

UNIVERSITÉ DE MONTRÉAL

OPTIMISATION DES ASSEMBLAGES PAR INTERFÉRENCE SOUMIS À DES  
SOLLICITATIONS DE FATIGUE

GUILLAUME BIRON

DÉPARTEMENT DE GÉNIE MÉCANIQUE  
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION  
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES  
(GÉNIE MÉCANIQUE)

AOÛT 2011

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

OPTIMISATION DES ASSEMBLAGES PAR INTERFÉRENCE SOUMIS À DES  
SOLLICITATIONS DE FATIGUE

présenté par : BIRON Guillaume

en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de :

M. TRÉPANIÉR Jean-Yves, Ph. D, président

M. VADEAN Aurelian, Doct., membre et directeur de recherche

M. AUDET Charles, Ph. D., membre

## DÉDICACE

*À mes parents, mes frères et Maude*

## REMERCIEMENTS

Je souhaite à remercier mon directeur de recherche Aurelian Vadean pour sa patience, sa confiance, son professionnalisme et son support.

Je remercie également M. Bernard Sanschagrin pour son support et sa gentillesse.

Je remercie M. Charles Audet et M. Jean-Yves Trépanier, professeurs à l'École Polytechnique de Montréal pour avoir accepté de faire partie du jury d'évaluation de ce mémoire.

Je remercie également les gens d'Altair Engineering Canada pour leurs précieux conseils sur les différents modules utilisés. Je suis également reconnaissant envers Frédéric Lanoue pour son support sur l'utilisation du logiciel ANSYS.

Je ne peux oublier de remercier mes collègues de travail et ami, notamment Lionel Michel, Marc Ouellet, Frédéric Lanoue, Amir, Francois Landry, Benoit Malouin, Pierre Legay. Nos différents échanges ont rendu cette expérience très enrichissante aussi bien d'un point de vue professionnel que personnel.

Enfin, un merci spécial à mes parents qui ont toujours cru en moi. À Maude, qui depuis le début de l'aventure, me supporte sans cesse dans cette démarche.

A tous ceux qui ont contribué de près ou de loin à ce projet, merci.

## RÉSUMÉ

Ce mémoire présente une méthode d'optimisation des assemblages par interférence soumis à des sollicitations de fatigue. Le phénomène de fatigue par fretting réduit la vie en fatigue des assemblages par interférences à l'interface de contact. Il existe toutefois des configurations géométriques qui atténuent ce phénomène. De plus, des résultats expérimentaux montrent que l'ajout d'une rainure circulaire au bord du contact permet de diminuer considérablement l'effet de la fatigue par fretting. Par contre, cet ajout introduit une importante concentration de contrainte dans l'arbre et peut provoquer la rupture par fatigue normale. Les essais expérimentaux montrent qu'il existe une zone de transition dans la localisation de la rupture à l'interface de contact et dans la rainure. Toutefois, ils démontrent également qu'il existe un compromis de paramètres, favorable à la vie en fatigue de l'assemblage.

Une méthodologie de calcul par éléments finis des assemblages par interférences sollicités en fatigue est présentée dans la revue de littérature. En comparant les résultats obtenus par la méthode des éléments finis et les essais expérimentaux, il a été possible d'établir l'existence d'un facteur de réduction de vie en fatigue causée par le fretting ( $k_{\text{fret}}$ ). Bien qu'il soit possible de simuler avec précisions la vie en fatigue de l'assemblage, il n'existe pas de méthode qui permet d'obtenir directement la valeur des paramètres selon des critères de conception spécifiques. La plastification du matériau et la présence d'éléments de contact nécessitent une modélisation par éléments finis non linéaires. Par conséquent, son temps de calcul est considérablement élevé. Le couplage entre le modèle d'éléments finis et un algorithme d'optimisation devient irréaliste compte tenu du nombre de simulations nécessaires. La méthode d'optimisation doit donc être conçue en utilisant un nombre restreint de simulations.

Afin de répondre à cette problématique, un algorithme d'optimisation séquentielle multi-objectifs (OSMO) a été développé. Cette méthode consiste à optimiser une fonction d'interpolation d'un plan d'expériences. La totalité de l'algorithme a été réalisée à partir des modules *Hypermath*, *Hyperstudy* et *Sampling* du logiciel Altair Hyperworks V10. La méthode du Krigeage a été utilisée pour interpoler le plan d'expériences. Le plan utilisé est le Latin Hypercube Optimal.

Ensuite, un algorithme génétique d'optimisation multi-objectifs est utilisé pour générer le Front de Pareto du problème.

Malgré qu'il a été développé pour répondre au problème d'optimisation des assemblages par interférence, l'algorithme peut résoudre efficacement des problèmes structurels. Ainsi, il a été appliqué à un problème académique connu de poutre à 10 membrures. Son temps de calcul négligeable par rapport aux éléments finis permet d'explorer davantage la souplesse de l'algorithme. Les résultats démontrent qu'il permet de résoudre des problèmes structurels avec précision. De plus, ses performances ont été démontrées par rapport à un algorithme d'optimisation séquentielle existante.

Une application de l'algorithme OSMO sur une configuration d'assemblage par interférence étudiée dans la littérature est présentée dans le cadre d'un article scientifique soumis à la revue *Multi-disciplinary and structural optimization*. Le problème d'optimisation consiste à minimiser la masse de l'assemblage soumis à un chargement de torsion alternée tout en s'assurant d'une vie infinie en fatigue.

Une étude *ANalysis Of Variance* (ANOVA) a tout d'abord permis d'effectuer une analyse de sensibilité des paramètres sur la vie en fatigue de l'assemblage. Ensuite, l'utilisation d'une approche multi-objectifs au problème de fretage a généré un Front de Pareto. Une analyse des solutions optimales obtenues a montré que pour une très faible influence sur la masse, il est possible d'obtenir un facteur de sécurité supérieur afin de pallier à d'autres incertitudes de calcul et/ou modélisation. Avec les connaissances supplémentaires acquises, il est donc possible pour l'ingénieur de faire une sélection adaptée de la solution en fonction d'autres critères de conception.

## ABSTRACT

This thesis presents a methodology for designing interference fit assemblies subjected to fatigue load. Due to the assembly, a fretting-fatigue phenomenon was found responsible of reduction of the fatigue life inside the contact. However, there are geometric configurations that contribute to reduce its impact on the life of the assembly. Experimental tests have shown that the addition of a fillet, circular groove or hub overhang improves the fatigue life of interference fit assemblies. It was also determinate that in contact, failure is caused by fretting fatigue while it is caused by normal fatigue in the fillet or groove. As a result of the competing maximum equivalent stresses in contact and in the fillet, a shift in the location of fatigue failure has been observed in both experimental tests and finite elements simulation.

A methodology for fatigue strength calculation of interference fit assemblies subjected to fatigue load is presented. A comparison between calculations by finite element analysis and experimental tests has show the existence of fatigue strength reduction factor ( $k_{fret}$ ). While it's possible to calculate the fatigue strength with precision, there is actually no method to directly obtain the parameters values according to design criteria. Plastic deformation and contact element needs non-linear calculations of finite element model. Therefore, obtaining the optimal solution with a classic procedure may be time consuming, hence inefficient.

In order to address this problem, a Sequential Approximate Multi-Objective Optimization (SAMOO) algorithm was developed in this work. This method consists of optimizing an interpolation function from design of experiments. Entire algorithm was programmed from modules *Hypermath*, *HyperStudy Sampling* of *Altair HyperWorks VII* suite. The kriging method was used to interpolate the *Optimal Latin Hypercube* design of experiments. The multi-objective objective genetic algorithm (MOGA) from *Hyperstudy* is used to generate the Pareto Front of the problem.

Although it was developed to address the problem of optimizing the interference fit joint, the algorithm can effectively solve structural problems. Thus, it was applied to a well known

academic problem of a 10-bar truss. Its computation time is negligible compared to finite element model and therefore allows more flexibility to validate the algorithm. The results of this application demonstrate that the SAMMO algorithm can solve efficiently structural problem with fewer calculations. Moreover, its performances have been compared to an existing algorithm.

An application of the SAMOO on a geometric configuration of interference fits assembly studied in the literature has been presented as a part of a scientific paper submitted to *Structural and Multidisciplinary optimization*. The optimization problem consisted in finding a tradeoff between mass and competing safety factors in two zones subjected to fatigue failures. The application of SAMOO algorithm on the interference fit problem produces good results. An ANOVA has generated a sensitivity study applied on the design of experiments. Moreover, optimizing the interference fit assembly with the multi-objective approach adds valuable information concerning the design specific behaviour around the optimum zone. Finally, it provides further arguments to help choosing an optimal design based on different criteria.



## TABLE DES MATIÈRES

DÉDICACE.....	III
REMERCIEMENTS .....	IV
RÉSUMÉ.....	V
ABSTRACT .....	VII
TABLE DES MATIÈRES .....	IX
LISTE DES TABLEAUX.....	XIV
LISTE DES FIGURES.....	XV
LISTE DES SIGLES ET ABRÉVIATIONS .....	XVIII
LISTE DES ANNEXES.....	XXI
INTRODUCTION.....	1
CHAPITRE 1 REVUE DE LITTÉRATURE SUR LES ASSEMBLAGES PAR INTERFÉRENCE 3	
1.1 Calculs de base .....	4
1.1.1 Paramètre de l'assemblage .....	4
1.1.2 Calcul de la pression de contact .....	5
1.1.3 Force axiale transmissible .....	6
1.1.4 Couple transmissible .....	7
1.1.5 Dilatation thermique des pièces .....	7
1.2 Configurations géométriques .....	7
1.2.1 Essais avec chargement en torsion alternée .....	7
1.2.2 Essais avec chargement en flexion alternée .....	10
1.2.3 Étude de sensibilité à partir de calcul par éléments finis .....	13
1.3 Modélisation par éléments finis de l'assemblage.....	14

1.3.1	Algorithmes de contacts .....	15
1.3.2	Symétrie cyclique .....	16
1.3.3	Sous-modélisation .....	17
1.4	Endommagement par fatigue .....	18
1.4.1	Critère de fatigue .....	18
1.4.2	Endommagement par fretting .....	20
1.5	Calcul de vie en fatigue par la méthode des éléments finis .....	21
1.6	Méthode de conception des assemblages par interférences présentée dans la littérature 22	
1.6.1	Méthode de conception basée sur les équations analytiques.....	22
1.6.2	Méthode de conception basée sur les éléments finis.....	23
CHAPITRE 2 ÉTAT ACTUEL DES CONNAISSANCES SUR L'OPTIMISATION SÉQUENTIELLE MULTI-OBJECTIFS .....		25
2.1	<i>Design and Analysis of Computer Experiments (DACE)</i> .....	25
2.2	Structure d'un algorithme d'optimisation séquentielle .....	26
2.3	Zone d'intérêt .....	28
2.4	Plans d'expériences .....	29
2.4.1	Plans de surface de réponses (SR).....	29
2.4.2	Plan uniforme ( <i>Space Filling Design</i> ).....	31
2.5	Modèle d'interpolation .....	33
2.5.1	Mesure de l'erreur .....	34
2.6	Amélioration séquentielle .....	34
2.6.1	Stratégie de déplacement des limites ( <i>Move Limit strategy - MLS</i> ).....	35
2.6.2	Méthode de remplissage ( <i>Space filling - SF</i> ) .....	35

CHAPITRE 3	ALGORITHME	D’OPTIMISATION	SÉQUENTIELLE	MULTI-	
OBJECTIFS (OSMO)	.....				37
3.1	Algorithme OSMO	.....			37
3.2	Schéma général de l’algorithme	.....			37
3.3	Paramétrisation du modèle et détermination des bornes	.....			40
3.4	Définition de la zone d’intérêt	.....			41
3.5	Plan d’expériences	.....			41
3.6	Modèle d’interpolation	.....			42
3.6.1	Méthode de krigeage dans <i>DACE Toolbox</i>	.....			42
3.6.2	Algorithme pour le calcul de krigeage avec <i>HyperMath</i>	.....			46
3.7	Optimisation du paramètre de corrélation $\theta$ par la méthode de validation croisée	.....			46
3.7.1	Méthode de validation croisée	.....			47
3.7.2	Programmation de la méthode de validation croisée	.....			48
3.7.3	Minimisation de l’erreur obtenue par validation croisée	.....			49
3.8	Optimisation Multi-objectifs	.....			49
3.9	Amélioration séquentielle	.....			50
3.9.1	Critère d’arrêt l’algorithme	.....			51
3.9.2	Méthode statistique de modification des limites	.....			51
3.9.3	Définition de la boîte à moustache	.....			52
3.9.4	Modification des limites (phase exploratoire)	.....			53
3.9.5	Modification des limites (phase recherche)	.....			54
3.9.6	Méthode de remplissage	.....			55
3.10	Programmation de l’algorithme	.....			58
CHAPITRE 4	APPLICATION	DE	L’ALGORITHME	OSMO	À
STRUCTUREL	THÉORIQUE	.....			60

4.1	Définition du problème .....	61
4.2	Phase exploratoire .....	62
4.2.1	Modification des limites.....	63
4.3	Phase recherche : optimisation mono-objective.....	65
4.3.1	Convergence de l'algorithme .....	66
4.3.2	Résultats de l'algorithme.....	67
4.3.3	Comparaison de la solution.....	68
4.4	Phase Recherche : Optimisation Multi-Objectifs.....	69
4.4.1	Amélioration séquentielle .....	69
4.4.2	Convergence de l'algorithme .....	70
4.4.3	Vérifications des contraintes .....	71
4.4.4	Résultat de l'algorithme OSMO.....	72
4.5	Limites de l'algorithme OSMO observées.....	73
4.5.1	<i>Sampling Module</i> .....	73
4.5.2	Performance de l'algorithme MOGA.....	73
4.6	Conclusion.....	74
CHAPITRE 5      ARTICLE : OPTIMAL DESIGN OF INTERFERENCE FIT ASSEMBLIES SUBJECTED TO FATIGUE LOADS .....		76
5.1	Présentation de l'article.....	76
5.2	Abstract .....	77
5.3	Introduction .....	78
5.4	Interference fit assembly .....	80
5.4.1	Parametrized model.....	80
5.4.2	Finite element model.....	81
5.4.3	Selection of fatigue criterion.....	83

5.4.4	Typical results of fatigue strength calculations.....	84
5.4.5	Problem definition.....	85
5.4.6	Constraint equations (CE) related to the interference fit problem .....	86
5.5	Sequential approximate multi-objective optimization (SAMOO) .....	87
5.5.1	Zone of interest.....	89
5.5.2	Design of Experiments .....	89
5.5.3	Kriging model .....	90
5.5.4	Error minimization using cross-validation .....	91
5.5.5	Multi-objective optimization algorithm .....	92
5.5.6	Sequential improvement.....	93
5.6	Results of SAMOO on interference fit assembly optimization problem .....	93
5.6.1	ANOVA study.....	93
5.6.2	Accuracy of SAMOO algorithm .....	94
5.6.3	Choosing the optimal solution.....	95
5.6.4	Discussion .....	98
5.6.5	Conclusion.....	98
5.6.6	Acknowledgements .....	99
	References .....	100
CHAPITRE 6	DISCUSSION GÉNÉRALE .....	102
CHAPITRE 7	CONCLUSION .....	104
	BIBLIOGRAPHIE .....	107
	ANNEXES .....	111

## LISTE DES TABLEAUX

Tableau 1.1 Définition des paramètres.....	5
Tableau 1.2 Nomenclature des termes utilisés pour les calculs de fatigue .....	19
Tableau 2.1 Recommandation pour l'utilisation des différents modèles d'interpolation [26].....	33
Tableau 3.1 Nombre de points d'expériences en fonction du nombre de paramètres. ....	42
Tableau 3.2 Paramètres de krigeage.....	43
Tableau 3.3 Options du solveur MOGA. ....	50
Tableau 3.4 Information sur le modèle d'interpolation .....	51
Tableau 4.1 Propriétés de la charpente à 10 barreaux.....	61
Tableau 4.2 Bornes de la zone d'intérêt pour l'itération 1 .....	64
Tableau 4.3 Erreur d'interpolation sur les contraintes .....	66
Tableau 4.4 Comparaison des résultats .....	69
Tableau 4.5 Temps de calcul (en minute) .....	75
Tableau 5.1 Definition of geometrics parameter.....	81
Tableau 5.2 Mechanical properties .....	81
Tableau 5.3 Contact definition.....	83
Tableau 5.4 Design space.....	87
Tableau 5.5 Optimal solution illustrated in Fig. 5-11 .....	97
Tableau 5.6 Parameter value of solution 2 .....	98
Tableau A.1 Définition des paramètres.....	113
Tableau A.2 Définition de la zone d'intérêt.....	116

## LISTE DES FIGURES

Figure 1-1 Assemblage par interférence, .....	3
Figure 1-2 Assemblage par interférence utilisé pour des essais expérimentaux .....	4
Figure 1-3 : Moyeu assemblé avec arbre simple.....	5
Figure 1-4 Géométries des essais de Hattori [4] .....	8
Figure 1-5 Configuration géométrique des essais de Juuma [7] .....	9
Figure 1-6 Configurations de Nishioka. [8] .....	11
Figure 1-7 Comparaison entre un modèle d'assemblage conventionnel et celui amélioré. [9].....	11
Figure 1-8 Modèle d'essai de Kubota [10] .....	12
Figure 1-9 : Configurations [11] .....	13
Figure 1-10 Maillage global [12] .....	15
Figure 1-11 Symmétrie cyclique .....	17
Figure 1-12 Sous-modèle du modèle global présenté à la Figure 1-10 [12].....	18
Figure 1-13 Amplitude de glissement, le taux d'usure et la vie en fatigue[15].....	21
Figure 1-14 Réduction de vie en fatigue [2] .....	22
Figure 2-1 Algorithme de DACE [18] .....	26
Figure 2-2 Algorithme typique d'optimisation séquentielle [19] .....	27
Figure 2-3 Zone d'intérêt [20] .....	28
Figure 2-4 Central Composite Design [22] .....	29
Figure 2-5 Influence des paramètres à la suite d'une modélisation de surface de réponse quadratique [23] .....	31
Figure 2-6 Comparaison de la couverture des plans d'expériences de type SR (gauche) et uniforme (droite) .....	32
Figure 2-7 Plan d'expériences avec 2 facteurs et 9 points d'expériences : 3 modes orthogonaux (gauche), Latin Hypercube aléatoire (centre) et Latin HyperCuble optimal (droite) [31].....	32

Figure 2-8 Minimisation d'une fonction de mérite.....	36
Figure 3-1 Schéma général de l'algorithme OSMO .....	39
Figure 3-2 Création d'une paramétrisation appropriée pour un algorithme d'optimisation de design structurel.....	40
Figure 3-3 Paramétrisation d'un assemblage par interférence : Bonne paramétrisation (gauche), Modèle disfonctionnel (droite).....	41
Figure 3-4 Comparaison entre $\theta$ initial et celui obtenu par la validation croisée.....	47
Figure 3-5 Programme d'évaluation de l'erreur <i>evaluation_errreur()</i> .....	48
Figure 3-6 Boite à moustache.....	52
Figure 3-7 Vérification d'une contrainte des variables X1 et X2 .....	54
Figure 3-8 Analyse de dispersions des résultats d'optimisation .....	55
Figure 3-9 Fonction de mérite présenté par Bonte [32] .....	57
Figure 3-10 Méthode de minimisation de fonction de mérite .....	58
Figure 3-11 Présentation des différents logiciels utilisés pour l'algorithme OSMO .....	59
Figure 4-1 Structure à 10 barreaux [40] .....	60
Figure 4-2 Contraintes dans les barreaux avec le plan d'expériences initial .....	63
Figure 4-3 Contraintes dans les barreaux à l'itération 1 .....	65
Figure 4-4 Erreur d'interpolation sur les contraintes .....	66
Figure 4-5 Convergence de la valeur des paramètres $X_i$ .....	67
Figure 4-6 Convergence de la masse de la structure .....	68
Figure 4-7 Erreur sur l'optimisation multi-objective.....	70
Figure 4-8 Conformité des contraintes.....	72
Figure 4-9 Résultats de l'optimisation Multi-Objective .....	72
Figure 4-10 Comparaison des résultats de l'algorithme MOGA et GMMO.....	74
Figure 5-1 Juuma's parameterized model.....	80



Figure 5-2 Finite element model with boundary conditions .....	82
Figure 5-3 Distribution of $VM$ in the critical zone .....	85
Figure 5-4 Structure of SAMOO algorithm .....	88
Figure 5-5 Design space and zone of interest .....	89
Figure 5-6 Random Latin Hypercube (left) and Optimal Latin HyperCube (right) designs.....	90
Figure 5-7 Improvement on a random function using Cross-Validation .....	91
Figure 5-8 Influence of each parameters on $VM_{contact}$ and $VM_{fillet}$ .....	94
Figure 5-9 Results of SAMOO on a stress minimization in contact and in the fillet.....	95
Figure 5-10 Isometric view of the Pareto front .....	96
Figure 5-11 Level curves of the Pareto front .....	96
Figure A-1 Structure à 10 barreaux [41] .....	113
Figure A-2 Dispersion des valeurs des paramètres pour la contrainte $\sigma_1$ .....	117
Figure A-3 Dispersion des valeurs des paramètres pour la contrainte $\sigma_2$ .....	117
Figure A-4 Dispersion des valeurs des paramètres pour la contrainte $\sigma_3$ .....	118
Figure A-5 Dispersion des valeurs des paramètres pour la contrainte $\sigma_4$ .....	118
Figure A-6 Dispersion des valeurs des paramètres pour la contrainte $\sigma_5$ .....	119
Figure A-7 Dispersion des valeurs des paramètres pour la contrainte $\sigma_6$ .....	119
Figure A-8 Dispersion des valeurs des paramètres pour la contrainte $\sigma_7$ .....	120
Figure A-9 Dispersion des valeurs des paramètres pour la contrainte $\sigma_8$ .....	120
Figure A-10 Dispersion des valeurs des paramètres pour la contrainte $\sigma_9$ .....	121
Figure A-11 Dispersion des valeurs des paramètres pour la contrainte $\sigma_{10}$ .....	121

## LISTE DES SIGLES ET ABRÉVIATIONS

CCD	Plan d'expériences central composite
DACE	Design and Analysis of Computer Experiments
dL	Longueur de porte-à-faux
E	Module de Young
$E_m, E_a$	Module de Young du Matériau de du moyeu et de l'arbre
$\epsilon$	Erreur d'approximation
F	Force axiale transmissible par l'assemblage
FEA	Analyse d'éléments finis ( <i>Finite Element Analysis</i> )
$FS_{VM}$	Facteur de sécurité selon le critère de Von Mises
GA	Algorithme génétique
GMMO	Algorithme de gradient multiobjectif
IMSE	Erreur des moindres carrés intégrés
$k_{fret}$	Facteur de réduction de vie en fatigue dans le fretage
L	Longueur de l'emmanchement
$L_a$	Longueur de l'assemblage
LHC	Plan d'expériences <i>Latin Hypercube</i>
$m$	nombre de points d'expériences
$n$	Nombre de paramètre d'entrée
MLS	Stratégie de déplacement des limites ( <i>Moving Limit Strategy</i> )
MSE	Erreur des moindres carrés
MOGA	Algorithme génétique multi-objectifs
OLHC	Plan d'expériences <i>Optimal Latin Hypercube</i>
OSMO	Optimisation séquentielle multi-objectifs

$P_i, P_e$	Pression interne et externe exercée sur un cylindre
$r_c$	Rayon nominal à l'interface de contact entre le moyeu et l'arbre
$r_e$	Rayon extérieur d'un cylindre
$r_s$	Rayon mineur de l'arbre
$r_i$	Rayon intérieur de l'arbre
$r_o$	Rayon extérieur de l'arbre
$r_f$	Rayon du congé
$RMSE_{VC}$	Erreur des moindres carrés selon la méthode de validation croisée
S	Matrice du plan d'expériences
SAMOO	Sequential Approximate Multi-Objective Optimization
$S_e$	Limite d'endurance
$S_L$	Limite de fatigue
$S_{ut}$	Limite de résistance
$S_Y$	Limite d'élasticité
SF	Méthode de remplissage ( <i>space fillin</i> )
$SF_{contact}$	Facteur de sécurité dans le contact ( <i>Safety factor</i> )
$SF_{congé}$	Facteur de sécurité dans le congé ( <i>Safety factor</i> )
T	Torque transmissible par l'assemblage
VM	Contrainte alternée équivalente selon le critère de Von Mises
$X_i$	Paramètre (facteur) d'un plan d'expériences
$Y_i$	Variable de réponse d'un plan d'expériences
$\hat{y}$	Réponse de la fonction d'interpolation
$\alpha$	Coefficient de dilatation d'un matériau
$\delta_r$	Interférence radiale entre l'arbre et le moyeu

$\Delta T$	Différence de température
$\rho$	Masse Volumique
$\sigma_a, \sigma_m$	Contrainte alternée et moyenne selon le critère de Von Mises
$\sigma_a^*$	Contrainte alternée équivalente selon le critère de Von Mises
$\sigma_r, \sigma_\theta, \sigma_z,$ $\tau_{r\theta}, \tau_{\theta z}, \tau_{rz}$	6 composantes du tenseur de contrainte dans un repère cylindrique
$\theta$	Paramètre de corrélation
$\mu$	Coefficient de Friction
$\nu$	Coefficient de Poisson

## LISTE DES ANNEXES

ANNEXE 1 – Méthode de sous-modélisation .....	111
ANNEXE 2 – Structure à 10 barreaux .....	114
ANNEXE 3 – Sélection de la zone d'intérêt.....	116

## INTRODUCTION

Avec la compétition active dans les marchés, les entreprises sont toujours à la recherche de produits optimaux. Elles accordent également beaucoup d'importance au temps de mise en marché d'un produit afin de profiter de l'avantage concurrentiel. Pour arriver à cette fin, il devient nécessaire d'avoir des méthodes de conception efficaces qui assurent la qualité, la fiabilité et l'optimalité d'un produit. L'évolution des logiciels de CAD et d'éléments finis permet à l'ingénieur de simuler plusieurs phénomènes mécaniques avec précision. Par contre, pour certains, la résolution des calculs numériques est longue et très complexe. Dans ces cas, la recherche de la solution optimale devient ardue et, par conséquent, beaucoup plus coûteuse en terme de ressources. De plus, les critères de conception et d'optimalité sont toujours de plus en plus sévères et rendent la prise de décision difficile pour le concepteur. L'ingénieur est donc souvent confronté à un choix parmi plusieurs objectifs, ce qui rend la sélection d'une solution optimale difficile. La conception d'un assemblage par interférence n'échappe pas à ces difficultés typiques.

La modélisation du comportement des assemblages par interférence par la méthode des éléments finis est connue. La présence du contact, du frettage et de la plastification dans l'assemblage complexifie sa modélisation numérique et par conséquent, augmente considérablement le temps de calcul. L'utilisation d'une méthode d'optimisation est donc préférable pour effectuer une sélection des paramètres optimaux. Étant donné le temps résolution du modèle par éléments finis, cette méthode doit être en mesure de limiter le nombre de calculs structurels. Ainsi, le principal objectif de ce travail consiste à élaborer une méthode de conception optimale des assemblages par interférences.

Le premier chapitre présente une vue d'ensemble sur les assemblages par interférence sollicités en fatigue. Une revue des différentes configurations géométriques des assemblages étudiées dans les divers essais expérimentaux portant sur la fatigue. Des informations pertinentes à la modélisation par éléments finis et la sélection d'un critère de fatigue sont détaillées. Les différentes méthodes de conception des assemblages par interférence proposées dans la littérature y sont également présentées.

Le deuxième chapitre présente l'état actuel des connaissances sur l'optimisation séquentielle. Des informations sur l'historique et les différentes méthodes utilisées sont présentées. Les différents plans d'expériences, modèle d'interpolation et amélioration séquentielle pertinents y sont également décrits.

Le troisième chapitre porte sur l'élaboration de l'algorithme d'Optimisation Séquentielle Multi-Objectifs (OSMO). Chaque étape et ses raisonnements associés y sont présentés afin de bien comprendre son fonctionnement. On y trouve également des détails sur sa programmation avec le logiciel Altair Hyperworks V10.

Le troisième chapitre porte sur l'application de l'algorithme OSMO sur un problème académique connu. Ce dernier consiste à minimiser la masse et le déplacement d'une structure à 10 barreaux sollicitée en porte-à-faux. Puisque les équations analytiques des contraintes et du comportement de la structure sont connues, le temps de calcul est négligeable. Ainsi, il a été possible de valider l'efficacité de l'algorithme à résoudre des problèmes structurels. Une comparaison des résultats avec un algorithme existant est également présentée.

Le dernier chapitre présente l'article s'intitulant : « *Optimal design of interference fits assembly using a Multi-Objective Sequential Approximate Optimization approach* ». Il a été soumis à la revue « *Multidisciplinary and structural optimization, ISSN 1615-1488* » (en cours d'évaluation). Cet article présente les résultats de l'application de l'algorithme d'optimisation OSMO sur un problème d'assemblage par interférence sollicité par un chargement de torsion alternée. Les informations obtenues par l'algorithme OSMO sur le problème décrit permettent une sélection justifiée de la solution optimale.

Enfin, une conclusion résume les différents éléments à retenir de ce travail. De plus, des perspectives d'amélioration de l'algorithme y sont également présentées.

## CHAPITRE 1 REVUE DE LITTÉRATURE SUR LES ASSEMBLAGES PAR INTERFÉRENCE

.Le frettage est un procédé d'assemblage de deux pièces avec serrage. Il est populaire puisqu'il est simple, rigide et efficace. Ce procédé d'assemblage est couramment utilisé dans l'industrie de la fabrication, automobile, ferroviaire et aéronautique. Ses applications sont principalement sur des pièces rotatives comme l'assemblage d'un arbre avec des roues d'engrainages, des roulements à billes, des roues de trains etc.

L'assemblage est constitué principalement de deux éléments : un arbre et un moyeu. Le frettage consiste à l'assemblage du moyeu et de l'arbre avec un ajustement serré. Ainsi, lorsqu'ils sont assemblés, ils peuvent transmettre les forces/moment appliqué sur l'arbre au moyeu. La Figure 1-1 présente les deux pièces avant l'assemblage. La Figure 1-2 montre les deux pièces assemblés dans le cadre d'essais expérimentaux.

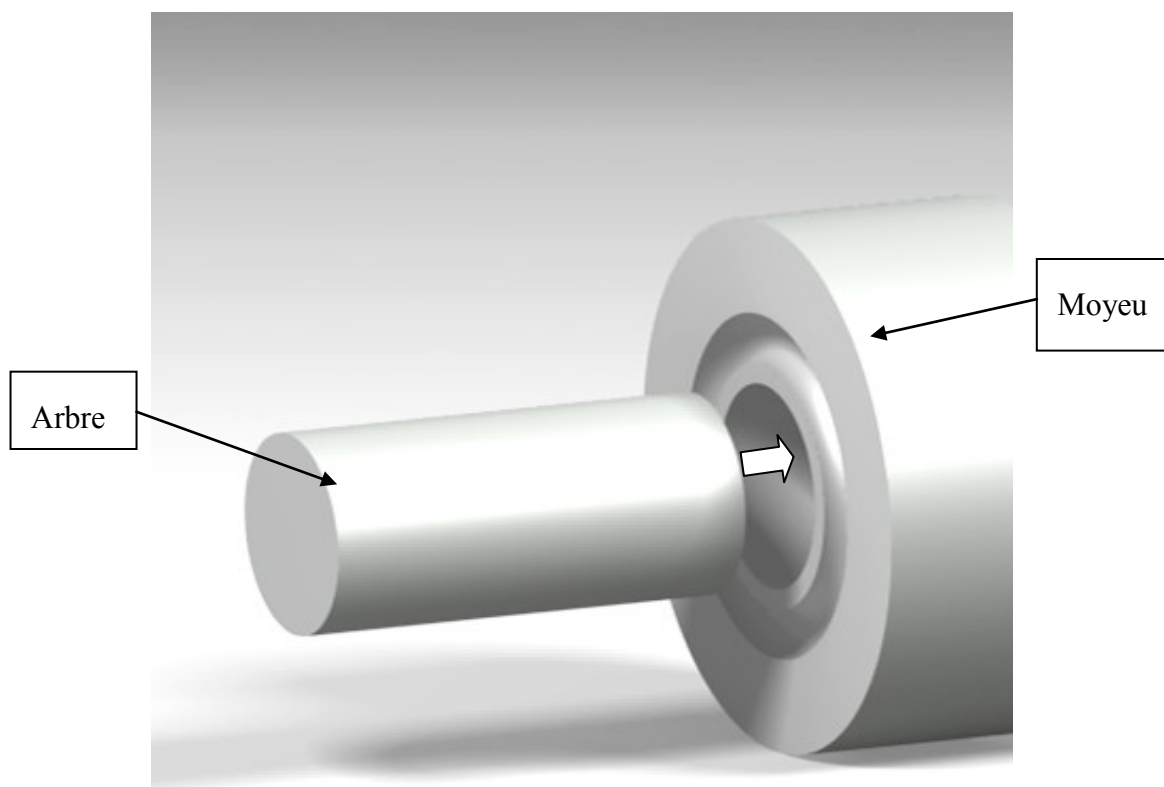


Figure 1-1 Assemblage par interférence



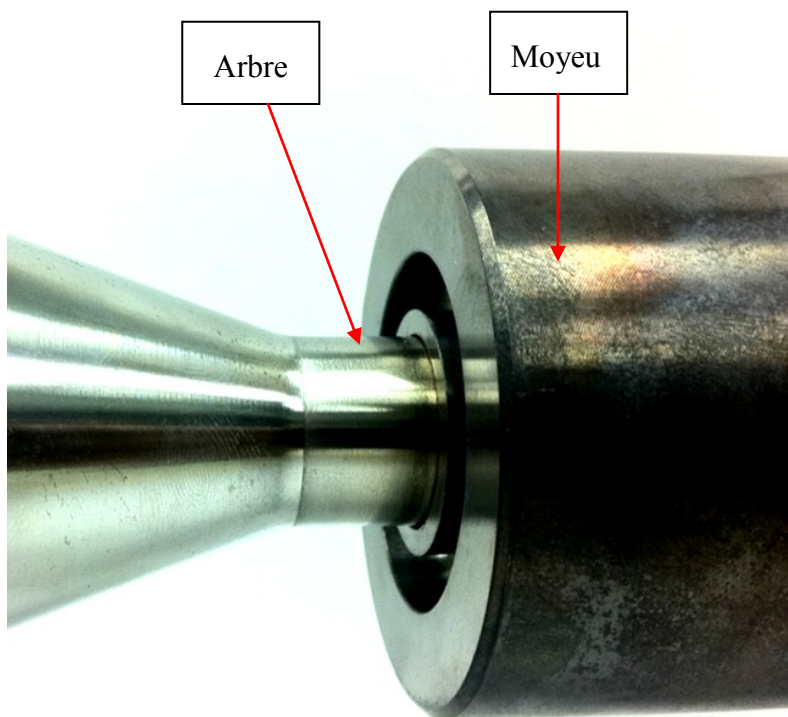


Figure 1-2 Assemblage par interférence utilisé pour des essais expérimentaux

Le but principal de ce travail est de présenter une méthode de conception optimale des assemblages par interférence. Ainsi, ce chapitre présente les connaissances actuelles sur les assemblages par interférence soumis à des sollicitations en fatigue. Les informations retenues serviront à la conception d'un modèle d'analyse générique robuste.

## 1.1 Calculs de base

Les équations analytiques sont utilisées pour des calculs approximatifs sur le comportement global du joint. Elles servent d'ailleurs à calculer le jeu nécessaire à l'assemblage des pièces et à exprimer la pression nominale de contact.

### 1.1.1 Paramètre de l'assemblage

L'assemblage par interférence est l'emmanchement de deux pièces nommées arbre et moyeu. La Figure 1-3 et le décrit les principaux paramètres de l'assemblage. Les équations de base du comportement sont développées pour cette géométrie.

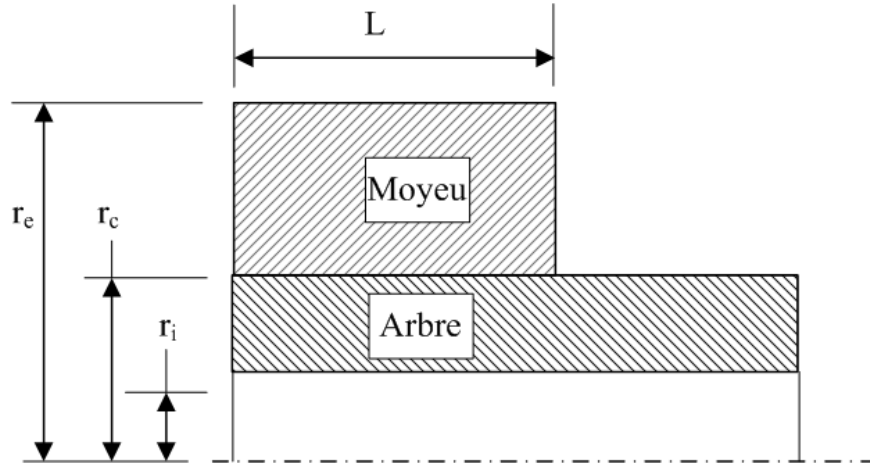


Figure 1-3 : Moyeu assemblé avec arbre simple

Tableau 1.1: Définition des paramètres

Paramètre	Définition
L	Longueur de l'emmanchement
$r_i$	Rayon intérieur de l'arbre
$r_e$	Rayon extérieur du moyeu
$r_c$	Rayon nominal à l'intersection des deux pièces
$\delta_r$	Interférence radiale entre les 2 pièces

### 1.1.2 Calcul de la pression de contact

La pression de contact se calcule en résolvant les équations de Lamé pour des cylindres à parois épaisses de longueur infinie. Dans un repère cylindrique  $(r, \theta)$ , pour une pression interne ( $P_i$ ) et externe ( $P_e$ ), les équations de Lamé permettent d'obtenir la contrainte radiale ( $\sigma_r$ ) et circonférentielle ( $\sigma_\theta$ ) d'un cylindre :

$$\sigma_r(r) = \frac{r_i^2 p_i - r_e^2 p_e}{r_e^2 - r_i^2} - \frac{(p_i - p_e) r_e^2 r_i^2}{r^2 (r_e^2 - r_i^2)} \quad (1.1)$$

$$\sigma_\theta(r) = \frac{r_i^2 p_i - r_e^2 p_e}{r_e^2 - r_i^2} + \frac{(p_i - p_e) r_e^2 r_i^2}{r^2 (r_e^2 - r_i^2)} \quad (1.2)$$

Ensuite, pour de mêmes conditions, le déplacement  $u_r$  est :

$$u_r = \frac{1 - \nu}{E} \left( \frac{r_i^2 p_i - r_e^2 p_e}{r_e^2 - r_i^2} \right) r + \frac{1 + \nu}{E} \left( \frac{(p_i - p_e) r_e^2 r_i^2}{r^2 (r_e^2 - r_i^2) r} \right). \quad (1.3)$$

Dans un assemblage par interférence, l'arbre aura une contrainte circonférentielle en compression tandis que pour le moyeu elle sera en tension. À l'interface de contact, la contrainte radiale entre les deux pièces est continue et égale à la pression de contact P en valeur absolue.

Après l'emmanchement, l'arbre subira ainsi une déformation - déplacement radial  $u_i$  et une pression externe égale à P. Le moyeu pour sa part aura une pression interne égale à P et un déplacement radial  $u_e$ . Ces conditions doivent être respectées et vérifiées lors de la modélisation du comportement de l'assemblage par la méthode des éléments finis de l'assemblage.

