigma^2 \frac{\partial^2 R}{\partial x_k \partial x_l}(\mathbf{x}^i, \mathbf{x}^j; \mathbf{l}) \end{aligned} \quad (5.26)$$

**Remarques :** Il est important de remarquer quelques propriétés nécessaires à la réalisation du cokrigage à gradients.

- Il est nécessaire que la fonction de corrélation  $R$  soit dérivable.
- Les fonctions de corrélation sont des fonctions paires mais leurs dérivées premières sont impaires, ce qui explique la relation  $c_{ij,k} = -c_{ji,k} \forall (i, j, k) \in \llbracket 1, n_s \rrbracket^2 \times \llbracket 1, n_{dim} \rrbracket$

On en déduit alors les matrices  $\mathbf{C}_{YW}$  et  $\mathbf{C}_{WW}$ . La matrice  $\mathbf{C}_{YW}$  s'exprime comme telle :

$$\mathbf{C}_{YW} = \begin{bmatrix} c_{11,1} & \cdots & c_{11,n_{dim}} & c_{12,1} & \cdots & c_{1n_s,n_{dim}} \\ c_{21,1} & \cdots & c_{21,n_{dim}} & c_{22,1} & \cdots & c_{2n_s,n_{dim}} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ c_{21,1} & \cdots & c_{21,n_{dim}} & c_{22,1} & \cdots & c_{2n_s,n_{dim}} \end{bmatrix} \quad (5.27)$$

Tandis que la matrice  $\mathbf{C}_{WW}$  s'exprime comme suit :

$$\mathbf{C}_{WW} = \begin{bmatrix} \mathbf{C}_{WW}^{11} & \mathbf{C}_{WW}^{12} & \cdots & \mathbf{C}_{WW}^{1n_s} \\ \mathbf{C}_{WW}^{21} & \mathbf{C}_{WW}^{22} & \cdots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{C}_{WW}^{n_s 1} & \mathbf{C}_{WW}^{n_s 2} & \cdots & \mathbf{C}_{WW}^{n_s n_s} \end{bmatrix} \quad (5.28)$$

$$\text{avec } \forall (i, j) \in \llbracket 1, n_s \rrbracket^2, \mathbf{C}_{WW}^{ij} = \begin{bmatrix} c_{ij,11} & c_{ij,12} & \cdots & c_{ij,1n_{dim}} \\ c_{ij,21} & c_{ij,22} & \cdots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ c_{ij,n_{dim}1} & c_{ij,n_{dim}2} & \cdots & c_{ij,n_{dim}n_{dim}} \end{bmatrix}$$

De même le vecteur  $\tilde{\mathbf{c}}_{grad}(\mathbf{x})$  peut s'exprimer sous la forme

$$\tilde{\mathbf{c}}_{grad}(\mathbf{x}) = \sigma^2 \tilde{\mathbf{r}}_{grad}(\mathbf{x}) = \sigma^2 \begin{bmatrix} \mathbf{r}_0(\mathbf{x}) \\ \mathbf{r}_{grad}(\mathbf{x}) \end{bmatrix} \quad (5.29)$$

avec

$$\mathbf{r}_0(\mathbf{x}) = [ R(\mathbf{x}, \mathbf{x}^{(1)}; \mathbf{l}) \quad \dots \quad R(\mathbf{x}, \mathbf{x}^{(n_s)}; \mathbf{l}) ]^T$$

et

$$\mathbf{r}_{grad}(\mathbf{x}) = \left[ R(\mathbf{x}, \frac{\partial \mathbf{x}^{(1)}}{\partial x_1}; \mathbf{l}) \quad \dots \quad R(\mathbf{x}, \frac{\partial \mathbf{x}^{(n_s)}}{\partial x_{n_{dim}}}; \mathbf{l}) \right]^T$$

La détermination des paramètres de cokrigeage se fait par maximisation de la fonction de vraisemblance  $L_{grad}$  :

$$L_{grad} = \frac{1}{\sqrt{(2\pi)^{n_s(n_{dim}+1)} (\sigma^2)^{n_s(n_{dim}+1)} |\tilde{\mathbf{C}}_{grad}|}} e^{-\frac{(\tilde{\mathbf{y}}_{grad} - \tilde{\mathbf{F}}_{grad} \tilde{\boldsymbol{\beta}}_{grad})^T \tilde{\mathbf{C}}_{grad}^{-1} (\tilde{\mathbf{y}}_{grad} - \tilde{\mathbf{F}}_{grad} \tilde{\boldsymbol{\beta}}_{grad})}{2\sigma^2}} \quad (5.30)$$

**Remarques :** Pour éventuellement plus de précisions sur le cokrigeage à gradients le lecteur pourra se référer à [Morris *et al.*, 1993, Chung et Alonso, 2002, Wachernagel, 2003, Laurent, 2013]. Les différentes équations présentées dans ce paragraphe peuvent être retrouvées

en suivant la même démarche que celle établie pour déterminer les équations du cokrigage multi-fidélité. La démarche pour déterminer ces équations est en effet identique. Il faut bien sûr adapter la matrice de covariance mais connaissant les relations décrites aux équations 5.26, ceci ne pose pas de difficultés majeures d'écriture.

### 3.1.2 Cokrigage multi-fidélité avec prise en compte des gradients

Dans le cadre du cokrigage multi-fidélité avec prise en compte des gradients, on peut exprimer les équations du métamodèle et de son estimateur d'erreur assez simplement. Il est juste nécessaire de définir les processus gaussiens qui vont être utilisés. Dans le cadre du cokrigage multi-fidélité à deux niveaux de fidélité, deux processus étaient nécessaires à la construction du métamodèle. Dans le cadre du cokrigage multi-fidélité avec prise en compte des gradients, trois processus gaussiens sont définis :

$$\begin{aligned} Y_{fcv}(\mathbf{x}) &= \mu_{fcv}(\mathbf{x}) + Z_{fcv}(\mathbf{x}) \\ Y_{pcv}(\mathbf{x}) &= \mu_{pcv}(\mathbf{x}) + Z_{pcv}(\mathbf{x}) \\ \forall i \in \llbracket 1, n_{dim} \rrbracket \quad W_{fcv}^i(\mathbf{x}) &= \frac{\partial \mu_{fcv}(\mathbf{x})}{\partial x_i} + Z_i^{fcv-grad}(\mathbf{x}) \end{aligned} \quad (5.31)$$

**Remarque :** La prise en compte des gradients dans le cadre de notre étude ne s'effectue qu'au niveau haute fidélité. Mathématiquement, il peut également se faire au niveau basse fidélité, cependant dans notre cas d'étude le calcul des gradients s'effectuant par différence finie, il est compliqué d'obtenir un gradient de qualité au niveau basse fidélité par les données partiellement convergées.

L'estimateur est recherché sous la forme d'une combinaison linéaire :

$$\hat{Y}^{m-grad}(\mathbf{x}) = \mathbf{w}_{fcv}^T(\mathbf{x})\mathbf{Y}_{fcv} + \mathbf{w}_{pcv}^T(\mathbf{x})\mathbf{Y}_{pcv} + \mathbf{w}_{fcv-grad}^T(\mathbf{x})\mathbf{dY}_{fcv} \quad (5.32)$$

avec  $\mathbf{Y}_{fcv}$  le vecteur des évaluations des données totalement convergées,  $\mathbf{Y}_{pcv}$  le vecteur des évaluations des données partiellement convergées et  $\mathbf{dY}_{fcv}$  le vecteur des évaluations des gradients des points totalement convergés.

De même que pour les méthodes de cokrigage à gradients ou cokrigage multi-fidélité, on recherche cet estimateur sous la forme du meilleur estimateur linéaire non-biaisé. (La démarche reste la même que dans les paragraphes précédents).

L'équation du métamodèle et sa variance s'écrivent alors :

$$\hat{y}^{m-grad}(\mathbf{x}) = \tilde{\mathbf{f}}_m(\mathbf{x})^T \tilde{\boldsymbol{\beta}}_{m-grad} + \mathbf{c}_{m-grad}(\mathbf{x})^T \tilde{\mathbf{C}}_{m-grad}^{-1} (\tilde{\mathbf{y}}_{m-grad} - \tilde{\mathbf{F}}_{m-grad} \tilde{\boldsymbol{\beta}}_{m-grad}) \quad (5.33)$$

$$\begin{aligned} s_{\hat{y}^{m-grad}}^2(\mathbf{x}) &= \sigma^2 + \tilde{\mathbf{u}}_{m-grad}(\mathbf{x})^T (\tilde{\mathbf{F}}_{m-grad}^T \tilde{\mathbf{C}}_{m-grad}^{-1} \tilde{\mathbf{F}}_{m-grad})^{-1} \tilde{\mathbf{u}}_{m-grad}(\mathbf{x}) \\ &\quad - \tilde{\mathbf{c}}_{m-grad}(\mathbf{x})^T \tilde{\mathbf{C}}_{m-grad}^{-1} \tilde{\mathbf{c}}_{m-grad}(\mathbf{x}) \end{aligned} \quad (5.34)$$

$$\text{avec : } \tilde{\mathbf{u}}_{m-grad}(\mathbf{x}) = \tilde{\mathbf{F}}_{m-grad}^T \tilde{\mathbf{C}}_{m-grad}^{-1} \tilde{\mathbf{c}}_{m-grad}(\mathbf{x}) - \tilde{\mathbf{f}}_m(\mathbf{x})$$

avec :

$$\tilde{\mathbf{y}}_{m\_grad} = \begin{bmatrix} \mathbf{y}_{fcv} \\ \mathbf{y}_{pcv} \\ \mathbf{d}\mathbf{y}_{fcv} \end{bmatrix} \quad (5.35)$$

$$\tilde{\mathbf{f}}_m = \begin{bmatrix} \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix} \text{ voir équation 3.19}$$

$$\tilde{\mathbf{F}}_{m\_grad} = \begin{bmatrix} \mathbf{F}_{fcv} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{d}\mathbf{F}_{fcv} & \mathbf{0} \end{bmatrix} \quad (5.36)$$

