

Chapitre 1

État de l’art et nouvelles tendances en conception collaborative

1.1. Introduction

Nous présentons dans ce chapitre les principales contributions en optimisation multidisciplinaire (ou MDO pour MultiDisciplinary Optimization). Le nom “optimisation multidisciplinaire” laisse sous-entendre à tort qu’il ne s’agit que d’un problème d’optimisation. Nous lui préférons ici le terme “conception collaborative”. En effet, l’optimisation ne représente qu’une composante qui ne peut être séparée du reste du problème de conception. D’autre part, le but n’est pas de créer un processus automatique, mais de faciliter les interactions entre les équipes des différentes disciplines.

Dans la littérature, la plupart des publications considèrent deux types de paramètres de conception : les paramètres publics (partagés par les disciplines) et les paramètres privés (spécifiques à une seule discipline). Nous considérons ici seulement les paramètres publics, les paramètres privés étant choisis à leur valeur optimale par chaque discipline. Si les variables privées ne sont pas traitées localement, elles sont traitées comme variables de conception globales et elles ne s’appellent plus variables privées.

La définition des paramètres publics et de leur rayon de validité est un enjeu majeur, trop souvent sous-estimé dans la littérature dédiée à la MDO. Ce choix est en

2 Conception collaborative

général dicté par le savoir-faire et l'expérience des ingénieurs. La configuration optimale obtenue dépend fortement de l'ensemble des paramètres publics sélectionnés. Par exemple, dans la conception d'une aile d'avion, la discipline "structure" a pour souhait d'augmenter l'épaisseur pour avoir une meilleure résistance, tandis que le but de la discipline "CFD" (Computational Fluid Dynamics ou dynamique des fluides numérique) est au contraire de réduire l'épaisseur de l'aile afin de diminuer la traînée. La paramétrisation est un moyen de trouver un compromis entre les différentes disciplines.

Par exemple, les paramètres de forme sont difficiles à manipuler. En effet, les ingénieurs utilisent leurs paramètres sur des maillages fixes. De plus chaque discipline possède son propre maillage : à l'intérieur de la structure pour la mécanique des structures, sur la surface pour l'acoustique, et en dehors pour la CFD. Il faut, pour pallier ce problème, créer des paramétrisations de forme indépendantes des disciplines.

Les logiciels actuels de CAO (ou Conception Assistée par Ordinateur) prennent en charge la gestion de paramètres géométriques et peuvent gérer différents types de contraintes sur ces paramètres (bornes, volume, encombrement, ...). Ces outils pourraient faciliter les interactions entre disciplines. Malheureusement, cela n'est actuellement pas utilisé par les ingénieurs, car les logiciels de CAO n'offrent pas la possibilité de paramétrer le maillage et il faut reconstruire un nouveau maillage pour chaque nouvelle configuration étudiée. Ce n'est donc pas compatible avec un processus de recherche de configuration optimale. L'erreur induite par un changement de topologie du maillage est trop importante, et risque de dégrader l'information à calculer (comme par exemple la variation des sorties entre deux configurations proches). Dans un contexte d'optimisation de forme, la paramétrisation de maillage (mesh morphing) est recommandée.

La conception multidisciplinaire est effectuée en pratique sur des plates-formes qui gèrent les interactions entre les logiciels d'analyse des disciplines et les algorithmes d'optimisation. Les logiciels commerciaux les plus connus sont iSight, modeFrontier, Model Center, et Optimus. Il y a aussi Boss Quatro, spécialisé en calcul des structures, et Dakota dans le domaine du logiciel libre [SAL 97, SAL 98, PAD 99, ALE 98]. Ces environnements prennent les logiciels d'analyse sous la forme de "boîtes noires". La communication par fichier est généralement prise en charge avec des langages de protocole d'échange tels que Python.

Le présent chapitre développe les points suivants. Dans la section 1.2, nous exposons le problème de définition des paramètres. Les approches les plus classiques d'optimisation multidisciplinaire sont décrites dans la section 1.3. La section 1.4 expose les résultats comparatifs de ces formulations sur un cas-test de conception d'avion. Les travaux de J. Périaux et al., concernant la théorie des jeux appliquée à la MDO, sont présentés dans la section 1.5. Dans la section 1.6, nous exposons les approches alternatives et les développements récents en MDO.

1.2. Définition des paramètres

1.2.1. Paramètres publics

Dans un contexte multidisciplinaire, le choix et la définition des paramètres publics donne lieu à des négociations entre les différentes disciplines. Ce choix est en lui-même une étape du processus de conception et agit fortement sur la configuration optimale. Le chapitre *Optimisation de forme paramétrique multiniveau* traite de la paramétrisation.

La définition de ces paramètres est une question majeure pour la conception collaborative. Malgré les avancées technologiques, ce problème reste difficile et ne peut pas être résolu par l'équipe de conception seule. Ce choix est le produit d'une discussion entre tous les différents départements acteurs dans la conception. En général, les premières simulations permettent de mesurer l'importance des différents paramètres et d'affiner ce choix.

1.2.2. Paramètres privés

Les paramètres privés sont ceux qui ne concernent directement qu'une seule discipline. Par exemple, pour une aile, le nombre de nervures, leur position et leurs dimensions ne concernent que la discipline "structure". Pour un jeu de paramètres publics donné, la discipline "structure" doit alors renvoyer des réponses optimisées par rapport à ses variables locales.

1.2.3. Définition des paramètres publics

Dans le cas de l'optimisation de forme, le nombre de paramètres peut être très grand. Par exemple chaque nœud du maillage peut contenir un paramètre. Dans ce cas, il est possible de trouver une solution optimale au problème d'optimisation, mais qui ne soit pas une configuration viable du point de vue de la conception. Les critères d'usinage ou d'esthétique sont difficiles à décrire et ne sont pas forcément pris en compte dans l'optimisation. En réduisant le nombre de paramètres de conception, on réduit les chances de tomber sur des solutions inutilisables.

Cependant, le fait de prendre en compte un grand nombre de paramètres peut être intéressant sous deux points de vue :

- nous pouvons tomber sur des configurations inattendues qui, à quelques modifications près pour satisfaire des critères subjectifs, peuvent se révéler très intéressantes ;
- le calcul des sensibilités (avec des méthodes adjointes), par rapport à chaque paramètre, permet de déterminer quels paramètres sont les plus significatifs et de créer un ensemble pertinent de paramètres publics.

4 Conception collaborative

Dans un contexte multidisciplinaire, nous ne pouvons considérer qu'un nombre réduit de paramètres.

1.2.3.1. *Art de l'ingénieur*

L'expérience des équipes d'ingénieurs sur le dimensionnement des paramètres de forme doit être prise en compte pour le choix des critères. Par exemple, en aérodynamique, le choix de la courbure du bord d'attaque de l'aile est le fruit d'une longue expérience.

1.2.3.2. *Méthodes adjointes*

La définition des paramètres peut se faire à l'aide des méthodes adjointes. Les objectifs de chaque discipline peuvent être combinés en un seul critère afin d'avoir un gradient commun. La distribution de ce gradient sur la surface de la configuration initiale indique les régions les plus importantes. Si l'on veut mesurer l'importance des paramètres sur chaque sortie, on peut aussi calculer les gradients de chaque critère séparément.

Une approche hiérarchique pour déterminer les paramètres les plus significatifs est présentée dans [TAN 02]. Une méthode basée sur les volumes de contrôle est présentée dans la partie 1.2.4.1. Pour la description des méthodes adjointes, nous renvoyons à [CEA 86, MOH 01, JAM 90]. Les méthodes de différentiation automatique peuvent être utilisées pour la génération du code adjoint [GRI 00, GRI 90, GRI 89, MOH 01, MOR 85, OST 71, ROS 92, SPE 80, STR 90, WEN 64]. Ces deux approches, les méthodes adjointes et l'art de l'ingénieur, sont complémentaires.

1.2.4. *Création des paramètres*

Depuis les années 90, les logiciels de CAO permettent la paramétrisation de forme sous contraintes. Ces contraintes peuvent exprimer une relation entre les paramètres (par exemple une surface ou un volume imposé) et elles peuvent être de nature géométrique : orthogonalité, parallélisme, etc.

La CAO, comme son nom l'indique, était liée à la conception. La CAO concerne désormais tous les cycles de vie d'un produit. Son rôle est étendu aux phases de fabrication, de maintenance, d'exploitation et même au recyclage en fin de vie du produit. C'est pour cette raison que l'on parle de PLM (Product Life Management). Toutes ces études sont faites avant le lancement de la fabrication, ce qui permet de revoir la conception de l'avion, si le remplacement d'un filtre pose un problème. Avant, il fallait improviser une solution pour le filtre pendant la phase de fabrication. Le calcul suit la même évolution et permet à la PLM de disposer d'informations quantitatives.

Revenons à la CAO, la géométrie paramétrée n'est toujours pas utilisée, malgré une forte incitation de la part des entreprises à développer la paramétrisation de forme. Les raisons principales sont :

– la difficulté de leur mise en œuvre : le dessin d'une géométrie fixe peut déjà être difficile à mettre en œuvre, et il est encore plus difficile de dessiner une infinité de géométries simultanément ;

– à l'intérieur d'une entreprise, la communication technique repose souvent sur la CAO. Cependant, contrairement à la CAO fixe (non paramétrée), il n'existe pas de format standard permettant de communiquer une géométrie paramétrée aux outils métiers de chaque discipline.

1.2.4.1. Volumes de contrôle

Nous proposons ici des méthodes correspondant aux besoins des ingénieurs. Nous considérons des maillages fixes. Cette méthode est basée sur des transformations de volume. Cela consiste à associer un volume paramétrique au maillage. Les nœuds du maillage suivent alors les modifications des points de contrôle du volume.

L'utilisation des NURBS à 2 variables (Non Uniform Rational B-Splines, ou B-Splines rationnelles non uniformes) est classique pour la représentation des surfaces. On peut alors imaginer des NURBS dépendant de trois variables (u, v, w) pour la définition de volumes. La perturbation d'un des points de contrôle des NURBS entraînera automatiquement la perturbation du maillage. On peut utiliser des éléments finis de type hexaèdre comme des NURBS de degré 1 ou 2. Ils sont généralement implémentés dans la plupart des logiciels de CAO.

L'avantage de cette méthode est de créer des perturbations de maillage régulières à l'intérieur de chaque volume.

1.2.4.1.1. Perturbation de maillage

L'utilisation de NURBS ou d'éléments finis nécessite une fonction explicite qui à chaque point (u, v, w) du cube de référence $((u, v, w) \in [0, 1]^3)$ associe un point (x, y, z) du volume de contrôle (voir Fig. 1.1).

$$\begin{pmatrix} x(u, v, w) \\ y(u, v, w) \\ z(u, v, w) \end{pmatrix} = \sum P_i(u, v, w) \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}, \quad (1.1)$$

où (x_i, y_i, z_i) sont les points de contrôle du volume de contrôle.