L'interférence est alors équivalente à la somme des déplacements radiaux des deux pièces :

$$\delta_r = |u_i| + |u_e|. \quad (1.4)$$

En combinant les conditions énumérées ci-dessus et les équations 1.1-1.4, on peut déterminer la pression nominale de contact à l'interface entre l'arbre et le moyeu [1] :

$$P = \frac{\delta_r}{r_c \left[ \frac{1}{E_m} \left( \frac{r_o^2 + r^2}{r_o^2 - r^2} + \nu_e \right) + \frac{1}{E_a} \left( \frac{r_i^2 + r^2}{r^2 - r_i^2} + \nu_s \right) \right]} \quad (1.5)$$

où  $E_m$  et  $E_a$  sont les modules de Young du moyeu et de l'arbre respectivement. Enfin, ces relations découlent de la théorie des cylindres composés dans un plan bidimensionnel.

### 1.1.3 Force axiale transmissible

Une relation entre la pression de contact P, le coefficient de friction  $\mu$  et la longueur d'emmanchement L, permet d'approximer la force axiale maximale que l'assemblage peut transmettre sans qu'il y a glissement relatif à l'interface [2]:

$$F = 2\pi r_c \mu L P. \quad (1.6)$$

Cette relation est établie en ne tenant pas compte des efforts de cisaillement induits après l'assemblage. Elle permet donc seulement d'approximer grossièrement le comportement.

### 1.1.4 Couple transmissible

Le couple maximal transmissible par l'assemblage est approximé par une relation similaire [2] :

$$T = 2\pi r_c^2 \mu L P. \quad (1.7)$$

### 1.1.5 Dilatation thermique des pièces

Deux moyens sont utilisés pour assembler le moyeu et l'arbre : la dilatation thermique (*shrink fit*) et avec une presse (*press fit*). Pour un assemblage pressé, la force nécessaire à l'assemblage est approximée par l'équation (1.6). Pour l'assemblage par dilatation thermique, le principe consiste à chauffer le moyeu et refroidir l'arbre suffisamment afin de créer un jeu qui permettra l'insertion de l'arbre dans le moyeu. Cette technique génère moins de contraintes résiduelles et offre de meilleurs résultats [3]. Pour chaque composante, l'équation qui permet de calculer le déplacement radial en fonction du rayon et de la variation de la température s'exprime ainsi :

$$u_r(r) = r\alpha\Delta T. \quad (1.8)$$

Où  $\alpha$  est le coefficient de dilatation thermique propre au matériau.

## 1.2 Configurations géométriques

Il existe plusieurs configurations géométriques ayant une influence prépondérante sur la vie en fatigue de l'assemblage par interférence. Dans une première partie, les essais expérimentaux disponibles dans la littérature sont présentés afin de déterminer l'influence des différents paramètres géométriques sur la vie en fatigue de l'assemblage. Ensuite, une étude de sensibilité sur les configurations géométriques réalisée avec des calculs par la méthode des éléments finis y sera présentée.

### 1.2.1 Essais avec chargement en torsion alternée

Hattori et al. [4] ont observé que le facteur de réduction de vie en fatigue d'un assemblage soumis à des sollicitations de flexion rotative versus un chargement en torsion alternée est deux fois plus élevé. Tel qu'illustré à la Figure 1-4, Hattori et al. [4] ont étudié trois types de configuration

d'assemblage : simple (a), avec épaulement (b) et avec une rainure circulaire (c). Ses principales observations sont les suivantes:

- La configuration c) avec une partie en porte-à-faux (*overhang* dans la figure) n'améliore pas de façon significative la vie en fatigue de l'assemblage. De plus, si les valeurs des paramètres ne sont pas contrôlées adéquatement, elle peut plutôt réduire la vie en fatigue.
- La configuration b) n'améliore pas la vie en fatigue de l'assemblage.

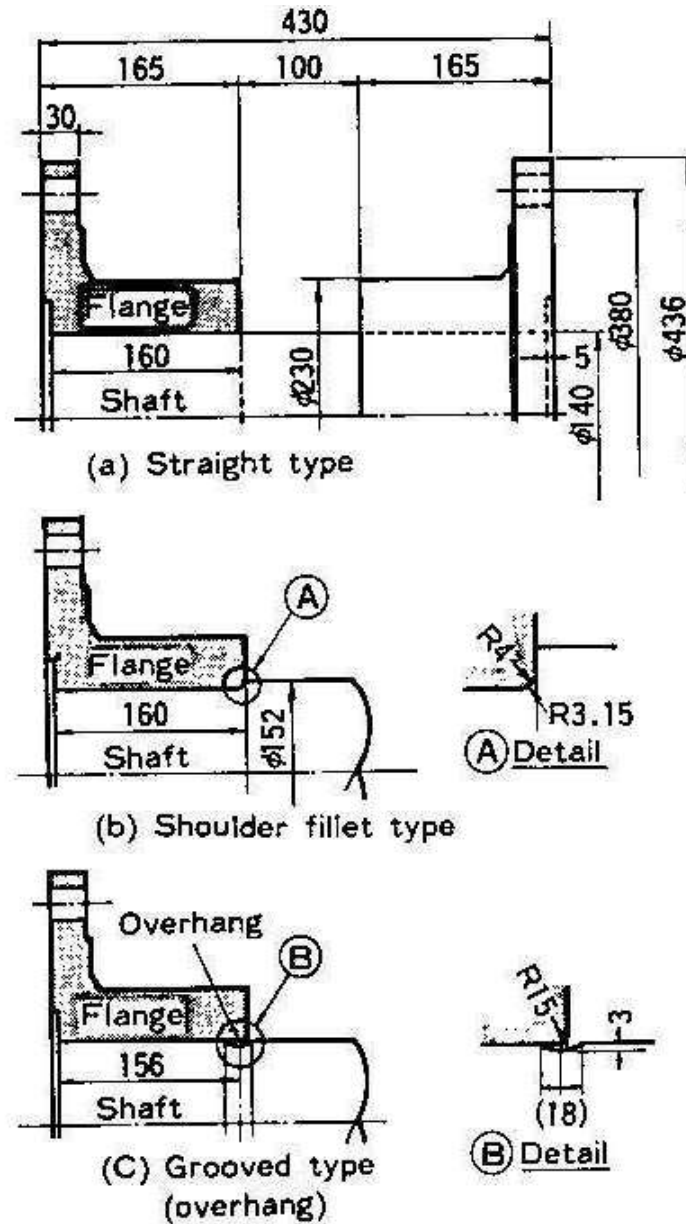


Figure 1-4 Géométries des essais de Hattori et al. [4]

Les essais de Juuma [5-7] traitent également des assemblages par interférence sollicités en torsion alternée. La géométrie utilisée lors de ses essais expérimentaux est présentée à la Figure 1-5. La configuration géométrique est conservée à chaque essai, par contre il a fait varié les dimensions  $D$ ,  $d$ ,  $r$ ,  $M$  et  $\delta L$  (présentés sur la Figure 1-3). Son étude porte sur un moyeu avec une partie en porte-à-faux assemblé à un arbre avec épaulement. Sur cette configuration, ses observations sont les suivantes:

- Il existe un niveau de contraintes où la rupture s'amorce soit dans le congé, soit dans l'arbre. On définit ce niveau comme étant une zone de transition de la localisation de la rupture.
- Lorsque la combinaison des paramètres  $r$  et  $d$  est adéquate, il semble que la partie en porte-à-faux soit bénéfique pour la résistance à la fatigue.
- La rupture à la surface de contact est provoquée par un phénomène de fretting-fatigue tandis que dans le congé c'est la fatigue normale de l'arbre.

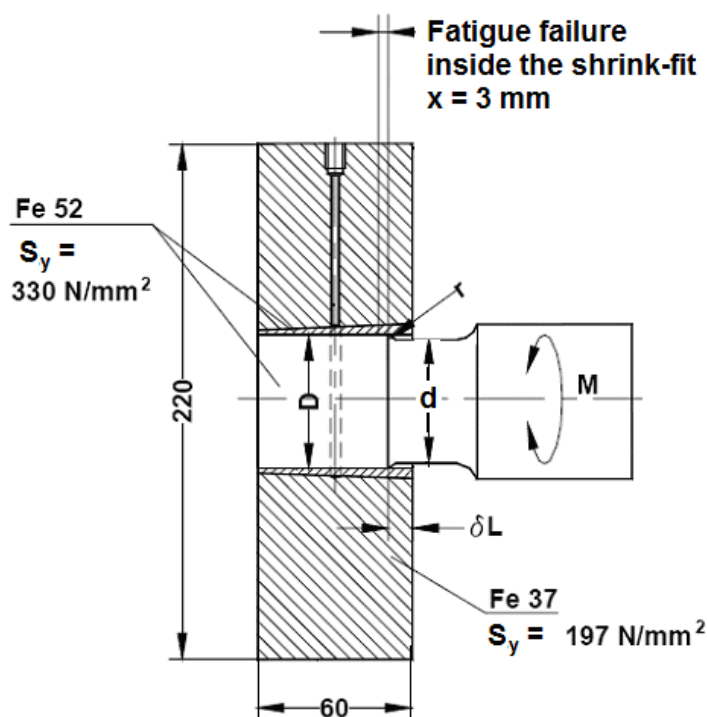


Figure 1-5 Configuration géométrique des essais de Juuma [7]

Les affirmations de Juuma, Hattori et al. sont contradictoires au sujet de la pertinence d'une partie en porte-à-faux du moyeu. Toutefois, les essais expérimentaux de Hattori et al. sont beaucoup moins complets quant à l'étude de la profondeur de la rainure et son rayon. Ses conclusions pourraient donc être le résultat d'une combinaison défavorable des valeurs des paramètres utilisées.

Ainsi, il est justifiable de croire que la configuration géométrique qui offre les meilleurs résultats en fatigue pour un chargement de torsion alternée est celle étudiée par Juuma. C'est-à-dire, un moyeu avec une partie en porte-à-faux assemblé avec un arbre avec épaulement.

### **1.2.2 Essais avec chargement en flexion alternée**

Nishioka et Komatsu [8] ont été dans les premiers à étudier l'effet des différentes configurations sur la vie en fatigue de l'assemblage par interférence soumis à des sollicitations de flexion rotative. Les configurations utilisées dans ses essais sont présentées à la Figure 1-6 et ses observations sont les suivantes :

- La configuration D est celle qui procure la meilleure résistance en fatigue.
- La configuration E est également bénéfique, mais est plus sensible aux valeurs des paramètres
- Il existe un compromis entre  $d/D$ ,  $\rho$  et  $\delta L$  qui permet d'obtenir une vie en fatigue optimale.

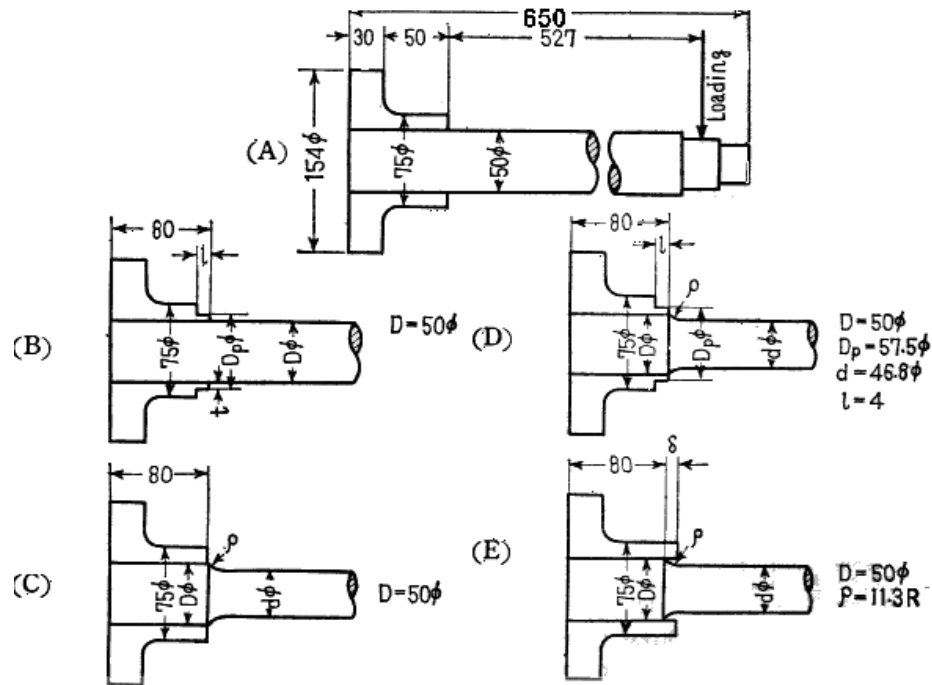


Figure 1-6 Configurations de Nishioka. [8]

Hirakawa et al. [9] ont réalisé une étude sur l'amélioration de la vie en fatigue d'une roue de train et d'engrenage assemblés par frettage sur un arbre. Les géométries utilisées sont présentées à la Figure 1-7. Cette étude porte sur la comparaison entre un assemblage conventionnel et un avec une rainure circulaire et une partie des roues assemblées en porte-à-faux. Bien que les détails soient limités, son observation principale est que selon la combinaison de  $t$  et de  $\rho$ , il est possible d'améliorer la vie en fatigue d'un facteur d'environ 2.2 comparativement à un assemblage conventionnel.

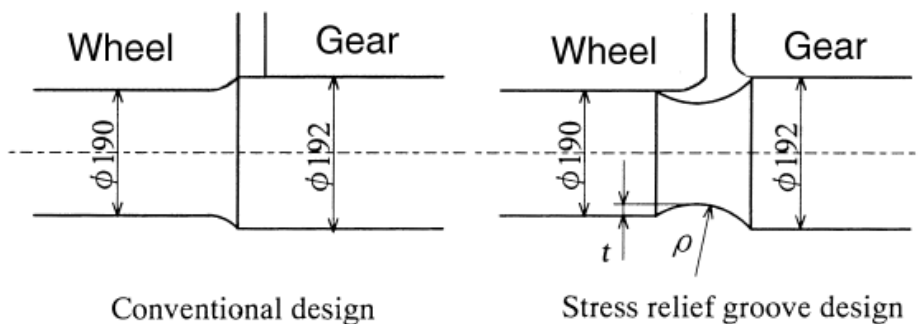


Figure 1-7 Comparaison entre un modèle d'assemblage conventionnel et celui amélioré. [9]



Kubota et al. [10] a proposé des essais expérimentaux afin de simuler des conditions de frettage. Ses spécimens d'essai sont rectangulaires et leurs géométries sont présentées à la Figure 1-8. Ses observations sont les suivantes :

- Avec un compromis entre  $\theta$ ,  $R$  et  $d$ , la limite de fatigue par rapport à un spécimen sans rainure a été améliorée d'un facteur 1.57.
- Il est possible de caractériser l'amélioration par rapport à un spécimen sans rainure en fonction d'un paramètre adimensionnel  $\theta d/D$ , où  $D$  serait le diamètre (méthode appliquée également aux essais de Nishioka et Komatsu [8]).

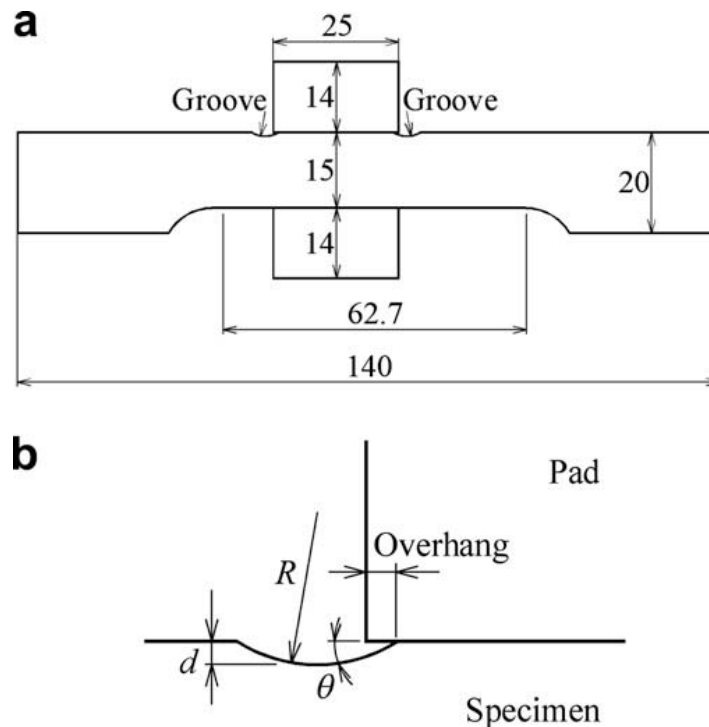


Figure 1-8 Modèle d'essai de Kubota et al. [10]

Pour les assemblages sollicités en flexion, les résultats des essais expérimentaux semblent converger vers l'assemblage d'un moyeu avec une partie en porte-à-faux assemblé à un arbre avec épaulement. Avec la littérature disponible sur les essais expérimentaux, il est possible d'affirmer que cette configuration semble avoir une vie en fatigue supérieure, peu importe le chargement. Toutefois, la plupart des auteurs [4-8, 10] mentionnent qu'il y a un compromis judicieux de la valeur des paramètres afin d'atteindre une vie en fatigue optimale.

### 1.2.3 Étude de sensibilité à partir de calcul par éléments finis

Lanoue [11] a effectué une analyse de sensibilité des différents paramètres selon 4 configurations avec à l'aide de calcul par éléments finis : a) arbre simple, b) arbre avec épaulement, c) arbre avec une rainure circulaire et d) Arbre avec moyeu en saillie. Il a étudié ces configurations pour un chargement de flexion rotative et de torsion, les géométries et leurs paramètres associés sont présentés à la Figure 1-9 :

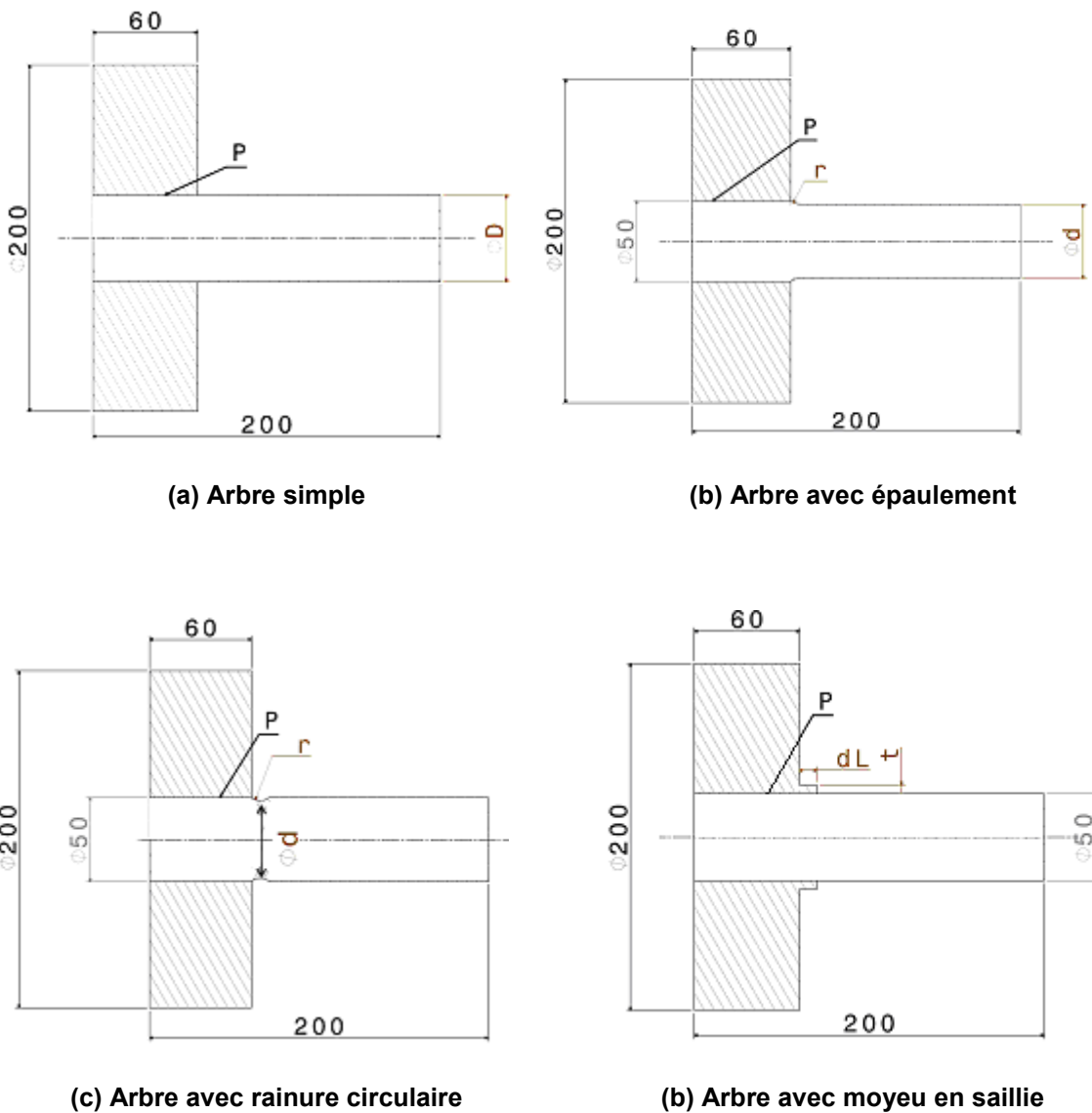


Figure 1-9 : Configurations [11]

ses recommandations ont été les suivantes :

- La configuration la plus avantageuse pour un chargement en flexion rotative ou alternée est celle d'un arbre avec épaulement.
- En augmentant le diamètre et en ajustant les paramètres  $r$  et  $P$ , il est possible de diminuer considérablement la concentration de contrainte présente dans le contact.
- Avec la configuration d'un arbre avec épaulement, il est possible d'augmenter la vie en fatigue de l'ordre de 50 % pour un chargement en flexion rotative et de 20 % pour un chargement de torsion alternée.
- Les tendances observées pour un arbre avec une rainure sont les mêmes qu'avec épaulement, mais avec un moindre gain.

Enfin, par ses études de sensibilité, Lanoue [11, 12] montre qu'une configuration d'un arbre avec épaulement est favorable. Ainsi, les résultats des études de sensibilité réalisées avec la méthode des éléments finis sont en accord avec ceux expérimentaux présentés précédemment.

### **1.3 Modélisation par éléments finis de l'assemblage**

Les différents essais expérimentaux ont permis d'étudier l'effet de certaines configurations géométriques sur la vie en fatigue. Parmi eux, les essais de Juuma [5-7], Nishioka et Komatsu [8] contiennent suffisamment de détails afin que Lanoue [12] puisse comparer les résultats des simulations par éléments finis avec leurs essais expérimentaux. La Figure 1-10 montre le modèle d'éléments finis utilisé par Lanoue pour effectuer les analyses de convergence du maillage et de l'algorithme de contact. Dans la région de l'arrêt de contact, il existe une zone de concentration de contraintes où l'amorce de la rupture a souvent lieu. Cette zone devra donc être raffinée. Afin de proposer un modèle par éléments finis de taille raisonnable et exploitable avec un ordinateur de bureau relativement performant, une technique de sous-modélisation ou de symétrie cyclique a été appliquée tout en générant de bons résultats. Au préalable, il est nécessaire de s'assurer de la convergence de l'algorithme de contact entre les deux pièces.

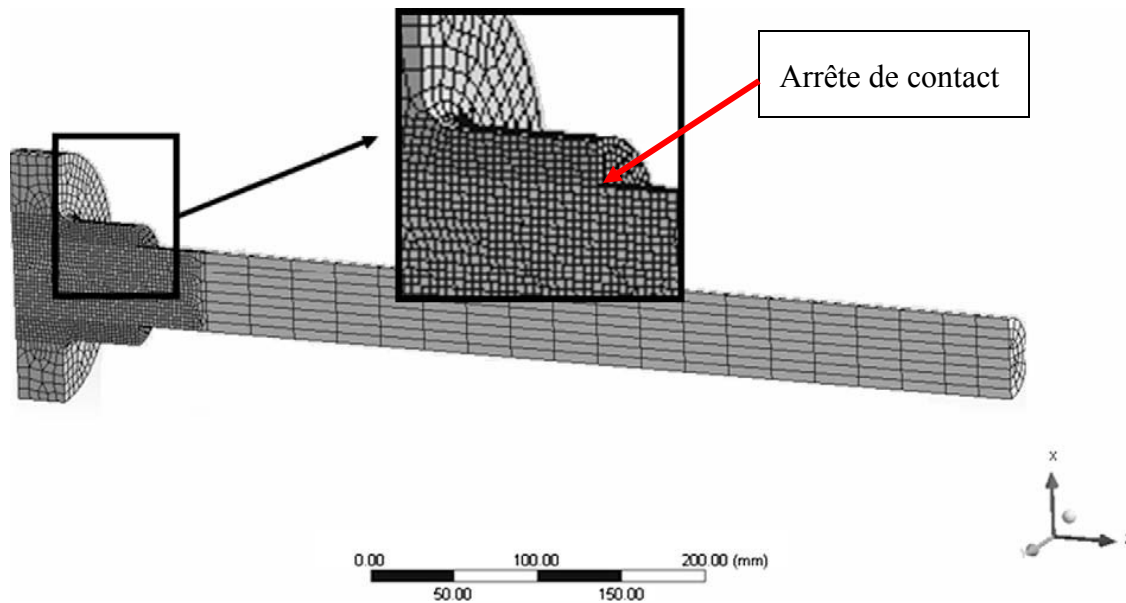


Figure 1-10 Maillage global [12]

### 1.3.1 Algorithmes de contacts

Les assemblages par interférence nécessitent une formulation de contact de type rugueux (avec friction) étant donné le glissement relatif possible entre les surfaces induisant la fretting-fatigue. Dans cette lignée, Lanoue a réalisé une étude de convergence sur les différents algorithmes de contact présents dans la librairie d'Ansys (Logiciel utilisé pour ce travail). Il s'est intéressé aux 4 algorithmes suivants : la méthode de pénalité, le Lagrangien augmenté, le Lagrangien normal et le Lagrangien pur. La méthode de pénalité et Lagrangien augmenté utilisent une fonction de pénalité, avec un contrôle direct sur la rigidité de contact pour le premier contrairement à un contrôle sur la pénétration pour le second. Le Lagrangien pur utilise les multiplicateurs de Lagrange dans la direction normale et tangentielle au contact. Enfin, le Lagrangien Normal utilise la méthode de pénalité dans la direction normale et les multiplicateurs de Lagrange dans la direction tangentielle.

L'étude de Lanoue a été réalisée sur une des configurations géométriques utilisées par Nishioka [8]. Enfin, dans un but d'obtenir la convergence du contact [12], ses recommandations sont les suivantes:

- Avec les assemblages par interférence, le meilleur indicateur de convergence est celui de l'interférence  $\delta_r$ .

- Le raffinement du maillage est nécessaire pour assurer une convergence.
- La méthode du Lagrangien augmenté est recommandée puisqu'elle utilise les notions de pénétration maximale et de glissement élastique maximal comme contrôle. Il est donc possible de quantifier l'erreur acceptable dans le contact.

Dans le cas des assemblages par interférences, le glissement entre les 2 surfaces a une amplitude de quelques microns ( $\mu\text{m}$ ). Une tolérance trop sévère entraîne souvent des problèmes de convergence tandis qu'une trop permissive augmente des risques d'erreur. Ainsi, Lanoue [12] a établi que l'utilisation de la méthode du Lagrangien augmenté avec une tolérance de pénétration de 0.1% de l'interférence radiale combiné à une tolérance de glissement de 0,1  $\mu\text{m}$  constituait la méthode la plus efficace pour modéliser ces assemblages. Pour chaque modèle, la convergence doit toutefois être validée.

### 1.3.2 Symétrie cyclique

Lorsque l'assemblage par interférence est cyclique et qu'il est soumis à des sollicitations de torsion alternée, il est possible de modéliser son comportement avec une tranche de  $1^\circ$  avec des conditions de symétrie cyclique [12]. Une macro a été développée afin de relier les nœuds des surfaces auxquels la symétrie cyclique est appliquée par les équations de contraintes. Pour y arriver, il est nécessaire d'avoir un maillage réglé entre les 2 surfaces axisymétriques auquel chaque nœud possède un homologue de même coordonné (x et y) dans un système de coordonnées cartésiennes, voir Figure 1-11. Les équations de contrainte s'écrivent ainsi :

$$\begin{aligned}u_{x2} &= u_{x1} \cos \theta - u_{y1} \sin \theta \\u_{y2} &= u_{x1} \sin \theta + u_{y1} \cos \theta \\u_{z2} &= u_{z1},\end{aligned}$$

où  $\theta$  représente l'angle de la tranche du modèle 3D pour la symétrie cyclique.

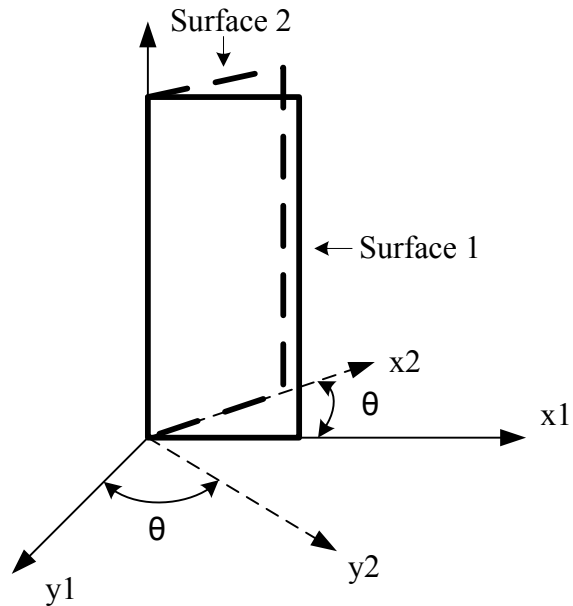


Figure 1-11 Symmétrie cyclique

### 1.3.3 Sous-modélisation

Dans un cas de chargement complexe où l'utilisation de symétrie cyclique n'est pas possible, une méthode de sous-modélisation permet de réduire la taille du modèle. Dans le cas des assemblages par interférence, dû à la forte concentration de contrainte générée par l'arrêt de contact, il est nécessaire d'avoir une taille de maillage locale beaucoup plus petite que le modèle global. La sous-modélisation permet donc d'atteindre cette taille de maillage à l'arête de contact tout en respectant le nombre de nœuds maximal imposé par la licence. La Figure 1-12 présente le sous-modèle étudié par Lanoue [12] et son maillage raffiné autour de l'arête de contact. Cette méthode de raffinement sera typiquement utilisée dans le cadre de ce travail.

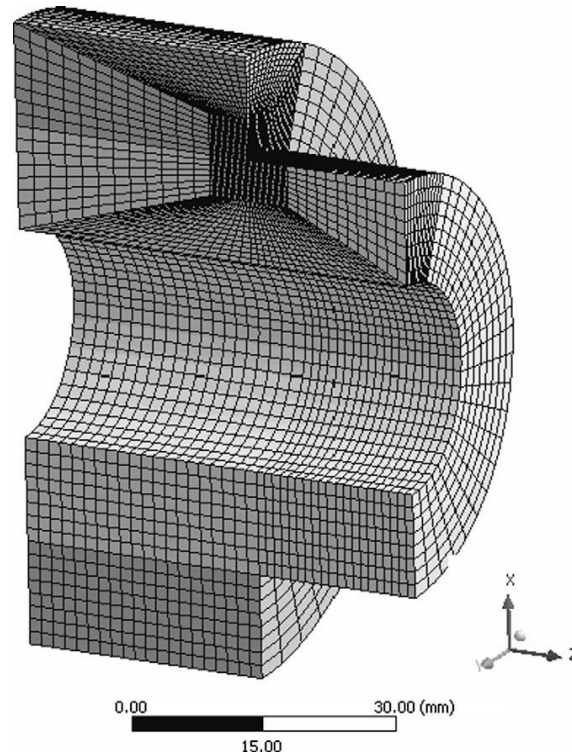


Figure 1-12 Sous-modèle du modèle global présenté à la Figure 1-10 [12]

Le principe de la sous-modélisation consiste à approximer les conditions limites de la frontière du sous-modèle dans son modèle global. Pour effectuer la sous modélisation, un algorithme a été développé par Cormier et al. [13] et traduit par Lanoue [11]. Cet algorithme est présenté à l'Annexe 1.

## 1.4 Endommagement par fatigue

Si la section précédente visait essentiellement à définir les méthodes permettant de simuler l'état de contraintes des assemblages par interférence, celle-ci vise particulièrement à calculer sa vie en fatigue.

### 1.4.1 Critère de fatigue

Le processus de fatigue se divise grossièrement en trois phases; l'amorce de la fissure, la propagation de la fissure et la rupture finale [1]. La période d'initiation de la fissure occupe la plus grande partie de la vie de la pièce. Une courbe S-N propre à chacun des matériaux établit la relation entre le nombre de cycles (N) menant à la l'amorce de la fissure en fonction de la contrainte S. Les données formant cette courbe sont généralement disponibles pour la plupart des

matériaux. Certains critères tels que Von Mises, Crossland et Sines basés sur l'état de contraintes multiaxiales permettent d'approximer la vie en fatigue du matériau à l'aide de la courbe S-N. Pour les aciers, lorsque la contrainte équivalente est inférieure à une limite d'endurance ( $S_e$ ), la vie en fatigue de l'assemblage est considérée comme infinie.

Lorsque l'état de contraintes est multiaxial, il est recommandé d'utiliser un critère de fatigue multiaxial. Weber [14] a fait un inventaire de l'ensemble des critères de fatigue disponibles et les a appliqués à 233 essais expérimentaux différents. Il a ensuite mesuré la justesse et la précision de chacun et Fogue, Robert et Dang Van ont retenu l'attention. Par contre, un critère de fatigue multiaxial nécessite des données sur la limite d'endurance en traction, torsion et flexion alternée du matériel. Ce genre de données n'est malheureusement pas disponible dans la littérature portant sur les assemblages par interférence.

Toutefois, Lanoue [11] a appliqué les critères de Sines, Crossland, Von Mises et Dang Van (basé sur une approximation de la limite d'endurance en torsion alternée) sur les résultats des simulations d'éléments finis obtenus pour les géométries des essais de Juuma, Nishioka et Komatsu. [6, 8]. Ses résultats ont démontré que Sines, Crossland et Von Mises étaient équivalents et pouvait modéliser avec précision la vie en fatigue de l'assemblage. Celui de Dang Van donne des résultats légèrement moins précis, mais semble moins sensible au type de chargement.

Tableau 1.2 Nomenclature des termes utilisés pour les calculs de fatigue

Paramètre	Définition
$S_e$	Limite d'endurance
$S_f$	Limite de fatigue
FS	Facteur de sécurité
$S_{ut}$	Limite de résistance
$\sigma_a, \sigma_m$	Contrainte alternée et moyenne

Étant donné sa simplicité et sa précision, le critère de Von Mises est retenu pour ce travail. Les contraintes moyenne et alternée s'écrivent comme suit :

$$\sigma_a = \sqrt{\frac{(\sigma_{z,a} - \sigma_{x,a})^2 + (\sigma_{y,a} - \sigma_{x,a})^2 + (\sigma_{z,a} - \sigma_{y,a})^2 + 6(\tau_{xy,a}^2 + \tau_{xz,a}^2 + \tau_{yz,a}^2)}{2}} \quad (1.9)$$



$$\sigma_m = \sqrt{\frac{(\sigma_{z,m} - \sigma_{x,m})^2 + (\sigma_{y,m} - \sigma_{x,m})^2 + (\sigma_{z,m} - \sigma_{y,m})^2 + 6(\tau_{xy,m}^2 + \tau_{xz,m}^2 + \tau_{yz,m}^2)}{2}} \quad (1.10)$$

Selon le diagramme de Goodman modifié, une correction due à la contrainte moyenne est appliquée à la contrainte alternée. On peut alors écrire le facteur de sécurité ( $FS$ ) ainsi :

$$FS_{VM} \begin{cases} = \frac{1}{\frac{\sigma_a}{S_f} + \frac{\sigma_m}{S_{ut}}} & \text{si } \sigma_m > 0 \\ = \frac{\sigma_a}{S_f} & \text{si } \sigma_m \leq 0. \end{cases} \quad (1.11)$$

Selon l'équation 1.10 , il est alors possible d'écrire la contrainte alternée équivalente ( $\sigma_a^*$ ), qui permet d'établir une relation avec les courbe S-N :

$$\sigma_a^* \begin{cases} = \frac{\sigma_a}{1 - \frac{\sigma_m}{S_{ut}}} & \text{si } \sigma_m > 0 \\ = \sigma_a & \text{si } \sigma_m \leq 0. \end{cases} \quad (1.12)$$

## 1.4.2 Endommagement par fretting

Le fretting est un phénomène d'endommagement qui est causé par un faible glissement oscillatoire entre deux surfaces en contact. Il réduit considérablement la vie en fatigue des assemblages dans la cette zone. Il est déterminé que l'endommagement par fretting dépend essentiellement de trois paramètres: la pression de contact, l'amplitude de glissement et le coefficient de friction [15].

Des observations montrent que l'amplitude de glissement des assemblages par interférences se stabilise et est inférieure à 25  $\mu\text{m}$ . [5, 11, 16] Selon la Figure 1-13, ils se trouvent donc dans un régime de glissement partiel. Pour ce régime, on remarque un lien étroit entre l'augmentation du glissement et la diminution de la vie en fatigue. Les études menées par Lanoue [11] montre également ce lien avec les calculs par éléments finis.

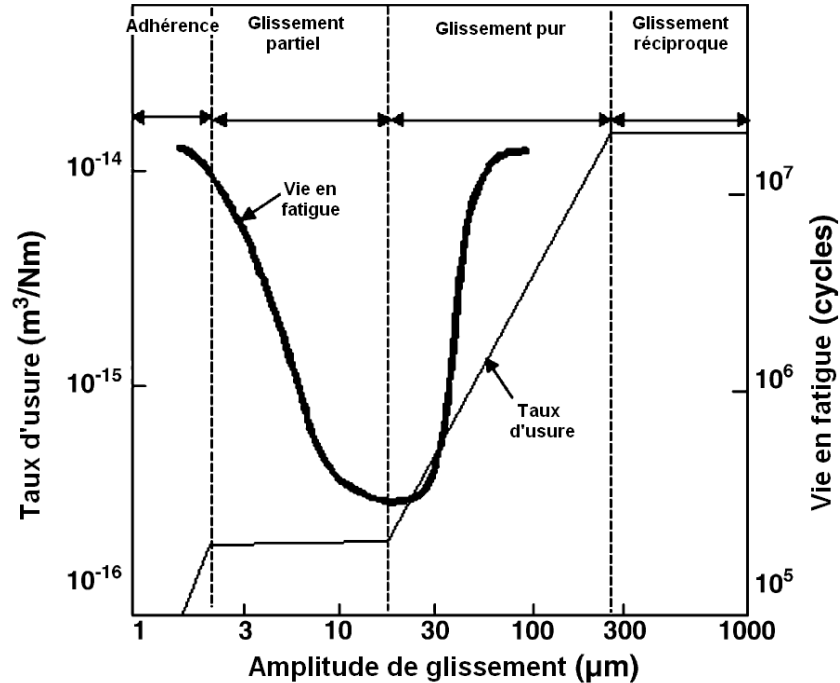


Figure 1-13 Amplitude de glissement, le taux d'usure et la vie en fatigue [15]

Dans le contact, la comparaison des résultats des analyses par éléments finis et des essais expérimentaux a permis de déterminer la présence d'un facteur de réduction de vie en fatigue ( $k_{\text{fret}}$ ) [11]. Par contre, lorsque la rupture se produit à l'extérieur du contact, le critère de Von Mises approxime avec justesse la vie en fatigue. Ainsi, les informations indiquent que la rupture dans le congé est causée par la fatigue normale tandis que dans le contact elle est causée par la fatigue par fretting.