La matrice de covariance s'exprime alors comme :

$$\tilde{\mathbf{C}}_{m\_grad} = \begin{bmatrix} \mathbf{C}^{fcv}(\mathbf{X}_{fcv}, \mathbf{X}_{fcv}) & \mathbf{C}^{f-pcv}(\mathbf{X}_{fcv}, \mathbf{X}_{pcv}) & \mathbf{C}_{YW}^{fcv}(\mathbf{X}_{fcv}, \mathbf{X}_{fcv}) \\ \mathbf{C}^{f-pcvT}(\mathbf{X}_{fcv}, \mathbf{X}_{pcv}) & \mathbf{C}^{pcv}(\mathbf{X}_{pcv}, \mathbf{X}_{pcv}) & \mathbf{C}_{YW}^{f-pcv}(\mathbf{X}_{fcv}, \mathbf{X}_{pcv}) \\ \mathbf{C}_{YW}^{fcvT}(\mathbf{X}_{fcv}, \mathbf{X}_{fcv}) & \mathbf{C}_{YW}^{f-pcvT}(\mathbf{X}_{fcv}, \mathbf{X}_{pcv}) & \mathbf{C}_{WW}^{fcv}(\mathbf{X}_{fcv}, \mathbf{X}_{fcv}) \end{bmatrix} \quad (5.37)$$

Il est bien sûr nécessaire d'adapter les notations et les formules comme il a été fait lors du chapitre 3 aux paragraphes 1.2.1, 1.2.3 et 1.2.5 selon l'approche de cokrigage multi-fidélité qui a été choisie.

La figure 5.10 illustre le cokrigage multi-fidélité à gradients (cokrigage Intercorrélation) sur l'exemple fourni par [Forrester *et al.*, 2007] et décrit au paragraphe 1.4 du chapitre 3.

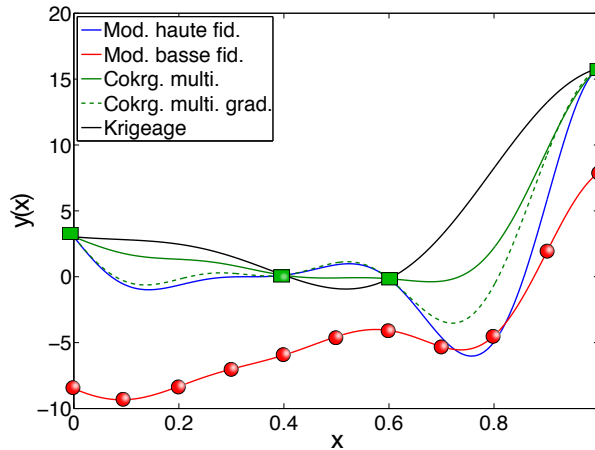


FIGURE 5.10: Métamodèle de cokrigage multi-fidélité à gradients

### 3.1.3 Krigeage hiérarchique à gradients

Le krigeage hiérarchique à gradients instauré dans la toolbox GREMAT ne souffre d'aucune difficulté technique supplémentaire par rapport à ce qui a été fait au cours des travaux de thèse de

L. Laurent [Laurent, 2013] et des présents travaux. Pour rappel, un métamodèle construit à l'aide du krigeage hiérarchique est un métamodèle de type krigeage mais dont la fonction de régression est définie à l'aide d'un métamodèle créé à partir des données basse fidélité. Ainsi, tout comme le krigeage hiérarchique ne nécessite qu'une modification de la matrice des données de régression  $\mathbf{F}$  utilisée pour le krigeage, le krigeage hiérarchique à gradients ne nécessite que l'adaptation de la matrice  $\tilde{\mathbf{F}}_{grad}$ . Les gradients à définir sont ceux du métamodèle utilisé comme fonction de régression du krigeage hiérarchique. Il n'est donc pas nécessaire de calculer les gradients au niveau basse fidélité.

### 3.1.4 Evofusion avec prise en compte des gradients

En s'inspirant des travaux de [Han *et al.*, 2013], on se propose de créer un modèle à gradients basé sur l'Evofusion. Le métamodèle d'erreur est ainsi évalué en tenant compte de la différence entre les évaluations des données partiellement et totalement convergées et de leurs gradients. Les gradients des données totalement convergées sont évalués via le solveur numérique. Les gradients des données partiellement convergées sont estimés à l'aide du métamodèle  $\hat{y}_{pcv}(\mathbf{x})$ . De ce fait :

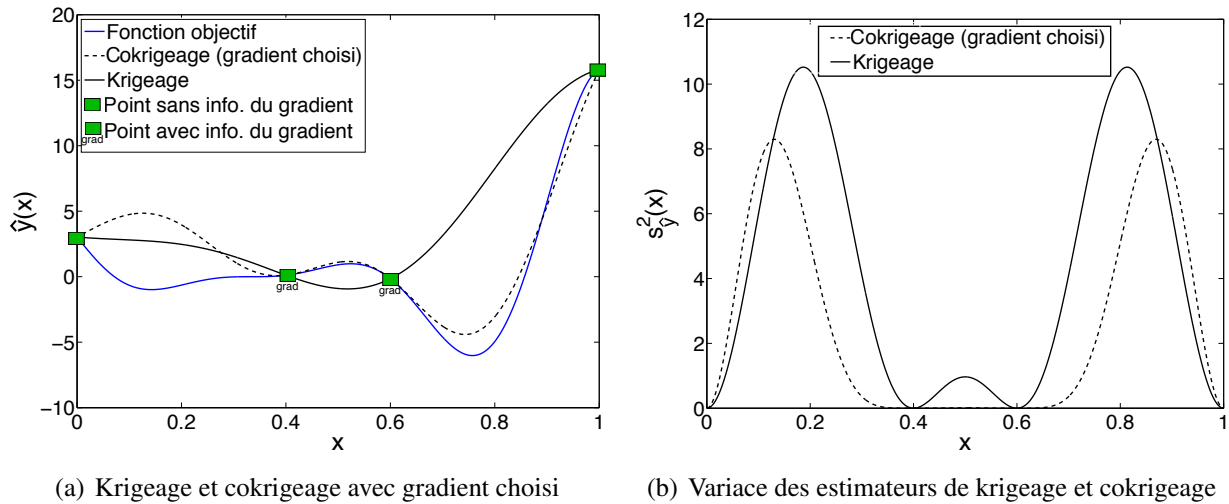
$$y_{err}(\mathbf{X}_{fcv}) = y(\mathbf{X}_{fcv}) - y_{pcv}(\mathbf{X}_{fcv})$$

$$\frac{\partial y_{err}(\mathbf{X}_{fcv})}{\partial x_i} = \frac{\partial (y(\mathbf{X}_{fcv}) - y_{pcv}(\mathbf{X}_{fcv}))}{\partial x_i} = \underbrace{\frac{\partial y(\mathbf{X}_{fcv})}{\partial x_i}}_{\substack{\text{Calculé à l'aide} \\ \text{du solveur} \\ \text{numérique par} \\ \text{différence finie}}} - \underbrace{\frac{\partial y_{pcv}(\mathbf{X}_{fcv})}{\partial x_i}}_{\substack{\text{Calculé à l'aide} \\ \text{du métamodèle} \\ \hat{y}_{pcv}}} \quad (5.38)$$

**Remarque :** pour rappel, la philosophie de l'Evofusion est de construire un métamodèle de krigeage à l'aide de données haute fidélité (totalement convergées) et de données basse fidélité (partiellement convergées) corrigées à l'aide du métamodèle d'erreur. **Le métamodèle évofusé avec prise en compte des gradients est donc un métamodèle de cokrigeage à gradients ne portant que sur les données haute fidélité.** Une option supplémentaire de la toolbox GREMAT a donc été nécessaire pour la construction de métamodèle ne prenant en compte l'information des gradients que pour certaines données. En figure 5.11, on illustre la réalisation d'un cokrigeage à gradients en certains points choisis. Un métamodèle évofusé sans prise ou avec prise en compte des gradients est présenté en figure 5.12.

La figure 5.11(a) montre la différence entre le métamodèle de krigeage et cokrigeage à gradients choisis ou sélectionnés (ces métamodèles sont construits avec la même fonction de corrélation). On constate sur la figure 5.11(b) que la variance de l'estimateur est plus faible avec le cokrigeage à gradients que celle avec le krigeage. On remarque également que le maximum de variance est plus proche des points dont l'information du gradient n'est pas disponible, ce qui est

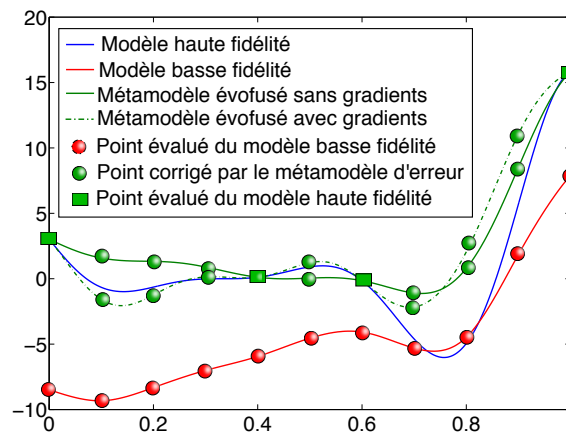




**FIGURE 5.11:** Comparaison des estimateurs et de cokrigage avec des gradients choisis

cohérent avec le procédé de construction. L'utilisation de gradients seulement en certains points choisis a été implémenté pour l'ensemble des méthodes multi-fidélité.

La figure 5.12 illustre la réalisation de la méthode d'Evofusion avec prise en compte des gradients.



**FIGURE 5.12:** Métamodèle évofofusé avec ou sans prise en compte des gradients

### 3.2 Illustrations des performances atteintes sur deux cas test mécaniques

Sur les cas test mécaniques *cubes* et *2\_rainures*, on se propose de comparer la qualité globale des métamodèles obtenus construits avec  $10 \times n_{dim}$  points partiellement convergés et  $3 \times n_{dim}$  points totalement convergés. Les gradients sont évalués à l'aide d'un calcul par différence finie.

