Etude et mise en oeuvre d'une méthode d'optimisation de forme couplant simulation numérique en aérodynamique et en calcul de structure

Meryem Marcelet

N°: 2008 ENAM 0039

Ecole doctorale n° 432 : Sciences des Métiers de l'Ingénieur

T H È S E

pour obtenir le grade de

Docteur

de

l'École Nationale Supérieure d'Arts et Métiers

Spécialité "Mécanique"

présentée et soutenue publiquement
par

Meryem MARCELET

le 10 décembre 2008

ETUDE ET MISE EN ŒUVRE D'UNE METHODE D'OPTIMISATION DE FORME COUPLANT SIMULATION NUMERIQUE EN AERODYNAMIQUE ET EN CALCUL DE STRUCTURE

Directeur de thèse : ***Alain LERAT***

Co-encadrement de la thèse : ***Jacques PETER et Gérald CARRIER***

Jury :

M. Pierre-Alain BOUCARD, Professeur, LMT Cachan ..... Rapporteur
M. Jean-Antoine DESIDERI, Directeur de recherche, INRIA, Sophia Antipolis.. Rapporteur
M. Pascal LARRIEU, Responsable des méthodes aérodynamiques, Airbus ..... Examinateur
M. Alain LERAT, Professeur, ENSAM, Paris ..... Examinateur
M. Mohamed MASMOUDI, Professeur, Institut de mathématiques de Toulouse . Président
M. Jacques PETER, Ingénieur de recherche, ONERA/DSNA, Châtillon ..... Examinateur

Laboratoire de Simulation Numérique pour la Mécanique des Fluides
ENSAM, CER de Paris

Remerciements

Je remercie Messieurs Pierre-Alain Boucard, professeur des universités affilié au LMT-Cachan, et Jean-Antoine Désidéri, directeur de recherche à l'INRIA, d'avoir jugé ce mémoire en tant que rapporteurs, Monsieur Mohamed Masmoudi, professeur à l'Institut de Mathématiques de Toulouse, d'avoir endossé le rôle de président de jury ainsi que Monsieur Pascal Larrieu, responsable des méthodes aérodynamiques chez Airbus, d'avoir accepté de prendre part à mon jury. J'adresse également tous mes remerciements à Monsieur Alain Lerat, professeur à l'ENSAM, pour avoir dirigé ma thèse, ainsi qu'à Messieurs Jacques Peter et Gérald Carrier, ingénieurs ONERA, pour m'avoir encadrée durant ces trois années. Mes remerciements vont également à toute l'équipe d'ingénieurs que j'ai côtoyée durant ces trois ans et dont la disponibilité, la facilité d'accès et les conseils m'ont été précieux ainsi qu'à tous les doctorants et stagiaires dont j'ai croisé la route - source intarissable de bonne humeur, d'entrain et d'humour. And last, but not least, les remerciements les plus inconditionnellement chaleureux vont à ma famille et à mes amis qui ont toujours su m'apporter le soutien et l'amour dont j'avais besoin.

Table des matières

Nomenclature

Problème d'optimisation

n	Nombre de paramètres de forme
p	Nombre de contraintes
$\mathcal{D}$	Domaine de conception ($\subset \mathbb{R}^n$)
α	Vecteur des paramètres de forme
J	Fonction objectif à minimiser
G_k	Fonction contrainte à satisfaire ($1 \leq k \leq p$) : $G_k(\alpha) \leq 0$

Notations générales

O	Origine du repère direct
(x, y, z)	Repère global associé au modèle (l'axe x pointe dans la direction du fuselage vers l'arrière de l'avion, l'axe y est dans le plan de l'avion et pointe à la droite du pilote, et l'axe z correspond à la verticale ascendante)
$\mathcal{F}$	Maillage du domaine occupé par le fluide
$\mathcal{F}^I$	Maillage de l'interface fluide-structure incluse dans le maillage $\mathcal{F}$
$\mathcal{S}$	Maillage de la structure
$\mathcal{S}^I$	Maillage de l'interface fluide-structure incluse dans le maillage $\mathcal{S}$

Modèle structure

n_p	Nombre de noeuds du maillage de la poutre
R_s	Système d'équations discrètes écrites sous forme de résidu pour la mécanique des structures
D	Champ des déplacements
ω_p	Champ de flexion sur la discrétisation de la poutre
θ_p	Champ de torsion sur la discrétisation de la poutre
Z	Maillage correspondant à une discrétisation de la structure (poutre)
Z_{rig}	Maillage correspondant à une discrétisation de la structure (poutre) avant déformation due aux effets aéroélastiques
P_i	Noeuds i du maillage de la poutre
(x_i, y_i, z_i)	Coordonnées du noeud P_i dans le maillage Z exprimées dans le repère global (x, y, z)
$(x_{rig_i}, y_{rig_i}, z_{rig_i})$	Coordonnées du noeud P_i dans le maillage Z_{rig} exprimées dans le repère global (x, y, z)

$\omega = (\omega_x, \omega_y, \omega_z)$	Déplacement de flexion exprimé dans le repère global (x, y, z)
$\theta = (\theta_x, \theta_y, \theta_z)$	Déplacement de torsion exprimé dans le repère global (x, y, z)
ω_{p_i}	Déplacement de flexion au noeud P_i
θ_{p_i}	Déplacement de torsion au noeud P_i
$\delta\omega_{p_i}$	Déplacement de flexion virtuel admissible du noeud P_i
$\delta\theta_{p_i}$	Déplacement de torsion virtuel admissible du noeud P_i
F	Matrice de flexibilité
${F_{ij}}^{zz}$	Déplacement selon z engendré au point P_i par un effort unitaire selon z au point P_j
${F_{ij}}^{z\theta_y}$	Déplacement selon z engendré au point P_i par un moment unitaire selon y au point P_j
${F_{ij}}^{z\theta_x}$	Déplacement selon z engendré au point P_i par un moment unitaire selon x au point P_j
${F_{ij}}^{\theta_y z}$	Déplacement angulaire autour de y engendré au point P_i par un effort unitaire selon z au point P_j
${F_{ij}}^{\theta_y\theta_y}$	Déplacement angulaire autour de y engendré au point P_i par un moment unitaire selon y au point P_j
${F_{ij}}^{\theta_y\theta_x}$	Déplacement angulaire autour de y engendré au point P_i par un moment unitaire selon x au point P_j
(F_x, F_y, F_z)	Effort d'origine aérodynamique exprimé dans le repère global (x, y, z)
(M_x, M_y, M_z)	Moment d'origine aérodynamique exprimé dans le repère global (x, y, z)
(X_i, Y_i, Z_i)	Repère local associé à la section $P_{i-1}P_i$ de la discrétisation de la poutre
$(F_{X_i}, F_{Y_i}, F_{Z_i})$	Effort d'origine aérodynamique exprimé dans le repère local (X_i, Y_i, Z_i)
$(M_{X_i}, M_{Y_i}, M_{Z_i})$	Moment d'origine aérodynamique exprimé dans le repère local (X_i, Y_i, Z_i)
$\omega = (\omega_{X_i}, \omega_{Y_i}, \omega_{Z_i})$	Déplacement de flexion exprimé dans le repère local (X_i, Y_i, Z_i)
$\theta = (\theta_{X_i}, \theta_{Y_i}, \theta_{Z_i})$	Déplacement de torsion exprimé dans le repère local (X_i, Y_i, Z_i)
E	Module d'Young du matériau
I	Moment quadratique autour de l'axe de flexion
G	Module d'élasticité de glissement
J	Moment quadratique d'une section transverse de la poutre autour de son axe neutre
L_S	Chargement aérodynamique équivalent appliqué à la structure
δD_S	Déplacement virtuel et admissible de l'interface $\mathcal{S}^I$
δW_S	Travail des efforts L_S lors du déplacement δD_S
$\mathcal{V}_i$	Volume d'influence du noeud P_i
$\overrightarrow{\mathcal{F}_i}$	Force appliquée au noeud P_i
$\overrightarrow{\mathcal{M}_i}$	Moment appliqué au noeud P_i
$\overrightarrow{\mathcal{F}}_S$	Somme des efforts appliqués aux noeuds du maillage $\mathcal{S}^I$
$\overrightarrow{\mathcal{M}}_S$	Somme des moments appliqués aux noeuds du maillage $\mathcal{S}^I$

Modèle aérodynamique

n_f	Nombre de noeuds du maillage du domaine fluide
n_{cell}	Nombre de cellules du maillage du domaine fluide
R_f	Système d'équations discrètes écrites sous forme de résidu décrivant le comportement du fluide

W	Grandeurs conservatives aérodynamiques
W^b	Grandeurs conservatives aérodynamiques à la frontière $\mathcal{F}^I$
X	Maillage structuré correspondant à une discrétisation du domaine fluide
X_{rig}	Maillage structuré correspondant à une discrétisation du domaine fluide avant déformation due aux effets aéroélastiques
N_{ijk}	Noeud d'indices (i, j, k) de la discrétisation du domaine fluide
$(x_{ijk}, y_{ijk}, z_{ijk})$	Coordonnées du noeud N_{ijk} dans le maillage X exprimées dans le repère global (x, y, z)
$(\tilde{x}_{ijk}, \tilde{y}_{ijk}, \tilde{z}_{ijk})$	Coordonnées du noeud N_{ijk} après transfert des déplacements de la structure au maillage X exprimées dans le repère global (x, y, z)
$(x_{rig_{ijk}}, y_{rig_{ijk}}, z_{rig_{ijk}})$	Coordonnées du noeud N_{ijk} dans le maillage X_{rig} exprimées dans le repère global (x, y, z)
d_{ijk}	Distance du noeud N_{ijk} à l'axe de la poutre
N'_{ijk}	Projeté du noeud N à coordonnée y sur la géométrie de la poutre
$(x'_{ijk}, y'_{ijk}, z'_{ijk})$	Coordonnées du noeud N'_{ijk} exprimées dans le repère global (x, y, z)
ω'	Déplacement de flexion interpolé au point N'_{ijk}
θ'	Déplacement de torsion interpolé au point N'_{ijk}
r	Rayon de propagation des déplacements de la structure
a	Coefficient d'amortissement du rayon de propagation des déplacements de la structure
ζ	Coefficient d'amortissement des déplacements de torsion
L_F	Chargement aérodynamique
δD_F	Déplacement virtuel de l'interface $\mathcal{F}^I$ équivalent au déplacement virtuel δD_S de l'interface $\mathcal{S}^I$
δW_F	Travail des efforts L_F lors du déplacement δD_F
n_f^I	Nombre d'interfaces dont est composée le maillage surfacique $\mathcal{F}^I$
I_l	Interface numéro l du maillage surfacique $\mathcal{F}^I$ $(1 \leq l \leq n_f^I)$
S_l	Aire de l'interface I_l
S_l^i	Aire de la partie de l'interface I_l incluse dans $\mathcal{V}_i$
G_l	Barycentre de l'interface I_l
y_l	Coordonnées selon $\vec{y}$ du barycentre G_l
$\overrightarrow{\delta G_l}$	Déplacement virtuel interpolé au point G_l
$\vec{n}_l = (n_{lx}, n_{ly}, n_{lz})$	Vecteur unitaire normal à l'interface I_l
$\vec{t}_l = (t_{lx}, t_{ly}, t_{lz})$	Vecteur unitaire tangent à l'interface I_l
p_l	Pression statique exercée par le fluide sur l'interface I_l du maillage surfacique $\mathcal{F}^I$
μ_l	Viscosité dynamique du fluide associée à l'interface I_l
V_t^l	Composante tangentielle de la vitesse du fluide sur l'interface I_l
μ_c	Viscosité dynamique du fluide au centre de la cellule adjacente à l'interface I_l
ρ_c	Masse volumique au centre de la cellule adjacente à l'interface I_l
$(\rho\overrightarrow{V})_c$	Quantité de mouvement au centre de la cellule adjacente à l'interface I_l
$(\rho u)_c$	Quantité de mouvement selon $\vec{x}$ au centre de la cellule adjacente à l'interface I_l
$(\rho v)_c$	Quantité de mouvement selon $\vec{y}$ au centre de la cellule adjacente à l'interface I_l
$(\rho w)_c$	Quantité de mouvement selon $\vec{z}$ au centre de la cellule adjacente à l'interface I_l

$(\rho e)_c$	Energie totale au centre de la cellule adjacente à l'interface I_l
c_c	Vitesse du son au centre de la cellule adjacente à l'interface I_l
p_c	Pression statique au centre de la cellule adjacente à l'interface I_l
Vol_c	Volume de la cellule adjacente à l'interface I_l
V_t^c	Projection de la vitesse au centre de la cellule adjacente à l'interface I_l sur le vecteur tangent à cette cellule
$\overrightarrow{\mathcal{F}}_F$	Somme des efforts appliqués aux noeuds du maillage $\mathcal{F}^I$
$\overrightarrow{\mathcal{M}}_F$	Somme des moments appliqués aux noeuds du maillage $\mathcal{F}^I$
Cl	Coefficient aérodynamique de portance
Clp	Coefficient aérodynamique de portance produit uniquement par le champ de pression
Cd	Coefficient aérodynamique de traînée
Cdp	Coefficient aérodynamique de traînée produit uniquement par le champ de pression

Calcul adjoint

λ_f	Vecteur adjoint de taille $(5n_{cell})$ (Euler) ou $(7n_{cell})$ (RANS) associé à la fonction objectif J et correspondant au système d'équations R_f
λ_s	Vecteur adjoint de taille $(2n_p)$ associé à la fonction objectif J et correspondant au système d'équations R_s
λ_{kf}	Vecteur adjoint de taille $(5n_{cell})$ (Euler) ou $(7n_{cell})$ (RANS) associé aux fonctions contraintes G_k, $1 \leq k \leq p$, et correspondant au système d'équations R_f
λ_{ks}	Vecteur adjoint de taille $(2n_p)$ associé aux fonctions contraintes G_k, $1 \leq k \leq p$, et correspondant au système d'équations R_s

Introduction

1. Position du problème

Pour répondre aux exigences de la Commission Européenne en termes de pollution environnementale et sonore, les avionneurs devront être capables, suivant les objectifs fixés par l'initiative technologique "Clean Sky", de fournir d'ici 2020 des avions dont les émissions de CO_2 auront été réduites de 50%, celles de NO_x de 80% et dont la pollution sonore aura été diminuée de 50%. Ces derniers doivent par ailleurs faire face à l'augmentation constante du prix du kérosène (qui, par exemple, a plus que doublé entre janvier 2007 et avril 2008 (+ 105%)). L'enjeu industriel auquel doivent répondre les avionneurs est donc double : fabriquer des avions moins polluants et d'exploitation plus économique. Pour répondre à ces objectifs, l'optimisation des formes aérodynamiques pour les phases de décollage et le point de croisière en particulier, constitue un moyen clef.

La recherche des formes aérodynamiques optimales est un domaine maintenant ancien dont la nécessité est apparue lors de la conception d'avions transsoniques et supersoniques (dont les performances aérodynamiques pour ces domaines de vol sont très sensibles au changement de forme). Cette dernière reposait et est encore dominée par le savoir-faire et l'expérience des ingénieurs de conception. Elle s'appuyait par ailleurs sur l'analyse des performances d'un nombre restreint de formes aérodynamiques préalablement sélectionnées. L'essor de la simulation numérique pour la résolution des équations de la mécanique des fluides a rendu possible le développement de l'optimisation numérique de forme, très prometteuse en terme d'accélération du processus de conception. Le problème de conception est considéré sous une forme mathématique plus ou moins simplifiée : on cherche à déterminer un ensemble de paramètres de forme, optimisant une fonction objectif (généralement la traînée) tout en satisfaisant un ensemble de contraintes. Ces fonctions dépendent habituellement de la réponse du système, cette dernière étant gouvernée par un système couplé d'équations discrétisant des équations aux dérivées partielles.

Ainsi, dès le milieu des années 1970, des ingénieurs de la NASA ont commencé à utiliser, lors de la conception d'aéronefs, les outils de simulation numérique pour la mécanique des fluides au sein de processus d'optimisation de forme [101, 100, 217, 103, 99, 215, 132]. Les configurations étudiées étaient simples (profils d'ailes, forme en plan de voilures) et les modèles aérodynamiques utilisés pour simuler le comportement du fluide étaient de faible fidélité. Les formes optimales étaient alors déterminées par des méthodes d'optimisation locale s'appuyant sur les gradients (méthodes de descente) - méthodes bien adaptées aux cas où le nombre de paramètres de forme est important. Les gradients des fonctions objectif et contraintes par rapport aux paramètres gouvernant la forme aérodynamique considérée, nécessaires au processus d'optimisation, étaient évalués par la méthode des différences finies. Enfin, seul le comportement aérodynamique de la forme aéronautique était pris en compte. Dès les années 1980, la puissance des calculateurs a rendu possible l'utilisation de modèles aérodynamiques de haute fidélité (équations d'Euler, de Navier-Stokes moyennées) pour l'optimisation numérique de forme. Même si, dès 1986, Sobieszcsanski-Sobieski [207] a mis au défi la communauté des aérodynamiciens d'inclure

le calcul des gradients dans les codes de simulation pour la mécanique des fluides, par rapport aux paramètres de forme, l'utilisation des différences finies s'est poursuivie jusqu'à très récemment.

A partir de ce contexte (méthode de descente, gradient par différences finies, fonction et modélisation exclusivement aérodynamique) l'optimisation de formes aéronautiques s'est ensuite généralisée dans trois directions.

La première concernait, de manière naturelle, le calcul des gradients. En effet, la méthode des différences finies présente, outre des problèmes d'imprécision, l'inconvénient de nécessiter au moins autant de calculs d'analyse qu'il y a de paramètres de forme par rapport auxquels le gradient des fonctions d'intérêt doit être évalué. Des méthodes analytiques ont ainsi été développées pour calculer ces gradients à moindre coût et de la manière la plus précise possible. Elles consistent toutes à résoudre un ensemble de systèmes d'équations linéaires issus du système non-linéaire des équations dont sont solutions les variables d'état du système. Si ce dernier est obtenu de façon directe par différenciation des équations d'état du système, on parle de la méthode de l'équation linéarisée ou de la méthode directe. Il y a alors autant de systèmes linéaires à résoudre que de paramètres de forme. En revanche, s'il est obtenu en passant par l'espace dual et par introduction des vecteurs adjoints aux équations d'état, on parle de la méthode de l'équation adjointe. Il y a alors autant de systèmes linéaires à résoudre que de fonctions à dériver. Si les équations linéarisées et adjointes sont déduites des équations de la mécanique sous forme continue, on parle de la méthode de l'équation linéarisée continue ou de la méthode de l'équation adjointe continue. Ces dernières doivent être discrétisées avant d'être résolues numériquement. Si au contraire, elles sont déduites des équations de la mécanique sous forme discrète, on parle de la méthode de l'équation linéarisée discrète ou de la méthode de l'équation adjointe discrète. Ces dernières, déjà sous forme discrète, peuvent alors être résolues numériquement. Bien que moins intuitive, c'est la méthode de l'équation adjointe qui, par diffusion du contrôle optimal [131], a tout d'abord été développée et appliquée à des configurations simples [170, 150, 171]. L'article de A. Jameson [113] rédigé en 1988 présentant la méthode de l'équation adjointe continue, avec transformations de coordonnées, pour les équations d'Euler est considéré par les publications récentes dans le domaine du calcul des gradients pour l'optimisation de forme en aérodynamique, comme la première contribution au domaine.

Dès le début des années 1990, la méthode de l'équation linéarisée a ensuite été étudiée dans le contexte de l'aérodynamique [199, 201, 20, 37]. La méthode de l'équation adjointe s'est cependant avérée plus adaptée aux problèmes concrets d'optimisation de forme puisque le nombre de paramètres de forme (plusieurs centaines pour des configurations réalistes) est en général bien supérieur au nombre de fonctions à dériver (pas plus d'une dizaine). Elle a ainsi évolué de manière à s'adapter aux progrès apportés aux codes de simulation numérique pour la mécanique des fluides [66, 184] et a été utilisée pour résoudre des problèmes de complexité croissante et avec des modélisations aérodynamiques de plus en plus fidèles : profils d'ailes [114, 116], voilures [115, 181], avions complets [184, 182, 183, 55].

La seconde extension des méthodes disponibles dans les années 80 a concerné l'optimisation globale. Ces méthodes d'optimisation globale ne font généralement appel qu'à l'évaluation des fonctions d'intérêt et ne nécessitent pas le calcul de leurs gradients par rapport aux paramètres de forme. Elles sont dès lors plus coûteuses que des méthodes d'optimisation par gradients pour des problèmes aérodynamiques complexes. Elles ont en revanche la capacité de localiser l'optimum global du domaine de conception. Parmi ces méthodes, citons les algorithmes évolutionnaires [177, 10] qui ont fait leur apparition en 1965, les algorithmes génétiques [104, 87] introduits en 1975, l'algorithme des colonies de fourmis [59, 60] développé en 1982, le recuit simulé [122, 44, 29] introduit dès 1983, les algorithmes à estimation de distribution [148, 130, 62] développés à la fin des années 1990, et l'algorithme d'essaims particulaires [121] élaboré au début des années 2000. Ces méthodes restent cependant, à quelques exceptions près [190, 154], moins utilisées dans le

domaine de l'optimisation de forme que les méthodes de gradients à cause du coût élevé de l'évaluation des fonctions d'intérêt pour les configurations complexes utilisant des modélisations aérodynamiques avancées.

Enfin, la troisième extension importante a concerné les méthodes d'optimisations multidisciplinaires et en particulier d'optimisation aéroélastique. Les interactions fluide-structure font partie des interactions prépondérantes sur les performances d'un avion en vol ; cependant, pour des raisons de complexité et de coût, les couplages entre les disciplines ont longtemps été négligés lors du développement et des applications concrètes des méthodes d'optimisation. Les aéronefs étaient ainsi traditionnellement conçus de façon séquentielle : la forme aérodynamique était définie de manière à induire le meilleur rapport portance/traînée - la structure étant supposée rigide, et la structure était conçue de sorte qu'après déformation en vol (sous une charge de 1-g), elle adopte la forme aérodynamique optimale prescrite - la dépendance du chargement aérodynamique aux critères d'optimisation de la structure étant ignorée [123, 21]. Négliger les couplages interdisciplinaires, en particulier les interactions fluide-structure, peut malheureusement entrainer des erreurs de conception importantes, surtout pour des conditions de vols transsoniques et supersoniques. Par exemple, ne pas prendre en compte le couplage aéroélastique lors de l'analyse d'une aile transsonique peut conduire à des valeurs de portance fortement erronées. De même, les phénomènes aéroélastiques indésirables exhibés par les avions de transport supersoniques et dus aux faibles raideurs en flexion et en torsion des ailes peuvent être supprimés par la prise en compte des interactions fluide-structure dès les premières étapes de sa conception [23]. De plus, une des motivations clef de la prise en compte des interactions fluide-structure est l'élimination de contraintes géométriques artificielles lors du processus d'optimisation de forme : si aucune contrainte n'est imposée sur l'épaisseur minimale des profils, une optimisation aérodynamique conduit à réduire le rapport épaisseur/corde jusqu'à un point où il devient impossible de concevoir une structure capable de supporter la charge aérodynamique et de loger à l'intérieur de l'aile optimisée. Pour des raisons d'ordre pratique, l'utilisation de modèles simples pour la mécanique des structures et de théories aérodynamiques linéaires ainsi que l'utilisation de la méthode des différences finies pour le calcul des dérivées, ont dominé, jusqu'à très récemment, le domaine de l'optimisation aéroélastique [92, 15, 30, 89, 75, 176, 16, 200, 223, 9, 85, 83]. En particulier dans tous les exemples suivants, que la modélisation physique soit avancée ou non, les dérivées associées au système aéroélastique considéré sont calculées par différences finies : optimisation aéroélastique par Grossman *et al.* [89] de la voilure d'un avion de transport transsonique utilisant la méthode des éléments finis pour la structure et la méthode linéaire du réseau de tourbillons ("*vortex lattice method*" en anglais) pour le fluide ; calcul des gradients associés à la voilure d'un avion de transport transsonique par Arslan et Carlson [9] utilisant un modèle de plaque plane pour la structure et les équations non-linéaires du potentiel des petites perturbations en régime transsonique pour le fluide ; optimisation aéroélastique d'un avion de transport supersonique par Giunta et Sobieszcsanski-Sobieski [85] couplant un modèle éléments finis linéaire pour la structure aux équations d'Euler pour le fluide.

En dépit des progrès qui ont été réalisés dans le domaine de l'optimisation multidisciplinaire en général et aéroélastique en particulier, la conception des systèmes aéronautiques complexes demeure un défi. Grâce à l'augmentation de la puissance des calculateurs, des modèles haute fidélité commencent aujourd'hui à être mis en oeuvre pour simuler le comportement des disciplines dans des processus d'optimisation aéroélastique : modèles éléments finis détaillés pour la structure et théories aérodynamiques non-linéaires (équations d'Euler, de Navier-Stokes moyennées) pour le fluide. En revanche, les gradients nécessaires au processus d'optimisation aéroélastique sont encore presque exclusivement calculés par différences finies, malgré les progrès effectués dans le domaine du calcul analytique des gradients aérodynamiques, comme nous l'avons vu précédemment, et des gradients de la structure [97, 124, 1]. Le cadre théorique du calcul analytique des

gradients associés à un système couplé quelconque a été développé en 1990 par Sobieszcsanski-Sobieski [209]. Dans le cadre plus particulier de l'optimisation aéroélastique, ce calcul nécessite la résolution d'un système complexe d'équations couplées fluide-structure appelé "*global sensitivity equation (GSE)*" en anglais. Ce système d'équations correspond en réalité au système d'équations couplées qui apparait lorsque la méthode de l'équation linéarisée est appliquée au système aéroélastique afin de calculer les gradients nécessaires au processus d'optimisation. Le calcul analytique des dérivées requises par un processus d'optimisation aéroélastique ou par un processus d'optimisation couplée en général s'appuyant sur une modélisation haute fidélité des disciplines concernées, constitue, encore aujourd'hui, un réel défi. Dans le domaine de l'aéroélasticité, la méthode directe développée par Sobieszcsanski-Sobieski a été utilisée, avec des modélisations haute fidélité, sur des problèmes bidimensionnels [78, 149], puis sur des problèmes tridimensionnels [110, 138]. La méthode de l'équation adjointe, mieux adaptée au cas où le nombre de paramètres de forme est largement supérieur au nombre de fonctions à dériver, a déjà été appliquée au calcul de gradients en thermoélasticité pour des problèmes stationnaires [143, 144], en thermoélastoplasticité pour des problèmes stationnaires et transitoires [146, 147] et en magnétohydrodynamique [145]. Dans le domaine de l'aéroélasticité, seuls Martins *et al.* [134, 135] ont utilisé la méthode adjointe en association avec une modélisation éléments finis simplifiée pour simuler le comportement de la structure et les équations d'Euler pour simuler le comportement du fluide (maillage volumique structuré) ; les gradients évalués ont servi à l'optimisation d'un avion d'affaire supersonique. Maute *et al.* [139] ont par ailleurs utilisé la méthode de l'équation adjointe en association avec une modélisation éléments finis détaillée pour simuler le comportement de la structure et les équations d'Euler pour simuler le comportement du fluide (maillage volumique non structuré) afin de calculer les gradients et optimiser une voilure transsonique théorique développée par la NASA.

2. Travail de thèse et contributions

L'objectif de cette étude était de proposer et d'étudier un cadre de calcul par la méthode de l'équation adjointe des gradients nécessaires à l'optimisation de forme d'un système aéroélastique statique. Ce travail s'est appuyé sur le cadre de calcul par la méthode adjointe des gradients aérodynamiques associés à une modélisation du comportement du fluide par les équations d'Euler ou par les équations de Navier-Stokes moyennées [164] et développé dans le logiciel elsA [40]. Afin que le calcul des gradients soit précis mais n'ait pas pour autant un coût prohibitif, nous avons choisi d'utiliser la théorie des poutres d'Euler-Bernoulli pour modéliser le comportement de la structure. Le champ des déplacements de la structure soumise au chargement aérodynamique est évalué en utilisant la matrice de flexibilité associée au modèle [24]. Cette dernière est construite itérativement en appliquant un chargement unitaire à chacun des noeuds du maillage de la structure et en évaluant le déplacement induit sur l'ensemble des autres noeuds. Le déplacement induit par un chargement quelconque sur la structure est alors égal au produit de la matrice de flexibilité et de ce chargement. Cette modélisation, bien que faiblement fidèle en toute généralité, prédit correctement le champ de déplacement des ailes à grand allongement [51, 162]. Le comportement du fluide peut, quant à lui, être modélisé par les équations d'Euler ou de Navier-Stokes moyennées. Le calcul analytique par la méthode de l'équation adjointe des gradients associés à un tel système aéroélastique n'a jamais été effectué. Il est cependant susceptible de fournir des résultats pertinents pour une phase de conception du type avant-projet par exemple. La démarche a consisté d'une part à mettre en place la résolution de l'équilibre statique du système aéroélastique décrit précédemment, et d'autre part à développer le cadre de calcul des gradients associés à ce système aéroélastique par la méthode de l'équation linéarisée puis par la méthode de l'équation adjointe.

Ces travaux ont fait l'objet de publications :

– J. Peter, M. Marcelet et S. Burguburu, "Introduction à l'optimisation de forme en aérodyna-

mique et quelques exemples d'applications", *Publication ONERA*, 2006-01, ISSN 0078-3780, 2006 ;

- M. Marcelet, J. Peter et G. Carrier, "Sensitivity analysis of a strongly coupled aero-structural system using the direct and adjoint methods", European Journal of Computational Mechanics, vol. 17/8, pp. 1077 – 1106, novembre 2008 ;

et de communications durant des congrès,

- M. Marcelet, J. Peter et G. Carrier, "Sensitivity analysis of a coupled aero-structural system using the direct differentiation and adjoint equations", *ECCOMAS* 2006, Egmond ann Zee, Pays-Bas, 5-6 septembre 2006 ;
- M. Marcelet, J. Peter et G. Carrier, "Sensitivity analysis of a coupled aero-structural system using both direct and adjoint methods", *AAAF,* 42^e *colloque d'aérodynamique appliquée, couplage et optimisation multidisciplinaire*, Sophia-Antipolis, France, 19-21 mars 2007.
- G. Carrier, S. Mouton, M. Marcelet et C. Blondeau, "Towards aerodynamic design by optimization of transonic transport aircraft in a multidisciplinary environment", 1^{st} *CEAS European air and space conference century perspective*, Berlin, Allemagne, 10-13 septembre 2007 ;
- M. Marcelet, J. Peter et G. Carrier, "Sensitivity analysis of a strongly coupled aero-structural system with direct and adjoint methods", AIAA Paper 2008−5863, 12^{th} *AIAA/ISSMO multidisciplinary analysis and optimization conference*, Victoria, Canada, 10-12 sepembre 2008.

Nous nous sommes de plus intéressés aux modèles réduits non physiques pour l'optimisation de forme. Nous avons considéré en particulier l'évaluation des performances de quatre modèles très utilisés en optimisation de forme globale : la régression polynômiale, les réseaux de neurones (perceptrons multi-couches et fonctions à base radiale) et le Kriging.

Depuis les années 1990, l'utilisation de modèles réduits non physiques ou métamodèles pour l'optimisation de forme est un domaine de recherche florissant [108, 203, 214, 174] - l'idée sous-jacente étant de remplacer en partie les évaluations exactes des fonctions d'intérêt requises par les processus d'optimisation globale par des évaluations approchées issues de modèles réduits mathématiques. Cette technique permet de réduire les coûts de calcul très importants associés aux processus d'optimisation globale effectués sur des modèles à haute fidélité. Un très grand nombre d'applications a été effectué parmi lesquelles : l'optimisation de forme globale, par algorithme génétique, d'un compresseur tridimensionnel par rapport à neuf paramètres et faisant appel à un réseau de neurones perceptron multi-couches [52] ; l'optimisation de forme globale, par algorithme génétique, d'une pale de turbine tridimensionnelle autour de laquelle le comportement du fluide est modélisé par les équations d'Euler et faisant appel soit à un réseau de neurones perceptron multi-couches, soit à un réseau de neurones par fonctions à base radiale [79] ; l'optimisation aéroélastique de forme globale d'un profil par rapport à deux paramètres faisant appel aux équations RANS (Reynolds Averaged Navier-Stokes) et au Kriging [118] ; ainsi que l'optimisation aéroélastique de forme globale d'une tuyère par rapport à deux paramètres, le comportement de la structure étant modélisé par des éléments finis, celui du fluide par les équations d'Euler et faisant appel à une régression polynômiale et au Kriging [202]. Les travaux sur ce sujet ont fait l'objet d'une publication :

- J. Peter et M. Marcelet, "Comparison of surrogate models for turbomachinery design", *WSEAS Transactions on fluid mechanics*, vol. 3, no. 1, janvier 2007 ;

d'un rapport du projet européen NODESIM CFD :

- J. Peter, M. Marcelet et S. Burguburu, "Comparison of surrogate models for the actual global optimization of a 2D turbomachinery flow. Assessment of design robustness w.r.t inlet angle", *rapport ONERA de la tâche 3.2 de NODESIM-CFD*, juillet 2008 ;

et d'un rapport ONERA,

– N. Bartoli, M. Samuélidès, D. Bailly et M. Marcelet, "Projet de recherche fédérateur DOOM - Revue de modèles réduits pour l'optimisation", RT 2/11576 DPRS/DTIM, décembre 2006.

Le chapitre 1 est consacré à la présentation du contexte de l'optimisation multidisciplinaire. Dans un premier temps, le problème d'optimisation à résoudre est formulé mathématiquement. Dans un deuxième temps, les principaux algorithmes d'optimisation de forme locale et globale sont présentés, ainsi que les modèles réduits non physiques auxquels ces derniers ont le plus souvent recours en pratique. Enfin, les principales méthodologies d'optimisation utilisées pour résoudre un problème d'optimisation multidisciplinaire sont détaillées.

Le chapitre 2 présente le calcul de l'équilibre aéroélastique statique. Le modèle fluide-structure utilisé est tout d'abord présenté. Puis les solveurs employés pour simuler le comportement du fluide et de la structure sont décrits. La méthode de couplage adoptée est alors exposée. Plus précisément, les méthodes de transfert du chargement aérodynamique à la structure d'une part, et de transfert du champ des déplacements de la structure au maillage du domaine fluide qui ont été retenues, sont décrites et justifiées au regard des conditions communément imposées lors d'une analyse aéroélastique. La méthode itérative choisie pour résoudre le système d'équations couplées décrivant l'état du système est ensuite présentée. Enfin les différents cas tests utilisés dans le cadre de cette étude sont décrits.

Le chapitre 3 est consacré au calcul des gradients associés au système aéroélastique présenté dans le chapitre 2. Dans un premier temps, le contexte de l'optimisation aéroélastique est rappelé. Dans un deuxième temps, les différentes méthodes numériques et analytiques du calcul des gradients d'un système couplé sont présentées, ainsi que la méthode itérative choisie pour résoudre les systèmes d'équations couplées qui peuvent leur être associées. Les dépendances des fonctions d'intérêt ainsi que le calcul des dérivées partielles sont détaillés. Enfin, ces méthodes sont appliquées aux cas tests présentés dans le chapitre 2.

Le chapitre 4 est quant à lui consacré à la présentation des modèles réduits non physiques étudiés ainsi qu'à la validation de l'utilisation de tels modèles approchés pour l'optimisation de forme globale d'une configuration 2D de turbomachine.

Chapitre 1

Principales méthodes d'optimisation multidisplinaire

1.1 Contexte et hypothèses

Ce chapitre a pour objectif la présentation la plus succincte et précise que possible des différentes notions relatives à l'analyse et l'optimisation d'un système multidisciplinaire, c'est-à-dire d'un système mettant en jeu de manière couplée plusieurs disciplines.

La conception d'une forme aéronautique peut notamment être vue comme un problème d'optimisation multidisciplinaire consistant à déterminer les valeurs d'un ensemble de paramètres de forme rendant une ou plusieurs fonctions objectifs optimales (trainée minimale, rayon d'action maximal, poids minimal ...) tout en satisfaisant un ensemble de contraintes (contraintes géométriques, portance minimale, poids maximal ...).

Cette étude s'intéresse aux problèmes d'optimisation comportant une fonction objectif et plusieurs fonctions contraintes. En général, lorsque le problème d'optimisation doit gérer simultanément plusieurs fonctions objectifs, soit il se ramène à une seule fonction objectif égale à la somme pondérée des fonctions objectifs du problème (la valeur des poids détermine alors l'importance donnée à chacune de ces fonctions, le processus d'optimisation s'en trouve alors en quelque sorte biaisé), soit il trace le front de Pareto, c'est-à-dire, pour un problème de minimisation, l'ensemble des solutions non-dominées ou encore des solutions parmi lesquelles on ne peut décider si une solution est meilleure qu'une autre, aucune n'étant systématiquement inférieure aux autres sur tous les objectifs.

On supposera dans toute la suite de l'étude d'une part que l'on cherche à minimiser la fonction objectif considérée par le problème d'optimisation et d'autre part que l'on s'est affranchi de toute contrainte d'égalité. Cette dernière hypothèse revient à supposer que les paramètres de conception sélectionnés sont indépendants, qu'ils ne sont liés par aucune relation implicite ou explicite (contraintes d'égalités). On supposera également que les contraintes d'inégalités ont été reformulées en contraintes d'infériorité. On fait de plus l'hypothèse pour l'ensemble de l'étude que la fonction objectif et les fonctions contraintes sont au moins de classe C^1. Afin d'assurer l'existence de minima locaux, on fera de plus l'hypothèse pour le reste de cette étude que la fonction objectif est au moins localement convexe.

Notons :

- n le nombre de paramètres de conception - "*design variables*" en anglais ;
- p le nombre de contraintes (d'infériorité) à satisfaire ;
- α le vecteur formé par l'ensemble de ces paramètres ;
- $\mathcal{D}$ (sous-ensemble de $\mathbb{R}^n$) le domaine de conception dans lequel ces paramètres peuvent évoluer ;

- J la fonction objectif (à minimiser) ;
- $(G_k)_{1\leq k\leq p}$ l'ensemble des contraintes (d'infériorité) à satisfaire.

Introduisons de plus :

- R le système d'équations discrètes non-linéaires écrites sous forme de résidu et décrivant l'état du système multidisciplinaire étudié ;
- M le vecteur constitué des coordonnées du maillage du domaine étudié ;
- Y le vecteur inconnu ou la variable d'état du système (de taille n_s)

L'équation d'état du système multidisciplinaire s'écrit alors :

$$R\left(M,Y\right)=0$$

On supposera dans toute la suite de l'étude que R est de classe C^1. La forme de l'entité aérodynamique étudiée est par définition une fonction continue et régulière du vecteur des paramètres de forme α, de telle sorte que le maillage du domaine considéré autour de cette entité est également une fonction continue et régulière de α :

$$M(\alpha).$$

L'équation précédente définit donc un système non-linéaire en Y de n_s équations à n_s inconnues.

Considérons les vecteurs (M_0,Y_0) tels que :

$$\begin{cases} R\left(M_0,Y_0\right)=0 \\ \\ det\left[\left(\dfrac{\partial R}{\partial Y}\right)\left(M_0,Y_0\right)\right]\neq 0 \end{cases}.$$

Le théorème des fonctions implicites permet alors de définir Y comme une fonction C^1 de M sur un voisinage de M_0. Par continuité et régularité de $M(\alpha)$, il est donc possible de définir Y comme une fonction C^1 de α dans un voisinage de α_0. On supposera dans la suite de cette étude que cette propriété est vraie sur $\mathcal{D}$.

Le problème d'optimisation multidisciplinaire, dont une solution est notée α^*, s'écrit alors sous forme canonique de la manière suivante :

$$\textit{trouver } \alpha\in\mathcal{D}\subset I\!R^n \textit{ tel que } J(\alpha) \textit{ soit minimale sur } \mathcal{D} \textit{ et que } \forall k\in[1,p]\ G_k(\alpha)\leq 0\ .$$

Le vecteur des paramètres de forme α est dit vecteur admissible si l'ensemble des contraintes d'infériorité sont satisfaites, en d'autres termes si :

$$\forall k\in[1,p]\ G_k(\alpha)\leq 0.$$

Pour un problème non contraint et en supposant en outre que la fonction objectif est de classe C^2, les conditions d'existence d'un optimum local ou global sont bien connues. En effet, pour que α^* soit un optimum local de J, il suffit que le gradient de J soit nul en α^* ($\nabla J(\alpha^*)=0$) et que la matrice hessienne de J en α^* ($\nabla^2 J(\alpha^*)$) soit définie positive ; en revanche, pour que α^* soit un optimum global de J, il suffit que le gradient de J soit nul en α^* ($\nabla J(\alpha^*)=0$) et que la matrice hessienne de J ($\nabla^2 J(\alpha)$) soit définie positive sur $\mathcal{D}$.

En revanche, pour un problème contraint, la condition d'optimalité est plus complexe. En effet, si par exemple l'optimum se situe sur une frontière de contrainte, c'est-à-dire s'il existe $k\in[1,p]$ tel que $G_k(\alpha^*)=0$, la contrainte k est alors dite active, alors le gradient c'est pas

nécessairement nul au minimum. La condition nécessaire d'optimalité du premier ordre, dite condition de Kuhn-Tucker, s'énonce comme suit :

- α^* est un vecteur admissible : $\forall k \in [1,p]\ G_k(\alpha^*) \leq 0$
- il existe p multiplicateurs λ_k tels que : $\begin{cases} \forall k \in [1,p],\ \lambda_k \geq 0 \\ \forall k \in [1,p],\ \lambda_k G_k(\alpha^*) = 0 \\ \nabla J(\alpha^*) + \sum_{k=1}^{p} \lambda_k \nabla G_k(\alpha^*) = 0 \end{cases}$

Si aucune contrainte n'est atteinte en α^*, la condition nécessaire d'optimalité est $\nabla J(\alpha^*) = 0$ (on retrouve bien la condition nécessaire d'existence d'un optimum pour un problème non contraint). En revanche, si une ou plusieurs contraintes sont actives ($\exists k \in [1,p],\ G_k(\alpha^*) = 0$) alors le gradient $\nabla J(\alpha^*)$ est une combinaison linéaire à coefficients négatifs ou nuls des gradients de ces contraintes.

La condition de Kuhn-Tucker est une condition suffisante d'optimalité si la fonction objectif et les fonctions contraintes sont convexes.

1.2 Différents algorithmes d'optimisation

Les calculs de gradients effectués dans le contexte de cette thèse tiennent compte des déformations aéroélastiques subies par les entités aérodynamiques que l'on cherche à optimiser mais sont destinés à l'optimisation de forme aérodynamique, sous-domaine de l'optimisation multidisciplinaire.

Introduisons les notations suivantes propres au calcul aéroélastique :

- R_f représente le système d'équations discrètes non-linéaires écrites sous forme de résidu et décrivant le comportement du fluide ;
- W est le vecteur des inconnues associées aux équations de la mécanique des fluides (champ conservatif) ;
- X est le maillage correspondant à une discrétisation du domaine fluide ;
- R_s représente le système d'équations discrètes (linéaires ou non-linéaires) écrites sous forme de résidu et décrivant le comportement de la structure ;
- D est le vecteur des déplacements de la structure ou vecteur des inconnues associées aux équations de la mécanique des structures ;
- Z est le maillage correspondant à une discrétisation de la structure.

En utilisant ces notations et les notations introduites dans la section précédente, le vecteur des inconnues (variables d'état) du système aéroélastique est le vecteur $Y = (W, D)$, le vecteur des coordonnées du maillage du système étudié est $M = (X, Z)$, et le système d'équation décrivant l'état du système aéroélastique est :

$$R(M,Y) = \begin{pmatrix} R_f(W, X(D,Z)) = 0 \\ \\ R_s(D,Z,W) = 0 \end{pmatrix} = 0.$$

La fonction objectif et les fonctions contraintes considérées dans cette étude sont donc fonctions du vecteur des paramètres de forme α et du champ aérodynamique W et du maillage

aérodynamique X,

$$\begin{cases} J(\alpha) = J(\alpha, X(\alpha), W(\alpha)) \\ \\ \forall k \in [1,p],\ G_k(\alpha) = G_k(\alpha, X(\alpha), W(\alpha)) \end{cases},$$

ces dernières étant calculées en résolvant le système d'équations décrivant l'équilibre aéroélastique :

$$\begin{cases} R_f(W, X(D,Z)) = 0 \\ \\ R_s(D,Z,W) = 0 \end{cases}.$$

L'amélioration des formes aérodynamiques peut se faire de manière expérimentale ou numérique. Du point de vue du numéricien, cela consiste à sélectionner une forme solide discrète, définie par son maillage de surface, parmi un ensemble de formes discrètes paramétrées - la forme sélectionnée optimisant une fonction objectif tout en satisfaisant un certain nombre de contraintes dont les valeurs dépendent directement ou indirectement de la géométrie et du champ aérodynamique.

De manière générale, on distingue deux types d'algorithmes d'optimisation : d'une part les algorithmes d'optimisation globale et d'autre part les algorithmes d'optimisation locale. Notons que quelque soit le type d'algorithme choisi, à chaque vecteur α est associé un maillage de la forme solide, auquel est associé un maillage du domaine occupé par le fluide qui l'entoure. Faire varier le vecteur des paramètres de forme au cours du processus d'optimisation entraine donc un remaillage du domaine fluide.

Les algorithmes d'optimisation globale recherchent un optimum global sur l'ensemble du domaine de conception $\mathcal{D}$. Ils travaillent directement sur un ensemble d'évaluations de la fonction objectif et les fonctions contraintes correspondant à un ensemble de paramètres de forme. Ces évaluations nécessitent le calcul des inconnues du système et en particulier du champ aérodynamique convergé, ce qui pour des modélisations à haut niveau de fidélité peut s'avérer très coûteux. A cette étape, l'algorithme d'optimisation fait donc en général appel à des modèles réduits. Ceux-ci peuvent être de deux types :

(a) des modèles réduits physiques, c'est-à-dire des modélisations à plus faible niveau de fidélité ;

(b) des modèles réduits non physiques ou méta-modèles, c'est-à-dire des approximations par interpolation ou régression des fonctions qui entrent en jeu dans le processus d'optimisation.

En d'autres termes, les algorithmes de recherche globale travaillent peu sur la fonction "exacte" (évaluations numériques avec la modélisation physique la plus fidèle pour le calcul des variables d'état puis des fonctions d'intérêt). Au contraire, ils utilisent des fonctions approchées pour sélectionner des zones d'intérêt où la fonction "exacte" est évaluée, ce qui permet à la fois d'enrichir la connaissance de cette fonction exacte et d'améliorer le méta-modèle utilisé pour l'approcher. Ces algorithmes font en général évoluer la population de manière non déterministe.

Les algorithmes d'optimisation locale cherchent un optimum local au voisinage d'un candidat initial en s'appuyant généralement sur les gradients des fonctions entrant en jeu dans le problème d'optimisation. Les algorithmes d'optimisation locale faisant appel au gradient partent d'une forme donnée, c'est-à-dire d'un vecteur de paramètres de forme initial, pour laquelle la fonction objectif, les fonctions contraintes ainsi que les gradients de ces fonctions par rapport au vecteur des paramètres de forme sont évalués. Le calcul des gradients requis par le processus d'optimisation n'est pas élémentaire car les variables à dériver sont couplées au maillage via les équations d'état discrètes du système. Ces gradients sont calculés soit par la résolution des équations linéarisées issues de la différentiation directe des équations d'état du système, il y a alors autant de systèmes linéaires à résoudre que de paramètres de forme, soit n, soit par la résolution

des équations adjointes, il y a alors $1+n_c$ systèmes linéaires à résoudre. L'algorithme de descente détermine un nouveau vecteur de paramètres de forme.

Les algorithmes d'optimisation globale et locale ne s'excluent pas. Au contraire il est possible et profitable de les combiner [41, 42]. Les algorithmes d'optimisation globale peuvent notamment servir à localiser les zones de "puits" et les algorithmes d'optimisation locale à affiner la recherche dans ces zones présentant un optimum local.

1.2.1 Méthodes de recherche globale

D'après Goldberg [87], les méthodes développées pour résoudre les problèmes d'optimisation globale peuvent être classées en trois types. Par ordre de robustesse croissante, il s'agit des méthodes déterministes, des méthodes énumératives et des méthodes stochastiques. Les méthodes déterministes ne font appel à aucun concept stochastique et utilisent toujours le même cheminement pour arriver à la solution. Parmi celles-ci on distingue :

(a) les méthodes énumératives qui consistent à évaluer tout simplement la fonction en chaque point de l'espace de recherche ; ces méthodes sont inutilisables en pratique car très peu adaptées à la CFD ;

(b) les méthodes d'exploration directes qui cherchent les minima locaux en se déplaçant dans une direction qui dépend du gradient des fonctions d'intérêt ;

(c) les méthodes d'exploration indirectes qui cherchent à atteindre les minima locaux en résolvant le système d'équations obtenu en annulant les dérivées de la fonction ; l'inconvénient de ces méthodes est qu'elles ne convergent vers l'optimum global que si le point de départ est proche de celui-ci ;

(d) les méthodes stochastiques utilisent exclusivement ou en partie un processus stochastique de telle sorte qu'à iso conditions initiales, un même algorithme peut proposer plusieurs solutions différentes ; cependant, les méthodes purement aléatoires qui explorent stochastiquement l'espace de recherche et mémorisent le meilleur élément trouvé ne sont ni efficaces ni robustes ; les algorithmes stochastiques de recherche d'optimum global sont en général itératifs et comprennent trois opérateurs : un mécanisme de perturbation, un critère d'acceptation et un critère d'arrêt ; l'inconvénient majeur des méthodes stochastiques est que leur convergence n'est garantie que de manière asymptotique.

Parmi ces méthodes, on distingue les algorithmes fondés sur la notion de parcours, des algorithmes qui utilisent la notion de population. Les premiers font évoluer α sur $\mathcal{D}$ à chaque itération, les plus connus sont le recuit simulé - "*simulate annealing*" en anglais - recherche avec tabous, recherche à voisinage variable, méthode GRASP, méthode de bruitage. Les seconds utilisent la notion de population, ces algorithmes manipulent simultanément un ensemble de solutions à chaque itération. Les plus connus sont les algorithmes génétiques, les algorithmes d'optimisation par essaims de particules et l'algorithme de colonies de fourmis. Tous ces algorithmes font en outre appel à la notion de mémoire, ils utilisent généralement l'historique de leur recherche pour guider les recherches aux itérations suivantes. Le cas le plus simple consiste à utiliser l'état des recherches à une itération donnée pour déterminer la prochaine itération, il s'agit alors d'une méthode sans mémoire appelée également processus de décision Markovien. Ces algorithmes de recherche globale peuvent également utiliser un algorithme général pour manipuler une population via des opérateurs (actions modifiant l'état d'un ou plusieurs vecteurs α) on dit alors qu'il est évolutionnaire. La structure générale des algorithmes évolutionnaires consiste à sélectionner un ensemble de α (individus), à les faire se reproduire, à les faire muter puis à en remplacer une partie.

Méthode de Monte-Carlo

La méthode de Monte-Carlo constitue, parmi les méthodes de recherche aléatoire, la méthode stochastique la plus simple. A chaque itération, un vecteur α est tiré au hasard dans l'espace de recherche $\mathcal{D}$, la valeur de la fonction objectif $J(\alpha)$ est comparée à la meilleure valeur obtenue jusqu'alors. Si elle est inférieure, alors ce vecteur α remplace le vecteur précédent gardé en mémoire, sinon un nouveau vecteur est tiré au hasard. Le processus continue ainsi jusqu'à ce qu'un critère de convergence soit atteint. Cette méthode n'est utilisable que lorsque le nombre de paramètres de formes n est faible et le coût d'évaluation de la fonction J faible. Elle est en particulier inapplicable pour des problèmes faisant intervenir une résolution numérique des équations de la mécanique des fluide - "*Computational Fluid Dynamic (CFD)*" en anglais. Ce qui est en particulier le cas de l'optimisation de forme.

Recherche tabou

Cette méthode - "*tabu search*" en anglais - a été proposée par F. Glover [86]. Elle consiste à partir d'un vecteur α dans l'espace $\mathcal{D}$, à explorer son voisinage et puis à choisir dans ce voisinage un vecteur α qui minimise la fonction objectif. Cette opération peut conduire à augmenter temporairement la valeur de la fonction (lorsque celle-ci prend des valeurs supérieures pour tous les points du voisinage) mais elle a l'avantage de permettre à l'algorithme de sortir des points d'optima locaux. Pour ne pas retomber dans un optimum local, l'algorithme garde en mémoire la liste des vecteurs α par lesquels il est passé "récemment", et s'interdit d'y repasser. Il s'agit de la liste des positions taboues. Pour certains problèmes, cet archivage peut représenter une grande quantité d'information, ce qui peut être problématique. Il existe plusieurs variantes suivant la façon de définir le voisinage et de gérer la mémoire. Il a été prouvé que l'algorithme de recherche tabou convergeait vers le minimum global de la fonction J moyennant des conditions strictes rarement satisfaites en pratique [86].

Algorithme génétique

L'utilisation des algorithmes génétiques dans le cadre de l'optimisation mathématique a été proposée pour la première fois par J. Holland [104] puis repris dans le cadre de la résolution de problèmes concrets par D. Goldberg [87]. Ces algorithmes sont des méthodes stochastiques définies originellement par analogie aux mécanismes génétiques. Elles font ainsi appel à des opérateurs de sélection, de croisement et de mutation. Les algorithmes n'utilisant pas l'opérateur de croisement sont qualifiés d'algorithmes évolutionnaires [177, 10]. Leur succès applicatif est dû à leur robustesse, leur capacité à traiter des variables discrètes et leur capacité à éviter les minima locaux. Ils présentent néanmoins l'inconvénient d'utiliser un grand nombre d'évaluations des fonctions objectif et contraintes.

Le principe général des algorithmes génétiques est le suivant : un ensemble de vecteurs $\alpha \in \mathcal{D}$ est généré aléatoirement, l'algorithme fait ensuite évoluer cet ensemble en sélectionnant les vecteurs pour lesquels $J(\alpha)$ est la plus faible possible tout en assurant l'exploration la plus complète possible de l'espace $\mathcal{D}$. Les algorithmes génétiques ne travaillent pas directement sur les vecteurs α mais utilisent un codage de l'espace $\mathcal{D}$. Ils ont de plus la caractéristique de travailler sur un ensemble de vecteurs α et de n'exiger aucune propriété de régularité de la fonction à minimiser.

La phase d'initialisation des algorithmes génétiques se scinde en deux étapes : la génération aléatoire d'une population initiale, c'est-à-dire d'un ensemble de candidats $\alpha \in \mathcal{D}$, et le codage des paramètres de forme. L'ensemble de départ doit être réparti uniformément sur l'espace $\mathcal{D}$ sauf si des zones d'intérêt sont connues a priori. L'algorithme doit être capable d'entretenir la

diversité de la population, c'est-à-dire la répartition des vecteurs α sur l'espace $\mathcal{D}$, pour permettre une exploration maximale de l'espace $\mathcal{D}$. Lors du codage de l'espace $\mathcal{D}$, une structure de données (appelée chromosome) est associée à chacun des vecteurs α. Les premiers algorithmes génétiques (1975) utilisaient un codage binaire [104]. Les codages les plus utilisés actuellement sont le codage binaire et la base dix [93]. L'usage est de concaténer les codages des composantes des vecteurs de paramètres de forme pour constituer une représentation génétique en une seule chaine. L'intérêt de ce type de codage est de permettre la création d'opérateurs de croisement et de mutation simple. Ce type de codage présente toutefois l'inconvénient de ne pas conserver la topologie de l'espace $\mathcal{D}$. En effet, deux vecteurs α proches dans l'espace $\mathcal{D}$ ne se retrouvent pas nécessairement proches après codage. Les codages qui ne conservent pas la topologie de l'espace de recherche $\mathcal{D}$ ont tendance à favoriser l'exploration de cet espace puisque le croisement de deux vecteurs parents peut générer des vecteurs enfants très éloignés dans l'espace de recherche $\mathcal{D}$. A l'inverse, les codages conservant la topologie de l'espace $\mathcal{D}$ favorisent l'exploitation de la population puisque l'application des opérateurs aura tendance à améliorer la performance des individus, c'est-à-dire à réduire la fonction J. L'utilisation du codage "réfléchi" ou codage de Gray [137] permet de conserver une distance constante après codage de deux vecteurs α successifs.

L'opérateur de sélection permet de choisir dans l'ensemble courant des vecteurs α, un sous-ensemble de vecteurs destinés à générer le nouvel ensemble, c'est à dire de nouveaux vecteurs d'évaluation de la fonction objectif $J(\alpha)$. Précisément, celui-ci a pour rôle d'identifier les vecteurs α pour lesquels les valeurs $J(\alpha)$ sont les plus faibles possibles et d'éliminer partiellement les autres vecteurs. Il est important de remarquer que tous les "mauvais" vecteurs ne doivent pas être éliminés. Une telle technique n'assurerait pas une exploration suffisamment exhaustive de l'espace $\mathcal{D}$ et se laisserait facilement emprisonner dans le voisinage d'optima locaux. La sélection ne doit donc être ni trop sévère ni trop lâche pour ne pas ralentir la convergence. L'algorithme historique utilisé par Holland consistait à accorder au vecteur α une probabilité de survie - c'est-à-dire d'appartenance au nouvel ensemble de candidats - proportionnelle à sa performance ou adaptation - "*fitness*" en anglais, cette dernière étant une fonction monotone décroissante de $J(\alpha)$. Les deux principes [87] de sélection les plus connus actuellement sont la sélection par roulette de casino - "*roulette wheel selection*" en anglais - et la stratégie du reste stochastique sans remplacement - "*stochastic remainder without replacement selection*" en anglais. Le premier associe à chacun des vecteurs α un segment dont la longueur est proportionelle à la performance. Les segments sont mis bout à bout et le segment total est normalisé entre 0 et 1. Un certain nombre de valeurs sont choisies aléatoirement entre 0 et 1. Les vecteurs α associés aux segments auxquels appartiennent ces valeurs sont sélectionnés. Un même vecteur α peut ainsi être sélectionné plusieurs fois. Le deuxième principe de sélection commence par une sélection déterministe des vecteurs α : pour chaque vecteur est calculée la partie entière du quotient r égal à la performance divisée par la performance moyenne, celle-ci sert à déterminer le nombre de fois où ce vecteur est utilisé pour générer de nouveaux vecteurs. Afin de ne pas éliminer tous les "mauvais" vecteurs, ce qui serait le cas avec cette sélection déterministe, une sélection aléatoire du type roulette de casino basée sur la performance $r - E(r)$ est effectuée ($E(r)$ représente la partie entière du quotient r). Ces deux principes de sélection sont en général combinés avec un principe de sélection élitiste. Celui-ci conserve automatiquement les meilleurs individus sans leur faire subir ni croisement ni mutation.

L'étape de croisement ou reproduction consiste à croiser les vecteurs α [173, 211]. Elle a pour but d'assurer la diversité de la population, l'idée sous-jacente étant de recombiner des séquences intéressantes des paramètres de forme. Le croisement peut être conditionné par un tirage aléatoire avec une certaine probabilité, une forte probabilité favorise l'exploration au risque de perdre de bonnes combinaisons. En général deux vecteurs parents, c'est-à-dire vecteurs destinés à être croisés, sont appairés aléatoirement et génèrent deux vecteurs enfants. L'algorithme historique développé par Holland utilisait une permutation des séquences de codage des deux paramètres à

droite et à gauche d'un point de coupure. Les techniques utilisées actuellement sont en général le "slicing crossover" ou le "k-point slicing crossover" : les chromosomes des parents sont découpés en k sous-chaînes et celles-ci sont ensuite échangées pour donner naissance aux vecteurs enfants.

L'étape de mutation permet d'améliorer la capacité des algorithmes génétiques à explorer l'ensemble du domaine $\mathcal{D}$ et d'améliorer ainsi la probabilité de l'algorithme de trouver une solution optimale. Elle a en effet pour objectif d'introduire ou de ré-introduire des caractéristiques génétiques en modifiant le codage de quelques candidats du nouvel ensemble de vecteurs α suivant une certaine probabilité de mutation. Elle permet également d'éviter la convergence prématurée de l'algorithme vers un minimum local. L'opérateur de mutation tire aléatoirement une séquence dans le chromosome associé à un vecteur α sélectionné et le remplace par une séquence choisie aléatoirement. La probabilité de mutation lors des changements de population ne doit pas être trop élevée pour éviter de rendre la recherche aléatoire.

Pour éviter une convergence trop rapide vers des zones de minima locaux, deux techniques sont usuellement employées. La première appelée mise à l'échelle - "*scaling*" en anglais - consiste à modifier la fonction d'adaptation pour qu'elle présente un gradient modéré. La seconde appelée partage - "*sharing*" en anglais - consiste à pénaliser la fonction d'adaptation en fonction du taux d'agrégation de la population dans le voisinage d'un vecteur α pour éviter le rassemblement des vecteurs α dans un puits.

Les algorithmes génétiques sont particuliérement robustes et permettent de minimiser des fonctions ne présentant aucune propriété de continuité ou de dérivabilité. Le théorème fondamental des algorithmes génétiques [65] indique que des sous-séquences courtes du paramètre optimal ont une probabilité forte d'être sélectionnées de génération en génération. Néanmoins, ils restent moins efficaces pour un problème donné qu'un algorithme déterministe spécifique. De plus, les nombreux paramètres de contrôle, en particulier probabilité de croisement et de mutation, sont délicats à régler. Enfin, un grand nombre d'évaluations de la fonction (pour le calcul de la fonction d'adaptation) est nécessaire pour assurer un bon niveau de robustesse et cet ensemble d'évaluation peut être très coûteux. Notons également que les algorithmes génétiques peuvent éprouver des difficultés à gérer de nombreuses contraintes.

Algorithmes à estimation de distribution

Les algorithmes à estimation de distribution - "*Estimation of Distribution Algorithms - EDA*" en anglais - ont été introduits par Muhlenbein en 1996 [148, 130, 62], ils résolvent le problème d'optimisation via un échantillonnage de la fonction $J(\alpha)$. Ils sont itératifs et travaillent sur un ensemble de vecteurs α. En revanche, à l'inverse des algorithmes évolutionnaires classiques, ils n'utilisent ni opérateur de croisement, ni opérateur de mutation pour générer le nouvel ensemble de vecteurs α à tester. Ces derniers sont choisis suivant une distribution de probabilité issue de la fonction J. Dans les deux cas, les algorithmes travaillent avec un ensemble de candidats α et les points ayant une bonne performance ("*fitness*") sont utilisés soit par les opérateurs de croisement et de mutation pour générer la nouvelle population de candidats, soit pour estimer une nouvelle distribution de probabilité pour la recherche suivante. Le comportement des algorithmes à estimation de distribution repose principalement sur le choix du modèle de distribution utilisé pour décrire le comportement de la fonction $J(\alpha)$. Celui-ci peut être sans dépendance, avec une dépendance bi-variante ou une dépendance multi-variante. La distribution de probabilité utilisée par un modèle sans dépendance ne dépend que d'une seule variable et est indépendant sur chaque variable. Les modèles sans dépendances sont simples à manipuler mais ne sont pas représentatifs des problèmes complexes pour lesquels les dépendances sont nombreuses. Pour un modèle de dépendance multi-variante toutes les dépendances possibles sont prises en compte. La loi de distribution la plus utilisée est la loi normale avec ou sans covariance. Les algorithmes à

estimation de distribution procèdent en trois étapes. Lors de la première étape un ensemble de vecteurs α est tiré au sort suivant une distribution de probabilité donnée, les vecteurs α pour lesquels la valeur de la fonction $J(\alpha)$ est la plus faible possible sont sélectionnés de manière déterministe et une loi de distribution décrivant cette nouvelle population est estimée lors de la seconde étape. Enfin, à l'étape trois, un nouvel ensemble de vecteurs α est tiré au sort suivant la loi de distribution estimée à l'étape deux. Ces étapes s'enchainent tant que le critère d'arrêt de l'algorithme d'optimisation n'est pas vérifié. Au fur et à mesure du déroulement de l'algorithme, l'échantillonnage se concentre autour de l'optimum. A condition que le nombre total de vecteurs α utilisés pour l'échantillonnage soit supérieur à une borne inférieure dépendant de la méthode employée pour évaluer la distribution de probabilité [228, 105], cette méthode converge vers le minimum global de la fonction J.

Recuit simulé

La stratégie du recuit simulé - "*simulated annealing*" en anglais - [29] s'appuie sur le phénomène physique décrit par la loi statistique de Boltzman. La technique du recuit thermique consiste à chauffer un matériau et à le refroidir dans certaines conditions. Lors du refroidissement, les atomes s'organisent en privilégiant la configuration dont le niveau d'énergie est le plus faible, c'est alors la configuration la plus stable. Tant que le niveau total d'énergie du matériau reste élevé, les atomes se réarrangent entre eux dans le but de trouver le niveau de plus faible énergie quitte à passer par des états moins stables. Comme au fur et à mesure du refroidissement, le niveau total d'énergie du système baisse, de sorte que les atomes ont de moins en moins de ressources pour passer à des états instables. Pour une température T, la probabilité qu'a un groupe d'atomes de passer à un niveau d'énergie supérieur d'un écart ΔE est fournie par l'équation de Boltzman : $exp(-\Delta E/T)$. Le recuit simulé imite ce phénomène pour chercher une solution au problème de minimisation. Il a tout d'abord été développé par trois chercheurs de la société IBM [122] puis indépendamment par V. Cerny [44].

Le recuit simulé utilise une méthode d'exploration de l'espace $\mathcal{D}$ à la fois aléatoire et dirigée afin de converger vers une solution proche de l'optimum. L'algorithme part d'une solution initiale et d'une température de départ qu'il fixe arbitrairement puis fait décroître la température par paliers selon une loi de décroissance choisie arbitrairement également. Il a été prouvé [129] que moyennant des conditions sur la loi de décroissance de la température, la méthode du recuit simulé convergeait asymptotiquement vers optimum global de J.

A chaque palier, l'algorithme teste un certain nombre de vecteurs α, ce nombre est en général proportionnel à la dimension de l'espace des paramètres de forme n. Si la valeur de $J(\alpha)$ est inférieure à la valeur minimale courante, le vecteur α est automatiquement accepté. Dans le cas contraire, le vecteur α est accepté avec une probabilité fonction de la température courante : l'algorithme calcule la probabilité $exp(\Delta J/T)$ et tire au hasard un nombre compris entre 0 et 1, si ce nombre est inférieur à la probabilité, la modification est acceptée. Les vecteurs α pour lesquels la valeur de la fonction objectif $J(\alpha)$ est supérieure à la valeur minimale courante ne sont pas nécessairement rejetés. La diminution progressive de la température limite cependant progressivement la liberté dont est pourvu l'algorithme de ne pas chercher à réduire la valeur de J à chaque étape. De cette manière, l'algorithme peut explorer un large domaine de $\mathcal{D}$ sans être piégé prématurément dans un puits. L'algorithme doit également choisir une loi de décroissance en température, la fonction la plus couramment utilisée pour cette dernière est :

$$T_{i+1} = aT_i$$

où a est une constante réglant la vitesse de décroissance, T_{i+1} et T_i respectivement les températures aux paliers $i+1$ et i de l'algorithme. L'algorithme itère jusqu'à ce que la température

d'arrêt fixée au préalable soit atteinte.

L'inconvénient majeur de cette technique est la dépendance au choix des paramètres qui la caractérisent : température initiale, température d'arrêt, nombre de paliers, longueur de chacun des paliers, et loi de décroissance de la température.

Intelligence en essaim

L'algorithme de colonies de fourmis ainsi que l'algorithme par essaims particulaires utilisent une méthodologie de collaboration entre différents agents de recherche ayant une connaissance limitée du système en vue de converger vers la solution optimale du problème. Ces techniques d'optimisation peuvent être regroupée sous le terme générique d'intelligence en essaim - "*swarm intelligence*" en anglais - [27].

• Algorithme des colonies de fourmis

Cet algorithme appelé "*Ant Colony Optimization*" en anglais a été proposé par Dorigo [59, 60] pour la recherche de chemins optimaux dans un graphe (problème du voyageur de commerce en particulier). Il peut cependant être appliqué à d'autres problèmes d'optimisation, en particulier les problèmes d'optimisation combinatoire. Comme l'algorithme précédent, il s'inspire d'un phénomène naturel, celui des colonies de fourmis cherchant le chemin le plus court entre leur colonie et une source de nourriture. Dans la nature, une colonie de fourmis réussit à trouver le chemin le plus court en procédant de la manière suivante : une fourmi éclaireuse parcourt au hasard l'environnement autour de la colonie, lorsqu'elle trouve une source de nourriture, elle retourne à la colonie en déposant sur son chemin des hormones volatiles appelées phéromones. Ces hormones étant attractives pour les autres fourmis, celles passant à proximité auront tendance à être attirées et à suivre cette piste, ce qui la renforce. Lorsque plusieurs pistes sont possibles, la plus courte est forcément parcourue dans le même temps par plus de fourmis, ce sera donc la piste la plus renforcée, et, à terme, la piste conservée. Sans la propriété de volatilité des phéromones, aucune piste ne serait choisie. Les fourmis collaborent donc en mélangeant exploration aléatoire et suivi de traces chimiques.

Originellement, l'algorithme de colonies de fourmis s'est inspiré de cette physique pour choisir le chemin le plus court sur un graphe (problème du voyageur de commerce). Cet algorithme est cependant capable de traiter un problème d'optimisation quelconque pour lequel la fonction à minimiser n'est plus la distance de parcours mais la valeur de la fonction objectif J et l'espace de recherche n'est plus l'ensemble des parcours possibles sur un graphe mais l'espace $\mathcal{D}$. Il procède par analogie au phénomène physique. Chaque vecteur α joue le rôle d'une fourmi et la valeur de la fonction objectif en ce point $J(\alpha)$ est associée à la distance parcourue par la fourmi. L'algorithme commence par choisir un ensemble de vecteurs α ou, avec l'analogie définie précédemment, un ensemble de fourmis. Dans l'algorithme original associé au problème du voyageur de commerce, chacune des fourmis parcourt un trajet parmi tous les trajets possibles et dépose sur celui-ci une quantité de phéromones proportionnelle à la distance parcourue et qui s'évapore au cours du temps. En utilisant l'hypothèse que plus un trajet est court, plus il a tendance à être emprunté par les fourmis, la quantité de phéromones associée à un trajet est d'autant plus forte que le trajet est court. Dans le contexte général de la minimisation de la fonction J, à chaque vecteur α testé est associée une quantité de phéromone d'autant plus forte que la valeur $J(\alpha)$ est faible. Les quantités de phéromones s'évaporent à chaque itération selon une loi choisie par l'algorithme. En d'autres termes, la valeur associée au vecteur α est modifiée au cours des itérations de l'algorithme. Les fourmis tiennent compte des marquages précédents pour optimiser leur recherche. L'algorithme général tient donc compte des valeurs de phéromone associées à α

(et reliées à $J(\alpha)$) calculées précédemment pour définir de nouveaux points d'évaluation α. Il a été prouvé [90] que l'algorithme des colonies de fourmis convergeait en un nombre fini d'itérations vers l'optimum global de la fonction J.

- **Algorithme par essaims particulaires**

Cet algorithme appelé "*particle swarm optimization*" en anglais a été proposé par R. Eberhart et J. Kennedy [121]. Comme l'algorithme des colonies de fourmis, il utilise la collaboration entre agents ayant une connaissance limitée pour trouver la solution optimale. Il procède ainsi en utilisant un essaim formé de particules qu'il fait évoluer au cours du temps. A chacune de ces particules est associée une vitesse. Celle-ci permet d'évaluer la position de la particule au pas de temps suivant. L'objectif est de faire converger l'essaim vers la solution optimale du problème.

Dans notre cas, les particules formant l'essaim se déplacent dans l'espace $\mathcal{D}$. A chaque étape du processus d'optimisation, chacune des particules définit (par la position qu'elle occupe) un vecteur α. L'objectif de l'algorithme est de faire évoluer ces particules dans l'espace $\mathcal{D}$ jusqu'à ce qu'elles s'agglomèrent en une certaine position α qui doit être un minimum de la fonction J. Pour cela, l'algorithme commence par initialiser l'essaim de recherche en définissant le nombre de particules qui le constitue ainsi que leur position de départ, c'est-à-dire le vecteur α que chacune représente. Le choix de ces vecteurs est fait de façon aléatoire ou de façon déterministe et régulière. L'algorithme doit également associer une vitesse initiale à chacune des particules. Le champ initial de vitesse est choisi de manière aléatoire. Enfin, l'algorithme doit définir la notion de voisinage pour chacune des particules. Deux types de voisinage peuvent être définis, le voisinage géographique qui doit être recalculé à chaque pas de temps, et le voisinage "social" défini une fois pour toute à l'initialisation. Ce dernier est le plus utilisé car il est plus simple à programmer, moins coûteux en temps de calcul et en cas de convergence, il tend vers le voisinage géographique. L'algorithme fait ensuite évoluer l'essaim en alternant calcul des nouvelles positions des particules (grâce aux vitesses associées à ces dernières) autrement dit choix de nouveaux vecteurs α à tester, et calcul des nouvelles vitesses associées aux particules. Le calcul de cette vitesse tient compte de trois tendances. La tendance à suivre sa propre voie, c'est-à-dire sa vitesse actuelle, la tendance conservatrice à suivre sa meilleure performance ainsi que la tendance panurgienne à suivre la meilleure performance de ses voisins. Chaque particule doit donc garder en mémoire sa meilleure performance, c'est-à-dire le vecteur α correspondant à la valeur $J(\alpha)$ minimum des vecteurs auxquels elle est passée. Pour calculer la vitesse de la particule au pas de temps suivante $v(t+1)$, l'algorithme recombine sa vitesse actuelle ($v(t)$), la position α_i à laquelle la particule a atteint sa meilleure performance ainsi que la position α_g à laquelle une particule de son voisinage a atteint sa meilleure performance. La nouvelle vitesse de la particule est alors calculée par la combinaison linéaire suivante :

$$v(t+1) = c_1 v(t) + c_2(\alpha_i - x(t)) + c_3(\alpha_g - x(t))$$

Le coefficient c_1 est fixé de manière arbitraire, et les coefficients c_2 et c_3 sont calculés aléatoirement. La position de la particule au pas de temps suivant est calculée par la formule :

$$\alpha(t+1) = \alpha(t) + v(t+1)$$

Avant de trouver l'optimum global, il est possible que l'essaim se décompose en sous-essaims autour d'optimum locaux.

Les points délicats de la méthode sont la définition de la taille de l'essaim, du voisinage des particules et du coefficient de confiance c_1 de la pondération linéaire. Une vitesse maximale est également souvent imposée afin d'éviter que le système "explose". Un ensemble de règles assurant la convergence vers l'optimum global de l'algorithme d'optimisation par essaim de particules ont été mises en évidence [213, 219].

1.2.2 Modèles approchés pour l'optimisation globale

Toutes les méthodes présentées dans la section précédente 1.2.1 peuvent substituer une valeur approchée de la fonction J en α à l'évaluation exacte $J(\alpha)$ au cours du processus d'optimisation. Le calcul exact de la fonction n'est alors utilisé que pour valider ou non le vecteur α candidat au minimum. Cette méthodologie est intéressante lorsque le calcul de la valeur exacte de la fonction J est coûteux, ce qui est en particulier le cas des calculs CFD. Le calcul de valeurs approchées de la fonction J est réalisé par des modèles de régression et d'interpolation. Les algorithmes d'optimisation présentés précédemment peuvent y faire appel pour approximer les valeurs des fonctions objectif et contraintes, en vue de réduire le coût du processus de recherche.

De façon générale, ces modèles sont qualifiés de modèles approchés non physiques par opposition aux modèles approchés physiques (pour un calcul aérodynamique, la théorie de la ligne portante de Prandtl est par exemple un modèle approché physique). Cette partie s'intéresse à la description succincte des principaux modèles utilisés dans le domaine de l'optimisation de forme, c'est-à-dire la régression par interpolation polynômiale, la régression par réseaux de neurones, l'interpolation ou régression par fonctions à base radiale, et l'interpolation par Kriging.

Ces modèles approchés ont pour objectif de fournir une approximation des fonctions considérées en tout point α de l'espace de recherche $\mathcal{D}$ (ou d'un sous-ensemble de celui-ci lorsque le processus utilise la notion de zone de confiance). On donnera les définitions pour la fonction J, la logique étant identique lorsque l'on s'intéresse à une fonction contrainte $G_k,\ k \in [1,p]$. L'objectif est donc de calculer une fonction $\tilde{J}$ approchant au mieux la fonction exacte J sur l'espace de recherche $\mathcal{D}$ (ou sur le sous-ensemble considéré). Le calcul de cette fonction approchée nécessite un échantillonnage initial de l'espace de recherche $\mathcal{D}$ (ou du sous-ensemble considéré) c'est-à-dire la sélection d'un ensemble discret et fini de points, notés $\{\alpha_1, ..., \alpha_{n_s}\}$, auxquels la fonction exacte est évaluée (on a noté n_s le nombre d'échantillons initials). L'objectif est donc de construire une fonction $\tilde{J}$ approximant la fonction J sur le domaine considéré à partir de l'échantillon $\{(\alpha_1, J(\alpha_1)), ..., (\alpha_{n_s}, J(\alpha_{n_s})\}$.

La difficulté technique générale sous-jacente à ce problème est le dilemne biais-variance : une description très riche de la fonction approchée (par un très grand nombre d'échantillons ou par un modèle d'approximation très précis comme un polynôme de haut degré) permet d'obtenir des évaluations très précises aux points de l'échantillon mais tend à avoir une variance très élevée et par conséquent à être imprécise en dehors du voisinage des points de l'échantillon. Inversement une description trop pauvre sera imprécise aux points de l'échantillon mais ne présentera pas de problème de variance trop élevée.

Construction du plan d'expérience initial

La qualité de l'approximation dépend de la répartition spatiale des points constituant l'échantillon appelé aussi plan d'expérience. Remarquons que l'initialisation des processus d'optimisation sans gradient (algorithme génétique par exemple) repose sur un échantillonnage de l'espace de recherche $\mathcal{D}$. Le choix des points échantillons ou la conception de plans d'expérience - "*design of experiment*" en anglais - est un sujet de recherche à part entière.

La première technique qui a été envisagée est le remplissage systématique ou factoriel - "*space filling*" en anglais - consistant à construire puis évaluer tous les vecteurs de paramètres dont la k-ième composante prend une valeur parmi ses bornes (α_{k-min},α_{k-max}), ou parmi les trois nombres (α_{k-min},$\dfrac{\alpha_{k-min}+\alpha_{k-max}}{2}$,$\alpha_{k-max}$). Le nombre d'évaluations pour le remplissage systématique est 2^n pour le premier choix et 3^n pour le second. Pour une géométrie à 18 paramètres de forme par exemple, le plan d'expérience systématique à trois valeurs par composante nécessite $3^{18} \simeq 4.10^8$ calculs. Cette technique est donc impraticable pour un nombre important

de paramètres de formes.

La méthode de l'hypercube latin définit pour chaque composante une suite arithmétique de p valeurs possibles, commençant et se terminant par les bornes de l'intervalle de variation de la composante. Un hypercube latin est une séquence de vecteurs α^l tels que les k^{ieme} coordonnées des vecteurs α^l aient toutes des valeurs distinctes (prises dans les suites). La taille de la séquence de vecteurs est inférieure à l'entier p. Inversement il est facile de construire des séquences de vecteurs α de taille p constituant un hypercube latin. Parmi les hypercubes latins, les meilleurs du point de vue de l'approximation de fonction sont ceux dont la somme des carrés des inverses des distances entre points est minimale (il s'agit évidemment d'un critère de bonne répartition entre points). Un second critère de sélection est la maximisation du déterminant de la matrice (qu'on notera X^TX) intervenant dans la régression polynômiale sur la base de l'échantillon. Les tableaux orthogonaux de Tagushi [196], enfin, sont des plans d'expérience construits dans le même cadre théorique, à partir d'hypercubes latins. La propriété des ces échantillonnages est que la matrice à inverser dans la régression polynomiale est orthogonale.

Régression par interpolation polynômiale

L'avantage de ce modèle d'approximation est son faible coût et le caractère explicite de la fonction approchée. Comme le nom de la méthode l'indique, elle cherche une approximation de la fonction exacte sous forme d'une fonction polynômiale. Illustrons-la par exemple lorsque le degré de la fonction polynômiale est choisi égal à 2. L'approximation quadratique de la fonction objectif en un vecteur $\alpha = [\alpha_1, ..., \alpha_n]$ de l'espace des paramètres de forme est de la forme [83] :

$$\tilde{J}(\alpha) = \psi_0 + \sum_{j=1}^{n} \psi_j \alpha_j + \sum_{j=1}^{n} \psi_{jj} \alpha_j^2 + \sum_{j=1}^{n} \sum_{\substack{k=1 \\ j<k}}^{n} \psi_{j,k} \alpha_j \alpha_k$$

où ψ_j, ψ_{jj} et $\psi_{j,k}$ sont les coefficients de l'approximation polynômiale. Il y a alors $n_t = (n+1)(n+2)/2$ coefficients à déterminer, ce qui nécessite donc au moins n_t évaluations exactes. Dans les cas pratiques, on dispose généralement d'un nombre de mesures très supérieur au nombre de coefficients. La détermination des coefficients ψ_j se fait alors par la méthode des moindres carrés.

En notant :

$$X = \begin{bmatrix} 1 & \alpha_1^1 & \alpha_2^1 & \dots & \alpha_n^1 & (\alpha_1^1)^2 & \dots & (\alpha_n^1)^2 & \alpha_1^1\alpha_2^1 & \alpha_1^1\alpha_3^1 & \dots & \alpha_{n-1}^1\alpha_n^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha_1^{n_s} & \alpha_2^{n_s} & \dots & \alpha_n^{n_s} & (\alpha_1^{n_s})^2 & \dots & (\alpha_n^{n_s})^2 & \alpha_1^{n_s}\alpha_2^{n_s} & \alpha_1^{n_s}\alpha_3^{n_s} & \dots & \alpha_{n-1}^{n_s}\alpha_n^{n_s} \end{bmatrix}$$

$$\Psi = \begin{bmatrix} \psi_0 & \psi_1 & \psi_2 & \dots & \psi_n & \psi_{1,1} & \dots & \psi_{n,n} & \psi_{1,2} & \psi_{1,3} & \dots & \psi_{n-1,n} \end{bmatrix}^T$$

$$J_s = \begin{bmatrix} J(\alpha_1) & \dots & J(\alpha_{n_s}) \end{bmatrix}^T$$

l'erreur d'approximation pour l'échantillon s'exprime par

$$X\Psi - J_s = \begin{bmatrix} \tilde{J}(\alpha_1) - J(\alpha_1) \\ \vdots \\ \tilde{J}(\alpha_{n_s}) - J(\alpha_{n_s}) \end{bmatrix}$$

L'interpolation au sens des moindres carrés cherche le vecteur Ψ des paramètres minimisant la norme de l'erreur quadratique - "*mean square error (MSE)*" en anglais :

$$\mathrm{MSE}(\Psi) = \sum_{i=1}^{n_s} (\tilde{J}(\alpha_i) - J(\alpha_i))^2 = ||X\Psi - J_s||_2^2 = \Psi^T X^T X \Psi - 2\Psi^T X^T J_s + J_s^T J_s$$

Le gradient de la fonction MSE(Ψ) est égal à ∇MSE $= 2X^T(X\Psi - J_s)$, en utilisant la convexité de la fonction on déduit que le vecteur Ψ minimisant cette fonction doit satisfaire la condition suivante :

$$X^T(X\Psi - J_s) = 0 \Longrightarrow \Psi = (X^T X)^{-1} X^T J_s$$

L'approximation polynômiale de degré deux est facile d'utilisation, elle est cependant d'une précision faible lorsque la fonction à approcher possède plusieurs extrema suivant certaines composantes. Des polynômes de degré plus élevé peuvent être utilisés, mais l'élévation du degré entraîne dans certains cas l'apparition d'oscillations artificielles. L'utilisation de fonctions splines multivariables permet de palier en partie à ce problème. Les splines multivariables sont des fonctions polynômiales définies sur des sous-domaines de l'espace D_α, dont les différents morceaux se raccordent de manière C^1 [11]. Elles permettent une meilleure approche des fonctions fortement variables. Enfin, lorsque les dérivées de la fonction à approcher sont connues, il est possible d'utiliser une méthode d'approximation multivariable de Hermite [224].

Régression par réseaux de neurones

L'idée des réseaux de neurones a pour la première fois été proposée par McCulloch et Pitts en 1943 [141] en s'appuyant sur une abstraction du neurone physiologique perçu comme un mécanisme matériel et logique. En 1949, Hebb [98] propose une règle d'apprentissage encore utilisée à l'heure actuelle, puis en 1958, Rosenblatt [186] présente le premier réseau perceptron composé d'une couche de perception (les entrées) et d'une couche de décision (la sortie), imaginant ainsi le premier système artificiel capable d'apprendre par expérience. Dans la même période, Widrow [225] développe "adaline" ("*adaptative linear element*") qui sera la base des réseaux multi-couches. Enfin en 1982, Hopfield [106] présente un réseau complétement rebouclé.

Dans le contexte des mathématiques appliquées, un neurone est une fonction algébrique non linéaire, paramétrée et à valeurs bornées. Un réseau de neurones est un assemblage par couches de neurones : la variable d'entrée des neurones d'une même couche est la combinaison linéaire des sorties des neurones de la couche précédente. Cet assemblage est construit de manière à définir une fonction de $\mathbb{R}^n$ dans $\mathbb{R}$. Pour définir un réseau de neurones, il faut donc choisir le nombre de couches, le nombre de neurones constituant chacune des couches, les coefficients associés à chacune des combinaisons linéaires ainsi que les paramètres définissant chacun des neurones. Ces coefficients représentent l'ensemble des degrés de liberté du réseau.

Les réseaux de neurones sont classiquement classés en deux types élémentaires :

- les neurones à "entrées paramétrées" pour lesquels les paramètres du réseau sont liés aux entrées du neurone (ce sont les coefficients associés aux différentes combinaisons linéaires) ;
- les neurones à "non-linéarités paramétrées", pour lesquels les paramètres sont attachés à la non-linéarité intrinsèque de la fonction.