## 1.5 Calcul de vie en fatigue par la méthode des éléments finis

Lanoue [11] a présenté une méthodologie qui permet de simuler avec précision un assemblage par frettage soumis à des sollicitations de fatigue par la méthode des éléments finis. Les principales étapes ont été détaillées à la section 1.3. Avec ses résultats, il a pu faire les observations suivantes :

- Il existe un facteur de réduction de vie en fatigue ( $k_{\text{fret}}$ ) à l'interface de contact
- L'amplitude de glissement semble être la conséquence d'une réduction de vie en fatigue plutôt que la cause.

- Il existe une transition dans la localisation de l'amorce de la fissure entre la zone de contact et un congé.

Bien que l'ajout d'un congé semble augmenter la vie en fatigue dans le contact, la concentration de contrainte induite diminue la vie en fatigue de l'arbre. Le compromis à faire pour assurer une vie en fatigue optimale est l'objet de l'optimisation présenté au Chapitre 5.

## 1.6 Méthode de conception des assemblages par interférences présentée dans la littérature

Tels que présentés précédemment, plusieurs ouvrages traitent de l'analyse des contraintes présentes dans les assemblages par interférences, mais peu s'attardent à sa conception. Parmi les méthodes disponibles, il y a ceux qui sont basées sur des équations analytiques décrites précédemment, et ceux qui se basent sur les résultats d'éléments finis.

### 1.6.1 Méthode de conception basée sur les équations analytiques

Norton [2] a proposé une méthode basée sur la sélection d'un facteur de réduction de vie en fatigue ( $K_t$ ) sur un graphique en fonction des paramètres adimensionnels ( $p/\sigma$ ) et ( $l/d$ ). Elle s'applique à un arbre simple assemblé dans un moyeu. C'est toutefois la seule méthode disponible qui traite de la longueur d'emmanchement. Elle est simpliste et n'est valide que pour approximer grossièrement l'état de contrainte présente dans l'arbre.

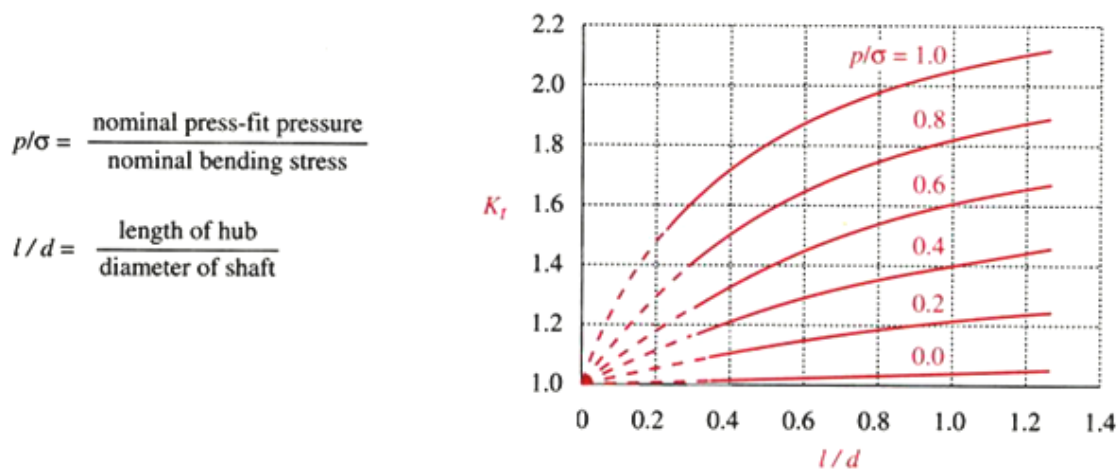


Figure 1-14 Réduction de vie en fatigue [2]

Castagnetti et Dragoni [17] traite de la conception des assemblages optimisés pour obtenir un transfert de charge maximum. Il utilise également un paramètre adimensionnel  $d/D$  et qui est basé sur les équations de Lamé présentées à la Section 1.1. Cette méthode est toutefois limitée, car elle nécessite la connaissance de plusieurs paramètres tels le diamètre nominal et la longueur d'emmanchement. Dans le cadre d'un problème d'optimisation, ces paramètres sont à déterminer. Cette méthode ne traite également pas de l'aspect fatigue.

### **1.6.2 Méthode de conception basée sur les éléments finis.**

Nishioka et Komatsu ont montré dans ses travaux expérimentaux que l'ajout d'une rainure circulaire à l'arbre améliorerait considérablement la vie en fatigue d'un assemblage soumis à des sollicitations de flexion rotative. Ainsi, Kubota et al. [10] ont étudié expérimentalement l'effet du rayon de courbure ( $R$ ) et l'angle de tangence ( $\theta$ ) sur la vie en fatigue avec la géométrie présentée à la Figure 1-8. Leurs résultats montrent qu'il existe une corrélation entre l'amélioration de la vie en fatigue et le paramètre  $(d\theta/D)$ . Cette même corrélation a également été appliquée sur les essais de Nishioka et observée avec les éléments finis. Cette méthode indique lorsqu'il y a rupture, mais ne prédit pas si elle a lieu dans le congé ou dans le contact. Elle peut toutefois être utile à déterminer les paramètres de la rainure afin d'obtenir une durée de vie améliorée de l'assemblage. Toutefois, la longueur d'emmanchement, la pression de contact, le diamètre du moyeu sont tous des paramètres qui demeurent à déterminer.

Lanoue [11] a présenté une méthode de conception basée sur les analyses par éléments finis. Une première partie consiste à établir approximativement le diamètre de l'arbre basé sur une étude de sensibilité et les équations analytiques de calcul de contraintes dans un arbre. Ensuite, une fois le diamètre déterminé, la seconde partie consiste à établir les paramètres initiaux de la configuration selon les études de sensibilité obtenues par les calculs par éléments finis. Après la validation d'une première géométrie, si la conception ne répond pas aux critères imposés, la sélection des valeurs des paramètres pour une seconde itération s'effectuera en fonction de la variation des paramètres observée lors de l'étude de sensibilité. Par exemple, s'il a été observé pour un diamètre  $D$  que le rayon contribue à la vie en fatigue, lors de la seconde itération on augmentera donc ce rayon.

Enfin, cette méthode permet d'obtenir une conception qui saura répondre à un facteur de sécurité avec peu d'itérations. Mais dans le cadre d'une conception optimale, trop d'éléments sont manquants afin de pouvoir s'en servir pour réaliser une optimisation. En effet, la méthode ne traite pas de la conception de la longueur d'emmanchement, du diamètre externe du moyeu et interne de l'arbre.

Toutes les méthodes présentées dans cette section simplifient le problème de conception afin de restreindre le nombre de paramètres à définir. Ainsi, l'objectif du présent ouvrage est donc de proposer une méthode de conception optimale, qui en plus de guider l'ingénieur dans la sélection de ses paramètres, saura répondre aux critères conception dans leur globalité. Cette méthode doit être également en mesure de considérer l'ensemble des paramètres définissant l'assemblage. Ainsi, l'approche utilisée consiste à l'élaboration d'un modèle générique d'éléments finis à partir des informations présentées dans ce chapitre. Ensuite, en effectuant une optimisation séquentielle selon des critères de conceptions spécifiques, il deviendra possible d'obtenir la valeur de ces paramètres.

## CHAPITRE 2 ÉTAT ACTUEL DES CONNAISSANCES SUR L'OPTIMISATION SÉQUENTIELLE MULTI-OBJECTIFS

Il arrive que l'état d'avancement des connaissances ou la complexité des phénomènes ne permettent pas l'obtention des modèles de calcul suffisamment simples et/ou précis. Dans ces situations, les éléments finis sont couramment utilisés. Cependant, la complexité des phénomènes et la disponibilité des ressources informatiques rendent le temps de résolution de calculs relativement élevé. Le but de ce travail est de mettre au point une méthode de conception optimale des assemblages par interférence. Pour y arriver, l'utilisation de l'optimisation séquentielle multi-objectifs a été identifiée comme appropriée.

Avec 10 paramètres, le nombre de simulations par éléments finis requis pour assurer une optimisation multi-objectifs convenable est de 80 000 calculs (obtenu du problème au Chapitre 4). Dans ce cas, coupler l'algorithme d'optimisation au modèle d'éléments finis n'est donc pas efficace du point de vue du temps de traitement. Le principe de l'optimisation séquentielle est donc de limiter le nombre de calculs à effectuer sur le modèle d'éléments finis tout en approximant la réponse du système dans un espace de conception donné.

Ce chapitre décrira l'état actuel des connaissances sur l'optimisation séquentielle. Les éléments présentés ont servi à l'élaboration de l'algorithme d'Optimisation Séquentielle Multi-Objective (OSMO) présentée au chapitre 3.

### ***2.1 Design and Analysis of Computer Experiments (DACE)***

Sacks et al. [18] ont été les précurseurs de l'optimisation séquentielle avec la méthode DACE pour modéliser les problèmes complexes à moindre coût. Pour pouvoir modéliser adéquatement un problème, il faut au préalable connaître les paramètres du problème; les variables d'entrée et les variables de réponses. Le but de DACE est donc de générer une fonction substitut qui permettra d'obtenir une réponse approximée en fonction des variables d'entrée.

Pour y arriver, la méthode stipule qu'il faut tout d'abord déterminer les variables d'entrées du problème. Ensuite, la deuxième partie consiste à élaborer un plan d'expériences pour lequel les variables de réponses sont définies. La troisième partie est la modélisation de la surface de réponse.

De plus, Sacks et al. insistent sur la validation par l'erreur des moindres carrés maximale (IMSE) ou moyenne (MMSE) de la surface de réponse. Il est donc possible d'optimiser les paramètres de cette surface de réponse afin d'obtenir une fonction substitut avec la précision requise. L'objectif de DACE est donc de générer une fonction substitut (fonction d'interpolation).

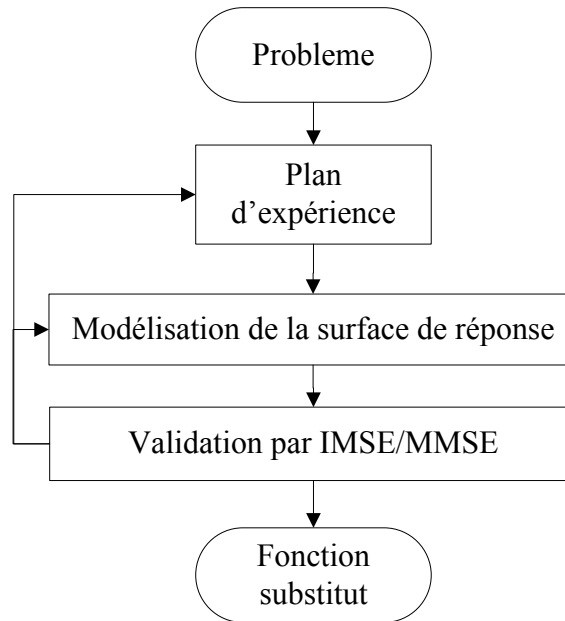


Figure 2-1 Algorithme de DACE [18]

## 2.2 Structure d'un algorithme d'optimisation séquentielle

L'optimisation séquentielle fait intervenir deux modèles distincts : le modèle numérique et le modèle d'interpolation. Le premier est celui pour lequel on connaît la réponse réelle, par exemple, un modèle d'éléments finis. Le second est celui sur lequel on réalise l'optimisation et consiste à interpoler le plan d'expériences réalisé sur le modèle numérique. La structure typique d'un algorithme d'optimisation séquentielle est présentée à la Figure 2-2 [19]. Selon cette nomenclature, on nomme le corps de l'algorithme un cycle et on le note par  $c^{(s)}$  auquel  $s$  représente son numéro d'itération. Un cycle contient plusieurs étapes nécessaires à sa réalisation :

1. Établir le modèle à résoudre.
2. Définir la zone d'intérêt  $\Omega_s$  à l'intérieur du domaine d'étude ( $\Omega_0$ ).
3. Réaliser le plan d'expériences pour  $\Omega_s$ .
4. Évaluer à l'aide du modèle numérique la réponse réelle aux points d'expériences.

5. Déterminer la fonction de transfert basée sur les points d'expériences.
6. Obtenir la solution de l'optimisation effectuée basée sur la fonction substitut.
7. Évaluer la valeur réelle de la réponse pour cette solution a l'aide du modèle d'interpolation.
8. Arrêter si le critère d'arrêt est satisfait, sinon retourner à l'étape 2.

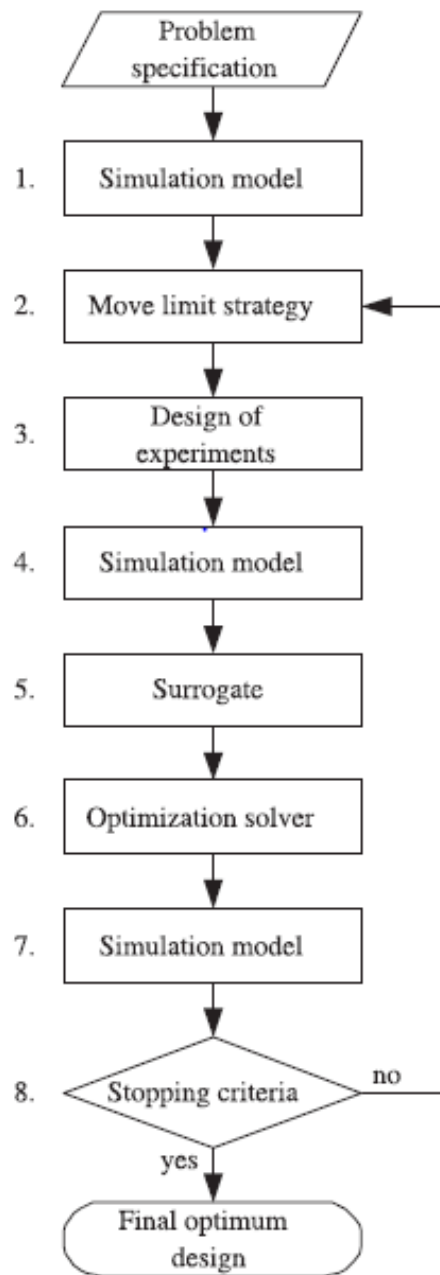


Figure 2-2 Algorithme typique d'optimisation séquentielle [19]



Pour obtenir une optimisation séquentielle efficace, il est nécessaire d'utiliser un plan d'expériences, un modèle d'interpolation et un solveur d'optimisation qui convient le mieux au besoin. À ce sujet, différents auteurs imbriquent des sous-algorithmes aux étapes présentées par Jacobs et al. avec un but d'améliorer la précision tout en diminuant le nombre d'évaluations du modèle. Les parties suivantes présenteront les divers travaux sur les plans d'expériences, les modèles d'interpolation et les différents algorithmes.

### 2.3 Zone d'intérêt

Un espace de conception est borné par les contraintes du problème d'optimisation. Il est toutefois normal que seulement une partie de cet espace de conception puisse contenir la solution optimale. Pour rendre encore plus efficace l'optimisation séquentielle, il est nécessaire de définir une zone d'intérêt pour laquelle le plan d'expériences sera défini. Cette zone d'intérêt a pour utilité de cerner la portion de l'espace de conception qui contient la solution optimale [20]. Telles que présentées par Jacobs et al. [19], les limites de cette zone d'intérêt seront modifiées au fil des itérations afin d'augmenter la sensibilité de la fonction d'interpolation par rapport au modèle numérique. La Figure 2-3 présente la distinction entre la zone d'intérêt et l'espace de conception.

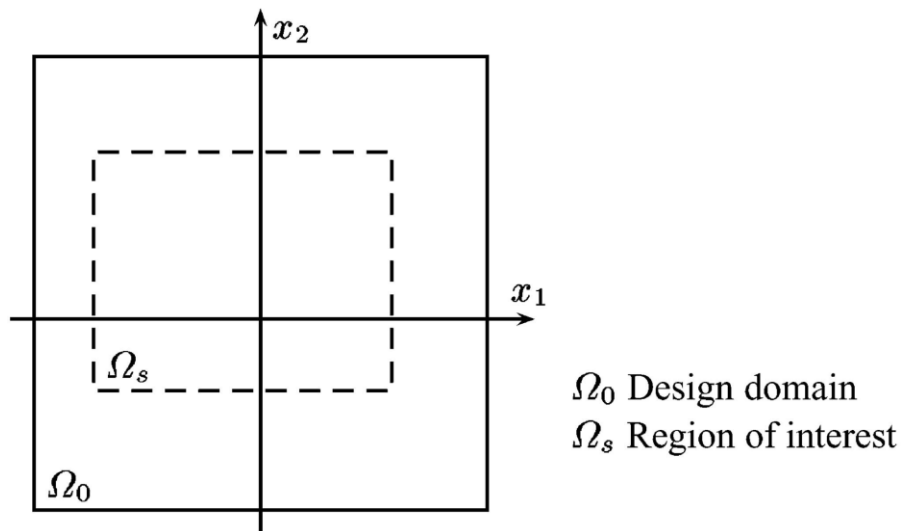


Figure 2-3 Zone d'intérêt [20]

## 2.4 Plans d'expériences

Les plans d'expériences sont des méthodes statistiques et mathématiques qui servent à pouvoir établir des liens entre les différents paramètres d'un phénomène. Dans ce travail, leur utilisation sert principalement à bâtir un modèle mathématique qui permettra d'observer l'influence des variables d'entrées sur les variables de sorties qu'on appellera « réponse ». Typiquement, on présente les plans d'expériences dans une matrice où les colonnes sont les valeurs des paramètres (variables d'entrée) et les rangées sont les différents essais. Dans le cas de plans d'expériences numériques, on utilise surtout 2 catégories : les plans de surfaces de réponses et les plans uniformes (*Space-Filling*) [21].

### 2.4.1 Plans de surface de réponses (SR)

Les plans de surface de réponses sont conçus de façon à pouvoir obtenir une surface quadratique avec un minimum d'essais. Leur principe est alors de discrétiser la zone d'intérêt en plusieurs modalités représentées par des valeurs codées  $-\alpha, -1, 0, 1, \alpha$ . Une modalité est la valeur spécifique que prend une variable lors des essais. Par exemple, pour un paramètre  $X_1 \in [a, b]$ ,  $-\alpha$  sera égal à  $a$  et  $+\alpha$  sera égal à  $b$  ainsi, la valeur 0 sera alors la moyenne des 2 bornes. La structure des plans d'expériences central composite est présentée à la Figure 2-4.

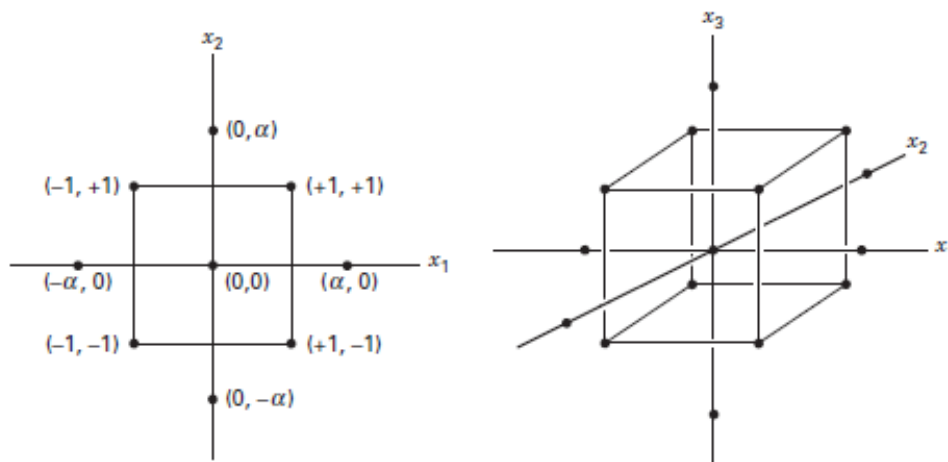


Figure 2-4 Central Composite Design [22]

Montgomery [22] suggère le plan d'expériences centrale composite (CCD) dans un cadre d'expériences numériques, c'est-à-dire lors d'un problème déterministe. Les plans à 5 modalités

offrent plusieurs points de contrôle pour la construction de la surface de réponse de second ordre. Schafer et Finke [23] proposent l'utilisation du plan D-Optimal pour un algorithme d'optimisation séquentielle appliqué à un maillage d'éléments finis.

Enfin, Unal et al. [24] ont comparé les résultats obtenus avec le CCD au plan D-optimal design dans le cadre d'optimisation séquentielle. Dans leurs conclusions, ils doutent de la nécessité d'utiliser autant de points pour le CCD, et dans une perspective de réduction du nombre d'expériences ils suggèrent le D-Optimal.

La structure de ce type de plan d'expériences permet l'obtention d'une surface de réponse quadratique à un moindre coût informatique :

$$Y_i = \beta_{i,0,0} \tag{2.1}$$

$$+ \sum_{j=1}^n \left( \beta_{i,j,0} X_j + \dots + \sum_{r=j}^n \beta_{i,j,r} X_j X_r \right) + \epsilon_i$$

où  $Y_i$  représente la réponse du phénomène étudié,  $X_i$  les facteurs étudiés,  $\beta$  les coefficients de régression,  $n$  le nombre de paramètres et  $\epsilon$  l'erreur d'approximation .

Ce type de plan offre également la possibilité d'évaluer la précision et la pertinence des paramètres par le test de Student et « P-Value ». Le « P-Value » suggère qu'un coefficient est statistiquement significatif s'il est inférieur à un seuil, comme 0.05 par exemple [22]. Ces données ne sont pas possibles pour les plans uniformes.

En plus de la modélisation du phénomène, les plans de surface de réponses sont utilisés pour des études de sensibilité des paramètres. Shafer et Finke [23] a intégré le plan D-optimal et la surface de réponse quadratique sur un modèle d'éléments finis d'une roue d'automobile. L'influence des paramètres d'entrée (ici différentes dimensions) sur la réponse (contraintes) est présentée typiquement comme à la Figure 2-5.

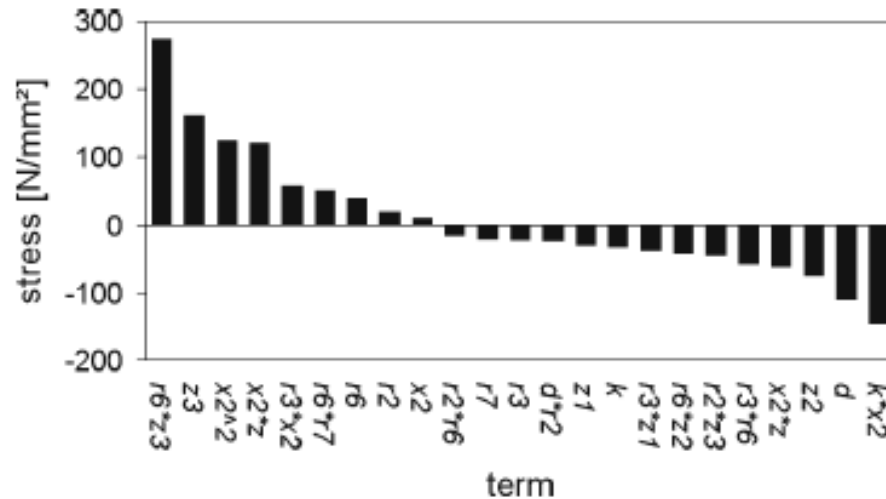


Figure 2-5 Influence des paramètres à la suite d'une modélisation de surface de réponse quadratique [23]

Enfin, les principaux avantages des plans de types surface de réponse quadratique sont [25, 26]:

- La connaissance des interactions entre les différents paramètres d'entrée (X) et de sortie (Y).
- Intégration facile aux algorithmes informatiques.
- Peu d'essais sont requis pour l'exploration de la zone d'intérêt et l'optimisation.

#### 2.4.2 Plan uniforme (*Space Filling Design*)

Dans le cas de problème déterministe, l'utilisation des plans uniformes est recommandée [26]. En fait, McKay et al. [27] ont initié le plan *Latin Hypercube* (LHC) qui consiste à répartir des points d'expériences à l'intérieur d'un espace de conception. La principale caractéristique de ce dernier est qu'il permet de discrétiser la zone d'intérêt en un nombre « n » de divisions déterminées par l'utilisateur. Ainsi, tel que présenté à la Figure 2-6, les plans uniformes présentent une distribution des points d'expériences plus uniforme dans l'espace de conception que les plans de surface de réponse.

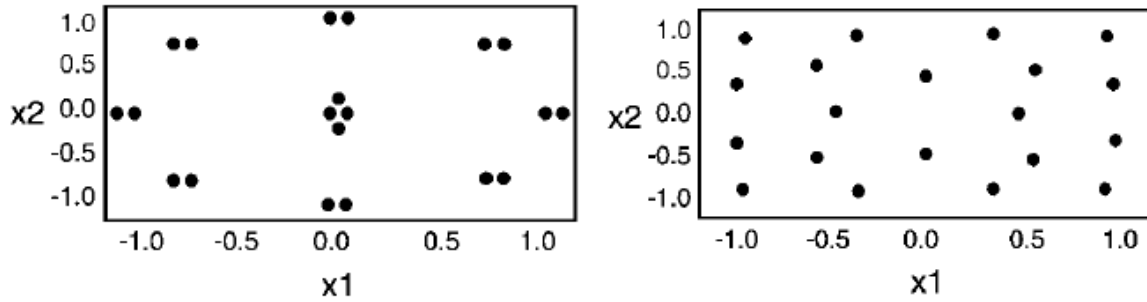


Figure 2-6 Comparaison de la couverture des plans d'expériences de type SR (gauche) et uniforme (droite)

Le plan d'expériences présenté par Mckay, dénoté comme *Random Latin Hypercube*, n'assure toutefois pas la répartition optimale des points dans l'espace. Afin de palier à ce problème, Johnson et al. [28] suggèrent une méthode *Maximin* qui assure une certaine qualité dans la répartition des points d'expériences dans l'espace. Elle consiste à maximiser la distance minimale entre n'importe quels des points du plan d'expériences. Tel que montré à la Figure 2-7, Morris [29] a intégré cette méthode au plan d'expériences Latin Hypercube et se nomme *Optimal Latin HyperCube* (OHLC).

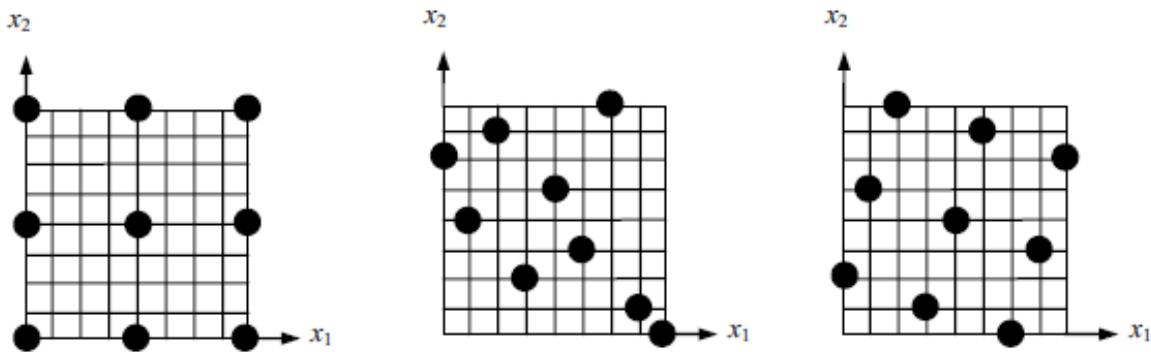


Figure 2-7 Plan d'expériences avec 2 facteurs et 9 points d'expériences : 3 modes orthogonaux (gauche), Latin Hypercube aléatoire (centre) et Latin HyperCube optimal (droite) [31]

De nouvelles méthodes permettent un échantillonnage plus efficace tel le *Quasi LHC* [30]. Elles ne sont pas encore disponibles dans les logiciels commerciaux et alourdissent l'algorithme d'optimisation puisqu'elles sont séquentielles elles-mêmes.

Enfin, les plans d'expériences uniformes sont conçus pour balayer l'espace le plus efficacement possible. Par contre, leur interpolation est beaucoup plus complexe que les plans SR [26]. En effet, la répartition des points dans l'espace des plans uniformes n'est pas ordonnée, ainsi la fonction d'interpolation repose donc sur des points aléatoires et indépendants dans l'espace.

## 2.5 Modèle d'interpolation

Le principal objectif du modèle d'interpolation est d'approximer avec un minimum d'erreur le comportement du phénomène étudié en se basant sur les résultats du plan d'expériences. Deux modèles retiennent l'attention, la fonction polynomiale et le Krigeage. La fonction polynomiale, sous sa forme quadratique, est utilisée dans les plans surface de réponse (SR). Dans un cadre d'optimisation structurelle, plusieurs auteurs présentent la méthode de Krigeage comme étant plus précis, mais plus complexe [18, 20, 26, 32, 33]. Simpson et al. [26] dans leur étude présentent un tableau qui décrit les meilleures utilisations des deux méthodes :

Tableau 2.1 Recommandations pour l'utilisation des différents modèles d'interpolation [26]

Choix de modèle	Caractéristique/utilisation
Surface de réponse	<ul style="list-style-type: none"> <li>• Facile d'utilisation</li> <li>• Bons résultats dans le cas d'expériences avec erreurs aléatoires</li> <li>• Approprié pour applications de moins de 10 facteurs</li> </ul>
Krigeage	<ul style="list-style-type: none"> <li>• Extrêmement flexible, mais complexe</li> <li>• Bons résultats dans le cas de problèmes déterministes</li> <li>• Peut être utilisé dans des problèmes de moins de 50 paramètres</li> <li>• Peu d'algorithmes commerciaux sont disponibles</li> </ul>

L'utilisation du krigeage sera préférée à celle des surfaces de réponse pour le développement de l'algorithme OSMO dans le prochain chapitre. Cette méthode a été initiée par D.C. Krige en 1951 pour évaluer les ressources d'une mine d'or. Ensuite, dans son algorithme d'interpolation

DACE, Sacks et al.[18] ont présenté l'utilisation du krigeage pour la modélisation de problèmes déterministes.

En effet, la méthode de krigeage utilisée a la propriété d'être exacte au point d'expériences. Ainsi, plus le nombre de points est élevé, plus la fonction sera précise, mais plus son calcul sera long et complexe. Cette méthode d'interpolation est en effet une combinaison linéaire de tous les points d'expériences. Chaque combinaison dépend essentiellement de la distance entre ses valeurs initiales et celles observées. L'algorithme utilisé dans ce travail est celui présenté par Lophaven & al. dans DACE (Design and Analysis of Computer Experiments) Matlab toolbox [34]. Les détails mathématiques de l'algorithme seront présentés au Chapitre 3.

### 2.5.1 Mesure de l'erreur

Lors d'une interpolation, il est important de valider la qualité du modèle. Cette étape a été incorporée à la méthode DACE établie par Sacks et al. [18]. Dans le cas de fonction polynomiale, il existe divers analyses statistiques qui peuvent fournir des informations quant à la qualité du modèle échéant : le test P-Level ou le coefficient de détermination. Sacks et al. ont proposé la méthode des moindres carrés intégrée moyenne (IMSE) pour évaluer l'erreur :

$$IMSE = \int_x MSE[\hat{y}(x)]\phi dx \quad (2.2)$$

Dans le krigeage, puisque la fonction est exacte aux points d'expériences, il n'est donc pas possible d'obtenir l'erreur directe de la fonction sur les points. Kleijnen [35] propose une méthode de validation croisée. Cette méthode suggère que pour chaque point d'expériences, on calcule la réponse  $\hat{y}_i$  du modèle d'interpolation à ce point sans celui-ci dans le plan d'expériences. Ensuite, puisque  $y_i$  est connu il est possible d'obtenir l'erreur estimée à ce point. Cette méthode sera discutée dans le prochain chapitre.

## 2.6 Amélioration séquentielle

Les éléments discutés précédemment sont essentiellement propres à chaque cycle. Si la précision et la solution n'atteignent pas la satisfaction d'un critère d'arrêt imposé par l'utilisateur, il doit y avoir une modification du plan d'expériences. Plusieurs types d'améliorations séquentielles sont présentées dans la littérature. La stratégie de déplacements des limites (MLS – *Moving Limit*

*Strategy*) consiste à définir une nouvelle zone d'intérêt à chaque itération afin de trouver la solution optimale. Les méthodes de remplissage (SF – *Space Filing*) eux, conservent le plan d'expériences, mais lui ajoutent des points afin de diminuer l'erreur d'interpolation. Les méthodes présentées dans la littérature traitent seulement des problèmes mono-objectifs.

### 2.6.1 Stratégie de déplacement des limites (*Move Limit strategy - MLS*)

Cette méthode consiste essentiellement à déplacer les limites de la zone d'intérêt à chaque itération dans un but de cerner la solution optimale. Jacobs et al.[19], Kim et Lee [36] ont utilisé cette approche dans leur algorithme. Kim et Lee l'ont appliqué au problème académique étudié du chapitre 4, ses résultats seront donc comparés à l'algorithme OSMO.

Les études de Bonte révèlent toutefois que les méthodes des MLS présentent des désavantages vis-à-vis des autres méthodes [32] :

- La solution n'est pas triviale lorsqu'il existe plusieurs optimums locaux.
- La méthode est moins efficace que le remplissage puisqu'elle ne conserve pas l'historique des itérations antérieures dans le cas où la solution optimale se trouve à l'extérieur de la nouvelle zone d'intérêt.

La programmation de ces méthodes est également complexe. En effet, elle est inspirée des méthodes d'optimisation à région de confiance. Ainsi, étant donné leur complexité et leur efficacité, ces méthodes ne sont pas retenues pour ce travail. Leurs détails mathématiques ne seront donc pas présentés.

### 2.6.2 Méthode de remplissage (*Space filling - SF*)

Cette méthode fonctionne à la fois pour le krigeage ou une série polynomiale. La philosophie de ces méthodes consiste à conserver le plan d'expériences initiales et d'y ajouter des points d'expériences afin d'y augmenter la précision de la fonction d'interpolation. Une méthode retenue est la minimisation d'une fonction de mérite. Cette méthode présentée par Bonte et al. [32] consiste à minimiser une fonction de mérite avec plusieurs points de départ afin d'y repérer les points qui sont le plus susceptibles d'être proches de la solution optimale.



La fonction de mérite prend donc la forme suivante [32] :

$$f_{merite}(x) = \hat{y}(x) - w * \hat{s}(x) \quad (2.3)$$

Où  $\hat{y}$  et  $\hat{s}$  sont respectivement la fonction interpolée et la variance au moindre carré;  $w$  est le poids.

Au sens physique, la variance des moindres carrés représente la probabilité d'erreur sur la surface. Ainsi, au point d'expériences cette variance est nulle et entre 2 points elle est maximale.

La Figure 2-8 présente typiquement la fonction de mérite, la fonction de krigeage et la variance aux moindres carrés.

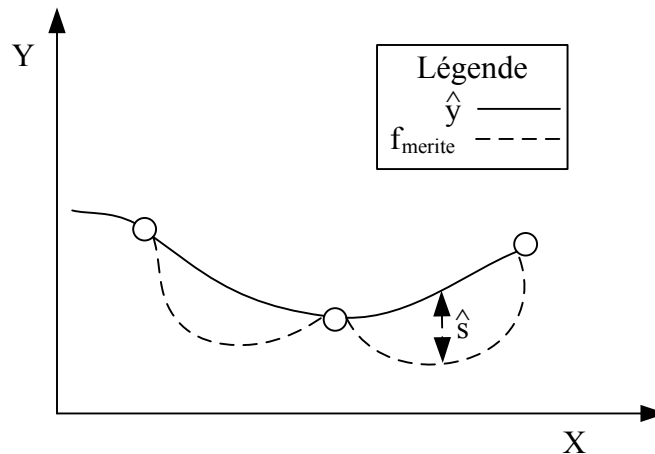


Figure 2-8 Minimisation d'une fonction de mérite

En observant la figure, on remarque si le poids tend vers  $\infty$ , la fonction de mérite sera largement inférieure à la fonction de krigeage entre les points d'expériences. Toutefois, s'il tend vers 0, le point d'expériences minimum sera ajouté au plan d'expériences puisque cela est l'équivalent de minimiser la fonction de krigeage. Les détails mathématiques concernant le calcul de la variance sera présenté au prochain chapitre.

## CHAPITRE 3 ALGORITHME D'OPTIMISATION SÉQUENTIELLE MULTI-OBJECTIFS (OSMO)

Ce chapitre décrit l'algorithme d'optimisation séquentielle multi-objective. Il explique son fonctionnement et les choix associés. Le principal objectif de l'algorithme OSO est de générer un ensemble de solutions optimales avec précision basées sur un nombre de calculs (éléments finis) minimal. Il faut spécifier qu'étant données les ressources disponibles, l'algorithme OSO a été conçu pour être utilisé avec la suite d'Altair Hyperworks V10 d'analyse et optimisation par éléments finis.

### 3.1 Algorithme OSO

L'algorithme OSO a été créé pour résoudre des problèmes structurels multi-objectifs. Essentiellement, la construction des problèmes multi-objectifs à résoudre a cette forme [41]:

$$\begin{aligned} \min_x \quad & F(x) = [f_1(x), f_2(x), \dots, f_m(x)] \\ \text{su} \text{jet à} \quad & G(x) = [g_1(x), g_2(x), \dots, g_j(x)] \geq 0 \\ & H(x) = [h_1(x), h_2(x), \dots, h_k(x)] = 0 \\ & x_i^L \leq x_i \leq x_i^U, i = 1, 2, \dots, n \end{aligned}$$

où  $n$  est le nombre de variable,  $m$  le nombre de fonctions objectives,  $j$  et  $k$  le nombre des contraintes d'inégalité  $G$  et d'égalité  $H$ ,  $x_i^L$  et  $x_i^U$  sont respectivement les bornes des variables  $x_i$ .

De plus étant donné que l'optimisation séquentielle est basée sur une fonction d'interpolation, il est nécessaire de définir un critère d'arrêt d'erreur relative minimale. On considéra donc l'algorithme convergé lors que son erreur sera inférieure à un seuil  $\varepsilon$ .

### 3.2 Schéma général de l'algorithme

L'algorithme OSO se divise en 2 phases : Exploratoire et Recherche. La phase exploratoire consiste essentiellement à explorer l'espace de conception afin d'y déterminer une zone d'intérêt fiable et adéquate pour l'optimisation séquentielle. La seconde phase consiste à rechercher une ou des solutions optimales à l'intérieur de cette même zone.

La question qui peut se poser c'est sur l'utilité d'ajouter une phase exploratoire aux algorithmes présentés dans la littérature [19, 23, 32]. L'optimisation séquentielle nécessite la définition d'une zone d'intérêt adéquate au problème. Lorsque la sensibilité des paramètres de même que les connaissances sur le phénomène sont limitées, il est difficile d'établir une zone d'intérêt qui saura respecter l'ensemble des contraintes du problème. Cette phase sert essentiellement à vérifier de façon statistique si la zone d'intérêt est adéquate.

Ensuite, la phase recherche est inspirée de l'algorithme présenté par Bonte et al [32]. L'ajout d'une étape de minimisation de l'erreur est jugé nécessaire tel que présenté par Sacks et al. [18]. Elle permet d'obtenir une meilleure sensibilité de la fonction de krigeage sur les différents points d'expériences.

La phase exploratoire consiste à 2 étapes principales :

1. Paramétriser le modèle.
2. Réaliser le plan d'expériences.
3. Effectuer les simulations afin d'obtenir la réponse du système pour chaque point.

La majorité du temps requis pour l'exécution de l'algorithme est dépensée dans la phase d'exploration. Chaque itération est très couteuse en temps de calcul. C'est donc pourquoi une paramétrisation du modèle et un choix des bornes doivent être faits avec attention. Cette phase est toutefois nécessaire puisqu'elle constitue la base de l'algorithme d'optimisation séquentielle. À chaque itération, une modification des limites de la zone d'intérêt est nécessaire afin d'obtenir un plan d'expériences convenable pour la phase recherche.