Les tableaux 5.5 et 5.6 présentent les résultats obtenus en terme de temps de calcul nécessaire pour évaluer les différentes données à l'aide du solveur numérique et de la qualité globale obtenue pour représenter la fonction objectif des deux cas test.

<i>Cubes</i>	Sans gradients				Avec gradients				
	Evofusion	Ckrg.	Krg. Hier.	Krg.	Evofusion	Ckrg.	Krg. Hier.	Krg.	
Tps (s)	50				41.2	83.9			74.5
$r_{ccc}$	0.88	0.89	0.87	0.81	0.92	0.94	0.93	0.87	
$r^2$	0.81	0.83	0.79	0.7	0.90	0.92	0.90	0.81	
$Q_2$	65.55	58.13	76.02	108.6	40.91	32.95	39.20	67.07	

**TABLE 5.5:** Résultats de la qualité des méthodes multi-fidélité avec prise en compte ou non des gradients pour le cas test *cubes*

<i>2_rainures</i>	Sans gradients				Avec gradients				
	Evofusion	Ckrg.	Krg. Hier.	Krg.	Evofusion	Ckrg.	Krg. Hier.	Krg.	
Tps (s)	287				224	553			490
$r_{ccc}$	0.71	0.7	0.73	0.34	0.82	0.83	0.84	0.75	
$r^2$	0.56	0.62	0.57	0.2	0.75	0.77	0.75	0.7	
$Q_2$	0.246	0.217	0.249	0.494	0.146	0.134	0.145	0.199	

**TABLE 5.6:** Résultats de la qualité des méthodes multi-fidélité avec prise en compte ou non des gradients pour le cas test *2\_rainures*

On constate que l'apport des gradients est indéniable pour chacune des méthodes testées. Il peut donc être très intéressant de calculer les gradients afin d'améliorer la qualité des métamodèles. Cependant, ce calcul de gradients se fait actuellement par une estimation à l'aide de différence finie et le coût de calcul associé à cette estimation peut s'avérer important [Laurent, 2013]. En revanche si le calcul des gradients est obtenu à l'aide d'un problème adjoint par exemple, cela peut s'avérer être très intéressant. De plus, on remarque sur le cas test *cubes* que les méthodes multi-fidélité (sans prise en compte des gradients) sont plus efficaces (moins coûteuses et de meilleure qualité) que le cokrigeage à gradients. Ceci tend donc à montrer de nouveau l'efficacité de ces méthodes multi-fidélité pour construire un métamodèle de meilleure qualité par rapport à une approche par krigeage en un temps de calcul réduit. A titre informatif le traitement de l'exemple *1\_rainure* a été en parti réalisé dans [Soulier *et al.*, 2015].



# Conclusions et perspectives

Dans le prolongement des travaux précédemment effectués au laboratoire du LMT-Cachan, cette thèse vise à élaborer une méthode d'optimisation globale des assemblages. En effet, la simulation numérique d'un problème mécanique d'assemblage peut requérir un temps de calcul relativement conséquent. Ce coût de calcul est notamment dû au traitement de non-linéarités représentatives du comportement de l'assemblage. Il est ainsi nécessaire en vue de mener à bien un processus d'optimisation de proposer une méthode alternative aux méthodes d'optimisation directe s'appuyant sur le modèle mécanique utilisé. L'approche utilisée est une approche **à deux niveaux de modèles** permettant de mener à bien cette optimisation d'assemblages.

Deux niveaux d'optimisation sont alors utilisés. Le premier s'appuie sur la construction d'un modèle de substitution permettant d'évaluer la fonction objectif en un temps de calcul très faible. Le second niveau s'appuie sur une optimisation locale menée sur le modèle mécanique utilisé et dont le point de départ est le résultat de l'optimisation au premier niveau.

Deux points importants ont constitué le fondement de cette thèse :

- La méthode de résolution numérique qui s'appuie sur la méthode LaTIn. Cette méthode permet de résoudre de manière robuste les problèmes d'assemblages traités. Ces problèmes ont notamment pour paramètres des coefficients de frottement, de jeux, de précharges ou de défauts géométriques. Ils sont donc relativement similaires et l'utilisation de la **stratégie LaTIn multiparamétrique** utilisée sous COFAST permet alors de réduire les coûts de simulation au fur et à mesure des jeux de paramètres simulés.
- Le modèle de substitution employé qui permet d'estimer la fonction objectif à optimiser. Les métamodèles employés ici sont des métamodèles **multi-fidélité** permettant la prise en compte de données de différents types. La possibilité offerte par la méthode LaTIn de disposer d'une estimation de la solution mécanique au problème posé en temps et en espace à chaque itération de l'algorithme de résolution, a notamment permis de construire ces métamodèles multi-fidélité à partir de **données calculées totalement convergées** (la solution calculée est supposée exacte) et **partiellement convergées** (la solution est inexacte mais peu coûteuse à estimer).

Les travaux effectués durant cette thèse s'appuient sur trois thématiques : **le calcul numérique d'assemblages**, la **construction de métamodèles** et finalement **l'optimisation**. Le thème d'étude central a cependant porté sur les méthodes de construction de métamodèles multi-fidélité couplées avec l'utilisation de la stratégie de calcul multiparamétrique.

Le premier chapitre présente les trois thématiques abordées au cours de cette thèse. Dans un premier temps, le contexte général de l'optimisation est rappelé ainsi que la définition des

problèmes classiques et que certaines méthodes de résolution utilisées au cours de ces travaux. Après avoir présenté la stratégie multi-niveaux utilisée, les différents éléments constituant cette approche ont été présentés. On s'est notamment intéressé au procédé de krigeage permettant une construction efficace de métamodèles quant à représenter la fonction objectif. Le krigeage est d'ailleurs le procédé de base utilisé tout au long de ces travaux. La méthode LaTIn pour les assemblages est ensuite rappelée. On peut notamment retenir que grâce au fait que l'algorithme associé à la méthode n'est pas basée sur un schéma incrémental comme la plupart des autres solveurs, il est alors possible d'obtenir à chaque itération une approximation de la solution au problème mécanique. Un indicateur d'erreur est d'ailleurs associé à chaque solution. Celui-ci permet de définir le niveau de convergence des données. Finalement, une présentation de différentes méthodes pouvant s'insérer dans un processus d'optimisation et prenant en compte des données pour différents niveaux de modélisation ou de convergence a été mise en avant.

Le deuxième chapitre a pour but de démontrer la possibilité d'utiliser de manière efficace et robuste des données partiellement convergées dans un processus de création de métamodèles de qualité (c'est-à-dire représentatif de la fonction objectif) ou d'optimisation. Il a été notamment mis en avant sur deux cas test mécaniques que l'on peut déterminer :

- un niveau de convergence pour les données basse fidélité,
- un nombre initial de points pour construire le métamodèle multi-fidélité,

qui s'avèrent efficaces pour atteindre les deux objectifs précédents. Sur les cas test en 2 dimensions proposés, il s'est avéré que le **niveau de convergence** efficace pour définir les métamodèles multi-fidélité était  $\eta = 0.05$  avec un **nombre initial de points partiellement convergés de  $10 \times n_{\text{dim}}$** . Des **gains de temps de calcul** par rapport à la stratégie de métamodélisation par krigeage sont alors à envisager.

Au vu des résultats précédemment obtenus, le troisième chapitre vise à comparer différentes méthodes de construction de métamodèles multi-fidélité. Les approches ont été notamment comparées sur des cas test mécaniques. Cinq approches ont été mises en œuvre :

- l'Evofusion (méthode de correction de surface)
- 3 approches de cokrigeage
- le krigeage hiérarchique

Deux grandes conclusions sont à retenir de ce chapitre :

- La première est que sur les cinq approches multi-fidélité testées, trois se sont avérées particulièrement intéressantes. Il s'agit de l'**Evofusion**, du **cokrigeage de type « Intercorrélation »** et du **krigeage hiérarchique**. D'un point de vue technique ces méthodes offrent également l'avantage d'être facilement implémentable dès lors que l'on dispose d'un code de krigeage. En effet, la méthode d'Evofusion utilisée ici ne s'appuie que sur des métamodèles de krigeage, le krigeage hiérarchique ne nécessite que la modification de la fonction de régression et le cokrigeage Intercorrélation la modification de la matrice de corrélation « étendue » aux données partiellement convergées.
- La seconde est qu'à **coût de calcul fixé** la qualité moyenne des métamodèles obtenus est meilleure avec les approches multi-fidélité citées précédemment qu'avec le procédé

de krigeage des données totalement convergées. Cependant, à **qualité exigée** le coût de calcul moyen pour obtenir les différents métamodèles est sensiblement le même que le procédé utilisé soit de krigeage ou de multi-fidélité. Ceci s'explique notamment par le fait que le calcul de plusieurs données partiellement convergées n'est pas nécessairement négligeable et que la stratégie multiparamétrique tend à réduire les coûts de calcul associés aux données totalement convergées. Ainsi, même si le procédé de krigeage exige plus de points calculés de manière totalement convergée que les procédés multi-fidélité, le coût de calcul n'en est pas nécessairement moindre. **Si l'on souhaite construire un métamodèle avec une qualité minimale exigée relativement importante, les procédés multi-fidélité ne sont pas nécessairement plus performants que le krigeage.** Cependant, les exemples testés sont de petites tailles et relativement rapides à calculer, cette remarque est alors à nuancer car sur l'exemple mécanique de taille industrielle traité au chapitre 5 un gain de calcul de l'ordre de 1.5 (en fonction de l'utilisation ou non de l'approche de calcul multiparamétrique) a alors été atteint.

