



**HAL**  
open science

# Solveurs de Krylov robustes pour la résolution partitionnée et monolithique du système adjoint couplé aéro-structure

Mehdi Jadoui

► **To cite this version:**

Mehdi Jadoui. Solveurs de Krylov robustes pour la résolution partitionnée et monolithique du système adjoint couplé aéro-structure. Analyse numérique [math.NA]. Sorbonne Université, 2023. Français. NNT : 2023SORUS350 . tel-04336964

**HAL Id: tel-04336964**

**<https://theses.hal.science/tel-04336964>**

Submitted on 12 Dec 2023

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

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

# Thèse de Doctorat de Sorbonne université

Spécialité : Mathématiques Appliquées

École doctorale de Sciences Mathématiques de Paris Centre 386

*Laboratoire Jacques-Louis Lions*

présentée par

**Mehdi Jadoui**

pour obtenir le grade de :

Docteur de Sorbonne Université

Sujet de thèse :

## **Solveurs de Krylov robustes pour la résolution partitionnée et monolithique du système adjoint couplé aéro-structure**

Thèse présentée et soutenue à Châtillon, le 16 novembre 2023, devant le jury composé de :

M. Luc GIRAUD	Directeur de recherche INRIA Bordeaux	Rapporteur
M. Michel VISONNEAU	Directeur de recherche CNRS	Rapporteur
M. Gilbert ROGE	Ingénieur de recherche Dassault Aviation	Président du jury
M. Pierre JOLIVET	Chargé de recherche CNRS	Examineur
M. Christophe BLONDEAU	Ingénieur de recherche ONERA	Examineur
M. François-Xavier ROUX	Professeur LJLL	Directeur de thèse



# Remerciements

Au moment où j'écris ces mots, je me trouve à la croisée des chemins entre le passé et l'avenir, et il est essentiel de prendre un moment pour exprimer ma gratitude à ceux qui m'ont accompagné tout au long de ce parcours académique.

Tout d'abord, je tiens à remercier l'ONERA de m'avoir donné cette opportunité de découvrir le travail de recherche qui représente à mes yeux, je dirais, une étape importante et décisive dans la vie d'un jeune chercheur. Je témoigne également de la qualité exceptionnelle de l'encadrement scientifique adopté à l'ONERA.

Ensuite, j'adresse toute ma reconnaissance et ma gratitude à mes encadrants de thèse Christophe Blondeau et François-Xavier Roux. La qualité de ce travail de thèse n'aurait pas atteint ce niveau sans leur rigueur, leur approche scientifique et leur créativité. Leur excellente qualité scientifique m'a permis surtout de faire mûrir mon esprit intellectuel qui est une étape nécessaire pour n'importe quel chercheur souhaitant viser l'excellence. Je remercie également l'équipe MSAE pour leur accueil chaleureux, leur sympathie et leur bonne humeur.

Je voudrais remercier très chaleureusement les rapporteurs de ce travail de thèse, M. Giraud et M. Visonneau. Leur remarque et leur analyse scientifique sur le sujet m'ont permis de prendre du recul scientifique et d'ouvrir des perspectives d'amélioration sur mes travaux.

Enfin, je ne serai jamais assez reconnaissant envers mes parents et plus particulièrement ma mère qui a su, par sa bienveillance et sa personnalité, me remonter le moral durant ce travail de thèse. Je remercie chaque membre de ma famille qui a pu contribuer de près ou de loin à ma réussite. Je pense notamment à mon très cher ami Rami qui a joué un rôle très important pendant les périodes difficiles. Je ne saurai comment le remercier.



## Abstract

Gradient-based multidisciplinary optimization is a technique widely used in the aeronautical industry and more particularly in the context of flexible wing design. There are basically two approaches for computing the gradient, namely the tangent approach and the adjoint approach. For a shape optimization of a flexible structure, there are generally few functions of interest and many design parameters. It is therefore appropriate to use the adjoint method. We are interested in improving numerical techniques for solving the fluid-structure coupled adjoint problem. In this work, we consider the partitioned approach on the one hand and the monolithic approach on the other hand. The first approach has the advantage of alternatively solving the fluid and structure problem by reusing existing disciplinary solvers. However, it suffers from numerical limitations when it comes to solving stiff coupled problems and can diverge in the case of strong aeroelastic coupling. The monolithic approach consists in solving the coupled adjoint system in a global way and makes it possible to immediately eliminate the limitations of the partitioned approach. However, it requires the implementation of robust preconditioning strategies typically based on an approximate Schur complement and also an appropriate scaling of the coupled system. This dissertation consists of three parts. First of all, we improved the solution of the pure fluid adjoint problem by proposing robust preconditioning strategies coupled with nested Krylov solvers. In a second step, the convergence of the partitioned approach was improved by introducing recycling techniques of approximate invariant subspace between fluid-structure cycles. Finally, a coupled monolithic adjoint solver was developed and compared to the partitioned approach. We demonstrate the efficiency of these solvers on a realistic case of aeroelastic adjoint calculation for the ONERA M6 wing in RANS fluid flow by considering a fully linearized turbulence model. The numerical results have indeed shown on the one hand the interest of recycling within the framework of a partitioned resolution and on the other hand the interest of the monolithic solution of stiff numerical problems.



## Résumé

L'optimisation multidisciplinaire à base de gradient est une technique largement utilisée dans l'industrie aéronautique et plus particulièrement dans le cadre du dimensionnement de voilures flexibles. Il existe essentiellement deux approches pour le calcul du gradient, à savoir l'approche tangente et l'approche adjointe. Pour une optimisation de forme d'une structure flexible, il y a en général peu de fonctions d'intérêt et de nombreux paramètres de conception. Il est donc adapté d'utiliser la méthode adjointe. Nous nous intéressons à l'amélioration des techniques numériques de résolution du problème adjoint couplé fluide-structure. Dans ce travail, nous considérons l'approche partitionnée d'une part et l'approche monolithique d'autre part. La première approche a l'avantage de résoudre alternativement le problème fluide et structure en réutilisant des solveurs disciplinaires existants. Elle souffre cependant de limitations numériques dès qu'il s'agit de résoudre des problèmes couplés raides et peut diverger dans le cas de fort couplage aéroélastique. L'approche monolithique consiste à résoudre le système adjoint couplé de manière globale et permet d'éliminer les limitations de l'approche partitionnée. En contrepartie elle requiert la mise en place de stratégies de préconditionnement robustes typiquement basées sur un complément de Schur approché et également une mise à l'échelle appropriée du système couplé. Ce mémoire de thèse se compose de trois parties. Tout d'abord nous avons amélioré la résolution du problème adjoint purement fluide en proposant des stratégies de préconditionnement robustes couplées à des solveurs de Krylov emboîtés. Dans un deuxième temps, la convergence de l'approche partitionnée a été améliorée en introduisant des techniques de recyclage d'espaces invariants approchés entre cycles fluide-structures. Enfin, un solveur monolithique adjoint couplé a été développé et comparé à l'approche partitionnée. Nous démontrons l'efficacité de ces solveurs sur un cas réaliste de calcul adjoint aéroélastique pour l'aile M6 ONERA en écoulement fluide RANS en considérant un modèle de turbulence entièrement linéarisé. Les résultats numériques ont effectivement montré d'une part l'intérêt du recyclage dans le cadre d'une résolution partitionnée et d'autre part l'intérêt de la résolution monolithique des problèmes numériques raides.