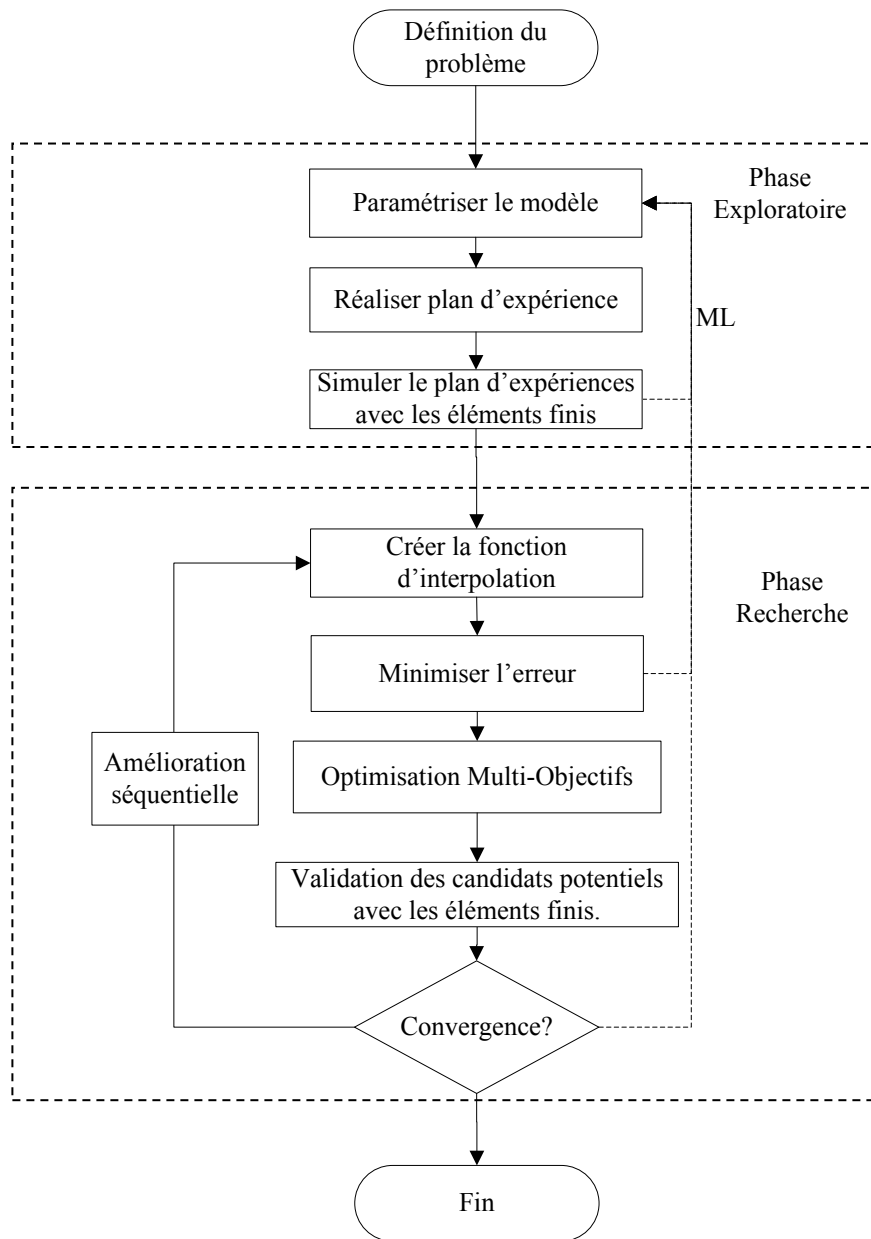


Figure 3-1 Schéma général de l'algorithme OSMO

La phase recherche vise à atteindre 2 objectifs : diminuer l'erreur d'interpolation dans la région optimale et trouver ces solutions optimales. Pour y arriver, les étapes ci-dessous sont nécessaires :

1. Créer le modèle d'interpolation.
2. Minimiser l'erreur.
3. Optimiser le modèle d'interpolation.
4. Effectuer une simulation avec les paramètres des candidats retenus.

5. Si la précision demandée est respectée et que la solution optimale converge, arrêter l'algorithme. Sinon, effectuer une amélioration séquentielle selon la méthode de remplissage par candidats potentiels établie dans ce chapitre.

### 3.3 Paramétrisation du modèle et détermination des bornes

L'utilisation de l'algorithme OSMO implique que la topologie du problème d'optimisation est déterminée. Elle impose également que le design structurel à optimiser soit paramétré et fonctionnel. Dominique [37] présente une démarche de paramétrisation appropriée pour un algorithme d'optimisation de design structurel (Figure 3-2). Cette démarche est utilisée pour définir l'espace de conception de l'algorithme OSMO.

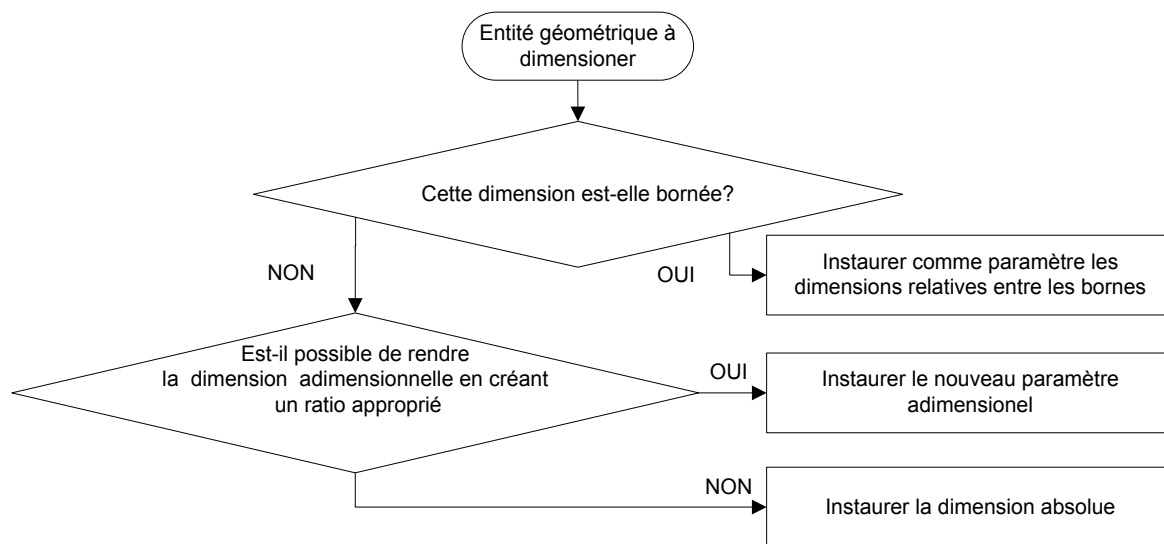


Figure 3-2 Création d'une paramétrisation appropriée pour un algorithme d'optimisation de design structurel.

La Figure 3-3 présente un exemple de la paramétrisation d'un assemblage par interférence. Dans l'exemple, le paramètre X1 et X3 en pourcentage relatif entre leur borne minimale (X1) ou maximale (X3) et la valeur du paramètre X2. De cette façon, on évite les modèles dysfonctionnels dû à une modification des valeurs des paramètres.

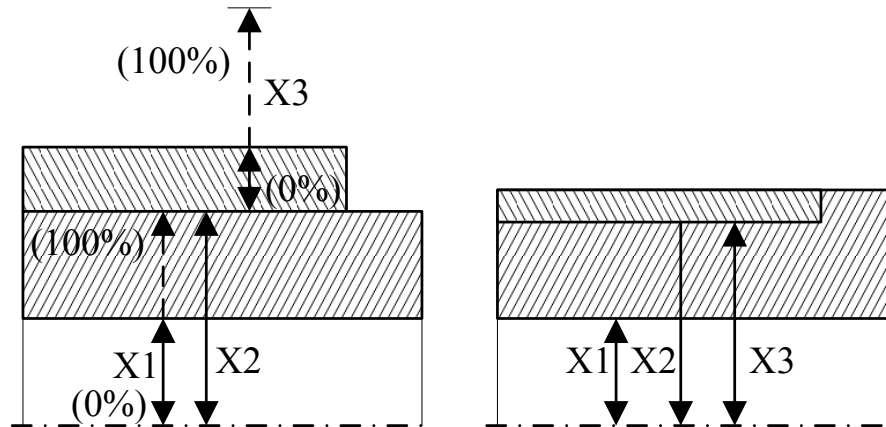


Figure 3-3 Paramétrisation d'un assemblage par interférence : Bonne paramétrisation (gauche), Modèle disfonctionnel (droite)

### 3.4 Définition de la zone d'intérêt

La zone d'intérêt (ZI) doit être comprise à l'intérieur de l'espace de conception. Initialement, il n'existe pas de méthodes permettant de définir la zone d'intérêt. Pour la première itération, celle-ci sera donc de même dimension que l'espace de conception. Si après une analyse statistique des valeurs obtenues, une majorité des solutions se trouve à l'extérieur des critères de conceptions imposés, une réduction des limites de la zone d'intérêt s'imposera telle que présentée à la section 3.9.4.

### 3.5 Plan d'expériences

Deux types de plans d'expériences ont été étudiés pour l'algorithme OSMO. Le premier est le CCD (*central Composite Design*) et le second, le OHLC (*Optimal Hyper Latin Cube*). Les CCD ont le désavantage d'imposer un nombre d'essais selon le nombre de paramètres. Le Tableau 3.1 présente le nombre de points d'expériences par paramètre que requiert le plan CCD selon le logiciel Altair Hyperworks V10.

Tableau 3.1 Nombre de points d'expériences en fonction du nombre de paramètres.

Nombre de paramètres	Nombre de points d'expériences (CCD)	Nombre de points d'expériences (OLHC)
2	9	20
3	15	30
4	25	40
5	43	50
6	77	60
7	143	70
8	273	80
9	531	90

Pour le OLHC, il est suggéré d'utiliser un nombre de points d'expériences de 10 fois le nombre de paramètres [32]. L'utilisation du plan d'expériences OHLC devient profitable lorsque le nombre de paramètres est supérieur à 6.

Pour OSMO, le plan d'expériences utilisé sera le OLHC de Altair Hyperworks V10. Le logiciel effectue la minimisation de la distance maximale entre chaque point d'expériences selon la méthode *Maximin* présentée au chapitre précédent.

### 3.6 Modèle d'interpolation

Pour assurer l'interpolation des résultats du plan d'expériences, la méthode du krigeage est utilisée. Celle-ci est présentée dans le *DACE toolbox* intégré dans le logiciel Matlab [34]. Une première partie de cette section décrira la partie mathématique de la méthode de krigeage. Dans une seconde partie, l'écriture informatique de la méthode dans le module *Hypermath* d'Altair Hyperworks sera présentée.

#### 3.6.1 Méthode de krigeage dans *DACE Toolbox*

La méthode présentée ici est reprise de Lophaven et al. [34]. Cette méthode de krigeage offre la possibilité d'utiliser différents modèles de régression et de corrélation. Des auteurs ont appliqué les modèles à différents problèmes d'optimisation structurelle et ce sont le modèle linéaire et gaussien qui ont offert les meilleures performances [32, 34]. Le Tableau 3.2 présente les paramètres requis pour la modélisation de la fonction de krigeage;  $n$  représente le nombre de paramètres et  $m$  le nombre de points d'expériences.

Tableau 3.2 Paramètres de krigeage

Paramètres	Dimensions	Définitions
S	n x m	Matrice contenant les points d'expériences
Y	m	Vecteur de réponse des points d'expériences
$\theta$	n	Paramètres de corrélation

Tout d'abord, afin d'adimensionaliser les paramètres, les données S et Y doivent être soumises aux conditions de normalisation, c'est à dire :

$$\begin{aligned} \mu[s_{[1,m],j}] &= 0, \quad V[s_{[1,m],j}, s_{\{1,m\},j}] = 1, \quad j = 1, \dots, n \\ \mu[Y] &= 0, \quad V[Y, Y] = 1, \end{aligned} \quad (3.1)$$

où  $\mu$  et  $V$  sont respectivement la moyenne et la covariance.

Tel que présenté par Sacks (1989), le modèle de krigeage est la somme d'un modèle de régression et une fonction de covariance généralisée  $Z(x)$  :

$$Y(x) = \mathcal{F}(\beta, x) + Z(x) \quad (3.2)$$

où  $\mathcal{F}$  représente le modèle de régression,  $\beta$  est un paramètre de régression et  $x \subseteq \mathbb{R}^n$ .

### 3.6.1.1 Choix du modèle de régression et de corrélation

Le modèle de régression est en effet la combinaison linéaire d'un nombre de  $n+1$  fonctions  $f_j: \mathbb{R}^n \rightarrow \mathbb{R}$

$$\begin{aligned} \mathcal{F}(\beta', x) &= \beta_1 f_1(x) + \dots + \beta_p f_p(x) \\ &= [f_1(x) \dots f_{n+1}(x)] \beta \\ &= f(x)^T \beta. \end{aligned} \quad (3.3)$$

Celui utilisé dans ce travail est linéaire :

$$f_1(x) = 1, \quad f_2(x) = x_1, \dots, \quad f_{n+1} = x_n. \quad (3.4)$$



Tel que mentionné, le choix du modèle de corrélation est celui de Gauss. Le modèle est une fonction dépendante du paramètre de corrélation et de deux points d'expériences quelconque, nommé ici pour le besoin  $w$  et  $z$ . Ces vecteurs variables sont de dimensions  $n$ . L'équation du modèle de corrélation s'écrit comme suit :

$$\mathcal{R}(\theta, w, z) = \prod_{j=1}^n \mathcal{R}_j(\theta, w_j - z_j) \quad (3.5)$$

$$\text{où } \mathcal{R}_j(\theta, d_j) = e^{-(\theta_j d_j^2)} \text{ et } d_j = w_j - z_j.$$

### 3.6.1.2 Fonction de Krigeage

Selon Sacks[18] et Lophaven [34] l'équation de krigeage s'écrit ainsi (les termes sont définis aux équations 3.7 à 3.9, et à la section 3.6.1.3 pour  $\beta^*$  et  $\gamma^*$ ) :

$$\hat{y}(x) = r^T R^{-1} Y - (F^T R^{-1} r - f) \beta^* \quad (3.6)$$

$$\hat{y}(x) = f^T \beta^* + r^T R^{-1} (Y - F \beta^*)$$

$$\hat{y}(x) = f(x)^T \beta^* + r^T(x) \gamma^*,$$

où la matrice  $f(x)$  est définie à l'équation 3.4, la matrice  $R$  est la distance entre les différents points d'expériences :

$$F_{ij} = f_j(s_i) \quad (3.7)$$

$$F = [f(s_1) \dots f(s_m)]^T$$

$$R_{ij} = \mathcal{R}(\theta, s_i, s_j) \quad i, j = 1 \dots m \quad (3.8)$$

$$r^T(x) = [\mathcal{R}(\theta, s_1, x) \dots \mathcal{R}(\theta, s_m, x)]^T. \quad (3.9)$$

### 3.6.1.3 Calcul des coefficients $\beta^*$ et $\gamma^*$

Les coefficients  $\beta^*$  et  $\gamma^*$  sont calculés à partir des données du plan d'expériences. Il n'est donc pas nécessaire de les calculer à chaque évaluation de la fonction mais plutôt lorsque le paramètre  $\theta$ , la matrice  $S$  et  $Y$  sont modifiés. Leur expression s'écrit ainsi :

$$\beta^* = (F^T R^{-1} F)^{-1} F^T R^{-1} Y \quad (3.10)$$

$$\gamma^* = R^{-1}(Y - F\beta^*). \quad (3.11)$$

Lophaven suggère d'utiliser la factorisation de Cholesky ( $C$ ) et décomposition  $QR$  pour calculer ces facteurs afin d'éviter des problèmes de conditionnement de matrice et ainsi alléger les calculs. Tout d'abord puisque  $R$  est symétrique, il est possible d'effectuer la factorisation de Cholesky :

$$R = CC^T \quad (3.12)$$

Ensuite, en supposant que :

$$\tilde{F}\beta = \tilde{Y} \text{ avec } C\tilde{F} = F \text{ et } C\tilde{Y} = Y. \quad (3.13)$$

Avec la décomposition  $QR$  de  $\tilde{F}$ , voir équation 3.14, si la matrice  $G$  n'est pas inversible, cela implique que le modèle de régression n'est pas bien conditionné. Dans ce cas, il faut donc arrêter le calcul et choisir un nouveau plan d'expériences:

$$\tilde{F} = QG^T. \quad (3.14)$$

En combinant 3.13 et 3.14, on trouve :

$$G^T\beta^* = Q^T\tilde{Y}. \quad (3.15)$$

Avec 3.15 on peut déterminer  $\gamma^*$  :

$$\gamma^* = (\tilde{Y} - \tilde{F}\beta^*)^T C^{-1} \quad (3.16)$$

### 3.6.2 Algorithme pour le calcul de krigeage avec *HyperMath*

L'algorithme d'interpolation par krigeage se divise en 2 parties : la modélisation de la fonction et l'estimation de la réponse. Le premier programme consiste essentiellement à déterminer les coefficients  $\beta^*$  et  $\gamma^*$  tandis que le second estime la réponse en fonction de  $x$ .

Le temps de calcul des coefficients  $\beta^*$  et  $\gamma^*$  représentent plus de 95% de l'évaluation de la fonction de krigeage. Lors de l'étape de l'optimisation, l'algorithme aura à évaluer plusieurs fois la fonction de krigeage. Puisqu'on sait que le calcul de ceux-ci est indépendant de  $x$ , un programme *generation\_donnees()* a été écrit afin de calculer ces coefficients et les stocker sous forme de fichier binaire Matlab. Ainsi, un programme *estimation()* a pour rôle d'évaluer la fonction, avec les coefficients  $\beta^*$  et  $\gamma^*$  calculé précédemment, en fonction de  $x$ . Ce programme est couplé à l'algorithme d'optimisation et a pour rôle d'estimer la réponse  $\hat{y}(x)$  sous les conditions présentées à la section précédente.

### 3.7 Optimisation du paramètre de corrélation $\theta$ par la méthode de validation croisée

Le paramètre de corrélation  $\theta$  a une importance prépondérante sur la précision du modèle. Puisque la fonction de krigeage est exacte aux points d'expériences, Kleijnen [35] suggère l'utilisation de la validation croisée pour approximer l'erreur entre la fonction d'interpolation pour des points ne faisant pas partie du plan d'expériences et les valeurs réelles de réponse. Ensuite, la minimisation de cette erreur permet de trouver le paramètre de corrélation optimal.

La Figure 3-4 montre la nécessité d'une telle étape pour assurer une interpolation avec précision. En effet, la méthode a été appliquée à une fonction quelconque (3.17) créée pour montrer l'influence du paramètre  $\theta$  sur l'interpolation.

$$Y(x) = (4 \sin 3x + 3 \sin 11x) * (15 - x^3) + x^4 - (2x)^3 + (2x)^2 \quad (3.17)$$

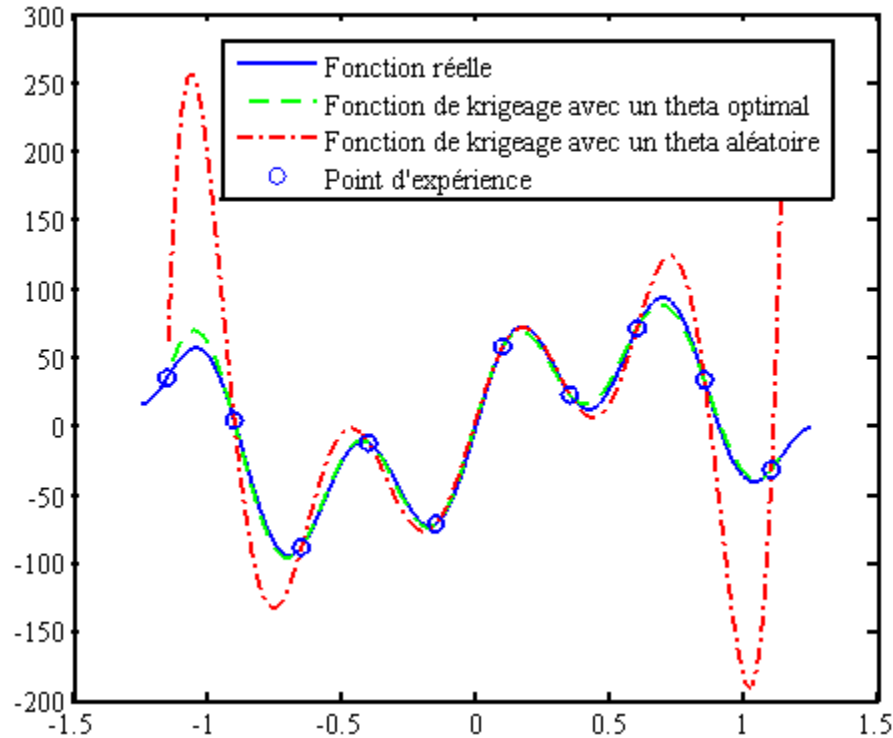


Figure 3-4 Comparaison entre  $\theta$  initial et celui obtenu par la validation croisée

### 3.7.1 Méthode de validation croisée

La méthode de validation croisée est un processus itératif. Ainsi, les étapes présentées ci-dessous sont à répéter de  $i=1..n$  où  $n$  est le nombre de points d'expériences :

- 1) Affecter à  $x_i$  les valeurs du point d'expériences  $S_i$  et à  $y_i$  la réponse  $Y_i$
- 2) Enlever à la matrice  $S$  (des points d'expériences) le vecteur  $S_i$  et à la matrice  $Y$  la valeur  $Y_i$
- 3) Calculer les valeurs  $\beta_i^*$  et  $\gamma_i^*$ . pour les nouvelles matrices,  $S$
- 4) Calculer  $\hat{y}_i(x_i)$

De cette façon, il est donc possible d'approximer l'erreur au moindre carré moyenne provenant de la validation croisée :

$$RMSE_{vc} = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i(\theta) - y_i)^2 \quad (3.18)$$

### 3.7.2 Programmation de la méthode de validation croisée

La fonction *evaluation\_erreur()* dépendante de  $\theta$  a été créée. Ce programme évalue l'erreur par la méthode de validation croisée en fonction du vecteur  $\theta$ , puisque chaque évaluation de cette fonction doit faire  $n$  calculs de fonction de krigeage. Afin de réduire le temps de minimisation de l'erreur des moindres carrée, le calcul de la matrice de distance entre chaque site pour les  $n$  itérations et le stockage dans un fichier *d.mat* sont faits dans un programme indépendant. Pour chaque itération  $i$  de la méthode, on a donc une matrice  $d$  contenant les distances entre les différents points d'expériences normalisées. Le fichier *d.mat* contient donc  $n$  matrices  $d$  :

$$d = s_i - s_j, \quad i, j = 1 \dots n - 1 \quad (3.19)$$

Le simple fait de calculer la matrice  $d$  à part du programme réduit le temps de calcul d'un facteur d'environ 100 pour un total de 0.34 seconde par évaluation. La Figure 3-5 montre le diagramme du programme d'évaluation de l'erreur par la méthode de validation croisée.

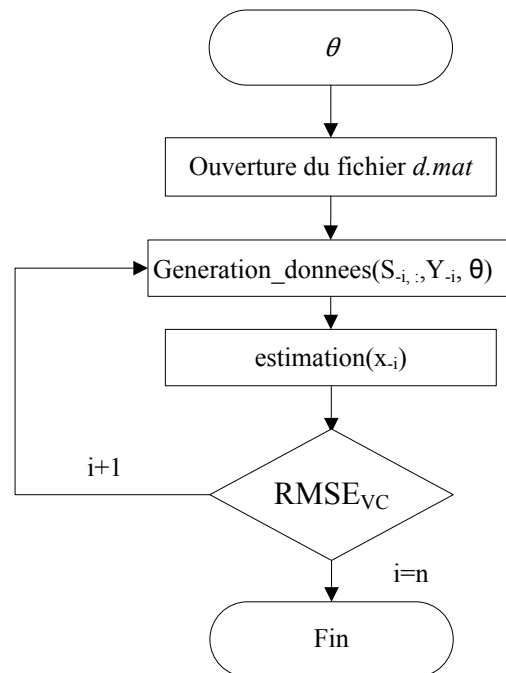


Figure 3-5 Programme d'évaluation de l'erreur *evaluation\_erreur()*

### 3.7.3 Minimisation de l'erreur obtenue par validation croisée

L'équation de calcul de  $RMSE_{VC}$  permet d'évaluer l'erreur d'interpolation par la méthode de validation croisée. En adoptant la minimisation de cette erreur, il est possible d'obtenir une fonction de krigeage adéquate. Dans un but d'utiliser un même logiciel pour effectuer l'algorithme OSMO, le solveur d'optimisation préconisé est l'algorithme génétique (GA) de Altair Hyperworks V10. Les paramètres du solveur sont toujours propres à chaque problème, toutefois, selon les différents essais effectués, les réglages par défaut offrent de bons résultats. Lorsque la minimisation de l'erreur génère des valeurs des composantes de  $\theta$  qui tendent vers l'infini ou 0, c'est souvent un signe d'une mauvaise corrélation de la fonction de krigeage. Ainsi, une fois le paramètre  $\theta$  connu, il est possible d'effectuer l'optimisation multi-objective sur la fonction de Krigeage.

## 3.8 Optimisation Multi-objectifs

Les fonctions de krigeage sont souvent appropriées à saisir un comportement non-linéaire présentant plusieurs minimas locaux. Dans ce cas, les algorithmes génétiques offrent de meilleurs résultats que les algorithmes de type gradient qui dépendent plutôt du point de départ. Pour OSMO, le solveur utilisé pour effectuer l'optimisation sur les fonctions de krigeage est l'algorithme génétique multi-objectifs (MOGA) intégré dans Altair Hyperstudy V10.

Le rôle de l'optimisation multi-objectifs consiste à générer un ensemble de solutions optimales. L'algorithme MOGA utilise la stratégie « *non-dominée* ». On dit qu'une solution est non-dominée lorsque qu'aucune amélioration ne peut être faite sur un objectif sans y dégrader une des autres. L'ensemble des solutions non-dominées forment le front de Pareto. Pour obtenir des résultats de qualité, ils doivent satisfaire deux critères [38] :

1. Les solutions convergent vers le front de Pareto.
2. Les solutions sont dispersées tout le long du front de Pareto.

En effectuant différents essais sur les problèmes d'optimisation présentés dans ce travail, les paramètres par défaut ont donné de bons résultats dans la majorité des cas. Toutefois, lorsqu'il y a plusieurs objectifs, pour obtenir une meilleure diversité dans les solutions obtenues, le nombre de points non-dominés peut être augmenté. D'autre part, pour permettre une meilleure recherche,

il est possible d'augmenter la taille de la population jusqu'à l'atteinte d'un seuil où la celle-ci n'améliora plus la qualité des résultats. Toutefois, dans les deux cas, l'augmentation du nombre de candidats augmentera également le temps de calculs. Les valeurs des options utilisées pour l'algorithme MOGA sont présentées dans le Tableau 3.3. Avec plusieurs essais, ce sont ces valeurs qui offraient les meilleurs résultats.

Tableau 3.3 Options du solveur MOGA.

Paramètre	Valeur
MOGA	
Nombre d'itérations max.	50
Nombre d'itérations min.	25
Points non-dominés	500
Tolérance de violation des contraintes	0.5%
GA	
Taille de la population	Automatique
Valeur	-
État discrète	1024
Ratio de mutation	0.01
Recherche globale	2
% population élite	10%
Initialisation aléatoire	1
Nombre de prétendants	2
Multiplicateur de pénalité	2.0
Puissance de pénalité	1

### 3.9 Amélioration séquentielle

Puisque l'optimisation repose sur l'interpolation d'un plan d'expériences, il arrive que l'erreur de cette fonction soit élevée. Pour pouvoir pallier à ce problème, on doit prévoir une amélioration séquentielle qui vise essentiellement à diminuer cette erreur. Pour l'algorithme OSMO, l'amélioration séquentielle à la phase exploratoire est basée sur une méthode statistique qui consiste à modifier les bornes de la zone d'intérêt en fonction de la conformité des solutions. Cette méthode est couteuse en temps de calcul, elle est donc seulement utilisée en phase exploratoire afin d'offrir un plan d'expériences adéquat à la phase recherche.

Ensuite, pour la phase recherche, le type d'améliorations séquentielles retenu est une méthode de remplissage. Celle-ci consiste essentiellement à trouver la solution optimale dans la zone d'intérêt définie lors de la phase exploratoire.

### 3.9.1 Critère d'arrêt l'algorithme

On considère l'algorithme convergé lorsque la variation de celui-ci est inférieure à un critère d'arrêt établi initialement. Pour le cas de l'algorithme OSMO, ce critère est l'erreur d'interpolation entre la fonction d'interpolation et le modèle numérique. Elle est valable autant pour les fonctions objectives que pour les contraintes. En effet, si l'erreur d'interpolation sur les contraintes est trop élevée, il n'est pas possible de savoir si la solution respecte les critères du problème.

Si le problème est mono-objectif, cette erreur est l'écart absolu entre la solution approximée et sa valeur réelle obtenue par le modèle numérique. Lors qu'il est multi-objectifs, on calcule cette erreur avec l'écart absolu moyen.

### 3.9.2 Méthode statistique de modification des limites

Cette technique consiste à modifier les limites de la zone d'intérêt, toutefois elle ne repose pas sur les fondements mathématiques des stratégies de déplacement de limites présentées au chapitre précédent. La méthode suggérée ici repose plutôt sur l'analyse statistique de la conformité des contraintes de la zone d'intérêt. En effet, l'optimisation séquentielle génère, à travers les étapes, ces informations qui serviront à ces analyses. Le Tableau 3.4 présente les informations obtenues indiquant un mauvais choix de zone d'intérêt.

Tableau 3.4 Information sur le modèle d'interpolation

Étape	Observations	Résultat
Plan d'expériences	Valeur qui ne respecte pas les contraintes	Forte variation dans la fonction d'interpolation.
Modèle d'interpolation (minimisation de l'erreur)	Erreur trop élevée	Fonction d'interpolation qui ne représente pas adéquatement le phénomène
Optimisation	Solutions concentrées sur une borne de la zone d'intérêt.	La solution optimale peut se trouver à l'extérieur de la zone d'intérêt.



Pour la réalisation de l'algorithme OSMO, il a été suggéré à la Section 2.3 de définir une zone d'intérêt équivalente à l'espace de conception et de la réduire afin de mieux cibler la zone des solutions optimales. La modification des limites peut s'effectuer à la phase recherche et à la phase exploratoire. En ce sens, l'approche est différente. Avant d'aborder le sujet, l'outil statistique utilisé pour effectuer l'analyse sera défini pour en faciliter la compréhension.

### 3.9.3 Définition de la boîte à moustache

Les boîtes à moustache sont des outils statistiques qui servent à étudier la dispersion des données. Celles-ci consistent à afficher : une boîte entre le premier (Q1) et troisième quartile (Q3) , une ligne médiane (M) et les valeurs extrêmes. Toutefois, les données dépassant une fois l'écart interquartile y sont affichées individuellement. L'écart interquartile est la différence entre la valeur du 1<sup>er</sup> et troisième quartie. L'équation 3.21 présente ce calcul et la Figure 3-6 illustre une boîte à moustache. Cet outil est disponible avec le logiciel MATLAB.

$$EIQ = Q3 - Q1 \quad (3.20)$$

Pt marginal > 1.5 EIQ → X

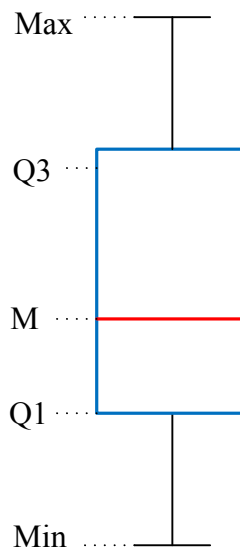


Figure 3-6 Boîte à moustache

### 3.9.4 Modification des limites (phase exploratoire)

Dans l'algorithme OS MO, la modification des limites de la zone d'intérêt se fait à partir d'analyses statistiques. À la phase d'exploratoire, après avoir obtenu les résultats  $Y$  de la réalisation du plan d'expériences, il est nécessaire de s'assurer que ceux-ci répondent en majorité aux critères de conception avant d'amorcer la phase recherche. Il a été observé que lorsque le plan d'expériences contient une majorité de points qui ne respecte pas les contraintes, la fonction d'interpolation est moins représentative et donc, la qualité des résultats est affectée.

Ainsi, on suggère de valider la zone d'intérêt par une analyse de dispersion des valeurs normalisées des variables. La normalisation des variables s'effectue tel que :

$$X_{norm} = \frac{X - X_{inf}}{X_{sup} - X_{inf}} \quad (3.21)$$

où  $X_{sup}$  et  $X_{inf}$  représentent les bornes supérieure et inférieure de la zone d'intérêt.

Pour détecter les régions de la zone d'intérêt qui ne respectent pas les contraintes, les diagrammes de boîtes à moustache ont été utilisés.

Le principe de modification de la zone d'intérêt consiste à répartir tous les points d'expériences selon deux catégories : ceux qui respectent la contrainte et ceux qui ne respectent pas la contrainte. Ensuite, en affichant les diagrammes à moustache de ces 2 catégories pour chaque variable  $X$ , il est possible d'observer si des régions de la zone d'intérêt peuvent être réduites afin que l'ensemble des points respecte la contrainte.

Un exemple de cette démarche est présenté à la Figure 3-7, il traite d'un problème type d'optimisation d' $Y$  soumis à une contrainte. On qualifie un point d'expériences de conforme lorsqu'il respecte la contrainte à l'étude.

Sur cette figure, pour la variable  $X_1$  on remarque :

- Aucun point d'expériences dont ses valeurs  $X_1$  sont supérieures à 0.6 n'est conforme.
- Tous les points d'expériences dont ses valeurs  $X_1$  sont inférieures à 0.4 sont conformes.
- Pour  $X_1 \in [0.4, 0.6]$ , autant les points d'expériences peuvent être conformes ou non-conformes.

Pour la variable X2, on remarque que autant les points d'expériences peuvent être conforme que non-conforme pour les bornes définies de X2. Il est donc impossible de relimiter sa borne avec les informations à disposition. Dans ce cas, il est donc possible de croire que la contrainte dépend beaucoup moins de la variable X2 que de la variable X1.

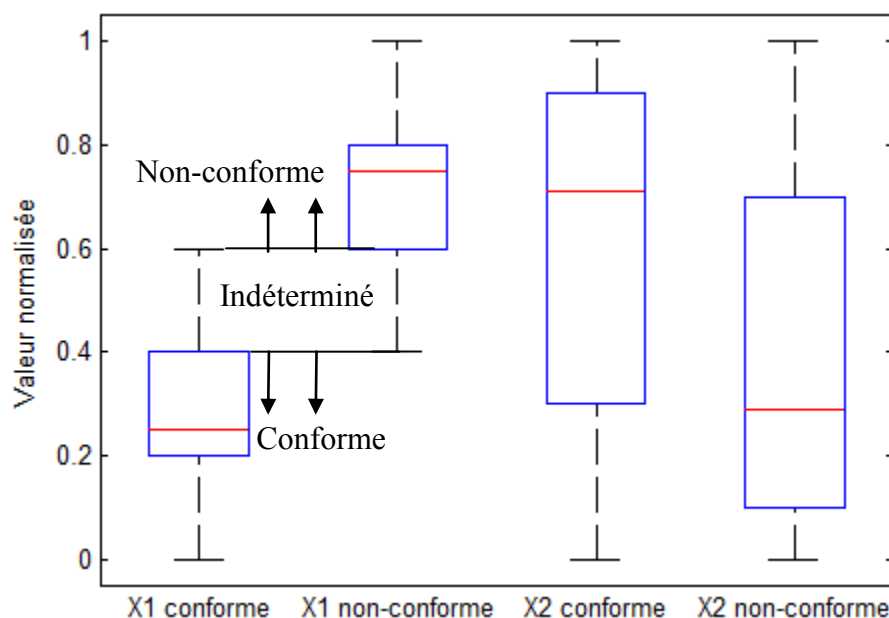


Figure 3-7 Vérification d'une contrainte des variables X1 et X2

### 3.9.5 Modification des limites (phase recherche)

Tel que présenté à la Figure 3-1, le schéma général de l'algorithme OSMO, il arrive qu'après l'obtention des résultats de l'optimisation multi-objectifs, on remarque que la zone d'intérêt n'est pas adéquate. Généralement à la phase recherche, il n'est pas souhaitable de modifier les limites de la zone d'intérêt puisque cela engendre des calculs supplémentaires. Par contre, lorsque les solutions semblent se trouver à l'extérieur de la zone d'intérêt ou que l'erreur est jugée très élevée, il est possible de réduire les limites de la zone d'intérêt. Ainsi, l'augmentation du nombre de points engendra une précision accrue de la fonction d'interpolation dans cette nouvelle zone.

L'utilisation des diagrammes à moustache tels que présentés à la section 3.9.3 permet de valider la zone d'intérêt. Sur la Figure 3-8, trois possibilités de résultats sont présentées selon les des variables X1, X2 et X3.

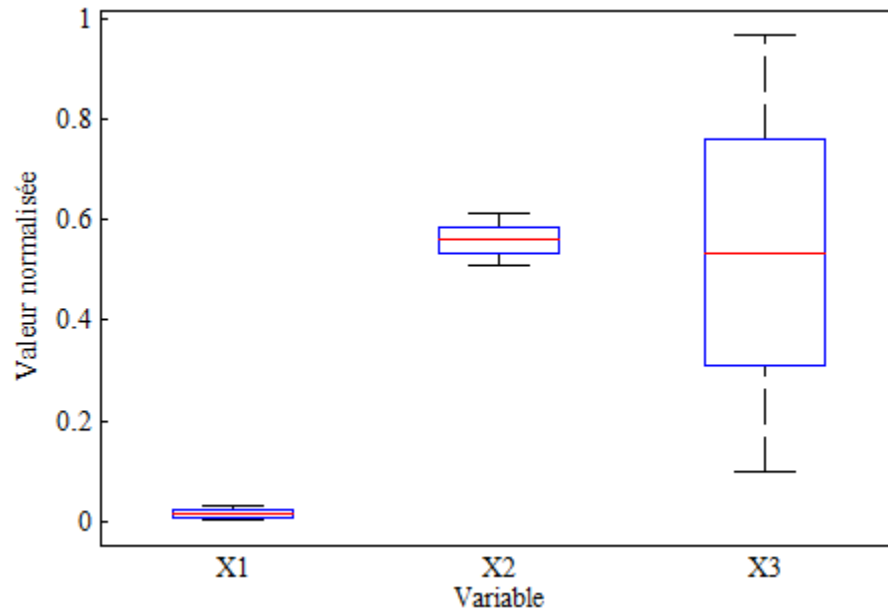


Figure 3-8 Analyse de dispersions des résultats d'optimisation

Pour la variable X1, les solutions sont concentrées sur la borne inférieure de la variable. Dans ce cas, il est fort probable que la solution optimale se trouve à l'extérieur de la zone d'intérêt. Si l'espace de conception permet le déplacement de cette borne, il est donc approprié de la déplacer et ensuite générer un nouveau plan d'expériences.

La variable X2 montre des résultats de l'optimisation très concentrés. Lorsque ceci arrive, il est souhaitable de cerner la zone d'intérêt afin d'augmenter la précision autour de ces valeurs.

Enfin, les résultats de la variable X3 montrent un bon choix de la zone d'intérêt.

### 3.9.6 Méthode de remplissage

Il arrive que la solution ne respecte pas un critère d'erreur maximal établi au préalable. Ceci est souvent dû à l'imprécision du modèle de krigeage en dehors des points d'expériences. La méthode de remplissage consiste à ajouter des points au plan d'expériences afin d'augmenter leur densité et ainsi, augmenter la sensibilité de la fonction d'interpolation. Bien que l'algorithme OSMO soit conçu pour les optimisations multi-objectifs, il est possible de l'utiliser pour un problème mono-objectif. En ce sens, la méthode remplissage diffère. Dans le cas, de l'optimisation multi-objectifs, une méthode de remplissage par candidats potentiels est développée. Pour un problème mono-objectif, la méthode appliquée est celle de Bonte [32], qui consiste à minimiser une fonction de mérite.