Nous nous restreignons à la présentation des perceptrons multicouches et des réseaux de fonctions à base radiale. Les perceptrons multicouches (figure 1.1) - "*multilayer perceptrons (MP)*" en anglais - sont des réseaux de neurones à "entrées paramétrées". La sortie $y_{j,k}$ du neurone j de la sous-couche k est de la forme :

$$y_{j,k} = g_P(\psi_0 + \sum_{i=1}^{n_{k-1}} \psi_{i,j,k} y_{i,k-1})$$

où n_{k-1} représente le nombre de neurones de la sous-couche $k-1$, $y_{i,k-1}$ $(1 \leq i \leq n_{k-1})$ la sortie du neurone i de la sous-couche $k-1$, et ψ_0, $0 \leq i \leq n_{k-1}$ les coefficients associés au neurone j $(1 \leq j \leq n_k)$. La fonction g_P est la "fonction d'activation" du réseau. En général une

approximation de la fonction palier est choisie, par exemple, la fonction sigmoïde :

$$g_P(x) = \frac{1}{1 + exp(-x)}.$$

En particulier, la sortie du neurone j $(1 \leq j \leq n_1)$ de la première couche est :

$$y_{j,1} = g_P(\psi_0 + \sum_{i=1}^{n} \psi_{i,j,1}\alpha_i)$$

où α_i est la $i^{\text{ième}}$ composante du vecteur α (en particulier $n_0 = n$). De manière générale, les réseaux (MP) ont plusieurs neurones de sorties dont les valeurs sont combinées linéairement pour donner la sortie effective du réseau (neurone de sortie linéaire).

Les réseaux de fonctions [157, 166, 17] à base radiale (figure 1.2) - "*radial basis functions (RBF)*" en anglais - sont au contraire des réseaux de neurones à "non-linéarités paramétrées". Pour un réseau constitué d'une seule couche de fonctions à base radiale, l'entrée du réseau est le vecteur α. Chacun des neurones a pour sortie $g_R(\alpha, c, r) = exp(-||\alpha - c||^2/r^2)$ où c représente le centre de la fonction gaussienne g_R et r son écart type. La sortie finale du réseau est la combinaison linéaire des sorties des neurones de la couche cachée. Elle est égale à :

$$\tilde{J}(\alpha) = \sum_{i=1}^{n_1} \psi_i \; g_R(\alpha, c_i, r_i)$$

où n_1 représente le nombre de neurones de la couche cachée. Dans de nombreuses applications, seuls les coefficients ψ_i de la combinaison linéaire sont ajustés. En d'autres termes les centres et les rayons des fonctions gaussiennes sont fixés a priori. Dans ce cas, la régression devient linéaire. En particulier, les centres coïncident généralement avec les points échantillons. L'écart type r_i représente une mesure de la "zone d'influence" du centre c_i, plus celui-ci est faible, plus sa zone d'influence est faible et inversement. Une pratique commune consiste à associer à chacun des centres le même écart type calculé de la manière suivante :

$$r_i = \frac{\max\limits_{j,k \in [1,n_1]} ||c_j - c_k||}{\sqrt{n_1}}$$

ou encore, si les centres coïncident avec les points échantillons :

$$r_i = \frac{\max\limits_{j,k \in [1,n_s]} ||\alpha_j - \alpha_k||}{\sqrt{n_s}}.$$

Notons que le réseau (MP) utilise une décomposition en composante du vecteur d'entrée (vecteur des paramètres de forme) alors que le réseau (RBF) s'exprime naturellement en fonction du vecteur α.

On note de manière générale $\tilde{J}(\alpha, \psi)$ la sortie du réseau de neurones où ψ est le vecteur constitué de l'ensemble des coefficients définissant le réseau de neurones. Les coefficients sont solutions de la minimisation de l'erreur quadratique MSE. Il s'agit de la phase d'apprentissage :

$$\psi = arg\min_{\psi} (\text{MSE}(\psi)) = arg\min_{\psi} \left(\sum_{i=1}^{n_s} (\tilde{J}(\alpha_i) - J(\alpha_i))^2 \right).$$

Ce problème est résolu par un algorithme d'optimisation (les plus utilisés étant les algorithmes de descente dans la direction du gradient). La structure du réseau en couches successives rend

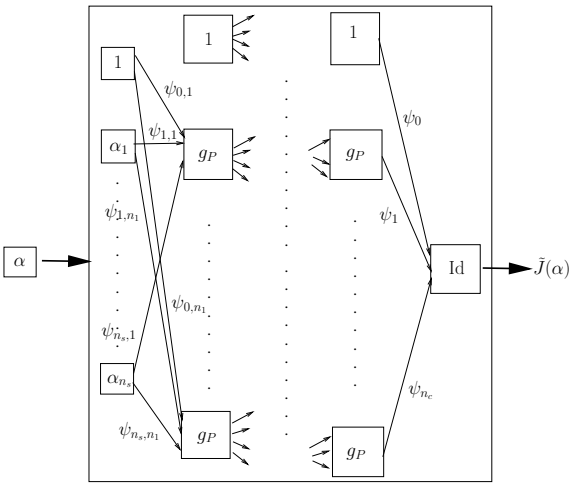

FIG. 1.1 – Schéma d'un perceptron multi-couche

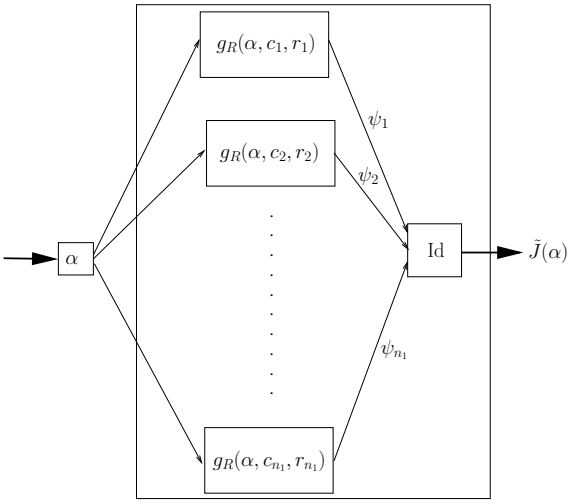

FIG. 1.2 – Schéma d'un réseau de fonctions à base radiale

le calcul des dérivées $\partial \text{MSE}/\partial\psi_i$ constituant le gradient ∇_ψ MSE particulièrement ardu. La méthode traditionnellement adoptée consiste à calculer ces dérivées en partant de la dernière couche puis en procédant par récurrence pour finir par la première couche du réseau. Il s'agit de l'algorithme dit de rétropropagation du gradient de l'erreur.

Pour le cas particulier des réseaux (RBF) dont les centres coïncident avec les points échantillons et pour lesquels les écarts types et le nombre de neurones ont été choisis a priori, le problème de minimisation de l'erreur quadratique dégénère en un problème linéaire d'inversion de matrice :

$$\begin{aligned} \psi &= arg\min_{\psi} \left(\sum_{i=1}^{n_s} (\tilde{J}(\alpha_i) - J(\alpha_i))^2 \right) \\ &= arg\min_{\psi} \left(\sum_{i=1}^{n_s} (\sum_{j=1}^{n_s} \psi_j \; exp(-\frac{||\alpha_i - \alpha_j||^2}{r_j^2}) - J(\alpha_i))^2 \right) \\ &= H^{-1} J_s \end{aligned}$$

où $H = \left(exp(-\frac{||\alpha_i - \alpha_j||^2}{r_j^2}) \right)_{1 \leq i,j \leq n_s}$.

Les techniques additionnelles dites de "validation croisée" permettent en outre de choisir le sous-ensemble de points échantillons à partir d'un ensemble de points échantillons plus important assurant la plus faible erreur de généralisation [61]. Les problématiques liées au choix du nombre de couches du réseau ainsi qu'au choix du nombre de neurones par couche ont fait l'objet de nombreux travaux [108, 109, 107, 14]. Il a en particulier été démontré que "toute fonction bornée suffisamment régulière peut être approchée uniformément et avec une précision arbitraire dans un domaine fini de l'espace de ses variables par un réseau de neurones comportant une couche de neurones cachés en nombre fini, possédant tous la même fonction d'activation et un neurone de sortie linéaire" [14]. Ce théorème assure la réalisabilité de la régression par réseaux de neurones.

Interpolation par Kriging (méthode DACE)

La méthode de régression introduite par D. G. Krige [126] dans le domaine des analyses minières dans les années 1950, nommée par la suite Kriging ou Krigeage, a été étendue à la géostatistique par Matheron [136] et Cressie [49], puis utilisée par Sacks, Welch, Mitchell et Wynn [188] dans le domaine de l'optimisation de forme. Ce processus a alors été nommé D.A.C.E

(Design and Analysis of Computer Experiments). Contrairement aux méthodes précédemment décrites, ces techniques déterminent une fonction approchée à partir de raisonnements statitiques. Le résultat des simulations numériques est, à la différences des expériences physiques, dépourvu d'erreur aléatoire, autrement dit, il est déterministe. Le raisonnement utilisé consiste cependant à considérer ses réponses $J(\alpha)$ comme étant les réalisations d'un processus aléatoire $\mathcal{J}(\alpha)$ de la forme :

$$\mathcal{J}(\alpha) = \mu(\alpha) + z(\alpha)$$

Dans cette expression $\mu(\alpha)$ une fonction de régression déterministe classique (polynômiale par exemple) tandis que $z(\alpha)$ est un processus stochastique représentant l'erreur de prédiction du modèle de régression $\mu(\alpha)$. On confondra in fine les échantillons $\mathcal{J}_s$ et J_s respectivement du processus $\mathcal{J}$ et de la fonction déterministe J que l'on cherche à approcher.

Le processus aléatoire $z(\alpha)$ est supposé avoir une moyenne (ou espérance) nulle $E(z(\alpha)) = 0$, une variance contante, notée σ^2 ($\sigma^2 = E(z(\alpha)^2)$), et la covariance entre deux erreurs $z(\alpha_1)$ et $z(\alpha_2)$ est égale à :

$$E(z(\alpha_1)z(\alpha_2)) = Cov(z(\alpha_1)z(\alpha_2)) = \sigma^2 Cor(\alpha_1, \alpha_2)$$

où $Cor(\alpha_1, \alpha_2)$ représente la fonction de corrélation entre les points α_1 et α_2. Ces hypothèses étant posées, il est nécessaire de préciser la fonction de corrélation utilisée. En général, celle-ci vérifie $Cor(\alpha_1, \alpha_2) = \phi^{\theta,p}(\alpha_1 - \alpha_2)$, et égale au produit des fonctions de corrélation $1D$ suivant :

$$Cor(\alpha_1, \alpha_2) = \phi^{\theta,p}(\alpha_1 - \alpha_2) = \prod_{i=1}^{n} e^{-\theta_i |\alpha_{1i} - \alpha_{2i}|^{p_i}}$$

où $\theta_i > 0$ et $0 < p_i \leq 2$. La fonction $\phi^{\theta,p}$ est une fonction décroissante de la norme du vecteur différence. Cela signifie que, lors d'une réalisation de la fonction $\mathcal{J}$ (et donc de la fonction écart), les variations de la fonction écart z sont d'autant plus fortement corrélées que les points sont proches. Cette fonction tend vers 0 quand la distance entre α_1 et α_2 augmente. L'influence d'un point de calcul α_i sur la valeur en α diminue donc à mesure que α s'éloigne de α_i. La valeur de θ permet de contrôler la zone d'influence : pour les fortes valeurs de θ, seules les valeurs proches sont bien corrélées (en 1D la fonction $\mathcal{J}(\alpha) - \mu(\alpha)$ tend vers des pics passant par les points de l'échantillon et s'appuyant sur la droite $y = 0$). Le paramètre p mesure la régularité de la fonction de corrélation. Les valeurs p_i, $1 \leq i \leq n$, sont en général prises égales à 2, de telle sorte que le processus aléatoire est un processus Gaussien.

Plusieurs formes de Kriging sont définies suivant le type de la fonction $\mu(\alpha)$:
- lorsque μ est une constante connue a priori ($\mu(\alpha) = m$) on parle de Kriging simple ;
- lorsque μ est une constante à déterminer ($\mu(\alpha) = \mu$) on parle de Kriging ordinaire,
- lorsque μ est une approximation par régression sur une base de fonctions (polynômiales par exemple) on parle de Kriging universel ($\mu(\alpha) = \sum_{j=1}^{k} \beta_j f_j(\alpha)$).

Dans le cadre du Kriging, la fonction $\mathcal{J}(\alpha)$ est approchée par une combinaison linéaire des valeurs du sampling, c'est à dire par l'estimateur linéaire suivant :

$$\tilde{J}(\alpha) = c^T(\alpha)\mathcal{J}_s$$

où $c(\alpha) = [c_1(\alpha)...c_{n_s}(\alpha)]^T$ est un vecteur de coefficients fonctions de α et, comme J est considérée comme la réalisation de $\mathcal{J}$, $\mathcal{J}_s = [\mathcal{J}(\alpha_1)...\mathcal{J}(\alpha_{n_S})]^T = [J(\alpha_1)...J(\alpha_{n_S})]^T = J_s$ représente le vecteur des valeurs de la fonction aux points de sampling.

L'estimateur choisi dans le cadre du Kriging est le meilleur estimateur linéaire non biaisé - "*best linear unbiaised predictor (BLUP)*" en anglais. En d'autres termes, il vérifie les deux conditions suivantes :
- il est non biaisé (condition (BLUP1)) :

$$E(\tilde{J}(\alpha)) = E(\mathcal{J}(\alpha)) \qquad \text{(BLUP1)}$$

- il minimise l'erreur quadratique (condition BLUP2) :

$$\text{Minimiser MSE}(\alpha) = E((\tilde{J}(\alpha) - \mathcal{J}(\alpha))^2) \qquad \text{(BLUP2)}.$$

Cet estimateur réalise une interpolation de la fonction J car il passe exactement par les points d'observation. La valeur MSE(α) permet d'évaluer le degré de confiance qui peut être accordée à la valeur prédite par l'estimateur linéaire. Comme la forme analytique de la fonction $J(\alpha)$ (calculée par simulation numérique) n'est pas connue, l'erreur MSE(α) est calculée en utilisant le modèle probabiliste $\mathcal{J}$ dont les paramètres dépendent exclusivement du sampling $\mathcal{J}_s = J_s$. La pertinence de cette erreur est donc fonction de la représentativité de ce sampling par rapport à la fonction réelle. Une faible valeur MSE(α) ne veut donc pas forcément dire que l'erreur réelle est faible.

Les fonctions $f_j, 1 \leq j \leq k$ et la forme de la fonction corrélation étant choisies, il reste à déterminer les paramètres : σ^2, $\theta = [\theta_1 \ ... \ \theta_{n_s}]$, $p = [p_1 \ ... \ p_{n_s}]$, et $\beta = [\beta_1 \ ... \ \beta_k]$. Pour choisir ces paramètres, Sacks, Welch, Mitchell et Wynn [188] proposent la technique dite du maximum de vraisemblance - "*maximum least estimatiom (MLE)*" en anglais. Connaissant un ensemble de réalisations de la fonction J, il s'agit de trouver les paramètres du problèmes qui maximisent la probabilité d'obtenir ces valeurs. Autrement dit, il s'agit d'optimiser la probabilité conditionnelle $P(\mathcal{J}_1, ..., \mathcal{J}_{n_s} | \sigma^2, \theta, p, \beta)$. En général, les paramètres $p_i, 1 \leq i \leq n_s$ sont choisis égaux à 2 et les paramètres σ^2, θ, β solutions du problème de maximisation (MLE). La méthode du maximum de vraisemblance et la méthode (BLUP) conduisent au même β.

En résumé, pour définir complétement la fonction de covariance Cov (choisie sous la forme $\sigma^2 \prod_{i=1}^{n} e^{-\theta_i |\alpha_{1i} - \alpha_{2i}|^{p_i}}$), les paramètres σ^2, $\theta = [\theta_1 \ ... \ \theta_{n_s}]$ et $p = [p_1 \ ... \ p_{n_s}]$ doivent être précisés. L'approche (BLUP) ne permet pas de calculer ces paramètres, ceux-ci doivent être fixés au préalable. En revanche, l'approche faisant appel au maximum de vraisemblance permet de les calculer (le vecteur p est cependant souvent fixé a priori). La méthode (BLUP) ne permet donc de calculer que les coefficients de l'approximation linéaire $\tilde{J}$, les paramètres θ et p étant fixés Tandis que la méthode (MLE) conduit à une expression de l'approximation linéaire identique et permet en outre de calculer les paramètres θ et p. La variance σ^2, calculable analytiquement grâce à la méthode (MLE), n'intervient que dans l'expression de l'erreur MSE mais pas dans le calcul de l'approximation linéaire $\tilde{J}$.

Que les paramètres θ et p aient été choisis ou calculés par la méthode (MLE), l'estimateur linéaire $\tilde{J}$ de la fonction J calculé par la méthode du Kriging est égal à* :

$$\tilde{J}(\alpha) = f_\alpha^T \hat{\beta} + K_\alpha^T C^{-1} (\mathcal{J}_s - F\hat{\beta})$$

où

$$\begin{cases} \bullet \ \hat{\beta} = (F^T C^{-1} F)^{-1} F^T C^{-1} \tilde{\mathcal{J}}_s \\ \bullet \ f_\alpha = [f_1(\alpha) \ \ f_2(\alpha) \dots f_k(\alpha)]^T \\ \bullet \ K_\alpha = [\phi^{\theta,p}(\alpha - \alpha_1) \ ... \ \phi^{\theta,p}(\alpha - \alpha_{n_s})]^T \\ \bullet \ F = \begin{bmatrix} f_1(\alpha_1) & f_2(\alpha_1) & \dots & f_k(\alpha_1) \\ \vdots & \vdots & \vdots & \vdots \\ f_1(\alpha_{n_s}) & f_2(\alpha_{n_s}) & \dots & f_k(\alpha_{n_s}) \end{bmatrix} \\ \bullet \ C = \begin{bmatrix} \phi^{\theta,p}(\alpha_1 - \alpha_1) & \phi^{\theta,p}(\alpha_2 - \alpha_1) & \dots & \phi^{\theta,p}(\alpha_{n_s} - \alpha_1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi^{\theta,p}(\alpha_1 - \alpha_{n_s}) & \phi^{\theta,p}(\alpha_2 - \alpha_{n_s}) & \dots & \phi^{\theta,p}(\alpha_{n_s} - \alpha_{n_s}) \end{bmatrix} \end{cases}$$

*l'exposant T indique le passage à la transposée

1.2.3 Méthodes de recherche locale

Les méthodes d'optimisation locale cherchent l'optimum de la fonction objectif dans le voisinage d'une valeur initiale. Ces méthodes fonctionnent dans le cas où la fonction objectif est localement convexe, elles permettent alors de trouver un minimum dans ce domaine. L'avantage des algorithmes d'optimisation locale réside dans leur rapidité. Leur inconvénient principal est leur incapacité à trouver le meilleur optimum (l'optimum global). Cette section a été rédigée principalement à partir des cours [43, 5].

Méthode du polytope de Nelder-Mead ou méthode du simplex

Cette méthode est déterministe et directe dans le sens où elle n'utilise pas d'information sur les dérivées de la fonction objectif. Elle a été introduite en 1965 par Nelder et Mead [152]. Elle a l'avantage d'être à la fois robuste et simple à programmer, et d'avoir des temps de calcul faibles. En particulier, elle est peu sensible au bruit. La fonction objectif peut donc être calculée de manière approximative au cours de l'algorithme de recherche de l'optimum. A la différence des autres algorithmes d'optimisation locale, cette méthode ne part pas d'un point de départ mais d'un polytope, c'est à dire d'une figure géométrique de $n+1$ points. Celui-ci est construit par tirage aléatoire d'un point, noté α_1, dans l'espace de recherche. Les n points restants ($i = 2, ..., n+1$) sont donnés par la relation :

$$\alpha_i = \alpha_1 + \lambda_i e_i$$

où les vecteurs e_i, $(i = 2, ..., n+1)$, forment une base de l'espace de recherche et les scalaires λ_i, $(i = 2, ..., n+1)$ sont des constantes (en général, elles sont toutes prises égales à la même valeur λ). L'algorithme de recherche du minimum procède de la manière suivante :

1. Il ordonne les points constituant le polytope de manière à avoir :

   $$J(\alpha_1) \leq J(\alpha_2) \leq ... \leq J(\alpha_{n+1})$$

   Le point α_{n+1} est donc le plus mauvais point en terme de minimisation de la fonction. Le polytope est alors noté $P =< \alpha_1, \alpha_2, ..., \alpha_{n+1} >$;
2. Il calcule le centre de gravité $\overline{\alpha}$ du polytope $P =< \alpha_1, \alpha_2, ..., \alpha_{n+1} >$;
3. Il effectue la réflexion du plus mauvais point (α_{n+1}) par rapport au centre de gravité du polytope par la relation :

   $$\alpha_r = (1+a)\overline{\alpha} - a\alpha_{n+1}$$

   où $a > 0$ est appelé coefficient de réflexion du polytope

   (a) Si $J(\alpha_r) < J(\alpha_1)$, ce point a du potentiel en terme de minimisation de la fonction objectif. L'algorithme tente une expansion dans la direction de α_r en calculant le point

   $$\alpha_e = b\alpha_r + (1-b)\overline{\alpha}$$

   où $b > 0$ est appelé coefficient d'expansion du polytope. Si $J(\alpha_e) < J(\alpha_r)$ alors le point α_e est accepté et remplace α_{n+1}. L'algorithme retourne à l'étape 1. Sinon, c'est le point α_r qui est finalement accepté, il remplace α_{n+1}, et l'algorithme retourne à l'étape 1.

   (b) Si $J(\alpha_1) \leq J(\alpha_r) < J(\alpha_n)$ alors le point α_r est accepté et remplace α_{n+1}, et l'algorithme retourne à l'étape 1.

(c) Si $J(\alpha_n) \leq J(\alpha_r) < J(\alpha_{n+1})$ alors l'algorithme tente une contraction externe en calculant le point

$$\alpha_c = c\overline{\alpha} + (1-c)\alpha_r$$

où c est le coefficient de contraction du polytope. Si $J(\alpha_c) < J(\alpha_r)$ alors le point α_c est accepté et remplace α_{n+1}. L'algorithme retourne à l'étape 1. Sinon, c'est le point α_r qui est finalement accepté, il remplace α_{n+1}, et l'algorithme retourne à l'étape 1.

(d) Si $J(\alpha_{n+1}) \leq J(\alpha_r)$, ce point est très mauvais en terme d'optimisation de la fonction objectif. L'algorithme tente une contraction interne en calculant le point

$$\alpha_c = c\overline{\alpha} + (1-c)\alpha_{n+1}$$

Si $J(\alpha_c) < J(\alpha_{n+1})$ alors le point α_c est accepté et remplace α_{n+1}. L'algorithme retourne à l'étape 1. Sinon, le polytope tout entier est rétréci. Tous les $\alpha_i, i = 2, ..., n+1$ sont alors remplacés par

$$\alpha_i = d\alpha_i + (1-d)\alpha_1$$

où d est le coefficient de rétrécissement du polytope.

L'algorithme s'arrête lorsque le déplacement du polytope d'une itération à l'autre (c'est-à-dire entre deux étapes 1 de l'algorithme ci-dessus) est suffisamment faible. En d'autres termes l'algorithme s'arrête lorsque

$$\frac{1}{n}\sum_{i=1}^{n} ||\alpha_i^{k+1} - \alpha_i^k||^2 < \epsilon$$

où α_i^{k+1} est le remplaçant du sommet α_i^k à l'itération $k+1$ et ϵ est le critère d'arrêt fixé au préalable. Les valeurs standards des constantes de réflexion (a), d'expansion (b), de contraction (c) et de rétrécissement (d) sont :

$$\begin{cases} a = 1 \\ b = 2 \\ c = 1/2 \\ d = 1/2 \end{cases}$$

Méthodes de gradient (restreintes aux cas d'optimisation sans contrainte)

Ces méthodes d'optimisation sont dites méthodes de descente. De manière générale, l'algorithme part d'un vecteur initial α_0, et tente ensuite de converger itérativement vers un optimum local en suivant une série de vecteurs définis de la manière suivante :

$$\alpha_{i+1} = \alpha_i + h_{i+1}u_{i+1}$$

où h_{i+1} et u_{i+1} représentent respectivement le pas et la direction de descente, $i+1$ l'itération en cours et i l'itération précédente. Une fois la direction de descente choisie, l'algorithme doit calculer le pas de la descente par une méthode d'optimisation monodirectionnelle. Le choix de la direction de descente et la façon de choisir le pas distinguent les différents algorithmes entre eux. Le pas h_{i+1} est qualifié d'optimal lorsque :

$$h_{i+1} = arg\min_h J(\alpha_i + hu_{i+1}).$$

En pratique, l'algorithme ne va pas jusqu'à trouver le minimum global dans la direction de descente mais se contente du premier optimum trouvé. En d'autres termes lorsque l'algorithme cherche le point qui minimise effectivement la fonction objectif dans la direction courante u_{i+1}.

- **Algorithme de gradient à pas constant**

Cet algorithme descend dans la direction opposée à celle du gradient. En effet soit le vecteur $\alpha - \nabla J(\alpha)h$ $(h \geq 0)$ situé sur la demi-droite passant par α et dirigée par l'opposé du gradient en α. Lorsque h est suffisamment faible, le développement limité de la fonction J en ce point est égal à :

$$\begin{aligned} J(\alpha - h\nabla J(\alpha)) &= J(\alpha) - (\nabla J(\alpha)|\nabla J(\alpha))\, h + O(h^2) \\ &= J(\alpha) \underbrace{-||\nabla J(\alpha)||^2 h + O(h^2)}_{\leq\ 0} \\ &\leq J(\alpha) \end{aligned}$$

Un déplacement dans la direction opposée au gradient conduit donc à une diminution de la fonction objectif. La direction du gradient est donc une bonne direction de descente lorsque l'on cherche à réduire la fonction objectif et le processus itératif choisi par les algorithmes de gradient s'écrit donc :

$$\alpha_{i+1} = \alpha_i - h_{i+1}\nabla J(\alpha_i).$$

L'algorithme de gradient à pas constant choisit le pas de la descente de manière arbitraire et constante au cours de l'algorithme de recherche du minimum ou évolutive (tendant progressivement vers zéro). Il est toutefois intéressant de remarquer que si le pas prend des valeurs trop faibles alors la convergence de l'algorithme sera lente.

- **Algorithme de gradient à pas optimal ou algorithme de plus grande descente**

Cet algorithme procède de la même manière que l'algorithme de gradient à pas constant à la différence près qu'il choisit le pas de la descente de telle sorte que la descente soit la plus efficace dans la direction locale du gradient ; en d'autres termes h_{i+1} est défini par :

$$h_{i+1} = arg \min_h J(\alpha_i - h\nabla J(\alpha_i))$$

Cet algorithme, appelé "*steepest descent*" en anglais est l'algorithme de descente le plus populaire. En pratique, ce n'est pas le minimum global dans la direction de descente qui est utilisé mais le premier minimum trouvé.

Pour appliquer cette méthode, il est indispensable de savoir trouver le pas h $(h > 0)$ minimisant la fonction $J(\alpha - hu)$, où J, α et u sont connus. Cette recherche est faite à chaque pas de l'algorithme de descente. En définissant la fonction $q(h)$ de la manière suivante :

$$q(h) = J(\alpha - hu) - J(\alpha)$$

le problème revient alors à trouver h tel que $q'(h) = 0$. Pour cela il existe plusieurs façons de procéder décrites par Vanderplaats [216].

La méthode la plus performante actuellement est la méthode de Wolfe. Le pas h est obtenu en minimisant un polynôme du troisième degré tangent au graphe de la fonction q aux points d'abscisses h^- et h^+. Des formules explicites donnent h en fonction de ces points. Cette méthode choisit deux réels m_1 et m_2 de telle sorte que $0 < m_1 < m_2 < 1$, le point h est ensuite déterminé de manière itérative selon le processus suivant (cf figure 1.3) :

- si $q(h) \leq q(h^-) + hm_1q'(h^-)$ et $m_2q'(h^-) \leq q'(h)$, alors le point h est accepté et le processus itératif de calcul de h s'arrête,

- si $q(h) \leq q(h^-) + hm_1q'(h^-)$ et $m_2q'(h^-) > q'(h)$, alors $h = h^-$,

- si $q(h) > q(h^-) + hm_1q'(h^-)$, alors $h = h^+$.

En pratique, ces méthodes de calcul du pas bien que plus précises sont beaucoup trop

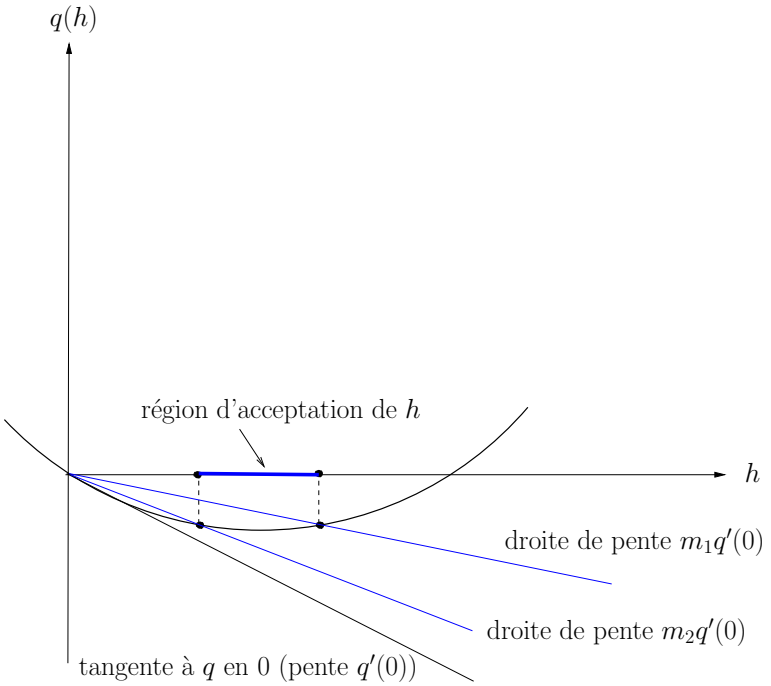

FIG. 1.3 – Construction de l'intervalle d'acceptation du pas h par la méthode de Wolfe.

coûteuses. Elles sont donc remplacées par des méthodes simplifiées. Le pas h est par exemple déterminé en évaluant la fonction J en trois points situés sur la demi-droite de direction opposée au gradient $\nabla J(\alpha)$, et en prenant le minimum de la parabole passant par ces trois points. Le pas de la descente est alors l'abscisse h de ce minimum.

- **Algorithme des gradients conjugués**

La méthode de descente des gradients conjugués n'est pas originellement une méthode d'optimisation mais une méthode de résolution des systèmes linéaires. Il s'agit d'une méthode de descente permettant de trouver le minimum en un nombre d'itérations majoré par la dimension de l'espace $\mathcal{D}$, à la différence de la méthode "steepest descent" qui peut nécessiter en théorie un nombre infini d'itérations [43].

En utilisant les résultats établis pour une fonctionnelle quadratique, l'algorithme de Polak-Ribière (1969) calcule les directions de descente successives de la manière suivante :

$$u_{i+1} = -\nabla J(\alpha_i) + \frac{||\nabla J(\alpha_i)||^2}{||\nabla J(\alpha_{i-1})||^2} u_i$$

avant d'effectuer une recherche du pas optimal dans cette direction.

Méthodes de Newton

Les méthodes suivantes sont décrites à des fins d'exhaustivité. En effet, bien qu'ayant fait l'objet d'études théoriques il y a une quinzaine d'années [198], la matrice hessienne était jusqu'à récemment inaccessible de façon numérique ou analytique pour les problèmes d'optimisation de

forme ayant un intérêt industriel. Des études récentes [159, 160] prêtent en revanche à penser que les dérivées secondes seront certainement accessibles dans les années à venir.

Comme l'optimum d'un problème d'optimisation sans contrainte annule nécessairement le gradient de la fonction objectif, l'idée des méthodes de Newton est d'utiliser une approximation quadratique de la fonction objectif et de construire une suite de points annulant le gradient de cette approximation quadratique. La forme quadratique utilisée est la suivante :

$$J(\alpha + u) \approx J(\alpha) + (\nabla J(\alpha)|u) + \frac{1}{2} u^T \nabla^2 J(\alpha) u.$$

Le gradient $g(u) = \nabla J(\alpha) + \nabla^2 J(\alpha) u$ de cette forme quadratique s'annule en

$$u = -\left[\nabla^2 J(\alpha)\right]^{-1} \nabla J(\alpha).$$

La nullité du gradient de la fonction objectif n'est cependant pas une condition nécessaire d'optimalité pour les problèmes d'optimisation contraints (cf théorème de Kuhn-Tucker). La formule ci-dessus est cependant applicable si le problème d'optimisation contraint est reformulé en un problème d'optimisation non contraint, par une méthode de pénalisation des contraintes, une méthode de gradient projeté ou une méthode lagrangienne (décrites ci-après) par exemple.

- **Algorithme classique**

L'algorithme de Newton procède donc itérativement de la manière suivante :

$$\alpha_{i+1} = \alpha_i - \left[\nabla^2 J(\alpha_i)\right]^{-1} \nabla J(\alpha_i)$$

L'inconvénient majeur de cet algorithme est qu'il requiert l'inversion d'un système linéaire associé à la matrice Hessienne de la fonction objectif. On lui préfère en général l'algorithme de quasi-Newton qui calcule de manière approchée l'inverse de la matrice Hessienne tout en gardant cependant les propriétés de convergence rapide de la méthode de Newton.

- **Algorithme de quasi-Newton**

Il s'agit des algorithmes actuellement utilisés pour la plupart des algorithmes d'optimisation sans contrainte pour trouver les minima locaux de fonctions différentiables dont on sait calculer le gradient. L'algorithme de recherche du minimum est le suivant :

1. à partir de l'état courant α_i
2. calculer le gradient $\nabla J(\alpha_i)$
3. calculer H_{i+1} une approximation de la matrice Hessienne $\left[\nabla^2 J(\alpha_i)\right]^{-1}$
4. déterminer la direction de descente $u_{i+1} = -H_{i+1} \nabla J(\alpha_i)$
5. chercher le pas optimal h_{i+1}
6. calculer l'état courant suivant $\alpha_{i+1} = \alpha_i + h_{i+1} u_{i+1}$

Pour calculer une approximation de la matrice Hessienne, l'idée est de tenir compte de l'information apportée par les calculs opérés depuis la dernière itération de l'étape 2, autrement dit de $y_i = \alpha_i - \alpha_{i-1}$ et de $\eta_i = \nabla J(\alpha_i) - \nabla J(\alpha_{i-1})$. La matrice Hessienne doit vérifier approximativement $y_i = H_{i+1} y_i$. En pratique deux approximations sont utilisées, la première est due à Davidon, Fletcher et Powell (méthode DFP) [50] et date du début des années 1960, la matrice Hessienne est approchée par :

$$H_{i+1} = H_i + \frac{1}{y_i^T \eta_i} y_i y_i^T - \frac{1}{\eta_i^T H_i \eta_i} \left(H_i \eta_i\right) \left(H_i \eta_i\right)^T,$$

la seconde due à Broyden, Fletcher, Goldfarb et Shanno (méthode BFGS) [73] est plus récente (elle date des années 1970), elle approche la matrice Hessienne de la manière suivante :

$$H_{i+1} = H_i + \left[1 + \frac{\eta_i^T H_i \eta_i}{y_i^T \eta_i}\right] \frac{1}{y_i^T \eta_i} y_i y_i^T - \frac{1}{y_i^T \eta_i} \left[y_i \left(H_i \eta_i\right)^T + H_i \eta_i y_i^T\right]$$

Algorithme du gradient à pas fixe avec projection ou algorithme du gradient projeté

Lorsque le problème d'optimisation est contraint, les algorithmes de descente selon la direction du gradient peuvent violer une ou des contraintes du problème. L'algorithme du gradient projeté fait donc appel à la projection sur l'ensemble des vecteurs admissibles (qui n'est pas un espace vectoriel en pratique). Il part d'une valeur initiale α_0 puis construit la suite des points $\alpha_{i+1} = \mathcal{P}(\alpha_i - h\nabla J(\alpha_i))$ où $\mathcal{P}$ représente la projection de $\mathbb{R}^n$ sur le sous-ensemble de $\mathbb{R}^n$ constitué des vecteurs α admissibles défini de la manière suivante :

$$\mathcal{D} \cap \{\alpha \ / \ \forall k \in [1,p] \quad G_k(\alpha) \leq 0\}.$$

Le critère d'arrêt de l'algorithme est par exemple $||\alpha_{i+1} - \alpha_i|| < \epsilon$. La difficulté essentielle de cette méthode réside dans le calcul de la projection. Celui-ci n'est en général pas élémentaire de telle sorte que la méthode est peu utilisée en pratique.

Pénalisation des contraintes

La pénalisation des contraintes permet de reformuler le problème d'optimisation avec contraintes en un problème d'optimisation sans contrainte. Toutefois, la résolution du nouveau problème d'optimisation sans contrainte (problème pénalisé) est souvent délicate, les problèmes pénalisés étant souvent "mal conditionnés". La nouvelle fonction objectif considérée par le problème pénalisé est la suivante :

$$\tilde{J}(\alpha) = J(\alpha) + \frac{1}{\epsilon} \sum_{k=1}^{p} (max(G_k(\alpha), 0))^2$$

où ϵ est un réel à fixer. Le problème pénalisé consiste alors à trouver $\alpha \in \mathbb{R}^n$ tel que $\tilde{J}(\alpha)$ soit minimale sur $\mathcal{D}$.

En notant α_ϵ^* une solution du problème pénalisé, il est possible de montrer [5] que

$$lim_{\epsilon->0} \quad \alpha_\epsilon^* = \alpha^*.$$

En d'autres termes plus ϵ est petit, plus le problème pénalisé approche bien le problème d'optimisation avec contraintes. En pratique, le choix de ϵ est crucial.

Méthode Lagrangienne

Le Lagrangien associé au problème d'optimisation avec contrainte est :

$$\mathcal{L}(\alpha, \lambda_1, ..., \lambda_p) = J(\alpha) + \sum_{k=1}^{p} \lambda_k G_k(\alpha)$$

La condition nécessaire d'optimalité de Kuhn-Tucker implique qu'à l'optimum $(\alpha^*, \lambda_1^*, ..., \lambda_p^*)$ est un point selle du Lagrangien. En effet,

$$\begin{cases} \mathcal{L}(\alpha^*, \lambda_1^*, ..., \lambda_p^*) = \min_{\alpha \in \mathbb{R}^n} \mathcal{L}(\alpha, \lambda_1^*, ..., \lambda_p^*) \\ \\ \mathcal{L}(\alpha^*, \lambda_1^*, ..., \lambda_p^*) = \max_{(\lambda_1 \geq 0, ..., \lambda_p \geq 0)} \mathcal{L}(\alpha^*, \lambda_1, ..., \lambda_p) \end{cases}$$

la deuxième inégalité provient simplement du fait que :

$$\mathcal{L}(\alpha^*, \lambda_1^*, ..., \lambda_p^*) = J(\alpha^*) \geq J(\alpha^*) + \underbrace{\sum_{k=1}^{p} \underbrace{\lambda_k}_{\geq 0} \underbrace{G_k(\alpha^*)}_{\leq 0}}_{\leq 0} = \mathcal{L}(\alpha^*, \lambda_1, ..., \lambda_p).$$

L'algorithme d'Uzawa est une méthode de recherche de point selle. A partir d'éléments quelconques $\lambda_{1,0}$, ..., $\lambda_{p,0}$, il construit des suites de points α_i, $\lambda_{1,i}$, ..., $\lambda_{p,i}$ telles que :

$$\begin{cases} \alpha_i = arg\min_{\alpha \in \mathbb{R}^n} \left(J(\alpha) + \sum_{k=1}^{p} \lambda_{k,i} G_k(\alpha) \right) & (1) \\ \\ \forall k \in [1,p], \quad \lambda_{k,i+1} = max\left(\lambda_{k,i} + \rho G_k(\alpha_i), 0\right) & (2) \end{cases}$$

où ρ est un réel strictement positif ($\rho > 0$) fixé a priori.

L'équation (1) est en fait un problème d'optimisation sans contrainte. Elle peut donc être résolue par exemple par l'un des algorithmes sans contraintes présentés précédemment (algorithme du gradient, méthode de Newton ...). Remarquons qu'en pratique ce problème d'optimisation sans contrainte est rarement résolu complétement au cours du processus itératif.

Programmation linéaire séquentielle

Cette méthode, appelée "*sequential linear programming (SLP)*" en anglais , a d'abord été introduite par Kelley [120] sous le nom de méthode du plan séquent (cutting-plane method en anglais) et par Griffith et Stewart [88] sous le nom de programmation approchée - "*method of approximate programming*" en anglais. Elle remplace la fonction objectif et les fonctions contraintes par des approximations linéaires à l'ordre 1 de ces fonctions. Elle substitue ainsi une suite de problèmes d'optimisation linéaires au problème d'optimisation non linéaire d'origine. Il s'agit d'une méthode d'optimisation locale, elle part donc d'un vecteur initial α_0. Le problème d'optimisation linéaire auquel la méthode se rapporte est le suivant :

$$\textit{trouver } \alpha \in \mathcal{D} \subset \mathbb{R}^n \textit{ tel que } \begin{cases} J(\alpha_0) + \nabla J(\alpha_0).(\alpha - \alpha_0) \textit{ est minimale} \\ \\ \forall k \in [1,p],\ G_k(\alpha_0) + \nabla G_k(\alpha_0).(\alpha - \alpha_0) \leq 0 \end{cases}$$

La solution du problème linéaire peut ne pas être admissible pour le problème non linéaire d'origine, c'est-à-dire qu'une ou plusieurs contraintes ($\forall k \in [1,p],\ G_k(\alpha) \leq 0$) peuvent ne pas être satisfaites, quand bien même l'ensemble des contraintes du problème d'optimisation linéaire sont satisfaites ($\forall k \in [1,p],\ G_k(\alpha_0) + \nabla G_k(\alpha_0).(\alpha - \alpha_0) \leq 0$). Rien n'empêche cependant l'algorithme de repartir de ce nouveau point, de chercher l'optimum du nouveau problème d'optimisation linéaire, et d'aboutir à la solution cherchée (optimum du problème d'optimisation non linéaire).

Il est parfaitement possible que la solution du problème d'optimisation linéaire soit située à l'infini, surtout si le problème est faiblement contraint. Pour éviter ce type de comportement, les algorithmes SLP introduisent souvent une borne de déplacement limitant l'amplitude entre deux vecteurs α_i et α_{i+1} au cours du processus itératif de recherche de l'optimum. La contrainte associée à cette borne de déplacement est souvent relachée à mesure de l'avancement du processus itératif (i.e. de l'augmentation progressive de la valeur de la borne). Les paramètres critiques de la méthode sont alors la valeur de la borne de déplacement et sa loi d'évolution au cours du processus itératif.

Méthode des centres

Comme la méthode précédente, cette méthode [45] se ramène à une suite de problèmes d'optimisation linéaires. Lorsque la fonction à minimiser ne varie pas de façon trop brusque, cette méthode ne travaille qu'avec des états admissibles au sens du problème d'optimisation non linéaire : la suite de vecteur α_i qu'elle construit pour approcher l'optimum est située dans la zone des états admissibles du problème d'optimisation non linéaire. Il s'agit d'une méthode d'optimisation locale : elle part d'un vecteur initial α_0 et à l'itération $i+1$ du processus itératif, le vecteur α_{i+1} est choisi comme étant le centre de la plus grande sphère incluse dans l'espace des α définis par :

$$\begin{cases} \nabla J(\alpha_i)(\alpha - \alpha_i) \leq 0 & \text{(réduction au premier ordre de la fonction objectif)} \\ \\ G_k(\alpha_i) + \nabla G_k(\alpha_i)(\alpha - \alpha_i) \leq 0 & \text{(satisfaction des contraintes au premier ordre)} \end{cases}$$

La distance du centre de la sphère (α_{i+1}) à l'hyperplan représentant la linéarisation de la fonction objectif au point α_i est [216] :

$$d = -\frac{\nabla J(\alpha_i).(\alpha_{i+1} - \alpha_i)}{|\nabla J(\alpha_i)|}$$

et la distance du centre de la sphère (α_{i+1}) à chacun des hyperplans représentant la linéarisation des contraintes au point α_i est [216] :

$$d_k = -\frac{G_k(\alpha_i) + \nabla G_k(\alpha_i).(\alpha_{i+1} - \alpha_i)}{|\nabla G_k(\alpha_i)|}.$$

L'incrément $\delta\alpha = \alpha_{i+1} - \alpha_i$ est cherché de tel sorte que le rayon (noté r) de la sphère inscrite

$$\begin{cases} r \leq d \\ r \leq d_k \ \forall k \in [1, p] \end{cases}$$

soit maximal.

A chaque itération de la méthode des centres, il faut donc trouver $\delta\alpha$ et r tels que :

$$\begin{cases} \text{r maximal} \\ \\ \nabla J(\alpha_i).\delta\alpha + |\nabla J(\alpha_i)|r \leq 0 \\ \\ \nabla G_k(\alpha_i).\delta\alpha + |\nabla G_k(\alpha_i)|r \leq -G_k(\alpha_i) \end{cases}$$

Une fois ce système d'optimisation linéaire résolu, le nouveau vecteur α_{i+1} peut être déterminé, et le processus itératif de construction des α_i peut reprendre son cours jusqu'à ce que la condition d'arrêt de la recherche soit satisfaite.

Méthode des directions admissibles

A la différence des deux méthodes précédentes, cette méthode [229, 67, 46, 47] travaille directement sur le problème d'optimisation non linéaire. Il s'agit d'une méthode de descente, elle construit donc une suite de vecteurs

$$\alpha_{i+1} = \alpha_i + h_{i+1}u_{i+1}$$

où la direction de descente u_{i+1} à l'itération $i+1$ du processus itératif est choisie de telle sorte que d'une part la fonction objectif soit réduite à l'ordre un :

$$\nabla J(\alpha_i).u_{i+1} \leq 0$$

et d'autre part que les fonctions contraintes actives, c'est-à-dire les fonctions $G_k, k \in [1,p]$, telles que $G_k(\alpha_i) = 0$, soient réduites à l'ordre un :

$$\nabla G_k(\alpha_i).u_{i+1} \leq 0.$$

Une fois la direction de descente choisie, le pas de la descente est déterminé par une méthode d'optimisation monodimensionnelle.

Pour être sûr de travailler avec des vecteurs admissibles, un coefficient de sécurité strictement positif θ_k est ajouté. La deuxième condition de calcul de la direction de descente devient donc :

$$\nabla G_k(\alpha_i).u_{i+1} + \underbrace{\theta_k}_{>0} \leq 0$$

pour toutes les fonctions contraintes actives. Lorsque les contraintes sont linéaires, il est inutile d'ajouter ce coefficient ($\theta_k = 0$). Afin de garantir une réduction optimale de la fonction objectif, un coefficient β est également introduit. Le problème à résoudre est alors

$$\text{trouver } \beta \text{ et } u_{i+1} \text{ tels que } \begin{cases} \beta \text{ est maximal} \\ \\ \nabla J(\alpha_i).u_{i+1} + \beta \leq 0 \\ \\ \nabla G_k(\alpha_i).u_{i+1} + \theta_k \leq 0 \end{cases}$$

Pour se prémunir des solutions infinies apparaissant fréquemment dans les problèmes d'optimisation linéaires, la norme de u_{i+1} est bornée.

Programmation quadratique séquentielle

Cette méthode [77, 94, 26], appelée "*sequential quadratic programming (SQP)*" en anglais, peut être vue comme une généralisation de la méthode de Newton aux problèmes d'optimisation contraints. En effet, elle remplace la fonction objectif par une approximation quadratique et les fonctions contraintes par des approximations linéaires. Elle substitue alors le problème d'optimisation ainsi construit au problème d'origine. Il s'agit d'une méthode d'optimisation locale, elle part donc d'un vecteur initial et construit une suite de vecteurs α_i tentant de converger vers l'optimum du problème d'optimisation non linéaire. A l'itération $i+1$ du processus itératif, le vecteur α_{i+1} est la solution du problème d'optimisation

$$\textit{trouver } \alpha \in \mathcal{D} \subset I\!R^n \textit{ tel que}$$

$$\begin{cases} J(\alpha_i) + \nabla J(\alpha_i).(\alpha - \alpha_i) + \frac{1}{2}(\alpha - \alpha_i)^T H(\alpha - \alpha_i) \textit{ est minimale} \\ \\ \forall k \in [1,p],\ G_k(\alpha_i) + \nabla G_k(\alpha_i).(\alpha - \alpha_i) \leq 0 \end{cases}$$

où la matrice H est une matrice définie positive approchant la matrice Hessienne ($\nabla^2 J(\alpha_i)$).

Comme pour la méthode de programmation linéaire séquentielle, cette méthode n'assure pas l'admissibilité des points de la suite de vecteurs α_i. Elle peut passer au cours du processus itératif par une série de points non admissibles. Une version de l'algorithme qui ne passe que par des états admissibles a cependant été développée [28].

1.3 L'optimisation multidisciplinaire - notations générales

Un système multidisciplinaire peut être vu comme un ensemble de boîtes noires. Chaque boîte ou sous-système représente une discipline (aérodynamique, structure, mécanique du vol, etc) ou un composant (fuselage, aile, moteur etc). Ces boîtes échangent des informations via un réseau d'entrées/sorties.

Le partitionnement du système peut être par nature hiérarchique ou non-hiérarchique. Dans un système hiérarchique, le transfert d'information se fait de manière unidirectionnelle, celui-ci va des disciplines "parent" vers les disciplines "enfant" (une discipline parent pouvant communiquer avec plusieurs disciplines enfant). Le calcul de son état d'équilibre ne nécessite pas d'itération ; il peut être effectué de façon directe. Un partitionnement hiérarchique n'est soumis à aucune limitation sur la communication entre systèmes. Dans un système non hiérarchique, il n'existe pas de notion de relation parent/enfant, les disciplines interagissent les unes avec les autres. Ce système est plus général que le précédent, son état d'équilibre est souvent résolu par un processus itératif.

De façon schématique, les méthodes de conception peuvent être classées par niveau croissant de performance et de complexité de la manière suivante :

(a) l'expérience ;

(b) les méthodes "classiques" de conception ;

(c) les méthodes d'optimisations numériques.

L'optimisation numérique d'un système multidisciplinaire - "*Multidisciplinary Optimization (MDO)*" en anglais - se distingue des méthodes "classiques" de conception de systèmes complexes par le fait qu'elle ne considère pas chaque entité de manière séquentielle mais qu'elle considère le système dans sa globalité. Celle approche présente de nombreux avantages. Plus précisément, elle permet une réduction du cycle de conception, l'utilisation d'une approche systématique, et l'exploration d'un large spectre de vecteurs α, spectre non biaisé par l'intuition puisque les vecteurs α candidats à la solution optimale sont déterminés numériquement par l'optimiseur. Elle présente cependant un certain nombre d'inconvénients, en particulier, le temps de calcul augmente rapidement en fonction de n, les résultats sont intrinséquement limités par le domaine d'application des codes d'analyse, et enfin, les algorithmes traditionnels décrits dans la section précédente ne sont pas toujours suffisants (problème de convergence, la solution trouvée n'apporte pas une amélioration suffisante).

Introduisons maintenant quelques notations qui seront utiles par la suite. Ces notations sont illustrées pour un système hiérarchique et non-hiérarchique par les figures 1.4 et 1.5.

On suppose que le système global est constitué de r sous-systèmes notés SS_i, $1 \leq i \leq r$. Les paramètres de forme se répartissent entre les différents sous-systèmes. Certains sont communs à plusieurs sous-systèmes et sont appelés **paramètres partagés ou publics**. D'autres sont propres à un sous-système et sont appelés **paramètres locaux**. Le vecteur des paramètres locaux propres au sous-système i (SS_i) est noté α_i, et le vecteur des paramètres partagés et utilisés par au moins deux sous-systèmes dont au moins le sous-système i est noté α_{pi}. D'autre part, l'ensemble des inconnues propres au sous-système i est noté de façon générale Z_i.

Les sous-systèmes constituant le système global interagissent via un ensemble d'entrées/sorties ou grandeurs couplantes notées y_{ij}, $1 \leq i, j \leq r$, où y_{ij} représentent la grandeur fournie par le sous-système i au sous-système j. Le système global étant composé de r sous-systèmes, si on suppose en outre que les grandeurs couplantes sont des scalaires, il y a au maximum $r(r-1)$ interactions possibles, donc grandeurs couplantes possibles.

La fonction objectif globale J peut-être fonction d'un certain nombre de fonctions objectifs propres à chaque sous-système. La fonction objectif propre au sous-système i est notée J_i.

L'ensemble des contraintes appliquées au système global ($G_k < 0$, $\forall\ k \in [1, p]$) représentent

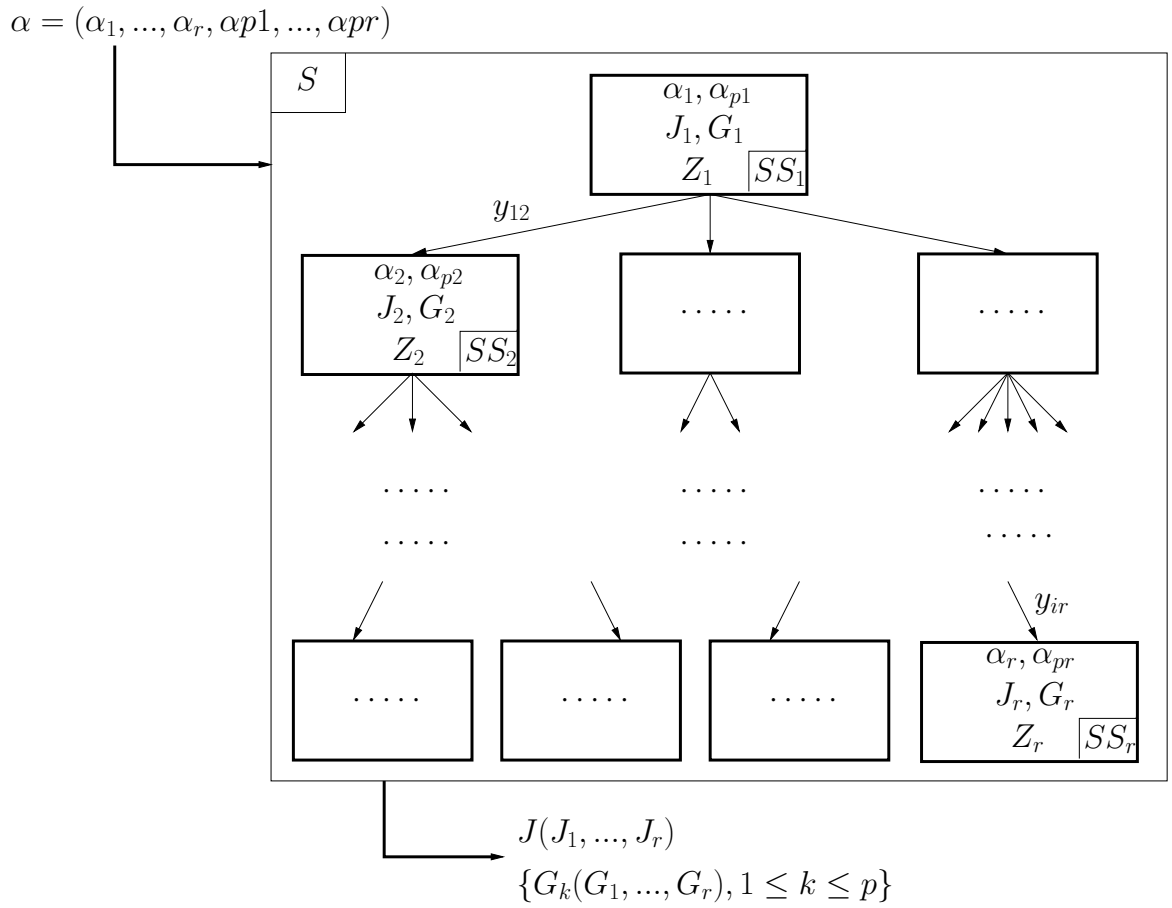

FIG. 1.4 – Système hiérarchique

l'union des contraintes propres à chacun des sous-systèmes auxquelles peuvent s'ajouter des contraintes imposées au niveau global. L'ensemble des contraintes propres au sous-système i est noté G_i.

Introduisons également quelques termes propres à l'optimisation d'un système multidisciplinaire.

D'un point de vue global, le calcul de l'état d'équilibre du système complet est appelé **analyse globale** du système et l'ensemble des programmes utilisés est appelé de façon générique **analyseur global**. Lorsque l'ensemble des inconnues du système multidisciplinaire a convergé vers un point d'équilibre, c'est à dire que les équations du système couplé sont satisfaites, on dit que le système a pris un état consistant - "**multidisciplinary feasibility**" en anglais. L'état du système multidisciplinaire peut être consistant d'un point de vue physique (analyse) sans que celui-ci soit admissible d'un point de vue conception (optimisation). En effet, certaines contraintes peuvent ne pas être satisfaites. L'ensemble des programmes permettant de chercher l'état optimal du système multidisciplinaire est appelé **optimiseur global**.

D'un point de vue local, le calcul de l'état d'équilibre d'un sous-système est appelé **analyse locale** et l'ensemble des programmes utilisés est appelé **analyseur local**. Lorsque l'ensemble des inconnues de ce sous-système a convergé vers un point d'équlibre, c'est à dire que les équations d'état du sous-système sont satisfaites, on dit que le sous-système a atteint un état consistant - "**disciplinary feasibility**" en anglais. L'ensemble des programmes permettant de chercher l'état optimal du sous-système est appelé **optimiseur local**.

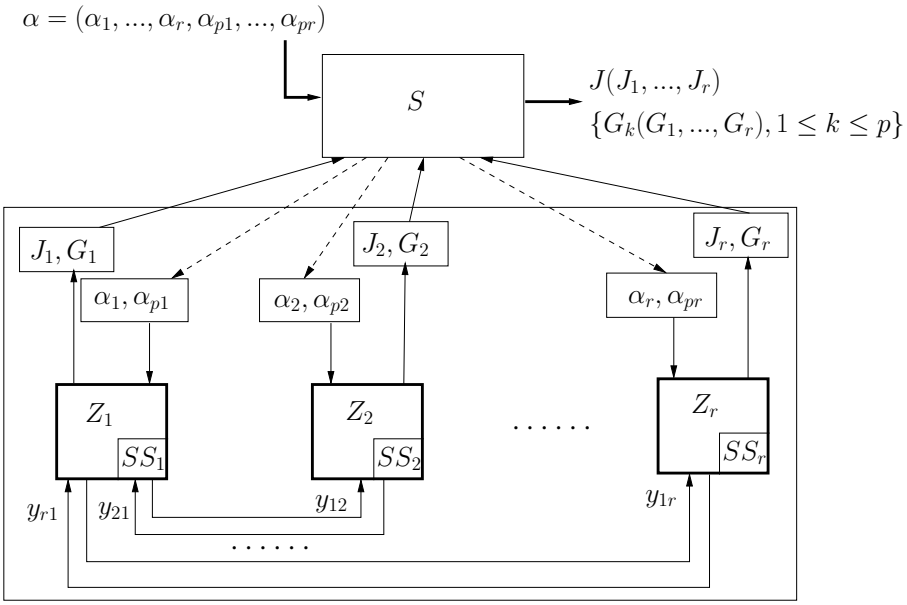

FIG. 1.5 – Système non-hiérarchique couplé

1.4 Différentes méthodologies d'optimisation

La difficulté liée au design de systèmes complexes réside dans le fait que pour avoir une conception réaliste il est nécesaire de prendre en compte un grand nombre de sous-systèmes et de détails (paramètres) de conception. Ce problème est beaucoup trop lourd en pratique pour être résolu de façon directe. Une pratique commune consiste à utiliser des résultats issus de calculs simplifiés pour l'optimisation du système global et des résultats issus de calculs haute-fidélité pour l'optimisation des sous-systèmes. Un certain nombre de méthodes MDO [208, 155, 13, 210, 111, 125, 25] ont été développées pour tenter de préserver une analyse précise tout au long du processus d'optimisation. Ces méthodes mettent en oeuvre une décomposition du problème global d'optimisation. Elles préservent le couplage des disciplines, et permettent au processus d'optimisation global d'utiliser les résultats des calculs détaillés effectués au niveau des sous-systèmes. Un certain nombre de ces méthodes, présentées dans la partie 1.4.3, ont en commun la propriété, très intéressante d'un point de vue temps de calcul, de permettre l'exécution des différentes tâches en parallèle.

1.4.1 Méthode sans couplage

La méthode de conception la plus simple consiste tout simplement à ignorer le couplage entre les disciplines. Cette technique peut donner des résultats d'une précision acceptable pour des systèmes très peu couplés, ce qui n'est pas le cas lorsque le couplage est fort. Deux approches sont couramment utilisées.

La première consiste à faire une étude paramétrique des différentes variables de conception. En d'autres termes à quantifier l'influence des α_i et α_{pi}, $1 \leq i \leq r$ sur les J et G_k,, $1 \leq k \leq p$. Puis à choisir les vecteurs α_i et α_{pi}, $1 \leq i \leq r$ influençant significativement le problème d'optimisation. Cette approche présente cependnat l'inconvénient de ne pas tenir compte des effets couplés des différentes variables entre elles.

Si le sous-système SS_1 représente la partie aérodynamique, la deuxième approche choisit α_1 et α_{p1} tels que J_1 soit minimale et que la contrainte G_1 soit satisfaite, puis détermine le reste des vecteurs α_i et α_{pi}, $2 \leq i \leq r$ tels que l'état du sous-système SS_1 (défini par α_1 et α_{p1}) soit atteint. De manière générale, cette seconde approche consiste à dessiner une forme aérodynamique

telle que les performances aérodynamiques soient optimales, puis à définir les caractéristiques de la structure de telle sorte qu'en vol et après déformation, celle-ci prenne la forme optimale prescrite par les aérodynamiciens. Cette approche est appelée approche "**Jig Shape**" en anglais. Elle présente deux inconvénients majeurs [140], d'une part la forme optimale prescrite par l'aérodynamique n'est atteinte que pour la configuration de vol considérée, d'autre part, elle n'a pas la capacité de tenir compte des couplages entre disciplines, en particulier elle ne tient pas compte de la dépendance des efforts aérodynamiques au champ de déplacement de la structure. Or, Wakayama [222, 223, 221] a montré l'importance de la prise en compte du couplage pour la conception de systèmes fortement couplés. En particulier, si la considération des effets aéroélastiques (couplage fluide-structure) a peu d'impact sur la forme de l'aile, en revanche, les effets sur la structure sont substantiels. En effet, ceux-ci ont tendance à décharger l'extrémité de la voilure, en particulier dans des conditions de bourrasque ou de manoeuvre critique pour la structure, ce qui rend possible l'utilisation de structure plus légères.

1.4.2 Méthode séquentielle

Traditionnellement, la conception des structures aéronautiques se faisaient de manière séquentielle. Cette méthode suppose que les variables de conception du système aient strictement été subdivisées et réparties entre les différents sous-systèmes, c'est-à-dire qu'il n'y ait aucun vecteur partagé α_{pi}, $1 \leq i \leq r$.

Le processus d'optimisation est le suivant : chacun des systèmes SS_i, $i \in [1, r]$, détermine le vecteur α_i qui minimise J_i et satisfait G_i, puis l'état global du système couplé correspondant à l'ensemble des α_i, $i \in [1, r]$, retenus est évalué, l'état de chacun des sous-systèmes peut donc être réévalué connaissant le nouvel état du système global, c'est-à-dire les nouvelles valeurs des y_{ij}, $1 \leq i, j \leq r$. Si les critères d'optimimalité du système S sont vérifiés, c'est-à-dire si J est minimale et les contraintes $G_k < 0$, $k \in [1, p]$, sont satisfaites, alors les vecteurs α_i, $i \in [1, r]$, sont retenus, sinon une autre itération du processus d'optimisation est réalisée.

Cette méthode a par exemple été utilisée pour la conception d'une voilure composite à flèche négative pour laquelle le problème d'optimisation consistait à minimiser le poids de la structure tout en imposant la suppression du phénomène de divergence aéroélastique propre aux ailes à flèche négative [176].

1.4.3 Méthodes à un niveau

Les méthodes d'optimisation à un niveau ne s'appuyent pas sur la décomposition du système principal S en sous-systèmes SS_i, $1 \leq i \leq r$ mais traitent celui-ci dans sa globalité (niveau unique). En d'autres termes, bien que l'analyse du système multidisciplinaire puisse être faite sur plusieurs niveaux, c'est à dire que chaque sous-système SS_i puisse avoir la charge de calculer ses inconnues et les grandeurs couplantes requises par le calcul de l'équilibre global y_{ij}, $1 \leq j \leq r$, les décisions de conception (d'optimisation) ne sont prises qu'au niveau du système global, d'où le qualificatif "à un niveau". Les formulations utilisées dans cette partie ont été introduites pour la première fois par Cramer [48] et sont maintenant utilisées classiquement par la communauté MDO.

Méthode MDF

La méthode la plus répandue, la plus simple conceptuellement mais aussi la plus coûteuse en temps de calcul, consiste à considérer le système multidisciplinaire dans sa globalité comme

un problème unique d'optimisation. Cette méthode est appelée "*Multiple Discipline Feasible (MDF)*" ou "*Nested Analysis and Design (NAND)*" ou encore "*single-NAND-NAND*" en anglais.

La recherche de l'optimum n'est orchestrée que par l'optimiseur global. Pour cela, il fait appel à l'analyseur global, qui peut lui même faire appel aux différents analyseurs locaux. Cette méthode alterne calcul d'analyse pour l'évaluation de $J(\alpha)$ et de $G_k(\alpha)$, $k \in [1,p]$ et calcul d'optimisation pour déterminer un nouveau vecteur candidat α.

De cette manière, à chaque pas du processus d'optimisation, c'est à dire à chaque vecteur α testé par le processus d'optimisation, le système multidisciplinaire est consistant : il est l'état d'équilibre associé à α. Cette méthode est donc particulièrement longue, surtout si le système est fortement couplé, c'est pourquoi elle est en général qualifiée de "force brute".

Lorsque l'optimiseur global s'appuie sur une méthode par gradient, la convergence vers un état optimal se fait de la manière suivante :

1. choix d'un vecteur initial α_0,
2. calcul de l'état d'équilibre du système associé,
3. calcul des gradients des fonctions objectif et contraintes :

$$\frac{dJ}{d\alpha}(\alpha) \quad \text{et} \quad \left\{\frac{dG_k}{d\alpha}(\alpha)\right\}_{1\leq k\leq p},$$

4. choix d'un nouveau vecteur α,
5. calcul de la fonction objectif J et des contraintes G_k, $1 \leq k \leq p$,
6. retour à l'étape 2 si la condition d'optimalité n'est pas satisfaite.

Lorsque la méthode d'analyse du système complet (2) utilise une stratégie du type point fixe - "*Fixed Point Iteration (FPI)*" en anglais - la méthode d'optimisation hérite automatiquement des défauts de celle-ci. En particulier, cette stratégie peut ne pas être convergente ou ne pas être capable de trouver tous les points fixes. De plus, elle est séquentielle par nature. Cette méthode de conception peut donc s'avérer peu robuste. Si, par exemple, l'analyseur ne parvient pas à trouver l'état d'équilibre ou s'il existe plusieurs points d'équilibre associés à un vecteur α, il est possible que l'optimiseur s'arrête sans qu'aucune solution optimale puisse être proposée. Ces imperfections ont donné naissance à d'autres formulations ayant de meilleures propriétés de convergence.

Méthode IDF

Cette méthode a été développée pour pallier aux limitations de la méthode MDF. Comme la méthode MDF, elle fait appel aux analyseurs locaux et à l'optimiseur global uniquement. L'état d'équilibre du système multidisciplinaire n'est cependant pas calculé par l'analyseur global. Cette tâche est déléguée à l'optimiseur (global). De telle sorte que les défauts inhérents à la méthode d'analyse globale ne sont pas répercutés sur la méthode d'optimisation. C'est l'optimiseur global qui fixe les valeurs prises par les vecteurs α_i, α_{pi}, $1 \leq i \leq r$ et par les grandeurs couplantes y_{ij}, $1 \leq i,j \leq r$ à chaque pas du processus d'optimisation.

Afin de garantir la consistance du système multidisciplinaire à la fin du processus d'optimisation, c'est-à-dire que l'état du système multidisciplinaire associé alors au vecteur α^* soit bien un état d'équilibre, il est nécessaire d'ajouter un ensemble de contraintes auxilliaires stipulant l'égalité entre les grandeurs couplantes évaluées par les sous-systèmes et celles fixées par l'optimiseur. Par conséquent, ces contraintes ne sont satisfaites de façon certaine qu'à la fin du processus d'optimisation. En particulier, si celui-ci est interrompu, seuls les différents sous-systèmes SS_i,

$1 \leq i \leq r$, seront consistants puisque leur état a été calculé par un analyseur local propre au sous-système. Le système S n'a a priori aucune raison d'être consistant. Pour cette raison, cette méthode est appelée "*Individual Discipline Feasible (IDF)*" ou "*Simultaneous Analysis and Design (SAND)*" ou encore "*single-SAND-NAND*" en anglais.

Elle procède de la manière suivante :

1. choix d'un vecteur α_0 initial et des grandeurs couplantes y_{ij_0}, $1 \leq i, j \leq r$, initiales,
2. calcul de l'état d'équilibre de chaque sous-système SS_i, $1 \leq i \leq r$,
3. calcul des valeurs des grandeurs couplantes $y_{ij_{\text{calculé par } SS_i}}$, $1 \leq i, j \leq r$, des fonctions objectifs et des contraintes (J_i, G_i, $1 \leq i, j \leq r$),
4. calcul de la fonction objectif J et des contraintes G_k, $1 \leq k \leq p$,
5. si la condition d'optimalité n'est pas satisfaite, choix d'un nouveau vecteur α, de nouvelles grandeurs couplantes y_{ij_0}, $1 \leq i, j \leq r$ et retour à l'étape 2.

Cette méthode permet la parallélisation des différents calculs d'analyse des sous-systèmes SS_i, $1 \leq i \leq r$, qui peuvent alors être effectués simultanément. Cette méthode a cependant un plus fort niveau de centralisation que la méthode MDF et peut par conséquent être impossible à mettre en oeuvre en pratique.

Méthode AAO

Comme les deux méthodes précédentes, cette méthode ne fait appel qu'aux analyseurs locaux et à l'optimiseur global. Cependant, les analyseurs locaux ne sont pas utilisés pour déterminer un état d'équilibre du système multidisciplinaire (méthode MDF) ou de chacun des sous-systèmes (méthode IDF). Ils ne servent qu'à évaluer le résidu des équations associées à chacun des sous-systèmes SS_i, $1 \leq i \leq r$ et ayant pour "solutions" Z_i, $1 \leq i \leq r$. C'est l'optimiseur global qui fixe les valeurs des vecteurs α, α_{pi}, $1 \leq i \leq r$, des grandeurs couplantes y_{ij}, $1 \leq i, j \leq r$ et des inconnues Z_i, $1 \leq i \leq r$ associées à chacun des sous-systèmes SS_i Z_i, $1 \leq i \leq r$.

Comme pour la méthode IDF, pour assurer la consistance du système multidisciplinaire à la fin du processus d'optimisation, il est nécessaire d'imposer un ensemble de contraintes supplémentaires stipulant d'une part l'égalité entre les grandeurs couplantes évaluées par les sous-systèmes et celles fixées par l'optimiseur (équilibre du système S) et d'autre part la nullité des résidus des équations associées à chacun des sous-systèmes SS_i, $1 \leq i \leq r$ (équilibre des sous-systèmes SS_i). Le processus d'optimisation inclut donc le calcul de l'état d'équilibre final du système multidisciplinaire, de sorte que cette méthode est appelée "*All-At-Once (AAO)*" ou "*single-SAND-SAND*" et parfois aussi "*Simultaneous Analysis and Design (SAND)*" en anglais.

Comme pour la méthode IDF, si le processus d'optimisation est interrompu, ni le système global S ni les sous-systèmes SS_i, $1 \leq i \leq r$, n'ont de raison d'être dans un état d'équilibre puisque et les grandeurs couplantes y_{ij}, $1 \leq i, j \leq r$, et les résidus des équations associées à chacun des sous-systèmes SS_i, $1 \leq i \leq r$ ont été ajoutées aux inconnues du problème d'optimisation.

Cette méthode procède de la manière suivante :

1. choix d'un vecteur α_0 initial, des grandeurs couplantes y_{ij_0}, $1 \leq i, j \leq r$, initiales et des inconnues disciplinaires Z_{i0}, $1 \leq i \leq r$, initiales,
2. calcul des résidus des équations associées à chacun des sous-systèmes SS_i, $1 \leq i \leq r$,
3. calcul des valeurs des grandeurs couplantes y_{ij_0}, $1 \leq i, j \leq r$, des fonctions objectifs et des contraintes (J_i, G_i, $1 \leq i, j \leq r$),
4. calcul de la fonction objectif J et des contraintes G_k, $1 \leq k \leq p$,

5. si la condition d'optimalité n'est pas satisfaite, choix d'un nouveau vecteur α, de nouvelles grandeurs couplantes y_{ij_0}, $1 \leq i, j \leq r$, de nouvelles inconnues disciplinaires Z_{i0}, $1 \leq i \leq r$ et retour à l'étape 2

La méthode AAO est très efficace mais peut poser des problèmes de mise en oeuvre pratique à cause de la forte centralisation des tâches. Lorsque le transfert et l'échange de données s'avère délicat, il est possible de recourir à des stratégies multiniveaux.

Conclusions sur les méthodes à un niveau

Quelque soit la méthode à un niveau utilisée, l'optimum obtenu est fonction du point d'initialisation du processus itératif et des paramètres de l'optimiseur, en particulier :

- du domaine $\mathcal{D}$ choisi ;
- du pas des différences finies lorsque certains gradients sont calculées par cette technique ;
- des coefficients de pénalités utilisés pour transformer les contraintes égalités introduites (égalité entre les grandeurs couplantes évaluées par les sous-systèmes et celles fixées par l'optimiseur pour la méthode IDF ou égalité entre les grandeurs couplantes évaluées par les sous-systèmes et celles fixées par l'optimiseur pour la méthode IDF et nullité des résidus des équations associées à chacun des sous-systèmes SS_i, $1 \leq i \leq r$ pour la méthode AAO) en contraintes inégalités ;
- du nombre d'itérations autorisé ;
- du critère de convergence du processus d'optimisation.

La solution proposée par la méthode MDF est toujours celle qui répond le mieux au problème d'optimisation. Elle a cependant le désavantage d'être beaucoup plus coûteuse, et est souvent inutilisable en pratique en particulier lorsque l'un des sous-systèmes fait appel à des calculs CFD.

La méthode AAO est la méthode la plus rapide mais elle introduit des difficultés propres liées au traitement des contraintes supplémentaires. Indépendamment de tout contexte particulier, ces remarques conduisent à penser que la méthode IDF est finalement un bon compromis puisqu'elle introduit moins de contraintes supplémentaires et que le temps de calcul requis est moins important que celui exigé par la méthode MDF.

Une comparaison pratique des trois grandes méthodes à un niveau (MDF, IDF, et AAO) a été réalisée par [200], celle-ci portait sur la conception d'un canal unidirectionnel dont les parois subissent des déformations aéroélastiques statiques. Les avantages (gain de temps des méthodes IDF et AAO par rapport à la méthode MDF) et les désavantages précédemment décrits (absence de résultats utiles si le processus d'optimisation est interrompu en cours de route), sont mis en évidence.

1.4.4 Méthodes multiniveaux

Le caractère centralisé des méthodes à un niveau, y compris des méthodes plus avancées du type IDF ou AAO, pour lesquelles l'optimiseur global a la charge de toutes les décisions de design, même les moins importantes, peut s'avérer problématique d'un point de vue industriel. Plutôt que de déléguer l'entière responsabilité de l'optimisation à l'optimiseur global, il peut être plus avantageux de faire appel aux optimiseurs locaux propres à chacune des disciplines. Cette remarque s'avère d'autant plus justifiée que le transfert d'information est délicat ou que certains sous-systèmes possèdent un optimiseur performant. Des stratégies d'optimisation faisant à la fois

appel à l'optimiseur global et aux optimiseurs locaux ont ainsi été développées. On parle de stratégies multiniveaux. Ces stratégies permettent à chaque sous-système d'opérer indépendamment, le couplage n'étant géré qu'au niveau global lors d'une phase de coordination.

La structure classique d'une stratégie multiniveaux consiste à confier à l'optimiseur global l'échange des informations entre les différents sous-systèmes, c'est-à-dire des grandeurs couplantes y_{ij}, $1 \leq i, j \leq r$, ainsi que la gestion des vecteurs α_i, α_{pi}, $1 \leq i \leq r$ L'optimiseur global a donc un rôle coordinateur.

Dans le cas particulier des structures hiérarchiques, il est possible d'utiliser la méthode Analytical Target Cascading (ATC) [6]. Celle-ci utilise la structure hiérarchique du système pour déléguer à chacun des sous-systèmes la responsabilité de son propre calcul d'analyse mais aussi de l'échange des données entre les systèmes situés en dessous de lui dans la hiérarchie.

Une structure de notation en trois termes a été mise en place par Balling [13] pour désigner de façon générique ces différentes stratégies. Le premier terme (soit single, soit multi) précise si seul l'optimiseur global est utilisé ou si les optimiseurs locaux sont également sollicités. Le second terme (soit SAND, soit NAND) décrit le le type de stratégie utilisé au niveau global. Le troisième terme (soit SAND, soit NAND) décrit le le type de stratégie utilisé au niveau local. Par exemple une stratégie multi-SAND-NAND est une stratégie d'optimisation faisant appel à la fois à l'optimiseur global et aux optimiseurs locaux, elle utilise une méthode SAND pour le processus d'optimisation global, et une méthode NAND pour les processus d'optimisation à l'échelle des sous-systèmes. Il est cependant important de remarquer que cette notation est très générale, et que deux méthodes portant le même nom peuvent recouvrir des stratégies différentes ne serait-ce que par les techniques d'optimisation ou d'analyse employées.

Méthode CSSO

La méthode CSSO - "*Concurrent Subspace Optimization*" en anglais - procède de la manière suivante : l'analyseur global calcule l'état du système couplé c'est-à-dire les inconnues Z_i, $1 \leq i \leq r$ associées à chacun des sous-systèmes SS_i, $1 \leq i \leq r$, et les grandeurs couplantes y_{ij}, $1 \leq i, j \leq r$ associées aux vecteurs α_i, α_{pi}, $1 \leq i \leq r$. Chacun des sous-systèmes SS_i, $1 \leq i \leq r$, fait ensuite appel à son optimiseur local pour trouver les vecteurs α_i et α_{pi} qui minimisent la fonction objectif J_i tout en satisfaisant les contraintes globales $G_k < 0$, $1 \leq k \leq p$. Il est possible que les sous-systèmes SS_i, $1 \leq i \leq r$, doivent fournir les gradients de la fonction objectif J_i et des contraintes globales G_k, $1 \leq k \leq p$ par rapport aux vecteurs α_i et α_{pi}. Comme tous les sous-systèmes SS_i, $1 \leq i \leq r$, travaillent en même temps et dans le même but, cette méthode est qualifiée de "concurrent" en anglais signifiant à la fois concurrente et concourante en français.

Il existe ensuite différents algorithmes de coordination utilisables par l'optimiseur global. Si les gradients

$$\frac{dJ}{d\alpha_i}(\alpha),\ \frac{dJ}{d\alpha_{pi}}(\alpha),\ \left\{\frac{dG_k}{d\alpha_i}(\alpha)\right\}_{1\leq k\leq p},\ \text{et}\ \left\{\frac{dG_k}{d\alpha_{pi}}(\alpha)\right\}_{1\leq k\leq p}, 1 \leq i \leq r,$$

sont calculés, l'optimiseur global peut chercher directement le meilleur compromis sans passer par les étapes successives, à l'intérieur de chaque sous-sytème SS_i, $1 \leq i \leq r$, d'optimisation de la fonction J par rapport aux vecteurs α_i et α_{pi} sous les contraintes $G_k < 0$, $1 \leq k \leq p$.

Si les critères de convergence vers le minimum sont satisfaits, la procédure s'arrête, sinon de nouveaux vecteurs α_i, α_{pi}, $1 \leq i \leq r$ sont choisis et une nouvelle analyse globale est effectuée. La phase de coordination peut être particulièrement délicate, cette difficulté explique le fait que cette méthode ait été peu appliquée [180, 194].

Une variante de la méthode CSSO a été appliquée à l'optimisation d'un avion de transport supersonique - "*High-Speed Commercial Transport, HSCT*" en anglais - par Baker [12]. La solution optimale fournie a été comparée à celle fournie par l'optimisation séquentielle utilisée lors

de la phase de conception précédant la commercialisation de l'avion. A la différence de la méthode CSSO, la variante employée faisait la distinction entre les paramètres globaux affectant au moins deux disciplines (ce sont par exemple la flêche, le rapport d'aspect, le ratio t/c maximal, la cambrure, le vrillage etc) et les paramètres locaux n'affectant de manière significative qu'une seule discipline (ce sont par exemple l'épaisseur de peau, la forme, l'espacement, l'aire des longerons etc). Le cas test envisagé ne considérait que deux disciplines (la mécanique des fluides et la structure) mais a cependant permis d'obtenir un dessin final meilleur que celui issu d'un processus d'optimisation séquentiel.

Méthode CO

Les méthodes multiniveaux, auxquelles se rattache la méthode CO - "*Collaborative Optimization*" en anglais - introduite par Braun et Kroo [34, 35, 127] ont remédié à cet inconvénient. Faire appel aux optimiseurs locaux permet de réduire le nombre d'évaluation des grandeurs couplantes y_{ij}, $1 \leq i, j \leq r$. Incidemment, cela permet de donner plus d'indépendance aux sous-sytèmes SS_i, $1 \leq i \leq r$ qui peuvent ainsi utiliser leurs propres techniques d'optimisation.

La méthode CO fait intervenir un nouvel ensemble d'inconnues appelées inconnues cibles - "*target variables*" en anglais - par la suite cet ensemble sera noté $t = (t_1, ..., t_r)$, où t_i correspond à l'ensemble cible associé au sous-système SS_i. L'ensemble des paramètres de conception partagé α_{pi} ainsi que les grandeurs couplantes entrantes y_{ji}, $1 \leq j \leq r$, et sortantes y_{ij}, $1 \leq j \leq r$ du sous-système SS_i sont associées à des valeurs cibles dans le vecteur t_i. Le vecteur t contrôle ainsi les interactions entre les sous-systèmes. Les variables de conception α_i , propres exclusivement au sous-système SS_i, n'ont pas de valeurs cibles dans le vecteur t_i. Ceci permet de garantir l'autonomie des sous-systèmes et n'ajoute pas de contraintes supplémentaires, à la différence des méthodes IDF et AAO. Ces variables de conception influencent cependant t_i puisque les grandeurs couplantes y_{ij}, $1 \leq i, j \leq r$, sont explicitement ou implicitement fonctions de toutes les variables de conception. A chaque étape du processus d'optimisation, l'optimiseur global associe à chacun des sous-systèmes un vecteur cible souhaité, noté $\hat{t} = \left(\hat{t_1}, ..., \hat{t_r}\right)$. Ce dernier permet de définir (pour chacun des sous-systèmes) une nouvelle fonction objectif mesurant la différence entre les valeurs cibles souhaitées par le système et celles effectivement calculées par le sous-système, par exemple par la norme L_2 de la différence :

$$\tilde{J}_i(\alpha_i, \alpha_{pi}, y_{ij}, y_{ji}) = ||t_i - \hat{t}_i||_2^2$$

Les contraintes sont subdivisées en contraintes locales propres à un sous-système particulier et fonctions uniquement de α_i, y_{ij} et y_{ji}, $1 \leq j \leq r$, et en contraintes globales fonctions de α_{pi}, $1 \leq i \leq r$, et éventuellement des inconnues Z_i, $1 \leq i \leq r$. Les contraintes globales sont écrites comme des fonctions du vecteur cible t.

Deux problèmes d'optimisation doivent alors être résolus :

1. le premier, à l'échelle locale de chacun des sous-systèmes :

$$\textit{trouver } \alpha_i \in D_i \textit{ tel que } \tilde{J}_i(\alpha_i) = ||t_i - \hat{t}_i||_2^2 \textit{ soit minimale et que } G_i(\alpha_i) < 0 \ ^*$$

2. le second, à l'échelle globale du système :

$$\textit{trouver } t \in D_t \textit{ tel que } J(t) \textit{ soit minimale et que } \begin{cases} G(t) < 0^* \\ \forall i \in [1, r], \quad \tilde{J}_i = 0^\dagger \end{cases}$$

*pour un vecteur $\vec{v}$, la notation $\vec{v} < 0$ signifie que chacune de ses composantes est strictement inférieure à 0

†cette contrainte égalité est remplacée en pratique par une contrainte inégalité, moyennant le choix d'une tolérance

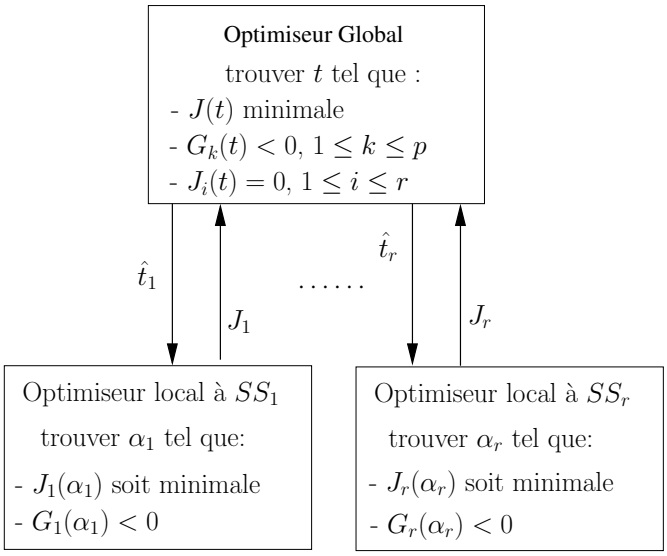

FIG. 1.6 – Organisation schématisée d'un système utilisant la méthode CO

Les échanges entre l'optimiseur global et les optimiseurs locaux s'effectuent alors de la manière suivante : l'optimiseur global détermine un vecteur cible $\hat{t}$ permettant de minimiser J et de satisfaire $G_k < 0$, $1 \leq k \leq p$, puis transmet les vecteurs t_i à chacun des sous-systèmes SS_i, $1 \leq i \leq r$. Chaque optimiseur local calcule alors le vecteur α_i lui permettant d'atteindre t_i. En d'autres termes, il cherche à minimiser l'écart entre les valeurs prises par le vecteur cible à l'intérieur du sous-système et celui requis par le système complet, tout en satisfaisant l'ensemble des contraintes attribuées au sous-système. Ces optimisations peuvent être effectuées en parallèle. A chaque étape du processus d'optimisation global, chacun des sous-systèmes doit effectuer une optimisation complète. Le système S n'atteint un état d'équilibre qu'à la fin du processus d'optimisation complet. La cohérence du système n'est pas assurée au cours du processus d'optimisation. Ce n'est qu'à la convergence de ce dernier que le système global est cohérent (par exemple, cohérence entre le chargement aérodynamique et le déplacement de la structure pour un système aéroélastique). Remarquons que les sous-systèmes ne communiquent pas directement entre eux, mais pas uniquement par l'intermédiaire de l'optimiseur global qui gère les variables couplantes y_{ij}, $1 \leq j \leq r$. La figure 1.6 illustre le mode de fonctionnement de la méthode CO.