Le quatrième chapitre vise sur un simple exemple mécanique 2D à incorporer les méthodes de modélisation multi-fidélité dans le processus d'optimisation à deux niveaux. Les trois approches multi-fidélité précédentes ont alors été testées avec différents critères d'arrêt d'enrichissement de l'échantillonnage de l'espace de conception lors du premier niveau d'optimisation et différentes méthodes de tirages. La méthode de cokrigeage de type Intercorrélation s'est révélée être la stratégie multi-fidélité la plus efficace. A l'aide d'une stratégie d'optimisation locale se basant sur les méthodes à région de confiance le second niveau d'optimisation a été présenté pour la méthode de cokrigeage. Un facteur de gain de calcul supérieur à 4 a été atteint grâce au couplage entre le cokrigeage multi-fidélité et l'approche de calcul multiparamétrique. **Sachant que le problème d'optimisation ne comporte que deux paramètres et est en 2D, ce gain est tout à fait prometteur et met en avant le fait que les méthodes de métamodèles multi-fidélité s'insèrent de manière efficace dans le processus d'optimisation.**

Finalement, il a été proposé de tester l'approche de cokrigeage multi-fidélité afin d'optimiser un **problème de type industriel** et construire une surface de réponse pour un autre cas industriel de grande taille. **Le cokrigeage s'est alors avéré être dans les deux cas plus performants en terme de précision et de temps de calcul que la méthode de référence basée sur le krigeage.** Dans un processus d'optimisation, l'utilisation d'approches multi-fidélité couplée avec la stratégie de calcul multiparamétrique amène alors des gains de calcul intéressants. Cependant, en vue de construire des surfaces de réponse, bien que pouvant être plus efficace que le krigeage comme vu sur le cas test industriel, privilégier la puissance de calcul comme il a été fait en parallélisant le solveur semble plus prometteur. De plus, des méthodes de calcul plus adaptées à ce genre de problématique comme la PGD multiparamétrique peuvent être utilisées [Relun, 2011].

Les différents travaux menés au cours de cette thèse ont conduit à développer différents moyens de construction de métamodèles multi-fidélité qui sont maintenant disponibles au sein de la toolbox GRENAT. Il a également été nécessaire de développer des outils d'optimisation multi-optima compatibles avec l'utilisation de plusieurs solveurs utilisés en parallèle.

Plusieurs perspectives à cette thèse peuvent être développées :

— à court terme :

- Afin de répondre plus précisément aux demandes des industriels, des applications sur des cas test avec un plus grand nombre de paramètres à optimiser et de fonctions contraintes à prendre en compte est à envisager.
- L'utilisation de critère d'enrichissement autre que l'*expected improvement* comme par exemple le *weighted expected improvement* est également à envisager afin de renforcer le caractère aléatoire de la recherche du minimum global sur des fonctions objectif à plusieurs paramètres. En effet, il a été montré que même sur un exemple à peu de paramètres, le minimum global pouvait ne pas être localisé avec le critère d'enrichissement d'*expected improvement*.
- L'intégration plus robuste de l'utilisation de solveur COFAST en parallèle est à approfondir (possibilité de choisir comme solution d'initialisation à un nouveau jeu de paramètres une des solutions calculée sur l'ensemble des processeurs).

— à moyen et à long terme :

- L'utilisation de correction de surface hybride [Han *et al.*, 2013] semble prometteuse et peut être développée au sein de la toolbox GREMAT.
- L'intégration d'indicateurs d'erreurs sur des quantités locales pourraient permettre de mieux maîtriser le niveau de convergence des données partiellement convergées considéré. Il pourrait également permettre d'adapter plus finement la réponse des données totalement convergées et de diminuer le bruit numérique qui a été problématique lors de la phase d'optimisation locale.
- L'utilisation des gradients (ou seulement de certains) pour la construction des métamodèles multi-fidélité a été mise en perspective durant cette thèse et demande à être plus largement testée sur différents cas test afin de déterminer si elle peut réellement amener une économie de temps de calcul.
- L'intégration de la méthode PGD dans le cadre de l'optimisation devrait également permettre des réductions de coûts de calcul. Des développements importants sur la prise en compte des variabilités paramétriques sont encore à mener mais la méthode semble prometteuse et des premiers travaux s'appuyant sur la POD existent déjà [Zimmermann, 2013].

# Bibliographie

- [Alexandrov *et al.*, 1998] ALEXANDROV, N., DENNIS JR., J. E., LEWIS, R. M. et TORCZON, V. (1998). A trust region framework for managing the use of approximation models in optimization. *Structural Optimization*, 15:16–23.
- [Alexandrov *et al.*, 2000] ALEXANDROV, N. M., LEWIS, R. M., GUMBERT, C. R., GREEN, L. L. et NEWMAN, P. A. (2000). Optimization with variable-fidelity models applied to wing design. In *Proceedings of the 38th Aerospace Sciences Meeting & Exhibit*. AIAA Paper 2000-0841.
- [Allaire, 2007] ALLAIRE, G. (2007). *Conception optimale de structures*. volume 58 de *Mathématiques & Applications*. Springer-Verlag, Berlin.
- [Allix, 1992] ALLIX, O. (1992). Damage analysis of delamination around a hole. In *New Advances in Computational Structural Mechanics*. Sous la dir. de : LADEVÈZE, P. et ZIENKIEWICZ, O. C. pages 411–421. Elsevier Science Publishers B.V.
- [Anderson, 1953] ANDERSON, R. L. (1953). Recent advances in finding best operating conditions. *Journal of the American Statistical Association*, 48(264):pp. 789–798.
- [Arrow *et al.*, 1958] ARROW, K., HURWICZ, L. et UZAWA, H. (1958). *Studies in Nonlinear Programming*. Stanford University Press, Stanford, CA.
- [Bäck *et al.*, 1997] BÄCK, T., HAMMEL, U. et SCHWEFEL, H.-P. (1997). Evolutionary computation : comments on the history and current state. *IEEE Transactions on Evolutionary Computation*, 1(1):3–17.
- [Bandler *et al.*, 1994] BANDLER, J. W., BIERNACKI, R. M., CHEN, S. H., GROBELNY, P. A. et HEMMERS, R. H. (1994). Space mapping technique for electromagnetic optimization. *IEEE Transactions on Microwave Theory and Techniques*, 42:2536–2544.
- [Barton, 1998] BARTON, R. R. (1998). Simulation metamodels. In *Proceedings of the 1998 Winter Simulation Conference*, Sous la dir. de : MEDEIROS, D. J., WATSON, E. F., CARSON, J. S. et MANIVANNAN, M. S. pages 167–174. IEEE Computer Society Press.
- [Barton et Meckesheimer, 2006] BARTON, R. R. et MECKESHEIMER, M. (2006). Metamodel-based simulation optimization. In *Handbooks in Operations Research and Management Science : Simulation*. Sous la dir. de : HENDERSON, S. G. et NELSON, B. L. volume 13, chapitre 19, pages 535–574. Elsevier.
- [Becker et Lago, 1970] BECKER, R. W. et LAGO, G. V. (1970). A global optimization algorithm. In *Proceedings of the 8th Allerton Conference on Circuits and Systems Theory*. pages 3–12, Monticello, Illinois.



- [Bendsøe, 1995] BENDSØE, M. P. (1995). *Optimization of Structural Topology, Shape and Material*. Springer-Verlag, Berlin Heidelberg.
- [Bergmann, 2004] BERGMANN, M. (2004). *Optimisation aérodynamique par réduction de modèle POD et contrôle optimal. Application au sillage laminaire d'un cylindre circulaire*. Thèse de doctorat, Institut national polytechnique de Lorraine, Nancy, France.
- [Bichon *et al.*, 2008] BICHON, B., ELDRED, M., SWILER, L., MAHADEVAN, S. et MCFARLAND, J. (2008). Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA Journal*, 46:2459–2468.
- [Birge, 2003] BIRGE, B. (2003). PSOt—A particle swarm optimization toolbox for use with Matlab. In *IEEE, Swarm Intelligence Symposium*. pages 182–186. IEEE.
- [Blanzé *et al.*, 2004] BLANZÉ, C., CHAMPANEY, L. et BOUCARD, P.-A. (2004). Advanced tools for design and analysis for assemblies of structure with uncertainties. In *ECCOMAS 2004 - 4th European Congress on Computational Methods in Applied Sciences and Engineering*, Sous la dir. de : NEITTAANMÄKI, P., ROSSI, T., MAJAVA, K. et PIRONNEAU, O.
- [Blanzé *et al.*, 1996] BLANZÉ, C., CHAMPANEY, L., COGNARD, J.-Y. et LADEVÈZE, P. (1996). A modular approach to structure assembly computations : application to contact problems. *Engineering Computations*, 13(1):15–32.
- [Boucard, 1996] BOUCARD, P.-A. (1996). *Approche à grand incrément de temps en grandes transformations*. Thèse de doctorat, ENS-Cachan, Cachan, France.
- [Boucard, 2001] BOUCARD, P.-A. (2001). Application of the LaTin method to the calculation of response surfaces. In *Proceeding of the First MIT Conference on Computational Fluid and Solid Mechanics*, Sous la dir. de : BATHE, K. J. pages 78–81, Cambridge, USA. Elsevier Science Ltd.
- [Boucard *et al.*, 2009] BOUCARD, P.-A., BUYTET, S. et GUIDAULT, P.-A. (2009). A multiscale strategy for structural optimization. *International Journal for Numerical Methods in Engineering*, 78(1):101–126.
- [Boucard et Champany, 2003] BOUCARD, P.-A. et CHAMPANEY, L. (2003). A suitable computational strategy for the parametric analysis of problems with multiple contact. *International Journal for Numerical Methods in Engineering*, 57(9):1259–1282.
- [Boucard et Champany, 2004] BOUCARD, P.-A. et CHAMPANEY, L. (2004). Approche multirésolution pour l'étude paramétrique d'assemblages par contact et frottement. *Revue Européenne des Eléments Finis*, 13(5/7):437–448.
- [Box et Wilson, 1951] BOX, G. E. P. et WILSON, K. B. (1951). On the experimental attainment of optimum conditions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 13(1):1–45.
- [Broyden, 1970] BROYDEN, C. G. (1970). The convergence of a class of double-rank minimization algorithms. *IMA Journal of Applied Mathematics*, 6(1):76–90.
- [Buytet *et al.*, 2006] BUYTET, S., BOUCARD, P.-A. et GUIDAULT, P.-A. (2006). Optimisation multi-niveaux de modèles - revue bibliographique. Rapport omd 5.1, LMT-Cachan.