# Table des figures

2.1	Organisation du module aéroélastique. . . . .	27
2.2	Illustration du manque de robustesse du solveur de Newton inexact pour le problème adjoint. À droite, restauration de la convergence par ajout de dissipation artificielle au second membre. . . . .	35
2.3	Convergence des solveurs FGMRES et BFGMRES avec et sans déflation pour différentes valeurs de $m$ (Figures extraites de [9]). . . . .	38
2.4	Historique de convergence de la norme du résidu relative pour deux fonctions d'intérêt. Mode séquentiel (à gauche) vs mode par bloc (à droite). . . . .	39
2.5	Historique de convergence de la norme du résidu pour 10 seconds membres. 50 vecteurs de Krylov. Trait épais : résidu global. Trait fin : résidu partiel de chaque second membre (Figure extraite de [27]). . . . .	40
2.6	Comparaison des deux approches pour la résolution du système couplé adjoint (Figure extraite de [7]). . . . .	43
2.7	Historique de convergence de la norme du résidu relatif pour le solveur monolithique GCROT (Figure extraite de [4]). . . . .	45
2.8	Performance des deux approches (partitionnée et monolithique) en fonction des valeurs de $q_\infty/E$ et pour $M_\infty = 0.785$ (Figure extraite de [4]). . . . .	47
2.9	Historique de convergence de la norme du résidu fluide pour l'approche partitionnée. GMRES emboîté préconditionné par un LUSGS et impact du CFL sur la convergence du système adjoint. . . . .	48
3.1	Échange d'incrément de la solution entre le Bloc 1 et le Bloc 2. . . . .	55
3.2	Stencils des différentes matrices Jacobiennes. . . . .	59
3.3	Structure par bloc de la Jacobienne d'ordre 1 approchée $\mathbf{J}_{O1}^{APP}$ en 2D. . . . .	60
3.4	Structure par bloc de la matrice Jacobienne d'ordre 1 exacte $\mathbf{J}_{O1}^{EXA}$ en 2D . . . . .	60
3.5	Profil OAT15A pour un écoulement fluide transsonique et turbulent : (a) illustre le maillage structuré découpé en 16 blocs tandis que (b) illustre la densité volumique à l'état stationnaire. . . . .	65
3.6	Courbe de convergence de la densité et de l'énergie cinétique turbulente. . . . .	66
3.7	Solution adjointe pour la variable densité. . . . .	66

3.8	Historique de convergence du résidu en fonction du nombre de produits matrices-vecteurs. L'impact de la déflation améliore drastiquement la convergence pour le LU-SGS et le BILU(0) appliqués à $\mathbf{J}_{O1}^{APP}$ . L'effet du préconditionneur BILU(0) appliqué à $\mathbf{J}_{O1}^{APP}$ améliore également la robustesse du solveur. . . . .	67
3.9	Impact de la réorthogonalisation sur la convergence du GMRES-DR. (a) courbe de convergence pour le BILU(0) et (b) courbe de convergence pour le LUSGS. . . . .	67
3.10	Courbe de scalabilité du FGMRES-DR(60,20,20) sur le système adjoint du profil OAT15A. . . . .	70
3.11	Vue tridimensionnelle de l'aile M6 ONERA pour un écoulement fluide transsonique turbulent : (a) et (b) représentent respectivement la peau de l'aile pour un maillage structuré et le champ de pression à l'état stationnaire. . . . .	70
3.12	Historique de convergence de la norme du résidu projeté (least-sq. res) et du résidu exact (true res.). Impact de la double orthogonalisation de Gram-Schmidt sur le comportement du GMRES-DR. Nous rappelons les paramètres numériques du GMRES-DR : $m = 140$ et $k = 46$ . . . . .	71
3.13	Historique de convergence de la norme relative du résidu en nombre d'itérations. Impact de la déflation pour les différents préconditionneurs. L'augmentation de la dominance diagonale du préconditionneur LUSGS en ajoutant une diagonale scalaire $I/\Delta\tau$ semble être prometteur mais reste moins performante que le BILU. L'impact de la déflation n'est pas concluant pour le BILU appliqué à la jacobienne d'ordre 1 exacte puisqu'un seul cycle est effectué. Nous rappelons les paramètres numériques pour le FGMRES-DR : $m = 60$ , $m_i = 20$ et $k = 20$ . . . . .	72
3.14	Impact de la déflation pour différentes tailles de l'espace de Krylov interne. Historique de convergence de la norme relative du résidu en nombre d'itérations (a) et en nombre de produits matrices-vecteurs (b) pour le solveur FGMRES(60, $m_i$ ) et FGMRES-DR(60, $m_i$ ,20) avec $m_i = 6, 10, 15, 20$ . . . . .	73
3.15	Courbe de scalabilité du FGMRES-DR(60,20,20) sur le système adjoint aile M6. . . . .	73
4.1	Contours du coefficient de pression pour les écoulements stationnaires rigides et aéroélastiques. . . . .	98
4.2	Modèle aéroélastique de l'aile M6 : maillage RANS CFD à 42 blocs et schéma interne FEM. . . . .	98
4.3	Comparaison des tracés de section des coefficients de pression rigides et aéroélastiques à $y=0.60\text{m}$ et $y=1.12\text{m}$ . . . . .	99
4.4	Distribution du déplacement vertical et de l'incrément de torsion à l'équilibre aéroélastique. . . . .	99

4.5	Historique de convergence de la norme du résidu relative. Équivalence algébrique et numérique des solveurs GMRES-DR et GCRO-DR préconditionnés par un BILU(0) appliqué à jacobienne exacte d'ordre 1. Nous rappelons les paramètres numériques : $m = 120$ , et $k = 40$ . . . . .	100
4.6	Historique de convergence de la norme du résidu relative. L'équivalence algébrique et numérique des solveurs FGMRES-DR et FGCRO-DR n'est pas systématiquement vérifiée. Nous rappelons les paramètres numériques : $m = 60$ , $m_i = 20$ et $k = 20$ . . . . .	101
4.7	Historique de convergence de la norme du résidu relative. Comparaison des trois stratégies de déflation. . . . .	102
4.8	Historique de convergence de la norme du résidu relative pour le solveur GMRES-DR(120,40). Impact des différents préconditionneurs. Dans (a), une certaine stagnation du résidu exact se produit en raison de la propagation des erreurs d'arrondi pendant le processus de redémarrage. Dans (b), un redémarrage sans déflation permet d'éliminer cette stagnation au prix d'un coût de calcul plus important. . . . .	102
4.9	Historique de convergence de la norme du résidu relative pour le solveur GCRO-DR(120,40). Impact des différents préconditionneurs. Dans (a), une convergence erratique du résidu des moindres carrés associée à une stagnation du résidu exact se produit dans le cas du BILU(0) appliquée à la matrice jacobienne approchée du premier ordre. Dans (b), un redémarrage sans déflation permet d'éliminer cette stagnation au prix d'un coût de calcul plus important. . . . .	103
4.10	Historique de convergence du résidu relative des solveurs GCRO-DR(60,20) et LGCRO-DR(30,10, $p$ ) pour $p = 1$ et $p = 2$ . Impact du recyclage des approximations de l'erreur sur la convergence du système adjoint fluide. . .	104
4.11	Angles séquentiels et de saut pour le solveur GCRO-DR(60,20). . . . .	104
4.12	Angles séquentiels et de saut pour le solveur GCRO-DR(30,10). . . . .	105
4.13	Angles séquentiels et de saut pour le solveur GCRO-DR(30,9,1). . . . .	105
4.14	Angles séquentiels et de saut pour le solveur GCRO-DR(30,8,2). . . . .	106
4.15	Historique de convergence de la norme du résidu relative. Comparaison des performances des préconditionneurs en partitionné. Nous rappelons les paramètres numériques : $m = 60$ , $m_i = 20$ et $k = 20$ . . . . .	108
4.16	Effet du recyclage inter-système du sous-espace invariant approché de l'opérateur $\mathbf{A}$ sur la résolution partitionnée GCRO-DR(140,42). . . . .	108
4.17	Effet du recyclage inter-système du sous-espace invariant approché de l'opérateur $\mathbf{A}$ sur la résolution partitionné FGCRO-DR(70,10,35). Les vecteurs de Ritz sont calculés selon la stratégie A exploitant l'espace $\text{range}(\mathbf{Z}_m)$ . . .	109

4.18	Effet du recyclage inter-système du sous-espace invariant approché de l'opérateur $\mathbf{A}$ sur la résolution partitionnée FGCRO-DR(70,10,35). Les vecteurs de Ritz sont calculés selon la stratégie B exploitant l'espace $\text{range}(\mathbf{V}_m)$ . . .	109
4.19	Effet du recyclage inter-système du sous-espace invariant approché de l'opérateur $\mathbf{A}$ sur la résolution partitionnée FGCRO-DR(70,10,35). Les vecteurs de Ritz sont calculés selon la stratégie C exploitant l'espace $\text{range}(\mathbf{W}_m)$ . . .	110
5.1	Répartition du temps de calcul pour les préconditionneurs bloc Jacobi et bloc Gauss–Seidel ( $\mathbf{M}_A^{-1}$ ) pour différentes tolérances. Les deux dernières lignes décrivent le nombre total et le nombre maximum d'itérations FGMRES lors du processus de Newton–Krylov. . . . .	116
5.2	Factorisation LU de la diagonale par bloc. . . . .	116
5.3	Courbes de convergence du solveur GMRES-DR(120,40) monolithique et partitionné. Cas monolithique : Application d'un préconditionneur diagonal par bloc dont le deuxième bloc diagonal est représenté par l'opérateur $\mathbf{I} - [\mathbf{D}]^T \mathbf{S}^T$ . Application d'une mise à l'échelle pour le cas monolithique (courbe rose). Cas partitionné : Application de la stratégie de recyclage dès le deuxième couplage fluide-structure (courbe orange). . . . .	119