Méthode BLISS

Tout comme les méthodes précédentes, la méthode BLISS - "*Bi-Level Integrated System Synthesis*" en anglais - se distingue de la méthodologie du "tout influence tout" adoptée par les méthodes à un niveau, et donne à chacun des sous-systèmes une grande part d'autonomie. A chaque pas du processus d'optimisation par la méthode CO, les sous-systèmes SS_i, $1 \leq i \leq r$, doivent procéder à un calcul d'analyse et d'optimisation. La méthode BLISS propose d'alléger ce processus en procédant de la façon suivante :

- calcul de l'état du système multidisciplinaire, c'est-à-dire des inconnues Z_i, $1 \leq i \leq r$ associées à chacun des sous-systèmes SS_i, $1 \leq i \leq r$, puis des grandeurs couplantes y_{ij}, $1 \leq i, j \leq r$ associées aux vecteurs α_i, α_{pi}, $1 \leq i \leq r$, ainsi que des gradients de ces grandeurs par rapport aux vecteurs α_i, α_{pi}, $1 \leq i \leq r$, par résolution du système constitué des équations du système couplé multi-disciplinaire dérivées par rapport aux vecteurs α_i, α_{pi}, $1 \leq i \leq r$ et dont les inconnues sont $\dfrac{dZ_i}{d\alpha}$, $1 \leq i \leq r$, ce système [209] est appelé

"*Global Sensitivity Equation (GSE)*" en anglais,

- pour chaque sous-système SS_i, $1 \leq i \leq r$: calcul du vecteur α_i minimisant J_i et satisfaisant G_i,
- calcul par l'optimiseur global des α_{pi}, $1 \leq i \leq r$, minimisant J et satisfaisant $G_k < 0$, $1 \leq k \leq p$, pour cette étape, l'optimiseur global est guidé par les gradients calculés précédemment.

Il est important de remarquer que dans cette méthode, les différentes disciplines ne sont pas amenées à optimiser une fonction traditionnelle du type traînée pour la mécanique des fluides ou poids pour la structure, mais une fonction mesurant l'influence de la discipline sur la fonction objectif globale.

Conclusions sur les méthodes multiniveaux

A la différence des méthodes à un niveau qui possédent un fort niveau de centralisation et qui requièrent un échange intensif d'informations entre les disciplines, les méthodes multiniveaux permettent d'utiliser le savoir-faire de chacune des disciplines tout en leur donnant une part importante d'autonomie (le calcul des variables de conception associées à la discipline est fait au niveau du sous-système et n'est pas fixé par le système global). Le principal attrait des méthodes multiniveaux, et en particulier de la méthode CO très utilisée, est donc de permettre à chacune des disciplines de garder une indépendance forte et de réduire la complexité du système couplé global. Elles sont ainsi souvent plus simples à mettre en oeuvre en pratique.

Cependant plusieurs auteurs [4, 3, 212, 81] ont mis en évidence un ensemble de contraintes pratiques liées à la structure algorithmique de ces méthodes. Les principaux défauts de ces méthodes sont liés aux difficultés que peuvent rencontrer les algorithmes d'optimisation non-linéaires à résoudre le problème d'optimisation reformulé (il est souvent profitable d'utiliser une cascade d'algorithmes d'optimisation [163] pour résoudre le problème d'optimisation associé à la méthodes multiniveaux considérée). Les méthodes multiniveaux peuvent ainsi s'avérer peu efficaces, peu robustes et succeptibles de fournir, souvent au prix d'un grand nombre d'itérations, des résultats peu fiables.

Les méthodes multiniveaux présentées ci-dessus ne sont ainsi applicables que lorsque le nombre de paramètres α_{pi}, $1 \leq i \leq r$, est faible. Si les sous-systèmes partagent un nombre élevé de paramètres de conception, les méthodes à un niveau sont les mieux adaptées.

1.4.5 Conclusions sur les méthodologies d'optimisation de systèmes couplés complexes

L'optimisation multidisciplinaire est un domaine relativement nouveau et peu connu dans l'industrie en raison des difficultés organisationnelles et techniques que sa mise en oeuvre induit (échange des valeurs à donner à chacun des paramètres de conception entre les disciplines protagonistes, calcul des variables d'état propres à chacune des disciplines et transfert des données aux autres diciplines). Elle permet de dépasser l'approche disciplinaire et de prendre en compte rigoureusement les couplages entre les disciplines lors de la conception de systèmes complexes. Elle requiert cependant l'existence d'outils d'analyse capables de prendre en compte ces interactions [226]. En pratique, la méthodologie n'est applicable que si la solution proposée est optimale et robuste et si le processus de recherche de l'optimum est automatisable (en particulier existence d'un logiciel de définition paramètrique des géométries, d'un logiciel de maillage automatique du type "*free form deformation*" [189] par exemple).

L'optimisation de systèmes complexes peut être abordée suivant deux grandes familles de méthodes, d'une part, par les méthodes à un niveau pour lesquelles, de façon schématique, le système S optimise et les sous-systèmes SS_i, $1 \leq i \leq r$, analysent, et d'autre part, par les méthodes multiniveaux pour lesquelles l'autonomie des sous-sytèmes est beaucoup plus importante (calcul des paramètres de conception associés à la discipline), les décisions d'optimisation étant toujours prises au niveau du système S. Notons que bien que toutes ces méthodes produisent une solution optimale correspondant à un équilibre du système couplé, toutes ne calculent pas l'équilibre de S au cours du processus d'optimisation. A chaque pas du processus d'optimisation, les méthodes MDF (méthode à un niveau), CSSO et BLISS (méthodes multiniveaux) calculent l'état d'équilibre de S, les méthodes IDF (méthode à un niveau) et CO (méthode multiniveaux) ne calculent que l'état d'équilibre des différents sous-systèmes SS_i, $1 \leq i \leq r$, et la méthode AAO (méthode à un niveau) ne calcule aucun état d'équilibre. Chacune de ces méthodes fait appel à un ou plusieurs optimiseurs (cf section 1.2). Elles peuvent en particulier utiliser des algorithmes d'optimisation par gradient. Seule la méthode BLISS utilise, par construction, les gradients de façon systématique. Les formulations multiniveaux restent en général plus efficaces mais moins robustes que les formulations à un niveau. Remarquons enfin que la mise en oeuvre des formulations MDO changent le mode de dialogue entre les équipes au sein d'une industrie, celui-ci doit passer du simple transfert d'information - "*data transfer*" en anglais - à la recherche de compromis - "*trade-off*" en anglais.

Les développements effectués dans le cadre de cette étude permettent de calculer les gradients d'un système couplé fluide-structure à l'équilibre aéroélastique, ces gradients étant les gradients des inconnues Z_i, $i = 1, 2$, des grandeurs couplantes y_{ij}, $1 \leq, i, j \leq 2$, et des fonctions J et G_k, $1 \leq k \leq p$ par rapport au vecteur α. Ceux-ci sont donc destinés au premier chef à être utilisés au sein d'une optimisation par gradient suivant la méthodologie MDF ou d'une optimisation multiniveaux du type BLISS de systèmes prenant en compte les interactions fluide-structure.

Chapitre 2

Calcul de la position d'équilibre aéroélastique

Ce chapitre décrit la méthode qui a été développée et utilisée pour calculer la position d'équilibre aéroélastique statique d'une structure aéronautique. Ces structures sont soumises à des efforts aérodynamiques (pression, efforts visqueux) importants. Sous cette charge, et à cause de sa nature flexible, la structure se déplace et se déforme de telle sorte que sa forme aérodynamique solide de base est modifiée. La nature flexible des structures aérodynamiques a un grand impact sur les performances des avions, sur leur manoeuvrabilité ainsi que sur le contrôle du vol. Pour effectuer un calcul réaliste des performances aérodynamiques, il est donc nécessaire de prendre ces déformations en compte, c'est-à-dire de travailler sur la forme aérodynamique après déplacement et déformation.

Le calcul de la position d'équilibre d'un système couplé fluide-structure peut être réalisé suivant deux approches. La première suppose qu'il existe un code de calcul permettant de résoudre simultanément les équations couplées (équations de la mécanique des structures et de la mécanique des fluides) sur un maillage coïncident à l'interface entre le domaine fluide et le domaine solide (surface de l'objet solide). La deuxième approche se place dans le cas où les codes de calcul sont distincts et travaillent sur des maillages non coïncidents à l'interface. Cette dernière est l'approche choisie dans la quasi-totalité des cas. En effet, les équations couplées étant résolues en faisant successivement appel à des outils monodisciplinaires, il est possible d'utiliser des schémas numériques différents et adéquats pour chacune des disciplines. Cette approche présente également l'avantage de préserver la modularité, et donc la possibilité de varier les modélisations et les méthodes de résolution. Beaucoup plus simple à mettre en oeuvre en pratique, nous avons choisi cette seconde approche dans le cadre de ce travail.

Dans ce cas, et entre autre sans l'hypothèse des petits déplacements, la simulation numérique d'un phénomène d'interaction fluide-structure nécessite un code de simulation résolvant les équations pour la dynamique des structures, un code de simulation résolvant les équations pour la dynamique des fluides, une interface de couplage des déformations, et, pour les problèmes instationnaires, un algorithme de couplage en temps. Le code de simulation associé à la structure a pour rôle de faire évoluer l'état de la structure au cours du temps (ce dernier converge vers un état constant pour les calculs stationnaires). Il doit ainsi être capable de recevoir des forces physiques qui peuvent varier au cours du temps, de calculer l'état de la structure au pas de temps suivant et de restituer les déplacements, voire les vitesses de la structure. Le code de simulation associé au fluide a pour rôle d'évaluer les efforts aérodynamiques s'exerçant sur la structure. Il doit être capable de recevoir une information relative aux déplacements et de calculer l'état du fluide inclus dans un domaine déformable (un maillage mobile) au pas de temps suivant. Pour cela la formulation ALE [58] - "*arbitrary Lagrangian-Eulerian*" en anglais - qui permet d'écrire les équations du fluide dans un repère mobile lié au point considéré peut être utilisée. L'inter-

face de couplage de l'espace a pour rôle de traduire les forces aérodynamiques calculées par le solveur fluide sur le maillage surfacique matérialisant l'interface fluide-structure pour le maillage du domaine fluide, en forces à appliquer sur le maillage matérialisant l'interface fluide-structure pour la structure. Réciproquement, cette interface de couplage de l'espace a également pour rôle de traduire le champ des déplacements (et éventuellement des vitesses) calculés par le solveur structure sur le maillage matérialisant l'interface fluide-structure du côté de la structure, en un champs de déplacements à appliquer au maillage surfacique matérialisant l'interface fluide-structure pour le maillage du domaine fluide. Ces déplacements appliqués d'abord à la structure sont ensuite propagés dans l'ensemble maillage volumique du domaine fluide. L'algorithme de couplage en temps permet d'organiser l'échange des informations dans le temps au niveau supérieur (niveau système, alors que les solveurs fluide et structure peuvent être considérés comme des sous-systèmes). C'est à son niveau que le type de couplage, fort ou faible, est choisi.

Les travaux présentés dans ce mémoire ont été effectués sur des configurations statiques correspondant aux réponses stationnaires à des conditions stationnaires. De plus, seuls les efforts d'origine aérodynamique sont pris en compte par la suite. La charge aérodynamique appliquée à la structure est donc stationnaire, ce qui est en particulier le cas en première approximation pour la phase croisière. L'hypothèse des petits déplacements a également été faite au niveau de l'interaction fluide-structure.

2.1 Description du modèle fluide-structure utilisé

L'objectif de la présente étude est de calculer les gradients nécessaires à la mise en oeuvre d'une optimisation de forme en régime de croisière réaliste dans le sens où la forme n'est pas supposée rigide mais au contraire se déforme sous la charge aérodynamique jusqu'à atteindre une position d'équilibre. Lors du processus d'optimisation, chacun des paramètres de forme choisis par l'optimiseur contrôle la forme rigide de l'objet aérodynamique considérée, c'est-à-dire la forme que cet objet prend en absence d'écoulement. Cette forme, entièrement déterminée par le vecteur des paramètres de forme choisis, est modifiée par la présence de l'écoulement fluide qui l'entoure. Elle atteint un état d'équilibre. C'est pour cet état d'équilibre que les différents gradients nécessaires au processus d'optimisation par gradient doivent être calculés si l'on veut tenir compte de la déformation aéroélastique de la forme aérodynamique à optimiser.

Cette étude s'est donc intéressée à la prise en compte des déformations aéroélastiques en phase de conception lors d'un processus d'optimisation de forme. Les gradients calculés étant destinés à une phase avant projet, il n'est pas nécessaire de les évaluer de façon très précise. Il est également légitime de supposer qu'en phase de croisière les déformations de la structure sont suffisamment faibles pour que l'on reste dans le domaine de validité d'un modèle de structure linéaire.

La théorie des poutres d'Euler-Bernoulli prédit le déplacement des ailes à forts allongements ou des pâles d'hélicoptère avec une bonne précision. Ce modèle a donc été retenu pour simuler le déplacement de la structure à l'origine de la déformation de la forme aéronautique. De plus, l'équilibre de la structure a été calculé grâce au calcul de la matrice de flexibilité associée [24]. Les déplacements de la structure sont alors reliés au chargement appliqué à la structure par la formule :

$$D = FL$$

où D représente le vecteur des déplacements des noeuds du maillage de la structure, F la matrice de flexibilité associée à la structure et L le vecteur du chargement aérodynamique (forces et moments) appliqué à la structure. L'hypothèse des petites déformations ayant été également faite - ce qui est acceptable pour les petites déformations habituellement expérimentées dans

un écoulement de type croisière - seul le maillage du domaine fluide évolue en correspondance avec le champ de déplacement prévu par la structure, le maillage de la structure sur lequel sont calculés ces déplacements demeurant fixe.

Les équations de la structure sont de façon pratique résolues par une procédure python, le modèle poutre ayant été défini comme une classe.

Le comportement du fluide peut être décrit par les équations d'Euler (non linéaires) ou les équations RANS (Reynolds Averaged Navier-Stokes). Ces équations sont résolues par le code elsA [39, 40] décrit dans la section 2.2.

Les inconnues du système aéroélastique sont le champ de déplacement de la structure noté D et calculé sur le maillage de la structure et les grandeurs conservatives aérodynamiques notées W et calculées sur le maillage du domaine fluide. Notons R_f le système d'équations discrètes non-linéaires écrites sous forme de résidu décrivant le comportement du fluide (équations d'Euler lorsque le fluide est supposé parfait et équations de Navier-Stokes moyennées (RANS) lorsque le fluide est visqueux et que l'écoulement est turbulent) :

$$R_f(W, X) = 0,$$

où X représente le maillage structuré correspondant à une discrétisation du domaine fluide au cours du processus de résolution de l'équilibre aéroélastique, et R_s le système d'équations discrètes linéaire à résoudre pour connaitre le champ de déplacement de la structure. Ce dernier, fonction de Z le maillage de la structure au cours du processus de résolution de l'équilibre aéroélastique, et D, fait en outre intervenir L le chargement aérodynamique appliqué à la structure :

$$R_s(D, Z) = D - FL = 0.$$

Ces deux systèmes d'équations sont couplés implicitement à travers les termes X et L. En effet, les coordonnées du maillage du domaine fluide X sont fonctions de D et L dépend explicitement de W et X. Le système d'équations couplées décrivant le comportement du système fluide-structure considéré est alors :

$$\begin{cases} R_f(W, X) = 0 \\ \\ R_s(D, Z) = 0 \end{cases} \quad \text{où} \quad \begin{cases} X(D, Z) \\ \\ L(W, X) \end{cases}$$

La procédure de calcul de l'équilibre aéroélastique est généralement itérative. La position d'équilibre est donc calculée en alternant calcul du chargement aérodynamique sur la forme aérodynamique courante grâce au code de simulation pour la mécanique des fluides et calcul du champ de déplacement induit sur la structure grâce au code de simulation en mécanique des structures. Entre ces deux étapes pivot du calcul de l'équilibre aéroélastique, les efforts aérodynamiques et le champ de déplacement de la structure doivent être transmis d'un code à l'autre. La figure 2.1 décrit de façon générale le schéma de calcul de la position d'équilibre aéroélastique.

2.2 Solveur fluide

Dans le cadre de cette étude, nous nous plaçons sous l'hypothèse d'un gaz parfait et le comportement du fluide peut être soit modélisé par les équations d'Euler, soit par les équations de Navier-Stokes moyennées. Ces équations font intervenir les variables suivantes :

- ρ la masse volumique du fluide ;
- V la vitesse ;
- T la température ;

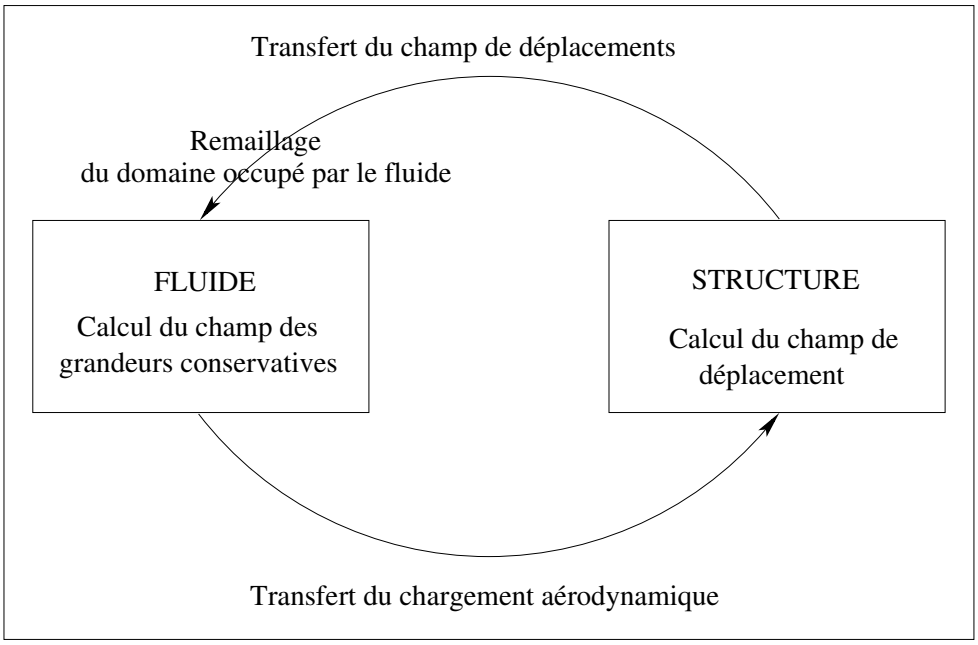

FIG. 2.1 – Principe de calcul de l'équilibre aéroélastique

- p la pression statique ;
- $H = e + \frac{p}{\rho}$ l'enthalpie totale ;
- $e = e_i + \frac{1}{2}\rho V^2$ l'énergie totale ;
- e_i l'énergie interne ;
- τ le tenseur des contraintes visqueuses ;
- s le flux de chaleur ;
- μ le coefficient de viscosité cinématique ;
- λ le coefficient de viscosité volumique ;
- K_T la conductivité thermique ;
- γ le rapport des chaleurs spécifiques du fluide (1.4 dans le cas de l'air) ;
- $R_g = 287 Jkg^{-1}K^{-1}$ la constante des gaz parfait.

Les équations d'Euler modélisent le comportement d'un fluide parfait (non visqueux et non conducteur de chaleur) compressible. Elles sont au nombre de cinq et font intervenir six inconnues.

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + div(\rho V) = 0 \\ \dfrac{\partial \rho V}{\partial t} + div(\rho V \otimes V + pI) = 0 \\ \dfrac{\partial \rho e}{\partial t} + div(\rho e V + pV) = 0 \end{cases} \tag{2.1}$$

Ces équations sont fermées par la relation d'état des gaz parfaits $p = \rho R_g T = (\gamma - 1)\rho e_i$.

Les équations de Navier-Stokes modélisent le comportement général d'un fluide visqueux,

newtonien et compressible (l'écoulement étant laminaire ou turbulent).

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + div(\rho V) = 0 \\ \dfrac{\partial \rho V}{\partial t} + div(\rho V \otimes V + pI - \tau) = 0 \\ \dfrac{\partial \rho e}{\partial t} + div(\rho e V + pV - \tau \cdot V + s) = 0 \end{cases} \tag{2.2}$$

où

$$\begin{cases} \tau &= \lambda(divV)I + 2\mu(\nabla V + \nabla V^T) \\ &= -\dfrac{2}{3}\mu(divV)I + \mu(\nabla V + \nabla V^T) \text{ avec l'hypothèse de Stokes } 3\lambda + 2\mu = 0 \\ s = -K_T \nabla T \end{cases}$$

Les lois d'état permettent d'exprimer l'énergie interne, la pression, la conductivité thermique et le coefficient de conductivité en fonction de la température :

$$\begin{cases} p = \rho R_g T = (\gamma - 1)\rho e_i \\ \mu(T) = \mu_0 \left(\dfrac{T}{T_0}\right)^{3/2} \dfrac{T_0 + C_0}{T + C_0} \end{cases}$$

où $\mu_0 = \mu(T_0)$ et $C_0 = 110.4$K.

Ces équations sont résolues, dans le cadre de ce travail, par le logiciel elsA [40] dans lequel la turbulence est modélisée en utilisant la méthode RANS ("Reynolds-Averaged Navier-Stokes") et formulée sous l'hypothèse de Boussinesq. En adoptant l'approche RANS et en utilisant la moyenne de Favre (ou moyenne pondérée par la masse), les équations de Navier-Stokes deviennent [*] :

$$\begin{cases} \dfrac{\partial \bar{\rho}}{\partial t} + div(\bar{\rho}\tilde{\bar{V}}) &= 0 \\ \dfrac{\partial \bar{\rho}\tilde{\bar{V}}}{\partial t} + div(\bar{\rho}\tilde{\bar{V}} \otimes \tilde{\bar{V}} + \bar{p}I) &= div(\tau + \tau_R) \\ \dfrac{\partial \bar{\rho}\tilde{\bar{e}}}{\partial t} + div(\bar{\rho}\tilde{\bar{e}}\tilde{\bar{V}} + \bar{p}\tilde{\bar{V}}) &= div((\tau + \tau_R)\tilde{\bar{V}} + \tilde{\bar{s}} + \tilde{\bar{s}}_t \end{cases} \tag{2.3}$$

où le tenseur des contraintes turbulentes appelé aussi tenseur de Reynolds est défini par la relation $\tau_R = -\overline{\rho V'' \otimes V''}$, et le tenseur des contraintes moyennes $\bar{\tau}$ est modélisé par la relation $\bar{\tau} = -\dfrac{2}{3}\mu div(\tilde{\bar{V}})I + \mu(\nabla\tilde{\bar{V}} + \nabla\tilde{\bar{V}}^T)$. Le flux de chaleur turbulent est égal à $\tilde{\bar{s}}_t = -\overline{pV''} - \overline{pe_i''V''} = -\lambda_t \nabla\tilde{\bar{T}}$ et le flux de chaleur laminaire à $\tilde{\bar{s}} = -\lambda\nabla\tilde{\bar{T}}$.

Deux modèles de turbulence ont été utilisés dans le cadre de ce travail : le modèle de Spalart-Allmaras - celui-ci résout une quantité proportionnelle à μ_t à l'aide d'une équation de transport

[*] $\begin{cases} \bar{q} = \lim\limits_{n\to\infty} \dfrac{1}{n}\sum\limits_{i=1}^{n} q_i(x,y,z,t) \text{ représente la moyenne statistique de la quantité } q \text{ alors égale à } \bar{q} + q' \\ \tilde{\bar{q}} = \dfrac{\overline{\rho q}}{\bar{\rho}} \text{ représente la moyenne de Favre de la quantité } q \text{ alors égale à } \tilde{\bar{q}} + q'' \end{cases}$

formulée de façon empirique - et le modèle k-ω de Wilcox - ce dernier fait intervenir deux équations supplémentaires - une pour l'énergie cinétique de la turbulence k et l'autre pour le taux spécifique de dissipation $\omega = \rho k/\mu_t$ [156].

Le logiciel elsA s'appuie sur une méthode de volumes finis avec valeurs aux centres des cellules pour la résolution des équations de Navier-Stokes. Le volume de calcul est discrétisé par des maillages structurés multiblocs et les inconnues sont les valeurs moyennes dans les cellules de discrétisation. Le schéma volumes finis s'écrit, sous sa forme semi-discrétisée en espace :

$$\frac{dW_\Omega}{dt} + \frac{1}{V(\Omega)} \left[\sum_{l=1}^{2\times dim} F_{\Gamma_l} \cdot \overrightarrow{n}_l \right] = 0 \tag{2.4}$$

où dim designe la dimension en espace, W le champ aérodynamique, $V(\Omega)$ le volume de la cellule Ω et F_{Γ_l} le flux à travers l'interface Γ_l de normale $\overrightarrow{n_l}$. Ce dernier est égal à la somme du flux convectif $F_c(W, s)$, où s désigne la vitesse par rapport à un repère absolu, et d'un flux visqueux $F_v(W, grad\ W)$. Dans le cadre de ce travail le flux convectif est résolu par le schéma décentré de Roe [185], avec correction de Harten et limiteur de Van Albada [2]. L'intégration temporelle se fait de façon implicite selon le schéma backward-Euler (ou quasi-Newton) [165] ; la matrice implicite décentrée et résolue par décomposition LU.

2.3 Solveur structure

Ce solveur est chargé d'évaluer le champ de déplacement de la structure induit par le chargement aérodynamique. En d'autres termes, il doit calculer les déplacements de la poutre utilisée pour simuler le comportement structurel de l'entité aérodynamique considérée par le processus d'optimisation de forme.

La structure est ainsi représentée par une poutre tridimensionnelle. En adoptant l'hypothèse que la structure se comporte comme une poutre d'Euler-Bernoulli, la poutre peut être réduite à une poutre unidirectionnelle dont l'axe correspond à l'axe neutre de la structure. De plus, les sections transverses, c'est-à-dire les sections perpendiculaires à l'axe neutre avant déformation, restent planes et perpendiculaires à l'axe neutre après déformation. Les forces de cisaillement perpendiculaires à l'axe neutre sont, de plus, négligées de telle sorte que les sections transverses se déplacent comme des objets solides : la déflection et la rotation de chacun des points de la section sont calculés à partir de la déflection et de la rotation de l'axe neutre par les formules de la mécanique des solides.

Le solveur structure développé et utilisé dans le cadre de ce travail doit donc évaluer les champs de déflection, noté ω, et de rotation, noté θ en chacun des points de l'axe neutre de la structure.

On suppose dans la suite de l'exposé que la poutre est constituée de n_p noeuds.

2.3.1 Approche matrice de flexibilité

Le repère associé à la structure aérodynamique est supposé défini de la manière suivante : l'axe x correspond à la direction du fuselage et pointe vers l'arrière de l'avion, l'axe y est dans le plan de l'avion et pointe vers la droite du pilote, et l'axe z correspond à la verticale ascendante. De manière générale, notons $\omega = (\omega_x, \omega_y, \omega_z)$ le champ de déplacement de flexion, $\theta = (\theta_x, \theta_y, \theta_z)$ le champ de déplacement de torsion, $(F_x, F_y.F_z)$ et (M_x, M_y, M_z) les trois composantes de la force et du moment du chargement aérodynamique agissant en un point de l'espace.

Nous notons de plus P_i , $1 \leq i \leq n_p$, les n_p noeuds constituant la géométrie de la poutre.

Comme précisé précédemment, nous nous plaçons dans le cadre de la théorie des poutres d'Euler-Bernoulli. Les déplacements de la poutre dans le plan (x, y) (mouvement du type avance-retard - "*lead lag motion*" en anglais - et mouvement d'allongement axial) sont ainsi négligés par rapport aux autres déplacements, à savoir les mouvements de flexion et de torsion de la poutre. En particulier, les sections droites restent droites et il n'y a aucune déformation due au cisaillement. Ainsi, en se référant au repère précédemment défini, seules les composantes F_z, M_x (sollicitation de la structure en flexion) et M_y (sollicitation de la structure en torsion) du torseur des efforts aérodynamiques sont prises en compte lors du calcul de l'équilibre de la poutre, et seuls les déplacements ω_z (mouvement de flexion de la poutre) et θ_y (mouvement de torsion de la poutre) sont considérés et par conséquent reportés sur le maillage du domaine fluide.

Ces déplacements sont calculés en utilisant la matrice de flexibilité ou matrice d'influence [24] - "*influence coefficient matrix*" en anglais, notée F, de la poutre. Précisément, celle-ci est définie de la manière suivante :

$$F = \begin{pmatrix} (F^{zz})_{1\leq i,j\leq n_p} & (F^{z\theta_y})_{1\leq i,j\leq n_p} & (F^{z\theta_x})_{1\leq i,j\leq n_p} \\ (F^{\theta_y z})_{1\leq i,j\leq n_p} & (F^{\theta_y\theta_y})_{1\leq i,j\leq n_p} & (F^{\theta_y\theta_x})_{1\leq i,j\leq n_p} \end{pmatrix}$$

Elle est donc de dimension $(2n_p * 3n_p)$ et ses coefficients correspondent physiquement au déplacement induit en chacun des points de la poutre par un effort unitaire en chacun des autres points de la poutre. D'après les hypothèses faites précédemment, seuls les efforts en flexion et en torsion sont pris en compte, de même, seuls les déplacement en flexion et en torsion sont retenus. De façon plus précise :

- le coefficient ${F_{ij}}^{zz}$ représente le déplacement vertical ω_z engendré au point P_i par un effort vertical F_z unitaire au point P_j ;
- le coefficient ${F_{ij}}^{z\theta_y}$ représente le déplacement vertical ω_z engendré au point P_i par un moment M_y unitaire au point P_j ;
- le coefficient ${F_{ij}}^{z\theta_y}$ représente le déplacement vertical ω_z engendré au point P_i par un moment M_x unitaire au point P_j ;
- le coefficient ${F_{ij}}^{\theta_y z}$ représente le déplacement angulaire θ_y engendré au point P_i par un effort vertical F_z unitaire au point P_j ;
- le coefficient ${F_{ij}}^{\theta_y\theta_y}$ représente le déplacement angulaire θ_y engendré au point P_i par un moment M_y unitaire au point P_j ;
- le coefficient ${F_{ij}}^{\theta_y\theta_x}$ représente le déplacement angulaire θ_y engendré au point P_i par un moment M_x unitaire au point P_j.

Introduisons le déplacement angulaire θ_x induit par la flexion ($\theta_x = \omega_z'$) dans la définition de la matrice de flexibilité :

$$F_{\text{généralisée}} = \begin{pmatrix} (F^{zz})_{1\leq i,j\leq n_p} & (F^{z\theta_y})_{1\leq i,j\leq n_p} & (F^{z\theta_x})_{1\leq i,j\leq n_p} \\ (F^{\theta_y z})_{1\leq i,j\leq n_p} & (F^{\theta_y\theta_y})_{1\leq i,j\leq n_p} & (F^{\theta_y\theta_x})_{1\leq i,j\leq n_p} \\ (F^{\theta_x z})_{1\leq i,j\leq n_p} & (F^{\theta_x\theta_y})_{1\leq i,j\leq n_p} & (F^{\theta_x\theta_x})_{1\leq i,j\leq n_p} \end{pmatrix}$$

Cette nouvelle matrice $F_{\text{généralisée}}$ est alors carrée. On peut ainsi tirer profit de sa propriété de symétrie (montrée par exemple dans [24]) pour alléger le calcul de la matrice F, alors considérée comme une sous matrice de la matrice carrée $F_{\text{généralisée}}$.

Ainsi,

- seules les parties supérieures droites des matrices $(F^{zz})_{1\leq i,j\leq n_p}$ et $(F^{\theta_y\theta_y})_{1\leq i,j\leq n_p}$ sont effectivement calculées (les parties inférieures gauches étant déduites par symétrie),
- seule une des matrices $(F^{z\theta_y})_{1\leq i,j\leq n_p}$ et $(F^{\theta_y z})_{1\leq i,j\leq n_p}$ devra être calculée (ces deux matrices étant identiques),
- en revanche, les matrices $(F^{z\theta_x})_{1\leq i,j\leq n_p}$ et $(F^{\theta_y\theta_x})_{1\leq i,j\leq n_p}$ devront être entièrement calculées.

2.3.2 Calcul de la matrice de flexibilité associée à une poutre rectiligne

Nous supposons dans cette partie que la structure, par exemple une aile droite sans cassure ni dièdre, peut être représentée par une poutre d'Euler-Bernoulli rectiligne et alignée avec l'axe y du repère précédemment défini. Dans ce cas, les mouvements de flexion et de torsion sont découplés, les termes $F^{z\theta_y}$, $F^{\theta_y z}$ et $F^{\theta_y\theta_x}$ sont donc nuls.

Notons :

- E le module d'Young du matériau ;
- I le moment quadratique d'une section droite de la poutre autour de l'axe de flexion ($I = \int_{\text{section}}[dist(P, \text{axe de flexion})]^2 dS$) ;
- G le module d'élasticité de glissement ;
- J le moment quadratique d'une section droite de la poutre autour de son axe principal ($J = \int_{\text{section}}[dist(P, \text{axe principal})]^2 dS$) ;
- s la coordonnée suivant l'axe principal d'un point de la poutre auquel on évalue les déplacements ;
- η la coordonnée suivant l'axe principal d'un point de la poutre auquel on applique les efforts ;
- s_i la coordonnée suivant l'axe principal du point P_i auquel on évalue les déplacements ;
- s_j la coordonnée suivant l'axe principal du point P_j auquel on applique les efforts ;
- λ variable muette servant d'intégration des déplacements ;
- $M(s)$ le moment de flexion au point de coordonnée s ;
- $T(s)$ le moment de torsion au point de coordonnée s.

Calcul de F^{zz}

Le déplacement induit au point d'abscisse curviligne s d'une poutre soumise à un effort vertical unitaire au point d'abscisse η (chargement en flexion) satisfait l'équation suivante :

$$\begin{cases} \text{si } s \leq \eta & EI(s)\dfrac{\partial^2\omega_z}{\partial s^2}(s,\eta) = EI(s)\dfrac{\partial\theta_x}{\partial s}(s,\eta) = M(s) = \eta - s \\ \\ \text{si } s > \eta & \dfrac{\partial^2\omega_z}{\partial s^2}(s,\eta) = \dfrac{\partial\theta_x}{\partial s}(s,\eta) = 0 \end{cases}$$

et vérifie les conditions aux limites suivantes au point d'encastrement avec le fuselage (situé en $s = 0$)

$$\begin{cases} \omega_z(0,\eta) = 0 \\ \\ \dfrac{\partial\omega_z}{\partial s}(0,\eta) = \theta_x(0,\eta) = 0 \end{cases}$$

L'intégration du système ci-dessus (sous sa forme continue) donne :

$$\begin{cases} \text{si } s \leq \eta \quad \theta_x(s,\eta) = \displaystyle\int_0^s \frac{\eta-\lambda}{EI(\lambda)} d\lambda \\ \\ \text{si } s > \eta \quad \theta_x(s,\eta) = \theta_x(\eta,\eta) = \displaystyle\int_0^\eta \frac{\eta-\lambda}{EI(\lambda)} d\lambda \end{cases}$$

$$\Longrightarrow \begin{cases} \text{si } s \leq \eta \quad F^{zz}(s,\eta) = \omega_z(s,\eta) = \displaystyle\int_0^s \theta_x(\alpha,\eta) d\alpha = \int_0^s \int_0^\alpha \frac{\eta-\lambda}{EI(\lambda)} d\lambda d\alpha \\ \\ \text{si } s > \eta \quad F^{zz}(s,\eta) = \omega_z(s,\eta) = \displaystyle\int_\eta^s \theta_x(\alpha,\eta) d\alpha + \omega_z(\eta,\eta) \\ \\ \qquad\qquad\qquad\qquad\quad = (s-\eta)\displaystyle\int_0^\eta \frac{\eta-\lambda}{EI(\lambda)} d\lambda + F^{zz}(\eta,\eta) \end{cases}$$

Soit sous forme discrète ($s_0 = 0$) :

$$\begin{cases} \text{si } i \leq j \qquad \theta_x(s_i) = \displaystyle\sum_{k=1}^{i} \frac{1}{2}\left(\frac{s_j - s_k}{EI(s_k)} + \frac{s_j - s_{k-1}}{EI(s_{k-1})}\right) \\ \\ \qquad\qquad\qquad\quad = \theta_x(s_{i-1}) + \displaystyle\frac{1}{2}\left(\frac{s_j - s_i}{EI(s_i)} + \frac{s_j - s_{i-1}}{EI(s_{i-1})}\right) \\ \\ \text{si } i \geq j+1 \quad \theta_x(s_i) = \theta_x(s_j) \end{cases}$$

$$\Longrightarrow \begin{cases} \text{si } i \leq j \qquad F_{ij}^{zz} = \omega_z(s_i) = \displaystyle\sum_{l=1}^{i} \frac{1}{2}(s_l - s_{l-1})(\theta_x(s_l) + \theta_x(s_{l-1})) \\ \\ \qquad\qquad\qquad\quad\; = F_{i-1j}^{zz} + \displaystyle\frac{1}{2}(s_i - s_{i-1})(\theta_x(s_i) + \theta_x(s_{i-1})) \\ \\ \text{si } i \geq j+1 \quad F_{ij}^{zz} = \omega_z(s_i) = F_{jj}^{zz} + (s_j - s_i)\theta_x(s_j) \end{cases}$$

Etant données les propriétés de symétrie de $\mathcal{F}$, à j fixé ($1 \leq j \leq n_p$) seuls les termes $\theta_x(s_i)$ et F_{ij}^{zz}, $1 \leq i \leq j$, sont calculés.

Calcul de $F^{z\theta_x}$

Le déplacement induit au point d'abscisse curviligne s d'une poutre soumise à un couple unitaire selon l'axe x au point d'abscisse η (chargement en flexion) satisfait l'équation suivante :

$$\begin{cases} \text{si } s \leq \eta \quad EI(s)\dfrac{\partial^2 \omega_z}{\partial s^2}(s,\eta) = EI(s)\dfrac{\partial \theta_x}{\partial s}(s,\eta) = M(s) = M(\eta) = 1 \\ \\ \text{si } s > \eta \quad \dfrac{\partial^2 \omega_z}{\partial s^2}(s,\eta) = \dfrac{\partial \theta_x}{\partial s}(s,\eta) = 0 \end{cases}$$

et vérifie les conditions aux limites suivantes au point d'encastrement avec le fuselage (situé en $s = 0$)

$$\begin{cases} \omega_z(0,\eta) = 0 \\ \\ \dfrac{\partial \omega_z}{\partial s}(0,\eta) = \theta_x(0,\eta) = 0 \end{cases}$$

L'intégration du système ci-dessus (sous sa forme continue) donne :

$$\begin{cases} \text{si } s \leq \eta \quad \theta_x(s,\eta) = \displaystyle\int_0^s \frac{1}{EI(\lambda)} d\lambda \\ \\ \text{si } s > \eta \quad \theta_x(s,\eta) = \theta_x(\eta,\eta) = \displaystyle\int_0^\eta \frac{1}{EI(\lambda)} d\lambda \end{cases}$$

$$\Longrightarrow \begin{cases} \text{si } s \leq \eta \quad F^{z\theta_x}(s,\eta) = \omega_z(s,\eta) = \displaystyle\int_0^s \theta_x(\alpha,\eta) d\alpha = \int_0^s \int_0^\alpha \frac{1}{EI(\lambda)} d\lambda d\alpha \\ \\ \text{si } s > \eta \quad F^{z\theta_x}(s,\eta) = \omega_z(s,\eta) = \displaystyle\int_\eta^s \theta_x(\alpha,\eta) d\alpha + w_z(\eta,\eta) \\ \\ \qquad\qquad\qquad\qquad\qquad\quad = (s-\eta)\displaystyle\int_0^\eta \frac{1}{EI(\lambda)} d\lambda + F^{z\theta_x}(\eta,\eta) \end{cases}$$

Soit sous forme discrète ($s_0 = 0$) :

$$\begin{cases} \text{si } i \leq j \qquad \theta_x(s_i) = \displaystyle\sum_{k=1}^{i} \frac{1}{2}(s_k - s_{k-1}) \left(\frac{1}{EI(s_k)} + \frac{1}{EI(s_{k-1})} \right) \\ \\ \qquad\qquad\qquad\quad = \theta_x(s_{i-1}) + \displaystyle\frac{1}{2}(s_i - s_{i-1}) \left(\frac{1}{EI(s_i)} + \frac{1}{EI(s_{i-1})} \right) \\ \\ \text{si } i \geq j+1 \quad \theta_x(s_i) = \theta_x(s_j) \end{cases}$$

$$\Longrightarrow \begin{cases} \text{si } i \leq j \qquad F_{ij}^{z\theta_x} = \omega_z(s_i) = \displaystyle\sum_{l=1}^{i} \frac{1}{2}(s_l - s_{l-1})(\theta_x(s_l) + \theta_x(s_{l-1})) \\ \\ \qquad\qquad\qquad\qquad\quad = F_{i-1j}^{z\theta_x} + \displaystyle\frac{1}{2}(s_i - s_{i-1})(\theta_x(s_i) + \theta_x(s_{i-1})) \\ \\ \text{si } i \geq j+1 \quad F_{ij}^{z\theta_x} = \omega_z(s_i) = F_{jj}^{z\theta_x} + (s_j - s_i)\theta_x(s_j) \end{cases}$$

Calcul de $F^{\theta_y\theta_y}$

Le déplacement induit au point d'abscisse curviligne s d'une poutre soumise à un couple unitaire selon l'axe y au point d'abscisse η (chargement en torsion) satisfait l'équation suivante :

$$\begin{cases} \text{si } s \leq \eta \quad GJ(s)\dfrac{\partial \theta_y}{\partial s}(s,\eta) = T(s) = T(\eta) = 1 \\ \\ \text{si } s > \eta \quad \dfrac{\partial \theta_y}{\partial s}(s,\eta) = 0 \end{cases}$$

et vérifie les conditions aux limites suivantes au point d'encastrement avec le fuselage (situé en $s = 0$)

$$\theta_y(0,\eta) = 0$$

L'intégration du système ci-dessus (sous sa forme continue) donne :

$$\begin{cases} \text{si } s \leq \eta \quad F^{\theta_y\theta_y}(s,\eta) = \theta_y(s,\eta) = \displaystyle\int_0^s \frac{d\lambda}{GJ(\lambda)} \\ \\ \text{si } s > \eta \quad F^{\theta_y\theta_y}(s,\eta) = \theta_y(s,\eta) = F^{\theta_y\theta_y}(\eta,\eta) \end{cases}$$

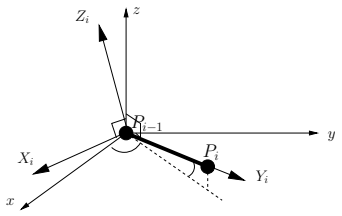

FIG. 2.2 – Repère local associé à la section $P_{i-1}P_i$ de la poutre

Soit sous forme discrète ($s_0 = 0$) :

$$\begin{cases} \text{si } i \leq j \qquad F_{ij}^{\theta_y\theta_y} = \theta_y(s_i, s_j) = \displaystyle\sum_{k=1}^{i} \frac{1}{2}(s_k - s_{k-1}) \left(\frac{1}{GJ(s_k)} + \frac{1}{GJ(s_{k-1})} \right) \\ \qquad\qquad\qquad\qquad\qquad\quad = F_{i-1j}^{\theta_y\theta_y} + \dfrac{1}{2}(s_i - s_{i-1}) \left(\dfrac{1}{GJ(s_i)} + \dfrac{1}{GJ(s_{i-1})} \right) \\ \text{si } i \geq j+1 \quad F_{ij}^{\theta_y\theta_y} = \theta_y(s_i, s_j) = F_{jj}^{\theta_y\theta_y} \end{cases}$$

Etant données les propriétés de symétrie de $\mathcal{F}$, à j fixé ($1 \leq j \leq n_p$) seuls les termes $F_{ij}^{\theta_y\theta_y}$, $1 \leq i \leq j$, sont calculés.

Que la poutre soit soumise à une force ou un moment de flexion ou à un moment de torsion, le déplacement de ses noeuds peut être calculé de manière itérative en partant de l'emplanture de l'aile (point d'encastrement de la poutre) et en allant vers le saumon.

2.3.3 Calcul de la matrice de flexibilité associée à une poutre rectiligne par morceaux quelconque

Lorsque la flèche ou le dièdre de la voilure ne sont pas nuls, la structure ne peut plus être assimilée à une poutre rectiligne. La géométrie de la poutre n'a donc plus lieu de suivre l'axe y. De plus si l'aile possède des points de cassures, la géométrie ne sera plus rectiligne mais rectiligne par morceaux. Les mouvements de flexion et de torsion sont alors couplés.

Toutefois, il est possible de se ramener localement au cas de la poutre rectiligne. Pour ce faire définissons le repère local (X_i, Y_i, Z_i) associé à une section $P_{i-1}P_i$ de la manière suivante : l'axe Y_i correspond à la direction locale de la poutre $P_{i-1}P_i$, l'axe Z_i est l'axe le plus proche de la direction verticale ascendante z, et l'axe X_i complète le trièdre direct (X_i, Y_i, Z_i). Par définition, ce repère varie d'une section à l'autre.

Dans ce cas, la matrice de flexibilité est calculée en se ramenant au cas de la poutre rectiligne présenté dans la sous-section précédente. Pour ce faire, les efforts appliqués à la poutre (flexion et torsion) et exprimés dans le repère global sont projetés dans le repère local associé au point auquel les déplacements en torsion et en flexion sont recherchés. Une fois que le déplacement induit en ce point est calculé dans le repère local, celui-ci est projeté à nouveau dans le repère global et seules les composantes du déplacement de flexion et de torsion sont conservées pour former la matrice de flexibilité.

Que l'effort appliqué à la poutre soit un effort de flexion F_z ou un couple de torsion M_y ou ou un couple de flexion M_x, les coefficients d'influence sont calculés itérativement (de proche en proche) en allant de l'emplanture (condition d'encastrement) de l'aile au saumon. Afin d'expliquer ce calcul, supposons que les déplacements de la poutre aient été calculés jusqu'au point P_i et décrivons alors le calcul du déplacement du point P_{i+1}.

L'effort appliqué au point P_j induit un moment non nul en chaque point de la poutre situé en

amont du point P_j ($i \leq j$). Cet effort peut être un effort unitaire F_z si l'objectif est de calculer F^{zz} ou $F^{\theta_y z}$, un couple unitaire M_y si l'objectif est de calculé $F^{z\theta_y}$ ou $F^{\theta_y\theta_y}$, ou un couple unitaire M_y si l'objectif est de calculer $F^{z\theta_x}$ ou $F^{\theta_y\theta_x}$.

Décrivons le calcul des coefficients d'influence en nous plaçant dans le cas général. Les cas où le chargement n'est composé que d'un effort de flexion ou que d'un couple de flexion ou que d'un couple de torsion sont alors des cas particuliers. Les champs de déplacements correspondants peuvent alors être déduits du déplacement général en annulant les autres sources de chargement dans la formule générale. Dans le cas général le tenseur des efforts appliqués en un point P_j de la poutre et considérés par le calcul de la position d'équilibre est de la forme :

$$\underbrace{\begin{pmatrix} 0 \\ 0 \\ F_z \end{pmatrix}}_{\text{effort}} \quad \underbrace{\begin{pmatrix} M_x \\ M_y \\ 0 \end{pmatrix}}_{\text{moment}}$$

Le moment appliqué au point P_{i+1} de la poutre est alors :

$$\left\{ \begin{array}{rcl} \overrightarrow{\mathcal{M}}(P_{i+1}) & = & \overrightarrow{\mathcal{M}}(P_j) + \overrightarrow{P_{i+1}P_j} \wedge F_z\vec{z} \\ & = & \mathcal{P}_{i+1}\left(M_x\vec{x} + M_y\vec{y} + \overrightarrow{P_{i+1}P_j} \wedge F_z\vec{z}\right) \\ & = & M_{X_{i+1}}\vec{X}_{i+1} + M_{Y_{i+1}}\vec{Y}_{i+1} + M_{Z_{i+1}}\vec{Z}_{i+1} \end{array} \right.$$

où $\mathcal{P}_{i+1}$ est la matrice de passage du repère (x, y, z) au repère $(X_{i+1}, Y_{i+1}, Z_{i+1})$. Ce moment se décompose donc en un moment de flexion $M_{X_{i+1}}$ (celui-ci dépend du bras de levier P_iP_j si cet effort est F_z et n'en dépend pas s'il s'agit de M_x (couple)), en un moment de torsion $M_{Y_{i+1}}$ (il s'agit nécessairement d'un couple puisque seul M_y génère de la torsion) et un moment $M_{Z_{i+1}}$ non pris en compte.

Il est alors possible de calculer les déplacements $\omega_{Z_{i+1}}$, $\theta_{X_{i+1}}$ et $\theta_{Y_{i+1}}$ en utilisant les équations dérivées précédemment pour une poutre droite rectiligne (l'axe y de la sous-section précédente est alors l'axe Y_{i+1}). Plus précisément, ces déplacements sont calculés en appliquant les formules suivantes :

- si le point P_{i+1} se situe en amont du point P_j, c'est-à-dire que $i+1 \leq j$:

$$\left\{ \begin{array}{l} (\theta_{X_{i+1}})_{i+1} = (\theta_{X_{i+1}})_i + \dfrac{1}{2}\left(\dfrac{(M_{X_{i+1}})_{i+1}}{EI_{i+1}} + \dfrac{(M_{X_{i+1}})_i}{EI_i}\right) \\ \\ (\omega_{Z_{i+1}})_{i+1} = (\omega_{Z_{i+1}})_i + \dfrac{1}{2}||P_{i+1}P_i||((\theta_{X_{i+1}})_{i+1} + (\theta_{X_{i+1}})_i) \\ \\ (\theta_{Y_{i+1}})_{i+1} = (\theta_{Y_{i+1}})_i + \dfrac{1}{2}\left(\dfrac{(M_{Y_{i+1}})_{i+1}}{GJ_{i+1}} + \dfrac{(M_{Y_{i+1}})_i}{GJ_i}\right) \end{array} \right.$$

- si le point P_{i+1} se situe strictement en aval du point P_j, c'est-à-dire que $i+1 > j$:

$$\left\{ \begin{array}{l} (\theta_{X_{i+1}})_{i+1} = (\theta_{X_{i+1}})_j \\ \\ (\omega_{Z_{i+1}})_{i+1} = (\omega_{Z_{i+1}})_i - ||P_{i+1}P_i||(\theta_{X_{i+1}})_j \\ \\ (\theta_{Y_{i+1}})_{i+1} = (\theta_{Y_{i+1}})_j \end{array} \right.$$

où $(\theta_{X_{i+1}})_{i+1}$ et $(\theta_{X_{i+1}})_i$ représentent respectivement la rotation induite par rapport à l'axe X_{i+1} aux points P_{i+1} et P_i, $(\theta_{Y_{i+1}})_{i+1}$ et $(\theta_{Y_{i+1}})_i$ représentent respectivement la rotation induite par

rapport à l'axe Y_{i+1} aux points P_{i+1} et P_i, et $(\omega_{Z_{i+1}})_{i+1}$ et $(\omega_{Z_{i+1}})_i$ représentent respectivement la translation induite suivant l'axe Z_{i+1} aux points P_{i+1} et P_i.

Les déplacements $(\theta_{X_{i+1}})_i$, $(\omega_{Z_{i+1}})_i$ et $(\theta_{Y_{i+1}})_i$ sont déduits de l'étape itérative précédente : ils ont été calculés dans le repère local (X_i, Y_i, Z_i) de l'élément de poutre $P_{i-1}P_i$ puis projetés dans le repère aérodynamique global (x, y, z), et sont projetés à nouveau dans le repère local courant $(X_{i+1}, Y_{i+1}, Z_{i+1})$ associé à l'élément de poutre P_iP_{i+1}.

Les déplacements $(\theta_{X_{i+1}})_{i+1}$, $(\omega_{Z_{i+1}})_{i+1}$ et $(\theta_{Y_{i+1}})_{i+1}$ sont ensuite projetés dans le repère global lié à l'avion (x, y, z), et seules les composantes ω_z et θ_y sont conservées pour former la matrice de flexibilité F.

2.3.4 Calcul des déplacements induits par le chargement aérodynamique

Une fois la matrice de flexibilité construite, il est simple de déduire des efforts appliqués, le vecteur des déplacements D de l'ensemble des points de la poutre. Celui-ci est alors simplement calculé par la formule $D = FL$, soit sous forme développée :

$$\begin{pmatrix} (\omega_p)_{1\leq i\leq n_p} \\ \\ (\theta_p)_{1\leq i\leq n_p} \end{pmatrix} = \begin{pmatrix} (F^{zz})_{1\leq i,j\leq n_p} & (F^{z\theta_y})_{1\leq i,j\leq n_p} & (F^{z\theta_x})_{1\leq i,j\leq n_p} \\ \\ (F^{\theta_y z})_{1\leq i,j\leq n_p} & (F^{\theta_y\theta_y})_{1\leq i,j\leq n_p} & (F^{\theta_y\theta_x})_{1\leq i,j\leq n_p} \end{pmatrix} \begin{pmatrix} (F_z)_{1\leq j\leq n_p} \\ \\ (M_y)_{1\leq j\leq n_p} \\ \\ (M_x)_{1\leq j\leq n_p} \end{pmatrix}$$

où $(\omega_p)_{1\leq i\leq n_p}$ et $(\theta_p)_{1\leq i\leq n_p}$ représentent respectivement les champs de déplacement en flexion, selon l'axe vertical z du repère global, et en torsion, autour de la direction moyenne de l'aile représentée par l'axe y du repère global dans notre cas, en chacun des points de la poutre. Notons que D, F, et L ont respectivement pour dimensions $(2n_p)$, $(2n_p * 3n_p)$ et $(3n_p)$.

2.4 Méthodes de couplage

On se place dans le cas général où les codes de calcul résolvant les équations de la mécanique des fluides et de la mécanique des structures sont distincts et de nature différente (formulation Eulérienne et volume finis pour le fluide, et formulation Lagrangienne et matrice de flexibilité pour la structure) et où les maillages de l'interface physique entre le domaine fluide et la structure, côté fluide d'une part et côté structure d'autre part, ne coïncident pas.

Résoudre le système couplé fluide-structure revient à résoudre les équations décrivant l'état du fluide et l'état de la structure tout en imposant un ensemble de conditions aux limites cohérentes. Apparait alors le problème du transfert de l'information entre les deux sous-systèmes. En d'autres termes, pour résoudre un tel problème il faut être capable de transmettre les efforts (d'origine aérodynamique dans notre cas) au code de simulation résolvant les équations de la structure et réciproquement, il faut être capable de transmettre les champs de vitesse et de déplacement de la paroi solide matérialisant l'interface fluide-structure au code de simulation décrivant l'état du fluide. Comme précisé au début du chapitre, nous nous restreignons au calcul de l'état d'équilibre aéroélastique statique. Ainsi seuls les déplacements de la structure doivent être transmis au domaine fluide.

Afin d'alléger les explications nous notons désormais $\mathcal{F}$ le maillage du domaine occupé par le fluide, $\mathcal{S}$ le maillage de la structure, $\mathcal{F}^I$ le maillage surfacique matérialisant l'interface fluide-structure pour le maillage du domaine fluide (maillage de la surface mouillée par le fluide) et $\mathcal{S}^I$ le maillage matérialisant cette interface pour le maillage de la structure.

Les deux principales difficultés à surmonter sont donc d'une part de transmettre les efforts aérodynamiques évalués sur le maillage du domaine fluide correspondant à la paroi solide (en

d'autres termes à la surface mouillée) et d'autre part de déformer le maillage du domaine fluide en fonction des déplacements calculés sur le maillage de la structure. Pour des géométries complexes, ce dernier point est résolu par étapes. En effet, dans un premier temps les déplacements évalués sur le maillage de la structure sont transmis aux noeuds du maillage du domaine fluide correspondant à la paroi solide (maillage surfacique), puis, dans un deuxième temps, ces déplacements sont propagés au reste du maillage du domaine fluide (maillage volumique).

2.4.1 Transfert des déplacements de la structure aux noeuds de paroi

Nous avons choisi de ne nous intéresser qu'à la résolution partitionnée des équations couplées associées au système fluide-structure. Cette étape est indispensable et non triviale car les noeuds des maillages de la structure et du domaine fluide ne coïncident pas à l'interface fluide-structure. La difficulté est liée à la nature distincte des deux grilles. En effet, elles ont une densité très différente. Le maillage fluide doit permettre une évaluation précise du flux autour de l'avion. En conséquence, il est très raffiné près des parois et en particulier dans les zones où les gradients des grandeurs conservatives sont potentiellement élevés. Au contraire, le maillage de la structure couvre à la fois la paroi mais aussi l'intérieur de l'avion. Il est en général beaucoup moins raffiné. Le transfert des déplacements évalués sur le maillage de la structure vers le maillage de la paroi solide inclus dans le maillage du domaine fluide fait donc à la fois appel à des techniques d'interpolation et d'extrapolation. L'objectif de ce transfert est d'assurer la condition :

$$D_{\mathcal{S}^I} = D_{\mathcal{F}^I}$$

à l'interface fluide-structure, où D représente de manière générique le champ de déplacement sur l'une ou l'autre des interfaces $\mathcal{S}^I$ ou $\mathcal{F}^I$.

L'importance du sujet pour la précision du calcul de l'état d'équilibre a ainsi donné lieu à différentes études [74, 205, 206] comparant les différentes techniques d'interpolation/extrapolation utilisables en aéroélasticité numérique. Le but est donc, connaisant le champ de déplacement en un ensemble de points (noeuds du maillage de la structure) d'en déduire le déplacement en ensemble de points disjoint (noeuds de l'interface $\mathcal{F}^{\mathcal{I}}$).

1) Infinite-plate splines (IPS)

Cette méthode [95] est la plus répandue en aéroélasticité numérique (le code MSC/NATRAN en particulier l'utilise). En s'appuyant sur les solutions de l'équilibre d'une plaque plane infinie, cette méthode calcule les forces qui, appliquées en chacun des noeuds du maillage de la structure, donneraient lieu à un tel champ de déplacement. Une fois ces forces déterminées, elles sont substituées dans la solution pour en déduire le déplacement associé à chacun des points de l'interface $\mathcal{F}^I$.

2) Finite-plate splines (FPS)

Cette méthode [8, 36] fait appel aux éléments finis (éléments quadrilatère et triangle) pour définir une surface passant par les points de l'interface fluide-structure appartenant à la fois au maillage de la structure et au maillage du domaine fluide. Une fois cette surface définie, les déplacements de la structure sont naturellement reliés aux déplacements des points de l'interface appartenant au maillage fluide. Cette méthode d'interpolation possède le grand avantage de conserver le travail effectué par les efforts aérodynamiques lorsqu'elle est utilisée pour calculer

la distribution nodale d'efforts équivalents à appliquer sur le maillage de la structure.

3) Multiquadratic-biharmonics (MQ)

Comme son nom l'indique, cette méthode utilise une décomposition sur la base des surfaces quadratiques (c'est-à-dire définie par une équation quadratique) pour définir la surface reliant les points des l'interfaces $\mathcal{F}^I$ et $\mathcal{S}^I$. Les surfaces les plus utilisées sont les hyperboloïdes à deux nappes [119, 96].

4) Thin-plate spline (TPS)

Cette méthode [63] est semblable à la méthode MQ, elle utilise les splines surfaces - "*thin-plate spline*" en anglais - pour reconstruire la surface passant par les points des interfaces $\mathcal{F}^I$ et $\mathcal{S}^I$. La surface reconstruite a la propriété de minimiser une fonctionnelle énergie. En dimension 1, les splines cubiques peuvent être interprétées physiquement comme la position d'équilibre d'une poutre en flexion.

5) B-splines non uniformes (NUBS)

Cette méthode [205] utilise une décomposition en B-spline (polynômiale ou rationnelle) pour définir la surface reliant les points des deux frontières $\mathcal{F}^I$ et $\mathcal{S}^I$. Elle est cependant déconseillée de manière générale car elle est susceptible d'introduire des oscillations entre les points interpolés.

6) Méthode des mortiers

A la différence des méthodes d'interpolations précédentes, cette méthode - "*mortar method*" en anglais - ne cherche pas à imposer directement la condition

$$D_{\mathcal{S}^I} = D_{\mathcal{F}^I}.$$

Elle essaie en revanche d'assurer cette égalité au sens faible en imposant [69, 22]

$$\forall \Psi_S \in \mathbb{R}^{n_{\mathcal{S}^I}},\ \Psi_F \in \mathbb{R}^{n_{\mathcal{F}^I}},\quad \int_I \Psi_S D_{\mathcal{S}^I} - \Psi_F D_{\mathcal{F}^I} = 0.$$

où $n_{\mathcal{S}^I}$ et $n_{\mathcal{F}^I}$ représente le nombre de noeuds des interfaces $\mathcal{S}^I$ et $\mathcal{F}^I$.

Cette condition est discrétisée en fonction du problème, la résolution du système linéaire obtenu nécessite l'inversion d'une matrice dont la taille peut être prohibitive lorsque le maillage du domaine fluide est tridimensionnel par exemple.

Parmi les méthodes d'interpolation/extrapolation existantes, aucune ne possède toutes les propriétés souhaitables. La qualité des interpolations/extrapolations obtenues dépend du problème. Les articles de revues cités ci-dessus conseillent donc de tester différentes méthodes. Cependant les méthodes MQ et TPS sont celles fournissant les résultats les plus précis et les surfaces d'interpolation les plus lisses. Ce sont également les méthodes les plus robustes ; elles sont en particulier capables de reproduire de fortes irrégularités.

De plus, en notant h_F une grandeur caractéristique du maillage du domaine fluide et h_S une grandeur caractéristique du maillage structure, il a été prouvé [69] que l'erreur de discrétisation associée au système couplé fluide-structure croissait asymptotiquement en $O(\sqrt{h_F})$

lorsque le déplacement de la structure était transmis au maillage du domaine fluide par l'intermédiaire d'une méthode d'interpolation. En revanche lorsque les déplacements sont propagés au maillage du domaine fluide en utilisant la méthode des mortiers, il est possible de prouver que l'erreur de discrétisation associée au système couplé fluide-structure croît asymptotiquement en $O(h_F)+O(h_S)$. Cette méthode est donc en théorie plus précise que les méthodes d'interpolation. Cependant, comme le maillage du domaine fluide est en pratique beaucoup plus raffiné que le maillage de la structure, l'erreur de discrétisation associée à la méthode des mortiers retrouve l'ordre de grandeur des méthodes d'interpolation, et comme elle est beaucoup plus lourde à mettre en oeuvre en pratique, elle est peu utilisée pour les problèmes d'aéroélasticité classique.

Dans le cadre des calculs couplés fluide-structure utilisés dans cette étude, les déplacements de la structure sont directement transmis au maillage du domaine fluide sans passer par l'étape intermédiaire de projection/interpolation/extrapolation sur l'ensemble des noeuds de la paroi $\mathcal{F}^I$, et ceci en raison du caractère simplifié du modèle utilisé pour prédire les déplacements de la structure. Le domaine fluide est donc remaillé suivant un processus entièrement analytique à chaque étape du calcul de l'équilibre aéroélastique statique. Cette procédure de remaillage est donc détaillée dans la partie 2.4.3.

2.4.2 Propagation du déplacement des noeuds de la paroi au reste du maillage du domaine fluide

Une fois les déplacements de la structure calculés sur $\mathcal{S}^I$ et transmis à $\mathcal{F}^I$, le maillage volumique correspondant au domaine fluide doit être remaillé. Pour cela il existe plusieurs techniques classiques. Ces dernières sont présentées brièvement dans cette section.

Les méthodes algébriques ou intégrales

Les méthodes algébriques ou intégrales [142, 197] donnent explicitement le déplacement de chaque point de maillage à partir du déplacement des points de la paroi. Elles font intervenir soit un calcul algébrique, soit un calcul intégral mais ne font pas intervenir de résolution de système linéaire. De manière générale, les déplacements sont interpolés entre la paroi et la frontière extérieure dont le déplacement est fixé à zéro. Il s'agit d'une méthode industrielle efficace et bien maitrisée sur des configurations complexes.

Notons $X(a)$ et $\delta X(a)$ la position et le déplacement transmis au noeud a du maillage du domaine fluide, $dS(b)$ le quart de la somme des surfaces des quatre mailles dont un coin est le noeud b, Ψ et ν deux facteurs sans dimension destinés à contrôler l'amortissement de la propagation des déplacements (lorsque Ψ vaut 1 le déplacement de la paroi est propagé dans tout le maillage). Les formules d'interpolations élémentaires peuvent être modifiées de sorte à s'affranchir de la topologie des lignes de maillage (formule (a)) et des effets dus à la densité de maillage (formule (b)) [167].

$$\delta X(a) = \sum_{b\in\mathcal{F}^I} \frac{\delta X(b)}{||X(a)X(b)||^\nu} \Big/ \left(\Psi \sum_{b\in\mathcal{F}^I} \frac{1}{||X(a)X(b)||^\nu}\right) \quad (a)$$

$$= \sum_{b\in\mathcal{F}^I} \frac{\delta X(b) dS(b)}{||X(a)X(b)||^\nu} \Big/ \left(\Psi \sum_{b\in\mathcal{F}^I} \frac{dS(b)}{||X(a)X(b)||^\nu}\right) \quad (b)$$

Les méthodes intégrales sont des méthodes simples et peu coûteuses. Elles nécessitent cependant un réglage préalable afin d'être adaptées aux différents types de maillages. La technique

présentée par Shankar et Ide [197] permet en outre, pour les calculs instationnaires, d'imposer une vitesse nulle au niveau de la frontière supérieure du domaine fluide.

La méthode des ressorts

La méthode des ressorts [64] consiste à envisager le maillage comme un treillis de ressorts, chaque arête porte alors un ressort dont il s'agit de déterminer la raideur. La déformation du maillage est obtenue en calculant la position du treillis lorsqu'il est soumis au déplacement forcé des points de la surface $\mathcal{F}^I$.

Cette méthode utilise le fait que le treillis de ressort est en équilibre à chaque itération de couplage entre le fluide et la structure, c'est-à-dire à chaque étape de propagation des déplacements et fait l'hypothèse que ces deux positions d'équilibre sont peu différentes pour effectuer un développement limité à l'ordre 1 des efforts appliqués au treillis à l'itération courante par rapport à ceux appliqués à l'itération précédente et exprimer ces efforts en fonction du champ de déplacement des extrémités des ressorts (noeuds du maillage du domaine fluide). En utilisant le fait qu'à l'itération précédente comme à l'itération courante la somme des efforts appliqués au treillis est nulle (position d'équilibre), cela conduit à résoudre le système [167] :

$$\sum_b K_{ab}\overrightarrow{n_{ab}}.(\overrightarrow{\delta X(b)} - \overrightarrow{\delta X(a)})\overrightarrow{n_{ab}} = \overrightarrow{0}$$

où $\delta X(a)$ et $\delta X(b)$ représentent les déplacements à appliquer aux noeuds a et b du maillage du domaine fluide, K_{ab} la raideur du ressort de l'arête $X(a)X(b)$ et $\overrightarrow{n_{ab}}$ le vecteur unitaire porté par ce ressort à l'itération précédente. En général, les raideurs sont définies en utilisant la formule suivante (où ν vaut généralement 1) :

$$K_{ab} = \frac{1}{||X_a^1 X_b^1||^\nu}.$$

Le calcul de la nouvelle position d'équilibre du treillis nécessite donc la résolution d'un système linéaire dont le nombre d'équations (scalaires) et d'inconnues (scalaires) est égal à trois fois le nombre de points du maillage du domaine fluide.

Les propriétés de régularité et d'orthogonalité du maillage peuvent être conservées grâce à l'utilisation de la technique proposée par Nakahashi et Deiwert [151].

La méthode de Batina

La méthode de Batina [19] est en réalité une simplification de la méthode des ressorts pour laquelle chaque ressort d'arête est remplacé par trois ressorts de même raideur, chacun des ressorts exerçant une force exclusivement dans les directions x, y ou z du repère physique. De telle sorte que l'écart entre les forces appliquées au point a entre les deux positions d'équilibre devient :

$$\overrightarrow{f}_{ab}^{1} = \overrightarrow{f}_{ab}^{0} + K_{ab}(\overrightarrow{\delta X_b} - \overrightarrow{\delta X_a})$$

et le système linéaire à résoudre :

$$\sum_b K_{ab}(\overrightarrow{\delta X_b} - \overrightarrow{\delta X_a}) = \overrightarrow{0}.$$

Les équations portant sur les coordonnées x, y et z sont alors découplées.

Méthode fondée sur analogie mécanique

La méthode fondée sur l'analogie avec l'élasticité linéaire des milieux continus [112] utilise une base de fonctions décrivant la déformation d'un héxaèdre de maillage pour calculer une matrice de rigidité associée au maillage, vu comme un solide élastique, et finalement relier le déplacement des noeuds au bilan des efforts (noté f) en chaque noeud. En distinguant les noeuds libres (noeuds du maillage du domaine fluide ne se trouvant pas sur la surface mouillée par le fluide) pour lesquels le déplacement n'est pas fixé par la structure et doit justement être estimé, des noeuds dont le déplacement est contraint par celui de la structure, le déplacement des noeuds du maillage $\mathcal{F}$ satisfait :

$$\begin{pmatrix} K_{\mathcal{F}_{\setminus\mathcal{F}^I}\ ,\ \mathcal{F}_{\setminus\mathcal{F}^I}} & K_{\mathcal{F}_{\setminus\mathcal{F}^I}\ ,\ \mathcal{F}^I} \\ K_{\mathcal{F}^I\ ,\ \mathcal{F}_{\setminus\mathcal{F}^I}} & K_{\mathcal{F}^I\ ,\ \mathcal{F}^I} \end{pmatrix} \begin{pmatrix} \delta X_{\mathcal{F}_{\setminus\mathcal{F}^I}} \\ \delta X_{\mathcal{F}^I} \end{pmatrix} \begin{pmatrix} 0 \\ f_{\mathcal{F}^I} \end{pmatrix}$$

où K est une matrice de rigidité à fixer par le concepteur en fonction de l'amortissement qu'il souhaite donner aux déplacements de la structure à l'intérieur du maillage fluide. Le système ci-dessus conduit au système linéaire suivant (dont la matrice est symétrique définie positive)qu'il suffit de résoudre pour connaitre le déplacement de chacun des points intérieurs au maillage fluide.

$$K_{\mathcal{F}_{\setminus\mathcal{F}^I}\ ,\ \mathcal{F}_{\setminus\mathcal{F}^I}}\ \delta X_{\mathcal{F}_{\setminus\mathcal{F}^I}} = -K_{\mathcal{F}^I\ ,\ \mathcal{F}^I}\ \delta X_{\mathcal{F}^I}$$

2.4.3 Technique de remaillage du domaine fluide utilisée dans le cadre de cette étude

Dans le cadre de cette étude, les déplacements de la structure sont prédits de telle sorte qu'il est possible de transmettre directement, par l'intermédiaire d'une formule analytique simple, les déplacements de la structure évalués sur le maillage de la poutre, à l'ensemble du maillage du domaine fluide. Cette méthode diffère donc des méthodes classiques du type "trois champs" [70, 138] qui ajoutent le champ des vitesses ou des déplacements aux noeuds du maillage du domaine fluide aux inconnnues standards du problème aéroélastique (grandeurs conservatives aérodynamiques et champ de déplacement de la structure) mais pour lesquelles les noeuds du maillage du domaine fluide conservent les mêmes coordonnées tout au cours du processus de calcul de l'équilibre aéroélastique. Dans notre cas, le maillage du domaine fluide n'est donc pas constant par opposition aux méthodes de transpiration, les coordonnées des noeuds changent au cours de la procédure de calcul de l'équilibre aéroélastique. Le nombre de noeuds du maillage du domaine fluide reste identique, mais chacun de ses noeuds voit ses coordonnées évoluer au cours du processus de calcul de l'équilibre aéroélastique.

En chacun des points de la poutre sont calculés, grâce au solveur structure présenté dans la section 2.3, un déplacement en flexion - déplacement vertical selon l'axe z du repère global (x, y, z) - noté ω_p - ainsi qu'un déplacement en torsion autour de la direction moyenne de l'aile représentée par l'axe y du repère global (x, y, z) - noté θ_p.

Le déplacement de l'aile, prédit par le solveur structure, est propagé au reste du domaine fluide en utilisant une analogie avec la mécanique du solide. En effet, de manière générale, en supposant connus le déplacement en flexion $\omega(P)$ selon l'axe z et le déplacement en torsion $\theta(P)$ autour de la direction moyenne y, en un point P, le déplacement du point Q, noté $\overrightarrow{\delta Q}$ est calculé de la manière suivante :

$$\overrightarrow{\delta Q} = \underbrace{\omega(P)\vec{z}}_{\text{translation selon } \vec{z}} + \underbrace{\overrightarrow{QP} \wedge \theta(P)\vec{y}}_{\text{rotation autour de l'axe } \vec{y}} .$$

Il résulte de l'addition d'un mouvement de translation vertical suivant l'axe z induit par la flexion de l'aile et d'un mouvement de rotation induit par la torsion de l'aile autour de l'axe y.

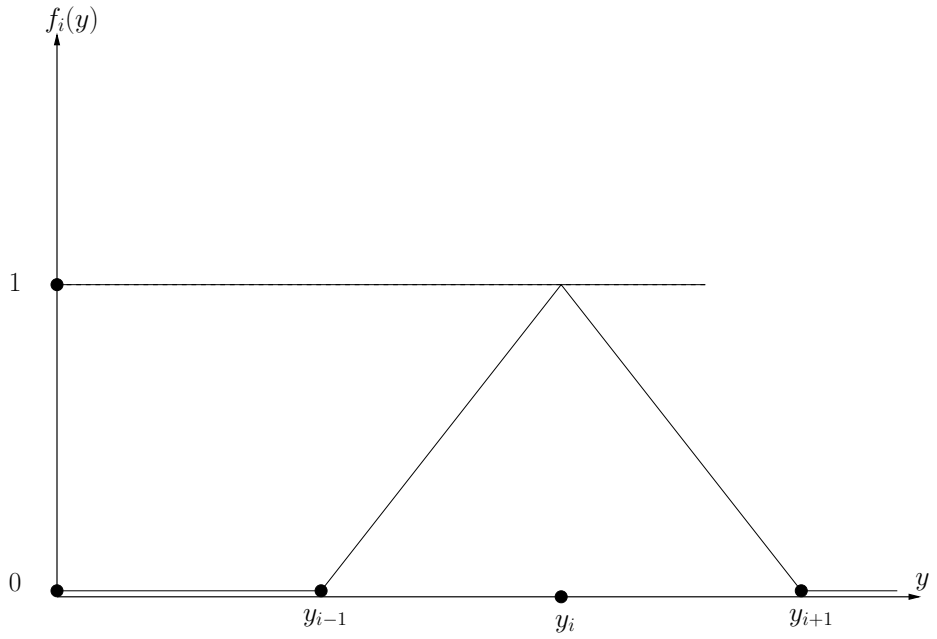

FIG. 2.3 – Fonction d'interpolation utilisée pour le calcul des déplacements

En utilisant cette formule, il est donc possible de déterminer le déplacement à appliquer en chacun des noeuds du maillage du domaine fluide à condition d'avoir déterminé le point P auquel sont évalués $\omega(P)$ et $\theta(P)$. Dans le cadre de ce travail, nous avons choisi d'utiliser le projeté du point Q à coordonnées constantes suivant la direction moyenne de la structure (i.e à y constant d'après la définition du repère (x, y, z)) sur la géométrie de la poutre. Il est peu probable que ce projeté coïncide avec un des noeuds du maillage de la poutre, points auxquels on connait les champs de déplacements en flexion et en torsion. Il faut donc choisir une méthode de calcul, c'est-à-dire d'interpolation, des déplacements au niveau du point projeté. Dans le cadre de cette étude nous avons choisi d'utiliser une simple interpolation linéaire, il s'agit donc d'un cas particulier de la méthode FPS décrite dans la section 2.4.1.

Décrivons en détail le processus de remaillage. Rappellons que X désigne le maillage structuré correspondant à une discrétisation du domaine fluide au cours du processus de résolution de l'équilibre aéroélastique et Z le maillage de la structure au cours du processus de résolution de l'équilibre aéroélastique, et notons X_{rig} le maillage structuré correspondant à une discrétisation du domaine fluide avant toute déformation causée par les effets aéroélastiques, il s'agit du maillage du domaine fluide autour de la forme de base de l'avion ou forme avant déformation en vol - "*jig shape*" en anglais. et Z_{rig} le maillage de la structure associée à la forme de base i.e. avant toute déformation due aux effets aéroélastiques. Au cours du processus de calcul de l'équilibre aéroélastique, les déplacements de la structure sont transmis au domaine fluide. Pour ce faire, deux techniques sont possibles : soit le maillage du domaine fluide est déformé (remaillage), soit un champ de vitesse / déplacement induit par le champ de vitesse / déplacement de la structure est associé à chacun des noeuds du maillage. Nous avons choisi dans le cadre de cette étude de procéder à un remaillage du domaine fluide. Ainsi, le nombre de noeuds constituant le maillage reste fixe, en revanche les coordonnées de chacun des noeuds évoluent en fonction du champ de déplacement de l'aile. En pratique, le nouveau champ de coordonnées est calculé à partir du champ de coordonnées du maillage du domaine fluide autour de la forme de base X_{rig}, du champ de déplacement de la structure D et des coordonnées du maillage de la structure Z. Dans le cadre de ce travail, nous nous restreignons volontairement aux configurations de vol à faibles déplacements de la structure. De telle sorte que l'**hypothèse des petits déplacements et**

déformations peut être faite sur la structure. Par conséquence, le maillage de la structure n'est pas déplacé au cours du processus de calcul de l'équilibre aéroélastique. Le champ de déplacement induit par le chargement aérodynamique sur la structure est ainsi calculé sur le maillage Z_{rig}. Ces déplacements sont transmis au maillage du domaine fluide qui se déforme, mais le maillage de la structure reste, lui, fixe au cours du processus de calcul de l'équilibre aéroélastique. En résumé, le nouveau champ de coordonnées X est calculé à partir de X_{rig}, de D et de Z_{rig}, dépendance que l'on écrira $X(X_{rig}, Z_{rig}, D)$

Introduisons les notations suivantes :

- $N_{rig_{ijk}}$ le noeud d'indices (i, j, k) du maillage structuré X_{rig} correspondant à une discrétisation du domaine fluide autour de la forme rigide,
- $(x_{rig_{ijk}}, y_{rig_{ijk}}, z_{rig_{ijk}})$ les coordonnées du noeud $N_{rig_{ijk}}$ exprimées dans le repère global (x, y, z) ,
- N_{ijk} le noeud d'indices (i, j, k) du maillage structuré X correspondant à une discrétisation du domaine fluide autour de la forme courante,
- $(x_{ijk}, y_{ijk}, z_{ijk})$ les coordonnées du noeud N_{ijk} exprimées dans le repère global (x, y, z),
- $(\tilde{x}_{ijk}, \tilde{y}_{ijk}, \tilde{z}_{ijk})$ les coordonnées du noeud N_{ijk} exprimées dans le repère global (x, y, z) après transfert des déplacement de la structure au maillage courant X,
- N'_{ijk} le projeté du noeud $N_{rig_{ijk}}$ sur la géométrie de la poutre à coordonnées constantes dans la direction moyenne de la structure (c'est-à-dire à y constant d'après la définition du repère (x, y, z)),
- $(x'_{ijk}, y'_{ijk}, z'_{ijk})$ les coordonnées du projeté N'_{ijk} exprimées dans le repère global (x, y, z),
- ω' le déplacement de flexion interpolé au point N'_{ijk},
- θ' le déplacement de torsion interpolé au point N'_{ijk}.

Le déplacement à appliquer au noeud $N_{rig_{ijk}}$ est calculé en projetant ce noeud sur le maillage de la structure perpendiculairement à la direction moyenne de la structure (axe y dans notre cas). Pour l'estimation des déplacements en flexion et en torsion au point N'_{ijk} de la géométrie de la poutre, nous avons choisi d'utiliser la fonction d'interpolation représentée par la figure 2.3. Il s'agit d'un processus d'interpolation linéaire. Les déplacements en flexion et en torsion au point N'_{ijk} sont alors calculés par la formule suivante :

$$\begin{cases} \omega' = \omega(y_{rig_{ijk}}) = \displaystyle\sum_{i=1}^{n_p} f_i(y_{rig_{ijk}})\ \omega_{p_i} \\ \\ \theta' = \theta(y_{rig_{ijk}}) = \displaystyle\sum_{i=1}^{n_p} f_i(y_{rig_{ijk}})\ \theta_{p_i} \end{cases} .$$

En d'autres termes si le projeté N'_{ijk} du noeud $N_{rig_{ijk}}$ sur le maillage de la structure se situe entre les noeuds P_{i-1} et P_i, c'est-à-dire si $y_{i-1} \leq y_{rig_{ijk}} \leq y_i$ (voir la figure 2.4), alors les déplacements en flexion et en torsion au point N'_{ijk} sont égaux à :

$$\begin{cases} \omega' = \omega(y_{rig_{ijk}}) = \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ \omega_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ \omega_{p_i} \\ \\ \theta' = \theta(y_{rig_{ijk}}) = \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ \theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ \theta_{p_i} \end{cases} .$$

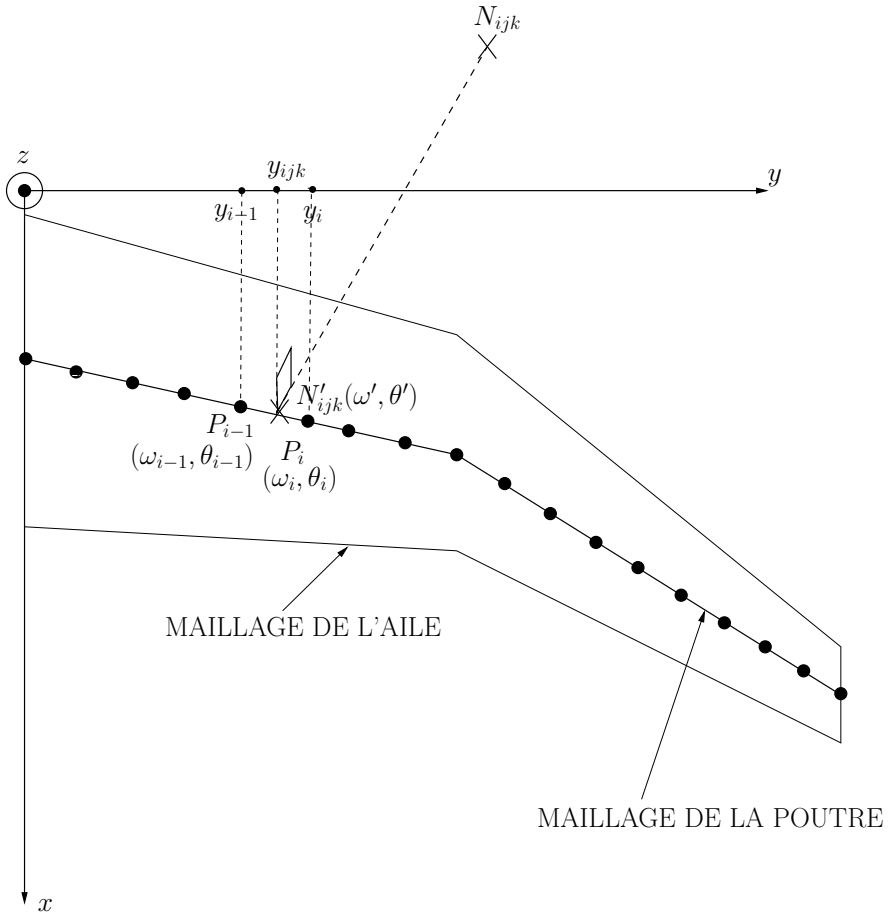

FIG. 2.4 – Projection des noeuds du maillage du domaine fluide sur la géométrie de la poutre

Les valeurs des déplacements en flexion et en torsion associées au projeté étant connues, il est alors possible de calculer le déplacement à appliquer au noeud $N_{rig_{ijk}}$ par la formule suivante :

$$\begin{aligned}
\overrightarrow{\delta N_{rig_{ijk}}} &= \omega'\vec{z} + \overrightarrow{N_{rig_{ijk}}N'_{ijk}} \wedge \theta'\vec{y} \\
&= \omega'\vec{z} + \begin{pmatrix} x_{rig_{ijk}} - x'_{ijk} \\ 0 \\ z_{rig_{ijk}} - z'_{ijk} \end{pmatrix} \wedge \begin{pmatrix} 0 \\ \theta' \\ 0 \end{pmatrix} \\
&= \omega'\vec{z} + \begin{pmatrix} -(z_{rig_{ijk}} - z'_{ijk})\theta' \\ 0 \\ (x_{rig_{ijk}} - x'_{ijk})\theta' \end{pmatrix} \\
&= -(z_{rig_{ijk}} - z'_{ijk})\theta'\vec{x} + (\omega' + (x_{rig_{ijk}} - x'_{ijk})\theta')\vec{z}
\end{aligned}$$

Les coordonnées du projeté N'_{ijk} sont approchées par la même méthode, précisément celles-ci sont égales à :

$$\begin{cases}
x'_{ijk} = \displaystyle\sum_{i=1}^{n_p} f_i(y_{rig_{ijk}})\ x_i \\
y'_{ijk} = y_{rig_{ijk}} \\
z'_{ijk} = \displaystyle\sum_{i=1}^{n_p} f_i(y_{rig_{ijk}})\ z_i
\end{cases},$$

soit, en utilisant la configuration décrite par la figure 2.4,

$$\begin{cases}
x'_{ijk} = \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ x_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ x_i \\
y'_{ijk} = y_{rig_{ijk}} \\
z'_{ijk} = \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ z_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ z_i
\end{cases}$$

Après transfert des déplacements de la structure au maillage courant X, les coordonnées du noeud N_{ijk} dans le nouveau maillage courant sont :

$$\begin{cases}
\tilde{x}_{ijk} = x_{rig_{ijk}} - (z_{rig_{ijk}} - z'_{ijk})\theta' \\
\tilde{y}_{ijk} = y_{rig_{ijk}} \\
\tilde{z}_{ijk} = z_{rig_{ijk}} + (\omega' + (x_{rig_{ijk}} - x'_{ijk})\theta')
\end{cases}$$

$$\left\{\begin{array}{l}
\tilde{x}_{ijk} = x_{rig_{ijk}} \\
\quad - \left(z_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ z_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ z_i \right) \left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ \theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ \theta_{p_i} \right) \\
\tilde{y}_{ijk} = y_{rig_{ijk}} \\
\tilde{z}_{ijk} = z_{rig_{ijk}} + \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ \omega_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ \omega_{p_i} + \\
\left(x_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ x_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ x_i \right) \left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ \theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ \theta_{p_i} \right)
\end{array}\right.$$

Le processus de calcul décrit ci-dessus permet donc de calculer les coordonnées de tous les noeuds du maillage du domaine fluide après transfert des déplacements de la structure soumise au chargement aérodynamique. Il peut ne pas être judicieux de déplacer l'ensemble des noeuds du maillage du domaine fluide, pour les maillages du type RANS en particulier. Dans ce cas, il est préférable d'amortir la propagation des déplacements. Dans le cadre de ce travail, nous avons choisi de propager les déplacements de la structure sans amortissement à l'intérieur d'un cylindre de rayon r centré autour de l'axe de la poutre, de les amortir linéairement entre le cylindre précédent et un cylindre de rayon supérieur égal à ar où $a > 1$, et de ne pas propager la déformation à l'extérieur de ce dernier cylindre (voir la figure 2.5).