Voici comment construire une perturbation de maillage. Soit (x_i^0, y_i^0, z_i^0) et (x_i^1, y_i^1, z_i^1) les coordonnées des points de contrôle respectivement avant et après la perturbation. Soit (x^0, y^0, z^0) un nœud du maillage avant perturbation. Nous calculons d'abord le point (u^0, v^0, w^0) tel que :

$$\begin{pmatrix} x^0 \\ y^0 \\ z^0 \end{pmatrix} = \sum P_i(u^0, v^0, w^0) \begin{pmatrix} x_i^0 \\ y_i^0 \\ z_i^0 \end{pmatrix}. \quad (1.2)$$

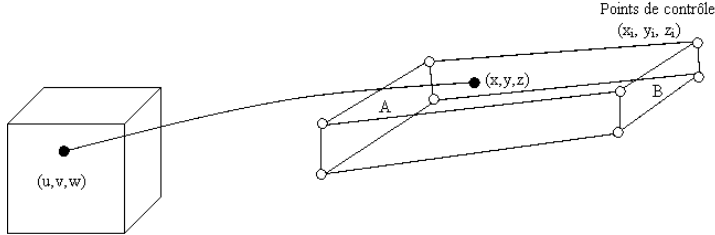


Figure 1.1. Le cube de référence (gauche) ; l'élément hexaédrique avec ses 8 points de contrôle (droite).

Les coordonnées de ce point après la perturbation sont donnés par :

$$\begin{pmatrix} x^1 \\ y^1 \\ z^1 \end{pmatrix} = \sum P_i(u^0, v^0, w^0) \begin{pmatrix} x_i^1 \\ y_i^1 \\ z_i^1 \end{pmatrix}. \quad (1.3)$$

Nous résolvons le système (1.2) à trois équations afin de trouver les trois inconnues à chaque nœud du maillage initial. Ce calcul est effectué une seule fois et sert pour toutes les perturbations.

1.2.4.1.2. Perturbation du maillage extérieur

En acoustique ou en électromagnétisme, le maillage s'applique à la surface de la structure. Dans ce cas, le processus décrit dans le paragraphe précédent permet de perturber les nœuds de la surface qui sont par définition à l'intérieur du volume de contrôle.

En aérodynamique, les phénomènes étudiés se déroulent en général hors de la structure et le maillage est alors en majeure partie en dehors du volume de contrôle.

Nous proposons d'utiliser les techniques de volumes de contrôle pour le calcul des perturbations de maillage, et les techniques traditionnelles pour le calcul des perturbations de volume. Nous allons maintenant voir qu'il y a plusieurs méthodes de propagation de maillage.

1.2.5. Propagation de maillage

Dans le cas d'un maillage externe, si le paramètre est défini par une perturbation de surface, on doit propager cette perturbation de surface dans le maillage avoisinant.

1.2.5.1. Méthode d'élasticité

La perturbation est considérée comme un déplacement imposé. À l'intérieur du domaine on résout un problème d'élasticité linéaire. En général, la rigidité est une fonction croissante du volume de l'élément. Nous utilisons le nouveau maillage obtenu pour les calculs à venir.

1.2.5.2. Méthode intégrale

Soit $F(x)$ l'image d'un point x . La fonction $F(x)$ est supposée connue sur tous les points de la frontière, elle est donnée par la méthode des volumes de contrôle. Le déplacement d'un point interne est alors pondéré par l'inverse de sa distance à la surface :

$$F(x) = \left(\int_{\partial\Omega_{r_0}} \frac{F(x')}{\|x - x'\|} dx' \right) / \left(\int_{\partial\Omega_{r_0}} \frac{1}{\|x - x'\|} dx' \right). \quad (1.4)$$

Cette intégrale peut être calculée en utilisant le maillage à perturber.

1.2.6. Application à une aile

Considérons une aile, dont les paramètres sont l'épaisseur, la vrille, la flèche (voir Fig. 1.2), la courbure et l'inflexion (voir Fig. 1.6 et 1.7). Nous allons illustrer les techniques de volumes de contrôle avec cet exemple.

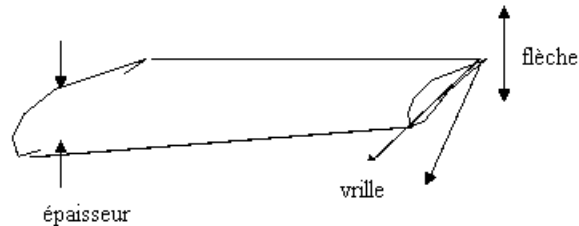


Figure 1.2. Paramètres de l'aile.

Une modification de l'épaisseur consiste en une perturbation verticale des points de contrôle de la face A (voir Fig. 1.1 et 1.3). La vrille consiste en une rotation rigide de la face B (voir Fig. 1.1 et 1.4). La modification de la flèche consiste en une translation de la face B (voir Fig. 1.1 et 1.5).

Nous pouvons utiliser un élément fini de degré 2 pour modifier la courbure du profil ou l'inflexion. Nous voyons sur la figure 1.6 un élément fini d'ordre 2 dans la direction correspondant à la courbure, et sur la figure 1.7 un autre élément fini d'ordre 2 dans la direction correspondant à l'inflexion.

La figure 1.8 montre le déplacement du moteur le long de l'aile. Les blocs $B1$ et $B3$ suivent le déplacement par le contrôle des points du bloc $B2$. Si le bord d'attaque et le bord de fuite sont parallèles, on peut considérer seulement les déplacements rigides de ce bloc. Si nous voulons prendre en compte l'angle existant entre ces deux bords, nous devons forcer les points de contrôle du bloc $B2$ à rester sur le bord.

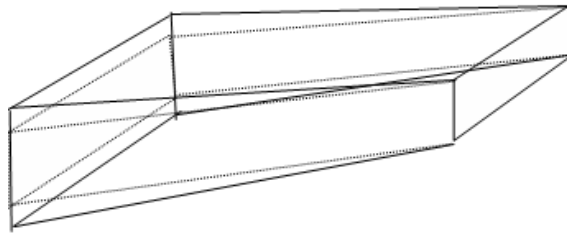


Figure 1.3. *Modification de l'épaisseur avec un élément fini de degré 1 dans chaque direction.*

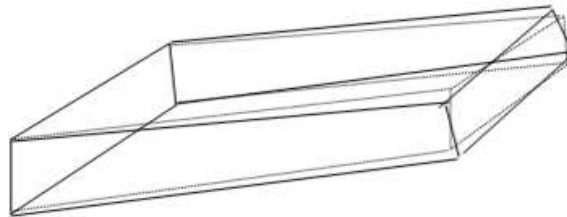


Figure 1.4. *Modification de la vrille avec un élément fini de degré 1 dans chaque direction.*

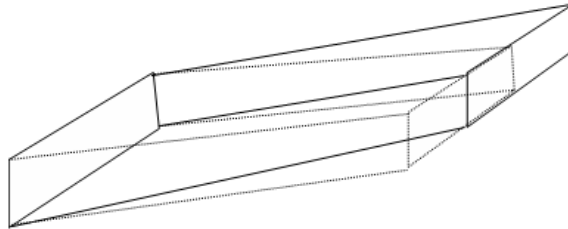


Figure 1.5. *Modification de la flèche avec un élément fini de degré 1 dans chaque direction.*

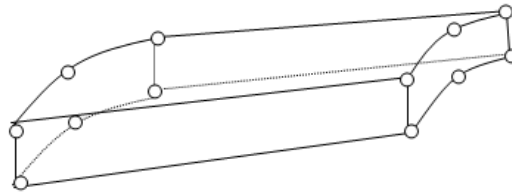


Figure 1.6. *Modification de la courbure avec un élément de degré 2 dans une direction et de degré 1 dans les deux autres directions.*

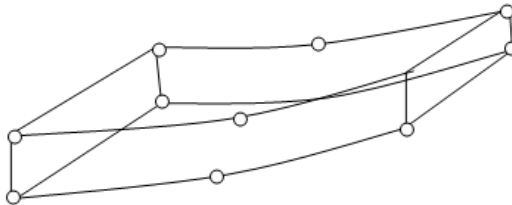


Figure 1.7. *Modification de l'inflexion avec un élément de degré 2 dans une direction et de degré 1 dans les deux autres directions.*

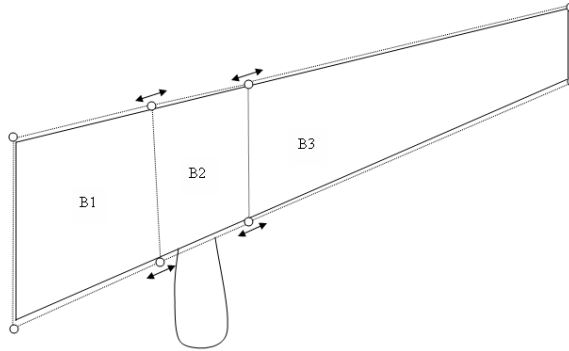


Figure 1.8. Déplacement du moteur. Les flèches correspondent aux degrés de liberté des différents points, les autres points étant fixés.

1.3. Formulations pour l'optimisation multidisciplinaire

Par souci de clarté, nous nous limitons ici à deux disciplines, et nous supposons que les paramètres privés sont cachés. Les équations d'état sont données sous la forme explicite suivante :

$$a_1 = A_1(x, a_2), \quad (1.5)$$

$$a_2 = A_2(x, a_1), \quad (1.6)$$

où x représente les paramètres publics de conception, et a_i sont les sorties de la discipline i . Dans la première (resp. deuxième) équation, a_2 (resp. a_1) correspond à la contribution de la discipline 2 (resp. 1) pour le calcul de a_1 (resp. a_2). Résoudre ce système d'équations d'état couplées se dit effectuer une analyse multidisciplinaire (ou MDA pour MultiDisciplinary Analysis).

Le problème d'optimisation à résoudre est le suivant :

$$\begin{aligned} \min_x \quad & f(x, a_1(x), a_2(x)). \\ g(x, a_1(x), a_2(x)) \leq & 0 \end{aligned} \quad (1.7)$$

Nous ne considérons pas ici les contraintes disciplinaires internes, nous supposons qu'elles sont prises en compte implicitement dans les équations d'état.

1.3.1. *MDF : Multi-Disciplinary Feasible*

À chaque itération de l'algorithme d'optimisation, les équations d'état sont résolues afin de déterminer les sorties [ALE 99]. C'est la méthode la plus naturelle dans laquelle chaque discipline fournit son code d'analyse et, lorsque cela est possible, le calcul des gradients.