# Liste des tableaux

2.1	Performance des méthodes de Krylov emboîtées pour la résolution du système adjoint (Figure reproduite à partir de [9]). . . . .	38
2.2	Performances du GMRES emboîté. Tolérance du résidu = $10^{-4}$ . . . . .	39
2.3	Paramètres caractéristiques du CRM (Table extraite de [7]). . . . .	42
2.4	Série de valeurs de $q_{\infty}/E$ pour l'analyse de performance (Figure extraite de [4]). . . . .	46
2.5	Comparaison des performances du solveur monolithique et partitionné pour des valeurs critiques de $q_{\infty}/E$ (Figure extraite de [4]). . . . .	46
3.1	Paramètres numériques du solveur FGMRES-DR . . . . .	65
3.2	Performance de FGMRES-DR . . . . .	69
3.3	Impact du nombre de sous-domaines sur le nombre d'itérations pour le solveur FMGRES-DR(60,20,20) et selon le type du préconditionneur. . . . .	74
4.1	Performances du solveur partitionné en fonction des stratégies de déflation. Les résultats sont rapportés en nombre de produits par $\mathbf{A}$ . . . . .	107



# Table des matières

<b>1</b>	<b>Introduction</b>	<b>19</b>
	<b>Introduction</b>	<b>23</b>
<b>2</b>	<b>Etat de l'art des solveurs adjoints couplés fluide-structures</b>	<b>25</b>
2.1	Le problème aéroélastique stationnaire . . . . .	27
2.2	Méthodes numériques et calcul du gradient . . . . .	28
2.2.1	Description du problème adjoint discret aérodynamique . . . . .	29
2.2.1.1	Méthode des différences finies et Complex Step . . . . .	30
2.2.1.2	Calcul du gradient par la méthode de l'équation linéarisée discrète . . . . .	31
2.2.1.3	Calcul du gradient par la méthode adjointe discrète . . . . .	31
2.2.2	Approche partitionnée pour la résolution d'un système adjoint aéro- structure . . . . .	32
2.2.2.1	Équilibre aéroélastique . . . . .	32
2.2.3	Approche partitionnée pour le système adjoint couplé . . . . .	32
2.3	Motivation du développement des solveurs de Krylov . . . . .	34
2.4	Résolution d'un problème adjoint aérodynamique par méthode de Krylov . . . . .	35
2.4.1	Contribution des méthodes de Krylov par bloc . . . . .	35
2.4.1.1	Configuration NACA 64212 . . . . .	37
2.5	Résolution du problème adjoint aéro-structure par méthode de Krylov . . . . .	40
2.5.1	Approche monolithique et approche partitionnée . . . . .	41
<b>3</b>	<b>Développement de solveurs de Krylov emboîtés associés à des stratégies de préconditionnement robustes</b>	<b>49</b>
3.1	Méthode de projection - Méthode de Krylov . . . . .	51
3.1.1	La méthode généralisée du résidu minimal (GMRES) . . . . .	51
3.1.2	Procédure d'orthogonalisation - Méthode d'Arnoldi . . . . .	52
3.2	Stratégies de préconditionnement . . . . .	53
3.2.1	Méthode LU-SGS couplée à la méthode de Schwarz additive . . . . .	54
3.2.2	Factorisation LU incomplète . . . . .	56
3.2.3	La méthode GMRES flexible (FGMRES) . . . . .	56



3.2.4	Matrices de préconditionnement . . . . .	58
3.2.4.1	Matrice jacobienne approchée d'ordre un . . . . .	59
3.2.4.2	Matrice jacobienne exacte d'ordre un . . . . .	60
3.3	Approche de déflation spectrale . . . . .	60
3.3.1	Calcul des vecteurs de Ritz harmoniques . . . . .	61
3.3.1.1	Méthode de Rayleigh-Ritz . . . . .	61
3.3.1.2	Traitement numérique des valeurs propres . . . . .	63
3.3.2	Description de l'algorithme FGMRES-DR . . . . .	63
3.4	Résultats numériques . . . . .	64
3.4.1	Profil OAT15A . . . . .	65
3.4.1.1	Résolution du système adjoint . . . . .	65
3.4.1.2	Étude de la scalabilité forte . . . . .	69
3.4.2	Aile M6 ONERA . . . . .	69
3.4.2.1	Résolution du système adjoint . . . . .	69
3.4.2.2	Étude de la scalabilité forte . . . . .	72
<b>4</b>	<b>Stratégie de recyclage d'espace invariant approché pour le solveur parti-</b>	
	<b>tionné</b>	<b>75</b>
4.1	Historique des méthodes de recyclage . . . . .	77
4.1.1	Correction de rang 1 . . . . .	77
4.1.2	Méthode généralisée des résidus conjugués - GCR . . . . .	79
4.1.3	Méthode GMRES Réursive - GMRESR . . . . .	80
4.1.4	Stratégies de troncature pour les méthodes de Krylov - GCROT . . . . .	81
4.1.5	Méthode de recyclage GCRO-DR . . . . .	83
4.1.5.1	Calcul des espaces de correction et de recyclage dans le cadre du GCRO . . . . .	85
4.1.5.2	Description de l'algorithme GCRO-DR flexible . . . . .	86
4.1.6	Cadre général des méthodes de recyclage . . . . .	90
4.1.7	Loose GCRO-DR . . . . .	93
4.1.7.1	Cas 1 : Recyclage d'une seule approximation d'erreur . . . . .	95
4.1.7.2	Cas 2 : Recyclage de plusieurs approximations de l'erreur . . . . .	95
4.1.8	Loose FGCRO-DR . . . . .	96
4.1.9	Critère d'activation pour le recyclage d'espace invariant approché . . . . .	96
4.2	Résultats numériques . . . . .	98
4.2.1	Aile M6 Analyse aéroélastique . . . . .	98
4.2.2	Évaluation des performances dans le cas d'un système adjoint aéro- dynamique . . . . .	100
4.2.3	Effet de l'approximation des erreurs sur le système adjoint aérody- namique . . . . .	103

---

4.2.4	Évaluation des performances du solveur partitionné dans le cas d'un système adjoint couplé . . . . .	106
<b>5</b>	<b>Mise en place d'un solveur monolithique couplé à des stratégies de pré-conditionnement robustes</b>	<b>111</b>
5.1	Stratégies de préconditionnement dans le cadre d'un solveur monolithique . . . . .	112
5.1.1	Factorisation par bloc et complément de Schur . . . . .	112
5.1.2	Stratégie de mise à l'échelle . . . . .	115
5.2	Étude des différentes approximations du complément de Schur . . . . .	117
5.2.1	Premier cas de figure : . . . . .	117
5.2.2	Deuxième cas de figure : . . . . .	117
5.2.3	Troisième cas de figure . . . . .	117
5.3	Résultats numériques . . . . .	118
<b>A</b>	<b>Reformulation du problème aux valeurs propres</b>	<b>125</b>
A.1	Reformulation des problèmes aux valeurs propres . . . . .	125
A.2	Simplification du problème aux valeurs propres généralisé du FGCRO-DR . . . . .	126
<b>B</b>	<b>Algorithmes</b>	<b>127</b>



# Chapitre 1

## Introduction

L'un des enjeux majeurs dans la phase de conception d'un avion de transport civil est de répondre à des exigences environnementales de plus en plus drastiques, au titre desquelles figure la réduction du taux de  $CO_2$  émis dans l'atmosphère. La stratégie consiste à réduire la quantité de carburant par passager par trajet. Diverses approches de conception ont été mises en oeuvre afin de répondre à ces exigences. L'une d'elles consiste à développer des outils numériques permettant d'optimiser conjointement la forme et la rigidité de l'avion, aboutissant par exemple à des configurations d'ailes présentant de très grands allongements. De telles structures à la fois légères et résistantes nécessitent le recours à des matériaux de nouvelle génération et, à ce titre, l'emploi des matériaux composites s'est généralisé dans la conception aéronautique. Néanmoins, la flexibilité accrue des ailes composites de grand allongement est à l'origine d'interactions fortes entre le fluide et la structure. Les problèmes aéroélastiques que nous étudions dans ce travail de thèse tombent dans cette catégorie. Plus particulièrement, le problème linéarisé autour d'une configuration d'équilibre aéroélastique va générer des systèmes linéaires couplés qui seront généralement très raides et donc difficiles à résoudre par des approches numériques itératives standards.