De manière plus précise, nous avons défini le facteur d'amortissement ζ par les relations

$$\left\{\begin{array}{lll}
\text{si } d_{ijk} \leq R & \text{alors} & \zeta = 1 \\
\text{si } R \leq d_{ijk} \leq ar & \text{alors} & \zeta = 1 - \dfrac{d_{ijk} - r}{ar - r} \\
\text{si } d_{ijk} \geq ar & \text{alors} & \zeta = 0
\end{array}\right.$$

où d_{ijk} représente la distance du noeud $N_{rig_{ijk}}$ à l'axe de la poutre. Celui-ci permet d'amortir les déplacements dus à la torsion, l'angle de rotation utilisé étant alors égal à $\zeta\theta'$ à la place de θ'. En revanche les déplacements dus à la flexion ne sont pas amortis.

Après transfert des déplacements de la structure au maillage courant X, les coordonnées du noeud N_{ijk} dans le nouveau maillage courant, sont alors égales à :

$$\left\{\begin{array}{l}
\tilde{x}_{ijk} = x_{rig_{ijk}} \\
\quad - \left(z_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ z_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ z_i \right) \left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ \zeta\theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ \zeta\theta_{p_i} \right) \\
\tilde{y}_{ijk} = y_{rig_{ijk}} \\
\tilde{z}_{ijk} = z_{rig_{ijk}} + \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ \omega_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ \omega_{p_i} + \\
\left(x_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ x_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ x_i \right) \left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\ \zeta\theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\ \zeta\theta_{p_i} \right)
\end{array}\right. \tag{2.5}$$

Les coordonnées des noeuds du maillage du domaine fluide vues par le solveur fluide sont mises à jour à chaque étape du processus de calcul de l'équilibre aéroélastique. Cette mise à jour

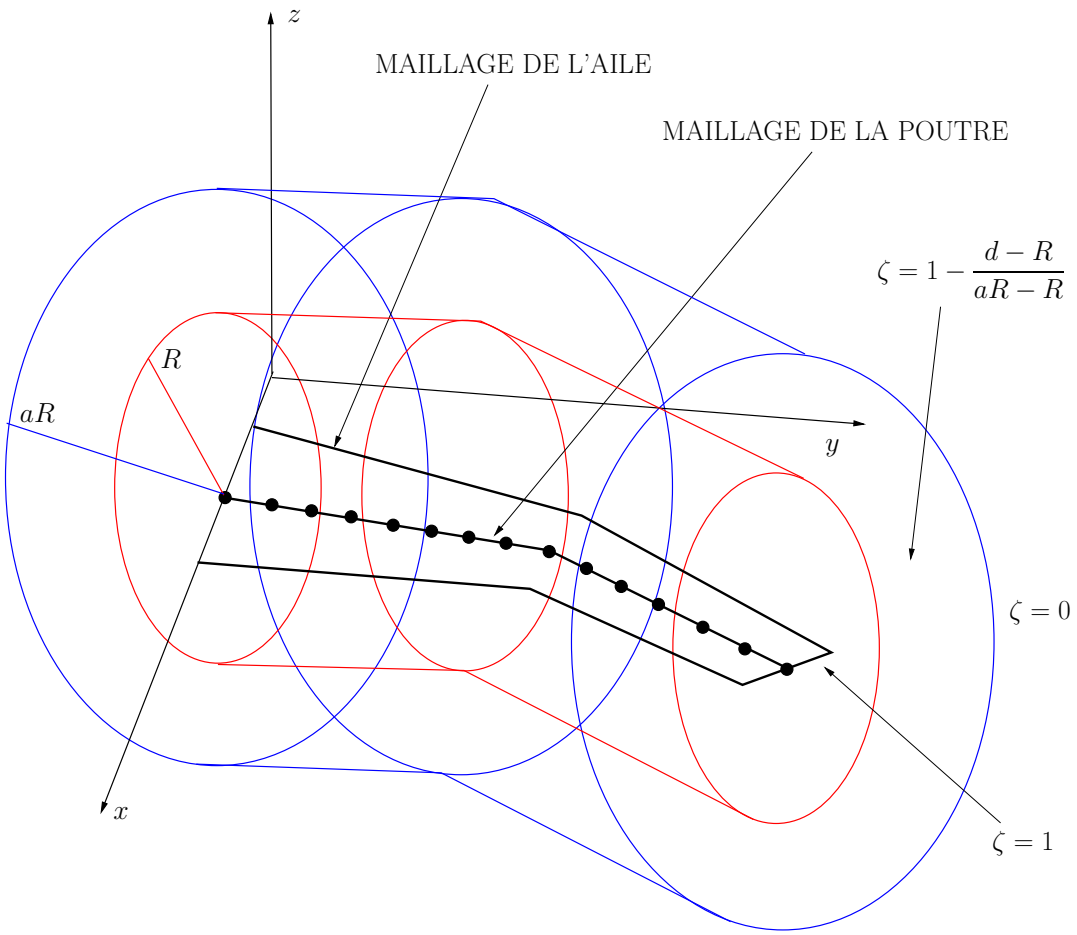

FIG. 2.5 – Amortissement de la propagation des déplacements de la structure dans le maillage du domaine fluide

ne passe pas par une lecture/écriture de fichier, mais se fait à l'intérieur du code de simulation. Toutes les données géométriques (les coordonnées des noeuds, les surfaces et les volumes du maillage) sont mises à jour à chaque transfert de déformation.

2.4.4 Transfert des efforts aérodynamiques au maillage de la structure

Les propriétés à satisfaire

Les efforts aérodynamiques à l'origine de la déformation de la structure doivent être traduits de manière à être interprétables par le solveur structure. Dans le cadre de cette étude, il s'agit de calculer le tenseur (force et moment) à appliquer en chacun des noeuds de la discrétisation de la poutre. Comme dans le cas du transfert des déplacements de la structure au maillage du domaine fluide, les difficultés proviennent de la non-coïncidence des maillages de la structure et du domaine fluide à l'interface fluide/solide. A la différence de la situation dans laquelle on cherche à transmettre les déplacements et pour laquelle seule la précision de l'interpolation utilisée est importante, le calcul des efforts équivalents doit tenir compte des propriétés physiques du système. Plus précisément le système fluide/structure que l'on considère est un système fermé si bien que l'équilibre des efforts et la conservation de l'énergie doivent être assurés. L'équilibre des efforts est satisfait si les efforts aérodynamiques calculés sur la discrétisation de l'interface fluide ($\mathcal{F}^I$) sont égaux aux efforts appliqués à la discrétisation de l'interface solide ($\mathcal{S}^I$). La conservation de l'énergie est assurée si l'énergie produite (respectivement absorbée) par la structure est égale à l'énergie absorbée (respectivement produite) par le fluide. Si une méthode de transfert des efforts aérodynamiques du fluide vers la structure vérifie ces deux exigences, alors elle est qualifiée de conservative.

La méthode la plus simple pour calculer le chargement équivalent sur la structure consiste à associer à chacun des noeuds du maillage de l'interface fluide-structure côté structure ($\mathcal{S}^I$) une cellule du maillage du domaine fluide et d'interpoler la pression et le tenseur des contraintes visqueuses en ce point à partir des valeurs prises par ces valeurs au centre de la cellule. Cette méthode est consistante avec la résolution du problème aérodynamique. Cependant, en aucun cas elle ne garantit l'égalité des efforts intégrés d'une part sur l'interface fluide-structure côté fluide ($\mathcal{F}^I$) et d'autre part sur l'interface fluide-structure côté structure ($\mathcal{S}^I$).

Un algorithme de transfert des efforts assurant la conservation de l'énergie entre les sous-systèmes fluide et structure a été proposé par Farhat *et al.* [70, 69, 68, 133] et détaillé pour le cas particulier où le comportement de la structure est prédit par les éléments finis. Pour assurer la conservation de l'énergie, cet algorithme impose l'égalité entre les travaux des efforts équivalents appliqués à la structure lors d'un déplacement virtuel mais admissible de l'interface $\mathcal{S}^I$ et les travaux des efforts aérodynamiques lors du déplacement virtuel équivalent reporté sur l'interface $\mathcal{F}^I$. Notons :

- δD_S un déplacement virtuel, admissible de l'interface $\mathcal{S}^I$;
- δD_F le déplacement virtuel de l'interface $\mathcal{F}^I$ équivalent au déplacement virtuel δD_S de l'interface $\mathcal{S}^I$ (calculé par la procédure de transfert des déplacements) ;
- δW_S le travail des efforts équivalents appliqués à la structure lors du déplacement virtuel, admissible δD_S de l'interface $\mathcal{S}^I$;
- δW_F le travail des efforts aérodynamiques lors du déplacement virtuel équivalent δD_F reporté sur l'interface $\mathcal{F}^I$,
- L_S les efforts aérodynamiques équivalents appliqués à la structure ;
- L_F les efforts aérodynamiques.

L'algorithme satisfait l'égalité suivante :

$$\left(\delta W_S = \int_{\mathcal{S}^I} L.\delta D = L_S.\delta D_S\right) = \left(\delta W_F = \int_{\mathcal{F}^I} L.\delta D = L_F.\delta D_F\right)$$

En utilisant la solution proposée par Farhat *et al.*, une méthode de transfert des efforts aérodynamiques à la structure est conservative si elle satisfait les deux conditions suivantes :

$$\int_{\mathcal{F}^I} L = \int_{\mathcal{S}^I} L \qquad \text{(C1)}$$

$$\left[\delta W_S = \int_{\mathcal{S}^I} L.\delta D\right] = \left[\delta W_F = \int_{\mathcal{F}^I} L.\delta D\right] \quad \text{(C2)}$$

où L représente de manière générique le champ des efforts appliqués à la surface considérée. En utilisant les notations introduites ci-dessus, ces conditions s'énoncent simplement comme suit :

$$L_S = L_F \qquad \text{(C1)}$$

$$L_S.\delta D_S = L_F.\delta D_F \quad \text{(C2)}$$

La plupart des méthodes de transfert des efforts aérodynamiques utilisant des interpolations ne sont pas conservatives. En particulier, les efforts aérodynamiques sommés sur l'interface $\mathcal{F}^I$ ne sont pas égaux aux efforts aérodynamiques équivalents transférés à la structure et sommés sur l'interface $\mathcal{S}^I$; la condition (C1) n'est donc pas satisfaite. Pour assurer l'équilibre des efforts, il est conseillé d'utiliser un unique support pour le calcul des efforts, soit la discrétisation associée au domaine fluide, soit la discrétisation de la structure. La discrétisation du domaine fluide étant en général beaucoup plus fine que la discrétisation de la structure, il est donc préférable de l'utiliser pour calculer les efforts aérodynamiques équivalents. Dans ce cas l'égalité des efforts est assuré [69].

Il a été démontré sur des cas pratiques par Farhat *et al.*, qu'une méthode de transfert des efforts consistante mais non conservative analytiquement peut s'avérer numériquement conservative et mener à des prédictions fiables des déplacements de la structure à l'équilibre aéroélastique. Ce bon comportement est assuré en pratique si les maillages à l'interface du domaine fluide et de la structure ne sont pas trop différents et si les conditions aérodynamiques ne sont pas trop sévères (absence de choc fort, de zone de transition laminaire/turbulent). En particulier, Farhat *et al.* ont montré [69] sur des algorithmes de transfert des efforts aérodynamiques utilisant des interpolations et ayant la propriété d'être consistants mais non conservatifs en théorie, qu'un algorithme de transfert peut être numériquement conservatif pour un écoulement supersonique sans choc, mais non conservatif pour un écoulement transsonique avec choc fort. Dans le cas général, lorsque les maillages du domaine fluide et de la structure partagent le même support géométrique (mais pas forcément la même discrétisation) et que les conditions aérodynamiques ne sont pas trop sévères (absence de choc fort, de zone de transition laminaire/turbulent), la propriété de conservativité n'est pas critique puisque des méthodes de transfert consistantes mais non conservatives en théorie fournissent de bons résultats en matière de prédiction des déplacements aéroélastiques [70, 169, 133].

Technique de transfert des efforts utilisée dans le cadre de cette étude

Afin de garantir la conservativité du transfert des efforts aérodynamiques à la structure, nous avons décidé, comme conseillé dans [69] par exemple, de les évaluer sur l'un des deux supports ($\mathcal{F}^I$ ou $\mathcal{S}^I$). Le maillage du domaine fluide étant beaucoup plus raffiné que le maillage de la structure, nous l'avons retenu pour l'évaluation des efforts , avant de les transmettre aux noeuds du maillage de la structure.

Afin de décrire précisément ce transfert, introduisons tout d'abord les notations suivantes :

- n_f^I le nombre d'interfaces dont est composée le maillage surfacique $\mathcal{F}^I$;
- I_l l'interface numéro l du maillage surfacique $\mathcal{F}^I$ $(1 \leq l \leq n_f^I)$;
- S_l l'aire de l'interface I_l ;
- G_l le barycentre de l'interface I_l ;
- $\vec{n}_l = (n_{lx}, n_{ly}, n_{lz})$ le vecteur unitaire normal à l'interface I_l.

L'objectif est de calculer le tenseur des efforts à transmettre en chacun des noeuds de la poutre P_i, $1 \leq i \leq n_p$. Celui-ci est noté : $\left(\overrightarrow{\mathcal{F}_i}, \overrightarrow{\mathcal{M}_i}\right)$, où $\overrightarrow{\mathcal{F}_i}$ et $\overrightarrow{\mathcal{M}_i}$ sont respectivement la force et le moment issus du chargement aérodynamique qui s'appliquent au noeud P_i, $1 \leq i \leq n_p$.

Notons $\mathcal{V}_i$, $1 \leq i \leq n_p$, la zone de l'espace comprise entre les deux plans parallèles et perpendiculaires à l'axe de la poutre et passant par les milieux des segments $[P_{i-1}, P_i]$ et $[P_i, P_{i+1}]$ (ce sont respectivement les points $\dfrac{P_i + P_{i-1}}{2}$ et $\dfrac{P_{i+1} + P_i}{2}$). Cette zone a été définie dans le cadre de ce travail comme la zone d'influence du noeud P_i, $1 \leq i \leq n_p$, du maillage de la structure. Par définition, le noeud P_0 correspond à l'emplanture de l'aile. Le déplacement de ce point étant toujours nul (conditions à la limite), les déplacements de la poutre sont calculés à partir du noeud P_1 jusqu'au saumon de l'aile. Il est alors possible d'associer à chacun des noeuds du maillage de la structure la zone de l'espace définie précédemment. Pour le noeud P_1 il s'agit de la partie de l'espace comprise entre les deux plans parallèles perpendiculaires à l'axe de la poutre et passant par les points milieux des segments $[P_0, P_1]$ et $[P_1, P_2]$ (ce sont respectivement les points $\dfrac{P_1 + P_0}{2}$ et $\dfrac{P_2 + P_1}{2}$). Ces notations sont illustrées par la figure 2.6.

Enfin, définissons pour chacune des interfaces I_l, $1 \leq l \leq n_f^I$, du maillage surfacique $\mathcal{F}^I$, l'aire S_l^i égale à l'aire de la surface correspondant à l'intersection entre l'interface I_l et la zone d'influence $\mathcal{V}_i$ associée au noeud P_i, $1 \leq i \leq n_p$, du maillage de la structure :

$$S_l^i = \text{aire}\left(I_l \ \cap \ \mathcal{V}_i\right)$$

Remarquons que l'aire S_l^i est égale à 0 lorsque l'interface I_l est en dehors de la zone d'influence $\mathcal{V}_i$ associée au noeud P_i. La définition de l'aire S_l^i est illustrée par la figure 2.7. De plus, le maillage de la structure est construit de telle sorte que la totalité du maillage surfacique $\mathcal{F}^I$ est incluse dans l'union des $\mathcal{V}_i$ $\mathcal{F}^I \subset \cup_{i=1}^{n_p} \mathcal{V}_i$, si bien que l'aire de l'interface I_l est égale à la somme des aires S_l^i, $1 \leq i \leq n_p$:

$$\forall 1 \leq l \leq n_f^I \quad , \quad S_l = \sum_{i=1}^{n_p} S_l^i.$$

L'aire S_l^i permet de pondérer les efforts aérodynamiques agissant sur l'interface I_l qui seront transmis au noeud P_i.

De manière générale et en simplifiant le tenseur des contraintes visqueuses d'après l'hypothèse de Stokes, les efforts exercés par un écoulement de vitesse $\overrightarrow{V}$ (avec pression statique infinie p_∞) sur une surface $\mathcal{S}$ de normale $\vec{n}$ sont égaux à :

$$\overrightarrow{\mathcal{F}} = \underbrace{\int_{\mathcal{S}} (p - p_\infty) \, .\vec{n} dS}_{\text{effort dû à la pression}} + \underbrace{\int_{\mathcal{S}} \left(-\frac{2}{3} \mu div \overrightarrow{V} \overrightarrow{\overrightarrow{I}} + \mu \left(\overrightarrow{\overrightarrow{grad} \overrightarrow{V}} + \overrightarrow{\overrightarrow{grad} \overrightarrow{V}}^{\,T} \right) \right) .\vec{n} dS}_{\text{effort dû à la viscosité}}$$

Pour un calcul du type Euler (hypothèse fluide parfait), les efforts aérodynamiques agissant sur la structure se limitant aux efforts de pression :

$$\overrightarrow{\mathcal{F}}_{\text{Euler}} = \int_{\mathcal{S}} (p - p_\infty) \, \vec{n} dS$$

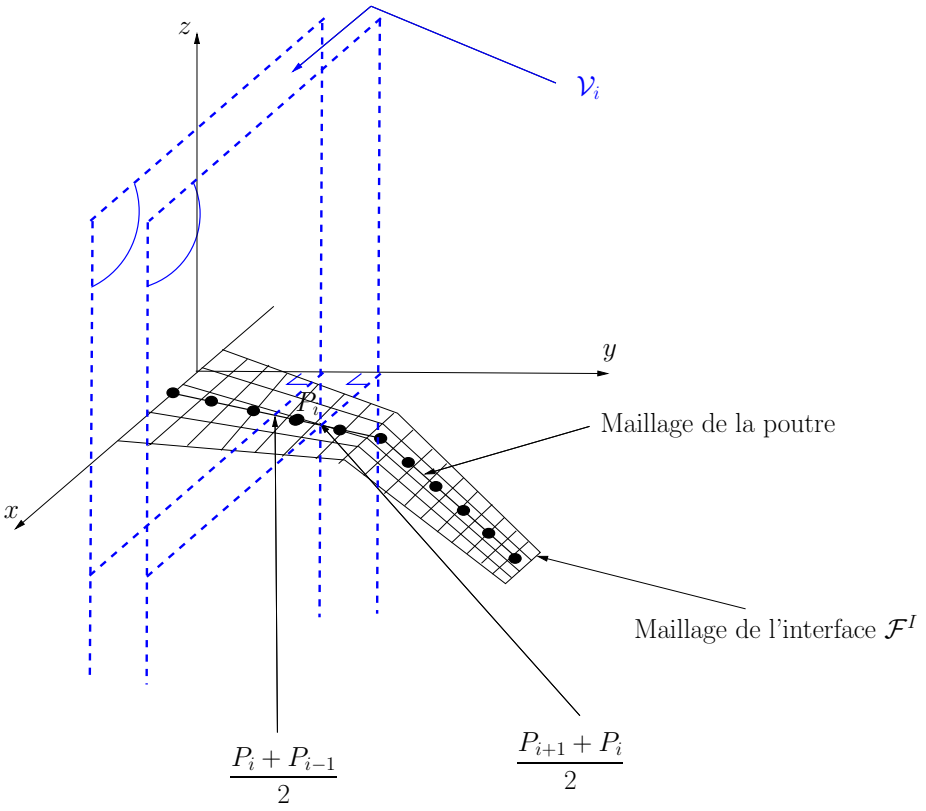

FIG. 2.6 – Schéma explicatif de la définition de la zone $\mathcal{V}_i$ associée à chacun des noeuds P_i, $1 \leq i \leq n_p$ du maillage de la structure.

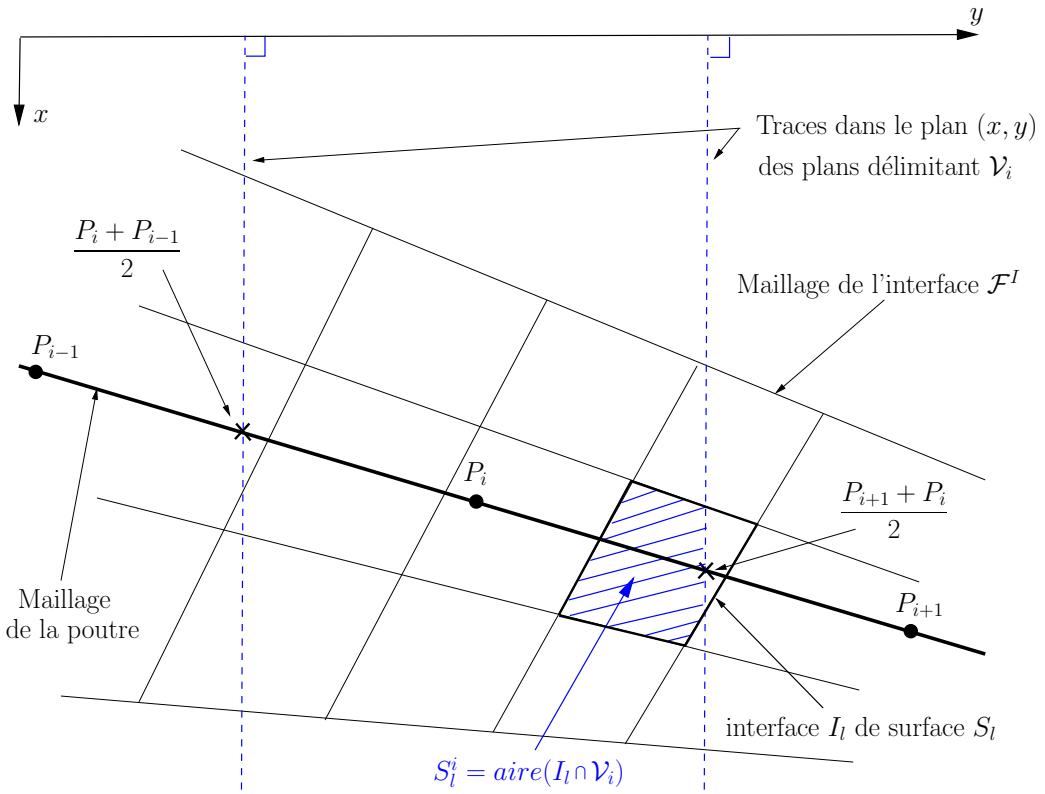

FIG. 2.7 – Schéma explicatif du calcul de l'aire S_l^i.

Le tenseur des efforts aérodynamiques transmis au noeud P_i, $1 \leq i \leq n_p$, du maillage de la structure sont alors calculés, dans le cadre de ce travail, de la manière suivante :

$$\begin{cases} \overrightarrow{\mathcal{F}_i} = \sum_{l=1}^{n_f^I} S_l^i \left(p_l - p_\infty\right) \vec{n}_l \\ \\ \overrightarrow{\mathcal{M}_i} = \sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge \left(S_l^i \left(p_l - p_\infty\right) \vec{n}_l\right) \end{cases}$$

En revanche, pour un calcul du type Navier-Stokes (fluide réel), les efforts aérodynamiques agissant sur la structure sont composés à la fois des efforts de pression et des efforts visqueux. Les efforts visqueux sont négligés dans la majorité des problèmes d'aéroélasticité. En effet, les efforts de pression sont en général prédominants dans la génération des déplacements aéroélastiques [128, 38, 76, 195, 57, 82]. La contrainte visqueuse tangentielle à la paroi est égale à :

$$\tau_p = \mu \left(\frac{\partial V_t}{\partial n_p}\right)_{\text{paroi}},$$

où τ_p représente la contrainte visqueuse tangentielle à la paroi, V_t la vitesse tangentielle à la paroi et n_p la direction normale à la paroi. Cette contrainte est portée par le vecteur $\vec{t}$ tangent à la paroi.

Les efforts aérodynamiques exercés par le fluide réel (équations de Navier Stokes) sur la structure sont alors approchés par la formule suivante :

$$\overrightarrow{\mathcal{F}}_{\text{réel}} = \int_{\mathcal{S}} \left(p - p_\infty\right) \vec{n} dS + \int_{\mathcal{S}} \mu \left(\frac{\partial V_t}{\partial n_p}\right) \vec{t} dS$$

Introduisons également les notations suivantes :

- p_l la pression statique exercée par le fluide sur l'interface I_l du maillage surfacique $\mathcal{F}^I$,
- μ_l la viscosité dynamique associée à l'interface I_l,
- V_t^l la composante tangentielle de la vitesse du fluide sur l'interface I_l,
- $\vec{t}_l$ le vecteur tangent à l'interface I_l.

Le tenseur des efforts aérodynamiques transmis au noeud P_i, $1 \leq i \leq n_p$, du maillage de la structure sont donc calculés, de la manière suivante :

$$\begin{cases} \overrightarrow{\mathcal{F}_i} = \sum_{l=1}^{n_f^I} S_l^i \left(p_l - p_\infty\right) \vec{n}_l + S_l^i \mu_l \left(\frac{\partial V_t^l}{\partial n_l}\right)_l \vec{t}_l \\ \\ \overrightarrow{\mathcal{M}_i} = \sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge \left(S_l^i \left(p_l - p_\infty\right) \vec{n}_l + S_l^i \mu_l \left(\frac{\partial V_t^l}{\partial n_l}\right)_l \vec{t}_l\right) \end{cases}$$

Notons ρ_c, $(\rho u)_c$, $(\rho v)_c$, $(\rho w)_c$ et $(\rho e)_c$ respectivement la masse volumique, les composantes selon $\vec{x}$, $\vec{y}$, et $\vec{z}$ de la quantité de mouvement, et l'énergie totale au centre de la cellule adjacente à la paroi au niveau de l'interface I_l du maillage $\mathcal{F}^I$ (I_l est une des faces de la cellule).

A partir de ces cinq grandeurs conservatives, il est possible de calculer la pression statique p_c et la vitesse du son c_c au centre de la cellule par la relation des gaz parfaits :

$$\begin{cases} p_c = (\gamma - 1)\rho_c e_c - \dfrac{1}{2}\dfrac{(\rho_c u_c)^2 + (\rho_c v_c)^2 + (\rho_c w_c)^2}{\rho_c} \\ \\ c_c = \sqrt{\gamma \dfrac{p_c}{\rho_c}} \end{cases}$$

Les grandeurs à la paroi sont évaluées grâce aux valeurs calculées au centre de la cellule adjacente à la paroi. Dans le cadre de ce travail, lorsque le fluide est supposé parfait (équations d'Euler), la pression statique p_l sur l'interface I_l est calculée de la manière suivante (issue de la théorie des relations caractéristiques [218]) :

$$p_l = p_c - c_c \left(\rho \overrightarrow{V}\right)_c \cdot \vec{n}_l = p_c - c_c \left((\rho u)_c \, n_{lx} + (\rho v)_c \, n_{ly} + (\rho w)_c \, n_{lz}\right)$$

Lorsque le fluide est visqueux (équations de Navier-Stokes) et que la paroi est adiabatique, la pression statique, la densité et la température statique à la paroi sont supposées égales à celles de cellule adjacente à la paroi. La pression statique p_l sur l'interface I_l est alors donnée par l'égalité :

$$p_l = p_c$$

De plus, dans ce cas, il faut évaluer le terme correspondant aux efforts visqueux à la paroi dans l'expression du chargement aérodynamique sur la structure. Ce terme a été approché par la relation suivante :

$$\begin{aligned} \mu_l \left(\frac{\partial V_t^l}{\partial n_l}\right)_l \vec{t}_l &\approx \mu_c \frac{V_t^c}{\frac{1}{2}\frac{\mathrm{Vol}_c}{S_l}} \\ &\approx \mu_c \frac{2S_l}{\mathrm{Vol}_c} \left(\frac{\left((\rho u)_c - (\rho u)_c \, n_{lx}\right)}{\rho_c} \vec{x} + \frac{\left((\rho v)_c - (\rho v)_c \, n_{ly}\right)}{\rho_c} \vec{y} + \frac{\left((\rho w)_c - (\rho w)_c \, n_{lz}\right)}{\rho_c} \vec{z} \right) \end{aligned}$$

dans laquelle Vol_c représente le volume de la cellule adjacente à la paroi au niveau de l'interface I_l, μ_c la viscosité dynamique du fluide au centre de la cellule adjacente et V_t^c la vitesse du fluide au centre de la cellule adjacente projetée sur le vecteur unitaire tangent à l'interface I_l.

Il est possible de vérifier que la méthode choisie pour transférer les efforts aérodynamiques à la structure conserve les efforts mais ne conserve pas l'énergie du système fermé fluide-structure. Notons $\overrightarrow{\mathcal{F}}_F$, $\overrightarrow{\mathcal{M}}_F$, $\overrightarrow{\mathcal{F}}_S$ et $\overrightarrow{\mathcal{M}}_S$ respectivement la somme des efforts et des moments (réduits à l'origine) sur les maillages $\mathcal{F}^I$ et $\mathcal{S}^I$. Montrons tout d'abord que la somme des efforts et des moments aérodynamiques évalués sur le maillage de la surface mouillée par le fluide $\mathcal{F}^I$ sont respectivement égaux à la somme des efforts et des moments appliqués au maillage de la structure $\mathcal{S}^I$. La somme des efforts aérodynamiques (forces et moments réduits en O) exercés par l'écoulement sur le maillage $\mathcal{F}^I$ est égale à :

$$\begin{cases} \overrightarrow{\mathcal{F}}_F = \displaystyle\sum_{l=1}^{n_f^I} S_l p_l \vec{n}_l \\ \\ \overrightarrow{\mathcal{M}}_F = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{OG_l} \wedge S_l p_l \vec{n}_l \end{cases} .$$

La somme des efforts aérodynamiques (forces et moments réduits en O) appliqués au maillage de la structure est, quant à elle, égale à :

$$\left\{\begin{array}{rcl}
\overrightarrow{\mathcal{F}}_S = \displaystyle\sum_{i=1}^{n_p}\sum_{l=1}^{n_f^I} p_l S_l^i\ \vec{n}_l = \sum_{l=1}^{n_f^I}\left(\sum_{i=1}^{n_p} S_l^i\right) p_l\vec{n}_l = \sum_{l=1}^{n_f^I} p_l S_l \vec{n}_l = \overrightarrow{\mathcal{F}}_F & & \\
\overrightarrow{\mathcal{M}}_S & = & \displaystyle\sum_{i=1}^{n_p}\left(\overrightarrow{\mathcal{M}_i} + \overrightarrow{OP_i}\wedge\overrightarrow{\mathcal{F}_i}\right) \\
& = & \displaystyle\sum_{l=1}^{n_f^I}\sum_{i=1}^{n_p} S_l^i\ \overrightarrow{P_iG_l}\wedge p_l\vec{n}_l + \sum_{l=1}^{n_f^I}\sum_{i=1}^{n_p} S_l^i\ \overrightarrow{OP_i}\wedge p_l\vec{n}_l \\
& = & \displaystyle\sum_{l=1}^{n_f^I}\left(\sum_{i=1}^{n_p} S_l^i\right)\overrightarrow{OG_l}\wedge p_l\vec{n}_l \\
& = & \displaystyle\sum_{l=1}^{n_f^I}\overrightarrow{OG_l}\wedge p_l S_l\vec{n}_l = \overrightarrow{\mathcal{M}}_F
\end{array}\right.$$

La somme des efforts aérodynamiques transmis à la structure est donc égale à la somme des efforts aérodynamiques exercés par l'écoulement sur la surface mouillée. L'algorithme de projection utilisé conserve les efforts.

D'après la section 2.4.3 le champ de déplacement virtuel $(\delta\omega_{p_i}, \delta\theta_{p_i})$ appliqué au noeud du maillage de la structure peut être propagé à la totalité du maillage du domaine fluide, en particulier à la sous-partie constituée des noeuds du maillage $\mathcal{F}^I$. Le déplacement virtuel $\overrightarrow{\delta G_l}$ associé au barycentre G_l de l'interface I_l est égal à :

$$\overrightarrow{\delta G_l} = \sum_{i=1}^{n_p} f_i(y_l)\left(\delta\omega_{p_i}\vec{z} + \overrightarrow{G_lP_i}\wedge\delta\theta_{p_i}\vec{y}\right)$$

Le travail des efforts aérodynamiques appliqués au maillage structure lors du déplacement virtuel de celle-ci est égal à :

$$\delta W_S = \sum_{i=1}^{n_p}\left(\overrightarrow{\mathcal{F}_i}\cdot\delta\omega_{p_i}\ \vec{z} + \overrightarrow{\mathcal{M}_i}\cdot\delta\theta_{p_i}\ \vec{y}\right) \quad = \sum_{i=1}^{n_p}\sum_{l=1}^{n_f^I}\left(\delta\omega_{p_i}\ S_l^i\ p_l\ (\vec{n}_l\cdot\vec{z}) + \delta\theta_{p_i}\ S_l^i\ p_l\ \left(\overrightarrow{P_iG_l}\wedge\vec{n}_l\right)\cdot\vec{y}\right)\ .$$

Le travail des efforts aérodynamiques sur le maillage de la surface mouillée est quant à lui égal à :

$$\delta W_F = \sum_{l=1}^{n_f^I}\overrightarrow{\delta G_l}\cdot S_l p_l\vec{n}_l = \sum_{l=1}^{n_f^I}\sum_{i=1}^{n_p}\left(\delta\omega_{p_i}\ S_l f_i(y_l)\ p_l\ (\vec{n}_l\cdot\vec{z}) + \delta\theta_{p_i}\ S_l f_i(y_l)\ p_l\ \left(\overrightarrow{G_lP_i}\wedge\vec{y}\right)\cdot\vec{n}_l\right)\ .$$

En utilisant le fait que

$$\left(\overrightarrow{G_lP_i}\wedge\vec{y}\right)\cdot\vec{n}_l = \left(\vec{n}_l\wedge\overrightarrow{G_lP_i}\right)\cdot\vec{y} = \left(\overrightarrow{P_iG_l}\wedge\vec{n}_l\right)\cdot\vec{y},$$

on trouve que

$$\delta W_F = \sum_{l=1}^{n_f^I}\overrightarrow{\delta G_l}\cdot S_l p_l\vec{n}_l = \sum_{l=1}^{n_f^I}\sum_{i=1}^{n_p}\left(\delta\omega_{p_i}\ S_l f_i(y_l)\ p_l\ (\vec{n}_l\cdot\vec{z}) + \delta\theta_{p_i}\ S_l f_i(y_l)\ p_l\ \left(\overrightarrow{P_iG_l}\wedge\vec{n}_l\right)\cdot\vec{y}\right)\ .$$

La condition de conservativité de l'énergie $\delta W_F = \delta W_S$ est en particulier réalisée si la condition suivante est satisfaite

$$\forall \;\; 1 \leq i \leq n_p, \quad S_l f_i(y_l) = S_l^i \quad => f_i(y_l) = \frac{S_l^i}{S_l}.$$

Cette condition n'est évidemment pas satisfaite dans notre cas. En effet, imaginons le cas d'une interface dont l'intersection avec la zone $\mathcal{V}_i$ ne soit pas nulle, mais qui serait suffisamment étirée pour que son centre de gravité G_l soit en dehors de $\mathcal{V}_i$. Par définition $S_l^i \neq 0$ alors que dans notre cas $f_i(y_l) = 0$ (voir la figure 2.3).

2.5 Procédure itérative de calcul de l'équilibre aéroélastique statique

Le comportement du système fluide-structure est décrit par le système d'équations couplées introduit dans la section 2.1 :

$$\left\{ \begin{array}{l} R_f(W, X) = 0 \\ \\ R_s(D, Z) = 0 \end{array} \right. \quad \text{où} \quad \left\{ \begin{array}{l} X(D, Z) \\ \\ L(W, X) \end{array} \right. \tag{2.6}$$

Le système d'équations R_f décrivant l'état du fluide est non-linéaire tandis que le système d'équations décrivant le déplacement de la structure est linéaire. Le système d'équations couplées décrivant l'état du système couplé est donc non-linéaire. Dans le cadre de cette étude et dans la grande majorité des problèmes aéroélastiques, les codes de calcul utilisés pour résoudre les équations associées à chacune des disciplines sont distincts et le système couplé est résolu de manière itérative.

Dans le cadre de ce travail, la technique itérative choisie est une technique de point fixe. Rappelons avant de la décrire que l'hypothèse des petits déplacements et petites déformations a été faite de telle sorte que le maillage de la structure n'est pas modifié au cours du processus itératif conduisant à l'équilibre aéroélastique ($Z = Z_{rig}$).

Précisons aussi les dépendances du systèmes couplés. D'après la section 2.1, les équations du système sont implicitement couplées par l'intermédiaire des coordonnées du maillage du domaine fluide X et du chargement aérodynamique transmis à la structure L. D'après la section 2.4.3, la technique de remaillage est telle que $X = X\left(X_{rig}, Z, D\right) = X\left(X_{rig}, Z_{rig}, D\right)$. La position d'équilibre aéroélastique du système couplé fluide-structure est calculée de la manière suivante (l'exposant (q) représente l'itération numéro q du processus de résolution) :

1. initialisation de la procédure itérative : le déplacement de la structure est supposé identiquement nul $D^{(0)} = 0$ de tel sorte que le maillage du domaine fluide coïncide avec le maillage de la forme rigide non déformée sous les effets aéroélastiques $X = X_{rig}$,
2. les grandeurs conservatives associées à l'écoulement fluide $W^{(q)}$ sont calculées sur le maillage $X^{(q)}\left(X_{rig}, Z_{rig}, D^{(q)}\right)$,
3. le chargement aérodynamique $L^{(q)}$ est calculé et transmis à la structure (voir la section 2.4.4),
4. le champ de déplacement induit par le chargement aérodynamique sur la structure est calculé $D^{(q+1)}$,
5. le maillage du domaine fluide est déformé, les nouvelles coordonnées du maillage satisfont $X^{(q+1)}\left(X_{rig}, Z_{rig}, D^{(q+1)}\right)$ (voir la section 2.4.3),
6. les grandeurs conservatives associées à l'écoulement fluide $W^{(q+1)}$ sont évaluées sur le maillage $X^{(q+1)}$,

7. les critères de convergence sont évalués (ϵ_f et ϵ_s sont deux seuils de tolérance fixés au préalable)

$$\begin{cases} ||R_f||_2 \leq \epsilon_f \\ \\ ||D^{(q+1)} - D^{(q)}||_2 \leq \epsilon_s ||D^{(q)}||_2 \end{cases}, \tag{2.7}$$

8. si les critères de convergence sont satisfaits la boucle itérative s'arrête, sinon elle repart à l'étape 2.

2.6 Présentation des cas tests

2.6.1 Aile-Fuselage F4

Les développements relatifs au calcul de gradient d'un système couplé fluide structure utilisant l'hypothèse de fluide parfait (équations d'Euler) ont été validés grâce à la configuration aile-fuselage F4 définie au DLR [178].

Le maillage du domaine fluide est aux dimensions de la maquette. Il est constitué de deux blocs et comporte au total 313.650 noeuds. Les conditions de vol sont celles de la croisière. L'écoulement est en particulier symétrique, de telle sorte que seul le domaine fluide situé à droite de la maquette est discrétisé. Les différentes conditions de vol qui ont été considérées sont résumées dans le tableau 2.1. En particulier, plusieurs incidences ont été considérées dans le but de déterminer l'influence de paramètres sur la vitesse et la qualité de la convergence du processus itératif de calcul de la position d'équilibre aéroélastique.

Le support du maillage de la structure est situé au centre de l'aile et est linéaire par mor-

Incidence	$-1°$, $0.93°$, $3°$
Densité à l'infini amont	$9.015\ kg/m^3$
Pression statique à l'infini amont	$6.977 \cdot 10^4\ N/m^2$
Nombre de Mach à l'infini amont	0.75

TAB. 2.1 – Conditions de vol de la configuration aile-fuselage F4.

ceaux. Il est constitué de 250 noeuds. Les caractéristiques de la structure sont résumées dans le tableau 2.2. Les moments quadratiques I et J relatifs respectivement aux déplacements en flexion et en torsion de la poutre ont été calculés en supposant que les sections transverses de l'aile en envergure étaient pleines et constituées d'un matériau isotrope (aluminium). Les variations de ces moments en envergure sont décrites par les figures 2.8 et 2.9 (l'envergure y est normalisée par la demi-envergure des ailes). Ces données mécaniques sont issues des tests aéroélastiques effectués en soufflerie sur la configuration aile-fuselage HiReTT décrite dans la section 2.6.3. Elles ne correspondent pas physiquement au comportement structurel de la configuration aile-fuselage F4, ce qui n'a pas d'importance quant à la validation numérique des calculs de gradients effectués dans le chapitre suivant.

Le maillage de la surface mouillée totale (aile et fuselage) et le maillage de la structure,

Module d'Young	$E = 1.813 \cdot 10^{11}$	N/m^2
Module d'élasticité de glissement	$G = 0.682 \cdot 10^{11}$	N/m^2
Coefficient de Poisson	$\nu = 0.33$	

TAB. 2.2 – Caractéristiques mécaniques du matériau dont est composée la structure.

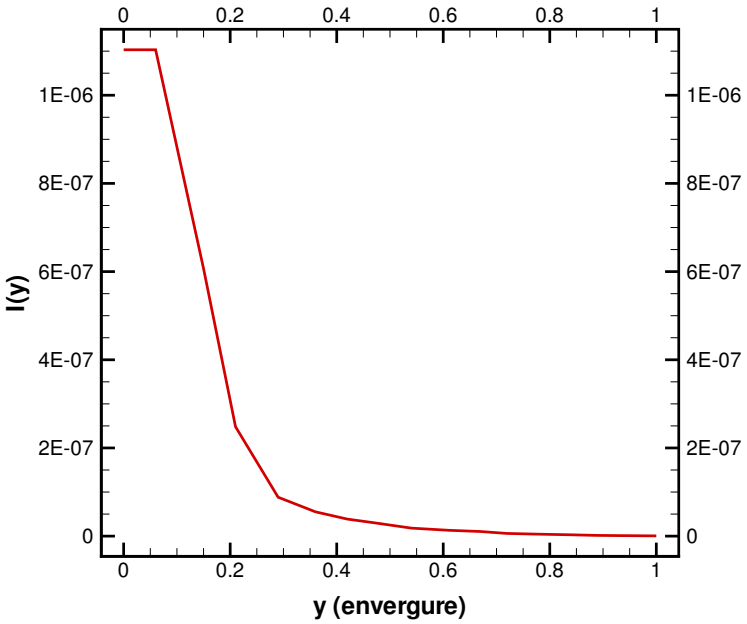

FIG. 2.8 – Moment quadratique d'une section de poutre autour de l'axe de flexion (noté $I = \int_{\text{section}} [dist(P, \text{axe de flexion})]^2 dS$ et exprimé en m^4)

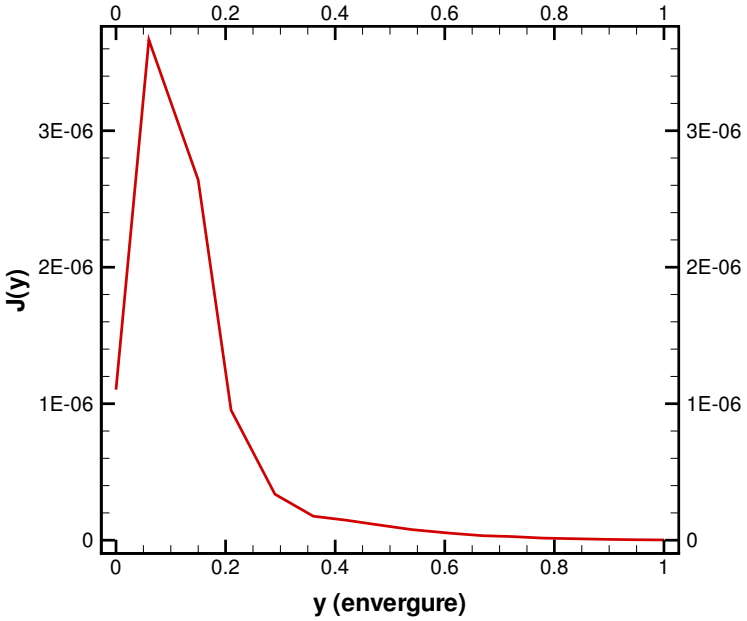

FIG. 2.9 – Moment quadratique d'une section de poutre autour de l'axe principal (noté $J = \int_{\text{section}} [dist(P, \text{axe principal})]^2 dS$ et exprimé en m^4)

c'est-à-dire de la poutre simulant son comportement, sont représentés par les figures 2.10 et 2.11. Seule l'aile se déforme sous l'effet du chargement aérodynamique. Cependant, comme le montrent ces figures, le maillage de la structure va au delà des limites de l'aile en envergure. Cette non-

coïncidence permet de faire en sorte que la totalité du chargement aérodynamique soit transmis à la structure représentant l'aile. Les noeuds de la structure pour lesquels le chargement aérodynamique est nul ne se déplaceront pas et ne nuiront donc pas à la simulation du comportement structurel de l'aile.

Après six échanges des efforts et des déplacements des codes de simulation pour la mécanique

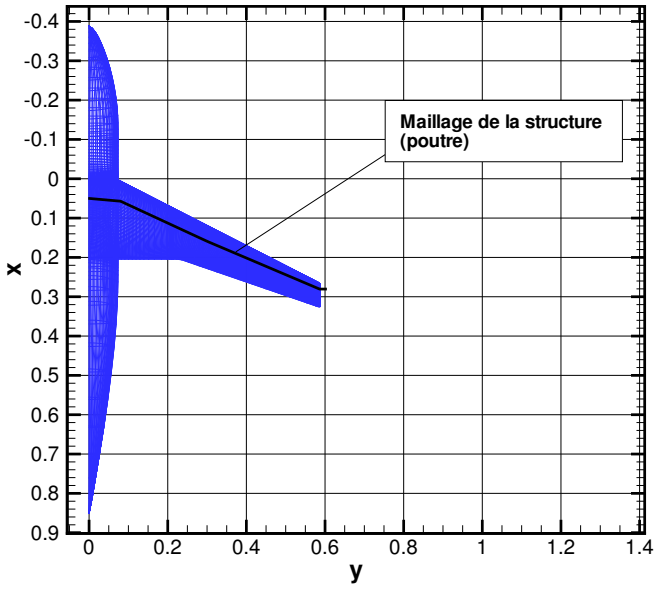

FIG. 2.10 – Configuration aile-fuselage F4 (rigide) vue du dessus.

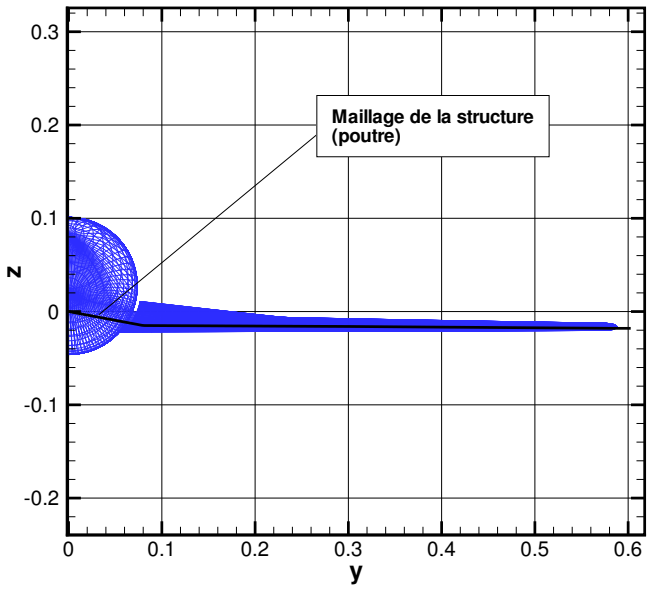

FIG. 2.11 – Configuration aile-fuselage F4 (rigide) vue de derrière.

des fluides et pour la mécanique des structures, la procédure itérative décrite dans la section 2.5 est convergée avec $\epsilon_f = 10^{-3}$ et $\epsilon_s = 10^{-4}$ (en utilisant les notations de l'équation (2.7)). L'équilibre aéroélastique statique a été trouvé. La position de l'aile à l'équilibre aéroélastique est présentée par les figures 2.12, 2.13, 2.14, 2.15 et 2.16. et les champs de déplacements de la structure correspondants (flexion et torsion) sont représentés par les figures 2.18 et 2.17. En croisière, à l'équilibre aéroélastique, le calcul prédit un dévrillage de l'aile accompagné d'un déplacement en flexion positif (vers le haut), ce qui est en accord avec les observations faites sur les ailes à flêche positive.

Les courbes de convergence de la norme L_2 du résidu des équations aérodynamiques R_f au

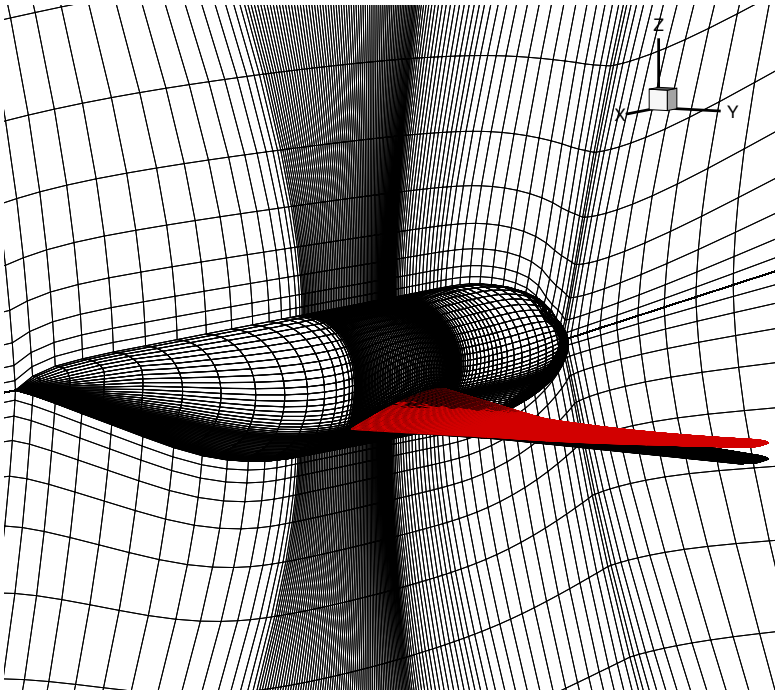

FIG. 2.12 – Comparaison entre la position de l'aile sur la configuration rigide (noir) et à l'équilibre aéroélastique (rouge) pour la configuration aile-fuselage F4

cours du processus itératif de calcul de l'équilibre aéroélastique décrit dans la section 2.5 pour les trois conditions de vol considérées (incidences) sont représentées par les figures 2.19, 2.20 et 2.21. La convergence du champ de déplacement de la structure au cours du processus itératif est illustrée par les figures 2.23 et 2.22. Ces figures représentent respectivement le déplacement en flexion et en torsion au saumon, point où le déplacement de l'aile atteint son maximum. Pour les trois incidences selectionnées d'après les courbes de convergence de $||R_f||_2$ présentées, l'incidence n'est pas un facteur déterminant de la convergence du calcul de l'équilibre aéroélastique.

La comparaison des coefficients aérodynamiques de portance et de traînée aux différentes conditions de vol calculés en prenant ou non en compte les effets aéroélastiques est présentée dans les tableaux 2.3 et 2.4. Pour ces conditions de vol, les couplages fluide/structure au niveau de l'aile conduisent à diminuer la portance et la traînée.

2.6.2 Aile M6

Ce cas test a été utilisé pour valider les développements relatifs au calcul du gradient d'un système couplé fluide-structure pour un fluide visqueux, et un écoulement turbulent modélisé par les équations RANS.

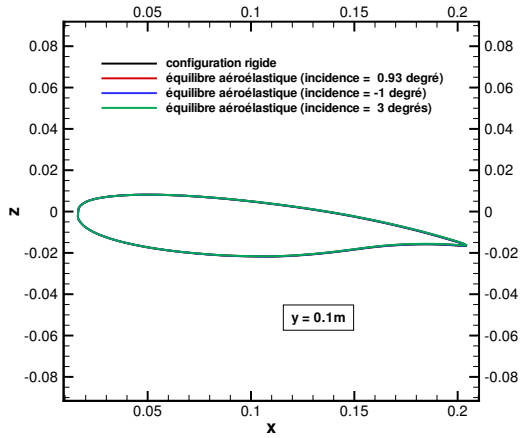

FIG. 2.13 – Coupe à l'emplanture (configuration aile-fuselage F4).

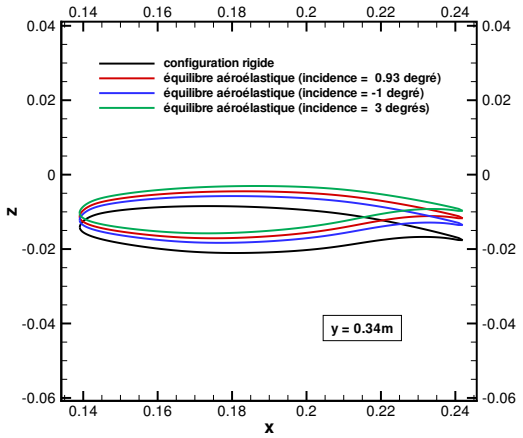

FIG. 2.14 – Coupe à mi-envergure (configuration aile-fuselage F4).

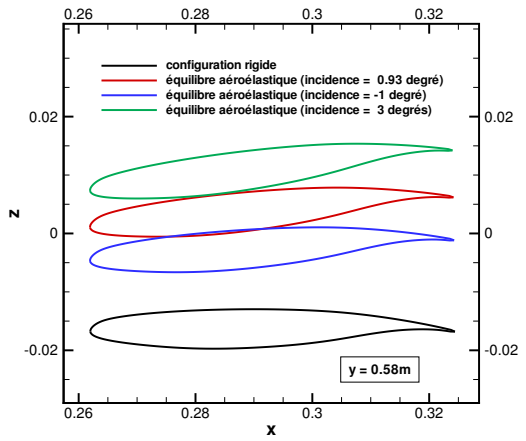

FIG. 2.15 – Coupe au saumon (configuration aile-fuselage F4).

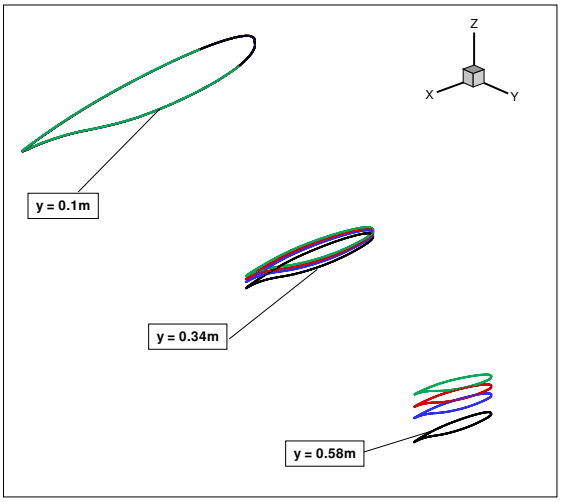

FIG. 2.16 – Vue générale des trois coupes (configuration aile-fuselage F4).

	Calcul aérodynamique seul	Calcul aéroélastique
$-1°$	0.554	0.454
0.93°	0.825	0.760
3°	1.073	0.911

TAB. 2.3 – Coefficient aérodynamique de portance (Cl_p) aux différentes conditions de vol pour la configuration aile-fuselage F4.

Le maillage du domaine fluide est adimensionné par la corde à l'emplanture (la longueur de référence pour le calcul adimensionné est $L = 1$ à l'emplanture). Il est mono-bloc et comporte 614.705 noeuds. Le calcul de l'écoulement est également effectué en grandeurs adimensionnées. Les grandeurs de références sont calculées à partir de la densité à l'infini amont et de la vitesse à l'infini amont. En particulier, le calcul de l'écoulement dans la veine d'essai se fait en considérant un écoulement infini amont tel que $\rho_\infty = 1$ et$V_\infty = 1$. L'incidence ainsi que les nombres de

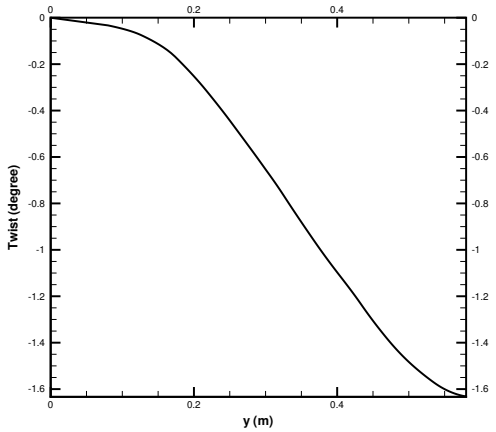

FIG. 2.17 – Champ de déplacement en torsion sur la poutre à l'équilibre aéroélastique (configuration aile-fuselage F4).

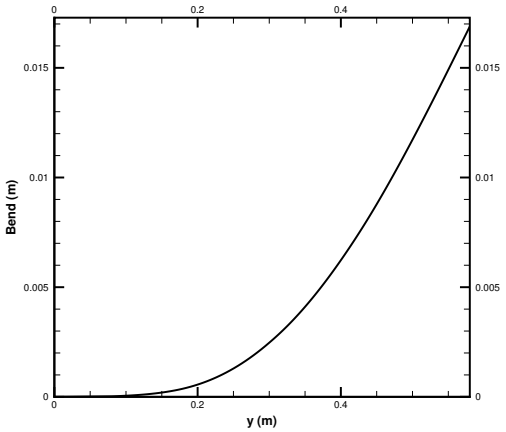

FIG. 2.18 – Champ de déplacement en flexion sur la poutre à l'équilibre aéroélastique (configuration aile-fuselage F4).

	Calcul aérodynamique seul	Calcul aéroélastique
$-1°$	0.0238	0.0205
$0.93°$	0.0409	0.0344
$3°$	0.0834	0.0542

TAB. 2.4 – Coefficient aérodynamique de traînée (Cd_p) aux différentes conditions de vol pour la configuration aile-fuselage F4.

Mach et de Reynolds à l'infini amont sont donnés par le tableau 2.5. Le fluide est visqueux, son écoulement est turbulent et modélisé par les équations RANS avec pour modèle de turbulence le modèle k-w Wilcox.

Le support du maillage de la structure est situé au centre de l'aile et est linéaire par mor-

Incidence	$3.06°$
Nombre de Mach à l'infini amont	0.84
Nombre de Reynolds ($Re = \frac{\rho_\infty V_\infty L}{\mu_\infty} = \frac{1}{\mu_\infty}$)	$1.4 \cdot 10^7$

TAB. 2.5 – Conditions de vol associées à la configuration aile M6.

ceaux. Son amplitude en envergure est légérement supérieur à la demi-envergure des ailes de sorte que la totalité du chargement aérodynamique soit pris en compte. Il est constitué de 320 noeuds. Les caractéristiques de la structure sont identiques à celles de la configuration aile-fuselage F4 (voir le tableau 2.2), et les moments quadratiques en flexion et en torsion obéissent à la même loi que ceux utilisés pour la configuration aile-fuselage F4 (voir les figures 2.8 et 2.9). Ces données mécaniques sont issues des tests aéroélastiques effectués en soufflerie sur la configuration aile-fuselage HiReTT décrite dans la section 2.6.3. Elles ne correspondent pas physiquement au comportement structurel de l'aile M6 expérimentale, ce qui n'a pas d'importance au regard de la validation numérique des calculs de gradients avec modélisation aéroélastique effectués dans le chapitre suivant. Le maillage de la surface de l'aile et de la structure sont représentés par la figure 2.24.

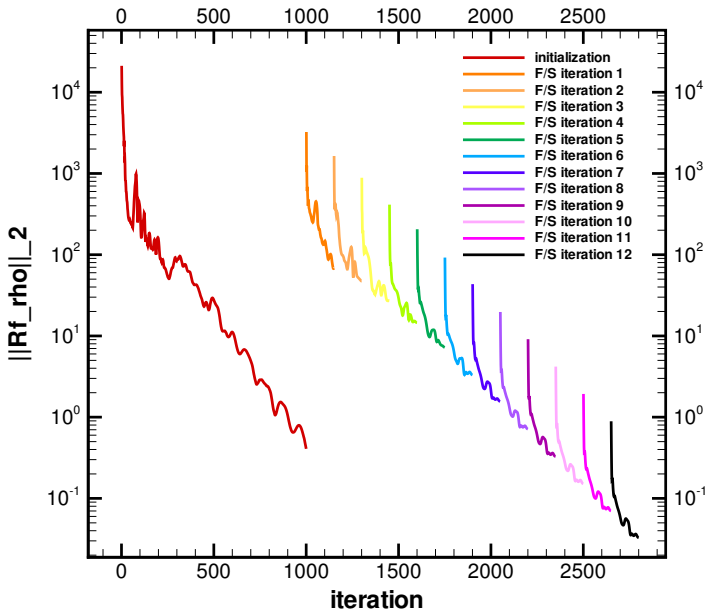

FIG. 2.19 – Convergence de la norme L_2 du résidu de l'équation de conservation de la masse au cours du processus itératif de calcul de l'équilibre aéroélastique (0.93° d'incidence) pour la configuration aile-fuselage F4.

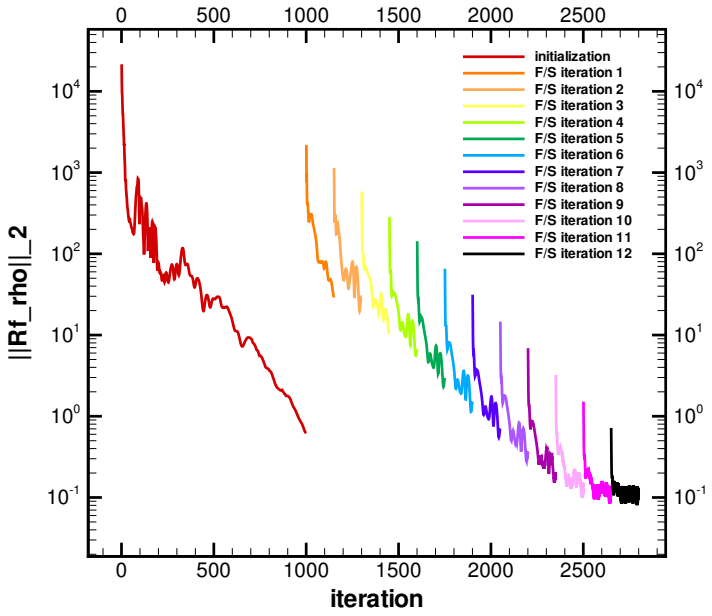

FIG. 2.20 – Convergence de la norme L_2 du résidu de l'équation de conservation de la masse au cours du processus itératif de calcul de l'équilibre aéroélastique ($-1°$ d'incidence) pour la configuration aile-fuselage F4.

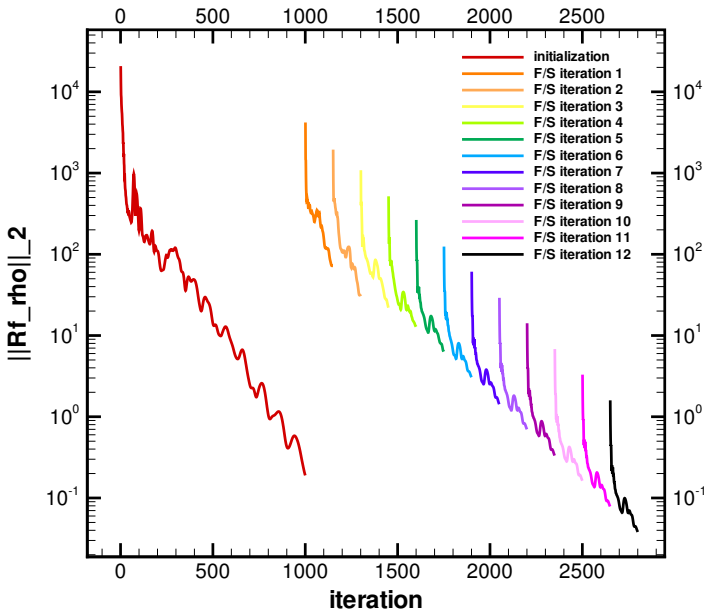

FIG. 2.21 – Convergence de la norme L_2 du résidu de l'équation de conservation de la masse au cours du processus itératif de calcul de l'équilibre aéroélastique (3° d'incidence) pour la configuration aile-fuselage F4.

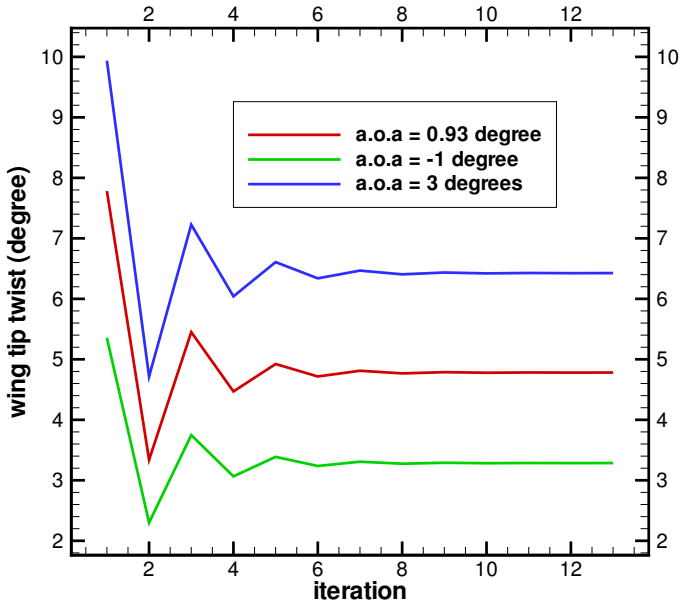

FIG. 2.22 – Convergence du déplacement en torsion au saumon au cours du processus itératif de calcul de l'équilibre aéroélastique pour les trois incidences considérées (configuration aile-fuselage F4).

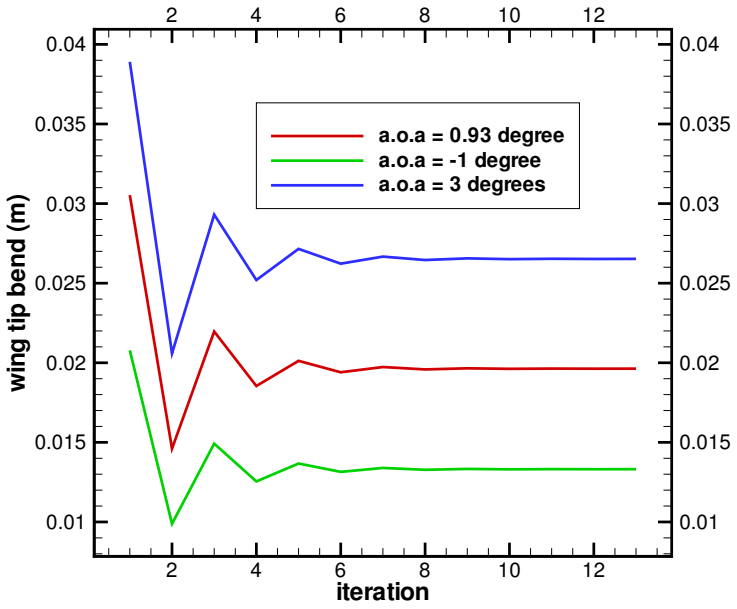

FIG. 2.23 – Convergence de la flexion au saumon au cours du processus itératif de calcul de l'équilibre aéroélastique pour les trois incidences considérées (configuration aile-fuselage F4).

Le processus de calcul de l'équilibre aéroélastique statique converge en une dizaine d'itéra-

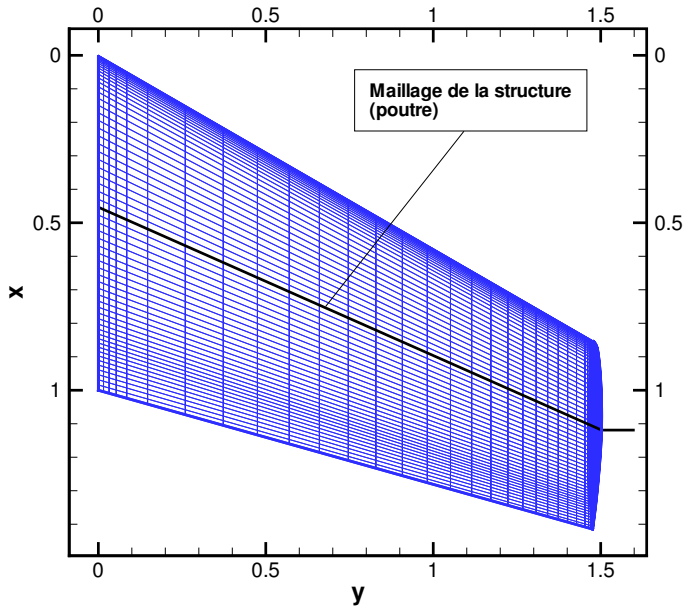

FIG. 2.24 – Configuration aile M6 (rigide).

tions avec $\epsilon_f = 10^{-5}$ et $\epsilon_s = 10^{-4}$ (en utilisant les notations de l'équation (2.7)). Les courbes de convergence de la norme L_2 de la première équation (conservation de la masse) du système des équations aérodynamiques écrites sous forme de résidu et des déplacements en flexion et en

torsion du saumon de l'aile (point auquel les effets aéroélastiques sont les plus importants) sont représentées par les figures 2.32, 2.33 et 2.34. Les figures 2.25, 2.26, 2.27, 2.28 et 2.29 illustrent l'écart entre la position de l'aile rigide et de l'aile souple soumise aux effets aéroélastiques.

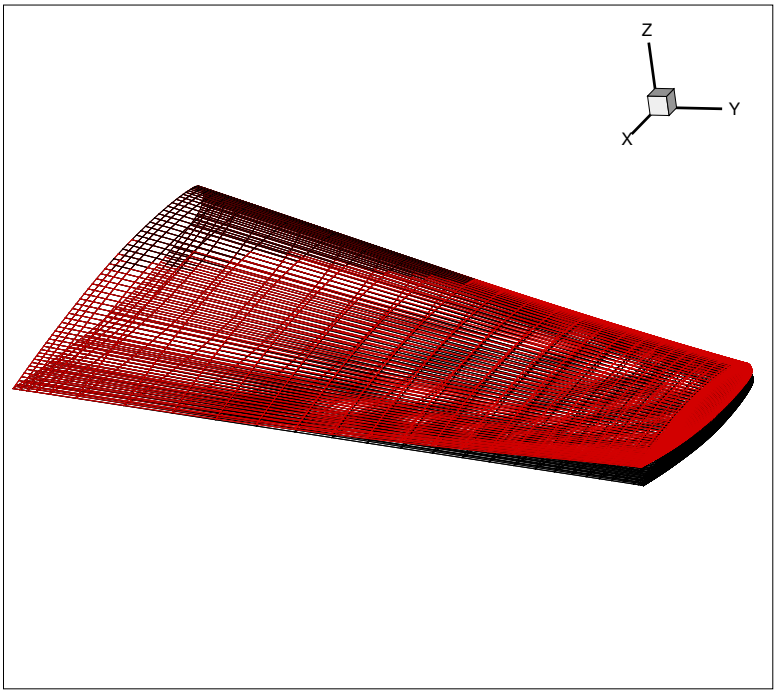

FIG. 2.25 – Comparaison entre la position de l'aile sur la configuration rigide (noir) et à l'équilibre aéroélastique (rouge) pour la configuration aile M6.

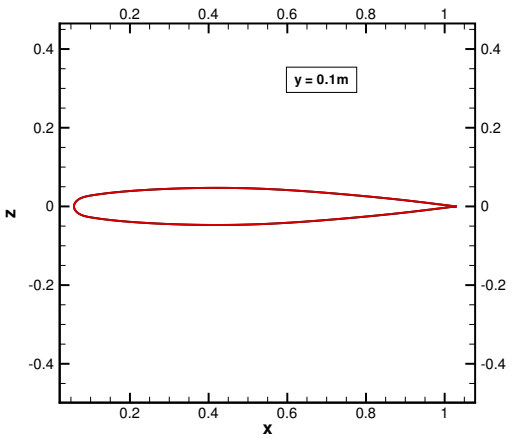

FIG. 2.26 – Coupe à l'emplanture (configuration aile M6).

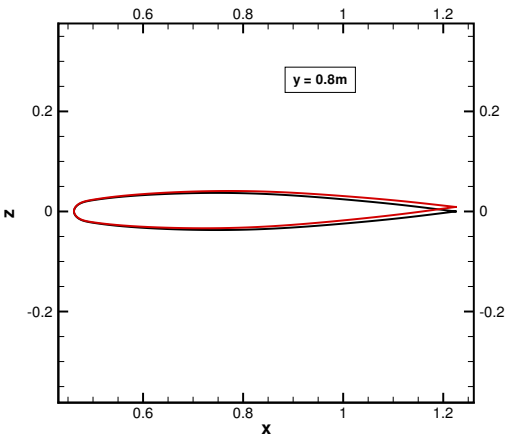

FIG. 2.27 – Coupe à mi-envergure (configuration aile M6).

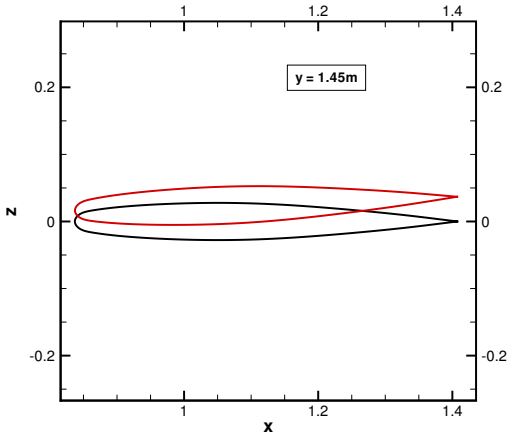

FIG. 2.28 – Coupe au saumon (configuration aile M6).

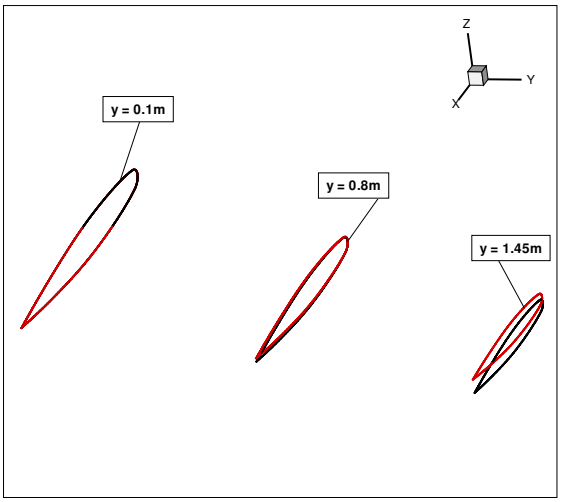

FIG. 2.29 – Vue générale des trois coupes (configuration aile M6).

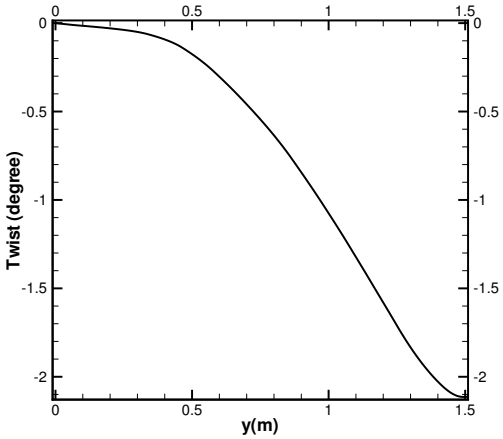

FIG. 2.30 – Champ de déplacement en torsion sur la poutre à l'équilibre aéroélastique (configuration aile M6).

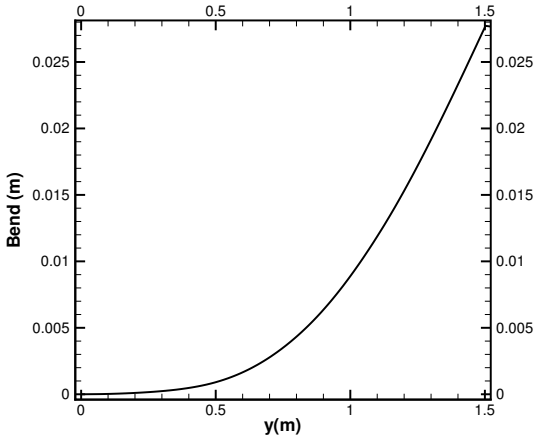

FIG. 2.31 – Champ de déplacement en flexion sur la poutre à l'équilibre aéroélastique (configuration aile M6).

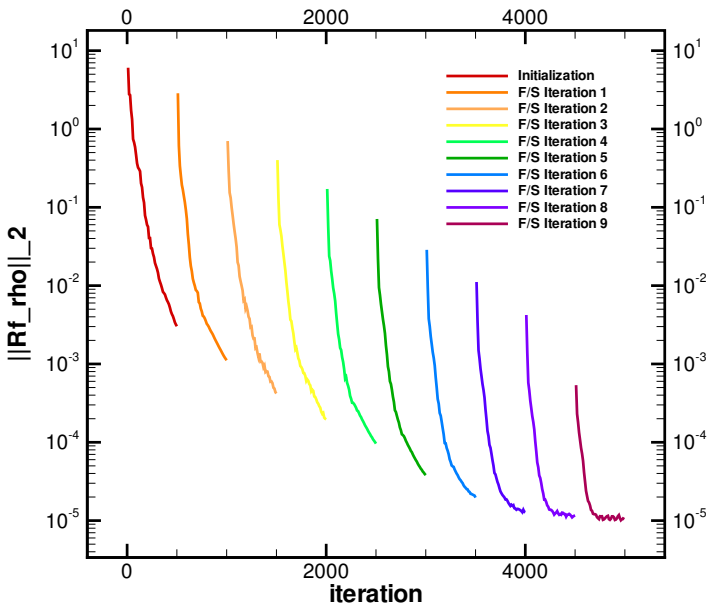

FIG. 2.32 – Aile M6 - Convergence de la norme L_2 du résidu de l'équation de conservation de la masse au cours du processus itératif de calcul de l'équilibre aéroélastique (configuration aile M6).

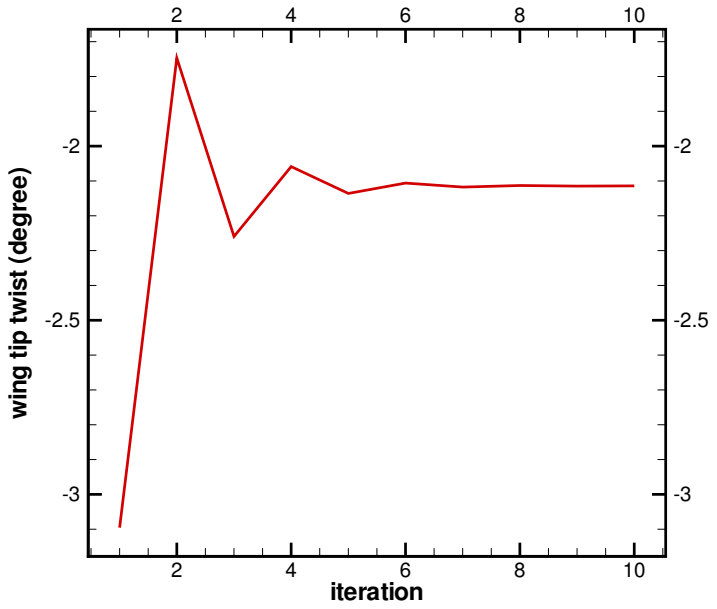

FIG. 2.33 – Convergence du déplacement en torsion au saumon au cours du processus itératif de calcul de l'équilibre aéroélastique (configuration aile M6).

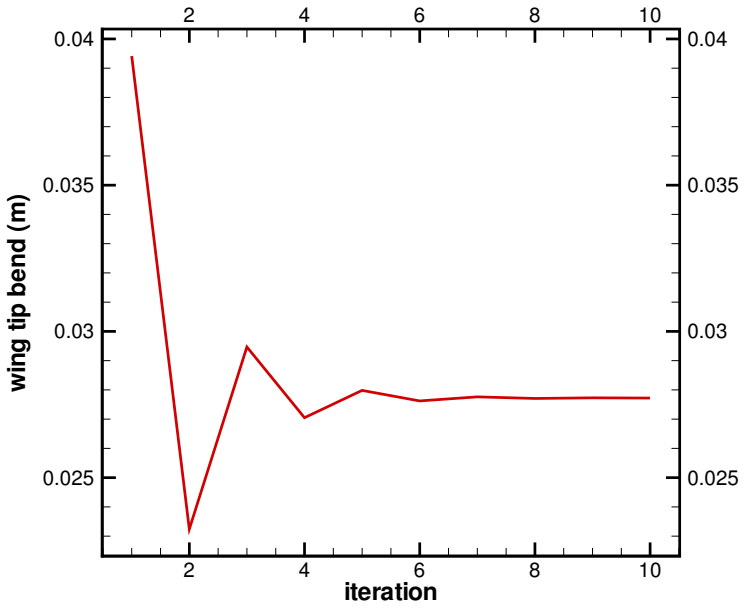

FIG. 2.34 – Aile M6 - Convergence de la flexion au saumon au cours du processus itératif de calcul de l'équilibre aéroélastique (configuration aile M6).

Comme pour la configuration aile-fuselage F4 présentée dans la section précédente, la comparaison des coefficients aérodynamiques de portance et de traînée calculés en prenant ou non en compte les effets aéroélastiques montre que pour la condition de vol considérée, le couplage fluide/structure induit une diminution de la portance et de la traînée produite (voir le

tableau 2.6).

	Calcul aérodynamique seul	Calcul aéroélastique
Cl	0.26783	0.22173
Cd	0.01526	0.01153

TAB. 2.6 – Coefficient aérodynamique de portance pour la configuration aile M6.

2.6.3 Aile-Fuselage HiReTT

Ce cas test [31, 33, 32] a été utilisé pour valider les développements relatifs au calcul du gradient d'un système couplé fluide-structure pour un fluide visqueux et un écoulement turbulent modélisé par les équations RANS. Il s'agit de la configuration étudiée par le projet "*European High Reynolds-Number Tools and Technique (HiReTT)*".

La forme solide autour de laquelle est construit le maillage du domaine fluide est aux dimensions de la maquette (au $1/50e$). Il comporte 1.900.000 noeuds et est constitué de 59 blocs, certains raccords entre blocs étant non-coïncidents. Les conditions de vol à l'infini amont sont reportées dans le tableau 2.7. Comme précisé ci-dessus le fluide est visqueux, son écoulement est turbulent et modélisé par les équations RANS avec pour modèle de turbulence le modèle de Spalart-Allmaras.