### 3.9.6.1 Remplissage par candidats potentiels

À la suite de l'optimisation multi-objectifs, des solutions optimales sont proposées à l'utilisateur. Ce dernier doit donc choisir certaines solutions afin de valider leur précision avec le modèle numérique. Ces solutions retenues sont nommées « candidat potentiel ». C'est avec ces candidats potentiels qu'on estime l'erreur d'interpolation avec l'écart absolu moyen entre leur réponse approximée et réelle.

Si l'erreur est plus petite que le critère d'arrêt, l'algorithme a donc convergé. Sinon, la méthode de remplissage discutée ici propose donc d'ajouter les candidats potentiels au plan d'expériences et ainsi recommencer le calcul des fonctions de Krigeage. En effet, ces candidats potentiels sont des points optimaux de la fonction d'interpolation, il a donc une certaine probabilité qu'il soit proche du Front de Pareto recherché. Ainsi, en les ajoutant au plan d'expériences, on augmente donc la sensibilité de la fonction de Krigeage dans cette région de la zone d'intérêt.

Après plusieurs itérations, la fonction d'interpolation sera donc plus précise et ainsi, l'algorithme convergera. Selon les tests de l'auteur, la sélection du 1er et 9e décile, du 1<sup>er</sup> et 4e quartile et de chaque variable de réponse Y du front de Pareto offre de bons résultats.

### 3.9.6.2 Minimisation d'une fonction de mérite

Le principe de la fonction de mérite est de soustraire la variance de la fonction de krigeage et d'y trouver les minimas potentiels entre les points d'expériences. Par exemple, sur la Figure 3-9, trois courbes, une d'interpolation et 2 de mérites sont présentées. Sur cette figure, les deux fonctions de mérites sont la fonction d'interpolation avec la somme et la différence de la variance des moindres carrés.

Les deux points ajoutés au point d'expériences seront donc les deux minimas de la fonction de mérite plus petit que la plus petite valeur du plan d'expériences (à  $x=2$ ).

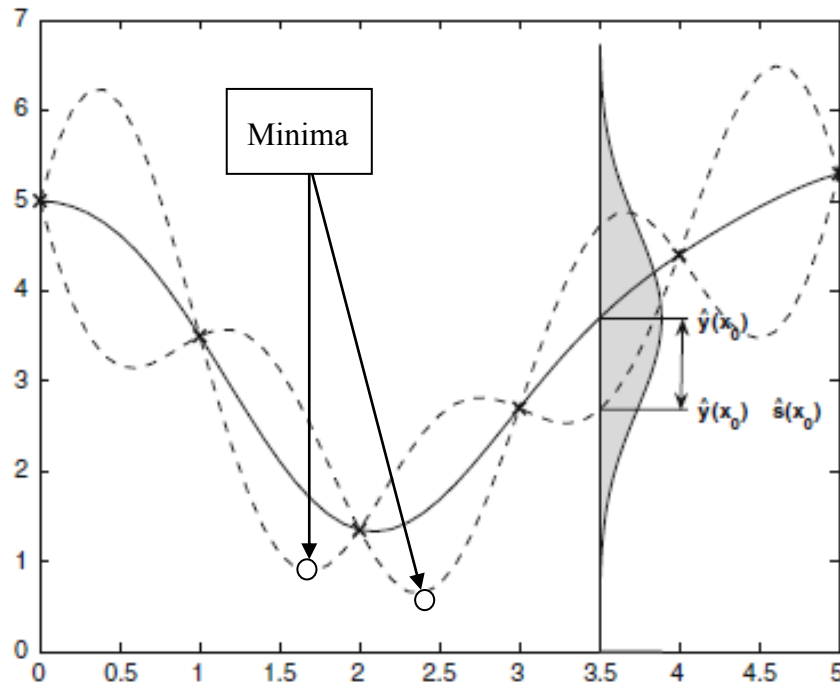


Figure 3-9 Fonction de mérite présenté par Bonte [32]

La fonction de mérite présentée par Bonte s'écrit alors ainsi :

$$f_{merite}(x) = \hat{y}(x) - w * \hat{s}(x). \quad (3.22)$$

L'évaluation de  $\hat{s}(x)$  s'évalue à partir des matrices calculées pour évaluer la réponse. Le développement de l'équation est celle présenté dans les travaux de Lophaven et al.[34]. Une fonction a été écrite :

$$\hat{s}(x) = \sigma^2 (1 + \|G^{-1}u\|_2^2 - \|\tilde{r}\|_2^2) \quad (3.23)$$

$$\text{où } \sigma^2 = \frac{1}{m} (Y - F\beta^*)R^{-1}(Y - F\beta^*), \quad \tilde{r} = C^{-1}r(x), u = \tilde{F}^T \tilde{r} - f(x).$$

Dans cette équation, les vecteurs  $Y$ ,  $r(x)$  et  $f(x)$  et les matrices  $G, F, \beta^*, R$  sont obtenus à partir des équations définies à la Section 3.6.1 portant sur la modélisation de la fonction de krigeage.

Pour trouver les différents minima étant susceptibles d'être ajoutés au plan d'expériences, Bonte suggère également d'utiliser un algorithme d'optimisation de type gradient et d'utiliser les différents points d'expériences comme point de départ. Si le minima trouvé est plus petit que le plus petit point d'expériences, on doit l'ajouter au plan. Le programme *MMF.m* a été écrit. Pour ce programme, le logiciel utilisé est Matlab étant donné qu'*Hypermath* ne permet pas d'introduire la minimisation dans une boucle.

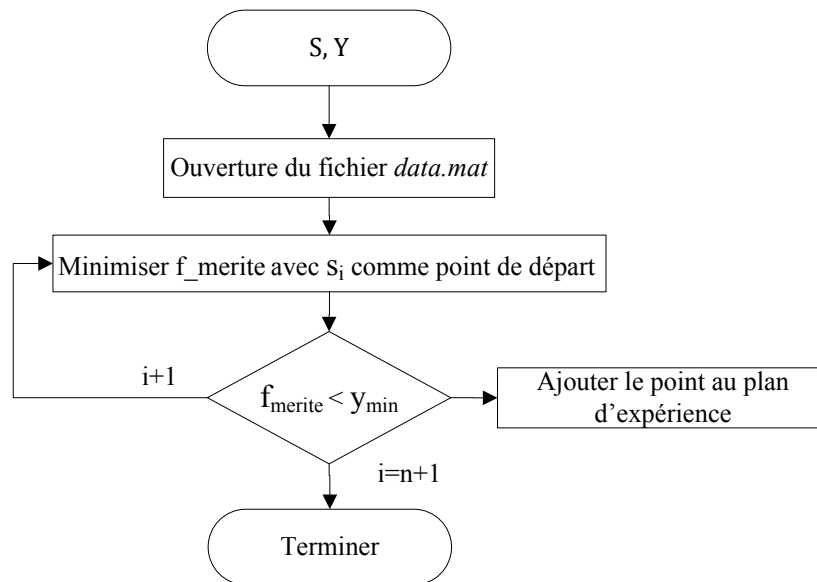


Figure 3-10 Méthode de minimisation de fonction de mérite

Ce processus d'amélioration séquentielle semble plutôt complexe et long. Cependant, les temps de calcul restent relativement acceptables, ne nécessitant qu'environ cinq minutes pour une fonction à 10 variables.

### 3.10 Programmation de l'algorithme

Tel que présenté dans ce chapitre, l'algorithme OSMO fait intervenir différents modules de la suite Altair Hyperstudy V10, des modules adjacents de Matlab et un logiciel d'éléments finis. Afin de bien visualiser les différentes fonctions utilisées la Figure 3-11 présente la structure de l'algorithme. Une application de l'algorithme à un problème structurel académique sera présentée au prochain chapitre.

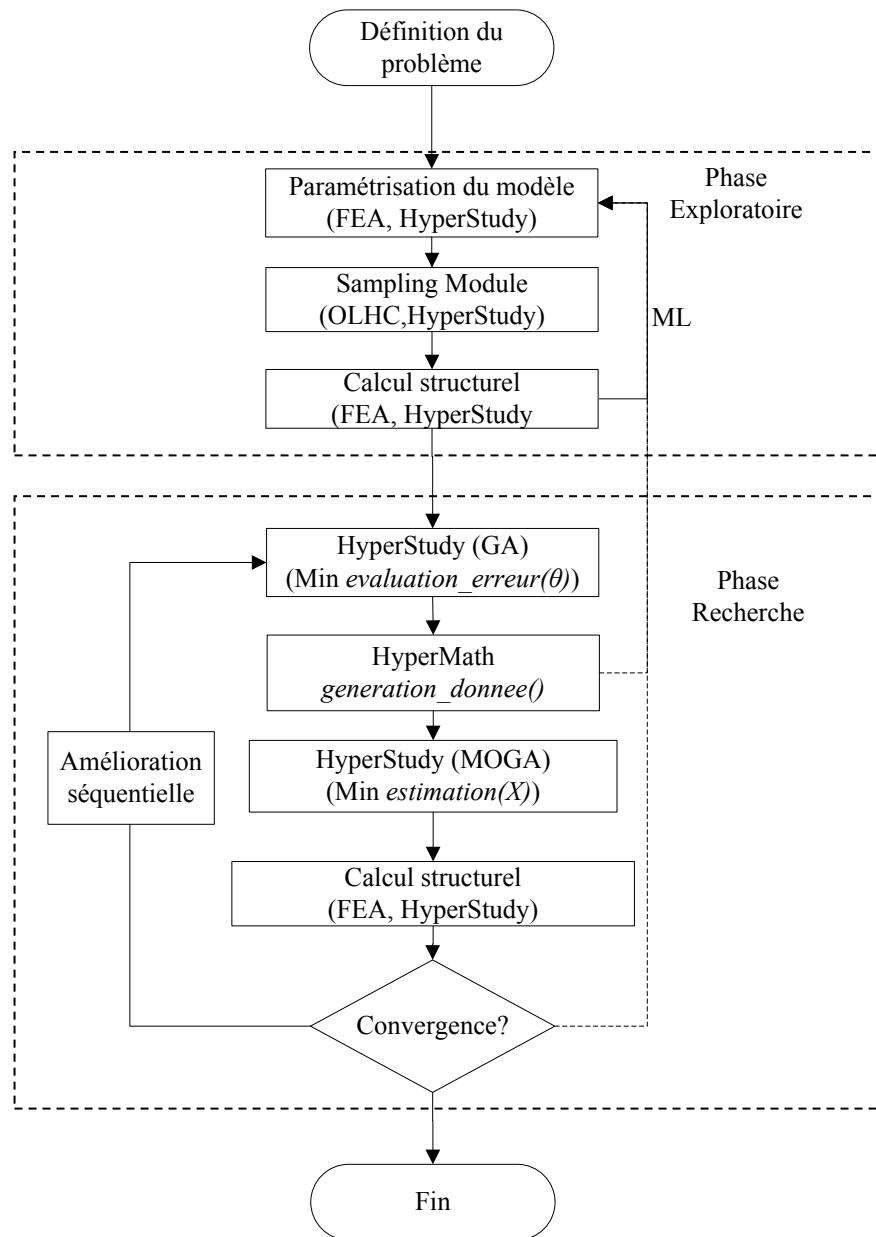


Figure 3-11 Présentation des différents logiciels utilisés pour l'algorithme OSMO



## CHAPITRE 4 APPLICATION DE L'ALGORITHME OSMO À UN PROBLÈME STRUCTUREL THÉORIQUE

Dans le chapitre précédent, les différentes étapes de l'algorithme OSMO ont été expliquées. Ce chapitre-ci décrira l'application de l'algorithme à un problème structurel théorique. Celui retenu a été initié par Haftka et Gürdal [39], il consiste à minimiser la masse et le déplacement d'une structure à 10 barreaux sollicitée en porte-à-faux (Figure 4-1). Ces barreaux sont sujets à une contrainte axiale maximale (Tension/compression). Plusieurs auteurs ont utilisé des variantes de ce problème afin de montrer une application de leur algorithme [19, 36, 40, 41].

L'utilisation d'un problème académique permet de valider l'algorithme d'optimisation tout en ayant un temps de calcul du modèle numérique négligeable. Le comportement analytique ainsi que la solution optimale du problème sont donc connus (voir Annexe 2). La résolution du problème se fera toutefois en dépit de ces connaissances théoriques afin pouvoir évaluer la performance de l'algorithme OSMO.

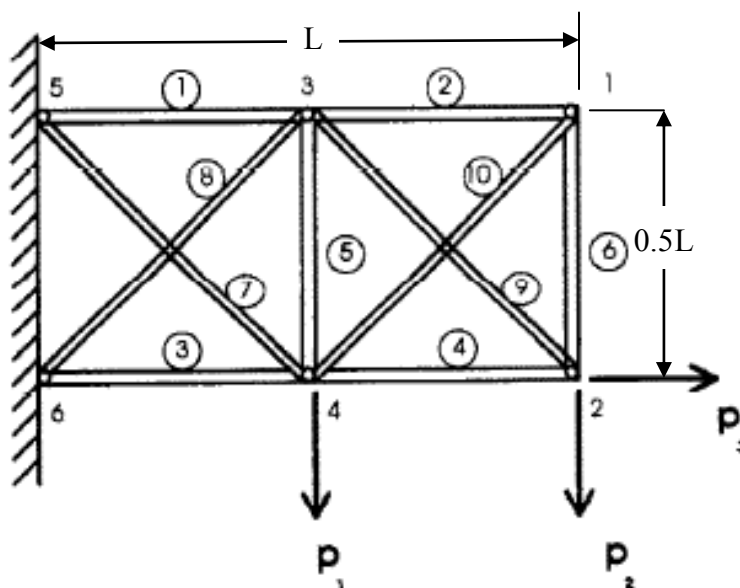


Figure 4-1 Structure à 10 barreaux [40]

Les conditions de chargement (Tableau 4.1) sont celles présentées dans les travaux de Kim et Lee [36]. En effet, ce dernier a récemment traité de ce problème avec son algorithme d'optimisation séquentielle. Ses travaux serviront donc de comparaison à l'algorithme OSMO.

## 4.1 Définition du problème

À l'instar d'utiliser un modèle d'éléments finis, des équations analytiques servent à obtenir la masse, la contrainte à l'intérieur de chaque barreau ainsi que le déplacement du nœud 2. Les barreaux sont reliés par des rotules, ils ne peuvent donc être sollicités que par une force axiale. Pour modifier la rigidité et les contraintes dans le montage, on doit donc augmenter ou diminuer les sections des 10 barreaux. Le problème fait donc intervenir 10 variables d'entrées, soit les sections des 10 barreaux nommés  $X_i$  ( $i=1,10$ ), et 12 variables de réponse, soit les contraintes axiales ( $\sigma_i(X)$ ) dans chaque barreau, la masse  $M(X)$  de la structure et le déplacement du point #2  $\delta_2(X)$  (Figure 4-1). Les relations analytiques permettant de calculer la masse  $M$ , les contraintes  $\sigma_i$  et le déplacement  $\delta_2$  sont décrites à l'Annexe 2.

Le premier cas étudié consiste à l'optimisation mono-objective de la structure et le second à celle multi-objectifs. Le problème se pose donc ainsi:

$$\begin{array}{ll}
 \text{Minimiser} & \text{Cas \#1 } M(X_i) \\
 & \text{Cas \#2 } \phi(M(X_i), \delta_2(X_i)) \\
 \text{Sujet à} & \sigma_{\text{inférieur}} \leq \sigma_i \leq \sigma_{\text{supérieur}} \\
 & X_{\text{inférieur}} \leq X_i \leq X_{\text{supérieur}}
 \end{array} \tag{4.1}$$

Pour une comparaison plus facile avec la littérature, le système d'unité impérial a été utilisé. Les paramètres relatifs au problème et leur valeur associée sont établis selon Kim et Lee [36] :

Tableau 4.1 Propriétés de la charpente à 10 barreaux

Paramètres	Valeurs
$\sigma_{\text{inférieur}}$	-25 ksi
$\sigma_{\text{supérieur}}$	25 ksi
$X_{\text{inférieur}}$	0.01in <sup>2</sup>
$X_{\text{supérieur}}$	17 in <sup>2</sup>
$F_1$	100 kips
$F_2$	100 kips
$F_3$	0 Kips
L	360 in.

Au préalable à la résolution du problème, il est nécessaire de définir un critère d'arrêt de l'algorithme. La solution présentée par Kim et Lee [36] affiche un écart de 2290 lbs sur la solution optimale. L'erreur sur l'interpolation de la masse n'est que de 10lbs, par contre celles sur les contraintes maximales ne sont pas détaillées. Dans l'algorithme OSMO, l'arrêt de l'algorithme s'effectue lorsque les erreurs d'interpolations ( $\epsilon_{masse}$ ,  $\epsilon_{cont}$  et  $\epsilon_{\delta}$ ) sont inférieurs à une valeur établie initialement. Ainsi, pour ce problème, la valeur des critères d'arrêt a été établie arbitrairement à :

$$\epsilon_{cont} < 0.05 \text{ ksi} \quad (4.2)$$

$$\epsilon_{masse} < 10 \text{ lbs}$$

$$\epsilon_{\delta} < 0.05 \text{ in.}$$

Le problème d'optimisation étant défini, la partie suivante consiste à réaliser la phase exploratoire.

## 4.2 Phase exploratoire

Lorsque l'utilisateur a des connaissances sur le comportement du modèle, il est souhaitable de définir une zone d'intérêt plus raffinée qui cerne la solution recherchée. Dans le cas du problème étudié, la première itération est établie selon l'espace de conception. La zone d'intérêt est donc bornée ainsi:

$$0.01 \leq X_i \leq 17 \text{ pour } i = 1,10$$

$X_i$  étant la section des barreaux.

Dans le cadre de la phase exploratoire, lorsque l'ensemble des réponses obtenues ne répond pas aux contraintes du problème d'optimisation, l'expérimentateur devra réduire la zone d'intérêt afin de cerner la région de faisabilité, c'est-à-dire là où les solutions sont conformes aux contraintes du problème.

Pour la première itération de la phase exploratoire, tel que présenté à la section 3.5, le nombre de points d'expériences est défini à 100, soit 10 fois le nombre de paramètres. La Figure 4-2

présente les résultats du plan d'expériences initial en fonction des contraintes du problème (nommée « C » sur la figure) sous forme de boîtes à moustache. On remarque qu'une bonne majorité des solutions sont à l'extérieur de cette région de faisabilité. Des valeurs de contraintes allant jusqu'à 12 fois la limite permise sont obtenues. Cet écart a pour effet de diminuer la sensibilité de la fonction de krigeage dans cette région. Ainsi, une réduction des limites de la zone d'intérêt sera effectuée.

Il est à noter que, sans avoir de critères particuliers, il est toujours souhaitable d'effectuer une itération de modification de la zone d'intérêt. Ainsi, on évite une erreur trop élevée entre la fonction d'interpolation et le modèle numérique.

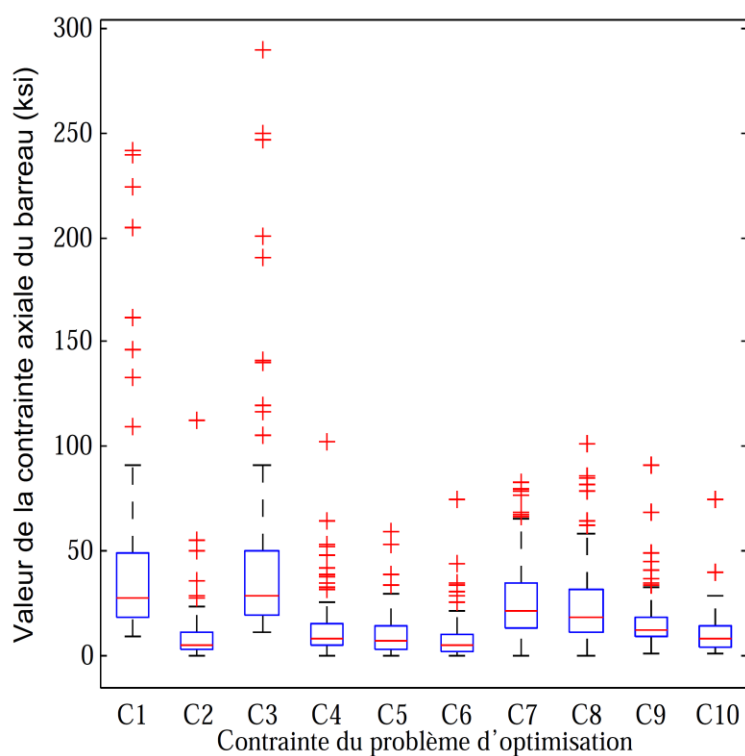


Figure 4-2 Contraintes dans les barreaux avec le plan d'expériences initial

#### 4.2.1 Modification des limites

Afin d'éliminer les solutions marginales qui ne respectent la région de faisabilité, la méthode de modifications des limites a été appliquée. La démarche a été effectuée telle que décrite à la Section 3.9.4. Celle-ci consiste à diviser pour chaque contrainte, les points d'expériences conformes et non-conformes à celle-ci, et y afficher la valeur de leurs paramètres sous forme de

boite à moustache. Ce raisonnement a été appliqué pour chaque contrainte. (Voir Annexe C ). Ceci a donc généré une nouvelle zone d'intérêt présentée dans le Tableau 4.2.

Tableau 4.2 Bornes de la zone d'intérêt pour l'itération 1

Paramètres	Borne inférieure (in <sup>2</sup> )	Borne supérieure (in <sup>2</sup> )
X1	6.8	17
X2	0.01	17
X3	6.8	17
X4	2.04	17
X5	0.01	17
X6	0.01	17
X7	4.25	17
X8	3.4	17
X9	1.7	17
X10	0.01	17

La valeur des contraintes pour ce plan d'expériences est présentée à la Figure 4-3 sous forme de boites à moustache. On remarque que les valeurs des paramètres sont en majeure partie inférieure à 25 ksi. Le fait d'obtenir certaines valeurs marginales soient supérieures à 25 ksi ne devraient pas affecter les résultats puisqu'il en demeure que l'algorithme d'optimisation sera contraint et ne permettra une recherche dans cette région

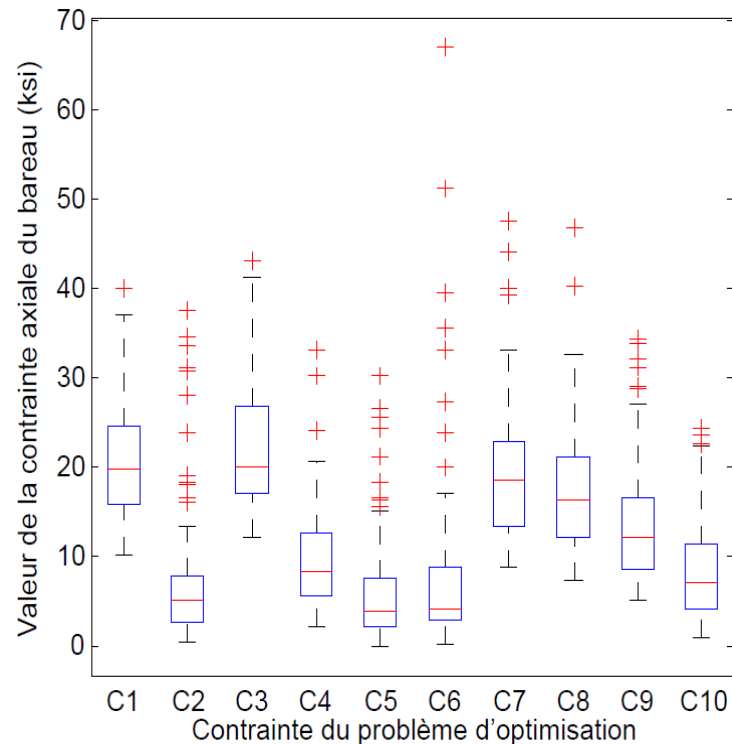


Figure 4-3 Contraintes dans les barreaux à l'itération 1

Lorsque la zone d'intérêt est définie, il est possible de passer à la phase recherche de l'algorithme. Cette phase aura pour but d'améliorer la précision du modèle d'interpolation et ainsi, de trouver la solution optimale.

### 4.3 Phase recherche : optimisation mono-objective

Cette partie traite du problème mono-objectif qui consiste à minimiser la masse de la structure tout en ayant les contraintes axiales à l'intérieur des barreaux inférieures à 25ksi. Le principe d'amélioration séquentielle est différent entre un problème mono-objectif et multi-objectifs. La méthode retenue est celle de Bonte et al. [32] décrite à la section 3.9.6.2. Le critère d'arrêt de l'algorithme est lorsque son erreur d'interpolation de toutes les contraintes et la masse seront respectivement inférieure à 0.05 ksi et 10 lbs. On quantifie cette erreur par l'écart absolu entre la valeur optimale approximée et sa validation avec le modèle numérique.

### 4.3.1 Convergence de l'algorithme

Dans le cas échéant, l'approximation de la masse par les fonctions de Krigage est quasi-exacte (erreur  $\approx 1 \times 10^{-6}$ ). La difficulté de l'optimisation repose donc sur l'interpolation des contraintes axiales présentes dans les barreaux. Enfin, la convergence de l'algorithme a été atteinte avec 7 itérations. La Figure 4-6 montre l'évolution de l'erreur au fil des itérations de la phase recherche pour chacune des contraintes. Le Tableau 4.3 présente les valeurs de l'erreur d'interpolation sur la section de chaque barreau au fil des itérations.

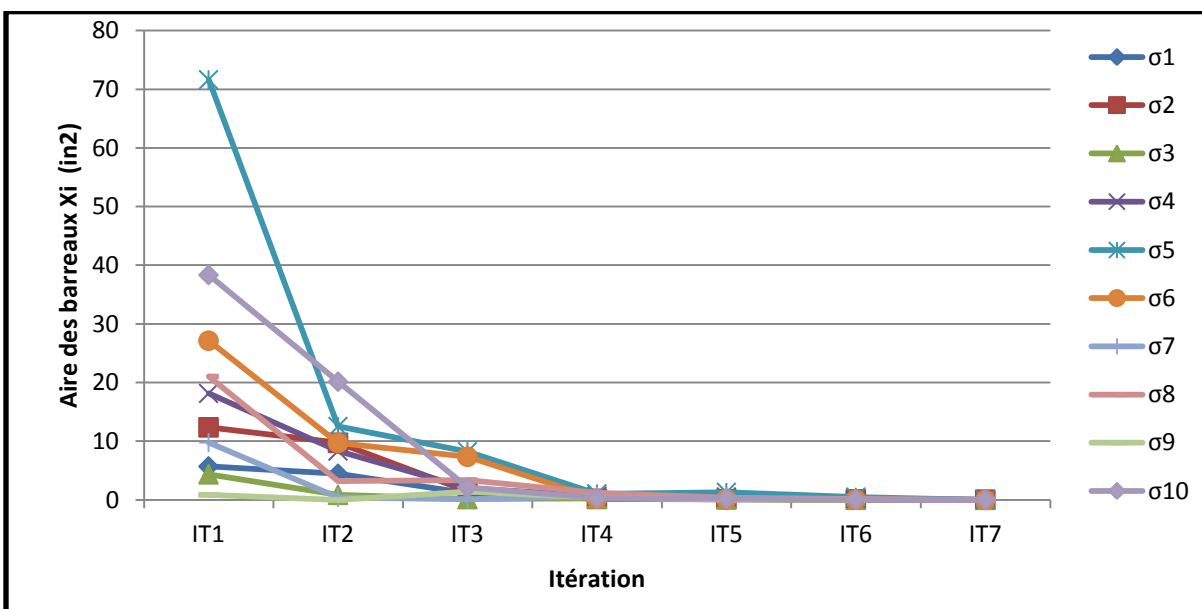


Figure 4-4 Erreur d'interpolation sur les contraintes

Tableau 4.3 Erreur d'interpolation sur les contraintes

	$\sigma_1$	$\sigma_2$	$\sigma_3$	$\sigma_4$	$\sigma_5$	$\sigma_6$	$\sigma_7$	$\sigma_8$	$\sigma_9$	$\sigma_{10}$
IT1	5.6968	12.3622	4.3592	18.1422	71.6302	27.1314	9.7652	21.0269	0.8407	38.3323
IT2	4.4446	9.7285	0.8546	8.3288	12.5109	9.6985	0.3938	3.2170	0.0008	20.1542
IT3	1.1327	1.4557	0.1329	2.0081	8.2235	7.3430	0.1085	3.3755	1.4018	2.1383
IT4	0.2081	0.1640	0.2385	0.6206	1.0344	0.1940	0.6407	1.1781	0.3570	0.3116
IT5	0.0903	0.1023	0.1312	0.4904	1.2900	0.1323	0.1581	0.3811	0.0782	0.2243
IT6	0.0126	0.0767	0.0100	0.2336	0.4647	0.0467	0.1039	0.1221	0.0150	0.0288
IT7	0.0034	0.0316	0.0064	0.0040	0.0017	0.0016	0.0046	0.0056	0.0004	0.0351

À chaque itération, la minimisation de la fonction de mérite a généré un seul point à ajouter. La raison étant que la fonction de krigage de la fonction objective est quasi exacte. On a observé

que la variance aux moindres carrés était très faible, donc ceci est équivalent à minimiser la fonction de krigeage (voir Equation 2.5, section 3.9.6.2). L'amélioration séquentielle présentée par Bonte présente donc une limite à ce niveau. L'algorithme d'optimisation a néanmoins convergé en 7 itérations, donc avec 7 calculs du modèle numérique pour un total de 207 calculs du modèle numérique incluant la phase exploratoire.

### 4.3.2 Résultats de l'algorithme

La convergence de l'algorithme est basée sur l'erreur d'interpolation entre la valeur optimale calculée et sa valeur réelle. Ceci permet d'obtenir implicitement également la convergence vers une solution optimale théorique et ainsi, les valeurs de ses paramètres. Les Figure 4-5 et Figure 4-6 présentent donc l'évolution de la masse et la valeur des paramètres au fil des itérations.

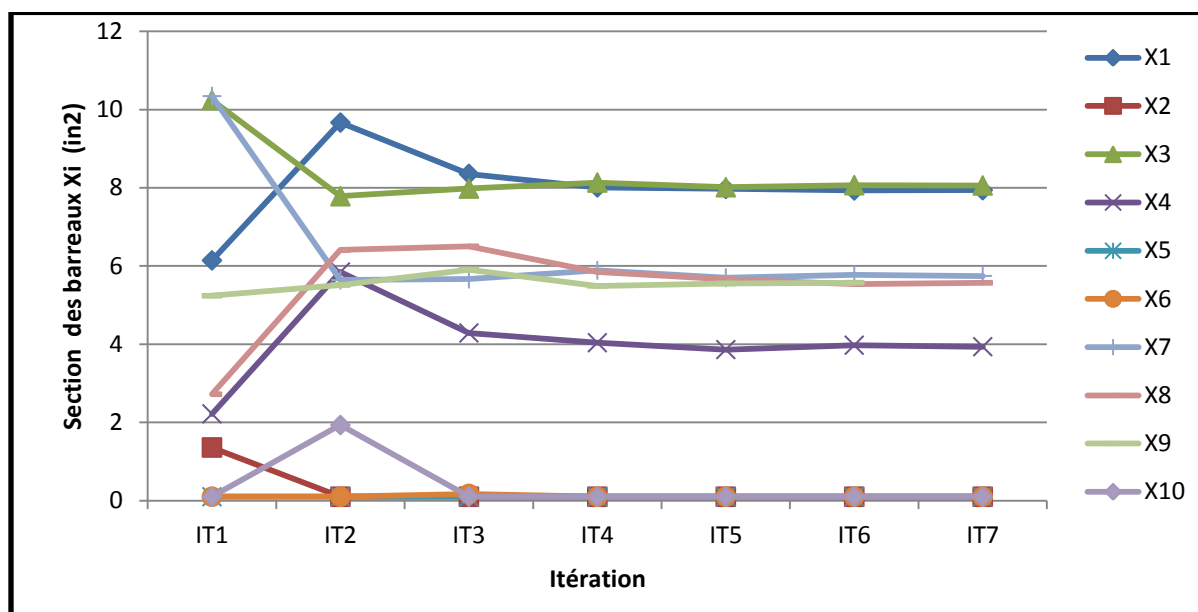


Figure 4-5 Convergence de la valeur des paramètres  $X_i$



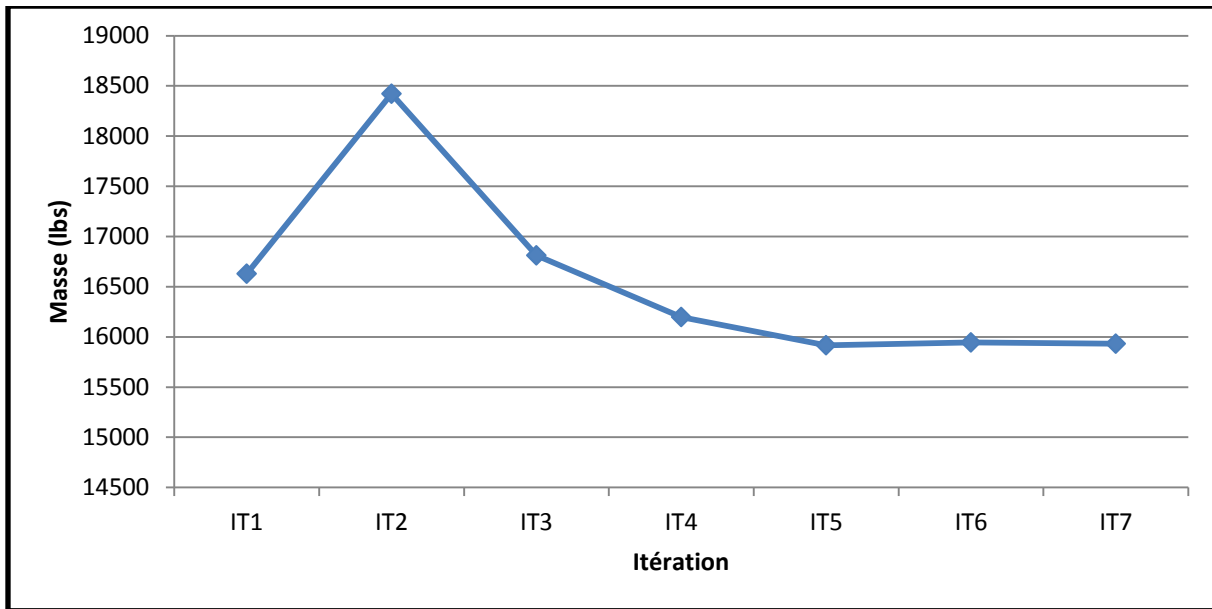


Figure 4-6 Convergence de la masse de la structure

### 4.3.3 Comparaison de la solution

Tel que mentionné, le problème de minimisation de la structure à 10 barreaux a une solution optimale connue. De plus, Kim et Lee [36] ont récemment optimisé ce problème à l'aide de son algorithme séquentiel. Le Tableau 4.4 présente la solution optimale du problème, celle obtenue par Kim et celle obtenue par l'algorithme OSMO. Avec ces résultats, on observe que le nombre d'itérations de calcul est nettement inférieur avec l'algorithme OSMO. De plus, la précision de l'algorithme OSMO est largement plus élevée que celui présenté par Kim.

Tableau 4.4 Comparaison des résultats

	Exacte	Kim	OSMO
X1	7.94	8.95	7.939
X2	0.1	1.1	0.100
X3	8.06	7.46	8.060
X4	3.94	3.52	3.937
X5	0.1	2.66	0.100
X6	0.1	0.065	0.100
X7	5.75	4.89	5.746
X8	5.57	7.73	5.570
X9	5.57	4.97	5.569
X10	0.1	0.95	0.100
Masse	15930	18220	15932.000
Calcul Structurel		1239	207

#### 4.4 Phase Recherche : Optimisation Multi-Objectifs

La convergence de l'algorithme OS MO dans la partie multi-objectifs est plus complexe. En effet, l'optimisation multi-objectifs génère plusieurs solutions optimales, ainsi chacune d'elle ont des erreurs d'interpolations différentes. La sensibilité et la précision de krigeage doivent donc être bonnes dans la région formée par le Front de Pareto et non seulement autour d'un point optimal.

De plus, les contraintes du problème sont souvent non-actives, c'est-à-dire qu'il ne limite pas le problème. En effet, il est plus rare de voir une contrainte active pour tous les objectifs en même temps. Imposer un critère d'arrêt sur les contraintes est plus sévère puisqu'elle varie selon le Front de Pareto. Ainsi, le critère de convergence sera plutôt appliqué au déplacement de la structure. La vérification des contraintes pour tout le Front de Pareto reste toutefois nécessaire.

Bien qu'il soit normal de s'assurer de la convergence du modèle d'interpolation, la qualité du front de Pareto ne doit cependant pas être négligée. Avec l'algorithme OS MO, le Front de Pareto est surtout contrôlé par les paramètres de l'algorithme d'optimisation MOGA.

##### 4.4.1 Amélioration séquentielle

Les candidats potentiels sont les solutions optimales retenues du Front de Pareto pour effectuer l'amélioration séquentielle. Chaque candidat potentiel doit être validé par le modèle numérique.

Pour ce problème, le temps de calcul est négligeable, mais dans le cadre d'une optimisation à partir d'un modèle d'éléments finis, il est important de sélectionner un nombre adéquat de candidats potentiels.

Dans le problème présent, il a été déterminé d'ajouter 5 candidats potentiels dispersés sur le front de Pareto. Ces points ont été sélectionnés aux environs du 10<sup>ième</sup>, 25<sup>ième</sup>, 50<sup>ième</sup>, 75<sup>ième</sup> et 90<sup>ième</sup> percentiles. Le calcul de l'erreur d'interpolation sera donc basé sur l'écart moyen absolu de ces points entre le déplacement interpolé et celui validé par le modèle numérique. Pour atteindre la convergence, il faudra donc que cette erreur d'interpolation sur le déplacement soit inférieure à 0.05 in., tel que spécifié dans la définition du problème.

#### 4.4.2 Convergence de l'algorithme

Il est possible de se demander si le calcul de l'erreur d'interpolation à partir de seulement 5 points est représentatif de l'erreur d'interpolation sur l'ensemble du Front de Pareto. Ainsi, sur la Figure 4-7, une courbe représentant l'écart absolu moyen de tous les points obtenus de l'algorithme MOGA et leurs valeurs réelles sont présentées. On remarque que le calcul de l'erreur pour seulement 5 points représente globalement celle sur tout le Front de Pareto.