Il existe deux grands types d'optimisation : l'optimisation globale et l'optimisation locale. Le premier consiste à rechercher la meilleure configuration dans tout le domaine formé par les paramètres de conception. En général, on fait appel à des algorithmes d'optimisation génétique associés traditionnellement à des surfaces de réponse. Cette stratégie peut s'avérer coûteuse dès que l'on considère des modèles numériques haute-fidélité. Le deuxième consiste à déterminer un optimum local en calculant les gradients des fonctions d'intérêt par rapport aux paramètres de conception. Ces gradients fournissent à l'optimiseur la direction de recherche vers cet optimum local. Dans un contexte d'optimisation multidisciplinaire haute-fidélité, l'optimisation locale à base de gradient paraît la meilleure approche puisqu'elle constitue un bon compromis entre coût de calcul et précision de l'optimum. La principale difficulté de cette approche est de calculer de manière efficace et précise le gradient de la fonction d'intérêt en question.

Il existe deux méthodes pour le calcul du gradient : la première méthode, appelée

méthode directe ou tangente, consiste à linéariser les équations d'équilibre autour d'une configuration stationnaire formant ainsi un système linéaire. La résolution de ce système permet d'obtenir le gradient de la fonction objectif par rapport aux paramètres de conception. Le coût de calcul du gradient est indexé sur le nombre de paramètres du système qui peut atteindre plusieurs centaines. Par conséquent, il est nécessaire de résoudre autant de systèmes linéaires qu'il y a de paramètres. Pour une optimisation haute-fidélité aéro-structure, on considère un nombre réduit de réponses du système et de nombreux paramètres de conception. Dans ce cas, il est préférable de calculer le gradient par la méthode adjointe. Elle permet de rendre le calcul du gradient indépendant du nombre de paramètres. Dans ce cas, le nombre de résolutions de systèmes linéaires est indexé sur le nombre de fonctions d'intérêt. Cette méthode a été introduite en 1988 par Jameson [1] dans le cadre d'une optimisation de forme aérodynamique.

Dans sa formulation directe ou adjointe, le calcul du gradient nécessite la résolution d'un système linéaire issu de la linéarisation des équations d'équilibre autour d'une configuration stationnaire. Résoudre de tels systèmes linéaires représente en soi un défi notamment à cause de leur très grande taille, leur structure creuse, non symétrique par blocs et de leur mauvais conditionnement. Par conséquent, il est indispensable de mettre en place des stratégies de résolution efficaces et robustes. Il existe principalement deux approches pour la résolution de systèmes couplés. L'approche partitionnée consiste à résoudre de manière approchée le problème fluide entre deux couplages avec le problème structure [2]. Cette approche se base essentiellement sur l'algorithme de Gauss-Seidel linéaire par blocs. Le problème structure est résolu le plus souvent par des méthodes directes. Le principal avantage de cette méthode est qu'elle ne demande pas un effort d'implémentation conséquent du fait de son caractère modulaire. Malgré cela, cette approche manque de robustesse dans le cas d'interactions fluide-structure fortes et peut même diverger. Il existe néanmoins des techniques qui permettent d'améliorer la stabilité de cette approche telles que les techniques de relaxation stationnaire ou adaptatives de type Aitken [3]. L'approche monolithique consiste à résoudre le système couplé de manière globale. Contrairement à l'approche partitionnée, elle permet de résoudre de manière robuste des systèmes présentant de forts couplages à condition de développer des stratégies adaptées de mise à l'échelle et de préconditionnement.

Pour ces deux approches, la résolution des systèmes linéaires adjoints se fait au moyen de méthodes itératives de type Krylov. Ce choix est justifié par le fait que ces méthodes sont robustes pour résoudre des systèmes linéaires qui n'ont pas de propriétés particulières exploitables. Elles possèdent également des propriétés de convergence remarquable par rapport aux méthodes itératives classiques de type point fixe. Néanmoins, les systèmes linéaires qui sont traités en mécanique des fluides sont généralement très mal conditionnés. Une phase de préconditionnement est donc indispensable pour que les méthodes de Krylov soient robustes. De manière générale, la mise en place d'un préconditionneur dépend fortement du problème physique. Il est donc primordial de prendre en compte les

---

caractéristiques physiques dans la construction du préconditionneur.

En raison de la robustesse de l'approche monolithique, certains auteurs ont concentré leurs efforts sur celle-ci afin d'améliorer la résolution des systèmes adjoints fortement couplés. À titre d'exemple, Zhang et Zingg [4] ont réalisé une étude comparative des approches partitionnée et monolithique pour diverses stratégies de préconditionnement. La résolution s'effectue au moyen de la méthode GCROT (Truncated Generalized Conjugate Residual with inner Orthogonalization) [5]. Cette dernière fait partie de la classe des méthodes de Krylov emboîtées (voir [6] pour une revue exhaustive). Dans le même esprit, Kenway et al. [7] comparent les deux approches en invoquant le solveur GMRES (Generalized Minimal Residual Method) [8]. Sachant que l'approche monolithique n'est autre qu'une méthode de Krylov appliquée directement au système linéaire global, il est crucial de mettre en place une stratégie de préconditionnement robuste permettant de résoudre de manière efficace ces systèmes adjoints fortement couplés. Dans les deux études, les auteurs emploient un préconditionneur diagonal par bloc et un préconditionneur triangulaire par bloc. Pour leurs cas d'application, ils concluent que l'approche monolithique est plus performante que l'approche partitionnée. Il faut cependant mentionner que, contrairement à la stratégie partitionnée, l'approche monolithique requiert l'application de l'opérateur de déformation de maillage à chaque itération. Il est donc crucial de disposer de techniques de déformation de maillage à la fois robustes, i.e., préservant la qualité du maillage et hautement parallélisées. Ces techniques doivent bien évidemment être disponibles en version linéarisée et adjointe.

Les travaux cités précédemment permettent de mettre en évidence l'intérêt de l'approche monolithique pour la résolution des problèmes adjoints aéro-structure par rapport à la version partitionnée. Néanmoins, aucune démarche n'a été proposée dans la littérature afin d'améliorer l'approche partitionnée. Des artifices numériques tels que la relaxation dynamique de Aitken [3] ont permis malgré tout d'améliorer le comportement de cette approche mais deviennent rapidement inefficace pour des problèmes numériques raides ou lorsque le profil de convergence des résidus n'est pas linéaire. En outre, les problèmes traités par les auteurs ne sont pas suffisamment raides pour mettre concrètement en valeur l'efficacité du solveur monolithique. En effet, les études citées se restreignent à des équations fluides non visqueuses d'Euler et la question de la transposition des conclusions à des équations de type RANS se pose. Pinel et Montagnac [9] ou encore Chen et al. [10] utilisent les équations RANS mais seulement pour des problèmes adjoints fluides. Historiquement, la viscosité turbulente a d'abord été systématiquement négligée lors de la linéarisation des équations de transport turbulentes [11]. L'avènement récent des techniques de différentiation automatique dans le développement des versions linéarisées et adjointes des grands codes de calcul a favorisé la linéarisation complète des équations turbulentes [12]. On parle alors de linéarisation exacte des équations RANS qui en pratique aboutit à des systèmes linéaires très mal conditionnés.