Le support du maillage de la structure est situé au centre de l'aile et est linéaire par mor-

Incidence	$1.5652°$
Température statique à l'infini amont	$100\ K$
Densité à l'infini amont	$5.495\ kg/m^3$
Pression statique à l'infini amont	$1.577 \cdot 10^5\ N/m^2$
Nombre de Mach à l'infini amont	0.85
Nombre de Reynolds	$32.5 \cdot 10^6$
$q/E = \frac{1}{2}\gamma p_\infty M_\infty^2/E$	$0.44 \cdot 10^{-6}$

TAB. 2.7 – Conditions d'essai associées à la configuration aile-fuselage HiReTT.

ceaux. Son amplitude en envergure est légérement supérieur à la demi-envergure des ailes de sorte que la totalité du chargement aérodynamique soit pris en compte. Il est constitué de 425 noeuds. Les caractéristiques de la structure sont résumées dans le tableau 2.2. Les moments quadratiques en flexion et en torsion sont décrits par les figures 2.8 et 2.9. Les valeurs de ces moments ont été extraites du rapport [31] où elles ont été calculées expérimentalement grâce à des essais en flexion et traction effectués à différentes stations de l'aile en envergure. Le maillage de la surface mouillée (de la forme rigide) par le fluide et le maillage de la structure sont représentés par les figures 2.35 et 2.36.

Le processus de calcul de l'équilibre aéroélastique statique converge en une dizaine d'itérations avec $\epsilon_f = 10^{-4}$ et $\epsilon_s = 10^{-5}$ (en utilisant les notations de l'équation (2.7)). Les courbes de convergence de la première équation (conservation de la masse) du système des équations aérodynamiques écrites sous forme de résidu et des déplacements en flexion et en torsion du saumon de l'aile (point auquel les effets aéroélastiques sont les plus importants) sont représentées par les figures 2.44, 2.46 et 2.45. Les figures 2.37, 2.38, 2.39, 2.40 et 2.41 illustrent la différence entre la position de l'aile rigide et de l'aile souple soumise aux effets aéroélastiques.

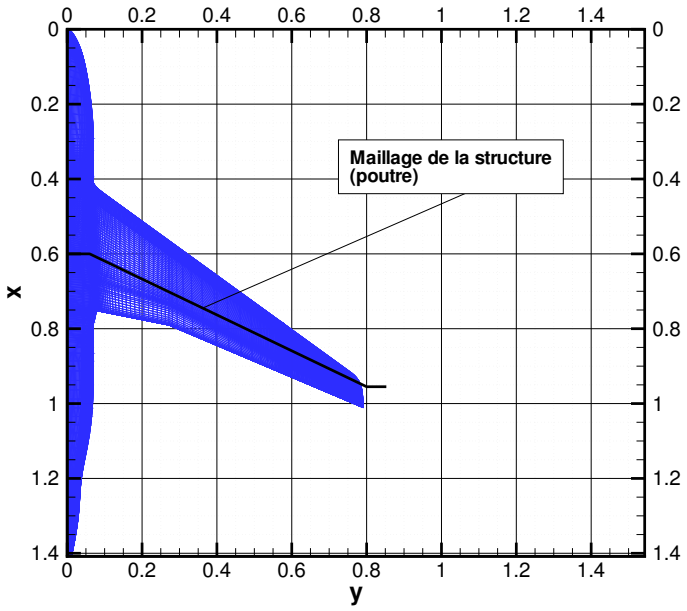

FIG. 2.35 – Configuration aile-fuselage HiReTT (rigide) vue du dessus.

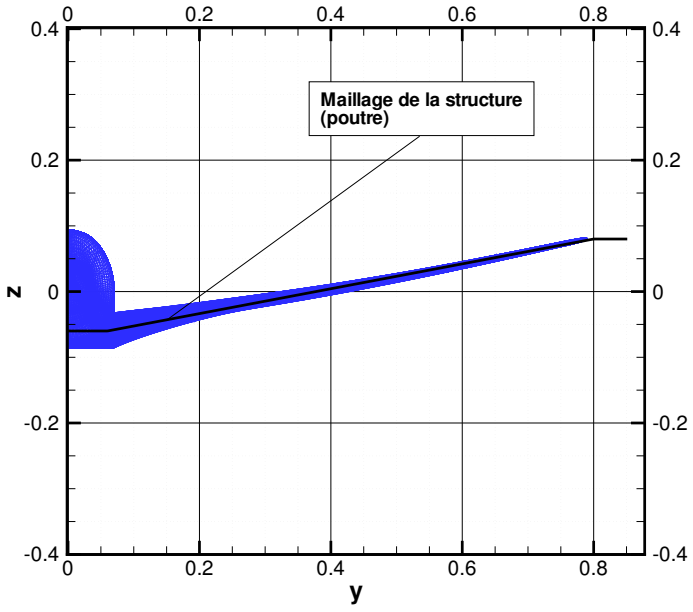

FIG. 2.36 – Configuration aile-fuselage HiReTT (rigide) vue de derrière.

Comme pour les deux cas tests précédents, la comparaison des coefficients aérodynamiques de portance et de traînée calculés en prenant ou pas en compte les effets aéroélastiques montre qu'à ce point de vol, les effets aéroélastiques ont tendance à réduire ces coefficients (voir le tableau 2.6).

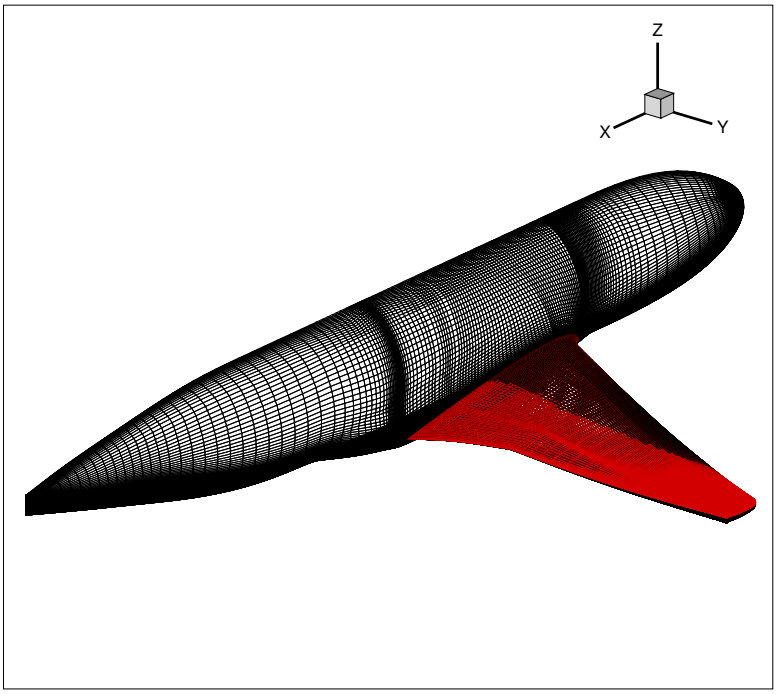

FIG. 2.37 – Comparaison entre la position de l'aile sur la configuration rigide (noir) et à l'équilibre aéroélastique (rouge) pour la configuration aile-fuselage HiReTT

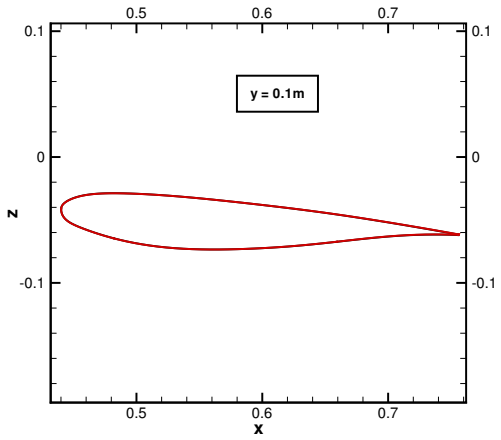

FIG. 2.38 – Coupe à l'emplanture (configuration aile-fuselage HiReTT).

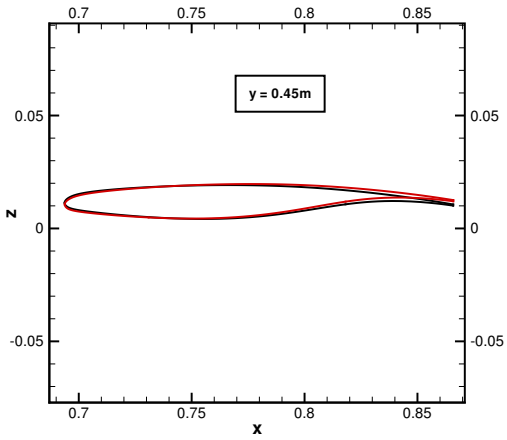

FIG. 2.39 – Coupe à mi-envergure (configuration aile-fuselage HiReTT).

	Calcul aérodynamique seul	Calcul aéroélastique
Cl	0.0717	0.0623
Cd	0.0052	0.0046

TAB. 2.8 – Coefficient aérodynamique de portance aux différentes conditions de vol pour la configuration aile-fuselage HiReTT.

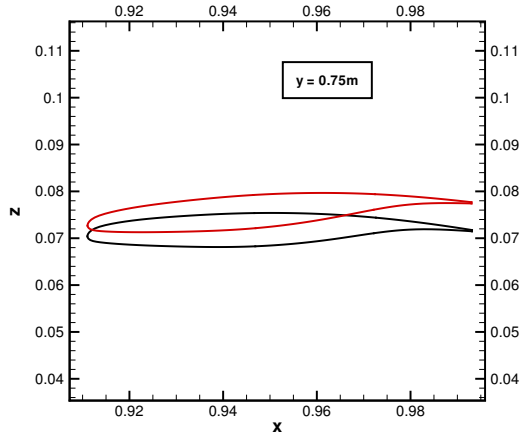

FIG. 2.40 – Coupe au saumon (configuration aile-fuselage HiReTT).

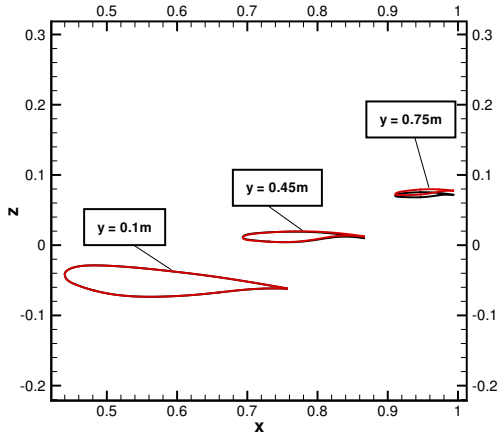

FIG. 2.41 – Vue générale des trois coupes (configuration aile-fuselage HiReTT).

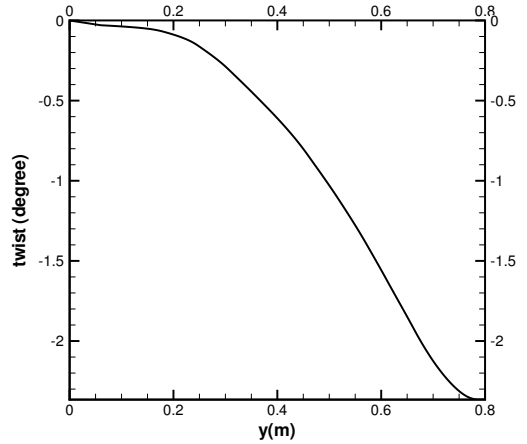

FIG. 2.42 – Champ de déplacement en torsion sur la poutre à l'équilibre aéroélastique (configuration aile-fuselage HiReTT).

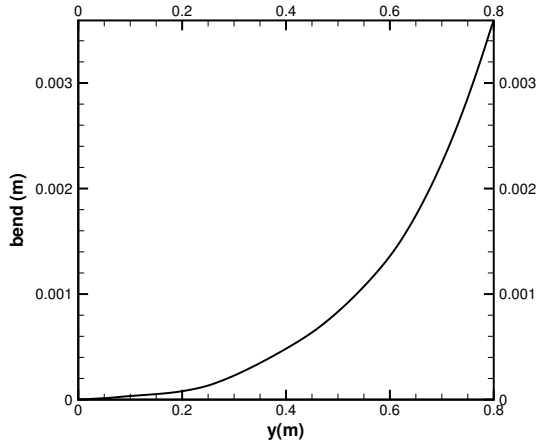

FIG. 2.43 – Champ de déplacement en flexion sur la poutre à l'équilibre aéroélastique (configuration aile-fuselage HiReTT).

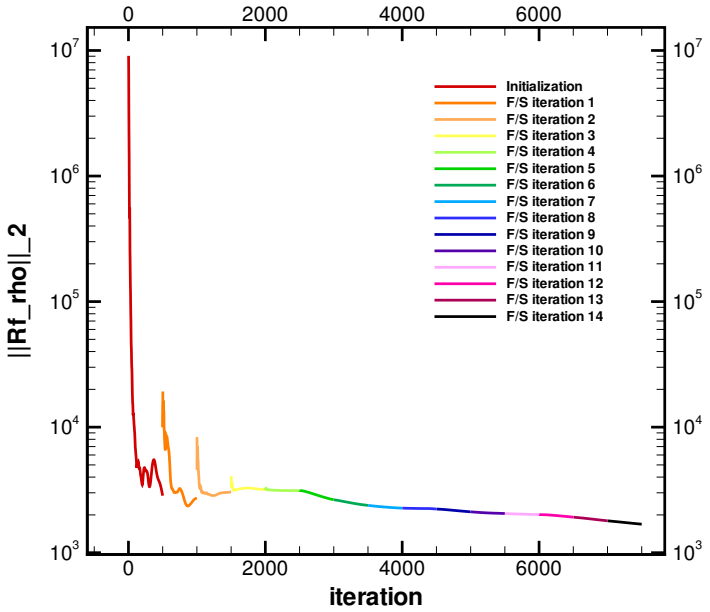

FIG. 2.44 – Convergence de la norme L_2 du résidu de l'équation de conservation de la masse au cours du processus itératif de calcul de l'équilibre aéroélastique (configuration aile-fuselage HiReTT).

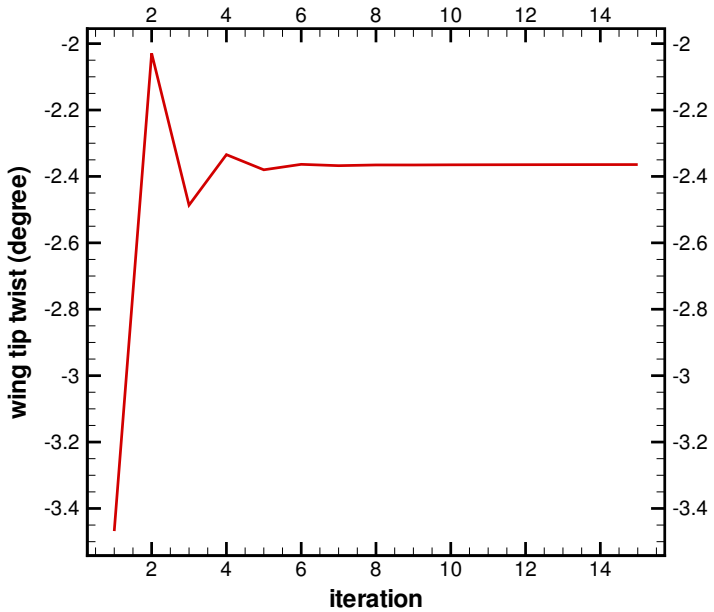

FIG. 2.45 – Convergence du champ de déplacement en torsion au saumon au cours du processus itératif de calcul de l'équilibre aéroélastique (configuration aile-fuselage HiReTT).

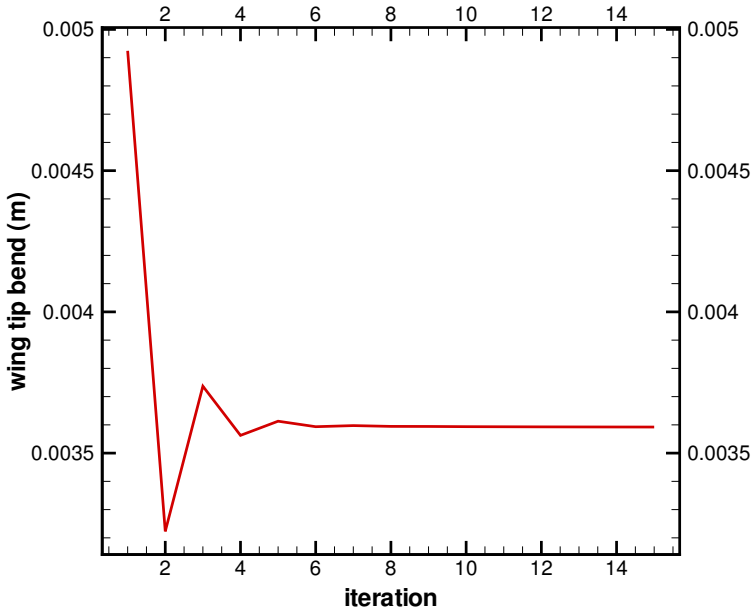

FIG. 2.46 – Convergence du champ de déplacement en flexion au saumon au cours du processus itératif de calcul de l'équilibre aéroélastique (configuration aile-fuselage HiReTT).

Chapitre 3

Calcul des gradients

3.1 Optimisation de forme aérodynamique avec modélisation aéroélastique

Ce chapitre s'intéresse au calcul des gradients nécessaires aux méthodes de descente pour l'optimisation de forme locale (voir section 1.2.3) d'objets aéronautiques pour lesquels les interactions fluide-structure sont prises en compte.

Le problème d'optimisation de forme introduit au chapitre 1 :

$$\textit{trouver } \alpha \in \mathcal{D} \subset R^n \textit{ tel que } J(\alpha) \textit{ soit minimale sur } \mathcal{D} \textit{ et que } \forall k \in [1,p]\ G_k(\alpha) \leq 0$$

se rattache au domaine de l'optimisation sur un espace de paramètres d'une fonction faisant intervenir la solution discrète d'une équation différentielle. Le paramètre figurant dans la fonction à optimiser et dans les fonctions contraintes, via le maillage et les variables d'état couplées par les équations discrètes, le calcul des gradients requis par le processus d'optimisation considéré n'est pas élémentaire.

Les premières optimisations numériques de forme aérodynamique par gradients, dont l'essor a été conditionné par l'utilisation de codes de simulations pour la résolution de la mécanique des fluides (CFD) [170, 101, 100, 102, 103, 99, 217], faisaient appel à la méthode des différences finies pour évaluer les gradients. Les gradients utilisés aujourd'hui pour l'optimisation numérique de forme peuvent être calculés de façon "exacte" par la résolution d'un système d'équations linéaires issu du système non-linéaire des équations différentielles discrétisées dont sont solutions les variables d'état du système. Si ce dernier est obtenu de façon directe par différentiation des équations d'état du système, on parle de la méthode de l'équation linéarisée discrète ou de méthode directe discrète ("*discrete direct differentiation method*" en anglais). En revanche, s'il est obtenu en passant par l'espace dual par introduction des vecteurs adjoints aux équations, on parle de la méthode de l'équation adjointe discrète ("*discrete adjoint method*" en anglais). Bien que moins intuitive, c'est cette dernière qui, par diffusion du domaine du contrôle optimal [131], a d'abord été développée et appliquée à des problèmes simplifiés [170, 150, 171]. C'est cependant à l'article de A. Jameson [113] écrit en 1988 et présentant la méthode de l'équation adjointe continue avec transformations de coordonnées, pour les équations d'Euler, que les publications récentes dans le domaine du calcul des gradients pour l'optimisation de forme aérodynamique font référence comme la première contribution au domaine. La méthode de l'équation linéarisée a ensuite été étudiée dans le contexte de l'aérodynamique [199, 201, 20].

Traditionnellement, dans le contexte de l'optimisation numérique de forme, les fonctions objectif et contraintes n'étaient évaluées qu'à partir de considérations aérodynamiques, c'est-à-dire sans prise en compte de l'interaction de cette dernière avec les autres disciplines, les fonctions considérées étant de nature purement aérodynamique (traînée, portance, moment de tangage

etc). Les autres disciplines étaient traitées séparément selon un processus d'optimisation et des critères spécifiques (optimisation séquentielle - section 1.4.2). Il a cependant été prouvé que ce type de méthodologie qui ne tient pas compte de façon directe des interactions entre les différentes disciplines ne converge pas nécessairement vers la solution optimale du problème multidisciplinaire [220]. Dans un contexte industriel, l'optimisation de forme s'intéresse à des systèmes caractérisés par des interactions complexes. Le couplage aéroélastique fait partie des interactions les plus importantes. Le négliger peut par exemple conduire à une surestimation allant jusqu'à 50% de l'efficacité des ailerons [72], ou comme nous l'avons mis en évidence sur la configuration aile-fuselage F4 développée au DLR (voir la section 2.6.1) à une sur-estimation de la portance et de la traînée allant respectivement jusqu'à 10 et 20%. Du point de vue du calcul de gradient, Arslan et Carlson [9] ont montré que négliger les effets aéroélastiques, lors de l'optimisation de la voilure d'un avion de transport transsonique, pouvaient fournir des gradients d'amplitude et même de signe contraire. Dès lors la motivation à l'origine de la prise en compte des interactions aéroélastiques lors du calculs de gradients utilisés par l'optimisation numérique de forme deviennent évidente.

Le cadre théorique du calcul des gradients d'un système complexe couplé par la résolution de la méthode de l'équation linéarisée a été formalisé par J. Sobieszcsanski-Sobieski [208, 209]. Ce dernier s'est inspiré des méthodes de gradients utilisées par les premières optimisations structurelles [97, 123, 21]. Les gradients sont alors accessibles grâce à la résolution du système des équations du gradient global - "*global sensitivity equations (GSE)*" en anglais.

Les premières études portant sur le calcul des gradients associés à un système aéroélastique, utilisaient un modèle analytique simple pour simuler le comportement de la structure et un modèle théorique linéaire pour évaluer le chargement aérodynamique transmis à la structure [92, 30, 75, 16, 200]. Cependant, depuis une douzaine d'années, les modèles aéroélastiques utilisés sont non-linéaires. Les configurations étudiées ont d'abord été bi-dimensionnelles [78, 149] et sont aujourd'hui tri-dimensionnelles [153, 85, 83, 138, 110]. Les calculs de gradients étaient exclusivement réalisés en utilisant la méthode de l'équation linéarisée discrète. Ils sont depuis peu effectués aussi grâce à la méthode de l'équation adjointe discrète [134, 139, 71] : Martins *et al.* [134, 135] ont utilisé la méthode adjointe en association avec une modélisation éléments finis simplifiée pour simuler le comportement de la structure et les équations d'Euler pour simuler le comportement du fluide (maillage volumique structuré) ; les gradients évalués ont servi à l'optimisation d'un avion d'affaire supersonique ; Maute *et al.* [139] ont par ailleurs utilisé la méthode de l'équation adjointe en association avec une modélisation éléments finis détaillée pour simuler le comportement de la structure et les équations d'Euler pour simuler le comportement du fluide (maillage volumique non structuré) pour calculer les gradients et optimiser une voilure transsonique théorique développée par la NASA. Dans le cadre de cette étude, est présenté le calcul analytique, par la méthode adjointe discrète, des gradients associés à un système aéroélastique dans lequel le comportement de la structure est modélisé par la théorie des poutres d'Euler-Bernoulli et résolu au moyen de la matrice de flexibilité, et celui du fluide par les équations d'Euler ou les équations de Navier-Stokes moyennées. La modélisation de la structure, bien que faiblement fidèle en toute généralité, prédit correctement le champ de déplacement des ailes à grand allongement [51, 162]. Un tel calcul n'a jamais été effectué, il est cependant susceptible de fournir des résultats pertinents pour une phase de conception du type avant-projet par exemple.

3.2 Ecriture des dépendances des différentes fonctions considérées

Nous nous intéressons aux équations aéroélastiques décrites dans le chapitre 2. Le système d'équations couplées discrètes décrivant l'état du système est écrit sous forme de résidus :

$$\begin{cases} R_f(W,X) = 0 \\ \\ R_s(D,Z) = D - FL = 0 \end{cases}$$

Ces équations sont couplées par l'intermédiaire des coordonnées des noeuds du maillage du domaine fluide X et des efforts aérodynamiques L appliqués à la structure. Plus précisément, d'après la section 2.4.3, les coordonnées des noeuds du maillage du domaine fluide X sont fonctions de X_{rig} - coordonnées des noeuds du maillage du domaine fluide avant tout déplacement dû aux effets aéroélastiques, de Z_{rig} - coordonnées des noeuds du maillage de la structure avant tout déplacement dû aux effets aéroélastiques (hypothèse des petits déplacements et petites déformations), et de D - champ des déplacements de la structure :

$$X = X(X_{rig}, Z_{rig}, D). \tag{3.1}$$

Le chargement aérodynamique appliqué à la structure est fonction des coordonnées X et $Z = Z_{rig}$ et du champ des variables conservatives fluides W au niveau de l'interface $\mathcal{F}^I$. Ce dernier - noté W^b - dépend de la(des) condition(s) à la(aux) limite(s) caractérisant l'interface $\mathcal{F}^I$ (paroi adiabatique sans glissement, paroi isotherme avec glissement etc). Il est de façon générale fonction de W et de X :

$$L = L(X(X_{rig}, Z_{rig}, D), Z_{rig}, W^b(W, X(X_{rig}, Z_{rig}, D))). \tag{3.2}$$

Précisément, le vecteur des paramètres de forme utilisé au sein du processus d'optimisation de forme définit de façon paramétrée le maillage du domaine fluide X_{rig} avant tout déplacement dû aux effets aéroélastique :

$$X_{rig} = X_{rig}(\alpha).$$

En faisant apparaitre les dépendances dans le système d'équations différentielles décrivant l'état du système, celui-ci devient :

$$\begin{cases} R_f(W, X(X_{rig}(\alpha), Z_{rig}, D)) = 0 \\ \\ R_s(D,Z) = D - FL(X(X_{rig}(\alpha), Z_{rig}, D), Z_{rig}, W^b(W, X(X_{rig}(\alpha), Z_{rig}, D))) = 0 \end{cases}$$

Rappelons que sous une hypothèse légitime signifiant qu'il y a localement unicité des variables d'état (écoulement et déplacements) associées à un maillage déterminé $X_{rig}(\alpha)$, les inconnues du système aéroélastique peuvent être considérées comme des fonctions de $X_{rig}(\alpha)$ et par conséquent de α : $W(\alpha)$, $D(\alpha)$. Le système d'équations différentielles décrivant l'état du système s'écrit donc de manière complète :

$$\begin{cases} R_f(W(\alpha), X(X_{rig}(\alpha), Z_{rig}, D(\alpha))) = 0 \\ \\ D(\alpha) - FL(X(X_{rig}(\alpha), Z_{rig}, D(\alpha)), Z_{rig}, W^b(W(\alpha), X(X_{rig}(\alpha), Z_{rig}, D(\alpha)))) = 0 \end{cases} \tag{3.3}$$

Les fonctions objectif et contraintes considérées par le processus d'optimisation de forme dépendent généralement du maillage X et du champ aérodynamique sur l'interface $\mathcal{F}^i$, $W^b(W,X)$. Elles peuvent également dépendre directement du champ aérodynamique W dans le domaine

occupé par le fluide. C'est le cas par exemple lorsque ces dernières correspondent aux coefficients aérodynamiques associés à un écoulement de fluide visqueux. Elles dépendent ainsi de façon implicite du vecteur α puisque X, W et W^b sont fonctions de α :

$$\begin{cases} X = X(X_{rig}(\alpha), Z_{rig}, D(\alpha)) = X(\alpha) \\ W = W(\alpha) \\ W^b = W^b(W(\alpha), X(\alpha)) = W^b(\alpha) \end{cases} . \tag{3.4}$$

De telle sorte qu'en faisant apparaitre les dépendances explicitées ci-dessus, ces fonctions sont égales à :

$$\begin{cases} J(X(\alpha), W^b(\alpha), W(\alpha)) \\ G_k(X(\alpha), W^b(\alpha), W(\alpha)), \ \forall k \in [1,p] \end{cases},$$

soit,

$$\begin{cases} J(X(X_{rig}(\alpha), Z_{rig}, D(\alpha)), W^b(W(\alpha), X(X_{rig}(\alpha), Z_{rig}, D(\alpha))), W(\alpha)) \\ G_k(X(X_{rig}(\alpha), Z_{rig}, D(\alpha)), W^b(W(\alpha), X(X_{rig}(\alpha), Z_{rig}, D(\alpha))), W(\alpha)), \ \forall k \in [1,p] \end{cases} . \tag{3.5}$$

Le processus d'optimisation de forme locale par gradient requiert le calcul des gradients des fonctions objectif et contraintes par rapport au vecteur α. En d'autres termes pour pouvoir effectuer une optimisation de forme locale par gradient, il faut être capable de calculer les dérivées totales $\frac{dJ}{d\alpha}$ et $\frac{dG_k}{d\alpha}$, $\forall k \in [1,p]$. soit, en utilisant les dépendances de ces fonctions, le gradient de la fonction objectif par rapport à α :

$$\begin{aligned} \frac{dJ}{d\alpha} = & \frac{\partial J}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{dX_{rig}}{d\alpha} + \frac{\partial X}{\partial D}\frac{dD}{d\alpha}\right) \\ & + \frac{\partial J}{\partial W^b}\left(\frac{\partial W^b}{\partial W}\frac{dW}{d\alpha} + \frac{\partial W^b}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{dX_{rig}}{d\alpha} + \frac{\partial X}{\partial D}\frac{dD}{d\alpha}\right)\right) + \frac{\partial J}{\partial W}\frac{dW}{d\alpha} \end{aligned} \tag{3.6}$$

et le gradient des fonctions contraintes G_k, $\forall k \in [1,p]$, par rapport à α :

$$\begin{aligned} \frac{dG_k}{d\alpha} = & \frac{\partial G_k}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{dX_{rig}}{d\alpha} + \frac{\partial X}{\partial D}\frac{dD}{d\alpha}\right) \\ & + \frac{\partial G_k}{\partial W^b}\left(\frac{\partial W^b}{\partial W}\frac{dW}{d\alpha} + \frac{\partial W^b}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{dX_{rig}}{d\alpha} + \frac{\partial X}{\partial D}\frac{dD}{d\alpha}\right)\right) + \frac{\partial G_k}{\partial W}\frac{dW}{d\alpha} \end{aligned} \tag{3.7}$$

Les termes $\frac{\partial J}{\partial X}$, $\frac{\partial J}{\partial W^b}$, $\frac{\partial J}{\partial W}$, et $\frac{\partial G_k}{\partial X}$, $\frac{\partial G_k}{\partial W^b}$, $\frac{\partial G_k}{\partial W}$, $\forall k \in [1,p]$, dépendent directement de la définition des fonctions objectif et contraintes. Ils sont donc connus (de façon analytique) dès la définition du problème d'optimisation de forme. Dans le contexte de ce travail, les fonctions considérées sont les coefficients aérodynamiques de traînée et de portance. Les dérivées de ces fonctions par rapport aux variables aérodynamiques conservatives et aux coordonnées du maillage du domaine fluide sont calculées grâce à une version dite "dérivée" du logiciel FFD [54].

Les termes $\frac{\partial X}{\partial X_{rig}}$ et $\frac{\partial X}{\partial D}$ dépendent du processus de remaillage du domaine fluide, ils peuvent donc être calculés de façon exacte dès lors que la méthode de remaillage est choisie (section 2.4.3).

Les termes $\frac{\partial W^b}{\partial W}$ et $\frac{\partial W^b}{\partial X}$ dépendent du code de simulation numérique utilisé pour résoudre

les équations de la mécanique des fluides et de la (ou des) condition(s) à(aux) limite(s) appliquée(s) à l'interface $\mathcal{F}^I$. Ils peuvent donc être calculés à l'intérieur même du code de simulation.

Enfin le terme $\frac{dX_{rig}}{d\alpha}$ dépend du choix des paramètres de forme, en fonction des outils de maillage de surface et de volume utilisés ; il peut être calculé de façon exacte. Si ceux-ci ne fournissent pas la dérivée $\frac{dX_{rig}}{d\alpha}$, cette dernière peut être évaluée de manière approchée par différences finies.

Tous les termes composant les gradients à calculer pour procéder à une optimisation de forme par gradient sont relativement facilement calculables à l'exception des termes $\frac{dW}{d\alpha}$ et $\frac{dD}{d\alpha}$. Ceux-ci représentent les gradients des variables d'état du système aéroélastique (solutions du système d'équations différentielles couplées) par rapport au vecteur α. Ces dérivées ne sont pas accessibles de façon simple ; ce sont les termes les plus délicats à calculer. Comme indiqué dans la section précédente, leur calcul peut être effectué selon différentes méthodes. Celles-ci sont décrites dans la section suivante.

3.3 Méthodes du calcul des gradients

3.3.1 Calcul des gradients par différences finies

Cette méthode a été utilisée jusque dans les années 1990. Elle a tendance à disparaitre au profit des méthodes "exactes" (méthodes de l'équation linéarisée discrète, de l'équation adjointe discrète et de l'équation adjointe continue).

Pour estimer à l'ordre 1 par exemple le gradient $\frac{dJ}{d\alpha}$, c'est-à-dire les dérivées partielles $\frac{\partial J}{\partial \alpha_i}$, $i \in [1, n]$, le composant, on calcule :

$$\begin{aligned}
\frac{\partial J}{\partial \alpha_i} &\approx \frac{1}{\delta\alpha_i}\Big(J(X(\alpha+\delta\alpha_i\delta^i_j), W^b(\alpha+\delta\alpha_i\delta^i_j), W(\alpha+\delta\alpha_i\delta^i_j)) \;-\; J(X(\alpha), W^b(\alpha), W(\alpha))\Big) \\
&\approx \frac{1}{\delta\alpha_i}\Big(J(X(X_{rig}(\alpha+\delta\alpha_i\delta^i_j), Z_{rig}, D(\alpha+\delta\alpha_i\delta^i_j)), \\
&\qquad W^b(W(\alpha+\delta\alpha_i\delta^i_j), X(X_{rig}(\alpha+\delta\alpha_i\delta^i_j), Z_{rig}, D(\alpha+\delta\alpha_i\delta^i_j))), W(\alpha+\delta\alpha_i\delta^i_j)) \\
&\qquad - J(X(X_{rig}(\alpha), Z_{rig}, D(\alpha)), W^b(W(\alpha), X(X_{rig}(\alpha), Z_{rig}, D(\alpha))), W(\alpha))\Big)
\end{aligned} \tag{3.8}$$

où δ^i_j représente le vecteur de $\mathbb{R}^n$ dont toutes les composantes sont nulles sauf la composante i qui est égale à 1 et $\delta\alpha_i$ est un scalaire représentant l'incrément choisi pour les différences finies. Ce calcul nécessite donc l'évaluation de $W(\alpha+\delta\alpha_i\delta^i_j)$ et $D(\alpha+\delta\alpha_i\delta^i_j)$, solutions du système couplé non-linéaire :

$$\begin{cases}
R_f(W(\alpha+\delta\alpha_i\delta^i_j), X(X_{rig}(\alpha+\delta\alpha_i\delta^i_j), Z_{rig}, D(\alpha+\delta\alpha_i\delta^i_j))) = 0 \\
\\
D(\alpha+\delta\alpha_i\delta^i_j) - FL(X(X_{rig}(\alpha+\delta\alpha_i\delta^i_j), Z_{rig}, D(\alpha+\delta\alpha_i\delta^i_j)), Z_{rig}, \\
\qquad\qquad W^b(W(\alpha+\delta\alpha_i\delta^i_j), X(X_{rig}(\alpha+\delta\alpha_i\delta^i_j), Z_{rig}, D(\alpha+\delta\alpha_i\delta^i_j)))) = 0
\end{cases}.$$

L'évaluation à l'ordre 1 du gradient $\frac{dJ}{d\alpha}$ requiert ainsi le calcul de n équilibres aéroélastiques (en plus du calcul de la solution $(W(\alpha), D(\alpha))$, soit la résolution de n systèmes non-linéaires couplés supplémentaires (problèmes pour lesquels on connait une bonne initialisation).

On vérifie facilement que le nombre de systèmes non-linéaires couplés à résoudre pour évaluer le gradient $\frac{dJ}{d\alpha}$ par différences finies à un ordre supérieur est multiplié par un facteur égal à

l'ordre de précision souhaité.

Pour certaines applications la valeur du gradient obtenu est très sensible à l'amplitude de l'incrément $\delta\alpha_i$, ce qui rend ce calcul très imprécis et inutilisable au sein d'un processus d'optimisation de forme automatisée [91].

Cette méthode, qui souffre des problèmes de coût et de précision mentionnés, est cependant simple à mettre en oeuvre. En effet, elle ne demande rien de plus que le calcul d'autant d'équilibres aéroélastiques supplémentaires qu'il y a de paramètres de forme. Elle est ainsi utilisée pour valider l'implémentation des méthodes "exactes" de calcul des gradients.

3.3.2 Méthode de l'équation linéarisée discrète (méthode directe)

On suppose que le résidu discret des équations de la mécanique des fluides et de la mécanique des structures sont continument dérivables. Quelque soit le vecteur des paramètres de forme considéré, les fonctions objectif et contraintes ainsi que leurs gradients sont évaluéees sur le système aéroélastique à l'équilibre. En d'autres termes les résidus des équations de la mécanique des fluides (R_f) et de la mécanique des structures (R_s) sont identiquement nuls sur l'intégralité du domaine $\mathcal{D}$ (on suppose implicitement ici que l'on sait faire converger la résolution du problème aéroélastique). Leur gradient par rapport au vecteur α est donc également identiquement nul sur l'intégralité du domaine $\mathcal{D}$:

$$\forall \alpha \in \mathcal{D}, \quad \begin{cases} \dfrac{dR_f}{d\alpha}(\alpha) = 0 \\ \dfrac{dR_s}{d\alpha}(\alpha) = 0 \end{cases} \tag{3.9}$$

En dérivant le système d'équations discrètes non-linéaires décrivant l'état d'équilibre du système aéroélastique (équation (3.3)) par rapport au vecteur α, on obtient :

$$\begin{cases} \dfrac{\partial R_f}{\partial W}\dfrac{dW}{d\alpha} + \dfrac{\partial R_f}{\partial X}\left(\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial X}{\partial D}\dfrac{dD}{d\alpha}\right) = 0 \\ \dfrac{dD}{d\alpha} - F\left(\dfrac{\partial L}{\partial X}\left(\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial X}{\partial D}\dfrac{dD}{d\alpha}\right)\right. \\ \qquad \left.\left(+\dfrac{\partial L}{\partial W^b}\left(\dfrac{\partial W^b}{\partial W}\dfrac{dW}{d\alpha} + \dfrac{\partial W^b}{\partial X}\left(\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial X}{\partial D}\dfrac{dD}{d\alpha}\right)\right)\right)\right) = 0 \end{cases}.$$

Ce système peut être réécrit en isolant les inconnues $\dfrac{dW}{d\alpha}$ et $\dfrac{dD}{d\alpha}$:

$$\begin{cases} \dfrac{\partial R_f}{\partial W}\dfrac{dW}{d\alpha} + \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D}\dfrac{dD}{d\alpha} = -\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} \\ -F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W}\dfrac{dW}{d\alpha} + \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right)\dfrac{dD}{d\alpha} = \\ \qquad F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha}\right) \end{cases} \tag{3.10}$$

ou encore sous forme matricielle (I représente la matrice identité dans un espace de taille $2n_p$) :

$$\begin{pmatrix} \dfrac{\partial R_f}{\partial W} & \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D} \\ -F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W} & \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right) \end{pmatrix} \begin{pmatrix} \dfrac{dW}{d\alpha} \\ \dfrac{dD}{d\alpha} \end{pmatrix} = \begin{pmatrix} -\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} \\ F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha}\right) \end{pmatrix} \tag{3.11}$$

L'inconnue $\begin{pmatrix} \dfrac{dW}{d\alpha} \\ \dfrac{dD}{d\alpha} \end{pmatrix}$ désigne la matrice rectangle $\begin{pmatrix} \dfrac{\partial W}{\partial \alpha_1} & \cdots & \dfrac{\partial W}{\partial \alpha_n} \\ \dfrac{\partial D}{\partial \alpha_1} & \cdots & \dfrac{\partial D}{\partial \alpha_n} \end{pmatrix}$. Chacun des vecteurs $\begin{pmatrix} \dfrac{\partial W}{\partial \alpha_i} \\ \dfrac{\partial D}{\partial \alpha_i} \end{pmatrix}$ est alors solution du système :

$$\begin{pmatrix} \dfrac{\partial R_f}{\partial W} & \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D} \\ -F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W} & \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right) \end{pmatrix} \begin{pmatrix} \dfrac{\partial W}{\partial \alpha_i} \\ \dfrac{\partial D}{\partial \alpha_i} \end{pmatrix} = \begin{pmatrix} -\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i} \\ F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i}\right) \end{pmatrix} \tag{3.12}$$

Notons n_{cell} le nombre de cellules du maillage du domaine fluide $\mathcal{F}$. Les dimensions des termes apparaissant dans l'équation (3.10) sont précisées dans les tableaux 3.1 et 3.2. Celles-ci dépendent en particulier du type d'équations utilisées pour modéliser le comportement du fluide (équations d'Euler ou équations RANS dans le cadre de cette étude).

Le principe de la méthode de l'équation linéarisée discrète est donc le suivant :

1. résoudre le système linéaire couplé d'équations discrètes (3.11) pour calculer les inconnues $\dfrac{dW}{d\alpha}$ et $\dfrac{dD}{d\alpha}$,
2. injecter les valeurs de ces gradients dans les équations (3.6) et (3.7) pour calculer les gradients $\dfrac{dJ}{d\alpha}$ et $\dfrac{dG_k}{d\alpha}$, $k \in [1, p]$, requis par le processus d'optimisation local par gradients.

La méthode de l'équation linéarisée discrète permet donc d'accéder aux $(p+1) * n$ dérivées nécessaires au processus d'optimisation local par gradients du système aéroélastique :

$$\left(\frac{\partial J}{\partial \alpha_i} \ , \ \left(\frac{\partial G_k}{\partial \alpha_i}\right)_{1 \leq k \leq p} \right)_{1 \leq i \leq n}$$

$\partial R_f/\partial W$	Euler : $(5n_{cell} * 5n_{cell})$ RANS : $(7n_{cell} * 7n_{cell})$
$\partial R_f/\partial X$	Euler : $(5n_{cell} * 3n_f)$ RANS : $(7n_{cell} * 3n_f)$
$\partial X/\partial X_{rig}$	$(3n_f * 3n_f)$
$\partial X/\partial D$	$(3n_f * 2n_p)$
$\partial L/\partial X$	$(3n_p * 3n_f)$
$\partial L/\partial W^b$	Euler : $(3n_p * 5n_f^I)$ RANS : $(3n_p * 7n_f^I)$
$\partial W^b/\partial W$	Euler : $(5n_f^I * 5n_{cell})$ RANS : $(7n_f^I * 7n_{cell})$
$\partial W^b/\partial X$	Euler : $(5n_f^I * 3n_f)$ RANS : $(7n_f^I * 3n_f)$

TAB. 3.1 – Dimensions des termes intervenant dans l'équation (3.10).

moyennant la résolutions de n systèmes linéaires couplés dont les inconnues sont $\left(\frac{\partial W}{\partial \alpha_i}, \frac{\partial D}{\partial \alpha_i}\right)$, $i \in [1, n]$.

La résolution de ces systèmes d'équations linéaires couplés nécessite l'évaluation des gradients indiqués dans le tableau 3.1. Les dérivées partielles $\partial R_f/\partial W$ et $\partial R_f/\partial X$ représentent respectivement la sensibilité * des équations non-linéaires discrètes écrites sous forme de résidu pour la mécanique des fluides au champ des grandeurs aérodynamiques conservatives et aux coordonnées du maillage. Celles-ci dépendent du code de simulation pour la mécanique des fluides utilisé ainsi que du modèle utilisé pour décrire le comportement du fluide (équations d'Euler ou équations

*anglicisme très utilisé dans le domaine de l'optimisation de forme issu du mot anglais "sensitivity" désignant la dérivée

$dW/d\alpha$	Euler : $(5n_{cell} * n)$ RANS : $(7n_{cell} * n)$
$dD/d\alpha$	$(2n_p * n)$
$dX_{rig}/d\alpha$	$(3n_f * n)$
$\partial W/\partial \alpha_i$	Euler : $(5n_{cell})$ RANS : $(7n_{cell})$
$\partial D/\partial \alpha_i$	$(2n_p)$
$\partial X_{rig}/\partial \alpha_i$	$(3n_f)$

TAB. 3.2 – Dimensions des inconnues du système d'équations linéarisées.

RANS dans le cadre de cette étude). La dérivée $\partial R_f/\partial W$ est calculée de façon exacte par dérivation des équations non-linéaires discrètes à l'intérieur du code de simulation pour la mécanique des fluides. La dérivée $\partial R_f/\partial X$ n'est pas accessible de façon exacte dans le code de simulation pour la mécanique des fluides utilisé. Les termes $\frac{\partial R_f}{\partial X}\frac{\partial X}{\partial X_{rig}}\frac{dX_{rig}}{d\alpha}$ et $\frac{\partial R_f}{\partial X}\frac{\partial X}{\partial D}\frac{dD}{d\alpha}$ sont évalués de manière approchée suivant des formules par différences finies à l'ordre 2 à l'intérieur du code de simulation pour la mécanique des fluides.

Les dérivées partielles $\partial L/\partial X$ et $\partial L/\partial W^b$ représentent respectivement les sensibilités du chargement aérodynamique transmis à la poutre selon le processus décrit dans la section 2.4.4 par rapport aux coordonnées du maillage du domaine fluide et aux champs des grandeurs aérodynamiques conservatives à l'interface $\mathcal{F}^I$. Ces dérivées peuvent être calculées de façon exacte par dérivation des formules d'intégration des efforts (à l'intérieur du code de simulation pour la mécanique des fluides) et de transfert du chargement aérodynamique (à l'intérieur du code de simulation pour la mécanique des structures).

3.3.3 Méthode de l'équation adjointe discrète

Les équations de la méthode de l'équation adjointe discrète peuvent être obtenues par différents formalismes et manipulations algébriques [167]. Présentons ici la méthode la plus classique consistant à introduire un lagrangien associé à chaque fonction dont on souhaite calculer le gradient. Nous définissons ainsi ici $2 * (1 + p)$ lagrangiens ou vecteurs adjoints :

- $\lambda_{f,i}$, le vecteur adjoint associé à α_i, $1 \leq i \leq n$, et à la fonction objectif J et correspondant au système d'équations non-linéaires discrètes pour la mécanique des fluides R_f, il est de

taille ($5n_{cell}$) si les équations d'Euler sont utilisées pour simuler le comportement du fluide ou ($7n_{cell}$) si ce sont les équations RANS additionnées d'un modèle de turbulence à deux équations de transport,

- $\lambda_{s,i}$, le vecteur adjoint associé à α_i, $1 \leq i \leq n$, et à la fonction objectif J et correspondant au système d'équations linéaires discrètes pour la mécanique des structures R_s, il est de taille $2n_p$,

- $\lambda_{kf,i}$, $1 \leq k \leq p$, les vecteurs adjoints associés à α_i, $1 \leq i \leq n$, et aux fonctions contraintes G_k, $1 \leq k \leq p$, et correspondant au système d'équations non-linéaires discrètes pour la mécanique des fluides R_f (taille indiquée ci-dessus),

- $\lambda_{ks,i}$, $1 \leq k \leq p$, les vecteurs adjoints associés à α_i, $1 \leq i \leq n$, et aux fonctions contraintes G_k, $1 \leq k \leq p$, et correspondant au système d'équations linéaires discrètes pour la mécanique des structures R_s(taille indiquée ci-dessus),

Le processus d'optimisation de forme locale par gradient requiert l'évaluation des gradients $\dfrac{\partial J}{\partial \alpha_i}$ et $\dfrac{\partial G_k}{\partial \alpha_i}$, $1 \leq k \leq p$, $1 \leq i \leq n$, . En utilisant l'équation (3.9), ces gradients peuvent être réécrits de la manière suivante[†]

$$\begin{cases} \dfrac{\partial J}{\partial \alpha_i} = \dfrac{\partial J}{\partial \alpha_i} + \lambda_{f,i}^T \dfrac{\partial R_f}{\partial \alpha_i} + \lambda_{s,i}^T \dfrac{\partial R_s}{\partial \alpha_i} \\ \dfrac{\partial G_k}{\partial \alpha_i} = \dfrac{\partial G_k}{\partial \alpha_i} + \lambda_{kf,i}^T \dfrac{\partial R_f}{\partial \alpha_i} + \lambda_{ks,i}^T \dfrac{\partial R_s}{\partial \alpha_i}, \ 1 \leq k \leq p \end{cases} \tag{3.13}$$

$$\begin{cases} \dfrac{\partial J}{\partial \alpha_i} = \dfrac{\partial J}{\partial \alpha_i} + \lambda_f^T \dfrac{\partial R_f}{\partial \alpha_i} + \lambda_s^T \dfrac{\partial R_s}{\partial \alpha_i} \\ \dfrac{\partial G_k}{\partial \alpha_i} = \dfrac{\partial G_k}{\partial \alpha_i} + \lambda_{kf}^T \dfrac{\partial R_f}{\partial \alpha_i} + \lambda_{ks}^T \dfrac{\partial R_s}{\partial \alpha_i}, \ 1 \leq k \leq p \end{cases} \tag{3.14}$$

On a vu dans la section 3.2 que les termes les plus délicats à calculer pour accéder aux valeurs de $\dfrac{\partial J}{\partial \alpha_i}$ et $\dfrac{\partial G_k}{\partial \alpha_i}$, $1 \leq k \leq p$, étaient les gradients des inconnues d'état du système aéroélastique par rapport au paramètre α_i, i.e. $\dfrac{\partial W}{\partial \alpha_i}$ et $\dfrac{\partial D}{\partial \alpha_i}$ (voir les équations (3.6) et (3.7)).

L'idée utilisée par la méthode de l'équation adjointe est de choisir les vecteurs adjoints définis ci-dessus de telle sorte que ces termes disparaissent des expressions permettant de calculer $\dfrac{\partial J}{\partial \alpha_i}$ et $\dfrac{\partial G_k}{\partial \alpha_i}$, $1 \leq k \leq p$.

Seul le calcul des vecteurs adjoints associés à la fonction objectif est détaillé, le calcul des vecteurs adjoints associés aux fonctions contraintes étant similaire. L'expression de l'ensemble des vecteurs adjoints est cependant donnée. Réécrivons dans un premier temps, sous forme dé-

[†]l'exposant T représente l'opérateur transposition.

veloppée, la première équation du système d'équations (3.14) :

$$\begin{aligned}
\frac{\partial J}{\partial \alpha_i} = & \frac{\partial J}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} + \frac{\partial X}{\partial D}\frac{\partial D}{\partial \alpha_i}\right) \\
& + \frac{\partial J}{\partial W^b}\left(\frac{\partial W^b}{\partial W}\frac{\partial W}{\partial \alpha_i} + \frac{\partial W^b}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} + \frac{\partial X}{\partial D}\frac{\partial D}{\partial \alpha_i}\right)\right) + \frac{\partial J}{\partial W}\frac{\partial W}{\partial \alpha_i} \\
& + \lambda_{f,i}^T \left(\frac{\partial R_f}{\partial W}\frac{\partial W}{\partial \alpha_i} + \frac{\partial R_f}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} + \frac{\partial X}{\partial D}\frac{\partial D}{\partial \alpha_i}\right) \right) \\
& + \lambda_{s,i}^T \left(\frac{\partial D}{\partial \alpha_i} - F\left(\frac{\partial L}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} + \frac{\partial X}{\partial D}\frac{\partial D}{\partial \alpha_i}\right)\right.\right. \\
& \qquad \left.\left. + \frac{\partial L}{\partial W^b}\left(\frac{\partial W^b}{\partial W}\frac{\partial W}{\partial \alpha_i} + \frac{\partial W^b}{\partial X}\left(\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} + \frac{\partial X}{\partial D}\frac{\partial D}{\partial \alpha_i}\right)\right)\right) \right)
\end{aligned}$$

Soit, en factorisant par dérivées des variables d'état :

$$\begin{aligned}
\frac{\partial J}{\partial \alpha_i} = & \frac{\partial J}{\partial X}\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} + \frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} \\
& + \lambda_{f,i}^T \frac{\partial R_f}{\partial X}\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} \\
& - \lambda_{s,i}^T F\left(\frac{\partial L}{\partial X}\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} + \frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i}\right) \\
& + \left(\frac{\partial J}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D} + \lambda_{f,i}^T \frac{\partial R_f}{\partial X}\frac{\partial X}{\partial D}\right. \\
& \qquad \left. + \lambda_{s,i}^T \left(I - F\left(\frac{\partial L}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)\right)\right)\frac{\partial D}{\partial \alpha_i} \\
& + \left(\frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial W} + \frac{\partial J}{\partial W} + \lambda_{f,i}^T \frac{\partial R_f}{\partial W} - \lambda_{s,i}^T F\frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial W}\right)\frac{\partial W}{\partial \alpha_i}
\end{aligned} \tag{3.15}$$

Les vecteurs adjoints $\lambda_{f,i}$ et $\lambda_{s,i}$ sont choisis de telle sorte que

$$\begin{cases}
\dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial W} + \dfrac{\partial J}{\partial W} + \lambda_{f,i}^T \dfrac{\partial R_f}{\partial W} - \lambda_{s,i}^T F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W} = 0 \\
\dfrac{\partial J}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D} + \lambda_{f,i}^T \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D} + \lambda_{s,i}^T \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right) = 0
\end{cases} \tag{3.16}$$

Ils sont donc solutions du système linéaire discret couplé suivant :

$$\left\{\begin{array}{l} \left(\dfrac{\partial R_f}{\partial W}\right)^T \lambda_{f,i} - \left(F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W}\right)^T \lambda_{s,i} = -\left(\dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial W}\right)^T - \left(\dfrac{\partial J}{\partial W}\right)^T \\ \left(\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D}\right)^T \lambda_{f,i} + \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right)^T \lambda_{s,i} = \\ \qquad -\left(\dfrac{\partial J}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)^T \end{array}\right. \tag{3.17}$$

soit sous forme matricielle :

$$\begin{array}{c} \begin{pmatrix} \dfrac{\partial R_f}{\partial W} & -F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W} \\ \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D} & I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right) \end{pmatrix}^T \begin{pmatrix} \lambda_{f,i} \\ \lambda_{s,i} \end{pmatrix} \\ = -\begin{pmatrix} \dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial W} + \dfrac{\partial J}{\partial W} \\ \dfrac{\partial J}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D} \end{pmatrix}^T \end{array} \tag{3.18}$$

De même les vecteurs adjoints associés aux fonctions contraintes sont solutions des systèmes linéaires ($1 \leq k \leq p$) :

$$\left\{\begin{array}{l} \left(\dfrac{\partial R_f}{\partial W}\right)^T \lambda_{kf,i} - \left(F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W}\right)^T \lambda_{ks,i} = -\left(\dfrac{\partial G_k}{\partial W^b}\dfrac{\partial W^b}{\partial W}\right)^T - \left(\dfrac{\partial G_k}{\partial W}\right)^T \\ \left(\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D}\right)^T \lambda_{kf,i} + \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right)^T \lambda_{ks,i} = \\ \qquad -\left(\dfrac{\partial G_k}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial G_k}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)^T \end{array}\right. \tag{3.19}$$

ou sous forme matricielle :

$$\begin{array}{c} \begin{pmatrix} \dfrac{\partial R_f}{\partial W} & -F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W} \\ \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D} & I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right) \end{pmatrix}^T \begin{pmatrix} \lambda_{kf,i} \\ \lambda_{ks,i} \end{pmatrix} \\ = -\begin{pmatrix} \dfrac{\partial G_k}{\partial W^b}\dfrac{\partial W^b}{\partial W} + \dfrac{\partial G_k}{\partial W} \\ \dfrac{\partial G_k}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial G_k}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D} \end{pmatrix}^T \end{array} \tag{3.20}$$

Les sytèmes linéaires discrets couplés dont sont solutions les vecteurs adjoints $(\lambda_{f,i}, \lambda_{s,i})_{1\leq i\leq n}$ sont identiques. De même, les sytèmes linéaires discrets couplés dont sont solutions les vecteurs

adjoints $(\lambda_{ks,i}, \lambda_{ks,i})_{(1\leq k\leq p\ ,\ 1\leq i\leq n)}$ sont tous identiques. Ce résultat, non évident a priori, signifie que les vecteurs adjoints dépendent des fonctions dont sont calculés les gradients, de R_f et de R_s mais pas des paramètres de forme. Il suffit donc de calculer les vecteurs adjoints λ_f et λ_s solutions du système linéaire discret couplé :

$$\left\{\begin{array}{l} \left(\dfrac{\partial R_f}{\partial W}\right)^T \lambda_f - \left(F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W}\right)^T \lambda_s = -\left(\dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial W}\right)^T - \left(\dfrac{\partial J}{\partial W}\right)^T \\ \left(\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D}\right)^T \lambda_f + \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right)^T \lambda_s = \\ \qquad -\left(\dfrac{\partial J}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)^T \end{array}\right. \tag{3.21}$$

ou les vecteurs adjoints λ_{kf} et λ_{ks}, $1 \leq k \leq p$, solutions du système linéaire discret couplé :

$$\left\{\begin{array}{l} \left(\dfrac{\partial R_f}{\partial W}\right)^T \lambda_{kf} - \left(F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W}\right)^T \lambda_{ks} = -\left(\dfrac{\partial G_k}{\partial W^b}\dfrac{\partial W^b}{\partial W}\right)^T - \left(\dfrac{\partial G_k}{\partial W}\right)^T \\ \left(\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D}\right)^T \lambda_{kf} + \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right)^T \lambda_{ks} = \\ \qquad -\left(\dfrac{\partial G_k}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial G_k}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)^T \end{array}\right. \tag{3.22}$$

pour accéder aux gradients des fonctions objectif et contraintes. Ces derniers sont alors calculés de la manière suivante :

$$\left\{\begin{array}{l} \dfrac{dJ}{d\alpha} = \dfrac{\partial J}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial J}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} \\ \qquad + \lambda_f^T\ \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} \\ \qquad - \lambda_s^T\ F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha}\right) \\ \dfrac{dG_k}{d\alpha} = \dfrac{\partial G_k}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial G_k}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} \\ \qquad + \lambda_{kf}^T\ \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} \\ \qquad - \lambda_{ks}^T\ F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{dX_{rig}}{d\alpha}\right) \quad , \quad 1 \leq k \leq p \end{array}\right. \tag{3.23}$$

Le principe de la méthode de l'équation adjointe discrète est donc le suivant :

1. résoudre les $1+p$ systèmes linéaires couplés d'équations discrètes (3.21) et (3.22) $(1 \leq k \leq p)$ pour calculer les inconnues λ_f, λ_s et λ_{kf}, λ_{ks}, $1 \leq k \leq p$

2. injecter les valeurs des vecteurs adjoints dans le système (3.23) pour calculer les gradients $\frac{dJ}{d\alpha}$ et $\frac{dG_k}{d\alpha}$, $k \in [1,p]$, requis par le processus d'optimisation local par gradients.

La méthode de l'équation adjointe discrète permet donc d'accéder aux $(p+1)*n$ dérivées nécessaires au processus d'optimisation local par gradient du système aéroélastique, à savoir :

$$\left(\frac{\partial J}{\partial \alpha_i} \ , \ \left(\frac{\partial G_k}{\partial \alpha_i} \right)_{1 \leq k \leq p} \right)_{1 \leq i \leq n}$$

moyennant la résolution de $1+p$ systèmes linéaires couplés dont les inconnues sont λ_f, λ_s et λ_{kf}, λ_{ks}, $i \in [1,p]$.

La méthode de l'équation adjointe discrète n'a été appliquée que très récemment pour le calcul de gradient de systèmes couplés fluide-structure [139, 140, 134, 135, 71].

Rappelons que la méthode de l'équation linéarisée discrète permet d'accéder aux $(p+1)*n$ gradients nécessaires au processus d'optimisation local par gradients moyennant la résolution de n systèmes linéaires couplés, tandis que la méthode de l'équation adjointe discrète n'en requiert que $1+p$. Pour les problèmes d'optimisation de forme ayant un intérêt industriel (voir par exemple [135]), le nombre de paramètres de formes est très supérieur aux nombres de fonctions contraintes à satisfaire. En général n est de l'ordre de plusieurs centaines alors que p n'excède pas la dizaine. De telle sorte que la relation suivante peut être considérée comme vraie pour la quasi-totalité des applications industrielles :

$$n >> 1+p.$$

Dans ces cas, il est alors préférable d'utiliser la méthode de l'équation adjointe pour calculer les gradients des fonctions objectif et contraintes.

3.4 Méthodes itératives de calcul des gradients utilisées dans le cadre de cette étude

3.4.1 Méthode de l'équation linéarisée discrète

Le calcul des gradients des fonctions objectif et contraintes par la méthode de l'équation linéarisée discrète nécessite la résolution des n systèmes linéaires d'équations discrètes (3.12). Comme pour le calcul de la position d'équilibre aéroélastique, c'est-à-dire le calcul des inconnues d'état W et D, solutions du système d'équations non-linéaires discrètes (2.6), les gradients des inconnues d'état par rapport à α, sont calculés suivant une procédure itérative du type point fixe.

Afin de découpler le système lors de la résolution, une procédure à retard est de plus utilisée [134, 138]. Celle-ci consiste à retarder l'avancée d'une discipline par rapport à l'autre lors de la résolution du système. En d'autres termes une des équations est choisie pour calculer

$$\frac{dW}{d\alpha} = \lim_{q->\infty} \left(\frac{dW}{d\alpha} \right)^{(q)}.$$

En général, il s'agit de l'équation $\frac{dR_f}{d\alpha}$ (voir équation (3.10)). Le gradient du champ de déplacement par rapport à α qui apparait dans cette équation est alors celui évalué à l'itération précédente $(q-1)$ de la procédure itérative. L'autre équation peut alors être utilisée pour calculer

$$\frac{dD}{d\alpha} = \lim_{q->\infty} \left(\frac{dD}{d\alpha} \right)^{(q)}.$$

Il s'agit en général de l'équation $\dfrac{dR_s}{d\alpha}$ (voir équation (3.10)). Le gradient du champ des grandeurs conservatives par rapport à α qui apparait dans cette équation est alors celui évalué à l'itération courante (q) de la procédure itérative.

Les systèmes,

$$\left\{\begin{array}{l} \dfrac{\partial R_f}{\partial W}\dfrac{\partial W}{\partial \alpha_i} + \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D}\dfrac{\partial D}{\partial \alpha_i} = -\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i} \\ \\ -F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W}\dfrac{\partial W}{\partial \alpha_i} + \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right)\dfrac{\partial D}{\partial \alpha_i} = \quad , \ 1 \leq i \leq n \\ \\ \qquad +F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i}\right) \end{array}\right. \tag{3.24}$$

sont résolus itérativement de la manière suivante (l'exposant (q) représente l'itération numéro q du processus de résolution) :

$$\left\{\begin{array}{l} \dfrac{\partial R_f}{\partial W}\left(\dfrac{\partial W}{\partial \alpha_i}\right)^{(q)} = -\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D}\left(\dfrac{\partial D}{\partial \alpha_i}\right)^{(q)} - \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i} \\ \\ \left(I - F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\right)\left(\dfrac{\partial D}{\partial \alpha_i}\right)^{(q+1)} = F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W}\left(\dfrac{\partial W}{\partial \alpha_i}\right)^{(q)} \\ \\ \qquad +F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i}\right) \end{array}\right. . \tag{3.25}$$

Pour s'affranchir de l'inversion de la matrice $F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)$ nous avons eu l'idée d'appliquer la même technique de retard à la seconde équation du système (3.25). plus précisément, d'utiliser $\left(\dfrac{\partial D}{\partial \alpha_i}\right)^{(q)}$ pour évaluer le terme

$$F\left(\frac{\partial L}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)\frac{\partial D}{\partial \alpha_i}$$

lors du calcul de $\left(\dfrac{\partial D}{\partial \alpha_i}\right)^{(q+1)}$.

A l'étape (q) du processus itératif qui le résout, le système (3.24) s'écrit :

$$\left\{\begin{array}{ll} \dfrac{\partial R_f}{\partial W}\left(\dfrac{\partial W}{\partial \alpha_i}\right)^{(q)} = -\dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial D}\left(\dfrac{\partial D}{\partial \alpha_i}\right)^{(q)} - \dfrac{\partial R_f}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i} & (a) \\ \\ \left(\dfrac{\partial D}{\partial \alpha_i}\right)^{(q+1)} = F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial D} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial D}\right)\left(\dfrac{\partial D}{\partial \alpha_i}\right)^{(q)} + F\dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial W}\left(\dfrac{\partial W}{\partial \alpha_i}\right)^{(q)} & . \\ \\ \qquad +F\left(\dfrac{\partial L}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i} + \dfrac{\partial L}{\partial W^b}\dfrac{\partial W^b}{\partial X}\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i}\right) & (b) \end{array}\right. \tag{3.26}$$

Ce processus itératif procède de la manière suivante :

1. initialisation de la procédure itérative :
   on se place à la position d'équilibre aéroélastique correspondant à α (le champ des grandeurs

conservatives sur le maillage $X(\alpha) = X(X_{rig}(\alpha), Z_{rig}, D(\alpha))$ est $W(\alpha)$ et le champ des déplacements appliqué au maillage $Z(\alpha) = Z_{rig}$ est $D(\alpha)$) et le gradient du champ des déplacements de la structure par rapport au vecteur α est supposé identiquement nul $\left(\frac{\partial D}{\partial \alpha_i}\right)^{(0)} = 0$

2. le gradient du champ des grandeurs aérodynamiques conservatives par rapport à α, $\left(\frac{\partial W}{\partial \alpha_i}\right)^{(q)}$, est calculé à partir du gradient $\left(\frac{\partial D}{\partial \alpha_i}\right)^{(q)}$ en utilisant l'équation (a) du système (3.26)
3. le gradient du champ des déplacements de la structure par rapport au vecteur α à l'itération suivante, $\left(\frac{\partial D}{\partial \alpha_i}\right)^{(q+1)}$, est calculé à partir des gradients $\left(\frac{\partial W}{\partial \alpha_i}\right)^{(q)}$ et $\left(\frac{\partial D}{\partial \alpha_i}\right)^{(q)}$ et de l'équation (b) du système (3.26)
4. les critères de convergence sont évalués(ϵ_1 et ϵ_2 sont deux seuils de tolérance fixés au préalable) :

$$\begin{cases} \left|\left|\frac{\partial R_f}{\partial \alpha_i}\right|\right|_2 \leq \epsilon_1 \\ \\ \left|\left|\left(\frac{\partial D}{\partial \alpha_i}\right)^{(q+1)} - \left(\frac{\partial D}{\partial \alpha_i}\right)^{(q)}\right|\right|_2 \leq \epsilon_2 \left|\left|\left(\frac{\partial D}{\partial \alpha_i}\right)^{(q)}\right|\right|_2 \end{cases} \tag{3.27}$$

5. si les critères de convergence sont satisfaits la boucle itérative s'arrête, sinon elle repart à l'étape 2.

A chaque pas de ce processus itératif, l'équation (a) du système (3.26) doit être résolue. La matrice jacobienne $\frac{\partial R_f}{\partial W}$ des équations pour la mécanique des fluides écrite sous forme de résidu discret est une matrice creuse très difficile à inverser par une méthode directe pour les problèmes de grande taille. Différentes méthodes itératives ont été développées pour résoudre ce système, parmi lesquelles la méthode "Generalized minimal residual method (GMRES)" [187] qui utilise une projection sur l'espace de Krylov et des techniques de minimisation du résidu mesurant l'écart entre l'équation à satisfaire et l'équation à l'itération courante du processus itératif de résolution, et des méthodes inspirées des techniques de simulation des écoulements stationnaires. C'est une méthode de ce type qui est utilisée dans le code de simulation pour la mécanique des fluides employé dans le cadre de cette étude. Décrivons là plus précisément. La matrice $\frac{\partial R_f}{\partial W}$ est approchée par une matrice $\frac{\partial R_f}{\partial W}^{(APP)}$. L'équation (a) est alors résolue selon l'algorithme itératif :

$$\begin{aligned} \frac{\partial R_f}{\partial W}^{(APP)} \left(\left(\frac{\partial W}{\partial \alpha_i}\right)^{(q)}_{(l)} - \left(\frac{\partial W}{\partial \alpha_i}\right)^{(q)}_{(l-1)}\right) = -\frac{\partial R_f}{\partial W}\left(\frac{\partial W}{\partial \alpha_i}\right)^{(q)}_{(l-1)} \\ -\frac{\partial R_f}{\partial X}\frac{\partial X}{\partial D}\left(\frac{\partial D}{\partial \alpha_i}\right)^{(q-1)} - \frac{\partial R_f}{\partial X}\frac{\partial X}{\partial X_{rig}}\frac{\partial X_{rig}}{\partial \alpha_i} \end{aligned} \tag{3.28}$$

où l'indice (l) représente l'itération numéro l du processus de résolution.

Le code de simulation utilisé, comme de nombreux codes pour la simulation numérique des équations de la mécanique des fluides, utilise le schéma numérique "backward-Euler" ou quasi-Newton, connu pour ses bonnes propriétés de convergence et de stabilité, pour résoudre les écoulements stationnaires. Celui-ci calcule le champ des grandeurs conservatives de manière itérative

selon la formule suivante[‡] :

$$\left(\frac{I}{\Delta t}+\frac{\partial R_f}{\partial W}^{(APP)}\right)\left(W_{(l)}-W_{(l-1)}\right)=-R_f(W_{(l-1)})=RHS(W_{(l-1)})\ .^{\S} \tag{3.29}$$

A chaque pas l, un système linéaire creux de grande taille ((3.28) ou (3.29)) doit être résolu, ce qui ne peut se faire de façon directe pour des applications tridimensionnelles usuelles. Ce système est généralement résolu soit par la méthode du gradient conjugué, soit par une méthode de relaxation. Le code de simulation pour la mécanique des fluides utilisé dans le cadre de cette étude résoud cette équation en utilisant une méthode de relaxation LU.

La matrice $\left(\frac{I}{\Delta t}+\frac{\partial R_f}{\partial W}^{(APP)}\right)$ tend vers la matrice $\left(\frac{\partial R_f}{\partial W}^{(APP)}\right)$ de l'équation (3.28) lorsque le pas de temps Δt tend vers l'infini. Il est donc possible d'adapter les routines de résolution du champ des grandeurs conservatives par le schéma "backward-Euler" de manière simple pour la résolution de l'équation linéarisée (a). C'est la solution qui a été adoptée dans le code de simulation pour la mécanique des fluides utilisé dans le cadre de cette étude. Cet algorithme itératif est décrit plus précisément dans [164, 168].

3.4.2 Méthode de l'équation adjointe discrète

Le calcul des gradients des fonctions objectif et contraintes par la méthode de l'équation adjointe discrète nécessite la résolution des $1+p$ systèmes linéaires d'équations discrètes (3.21) et (3.22). Les vecteurs adjoints associés aux fonctions objectif et contraintes et correspondant aux équations de la mécanique des fluides et de la mécanique des structures sont calculés suivant une procédure itérative du type point fixe, c'est-à-dire analogue à celle déterminant les solutions de l'équilibre aéroélastique et de l'équation linéarisée discrète.

Afin de découpler les systèmes à résoudre lors du processus itératif, la même procédure à retard que celle introduite dans la section 3.4.1 est utilisée. Détaillons cette procédure pour le système (3.21). Une des équations (en général la première) est choisie pour calculer

$$\lambda_f=\lim_{q->\infty}(\lambda_f)^{(q)}\ .$$

Le vecteur λ_s qui apparait dans cette équation est alors celui évalué à l'itération précédente $(q-1)$ de la procédure itérative. L'autre équation (en général la deuxième) peut alors être utilisée pour calculer

$$\lambda_s=\lim_{q->\infty}(\lambda_s)^{(q)}\ .$$

Le vecteur λ_f qui apparait dans cette équation est alors celui évalué à l'itération courante (q) de la procédure itérative.

Le système

$$\left\{\begin{array}{l}\left(\frac{\partial R_f}{\partial W}\right)^T\lambda_f-\left(F\frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial W}\right)^T\lambda_s=-\left(\frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial W}\right)^T-\left(\frac{\partial J}{\partial W}\right)^T\\ \\ \left(\frac{\partial R_f}{\partial X}\frac{\partial X}{\partial D}\right)^T\lambda_f+\left(I-F\left(\frac{\partial L}{\partial X}\frac{\partial X}{\partial D}+\frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)\right)^T\lambda_s=\\ \\ \qquad-\left(\frac{\partial J}{\partial X}\frac{\partial X}{\partial D}+\frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)^T\end{array}\right.$$

[‡]lorsque le pas de temps Δt tend vers l'infini, on retrouve le schéma de Newton classique, d'où la dénomination quasi-Newton

[§]l'abbréviation RHS désigne de manière générale le membre de droite de l'équation à résoudre.

est alors résolu itérativement de la manière suivante (l'exposant (q) représente l'itération numéro q du processus de résolution) :

$$\begin{cases} \left(\frac{\partial R_f}{\partial W}\right)^T (\lambda_f)^{(q)} = \left(F\frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial W}\right)^T (\lambda_s)^{(q)} - \left(\frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial W}\right)^T - \left(\frac{\partial J}{\partial W}\right)^T \\ \left(I - F\left(\frac{\partial L}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)\right)^T (\lambda_s)^{(q+1)} = -\left(\frac{\partial R_f}{\partial X}\frac{\partial X}{\partial D}\right)^T (\lambda_f)^{(q)} \\ \qquad -\left(\frac{\partial J}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)^T \end{cases} \tag{3.30}$$

Pour s'affranchir de l'inversion de la matrice $F\left(\frac{\partial L}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)$ nous avons eu l'idée d'appliquer, comme pour la méthode de l'équation linéarisée discrète décrite dans la section précédente, la technique à retard à la seconde équation du système (3.30). Plus précisément, d'utliser $(\lambda_s)^{(q)}$ pour évaluer le terme $F\left(\frac{\partial L}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)\lambda_s$ lors du calcul de $(\lambda_s)^{(q+1)}$.

A l'étape (q) du processus itératif qui le résout, le système (3.30) s'écrit :

$$\begin{cases} \left(\frac{\partial R_f}{\partial W}\right)^T (\lambda_f)^{(q)} = \left(F\frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial W}\right)^T (\lambda_s)^{(q)} - \left(\frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial W}\right)^T - \left(\frac{\partial J}{\partial W}\right)^T & (a) \\ (\lambda_s)^{(q+1)} = \left(F\left(\frac{\partial L}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)\right)^T (\lambda_s)^{(q)} - \left(\frac{\partial R_f}{\partial X}\frac{\partial X}{\partial D}\right)^T (\lambda_f)^{(q)} & \\ \qquad -\left(\frac{\partial J}{\partial X}\frac{\partial X}{\partial D} + \frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial X}\frac{\partial X}{\partial D}\right)^T & (b) \end{cases} \tag{3.31}$$

Ce processus itératif procède de la manière suivante :

1. initialisation de la porcédure itérative :
   on se place à la position d'équilibre aéroélastique correspondant à α (le champ des grandeurs conservatives sur le maillage $X(\alpha) = X(X_{rig}(\alpha), Z_{rig}, D(\alpha))$ est $W(\alpha)$ et le champ des déplacements appliqué au maillage $Z(\alpha) = Z_{rig}$ est $D(\alpha)$) et le vecteur adjoint associé à J et correspondant à R_s, $(\lambda_s)^{(0)}$, est initialisé par un champ nul ;
2. le vecteur adjoint associé à J et correspondant à R_f, $(\lambda_f)^{(q)}$, est calculé à partir du vecteur adjoint $(\lambda_s)^{(q)}$ en utilisant la première équation du système (3.31) ;
3. le vecteur adjoint, associé à J et correspondant à R_s à l'itération suivante, $(\lambda_s)^{(q+1)}$, est calculé à partir des vecteurs adjoints $(\lambda_f)^{(q)}$ et $(\lambda_s)^{(q)}$
4. les critères de convergence sont évalués(ϵ_3 et ϵ_4 sont deux seuils de tolérance fixés au préalable) :
$$\begin{cases} \left|\left| (\lambda_f)^{(q)} - (\lambda_f)^{(q-1)} \right|\right|_2 \leq \epsilon_3 \left|\left| (\lambda_f)^{(q-1)} \right|\right|_2 \\ \left|\left| (\lambda_s)^{(q)} - (\lambda_s)^{(q-1)} \right|\right|_2 \leq \epsilon_4 \left|\left| (\lambda_s)^{(q-1)} \right|\right|_2 \end{cases} \tag{3.32}$$
5. si les critères de convergence sont satisfaits la boucle itérative s'arrête, sinon elle se poursuit à l'étape 2.

Comme lors de la résolution du système d'équations linéarisées décrite dans la section précédente, à chaque pas du processus itératif détaillé ci-dessus, l'équation (a) du système (3.31) doit être résolue. Cette résolution est effectuée en utilisant la même logique que pour l'équation linéarisée, c'est-à-dire en s'inspirant du schéma numérique "backward-Euler" utilisé pour le calcul d'écoulements stationnaires. Plus précisément, le vecteur adjoint λ_f est calculé en résolvant le système linéaire d'équation

$$\begin{aligned}
&\left(\frac{I}{\Delta t} + \left(\frac{\partial R_f}{\partial W}^T\right)^{(APP)}\right)\left((\lambda_f)_{(l)}^{(q)} - (\lambda_f)_{(l-1)}^{(q)}\right) \\
&\quad = \left(F\frac{\partial L}{\partial W^b}\frac{\partial W^b}{\partial W}\right)^T (\lambda_s)^{(q-1)} - \left(\frac{\partial J}{\partial W^b}\frac{\partial W^b}{\partial W}\right)^T - \left(\frac{\partial J}{\partial W}\right)^T \\
&\quad = -RHS\left((\lambda_f)_{(l-1)}^{(q)}\right)
\end{aligned}$$

Les routines de calcul du champ des grandeurs conservatives d'un écoulement stationnaire suivant le schéma numérique de "backward-Euler" résolu par relaxation LU ont été adaptées pour la résolution de l'équation ci-dessus et ainsi de l'équation (a) du système (3.31). A la différence de la méthode de l'équation linéarisée, cette démarche nécessite ici la transposition de la matrice jacobienne approchée $\frac{\partial R_f}{\partial W}^{(APP)}$.

3.5 Calcul des dérivées partielles apparaissant dans les équations linéarisées et adjointes

Cette section a pour objectif de détailler le calcul des dérivées partielles qui apparaissent dans les équations de calcul des gradients. Dans le cadre de cette étude, aucune des matrices faisant l'objet des sections suivantes n'est stockée en mémoire (ni écrite dans un fichier). Celles-ci interviennent dans des produits, seuls ces derniers sont calculés par assemblage "à la volée".

3.5.1 Calcul du terme $(\partial X/\partial X_{rig})$

Cette section s'intéresse au calcul du terme $\frac{\partial X}{\partial X_{rig}}$, dérivée partielle du processus de remaillage par rapport aux coordonnées du maillage autour de la forme solide. Il apparait dans les équations linéarisées et adjointes. Le processus de remaillage du domaine fluide s'effectue selon l'équation (2.5). Si on ne considère que la dépendance de X en X_{rig}, les nouvelles coordonnées du noeud N_{ijk} ne dépendent que des coordonnées de ce noeud dans le maillage du domaine fluide autour de la forme rigide X_{rig}.

La matrice $\frac{\partial X}{\partial X_{rig}}$ est donc creuse et seuls les termes $\frac{\partial \tilde{x}_{ijk}}{\partial x_{rig_{ijk}}}$, $\frac{\partial \tilde{x}_{ijk}}{\partial y_{rig_{ijk}}}$, $\frac{\partial \tilde{x}_{ijk}}{\partial z_{rig_{ijk}}}$, $\frac{\partial \tilde{y}_{ijk}}{\partial y_{rig_{ijk}}}$, $\frac{\partial \tilde{z}_{ijk}}{\partial x_{rig_{ijk}}}$, $\frac{\partial \tilde{z}_{ijk}}{\partial y_{rig_{ijk}}}$ et $\frac{\partial \tilde{z}_{ijk}}{\partial z_{rig_{ijk}}}$ sont non nuls.

Plus précisément, ces dérivées partielles sont égales à :

$$\begin{cases} \dfrac{\partial \tilde{x}_{ijk}}{\partial x_{rig_{ijk}}} &= 1 \\ \dfrac{\partial \tilde{x}_{ijk}}{\partial y_{rig_{ijk}}} &= -\left(\dfrac{z_{i-1}}{y_i - y_{i-1}} + \dfrac{z_i}{y_i - y_{i-1}}\right)\left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\,\zeta\theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\,\zeta\theta_{p_i}\right) \\ & \quad -\left(z_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\,z_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\,z_i\right)\left(-\dfrac{\zeta\theta_{p_{i-1}}}{y_i - y_{i-1}} + \dfrac{\zeta\theta_{p_i}}{y_i - y_{i-1}}\right) \\ \dfrac{\partial \tilde{x}_{ijk}}{\partial z_{rig_{ijk}}} &= -\left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\,\zeta\theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\,\zeta\theta_{p_i}\right) \\ \dfrac{\partial \tilde{y}_{ijk}}{\partial y_{rig_{ijk}}} &= 1 \end{cases} \tag{3.33}$$

$$\begin{cases} \dfrac{\partial \tilde{z}_{ijk}}{\partial x_{rig_{ijk}}} &= \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\,\zeta\theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\,\zeta\theta_{p_i} \\ \dfrac{\partial \tilde{z}_{ijk}}{\partial y_{rig_{ijk}}} &= -\dfrac{\omega_{p_{i-1}}}{y_i - y_{i-1}} + \dfrac{\omega_{p_i}}{y_i - y_{i-1}} + \\ & \quad +\left(\dfrac{x_{i-1}}{y_i - y_{i-1}} + \dfrac{x_i}{y_i - y_{i-1}}\right)\left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\,\zeta\theta_{p_{i-1}} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\,\zeta\theta_{p_i}\right) \\ & \quad +\left(x_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\,x_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\,x_i\right)\left(-\dfrac{\zeta\theta_{p_{i-1}}}{y_i - y_{i-1}} + \dfrac{\zeta\theta_{p_i}}{y_i - y_{i-1}}\right) \\ \dfrac{\partial \tilde{z}_{ijk}}{\partial z_{rig_{ijk}}} &= 1 \end{cases} \tag{3.34}$$

La matrice $\dfrac{\partial X}{\partial X_{rig}}$ n'est pas stockée en mémoire lors de la résolution itérative des équations linéarisées. Le terme $\dfrac{\partial X}{\partial X_{rig}}\dfrac{\partial X_{rig}}{\partial \alpha_i}$ est en revanche assemblé "à la volée", ce qui évite le stockage d'une matrice tri-diagonale de grande taille.

3.5.2 Calcul du terme $(\partial X/\partial D)$

Cette section s'intéresse au calcul du terme $\dfrac{\partial X}{\partial D}$, dérivée partielle du processus de remaillage par rapport aux déplacements de la poutre. Il apparait dans les équations linéarisées et adjointes. Le processus de remaillage du domaine fluide s'effectue selon l'équation (2.5). Si on ne considère que la dépendance de X en D, les nouvelles coordonnées du noeud N_{ijk} ne dépendent que de la flexion et de la torsion des deux points de la poutre les plus près de N_{ijk}. Ces points sont notés P_{i-1} et P_i dans la section 2.4.3 et sont tels que $y_{i-1} \leq y_{rig_{ijk}} \leq y_i$.

Le processus de remaillage est tel que seuls les termes $\dfrac{\partial \tilde{x}_{ijk}}{\partial \theta_{p_{i-1}}}$, $\dfrac{\partial \tilde{x}_{ijk}}{\partial \theta_{p_i}}$, $\dfrac{\partial \tilde{z}_{ijk}}{\partial \omega_{p_{i-1}}}$, $\dfrac{\partial \tilde{z}_{ijk}}{\partial \omega_{p_i}}$, $\dfrac{\partial \tilde{z}_{ijk}}{\partial \theta_{p_{i-1}}}$

et $\dfrac{\partial \tilde{z}_{ijk}}{\partial \theta_{p_i}}$ sont non nuls.

$$\begin{cases} \dfrac{\partial \tilde{x}_{ijk}}{\partial \theta_{p_{i-1}}} &= -\left(z_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\, z_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\, z_i \right) \left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\, \zeta \right) \\ \dfrac{\partial \tilde{x}_{ijk}}{\partial \theta_{p_i}} &= -\left(z_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\, z_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\, z_i \right) \left(\dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\, \zeta \right) \end{cases} \tag{3.35}$$

$$\begin{cases} \dfrac{\partial \tilde{z}_{ijk}}{\partial \omega_{p_{i-1}}} &= \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}} \\ \dfrac{\partial \tilde{z}_{ijk}}{\partial \omega_{p_i}} &= \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}} \\ \dfrac{\partial \tilde{z}_{ijk}}{\partial \theta_{p_{i-1}}} &= \left(x_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\, x_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\, x_i \right) \left(\dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\, \zeta \right) \\ \dfrac{\partial \tilde{z}_{ijk}}{\partial \theta_{p_i}} &= \left(x_{rig_{ijk}} - \dfrac{y_i - y_{rig_{ijk}}}{y_i - y_{i-1}}\, x_{i-1} + \dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\, x_i \right) \left(\dfrac{y_{rig_{ijk}} - y_{i-1}}{y_i - y_{i-1}}\, \zeta \right) \end{cases} \tag{3.36}$$

Comme pour la matrice $\dfrac{\partial X}{\partial X_{rig}}$, la matrice $\dfrac{\partial X}{\partial D}$ n'est pas stockée en mémoire lors de la résolution itérative des équations linéarisées. Le terme $\dfrac{\partial X}{\partial D}\dfrac{\partial D}{\partial \alpha_i}$ est assemblé "à la volée", ce qui évite de stocker en mémoire une matrice de grande taille dont la majorité des termes sont nuls.