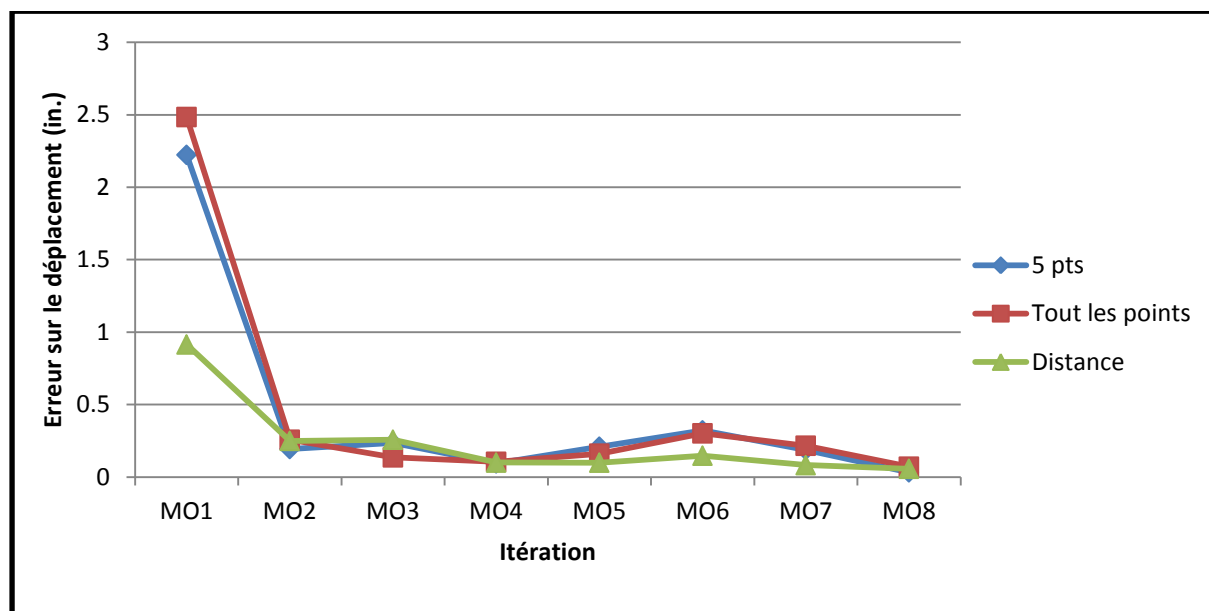


Figure 4-7 Erreur sur l'optimisation multi-objectifs

De plus, une solution théorique a été obtenue en optimisant les fonctions analytiques. Il est donc possible de calculer la distance entre Front de Pareto théorique et celui approximé au fil des

itérations afin de déterminer si la tendance est celle recherchée, soit la diminution des écarts. Cette distance est également quantifiée par l'écart absolu moyen et est présentée dans la Figure 4-7. Étant donné que l'approximation de la masse est quasi-exacte ( $<10^{-6}$ ), la distance entre le front de Pareto théorique et celle approximée revient donc à calculer celle entre le déplacement théorique et approximé pour une même masse.

Ce calcul de distance et d'erreurs absolues moyennes de tous les points a été possible puisqu'un modèle numérique théorique a été utilisé. Ceci montre que la méthode d'amélioration séquentielle basée sur la sélection de 5 candidats potentiels permet de faire converger l'algorithme vers le Front de Pareto théorique et également représenter l'erreur sur l'ensemble des points du Front de Pareto.

#### **4.4.3 Vérifications des contraintes**

Bien qu'elles ne soient pour la plupart non-actives, il est nécessaire de valider que les candidats potentiels respectent les contraintes. En effet, lorsque l'erreur d'interpolation des contraintes est trop élevée, il arrive que les contraintes ne soient pas respectées après la validation avec le modèle numérique.

Normalement, avec la méthode d'amélioration séquentielle retenue, le nombre de contraintes des candidats potentiels qui ne sont pas respectées doit diminuer. Ainsi la Figure 4-8 présente le nombre de contraintes non-respectées en fonction des itérations. Sur cette figure on remarque que plus l'algorithme évolue, plus les contraintes sont respectées. Ainsi, cela montre également que la méthode d'amélioration séquentielle fonctionne adéquatement.

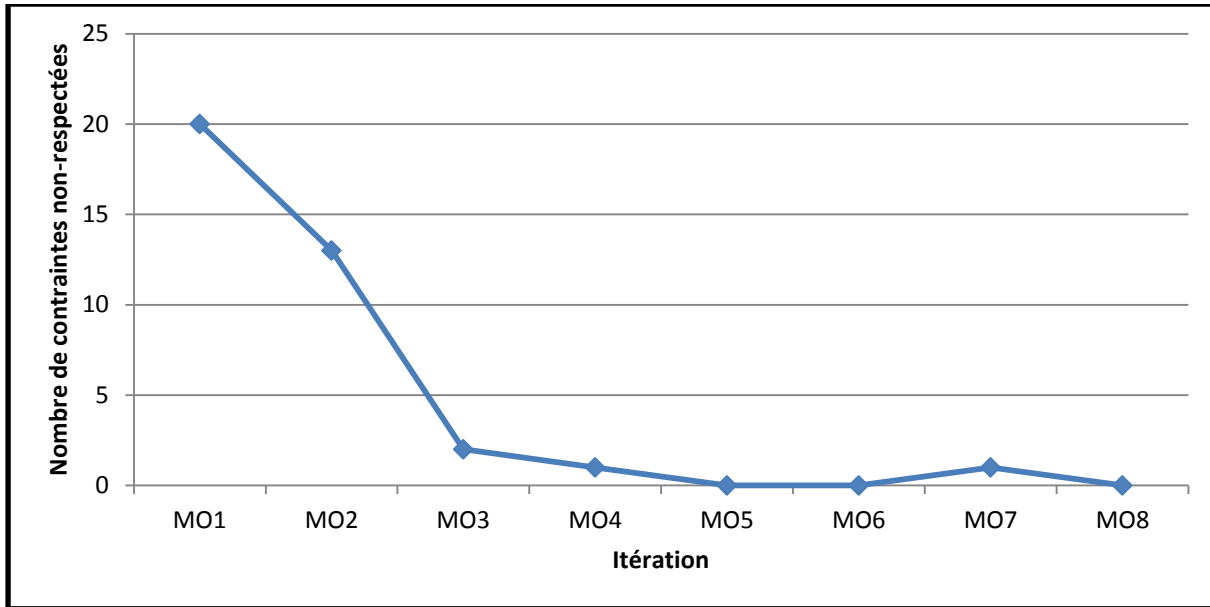


Figure 4-8 Conformité des contraintes

#### 4.4.4 Résultat de l'algorithme OSMO.

La Figure 4-9 présente les résultats de l'algorithme d'optimisation OSMO. On observe que l'algorithme a été en mesure de générer un Front de Pareto similaire au celui théorique. On remarque également que l'algorithme génère des résultats nettement meilleur que l'ensemble des points du plan d'expériences.

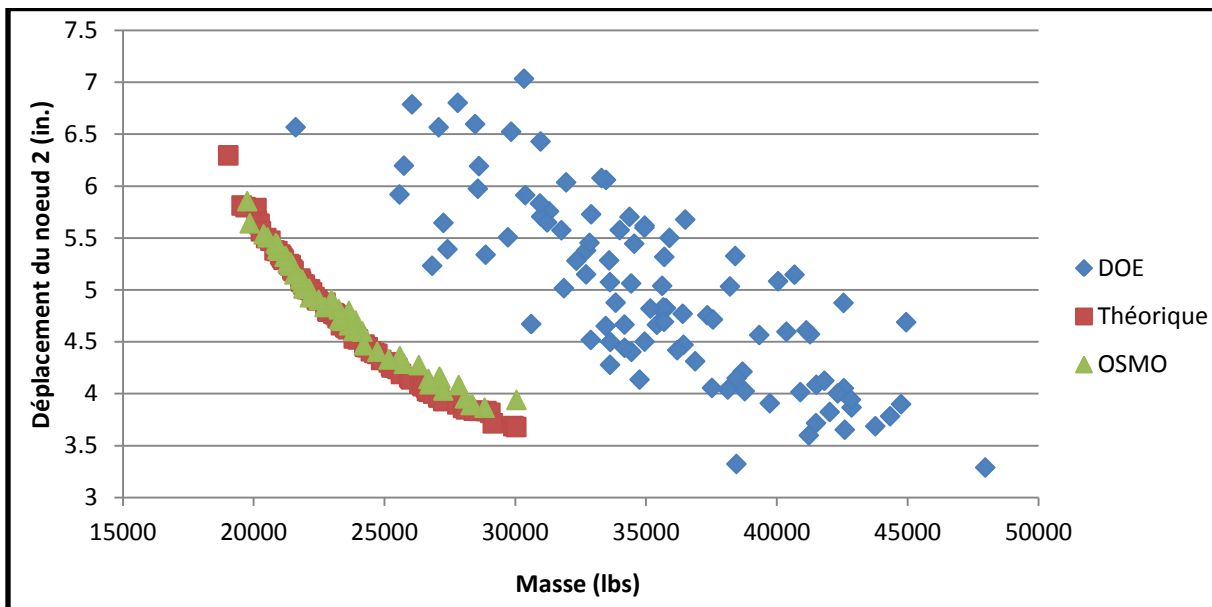


Figure 4-9 Résultats de l'optimisation Multi-Objectifs

## 4.5 Limites de l'algorithme OSMO observées

Le but de ce travail était de mettre en place un algorithme d'optimisation séquentielle qui permettra de résoudre les problèmes de frettage. Dans ce chapitre, l'algorithme a été appliqué à un problème académique afin de le valider et également d'observer ses limites. Le logiciel retenu pour programmer l'algorithme a été la suite Altair Hyperworks V10. Trois interfaces différentes et indépendantes de ce logiciel ont été utilisées dans l'algorithme OSMO. Le premier est *Hyperstudy* pour définir les paramètres, les bornes et les contraintes et ainsi réaliser l'optimisation. Le second est le *Sampling Module*, servant à générer le plan d'expériences *Optimal Latin HyperCube*. Le dernier est le *Hypermath* utilisé pour l'écriture de la fonction de krigeage. Chacun de ces modules a des limitations qui seront présentées et discutées dans cette section.

### 4.5.1 *Sampling Module*

Ce module permet de générer un plan d'expériences en fonction des paramètres et des bornes qu'on lui impose. Toutefois, il ne permet pas d'appliquer des contraintes. Par exemple, s'il y a une forme d'îlot à l'intérieur du domaine de conception où les solutions ne respecteraient pas les contraintes, il ne sera pas possible de l'imposer au logiciel.

Dans ce cas, si les solutions sont faisables numériquement, il est conseillé d'exécuter le plan d'expériences en ne considérant pas cet îlot et d'imposer ces contraintes à l'algorithme d'optimisation. Toutefois, le scénario est pour le moment indéterminé pour des solutions qui ne seraient pas réalisables numériquement à l'intérieur de l'espace de conception.

### 4.5.2 Performance de l'algorithme MOGA

Les fonctions de krigeage créent souvent des minimas entre les points d'expériences. Ainsi, lorsque le solveur multi-objectifs de type gradient de Altair (GMMO) a été appliqué, il a rapidement convergé vers un minima local et était donc, peu diversifié. Pour éviter ce cas, l'utilisation de l'algorithme d'algorithme génétique MOGA a été préconisée et offre de meilleurs résultats dans les circonstances.

Une optimisation du problème multi-objectifs a été réalisée sur les fonction analytiques afin de comparer la performance de l'algorithme MOGA et MGGO. La Figure 4-10 présente une comparaison entre le front de Pareto obtenu avec l'algorithme MOGA et celui obtenu avec l'algorithme GMMO. Le nombre d'évaluations requises des fonctions pour le GMMO est de 20 000 et pour le MOGA de 80 000. Plusieurs essais avec différents paramètres de réglage de l'algorithme MOGA ont été effectués afin d'améliorer la qualité du Front de Pareto jusqu'à un seuil où il ne variait plus. On remarque que dans l'ensemble le front de Pareto des deux algorithmes a convergé.

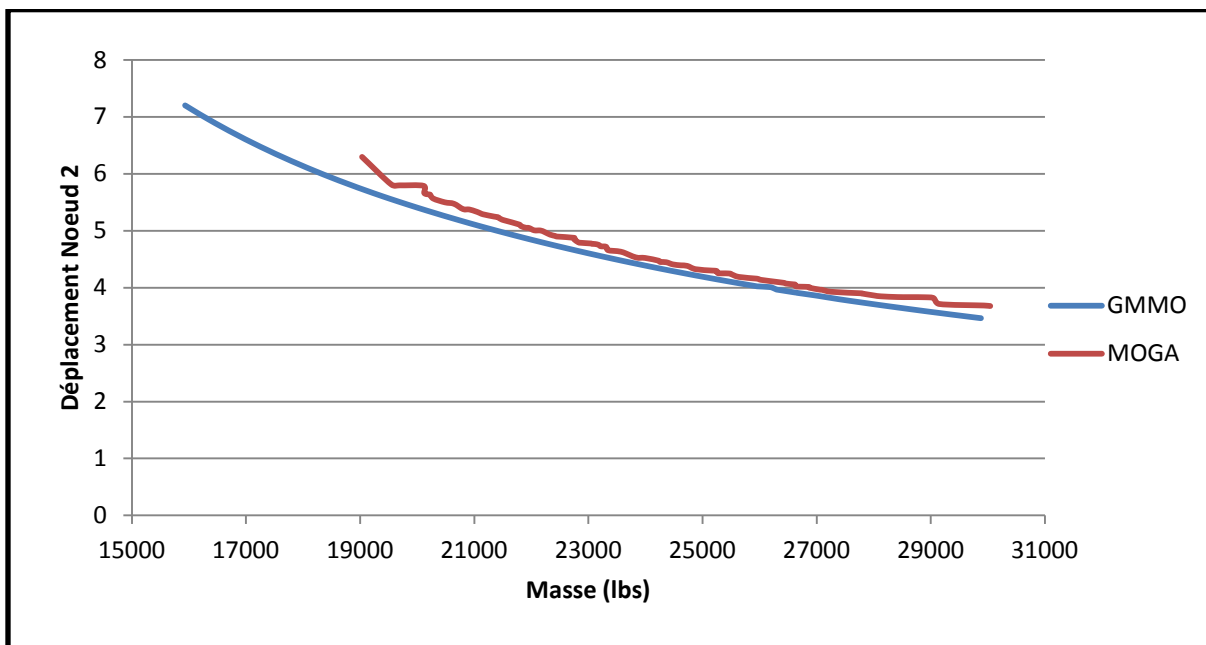


Figure 4-10 Comparaison des résultats de l'algorithme MOGA et GMMO

## 4.6 Conclusion

Le but de l'optimisation séquentielle est d'offrir une solution optimale tout en réduisant le nombre de calculs structurels. La résolution d'un problème d'optimisation multi-objectifs requiert beaucoup de calculs. Avec 80 000 évaluations de fonctions nécessaires, coupler un algorithme multi-objectifs à un modèle d'éléments finis non-linéaire est donc impensable. Ainsi dans le cadre d'un problème structurel, pour pouvoir utiliser l'approche multi-objectifs, l'optimisation séquentielle est donc jugée nécessaire.

Dans ce chapitre, on a vu que l'algorithme OSMO limitait le nombre de calcul structurel et assure de bons résultats. Tel que présenté, l'algorithme OSMO contient 2 phases : exploration et recherche. La phase d'exploration consiste à générer un plan d'expériences adéquat. Dans le cas de l'optimisation d'une structure à 10 barreaux, la majorité des points d'expériences générés lors de la première itération ne répondait pas aux critères de conception. Il a été décidé de recibler la zone d'intérêt afin d'obtenir une meilleure sensibilité pour les solutions recherchées. Ceci a permis de fournir un plan d'expériences de meilleure qualité. Pour la phase recherche de l'algorithme, une application sur un problème mono-objectif a été faite afin de comparer les résultats de l'algorithme OSMO avec un algorithme existant. Ensuite, une variante multi-objectifs du problème a été également résolue. Dans les deux cas, l'algorithme OSMO a convergé vers la solution optimale, et ce, efficacement.

Le Tableau 5.5 présente les temps de calcul pour chaque opération. Celui nécessaire à l'exécution de l'algorithme OSMO a été 270 minutes. Ainsi, lors d'une optimisation sur un modèle d'éléments finis, un estimé du temps nécessaire pourra donc être fait en ajoutant ce temps d'opération à celui nécessaire pour effectuer les calculs du modèle.

Tableau 4.5 Temps de calcul (en minute)

Étapes	Temps/calcul	Fréquence	Itération	Total
Plan d'expériences	0	100	2	0
Évaluation fonction krigeage	0.0045	2000	8	72
Theta	15	1	8	120
Simulation	15	5	8	0
Opt Multi-Obj	10	1	8	80
				272

Enfin, ce chapitre a permis de constater que l'algorithme OSMO permettait de résoudre efficacement des problèmes structurels. Ainsi, le prochain chapitre présente l'application de l'algorithme au problème de conception d'assemblage par interférence décrit dans le premier chapitre.



## CHAPITRE 5    ARTICLE : OPTIMAL DESIGN OF INTERFERENCE FIT ASSEMBLIES SUBJECTED TO FATIGUE LOADS

### 5.1 Présentation de l'article

Tel que présenté au Chapitre 1, Lanoue [11, 12] a mis en place une méthode de calcul de vie en fatigue des assemblages par interférence avec la méthode des éléments finis. Pour les diverses raisons discutées dans ce travail, coupler un modèle d'éléments finis avec un algorithme d'optimisation n'est pas considérable. Ainsi, le but de cet article a été d'appliquer l'algorithme OSMO au modèle d'éléments finis calculant la vie en fatigue de l'assemblage par interférence soumis à des sollicitations de torsion alternée.

Ainsi, la méthode de Lanoue [11] a été reprise et écrite avec la version APDL (*A Parametric Design Language*) du Logiciel Ansys V11 afin de pouvoir y effectuer un calcul automatisé et facilement modifiable en fonction des paramètres qu'on lui fournit. La géométrie utilisée dans cet article consiste en un arbre avec épaulement assemblé avec un moyeu avec une partie en porte-à-faux. La géométrie est celle présentée dans les essais de Juuma [7]. Le critère de fatigue utilisé est celui de Von Mises.

Le problème d'optimisation consiste à minimiser la masse et les facteurs de sécurité dans le congé et dans le contact. Étant donné une incertitude additionnelle sur la prédiction de la vie en fatigue dans le contact du au phénomène de fretting, il serait souhaitable d'obtenir un facteur de sécurité plus élevé. Il aura donc un compromis à faire entre la masse et les facteurs de sécurité.

Dans cet article, la présentation de l'algorithme OSMO a été simplifiée puisque le but de cet ouvrage n'était pas de détailler l'algorithme, mais plutôt de montrer qu'il peut résoudre efficacement un problème de conception des assemblages par interférence soumis à des sollicitations de fatigue. L'approche multi-objectifs pour résoudre le problème spécifique de conception sera également discutée.

# OPTIMAL DESIGN OF INTERFERENCE FIT ASSEMBLIES SUBJECTED TO FATIGUE LOADS

a Sequential Approximate Multi-Objective Optimization approach

## Authors

G. Biron and A. Vadean\*

Department of mechanical engineering

Ecole Polytechnique, Montreal

Tel.: +514-340-4711 ext. 4589

aurelian.vadean@polymtl.ca

L. Tudose

Technical University of Cluj-Hapoca, Romania

\* Corresponding authors

Submitted: July 9<sup>th</sup> 2011

## 5.2 Abstract

This paper presents a methodology for an optimal design of interference fit subjected to fatigue loads. Optimization consists in finding a tradeoff between mass and competing safety factors at hub and shaft contact zone as well as in shaft fillet. Developing an effective calculation method for the fatigue strength of interference fitted assembly using the finite element method is one of the main steps of the procedure. Meanwhile, coupling the finite elements model of interference fit with an optimization algorithm is not adequate considering the computing time and the important amount of calculations necessary to portrait the assembly behavior. Therefore, a sequential

approximate multi-objective optimization algorithm (SAMOO) is presented. The method involves Design Of Experiments (DOE), interpolation with kriging functions and multi-objective optimization. Preliminary study of parameters' variance as well as an advanced postprocessing of multi-objective optimization provide engineers with valuable informations to identify an optimal design of interference fit assembly using fewer finite elements calculations.

keywords: Interference fit; Fatigue strength; Sequential approximate optimization; Multi-objective; Finite elements

### **5.3 Introduction**

In today's highly competitive markets, the engineer must be able to suggest optimal solutions while meeting ever more complex design criteria. This paper promote a methodology for the optimal design of interference fit assembly using few finite element simulations.

Interference fit is a pre-stressed method of assembling a hub and a shaft. It is popular in the industry because it has the advantage of being compact, rigid and effective. Contact of the hub and the shaft creates a sharp edge which causes important stress concentration. Moreover, the fretting fatigue phenomenon occurs when the assembly is subjected to alternating stress. In order to improve fatigue life, Juuma (1999,2000), Nishioka and Komatsu (1967) studied the effects of many interference fit configurations.

Experimental tests have shown that the addition of a fillet, circular groove or hub overhang improves the fatigue life of interference fit assemblies by 20 up to 50 %. However, these features increases the number of design parameters and consequently, the complexity of design. In contact, failure is caused by fretting fatigue while it is caused by normal fatigue in the fillet or groove (Lanoue, 2008) As a result of the competing maximum equivalent stresses in contact and in the fillet, a shift in the location of fatigue failure has been observed in experimental tests.

A major challenge of design is to find a set of parameters values that will ensure the desired safety factor of the assembly. Kubota (2009) suggests empirical methods to find the optimal values of parameter targeting the balance between damage by fretting fatigue in contact and normal fatigue in the groove. However, this approach is limited to the design of the circular groove.

Lanoue et al. (2009) has developed a method to accurately simulate interference fit assembly subjected to fatigue loads using finite elements. It consists in choosing contact algorithms and mesh sizes wisely to ensure converged results. Juuma (1999,2000), Nishioka and Komatsu (1967) experimental tests on shrink fit joints were used as comparison to the finite element model. Resolution of contact algorithm and plastic deformation at contact edge considerably increase computing time. Optimal selection of parameters values cannot be made within a reasonable time without using complementary numerical methods. Coupling the finite element model with an optimization algorithm is not considered feasible either given the number of simulations required.

Sacks et al. (1989) introduced an approximation method called Design and Analysis of Computer Experiments (DACE), which requires knowing the input and output variables in order to model the problem. This method consists in designing experiments and ensuring the results interpolation using a response surface or kriging method. Based on this principle, Jacobs et al. (2004) laid out a structure for sequential approximate optimization (SAO), which optimizes the interpolation function obtained. Recently, several authors have published their work aimed at improving design of experiments, interpolation functions and SAO structure (Forsberg and Nilsson 2005, Kim and Lee 2010, Bonte et al. 2010, Schafer and Finke 2008, Van Beers and Kleijnen 2008). These approaches are the basis for the sequential approximate multi-objective optimization (SAMOO) algorithm developed in this paper for obtaining optimal design of interference fit assembly.

In order to promote this approach, Section 2 presents modeling considerations of fatigue strength calculations for interference fit assembly. Section 3 describes the main steps of the

SAMOO algorithm. Finally, Section 4 shows the results of SAMOO's application on the interference fit problem.

## 5.4 Interference fit assembly

Modeling interference fit assemblies subjected to fatigue loads involves three aspects: defining the geometry, finite elements analysis model and fatigue criteria. These main steps constitute the basis of the generic model for the optimization procedure using SAMOO algorithm.

### 5.4.1 Parametrized model

Different topologies of interference fit assembly improving fatigue limit are found in the literature. Several authors mentioned above have suggested that the design of a hub overhanging and a shoulder fillet on a shaft substantially improves fatigue life. Dimensional parameters used by Juuma (1999,2000) studied also by Lanoue (2008), are reproduced in this work for the optimal design of interference fit assembly. The topology chosen is illustrated in Fig. 5-1 and the parameters used are defined in Table 5.1.

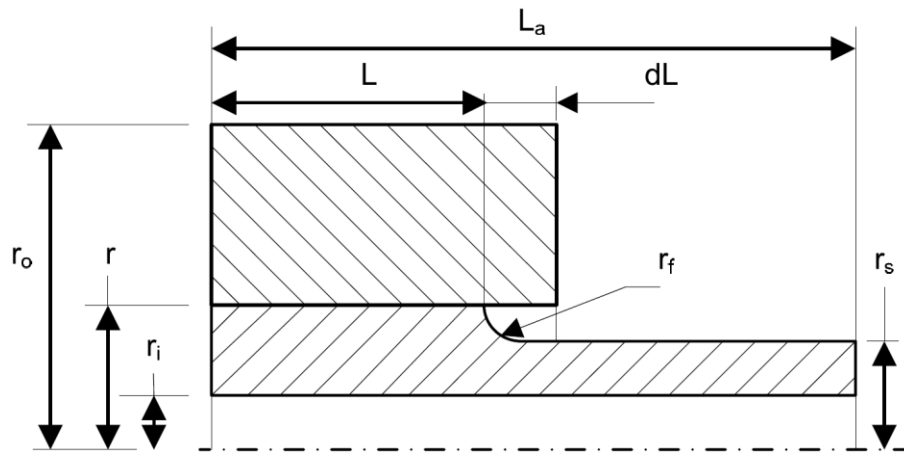


Figure 5-1 Juuma's parameterized model

Tableau 5.1 Definition of geometrics parameter

Parameter	Definition
L	Length of the hub engagement
$L_a$	Length of the assembly
$r_i$	Inside radius
$r_o$	Outside radius
r	Contact interface radius
$r_f$	Fillet radius
$r_s$	Shaft radius
dL	Hub overhang
$\delta_r$	Radial interference

According to Juuma's experimental tests, the torque applied to the model is  $3.8E+06$  Nmm. The mechanical proprieties of the material used for the finite elements model of assembly are presented in Table 5.2.

Tableau 5.2 Mechanical properties

Property	Value
Young's modulus E (MPa)	200 000
Poisson's ratio $\nu$	0.3
Yield strength $S_y$ (MPa)	330
Ultimate strength $S_{ut}$ (MPa)	545
Tangent modulus $E_T$ (MPa)	1407
Endurance limit $S_e$ (MPa)	345

### 5.4.2 Finite element model

While modeling interference fit assembly using the finite element method, contact algorithm, mesh size and simulation options must be carefully taken into consideration. First, the use of axisymmetric elements with contact is impossible when considering alternated torsional loads. To reduce calculating time, Lanoue (2008) has demonstrated that it is possible to achieve the same level of precision as that of a full model with a 1 degree slice subjected to cyclic symmetry conditions. Furthermore, this slice can be modeled with only one element in the circumferential direction, which reduces the number of required elements and, consequently, calculation time.

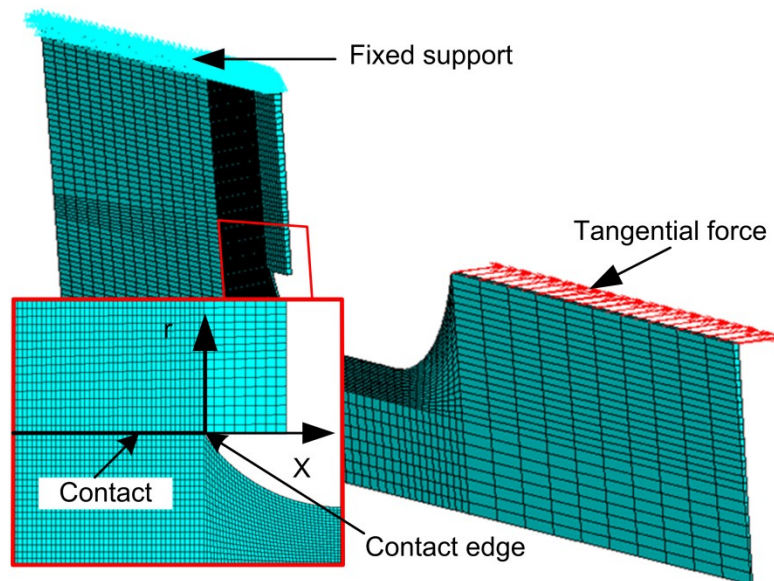


Figure 5-2 Finite element model with boundary conditions

The software chosen for the simulations is Ansys V11. The APDL (Ansys Parametric Design Language) version allows the development of a generic model with a robust mesh despite the variation of dimensions. Brick with 20 nodes (Solid186) is the type of element used. The area shown in Fig. 2 is of great interest because that is where the fatigue failure occurs. Since the model is subjected to cyclic conditions, it becomes possible to simulate the torque with tangential forces accounting for the alternating torque. Other boundary conditions are shown in Fig. 5-2

Following Lanoue et al. (2009) recommendations, the contact algorithm chosen for this work is Augmented Lagrangian. This algorithm allows for the control of maximum penetration and maximum elastic slip. The simulation options ensuring the convergence of the model are given in Table 5.3.

Tableau 5.3 Contact definition

Parameter	Value
Contact model	Surface to Surface
Coefficient of friction	0.7
Behaviour	Asymmetric
Contact algorithm	Augmented lagrangian
Penetration tolerance	0.001 mm
Allowable elastic slip	0.0001 mm
Element type	CONTA174 and TARGE170

### 5.4.3 Selection of fatigue criterion

As mentioned before, the interference fit assembly is subjected to alternated torsional loads causing fatigue failure. Fatigue life approximations can be obtained by applying a fatigue criterion on the finite element analysis results. Lanoue's studies (2011) have established that it is possible to calculate a realistic safety factor using Von Mises' fatigue criterion. Other criterion offering similar results, such as Sines, Crossland and Dang Van, have also been considered. However, because of its simplicity and fair precision, Von Mises' criterion was selected the present work. Alternating ( $\sigma_a$ ) and mean ( $\sigma_m$ ) stress in a cylindrical coordinate system is expressed as follows:

$$\sigma_a = \frac{1}{\sqrt{2}} \sqrt{(\sigma_{z,a} - \sigma_{x,a})^2 + (\sigma_{y,a} - \sigma_{x,a})^2 + (\sigma_{z,a} - \sigma_{y,a})^2 + 6(\tau_{xy,a}^2 + \tau_{xz,a}^2 + \tau_{yz,a}^2)} \quad (5.1)$$

$$\begin{aligned} \sigma_m & \quad (5.2) \\ &= \frac{1}{\sqrt{2}} \sqrt{(\sigma_{z,m} - \sigma_{x,m})^2 + (\sigma_{y,m} - \sigma_{x,m})^2 + (\sigma_{z,m} - \sigma_{y,m})^2 + 6(\tau_{xy,m}^2 + \tau_{xz,m}^2 + \tau_{yz,m}^2)}. \end{aligned}$$

Thus, with Goodman's modified diagram, it is possible to calculate the equivalent alternating Von Mises stress VM:

$$\begin{cases} VM = \frac{\sigma_a}{1 - \frac{\sigma_m}{S_{ut}}} & \text{if } \sigma_m > 0 \\ VM = \sigma_a & \text{if } \sigma_m \leq 0. \end{cases} \quad (5.3)$$



Due to the slip between the two parts, a fretting fatigue phenomenon is responsible for the premature damage in contact (Madge et al. 2007). A comparison between experimental tests and finite element model have shown the existence of a fretting fatigue strength reduction factor in contact (Lanoue et al. 2001). While Von Mises' criterion accurately approximates the fatigue strength in the fillet, a fatigue reduction factor ( $k_{fret}$ ) should be applied in contact. Therefore, the safety factor in the contact ( $SF_{contact}$ ) and in the fillet ( $SF_{fillet}$ ) are calculated as follows:

$$\begin{cases} SF_{contact} = \frac{S_e}{k_{fret} * VM} \\ SF_{fillet} = \frac{S_e}{VM} \end{cases} \quad (5.4)$$

The fatigue reduction factor ( $k_{fret}$ ) of 1.45 used in the calculation of  $SF_{contact}$  is in line with Juuma's experimental tests and Lanoue's (2008) studies.

#### 5.4.4 Typical results of fatigue strength calculations

Fig. 5-3 shows the distribution of equivalent alternating Von Mises stress in the contact (Fig. 5-2) and in the fillet. In this figure, it is also possible to observe the location of the two stress maximas. A shift in the location of fatigue failure occurs when the two maximas are equivalent. Actually, certain parameters contribute to stress reduction in the fillet while increasing it in contact and viceversa. Therefore, choosing the right parameters values meeting the stress criteria represents certainly a challenge. By optimizing the safety factors, it is possible to obtain optimal parameters' s values.

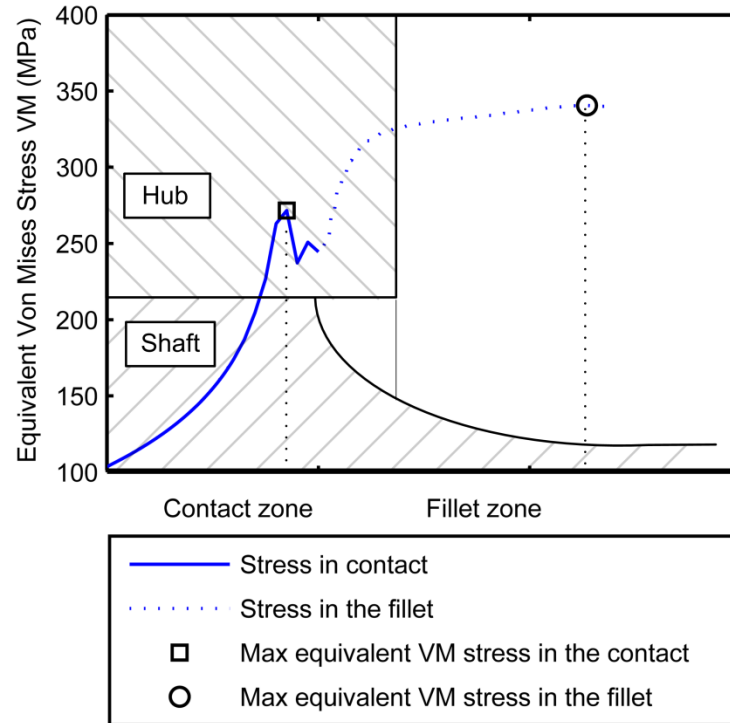


Figure 5-3 Distribution of VM in the critical zone

### 5.4.5 Problem definition

In designing interference fit assemblies, the main objective is to obtain a safety factor that meets the design criteria with minimum mass. Before using the SAMOO algorithm presented in Section 3, the optimization problem as well as the design space should be defined. A well-defined problem is essential to provide good results.

The optimization problem for this work consists in minimizing the assembly's mass while satisfying the safety factors. Since fatigue failures are from different nature in contact and fillet (fretting-fatigue and normal fatigue), they are interpolated by two separate kriging functions called  $SF_{contact}$  and in the fillet  $SF_{fillet}$ .

An analytic function which is rather ordinary to develop here is used to find out the mass value (W) in relation to its parameters. The tri-objectives optimization problem is therefore defined as follows:

- Maximize  $SF_{contact}(r_o, r_i, \delta_r, r, r_f, dL, r_s, L)$
- Maximize  $SF_{fillet}(r_o, r_i, \delta_r, r, r_f, dL, r_s, L)$
- Minimize  $W(r_o, r_i, \delta_r, r, r_f, dL, r_s, L)$

#### 5.4.6 Constraint equations (CE) related to the interference fit problem

While the objectives of this optimization problem are to obtain an infinite life,  $SF_{contact}$  (CE1) and  $SF_{fillet}$  (CE2) should be higher than 1. Since the shrink fit technique is used to assemble the hub and the shaft, the ability to fit the parts is mandatory. This process involves heating the hub to create a radial clearance between the two parts and thus allowing the insertion of the shaft in the hub. If the radial interference is too large, the temperature required for the assembly may be too high and the material's properties be affected. Therefore, radial interference ( $\delta_r$ ) was subjected to a maximal value (CE3).

It's necessary to set the length of the assembly ( $L_a$ ) for calculating his mass (CE4). For space reasons, the shaft's radius ( $r_s$ ) and hub's outer radius ( $r_o$ ) was constrained by a maximal dimension (CE5-6). There are also necessary constraints to consider in order to obtain a functionally generic model (CE7-CE9). Thus, the design domain used is given in table 5.4.

$$CE1 : SF_{contact} \geq 1$$

$$CE2 : SF_{fillet} \geq 1$$

$$CE3: \delta \leq .07mm$$

$$CE4: L_a = 85mm$$

$$CE5: r_h \leq 120 mm$$

$$CE6: r_s \leq 23 mm$$

$$CE7: r_i < r_s$$

$$CE8: r < r_s + r_f$$

$$CE9: r < r_h.$$

Tableau 5.4 Design space

Parameter	Inner bound (mm)	Outer Bound (mm)
L	30	60
$r_i$	2	12
$r_o$	35	120
r	25	28.5
$r_f$	8	12
$r_s$	19.5	23
dL	3	9
$\delta_f$	0.02	0.07

### 5.5 Sequential approximate multi-objective optimization (SAMOO)

In the optimization process using a finite element model, time is mostly spent on numerical simulations. Therefore, the main goal of Sequential Approximate Optimization (SAO) is to reduce calculations number while obtaining maximum information on the parameters' influence on the assembly behavior. The SAO principle consists of optimizing an interpolation function developed on the basis of a on a design of experiments (DOE). The number of finite elements calculations is thus defined by the number of design points. The SAMOO algorithm is designed to solve multi-objective structural problems using the finite element method. The workflow is illustrated in Fig. 5-4 and consists in the following steps:

1. Build a parameterized finite element model.
2. Determine the parameter boundaries in order to define the zone of interest.
3. Produce the design of experiments using the Optimal Latin Hypercube method.
4. Evaluate each design point with the FEA model.
5. Create a kriging interpolation.
6. Minimize errors using cross-validation.
7. Perform the optimization of the interpolation function using a multi-objective genetic algorithm (MOGA) solver.
8. Select candidates as a soft design from Pareto front.
9. Stop the algorithm if the accuracy criteria are satisfactory and if not, perform a sequential improvement

Notes:

- Soft design responses are calculated using the interpolation function.
- Hard design values are obtained from finite element analysis with the same parameters as the soft design.

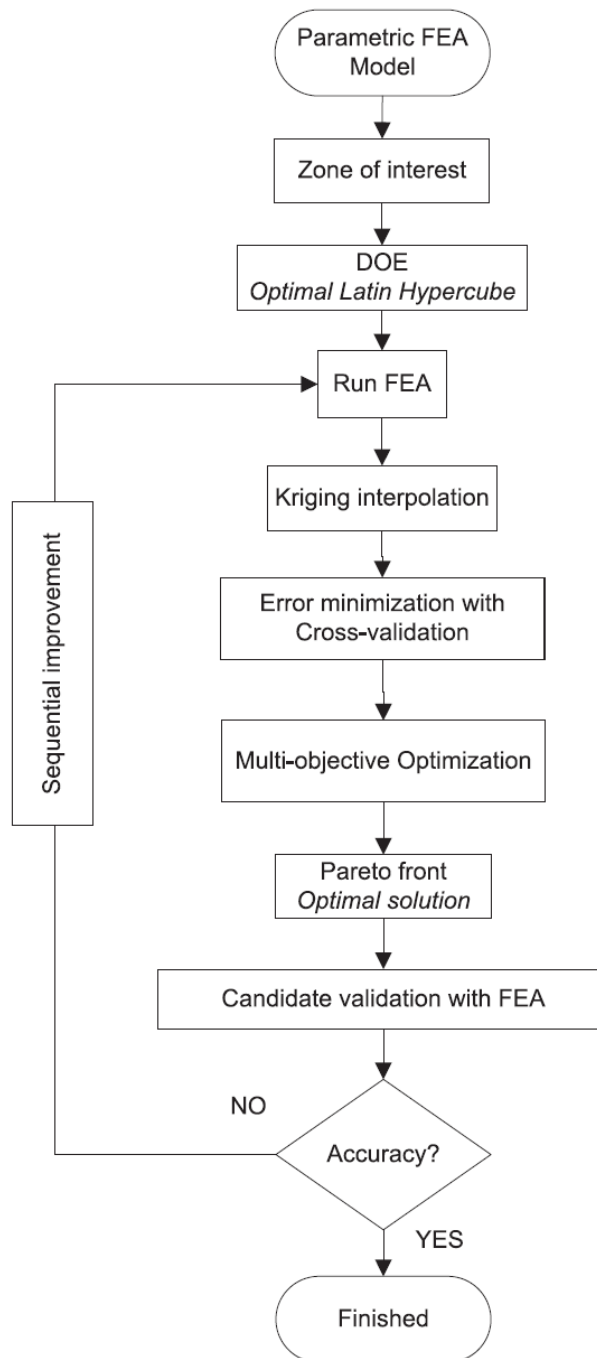


Figure 5-4 Structure of SAMOO algorithm

### 5.5.1 Zone of interest

Design space is limited by the constraints of the optimization problem. As explained above, sequential approximated optimization is performed on a DOE using a kriging function. Therefore, it is necessary to define a zone of interest where the DOE is created. As shown in Fig. 5-5 this zone must be located within a design space. The zone of interest may be modified whereas the design domain is fixed throughout the optimization. However, if the zone of interest is too large, the risk of error and the loss of sensitivity of the function in the optimal zone are increased. On the other hand, if the zone of interest is too specific, the optimal solution may lie outside of this zone.