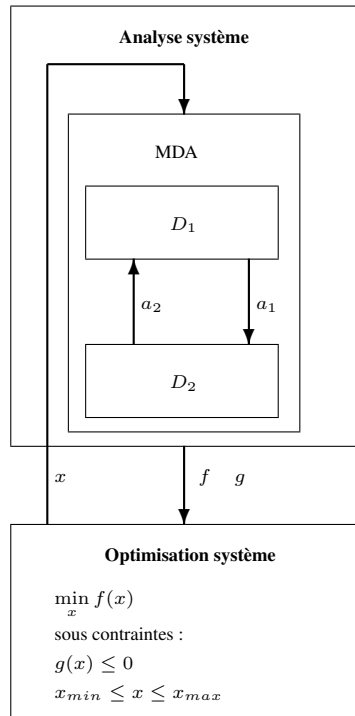


Figure 1.9. Diagramme de la formulation MDF

1.3.2. *IDF : Individual Discipline Feasible*

Dans cette méthode, chaque discipline résout sa propre équation d'état. Les variables d'interaction deviennent des paramètres supplémentaires pour le problème d'optimisation. L'optimiseur, en plus de résoudre le problème d'optimisation, doit aussi assurer la cohérence interdisciplinaire.

Le problème d'optimisation est alors le suivant :

$$\begin{aligned}
 & \min_{x, b_1, b_2} f(x, b_1, b_2). \\
 & g(x, b_1, b_2) \leq 0 \\
 & b_1 = a_1(x, b_2) \\
 & b_2 = a_2(x, b_1)
 \end{aligned} \tag{1.8}$$

$a_1(x, b_2)$ et $a_2(x, b_1)$ sont les solutions respectives des équations d'état 1.5 et 1.6.

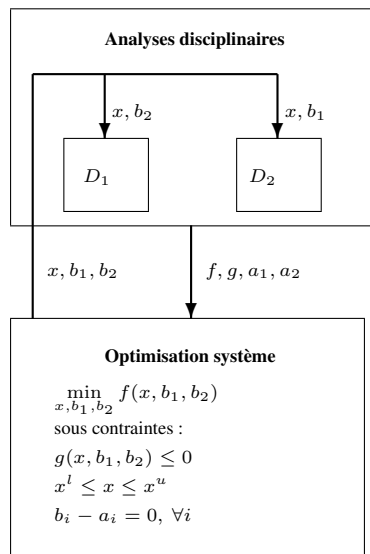


Figure 1.10. Diagramme de la formulation IDF

1.3.3. AAO : All-At-Once

La méthode AAO est aussi appelée "one shot", SAND ou SAD (Simultaneous Analysis and Design) [BRA 96, ALE 00b, GUM 01]. On résout simultanément les équations d'état de chaque discipline, la cohérence interdisciplinaire et le problème d'optimisation. Contrairement à la formulation MDF, les équations d'état ne sont pas forcément satisfaites à chaque pas de l'optimisation, mais elles doivent l'être à la convergence de l'algorithme. Les équations d'état sont considérées comme des contraintes d'égalité.

Le problème d'optimisation est le suivant :

$$\begin{aligned}
 & \min_{x, b_1, b_2} f(x, b_1, b_2). \\
 & g(x, b_1, b_2) \leq 0 \\
 & b_1 = t_1 \\
 & b_2 = t_2 \\
 & t_1 = A_1(x, b_2) \\
 & t_2 = A_2(x, b_1)
 \end{aligned} \tag{1.9}$$

Cette méthode peut être très efficace, mais sa convergence est compromise lorsque certaines des équations d'état sont fortement non-linéaires. Même dans un contexte monodisciplinaire, la convergence n'est pas toujours acquise.

Dans la formulation IDF, l'équation d'état de chaque discipline est résolue par la discipline elle-même, tandis que dans la formulation AAO, les équations d'état sont toutes résolues par l'optimiseur.

1.3.4. BLISS : Bi-Level Integrated System Synthesis

Cette méthode est décrite dans [SOB 98] À chaque étape, on résout les équations d'état (1.5) et (1.6), puis on calcule les gradients correspondants avec une méthode adjointe. On utilise une approximation de la fonction coût globale et des contraintes. La fonction coût est enrichie de l'information apportée par les coefficients de Lagrange λ provenant des optimisations locales. On utilise aussi une région de confiance pour s'assurer de la fiabilité de ces approximations. Le problème au point x_{cur} est le suivant :

$$\begin{aligned}
 \partial_x f_0 &= \partial_x f(x_{cur}, a_1(x_{cur}), a_2(x_{cur})) \\
 \partial_x g_0 &= \partial_x g(x_{cur}, a_1(x_{cur}), a_2(x_{cur})) \\
 g_0 &= g(x_{cur}, a_1(x_{cur}), a_2(x_{cur}))
 \end{aligned} \tag{1.10}$$

$$\begin{aligned}
 & \min_x (\partial_x f_0 + \lambda \cdot \partial_x g_0)(x - x_{cur}). \\
 & g_0 + \partial_x g_0(x - x_{cur}) \leq 0 \\
 & (1 - \Delta)x_{cur} \leq x \leq (1 + \Delta)x_{cur}
 \end{aligned}$$

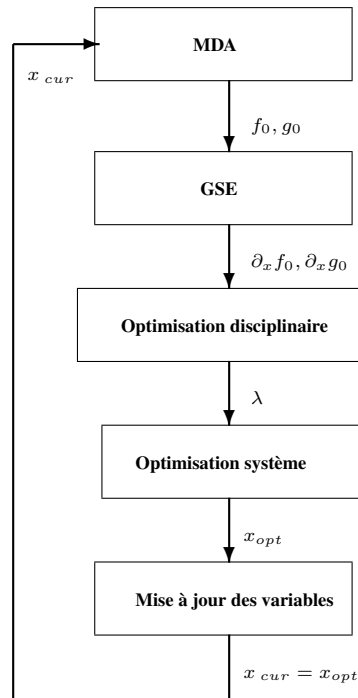


Figure 1.11. Diagramme de la formulation BLISS

La formulation BLISS-RSM [ALT 02] utilise des approximations polynomiales comme méta-modèles disciplinaires. La formulation BLISS-2000 [AGT 00] généralise les méta-modèles disciplinaires.

1.3.5. DAO : *Disciplinary Analysis Optimization*

Dans la méthode DAO [ALE 98, ALE 00b], l'optimisation n'est pas effectuée par les disciplines. Cependant, chaque discipline peut changer ses paramètres pour obtenir un point admissible. Nous avons vu précédemment que les contraintes disciplinaires faisaient partie de l'équation d'état. Les variables privées sont généralement utilisées pour satisfaire ces contraintes. Cependant, il peut exister un vecteur x , représentant les paramètres publics, pour lequel ces contraintes ne seront jamais satisfaites. On cherche alors à relaxer x en plusieurs jeux de variables partagées : z_i . Chaque discipline choisit ses variables z_i afin de satisfaire ses contraintes privées. Les différents z_i convergent à l'optimum grâce à l'ajout de contraintes d'égalité dans le problème d'optimisation global :

$$\begin{aligned}
 & \min_{x, b_1, b_2} f_{DAO}(x, b_1, b_2). \\
 & g(z_0, b_1, b_2) \leq 0 \\
 & b_1 = a_1(z_1, b_2) \\
 & b_2 = a_2(z_2, b_1) \\
 & z_0 = x \\
 & z_1 = x \\
 & z_2 = x
 \end{aligned} \tag{1.11}$$

1.3.6. CO : Collaborative Optimization

Dans la méthode CO [BRA 96, ALE 98, ALE 00a, ALE 00b, ALE 00c], toutes les disciplines contribuent à l'optimisation. Nous utilisons les même notations que pour la méthode DAO. La méthode "Collaborative Optimization" est une approche bi-niveaux : le niveau système et le niveau disciplinaire.

Le problème d'optimisation au niveau système est le suivant :

$$\left\{ \begin{array}{l} \min_{x, b_1, b_2} f(x, b_1, b_2). \\ [\|x - x_1\|^2 + \|x - x_2\|^2 + \|a_1 - b_1\|^2 + \|a_2 - b_2\|^2] = 0 \\ g(z_0, b_1, b_2) \leq 0 \end{array} \right. \tag{1.12}$$

Le niveau système impose à chaque discipline des objectifs à atteindre au sens des moindres carrés. L'objectif à atteindre pour la discipline i est b_i et la solution x_i doit être proche de x . Le problème disciplinaire pour la discipline 1 est alors :

$$\min_{x_1} \frac{1}{2} [\|x_1 - x\|^2 + \|b_1 - a_1(x_1, b_2)\|^2], \tag{1.13}$$

où a_1 est la solution de l'équation d'état (1.5).

Le problème disciplinaire pour la discipline 2 est alors :

$$\min_{x_2} \frac{1}{2} [\|x_2 - x\|^2 + \|b_2 - a_2(x_2, b_1)\|^2], \tag{1.14}$$

où a_2 est la solution de l'équation d'état (1.6).

1.3.7. CSSO : Collaborative Sub-Space Optimization

Nous donnons ici une description rapide de la méthode CSSO (pour plus de détails, nous renvoyons à [SHA 93]). Les paramètres de conception sont distribués entre les différentes disciplines. Chaque discipline optimise certains paramètres.

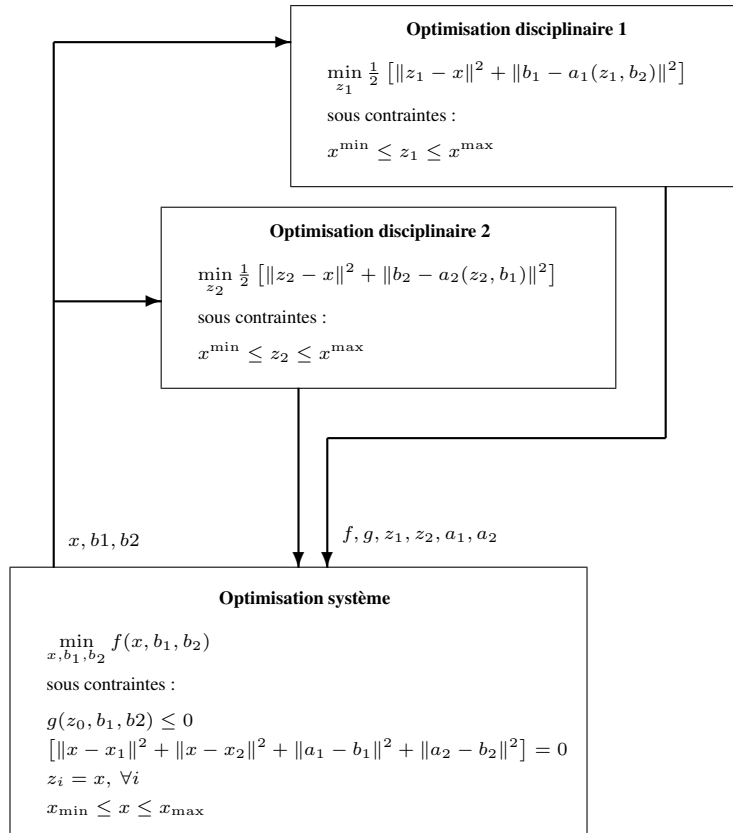


Figure 1.12. Diagramme de la formulation CO

La méthode CSSO est similaire à la méthode de décomposition de domaine et son application à la résolution des équations d'état. Elle diffère toutefois de cette dernière par un aspect : avec la méthode de décomposition de domaine, une inconnue peut être facilement associée à une équation d'état. En optimisation multidisciplinaire, il paraît difficile d'associer un paramètre public à une seule discipline car ils sont partagés entre les disciplines par définition.