L'objectif de cette thèse est de contribuer à l'amélioration de la résolution des sys-

tèmes linéaires adjoints fortement couplés. Pour cela, notre démarche se décompose en trois grandes étapes. La première étape consiste à améliorer la résolution des systèmes adjoints purement fluides en mettant en place des stratégies de préconditionnement robustes couplées à des solveurs de Krylov emboîtés, c'est l'objet du Chapitre 1. Dans le Chapitre 2, nous chercherons à améliorer l'approche partitionnée d'une part en intégrant les contributions du Chapitre 1 et d'autre part en introduisant des techniques originales de recyclage d'espaces invariants approchés entre cycles fluide-structures. Cette idée de recyclage vient du fait que la résolution partitionnée est vue algébriquement comme la résolution d'une séquence de systèmes linéaires où le second membre varie à chaque cycle fluide-structure. Plusieurs auteurs ont contribué au développement de différentes stratégies qui ont permis d'améliorer la résolution de séquences de systèmes linéaires [13, 5, 14] mais, à notre connaissance, pas dans le cadre de stratégies partitionnées. Il est donc intéressant d'intégrer de telles stratégies dans le but d'accélérer la résolution partitionnée. Le troisième et dernier Chapitre a pour objectif de mettre en place un solveur monolithique couplé à des stratégies de préconditionnement robustes. Typiquement, nous étudierons un préconditionneur basé sur un complément de Schur approché. Ce dernier, bien que connu, n'a jamais fait l'objet d'une étude approfondie dans le contexte d'une résolution monolithique. Nous proposerons une approximation de cet opérateur afin de trouver le bon compromis entre coût et robustesse du préconditionneur. Nous illustrerons les avancées méthodologiques proposées dans cette thèse en termes de gain au niveau des résolutions partitionnée et monolithique appliquées à un cas test aéroélastique stationnaire tridimensionnel réaliste, i.e., l'aile M6 ONERA en écoulement transsonique en fluide visqueux RANS [15].

---

Ce travail a donné lieu à la publication de plusieurs articles.

### **Article de revue**

- M. Jadoui, C. Blondeau, E. Martin, F. Renac, F.-X. Roux. Comparative Study of Inner-Outer Krylov Solvers for Linear Systems in Structured and High-Order Unstructured CFD problems. *Computer and Fluids*, 15 August 2022, p. 105575-105587.

### **Actes de conférences**

- M.Jadoui, C. Blondeau, E. Martin, F. Renac, F.-X. Roux. Comparative Study of Inner-Outer Krylov Solvers for Linear Systems in Structured and High-Order Unstructured CFD problems. *14th WCCM ECCOMAS Virtual Congress*, 2021.

- M.Jadoui, C. Blondeau, F.-X. Roux. Recycling Krylov Subspaces for Efficient Partitioned Solution of Aerostructural Coupled Adjoint Systems. *AIAA Aviation, Chicago*, 2022.

### **Communication orale**

- M.Jadoui, C. Blondeau, E. Martin, F. Renac, F.-X. Roux. Comparative Study of Inner-Outer Krylov Solvers for Linear Systems in Structured and High-Order Unstructured CFD problems. *PARCFD2021 Virtual Congress*, 2021.





## Chapitre 2

# Etat de l'art des solveurs adjoints couplés fluide-structures

### Sommaire

---

<b>2.1</b>	<b>Le problème aéroélastique stationnaire</b>	<b>27</b>
<b>2.2</b>	<b>Méthodes numériques et calcul du gradient</b>	<b>28</b>
2.2.1	Description du problème adjoint discret aérodynamique	29
2.2.2	Approche partitionnée pour la résolution d'un système adjoint aéro-structure	32
2.2.3	Approche partitionnée pour le système adjoint couplé	32
<b>2.3</b>	<b>Motivation du développement des solveurs de Krylov</b>	<b>34</b>
<b>2.4</b>	<b>Résolution d'un problème adjoint aérodynamique par méthode de Krylov</b>	<b>35</b>
2.4.1	Contribution des méthodes de Krylov par bloc	35
<b>2.5</b>	<b>Résolution du problème adjoint aéro-structure par méthode de Krylov</b>	<b>40</b>
2.5.1	Approche monolithique et approche partitionnée	41

---

### Introduction

L'objectif de ce chapitre est de mettre en avant la contribution des méthodes de Krylov dans la résolution des systèmes linéaires adjoints. Tout d'abord, une description du problème aéroélastique standard est proposée dans la section 2.1. La section 2.2 propose un rappel de la méthode des volumes finies ainsi que des différentes méthodes de calcul des gradients. Ensuite, nous expliquons dans la section 2.3 l'intérêt des méthodes de Krylov pour la résolution des systèmes adjoints couplés. Enfin, la section 2.4 expose les différentes contributions des méthodes de Krylov pour la résolution d'un problème adjoint aérodynamique suivi de la section 2.5 qui décrit les contributions dans le cadre d'une résolution d'un problème adjoint couplé aéro-structure.

## 2.1 Le problème aéroélastique stationnaire

Cette section décrit la résolution classique partitionnée d'un problème couplé aéroélastique au sein du solveur aérodynamique *elsA* de l'ONERA [16]. La résolution d'un problème couplé aéroélastique consiste à calculer la position d'équilibre entre le fluide et la structure. Celle-ci utilise une méthode de point fixe qui alterne entre résolution des équations qui régissent l'écoulement fluide et la résolution des équations de la structure. La Figure 2.1 synthétise les étapes de la résolution partitionnée.

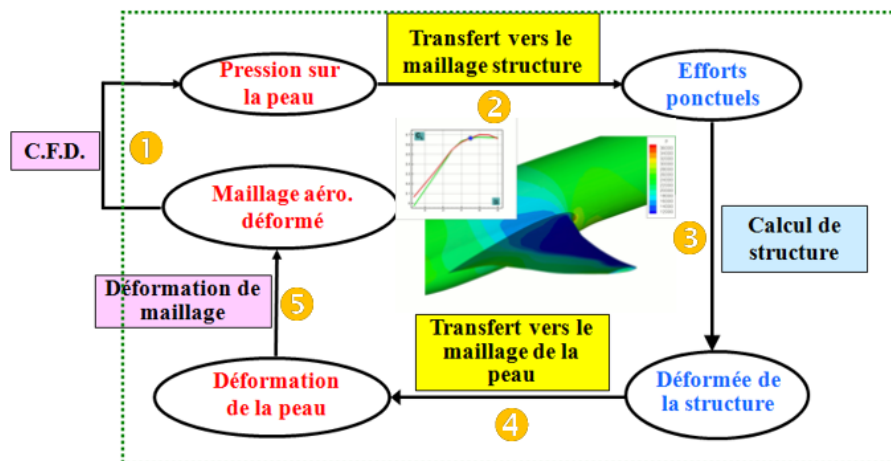


FIGURE 2.1 : Organisation du module aéroélastique.

La boucle de couplage statique entre fluide et structure comporte 5 étapes :

1. **Résolution du problème CFD** : les équations de Navier-Stokes ou d'Euler sont résolues sur le maillage fluide. Un nombre fixé d'itérations fluide est réalisé. Le problème fluide étant partiellement résolu, les efforts aérodynamiques sont déterminés sur l'interface aéroélastique.

2. **Transfert Fluide/Structure** : transfert des efforts aérodynamiques au maillage structure. Théoriquement, cette phase de transfert doit être conservative (au sens du respect des équations d'équilibre et de la conservation du travail au niveau de l'interface aéroélastique) car elle traduit le transfert d'énergie du fluide vers la structure.

3. **Résolution du problème CSM** : calcul des déplacements de la structure chargée avec les efforts aérodynamiques.

4. **Transfert Structure/Fluide** : connaissant la déformée de la structure, lissage de cette solution sur l'interface aéroélastique par des techniques de transfert de champ.

5. **Déformation de maillage** : étape de mise à jour du maillage fluide. Propagation de la déformation surfacique à l'ensemble des blocs du maillage aérodynamique par des

techniques de déformation de maillage.

## 2.2 Méthodes numériques et calcul du gradient

Cette section décrit les méthodes de discrétisation numériques des équations de Navier-Stokes. Plus précisément, nous utiliserons le modèle RANS (*Reynolds-Averaged Navier-Stokes*). Nous cherchons à calculer le champ des variables conservatives  $\mathbf{W}$  à l'état stationnaire dans le domaine  $\mathcal{D}$ . La solution stationnaire calculée dans le domaine  $\mathcal{D}$  s'exprime par la nullité du résidu non linéaire discrétisé :

$$\mathbf{R}_a(\mathbf{W}, \mathbf{X}) = \mathbf{0} \quad (2.1)$$

où  $\mathbf{X}$  représente le maillage aérodynamique. Pour calculer le champ  $\mathbf{W}$  à l'état stationnaire dans  $\mathcal{D}$ , nous allons discrétiser ce domaine par la méthode des Volumes Finis. Nous noterons ce domaine discrétisé par  $\mathcal{D}_\Omega$  qui sera constitué de cellules élémentaires  $\Omega$  de volume  $V_\Omega$ . Cette méthode est basée sur la discrétisation de la forme intégrale des équations de conservation.