3.5.3 Calcul du terme $(\partial L/\partial W^b) * (\partial W^b/\partial W)$

Le comportement du fluide est modélisé par les équations d'Euler

D'après la section 2.4.4 les efforts transmis au noeud P_i, $1 \leq i \leq n_p$, du maillage de la poutre sont égaux à :

$$\begin{cases} \overrightarrow{\mathcal{F}_i} = \displaystyle\sum_{l=1}^{n_f^I} S_l^i \left(p_l - p_\infty\right) \vec{n}_l \\ \overrightarrow{\mathcal{M}_i} = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge S_l^i \left(p_l - p_\infty \vec{n}_l\right) \end{cases}$$

Le champ aérodynamique sur l'interface I_l ne dépend, en terme de champ aérodynamique, que des valeurs de ce dernier dans la cellule adjacente à l'interface (voir la figure 3.1). Lorsque le comportement du fluide est modélisé par les équations d'Euler, la pression à l'interface I_l est égale à :

$$p_l = p_c - c_c \left((\rho u)_c\, n_{l_x} + (\rho v)_c\, n_{l_y} + (\rho w)_c\, n_{l_z} \right)$$

où $p_c = (\gamma - 1)\left(\rho_c e_c - \dfrac{1}{2}\dfrac{(\rho u)_c^2 + (\rho v)_c^2 + (\rho w)_c^2}{\rho_c} \right)$ et $c_c = \sqrt{\gamma \dfrac{p_c}{\rho_c}}$. Les dérivées partielles des efforts transmis à la poutre par rapport au champ aérodynamique conservatif à l'intérieur du

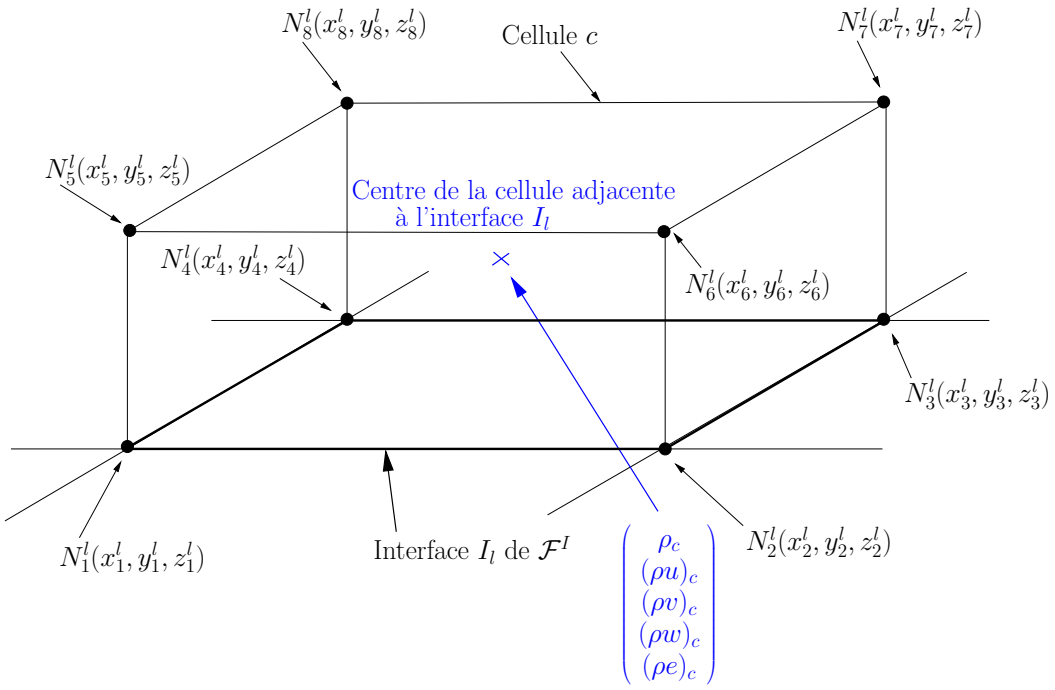

FIG. 3.1 – Interface I_l et cellule adjacente au maillage de surface $\mathcal{F}^I$ en I_l.

domaine fluide se calculent précisément de la manière suivante, où $1 \leq c \leq n_{cell}$:

$$\left\{\begin{array}{lllll}
\dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial \rho_c} & = & \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_l}{\partial \rho_c} n_{lz} & , \quad \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial \rho_c} & = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge S_l^i \frac{\partial p_l}{\partial \rho_c} \vec{n}_l \\
\dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (\rho u)_c} & = & \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_l}{\partial (\rho u)_c} n_{lz} & , \quad \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial (\rho u)_c} & = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge S_l^i \frac{\partial p_l}{\partial (\rho u)_c} \vec{n}_l \\
\dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (\rho v)_c} & = & \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_l}{\partial (\rho v)_c} n_{lz} & , \quad \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial (\rho v)_c} & = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge S_l^i \frac{\partial p_l}{\partial (\rho v)_c} \vec{n}_l \\
\dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (\rho w)_c} & = & \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_l}{\partial (\rho w)_c} n_{lz} & , \quad \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial (\rho w)_c} & = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge S_l^i \frac{\partial p_l}{\partial (\rho w)_c} \vec{n}_l \\
\dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (\rho e)_c} & = & \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_l}{\partial (\rho e)_c} n_{lz} & , \quad \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial (\rho e)_c} & = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge S_l^i \frac{\partial p_l}{\partial (\rho e)_c} \vec{n}_l
\end{array}\right. \tag{3.37}$$

Si on ne considère que les dépendances au champ W, la pression appliquée à l'interface I_l ne dépend que des grandeurs conservatives dans la cellule adjacente à I_l. Les dérivées partielles de la pression p_l par rapport aux grandeurs conservatives calculées aux centres des cellules du domaine fluide sont donc nulles sauf dans le cas particulier où la cellule c est la cellule adjacente

à l'interface I_l. Dans ce cas, ces dérivées partielles sont égales à :

$$\left\{\begin{array}{lcl}
\dfrac{\partial p_l}{\partial \rho_c} & = & \dfrac{\partial p_c}{\partial \rho_c} - \dfrac{\partial c_c}{\partial \rho_c}\left((\rho u)_c\, n_{lx} + (\rho v)_c\, n_{ly} + (\rho w)_c\, n_{lz}\right) \\
\dfrac{\partial p_l}{\partial (\rho u)_c} & = & \dfrac{\partial p_c}{\partial (\rho u)_c} - \dfrac{\partial c_c}{\partial (\rho u)_c}\left((\rho u)_c\, n_{lx} + (\rho v)_c\, n_{ly} + (\rho w)_c\, n_{lz}\right) - c_c n_{lx} \\
\dfrac{\partial p_l}{\partial (\rho v)_c} & = & \dfrac{\partial p_c}{\partial (\rho v)_c} - \dfrac{\partial c_c}{\partial (\rho v)_c}\left((\rho u)_c\, n_{lx} + (\rho v)_c\, n_{ly} + (\rho w)_c\, n_{lz}\right) - c_c n_{ly} \\
\dfrac{\partial p_l}{\partial (\rho w)_c} & = & \dfrac{\partial p_c}{\partial (\rho w)_c} - \dfrac{\partial c_c}{\partial (\rho w)_c}\left((\rho u)_c\, n_{lx} + (\rho v)_c\, n_{ly} + (\rho w)_c\, n_{lz}\right) - c_c n_{lz} \\
\dfrac{\partial p_l}{\partial (\rho e)_c} & = & \dfrac{\partial p_c}{\partial (\rho e)_c} - \dfrac{\partial c_c}{\partial (\rho e)_c}\left((\rho u)_c\, n_{lx} + (\rho v)_c\, n_{ly} + (\rho w)_c\, n_{lz}\right)
\end{array}\right. \tag{3.38}$$

D'après les formules précédentes, les dérivées partielles de la pression et de la vitesse du son au centre de la cellule c par rapport aux grandeurs conservatives au centre de cette même cellule sont égales à :

$$\left\{\begin{array}{lcl@{\quad,\quad}lcl}
\dfrac{\partial p_c}{\partial \rho_c} & = & (\gamma-1)\left(e_c + \dfrac{1}{2}\dfrac{(\rho u)_c^2 + (\rho v)_c^2 + (\rho w)_c^2}{\rho_c^2}\right) & \dfrac{\partial c_c}{\partial \rho_c} & = & \dfrac{1}{2c_c}\left(\dfrac{\gamma}{\rho_c}\dfrac{\partial p_c}{\partial \rho_c} - \gamma\dfrac{P_c}{\rho_c^2}\right) \\
\dfrac{\partial p_c}{\partial (\rho u)_c} & = & -(\gamma-1)\dfrac{(\rho u)_c}{\rho_c} & \dfrac{\partial c_c}{\partial (\rho u)_c} & = & \dfrac{1}{2c_c}\dfrac{\gamma}{\rho_c}\dfrac{\partial p_c}{\partial (\rho u)_c} \\
\dfrac{\partial p_c}{\partial (\rho v)_c} & = & -(\gamma-1)\dfrac{(\rho v)_c}{\rho_c} & \dfrac{\partial c_c}{\partial (\rho v)_c} & = & \dfrac{1}{2c_c}\dfrac{\gamma}{\rho_c}\dfrac{\partial p_c}{\partial (\rho v)_c} \\
\dfrac{\partial p_c}{\partial (\rho w)_c} & = & -(\gamma-1)\dfrac{(\rho w)_c}{\rho_c} & \dfrac{\partial c_c}{\partial (\rho w)_c} & = & \dfrac{1}{2c_c}\dfrac{\gamma}{\rho_c}\dfrac{\partial p_c}{\partial (\rho w)_c} \\
\dfrac{\partial p_c}{\partial (\rho e)_c} & = & (\gamma-1)\rho_c & \dfrac{\partial c_c}{\partial (\rho e)_c} & = & \dfrac{1}{2c_c}\dfrac{\gamma}{\rho_c}\dfrac{\partial p_c}{\partial (\rho e)_c}
\end{array}\right. \tag{3.39}$$

D'après la section 2.3.1 seules les sollicitations d'origine aérodynamique en flexion et torsion $\overrightarrow{\mathcal{F}_i}.\vec{z}$, $\overrightarrow{\mathcal{M}_i}.\vec{x}$ et $\overrightarrow{\mathcal{M}_i}.\vec{y}$ sont transmises à la poutre, de telle sorte que[¶]

$$\left\{\begin{array}{lcl}
\dfrac{\partial F_{iz}}{\partial (-)_c} & = & \dfrac{\partial (\overrightarrow{\mathcal{F}_i}.\vec{z})}{\partial (-)_c} = \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (-)_c}.\vec{z} \\
\dfrac{\partial M_{ix}}{\partial (-)_c} & = & \dfrac{\partial (\overrightarrow{\mathcal{M}_i}.\vec{x})}{\partial (-)_c} = \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial (-)_c}.\vec{x} \\
\dfrac{\partial M_{iy}}{\partial (-)_c} & = & \dfrac{\partial (\overrightarrow{\mathcal{M}_i}.\vec{y})}{\partial (-)_c} = \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial (-)_c}.\vec{y}
\end{array}\right. \tag{3.40}$$

[¶] $(-)$ représente une des cinq grandeurs aérodynamiques conservatives.

Le comportement du fluide est modélisé par les équations RANS

D'après la section 2.4.4 les efforts transmis au noeud P_i, $1 \leq i \leq n_p$, du maillage de la poutre sont égaux à :

$$\begin{cases} \overrightarrow{\mathcal{F}_i} = \displaystyle\sum_{l=1}^{n_f^I} S_l^i \left(p_l - p_\infty\right) \vec{n}_l \\ \qquad + S_l^i \mu_c \dfrac{2S_l}{\mathrm{Vol}_c} \left(\dfrac{(\rho u)_c - (\rho u)_c n_{lx}}{\rho_c} \vec{x} + \dfrac{(\rho v)_c - (\rho v)_c n_{ly}}{\rho_c} \vec{y} + \dfrac{(\rho w)_c - (\rho w)_c n_{lz}}{\rho_c} \vec{z} \right) \\ \overrightarrow{\mathcal{M}_i} = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{P_i G_l} \wedge \left(S_l^i \left(p_l - p_\infty\right) \vec{n}_l \right. \\ \qquad \left. + S_l^i \mu_c \dfrac{2S_l}{\mathrm{Vol}_c} \left(\dfrac{(\rho u)_c - (\rho u)_c n_{lx}}{\rho_c} \vec{x} + \dfrac{(\rho v)_c - (\rho v)_c n_{ly}}{\rho_c} \vec{y} + \dfrac{(\rho w)_c - (\rho w)_c n_{lz}}{\rho_c} \vec{z} \right) \right) \end{cases}$$

On suppose en outre dans cette partie que les grandeurs turbulentes ne sont pas impactées par une modification du vecteur α. De telle sorte que les grandeurs turbulentes sont supposées constantes au cours du processus d'optimisation. En particulier, la viscosité $\mu + \mu_T$ est supposée constante par rapport à α et les efforts aérodynamiques transmis à la poutre ne sont fonction, en terme de champ aérodynamique, que des cinq grandeurs conservatives ρ_c, $(\rho u)_c$, $(\rho v)_c$, $(\rho w)_c$, $(\rho e)_c$. Les dérivées partielles des efforts transmis à la poutre par rapport au champ aérodynamique conservatif à l'intérieur du domaine fluide se calculent précisément de la manière suivante, où $1 \leq c \leq n_{cell}$:

$$\begin{cases} \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial \rho_c} = \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_c}{\partial \rho_c} n_{lz} - S_l^i \mu_c \frac{2S_l}{\mathrm{Vol}_c} \frac{(\rho w)_c - (\rho w)_c n_{lz}}{\rho_c^2} \\ \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (\rho u)_c} = \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_c}{\partial (\rho u)_c} n_{lz} \\ \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (\rho v)_c} = \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_c}{\partial (\rho v)_c} n_{lz} \\ \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (\rho w)_c} = \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_c}{\partial (\rho w)_c} n_{lz} + S_l^i \mu_c \frac{2S_l}{\mathrm{Vol}_c} \frac{1 - n_{lz}}{\rho_c} \\ \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial (\rho e)_c} = \displaystyle\sum_{l=1}^{n_f^I} S_l^i \frac{\partial p_c}{\partial (\rho e)_c} n_{lz} \end{cases} \tag{3.41}$$

$$
\left\{
\begin{array}{rcl}
\dfrac{\partial\overrightarrow{\mathcal{M}_i}}{\partial\rho_c} &=& \displaystyle\sum_{l=1}^{n_f^I}\overrightarrow{P_iG_l}\wedge\left(S_l^i\frac{\partial p_c}{\partial\rho_c}\vec{n}_l\right.\\
&& \left.-\,S_l^i\mu_c\dfrac{2S_l}{\mathrm{Vol}_c}\left(\dfrac{(\rho u)_c-(\rho u)_c\,n_{lx}}{\rho_c^2}\vec{x}+\dfrac{(\rho v)_c-(\rho v)_c\,n_{ly}}{\rho_c^2}\vec{y}+\dfrac{(\rho w)_c-(\rho w)_c\,n_{lz}}{\rho_c^2}\vec{z}\right)\right)\\
\dfrac{\partial\overrightarrow{\mathcal{M}_i}}{\partial(\rho u)_c} &=& \displaystyle\sum_{l=1}^{n_f^I}\overrightarrow{P_iG_l}\wedge\left(S_l^i\frac{\partial p_c}{\partial(\rho u)_c}\vec{n}_l+\;S_l^i\mu_c\frac{2S_l}{\mathrm{Vol}_c}\frac{1-n_{lx}}{\rho_c}\vec{x}\right)\\
\dfrac{\partial\overrightarrow{\mathcal{M}_i}}{\partial(\rho u)_c} &=& \displaystyle\sum_{l=1}^{n_f^I}\overrightarrow{P_iG_l}\wedge\left(S_l^i\frac{\partial p_c}{\partial(\rho u)_c}\vec{n}_l+\;S_l^i\mu_c\frac{2S_l}{\mathrm{Vol}_c}\frac{1-n_{ly}}{\rho_c}\vec{y}\right)\\
\dfrac{\partial\overrightarrow{\mathcal{M}_i}}{\partial(\rho u)_c} &=& \displaystyle\sum_{l=1}^{n_f^I}\overrightarrow{P_iG_l}\wedge\left(S_l^i\frac{\partial p_c}{\partial(\rho u)_c}\vec{n}_l+\;S_l^i\mu_c\frac{2S_l}{\mathrm{Vol}_c}\frac{1-n_{lz}}{\rho_c}\vec{z}\right)\\
\dfrac{\partial\overrightarrow{\mathcal{M}_i}}{\partial(\rho e)_c} &=& \displaystyle\sum_{l=1}^{n_f^I}\overrightarrow{P_iG_l}\wedge\left(S_l^i\frac{\partial p_c}{\partial(\rho e)_c}\vec{n}_l\right)
\end{array}
\right. \tag{3.42}
$$

Les dérivées partielles de la pression p_c par rapport aux grandeurs conservatives dans la cellule adjacente sont calculées en utilisant les équations (3.39).

Seules les sollicitations d'origine aérodynamique en flexion et en torsion sont transmises à la poutre. Les dérivées partielles de ce chargement par rapport au champ aérodynamique sont calculées à partir des formules ci-dessus en utilisant de plus les relations (3.40).

3.5.4 Calcul du terme $(\partial L/\partial W^b) * (\partial W^b/\partial X)$

Le comportement du fluide est modélisé par les équations d'Euler

Le champ aérodynamique transmis à l'interface I_l ne dépend, en terme de coordonnées, que des coordonnées du vecteur normal à l'interface I_l, c'est-à-dire des coordonnées des huit noeuds N_1^l, N_2^l, N_3^l, N_4^l sommets de l'interface I_l (voir figure 3.1). Lorsque le comportement du fluide est modélisé par les équations d'Euler, ce terme est égal à$(X=\left\{x_r^l,y_r^l,z_r^l,r\in[1,4]\right\})$:

$$
r\in[1,4],\quad
\left\{
\begin{array}{rcl}
\dfrac{\partial\overrightarrow{\mathcal{F}_i}}{\partial W^b}\dfrac{\partial W^b}{\partial X} &=& \displaystyle\sum_{l=1}^{n_f^I}S_l^i\frac{\partial p_l}{\partial W^b}\frac{\partial W^b}{\partial X}n_{lz}\\
\dfrac{\partial\overrightarrow{\mathcal{M}_i}}{\partial W^b}\dfrac{\partial W^b}{\partial X} &=& \displaystyle\sum_{l=1}^{n_f^I}\overrightarrow{P_iG_l}\wedge S_l^i\frac{\partial p_l}{\partial W^b}\frac{\partial W^b}{\partial X}\vec{n}_l
\end{array}
\right. \tag{3.43}
$$

avec

$$r \in [1,4], \quad \left\{ \begin{array}{rcl} \dfrac{\partial p_l}{\partial x_r^l} & = & p_c - c_c \left((\rho u)_c \dfrac{\partial n_{lx}}{\partial x_r^l} + (\rho v)_c \dfrac{\partial n_{ly}}{\partial x_r^l} + (\rho w)_c \dfrac{\partial n_{lz}}{\partial x_r^l} \right) \\ \dfrac{\partial p_l}{\partial y_r^l} & = & p_c - c_c \left((\rho u)_c \dfrac{\partial n_{lx}}{\partial y_r^l} + (\rho v)_c \dfrac{\partial n_{ly}}{\partial y_r^l} + (\rho w)_c \dfrac{\partial n_{lz}}{\partial y_r^l} \right) \\ \dfrac{\partial p_l}{\partial z_r^l} & = & p_c - c_c \left((\rho u)_c \dfrac{\partial n_{lx}}{\partial z_r^l} + (\rho v)_c \dfrac{\partial n_{ly}}{\partial z_r^l} + (\rho w)_c \dfrac{\partial n_{lz}}{\partial z_r^l} \right) \end{array} \right. . \tag{3.44}$$

Seules les sollicitations d'origine aérodynamique en flexion et en torsion sont transmises à la poutre. La dérivée partielle de ces termes par rapport aux coordonnées du maillage du domaine fluide via le champ aérodynamique transmis à l'interface I_l est calculée en utilisant les égalités suivantes ($X = \left\{x_r^l, y_r^l, z_r^l, r \in [1,4]\right\}$) :

$$\left\{ \begin{array}{rcl} \dfrac{\partial F_{iz}}{\partial W^b}\dfrac{\partial W^b}{\partial X} & = & \dfrac{\partial(\overrightarrow{\mathcal{F}_i}.\vec{z})}{\partial W^b}\dfrac{\partial W^b}{\partial X} = \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial W^b}\dfrac{\partial W^b}{\partial X}.\vec{z} \\ \dfrac{\partial M_{ix}}{\partial W^b}\dfrac{\partial W^b}{\partial X} & = & \dfrac{\partial(\overrightarrow{\mathcal{M}_i}.\vec{x})}{\partial W^b}\dfrac{\partial W^b}{\partial X} = \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial W^b}\dfrac{\partial W^b}{\partial X}.\vec{x} \\ \dfrac{\partial M_{iy}}{\partial W^b}\dfrac{\partial W^b}{\partial X} & = & \dfrac{\partial(\overrightarrow{\mathcal{M}_i}.\vec{y})}{\partial W^b}\dfrac{\partial W^b}{\partial X} = \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial W^b}\dfrac{\partial W^b}{\partial X}.\vec{y} \end{array} \right. . \tag{3.45}$$

Le comportement du fluide est modélisé par les équations RANS

Le champ aérodynamique transmis à l'interface I_l dépend des coordonnées des huit noeuds N_1^l, N_2^l, N_3^l, N_4^l, N_5^l, N_6^l, N_7^l et N_8^l (et non plus de quatre comme pour les équations d'Euler) délimitant la cellule adjacente au maillage de surface $\mathcal{F}^I$ en I_l. Dans ce cas, cette dérivée est cependant égale à ($X = \left\{x_r^l, y_r^l, z_r^l, r \in [1,8]\right\}$) :

$$\left\{ \begin{array}{l} \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial W^b}\dfrac{\partial W^b}{\partial X} = \displaystyle\sum_{l=1}^{n_f^I} S_l^i \dfrac{\partial p_l}{\partial W^b}\dfrac{\partial W^b}{\partial X}\,\vec{n}_l + S_l^i \mu_c \dfrac{\partial((2S_l)/(\mathrm{Vol}_c))}{\partial W^b}\dfrac{\partial W^b}{\partial X}\left(\dfrac{(\rho u)_c - (\rho u)_c (\partial n_{lx})/(\partial X)}{\rho_c}\,\vec{x} + \right. \\ \qquad \left. \dfrac{(\rho v)_c - (\rho v)_c (\partial n_{ly})/(\partial X)}{\rho_c}\,\vec{y} + \dfrac{(\rho w)_c - (\rho w)_c (\partial n_{lz})/(\partial X)}{\rho_c}\,\vec{z} \right) \\ \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial W^b}\dfrac{\partial W^b}{\partial X} = \displaystyle\sum_{l=1}^{n_f^I} \overrightarrow{P_iG_l} \wedge S_l^i \dfrac{\partial p_l}{\partial W^b}\dfrac{\partial W^b}{\partial X}\,\vec{n}_l + \\ \qquad \overrightarrow{P_iG_l} \wedge S_l^i \mu_c \dfrac{\partial((2S_l)/(\mathrm{Vol}_c))}{\partial W^b}\dfrac{\partial W^b}{\partial X}\left(\dfrac{(\rho u)_c - (\rho u)_c (\partial n_{lx})/(\partial X)}{\rho_c}\,\vec{x} + \right. \\ \qquad \left. \dfrac{(\rho v)_c - (\rho v)_c (\partial n_{ly})/(\partial X)}{\rho_c}\,\vec{y} + \dfrac{(\rho w)_c - (\rho w)_c (\partial n_{lz})/(\partial X)}{\rho_c}\,\vec{z} \right) \end{array} \right. \tag{3.46}$$

La dérivée partielle du chargement aérodynamique transmis à la poutre par rapport aux coordonnées du maillage du domaine fluide via le champ aérodynamique transmis à l'interface I_l est calculée en utilisant les égalités données par les équations (3.45).

3.5.5 Calcul du terme $(\partial L/\partial X)$

Le comportement du fluide est modélisé par les équations d'Euler

De façon directe, le champ aérodynamique transmis à l'interface I_l ne dépend, en terme de coordonnées, que de la surface intersectée S_i^l (voir figure 2.7) et des coordonnées du vecteur normal à l'interface I_l. Ces deux grandeurs ne sont elles-même fonctions que des coordonnées des quatre noeuds N_1^l, N_2^l, N_3^l et N_4^l, sommets de l'interface I_l (voir figure 3.1). Lorsque le comportement du fluide est modélisé par les équations d'Euler, la dérivée partielle $\dfrac{\partial L}{\partial X}$ est égale à $(X = \left\{x_r^l, y_r^l, z_r^l, r \in [1,4]\right\})$:

$$\begin{cases} \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial X} = \displaystyle\sum_{l=1}^{n_f^I} \dfrac{\partial S_l^i}{\partial X}\left(p_l - p_\infty\right)\vec{n}_l + S_l^i\left(p_l - p_\infty\right)\dfrac{\partial \vec{n}_l}{\partial X} \\ \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial X} = \displaystyle\sum_{l=1}^{n_f^I} \dfrac{\partial \overrightarrow{P_iG_l}}{\partial X} \wedge S_l^i\left(p_l - p_\infty\right)\vec{n}_l + \overrightarrow{P_iG_l} \wedge \dfrac{\partial S_l^i}{\partial X}\left(p_l - p_\infty\right)\vec{n}_l + \overrightarrow{P_iG_l} \wedge S_l^i\left(p_l - p_\infty\right)\dfrac{\partial \vec{n}_l}{\partial X} \end{cases} . \tag{3.47}$$

La dérivée partielle du chargement aérodynamique transmis à la poutre par rapport aux coordonnées du maillage du domaine fluide est calculée en utilisant les égalités données par les équations (3.45).

Le comportement du fluide est modélisé par les équations RANS

Comme lorsque l'écoulement du fluide est modélisé par les équations d'Euler, le champ aérodynamique transmis à l'interface I_l ne dépend, en terme de coordonnées et de façon directe, que de la surface intersectée S_i^l (voir figure 2.7) et des coordonnées du vecteur normal à l'interface I_l, c'est-à-dire des coordonnées des quatre noeuds N_1^l, N_2^l, N_3^l et N_4^l. Ces deux grandeurs ne sont elles-même fonctions que des coordonnées des quatre noeuds N_1^l, N_2^l, N_3^l et N_4^l. Lorsque le comportement du fluide est modélisé par les équations RANS, la dérivée partielle $\dfrac{\partial L}{\partial X}$ est égale à $(X = \left\{x_r^l, y_r^l, z_r^l, r \in [1,4]\right\})$:

$$\begin{cases} \dfrac{\partial \overrightarrow{\mathcal{F}_i}}{\partial X} = \displaystyle\sum_{l=1}^{n_f^I} \dfrac{\partial S_l^i}{\partial X}\left(p_l - p_\infty\right)\vec{n}_l + S_l^i\left(p_l - p_\infty\right)\dfrac{\partial \vec{n}_l}{\partial X} \\ \qquad + \dfrac{\partial S_l^i}{\partial X}\mu_c \dfrac{2S_l}{\mathrm{Vol}_c}\left(\dfrac{(\rho u)_c - (\rho u)_c\, n_{lx}}{\rho_c}\vec{x} + \dfrac{(\rho v)_c - (\rho v)_c\, n_{ly}}{\rho_c}\vec{y} + \dfrac{(\rho w)_c - (\rho w)_c\, n_{lz}}{\rho_c}\vec{z}\right) \\ \dfrac{\partial \overrightarrow{\mathcal{M}_i}}{\partial X} = \displaystyle\sum_{l=1}^{n_f^I} \dfrac{\partial \overrightarrow{P_iG_l}}{\partial X} \wedge S_l^i(p_l - p_\infty)\vec{n}_l + \overrightarrow{P_iG_l} \wedge \dfrac{\partial S_l^i}{\partial X}(p_l - p_\infty)\vec{n}_l + \overrightarrow{P_iG_l} \wedge S_l^i(p_l - p_\infty)\dfrac{\partial \vec{n}_l}{\partial X} \\ \qquad + \overrightarrow{P_iG_l} \wedge \dfrac{\partial S_l^i}{\partial X}\mu_c \dfrac{2S_l}{\mathrm{Vol}_c}\left(\dfrac{(\rho u)_c - (\rho u)_c\, n_{lx}}{\rho_c}\vec{x} + \dfrac{(\rho v)_c - (\rho v)_c\, n_{ly}}{\rho_c}\vec{y} + \dfrac{(\rho w)_c - (\rho w)_c\, n_{lz}}{\rho_c}\vec{z}\right) \end{cases} \tag{3.48}$$

La dérivée partielle du chargement aérodynamique transmis à la poutre par rapport aux coordonnées du maillage du domaine fluide est calculée en utilisant les égalités données par les équations (3.45).

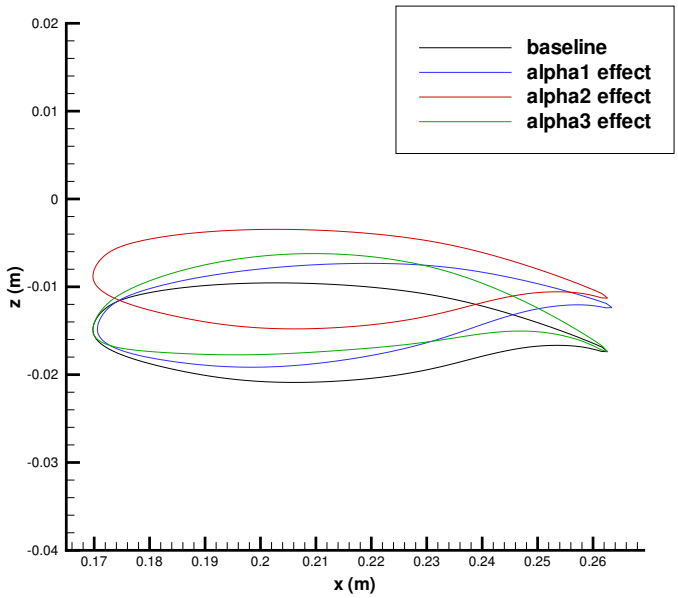

FIG. 3.2 – Effet des paramètres de forme α_1, α_2 et α_3 sur la géométrie des profils de l'aile en $y = 0.3m$ (configuration aile-fuselage F4).

3.6 Calculs des gradients pour trois cas tests aéronautiques

3.6.1 Aile-Fuselage F4

Les méthodes analytiques de calcul des gradients ont tout d'abord été testées et validées sur le cas aile-fuselage F4 introduit dans la section 2.6.1. Pour ce cas, le comportement du fluide est simulé par les équations d'Euler.

Trois paramètres de formes ont été choisis, aucun ne modifie la forme en plan de l'aile. On peut donc supposer que ces paramètres de forme n'ont pas d'influence sur les caractéristiques mécaniques de l'aile, et que l'on se situe bien dans le cadre fixé pour cette étude. Le premier paramètre de forme - noté α_1 - modifie le vrillage de l'aile rigide le long de son emplanture. Le second paramètre de forme - noté α_2 - introduit une bosse sur l'aile suivant son envergure. Enfin, le troisième paramètre de forme - noté α_3 - engendre une variation linéaire du maximum de cambrure des profils le long de l'envergure de l'aile. Les profils résultant d'une variation positive de α_1, α_2, et α_3 sont représentés par les profils respectivement bleu, rouget et vert sur la figure 3.2.

Les fonctions dont on cherche les dérivées par rapport à ces trois paramètre sont les coefficients aérodynamiques de traînée - noté C_d - et de portance - noté C_l.

Convergence de la méthode de l'équation linéarisée

Avant de calculer les valeurs de ces gradients, la méthode de l'équation linéarisée évalue les gradients des inconnues d'état du système aéroélastique (W et D) par rapport à chacun des trois paramètres de formes considérés (équation (3.26)). Pour la configuration aile-fuselage F4, le processus itératif de résolution du système bi-disciplinaire d'équations (3.26) converge en six itérations des solveurs fluide et structure avec $\epsilon_1 = 10^{-3}$ et $\epsilon_2 = 10^{-4}$ (en utilisant les notations de l'équation (3.27)). Les courbes 3.3, 3.5, 3.7, 3.4, 3.6 et 3.8 présentent l'évolution des gradients

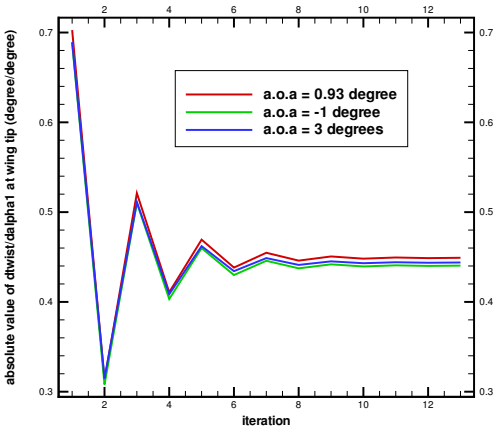

FIG. 3.3 – Convergence du gradient du champ de torsion par rapport à α_1, pour les trois incidences au saumon, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile-fuselage F4).

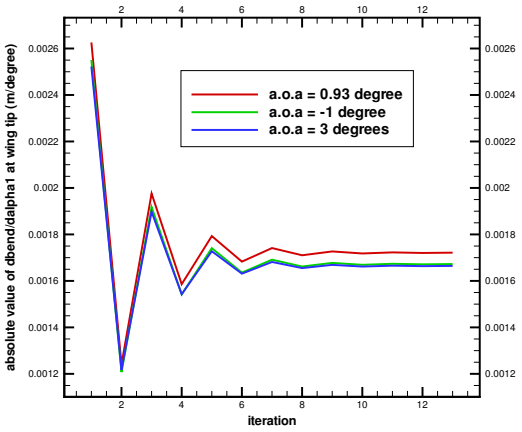

FIG. 3.4 – Convergence du gradient du champ de flexion par rapport à α_1 au saumon, pour les trois incidences, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile-fuselage F4).

du champ de torsion et du champ de flexion au saumon de l'aile au cours du processus itératif de résolution. Le saumon a été choisi car il s'agit du point où les déplacements de l'aile dus aux effets aéroélastiques sont les plus importants. Le nombre de pas du processus itératif est fixé au préalable et de façon arbitraire (guidée par l'expérience). Les courbes 3.9, 3.9 et 3.9 présentent l'évolution de la norme L_2 du membre de droite de la première équation de l'équation (a) du système d'équations (3.26) (équation correspondant à la dérivée de la relation de conservation de masse par rapport à α) au cours du processus de résolution du système (3.26). Ces courbes permettent de visualiser la convergence des gradients $\partial W/\partial\alpha_i$ et $\partial D/\partial\alpha_i$, $1 \leq i \leq 3$ au cours du processus de résolution.

Le calcul des gradients intermédiaires $\partial W/\partial\alpha_i$ et $\partial D/\partial\alpha_i$, $1 \leq i \leq 3$ peut être comparé au calcul par différences finies. Les figures 3.16, 3.12 et 3.13 comparent les gradients du champ de torsion au saumon de l'aile par rapport aux trois paramètres de forme étudiés (pour les trois conditions de vol considérées) calculés par la méthode de l'équation linéarisée d'une part et par différences finies d'autre part. Les figures 3.17, 3.14 et 3.15 comparent quant à elles les gradients du champ de flexion au saumon de l'aile par rapport aux trois paramètres de forme étudiés et pour les trois conditions de vol considérées, calculés par la méthode de l'équation linéarisée d'une part et par différences finies d'autre part. La demi-envergure de l'aile mesure $0.6m$. La partie des courbes s'étendant de $y = 0.6m$ à $y = 1m$ correspond à l'amortissement spatial de ces valeurs lors de la transmission du champ $(dD)/(d\alpha)$ au maillage du domaine $\mathcal{F}$ occupé par le fluide. L'écart absolu maximal observé entre les résultats donnés par ces deux méthodes de calcul est de 10%. Cette étape permet de s'assurer de la cohérence des calculs des gradients intermédiaires par la méthode de l'équation linéarisée.

La figure 3.16 met en évidence, pour le paramètre de forme α_2, un des problèmes liés à la méthode des différences finies et évoqué dans la section 3.3.1. En effet, lorsque le pas de la différence finie est trop grand, la précision de cette dernière est mauvaise, en revanche lorsque celui-ci est plus faible, les valeurs obtenues par différence finies sont bruitées. Le choix du pas approprié à l'évaluation d'une dérivée par différence finie se fait en localisant la zone de stabilité sur la courbe représentant la valeur approchée de la dérivée calculée par différence finie en fonction du pas de la différence finie. La figure 3.18 représente les valeurs approchées des dérivées $dC_d/d\alpha_3$

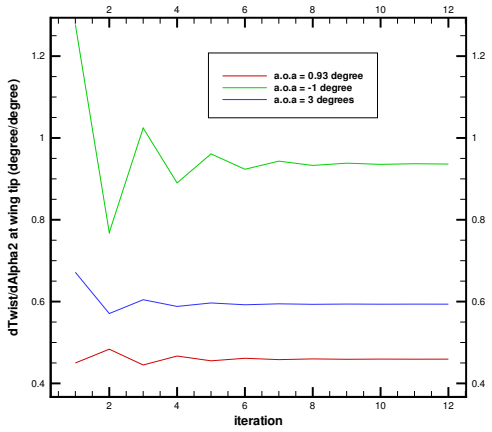

FIG. 3.5 – Convergence du gradient du champ de torsion par rapport à α_2, pour les trois incidences au saumon, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile-fuselage F4).

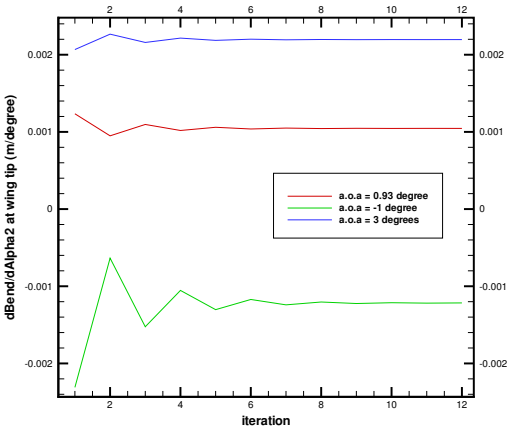

FIG. 3.6 – Convergence du gradient du champ de flexion par rapport à α_2 au saumon, pour les trois incidences, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile-fuselage F4).

et $dC_l/d\alpha_3$ en fonction du pas de la différence finie. Dans ce cas, le pas retenu est $\delta\alpha_3 = 0.0025m$.

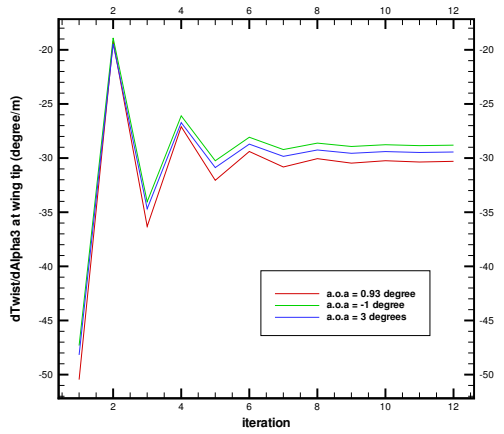

FIG. 3.7 – Convergence du gradient du champ de torsion par rapport à α_3, pour les trois incidences au saumon, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile-fuselage F4).

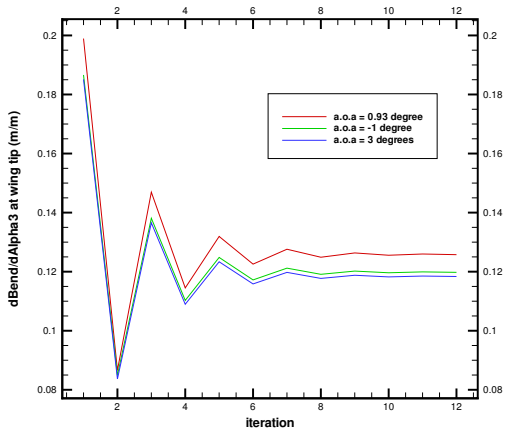

FIG. 3.8 – Convergence du gradient du champ de flexion par rapport à α_3 au saumon, pour les trois incidences, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile-fuselage F4).

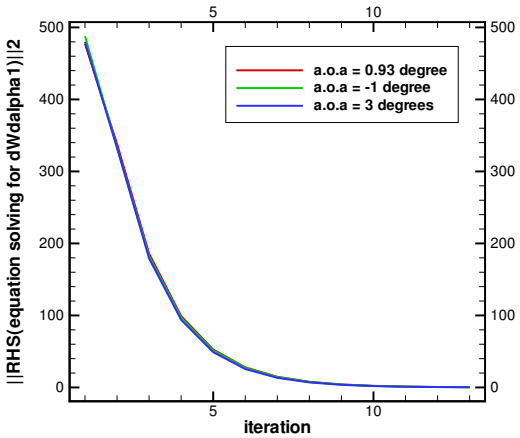

FIG. 3.9 – Convergence de la norme L_2 du résidu de l'équation (a) du système d'équations (3.24) (correspondant à la dérivée de la relation de conservation de la masse) au cours du processus itératif de résolution, pour les trois incidences et α_1 (configuration aile-fuselage F4).

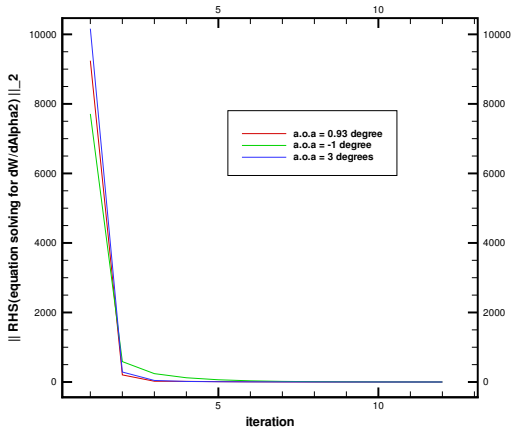

FIG. 3.10 – Convergence de la norme L_2 du résidu de l'équation (a) du système d'équations (3.24) (correspondant à la dérivée de la relation de conservation de la masse) au cours du processus itératif de résolution, pour les trois incidences et α_2 (configuration aile-fuselage F4).

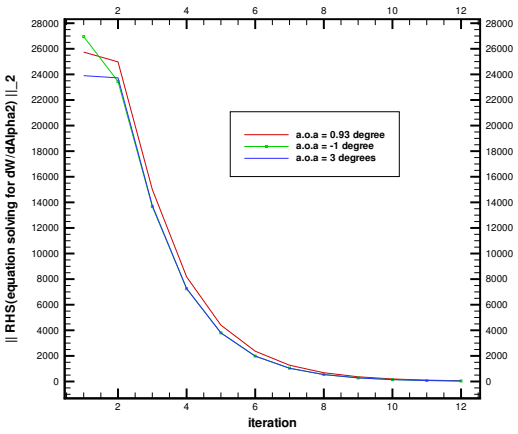

FIG. 3.11 – Convergence de la norme L_2 du résidu de l'équation (a) du système d'équations (3.24) (correspondant à la dérivée de la relation de conservation de la masse) au cours du processus itératif de résolution, pour les trois incidences et α_3 (configuration aile-fuselage F4).

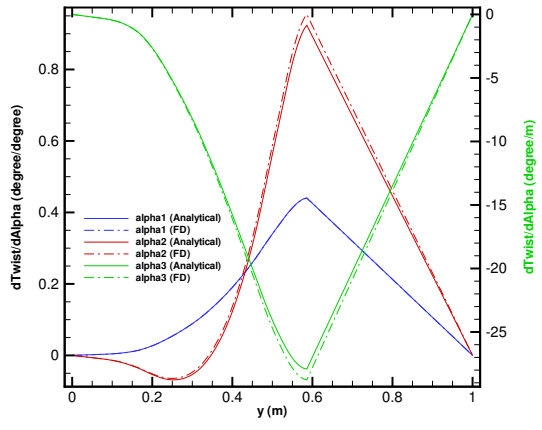

FIG. 3.12 – Gradient du champ de torsion le long de la poutre par rapport à α_1, α_2 et α_3 (incidence de $-1°$) (configuration aile-fuselage F4).

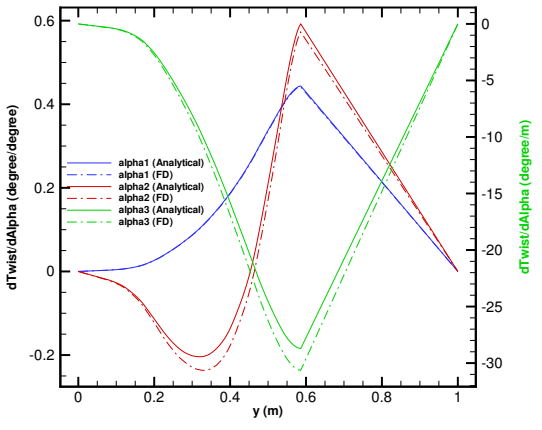

FIG. 3.13 – Gradient du champ de torsion le long de la poutre par rapport à α_1, α_2 et α_3 (incidence de $3°$) (configuration aile-fuselage F4).

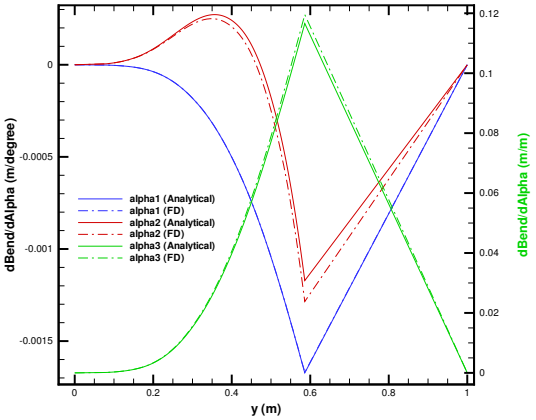

FIG. 3.14 – Gradient du champ de flexion le long de la poutre par rapport à α_1, α_2 et α_3 (incidence de $-1°$) (configuration aile-fuselage F4).

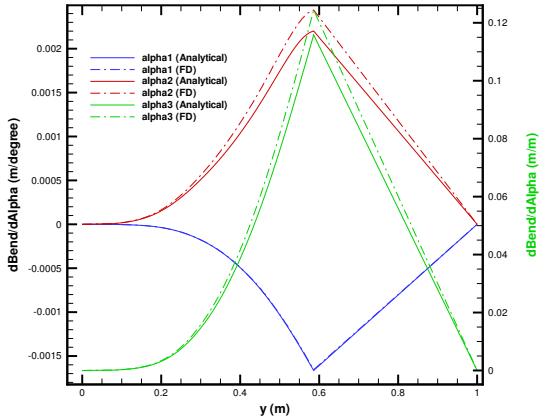

FIG. 3.15 – Gradient du champ de flexion le long de la poutre par rapport à α_1, α_2 et α_3 (incidence de $3°$) (configuration aile-fuselage F4).

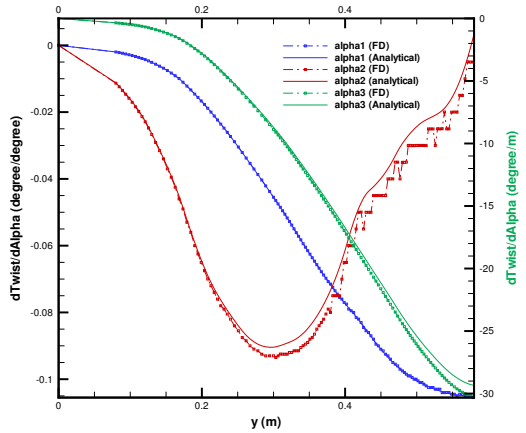

FIG. 3.16 – Gradient du champ de torsion le long de la poutre par rapport à α_1, α_2 et α_3 (incidence de $0.93°$) (configuration aile-fuselage F4).

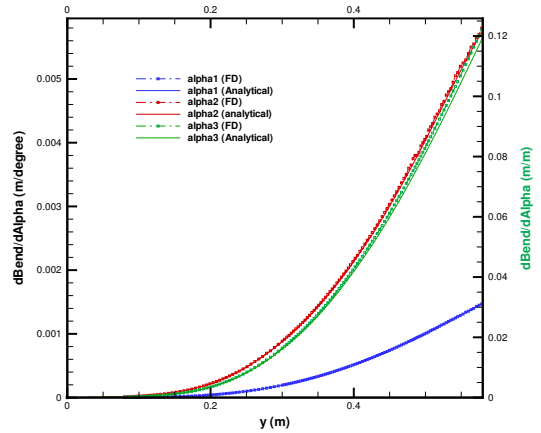

FIG. 3.17 – Gradient du champ de flexion le long de la poutre par rapport à α_1, α_2 et α_3 (incidence de $0.93°$) (configuration aile-fuselage F4).

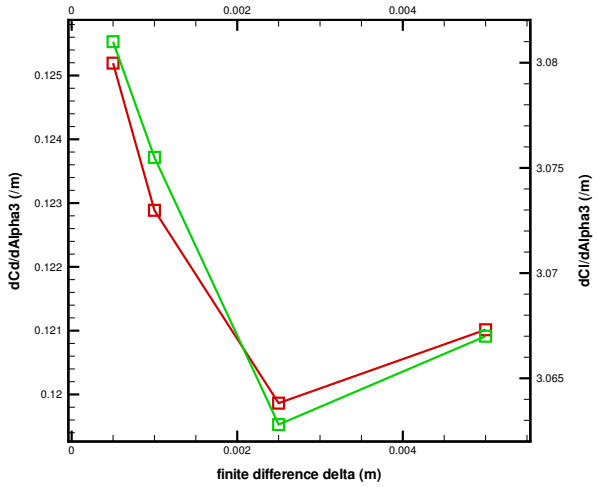

FIG. 3.18 – Valeurs approchées des dérivées des coefficients de traînée et de portance en fonction du pas de la différence finie.

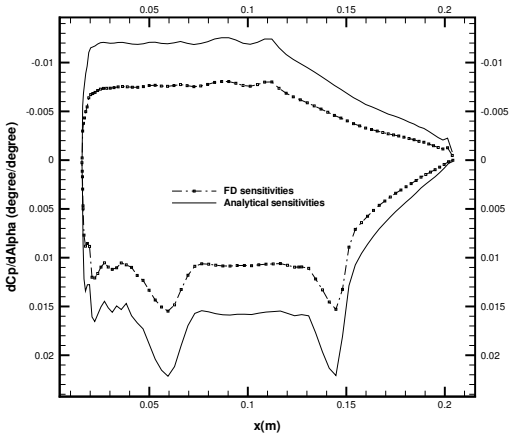

FIG. 3.19 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.1m$ et incidence de $-1°$.

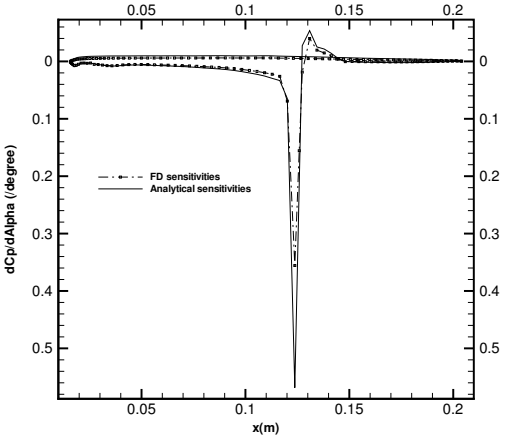

FIG. 3.20 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.1m$ et incidence de $3°$.

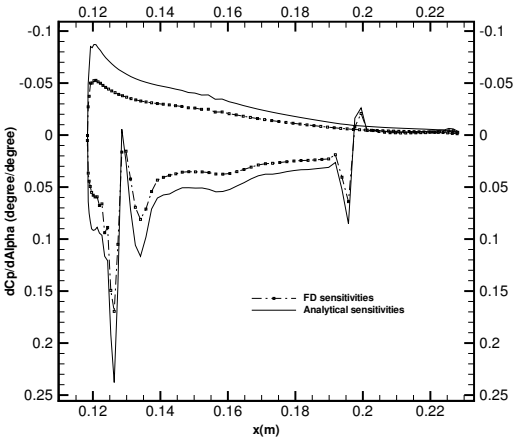

FIG. 3.21 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.3m$ et incidence de $-1°$.

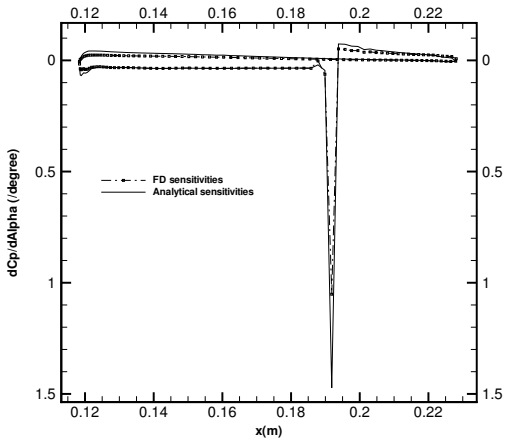

FIG. 3.22 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.3m$ et incidence de $3°$.

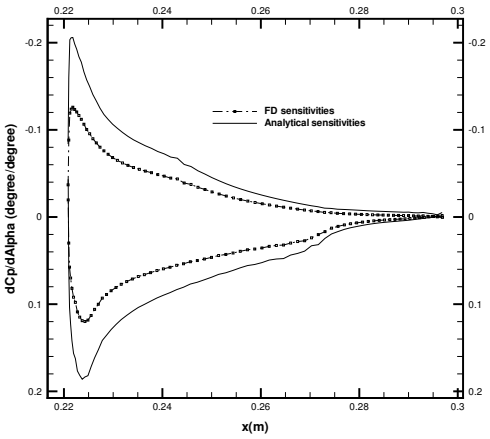

FIG. 3.23 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.5m$ et incidence de $-1°$.

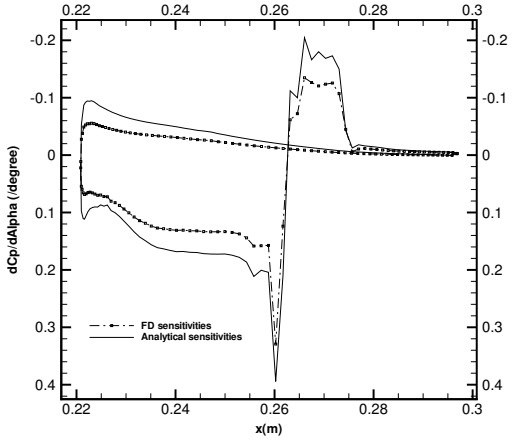

FIG. 3.24 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.5m$ et incidence de $3°$.

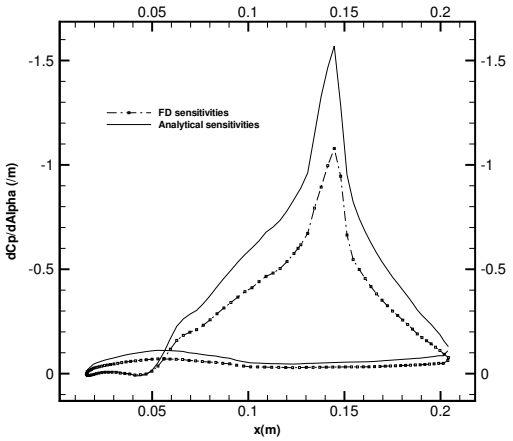

FIG. 3.25 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.1m$ et incidence de $-1°$.

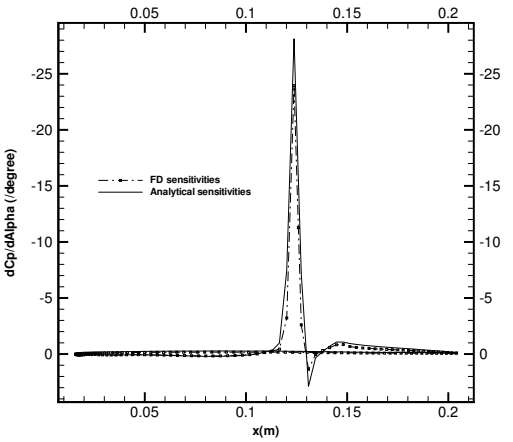

FIG. 3.26 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.1m$ et incidence de $3°$.

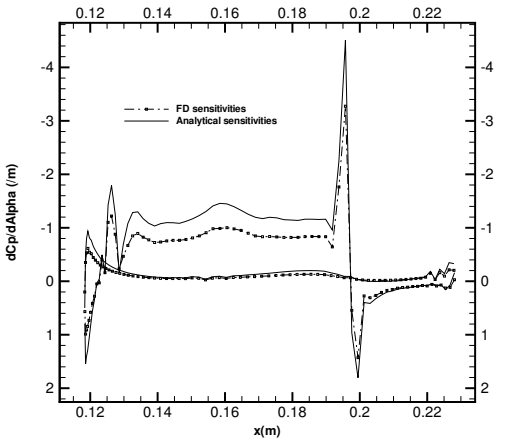

FIG. 3.27 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.3m$ et incidence de $-1°$.

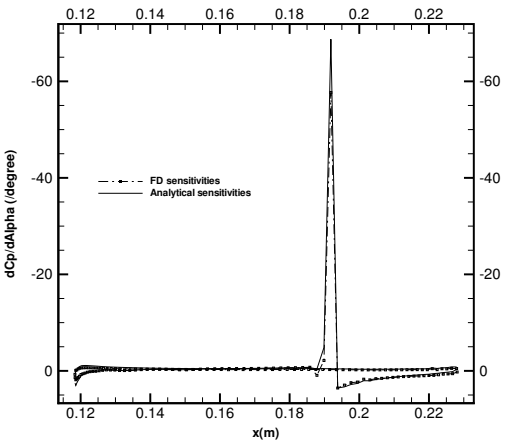

FIG. 3.28 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.3m$ et incidence de $3°$.

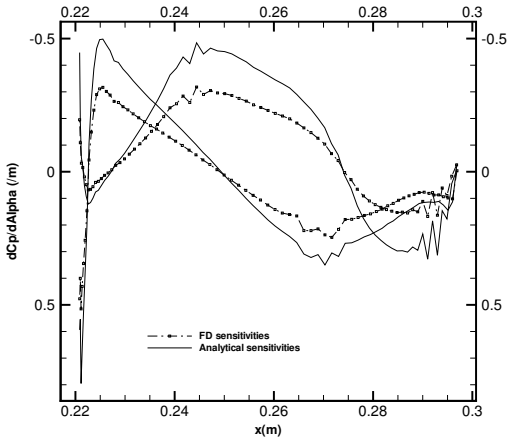

FIG. 3.29 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.5m$ et incidence de $-1°$.

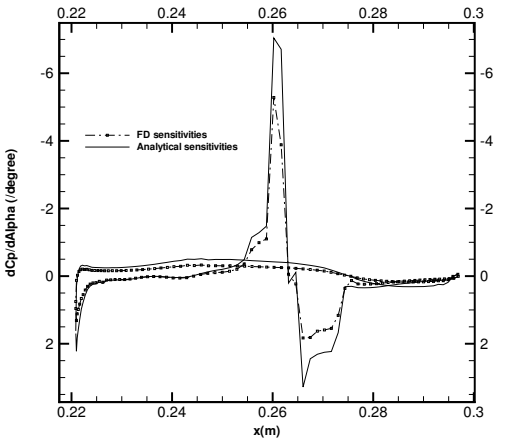

FIG. 3.30 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.5m$ et incidence de $3°$.

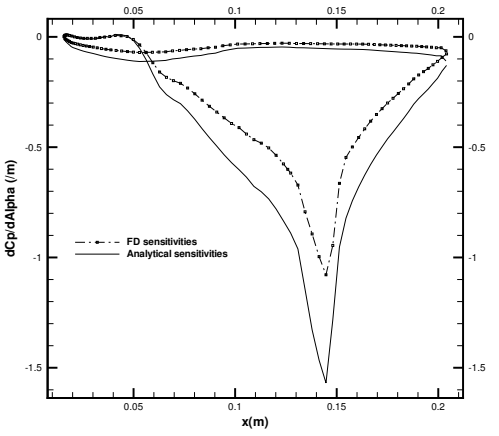

FIG. 3.31 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.1m$ et incidence de $-1°$.

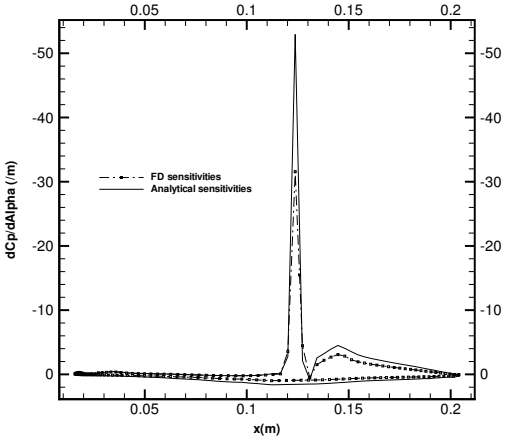

FIG. 3.32 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.1m$ et incidence de $3°$.

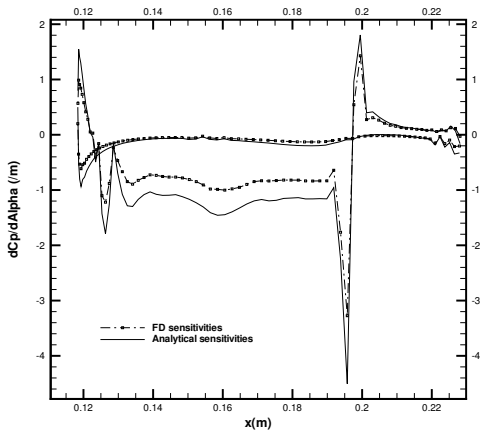

FIG. 3.33 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.3m$ et incidence de $-1°$.

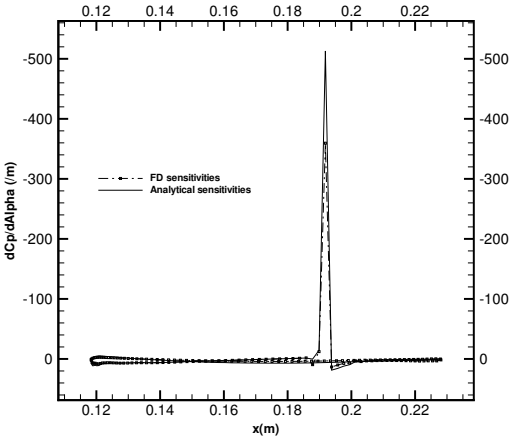

FIG. 3.34 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.3m$ et incidence de $3°$.

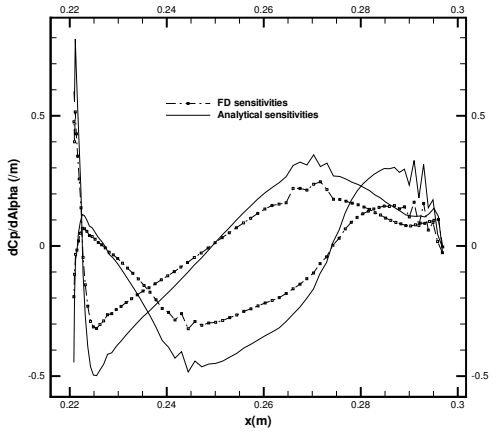

FIG. 3.35 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.5m$ et incidence de $-1°$.

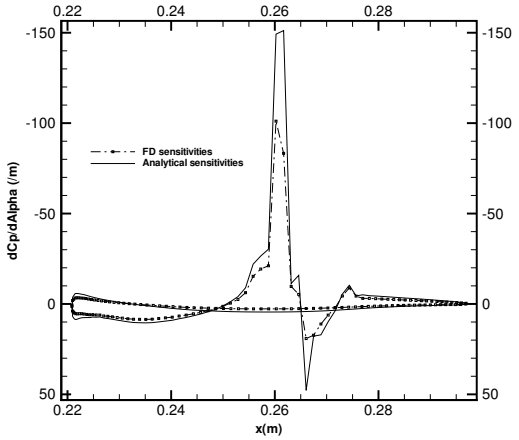

FIG. 3.36 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.5m$ et incidence de $3°$.

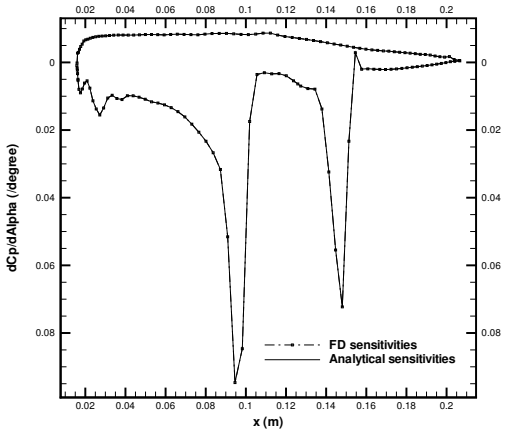

FIG. 3.37 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.1m$ et incidence de $0.93°$.

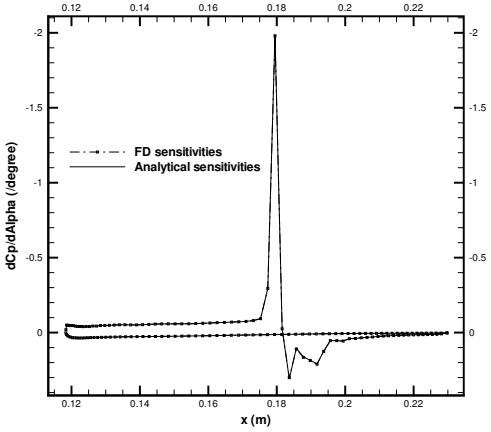

FIG. 3.38 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.3m$ et incidence de $0.93°$.

Il est plus difficile de comparer le champ volumique $\partial W/\partial \alpha_i$, $i \in [1, 3]$, calculé par la méthode de l'équation linéarisée au champ évalué par différence finie. Pour se ramener à une représentation monodimensionnelle, on peut par exemple comparer le gradient du coefficient de pression C_p par rapport à α au niveau de différentes sections de l'aile pour les cellules du domaine fluide adjacent à la paroi. Ce dernier se calcule facilement à partir du champ W à l'équilibre aéroélastique et du

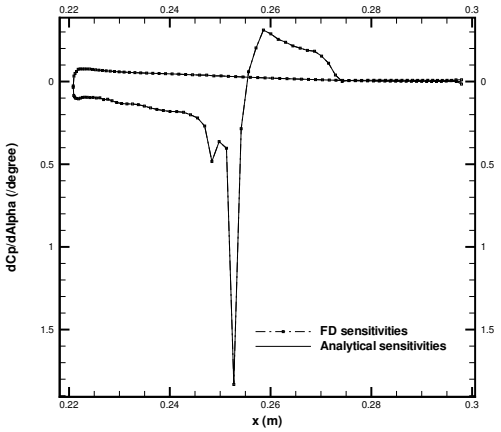

FIG. 3.39 – Configuration aile-fuselage F4 - radient du coefficient de pression C_p par rapport à α_1, coupe en $y = 0.5m$ et incidence de 0.93°.

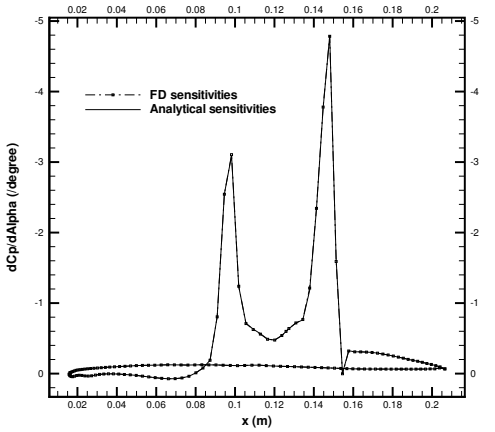

FIG. 3.40 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.1m$ et incidence de 0.93°.

champ $\partial W/\partial\alpha_i$, $i \in [1,3]$:

$$\frac{\partial C_p}{\partial \alpha_i} = (\gamma - 1)(\frac{\partial \rho_c}{\partial \alpha_i} e_c + \rho_c \frac{\partial e_c}{\partial \alpha_i} - \frac{(\rho u)_c \frac{\partial (\rho u)_c}{\partial \alpha_i} + (\rho v)_c \frac{\partial (\rho v)_c}{\partial \alpha_i} + (\rho w)_c \frac{\partial (\rho w)_c}{\partial \alpha_i}}{\rho_c}$$

$$+\frac{1}{2}\frac{(\rho u)_c^2 + (\rho v)_c^2 + (\rho w)_c^2}{\rho_c^2})/(\frac{1}{2}\rho_\infty^2 V_\infty^2)$$

Les gradients $\partial C_p/\partial\alpha_i$, $i \in [1,3]$ calculés en utilisant les gradients donnés par la méthode de l'équation linéarisée et ceux évalués par différences finies sont comparés pour trois profils situés respectivement à l'emplenture ($y = 0.1m$), à mi-envergure ($y = 0.3m$) et au saumon de l'aile ($y = 0.5m$). Ces résultats sont présentés par les figures 3.37, 3.38, 3.39, 3.40, 3.41, 3.42, 3.43, 3.44, et 3.45 pour les trois cas de vol considérés et les trois paramètres de forme étudiés.

Convergence de la méthode de l'équation adjointe

Comme pour la méthode de l'équation linéarisée, le processus itératif de résolution du système d'équations (3.31) converge rapidement : il n'a besoin que de six itérations entre les solveurs fluide et structure pour converger vers les valeurs finales des vecteurs adjoints avec $\epsilon_3 = 10^{-5}$ et $\epsilon_4 = 10^{-4}$ (en utilisant les notations de l'équation (3.32)). En revanche, à la différence de cette dernière, il n'est pas possible de comparer les vecteurs adjoints à des vecteurs adjoints calculés par différences finies. La seule comparaison possible avec la méthode des différences finies réside dans la comparaison des valeurs des gradients des fonctions dont on cherche à calculer les gradients. Il est toutefois possible de tracer l'évolution des vecteurs adjoints évalués pour visualiser la convergence du processus itératif. La convergence des vecteurs adjoints associés au champ de flexion au saumon de l'aile et correspondant aux coefficients C_d et C_l au cours du processus itératif est représentée par les figures 3.49 et 3.49. La convergence des vecteurs adjoints associés au champ de torsion au saumon de l'aile et correspondant aux coefficients C_d et C_l au cours du processus itératif est représentée par les figures 3.47 et 3.47. La convergence des

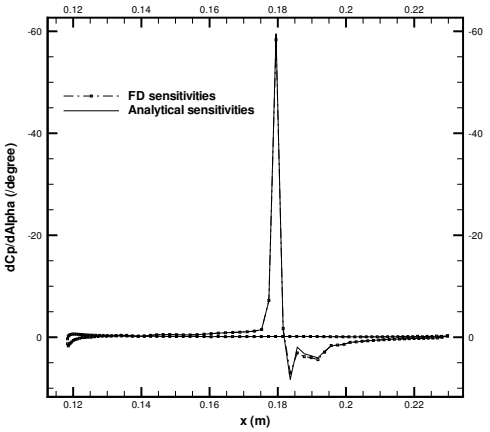

FIG. 3.41 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.3m$ et incidence de 0.93°.

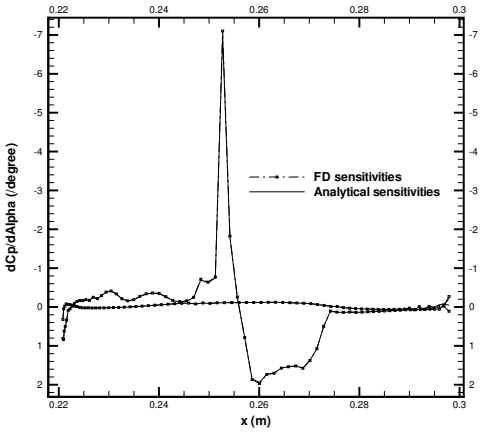

FIG. 3.42 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_2, coupe en $y = 0.5m$ et incidence de 0.93°.

vecteurs adjoints associés au champ aérodynamique W est illustrée grâce au tracé de la norme L_2 du membre de droite de la première équation du système (a) (état adjoint associé à la relation de conservation de la masse) du système d'équation (3.31) correspondant à C_l. Ce dernier est représenté par la figure 3.50.

Comparaison des valeurs de gradients obtenues par les différentes méthodes

Les gradients des coefficients de traînée et de portance sont assemblés en utilisant les dérivées partielles $\partial W/\partial\alpha_i$ et $\partial D/\partial\alpha_i$, $i \in [1,3]$, suivant la formule (3.6) pour la méthode de l'équation linéarisée ou en utilisant les vecteurs adjoints λ_f et λ_s associés à chaque système d'équations et correspondant à chacune des fonctions étudiées (coefficients C_d et C_l) suivant la formule (3.23) pour la méthode de l'équation adjointe. Ces gradients sont comparés aux valeurs calculées par différences finies à l'ordre 2. Les valeurs obtenues pour les trois conditions de vol étudiées sont assemblées dans les tableaux 3.3, 3.4 et 3.5. Les écarts relatifs (calculés par référence aux différence finies pour les méthodes de l'équation linéarisée et adjointe et par référence à la méthode de l'équation linéarisée dans le cas de l'équation adjointe) sont indiqués dans les tableaux 3.6, 3.7 et 3.8.

Les méthodes dites "exactes" de l'équation linéarisée et de l'équation adjointe permettent de

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_d/d\alpha_3$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$	$dC_l/d\alpha_3$
Méthode linéarisée	$3.14.10^{-3}$	$3.18.10^{-2}$	0.12	0.0396	0.358	3.04
Méthode adjointe	$3.14.10^{-3}$	$3.18.10^{-2}$	0.12	0.0396	0.358	3.04
Différences finies	$3.13.10^{-3}$	$3.18.10^{-2}$	0.12	0.0396	0.359	3.06

TAB. 3.3 – Gradient des coefficients aérodynamiques de traînée et de portance (incidence de 0.93°) (configuration aile-fuselage F4).

retrouver les valeurs prédites par différences finies à moins de 1%. Etant donnée la complexité de programmation de ces méthodes, ces résultats permettent de valider l'implémentation qui en a été

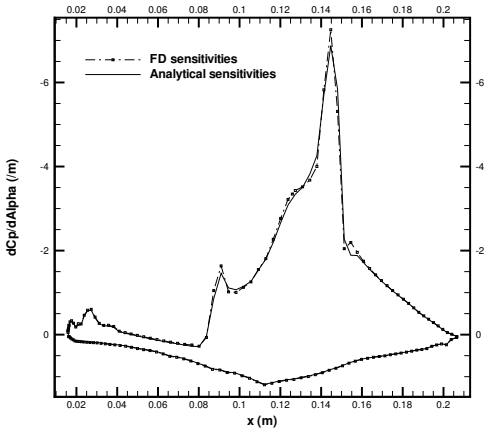

FIG. 3.43 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.1m$ et incidence de 0.93°.

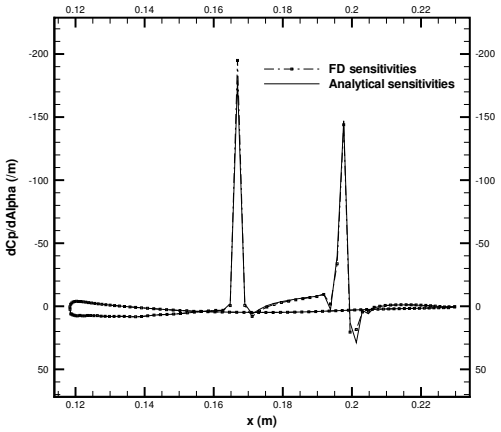

FIG. 3.44 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.3m$ et incidence de 0.93°.

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_d/d\alpha_3$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$	$dC_l/d\alpha_3$
Méthode linéarisée	$-7.79.10^{-4}$	$1.20.10^{-2}$	0.10	-0.029	0.216	2.93
Méthode adjointe	$-7.79.10^{-4}$	$1.20.10^{-2}$	0.10	-0.029	0.216	2.93
Différences finies	$-7.78.10^{-4}$	$1.20.10^{-2}$	0.10	-0.029	0.215	2.95

TAB. 3.4 – Gradient des coefficients aérodynamiques de traînée et de portance (incidence de $-1°$) (configuration aile-fuselage F4).

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_d/d\alpha_3$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$	$dC_l/d\alpha_3$
Méthode linéarisée	$-3.6.10^{-3}$	$3.97.10^{-2}$	0.10	-0.028	0.351	2.75
Méthode adjointe	$-3.6.10^{-3}$	$3.97.10^{-2}$	0.10	-0.028	0.351	2.75
Différences finies	$-3.7.10^{-3}$	$3.98.10^{-2}$	0.10	-0.028	0.353	2.76

TAB. 3.5 – Gradient des coefficients aérodynamiques de traînée et de portance (incidence de 3°) (configuration aile-fuselage F4).

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_d/d\alpha_3$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$	$dC_l/d\alpha_3$
Adjoint/Linéarisé	0%	0%	0%	0%	0%	0%
Linéarisé/DF	0.3%	0%	0%	0%	0.2%	0.6%
Adjoint/DF	0.3%	0%	0%	0%	0.2%	0.6%

TAB. 3.6 – Ecart relatif entre les différentes méthodes du calcul de gradient (incidence de 0.93°) (configuration aile-fuselage F4).

faite lorsque le comportement du fluide est simulé par les équations d'Euler ainsi que la fiabilité des méthodes analytiques (méthodes de l'équation linéarisée et adjointe). Les valeurs exactes des gradients ne sont pas celles calculées par différences finies. En effet, on est parvenu à faire converger la valeur des gradients en fonction du pas de la différence finie seulement pour le cas de vol à incidence 0.93° et pour le paramètre α_1. Toutes les autres valeurs de gradients calculées

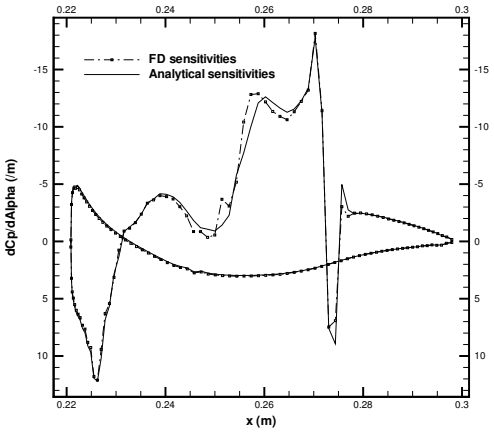

FIG. 3.45 – Configuration aile-fuselage F4 - Gradient du coefficient de pression C_p par rapport à α_3, coupe en $y = 0.5m$ et incidence de 0.93°.

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_d/d\alpha_3$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$	$dC_l/d\alpha_3$
Adjoint/Linéarisé	0%	0%	0%	0%	0%	0%
Linéarisé/DF	0.1%	0%	0%	0%	0.4%	0.6%
Adjoint/DF	0.1%	0%	0%	0%	0.4%	0.6%

TAB. 3.7 – Ecart relatif entre les différentes méthodes du calcul de gradient (incidence de $-1°$) (configuration aile-fuselage F4).

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_d/d\alpha_3$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$	$dC_l/d\alpha_3$
Adjoint/Linéarisé	0%	0%	0%	0%	0%	0%
Linéarisé/DF	2%	0.3%	0%	0%	0.5%	0.4%
Adjoint/DF	2%	0.3%	0%	0%	0.5%	0.4%

TAB. 3.8 – Ecart relatif entre les différentes méthodes du calcul de gradient (incidence de 3°) (configuration aile-fuselage F4).

par différences finies ont été interpolées sur une zone de stabilité de la fonction représentant la valeur de la différence en fonction du pas. L'objectif des calculs présentés ci-dessus était avant tout de retrouver, pour les gradients, des valeurs proches de celles fournies par un calcul par différences finies. Remarquons toutefois qu'en phase de croisière (cas de vol à incidence 0.93°) une augmentation du vrillage de l'aile rigide, c'est-à-dire une augmentation de α_1, ou une augmentation de la cambrure de l'aile, c'est-à-dire une augmentation de α_3, ont tendance à augmenter le chargement aérodynamique de la partie externe de l'aile, ce qui a tendance à augmenter les coefficients de portance et de traînée, en particulier à cause de l'augmentation de la traînée induite. D'autre part, une augmentation de α_1 et α_3 résulte également, à cause du couplage aéroélastique, en une augmentation de la flexion de l'aile et une diminution de la torsion de l'aile à l'équilibre aéroélastique. Cette tendance stabilisatrice peut être expliquée par le couplage géométrique entre les mouvements de flexion et de torsion propre aux ailes à flêche positive.

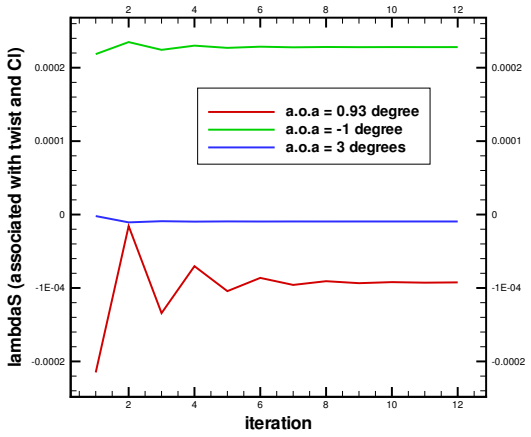

FIG. 3.46 – Convergence du vecteur adjoint associé au champ de torsion au saumon et correspondant à la fonction C_l au cours du processus itératif de résolution de l'équation (3.31) (configuration aile-fuselage F4).

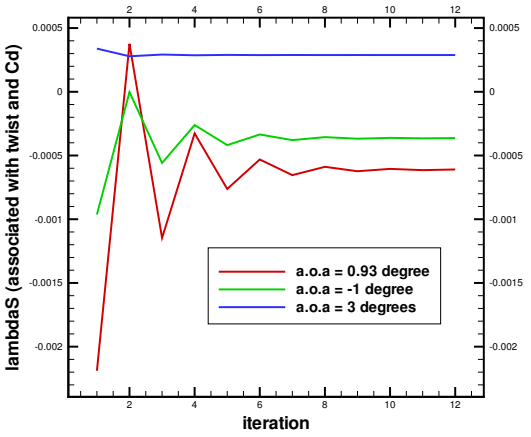

FIG. 3.47 – Convergence du vecteur adjoint associé au champ de torsion au saumon et correspondant à la fonction C_d au cours du processus itératif de résolution de l'équation (3.31) (configuration aile-fuselage F4).

Enfin, afin d'illustrer l'importance du couplage entre le fluide et la structure lors du calcul des gradients - même des fonctions purement aérodynamiques comme les coefficients de portance et de traînée - nous avons comparé, en phase de croisière, les valeurs des gradients de ces coefficients avec et sans prise en compte du couplage aéroélastique. Les valeurs de ces gradients ainsi que les écarts relatifs absolus (par référence aux calculs prenant en compte le couplage) sont résumés dans le tableau 3.9. Ces écarts peuvent être significatifs jusqu'à 50% pour la configuration envisagée.

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_d/d\alpha_3$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$	$dC_l/d\alpha_3$
pas de couplage	$2.34.10^{-3}$	$3.78.10^{-2}$	0.18	0.0402	0.327	4.25
couplage F/S	$3.14.10^{-3}$	$3.18.10^{-2}$	0.12	0.0396	0.358	3.04
écart relatif	25%	19%	50%	1.5%	9%	40%

TAB. 3.9 – Gradients des coefficients aérodynamiques calculés avec et sans prise en compte du couplage aéroélastique (configuration aile-fuselage F4).

3.6.2 Aile M6

Les méthodes analytiques pour le calcul des gradients nécessaires à l'optimisation de forme de systèmes couplés fluide-structure ont ensuite été testées et validées sur le cas monobloc de l'aile M6 introduit dans la section 2.24. Pour ce cas, le comportement du fluide est simulé par les équations RANS avec pour modèle de turbulence le modèle k-w Wilcox.

Deux paramètres de forme ont été choisis. Le premier - noté α_1 - introduit une variation linéaire du vrillage de la forme rigide de l'aile suivant son envergure et le second - noté α_2 - introduit une variation linéaire de la cambrure de l'aile suivant son envergure. Ces deux paramètres de forme n'ont pas d'impact sur la forme en plan de l'aile, de telle sorte qu'une variation de ces derniers ne change pas les caractéristiques de la structure. Ils permettent de respecter les hypothèses de cette étude.

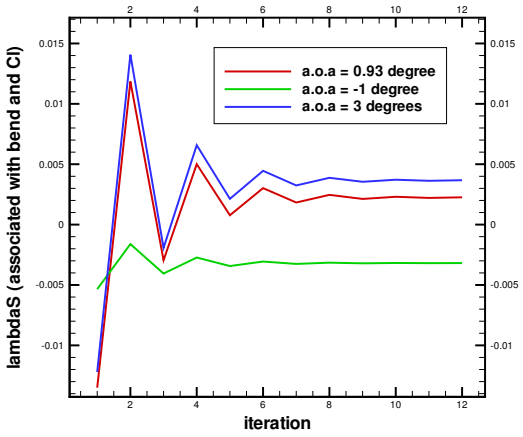

FIG. 3.48 – Convergence du vecteur adjoint associé au champ de flexion au saumon et correspondant à la fonction C_l au cours du processus itératif de résolution de l'équation (3.31) (configuration aile-fuselage F4).

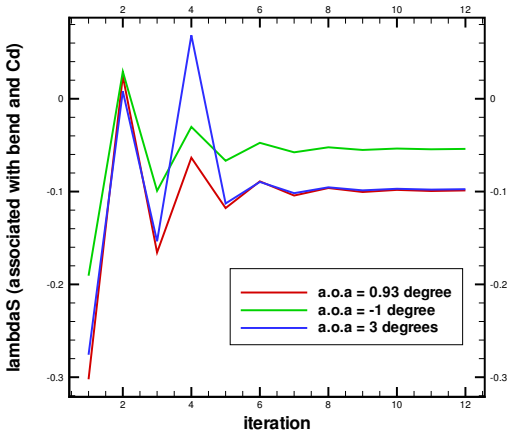

FIG. 3.49 – Convergence du vecteur adjoint associé au champ de flexion au saumon et correspondant à la fonction C_d au cours du processus itératif de résolution de l'équation (3.31) (configuration aile-fuselage F4).

Comme précédemment, les fonctions considérées sont d'origine purement aérodynamique puisqu'il s'agit des coefficients de traînéee et de portance.