Nous allons utiliser des méta-modèles de la discipline 2 (notées \tilde{A}_2) dans l'équation correspondant à la discipline 1, et réciproquement nous utilisons des méta-modèles de la discipline 1 (notées \tilde{A}_1) dans l'équation correspondant à la discipline 2.

Rappelons le problème initial :

$$a_1 = A_1(x, a_2), \quad (1.15)$$

$$a_2 = A_2(x, a_1), \quad (1.16)$$

$$\min_x f(x, a_1(x), a_2(x)).$$

$$g(x, a_1(x), a_2(x)) \leq 0 \quad (1.17)$$

Nous notons $x = (x_1, x_2)$, où x_i est le vecteur des variables associé à la discipline i et \tilde{a}_i les sorties du méta-modèle de la discipline a_i . Plus de détails sur les méta-modèles seront donnés par la suite.

Le problème pour la discipline 1 est le suivant :

$$a_1 = A_1(x, \tilde{a}_2), \quad (1.18)$$

$$\tilde{a}_2 = \tilde{A}_2(x, a_1), \quad (1.19)$$

$$\min_{x_1} f(x_1, x_2, a_1(x), \tilde{a}_2(x)). \quad (1.20)$$

$$g(x, a_1(x), \tilde{a}_2(x)) \leq 0$$

$$(1 - \Delta_2)x_{1\text{cur}} \leq x_1 \leq (1 + \Delta_2)x_{1\text{cur}} \quad (1.21)$$

La dernière contrainte introduite traduit la région de confiance du méta-modèle \tilde{a}_2 autour du point courant $x_{1\text{cur}}$. \tilde{a}_2 et Δ_2 sont fournis par la discipline 2.

De la même manière, nous aurons pour la discipline 2 :

$$\tilde{a}_1 = \tilde{A}_1(x, a_2), \quad (1.22)$$

$$a_2 = A_2(x, \tilde{a}_1), \quad (1.23)$$

$$\min_{x_2} f(x_1, x_2, \tilde{a}_1(x), a_2(x)). \quad (1.24)$$

$$g(x, \tilde{a}_1(x), a_2(x)) \leq 0$$

$$(1 - \Delta_1)x_{2\text{cur}} \leq x_2 \leq (1 + \Delta_1)x_{2\text{cur}} \quad (1.25)$$

Plusieurs publications traitent des méthodes de paramétrisation avec le but d'accroître l'autonomie de chaque discipline et d'agrandir la région de confiance [GUI 92,

GUI 93, GUI 94, GUI 95, MAD 00, LAW 96, LEG 04, ROD 00, SAN 00, SER 98, SOB 00].

Dans la plupart des méthodes de décomposition de domaine [ACH 95], il y a une façon naturelle d'associer inconnues et équations qui sont en même quantité. Cette association rend la résolution plus simple. À l'inverse, en conception collaborative, ce n'est pas trivial d'associer un sous-ensemble de paramètres à une discipline en particulier. Le chapitre *Partage de territoire en ingénierie concourante* propose des solutions de partage des variables.

1.3.8. DIVE : Discipline Interaction Variable Elimination

Cette formulation est décrite dans [MAS 06]. Elle s'inspire de la méthode BLISS 2000 [AGT 00]. Cette méthode cherche à réduire la complexité du problème d'optimisation par une élimination progressive des différents types de variables.

Les disciplines utilisent des méta-modèles (voir la section 1.3.9 pour plus de détails) pour leurs sorties. Pour une configuration de variable publique x , les sorties fournies par le méta-modèle sont optimales par rapport aux variables privées. Ces méta-modèles disciplinaires renvoient, en plus des sorties \tilde{A} , un indice α_i de fiabilité de la réponse fournie par le méta-modèle. Lorsque $\alpha_i < 0$, le méta-modèle est valide en ce point. Sinon, il faut le réévaluer, afin d'avoir une évaluation disciplinaire correcte. Chaque discipline est responsable de la construction de son méta-modèle et choisit celui qui lui convient le mieux. Le méta-modèle peut être construit en utilisant les techniques de réseaux neuronaux, de krigeage, etc. Lorsque le méta-modèle est une approximation linéaire ou quadratique, la contrainte de fiabilité $\alpha_i < 0$ définit ce que l'on appelle la région de confiance.

Les variables de couplage a_1 et a_2 doivent minimiser les résidus de l'équation d'état au sens des moindres carrés :

$$\min_{a_1, a_2} \|\tilde{A}_1(x, a_2) - a_1\|^2 + \|\tilde{A}_2(x, a_1) - a_2\|^2. \quad (1.26)$$

Cette approche spécifique à DIVE pour résoudre les équations d'état du système diffère des autres formulations où l'on trouve le plus souvent l'utilisation d'une méthode de point fixe (MDF, BLISS) ou bien l'ajout de contraintes d'égalité (AAO, IDF, CO). Cette caractéristique fait de DIVE une méthode plus robuste car elle pose mieux le problème de résolution de l'équation d'état et permet de le traiter de façon plus efficace avec des techniques de minimisation classiques, par exemple la méthode de Levenberg-Marquardt ou des méthodes adjointes.

Les variables partagées sont traitées au niveau de l'optimiseur système. Elles sont choisies pour satisfaire le problème d'optimisation global :

$$\begin{aligned} \min_x f(x, a_1(x), a_2(x)). \\ g(x, a_1(x), a_2(x)) \leq 0 \\ \alpha_i \leq 0 \end{aligned} \quad (1.27)$$

La solution obtenue doit satisfaire plusieurs conditions pour être considérée comme solution du problème de conception :

- la fiabilité de chaque méta-modèle doit être au moins vérifiée, sinon maximale, au point solution ;
- si l'optimum est situé sur le bord du domaine de validité du méta-modèle (c'est-à-dire lorsque $\alpha_i = 0$), il faut réévaluer ce méta-modèle autour de l'optimum et relancer l'optimisation ;
- lorsque les équations d'état ne sont pas résolues de manière assez précise, il faut aussi réévaluer les méta-modèles.

La configuration va ainsi évoluer dans l'espace de conception jusqu'à l'obtention d'une solution globale.

Les particularités de cette formulation sont les suivantes :

- une importance particulière est donnée à la résolution précise des équations d'état en formulant cette étape comme un problème d'optimisation et en vérifiant qu'à l'optimum ces équations sont bien résolues de manière précise ;
- l'évaluation de la fiabilité des méta-modèles disciplinaires fait de DIVE une généralisation de la méthode des régions de confiance [CON 98] ;
- dans certaines circonstances, la formulation DIVE est similaire à d'autres formulations :
 - si l'on utilise pas de méta-modèles et que l'on considère toutes les variables au niveau global, cela devient la formulation MDF ;
 - l'évaluation de la qualité de l'optimum, par rapport à la fiabilité des méta-modèles disciplinaires et à la précision de la résolution des équations d'état, assure la robustesse de cette méthode.

1.3.9. Modèles réduits et approximations

Cette section effectue une revue des différents méta-modèles. On se référera aux chapitres *Surfaces de réponse et réduction de modèles* et *Réduction de modèles, Éléments Finis par POD pour les problèmes paramétrés aux EDP* pour plus de détails. Les modèles réduits fournissent un moyen pratique pour réduire la complexité

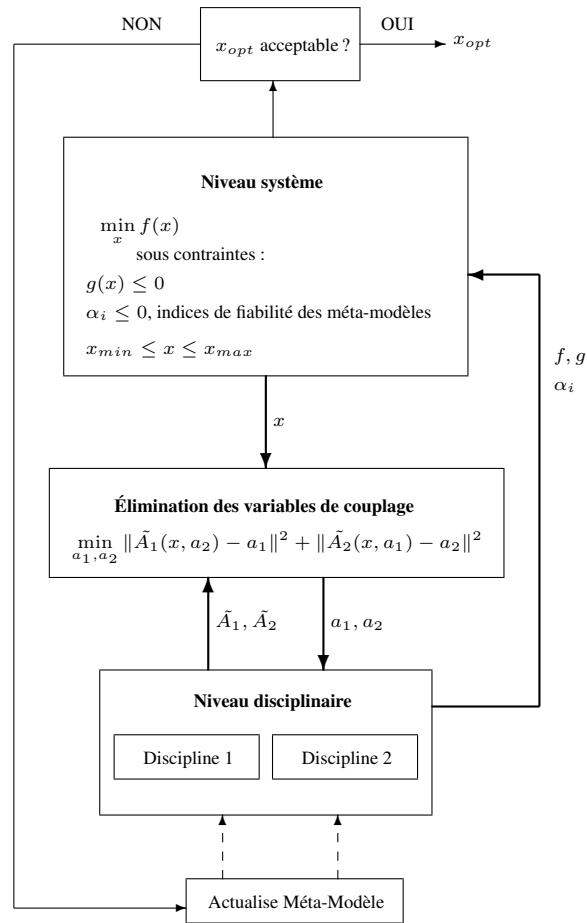


Figure 1.13. Diagramme de la formulation DIVE

en MDO. Ces derniers possèdent une validité locale à l'intérieur d'une région de confiance appropriée :

- la linéarisation est valide pour un grand nombre de paramètres si l'on utilise les méthodes adjointes, mais uniquement dans une petite région de confiance ;
- les méthodes de surface de réponse ont en général un domaine de validité assez grand pour un petit nombre de paramètres.

1.3.9.1. Approximation du premier ordre

La façon la plus simple de construire un méta-modèle local est de différencier (1.5) et (1.6). On obtient :

$$\begin{aligned} \tilde{a}_1 &= \partial_x A_1(x_{cur}, a_2(x_{cur}))(x - x_{cur}) \\ + \partial_{a_2} A_1(x_{cur}, a_2(x_{cur}))(a_2(x) - a_2(x_{cur})), \end{aligned} \quad (1.28)$$

et

$$\begin{aligned} \tilde{a}_2 &= \partial_x A_2(x_{cur}, a_1(x_{cur}))(x - x_{cur}) \\ + \partial_{a_1} A_2(x_{cur}, a_1(x_{cur}))(a_1(x) - a_1(x_{cur})). \end{aligned} \quad (1.29)$$

Cependant, la région de confiance est relativement petite. La méthode adjointe (mode inverse en différentiation automatique) peut être considérée pour les problèmes de grande dimension [GRI 00, GRI 90, GRI 89, MOH 01, MOR 85, OST 71, ROS 92, SPE 80, STR 90, WEN 64].