Chaque cellule hexaédrique possède  $N_f = 6$  faces notées  $\Gamma_i$  avec  $i = 1, \dots, N_f$ . Le champ moyen des variables conservatives au centre d'une cellule  $\Omega$  est défini par :

$$\bar{\mathbf{W}} = \frac{1}{V_\Omega} \int_\Omega \mathbf{W} \, d\Omega \quad (2.2)$$

De même, on écrit la valeur moyenne du flux à travers  $\Gamma_i$  :

$$\mathbf{F}_{\Gamma_i}(\bar{\mathbf{W}}) = \int_{\Gamma_i} [\mathbf{F}_c(\bar{\mathbf{W}}) + \mathbf{F}_d(\bar{\mathbf{W}})] \mathbf{n} \, d\Gamma_i \quad (2.3)$$

où  $\mathbf{F}_c$  et  $\mathbf{F}_d$  représentent les flux discrets convectif et diffusif respectivement et  $\mathbf{n}$  le vecteur normal à la face  $\Gamma_i$ . On peut donc réécrire la discrétisation de la forme intégrale des équations de conservation sur les variables moyennées :

$$\frac{\partial}{\partial t} (V_\Omega \bar{\mathbf{W}}) + \sum_{i=1}^6 \mathbf{F}_{\Gamma_i}(\bar{\mathbf{W}}) = 0 \quad (2.4)$$

Pour finir, nous introduisons une approximation des valeurs et flux moyens de l'équation (2.4) :

$$\frac{\partial}{\partial t} (V_\Omega \mathbf{W}_\Omega) + \sum_{i=1}^6 \mathbf{F}_i(\mathbf{W}_\Omega, \mathbf{W}_{\Omega_i}) = 0 \quad (2.5)$$

L'équation (2.5) correspond à une formulation semi-discrète en espace, ce qui permet de découpler l'intégration spatiale et temporelle des équations de conservation.

Le terme  $\mathbf{F}_i(\mathbf{W}_\Omega, \mathbf{W}_{\Omega_i})$  correspond à une approximation numérique du flux physique

fonction de  $\mathbf{W}_\Omega$  et des approximations numériques  $\mathbf{W}_{\Omega_i}$  calculées dans un certain nombre de cellules voisines à chacune des interfaces  $\Gamma_i$ .

On introduit également  $\mathbf{R}_{a,\Omega}$  le résidu numérique discret de la modélisation qui correspond à un bilan de flux au centre de la cellule  $\Omega$ . Autrement dit,

$$\sum_{i=1}^6 \mathbf{F}_i(\mathbf{W}_\Omega, \mathbf{W}_{\Omega_i}) = \mathbf{R}_{a,\Omega} \quad (2.6)$$

À l'état stationnaire (i.e.  $\partial/\partial t = 0$ ), on aboutit à :

$$\mathbf{R}_{a,\Omega}(\mathbf{W}_\Omega, \mathbf{X}_\Omega) = 0 \quad (2.7)$$

Pour le calcul du champ stationnaire  $\mathbf{W}_\Omega$ , on adopte une approche pseudo-instationnaire en maillage fixe ( $\partial V_\Omega/\partial t = 0$ ) en introduisant le terme  $V_\Omega (\partial \mathbf{W}_\Omega/\partial \tau)$  dans l'équation (2.7). Finalement, nous obtenons :

$$V_\Omega \frac{\partial \mathbf{W}_\Omega}{\partial \tau} = -\mathbf{R}_{a,\Omega} \quad (2.8)$$

On fait appel au schéma implicite Backward-Euler pour la discrétisation temporelle. Si on note  $\Delta\tau$  le pseudo de pas de temps local et  $n \in \mathbb{N}$  l'itération en temps, la discrétisation de (2.8) entre les instants  $\tau^n$  et  $\tau^{n+1} = \tau^n + \Delta\tau$  donne :

$$V_\Omega = \frac{\mathbf{W}_\Omega^{n+1} - \mathbf{W}_\Omega^n}{\Delta\tau} = -\mathbf{R}_{a,\Omega}^{n+1} \quad (2.9)$$

Une linéarisation à l'ordre 1 du résidu discret autour du champ  $\mathbf{W}_\Omega^n$  donne :

$$\mathbf{R}_{a,\Omega}^{n+1} \simeq \mathbf{R}_{a,\Omega}^n + \frac{\partial \mathbf{R}_{a,\Omega}}{\partial \mathbf{W}_\Omega} \Delta \mathbf{W}_\Omega^{n+1} \quad (2.10)$$

où  $\Delta \mathbf{W}_\Omega^{n+1} = \mathbf{W}_\Omega^{n+1} - \mathbf{W}_\Omega^n$ . La matrice  $\frac{\partial \mathbf{R}_{a,\Omega}}{\partial \mathbf{W}_\Omega}$  constitue la matrice jacobienne des flux. En pratique, on introduit dans la phase implicite une approximation de la jacobienne exacte plus facile à inverser.

Finalement, le schéma Backward-Euler s'écrit :

$$\left( \frac{V_\Omega}{\Delta\tau} \mathbf{I} + \frac{\partial \mathbf{R}_{a,\Omega}^{app}}{\partial \mathbf{W}_\Omega} \right) (\mathbf{W}_\Omega^{n+1} - \mathbf{W}_\Omega^n) = -\mathbf{R}_{a,\Omega}, \quad (2.11)$$

où  $\mathbf{I}$  est la matrice identité de taille  $N$ . Pour la résolution du système linéaire (2.11), on fait appel à la méthode de relaxation LU-SSOR couplée à une méthode Restrictive Additive Schwarz (RAS) [17].

### 2.2.1 Description du problème adjoint discret aérodynamique

Considérons le problème d'optimisation aérodynamique suivant :

$$\mathcal{J}(\alpha) = \min_{\alpha \in \mathbb{R}^{n_\alpha}} J(\mathbf{W}(\alpha), \mathbf{X}(\alpha)) \quad (2.12)$$

où  $\mathbf{W}$  correspond aux variables conservatives,  $\mathbf{X}$  aux coordonnées du maillage fluide et  $\alpha$  aux paramètres d'optimisation contrôlant la forme aérodynamique dont la taille est  $n_\alpha$ . La fonction objectif  $\mathcal{J}$  correspond en général soit à la traînée, soit à la finesse (rapport entre la portance et la traînée) d'un avion. La résolution de (2.12) s'effectue sous la contrainte  $\mathbf{R}_a(\mathbf{W}, \mathbf{X}) = \mathbf{0}$ , où  $\mathbf{R}_a$  correspond au résidu discret aérodynamique.

La résolution du problème (2.12) est basée principalement sur la méthode des gradients dont le calcul s'effectue selon plusieurs méthodes. Il existe deux façons de traiter un calcul de gradient. La première consiste d'abord à discrétiser les équations de la mécanique des fluides par une méthode de discrétisation (Volumes Finis, Éléments Finis, Galerkin Discontinue) puis de linéariser ces équations autour d'un état stationnaire. C'est la forme discrète de la méthode. La forme continue effectue le processus inverse, c'est-à-dire linéariser les équations fluides puis discrétiser l'équation linéarisée. Les sections suivantes présentent les différentes méthodes de calcul du gradient.

### 2.2.1.1 Méthode des différences finies et Complex Step

Historiquement, le calcul du gradient s'effectuait par la méthode des différences finies. Même si cette dernière est facile à implémenter, elle reste tout de même une méthode très coûteuse car le calcul à l'ordre 1 du gradient  $\frac{dJ}{d\alpha}$  nécessite  $n_\alpha$  résolutions fluides et  $(2n_\alpha + 1)$  résolutions fluides pour le calcul du gradient à l'ordre 2.

Soit  $\delta\alpha_i$  l'incrément lié à la méthode des différences finies pour le paramètre de forme  $\alpha_i$  où  $i \in \llbracket 1, n_\alpha \rrbracket$ . Soient les champs aérodynamiques  $\mathbf{W}(\alpha)$  et  $\mathbf{W}(\alpha + \delta\alpha_i)$  qui correspondent aux maillages respectifs  $\mathbf{X}(\alpha)$  et  $\mathbf{X}(\alpha + \delta\alpha_i)$ . Le calcul de gradient est donné par la formule suivante :

$$\frac{d\mathcal{J}}{d\alpha_i} \simeq \frac{J(\mathbf{W}(\alpha + \delta\alpha_i), \mathbf{X}(\alpha + \delta\alpha_i)) - J(\mathbf{W}(\alpha), \mathbf{X}(\alpha))}{\delta\alpha_i} \quad (2.13)$$

Pour certaines applications, la valeur du gradient peut être très sensible à l'amplitude de la perturbation  $\delta\alpha_i$  : si ce dernier est trop grand, la précision du gradient peut être mauvaise, et s'il est trop faible, les valeurs obtenues peuvent être bruitées.