Convergence de la méthode de l'équation linéarisée

Comme pour la configuration aile-fuselage, le processus itératif de résolution du système d'équation (3.26) converge rapidement, A partir de six itérations entre les solveurs fluide et structure, les valeurs des gradients $\partial W/\partial\alpha_i$ et $\partial D/\partial\alpha_i$, $i = 1, 2$, peuvent être considérées comme convergées (avec $\epsilon_1 = 10^{-6}$ et $\epsilon_2 = 10^{-7}$, en utilisant les notations de l'équation (3.27)). L'évolution des gradients du champ de flexion au saumon par rapport aux paramètres de forme α_1 et α_2 au cours du processus itératif de résolution est illustré par la figure 3.52, celle des gradients du champ de torsion au saumon par rapport aux paramètres de forme α_1 et α_2 par la figure 3.51. Enfin la convergence de $\partial W/\partial\alpha_1$ et $\partial W/\partial\alpha_2$ est illustrée figure 3.53 par le tracé de la norme L_2 du membre de droite de la première équation de l'équation (a) du système d'équations (3.26) (équation correspondant à la dérivée de la relation de conservation de masse par rapport à α) au cours du processus itératif de résolution.

Les valeurs des gradients intermédiaires $\partial W/\partial\alpha_i$ et $\partial D/\partial\alpha_i$, $i = 1, 2$, sont comparées aux évaluations par différences finies. Les figures 3.55 et 3.57 comparent les gradients du champ de flexion au saumon de l'aile par rapport α_1 et α_2 calculés par la méthode de l'équation linéarisée d'une part et par différences finies d'autre part. De même, les figures 3.54 et 3.56 comparent les gradients du champ de torsion au saumon de l'aile par rapport à α_1 et α_2 calculés par la méthode de l'équation linéarisée d'une part et par différences finies d'autre part. La comparaison des champs $\partial W/\partial\alpha_1$, $i = 1, 2$, aux valeurs calculées par différences finies se fait par l'intermédiaire de la comparaison du gradient du coefficient de pression à la paroi de l'aile. Les gradients $\partial C_p/\partial\alpha_i$, $i = 1, 2$ calculés en utilisant les gradients donnés par la méthode de l'équation linéarisée et ceux évalués par différences finies sont comparés pour trois profils situés respectivement à l'emplenture ($y = 0.2m$), à mi-envergure ($y = 0.8m$) et au saumon de l'aile ($y = 1.4m$). Ces comparaisons sont présentées sur les figures 3.58, 3.60 et 3.62 pour le paramètre de forme α_1 et

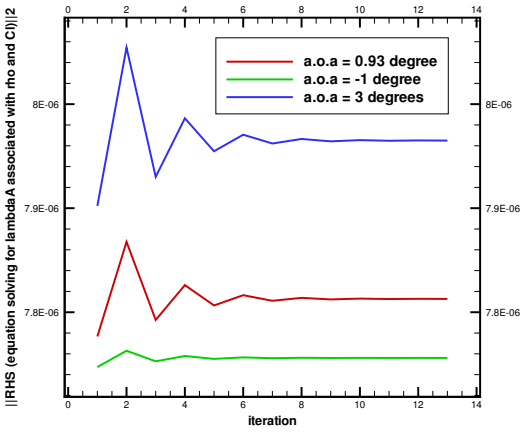

FIG. 3.50 – Convergence de la norme L_2 du résidu de l'équation (a) du système d'équations (3.31) associé à la valeur conservative ρ et correspondant à la fonction C_l au cours du processus itératif de résolution, pour les trois incidences (configuration aile-fuselage F4).

par les figures 3.59, 3.61 et 3.63 pour le paramètre de forme α_2. Les pas utilisés pour les calculs par différences finies sont ceux utilisés par la suite pour évaluer les gradients des coefficients aérodynamiques de traînée et de portance. Globalement, les valeurs des gradients calculés par la méthode de l'équation linéarisée sont peu différentes des valeurs calculées par différence finies, on observe cependant un écart allant jusqu'à 15% sur le gradient du champ de torsion au saumon par rapport à α_1 et jusqu'à 50% dans les zones de fortes variation du coefficient de pression. Ces écarts restent locaux et ne se traduisent pas en différences importantes sur les gradients des valeurs intégrées que sont les coefficients de traînée et de portance. Ils sont dus à la non modélisation de la turbulence dans le terme $\dfrac{\partial R_f}{\partial W}$ (hypothèse du μ_t figé) et à l'hypothèse de couche mince.

Convergence de la méthode de l'équation adjointe

Comme pour la méthode de l'équation linéarisée, le processus itératif de résolution du système d'équations (3.31) converge rapidement : il a besoin de cinq itérations entre les solveurs fluide et structure pour converger vers les valeurs finales des vecteurs adjoints (avec $\epsilon_3 = 10^{-8}$ et $\epsilon_4 = 10^{-8}$, en utilisant les notations de l'équation (3.32)). La convergence de ces vecteurs est illustrée par les figures suivantes : les figures 3.65 et 3.64 représentent l'évolution du vecteur adjoint associé au champ de torsion au saumon de l'aile et correspondant respectivement aux coefficients C_d et C_l, les figures 3.67 et 3.66 représentent l'évolution du vecteur adjoint associé au champ de flexion au saumon de l'aile et correspondant respectivement aux coefficients C_d et C_l. La figure 3.68 montre l'évolution de la norme L_2 du membre de droite de la première équation de l'équation (a) (état adjoint associé à la relation de conservation de la masse) du système d'équation (3.31) correspondant à C_l.

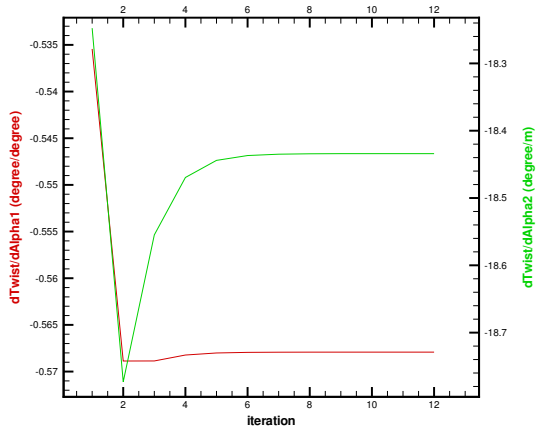

FIG. 3.51 – Convergence du gradient du champ de torsion par rapport à α_1 et α_2, au saumon, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile M6).

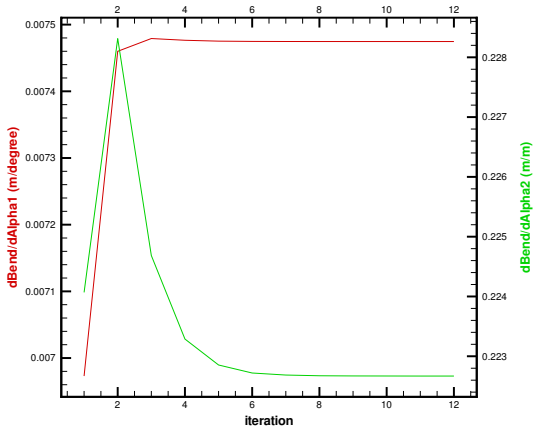

FIG. 3.52 – Convergence du gradient du champ de flexion par rapport à α_1 et α_2, au saumon, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile M6).

Comparaison des valeurs de gradients obtenues par les différentes méthodes

Les gradients des coefficients de traînée et de portance sont assemblés en utilisant les dérivées partielles $\partial W/\partial\alpha_i$ et $\partial D/\partial\alpha_i$, $i \in [1,3]$, suivant la formule (3.6) pour la méthode de l'équation linéarisée ou en utilisant les vecteurs adjoints λ_f et λ_s associés à chaque système d'équations et correspondant à chacune des fonction étudiées (coefficients C_d et C_l) suivant la formule (3.23) pour la méthode de l'équation adjointe. Ces gradients sont comparés aux valeurs calculées par différences finies à l'ordre 2. Les valeurs obtenues sont assemblées dans le tableau 3.10. Les écarts relatifs (calculés par référence aux différence finies pour les méthodes de l'équation linéarisée et adjointe et par référence à la méthode de l'équation linéarisée dans le cas de l'équation adjointe) sont indiqués dans le tableau 3.11.

Les écarts entre les valeurs obtenues sont plus importants que pour la configuration aile-

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$
Méthode linéarisée	$3.313.10^{-3}$	$2.55.10^{-2}$	$5.257.10^{-2}$	0.998
Méthode adjointe	$3.313.10^{-3}$	$2.55.10^{-2}$	$5.257.10^{-2}$	0.998
Différences finies	$3.422.10^{-3}$	$2.39.10^{-2}$	$5.655.10^{-2}$	0.988

TAB. 3.10 – Gradient des coefficients aérodynamiques de traînée et de portance (configuration aile M6).

fuselage F4 pour laquelle on avait fait l'hypothèse de fluide parfait (équations d'Euler) mais restent acceptables pour un calcul effectué sur une configuration 3D avec écoulement visqueux et turbulent (sous les hypothèses de couche mince et de viscosité turbulente constante). Comme pour la configuration aile-fuselage F4, on observe qu'une augmentation du vrillage de la forme rigide ou de la loi de cambrure entraine une augmentation des coefficients de traînée et de portance, ceci étant en particulier dû à une augmentation du chargement de l'aile. Une telle augmentation tend également à augmenter la flexion et à diminuer la torsion de l'aile à l'équilibre aéroélastique, ce comportement est la conséquence du couplage géométrique entre les mouvements de flexion

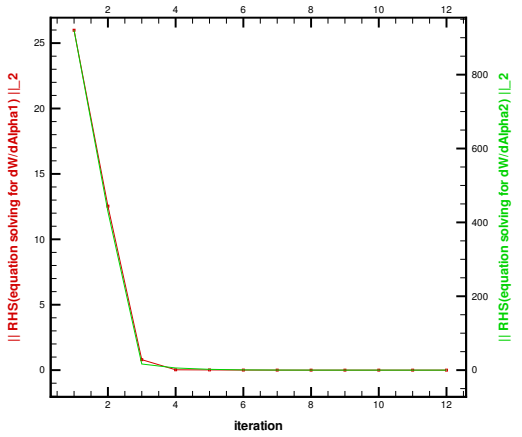

FIG. 3.53 – Convergence de la norme L_2 du résidu de l'équation (a) du système d'équations (3.24) (correspondant à la dérivée de la relation de conservation de la masse) au cours du processus itératif de résolution pour les paramètres de forme α_1 et α_2 (configuration aile M6).

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$
Adjoint/Linéarisé	0%	0%	0%	0%
Linéarisé/DF	3%	1%	7%	6%
Adjoint/DF	3%	1%	7%	6%

TAB. 3.11 – Ecart relatif entre les différentes méthodes du calcul de gradient (configuration aile M6).

et de torsion propres aux ailes à flêche positive.

3.6.3 Aile-Fuselage HiReTT

Les méthodes analytiques pour le calcul des gradients nécessaires à l'optimisation de forme de systèmes couplés fluide-structure ont enfin été testées et validées sur le cas multibloc non coïncidents de l'aile-fuselage HiReTT introduit dans la section 2.6.3. Pour ce cas, le comportement du fluide est simulé par les équations RANS avec pour modèle de turbulence le modèle de Spalart-Allmaras.

Deux paramètres de forme ont été choisis, ceux-ci correspondent à la coordonnée suivant z de deux points de contrôle de la méthode de déformation de maillage utilisée, "free-form deformation" [193, 227]. Le premier point se situe à mi-envergure et le second en bout d'aile.

Comme précédemment, les fonctions considérées sont les coefficients aérodynamiques de traînéee et de portance.

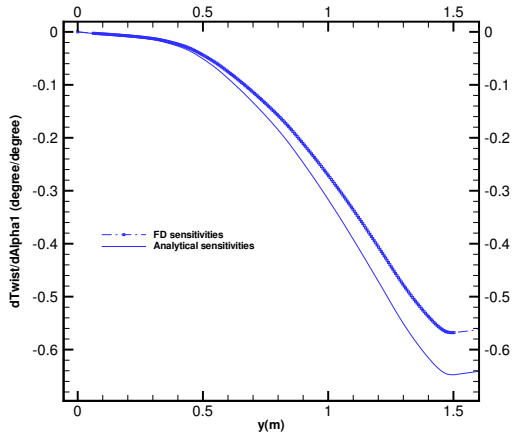

FIG. 3.54 – Gradient du champ de torsion le long de la poutre par rapport à α_1 (configuration aile M6).

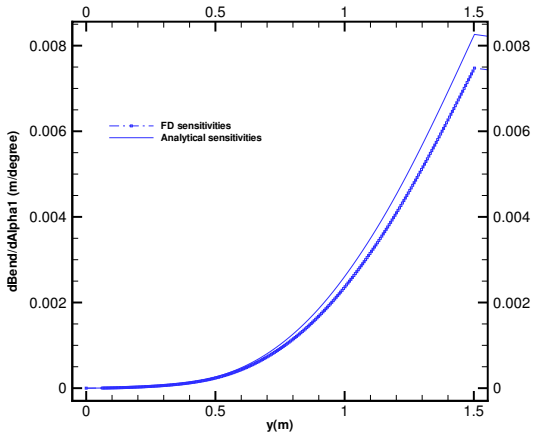

FIG. 3.55 – Gradient du champ de flexion le long de la poutre par rapport à α_1 (configuration aile M6).

Convergence de la méthode de l'équation linéarisée

Le processus itératif de résolution du système d'équation (3.26) converge un peu moins rapidement que pour les configurations précédentes, il a besoin d'au moins huit itérations pour converger (avec $\epsilon_1 = 10^{-3}$ et $\epsilon_2 = 10^{-3}$, en utilisant les notations de l'équation (3.27)). L'évolution des gradients du champ de flexion au saumon par rapport aux paramètres de forme α_1 et α_2 au cours du processus itératif de résolution est illustrée par la figure 3.70 ; celle des gradients du champ de torsion au saumon par rapport aux paramètres de forme α_1 et α_2 par la figure 3.69. Enfin la convergence des dérivées $\partial W/\partial\alpha_1$ et $\partial W/\partial\alpha_2$ est illustrée par le tracé de la norme L_2 du membre de droite de la première équation de l'équation (a) du système d'équations (3.26) (équation correspondant à la dérivée de la relation de conservation de masse par rapport à α) au cours du processus de résolution du système (3.26), figure 3.71.

Les valeurs des gradients intermédiaires $\partial W/\partial\alpha_i$ et $\partial D/\partial\alpha_i$, $i = 1, 2$, sont comparées aux évaluations par différences finies. Les figures 3.73 et 3.75 comparent les gradients du champ de flexion au saumon de l'aile par rapport à α_1 et α_2 calculés par la méthode de l'équation linéarisée d'une part et par différences finies d'autre part. De même, les figures 3.72 et 3.74 comparent les gradients du champ de torsion au saumon de l'aile par rapport à α_1 et α_2 calculés par la méthode de l'équation linéarisée d'une part et par différences finies d'autre part. La comparaison des champs $\partial W/\partial\alpha_1$, $i = 1, 2$, aux valeurs calculées par différences finies se fait par l'intermédiaire de la comparaison du gradient du coefficient de pression à la paroi de l'aile. Les gradients $\partial C_p/\partial\alpha_i$, $i = 1, 2$ calculés en utilisant les gradients donnés par la méthode de l'équation linéarisée et ceux évalués par différences finies sont comparés pour trois profils situés respectivement à l'emplenture ($y = 0.2m$), à mi-envergure ($y = 0.45m$) et au saumon de l'aile ($y = 0.7m$). Ces comparaisons sont représentées par les figures 3.76, 3.78 et 3.80 pour le paramètre de forme α_1 et par les figures 3.77, 3.79 et 3.81 pour le paramètre de forme α_2. Les pas utilisés pour les calculs par différences finies sont ceux utilisés par la suite pour évaluer les gradients des coefficients aérodynamiques de traînée et de portance. Globalement, les valeurs des gradients calculées par la méthode de l'équation linéarisée suivent les mêmes tendances que les valeurs calculées par différence finies. Elles peuvent cependant être localement très différentes, en particulier en bout d'aile. Ces écarts, dus à la non modélisation de la turbulence dans le terme $\dfrac{\partial R_f}{\partial W}$ (hypothèse du μ_t figé) et à l'hypothèse de couche mince, ne se traduisent cependant pas en différences excessives

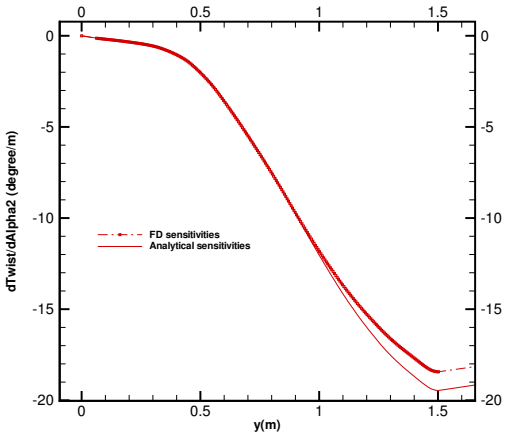

FIG. 3.56 – Gradient du champ de torsion le long de la poutre par rapport à α_2 (configuration aile M6).

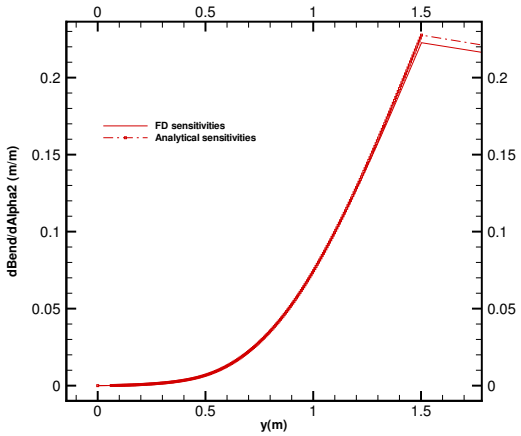

FIG. 3.57 – Gradient du champ de flexion le long de la poutre par rapport à α_2 (configuration aile M6).

sur les gradients des valeurs intégrées que sont les coefficients de traînée et de portance.

Convergence de la méthode de l'équation adjointe

Le processus itératif de résolution du système d'équation (3.31) converge également moins rapidement que pour les configurations précédentes, il a besoin d'au moins huit itérations pour converger (avec $\epsilon_1 = 10^{-3}$ et $\epsilon_2 = 10^{-4}$, en utilisant les notations de l'équation (3.32)). La convergence de ces vecteurs est illustrée par les figures suivantes : les figures 3.83 et 3.82 représentent l'évolution du vecteur adjoint associé au champ de torsion au saumon de l'aile et correspondant respectivement aux coefficients C_d et C_l, les figures 3.85 et 3.84 représentent l'évolution du vecteur adjoint associé au champ de flexion au saumon de l'aile et correspondant respectivement aux coefficients C_d et C_l. La figure 3.86 montre l'évolution de la norme L_2 du membre de droite de la première équation de l'équation (a) (état adjoint associé à la relation de conservation de la masse) du système d'équation (3.31) correspondant à C_l.

Comparaison des valeurs de gradients obtenues par les différentes méthodes

Les gradients des coefficients de traînée et de portance sont assemblés en utilisant les dérivées partielles $\partial W/\partial\alpha_i$ et $\partial D/\partial\alpha_i$, $i \in [1,3]$, suivant la formule (3.6) pour la méthode de l'équation linéarisée ou en utilisant les vecteurs adjoints λ_f et λ_s associés à chaque système d'équations et correspondant à chacune des fonction étudiées (coefficients C_d et C_l) suivant la formule (3.23) pour la méthode de l'équation adjointe. Ces gradients sont comparés aux valeurs calculées par différences finies à l'ordre 2. Les valeurs obtenues sont assemblées dans le tableau 3.12. Les écarts relatifs (calculés par référence aux différence finies pour les méthodes de l'équation linéarisée et adjointe et par référence à la méthode de l'équation linéarisée dans le cas de l'équation adjointe) sont indiqués dans le tableau 3.13.

Les écarts entre les valeurs obtenues sont comparables à ceux obtenus avec la configuration aile M6 et restent également acceptables pour un calcul effectué sur une configuration visqueuse et turbulente (sous les hypothèses de couche mince et de viscosité turbulente constante). Une augmentation des coordonnées suivant z des deux points de contrôle considérés (un à mi-envergure

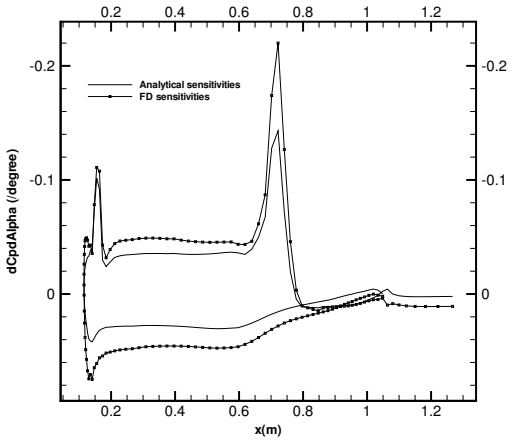

FIG. 3.58 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.2m$ par rapport à α_1 (configuration aile M6).

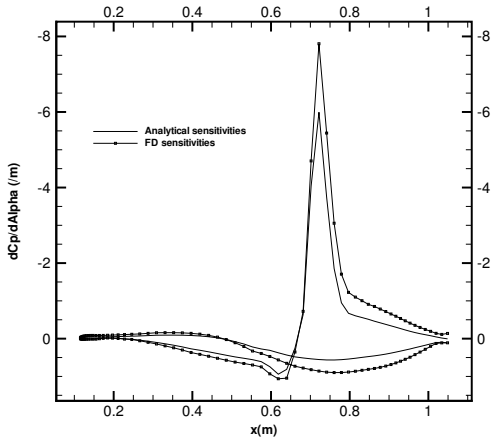

FIG. 3.59 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.2m$ par rapport à α_2 (configuration aile M6).

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$
Méthode linéarisée	$-5.69.10^{-2}$	-1.146	-1.267	-20.171
Méthode adjointe	$-5.71.10^{-2}$	-1.146	-1.267	-20.128
Différences finies	$-6.87.10^{-2}$	-0.991	-1.327	-20.750

TAB. 3.12 – Gradient des coefficients aérodynamiques de traînée et de portance (configuration aile-fuselage HiReTT).

	$dC_d/d\alpha_1$	$dC_d/d\alpha_2$	$dC_l/d\alpha_1$	$dC_l/d\alpha_2$
Adjoint/Linéarisé	0.3%	0%	0%	0.2%
Linéarisé/DF	17%	15%	4.5%	3%
Adjoint/DF	17%	15%	4.5%	3%

TAB. 3.13 – Ecart relatif entre les différentes méthodes du calcul de gradient (configuration aile-fuselage HiReTT).

et l'autre en bout d'aile) entrainent une diminution des coefficients de portance et de traînée de l'aile (conséquence en particulier d'une réduction de la traînée induite). Elle induit également une diminution de la flexion et une augmentation de la torsion de l'aile à l'équilibre aéroélastique, ce comportement est la conséquence du couplage géométrique entre les mouvements de flexion et de torsion propre aux ailes à flêche positive.

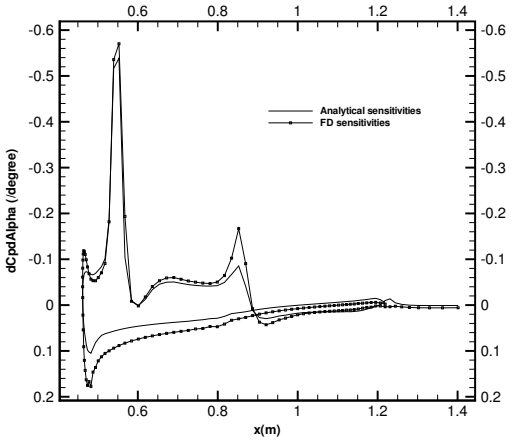

FIG. 3.60 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.8m$ par rapport à α_1 (configuration aile M6).

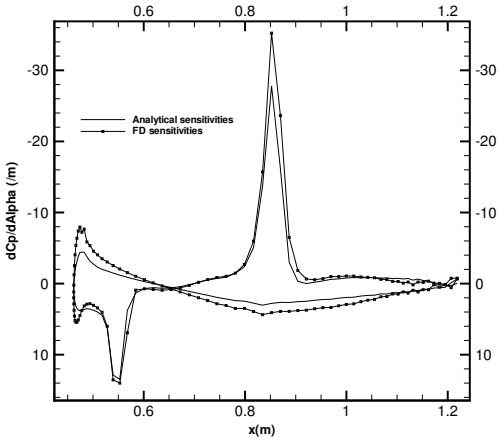

FIG. 3.61 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.8m$ par rapport à α_2 (configuration aile M6).

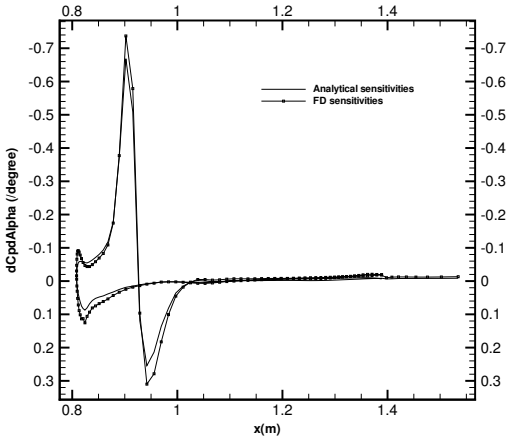

FIG. 3.62 – Gradient du coefficient de pression C_p sur le profil situé en $y = 1.4m$ par rapport à α_1 (configuration aile M6).

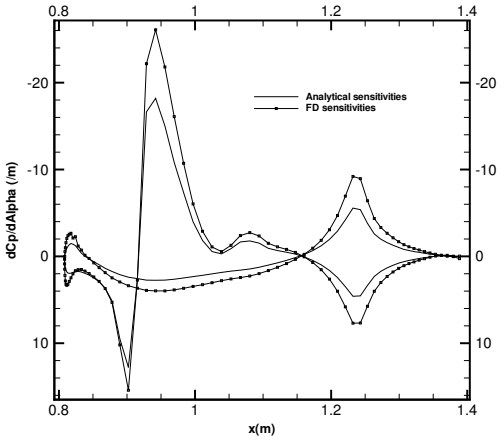

FIG. 3.63 – Gradient du coefficient de pression C_p sur le profil situé en $y = 1.4m$ par rapport à α_2 (configuration aile M6).

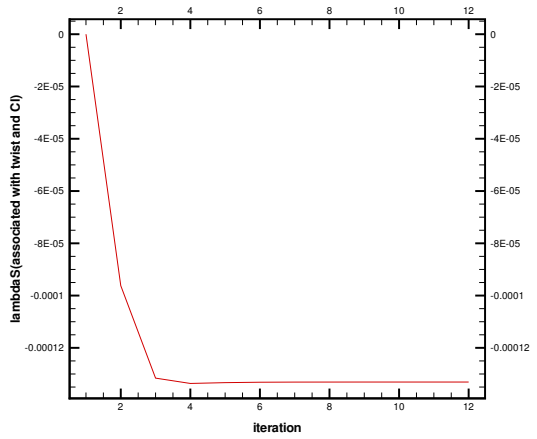

FIG. 3.64 – Convergence du vecteur adjoint associé au champ de torsion au saumon et correspondant à la fonction C_l au cours du processus itératif de résolution de l'équation (3.31) (configuration aile M6).

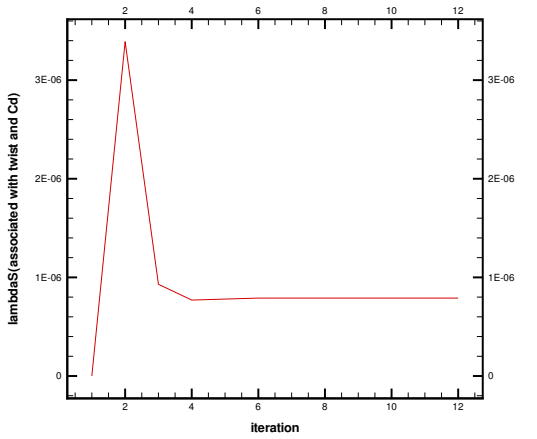

FIG. 3.65 – Convergence du vecteur adjoint associé au champ de torsion au saumon et correspondant à la fonction C_d au cours du processus itératif de résolution de l'équation (3.31) (configuration aile M6).

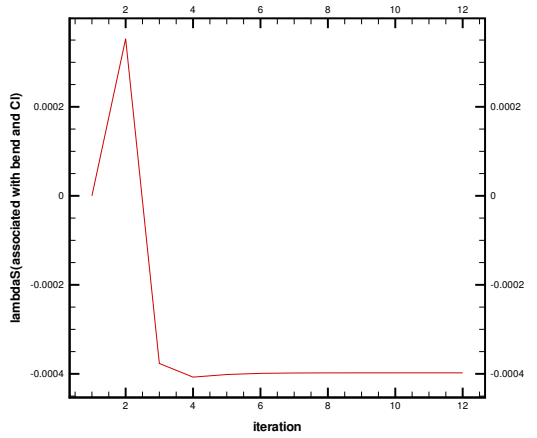

FIG. 3.66 – Convergence du vecteur adjoint associé au champ de flexion au saumon et correspondant à la fonction C_l au cours du processus itératif de résolution de l'équation (3.31) (configuration aile M6).

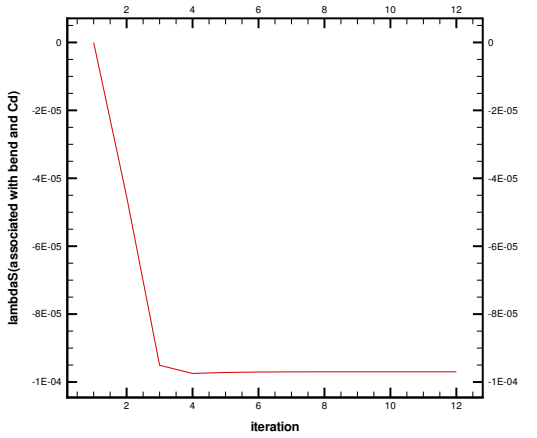

FIG. 3.67 – Convergence du vecteur adjoint associé au champ de flexion au saumon et correspondant à la fonction C_d au cours du processus itératif de résolution de l'équation (3.31) (configuration aile M6).

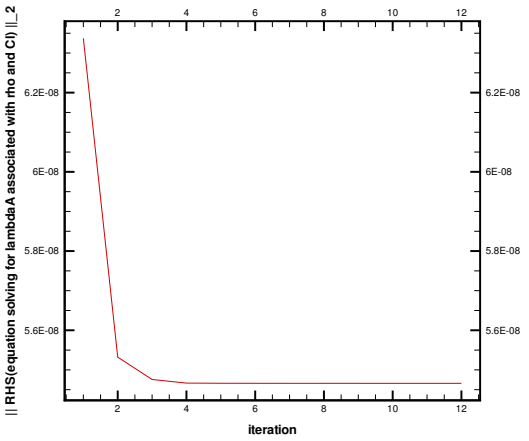

FIG. 3.68 – Convergence de la norme L_2 du résidu de l'équation (a) du système d'équations (3.31) associé à la valeur conservative ρ et correspondant à la fonction C_l au cours du processus itératif de résolution (configuration aile M6).

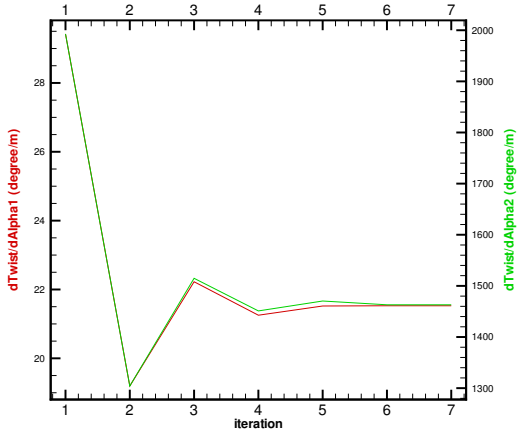

FIG. 3.69 – Convergence du gradient du champ de torsion par rapport à α_1 et α_2, au saumon, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile-fuselage HiReTT).

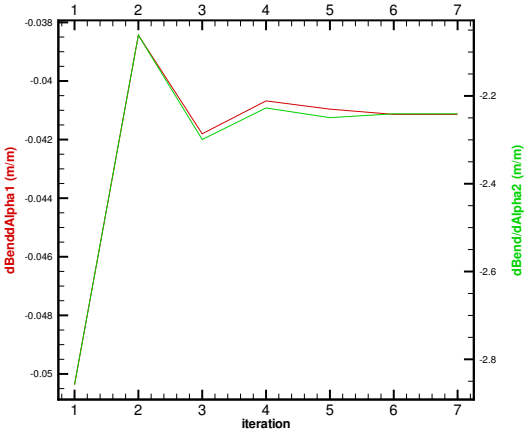

FIG. 3.70 – Convergence du gradient du champ de flexion par rapport à α_1 et α_2, au saumon, au cours du processus itératif de résolution de l'équation (3.24) (configuration aile-fuselage HiReTT).

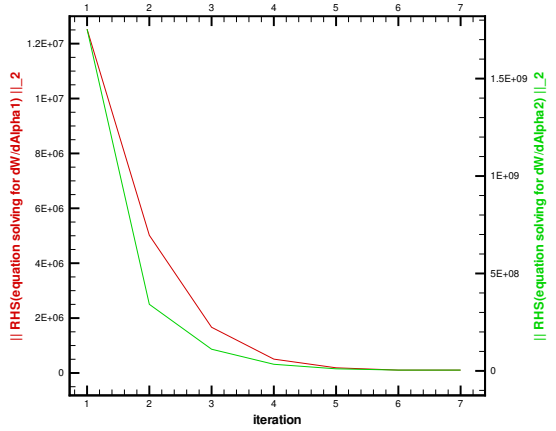

FIG. 3.71 – Convergence de la norme L_2 du résidu de l'équation (a) du système d'équations (3.24) (correspondant à la dérivée de la relation de conservation de la masse) au cours du processus itératif de résolution pour les paramètres de forme α_1 et α_2 (configuration aile-fuselage HiReTT).

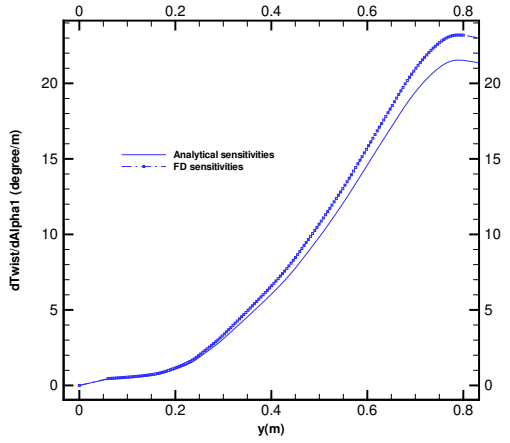

FIG. 3.72 – Gradient du champ de torsion le long de la poutre par rapport à α_1 (configuration aile-fuselage HiReTT).

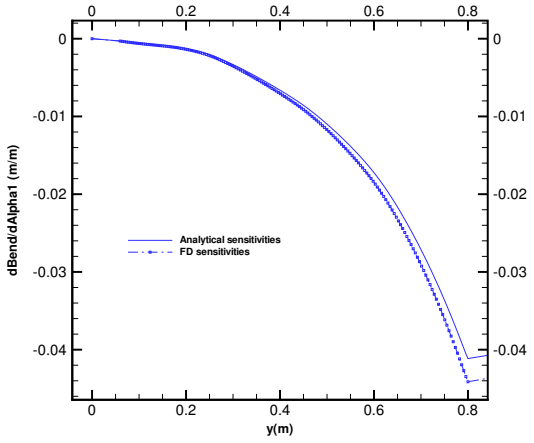

FIG. 3.73 – Gradient du champ de flexion le long de la poutre par rapport à α_1 (configuration aile-fuselage HiReTT).

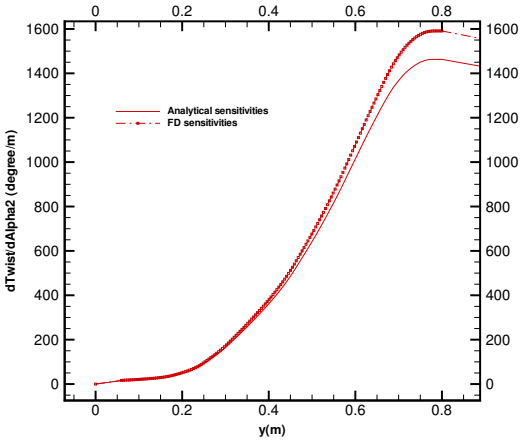

FIG. 3.74 – Gradient du champ de torsion le long de la poutre par rapport à α_2 (configuration aile-fuselage HiReTT).

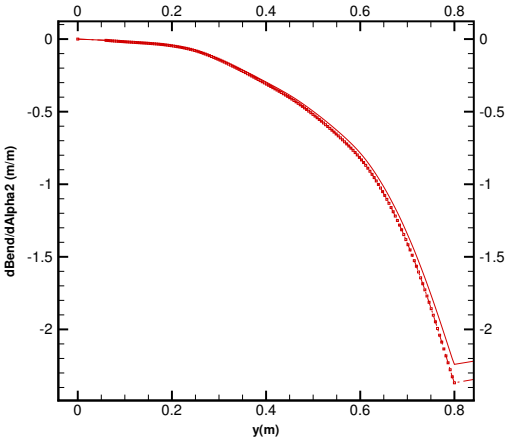

FIG. 3.75 – Gradient du champ de flexion le long de la poutre par rapport à α_2 (configuration aile-fuselage HiReTT).

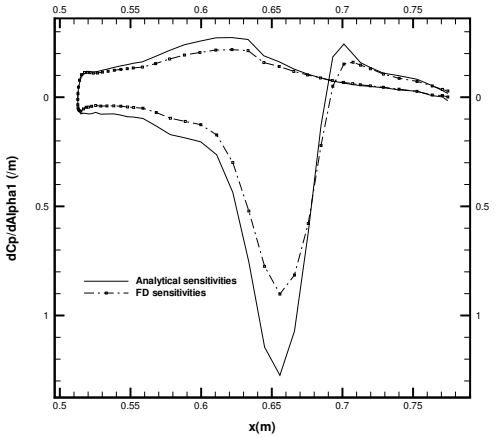

FIG. 3.76 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.2m$ par rapport à α_1 (configuration aile-fuselage HiReTT).

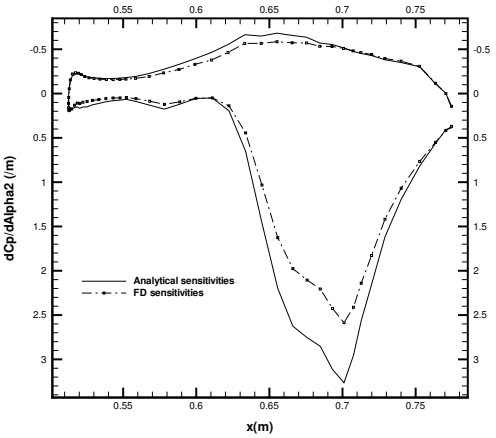

FIG. 3.77 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.2m$ par rapport à α_2 (configuration aile-fuselage HiReTT).

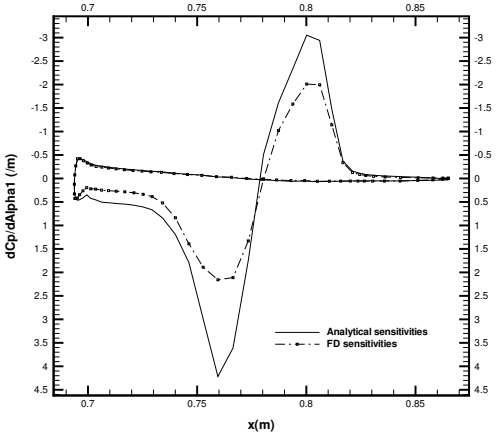

FIG. 3.78 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.45m$ par rapport à α_1 (configuration aile-fuselage Hi-ReTT).

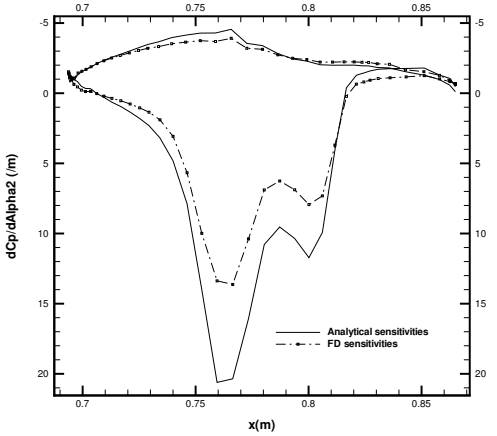

FIG. 3.79 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.45m$ par rapport à α_2 (configuration aile-fuselage Hi-ReTT).

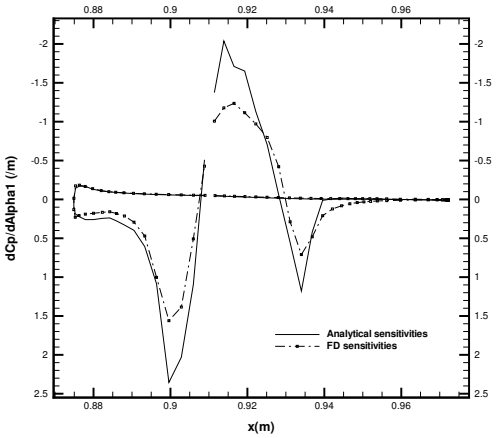

FIG. 3.80 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.7m$ par rapport à α_1 (configuration aile-fuselage Hi-ReTT).

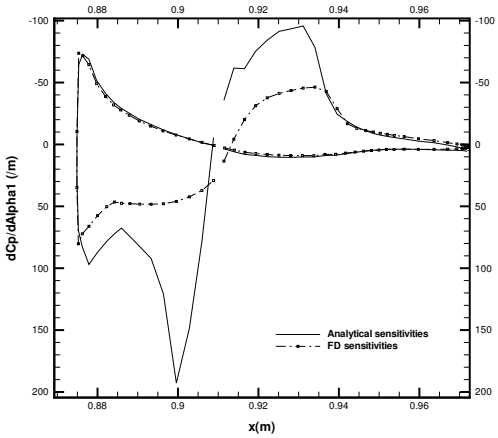

FIG. 3.81 – Gradient du coefficient de pression C_p sur le profil situé en $y = 0.7m$ par rapport à α_2 (configuration aile-fuselage Hi-ReTT).

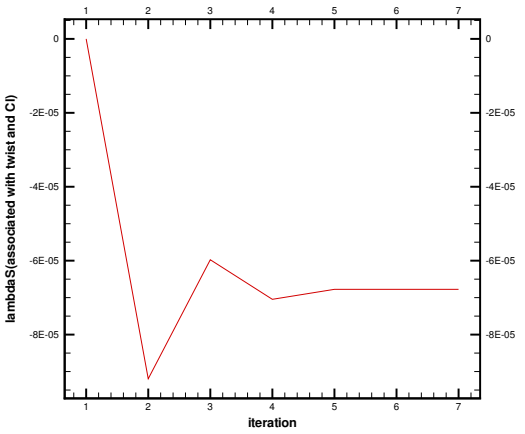

FIG. 3.82 – Convergence du vecteur adjoint associé au champ de torsion au saumon et correspondant à la fonction C_l au cours du processus itératif de résolution de l'équation (3.31) (configuration HiReTT).

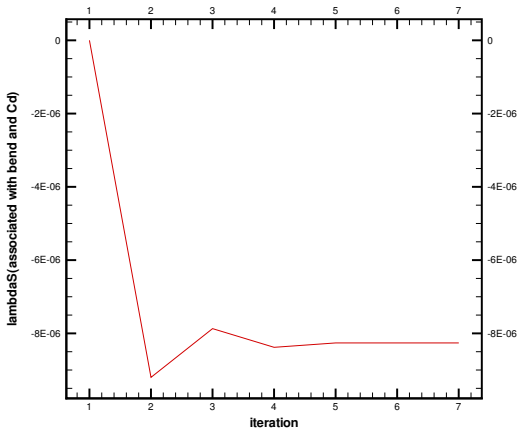

FIG. 3.83 – Convergence du vecteur adjoint associé au champ de torsion au saumon et correspondant à la fonction C_d au cours du processus itératif de résolution de l'équation (3.31) (configuration HiReTT).

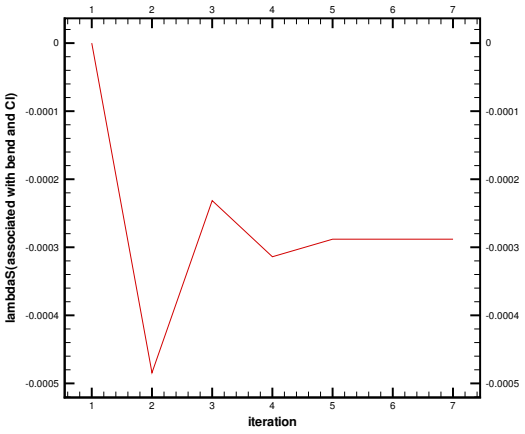

FIG. 3.84 – Convergence du vecteur adjoint associé au champ de flexion au saumon et correspondant à la fonction C_l au cours du processus itératif de résolution de l'équation (3.31) (configuration HiReTT).

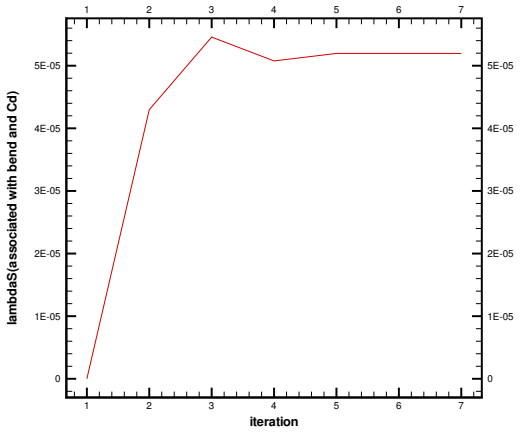

FIG. 3.85 – Convergence du vecteur adjoint associé au champ de flexion au saumon et correspondant à la fonction C_d au cours du processus itératif de résolution de l'équation (3.31) (configuration HiReTT).

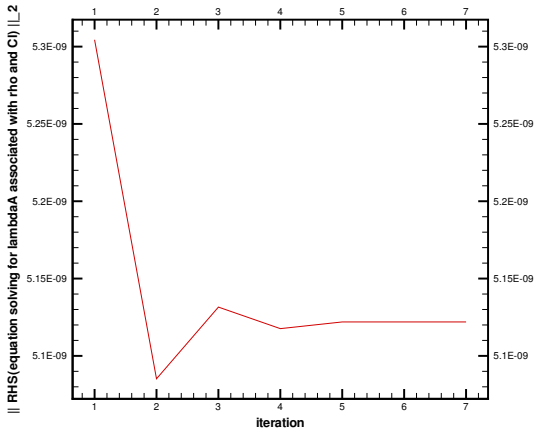

FIG. 3.86 – Convergence de la norme L_2 du résidu de l'équation (a) du système d'équations (3.31) associé à la valeur conservative ρ et correspondant à la fonction C_l au cours du processus itératif de résolution (configuration HiReTT).

Chapitre 4

Modèles approchés pour l'optimisation de forme

4.1 Contexte historique

La principale difficulté associée à l'utilisation des méthodes d'optimisation globale pour l'optimisation de forme (voir chapitre 1) réside dans le coût des évaluations nécessaires i.e. dans le coût de calcul de la fonction objectif J et des fonctions contraintes G_k, $1 \leq k \leq p$. Celui-ci est d'autant plus important que la modélisation physique adoptée est précise (par exemple, équations RANS pour simuler le comportement du fluide et modèle élément fini pour simuler le comportement de la structure). Afin de rendre acceptables les coûts de calcul, de nombreux auteurs ont, depuis le milieu des années 90, mis en avant l'intérêt de remplacer une partie des évaluations du modèle complet par des évaluations approchées, bien définies, issues d'un modèle réduit non physique. Ce dernier est qualifié de métamodèle, et la définition d'un modèle approché d'une certaine fonction est aussi appelée régression. Les évaluations associées à la modélisation la plus précise, susceptible d'être considérée pour un problème donné, sont qualifiées d'exactes.

Depuis une vingtaine d'années, un grand nombre de publications traitant de la mise en oeuvre et des performances de différents types de métamodèles ont vu le jour. En 1994, un état de l'art relatif aux méthodes de régression par fonctions polynômiales, par fonctions splines, par fonctions à base radiale, par lissage à noyau ou par Kriging (aussi appelé corrélation spatiale) a été publié par Barton [18]. En 2000, Jin *et al.* [117] ont publié à leur tour une description de ces modèles accompagnée d'une comparaison de leur efficacité relative sur des fonctions mathématiques. Un an plus tard, Simpson *et al.* [203] ont publié un résumé complet relatif à l'utilisation de modèles approchés classiques, parmi lesquels les surfaces de réponse par fonctions polynômiales ou par fonctions splines, les réseaux de neurones et le Kriging, et renvoyant à d'autres articles pour la description de modèles approchés moins répandus. Un état de l'art spécialement dédié à la méthode du Kriging a, de plus, été publié en 2004 par Van Beers et Kleijen [214]. Enfin, Queipo *et al.* [174] ont récemment publié un article traitant des performances des modèles de régression par surfaces de réponse polynômiales, par Kriging et par réseaux de neurones pour différents plans d'expérience initiaux.

Par ailleurs, un certain nombre de publications récentes traitent plus particulièrement de l'emploi de modèles approchés dans le cadre de l'optimisation de forme. Ces derniers sont destinés à fournir des évaluations approchées, certes, mais peu coûteuses, des fonctions objectif et contraintes en vue d'être utilisées par les algorithmes d'optimisation globale mis en oeuvre (en général algorithme génétique). Demeulaere et Pierret [52] ont, par exemple, publié un exemple d'application mettant en oeuvre un réseau de neurones de type "perceptron multicouche" au sein d'un algorithme génétique utilisé pour l'optimisation d'une aube tridimensionnelle de compres-

seur par rapport à neuf paramètres de forme. Giannakoglou *et al.* ont également apporté une large contribution à ce sujet : en évaluant les performances des réseaux de neurones de type "perceptron multicouche" et des réseaux de neurones faisant appel aux fonctions à base radiales [79], en étudiant les moyens et les avantages de la prise en compte des dérivées de la fonction à approcher lors de la construction de modèles approchés par réseaux de neurones et en appliquant la méthodologie développée à l'optimisation d'une aube tridimensionnelle de turbine autour de laquelle le comportement du fluide était gouverné par les équations d'Euler [80]. Polini *et al.* [172] ont par ailleurs effectué l'optimisation globale de la forme d'une aile autour de laquelle le comportement du fluide était gouverné par les équations d'Euler ou de Navier-Stokes avec l'hypothèse de couche mince ; ils faisaient appel à un algorithme génétique et à une régression par Kriging. Giunta *et al.* [84] ont également étudié l'utilisation de surfaces de réponses construites à partir de fonctions polynômiales ou de fonctions rationnelles pour l'optimisation de forme globale d'un avion de transport supersonique par rapport à deux paramètres de forme. En 1999, Papila *et al.* [161] ont utilisé des régressions par réseaux de neurones et par surfaces de réponse polynômiales pour l'optimisation de forme d'un profil et d'une aile par rapport à deux paramètres de forme. Le comportement du fluide autour de ces formes aérodynamiques était gouverné par les équations du potentiel. Rai *et al.* [175] ont réalisé l'optimisation de forme globale d'un profil bidimensionnel d'aube de turbine par rapport à quinze paramètres de forme en combinant régression par réseaux de neurones et régression par surfaces de réponse. Simpson *et al.* [202] ont également procédé à l'optimisation d'une tuyère par rapport à deux paramètres de forme, en prenant en compte des critères aérodynamiques et structurels et en faisant appel à des régressions par surfaces de réponse polynômiales et à des régressions par Kriging. Le comportement structurel de la tuyère était décrit par un modèle élément fini et le comportement du fluide, gouverné par les équations d'Euler. Plus récemment, Jouhaud *et al.* [118] ont décrit l'optimisation globale d'un profil par rapport à deux paramètres de forme faisant appel à une régression par Kriging, le comportement du fluide autour du profil étant gouverné par les équations RANS.

Notre objectif est d'évaluer les performances de quatre modèles approchés parmi les plus répandus sur un cas d'optimisation aérodynamique pratique. Plus précisément nous chercherons dans cette partie à déterminer le modèle approché le mieux adapté à l'optimisation de forme en turbomachine.

4.2 Description du problème d'optimisation

4.2.1 Géométrie du cas test

Nous nous sommes intéressés à l'optimisation de forme d'une aube de stator de la configuration VEGA2 [56] qui est une configuration stator-rotor de turbine classique. L'optimisation globale d'une telle configuration est très coûteuse. Nous nous sommes donc restreint à l'optimisation de la projection plane d'une coupe azimutale au moyeu (géométrie bidimensionnelle).

Le maillage du domaine fluide entourant le profil bidimensionnel est composé de quatre blocs avec raccords coïncidents. Ce dernier est représenté sur la figure 4.1. La longueur caractéristique de la aube est $L = 90\ mm$.

Le maillage est constitué de 11816 noeuds au total et la condition de périodicité est appliquée aux frontières inférieure et supérieure.

Le flux qui entre dans ce domaine est subsonique et dirigé suivant l'axe x. Sa température d'arrêt est $T_i = 288.27\ K$ et sa pression d'arrêt vaut $p_i = 101325\ Pa$. La pression statique en sortie est imposée : le rapport entre la pression statique en sortie et la pression d'arrêt en entrée est égal à $p_{s,\text{sortie}}/p_i = 0.35$. La viscosité turbulente est calculée par la loi de Sutherland. Le nombre de Reynolds de l'écoulement entrant dans le domaine est calculé en prenant comme

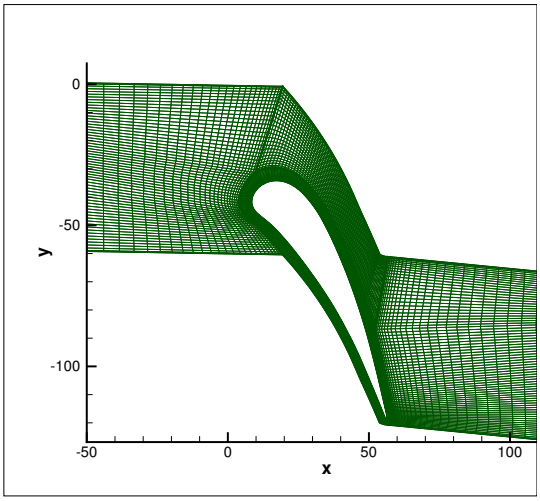

FIG. 4.1 – Maillage du domaine fluide autour de la aube (configuration VEGA2-2D).

longueur de référence L et vaut $Re = \dfrac{\rho_i a_i L}{\mu(T_i)} = \dfrac{\rho_i L}{\mu(T_i)}\sqrt{\dfrac{p_i}{\rho_i}} = 2.09\ 10^5$.

4.2.2 Calcul de l'écoulement fluide

Le comportement du fluide est gourverné par les équations RANS avec comme modèle de turbulence le modèle $k-\omega$ de Smith [204]. Le système nonlinéaire à sept équations associé est résolu numériquement par le code de simulation pour la mécanique des fluides elsA, présenté dans le chapitre 2. Les termes convectifs associés à l'écoulement moyenné sont calculés par les formules de Roe du second ordre (avec l'approche MUSCL et le limiteur de Van Albada). Le terme convectif associé à l'écoulement turbulent est calculé par la formule du flux de Roe du premier ordre, et les termes diffusifs associés aux deux écoulements sont calculés par une formule de flux centrée avec une évaluation des gradients aux centres des interfaces. De plus, les termes source des équations turbulentes sont calculés par des formules centrées. Ces calculs sont détaillés dans l'article [179]. Ce dernier montre également la cohérence entre les résultats numériques et expérimentaux sur la configuration tridimensionnelle originale.

A cause de la faible valeur de la pression statique en sortie, l'écoulement devient sonique entre les deux aubes, à proximité du bord de fuite, à l'endroit où la distance entre ces dernières est la plus faible. Deux ondes de choc prennent alors naissance au niveau du bord d'attaque de l'aube : la première est dirigée selon l'axe x et la seconde est oblique. Les lignes iso-Mach associées à l'écoulement sont représentées par la figure 4.2.

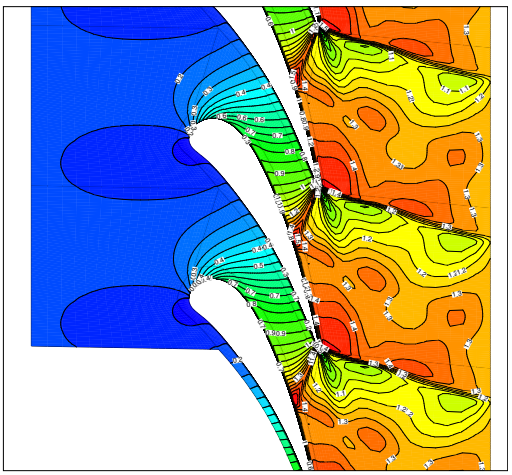

FIG. 4.2 – Lignes iso-Mach autour de l'aube (configuration VEGA2-2D).

4.2.3 Problème d'optimisation

On s'intéresse dans le cadre de ce problème à la modification de la forme de la coupe bidimensionnelle de l'aube lorsque le bord de fuite subit un déplacement suivant les axes x et y, le bord d'attaque restant à une position fixe. La déformation se fait de manière continue, de telle sorte que la déformée de la projection bidimensionnelle de l'aube constitue une fonction algébrique continue et régulière en coordonnées curvilignes. Une forme particulière est associée à chaque position du bord de fuite, c'est-à-dire à chacune des valeurs de ses coordonnées (x, y).

La déformation de la forme de l'aube induit une déformation du domaine fluide entourant directement l'aube (voir figure 4.1). Cette déformation est amortie de sorte qu'elle soit nulle à l'extérieur de ce domaine, c'est-à-dire dans les autres domaines.

Le déplacement maximal du bord de fuite autorisé selon les axes x ou y est de $+/-0.4\ mm$. Le domaine de conception est donc défini de la manière suivante :

$$\mathcal{D} = \{(x, y)_{\text{bord de fuite}} \in [-0.4; +0.4] \times [-0.4; +0.4]\} \ .$$

Le déplacement du bord de fuite selon l'axe x est défini comme étant le premier paramètre de forme α_1, et le déplacement du bord de fuite selon l'axe y comme le deuxième paramètre de forme α_2, d'où la définition du domaine de conception :

$$\mathcal{D} \quad = \quad \{(\alpha_1, \alpha_2) \in [-0.4; +0.4] \times [-0.4; +0.4]\} \quad .$$

La performance de chacune des déformées de l'aube est mesurée par la valeur de la pression totale en sortie. L'objectif du problème d'optimisation est de minimiser la perte de pression totale, c'est-à-dire le rapport $1 - p_{\text{totale,sortie}}/p_{\text{totale,entrée}}$, lors du passage dans le canal délimité par deux aubes. Il s'agit d'un problème d'optimisation de forme classique pour les turbomachines. Le rapport $p_{\text{totale,sortie}}/p_{\text{totale,entrée}}$ varie entre 0.918 et 0.924 sur le domaine de conception $\mathcal{D}$. L'écart

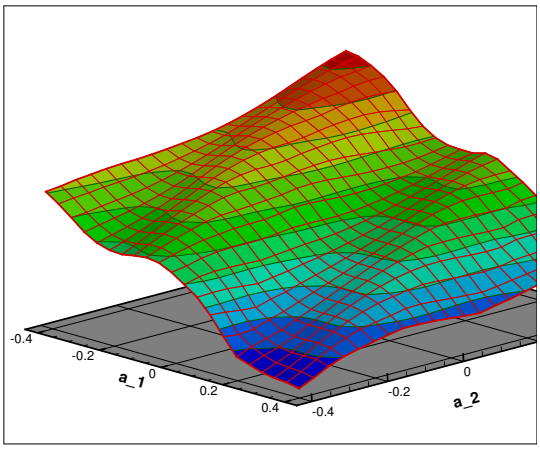

FIG. 4.3 – Pression totale moyenne en sortie en fonction des paramètres de forme α_1 et α_2 sur $\mathcal{D}$ (configuration VEGA2-2D).

est jugé significatif et une optimisation peut être effectuée. Les faibles valeurs de ce rapport s'expliquent par la présence de chocs forts. Une variation de la forme de l'aube due à une variation de la position du bord de fuite induit une variation de la pression totale en sortie correspondant à une variation de l'intensité des chocs.

Un échantillonnage régulier du domaine de conception $\mathcal{D}$ en 21×21 points est réalisé. L'écoulement correspondant à chacun de ces points est alors calculé avec exactement les mêmes paramètres numériques. Pour chacun de ces calculs, l'ordre de grandeur du résidu explicite associé au schéma est réduit d'un facteur quatre ou cinq. Ces calculs étant effectués, il est possible de tracer la pression totale moyenne en sortie en fonction des paramètres de forme α_1 et α_2 sur le domaine de conception (voir figure 4.3).

4.3 Modèles approchés utilisés

Nous nous sommes intéressés aux régressions par surfaces de réponse polynômiales du second degré, par réseaux de fonctions à base radiale, par perceptrons multicouches et par Kriging. La méthode de régression par surfaces polynômiales est décrite en détails dans le chapitre 1. Dans cette section, nous détaillons en revanche les trois autres méthodes d'approximation.

Afin de rester cohérent avec la présentation des modèles approchés donnée au chapitre 1, nous notons $J(\alpha)$ la fonction exacte que l'on cherche à approcher, c'est-à-dire le rapport entre les pressions totales en sortie et en entrée, et $\tilde{J}(\alpha)$ la fonction approchée. Le vecteur des paramètres de forme α a, dans notre cas, deux composantes α_1 et α_2 définies dans le paragraphe 4.2.3. L'erreur quadratique MSE - "*mean square error*" en anglais - entre le modèle approché et la fonction

exacte sur l'échantillon de taille n_s que l'on considère est égale à :

$$\text{MSE} = \frac{1}{2}\sum_{i=1}^{n_s}\Big(J(\alpha_1^i,\alpha_2^i) - \tilde{J}(\alpha_1^i,\alpha_2^i)\Big)^2$$

où $\alpha_i = (\alpha_1^i, \alpha_2^i)$ représente l'échantillon i, $1 \leq i \leq n_s$. Le vecteur des évaluations exactes est alors égal à

$$J_s = \begin{bmatrix} J(\alpha_1) \\ \vdots \\ J(\alpha_{n_s}) \end{bmatrix}.$$

4.3.1 Régression par surface de réponse polynômiale du degré deux

Dans le contexte de notre cas test, nous avons effectué des approximations de la fonction exacte par des surfaces de réponse polynômales de degré 2, 4, 6 et 8. Nous ne présentons cependant, pour des raisons de simplicité, que les formules associées à la régression polynômiale de degré 2. La fonction exacte J est, dans ce cas, approchée par une fonction quadratique de la forme

$$\tilde{J}(\alpha) = \psi_0 + \psi_1\alpha_1 + \psi_2\alpha_2 + \psi_{11}\alpha_1^2 + \psi_{12}\alpha_1\alpha_2 + \psi_{22}\alpha_2^2$$

Comme nous l'avons décrit au chapitre 1, les coefficients de ce polynôme sont calculés en minimisant l'erreur MSE sur l'échantillon considéré, de telle sorte que

$$\Psi = (X^T X)^{-1} X^T J_s$$

où

$$\Psi = \begin{bmatrix} \psi_0 \\ \psi_1 \\ \psi_2 \\ \psi_{11} \\ \psi_{12} \\ \psi_{22} \end{bmatrix} \quad et \quad X = \begin{bmatrix} 1 & \alpha_1^1 & \alpha_2^1 & (\alpha_1^1)^2 & \alpha_1^1\alpha_2^1 & (\alpha_2^1)^2 \\ 1 & \alpha_1^2 & \alpha_2^2 & (\alpha_1^2)^2 & \alpha_1^2\alpha_2^2 & (\alpha_2^2)^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha_1^{n_s} & \alpha_2^{n_s} & (\alpha_1^{n_s})^2 & \alpha_1^{n_s}\alpha_2^{n_s} & (\alpha_2^{n_s})^2 \end{bmatrix}.$$

4.3.2 Régression par réseaux de neurones

Le principe général de la régression de fonctions par réseaux de neurones de type réseaux de fonctions à base radiale (RBF) ou perceptrons multi-couches (MP) a déjà été exposé au chapitre 1. Complétons le, tout d'abord, par quelques résultats théoriques.

Hornik [108, 109, 107] a démontré que "toute fonction bornée suffisamment régulière pouvait être approchée uniformément et avec une précision arbitraire dans un domaine fini de l'espace des variables par un réseau de neurones comportant une couche de neurones cachés, en nombre fini, possédant tous la même fonction d'activation et un neurone de sortie linéaire". Ce théorème assure la réalisabilité de la régression d'une fonction nonlinéaire par réseaux de neurones.

Comme on l'a vu au chapitre 1, la recherche de la fonction approchée est réalisée non pas par interpolation directe mais par minimisation de l'erreur au sens des moindres carrés par rapport aux paramètres définissant le réseau. Ce procédé est d'autant plus justifié que la fonction

à approcher est de nature statistique, les évaluations pouvant alors être entachées d'erreur (par exemple, des erreurs de mesures).

Après avoir réalisé un nombre suffisamment grand d'évaluations de la fonction $\{J(\alpha_i),\ i \in [1, n_s]\}$, la mise en oeuvre de la régression se fait par l'enchainement des étapes suivantes :

1. trouver le nombre de neurones cachés nécessaire à une approximation suffisante ;
2. déterminer l'ensemble Ψ des coefficients définissant le réseau de neurones et, si ce dernier est un réseau (RBF), les centres et les rayons des fonctions gaussiennes ;
3. évaluer les performances du réseau à l'issue de l'apprentissage.

Baron [14] a démontré plusieurs propriétés sur le nombre et le type de neurones cachés à utiliser. Pour atteindre une précision donnée aux points de l'échantillon, le nombre d'évaluations n_s étant fixé, le nombre de paramètres ajustables nécessaires est inférieur lorsque l'approximation dépend des paramètres ajustables de manière non linéaire, comparativement à un réseau avec dépendance linéaire en fonction des paramètres ajustables. Il s'agit de la propriété de parcimonie. Pour une précision donnée et un nombre d'échantillons n_s fixés, le nombre de paramètres ajustables croit exponentiellement avec la taille n de l'espace lorsque l'approximation est linéaire par rapport à ces paramètres, alors qu'il croit linéairement avec celle-ci lorsque l'approximation est non linéaire par rapport à ces paramètres. L'utilisation de neurones de type (MP) est donc plus parcimonieuse que celle de type (RBF) lorsque les centres et les rayons des fonctions gaussiennes sont choisis a priori.

Par ailleurs, le même auteur a montré qu'à nombre d'échantillons fixé, l'écart entre l'approximation par réseau de neurones et la fonction à approcher est une fonction décroissante du nombre de neurones cachés. Il n'existe cependant pas d'estimation théorique permettant de prévoir le nombre de neurones cachés nécessaires pour obtenir une précision déterminée en fonction du nombre d'échantillons disponibles.

En supposant que le type et le nombre de neurones cachés aient été définis, il reste à déterminer l'ensemble des paramètres ajustables du réseau. Ces derniers sont choisis de telle sorte que l'erreur quadratique MSE(Ψ) soit minimale, en d'autres termes, en résolvant un problème d'optimisation. Pour cela, il existe deux méthodes, les méthodes non-adaptatives qui tiennent compte simultanément de toutes les valeurs échantillons disponibles, et les méthodes adaptatives qui incorporent de nouvelles valeurs échantillons au cours du processus d'apprentissage. Nous ne décrivons ici que les méthodes d'apprentissage non-adaptatives.

Considérons tout d'abord un réseau de neurones de type (RBF), les centres et les rayons ayant été choisis a priori (égaux ou non aux points échantillons). Le réseau de neurones est alors linéaire par rapport aux paramètres ajustables Ψ et la fonction MSE(Ψ) est quadratique par rapport à Ψ. En reprenant les notations du chapitre 1, en particulier que n_1 représente le nombre de fonctions à base radiale du réseaux, la fonction de sortie du neurone est égale à :

$$\tilde{J}(\alpha, \Psi) = \sum_{i=1}^{n_1} \psi_i \; g_R(\alpha, c_i, r_i) \; .$$

En définissant

$$H = \begin{bmatrix} g_R(\alpha_1, c_1, r_1) & g_R(\alpha_1, c_2, r_2) & \dots & g_R(\alpha_1, c_{n_1}, r_{n_1}) \\ \vdots & \vdots & \vdots & \vdots \\ g_R(\alpha_{n_s}, c_1, r_1) & g_R(\alpha_{n_s}, c_2, r_2) & \dots & g_R(\alpha_{n_s}, c_{n_1}, r_{n_1}) \end{bmatrix} ,$$

$$\mathcal{J}_s = \begin{bmatrix} J(\alpha_1) \\ \vdots \\ J(\alpha_{n_s}) \end{bmatrix} \quad et \quad \Psi = \begin{bmatrix} \psi_1 \\ \vdots \\ \psi_{n_1} \end{bmatrix} ,$$

l'erreur d'évaluation sur les données d'entrée est $\mathrm{MSE}(\Psi) = ||\mathcal{J}_s - H\Psi||^2$. Les dérivées de cette erreur par rapport aux paramètres ajustables sont donc égales à :

$$\begin{aligned} 1 \leq j \leq n_1 \quad , \quad \frac{\partial \mathrm{MSE}}{\partial \psi_j}(\Psi) &= 2\sum_{i=1}^{n_s} \left(\tilde{J}(\alpha_i, \Psi) - J(\alpha_i) \right) \frac{\partial \tilde{J}}{\partial \psi_j}(\alpha_i, \Psi) \\ &= 2\sum_{i=1}^{n_s} \left(\tilde{J}(\alpha_i, \Psi) - J(\alpha_i) \right) g_R(\alpha_i, c_j, r_j) \end{aligned} .$$

Le vecteur des paramètres ajustables minimisant la fonction erreur MSE doit donc satisfaire la condition nécessaire suivante :

$$\begin{aligned} \frac{\partial \mathrm{MSE}}{\partial \psi_j}(\Psi) = 0 \Rightarrow \sum_{i=1}^{n_s} J(\alpha_i) g_R(\alpha_i, c_j, r_j) &= \sum_{i=1}^{n_s} \tilde{J}(\alpha_i, \Psi) g_R(\alpha_i, c_j, r_j) \\ &= \sum_{i=1}^{n_s} \sum_{p=1}^{n_1} \psi_p \; g_R(\alpha_i, c_p, r_p) g_R(\alpha_i, c_j, r_j) \; . \end{aligned}$$

Soit sous forme matricielle

$$H^T H \Psi = H^T \mathcal{J}_s$$

Le vecteur des paramètres ajustables doit donc satisfaire la condition nécessaire suivante :

$$\Psi = (H^T H)^{-1} H^T \mathcal{J}_s \; .$$

Pour les fonctions fortement variables ou bruitées, il est nécessaire de pénaliser la norme du vecteur de taille n_1 des paramètres ajustables [157, 158]. La fonction à minimiser est alors :

$$\mathrm{MSE}(\Psi) = \sum_{i=1}^{n_s} \left(\tilde{J}(\alpha_i, \Psi) - J(\alpha_i) \right)^2 + \sum_{j=1}^{n_1} \lambda_j \psi_j^2 \; .$$

Le vecteur des paramètres ajustables doit alors satisfaire la nouvelle condition nécessaire :

$$\Psi = (H^T H - \Lambda)^{-1} H^T \mathcal{J}_s \quad , \text{où} \quad \Lambda = \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_{n_1} \end{bmatrix} \; .$$

Considérons maintenant un réseau de neurones de type (MP). Ce réseau de neurones dépend non linéairement des paramètres ajustables. La procédure de calcul du minimum de l'erreur MSE se fait de manière itérative par calcul du gradient de cette fonction. Le vecteur des paramètres ajustables à l'itération $(p+1)$ de cette procédure itérative est défini de la manière suivante :

$$\Psi^{(p+1)} = \Psi^{(p)} + \mu^p \nabla \mathrm{MSE}(\Psi^{(p)}) \; .$$

Le facteur μ^p est calculé par une optimisation unidirectionnelle dans la direction du gradient. Il s'agit d'une méthode de descente classique.

Le point important est le calcul du gradient ∇MSE. Il peut notamment se faire de façon

directe ou en utilisant un algorithme de rétropropagation évoqué dans le chapitre 1. Cette seconde méthode est la plus rapide. La sortie $y_{j,k}$ du neurone j de la sous-couche k est

$$y_{j,k} = g_P(\psi_0 + \sum_{i=1}^{n_{k-1}} \psi_{i,j,k} y_{i,k-1}) = g_P(x_{j,k})$$

où n_{k-1} représente le nombre de neurones de la sous-couche $k-1$, $y_{i,k-1}$ $(1 \leq i \leq n_{k-1})$ la sortie du neurone i de la sous-couche $k-1$, ψ_0, $0 \leq i \leq n_{k-1}$ les coefficients associés au neurone j $(1 \leq j \leq n_k)$ et $x_{j,k}$ représente l'entrée du neurone j de la sous-couche k. La fonction erreur relative à l'observation $o \in [1, n_s]$, définie par la relation $\text{MSE}_o = \left(\tilde{J}(\alpha_o, \Psi) - J(\alpha_o)\right)^2$, ne dépend de $\psi_{i,j,k}$ qu'à travers $x_{j,k}$. D'où

$$\frac{\partial \text{MSE}_o}{\partial \psi_{i,j,k}} = \frac{\partial \text{MSE}_o}{\partial x_{j,k}} \frac{\partial x_{j,k}}{\partial \psi_{i,j,k}} = \delta^o_{j,k} y^o_{i,k-1}$$

où $y^o_{i,k-1}$ correspond à la sortie du neurone i de la sous-couche $k-1$ lorsque l'entrée du réseau est l'observation o, et

$$\delta^o_{j,k} = \frac{\partial \text{MSE}_o}{\partial x_{j,k}} = 2\tilde{J}(\alpha_o, \Psi) \frac{\partial \tilde{J}(\alpha_o, \Psi)}{\partial x_{j,k}} \ .$$

Si le neurone considéré est le neurone de sortie du réseau, alors $\tilde{J}(\alpha_o, \Psi) = y_{j,k} = g_P(x_{j,k})$, et donc

$$\delta^o_{j,k} = 2\tilde{J}(\alpha_o, \Psi) g'_P(x^o_{j,k})$$

où $x^o_{j,k}$ correspond à l'entrée du neurone j de la sous-couche k lorsque les entrées du réseau global sont celles de l'observation o. Dans le cas particulier où le neurone de sortie effectue une somme pondérée des sorties des neurones qui lui sont connectés, on a $y_{j,k} = x_{j,k}$ et $\delta^o_{j,k} = 2\tilde{J}(\alpha_o, \Psi)$. En revanche, lorsque le neurone considéré n'est pas le neurone de sortie, ce dernier est relié aux n_{k+1} neurones de la sous-couche $k+1$. On a donc

$$\delta^o_{j,k} = \frac{\partial \text{MSE}_o}{\partial x_{j,k}} = \sum_{q=1}^{n_{k+1}} \frac{\partial \text{MSE}_o}{\partial x_{q,k+1}} \frac{\partial x_{q,k+1}}{\partial x_{j,k}}$$

où $x_{q,k+1}$ représente l'entrée du neurone q de la sous-couche $k+1$. Comme le neurone j de la sous-couche k est une entrée du neurone q de la sous-couche $k+1$:

$$x_{q,k+1} = \psi_0 + \sum_{j=1}^{n_k} \psi_{j,q,k+1} y_{j,k} = \psi_0 + \sum_{j=1}^{n_k} \psi_{j,q,k+1} g_P(x_{j,k}) \ ,$$

ce qui implique que

$$\frac{\partial x_{q,k+1}}{\partial x_{j,k}} = \psi_{j,q,k+1} g'_P(x_{j,k})$$

et finalement

$$\delta^o_{j,k} = \sum_{q=1}^{n_{k+1}} \frac{\partial \text{MSE}_o}{\partial x_{q,k+1}} \psi_{j,q,k+1} g'_P(x_{j,k}) = g'_P(x^o_{j,k}) \sum_{q=1}^{n_{k+1}} \delta^o_{q,k+1} \psi_{j,q,k+1} \ .$$

Les termes $\delta^o_{j,k}$ peuvent donc être calculés récursivement en parcourant les connexions à l'envers. Pour un réseau de neurones à dépendance non linéaire, notons donc que le calcul du vecteur Ψ dépend de l'initialisation.

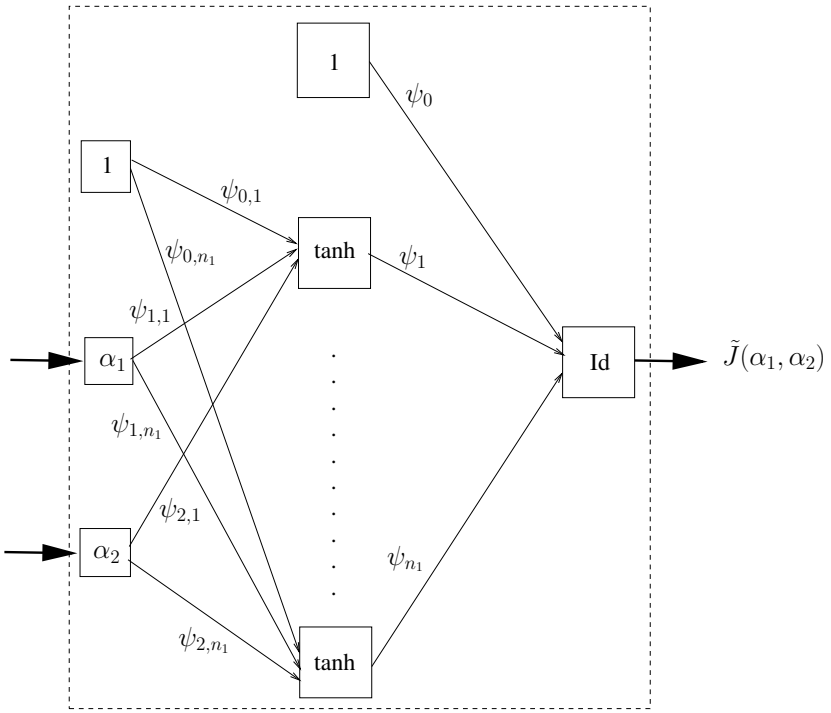

FIG. 4.4 – Réseau de neurones de type (MP) à une couche cachée utilisé dans le cadre du cas test.

Le modèle de régression choisi doit être d'une complexité suffisante pour rendre compte de la relation déterministe entre les facteurs selectionnés sans toutefois être trop complexe pour ne pas avoir une variance trop forte ; le modèle serait alors surajusté. En d'autres termes le choix du modèle consiste à résoudre le problème biais-variance. Pour déterminer si un modèle est surajusté on peut par exemple calculer la matrice jacobienne de la fonction coût $\partial \mathrm{MSE}/\partial \Psi$. Si son rang n'est pas égal au nombre de paramètres (dimension du vecteur Ψ), alors certains d'entre eux ne sont pas indépendants ; le modèle est surajusté et sa variance est certainement trop importante.

Réseau de neurones de type perceptron multi-couche

Le réseau de neurone de type (MP) considéré dans le cadre de notre cas test, ne comporte qu'une couche cachée. La fonction d'activation des neurones de la couche cachée est la fonction sigmoïde, et la fonction d'activation du neurone de sortie est la fonction identité, de sorte que la sortie du réseau est la somme pondérée des sorties des neurones de la couche cachée. Un biais est de plus ajouté aux entrées et aux sorties de la couche cachée. Ce réseau est décrit par la figure 4.4.
La fonction de sortie du neurone est donnée de manière explicite par la formule

$$\tilde{J}(\alpha) = \psi_0 + \sum_{i=1}^{n_1} \psi_i \tanh(\psi_{0,i} + \psi_{1,i}\alpha_1 + \psi_{2,i}\alpha_2)$$

Dans ce cas, le vecteur Ψ des paramètres ajustables du réseau est

$$\Psi = [\psi_0 \ \psi_1 \ \ldots \ \psi_{n_1} \ \psi_{0,1} \ \psi_{1,1} \ \psi_{2,1} \ \ldots \ \psi_{0,n_1} \ \psi_{1,n_1} \ \psi_{2,n_1}]^T \ .$$

Ce vecteur est choisi de manière à minimiser l'erreur MSE. La méthode d'optimisation choisie pour minimiser cette erreur est la méthode de plus grande descente. Dans ce cas simple, il est

inutile d'utiliser la méthode de rétropropagation du gradient de l'erreur pour calculer ∇MSE. Celui-ci peut être calculé de manière directe. Le vecteur Ψ pour lequel $\nabla\text{MSE}(\Psi) = 0$ est donc calculé de manière itérative. Le critère d'arrêt du processus est fixé par l'utilisateur. Ainsi, le processus itératif prend fin lorsque

$$|\nabla\text{MSE}(\Psi)| < \epsilon \ .$$

Le vecteur Ψ à l'itération $(p+1)$ est calculé en fonction du vecteur Ψ à l'itération (p) selon la formule

$$\Psi^{(p+1)} = \Psi^{(p+1)} - \arg\min_h \left(\text{MSE}(\Psi^{(p)}) - h \frac{d\text{MSE}}{d\Psi}(\Psi^{(p)}) \right) \frac{d\text{MSE}}{d\Psi}(\Psi^{(p)})$$

La méthode de Wolfe [43] est utilisée pour minimiser l'erreur MSE dans la direction opposée au gradient, c'est-à-dire pour calculer

$$\arg\min_h \left(\text{MSE}(\Psi^{(p)}) - h \frac{d\text{MSE}}{d\Psi}(\Psi^{(p)}) \right)$$

Réseau de neurones de type réseau de fonctions à base radiale

Les centres des fonctions à base radiale considérées coïncident exactement avec les points échantillons, en d'autres termes $n_1 = n_s$. La fonction de sortie du réseau est alors égale à

$$\begin{aligned} \tilde{J}(\alpha) &= \sum_{i=1}^{n_1} \psi_i g_R(\alpha, \alpha_i, r_i) \\ &= \sum_{i=1}^{n_1} \psi_i exp\left(-\frac{1}{2} \frac{(\alpha_1 - \alpha_1^i)^2 + (\alpha_2 - \alpha_2^i)^2}{r_i^2} \right) \end{aligned}$$

Le vecteur Ψ des paramètres ajustables du réseau est

$$\Psi = \begin{bmatrix} \psi_0 \\ \vdots \\ \psi_{n_1} \end{bmatrix} \ .$$

Dans ce cas, le vecteur Ψ minimisant de l'erreur MSE est égal à

$$\Psi = (X^T X)^{-1} X^T J_s$$

où la matrice X est définie de la manière suivante

$$X = \begin{bmatrix} g_R(\alpha_1, \alpha_1, r_1) & \dots & g_R(\alpha_1, \alpha_{n_1}, r_{n_1}) \\ \vdots & \vdots & \vdots \\ g_R(\alpha_{n_1}, \alpha_1, r_1) & \dots & g_R(\alpha_{n_1}, \alpha_{n_1}, r_{n_1}) \end{bmatrix} \ .$$

Le rayon associé aux fonctions à base radiale est égal à :

$$r = \frac{1}{\sqrt{n_1}} \max_{1 \le i,j \le n_s} \sqrt{\left((\alpha_1^i - \alpha_1^j)^2 + (\alpha_2^i - \alpha_2^j)^2 \right)} \ .$$

Ce rayon est le même pour toutes les fonctions à base radiale. Il est choisi de cette manière pour la raison suivante : imaginons que le nombre d'échantillons soit suffisamment élevé pour que

$$\int \tilde{J}(\alpha) = \sum_{i=1}^{n_1} \psi_i \int g_R(\alpha, \alpha_i, r_i) \approx \int J(\alpha) \ ,$$

il ne faut pas alors qu'en enrichissant la base d'échantillons, on détériore la valeur de l'intégrale $\int \tilde{J}$, pour cela on fait en sorte que $\int g_R(\alpha, \alpha_i, r_i) \sim \frac{1}{n_1}$, ce qui en dimension 2, revient à choisir $r \sim \frac{1}{\sqrt{n_1}}$.

Enfin, comme les centres des fonctions à base radiale coïncident avec les points échantillons, le vecteur Ψ minimisant de l'erreur MSE est égal à

$$\Psi = X^{-1} J_s \ .$$

4.3.3 Régression par méthode de Kriging

Le principe général de la régression de fonctions par Kriging a déjà été exposé au chapitre 1. En particulier, plusieurs formes de Kriging ont été définies. Précisons ici les méthodes de mise en oeuvre pratique de ces différentes formes.

Kriging simple

Etant donnée une moyenne m estimée a priori de $\mathcal{J}$, on interpole une fonction écart $z(\alpha) = \mathcal{J}(\alpha) - m$ par Kriging simple en réalisant une combinaison linéaire de n_s valeurs, dont les coefficients seront obtenus par un critère sur l'erreur quadratique moyenne.

Soit $\overline{z}(\alpha) = \sum_{i=1}^{n_s} \zeta_i(\alpha) z(\alpha_i)$ où $\overline{z}(\alpha) = \zeta^T(\alpha) \mathcal{Z}_s$ la fonction de régression. Le vecteur

$$\zeta = \begin{bmatrix} \zeta_1(\alpha) \\ \zeta_2(\alpha) \\ \vdots \\ \zeta_{n_s}(\alpha) \end{bmatrix}$$

est le vecteur des poids de la combinaison linéaire ; il est considéré comme déterministe dans les calculs d'espérance qui permettent de le calculer. Le vecteur

$$\mathcal{Z}_s = \begin{bmatrix} z(\alpha_1) \\ z(\alpha_2) \\ \vdots \\ z(\alpha_{n_s}) \end{bmatrix}$$

est le vecteur des évaluations de l'échantillon.