1.3.9.2. Approximation par des méthodes de paramétrisation

Les méthodes de paramétrisation sont très appropriées lorsque les disciplines sont faiblement couplées. Plus précisément, lorsqu'une discipline est indépendante des autres :

$$a_1 = A_1(x), \quad (1.30)$$

$$a_2 = A_2(x, a_1). \quad (1.31)$$

Dans cet exemple, la discipline 1 ne nécessite aucune information provenant de la discipline 2.

Nous pouvons trouver ce type de méta-modèle dans le domaine de l'aéro-élasticité. La sortie a_1 représente généralement la réponse de la structure à une excitation mécanique (une analyse modale est souvent nécessaire pour déterminer la réponse de la structure). Une bonne approximation consiste à utiliser la même base modale lorsque les paramètres de conception varient.

Pour construire des modèles réduits, on procède de la façon suivante. On résout d'abord les équations (1.5) et (1.6) pour plusieurs vecteurs x . Nous utilisons des méthodes DOE (Design Of Experience) pour sélectionner un échantillon de points : x^k , $k = 1 \dots n$, où n est petit.

La deuxième étape consiste à construire le modèle approché. Il existe deux catégories de méthodes :

1) l'approximation des fonctions objectif et contraintes utilisant les méthodes de surface de réponse [CRI 00, VAP 95, PLA 99], telles que les approximations polynomiales, les réseaux de neurones, SVM (Support Vecteur Machine), RBF (Radial Basis Functions), etc. ;

2) pour chaque vecteur x^k , on considère la solution u^k correspondante (c'est-à-dire le résultat de l'analyse du système). Pour un vecteur quelconque, on cherche la solution correspondante comme une combinaison linéaire des vecteurs u^k . Ce problème est petit et possède seulement une inconnue de dimension n . D'un point de vue numérique, il faut construire une base orthonormale à partir des vecteurs u^k . Pour cette raison, cette méthode est appelée POD (Proper Orthogonal Decomposition) [GUI 92, GUI 93, GUI 94, GUI 95, MAD 00, LAW 96, LEG 04, ROD 00, SAN 00, SER 98, SOB 00].

Si nous comparons les méthodes des catégories 1 et 2, la paramétrisation fournie par la méthode POD présente l'avantage de prendre en compte la connaissance du modèle.

Avec la paramétrisation, l'évaluation de la fonction coût est si peu coûteuse qu'il est possible d'utiliser des algorithmes génétiques [MÄK 98, PÉR 98], et d'autres algorithmes d'optimisation globale [KEL 99b, KEL 99a, MON 99].

1.4. Application à un cas-test d'avion supersonique

1.4.1. Description du problème de conception avant-projet d'avion supersonique

Le cas-test d'avion d'affaire supersonique, proposé par DASSAULT dans le cadre du projet OMD-RNTL, a permis d'expérimenter les formulations vues précédemment. Il est décrit brièvement dans cette section. Pour plus de détails, nous renvoyons à [RAV 07]. Ce cas test s'inspire largement de [RAY 94], qui est l'une des grandes références en conception avion avant-projet.

C'est un cas-test de type conception avant-projet. Il permet de calculer les performances d'un avion d'affaire supersonique avec des codes de calcul simplifiés. Le but de cet exercice de conception est d'obtenir une première configuration. Elle servira de base pour des études plus précises qui permettront d'assurer la viabilité du projet.

Il y a trois disciplines de conception :

- la discipline "structure" évalue les masses, dont la masse de l'avion au décollage, en fonction des paramètres de forme, de la masse au décollage elle-même, de la portance et de la masse moteur. La masse au décollage est alors réactualisée ;

- l'aérodynamique calcule la portance nécessaire en croisière en fonction de la masse, ainsi que la traînée en fonction des paramètres de forme, dont les dimensions du moteur fournies par la propulsion ;

– la discipline “propulsion” dimensionne les moteurs pour atteindre la poussée nécessaire. Elle influe donc sur la discipline “aérodynamique” au travers des dimensions des moteurs et sur la discipline “structure” avec la masse des moteurs.

L'organisation des disciplines est présentée sur le diagramme 1.14.

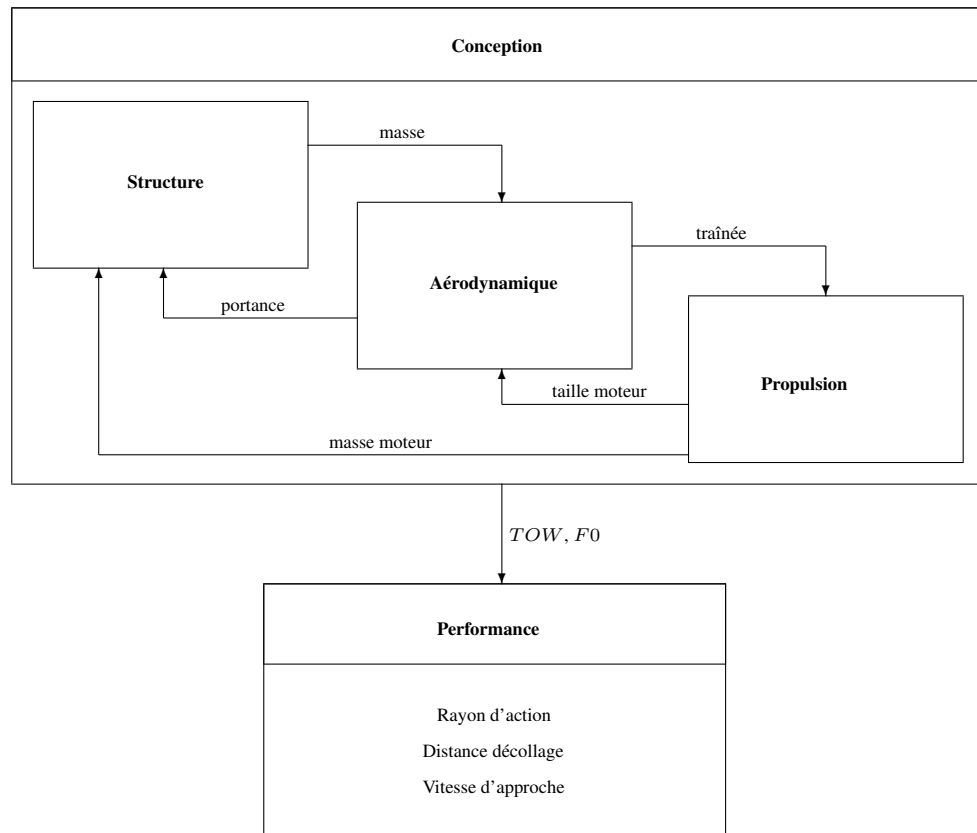


Figure 1.14. Organisation des disciplines

Le tableau 1.1 présente les différents paramètres choisis pour cette application.

Dans ce cas-test, l'équilibre entre ces trois disciplines est assuré par la convergence des variables que sont la masse au décollage TOW et la poussée $F0$. Les autres sorties, telles que la portance, la traînée ou la masse du moteur dépendent directement de ces dernières, et sont donc considérées comme transparentes.

Le tableau 1.1 présente les variables d'interaction de cette application.

Rubrique	Variable	Unité	Description
Conditions de vol	Z	m	altitude
	X_{mach}		nombre de Mach
	W_{fuel}	kg	masse de carburant
Géométrie Voilure	S	m^2	surface de référence voilure
	ϕ_0^w	deg	flèche bord d'attaque voilure
	ϕ_{100}^w	deg	flèche bord de fuite voilure
	Xl^w		effilement voilure
	t_c^w		épaisseur relative voilure
Géométrie Dérive	ϕ_0^t	deg	flèche bord d'attaque dérive
	ϕ_{100}^t	deg	flèche bord de fuite dérive
	Xl^t		effilement dérive
	t_c^t		épaisseur relative pour la dérive
Géométrie Fuselage	D_{fus}	m	diamètre du fuselage
Divers	α	deg	incidence max
	$Xfac$		$\frac{\text{masse à l'atterissage}}{\text{masse au décollage}}$

Tableau 1.1. Les variables de conception

Rubrique	Variable	Unité	Description
Interaction	TOW	kg	masse au décollage
	$F0$	N	poussée

Tableau 1.2. Les variables d'interaction

Lorsque l'équilibre disciplinaire est effectué, c'est-à-dire que l'on a défini les valeurs des variables TOW et $F0$ pour une configuration donnée, on peut alors calculer certaines sorties qui seront les critères dimensionnants pour l'avion. Ces critères peuvent être classés en deux catégories.

– Les performances en croisière :

- le rayon d'action : un avion doit pouvoir franchir une distance donnée afin de s'inscrire dans le segment de marché souhaité. Le rayon d'action est calculé avec la formule de Breguet.

- Les performances basse vitesse, l'avion doit respecter les contraintes réglementaires et liées aux aéroports :
 - la distance au décollage : l'avion doit pouvoir décoller sur une distance acceptable afin d'accéder à la plupart des aéroports,
 - la vitesse d'approche : l'avion doit pouvoir voler à une vitesse suffisamment basse sans risquer de décrocher afin d'atterrir dans de bonnes conditions.

Les critères en croisière et les critères basse vitesse sont en contradiction. En effet, une configuration optimisée pour le rayon d'action seulement n'aura jamais la portance nécessaire en basse vitesse pour satisfaire les critères de distance de décollage et de vitesse d'approche.

Le problème proposé ici est de minimiser la masse totale au décollage (TOW) tout en respectant les contraintes suivantes sur les sorties de la discipline performance :

- $R_{\text{action}} \geq 6500 \text{ km}$
- $D_{\text{décollage}} \leq 1828 \text{ m}$
- $V_{\text{approche}} \leq 70 \text{ m.s}^{-1}$

1.4.2. Résultats obtenus avec différentes formulations

Les formulations testées ici sont DIVE, MDF, IDF (AA0) et BLISS (voir les sections précédentes pour plus de détails sur ces formulations). Pour la méthode DIVE, les méta-modèles utilisés sont des approximations linéaires basées sur les gradients des sorties disciplinaires. Concernant la méthode BLISS, on considère ici une version mono-niveau de la méthode qui consiste à linéariser les sorties du système global. Naturellement, toutes les conclusions tirées de cette expérience doivent être généralisées avec circonspection ; elles constituent essentiellement des indicateurs de tendance demandant à être confirmées sur un ensemble plus vaste d'expérience utilisant des codes de calcul et des couplages interdisciplinaires de natures variées.