Il existe une autre méthode de calcul des gradients qui est la méthode de la *complex-step* qui repose sur l'utilisation de variables complexes pour estimer les dérivées. L'avantage par rapport aux différences finies est que cette méthode est robuste en termes de troncature numérique, ce qui la rend plus précise. Cependant, elle nécessite d'avoir un code de calcul en arithmétique complexe, ce qui n'est pas notre cas dans le cadre de cette thèse. Une analyse détaillée de cette méthode est présentée dans [18]. Si on note  $j$  le vecteur unitaire de la droite complexe, le calcul de gradient est donné par la formule suivante

$$\frac{d\mathcal{J}}{d\alpha_i} \simeq \frac{\Im(J(\mathbf{W}(\alpha + j\delta\alpha_i), \mathbf{X}(\alpha + j\delta\alpha_i)))}{\delta\alpha_i} \quad (2.14)$$

La section suivante présente les techniques de calcul de gradient qui ont la particularité d'être robustes surtout pour un problème d'optimisation aérodynamique. Le lecteur pourra se référer au papier de J. Petter et R. P. Dwight à ce sujet [19].

### 2.2.1.2 Calcul du gradient par la méthode de l'équation linéarisée discrète

Cette méthode consiste à linéariser le résidu discret aérodynamique par rapport aux paramètres de forme  $\alpha_i$  :

$$\frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \frac{d\mathbf{W}}{d\alpha_i} = - \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}} \frac{d\mathbf{X}}{d\alpha_i} \quad i \in \llbracket 1, n_\alpha \rrbracket \quad (2.15)$$

En résolvant les  $n_\alpha$  équations définies dans (2.15), nous obtenons la sensibilité de l'écoulement  $\frac{d\mathbf{W}}{d\alpha}$ . On peut alors calculer les dérivées totales de la fonction d'intérêt par rapport à  $\alpha$  :

$$\nabla_\alpha \mathcal{J}(\alpha) = \frac{\partial J}{\partial \mathbf{X}} \frac{d\mathbf{X}}{d\alpha} + \frac{\partial J}{\partial \mathbf{W}} \frac{d\mathbf{W}}{d\alpha} \quad (2.16)$$

Si on élimine le terme  $\frac{d\mathbf{W}}{d\alpha}$  dans (2.15) nous obtenons alors :

$$\nabla_\alpha \mathcal{J}(\alpha) = \frac{\partial J}{\partial \mathbf{X}} \frac{d\mathbf{X}}{d\alpha} - \frac{\partial J}{\partial \mathbf{W}} \left( \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right)^{-1} \left( \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}} \frac{d\mathbf{X}}{d\alpha} \right). \quad (2.17)$$

La valeur du gradient s'obtient en résolvant  $n_\alpha$  systèmes linéaires ce qui est coûteux dans le cadre d'un calcul haute-fidélité.

### 2.2.1.3 Calcul du gradient par la méthode adjointe discrète

Il existe deux façons d'obtenir les équations adjointes : soit en faisant intervenir le Lagrangien, soit en transposant l'équation linéarisée (2.15). En transposant la méthode de l'équation linéarisée, nous obtenons :

$$\frac{d\mathcal{J}(\alpha)}{d\alpha_i} = \left( \frac{\partial J}{\partial \mathbf{X}} \frac{d\mathbf{X}}{d\alpha_i} \right)^T - \left( \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}} \frac{d\mathbf{X}}{d\alpha_i} \right)^T \left( \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right)^{-T} \left( \frac{\partial J}{\partial \mathbf{W}} \right)^T \quad (2.18)$$

Nous remarquons qu'une inversion d'un système linéaire est nécessaire pour le calcul du gradient. Notre travail sera donc de résoudre le système adjoint suivant :

$$\begin{bmatrix} \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \end{bmatrix}^T \lambda = - \begin{bmatrix} \frac{\partial J}{\partial \mathbf{W}} \end{bmatrix}^T \quad (2.19)$$

où  $\lambda \in \mathbb{R}^N$  correspond au vecteur adjoint aérodynamique. L'avantage de la formulation adjointe est qu'elle est indépendante du nombre de paramètres de conception. Dans notre cas, nous optons pour cette formulation qui est adaptée à notre problématique.



## 2.2.2 Approche partitionnée pour la résolution d'un système adjoint aéro-structure

### 2.2.2.1 Équilibre aéroélastique

Notons les variables d'état du système couplé  $\mathbf{W}$  et  $\mathbf{U}$ , représentant respectivement les variables conservatives et le déplacement structure. À l'équilibre aéroélastique, les variables d'état ainsi que les maillages aérodynamique et structure satisfont de manière simultanée les équations discrétisées de la mécanique des fluides et de la mécanique des structures :

$$\begin{cases} \mathbf{R}_a(\mathbf{X}_a, \mathbf{W}, \mathbf{U}) = \mathbf{0} \\ \mathbf{R}_s(\mathbf{X}_s, \mathbf{W}, \mathbf{U}) = \mathbf{0} \end{cases} \quad (2.20)$$

où  $\mathbf{R}_a$  représente le résidu aérodynamique discret et  $\mathbf{R}_s$  le résidu structure discret. Ces deux blocs d'équations sont couplés via les efforts aérodynamiques  $\mathbf{Q}_a$  appliqués sur l'interface aéroélastique, qui induisent une déformation du maillage structure  $\mathbf{X}_s$ . Ce dernier modifie en retour le maillage aérodynamique  $\mathbf{X}_a$ . Nous introduisons le maillage aérodynamique de référence  $\mathbf{X}_{a0}$  qui est utilisé pour construire tous les opérateurs de transfert impliqués dans l'analyse aéroélastique. Dans la suite de l'étude, ce maillage est aligné avec le maillage bâti (c'est-à-dire sans charge extérieure) et avec le maillage structure  $\mathbf{X}_s$ . Une technique de transfert adaptée est appliquée à  $\mathbf{Q}_a$  pour obtenir les efforts structuraux  $\mathbf{Q}_s$  :

$$\mathbf{Q}_s(\mathbf{Q}_a(\mathbf{W}, \mathbf{X}_a), \mathbf{X}_{a0}, \mathbf{X}_s) = \mathbf{T}_{surf}^Q(\mathbf{X}_{a0}, \mathbf{X}_s)\mathbf{Q}_a(\mathbf{W}, \mathbf{X}_a) \quad (2.21)$$

où  $\mathbf{T}_{surf}^Q$  représente l'opérateur linéaire de transfert d'efforts. Connaissant les déplacements structuraux  $\mathbf{U}$ , on obtient le mouvement correspondant de la peau aérodynamique  $\delta\mathbf{X}_{a,surf}$  par application d'un opérateur de lissage tel que

$$\delta\mathbf{X}_{a,surf} = \delta\mathbf{X}_{a,surf}(\mathbf{X}_{a0}, \mathbf{X}_s, \mathbf{U}) = \mathbf{T}_{surf}^U(\mathbf{X}_{a0}, \mathbf{X}_s)\mathbf{U} \quad (2.22)$$

À nouveau, l'opérateur de transfert des déplacements  $\mathbf{T}_{surf}^U$  ne dépend que des maillages de référence. Finalement, l'intérieur du domaine fluide est déformé en propageant les déplacements surfaciques grâce à l'opérateur volumique  $\mathbf{T}_{vol}$  :

$$\mathbf{X}_a = \mathbf{X}_{a0} + \delta\mathbf{X}_a(\delta\mathbf{X}_{a,surf}, \mathbf{X}_{a0}) = \mathbf{X}_{a0} + \mathbf{T}_{vol}(\mathbf{X}_{a0})\delta\mathbf{X}_{a,surf} \quad (2.23)$$

où  $\mathbf{T}_{vol}(\mathbf{X}_{a0})$  dépend uniquement du maillage de référence  $\mathbf{X}_{a0}$ .