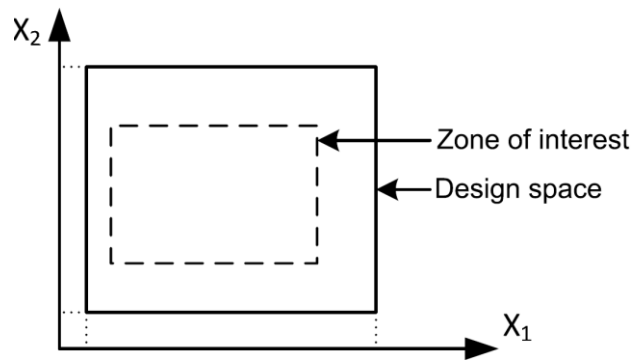


Figure 5-5 Design space and zone of interest

### 5.5.2 Design of Experiments

SAMOO uses a type of DOE called Latin Hypercube. Random Latin Hypercube (RLHC) was introduced by McKay et al. (1979) and is designed to efficiently cover the zone of interest. It discretizes this zone in “n” divisions. Therefore, it involves “n” experiments for which parameters values cannot be repeated for another design point. Typically, the design established by McKay does not guarantee an optimal point distribution in a zone of interest.

In an Optimal Latin Hypercube DOE showed in Fig. 5-6 point distribution is based on the Maximin Design principle. It provides optimal point distribution in the zone of interest (Johnson et al. 1990, Morris and Mitchell 1995). This technique consists in maximizing the minimum distance between any two points in the zone of interest to ensure that points are not too far apart and that there are no uncovered spaces.

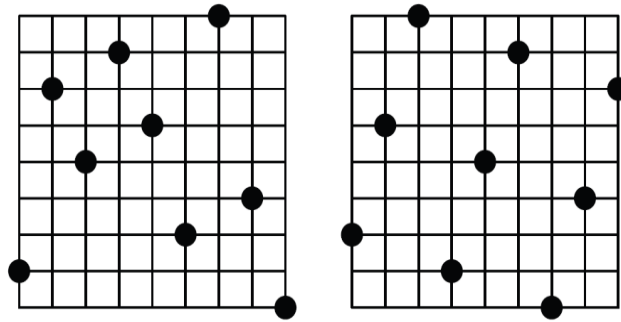


Figure 5-6 Random Latin Hypercube (left) and Optimal Latin HyperCube (right) designs

For computer experiments, the initial size of the DOE is 10 times the number of parameters as suggested by Bonte et al. (2010).

### 5.5.3 Kriging model

The main goal of a kriging function is to interpolate the zone of interest with minimal error. SAMOO uses the kriging method from the DACE (Design and Analysis of Computer Experiments) Matlab toolbox introduced by Lophaven et al. (2002). The kriging model used has the property of being exact at a design point. So the greater the density of points, the greater the function accuracy is.

Kriging function is a weighted linear combination of all the design points.

$$\hat{y}(x) = f(x)\beta^* + r(x)^T\gamma^* \quad (5.5)$$

where  $\beta^*$  is the generalized mean squared error,  $\gamma^*$  is the correlation factor,  $f(x)$  is the regression vector and  $r(x)$  is the correlation vector between  $x$  and the design points.

In SAMOO, it was determined that the linear regression and the Gaussian correlation offered better performances than other models. The use of gaussian functions are also suggested in the literature by Bonte et al. (2010), Lophaven et al. (2002). As described by Sacks et al. (1989), the regression model takes the following form:

$$f_1(x) = 1, f_2(x) = x_1, \dots, f_{m+1} = x_m \quad (5.6)$$

whereas the correlation model is expressed as:

$$\mathcal{R}(\theta, w, z) = \prod_{j=1}^n e^{-(\theta_j d_j^2)} \quad (5.7)$$

where  $m$  is the number of parameters,  $n$  is the number of design points,  $\theta$  is the correlation parameter and  $d_j$  is a matrix  $m \times n$  of the difference between design points. Correlation has a great influence on the model and deserves careful consideration. More development and justification could be added here but in order to lighten the text it was chosen not to.

#### 5.5.4 Error minimization using cross-validation

Obtaining good correlation between the real response and the kriging function outside the design points needs adjusting the correlation parameter  $\theta$ . The process of cross-validation is used to estimate the interpolation error of the kriging function. Therefore, error minimization can generate values of parameter  $\theta$  that will ensure a certain correlation of the kriging function in the zone of interest. To illustrate the benefits of error minimization by cross-validation, an application of the method on a random function is showed in Fig. 5-7.

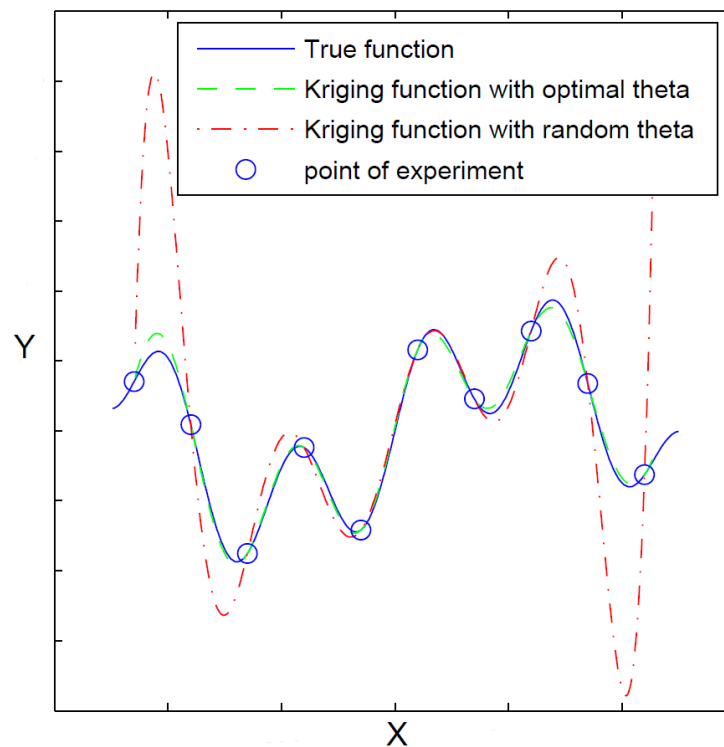


Figure 5-7 Improvement on a random function using Cross-Validation



The cross-validation method used in SAMOO is *one leave out one* (Kleijnen 2005). It consists in calculating the estimated response at a design point and developing a kriging function without the point in the DOE. The error between the estimated value  $\hat{y}_i$  and the real value  $y_i$  is then calculated for each point using root mean square error (RMSE) method:

$$RMSE(\theta) = \sum_{i=1}^N (\hat{y}_i(\theta) - y_i)^2 \quad (5.8)$$

Therefore, it's possible to obtain the optimal value of  $\theta$  by minimizing the RMSE function.

### 5.5.5 Multi-objective optimization algorithm

The main goal of multi-objective optimization is to create a set of soft potential candidates for the optimization problem. Since optimization is performed on interpolated functions, there are several local minimas. In this case, it is recognized that genetic algorithms offer better results than gradient type algorithms, which highly depend on the starting point. In SAMOO, the optimizer used is MOGA available in Altair HyperStudy V10. The MOGA algorithm also accepts non-linear constraints.

The MOGA algorithm uses a “non-dominated” strategy. It means that a solution is non-dominated if an objective cannot be improved without degrading others. The combination of all non-dominated solutions forms the Pareto front. For accurate results, solutions should (Deb et al. 2002):

1. converge to the Pareto front.
2. be spread along the Pareto front.

In sequential optimization, potential candidates are the optimal values of the kriging functions (soft design). Therefore, these values must be validated by finite element analysis (hard design). If the error between the soft and hard design points are too high, a sequential improvement must be performed in order to obtain an accurated kriging function.

### 5.5.6 Sequential improvement

The main goal of the sequential improvement is to increase the accuracy of the kriging function in order to meet the criteria. (Bonte et al. 2010) In SAMOO, the accuracy is calculated by the mean absolute deviation between the soft design and hard design points. The sequential improvement used in SAMOO algorithm consists in adding the hard design points to the current Design Of Experiments. By increasing the number of design points in the Pareto Front region, the function will be also more accurate in this region.

## 5.6 Results of SAMOO on interference fit assembly optimization problem

As the technical aspects of the SAMOO algorithm are now set, the next sections present its application on the interference fit problem.

### 5.6.1 ANOVA study

Applying SAMOO algorithm on the interference fit assembly generates some valuable informations for decision making. From the established DOE, the parameters' influence can be observed using an analysis of variance (ANOVA). In the case of a preliminary design with a high number of parameters, it is possible to validate the relevance of every parameter in the problem. The ANOVA analysis was performed using the Moving Least Square Model from Altair Hyperstudy V10.

By using this method for the fretting problem (Fig. 5-8), the following observations can be made on the selected zone of interest:

- The most influential parameters on safety factor in fillet are the inside radius ( $r_i$ ) and the radius of the shaft ( $r_s$ ).
- item The most influential parameters on safety factor in the contact are the radius at the contact interface ( $r$ ) and the radius of the shaft ( $r_s$ ).

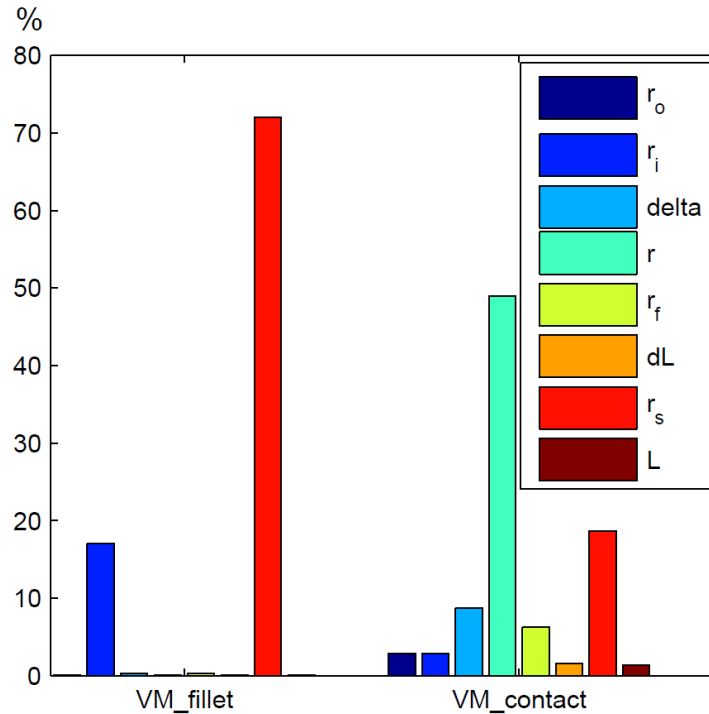


Figure 5-8 Influence of each parameters on  $VM_{contact}$  and  $VM_{fillet}$

### 5.6.2 Accuracy of SAMOO algorithm

In order to show the accuracy of SAMOO algorithm, the method was applied to the optimization problem of interference fit. The results are illustrated in Fig.5-9.

Design points are represented by diamonds. Soft design points obtained through the optimization of the interpolation functions are represented by squares. Hard designs of potential candidates are represented by triangles. Hard design consists in validating potential candidates using the structural model by FEA.

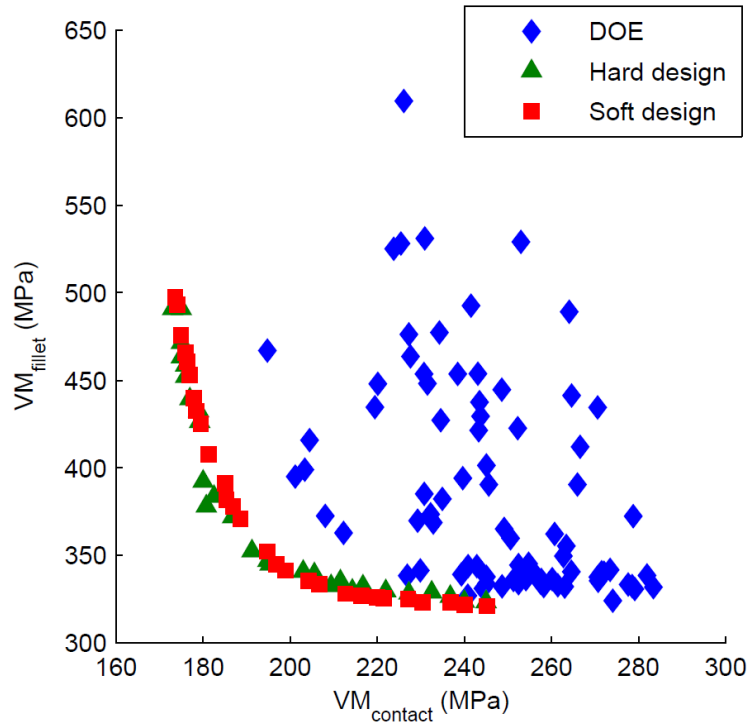


Figure 5-9 Results of SAMOO on a stress minimization in contact and in the fillet

In this application, one can see that SAMOO is highly accurate and it offers better candidates than the entire set of design points. Actually, the mean absolute deviation between potential candidates and hard design is 0.6 % in the contact and 1.2 % in the fillet , which confirms the accuracy of the interpolation model.

### 5.6.3 Choosing the optimal solution

Since the results obtained using SAMOO are all optimal solutions, choosing one can be a difficult task. In fact, The multi-objective approach allows to select solution based on other design criterias.

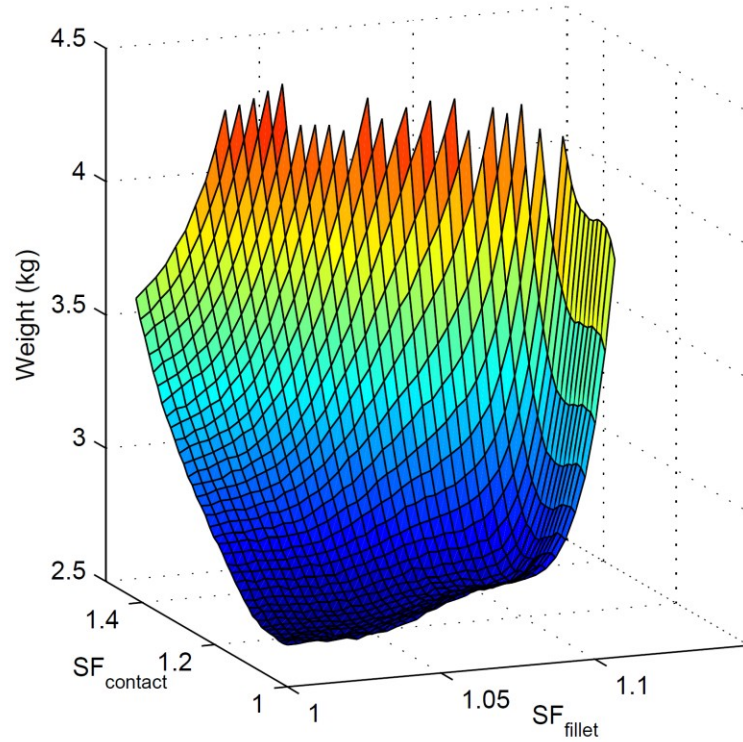


Figure 5-10 Isometric view of the Pareto front

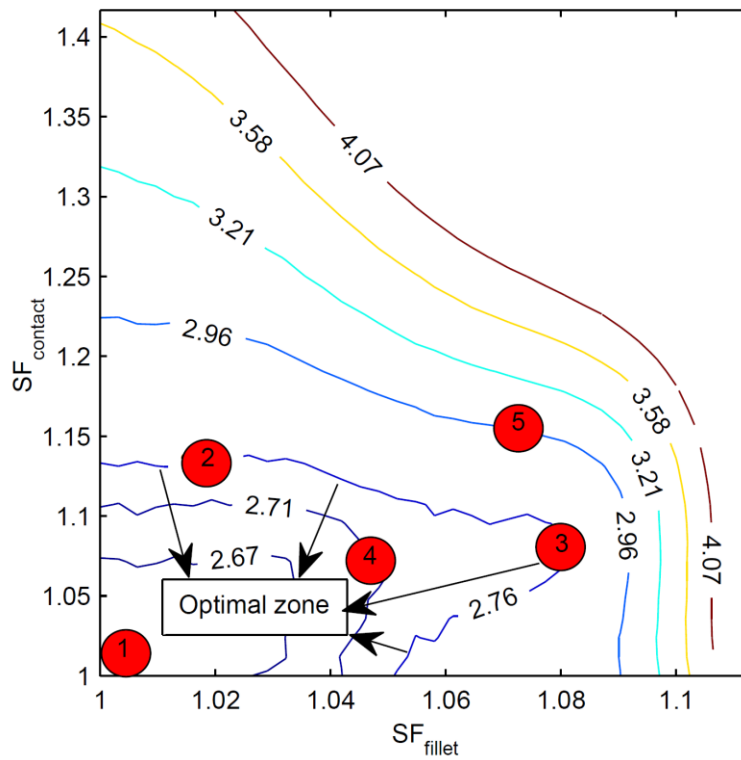


Figure 5-11 Level curves of the weight in the Pareto front

In the case of interference fit assemblies, Lanoue et al. (2011) studies have shown that there is additional uncertainty due to the estimated fatigue reduction factor in contact  $k_{fret}$ . This additional uncertainty of the fatigue strength in contact was estimated at +/-10% (Lanoue et al. 2001). It is than natural to prioritize a safety factor higher for this parameter if it barely affects the assembly's mass. Thus, an optimization was performed by considering the maximum stress in the fillet and the stress in contact  $SF_{contact}$  and  $SF_{fillet}$ . The results are shown in Fig. 5-10.

As a different representation of the results, Fig. 11 shows that for a mass less than 2.76, stress gradient remains low, which is the opposite for a greater mass values. Consequently, choosing a optimal solution inferior to this value is beneficial.

Among all the possible solutions, those that come to attention are shown in Fig. 5-11 and table 5.5. Solution 1 has the minimal mass allowed by optimization constraints. Solution 5 has the highest safety factor in contact. At this point, it's possible to see that the mass of the assembly is still minimally affected. Solution 2 to 4 seems to be a good compromise between the three objectives. In reality, a final decision between these solutions would be taken considering other design criteria that might arose in the design process.

From our point of view, solution 2 was selected as it does not affect the mass while assuring additional security in contact. Also, choosing a higher safety factor complies to the uncertainty caused by the dimensional tolerance range affecting the value of the interference and thus the real contact pressure. Table 6 shows the parameter values for solution 2.

Tableau 5.5 Optimal solution illustrated in Fig. 5-11

Solution	Weight ( Kg)	SF <sub>contact</sub>	SF <sub>contact</sub>
1	2.650	1	1
2	2.76	1.13	1.02
3	2.76	1.08	1.08
4	2.705	1.07	1.047
5	3.003	1.16	1.07

Tableau 5.6 Parameter value of solution 2

L	$r_i$	$r_o$	r	$r_f$	$r_s$	dL	$\delta r$
30.18	12.00	35.00	28.50	9.34	22.59	3.02	0.07

### 5.6.4 Discussion

Multi-objective optimization using genetic algorithm on kriging functions requires 80 000 evaluations and 183 iterations. The ability to estimate the accurate response leads to providing optimal solutions at a computationally low cost. The optimization of this structural problem would not be possible without the use of interpolation models. The optimization using SAMOO performed on the interference fit assembly problem required 97 finite element calculations.

The ANOVA has also generated a sensitivity study applied on the DOE. This study gives the engineer valuable information about the problem at hand that he would not get from a standard optimization algorithm. Multi-objective optimization produces a set of optimal solutions and gives arguments for selecting a candidate. If the problem had been defined with the objective of minimizing mass with stress less than  $S_e$  solution 1 in Table 5.5 would have been selected. But nothing would suggest that for a similar weight, it is possible to increase the safety factor on the maximum equivalent stress in contact. The choices are difficult and the appropriate post processing of the multi-objective optimization can influence the decision-making.

Finally, further experimental tests are needed to validate the fretting reduction factor ( $k_{fret}$ ). In the case studied, it was adopted from a previous study. However, when this factor is unknown or uncertain, it is always possible to estimate it, then perform experimental tests and choose the optimal solution that meets the criteria of this new  $k_{fret}$ . If the solutions are within the zone of interest, soft design can be obtained without having to make additional finite element calculations.

### 5.6.5 Conclusion

This work has shown that the SAMOO algorithm can be used to obtain a set of optimal solutions, but mostly that it can be an important design tool. The benefits of using a design of experiments and an interpolation method consist in supplying engineer with valuable informations in order to identify an optimal solution with fewer finite element calculations.

In this paper, applying the SAMOO algorithm to an interference fit assembly returns good results. First, modeling interference fit assembly with finite elements was presented. Then, the SAMOO algorithm was detailed to show the relevance of each step. Finally, it was demonstrated that its application could solve the specific problem of interference fit assembly. In fact, by suggesting several potential candidates, it provides an increased safety factor in the contact zone while the mass is kept to a low value.

Although the SAMOO algorithm has been applied here to a fretting problem, its flexibility allows it to be used for any structural multi-objective optimization problem. Its competitive edge emerges when simulations are non-linear and/or computational resources are limited. By suggesting a set of optimal solutions, it stands as a very useful decision-making tool for engineers.

### **5.6.6 Acknowledgements**

We thank Altair Engineering Canada for the support in the integration of different software modules as well as NSERC for funding this research.



## References

- M. Bonte, L. Fourment, T-H. Do, A. van den Boogaard, and J. Hutink. Optimization of forging processes using finite element simulations. *Structural and Multidisciplinary Optimization*, 42(25):797–810, 2010.
- K. Deb, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- J. Forsberg and L. Nilsson. On polynomial response surfaces and kriging for use in structural optimization of crashworthiness. *Structural and Multidisciplinary Optimization*, 29(3):232–243, 2005.
- J. H. Jacobs, L. F. P. Etman, F. van Keulen, and J. E. Rooda. Framework for sequential approximate optimization. *Structural and Multidisciplinary Optimization*, 27 (5):384–400, 2004.
- M. E. Johnson, L. M. Moore, and D. Ylvisaker. Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, 26(2):131–148, 1990.
- T. Juuma. Torsional fretting fatigue strength of a shrink fitted shaft. *Wear*, 231(2):310–318, 1999.
- T. Juuma. Torsional fretting fatigue strength of a shrinkfitted shaft with a grooved hub. *Tribology International*, 33(8):537–543, 2000.
- D. W. Kim and J. Lee. An improvement of kriging based sequential approximate optimization method via extended use of design of experiments. *Engineering Optimization*, 2010.
- J. P. C. Kleijnen. An overview of the design and analysis of simulation experiments for sensitivity analysis. *European Journal of Operational Research*, 164(2):287–300, 2005.
- M. Kubota, S. Kataoka, and Y. Kondo. Effect of stress relief groove on fretting fatigue strength and index for the selection of optimal groove shape. *International Journal of Fatigue*, 31(3):439–46, 2009.
- F. Lanoue. Modelisation et simulation des assemblages mecaniques par frettages sollicites en fatigue par la methode des elements finis, M. A. Sc., 2008.

- F. Lanoue, A. Vadean, and B. Sanschagrín. Finite element analysis and contact modelling considerations of interference fits for fretting fatigue strength calculations. *Simulation Modelling Practice and Theory*, 17, 1587–1602, 2009.
- F. Lanoue, A. Vadean, and B. Sanschagrín. Fretting fatigue strength reduction factor for interference fits. *Simulation Modelling Practice and Theory*, 2011. doi: 10.1016/j.simpat.2011.05.004.
- S. Lophaven, H. Nielsen, and J. Sondergaard. Dace : A matlab kriging toolbox. *Technical Report IMM-TR-2002-12*, 2002.
- J. J. Madge, S. B. Leen, I. R. McColl, and P. H. Shipway. Contact-evolution based prediction of fretting fatigue life: Effect of slip amplitude. *Wear*, 262(9-10):1159–1170, 2007.
- M. D. McKay, R. J. Beckman, and W. J. Conover. 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, 1979.
- M. D. Morris and T. J. Mitchell. Exploratory designs for computational experiments. *Journal of Statistical Planning and Inference*, 43(3):381–402, 1995.
- K. Nishioka and H. Komatsu. Researches on increasing the fatigue strength of press-fitted shaft assembly. *Bulletin of the Japan Society of Mechanical Engineers*, 10(42):880–9, 1967.
- J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4):409–435, 1989.
- C. Schafer and E. Finke. Shape optimisation by design of experiments and finite element methods-an application of steel wheels. *Structural and Multidisciplinary Optimization*, 36(5):477–491, 2008.
- W.C. M. van Beers and J. P. C. Kleijnen. Customized sequential designs for random simulation experiments: Kriging metamodeling and bootstrapping. *European Journal of Operational Research*, 186(3):1099–1113, 2008.

## CHAPITRE 6 DISCUSSION GÉNÉRALE

Le principal objectif de ce mémoire est d'établir une méthode de conception optimale des assemblages par interférences soumis à des sollicitations en fatigue. Cet article présente l'application de l'algorithme d'optimisation séquentielle présenté au Chapitre 3 sur un problème d'assemblage par interférence. Au préalable de la résolution du problème d'optimisation, il était nécessaire de définir un modèle numérique robuste.

Ainsi, une étude des différents essais expérimentaux présentés dans la littérature a été effectuée. Parmi les configurations géométriques qui ont été étudiées, celle d'un arbre avec épaulement assemblé à un moyeu avec une partie en porte-à-faux a été celle qui a offert la meilleure résistance à la fatigue. Une fois la configuration choisie, la modélisation par éléments finis de l'assemblage par interférence soumis à des sollicitations de torsion alternée a été effectuée selon les recommandations établies par Lanoue [11]. En effet, ce modèle avait été comparé et validé avec les essais de Juuma [5]. Le logiciel retenu pour modéliser l'assemblage est la version APDL (*ANSYS Parametric Design Language*) du logiciel ANSYS v11. Celui-ci est plus souple que la version *Workbench* et permet un meilleur contrôle de la qualité du maillage dans la région critique. De plus, une macro permettant de lire les paramètres du modèle dans un tableau contenant les points d'expériences a été écrite. Ensuite, avec l'état de contraintes obtenus des calculs par éléments finis, la contrainte équivalente de Von Mises est calculée et écrite dans un fichier de sortie. Ce dernier est utilisé pour la modélisation de la fonction de krigeage.

En ayant un modèle par éléments finis paramétré et robuste, un problème d'optimisation a été défini et présenté dans l'article. Ce problème consiste à minimiser la masse et les contraintes équivalentes de Von Mises présentes dans le contact et le congé. Le chargement appliqué au modèle est celui que Juuma [5] a utilisé pour ses essais expérimentaux. En effet, puisqu'une corrélation entre le modèle d'éléments finis et les résultats expérimentaux a été fait par Lanoue, il était possible d'utiliser un facteur de réduction de vie en fatigue ( $k_{fret}$ ).

Considérant le temps d'un calcul par éléments finis, la méthode d'optimisation développée et présentée au chapitre 3 a été utilisée. Celle-ci permet de résoudre des problèmes structurels

efficacement. En effet, la réalisation de l'optimisation a nécessité 98 calculs par éléments finis et environ 80 000 évaluations des fonctions de krigeages.

L'utilisation d'un algorithme multi-objectifs pour résoudre le problème de conception optimale des assemblages par interférence permet de générer des solutions optimales diversifiées. Ainsi, l'ingénieur peut donc obtenir une conception mieux adaptée à ses besoins. Par exemple, les résultats de l'optimisation ont permis d'observer que pour un espace de conception et chargement définis, une solution ayant un facteur de sécurité supérieur à 1.15 affectait considérablement la masse de l'assemblage. Toutefois, pour un facteur de sécurité inférieur à cette valeur, le choix de la solution optimale peut être basé sur d'autres critères de conception puisqu'elle n'affecte peu la masse. Pour ce problème spécifique, les non-linéarités causées par le fretting introduisent une incertitude supplémentaire dans les calculs de vie en fatigue. Ainsi, il était donc souhaitable d'obtenir un facteur de sécurité supplémentaire dans le contact. C'est donc pourquoi une solution préconisant une contrainte équivalente de Von Mises plus basse dans le contact a été choisie.

Enfin, cet article a démontré la pertinence de l'utilisation d'un algorithme d'optimisation séquentielle multi-objectifs. Sans l'approche séquentielle, la résolution du problème de conception de l'assemblage par interférence n'aurait pas été possible étant donné les ressources informatiques disponibles.

## CHAPITRE 7 CONCLUSION

Ce travail de recherche avait comme objectif de proposer une méthode de conception optimale des assemblages par interférence soumis à des sollicitations de fatigue.

Pour cela, une revue de littérature sur la conception des assemblages par interférence soumis à des sollicitations de fatigue a été effectuée. Parmi les différentes configurations géométriques présentées dans la littérature, celle d'un arbre avec épaulement assemblé avec un moyeu et une partie en porte-à-faux a retenu l'attention. En effet, différents auteurs ont conclu avec leurs essais expérimentaux que c'était la configuration qui offrait les meilleurs résultats en fatigue. Ensuite, les différents aspects du calcul de la vie en fatigue des assemblages par interférence avec la méthode des éléments finis ont été présentés. Ceci a donc constitué la base du modèle d'élément fini générique utilisé dans l'article présenté au Chapitre 5. Enfin, les différentes méthodes de conception de l'assemblage par interférence étudiées ont tous démontré des lacunes. En effet, elles ne permettent pas d'obtenir la valeur de tous les paramètres en fonction d'un critère de conception, mais seulement ceux sélectionnés. Il a donc été déterminé de conserver tout les paramètres utiles à la modélisation de l'assemblage et d'en effectuer une optimisation.

Le second chapitre présente donc l'état des connaissances sur l'optimisation séquentielle. Ce chapitre a présenté les différents plans d'expériences, fonction d'interpolation et améliorations séquentielles utilisées. Tout d'abord, une comparaison des performances établies dans la littérature entre les plans d'expériences de type surface de réponse et uniforme ont permis de constater que pour les problèmes déterministes, les plans uniformes étaient plus adéquats. Ensuite, une comparaison entre les modèles d'interpolation quadratique et de krigeage a établie que l'utilisation du krigeage donnait de meilleurs résultats mais était plus complexe à programmer. Deux type d'amélioration séquentielle ont été présentés : la stratégie de déplacements des limites et celle de remplissage. La méthode de stratégie de déplacement de limite consiste à déplacer la zone d'intérêt vers la solution recherchée tandis que la méthode de remplissage consiste à ajouter des points au plan d'expériences afin d'améliorer la qualité du modèle d'interpolation.

Ainsi, le Chapitre 3 présente donc l'algorithme d'optimisation séquentielle multi-objectifs (OSMO). Tout d'abord, il a été établi de diviser l'algorithme en 2 phases. La première phase, exploratoire, consiste à générer une zone d'intérêt adéquate à la phase recherche. Une méthode de modifications des limites basée sur une analyse statistique de conformité des points d'expériences aux contraintes a été développée dans un but d'améliorer la qualité de la zone d'intérêt. Ensuite, la seconde phase, nommée recherche, consiste à améliorer la précision de la fonction d'interpolation et ainsi, trouver la solution optimale. Le plan d'expériences utilisé est le *Optimal Latin Hyper-Cube* et le modèle d'interpolation est le *Krigeage*. Étant donné la disponibilité du logiciel, la totalité de la méthode a été programmée avec la suite Altair HyperWorks V10. L'algorithme d'optimisation choisi est un algorithme génétique multi-objectifs (MOGA). Après l'obtention des résultats de ce dernier, il est nécessaire de valider les candidats potentiels avec un modèle d'éléments finis. Si l'écart absolu est inférieur à un critère de convergence établi initialement, l'algorithme aura convergé, toutefois s'il est supérieur une amélioration doit être effectuée. En ce sens, pour les problèmes multi-objectifs, puisque les candidats potentiels et leur validation sont déjà calculés, la méthode propose d'ajouter ces points au plan d'expériences existant. Ainsi, le modèle sera donc plus précis dans cette zone et par conséquent, la convergence sera atteinte.

Pour valider cette démarche, une application de l'algorithme sur un problème académique est présentée au Chapitre 4. L'utilisation du problème académique permet donc une meilleure analyse considérant que le temps de calcul est négligeable et que la solution est connue. Le problème retenu consiste à minimiser la masse et le déplacement d'une structure à 10 barreaux soumis à des contraintes maximales de 25 ksi dans les barreaux. L'application de la phase exploratoire a montré les bienfaits de la méthode de modification des limites la zone d'intérêt. Ensuite, la phase recherche a été appliquée sur une version mono-objective du problème afin de pouvoir le comparer avec un algorithme d'optimisation séquentielle présenté dans la littérature récente. Ainsi, les résultats ont démontré que l'algorithme OSO a convergé plus rapidement et a offert de meilleurs résultats que ceux présentés dans la littérature. Sur le problème multi-objectifs, la phase recherche a permis également de constater que l'algorithme était efficace et permettait de converger vers le Front de Pareto théorique avec précision. Toutefois, les résultats ont révélé que l'algorithme MOGA était instable et présentait des lacunes incontrôlables.

Le chapitre 5 constitue la raison de ce travail, soit l'application de l'algorithme MOGA à un problème de conception tri-objectifs des assemblages par interférence soumis à des sollicitations de torsion. Le modèle d'éléments finis utilisé a été établi en considérant les différents aspects établis dans le chapitre 1. Étant donné l'utilisation d'un plan d'expériences dans l'algorithme, une analyse de sensibilité selon la méthode ANOVA a permis de constater quels paramètres étaient les plus influents selon les spécifications du problème. Ensuite, les résultats de l'optimisation ont montré que l'algorithme OSMO était efficace et précis. De plus, en considérant une approche multi-objectifs, il a été possible d'obtenir un facteur de sécurité supplémentaire dans le contact sans toutefois pénaliser la masse de manière considérable. Un avantage non-négligeable à l'ère où l'ingénieur est souvent confronté à des critères de conceptions de plus en plus rigoureux.

L'algorithme présenté dans ce travail est en phase préliminaire. Ainsi, plusieurs points ont été relevés et sont intéressants à présenter pour les futurs travaux. Tout d'abord, la programmation d'un algorithme de génération de plan d'expériences OLHC qui prendrait en compte les contraintes pourrait donner beaucoup plus de souplesse à l'utilisateur dans le cas de domaine de conception complexe. De plus, il serait intéressant d'utiliser un algorithme d'optimisation multi-objectifs qui saurait avoir les bienfaits de l'algorithme de type gradient et de type génétique en même temps. Enfin, l'algorithme OSMO n'est pas automatique, c'est-à-dire qu'il nécessite l'intervention d'un utilisateur pour effectuer les diverses transitions entre les étapes, la programmation de celui-ci pourrait donc être souhaitable.

**BIBLIOGRAPHIE**

- [1] A. Bazergui, *Résistance des matériaux*, 3e éd. -- ed. [Montréal]: Presses internationales Polytechnique, 2002.
- [2] R. L. Norton, *Machine design : an integrated approach*, 3rd ed. Upper Saddle River, N.J.: Pearson Prentice Hall, 2006.
- [3] J.-S. Jang, *et al.*, "3-Dimensional finite element analysis of the residual stress by cold expansion method and interference fit," 2006, pp. 1079-1082.
- [4] T. Hattori, *et al.*, "Torsional Fatigue Strength of a Shrink Fitted Shaft," *Bulletin of the JSME*, vol. 24, pp. 1893-1900, 1981.
- [5] T. Juuma, "Effect of Contact Pressure, Slip Amplitude and Geometry on the Torsional Fretting Fatigue Strength of a Shrink-fitted Shaft," in *11th International Congress and Exhibition on Condition Monitoring and Diagnostic Engineering Management*, Novotel Launceston, Tasmania, Australia, 1998, pp. 477 -485.
- [6] T. Juuma, "Torsional fretting fatigue strength of a shrink-fitted shaft," *Wear*, vol. 231, pp. 310-318, Jul 1999.
- [7] T. Juuma, "Torsional fretting fatigue strength of a shrink-fitted shaft with a grooved hub," *Tribology International*, vol. 33, pp. 537-543, Aug 2000.
- [8] K. Nishioka and H. Komatsu, "Researches on increasing the fatigue strength of press-fitted shaft assembly," *Bulletin of the Japan Society of Mechanical Engineers*, vol. 10, pp. 880-9, 1967.
- [9] K. Hirakawa, *et al.*, "The analysis and prevention of failure in railway axles," *International Journal of Fatigue*, vol. 20, pp. 135-144, Feb 1998.
- [10] M. Kubota, *et al.*, "Effect of stress relief groove on fretting fatigue strength and index for the selection of optimal groove shape," *International Journal of Fatigue*, vol. 31, pp. 439-46, 2009.
- [11] F. Lanoue, "Modélisation et simulation des assemblages mécaniques par frettages sollicités en fatigue par la méthode des éléments finis," *Maîtrise, Génie Mécanique*, École Polytechnique de Montréal, 2008.



- [12] F. Lanoue, *et al.*, "Finite element analysis and contact modelling considerations of interference fits for fretting fatigue strength calculations," *Simulation Modelling Practice and Theory*, vol. 17, pp. 1587-1602, 2009.
- [13] N. G. Cormier, *et al.*, "Aggressive submodelling of stress concentrations," *International Journal for Numerical Methods in Engineering*, vol. 46, pp. 889-909, 1999.
- [14] B. Weber, "Fatigue multiaxiale des structures industrielles sous chargement quelconque," Institut National des Sciences Appliquées de Lyon, 1999.
- [15] J. J. Madge, *et al.*, "Contact-evolution based prediction of fretting fatigue life: Effect of slip amplitude," *Wear*, vol. 262, pp. 1159-1170, 2007.
- [16] K. Nishioka and K. Hirakawa, "Fundamental investigations of fretting fatigue, V. The effect of relative slip amplitude," *Bulletin of the Japan Society of Mechanical Engineers*, vol. 12, pp. 692-7, 1969.
- [17] D. Castagnetti and E. Dragoni, "Optimal aspect ratio of interference fits for maximum load transfer capacity," *Journal of Strain Analysis for Engineering Design*, vol. 40, pp. 177-184, 2005.
- [18] J. Sacks, *et al.*, "Design and Analysis of Computer Experiments," *Statistical Science*, vol. 4, pp. 409-435, 1989.
- [19] J. H. Jacobs, *et al.*, "Framework for sequential approximate optimization," *Structural and Multidisciplinary Optimization*, vol. 27, pp. 384-400, Jul 2004.
- [20] J. Forsberg and L. Nilsson, "On polynomial response surfaces and Kriging for use in structural optimization of crashworthiness," *Structural and Multidisciplinary Optimization*, vol. 29, pp. 232-243, 2005.
- [21] B. Clément, "Note de cour (Mth 6301)," 2006.
- [22] D. C. Montgomery, *Design and analysis of experiments*, 6th ed. Hoboken, NJ: John Wiley & Sons, 2005.
- [23] C. Schafer and E. Finke, "Shape optimisation by design of experiments and finite element methods-an application of steel wheels," *Structural and Multidisciplinary Optimization*, vol. 36, pp. 477-491, Nov 2008.