Les tableaux 1.3 à 1.5 présentent les résultats obtenus.

Le tableau 1.3 présente les variables optimales obtenues pour chacune des formulations testées. Les différences les plus marquantes concernent les variables suivantes :

- la flèche de bord de fuite de la voilure Φ_{100}^w atteint des valeurs autour de 5.5° pour DIVE, MDF et IDF et -5.3° pour BLISS ;
- l'effilement de la voilure Xl^w atteint sa borne inférieure de 0.05 pour MDF, IDF et BLISS, et la valeur 0.11 pour DIVE ;
- la flèche des bords d'attaque ϕ_0^t et de fuite ϕ_{100}^t de la dérive, et l'épaisseur relative pour la dérive t_c^t atteignent leurs bornes avec la méthode BLISS alors qu'elles ne s'éloignent pas de leurs valeurs de départ dans les autres cas.

Variables		DIVE	MDF	IDF	BLISS	min	max
Z	m	14585	14985	14845	14895	8000	18500
X_{mach}		1.6	1.6	1.6	1.6	1.6	2
S	m^2	100	100	100	100	100	200
ϕ_0^w	deg	46.1	51.8	51.9	46.8	40	70
ϕ_{100}^w	deg	5.5	5.6	5.4	-5.3	-10	20
Xl^w		0.11	0.05	0.05	0.05	0.05	0.5
t_c^w		0.04	0.046	0.0463	0.04	0.04	0.08
ϕ_0^t	deg	56	56	57	70	40	70
ϕ_{100}^t	deg	5	5	5	0	0	10
Xl^t		0.274	0.274	0.274	0.500	0.05	0.5
t_c^t		0.063	0.060	0.059	0.05	0.05	0.08
D_{fus}	m	2	2	2	2	2	2.5
W_{fuel}	kg	15000	15000	15000	15000	15000	40000
α	deg	15	15	15	15	10	15
X_{fac}		0.85	0.85	0.85	0.85	0.85	0.95
TOW	kg	33066	33207	32862	33065	1	100000
$F0$	N	62221	62704	61963	62220	1	300000

Tableau 1.3. Variables de conception optimales.

Mis à part ces quelques détails, les configurations obtenues avec les différentes méthodes sont semblables.

Sorties		DIVE	MDF	IDF	IDF +MDA	BLISS	min	max
TOW	kg	33066	33207	32862	33141	33065		50000
$F0$	N	62221	62704	61963	62356	62220		
R_{action}	m	6500	6501	6510	6443	6500	6500	
$D_{d\acute{e}collage}$	N	1828.0	1827.3	1826.4	1842.0	1828.0		1828
$V_{approche}$	$m.s^{-1}$	70.0	70.0	69.9	70.2	70.0		70

Tableau 1.4. Valeurs des sorties.

Le tableau 1.4 présente les sorties obtenues pour chacune des différentes formulations MDO. Les résultats obtenus sur la masse à minimiser sont assez semblables, de 32862 pour IDF à 33207 pour MDF, ce qui représente un écart de l'ordre de 1% entre les résultats les plus éloignés.

Les sorties obtenues avec IDF présentent une variabilité qu'il faut réduire au moyen d'une analyse couplée post-optimale. Celle-ci est effectuée en partant des variables optimales de la méthode IDF pour obtenir les résultats contenus dans la colonne IDF+MDA. Non seulement, la masse obtenue par la méthode IDF+MDA est légèrement inférieure, mais cette tendance est renforcée par le non respect des contraintes : la contrainte sur le rayon d'action, avec une vraie valeur de 6443 km, n'est plus respectée. Il en est de même pour les deux autres contraintes sur la distance de décollage et la vitesse d'approche.

	DIVE	MDF	IDF	BLISS
Calcul des masses	630	1459	1144	966
Aérodynamique	630	1459	1144	966
Propulsion	636	1490	1144	992
Performance	606	1552	1144	1044

Tableau 1.5. Nombre d'appels aux disciplines.

Le tableau 1.5 montre le nombre d'appels aux différentes disciplines. Ces nombres d'appels comprennent les appels nécessaires aux calculs des gradients et aux constructions des méta-modèles. Les temps d'appel aux méta-modèles sont considérés comme négligeables par rapport aux temps d'appel aux disciplines et ne sont pas comptabilisés dans ce tableau. La méthode DIVE présente le nombre d'appels aux disciplines le plus bas, de l'ordre de 630. Viennent ensuite les méthodes BLISS avec un nombre d'appels de l'ordre de 1000, IDF avec 1144, et MDF avec 1459.

Ces résultats montrent l'avantage d'utiliser des méta-modèles au niveau disciplinaire avec la méthode DIVE plutôt que d'utiliser des méta-modèles au niveau système avec la méthode BLISS. Nous voyons aussi que les méthodes utilisant des méta-modèles permettent d'avoir un nombre d'appels moins élevé aux disciplines. Enfin, pour ce cas-test qui ne possède que deux variables de couplage, la formulation IDF est plus rapide que la formulation MDF, mais elle reste toutefois moins précise et moins évidente au niveau de l'utilisation.

Les configurations obtenues étant assez semblables, la différence entre les formulations se fait par le nombre d'appels aux différentes disciplines sur ce cas-test. Ces

résultats montrent donc l'avantage de l'utilisation de méta-modèles disciplinaires avec la méthode DIVE par rapport aux autres méthodes. Il faut cependant pondérer cette conclusion en intégrant la construction des méta-modèles, qui peut s'avérer délicate et coûteuse selon la nature du modèle disciplinaire qu'il remplace.

1.5. Théorie des jeux

La théorie des jeux a été utilisée pour l'optimisation multi-critères [RAM 02]. Les travaux de J. Périaux et al. [PÉR 98, MÁK 98, WAN 92] l'ont révélée comme une excellente façon de traiter l'optimisation multidisciplinaire. La théorie des jeux semble être plus appropriée pour la conception multidisciplinaire que la plupart des méthodes classiques. Les critères comme le maximum de la contrainte de Von Mises, la traînée ou la portance, le maximum du déplacement et la consommation de carburant sont des critères de natures totalement différentes. Dans la section 1.3, nous considérons seulement la somme pondérée de ces critères, mais cette opération est arbitraire et dénature souvent le critère global. L'avantage de la théorie des jeux est de travailler dans un contexte multi-critères, et chaque discipline a pour but de défendre son propre objectif. Les développements exposés ici concernent les problèmes à deux objectifs.

Nous considérons ici chaque discipline comme un joueur. Soit f_A et f_B deux fonctions réelles définies sur A et B :

$$\begin{aligned} f_A : A \times B &\rightarrow \mathbb{R} \\ (x, y) &\mapsto f_A(x, y), \end{aligned} \quad (1.32)$$

et

$$\begin{aligned} f_B : A \times B &\rightarrow \mathbb{R} \\ (x, y) &\mapsto f_B(x, y), \end{aligned} \quad (1.33)$$

où A et B sont deux sous-ensembles de \mathbb{R}^n et \mathbb{R}^m . Le joueur A minimise la fonction f_A selon la variable x et le joueur B minimise la fonction f_B selon la variable y .

1.5.1. Équilibre de Pareto

1.5.1.1. Définition

Le point (x^*, y^*) est un optimum de Pareto (ou point d'équilibre de Pareto) s'il n'existe aucun point $(x, y) \in A \times B$ tel que :

$$\begin{cases} f_A(x, y) \leq f_A(x^*, y^*) \\ f_B(x, y) < f_B(x^*, y^*), \end{cases}$$

ou

$$\begin{cases} f_A(x, y) < f_A(x^*, y^*) \\ f_B(x, y) \leq f_B(x^*, y^*). \end{cases}$$

Cela veut dire que l'on ne peut pas améliorer un objectif sans dégrader l'autre. L'ensemble des points d'équilibre de Pareto est appelé front de Pareto. Lorsque le front de Pareto est convexe, on peut le déterminer en minimisant la fonction suivante :

$$f_\lambda = \lambda f_A + (1 - \lambda) f_B \text{ pour } \lambda \in [0, 1].$$

Chaque λ nous donne un point d'équilibre de Pareto. Ce front de Pareto sert de support pour trouver un compromis entre les différentes disciplines.

1.5.1.2. Exemple

Considérons les fonctions :

$$\begin{cases} f_A = (x - 1)^2 + (x - y)^2 \\ f_B = (x - 3)^2 + (x - y)^2. \end{cases} \quad (1.34)$$

On peut trouver les points d'équilibre de Pareto en considérant une somme convexe des deux critères :

$$f_\lambda = \lambda(x - 1)^2 + (1 - \lambda)(y - 3)^2 + (x - y)^2, \lambda \in [0, 1]. \quad (1.35)$$

Les conditions d'optimalité du premier ordre donnent :

$$\begin{cases} \partial_x f_\lambda = 2[\lambda(x - 1) + (x - y)] = 0 \\ \partial_y f_\lambda = 2[(1 - \lambda)(y - 3) - (x - y)] = 0. \end{cases} \quad (1.36)$$

Nous obtenons finalement :

$$\begin{cases} (1 + \lambda)x - y = \lambda \\ -x + (2 - \lambda)y = 3(1 - \lambda). \end{cases} \quad (1.37)$$

1.5.2. Équilibre de Nash

1.5.2.1. Définition

Le point (x^*, y^*) est un point d'équilibre de Nash si :

$$\begin{cases} f_A(x^*, y^*) = \inf_x f_A(x, y^*) \\ f_B(x^*, y^*) = \inf_y f_B(x^*, y). \end{cases} \quad (1.38)$$

Ce point d'équilibre doit satisfaire les conditions d'optimalité du premier ordre :

$$\begin{cases} \nabla_x f_A(x^*, y^*) = 0 \\ \nabla_y f_B(x^*, y^*) = 0. \end{cases} \quad (1.39)$$

1.5.2.2. *Exemple*

Nous considérons de nouveau l'exemple (1.34) :

$$\begin{cases} f_A = (x - 1)^2 + (x - y)^2 \\ f_B = (y - 3)^2 + (x - y)^2, \end{cases} \quad (1.40)$$

pour lequel l'équilibre de Nash est donné par :

$$\begin{aligned} \begin{cases} \partial_x f_A = 0 \\ \partial_y f_B = 0 \end{cases} &\iff \begin{cases} y = 2x - 1 \\ x = 2y - 3 \end{cases} &\iff \begin{cases} x = \frac{5}{3} \\ y = \frac{7}{3} \end{cases}. \end{aligned} \quad (1.41)$$

Nous pouvons remarquer que la solution (1.41) est la même que (1.37) avec $\lambda = 1$ dans la première équation et $\lambda = 0$ dans la seconde.