- [Buytet *et al.*, 2008] BUYTET, S., BOUCARD, P.-A. et GUIDAULT, P.-A. (2008). Optimisation multi-niveaux de modèles-Application à l'optimisation des paramètres de liaison dans les assemblages. Rapport technique, ANR-05-RTLN-009. ANR Projet OMD.
- [Carter, 1991] CARTER, R. G. (1991). On the global convergence of trust region algorithms using inexact gradient informations. *SIAM Journal on Numerical Analysis*, 28(1):251–265.
- [Champaney, 1996] CHAMPANEY, L. (1996). *Une nouvelle approche modulaire pour l'analyse d'assemblages de structures tridimensionnelles*. Thèse de doctorat, ENS-Cachan, Cachan, France.
- [Champaney *et al.*, 2005] CHAMPANEY, L., BLANZÉ, C. et BOUCARD, P.-A. (2005). Outils de dimensionnement et d'analyse de sensibilité pour les assemblages de structures. *Mécanique & Industries*, 6:89–99.
- [Champaney et Boucard, 2007] CHAMPANEY, L. et BOUCARD, P.-A. (2007). Sur la prise en compte des variations de chargement et de géométrie dans les assemblages. In *18ème Congrès Français de Mécanique*. Grenoble.
- [Champaney *et al.*, 2008] CHAMPANEY, L., BOUCARD, P.-A. et GUINARD, S. (2008). Adaptive multi-analysis strategy for contact problems with friction. application to aerospace bolted joints. *Computational Mechanics*, 42(2):305–316.
- [Chen et Barnhart, 2008] CHEN, C.-C. et BARNHART, H. X. (2008). Comparison of ICC and CCC for assessing agreement for data without and with replications. *Computational Statistics & Data Analysis*, 53:554–564.
- [Chen et Yang, 2005] CHEN, T.-Y. et YANG, C.-M. (2005). Multidisciplinary design optimization of mechanisms. *Advances in engineering software*, 36(5):301–311.
- [Chinesta *et al.*, 2011] CHINESTA, F., LADEVEZE, P. et CUETO, E. (2011). A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18(4):395–404.
- [Choi *et al.*, 2005] CHOI, S., ALONSO, J. J., KIM, S., KROO, I. M. et WINTZER, M. (2005). Two-level multi-fidelity design optimization studies for supersonic jets. In *43rd AIAA Aerospace Sciences Meeting and Exhibit*. Reno, Nevada. AIAA Paper 2005 - 0531.
- [Chung et Alonso, 2002] CHUNG, H. et ALONSO, J. (2002). Using gradients to construct co-kriging approximation models for high-dimensional design optimization problems. In *40th AIAA Aerospace Sciences Meeting and Exhibit*, Sous la dir. de : CITESSEER. Reno, Nevada.
- [Clerc, 2005] CLERC, M. (2005). *L'optimisation par essaims particulières : versions paramétriques et adaptatives*. Hermes-Lavoisier. Paris.
- [Clerc et Kennedy, 2002] CLERC, M. et KENNEDY, J. (2002). The particle swarm-explosion, stability, and convergence in multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6(1):58–73.
- [Clerc et Siarry, 2004] CLERC, M. et SIARRY, P. (2004). Une nouvelle métaheuristique pour l'optimisation difficile : la méthode des essaims particulières. *J3eA*, 3:1–13 DOI : 10.1051/bib-j3ea :2004007.

- [Cognard, 1989] COGNARD, J.-Y. (1989). *Une nouvelle approche des problèmes de plasticité et viscoplasticité par la méthode à grand incrément de temps*. Thèse de doctorat, Université Pierre et Marie Curie, Paris, France.
- [Conn *et al.*, 2000] CONN, A. R., GOULD, N. I. M. et TOINT, P. L. (2000). *Trust-region methods*. SIAM, Philadelphia, PA, USA.
- [Cook et Nachtsheim, 1980] COOK, R. D. et NACHTSHEIM, C. J. (1980). A comparison of algorithms for constructing exact d-optimal designs. *Technometrics*, pages 315–324.
- [Courrier *et al.*, 2014] COURRIER, N., BOUCARD, P.-A. et SOULIER, B. (2014). The use of partially converged simulations in building surrogate models. *Advances in Engineering Software*, 67:186–197.
- [Courrier *et al.*, 2015] COURRIER, N., BOUCARD, P.-A. et SOULIER, B. (2015). Variable-fidelity modeling of structural analysis of assemblies. *Journal of Global Optimization*, A paraître.
- [Dadone et Grossman, 2000] DADONE, A. et GROSSMAN, B. (2000). Progressive optimization of inverse fluid dynamic design problems. *Computer and fluids*, 29:1–32.
- [Dadone et Grossman, 2003] DADONE, A. et GROSSMAN, B. (2003). Fast convergence of inviscid fluid dynamic design problems. *Computer and fluids*, 32:607–627.
- [Davidon, 1959] DAVIDON, W. C. (1959). Variable metric method for minimization. Rapport technique, ANL-5990 Rev., Argonne National Laboratories.
- [Dorigo, 1992] DORIGO, M. (1992). *Optimization, learning and natural algorithms*. Thèse de doctorat, Politecnico di Milano, Italy.
- [Dréo, 2004] DRÉO, J. (2004). *Adaptation de la méthode des colonies de fourmis pour l'optimisation en variables continues. Application en génie biomédical*. Thèse de doctorat, Université Paris XII - Val de Marne.
- [Dréo *et al.*, 2006] DRÉO, J., PÉTROWSKI, A., SIARRY, P. et TAILLARD, E. (2006). *Metaheuristics for Hard Optimization*. Springer-Verlag.
- [Dubourg *et al.*, 2011] DUBOURG, V., SUDRET, B. et BOURINET, J.-M. (2011). Reliability-based design optimization using kriging surrogates and subset simulation. *Structural and Multidisciplinary Optimization*, 44(5):673–690.
- [Duisinx, 1996] DUYSINX, P. (1996). Introduction à l'optimisation topologique. Rapp. tech. of-43. Itas, Université de Liège.
- [Duisinx et Bendsøe, 1998] DUYSINX, P. et BENDSØE, M. P. (1998). Topology optimization of continuum structures with local stress constraints. *International Journal for Numerical Methods in Engineering*, 43:1453–1478.
- [Eberhart *et al.*, 1996] EBERHART, R. C., DOBBINS, R. W. et SIMPSON, P. K. (1996). *Computational Intelligence PC Tools*. Boston : Academic Press Professional.
- [Echard *et al.*, 2011] ECHARD, B., GAYTON, N. et LEMAIRE, M. (2011). AK-MCS : An active learning reliability method combining Kriging and Monte Carlo Simulation. *Structural Safety*, 33(2):145–154.

- [Echard *et al.*, 2013] ECHARD, B., GAYTON, N., LEMAIRE, M. et RELUN, N. (2013). A combined Importance Sampling and Kriging reliability method for small failure probabilities with time-demanding numerical models. *Reliability Engineering & System Safety*, 111:232–240.
- [Engels *et al.*, 2004] ENGELS, H., BECKER, W. et MORRIS, A. (2004). Implementation of a multi-level optimisation methodology within the e-design of a blended wing body. *Aerospace Science and Technology*, 8(2):145–153.
- [Farhat *et al.*, 2000] FARHAT, C., LESOINNE, M. et PIERSON, K. (2000). A scalable dual-primal domain decomposition method. *Numerical Linear Algebra with Applications*, 7(7-8):687–714.
- [Fedorov, 1972] FEDOROV, V. V. (1972). *Theory of Optimal Experiments*. Academic Press.
- [Feo et Resende, 1995] FEO, T. A. et RESENDE, M. G. C. (1995). Greedy Randomized Adaptive Search Procedures. *Journal of Global Optimization*, 6(2):109–133.
- [Fletcher, 1970] FLETCHER, R. (1970). A new approach to variable metric algorithms. *Computer Journal*, 13(3):317–322.
- [Fletcher, 1987] FLETCHER, R. (1987). *Practical methods of optimization*. John Wiley & Sons, 2nd édition.
- [Fletcher et Powell, 1963] FLETCHER, R. et POWELL, M. J. D. (1963). A rapidly convergent descent method for minimization. *Computer Journal*, 6(2):163–168.
- [Fogel, 1964] FOGEL, L. J. (1964). *On the organization of intellect*. Thèse de doctorat, University of California, Los Angeles-Engineering, USA.
- [Forrester, 2004] FORRESTER, A. I. J. (2004). *Efficient global aerodynamic optimisation using expensive computational fluid dynamics simulations*. Thèse de doctorat, University of Southampton, School of Engineering Sciences, Southampton, England.
- [Forrester *et al.*, 2003] FORRESTER, A. I. J., BRESSLOFF, N. W. et KEANE, A. J. (2003). Response surface model evolution. In *16th AIAA Computational Fluid Dynamics Conference*. Orlando, USA. AIAA-2002-4089.
- [Forrester *et al.*, 2006a] FORRESTER, A. I. J., BRESSLOFF, N. W. et KEANE, A. J. (2006a). Optimization using surrogate models and partially converged computational fluid dynamics simulations. *Proceedings of the Royal Society A*, 462:2177–2204.
- [Forrester et Keane, 2009] FORRESTER, A. I. J. et KEANE, A. J. (2009). Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, 45:50–79.
- [Forrester *et al.*, 2006b] FORRESTER, A. I. J., KEANE, A. J. et BRESSLOFF, N. W. (2006b). Design and analysis of “noisy” computer experiments. *AIAA Journal*, 44(10):2331–2339.
- [Forrester *et al.*, 2007] FORRESTER, A. I. J., SOBESTER, A. et KEANE, A. J. (2007). Multi-fidelity optimization via surrogate modelling. *Proceedings of the Royal Society A*, 463:3251–3269.
- [Forrester *et al.*, 2008] FORRESTER, A. I. J., SOBESTER, A. et KEANE, A. J. (2008). *Engineering Design via Surrogate Modelling : A Practical Guide*. John Wiley & Sons, Chichester.