Dans ce cadre, on comprend immédiatement que la condition (BLUP1) est satisfaite automatiquement : $E(\overline{z}(\alpha)) = \sum_{i=1}^{n_s} \zeta_i(\alpha) E(z(\alpha_i)) = 0$. En supposant qu'une loi de covariance de la fonction z a été choisie, les coefficients de la combinaison linéaire sont choisis de telle sorte que l'erreur MSE soit minimale (condition (BLUP2))

$$\mathrm{MSE}(\alpha) = E((z(\alpha) - \overline{z}(\alpha))^2) = E((z(\alpha) - \sum_{i=1}^{n_s} \zeta_i(\alpha) z(\alpha_i))^2)$$

L'erreur peut être réécrite sous forme développée :

$$\mathrm{MSE}(\alpha) = \sum_{i=1}^{n_s} \sum_{j=1}^{n_s} \zeta_i(\alpha) \zeta_j(\alpha) E(z(\alpha_i) z(\alpha_j)) - 2 \sum_{i=1}^{n_s} \zeta_i(\alpha) E(z(\alpha_i) z(\alpha)) + E(z(\alpha)^2)$$

Par hypothèse sur le processus stochastique $E(z(\alpha)^2) = \sigma^2$. Le vecteur ζ doit donc satisfaire la condition nécessaire suivante

$$\forall i \in [1, \ldots, n_s] \qquad \frac{\partial \text{MSE}(\alpha)}{\partial \zeta_i} = 2\sum_{j=1}^{n_s} \zeta_j(\alpha) E(z(\alpha_i)z(\alpha_j)) - 2\sum_{i=1}^{n_s} E(z(\alpha_i)z(\alpha)) = 0$$

Pour aboutir aux équations usuellement présentées, il convient de diviser chaque équation du système linéaire précédent par le carré de l'écart type du processus statistique σ^2. On rappelle que les définitions, données au chapitre 1, de la matrice C dont les composantes sont $C_{ij} = E(z(\alpha_i), z(\alpha_j)/\sigma^2 = Cov(z(\alpha_i), z(\alpha_j))/\sigma^2$ et du vecteur K_α dont les composantes sont $K_{\alpha\ i} = E(z(\alpha_i), z(\alpha))/\sigma^2 = Cov(z(\alpha_i), z(\alpha))/\sigma^2$, de telle sorte que

$$C\zeta = K_\alpha \Rightarrow \zeta = C^{-1}K_\alpha \qquad \overline{z}(\alpha) = (C^{-1}K_\alpha)^T \mathcal{Z}_s = {K_\alpha}^T (C^{-1})^T \mathcal{Z}_s = {K_\alpha}^T C^{-1} \mathcal{Z}_s$$

Comme indiqué précédemment, on substitue en pratique un vecteur d'évaluation d'une fonction déterministe $Z_s = J_s - m$ au vecteur $\mathcal{Z}_s$.

Ces termes apparaîtront dans des calculs plus complexes, mais dont le principe est le même, pour le Kriging complet et le Kriging ordinaire.

Dans le cadre de notre cas test, nous avons considéré que la valeur de la fonction J au centre de l'espace de conception $\mathcal{D}$ était une bonne approximation de la valeur moyenne m de cette fonction sur l'ensemble de l'espace. La fonction de covariance choisie est

$$Cov(z(\alpha_i), z(\alpha_j)) = exp(-\theta||\alpha_i - \alpha_j||) = exp\left(-\theta\sqrt{\left((\alpha_1^i - \alpha_1^j)^2 + (\alpha_2^i - \alpha_2^j)^2\right)}\right) \ , \ 1 \leq i, j \leq n_s \, .$$

Le paramètre θ est déterminé selon une heuristique classique [118]. Sa valeur est choisie telle que les termes $exp(-\theta||\alpha_i - \alpha_j||)$, $1 \leq i, j \leq n_s$ soient tous supérieurs à 0.2.

Kriging universel

Le Kriging universel utilise un processus stochastique $\mathcal{J}(\alpha)$ de la forme :

$$\mathcal{J}(\alpha) = \sum_{j=1}^{k} \beta_j f_j(\alpha) + z(\alpha) = f_\alpha^T \beta + z(\alpha)$$

où les vecteurs f_α et β sont égaux à $f_\alpha = [f_1(\alpha) \ \ f_2(\alpha) \ldots f_k(\alpha)]^T$ et $\beta = [\beta_1 \ \ \beta_2 \ldots \beta_k]^T$, et cherche une fonction approchée de la forme $\tilde{J}(\alpha) = c^T(\alpha)\mathcal{J}_s$.

On rappelle les définitions du vecteur $K_\alpha = [\phi^{\theta,p}(\alpha - \alpha_1) \ ... \ \phi^{\theta,p}(\alpha - \alpha_{n_s})]^T$ de taille n_s et dépendant du point d'évaluation α, ainsi que des matrices F de taille (n_s,k) et C (symétrique) de taille (n_s,n_s) effectuée au chapitre 1

$$F = \begin{bmatrix} f_1(\alpha_1) & f_2(\alpha_1) & \ldots & f_k(\alpha_1) \\ \vdots & \vdots & \vdots & \vdots \\ f_1(\alpha_{n_s}) & f_2(\alpha_{n_s}) & \ldots & f_k(\alpha_{n_s}) \end{bmatrix}$$

$$C = \begin{bmatrix} \phi^{\theta,p}(\alpha_1 - \alpha_1) & \phi^{\theta,p}(\alpha_2 - \alpha_1) & \ldots & \phi^{\theta,p}(\alpha_{n_s} - \alpha_1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi^{\theta,p}(\alpha_1 - \alpha_{n_s}) & \phi^{\theta,p}(\alpha_2 - \alpha_{n_s}) & \ldots & \phi^{\theta,p}(\alpha_{n_s} - \alpha_{n_s}) \end{bmatrix} .$$

Les coefficients de l'approximation linéaire $\tilde{J}$ sont calculés de telle sorte qu'ils satisfassent les conditions (BLUP1) et (BLUP2). En utilisant le fait que $E(\mathcal{Z}_s) = 0 = E(z(\alpha))$ et en supposant que le vecteur des coefficients $c(\alpha)$ est déterministe, on a :

$$E(\tilde{J}(\alpha)) = E(c^T(\alpha)\mathcal{J}_s) = E(c^T(\alpha)(F\beta + \mathcal{Z}_s)) = E(c^T(\alpha)F\beta) + c^T(\alpha)E(\mathcal{Z}_s) = E(c^T(\alpha)F\beta)$$

$$E(\mathcal{J}(\alpha)) = E(f_\alpha^T\beta + z(\alpha)) = E(f_\alpha^T\beta) + E(z(\alpha)) = E(f_\alpha^T\beta)$$

La condition (BLUP1) impose donc l'égalité suivante :

$$F^T c(\alpha) = f_\alpha$$

Toujours en supposant que le vecteur des coefficients $c(\alpha)$ est déterministe et en utilisant la condition (BLUP1), l'erreur MSE(α) associée au Kriging universel est égale à :

$$\begin{aligned}
\text{MSE}(\alpha) &= E((\tilde{J}(\alpha) - \mathcal{J}(\alpha))^2) = E((c^T(\alpha)(F\beta + \mathcal{Z}_s) - (f_\alpha^T\beta + z(\alpha)))^2) \\
&= E(((c^T(\alpha)F - f_\alpha^T)\beta + c^T(\alpha)\mathcal{Z}_s - z(\alpha))^2) \\
&= E((c^T(\alpha)\mathcal{Z}_s - z(\alpha))^2) \\
&= E((\sum_{i=1}^{n_s} c_i(\alpha)z(\alpha_i) - z(\alpha))^2) \\
&= E(\sum_{1\le i,j\le n_s} c_i(\alpha)c_j(\alpha)z(\alpha_i)z(\alpha_j) - 2\sum_{i=1}^{n_s} c_i(\alpha)z(\alpha_i)z(\alpha) + z(\alpha)^2) \\
&= \sum_{1\le i,j\le n_s} c_i(\alpha)c_j(\alpha)E(z(\alpha_i)z(\alpha_j)) - 2\sum_{i=1}^{n_s} c_i(\alpha)E(z(\alpha_i)z(\alpha)) + E(z(\alpha)^2) \\
&= \sigma^2(1 - 2c^T(\alpha)K_\alpha + c^T(\alpha)Cc(\alpha))
\end{aligned}$$

puisque par hypothèse $E(z(\alpha)^2) = \sigma^2$. Choisir l'estimateur linéaire satisfaisant les conditions (BLUP1) et (BLUP2) revient à résoudre le problème d'optimisation suivant :

α étant fixé, trouver $c(\alpha)$ tel que MSE$(\alpha) = \sigma^2(1 - 2c^T(\alpha)K_\alpha + c^T(\alpha)Cc(\alpha))$ soit minimale sous la containte $F^T c(\alpha) = f_\alpha$.

Ce problème d'optimisation sous contrainte peut être résolu en introduisant le lagrangien :

$$\mathcal{L}(\lambda, c) = \text{MSE}(\alpha) + 2\sigma^2\lambda^T(F^T c(\alpha) - f_\alpha) = \sigma^2(1 - 2c^T(\alpha)K_\alpha + c^T(\alpha)Cc(\alpha) + 2\lambda^T(F^T c(\alpha) - f_\alpha))$$

Les vecteurs c et λ solutions du problème d'optimisation satisfont (on utilise le fait que C est une matrice symétrique)

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial \lambda} = 2\sigma^2(F^T c(\alpha) - f_\alpha) = 0 \\ \dfrac{\partial \mathcal{L}}{\partial c} = 2\sigma^2(-K_\alpha + Cc(\alpha) + F\lambda) = 0 \end{cases}$$

Le vecteur $[c(\alpha)\ \lambda(\alpha)]^T$ solution vérifie donc

$$\begin{bmatrix} C & F \\ F^T & 0 \end{bmatrix} \begin{bmatrix} c(\alpha) \\ \lambda(\alpha) \end{bmatrix} = \begin{bmatrix} K_\alpha \\ f_\alpha \end{bmatrix}$$

On calcule d'abord λ :

$$c(\alpha) = C^{-1}(K_\alpha - F\lambda) => F^T c(\alpha) = f_\alpha = F^T C^{-1} K_\alpha - F^T C^{-1} F\lambda$$
$$\Rightarrow \lambda = (F^T C^{-1} F)^{-1}(F^T C^{-1} K_\alpha - f_\alpha)$$

Puis en réinjectant dans la première équation, on trouve :

$$c(\alpha) = C^{-1}(K_\alpha - F\lambda) = C^{-1}(K_\alpha - F(F^T C^{-1} F)^{-1}(F^T C^{-1} K_\alpha - f_\alpha))$$

En d'autres termes,

$$\begin{bmatrix} c(\alpha) \\ \lambda(\alpha) \end{bmatrix} = \begin{bmatrix} C^{-1} - C^{-1}F(F^T C^{-1} F)^{-1} F^T C^{-1} & C^{-1}F(F^T C^{-1} F)^{-1} \\ (F^T C^{-1} F)^{-1} F^T C^{-1} & -(F^T C^{-1} F)^{-1} \end{bmatrix} \begin{bmatrix} K_\alpha \\ f_\alpha \end{bmatrix}$$

$$= \begin{bmatrix} C^{-1} - (F^T C^{-1})^T (F^T C^{-1} F)^{-1} (F^T C^{-1}) & (F^T C^{-1})^T (F^T C^{-1} F)^{-1} \\ (F^T C^{-1} F)^{-1}(F^T C^{-1}) & -(F^T C^{-1} F)^{-1} \end{bmatrix} \begin{bmatrix} K_\alpha \\ f_\alpha \end{bmatrix}$$

$$= \begin{bmatrix} C & F \\ F^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} K_\alpha \\ f_\alpha \end{bmatrix}$$

L'approximation linéaire $\tilde{J}$ est donc égale à :

$$\tilde{J}(\alpha) = f_\alpha^T (F^T C^{-1} F)^{-1} F^T C^{-1} \mathcal{J}_s + K_\alpha^T C^{-1}(\mathcal{J}_s - F(F^T C^{-1} F)^{-1} F^T C^{-1} \mathcal{J}_s) \ .$$

Il est alors possible de réécrire $\tilde{J}$, cherché sous la forme $c^T(\alpha)\mathcal{J}$, sous une forme semblable à $\mathcal{J}$:

$$\tilde{J}(\alpha) = f_\alpha^T \hat{\beta} + K_\alpha^T C^{-1}(\mathcal{J}_s - F\hat{\beta})$$

Par analogie, le vecteur $\hat{\beta} = (F^T C^{-1} F)^{-1} F^T C^{-1} \mathcal{J}_s$ est appelé estimateur de β. Remarquons que l'écart type σ^2 n'intervient pas dans l'expression de l'approximation $\tilde{J}$ de la fonction J par Kriging. Comme indiqué précédemment, l'échantillon de la fonction déterministe J_s est *in fine* subsitué à $\mathcal{J}_s$ dans les formules précédentes.

A partir d'un échantillon de taille n_s, il convient donc de calculer les matrices F, C et C^{-1}, puis le vecteur $\hat{\beta}$ de taille k. Pour chaque évaluation, il faut calculer les vecteurs K_α et f_α. La formule de régression ainsi définie est linéaire en les composantes de J_s mais non linéaire en α.

Le vecteur $c(\alpha)$ étant désormais connu, on peut calculer explicitement la valeur de l'erreur quadratique $\mathrm{MSE}(\alpha)$ dans le cadre statistique du Kriging :

$$\begin{aligned}
\mathrm{MSE}(\alpha) &= \sigma^2(1 - 2c^T(\alpha)K_\alpha + c^T(\alpha)Cc(\alpha)) \\
&= \sigma^2(1 - 2c^T(\alpha)K_\alpha + c^T(\alpha)(K_\alpha - F\lambda)) \\
&= \sigma^2(1 - c^T(\alpha)K_\alpha - c^T(\alpha)F\lambda)) \\
&= \sigma^2(1 - c^T(\alpha)K_\alpha - f_\alpha^T\lambda)) \\
&= \sigma^2(1 - K_\alpha^T c(\alpha) - f_\alpha^T\lambda)) \\
&= \sigma^2\left(1 - [K_\alpha^T \ f_\alpha^T]\begin{bmatrix} c(\alpha) \\ \lambda \end{bmatrix}\right) \\
&= \sigma^2\left(1 - [K_\alpha^T \ f_\alpha^T]\begin{bmatrix} C & F \\ F^T & 0 \end{bmatrix}^{-1}\begin{bmatrix} K_\alpha \\ f_\alpha \end{bmatrix}\right)
\end{aligned}$$

Kriging ordinaire

Rappelons que dans cette approche la fonction μ se réduit à une constante à déterminer et la fonction z est l'écart entre la fonction approchée et cette constante. Il suffit d'appliquer dans les formules du Kriging universel les simplifications $k = 1$ et $f_1 = 1$ pour obtenir du Kriging ordinaire. La matrice rectangle F devient un vecteur colonne à n_s lignes où chaque composante vaut 1 ; ce vecteur est noté $\underline{1}$. Soit finalement

$$\beta = \frac{\underline{1}^T C^{-1} \mathcal{J}_s}{\underline{1}^T C^{-1} \underline{1}} \qquad \tilde{J}(\alpha) = \beta + K_\alpha^T C^{-1} (\mathcal{J}_s - \underline{1}\beta)$$

Interpolation

Vérifions que le Kriging est un processus exact d'interpolation, c'est à dire que $\tilde{J}(\alpha_i) = J(\alpha_i)$ pour $1 \leq i \leq n_s$. La fonction approchée $\tilde{J}(\alpha)$ est définie par :

$$\tilde{J}(\alpha) = c^T(\alpha) \mathcal{J}_s$$

où $c(\alpha)$ satisfait les deux conditions suivantes :

$$\begin{cases} F^T c(\alpha) = f_\alpha \iff \begin{bmatrix} f_1(\alpha_1) & f_1(\alpha_2) & \dots & f_1(\alpha_{n_s}) \\ f_2(\alpha_1) & f_2(\alpha_2) & \dots & f_2(\alpha_{n_s}) \\ \vdots & \vdots & \vdots & \vdots \\ f_k(\alpha_1) & f_k(\alpha_2) & \dots & f_k(\alpha_{n_s}) \end{bmatrix} \begin{bmatrix} c_1(\alpha) \\ c_2(\alpha) \\ \vdots \\ c_{n_s}(\alpha) \end{bmatrix} = \begin{bmatrix} f_1(\alpha) \\ f_2(\alpha) \\ \vdots \\ f_k(\alpha) \end{bmatrix} \\ \\ \text{MSE}(\alpha) = c(\alpha)^T C c(\alpha)^T - 2c(\alpha)^T K_\alpha + 1 \geq 0 \text{ est minimal} \end{cases}$$

Lorsque α prend une valeur particulière α_i, le vecteur $\delta_i = [0 \ \dots \ 1 \ \dots \ 0]^T$ ayant toutes ses composantes nulles sauf la $i^{\text{ième}}$ composante qui est égale à 1 satisfait la première condition. Vérifions qu'il vérifie également la deuxième :

$$\delta_i^T C \delta_i^T - 2\delta_i^T K_{\alpha_i} + 1 = Cor(\alpha_i, \alpha_i) - 2Cor(\alpha_i, \alpha_i) + 1 = 0$$

puisque $Cor(\alpha_i, \alpha_i) = 1$. D'où la propriété recherchée

$$\tilde{J}(\alpha_i) = \delta_i J_s = \mathcal{J}(\alpha_i) = J(\alpha_i)$$

Calcul des paramètres θ, σ, p de la fonction de covariance de l'écart

Les fonctions $f_j, 1 \leq j \leq k$ et la forme de la fonction corrélation étant choisies, il reste à déterminer les paramètres : σ^2, $\theta = [\theta_1 \ ... \ \theta_{n_s}]$, $p = [p_1 \ ... \ p_{n_s}]$, et $\beta = [\beta_1 \ ... \ \beta_k]$. Le principe de ce choix a déjà été donné au chapitre 1. Nous le décrivons ici en détails.

Pour choisir ces paramètres, Sacks, Welch, Mitchell et Wynn [188] proposent la technique dite du maximum de vraisemblance - "*Maximum Least Estimation MLE*" en anglais. Connaissant un ensemble de réalisations de la fonction $\mathcal{J}$, il s'agit de trouver les paramètres du problèmes qui maximisent la probabilité d'obtenir ces valeurs. Autrement dit, il s'agit d'optimiser la probabilité conditionnelle $P(\mathcal{J}_1, ..., \mathcal{J}_{n_s} | \sigma^2, \theta, p, \beta)$. En supposant que les observations $\mathcal{J}_1, ..., \mathcal{J}_{n_s}$ sont indépendantes et que le processus stochastique mis en jeu est Gaussien, cette probabilité est égale à

[192] :

$$\begin{aligned} \text{MLE} = P(\mathcal{J}_1, ..., \mathcal{J}_{n_s}|\sigma^2, \theta, p, \beta) &= \prod_{i=1}^{n_s} P(\mathcal{J}_i|\sigma^2, \theta, p, \beta) \\ &= \frac{1}{(2\pi\sigma^2 det(C))^{n_s/2}} exp(\frac{-(\mathcal{J}_s - F\beta)^T C^{-1}(\mathcal{J}_s - F\beta)}{2\sigma^2}) \end{aligned}$$

En général, pour des raisons de simplicité, on n'optimise pas cette fonction, mais son logarithme appelé fonction log-vraisemblance et égale à :

$$\ln(\text{MLE}) = \frac{-1}{2}\left(n_s \ln(\sigma^2) + \ln(det(C)) + \frac{(\mathcal{J}_s - F\beta)^T C^{-1}(\mathcal{J}_s - F\beta)}{\sigma^2}\right)$$

En général, les paramètres $p_i, 1 \leq i \leq n_s$ sont choisis égaux à 2. Les paramètres σ^2, θ, β solutions du problème de maximisation de la fonction ln(MLE) doivent satisfaire les trois conditions suivantes :

$$(1) \quad \frac{\partial}{\partial\beta}(\ln(\text{MLE})) = \frac{\partial}{\partial\beta}((\mathcal{J}_s - F\beta)^T C^{-1}(\mathcal{J}_s - F\beta)) = 0$$

$$\Rightarrow \beta_{max} = \hat{\beta} = (F^T C^{-1} F)^{-1} F^T C^{-1} \mathcal{J}_s$$

$$(2) \quad \frac{\partial}{\partial\sigma^2}(\ln(\text{MLE})) = \frac{\partial}{\partial\sigma^2}(n_s \ln(\sigma^2) + \frac{(\mathcal{J}_s - F\beta)^T C^{-1}(\mathcal{J}_s - F\beta)}{\sigma^2}) = 0$$

$$\Rightarrow \sigma^2_{max} = \frac{1}{n_s}((\mathcal{J}_s - F\beta)^T C^{-1}(\mathcal{J}_s - F\beta))$$

$$(3) \quad \frac{\partial}{\partial\theta}(\ln(\text{MLE})) = 0$$

On remarque tout d'abord que les paramètres β et σ^2 sont découplés et fonction de θ. Le problème d'optimisation revient donc à trouver le paramètre θ conduisant à la valeur maximale de la fonction log-vraisemblance. Remarquons également que la méthode du maximum de vraisemblance et la méthode (BLUP) conduisent au même β. On trouve, en injectant β_{max} et σ^2_{max} dans la fonction ln(MLE) que la valeur θ à choisir est :

$$\begin{aligned} \theta &= \text{Argmax}\left[\frac{-1}{2}\left(n_s \ln(\sigma^2_{max}) + \ln(det(C))\right)\right] \\ &= \text{Argmin}\left[det(C)^{\frac{1}{n_s}}\sigma^2\right] \end{aligned}$$

Pour le Kriging ordinaire les deux premières formules sont bien sûr à remplacer par

$$\sigma^2 = \frac{1}{n_s}(\mathcal{J}_s - \underline{1}\beta)^T C^{-1}(\mathcal{J}_s - \underline{1}\beta) \qquad \beta = \frac{\underline{1}^T C^{-1}\mathcal{J}_s}{\underline{1}^T C^{-1}\underline{1}}$$

La matrice de covariance C tend à être de moins en moins bien conditionnée au fur et à mesure que les points α_i, $1 \leq i \leq n$, se rapprochent de l'optimum [191]. Une manière d'améliorer le conditionnement de cette matrice est d'éviter la valeur 2, p est alors choisi dans l'intervalle

$[0, 1.99]$.

En résumé, pour définir complétement la fonction de covariance Cov (choisie sous la forme $\sigma^2 \prod_{i=1}^{n_f} e^{-\theta_i |\alpha_{1i} - \alpha_{2i}|^{p_i}}$), les paramètres σ^2, $\theta = [\theta_1 \ ... \ \theta_{n_s}]$ et $p = [p_1 \ ... \ p_{n_s}]$ doivent être précisés. L'approche (BLUP) ne permet pas de calculer ces paramètres, ceux-ci doivent être fixés au préalable. En revanche, l'approche faisant appel au maximum de vraisemblance permet de les calculer (le vecteur p est cependant souvent fixé a priori). Notons toutefois que la variance σ^2 n'intervient que dans l'expression de l'erreur MSE mais pas dans le calcul de l'approximation linéaire $\tilde{J}$. Finalement, la méthode (BLUP) ne permet de calculer que les coefficients de l'approximation linéaire $\tilde{J}$, les paramètres θ et p étant fixés, alors que la méthode du maximum de vraisemblance, conduit à une expression de l'approximation linéaire identique et permet en outre de calculer les paramètres θ et p.

Remarque : comme l'expliquent den Hertog, Kleijnen et Siem [53], l'hypothèse sur laquelle repose l'expression de la variance du modèle de Kriging est fausse : $c(\alpha)$ dépend de Cor donc de θ, déterminé par le critère ci-dessus, et calculé en utilisant le vecteur $\mathcal{J}_s$. Le vecteur $c(\alpha)$ n'est donc pas indépendant de $\mathcal{J}_s$. Il est prouvé dans [53] que l'expression donnée ci-dessus est en fait une borne inférieure de l'erreur réelle.

4.3.4 Evaluations des performances des modèles approchés retenus

L'objectif de cette section est de discuter des performances relatives des différents modèles approchés sélectionnés sur le cas test considéré, et ainsi d'évaluer leur efficacité pour un problème d'optimisation. Nous n'avons pas utilisé de boîte à outil dédiée à l'optimisation, mais avons tout de même eu recours au "package" d'algèbre linéaire LAPACK [7].

Dans le cadre particulier de notre cas test, seuls deux paramètres de forme sont considérés. Le coût de calcul des modèles approchés est négligeable devant le coût d'une simulation stationnaire d'un écoulement fluide. Pour cette raison, l'efficacité d'un modèle approché pour l'optimisation est mesuré par le nombre d'échantillons, en d'autres termes, de calculs stationnaires d'écoulements, nécessaires pour atteindre une certaine précision.

La stratégie d'enrichissement de la base d'échantillons est la même pour tous les modèles approchés, elle procède de la manière suivante :
A - un nombre d'échantillons suffisant pour permettre la détermination des paramètres ajustables du modèle est choisi et la base d'échantillons de départ est construite sous forme d'hypercubes latins ;
B - des points échantillons sont ajoutés à la base d'échantillons si le critère C ($C1$, $C2$ ou $C3$ décrits ci-dessous) n'est pas satisfait ;
$B1$ - si les minima et les maxima locaux du modèle approché ne sont pas tous dans la base d'échantillons, alors jusqu'à quatre des points manquants sont ajoutés à cette base ;
$B2$ - sinon quatre points de l'espace de conception $\mathcal{D}$ et situés à une distance maximale des points échantillons sont ajoutés à la base d'échantillons.

Définition des critères d'évaluation

Sur l'échantillonnage complet du domaine de conception $\mathcal{D}$ selon un ensemble de 21×21 points, la fonction exacte J possède une maximum global situé en $(-0.4, 0.4)$, un minimum global situé en $(0.4, -0.4)$ un maximum local situé en $(0.04, 0.4)$ et deux minima locaux situés en $(-0.2, -0.2)$ et en $(0.4, 0.08)$.

Modèle approché	C1	Itérations	C2	Itérations	C3	Itérations	E(100)
Polynôme de degré 2	KO	-	KO	-	KO	-	$2.3\ 10^{-3}$
Polynôme de degré 4	KO	-	KO	-	KO	-	$2.1\ 10^{-3}$
Polynôme de degré 6	OK	47	KO	-	KO	-	$1.2\ 10^{-3}$
Polynôme de degré 8	OK	53	OK	65	OK	73	$4.9\ 10^{-4}$
Réseau de neurones (MP)	OK	300	KO	-	KO	-	$2.5\ 10^{-3}$
Réseau de neurones (RBF)	OK	37	OK	38	OK	46	$1.6\ 10^{-3}$
Kriging simple	OK	25	OK	62	OK	170	$3.7\ 10^{-4}$

TAB. 4.1 – Performances des modèles approchés étudiés.

La performance des modèles approchés est évaluée selon leur capacité à satisfaire les critères suivants :
$C1$ - leur capacité à reconstruire une approximation telle que l'erreur moyenne sur l'échantillonnage 21×21 du domaine de conception $\mathcal{D}$ soit inférieure à 2.10^{-3}, cette erreur, notée E, est adimensionnée par la variation totale de la fonction J sur le domaine $\mathcal{D}$. Elle est donnée par la formule

$$E = \frac{1}{21^2(J_{max} - J_{min})} \sqrt{\sum_{1 \leq i,j \leq 21} \left(J(\alpha_1^{i,j}, \alpha_2^{i,j}) - \tilde{J}(\alpha_1^{i,j}, \alpha_2^{i,j}) \right)^2} \ ;$$

$C2$ - leur capacité à localiser les maxima globaux et locaux sur l'échantillonnage 21×21 du domaine $\mathcal{D}$ (une tolérance spatiale discrète est précisée) ;
$C3$ - leur capacité à trouver les minima globaux et locaux de façon exacte sur l'échantillonnage 21×21 du domaine $\mathcal{D}$ (une tolérance spatiale discrète, significativement plus faible que la précédente, est précisée).

On ne cherche pas à localiser les maxima avec une précision aussi importante que les minima car la dérivée seconde prend des valeurs faibles dans le voisinage des maxima, de sorte qu'il est difficile de localiser ces derniers.

Les résultats obtenus avec les différents modèles de régression considérés sont résumés dans le tableau 4.1. Le nombre d'itérations indiqué correspond au nombre d'évaluations exactes requises pour satisfaire un des critères $C1$, $C2$ ou $C3$. L'erreur E indiquée correspond à l'erreur du modèle approché lorsque la taille de la base d'échantillons a atteint la valeur 100.

Interprétation des résultats

D'un point de vue général, le réseau de neurones (RBF) (cf figure 4.5) et le Kriging simple (cf figure 4.6) conduisent aux meilleurs résultats. Ces deux modèles approchés satisfont le critère $C2$ en ayant recours à un nombre raisonnable d'évaluations exactes. En revanche, seuls le réseau de neurones (RBF) et la surface de réponse polynômiale de degré 8 réussissent à satisfaire le critère $C3$ avec un nombre raisonnable d'évaluations exactes.

Les vues en coupe de la fonction exacte (coupes α_1 = constante ou α_2 = constante sur la figure 4.3) ont une forme beaucoup plus compliquée que de simples paraboles. La fonction exacte ne peut donc pas être approchée de manière suffisamment précise par un polynôme du degré deux. L'observation des courbes correspondant aux surfaces de réponse polynômiales de degré 4 et 6 conduit à la même conclusion. Il faut au moins un polynôme de degré 8 pour parvenir à une précision suffisante.

La fonction approchée calculée par le réseau de neurones (RBF) ne dépend que faiblement de l'échantillonnage initial. Deux réseaux de neurones (MP) ont été considérés, le premier avec une

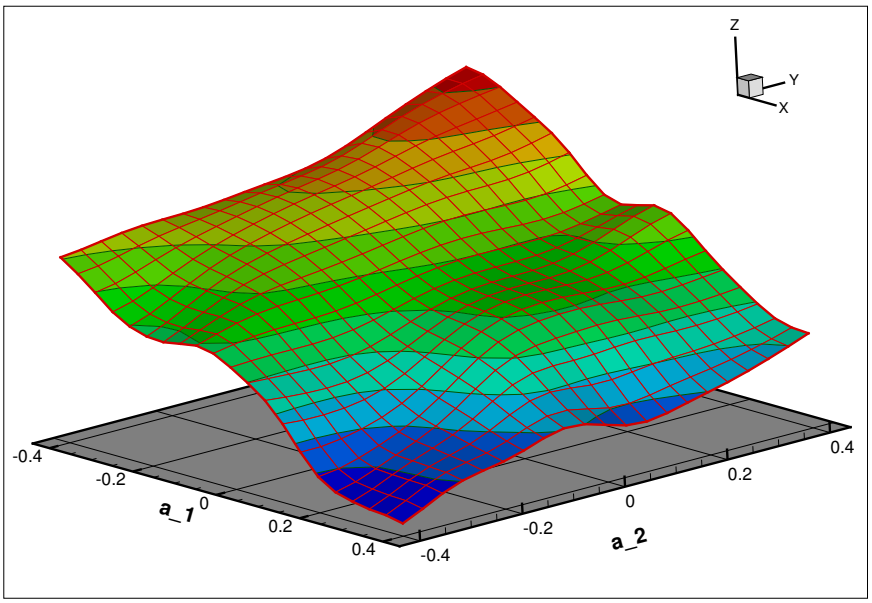

FIG. 4.5 – Fonction approchée issue du réseau de neurones (RBF) satisfaisant le critère $C3$.

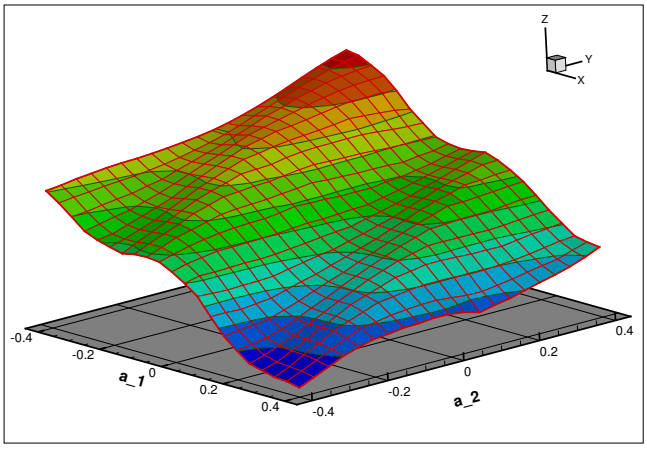

FIG. 4.6 – Fonction approchée calculée par Kriging simple satisfaisant le critère $C3$.

Modèle approché	C1	Itérations	C2	Itérations	C3	Itérations	E(100)
Kriging simple	OK	25	OK	62	OK	170	$3.7\ 10^{-4}$
Kriging ordinaire	OK	25	OK	62	OK	170	$3.7\ 10^{-4}$
Kriging universel (f_α^1)	OK	28	OK	65	OK	65	$3.6\ 10^{-4}$
Kriging universel (f_α^2)	OK	28	OK	65	OK	65	$3.6\ 10^{-4}$

TAB. 4.2 – Performances des modèles de régression par Kriging.

Modèle approché	C1	Itérations	C2	Itérations	C3	Itérations	E(100)
Enrichissement $B2$	OK	28	OK	65	OK	65	$3.6\ 10^{-4}$
Enrichissement $B2'$	OK	28	OK	57	OK	57	$4.1\ 10^{-4}$

TAB. 4.3 – Régression par Kriging universel avec fonction de régression polynômiale de degré 1.

couche cachée comportant cinq neurones et le deuxième avec une couche cachée comportant dix neurones. Le premier a quasiment besoin de la totalité de la discrétisation 21×21 du domaine $\mathcal{D}$ pour satisfaire le critère $C1$, et le second requiert jusqu'à 200 évaluations exactes. Si ces réseaux parviennent à localiser les optima globaux au bout de 42 évaluations exactes, il ne réussissent pas en revanche à localiser les optima locaux. De plus, l'erreur moyenne ne varie plus beaucoup de la valeur $2.4\ 10^{-3}$ à partir d'une base d'échantillons de taille 50.

C'est la méthode du Kriging simple qui, après un nombre fixé d'itérations, conduit à la plus faible erreur moyenne. Cependant, elle ne conduit pas au modèle approché le plus efficace en matière de localisation des deux maxima.

Méthodes de Kriging plus avancée

Nous étudions dans cette partie le bénéfice que l'on peut avoir à utiliser une méthode de Kriging ordinaire ou de Kriging universel à la place du Kriging simple. Pour la méthode de Kriging ordinaire $f_\alpha = [1]$. Deux fonctions de régression ont été considérées pour le Kriging universel, la première est une fonction polynômiale de degré 1, on a alors $f_\alpha = f_\alpha^1 = [1, \alpha_1, \alpha_2]$, la seconde est une fonction polynômiale de degré 2, dans ce cas $f_\alpha = f_\alpha^2 = [1, \alpha_1, \alpha_2, \alpha_1^2, \alpha_1\alpha_2, \alpha_2^2]$. La façon dont ces modèles satisfont les conditions $C1$, $C2$ et $C3$ est résumée dans le tableau 4.2.

Les performances de ces méthodes de Kriging plus élaborées sont semblables à celles de la méthode la plus simple (Kriging simple) dans le cadre de notre cas test, à l'exception de la capacité à localiser de façon précise les deux maxima, pour laquelle ces modèles sont plus performants.

Le calcul de l'erreur MSE peut être utilisé pour élaborer une stratégie d'enrichissement de la base d'échantillons sensiblement différente. En effet, l'étape $B2$ peut être remplacée par l'étape $B2'$:

$B2'$ - sinon quatre points (de l'échantillonnage complet 21×21 du domaine $\mathcal{D}$) pour lesquels l'erreur MSE est maximale sont ajoutés à la base d'échantillons.

Cette stratégie a été testée avec la méthode de Kriging universel utilisant une fonction de régression polynômiale de degré 1 (f_α^1). La comparaison avec la stratégie $B2$ est récapitulée dans le tableau 4.3. Pour une raison non identifiée, la stratégie $B2'$ conduit à une erreur moyenne après 100 évaluations exactes supérieure à celle donnée par la stratégie $B2$. En revanche, elle permet de localiser plus rapidement les maxima.

Enfin, des tests ont été effectués quant à l'influence de la valeur du coefficient θ apparaissant dans la fonction de corrélation. Pour les résultats présentés ci-dessus, la valeur 2.84 a été utilisée.

Celle-ci a été calculée en s'appuyant sur l'heuristique consistant à assurer qu'aucune des composantes de la matrice C n'est inférieur à 0.2. En réalité, lorsque θ varie dans l'intervalle $[2\ 10^{-5}, 5]$, les performances (relativement aux critères $C1$, $C2$ et $C3$) de ces modèles de régression par Kriging sont quasiment inchangées. Pour des valeurs plus élevées, la fonction approchée présente quasiment autant d'optima locaux que de points dans la base d'échantillons, tandis que pour des valeurs plus faibles, la matrice C est mal conditionnée. En raison de ces considérations d'ordre pratique, la valeur théoriquement optimale de θ (voir paragraphe 4.3.3) n'a pas été utilisée.

En conclusion, parmi tous ces modèles de régression, les modèles de régression par Kriging et par réseau de fonctions à base radiale sont ceux qui donnent les meilleurs résultats en terme d'approximation de la fonction exacte.

Conclusion

Ce travail a principalement consisté en l'étude et la mise en oeuvre d'une méthode de calcul des gradients des fonctions aérodynamiques par rapport à des paramètres géométriques pour un système aéroélastique soumis à un écoulement lointain stationnaire.

Une technique de calcul de l'équilibre aéroélastique statique a tout d'abord été développée. Dans ce cadre, le comportement du fluide peut être modélisé par les équations d'Euler ou par les équations de Navier-Stokes moyennées (RANS), celles-ci étant numériquement résolues par elsA [40], code de simulation numérique pour la mécanique des fluides développé à l'ONERA. Le comportement de la structure est, quant à lui, prédit par la théorie des poutres et les équations d'Euler-Bernoulli. Le chargement aérodynamique est transmis à la structure par l'intermédiaire de la matrice des coefficients d'influence également appelée matrice de flexibilité. Seuls les efforts de torsion et de flexion, constituant les sollicitations aérodynamiques prépondérantes, sont transmis, de manière consistante, à la structure dont seuls les mouvements de flexion et de torsion sont calculés sous l'hypothèse des petits déplacements. La déformation résultante sur le maillage du domaine fluide est prédite analytiquement par analogie avec la mécanique du solide. Enfin, le système aéroélastique couplé est résolu selon un processus itératif inspiré de la méthode du point fixe.

Cette étape accomplie, il a été possible de mettre en oeuvre un cadre de calcul, pour le système aéroélastique considéré, des gradients de fonctions d'intérêt (objectif, contraintes) par rapport à un vecteur de paramètres géométriques de la forme solide. Les gradients peuvent être calculés par la méthode de l'équation linéarisée discrète ou par la méthode du vecteur adjoint discret. Ces méthodes reposent sur la résolution de systèmes linéaires couplés. Dans le cadre de ce travail, la résolution des systèmes linéaires couplés est effectuée par un processus itératif doublement retardé. D'une part, l'inconnue vectorielle relative à une discipline est calculée à l'itération courante en utilisant l'inconnue relative à l'autre discipline évaluée à l'itération précédente - cette technique, également appelée "*lagged process*" en anglais, a été proposée par Martins [135]. D'autre part, pour s'affranchir de l'inversion de matrices monodisciplinaires de grande taille, nous avons introduit une notion de retard semblable pour le calcul des inconnues de la structure lors de la résolution des équations associées à ces dernières.

Ces développements ont enfin été appliqués au calcul des gradients des coefficients aérodynamiques de traînée et de portance par rapport à un ensemble de paramètres de forme pour trois configurations de complexité croissante : équations d'Euler résolues sur un maillage multibloc coïncident, équations RANS résolues sur un maillage monobloc, et finalement, équations RANS résolues sur un maillage multibloc non-coïncident. La validité des résultats a été établie par comparaison aux gradients calculés par différences finies. Lorsque le comportement du fluide est modélisé par les équations d'Euler, les valeurs des gradients obtenues par les méthodes linéarisée et adjointe sont quasiment identiques à celles évaluées par différences finies. En revanche lorsque le comportement du fluide est modélisé par les équations RANS, la comparaison est moins satisfaisante : l'écart avec les valeurs calculées par différences finies est de l'ordre de 10 à 20%. Ce biais s'explique en partie par l'hypothèse selon laquelle le coefficient de viscosité turbulente est invariant avec la forme, faite lors de l'établissement des équations linéarisée et adjointe. Les

gradients calculés, bien que sensiblement différents des valeurs obtenues par différences finies, fournissent cependant des directions de descente convenables pour un processus d'optimisation de forme locale.

Une dernière partie du travail a été consacrée à l'évaluation des performances de quatre modèles réduits non physiques (surfaces de réponse polynômiale, réseaux de neurones de type perceptron multi-couches, réseaux de fonctions à base radiale, régression par la méthode de Krige) dans le cadre d'un processus d'optimisation de forme. Ces dernières ont été estimées sur une configuration bidimensionnelle de turbomachine.

Un ensemble de compléments peuvent être apportés à ce travail. D'une part, les paramètres de forme que nous avons considérés lors du calcul pratique des gradients avaient la propriété de ne pas modifier la forme en plan de la voilure ; de telle sorte que les propriétés de la structure ont été supposées invariables et que le maillage de la structure ne dépendait pas du vecteur des paramètres de forme. En complément du travail réalisé ici, il pourrait être intéressant de prendre en compte les effets d'une modification des paramètres de forme sur les caractéristiques de la structure en introduisant une dépendance à la fois géométrique et mécanique au vecteur des paramètres de forme. D'autre part, en dépit du fait que la croisière, régime décrit par notre cadre de calcul de l'équilibre aéroélastique statique, n'est pas la phase réellement dimensionnante pour la structure, dont les caractéristiques sont en revanche déterminées par les phases de décollage, d'atterrissage et de descente d'urgence, il pourrait s'avérer intéressant, à des fins de généralisation, d'envisager le calcul des gradients de fonctions propres à la structure.

Enfin, les développements présentés dans ce manuscrit sont applicables en phase d'avant projet. Afin de les rendre adéquats à des stades de conception plus avancés, il pourrait être bénéfique d'améliorer le niveau de fidélité de la représentation de la structure, en considérant par exemple une modélisation de type éléments finis de cette dernière.

Ce travail offre un nombre de perspectives intéressantes. Plus particulièrement, le calcul des gradients aéroélastiques est une discipline particulière d'un domaine plus vaste : l'optimisation de forme, mais il sera une des clés des évolutions aérodynamiques futures. En effet, bien que la tendance actuelle ne soit plus aux optimisations locales par gradients, comme aux prémices de l'optimisation de forme aérodynamique, mais aux optimisations de formes globales, celles-ci seront, à l'avenir, de plus en plus couplées à des techniques d'optimisation locales pour l'analyse fine des zones potentielles d'optimum. De plus, les modèles approchés auxquels ces optimisations globales ont recours pour diminuer leurs coûts de calcul, cherchent actuellement à élever leur niveau de précision en tenant compte non seulement de données sur la fonction à approcher, mais aussi de données sur son gradient. En outre, depuis quelques années déjà, la tendance n'est plus à l'optimisation isolée du critère d'une seule discipline sans prise en compte des interactions avec les autres disciplines, mais plutôt à l'optimisation multidisciplinaire, c'est-à-dire à l'optimisation conjointe des critères des disciplines pertinentes ou à la prise en compte des interactions multidisciplinaires lors de l'optimisation d'une discipline particulière. L'interaction fluide-structure étant une des interactions prépondérantes sur le comportement d'un aéronef, le calcul des gradients aéroélastiques trouve toute sa légitimité dans ce contexte de développement futur. Enfin, plus encore qu'à l'inclusion des interactions interdisciplinaires, la tendance actuelle est à l'optimisation robuste, en d'autres termes à la prise en compte, lors de la conception, des incertitudes possibles sur les paramètres du problème.

Bibliographie

[1] M. Akgun, R. Haftka, K. Wu et J. Walsh : Sensitivity of lumped constraints using the adjoint method. AIAA Paper 99-1314, 1999.

[2] G. Van Albada, B. Van Leer et W. Roberts : A comparative study of computational methods in cosmic gaz dynamics. *Astronomy and Astrophysics*, 108:76–84, 1982.

[3] N. Alexandrov et R. Lewis : Comparative properties of collaborative optimization and other approaches to mdo. pages 39–46, Bradford, England, UK, 1999. Engineering Design Optimization, MCB Univ. Press.

[4] N. Alexandrov et R. Lewis : Analytical and computational aspects of collaborative optimization for multidisciplinary design. *AIAA Journal*, 40(2):301–309, February 2002.

[5] G. Allaire : *Analyse numérique et optimisation.* Ecole Polytechnique, 2002.

[6] J. Allison : Complex system optimization : A review of analytical target cascading, collaborative optimization, and other formulations. Mémoire de D.E.A., University of Michigan, 2004.

[7] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney et D. Sorensen : *LAPACK Users' Guide.* SIAM, 1999.

[8] K. Appa : Finite-surface spline. *Journal of Aircraft*, 26(5), 1989.

[9] A. Arslan et L. Carlson : Determination of sensitivity derivatives for an aeroelastic transonic wing. *Journal of Aircraft*, 33(1), 1996.

[10] A. Auger, S. Kern et N. Hansen : A restart CMA evolution strategy with increasing population size. Special Session on Real-Parameter Optimization, Conference on Evolutionary Computation, CEC-05, Edinburgh, UK, 2-5 septembre 2005.

[11] G. Awanou, M-J. Lai et P. Wenston : The multivariate spline method for scattered data fitting and numerical solutions of partial differential equations. Wavelet and Splines Conference, Athens, GA, 2005.

[12] M. Baker et J. Giesing : A practical approach to mdo and its application to an hsct aircraft. Los Angeles, CA, September 1995. 1st AIAA Aircraft Engineering, Technology, and Operations Congress, AIAA Paper 95-3885.

[13] R. Balling et J. Sobieszczanski-Sobieski : Optimization of coupled systems : a critical overview of approaches. Rapport technique, NASA Contractor Report 195019 - ICASE Report 94-100, 1994.

[14] A Barron : Universal approximation bounds for superposition of a sigmoidal function. *IEEE Transactions on Information Theory*, 39:930–945, 1993.

[15] J-F. Barthelemy et F. Bergen : Shape sensitivity analysis of wing static aeroelastic characteristics. Rapport technique, NASA, 1988.

[16] J-F. Barthelemy, G. Wrenn, A. Dovi et L. Hall : Supersonic transport wing minimum design integrating aerodynamics and structures. *AIAA Journal of Aircraft*, 31, 1994.

[17] N. Bartoli, M. Samuelides, D. Bailly et M. Marcelet : Projet de recherche fédérateur DOOM - revue de modèles réduits pour l'optimisation. Rapport technique, ONERA, 2006.

[18] R. Barton : Kriging interpolation in simulation : a state of art review. Proceedings of the 1994 winter simulation conference, 1994.

[19] J. Batina : Unsteady Euler airfoil solutions using unstructured dynamic meshes. Reno, NV, 1989. 27^{th} Aerospace Sciences Meeting, AIAA Paper, 89-0115.

[20] O. Baysal et M. Eleshaky : Aerodynamic design sensitivity analysis methods for the compressible euler equations. *Journal of Fluids Engineering*, 113(4):681–688, 1991.

[21] M. Bendsoe : *Optimization of Structural Topology, Shape, and Material.* Berlin : Springer, 1995.

[22] C. Bernardi et Y. Maday : Mesh adaptivity in finite elements using the mortar method. *Revue Européenne de Éléments Finis*, 9(4):451–465, 2000.

[23] K. Bhatia et J. Wertheimer : Aeroelastic challenges for a high speed civil transport. AIAA Paper 93-1478, 1993.

[24] R.L. Bisplinghoff, H. Ashley et R.L. Halfman : *Aeroelasticity.* Dover Science Books, 1996.

[25] C. Blondeau et Jean-Sébastien Schotté : Formulations MDO. Rapport technique, ONERA-DADS, 2006. RT 3/11576.

[26] P. Boggs et J. Tolle : Sequential quadratic programming. *Acta Numerica*, 1996.

[27] E. Bonabeau, M. Dorigo et G. Theraulaz : *Swarm Intelligence : From Natural to Artificial Systems.* Oxford University Press, 1999.

[28] J. Bonnans, E. Panier, L. Tits et J. Zhou : Avoiding the maratos effect by means of a nonmonotone line search ii. inequality constrained problems - feasible iterates. *SIAM Journal on Numerical Analysis*, 29:1187–1202, 1992.

[29] E. Bonomi et J.-L. Lutton : Le recuit simulé. *Pour la Science*, (129):68–77, 1988.

[30] K. Bowman, R. Grandhi et F. Eastep : Structural optimization of lifting surfaces with divergence and control reversal constraints. *Structural Optimization*, 1, 1989.

[31] C. Braun, A. Boucke, J. Ballmann, M. Hanke et A. Karavas : Numerical prediction of the model deformation of a high speed transport aircraft type wing by direct aeroelastic simulation. Rapport technique HIRETT/TR/RWTH/CB/18032003/1, Rheinisch-Westfalische Technische Hochschule Aachen Lehr-und Forschungsgebiet fur Mechanik, 2003.

[32] C. Braun, A. Boucke, G. Wellmer, A. Karavas et J. Ballmann : Numerical prediction of the influence of deployed ailerons and dynamic pressure on the model deformation of a high speed transport aircraft type wing by direct aeroelastic simulation (WP 3.2). Rapport technique HIRETT/TR/RWTH/CB/31072003/1, Rheinisch-Westfalische Technische Hochschule Aachen Lehr-und Forschungsgebiet fur Mechanik, 2003.

[33] C. Braun, G. Wellmer et J. Ballmann : Influence of altered wing twist at inboardwingstations on pressure distributions. Rapport technique HIRETT/M/RWTH/CB/23052003/1, Rheinisch-Westfalische Technische Hochschule Aachen Lehr-und Forschungsgebiet fur Mechanik, 2003.

[34] R. Braun : *Collaborative Optimization : An Architecture For Large-Scale Distributed Design.* Thèse de doctorat, Stanford University, April 1996.

[35] R. Braun et R. Kroo : Development and application of the collaborative optimization architecture in a multidisciplinary design environment. In Multidisciplinary Design Optimization - State of the Art. Proceedings of the ICASE/NASA Langley Workshop on Multidisciplinary Design Optimization, SIAM, 1997.

[36] S. Brown : Displacement extrapolations for cfd+csm aeroelastic analysis. AIAA-97-1090, 1997.

[37] G. Burgreen et O. Baysal : Three-dimensional aerodynamic shape optimization using discrete sensitivity analysis. *AIAA Journal*, 34:1761–1770, 1996.

[38] C. Byun et G. Guruswamy : Static aeroelasticity computations for flexible wing-body-control configurations. AIAA-96-4059.

[39] L. Cambier et M. Gazaix : Elsa : an efficient object-oriented solution to cfd complexity. 40*th* AIAA Aerospace Science Meeting and Exhibit, AIAA Paper 2002-108.

[40] L. Cambier et J.P. Veuillot : Status of the elsa CFD software for flow simulation and multidisciplinary applications. 46*th* AIAA Aerospace Science Meeting, AIAA Paper 2008-664.

[41] G. Carrier : Multi-disciplinary optimization of a supersonic transport aircraft wing planform. Jyväskylä, July 2004. ECCCOMAS 2004, European Congress on Computational Methods in Applied Sciences and Engineering.

[42] G. Carrier : Single and multi-point aerodynamic optimizations of a supersonic transport aircraft wing using optimization strategies involving adjoint method and genetic algorithm. Las Palmas, April 2006. ERCOFTAC.

[43] G. Casalis, B. Lécussan, F. Rogier, M. Samuélidès et P. Villedieu : *Calcul Scientifique*. SUPAERO, 2002.

[44] V. Cerny : Thermodynamical approach to the travelling salesman problem : an efficient simulation algorithm. *Journal of Optimization, Theory and Application*, 45, 1985.

[45] R. Chaney : On the pironneau-polak method of centers. *Journal of Optimization Theory and Applications*, 20:269–295, 1976.

[46] X. Chen : *Methods of Feasible Directions for Nonlinear Programming : Theory and Algorithms*. Thèse de doctorat, Department of Mathematical Sciences, Clemson University, 1999.

[47] X. Chen et M. Kostreva : A generalization of the norm-relaxed method of feasible directions. *Applied Mathematics and Computation*, 102(2-3):257–272, 1999.

[48] E. Cramer, J. Dennis, P. Frank et R. Lewis : Problem formulation for multidisciplinary optimization. *SIAM Journal of Optimization*, 4:754–776, 1994.

[49] N. Cressie : *Statistics for spatial data*. Wiley-Interscience, New-York, 1991.

[50] W. Davidon : Variable metric method for minimization. *SIAM Journal on Optimization*, pages 1–17, 1991.

[51] T. Deman et E. Dowell : Experimental and theoretical study of gust response for high-aspect-ratio wing. *AIAA Journal*, 40(3):419–429, 2002.

[52] A. Demeulenaere et S. Pierret : An integrated optimization system for multistage gas turbine design. Proceedings of Eurogen 2003, 2003.

[53] D. den Hertog, J. Kleijnen et A. Siem : The correct kriging variance estimated by bootstapping. Rapport technique, Tilburg University, May 2004.

[54] D. Destarac : Far-field / near-field drag balance and applications of drag extraction in cfd. *VKI Lecture Series 2003, CFD-based Aircraft Drag Prediction and Reduction, National Institute of Aerospace, Hampton (VA)*, November 2003.

[55] I. Salah El Din, G. Carrier et S. Mouton : Discrete adjoint method in elsA (Part II) : application to aerodynamic design optimization.

[56] J. Délery, R. Gaillard, G. Losfeld et C. Pendria : Study of the iso cascade in the ONERA s5ch wind tunnel, pressure probe and LDV measurement. Rapport technique RT 192/1865 DAFE, ONERA, 1999.

[57] H. Doi et J. Alonso : Fluid/structure coupled aeroelastic computations for transonic flows in turbomachinery. ASME Turbo Expo 2002, Amsterdam, The Netherlands, GT-2002-30313, 3-6 June 2002.

[58] J. Donea : An arbitrary Lagrangian-Eulerian finite element method for transient fluid-structure interactions. *Computational Methods Applied to Mechanical Engineering*, 33:689–723, 1982.

[59] M. Dorigo : *Optimization, Learning and Natural Algorithms.* Thèse de doctorat, Politecnico di Milano, Italie, 1992.

[60] M. Dorigo, M. Birattari et T. Stützle : Ant colony optimization : Artificial ants as a computational intelligence technique. *IEEE Computational Intelligence Magazine*, 1(4):28–39, 2006.

[61] G. Dreyfus, J.-M. Martinez, M. Samuélidès, M. Gordon, F. Badran, S. Thiria et L. Hérault : *Réseaux de neurones, Méthodologies et applications.* Eyrolles, 2002.

[62] J. Dréo, A. Petrowski, E. Taillard et P. Siarry : *Métaheuristiques pour l'optimisation difficile.* 2003.

[63] J. Duchon : Splines minimizing rotation-invariant semi-norms in Sobolev spaces. *In* W. Schempp et K. Zeller, éditeurs : *Constructive theory of functions of several variables*, pages 85–100. Springer-Verlag, Berlin.

[64] A. Dugeai : Méthode de mouvement de maillage pour l'aéroélasticité. Rapport technique, Rapport ONERA (sans mention de protection) RTS 101/3064 DDSS/Y. 1998, 1998.

[65] R. Duvigneau : Approche "boîtes noires" 1. Cours de la première Ecole de Printemps. Optimisation et Contrôle des Ecoulements et des Transferts. Mai 2006.

[66] M. Eleshaky et O. Baysal : Shape optimization of a 3D nacelle near a flat plate wing using multiblock sensitivity analysis. AIAA Paper 94-0160, 1994.

[67] O. Elwakeil et J. Arora : Methods for finding feasible points in constrained optimization. *AIAA Journal*, 33(9):1715–1719, 1995.

[68] C. Farhat, M. Lesoinne et P. LeTallec : A conservative algorithm for exchanging aerodynamic and elastodynamic data in aeroelastic systems. AIAA Paper 98-16371.

[69] C. Farhat, M. Lesoinne et P. LeTallec : Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces : Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 157(1), 1998.

[70] C. Farhat, M. Lesoinne et N. Maman : Mixed explicit/implicit time integration of coupled aeroelastic problems : Three-field formulation, geometric conservation and distributed solution. *International Journal for Numerical Methods in Fluids*, 21:807–835, 1995.

[71] A. Fazzolari, N. Gauger et J. Brezillon : Efficient aerodynamic shape optimization in MDO context. *Proc. Appl. Math. Mech.*, 5, 2005.

[72] G. Fillola, M.-C. Le Pape et M. Montagnac : Numerical simulations around wing control surfaces. ICASE 2004, 2004.

[73] R. Fletcher : A new approach to variable metric algorithms. *Computer Journal*, 13:317–322, 1970.

[74] R. Franke : Scattered data interpolation : Tests of some methods. *Mathematics of Computations*, 38(157):181–200, 1982.

[75] P. Friedman : Helicopter vibration reduction using structural optimization with aeroelastic/multidisciplinary constraints - a survey. *AIAA Journal of Aircraft*, 28, 1991.

[76] P. Friedman : Renaissance of aeroelasticity and its future. *Journal of Aircraft*, 36(1):105–121, Jan-Feb 1999.

[77] U. Garcia-Palomares et O. Mangasarian : Superlinearly convergent quasi-newton methods for nonlinearly constrained optimization problems. *Mathematical Programing*, 11:1–13, 1976.

[78] O. Ghattas et X. Li : Domain decomposition methods for sensitivity analysis of a nonlinear aeroelastic problem. *International Journal of Computational Fluid Dynamics*, 11, 1998.

[79] K. Giannakoglou, D. Papadimitriou et I. Kampolis : Coupling evolutionnary algorithms, surrogate models and adjoint methods in inverse design and optimization problems. *VKI Lecture Series* 2004-07, 2004.

[80] K. Giannakoglou, D. Papadimitriou et I. Kampolis : Aerodynamic shape design using evolutionnary algorithms and new gradient assisted metamodels. *computer methods in applied mechanics and engineering*, 195:6312–6329, 2006.

[81] J. Giesing et J. Barthelemy : A summary of industry mdo applications and needs. AIAA Paper 98-4737, 1998.

[82] P. Girodroux-Lavigne et A. Dugeai : Fluid-structure coupling with elsa : status of developments and examples of application. 42^e Colloque d'aérodynamique appliquée, Couplages et optimisation multidisciplinaire, Nice, France, 19-21 march 2007.

[83] A. Giunta : A novel sensitivity analysis method for high-fidelity multidisciplinary optimization of aero-structural systems. AIAA Paper 2000-16542.

[84] A. Giunta, J. Dudley, R. Narducci, B. Grossman, R. Haftka, W. Mason et L. Watson : Noisy aerodynamic response and smooth approximations in HSCT design. AIAA Paper 1994-4376, 1994.

[85] A. Giunta et J. Sobieszcsanski-Sobieski : Progress toward using sensitivity derivatives in a high-fidelity aeroelastic analysis of a supersonic transport. AIAA Paper 98-4763.

[86] F. Glover et M. Laguna : *Tabu search*. Kluwer Academic Publishers, Dordrecht, 1997.

[87] D. Goldberg : *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., 1989.

[88] R. Griffith et R. Stewart : A nonlinear programming technique for the optimization of continuous processing systems. *Management Science*, 7:379–392, 1961.

[89] B. Grossman, R. Haftka, P. Kao, D. Polen, M. Rais-Rohani et J. Sobieszczanski-Sobieski : Integrated aerodynamic-structural design of a transport wing. *Journal of aircraft*, 27(12):1050–1056, 1990.

[90] W. Gutjahr : A graph-based ant system and its convergence. *Future Generation Computer Systems*, 16:873–888, 2000.

[91] R. Haftka : Sensitivity calculations for iteratively solved problems. *International journal of numerical methods in engineering*, 21:1535–1546, 1985.

[92] R. Haftka : Structural optimization with aeroelastic constraints - a survey of u.s. applications. *International Journal of Vehicle Design*, 7, 1986.

[93] P. Hajela et C-Y Lin : *Computational engineering using metaphors from nature*, chapitre Real versus binary coding in genetic algorithms : a comparative study, pages 77–83. Civil-Comp press, Edinburgh, UK, 2000.

[94] S. Han : A globally convergent variable metric algorithms for general nonlinear programming problems. *Journal of Optimization Theory and Applications*, 22:297–309, 1977.

[95] R. Harder et R. Demarais : Interpolation using surface splines. *AIAA Journal*, 1972.

[96] R. Hardy : Multiquadratic equations of topography and other irregular surfaces. *Journal of geophysical research*, 1971.

[97] E. Haug, K. Choi et V. Komkov : *Design Sensitivity Analysis of Structural Systems.* Academic Press : Orlando, 1986.

[98] D. Hebb : *The organization of behavior.* John Wiley, New York, 1949.

[99] R. Hicks et P. Henne : Wing design by numerical optimization. AIAA Paper 77-1247.

[100] R. Hicks, E. Murman et G. Vanderplaats : An assesment of airfoil design by numerical optimization. Rapport technique, NASA, 1976.

[101] R. Hicks et G. Vanderplaats : *Design of Low-Speed Airfoil by Numerical Optimization.* SAE 750524, 1975.

[102] R. Hicks et G. Vanderplaats : *Airfoil Section Drag Reduction at Transonic Speeds by Numerical Optimization.* SAE 760477, 1976.

[103] R. Hicks et G. Vanderplaats : *Application of Numerical Optimization to the Design of Supercritical Airfoil Without Drag Creep.* SAE 770440, 1977.

[104] J. Holland : *Adaptation in Natural and Artificial Systems.* University of Michigan Press, 1975.

[105] Y. Hong, Q. Ren, J. Zeng et Y. Chang : Convergence of estimation of distribution algorithms in optimization of additively noisy fitness functions. 17^{th} IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05), 2005.

[106] J. Hopfield : Neural networks and physical systems with emergent collective computational abilities. Proceedings in the National Academy of Sciences, USA, 1982.

[107] K. Hornik : Approximation capabilities of multilayer feedforward networks. *Neural Network*, 4:251–257, 1991.

[108] K. Hornik, M. Stinchcombe et H.White : Multi-layer feedforward networks are universal approximators. *Neural Network*, 2:359–366, 1989.

[109] K. Hornik, M. Stinchcombe et H.White : Universal approximation of an unknown mapping and its derivative using multilayer feedforward networks. *Neural Network*, 3:551–560, 1990.

[110] G. Hou et A. Satyanarayana : Analytical sensitivity analysis of a static aeroelastic wing. AIAA Paper 2000-40084.

[111] K. Hulme et C. Bloebaum : A comparison of solution strategies for simulation-based multidisciplinary design optimization. AIAA Paper 98-4977, 1998.

[112] O.-P. Jacquotte, F. Montigny et G. Coussement : Generation, optimization, and adaptation of multiblock structured grids for complex configurations. *Survey on Mathematics for Industry*, 4:267–277, 1995.

[113] A. Jameson : Aerodynamic design via control theory. *Journal of Scientific Computing*, 3, 1988.

[114] A. Jameson : Automatic design of transonic airfoils to reduce the shock induced pressure drag. 31^{th} Israel Annual Conference on Aviation and Aeronautics, Tel-Aviv, February 1990.

[115] A. Jameson, L. Martinelli et N. Pierce : Optimum aerodynamic design using the Navier-Stokes equations. *Theoretical and computational fluid dynamics*, 10:213–237, 1998.

[116] A. Jameson et J. Reuther : Control theory based airfoil design using the euler equations. *AIAA Journal*, 34, 1996.

[117] R. Jin, W. Chen et T. Simpson : Comparative studies of metamodeling techniques under multiple modeling criteria. AIAA Paper 2000-4801, 2000.

[118] J. Jouhaud, P. Sagaut, M. Montagnac et J. Laurenceau : A surrogate-model based multidisciplinary shape optimization method application to a 2d subsonic airfoil. *Computers & Fluids*, 36:520–529, 2007.

[119] E. Kansa : Multiquadratics - a scattered data approximation scheme with applications to computational fluid dynamics, parts I and II. *Computer and mathematics with applications*, 1990.

[120] J. Kelley : The cutting-plane method for solving convex programs. *Industrial and Applied Mathematics*, 1960.

[121] J. Kennedy et R. Eberhart : *Swarm Intelligence*. Morgan Kaufmann Publishers, San Francisco, California, 2001.

[122] S. Kirkpatrick, S. Gelatt et M. Vecchi : Optimization by simulate annealing. *Science*, 220(4598), 1983.

[123] U. Kirsch : *Structural Optimization : Fundamentals and Applications*. Berlin : Springer, 1993.

[124] M. Kleiber, H. Antunez, H. Hien et P. Kowalczyk : *Parameter sensitivity in nonlinear mechanics*. Wiley : Chichester, 1997.

[125] S. Kodiyalam et J. Sobieszcsanski-Sobieski : Multidisciplinary design optimization - some formal methods, framework requirements, and application to vehicle design. *International Journal of Vehicle Design*, 25:3–22, 2001.

[126] D. Krige : A statistical approach to some mine valuations and allied problems at the witwatersrand. Mémoire de D.E.A., University of Witwatersrand, 1951.

[127] I. Kroo : Distributed multidisciplinary design and collaborative optimization. VKI lecture series on Optimization Methods and Tools for Multicriteria/Multidisciplinary Design, November 2004.

[128] I. Kroo, J. Gallman et S. Smith : Aerodynamic and structural studies of joined wing aircraft. AIAA/AHS/ASEE Aircraft Design, Systems and Operations Meeting, St. Louis, Missouri, AIAA Paper 87-2931, 14-16 September 1987.

[129] P. Laarhoven et E. Aarts : *Simulated annealing : theory and applications*. Springer, 1987.

[130] P. Larranaga et J. Lozano, éditeurs. *Estimation of Distribution Algorithms : A New Tool for Evolutionary Computation (Genetic Algorithms and Evolutionary Computation)*. Kluwer Academic Publishers, 2001.

[131] J.L. Lions : *Optimal control of systems governed by partial differential equations*. Springer Verlag, New-York, 1971.

[132] M. Lores, P. Smith et R. Hicks : Supercritical wing design using numerical optimization and comparisons with experiment. AIAA Paper 79-0065, 1979.

[133] N. Maman et C. Farhat : Matching fluid and structure meshes for aeroelastic computations : a parallel approach. *Computers & Structures*, 54(4), 1995.

[134] J. Martins : *A coupled-adjoint method for high-fidelity aero-structural optimization*. Thèse de doctorat, Stanford University CA, November 2000.

[135] J. Martins, J. Alonso et J. Reuther : High-fidelity aerostructural design optimization of a supersonic businees jet. *Journal of Aircraft*, 41(3):523–530, 2004.

[136] G. Matheron : *Les variables régionalisées et leur estimation.* Masson, Paris, 1965.

[137] K. Mathias et D. Whitley : Transforming the search space with gray coding. pages 513–518. IEEE Conference on Evolutionary Computation, 1994.

[138] K. Maute, M. Nikbay et C. Farhat : Coupled analytical sensitivity analysis and optimization of three-dimensional nonlinear aeroelastic systems. *AIAA Journal*, 39(11), 2001.

[139] K. Maute, M. Nikbay et C. Farhat : Sensitivity analysis and design optimization of three-dimensional nonlinear aeroelastic systems by the adjoint method. *International Journal for Numerical Methods in Engineering*, 56(6), 2003.

[140] K. Maute et G. Reich : Integrated multidisciplinary topology optimization approach to adaptative wing design. *Journal of Aircraft*, 43(1):253–263, January-February 2006.

[141] W. McCulloch et W. Pitts : A logical calculus of the ideas immanent in neurons activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

[142] M. Meaux : Amélioration de la chaîne d'optimisation numérique de formes aérodynamiques en vue de traitement de configuration complexes en écoulement visqueux. rapport de stage ENSICA, 2001.

[143] R. Meric : Coupled optimization in steady-state thermoelasticity. *Journal of thermal stresses*, 8:333–347, 1985.

[144] R. Meric : Material and load optimization of thermoelastic solids. Part I : sensitivity analysis. Part II : numerical results. *Journal of thermal stresses*, 9:359–388, 1986.

[145] R. Meric : Optimal cross-sectional shape for MHD channel flows. *International journal for numerical methods in engineering*, 30:919–929, 1990.

[146] P. Michaleris, D. Tortorelli et C. Vidal : Tangent operators and design sensitivity formulations for transient nonlinear coupled problems with applications to elastoplasticity. *International journal for numerical methods in engineering*, 37:2471–2499, 1994.

[147] P. Michaleris, D. Tortorelli et C. Vidal : Analysis and optimization of weakly coupled thermoelastoplastic systems with application to weldment design. *International journal for numerical methods in engineering*, 378:1259–1285, 1995.

[148] H. Mülhenbein et Paass : From recombination of genes to the estimation of distribution i. binary parameters. *Lectures Notes in Computer Science*, 1411, 1996.

[149] H. Moller et E. Lund : Shape sensitivity analysis of strongly coupled fluid-structure interaction problems. AIAA Paper 2000-4823, 2000.

[150] F. Murat et J. Simon : Etude de problèmes d'optimum design. volume 41, pages 54–62. 7^{th} IFIP Conference Lecture Notes in COmputer Sciences, 1976.

[151] K. Nakahashi et G. Deiwert : Self-adaptative-grid method with application to airfoil flow. *AIAA Journal*, 25:513–520, 1987.

[152] J. Nelder et R. Mead : A simplex method for function minimization. *Computer Journal*, 7(4):308–313, 1965.

[153] C. Newman : *Integrated multidisciplinary design optimization using discrete sensitivity analysis for geometrically complex aeroelastic configurations.* Thèse de doctorat, Virginia Tech, VA, 1997.

[154] S. Obayashi et D. Sasaki : Self-organizing map of Pareto solutions obtained from multiobjective supersonic wing design. AIAA Paper 2002-0991, 2002.

[155] J. Olds : The suitability of selected multidisciplinary design and optimization techniques to conceptual aerosapce vehicle design. Cleveland, Ohio, September 1992. 4^{th} AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, AIAA Paper 92-4791.

[156] ONERA. *elsA, Theoretical Manual*, 2007.

[157] M. Orr : Introduction to radial basis function networks. Center for cognitive science, University of Edinburgh, Scotland, April 1996.

[158] M. Orr : Recent advances in radial basis functions networks. Institute for adaptative and neural computation, Division of informatics, University of Edinburgh, Scotland, June 1999.

[159] D. Papadimitriou et K. Giannakoglou : Direct, adjoint and mixed approaches for the computation of hessian in airfoil design problems. *International Journal for Numerical Methods in Fluids*, 53:455–469, 2007.

[160] D. Papadimitriou et K. Giannakoglou : Computation of the hessian matrix in aerodynamic inverse design using continuous adjoint formulations. *Computers & Fluids*, 2008.

[161] N. Papila, W. Shyy, N. Fitz-Coy et R. Haftka : Assessment of neural network polynomial based techniques for aerodynamic applications. AIAA Paper 1999-33401, 1999.

[162] M. Patil et D. Hodges : On the importance of aerodynamic and structural geometrical nonlinearities in aeroelastic behavior of high-aspect-ratio wings. *Journal of fluids and structures*, 19(7):905–915, 2004.

[163] S. Patnaik, R. Coroneos, D. Hopkins et T. Lavelle : Lessons learned during solutions of multidisciplinary design optimization problems. *Journal of Aircraft*, 39(3):386–393, May-June 2002.

[164] J. Peter : Discrete adjoint method in elsA (Part I) : method/theory. 7^{th} ONERA-DLR Aerospace Symposium, 2006.

[165] J. Peter et F. Drullion : Large stencil viscous flux linearization for the simulation of 3D compressible turbulent flows with backward-Euler schemes. *Computers and fluids*, 2007.

[166] J. Peter et M. Marcelet : Comparison of surrogate models for turbomachinery design. *WSEAS Transactions on Fluid Mechanics*, 3(1):10–17, January 2007.

[167] J. Peter, M. Marcelet et S. Burguburu : Introduction à l'optimisation de formes en aérodynamique et quelques exemples d'application. *Publication ONERA*, 2006. 2006-01 ISSN 0078-3780.

[168] C.-T. Pham : *Linéarisation du flux visqueux des équations de Navier-Stokes et de modèles de turbulence pour l'optimisation aérodynamique en turbomachines.* Thèse de doctorat, Ecole Nationale Supérieure d'Arts et Métiers, 2006.

[169] S. Piperno, C. Farhat et B. Larrouturou : Partitioned procedures for the transient solution of coupled aeroelastic problems. *Computational Methods Applied to Mechanical Engineering*, 124:79–112, 1995.

[170] O. Pironneau : On optimal shapes for stokes flow. *Journal of Fluid Mechanics*, 70(2), 1973.

[171] O. Pironneau : *Optimal Shape Design for Elliptic Systems.* Springer Verlag, 1984.

[172] C. Polini, P. Loris, L. Larussi, S. Pieri et V. Pediroda : Robust design of aircraft components : a multi-objective optimization problem. *VKI lecture series 2004-07*, 2004.

[173] X. Qi et F. Palmieri : The diversification role of the crossover in the genetic algorithm. Proceedings of the Fifth European Conference on Genetic Algorithm, 1993.

[174] N. Queipo, R. Haftka, W. Shy, T. Goel, R. Vaidyanathan et P. Tucker : Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41:1–28, 2005.

[175] M. Rai et N. Madavan : Aerodynamic design using neural networks. *AIAA Journal*, 38:173–182, 2000.

[176] M. Rais-Rohani, R. Haftka, B. Grossman et E. Unger : Integrated aerodynamic-structural-control wing design. Cleveland, Ohio, September 1992. 4^{th} AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, AIAA Paper 92-4694.

[177] I. Rechenberg : *Cybernetic Solution Path of an Experimental Problem.* Royal Aircraft Establishment Library Translation, 1965.

[178] G. Redeker et R. Muller : A comparison of experimental results for the transonic flow around the DFVLR-F4 wing-body configuration. Rapport technique, GARTEUR-TP018, DLR-IB 129-83/21, 1983.

[179] F. Renac, C.-T. Pham et J. Peter : Sensitivity analysis for the RANS equations coupled with linearized turbulence model. AIAA Paper 2007-76839, 2007.

[180] J. Renaud et G. Gabriele : Improved coordination in non-hierarchic system optimization. *AIAA Journal*, 31(12):2367–2373, December 1993.

[181] J. Reuther : *Aerodynamic shape optimization using control theory.* Thèse de doctorat, University of California Davis, 1996.

[182] J. Reuther, A. Jameson, J. Alonso, M. Rimlinger et D. Saunders : Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers, part I. *Journal of Aircraft*, 36(1):51–60, 1999.

[183] J. Reuther, A. Jameson, J. Alonso, M. Rimlinger et D. Saunders : Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers, part II. *Journal of Aircraft*, 36(1):61–74, 1999.

[184] J. Reuther, A. Jameson, J. Farmer, L. Martinelli et D. Saunders : Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation. AIAA Paper 96-0094, January 1996.

[185] P. Roe : Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of computational physics*, 43:357–372, 1981.

[186] F. Rosenblatt : The perceptron : a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[187] Y. Saad et M.H. Schultz : Gmres : A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific Statistical Computing*, 7(3):856–869, 1986.

[188] J. Sack, W. Welch, T. Mitchell et H. Wynn : Design and analysis of computer experiments. *Statistical science*, 4:409–435, 1989.

[189] J. Samareh : Multidisciplinary aerodynamic-structural shape optimization using deformation (MASSOUD). Long Beach, CA, September 2000. 8^{th} AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA Paper 2000-4911.

[190] D. Sasaki, S. Obayashi et K. Nakahashi : Navier-Stokes optimization of supersonic wings with four design objectives using evolutionary algorithm. AIAA Paper 2001-2531, 2001.

[191] M. Sasena : *Flexibility and efficiency enhancements for constrained global design optimization with kriging approximations.* Thèse de doctorat, University of Michigan, 2002.

[192] M. Schonlau : *Computer experiments and global optimization.* Thèse de doctorat, University of Waterloo, Waterloo, Canada, 1997.

[193] T. Sederberg et S. Parry : Free-form deformation of solid geometric models. *Computer Graphics*, 20(4), 1986.

[194] R. Seller, S. Batill et J. Renaud : Response surface based, concurrent subspace optimization for multidisciplinary system design. 34^{th} AIAA Aerospace Sciences Meeting, AIAA Paper 96-0714, January 1996.

[195] W. Send : Coupling of fluid and structure for transport aircraft wings. International forum on aeroelasticity and structural dynamics, CEAS/AIAA/ICASE/NASA Langley, Williamsburg, Virginia, 22-25 June 1999.

[196] S. Shahpar : *Design of Experiments and Response Surface Modelling to Minimise the Design Cycle Time.* VKI Course on Optimization Methods and Tools for Multigcriteria/Multidisciplinary Design, 2004.

[197] V. Shankar et H. Ide : Aeroelastic computations of flexible configurations. *Computational Structure*, 30:15–28, 1988.

[198] L. Sherman, A. Taylor III, L. Green, A. Newman, G. Hou et V. Korivi : First- and second-order aerodynamic sensitivity derivatives via automatic differentiation with incremental iterative methods. *Journal of computational Physics*, 129:307–331, 1996.

[199] G. Shubin : Obtaining "cheap" optimization gradients from computational aerodynamics codes. Boeing computer service. Applied mathematics and statistics technical report - AMS-TR-164, June 1991.

[200] G. Shubin : Application of alternative multidisciplinary optimization formulations to a model problem for static aeroelasticity. *Journal of Computational Physics*, (118):73–85, 1995.

[201] G. Shubin et P. Frank : A comparison of implicit gradient approach and the variational approach to aerodynamic design optimization. Boeing computer service. Applied mathematics and statistics technical report - AMS-TR-163, June 1991.

[202] T. Simpson, T. Mauery, J. Korte et F. Mistree : Kriging model for global approximation in simulation based multidisciplinary design optimization. *AIAA Journal*, 39:2233–2241, 2001.

[203] T. Simpson, J. Peplinsky, P. Koch et J. Allen : Metamodels for computer-based engineering design : survey and recommendation. *Engineering with computers*, 17:129–150, 2001.

[204] B. Smith : A near wall model for the $k-\omega$ two equation turbulence model. AIAA Paper 1994-2386, 1994.

[205] M. Smith, C. Cesnik et D. Hodges : An evaluation of computational algorithms to interface between CFD and CSD methodologies. AIAA Paper 1996-1400, 1996.

[206] M. Smith, D. Hodges et C. Cesnik : Evaluation of computational algorithms suitable for fluid-structure interactions. *Journal of Aircraft*, 37(2), 2000.

[207] J. Sobieszcsanski-Sobieski : The case for aerodynamic sensitivity analysis. NASA/VPI&SU Symposium on Sensitivity Analysis in Engineering, September 1986.

[208] J. Sobieszcsanski-Sobieski : Sensitivity analysis and multidisciplinary optimization for aircraft design : recent advances and results. *Journal of Aircraft*, 27(12), Deccember 1990.

[209] J. Sobieszcsanski-Sobieski : Sensitivity of complex, internally coupled systems. *AIAA Journal*, 28(1), 1990.

[210] J. Sobieszcsanski-Sobieski et R. Haftka : Multidisciplinary aerospace design optimization : Survey of recent developments. Reno, Nevada, January 1996. 34^{th} Aerospace Sciences Meeting and Exhibit, AIAA Paper 96-0711.

[211] G. Syswerda : Uniforme crossover in genetic algorithms. Proceedings of the Third European Conference on Genetic Algorithm, 1989.

[212] R. Thareja et R. Haftka : Numerical difficulties associated with using equality constraints to achieve multi-level decomposition in structural optimization. AIAA Paper 86-0854, 1986.

[213] I. Trelea : L'essaim de particules vu comme un système dynamique : convergence et choix des paramètres. Séminaire "L'optimisation par essaim de particules (OEP)", Paris, 2003.

[214] W. van Beers et J. Kleijnen : Kriging interpolation in simulation : a survey. Proceedings of the 2004 Winter Simulation Conference, 2004.

[215] G. Vanderplaats : Approximation concepts for numerical airfoil optimization. Rapport technique, NASA, 1979.

[216] G. Vanderplaats : *Numerical Optimization Techniques for Engineering Design : with Applications.* McGraw-Hill, 1984.

[217] G. Vanderplaats et R. Hicks : Numerical airfoil optimization using a reduced number of design coordinates. Rapport technique, NASA, 1976.

[218] H. Viviand et J.P. Veuillot : Méthodes pseudo-instationnaires pour le calcul d'écoulements transsoniques. Publication ONERA, 1978.

[219] M. Vrahatis et K. Parsopoulos : Parameter selection and adaptation in unified particle swarm optimization. *Mathematical and computer modelling*, 46, 2007.

[220] S. Wakayama : *Lifting Surface Design Using Multidisciplinary Optimization.* Thèse de doctorat, Stanford University CA, December 1994.

[221] S. Wakayama : Multidisciplinary design optimization of the blended-wing-body. St. Louis, MO, September 1998. 7^{th} AIAA/USAF/NASA/I-SMO Symposium on Multidisciplinary Analysis and Optimization, AIAA Paper 98-39885.

[222] S. Wakayama et I. Kroo : A method for lifting surface design using nonlinear optimization. AIAA Paper 90-3290, September 1990.

[223] S. Wakayama et I. Kroo : Subsonic wing planform design using multidisciplinary optimization. *Journal of Aircraft*, 32(4):746–753, July-August 1995.

[224] L. Wang et R. Grandhi : Multivariate hermite approximation for design optimization. *International journal for numerical methods in engineering*, 39:787–803, 1996.

[225] B. Widrow : *Generalization and information storage in networks of Adaline "Neurons"*, pages 435–461. Self-Organizing Systems, Spartan Books, Washington, DC, 1962.

[226] Y. Xiong, M. Moscinski, M. Frontera, S. Yin, M. Dede et M. Paradis : Multidisciplinary design optimization of aircraft combustor structure : An industry application. *AIAA Journal*, 43(9):2008–2014, September 2005.

[227] W. Yamazaki, S. Mouton et G. Carrier : Efficient design optimization by physics-based direct manipulation free-form deformation. Victoria, BC, Canada, September 2008. 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA Paper 2008-5953.

[228] Q. Zhang et H. Mülhenbein : On the convergence of a class of estimation of distribution algorithms. *IEEE Transactions on evolutionary computation*, 8(2), April 2004.

[229] G. Zoutendijk : *Methods of Feasible Directions.* Elsevier Publishing Company, Amsterdam, Netherlands, 1960.

ETUDE ET MISE EN ŒUVRE D'UNE METHODE D'OPTIMISATION DE FORME COUPLANT SIMULATION NUMERIQUE EN AERODYNAMIQUE ET EN CALCUL DE STRUCTURE

RESUME: L'objet de ce travail a consisté en l'étude et la mise en œuvre d'une méthode de calcul des dérivées des fonctions d'intérêt d'un problème d'optimisation de forme par rapport aux paramètres géométriques décrivant la forme solide, appelées de façon générique ''gradients'', pour un système aéroélastique soumis à un écoulement lointain stationnaire. Une méthodologie de calcul de l'équilibre aéroélastique statique a tout d'abord été développée dans le cadre de laquelle le comportement du fluide peut être modélisé par les équations d'Euler ou de Navier-Stokes moyennées (RANS), résolues numériquement par elsA, code de simulation numérique pour la mécanique des fluides développé à l'ONERA, et le comportement de la structure est prédit par la théorie des poutres (équations d'Euler-Bernoulli écrites en formulation matrice de flexibilité). Un cadre de calcul des gradients relatifs à ce système a ensuite pu être mis en place. Ceux-ci sont calculés de manière analytique en utilisant, soit la méthode de l'équation linéarisée discrète, soit la méthode du vecteur adjoint discret. Ces méthodes impliquent la résolution de systèmes linéaires couplés effectuée, dans le cadre de cette étude, par un processus itératif doublement retardé. Enfin, ces développements ont été appliqués au calcul des gradients des coefficients aérodynamiques de traînée et de portance par rapport à un ensemble de paramètres de forme pour trois configurations de complexité croissante, et validés par comparaison aux valeurs des gradients calculés par différences finies. Une dernière partie de ce travail a été consacrée à l'évaluation des performances de quatre modèles réduits non physiques dans le cadre d'un processus d'optimisation de forme d'une configuration bidimensionnelle de turbomachine.

Mots-clés: *optimisation de forme, aéroélasticité, paramètre de forme, gradient, méthode de l'équation linéarisée discrète, méthode du vecteur adjoint discret, modèle réduit*

SHAPE OPTIMIZATION OF A FLUID-STRUCTURE SYSTEM

ABSTRACT: This work is mainly dedicated to the sensitivity analysis of a static aeroelastic system with respect to design parameters governing its jig-shape, in order to proceed to its shape optimization. First, a framework able to predict static aeroelastic equilibrium has been set up. The fluid behavior can be governed either by the nonlinear Euler equations or by the nonlinear averaged Navier-Stokes equations (RANS), which are numerically solved by the ONERA CFD solver elsA. The structural behavior is governed by the Euler-Bernoulli equations (beam theory) formulated using the flexibility matrix approach. Then, a framework computing the sensitivities of the shape optimization functions of interest to the set of design parameters related to the jig-shape of the aeroelastic system previously depicted, has been raised. These gradients can be evaluated either by the discrete direct differentiation method or by the discrete adjoint vector method. In both cases, a coupled linear system has to be solved, which is carried out using a doubly lagged iterative process. Finally, this framework has been applied to the computation of the drag and lift aerodynamic coefficient sensitivities with respect to different shape parameters for three configurations of growing complexity and validated by comparison with the finite difference gradient values. A last part of this work has been dedicated to the evaluation of the performance of four surrogate models within the shape optimization of a bidimensional turbomachinery configuration.

Keywords: *shape optimization, aeroelasticity, design parameter, gradient, discrete direct differentiation method, discrete adjoint vector method, surrogate model*

ARTS ET MÉTIERS ParisTech