L'équilibre de Nash permet d'éviter des pondérations arbitraires des fonctions. Cependant, le compromis entre les disciplines est masqué par la définition des paramètres.

1.5.3. *L'équilibre de Stackelberg*1.5.3.1. *Définition*

Nous supposons désormais que l'un des deux joueurs est le meneur du jeu, par exemple le joueur A . Le problème devient alors :

$$\begin{aligned} \min_x f_A(x, y_x), \\ \text{où } y_x = \arg \min_y f_B(x, y). \end{aligned} \quad (1.42)$$

C'est une méthode qui permet de réduire le nombre de variables. Ici, nous avons donc à minimiser :

$$j(x) := f_A(x, y_x). \quad (1.43)$$

Supposons que f_A est une fonction relative à l'aérodynamique, par exemple la traînée, et que f_B est une fonction relative à la discipline "structure". Alors y représente les paramètres privés de la mécanique des structures (nombre de longerons, position, épaisseur). À chaque évaluation de f_B , on résout le problème d'optimisation de la discipline "structure".

1.5.3.2. Exemple

Nous considérons une dernière fois l'exemple (1.34) :

$$\begin{cases} f_A = (x - 1)^2 + (x - y)^2 \\ f_B = (y - 3)^2 + (x - y)^2. \end{cases} \quad (1.44)$$

L'équilibre de Stackelberg est le même que celui de Nash :

$$y_x = \frac{(x + 3)}{2},$$

$$j(x) = (x - 1)^2 + \left(\frac{x - 3}{2}\right)^2.$$

La minimisation de $j(x)$ nous donne :

$$x = \frac{5}{3} \quad \text{et} \quad y_x = \frac{7}{3}.$$

1.5.4. Application de la théorie des jeux à l'optimisation multidisciplinaire

Nous considérons les équations (1.23) et la fonction coût (1.7) :

$$a_1 = A_1(x, a_2), \quad (1.45)$$

$$a_2 = A_2(x, a_1), \quad (1.46)$$

$$\min_x f(x, a_1(x), a_2(x)), \quad (1.47)$$

$$g(x, a_1(x), a_2(x)) \leq 0$$

où le critère pondéré $f(x, a_1(x), a_2(x))$ est remplacé par $f_1(x, a_1(x), a_2(x))$ pour la discipline 1 et par $f_2(x, a_1(x), a_2(x))$ pour la discipline 2. Par exemple, si la discipline 1 est la discipline "structure", f_1 peut représenter le poids de la structure. Si la discipline 2 est l'aérodynamique, alors f_2 peut être la portance (avec une contrainte sur la traînée).

Nous considérons ensuite la décomposition introduite par la méthode CSSO (voir la section 1.3.7). Mais ici, la fonction coût système est remplacée par des fonctions

coût disciplinaires. La discipline 1 résout le problème suivant :

$$a_1 = A_1(x, \tilde{a}_2), \quad (1.48)$$

$$\tilde{a}_2 = \tilde{A}_2(x, a_1), \quad (1.49)$$

$$\min_{x_1} f_1(x_1, x_2, a_1(x), \tilde{a}_2(x)) \quad (1.50)$$

$$g(x, a_1(x), \tilde{a}_2(x)) \leq 0$$

$$(1 - \Delta_2)x_{1\ cur} \leq x_1 \leq (1 + \Delta_2)x_{1\ cur}, \quad (1.51)$$

et la discipline 2 résout :

$$\tilde{a}_1 = \tilde{A}_1(x, a_2), \quad (1.52)$$

$$a_2 = A_2(x, \tilde{a}_1), \quad (1.53)$$

$$\min_{x_2} f_2(x_1, x_2, \tilde{a}_1(x), a_2(x)) \quad (1.54)$$

$$g(x, \tilde{a}_1(x), a_2(x)) \leq 0$$

$$(1 - \Delta_1)x_{2\ cur} \leq x_2 \leq (1 + \Delta_1)x_{2\ cur}. \quad (1.55)$$

On peut exprimer les contraintes dans la fonction coût pour être exactement dans le contexte de la théorie des jeux.

La théorie des jeux est peu utilisée en optimisation multidisciplinaire alors qu'elle propose des solutions intéressantes :

- le front de Pareto est un bon outil pour les problèmes d'optimisation multidisciplinaire : parmi tous ses points, nous pouvons choisir celui qui traduit le mieux les critères implicites, tels que l'esthétique ou la faisabilité ;
- l'équilibre de Nash permet d'éviter les problèmes de compromis, bien qu'un compromis reste présent en amont dans le choix des paramètres des différentes disciplines ;
- l'équilibre de Stackelberg correspond à l'optimisation des variables privées de la discipline.

1.6. Autres approches pour la conception collaborative

Dans la section 1.3 nous avons présenté des formulations MDO dans lesquelles le système global complexe était divisé en sous-systèmes, chacun représentant une

partie spécifique de la conception. Ces méthodes reflètent l'organisation actuelle de l'industrie aérospatiale, dans laquelle la conception se partage entre plusieurs disciplines, comme l'aérodynamique, la discipline "structure", la propulsion, etc. Dans ces méthodes, tous les sous-systèmes sont considérés sur un pied d'égalité, et en principe il n'y a pas de restriction au niveau de la communication entre les disciplines.

Il existe cependant d'autres modes d'organisation de la conception qui ne sont pas basés sur un découpage disciplinaire, mais sur une organisation hiérarchique et sur une restriction de la communication entre les différents niveaux [MIC 00]. L'utilisation des méthodes présentées en section 1.3 reviendrait à changer intégralement leur organisation pour la conception. Une méthode telle que ATC (Analytical Target Cascading) [KIM 01, MIC 00, MIC 99] est plus appropriée dans ce cas d'organisation hiérarchique. La comparaison des méthodes ATC, CO et d'autres approches, est exposée dans [ALL 04]. Une nouvelle méthode basée sur ATC et CO est exposée dans [ALL 05].

D'autres méthodes, basées sur la théorie des jeux et les multi-agents, sont présentées dans [WOL 99, KRO 04]. Le système global n'est plus vu comme un ensemble de sous-systèmes, mais plutôt comme un ensemble de joueurs. Chaque joueur cherche à optimiser à la fois sa propre fonction objectif mais aussi une fonction globale commune à tous les joueurs. Un équilibre est atteint lorsque personne ne peut plus améliorer ni son objectif, ni l'objectif global. Ces techniques ont d'abord été appliquées aux problèmes de routage de réseaux et d'allocation de ressources [WOL 02, WOL 00, TUM 02], et seulement récemment aux problèmes aéronautiques [BIE 04, BIE 05, TAN 07].

1.7. Conclusion

La plupart des contributions considèrent la conception collaborative du point de vue de l'optimisation. Nous avons souligné ici l'importance de la définition des paramètres comme étape préliminaire pour l'optimisation multidisciplinaire. Cette étape possède un fort impact sur la configuration optimale obtenue. Ce choix des paramètres ne peut pas être fait entièrement de façon automatique, mais provient aussi de l'expérience des ingénieurs et d'un compromis à trouver entre les disciplines.

La conception collaborative consiste à fournir un environnement qui permette la communication entre les différentes disciplines et l'utilisation d'outils d'optimisation performants, mais aussi où les ingénieurs et les concepteurs gardent une place prépondérante.

Les étapes les plus importantes de la conception collaborative sont :

- la définition de l'ensemble des paramètres publics qui sont partagés par toutes les disciplines ;

- la création de modèles réduits disciplinaires, permettant une réponse rapide de chaque discipline dans le processus d’optimisation ;
- l’exploration de l’espace de conception afin de trouver les configurations les plus intéressantes.

Nous pouvons aussi définir deux types de fonctions coût : objectives et subjectives. Les critères subjectifs, comme l’esthétique ou la faisabilité, sont difficiles à traduire sous forme de modèles mathématiques, et ces notions sont laissées au jugement des ingénieurs et des concepteurs. Seuls les critères objectifs sont concernés par la modélisation mathématique et interviennent dans l’optimisation.

1.8. Bibliographie

- [ACH 95] ACHDOU Y., KUZNETSOV Y. A., PIRONNEAU O., « Substructuring preconditioners for the Q_1 mortar element method », *Numer.Math.*, vol. 71, p. 419-449, 1995.
- [AGT 00] AGTE J. S., A tool for application of bi-level integrated system synthesis (BLISS) to multi-disciplinary optimization problem, 2000.
- [ALE 98] ALEXANDROV N. M., KODIYALAM S., Initial results of an MDO method evaluation study, Rapport n°AIAA-98-4884, NASA Langley, 1998.
- [ALE 99] ALEXANDROV N. M., Comparative Properties of Collaborative Optimization and Other Approaches to MDO, Rapport n°ICASE-99-209354, NASA Langley, 1999.
- [ALE 00a] ALEXANDROV N. M., Analytical and Computational Aspects of Collaborative Optimization, Rapport n°NASA-2000-210104, NASA Langley, 2000.
- [ALE 00b] ALEXANDROV N. M., LEWIS R., « Algorithmic perspectives on problem formulations in MDO », *8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2000, AIAA Paper 2000-4719.
- [ALE 00c] ALEXANDROV N. M., LEWIS R. M., Analytical and Computational Properties of Distributed Approaches to MDO, Rapport n°AIAA-2000-4718, NASA Langley, 2000.
- [ALL 04] ALLISON J. T., Complex system optimization : A review of analytical target cascading, collaborative optimization, and other formulations, Master’s thesis, Department of Mechanical Engineering, University of Michigan, 2004.
- [ALL 05] ALLISON J. T., KOKKOLARAS M., ZAWISLAK M., PAPALAMBROS P., « On the use of analytical target cascading and collaborative optimization for complex system design », *6th World congress on Structural and Multidisciplinary Optimization*, Rio de Janeiro, 30 May-03 June 2005.
- [ALT 02] ALTUS T. D., A Response Surface Methodology for Bi-Level Integrated System Synthesis (BLISS), Rapport n°NASA/CR-2002-211652, NASA Langley Research Center, 2002.
- [BIE 04] BIENIAWSKI S. F., WOLPERT D., KROO I., « Discrete, Continuous, and Constrained Optimization Using Collectives », *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, NY, August 30-September 1 2004.