- [Frisch, 1955] FRISCH, K. R. (1955). The logarithmic potential method of convex programming. Rapport technique, Memorandum of May 13.
- [Gano et Renaud, 2004] GANO, S. E. et RENAUD, J. E. (2004). Variable Fidelity Optimization Using a Kriging Based Function. *In Proceedings of 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. Albany, New-York.
- [Gano et al., 2006] GANO, S. E., RENAUD, J. E., MARTIN, J. D. et SIMPSON, T. W. (2006). Update strategies for kriging models used in variable fidelity optimization. *Structural and Multidisciplinary Optimization*, 32(4):287–298.
- [Garcia Palomares et Mangasarian, 1976] GARCIA PALOMARES, U. M. et MANGASARIAN, O. L. (1976). Superlinearly convergent quasi-newton algorithms for nonlinearly constrained optimization problems. *Mathematical Programming*, 11(1):1–13.
- [Gibbs, 1997] GIBBS, M. N. (1997). *Bayesian Gaussian processes for regression and classification*. Thèse de doctorat, University of Cambridge, USA.
- [Giunta et Eldred, 2000] GIUNTA, A. A. et ELDRED, M. R. (2000). Implentation of a trust region model management strategy in the DAKOTA optimization toolkit. *In Proceedings of the 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. Long Beach, USA. AIAA-2000-4935.
- [Giunta et al., 1995] GIUNTA, A. A., NARDUCCI, R., BURGEE, S., GROSSMAN, B., MASON, W. H., WATSON, L. T. et HAFTKA, R. T. (1995). Variable-complexity response surface aerodynamic desing of an hsct wing. *In 13th AIAA Applied Aerodynamics Conference*. pages 994–1002, San Diego, California. AIAA Paper 95- 1886.
- [Glover, 1986] GLOVER, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5):533–549.
- [Glowinski et Le Tallec, 1990] GLOWINSKI, R. et LE TALLEC, P. (1990). Augmented lagrangian interpretation of the nonoverlapping schwarz alternating method. *In Third International symposium on Domain Decomposition Methods for Partial Differential Equations*. pages 224–231. SIAM.
- [Goldberg, 1989] GOLDBERG, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- [Goldfarb, 1970] GOLDFARB, D. (1970). A family of variable metric updates derived by variational means. *Mathematics of Computation*, 24(109):23–26.
- [Goovaerts, 1997] GOOVAERTS, P. (1997). *Geostatistics for natural resources evaluation*. Oxford University Press, USA.
- [Gosselet et Rey, 2006] GOSSELET, P. et REY, C. (2006). Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational Methods in Engineering*, 13(4):515–572.
- [Han et Görtz, 2012] HAN, Z.-H. et GÖRTZ, S. (2012). A hierarchical kriging model for variable-fidelity surrogate modeling of aerodynamic functions. *AIAA Journal*, 50(9):1885–1896.

- [Han *et al.*, 2013] HAN, Z.-H., GÖRTZ, S. et ZIMMERMANN, R. (2013). Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function. *Aerospace Science and technology*, 25(1):177–189.
- [Han *et al.*, 2010] HAN, Z.-H., ZIMMERMANN, R. et GÖRTZ, S. (2010). A new cokriging method for variable-fidelity surrogate modeling of aerodynamic data. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*. Orlando, FL. AIAA 2010-1225.
- [Han *et al.*, 2012] HAN, Z.-H., ZIMMERMANN, R. et GÖRTZ, S. (2012). Alternative cokriging model for variable-fidelity surrogate modeling. *AIAA Journal*, 50(5):1205–1210.
- [Hassan, 2014] HASSAN, M. H. (2014). *Méthodologies de conception optimale de systèmes de conversion électromécanique*. Thèse de doctorat, Université de Paris Sud-Paris XI.
- [Holland, 1975] HOLLAND, J. H. (1975). *Adaptation in natural and artificial systems an introductory analysis with applications to biology, control, and artificial intelligence*. The University of Michigan Press.
- [Huang *et al.*, 2006a] HUANG, D., ALLEN, T. T., NOTZ, W. I. et MILLER, R. A. (2006a). Sequential kriging optimization using multiple-fidelity evaluations. *Structural and Multidisciplinary Optimization*, 32(5):369–382.
- [Huang *et al.*, 2006b] HUANG, D., ALLEN, T. T., NOTZ, W. I. et ZENG, N. (2006b). Global optimization of stochastic black-box via sequential kriging meta-models. *Journal of Global Optimization*, 34(3):441–466.
- [Isaaks et Srivastana, 1989] ISAAKS, E. et SRIVASTANA, R. (1989). *An introduction to applied geostatistics*. Oxford University Press, Oxford University Press, New York.
- [Janusevskis et Le Riche, 2013] JANUSEVSKIS, J. et LE RICHE, R. (2013). Simultaneous kriging-based estimation and optimization of mean response. *Journal of Global Optimization*, 55(2):313–336.
- [Jin *et al.*, 2002a] JIN, R., CHEN, W. et SUDJANTO, A. (2002a). On Sequential Sampling for Global Metamodeling in Engineering Design. In *Proceedings of DETC02 ASME 2002 Design Engineering Technical Conferences And Computers and Information in Engineering Conference*. Montreal, Canada.
- [Jin *et al.*, 2002b] JIN, R., CHEN, W. et SUDJANTO, A. (2002b). On sequential sampling for global metamodeling in engineering design. In *Proceedings of DETC02 ASME 2002 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. pages 1–10, Montreal, Canada.
- [Jones, 2001] JONES, D. R. (2001). A taxonomy of global optimization methods based on response surface. *Journal of Global Optimization*, 21:345–383.
- [Jones *et al.*, 1993] JONES, D. R., PERTTUNEN, C. D. et STUCKMAN, B. E. (1993). Lipschitzian optimization without the lipschitz constant. *Journal of Optimization Theory and Applications*, 79(1):157–181.
- [Jones *et al.*, 1998] JONES, D. R., SCHONLAU, M. et WELCH, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal Global Optimization*, 13:455–492.

- [Kaelbling *et al.*, 1996] KAEHLING, L. P., LITTMAN, M. L. et MOORE, A. W. (1996). Reinforcement learning : A survey. *Journal of Artificial Intelligence Research*, 4:237–285.
- [Karmarkar, 1984] KARMARKAR, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395.
- [Karush, 1939] KARUSH, W. (1939). Minima of functions of several variables with inequalities as side constraints. Mémoire de D.E.A., Dept. of Mathematics, University of Chicago.
- [Keane et Nair, 2005] KEANE, A. J. et NAIR, P. B. (2005). *Computational Approaches for Aerospace Design : The Pursuit of Excellence*. John Wiley & Sons, Chichester.
- [Keane et Petruzzeli, 2000] KEANE, A. J. et PETRUZZELI, N. (2000). Aircraft wing design using ga-based multi-level strategies. In *Proceedings 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. Long Beach, USA. AIAA-2000-4938.
- [Kennedy, 1998] KENNEDY, J. (1998). The behavior of particles. In *Proceedings of the 7th Conference on Evolutionary Computation*, Sous la dir. de : PORTO, V. W., SARAVANAN, N., WAAGEN, D. et EIBEN, A. E. pages 581–589, San Diego, California. Berlin : Springer.
- [Kennedy et Eberhart, 1995] KENNEDY, J. et EBERHART, R. C. (1995). Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks IV*. pages 1942–1948. Piscataway : IEEE.
- [Kennedy et O’Hagan, 2000] KENNEDY, M. C. et O’HAGAN, A. (2000). Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13.
- [Khalil, 2009] KHALIL, H. (2009). *Développement des techniques d’optimisation de forme pour la conception de composants hyperfréquences*. Thèse de doctorat, Université de Limoges, France.
- [Kirkpatrick *et al.*, 1983] KIRKPATRICK, S., GELATT, C. D. et VECCHI, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598):671–680.
- [Kitanidis, 1983] KITANIDIS, P. K. (1983). Statistical estimation of polynomial generalized covariance functions and hydrologic applications. *Water Resources Research*, 19(4):909–921.
- [Koehler et Owen, 1996] KOEHLER, J. et OWEN, A. (1996). Computer experiments. *Handbook stat*, 13:261–308.
- [Kolda *et al.*, 2003] KOLDA, T. G., LEWIS, R. M. et TORCZON, V. (2003). Optimization by direct search : New perspectives on some classical and modern methods. *Siam Review*, 45(3): 385–482.
- [Kravanja *et al.*, 2003] KRAVANJA, S., SORSAK, A. et KRAVANJA, Z. (2003). Efficient multilevel MINLP strategies for solving large combinatorial problems in engineering. *Optimization and engineering*, 4(1/2):97–151.
- [Krige, 1951] KRIGE, D. G. (1951). A statistical approach to some mine valuation and allied problems on the witwatersrand. Mémoire de D.E.A., University of the Witwatersrand.
- [Kuhn et Tucker, 1951] KUHN, H. W. et TUCKER, A. W. (1951). Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, 1950*. pages 481–492, Berkeley and Los Angeles. University of California Press.