- [24] R. Unal, *et al.*, "Response surface model building and multidisciplinary optimization using d-optimal designs," *AIAA-98-4759*, 1998.
- [25] T. W. Simpson, *et al.*, "Spatial correlation metamodels for global approximation in structural design optimization.," *ASME Design Engineering Technical Conference*, 1998.
- [26] T. W. Simpson, *et al.*, "Metamodels for computer-based engineering design: survey and recommendations," *Engineering with Computers*, vol. 17, pp. 129-150, 2001.
- [27] M. D. McKay, *et al.*, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, pp. 239-245, 1979.
- [28] M. E. Johnson, *et al.*, "Minimax and Maximin Distance Designs," *Journal of Statistical Planning and Inference*, vol. 26, pp. 131-148, Oct 1990.
- [29] M. D. Morris and T. J. Mitchell, "Exploratory Designs for Computational Experiments," *Journal of Statistical Planning and Inference*, vol. 43, pp. 381-402, Feb 1995.
- [30] F. Xiong, *et al.*, "Optimizing Latin hypercube design for sequential sampling of computer experiments," *Engineering Optimization*, vol. 41, pp. 793-810, 2009.
- [31] M. Zhao and W.-c. Cui, "Application of the optimal Latin hypercube design and radial basis function network to collaborative optimization," *Journal of Marine Science and Application*, vol. 6, pp. 24-32, 2007.
- [32] M. Bonte, *et al.*, "Optimization of forging processes using Finite Element simulations," *Structural and Multidisciplinary Optimization*, vol. 42, pp. 797-810, 2010.
- [33] T. Goel, *et al.*, "Comparing error estimation measures for polynomial and kriging approximation of noise-free functions," *Structural and Multidisciplinary Optimization*, vol. 38, pp. 429-442, 2009.
- [34] H. B. N. Soren N. Lophaven, Jacob Sondergaard, "DACE : A Matlab Kriging Toolbox," *Technical Report IMM-TR-2002-12*, 2002.
- [35] J. P. C. Kleijnen, "An overview of the design and analysis of simulation experiments for sensitivity analysis," *European Journal of Operational Research*, vol. 164, pp. 287-300, Jul 16 2005.

- [36] D. W. Kim and J. Lee, "An improvement of Kriging based sequential approximate optimization method via extended use of design of experiments," *Engineering Optimization*, 2010.
- [37] S. Dominique, "Algorithme intelligent d'optimisation d'un design structurel de grande envergure," Ph.D. NR62352, Ecole Polytechnique, Montreal (Canada), Canada, 2010.
- [38] K. Deb, *et al.*, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *Ieee Transactions on Evolutionary Computation*, vol. 6, pp. 182-197, Apr 2002.
- [39] R. T. Haftka and Z. Gürdal, *Elements of structural optimization*, 3rd rev. and expanded ed. Dordrecht ; Boston: Kluwer Academic Publishers, 1992.
- [40] I. Elishakoff, *et al.*, "Structural Design under Bounded Uncertainty - Optimization with Anti-Optimization," *Computers & Structures*, vol. 53, pp. 1401-1405, Dec 17 1994.
- [41] J. E. Herencia and R. T. Haftka, "Structural optimization with limited number of element properties," *Structural and Multidisciplinary Optimization*, vol. 41, pp. 817-820, May 2010.

## ANNEXE 1– Méthode de sous-modélisation

Pour effectuer la sous modélisation, un algorithme a été développé par Cormier [13] et traduit par Lanoue [11]. Les conditions limites imposées au sous-modèle sont donc le déplacement  $u$  des nœuds de sa frontière interpolés dans le modèle global. Voici la description de l'algorithme tel que présenté par Lanoue [11] :

1. *Analyse sur un modèle global d'une série de 3 maillages systématiquement raffinés et nommés : grossier (c - coarse), moyen (m) et fin (f).*
2. *Vérification de la convergence des déplacements à l'interface de sous-modélisation.*
  - a. *On dit que les déplacements convergent si :  $|u_i^c - u_i^m| > |u_i^m - u_i^f|$  où  $u$  est le déplacement des différents nœuds (i) de la région frontière*
  - b. *Les déplacements sont convergés si :  $|u_i^m - u_i^f| < \varepsilon$  (ex :  $\varepsilon = 5\%$ )*
  - c. *Si le critère de convergence n'est pas respecté, refaire des analyses supplémentaires selon l'étape 1 avec un maillage plus fin*
3. *Si l'étape 2 converge, un premier sous-modèle avec un maillage plus fin est testé en utilisant les résultats de déplacement du dernier et de l'avant-dernier maillage.*
  - a. *Les résultats sont convergés (donc l'erreur du aux conditions aux limites est jugée négligeable). Si la différence de contrainte maximale dans le sous-modèle varie moins que le critère défini en fonction des déplacements imposés à l'interface de contact (ic) :  $|\sigma_{\max}^{fic} - \sigma_{\min}^{mic}| < \epsilon$*
  - b. *Si le critère de convergence n'est pas respecté, recommencer depuis l'étape 1.*
4. *Si l'étape 3 converge, on peut raffiner le sous-modèle, puisque les conditions à l'interface de sous-modélisation sont jugées convergées.*
  - a. *L'étude de convergence se fait de la même manière que spécifiée en 2a et 2b avec un raffinement de maillage systématique par un facteur 2 sur la taille des éléments.*
  - b. *Si les contraintes maximales ne convergent pas aussi rapidement que souhaité par le critère défini, un autre sous-modèle à l'intérieur du premier peut être effectué en effectuant à nouveau la procédure complète de l'étude de convergence*

- c. *Si plusieurs ( $N$ ) sous-modèles successifs sont effectués, les erreurs à l'interface de sous-modélisation s'accumule et le critère de convergence pour chaque étape devrait être fixé selon :  $\varepsilon \rightarrow \frac{\varepsilon}{N}$*

## ANNEXE 2 – Structure à 10 Barreaux

Les équations analytique de la structure à 10 barreaux sollicitée en porte à faux sont connues. La géométrie est présentée à la figure et les paramètres utilisés dans le Tableau A.1

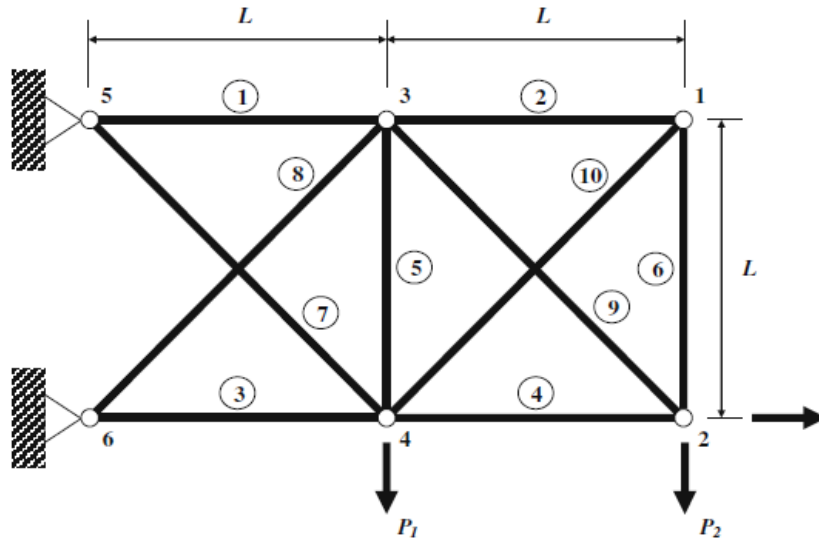


Figure A-1 Structure à 10 barreaux [41]

Tableau A.1 Définition des paramètres

Paramètre	Définition
$L$	Longueur de la structure
$N_i$	Force axiale présente dans les barreaux
$P_i$	Force appliquée à la structure
$A_i$	Aire de la section du barreau
$\sigma$	Contrainte normale
$E$	Module de Young
$W$	Masse de la structure.
$\rho$	Masse volumique
$\delta_2$	Déplacement du nœud 2.

Les forces axiales présentes dans les barreaux ( $N_i, i = 1, \dots, 10$ ) sont définies par les équations suivantes :

$$N_2 = -\frac{\sqrt{2}}{2} N_{10}$$

$$N_3 = -P_1 - 2P_2 + P_3 - \frac{\sqrt{2}}{2} N_8$$

$$N_4 = -P_2 + P_3 - \frac{\sqrt{2}}{2} N_{10}$$

$$N_5 = -P_2 - \frac{\sqrt{2}}{2} N_8 - \frac{\sqrt{2}}{2} N_{10}$$

$$N_6 = -\frac{\sqrt{2}}{2} N_{10}$$

$$N_7 = \sqrt{2}(P_1 + P_2) + N_8$$

$$N_8 = -\frac{b_1 - a_{12}N_{10}}{a_{11}}$$

$$N_9 = \sqrt{2}P_2 + N_{10}$$

$$N_{10} = -\frac{b_2 a_{11} - b_1 a_{12}}{a_{11} a_{22} - a_{12} a_{21}}$$

où

$$a_{11} = \left( \frac{1}{A_1} + \frac{1}{A_3} + \frac{1}{A_5} + \frac{2\sqrt{2}}{A_7} + \frac{2\sqrt{2}}{A_8} \right) \frac{L}{2E}$$

$$a_{12} = a_{21} = \frac{L}{2A_5 E}$$

$$a_{22} = \left( \frac{1}{A_2} + \frac{1}{A_4} + \frac{1}{A_5} + \frac{1}{A_6} + \frac{2\sqrt{2}}{A_9} + \frac{2\sqrt{2}}{A_{10}} \right) \frac{L}{2E}$$

$$b_1 = \left( \frac{P_2}{A_1} - \frac{P_1 + 2P_2 - P_3}{A_3} - \frac{P_2}{A_5} - \frac{2\sqrt{2}(P_1 + P_2)}{A_7} \right) \frac{\sqrt{2}L}{2E}$$

$$b_2 = \left( \frac{\sqrt{2}(P_3 + P_2)}{A_4} - \frac{\sqrt{2}P_2}{A_5} - \frac{4P_2}{A_9} \right) \frac{L}{2E}$$

Une fois les forces axiales dans les barreaux établies, il est possible de calculer la contrainte normale avec l'équation suivante:

$$\sigma_i = \frac{N_i}{A_i}$$

La masse de la structure se calcul ainsi :

$$M = \rho L \left( \sum_{j=1}^6 A_j + \sqrt{2} \sum_{j=7}^{10} A_j \right)$$

Enfin, le calcul du déplacement au nœud 2 se calcul selon l'équation suivante:

$$\delta_2 = \left[ \sum_{j=1}^6 N_j^0 N_j + \sqrt{2} \sum_{j=7}^{10} N_j^0 N_j \right] \frac{L}{E}$$

où les  $N_j^0$  sont calculés à partir des équations ci-dessus en substituant les forces appliquées P, par celle-ci  $P_1=P_3=0$  et  $P_2=1$ .



### ANNEXE 3 – Sélection de la zone d'intérêt

Le principe de modification des limites de la zone d'intérêt consiste à répartir tous les points d'expériences selon deux catégories : ceux qui sont conformes la contrainte et ceux qui ne sont pas conformes à la contrainte. Ensuite, en affichant les diagrammes à moustache de ces 2 catégories pour chaque variable X, il est possible d'observer si des régions de la zone d'intérêt peuvent être réduites.

Ainsi, pour le problème de structure à 10 membrures, une itération a été effectuée. Les figures suivantes présentent la sélection de la zone d'intérêt. Les points d'expériences conformes sont représentés par un « C » et ceux qui ne sont pas conformes par un « N ». De plus, le chiffre indique de quel barreau il s'agit. Par exemple, les valeurs du barreau #1, pour les points qui sont conformes, sont représentées par la variable « C1 ». Les valeurs des paramètres sont normalisées en suivant l'équation 3.22.

Enfin sur chaque figure, les parties de la zone d'intérêt à **rejeter** sont encadrées en rouge. Les valeurs normalisées et réelles de la nouvelle zone de conception sont présentées dans le Tableau A.2.

Tableau A.2 Définition de la zone d'intérêt

Paramètre	Borne inférieure (normalisée)	Borne supérieure (normalisée)	Borne inférieure (in.2)	Borne supérieure (in.2)
X <sub>1</sub>	0.40	1	6.8	17
X <sub>2</sub>	0.01	1	0.17	17
X <sub>3</sub>	0.40	1	6.8	17
X <sub>4</sub>	0.12	1	2.04	17
X <sub>5</sub>	0.01	1	0	17
X <sub>6</sub>	0.01	1	0	17
X <sub>7</sub>	0.25	1	4.25	17
X <sub>8</sub>	0.2	1	3.4	17
X <sub>9</sub>	0.1	1	1.7	17
X <sub>10</sub>	0.01	1	0	17

Note : Pour les contraintes 9 et 7, deux points marginaux ont été retirés puisqu'ils ne rencontraient pas la contrainte 1.

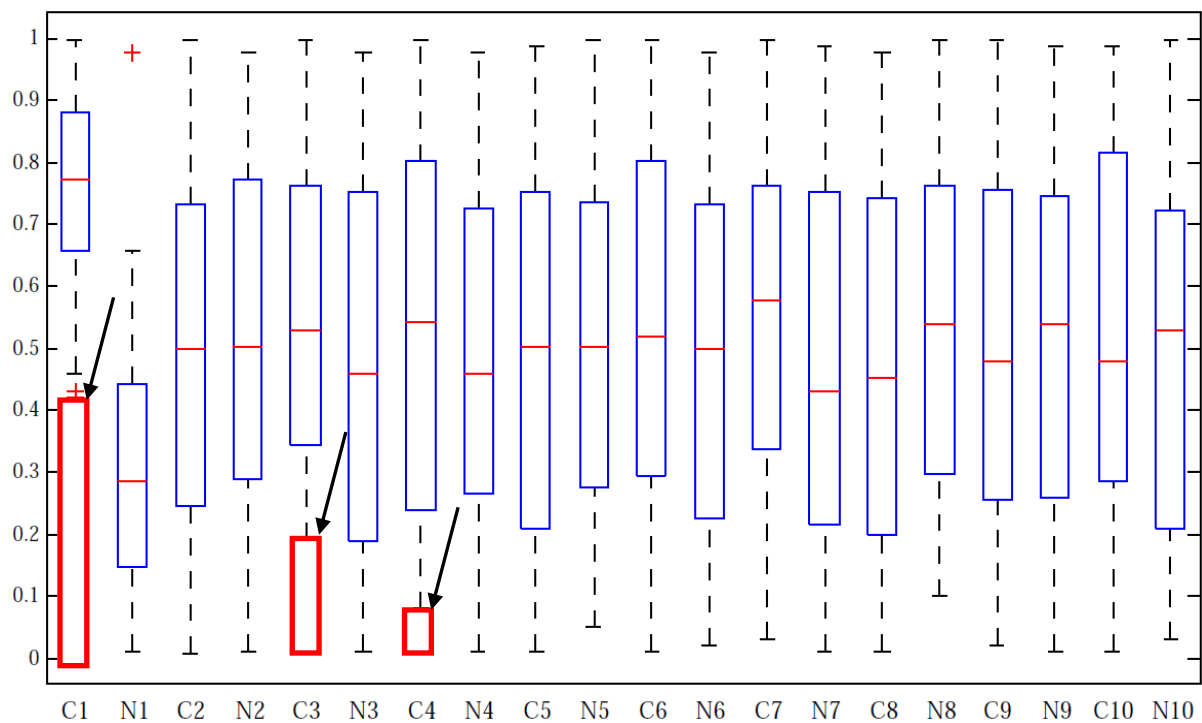


Figure A-2 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_1$

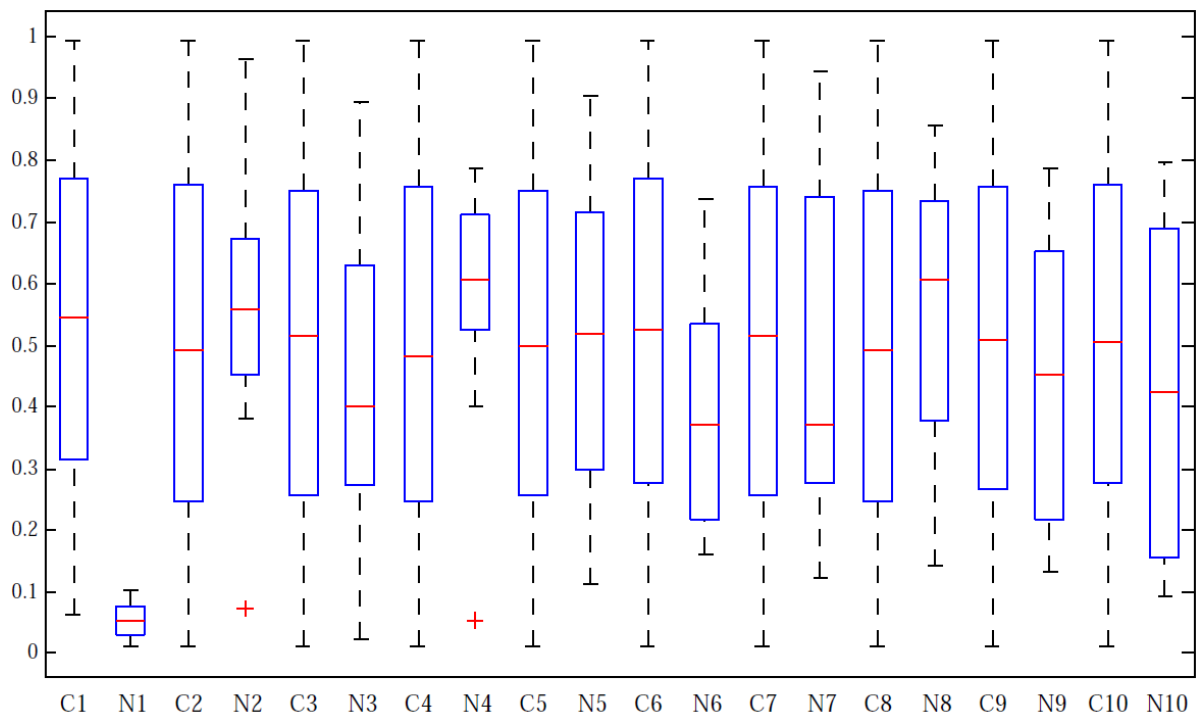


Figure A-3 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_2$

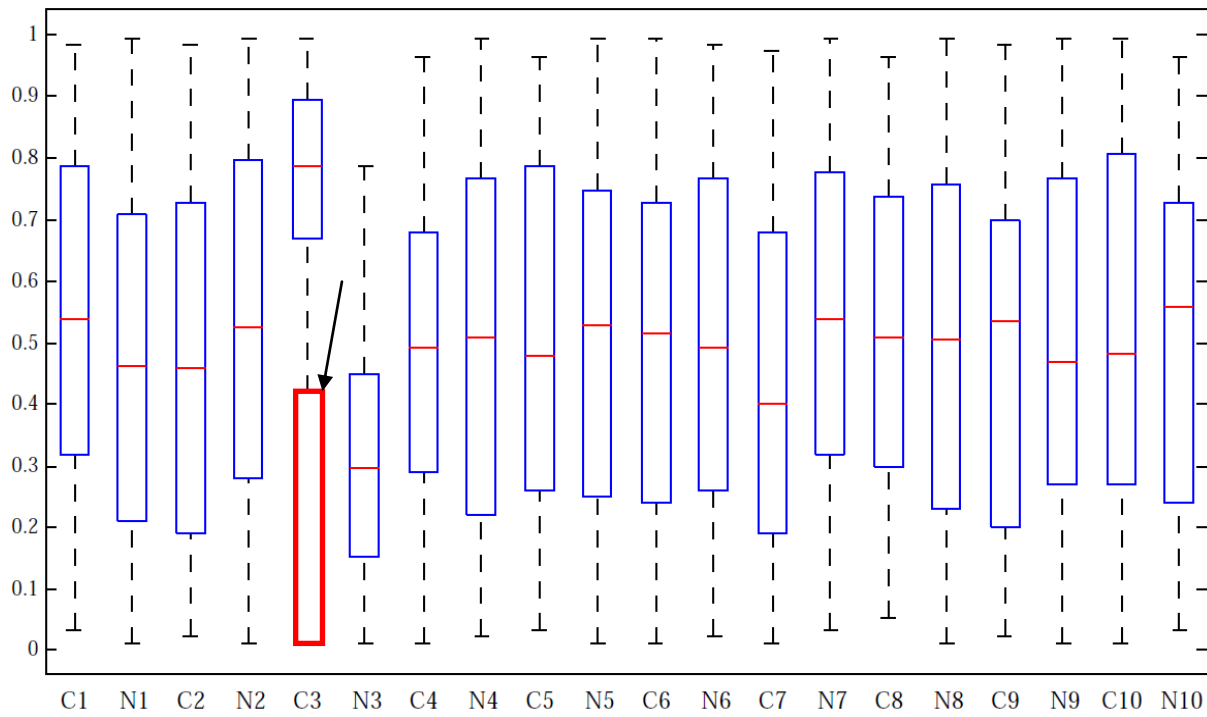


Figure A-4 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_3$

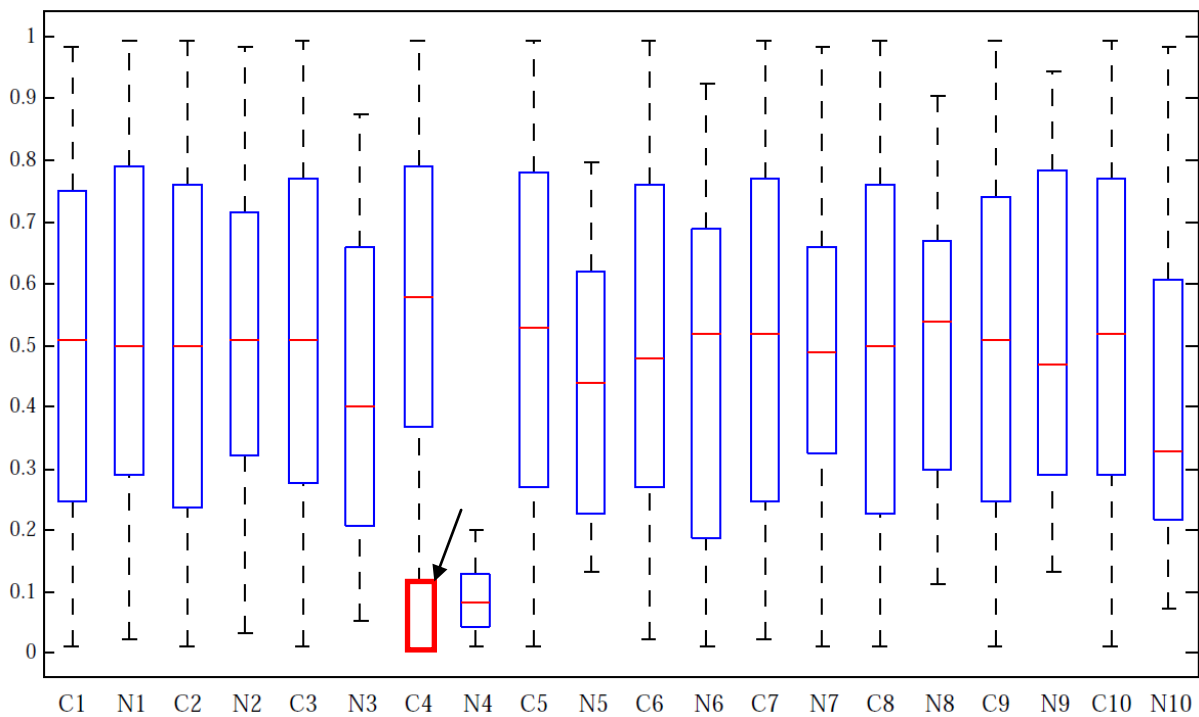


Figure A-5 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_4$

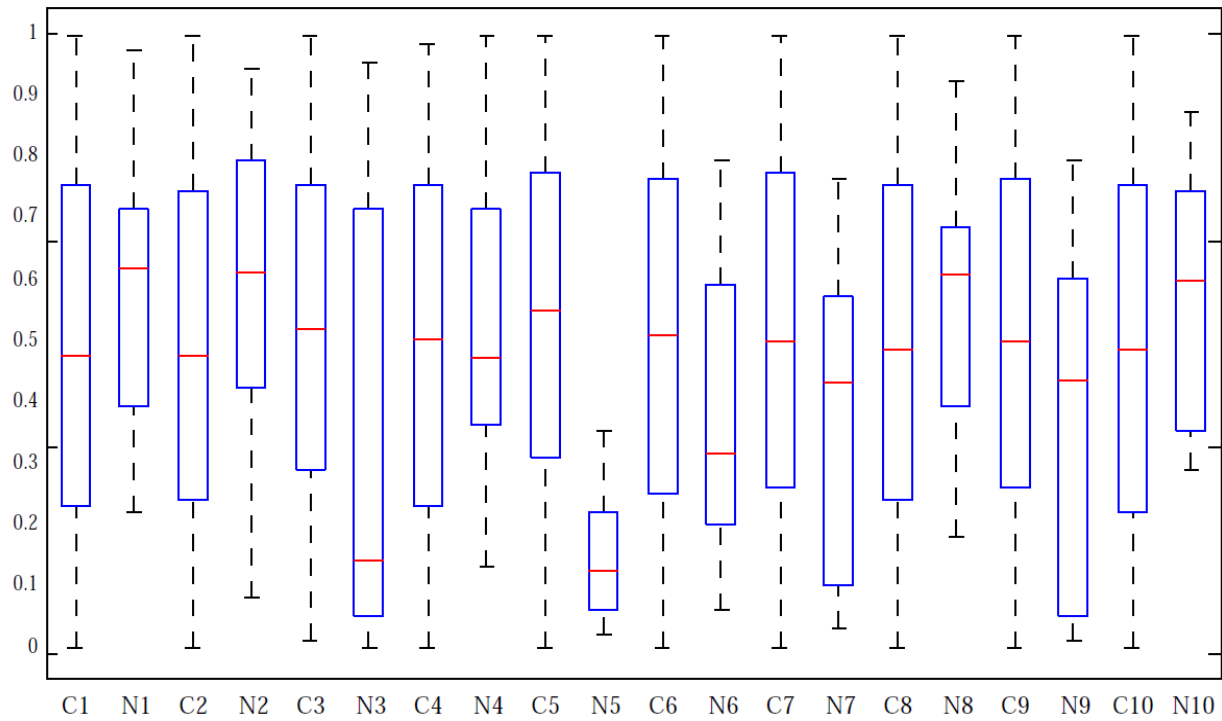


Figure A-6 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_5$

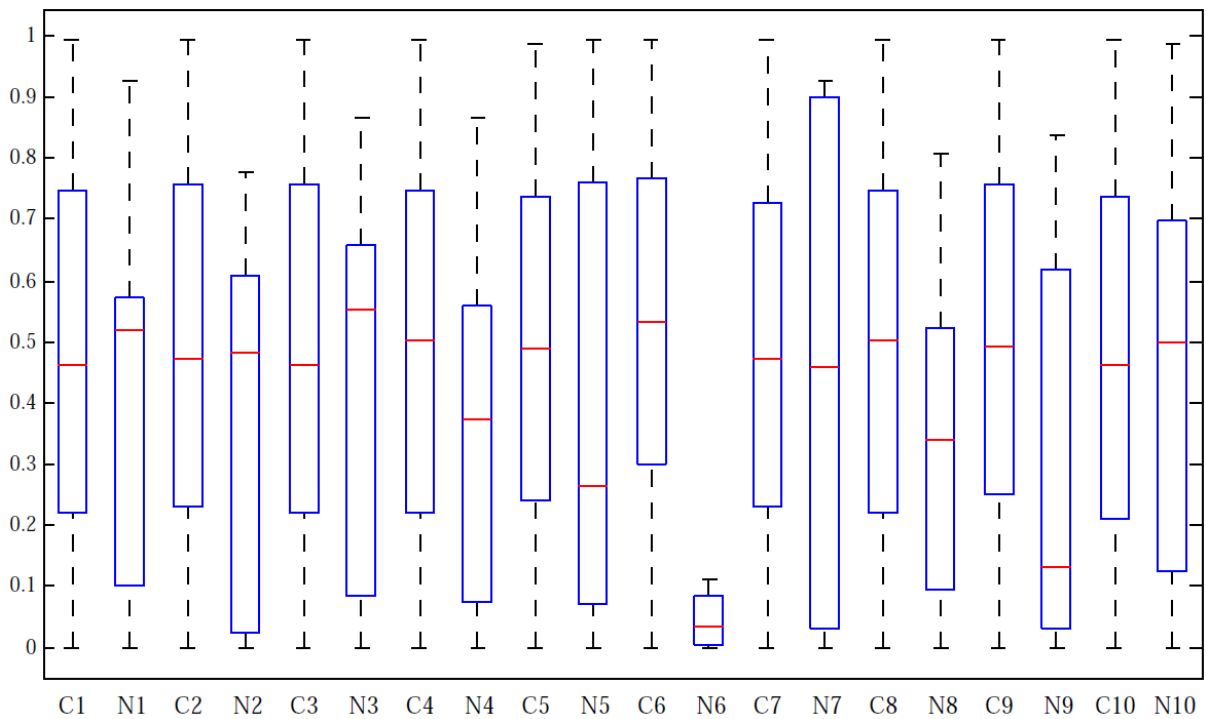


Figure A-7 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_6$

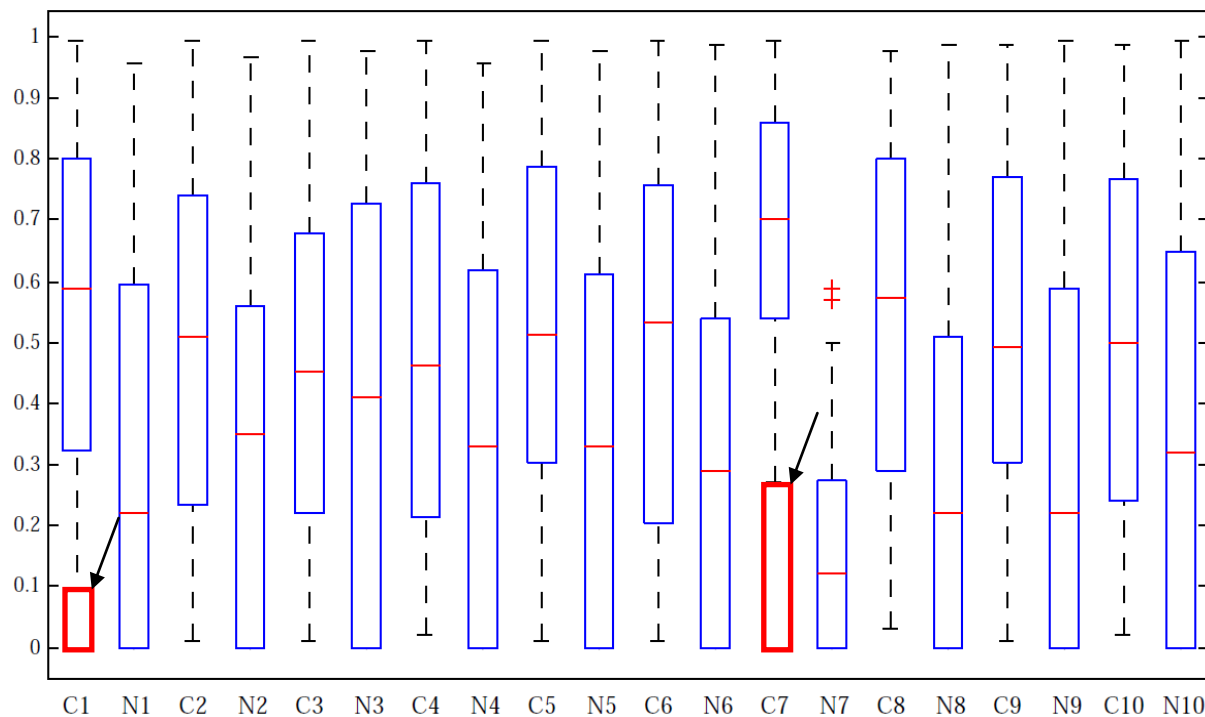


Figure A-8 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_7$

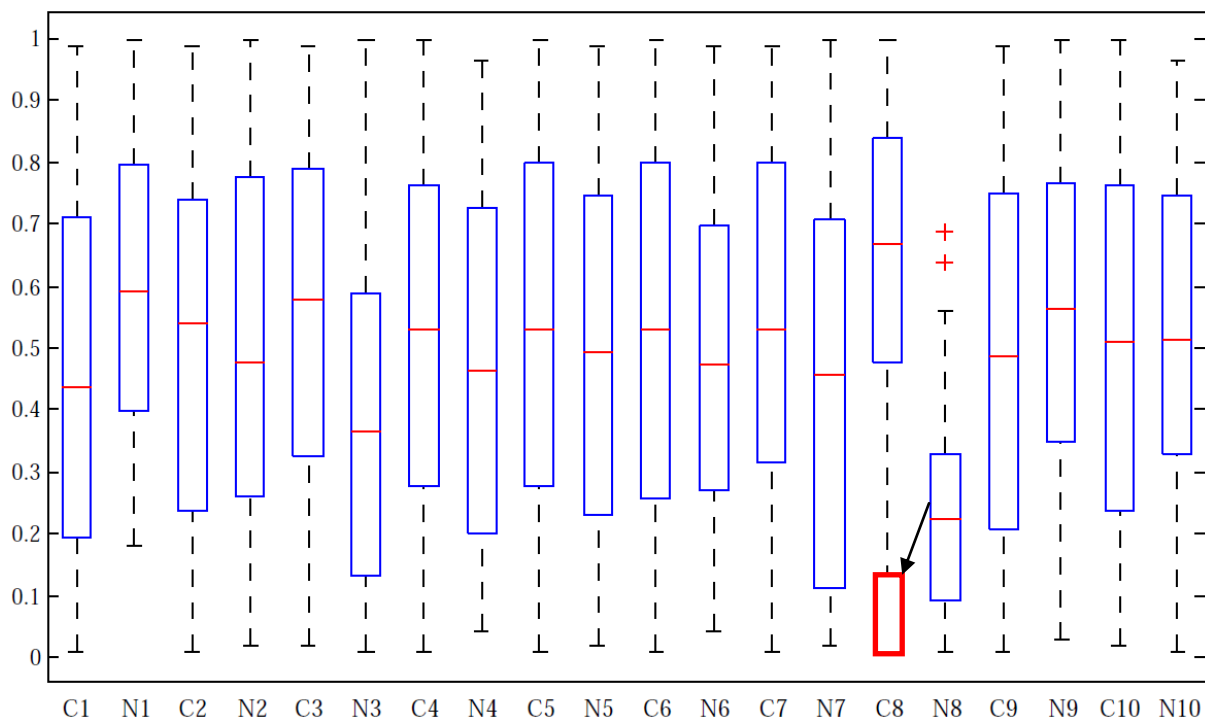


Figure A-9 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_8$

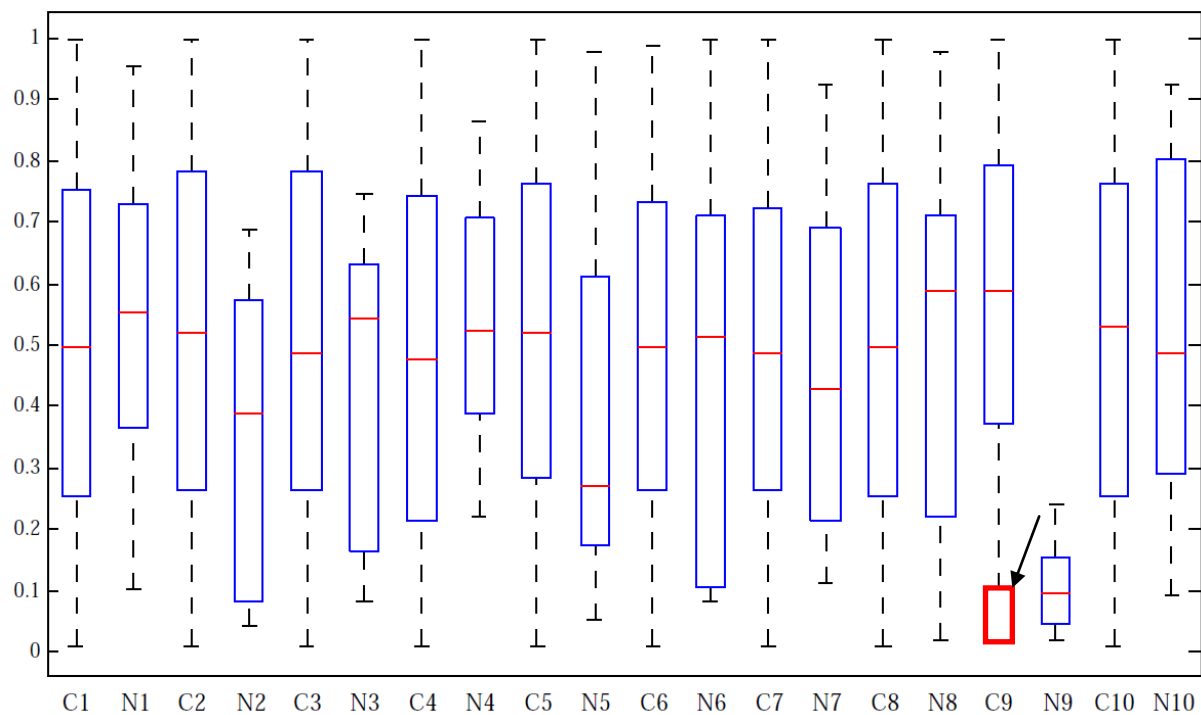


Figure A-10 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_9$

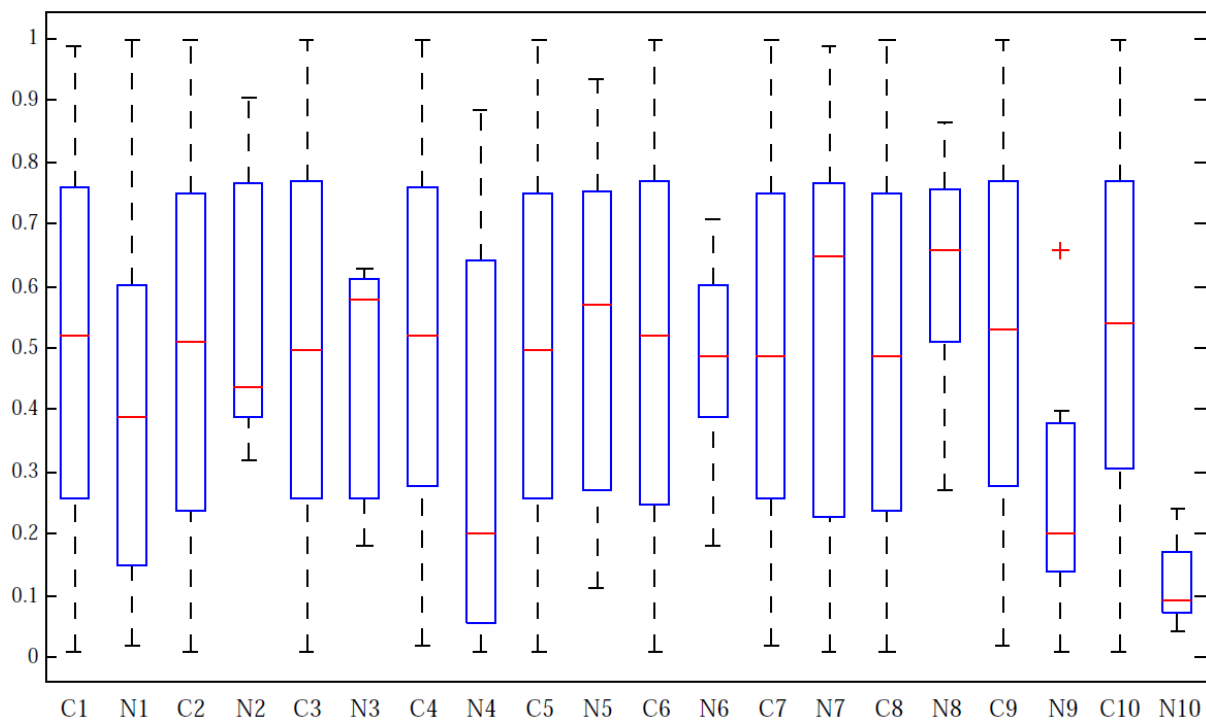


Figure A-11 Dispersion des valeurs des paramètres pour la contrainte  $\sigma_{10}$