- [BIE 05] BIENIAWASKI S. F., Distributed optimization and flight control using collectives, PhD thesis, University of Stanford, U.S.A., October 2005.
- [BRA 96] BRAUN R., GAGE P., KROO I., SOBIESKI I., Implementation and performance issues in collaborative optimization, Rapport, AIAA, 1996.
- [CEA 86] CEA J., « Conception optimale ou identification de forme, calcul rapide de la dérivée directionnelle de la fonction coût », *M. A. A. N.*, vol. 20(3), p. 371–402, 1986.
- [CON 98] CONN A. R., SCHEINBERG K., TOINT P., « A derivative free optimization algorithm in practice », *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, 1998.
- [CRI 00] CRISTIANINI N., SHAWE-TAYLOR J., *An introduction to support vector machines*, Cambridge University Press, 2000.
- [GRI 89] GRIEWANK A., « *Mathematical Programming : Recent Developments and Applications* », Chapitre On Automatic Differentiation, Kluwer Academic Publishers, 1989.
- [GRI 90] GRIEWANK A., JUEDES D., SRINIVASAN J., TYNER C., « ADOL-C, a Package for the Automatic Differentiation of Algorithms Written in C/C++ », *ACM Trans. Math. Software*, 1990.
- [GRI 00] GRIEWANK A., « Evaluating Derivatives : Principles and Techniques of Algorithmic Differentiation », *SIAM*, 2000.
- [GUI 92] GUILLAUME P., MASMOUDI M., « Dérivées d'ordre supérieur en optimisation de domaines », *C. R. Acad. Sci. Paris, Série I*, vol. 315, p. 859–862, 1992.
- [GUI 93] GUILLAUME P., MASMOUDI M., « Calcul numérique des dérivées d'ordre supérieur en conception optimale de forme », *C. R. Acad. Sci. Paris, Série I*, p. 1091-1096, 1993.
- [GUI 94] GUILLAUME P., MASMOUDI M., « Computation of High Order Derivatives in Optimal Shape Design », *Numerische Mathematik*, vol. 67, p. 231–250, 1994.
- [GUI 95] GUILLAUME P., MASMOUDI M., « Solution to the Time-Harmonic Maxwell's Equations in a Waveguide, Use of High Order Derivatives for Solving the Discrete Problem », *SIAM Journal on Numerical Analysis*, 1995.
- [GUM 01] GUMBERT C. R., HOU J. W., NEWMAN P. A., Simultaneous aerodynamic and structural design optimization (sasdo) for a 2-d wing, Rapport n°AIAA-2001-2527, AIAA, 2001.
- [JAM 90] JAMESON A., « Control Theory Applications for Optimum Design of Aerodynamic Shapes », *29th IEEE Conference on Decision and Control*, Honolulu, p. 176-179, Dec. 1990.
- [KEL 99a] KELLEY C. T., « Detection and remediation of stagnation in the Nelder-Mead algorithm using a sufficient decrease condition », *SIAM Journal of Optimization*, vol. 10, n°1, p. 43–55, 1999.
- [KEL 99b] KELLEY C. T., « Iterative methods for optimization », *Frontiers in Applied Mathematics*, *SIAM*, 1999.
- [KIM 01] KIM H. M., Target cascading in optimal system design, PhD thesis, University of Michigan, U.S.A., 2001.

- [KRO 04] KROO I., « Collectives and complex design », *VKI lecture series on Optimization Methods and Tools for Multicriteria/Multidisciplinary Design*, November 15-19 2004.
- [LAW 96] LAWRENCE S., TSOI A. C., BACK A. D., « Function approximation with neural networks and local methods : bias, variance and smoothness », *Australian Conference on Neural Networks*, Australian National Univ., p. 16–21, 1996.
- [LEG 04] LEGRESLEY P. A., ALONSO J. J., « Improving the Performance of Design Decomposition Methods with POD », *10th AIAA/ISSMO Multidisciplinary Analysis*, AIAA, 2004, AIAA 2004-4465.
- [MAD 00] MADAY Y., PATERA A., TURINICI G., « A priori convergence theory for reduced-basis approximations of single-parameter elliptic partial differential equations », *J. Sci. Comput.*, vol. 17, n°1-4, p. 437-446, 2000.
- [MÄK 98] MÄKINEN R., NEITTAANMÄKI P., PÉRIAUX J., « A genetic algorithm for multiobjective design optimization in aerodynamics and electromagnetics », ET AL. P., Ed., *Computation Fluid Dynamics 98, Proceedings of the ECCOMAS 98 Conference*, vol. 2, Athens, Greece, Wiley, p. 418–422, 1998.
- [MAS 06] MASMOUDI M., PARTE Y., « Disciplinary Interaction Variable Elimination (DIVE) Approach for MDO », *ECCOMAS CFD 2006*, 2006.
- [MIC 99] MICHELENA N., KIM H. M., PAMPALAMBROS P., « A system partitioning and optimization approach to target cascading », *12th International Conference on Engineering Design*, Munich, Germany, 1999.
- [MIC 00] MICHELENA N., PAMPALAMBROS P., « Trends and challenges in system Design optimization », *International Workshop on Multidisciplinary Design Optimization*, Pretoria, S. Africa, August 7-10 2000.
- [MOH 01] MOHAMMADI B., PIRONNEAU O., *Applied Shape Optimization for Fluids*, Oxford University Press, 2001.
- [MON 99] MONGEAU M., KARSENTY H., ROUZÉ V., HIRIART-URRUTY J.-B., « Comparison of public-domain software for black box global optimization », *Optimization Methods and Software*, 1999.
- [MOR 85] MORGENSTERN J., « How to compute fast a function and all its derivatives, a variation on the theorem of Baur-Strassen », *SIGACT News*, 1985.
- [OST 71] OSTROVSKII G. M., VOLIN J. M., BORISOV W. W., « Über die Berechnung von Ableitungen », *Wissenschaftliche Zeitschrift der Technischen Hochschule für Chemie, Leuna Merseburg*, vol. 13, p. 382–384, 1971.
- [PAD 99] PADULA S. L., KORTE J. J., DUNN H. J., SALAS A. O., *Multidisciplinary Optimization Branch Experience Using iSight Software*, Rapport n°NASA-99-209714, NASA, 1999.
- [PÉR 98] PÉRIAUX J., *Genetic Algorithms and Evolution Strategy in Engineering and Computer Science : Recent Advances and Industrial Applications*, John Wiley & Son Ltd, 1998.
- [PLA 99] PLATT J. C., « *Advances in Kernel Methods - Support Vector Learning* », Chapitre Fast training of support vector machines using sequential minimal optimization, p. 185-208, N° 12, MIT Press, Cambridge, Massachusetts, 1999.

- [RAM 02] RAMOS A. M., GLOWINSKI R., PÉRIAUX J., « Nash equilibria for the multiobjective control of linear partial differential equations », *Journal of Optimization Theory and Applications*, vol. 112, n°3, p. 457–498, 2002.
- [RAV 07] RAVACHOL M., Rapport d'avancement. "Optimisation Multidisciplinaire", Rapport, DASSAULT AVIATION / OMD-RNTL, 2007.
- [RAY 94] RAYMER D. P., Aircraft Design : A Conceptual Approach, AIAA Education Series, New York, 1994.
- [ROD 00] RODRIGUEZ J. F., RENAUD J. E., WUJEK B. A., TRAPPETA R. V., « Trust Region Model Management In Multidisciplinary Design Optimization », *Journal of Computational and Applied Mathematics*, vol. 124, n°1-2, p. 139–154, 2000.
- [ROS 92] ROSTAING N., DALMAS S., « Automatic Differentiation Analysis and Transformation of Fortran Program Using a Typed Functional Language », *International Conference on Computing Methods in Applied Sciences and Engineering*, 1992.
- [SAL 97] SALAS A. O., ROGERS J. L., A Web-Based System for Monitoring and Controlling Multidisciplinary Design, Rapport n°NASA 97-206287, NASA, 1997.
- [SAL 98] SALAS A. O., TOWNSEND J. C., Framework Requirement for MDO Application Development, Rapport n°AIAA-98-4740, AIAA, 1998.
- [SAN 00] SANDY D., BALKIN D., LIN K., « A neural approach to response surface methodology », *Communications in Statistics, Theory and Methods*, vol. 29, n°9-10, p. 2215-2227, 2000.
- [SER 98] SERAFINI D. B., A framework for managing models in nonlinear optimization of computationally expensive functions, PhD thesis, Rice University, 1998.
- [SHA 93] SHANKAR J., RIBBENS C., HAFTKA R., WATSON L., « Computational study of nonhierarchical decomposition algorithm », *Computational Optimization and Applications*, 1993.
- [SOB 98] SOBIESKI J., AGTE J. S., SANDUSKY R. R., « Bi-Level Integrated System Synthesis (BLISS) », , n°AIAA 98-4916, 1998.
- [SOB 00] SOBIESKI I. P., KROO I. M., « Collaborative Optimization Using Response Surface Estimation », *AIAA Journal*, vol. 38, n°10, p. 1931–1938, 2000.
- [SPE 80] SPEELPENNING B., computing Fast Partial Derivatives of Functions Given by Algorithms, PhD thesis, University of Illinois, 1980.
- [STR 90] STRASSEN V., « *Handbook of Theoretical Computer Science* », Chapitre Algebraic Complexity Theory, Elsevier, Amsterdam, 1990.
- [TAN 02] TANG Z. L., DÉSIDÉRI J. A., Towards Self-Adaptive Parameterization of Bezier Curves for Airfoil Aerodynamic Design, Rapport n°RR-4572, INRIA, October 2002, URL :<http://www-sop.inria.fr/rapports/sophia/RR-4572.html>.
- [TAN 07] TANG Z. L., DÉSIDÉRI J. A., PÉRIAUX J., « Multicriterion Aerodynamic Shape Design Optimization and Inverse Problems using Control Theory and Nash Games », *Journal of Optimization, Theory and Applications (JOTA)*, vol. 135, n°1, Octobre 2007.

- [TUM 02] TUMER K., AGOGINO A., WOLPERT D. H., « Learning sequences of actions in collectives of autonomous agents », *First International Joint Conference on Autonomous and Multi-Agent Systems*, Bologna, Italy, July 2002.
- [VAP 95] VAPNIK V. N., *The Nature of Statistical Learning Theory*, Springer-Verlag, 1995.
- [WAN 92] WANG J. F., Optimisation distribuée multicritère par algorithmes génétiques et théorie des jeux - Application à la simulation numérique de problèmes d'hypersustentation en aérodynamique, PhD thesis, Université Paris VI, 1992.
- [WEN 64] WENGERT R. E., « A Simple Automatic Derivation Evaluation Program », *Comm. ACM*, vol. 7, p. 463–464, 1964.
- [WOL 99] WOLPERT D. H., TUMER K., An Introduction to Collective Intelligence, Rapport n°NASA-ARC-IC-99-63, NASA Ames Research Center, 1999.
- [WOL 00] WOLPERT D. H., WHEELER K., TUMER K., « Collective Intelligence for Control of Distributed Dynamical Systems », *Europhysics Letters*, vol. 49, n°6, p. 708-714, 2000.
- [WOL 02] WOLPERT D. H., TUMER K., « Collective Intelligence, Data Routing, and Braess' Paradox », *Journal of Artificial Intelligence Research*, 2002.