- [Ladevèze, 1999] LADEVÈZE, P. (1999). *Nonlinear computational structural mechanics : new approaches and non-incremental methods of calculation*. Springer Verlag, Mechanical Engineering Series.
- [Lancaster et Šalkauskas, 1981] LANCASTER, P. et ŠALKAUSKAS, K. (1981). Surfaces generated by moving least squares methods. *Mathematics of computation*, 37(155):141–158.
- [Land et Doig, 1960] LAND, A. H. et DOIG, A. G. (1960). An automatic method of solving discrete programming problems. *Econometrica*, 28(3):497–520.
- [Laurenceau et Sagaut, 2008] LAURENCEAU, J. et SAGAUT, P. (2008). Building efficient response surfaces of aerodynamic functions with kriging and cokriging. *AIAA Journal*, 46(2):498–507.
- [Laurent, 2013] LAURENT, L. (2013). *Stratégie multiparamétrique et métamodèles pour l'optimisation multiniveaux de structures*. Thèse de doctorat, ENS-Cachan, Cachan, France.
- [Laurent et al., 2013a] LAURENT, L., BOUCARD, P.-A. et SOULIER, B. (2013a). A dedicated multiparametric strategy for the fast construction of a cokriging metamodel. *Computers and Structures*, 124:61–73.
- [Laurent et al., 2013b] LAURENT, L., BOUCARD, P.-A. et SOULIER, B. (2013b). Generation of a cokriging metamodel using a multiparametric strategy. *Computational Mechanics*, 51(2):151–169.
- [Le Gratiot, 2013] LE GRATIOT, L. (2013). *Multi-fidelity Gaussian process regression for computer experiments*. Thèse de doctorat, Université de Paris VII.
- [Levy et Montalvo, 1985] LEVY, A. V. et MONTALVO, A. (1985). The tunneling algorithm for the global minimization of functions. *SIAM Journal on Scientific and Statistical Computing*, 6(1):15–29.
- [Lewis et al., 2000] LEWIS, R. M., TORCZON, V. et TROSSET, M. W. (2000). Direct search methods : then and now. *Journal of Computational and Applied Mathematics*, 124(1-2):191–207.
- [Lin, 1989] LIN, L. I.-K. (1989). A concordance correlation coefficient to evaluate reproducibility. *Biometrics*, 45:255–268.
- [Liu et al., 2004] LIU, B., HAFTKA, R. T. et WATSON, L. T. (2004). Global-local structural optimization using response surfaces of local optimization margins. *Structural and Multidisciplinary Optimization*, 27(5):352–359.
- [Liu, 2003] LIU, W. (2003). *Development of Gradient-Enhanced Kriging Approximations for Multidisciplinary Design Optimization*. Thèse de doctorat, University of Notre Dame, Indiana.
- [Lophaven et al., 2002a] LOPHAVEN, S. N., NIELSEN, H. B. et SØNDERGAARD, J. (2002a). Aspects of the matlab toolbox dace 2002. Rapport technique, Technical report IMM-REP-2002-13.
- [Lophaven et al., 2002b] LOPHAVEN, S. N., NIELSEN, H. B. et SØNDERGAARD, J. (2002b). Dace : a matlab kriging toolbox version 2.0. Rapport technique, Technical report IMM-TR-2002-12.



- [Martin et Simpson, 2002] MARTIN, J. D. et SIMPSON, T. W. (2002). Use of Adaptive Metamodeling for Design Optimization. *In 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Sous la dir. de : AIAA. Atlanta, USA.
- [Matheron, 1963] MATHERON, G. (1963). Principles of geostatistics. *Economic Geology*, 58(8): 1246–1266.
- [Matheron, 1969] MATHERON, G. (1969). Le krigeage universel. *Cahiers du centre de morphologie mathématiques*, 1.
- [Matheron, 1970] MATHERON, G. (1970). La théorie des variables régionalisées et leur estimation. *Les Cahiers du Centre de Morphologie Mathématique de Fontainebleau, école Nationale des Mines de Paris*, Fascicule 5.
- [McKay et al., 1979] MCKAY, M. D., BECKMAN, R. J. et CONOVER, W. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245.
- [Miller et al., 2012] MILLER, R. L., HARDING, L. B., DAVIS, M. J. et GRAY, S. K. (2012). Bi-fidelity fitting and optimization. *The Journal of Chemical Physics*, 136.
- [Morris et Mitchell, 1995] MORRIS, M. D. et MITCHELL, T. J. (1995). Exploratory designs for computational experiments. *Journal of Statistical Planning and Interference*, 43(3):381–402.
- [Morris et al., 1993] MORRIS, M. D., MITCHELL, T. J. et YLVISAKER, D. (1993). Bayesian desing and analysis of computer experiments : use of derivatives in surface prediction. *Technometrics*, 35(3):243–255.
- [Myers, 1982] MYERS, D. E. (1982). Matrix fomulation of co-kriging. *Mathematical Geology*, 14(3).
- [Myšáková et al., 2014] MYŠÁKOVÁ, E., POSPÍŠILOVÁ, A. et LEPŠ, M. (2014). Optimized design of computer experiments : A review. *Computational Methods for Engineering Technology*, pages 325–344.
- [Nelder et Mead, 1965] NELDER, J. A. et MEAD, R. (1965). A simplex method for function minimization. *The Computer Journal*, 7(4):308–313.
- [Néron et al., 2015] NÉRON, D., BOUCARD, P.-A. et RELUN, N. (2015). Time-space pgd for the rapid solution of 3d nonlinear parametrized problems in the many-query context. *International Journal for Numerical Methods in Engineering*, A paraître.
- [Nguyen et Miller, 1992] NGUYEN, N.-K. et MILLER, A. J. (1992). A review of some exchange algorithms for construction discrete D-optimal designs. *Computational Statistics & Data Analysis*, 14:489–498.
- [Nickerson, 1997] NICKERSON, C. A. E. (1997). A note on “a concordance correlation coefficient to evaluate reproducibility”. *Biometrics*, 53:1503–1507.
- [Nocedal et Wright, 2006] NOCEDAL, J. et WRIGHT, S. J. (2006). *Numerical Optimization*. Springer, 2nd édition.
- [Orr, 1996] ORR, M. J. L. (1996). Introduction to radial basis function networks. Rapport technique, University of Edinburg.

- [Ozcan et Mohan, 1999] OZCAN, E. et MOHAN, C. (1999). Particle swarm optimization : surfing the waves. *In Proceeding of the 1999 IEEE Congress on Evolutionary Computation*. pages 1939–1944.
- [Parr et al., 2012] PARR, J. M., FORRESTER, A. I. J. et KEANE, A. J. (2012). Enhancing infill sampling criteria for surrogate-based constrained optimization. *Journal of Computational Methods in Sciences and Engineering*, 12:25–45.
- [Parr et al., 2010] PARR, J. M., HOLDEN, C. M. E., FORRESTER, A. I. J. et KEANE, A. J. (2010). Review of efficient surrogate infill sampling criteria with constraint handling. *In 2nd International Conference on Engineering Optimization*. Lisbon, Portugal.
- [Picheny, 2009] PICHENY, V. (2009). *IMPROVING ACCURACY AND COMPENSATING FOR UNCERTAINTY IN SURROGATE MODELING*. Thèse de doctorat, Ecole Nationale Supérieure des Mines de Saint-Etienne.
- [Picheny et al., 2013] PICHENY, V., GINSBOURGER, D., RICHET, Y. et CAPLIN, G. (2013). Quantile-based optimization of noisy computer experiments with tunable precision. *Technometrics*, 55(1):2–13 <hal-00578550v3>.
- [Picheny et al., 2010] PICHENY, V., GINSBOURGER, D., ROUSTANT, O., HAFTKA, R. T. et KIM, N.-H. (2010). Adaptive Designs of Experiments for Accurate Approximation of Target Regions. *Journal of Mechanical Design*, 132(7):1–9.
- [Powell, 1975] POWELL, M. J. D. (1975). Convergence properties of a class of minimization algorithms. *In Nonlinear Programming 2*, Sous la dir. de : MANGASARIAN, O. L., MEYER, R. R. et ROBINSON, S. M. pages 1–27. Academic Press, New York, NY.
- [Powell, 1998] POWELL, M. J. D. (1998). Direct search algorithms for optimization calculations. *Acta Numerica*, 7:287–336.
- [Queipo et al., 2005] QUEIPO, N. V., HAFTKA, R. T., SHYY, W., GOEL, T., VAIDYANATHAN, R. et TUCKER, P. K. (2005). Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41:1–28.
- [Rasmussen et Williams, 2006] RASMUSSEN, C. E. et WILLIAMS, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press, Adaptive Computation and Machine Learning, Cambridge, MA, USA.
- [Rastrigin, 1963] RASTRIGIN, L. A. (1963). The convergence of the random search method in the extremal control of a many-parameter system. *Automation and Remote Control*, 24(10): 1337–1342.
- [Rechenberg, 1973] RECHENBERG, I. (1973). *Evolutionsstrategie : Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog.
- [Relun, 2011] RELUN, N. (2011). *Approche multiparamétrique pour la conception robuste en fatigue*. Thèse de doctorat, ENS-Cachan, Cachan, France.
- [Rippa, 1999] RIPPA, S. (1999). An algorithm for selecting a good value for the parameter  $c$  in radial basis function interpolation. *Advances in Computational Mathematics*, 11(2):193–210.
- [Robinson et Keane, 1999] ROBINSON, G. M. et KEANE, A. J. (1999). A case for multi-level optimisation in aeronautical design. *Aeronautical Journal*, 103(1028):481–485.

- [Robinson *et al.*, 2008] ROBINSON, T. D., ELDRED, M. R., WILLCOX, K. E. et HAIMES, R. (2008). Surrogate-based optimization using multifidelity models with variable parametrization and corrected space mapping. *AIAA Journal*, 46:2814–2822.
- [Rosenbaum, 2013] ROSENBAUM, B. (2013). *Efficient Global Surrogate Models for Responses of Expensive Simulations*. Thèse de doctorat, Universität Trier.
- [Rosenbaum et Schulz, 2012] ROSENBAUM, B. et SCHULZ, V. (2012). Comparing sampling strategies for aerodynamic kriging surrogate models. *ZAMM - Journal of Applied Mathematics and Mechanics/ Zeitschrift für Angewandte Mathematik und Mechanik*, 92:852–868.
- [Roulet, 2011] ROULET, V. (2011). *Stratégie multiparamétrique pour la simulation d'assemblages de structures stratifiées*. Thèse de doctorat, ENS-Cachan, Cachan, France.
- [Saavedra Redlich, 2012] SAAVEDRA REDLICH, K. (2012). *Stratégie multiéchelle pour l'analyse du couplage flambage-délamination de composites stratifiés*. Thèse de doctorat, ENS-Cachan, Cachan, France.
- [Sacks *et al.*, 1989] SACKS, J., WELCH, W. J., MITCHELL, T. J. et WYNN, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, 4:409–423.
- [Santner *et al.*, 2003] SANTNER, T. J., WILLIAMS, B. J. et NOTZ, W. I. (2003). *The Design and Analysis of Computer Experiments*. Springer.
- [Schonlau, 1997] SCHONLAU, M. (1997). *Computer experiments and global optimization*. Thèse de doctorat, University of Waterloo, Waterloo, Ontario, Canada.
- [Schonlau *et al.*, 1997] SCHONLAU, M., WELCH, W. J. et JONES, D. R. (1997). A data-analytic approach to bayesian global optimization. *Proceedings of the ASA, Section on Physical and Engineering Sciences. Amer. Statist. Assoc.*, pages 186–191.
- [Series *et al.*, 2003] SERIES, L., FEYEL, F. et ROUX, F.-X. (2003). Une méthode de décomposition de domaine avec deux multiplicateurs de lagrange. *In Actes du 16<sup>e</sup> Congrès Français de Mécanique*.
- [Shang et Qiu, 2006] SHANG, Y.-W. et QIU, Y.-H. (2006). A note on the Extended Rosenbrock Function. *Evolutionary Computation*, 14:119–126.
- [Shanno, 1970] SHANNO, D. F. (1970). Conditioning of quasi-newton methods for function minimization. *Mathematics of Computation*, 24(111):647–656.
- [Sigmund, 2001] SIGMUND, O. (2001). Design of multiphysics actuators using topology optimization. *Computer methods in applied mechanics and engineering*, 190(49/50):6577–6604.
- [Silva et Kikuchi, 1999] SILVA, E. C. N. et KIKUCHI, N. (1999). Design of piezoelectric transducers using topology optimization. *Smart Mater. Struct.*, 8:350–364.
- [Simpson *et al.*, 2001a] SIMPSON, T. W., LIN, D. K. J. et CHEN, W. (2001a). Sampling Strategies for Computer Experiments : Design and Analysis. *International Journal of Reliability and Applications*, 2(3):209–240.
- [Simpson *et al.*, 2001b] SIMPSON, T. W., PEPLINSKI, J. D., KOCH, P. N. et ALLEN, J. K. (2001b). Metamodels for Computer-based Engineering Design : Survey and recommendations. *Engineering with Computers*, 17:129–150.

- [Sobester *et al.*, 2004] SOBESTER, A., LEARY, S. J. et KEANE, A. J. (2004). A parallel updating scheme for approximating and optimizing high fidelity computer simulations. *Structural and Multidisciplinary Optimization*, 27:371–383.
- [Sóbester *et al.*, 2005] SÓBESTER, A., LEARY, S. J. et KEANE, A. J. (2005). On the design of optimization strategies based on global response surface approximation models. *Journal of Global Optimization*, 33(1):31–59.
- [Soulier, 2003] SOULIER, B. (2003). Crashworthiness optimization using a surrogate approach by stochastic response surface. *Recent Advances in Integrated Design and Manufacturing in Mechanical Engineering*, pages 159–168.
- [Soulier et Boucard, 2009] SOULIER, B. et BOUCARD, P.-A. (2009). A multiparametric strategy for the large-scale multilevel optimization of structural assemblies. In *8th World Congress on Structural and Multidisciplinary Optimization*. Lisbon, Portugal.
- [Soulier et Boucard, 2013] SOULIER, B. et BOUCARD, P.-A. (2013). A multiparametric strategy for the two step optimization of structural assemblies. *Structural and Multidisciplinary Optimization*, 47(4):539–553.
- [Soulier *et al.*, 2015] SOULIER, B., COURRIER, N., LAURENT, L. et BOUCARD, P.-A. (2015). Métamodèles à gradients et multiniveaux de fidélité pour l’optimisation d’assemblages. In *CSMA2015, 12e Colloque National en Calcul des Structures*.
- [Soulier *et al.*, 2014] SOULIER, B., LAURENT, L. et BOUCARD, P.-A. (2014). Improvement of multiparametric strategy in two-level model optimization of assemblies. In *Proceedings of the 1st International Conference on Engineering and Applied Sciences Optimization, OPT-i*. Kos Island, Grèce.
- [Torczon, 1989] TORCZON, V. (1989). *Multi-Directional Search : A Direct Search Algorithm for Parallel Machines*. Thèse de doctorat, Rice University, Houston, Texas, USA.
- [Ulaganathan *et al.*, 2014] ULAGANATHAN, S., COUCKUYT, I., FERRANTI, F., LAERMANS, E. et DHAENE, T. (2014). Performance study of multi-fidelity gradient enhanced kriging. *Structural and Multidisciplinary Optimization*.
- [Vauchez, 1991] VAUCHEZ, P. (1991). *Simulation numérique des processus de mise en forme par la méthode à grands incrément de temps*. Thèse de doctorat, Université Pierre et Marie Curie, Paris, France.
- [Vitali *et al.*, 2002] VITALI, R., HAFTKA, R. T. et SANKAR, B. V. (2002). Multi-fidelity design of stiffened composite panel with a crack. *Structural and Multidisciplinary Optimization*, 23(5):347–356.
- [Wachernagel, 2003] WACHERNAGEL, P. (2003). *Multivariate geostatistics : an introduction with applications*. Springer, 3rd édition.
- [Wright, 1996] WRIGHT, M. H. (1996). Direct search methods : Once scorned, now respectable. In *Numerical Analysis : Proceedings of the 1995 Dundee Biennial Conference in Numerical Analysis*. Sous la dir. de : GRIFFITHS, D. et WATSON, G. pages 191–208. Addison Wesley Logman (Harlow), U.K.
- [Wright, 1997] WRIGHT, S. J. (1997). *Primal-dual interior-point methods*. volume 54. SIAM.

- [Yamazaki et Mavriplis, 2011] YAMAZAKI, W. et MAVRIPLIS, D. J. (2011). Derivative-enhanced variable fidelity surrogate modeling for aerodynamic functions. *In Proceedings of 49th AIAA Aerospace Sciences Meeting and Exhibit*. Numéro 1172, Orlando, USA.
- [Yamazaki et Mavriplis, 2013] YAMAZAKI, W. et MAVRIPLIS, D. J. (2013). Derivative-enhanced variable fidelity surrogate modeling for aerodynamic functions. *AIAA Journal*, 51(1):126–137.
- [Zimmermann, 2013] ZIMMERMANN, R. (2013). Gradient-enhanced surrogate modeling based on proper orthogonal decomposition. *Journal of Computational and Mathematics*, 237:403–418.
- [Zimmermann et Han, 2010] ZIMMERMANN, R. et HAN, Z.-H. (2010). Simplified cross-correlation estimation for multi-fidelity surrogate cokriging models. *Advances and Applications in Mathematical Sciences*, 7(2):181–201.

**Titre :** Couplage optimisation à convergence partielle et stratégie multiparamétrique en calcul de structures

**Mots clés :** Cokrigage, Multi-fidélité, Données partiellement convergées

**Résumé :** Dans le cadre du calcul des assemblages de structures, les bureaux d'études sont à l'heure actuelle encore limités dans la possibilité de mener des travaux d'optimisation. En effet, le traitement numérique des assemblages nécessite la mise en œuvre de méthodes capables de prendre en compte différents types de non-linéarités (frottement, contact...). Le coût de calcul associé à ces méthodes est généralement trop important pour mener une optimisation globale nécessitant un trop grand nombre d'évaluations. Afin de pallier à ce problème, ce travail s'appuie sur une démarche d'optimisation à deux niveaux de modèles. Le premier niveau d'optimisation consiste à la création d'un métamodèle sur lequel est effectuée une optimisation globale. Le second niveau d'optimisation consiste alors à mener à bien une optimisation locale sur le modèle mécanique réel. Cette optimisation locale s'appuie sur les résultats obtenus au premier niveau. Deux outils sont principalement utilisés au cours de cette thèse.

Tout d'abord les simulations numériques sont réalisées à l'aide de la méthode LaTIn multiparamétrique qui assure la réduction des temps de calcul associés aux multiples résolutions du problème mécanique. L'autre outil plus largement développé au cours de ce travail s'appuie sur la construction de métamodèles multi-fidélité. En effet, la méthode LaTIn est une méthode de calcul itérative, il est alors possible d'avoir accès à un indicateur d'erreur servant de niveau de convergence pour les différents calculs numériques effectués. La construction de métamodèles multi-fidélité a pour particularité de pouvoir incorporer différentes sources d'informations qui sont dans ce travail dites "totalement convergé" lorsqu'un calcul est effectué à convergence et "partiellement convergé" lorsqu'un calcul est arrêté avant convergence. Différentes méthodes multi-fidélité sont testées sur plusieurs exemples mécaniques afin de déterminer les plus performantes. Deux cas industriels sont également traités.

**Title :** Couplage optimisation à convergence partielle et stratégie multiparamétrique en calcul de structures

**Keywords :** Cokriging, Multifidelity, Partially converged data

**Abstract :** Optimisation strategies on assembly design are often time expensive on industrial case. The main difficulties are due to the non-linearities of the calculation (contact, friction...). The computation cost can be too expensive to perform a global optimization with a large number of evaluations of the mechanical problem. In order to achieve this kind of optimization problems, this work uses a two-levels models optimization strategy. The first level is defined thanks to the construction of a metamodel which is used to perform a global optimization. On the second level, a local optimization is used on the real mechanical model thanks to the results obtained from the first level. Two main tools are used in this work. The first one is the multiparametric LaTIn method which enables

to reduce drastically the computational time for solving several similar mechanical assembly design problems.

The other tool is the one which is the most developed in this work is the construction of multi-fidelity surrogate models

Indeed, the LaTIn method in an iterative method, so it is possible to define an error indicator which can be used as a level of convergence. The construction of multi-fidelity metamodels has for particularity to incorporate several kinds of information which are named as "fully converged" if the calculation has been converged and "partially converged" if the calculation has been stopped prematurely. Different multi-fidelity methods have been investigated in this work on several mechanical examples in the aim to define the most performant. Industrial case test are also treated.