## 2.2.3 Approche partitionnée pour le système adjoint couplé

Considérons la fonction d'intérêt  $J(\mathbf{W}, \mathbf{X}_a, \mathbf{X}_s)$  ainsi que le paramètre d'optimisation  $p$ . Afin d'obtenir le système d'équations adjointes, on introduit une variable adjointe et on formule la fonction objectif augmentée dont le gradient total a pour expression :

$$\frac{dJ}{dp} = \frac{\partial J}{\partial \mathbf{W}} \frac{d\mathbf{W}}{dp} + \frac{\partial J}{\partial \mathbf{X}_a} \left( [\mathbf{A}] \frac{d\mathbf{U}}{dp} + [\mathbf{B}] \frac{\partial \mathbf{X}_{a0}}{\partial p} \right) + \lambda_a^T \frac{d\mathbf{R}_a}{dp} + \lambda_s^T \frac{d\mathbf{R}_s}{dp} \quad (2.24)$$

L'équation (2.24) est valable pour tous les vecteurs adjoints  $\lambda_a$  et  $\lambda_s$  et utilise l'hypothèse fondamentale qui dit que la variation totale des résidus doit être nulle pour toute variation de paramètre.

En développant  $d\mathbf{R}_a/dp$  et  $d\mathbf{R}_s/dp$  et en factorisant les termes  $d\mathbf{U}/dp$  et  $d\mathbf{W}/dp$ , on obtient :

$$\begin{aligned} \frac{dJ}{dp} = & \left( \frac{\partial J}{\partial \mathbf{W}} + \lambda_a^T \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} - \lambda_s^T [\mathbf{C}] \right) \frac{d\mathbf{W}}{dp} + \left( \frac{\partial J}{\partial \mathbf{X}_a} [\mathbf{A}] + \lambda_a^T \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} [\mathbf{A}] + \lambda_s^T (\mathbf{K} - [\mathbf{D}]) \right) \frac{d\mathbf{U}}{dp} \\ & + \left( \frac{\partial J}{\partial \mathbf{X}_a} + \lambda_a^T \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \right) [\mathbf{B}] \frac{d\mathbf{X}_{a0}}{dp} - \lambda_s^T [\mathbf{E}] \frac{d\mathbf{X}_{a0}}{dp} \end{aligned} \quad (2.25)$$

où  $\mathbf{K}$  représente la matrice de rigidité. Tout au long de notre étude, nous nous restreignons au cas où les paramètres de forme n'affectent ni la géométrie ni la rigidité de la structure. On pourra retrouver dans [20] la dérivation complète dans le cas général. Il est à noter qu'une telle hypothèse n'entraîne aucune perte de généralité dans ce travail de thèse puisque que notre objectif est centré sur les techniques de résolution des systèmes adjoints couplés. Les opérateurs constants  $[\mathbf{A}]$  à  $[\mathbf{E}]$  sont définis analytiquement par les expressions [20] :

$$[\mathbf{A}] = \mathbf{T}_{vol} \mathbf{T}_{surf}^U \quad (2.26)$$

$$[\mathbf{B}] = \mathbf{I} + \frac{\partial \delta \mathbf{X}_a}{\partial \mathbf{X}_{a0}} + \mathbf{T}_{vol} \frac{\partial \delta \mathbf{X}_{a,surf}}{\partial \mathbf{X}_{a0}} \quad (2.27)$$

$$[\mathbf{C}] = \mathbf{T}_{surf}^Q \frac{\partial \mathbf{Q}_a}{\partial \mathbf{W}} \quad (2.28)$$

$$[\mathbf{D}] = \mathbf{T}_{surf}^Q \frac{\partial \mathbf{Q}_a}{\partial \mathbf{X}_a} \mathbf{T}_{vol} \mathbf{T}_{surf}^U \quad (2.29)$$

$$[\mathbf{E}] = \mathbf{T}_{surf}^Q \frac{\partial \mathbf{Q}_a}{\partial \mathbf{X}_a} [\mathbf{B}] + \frac{\partial \mathbf{Q}_s}{\partial \mathbf{X}_{a0}} \quad (2.30)$$

Le système adjoint couplé est obtenu en annulant les termes factorisés relatifs à  $d\mathbf{W}/dp$  et  $d\mathbf{U}/dp$  dans (2.25) ce qui donne

$$\begin{bmatrix} \left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right]^T & -[\mathbf{C}]^T \\ [\mathbf{A}]^T \left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \right]^T & [\mathbf{K}]^T - [\mathbf{D}]^T \end{bmatrix} \begin{bmatrix} \lambda_a \\ \lambda_s \end{bmatrix} = \begin{bmatrix} - \left[ \frac{\partial J}{\partial \mathbf{W}} \right]^T \\ -[\mathbf{A}]^T \left[ \frac{\partial J}{\partial \mathbf{X}_a} \right]^T \end{bmatrix} \quad (2.31)$$

La stratégie de résolution du système adjoint (2.31) consiste à appliquer l'algorithme de Gauss-Seidel linéaire par bloc (LBGS) qui est un schéma itératif par bloc (voir Annexe B p. 128, Algorithme 3). Une fois les vecteurs adjoints calculés, le gradient total de la

fonction objectif est donné par l'expression

$$\frac{dJ}{dp} = \left( \frac{\partial J}{\partial \mathbf{X}_a} + \lambda_a^T \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \right) [\mathbf{B}] \frac{d\mathbf{X}_{a0}}{dp} - \lambda_s^T [\mathbf{E}] \frac{\partial \mathbf{X}_{a0}}{\partial p} \quad (2.32)$$

L'objectif de ce chapitre est donc d'améliorer la résolution du système adjoint fluide ci-dessus (étape 17 de l'Algorithme 3) à chaque couplage fluide-structure :

$$\left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right]^T \lambda_a^k = - \left[ \frac{\partial J}{\partial \mathbf{W}} \right]^T + \mathbf{RHS}_{stru} \quad (2.33)$$

Si nous interprétons l'expression (2.33) d'un point de vue purement algébrique, nous remarquons que cette dernière est équivalente à une séquence de systèmes linéaires dont le second membre seulement varie entre chaque couplage fluide-structure (F/S). La variation du second membre est issue de la mise à jour du terme source structure (étape 3 de l'Algorithme 3). Une propriété intéressante de (2.33) est que le terme source structure converge vers une valeur constante après un certain nombre d'itérations F/S. Cette propriété implique au final une résolution purement fluide. Cette propriété sera utile notamment pour la stratégie de recyclage d'espace invariant approché.

### 2.3 Motivation du développement des solveurs de Krylov

L'idée de cette section est de justifier l'utilisation des méthodes de Krylov dans le cadre de la résolution des systèmes adjoints aérodynamiques (2.15) ou aéro-structures (2.31). Historiquement, les numériciens de la mécanique des fluides avaient pour habitude d'appliquer le solveur historique *Backward-Euler Newton* (BWE-Newton) aux systèmes adjoints. L'intérêt principal de cette approche est de réutiliser les briques logicielles disponibles pour la résolution par avance en temps fictif (*Time marching*) du problème stationnaire non-linéaire sous-jacent. La forme du schéma BWE dans le cas d'un problème adjoint est la suivante :

$$\left[ \frac{Vol}{\Delta\tau} + \left( \frac{\partial \mathbf{R}^{app}}{\partial \mathbf{W}} \right)^T \right] \Delta\lambda_k = - \left( \frac{\partial \mathbf{R}}{\partial \mathbf{W}} \right)^T \lambda_k - \left( \frac{\partial J}{\partial \mathbf{W}} \right)^T \quad (2.34)$$

Deux phases distinctes apparaissent dans (2.34) : la phase implicite et la phase explicite qui désignent respectivement la partie gauche et la partie droite de (2.34). La précision de la solution convergée ne dépend que de la phase explicite tandis que la convergence dépend essentiellement de la phase implicite. En pratique, on procède à une approximation de la jacobienne exacte de la discrétisation intervenant dans le membre de gauche pour des raisons de coût. Le système (2.34) est alors résolu de manière approchée par une méthode de relaxation LU-SGS (Lower-Upper Symmetric Gauss-Seidel).

La Figure 2.2 illustre le comportement typique de ce type de solveur. À gauche de la figure, on observe la convergence des résidus pour les 4 variables adjointes (densité de masse  $\rho$ , vitesse du fluide  $(u,w)$  et énergie totale  $\rho E$ ) qui affiche une divergence soudaine

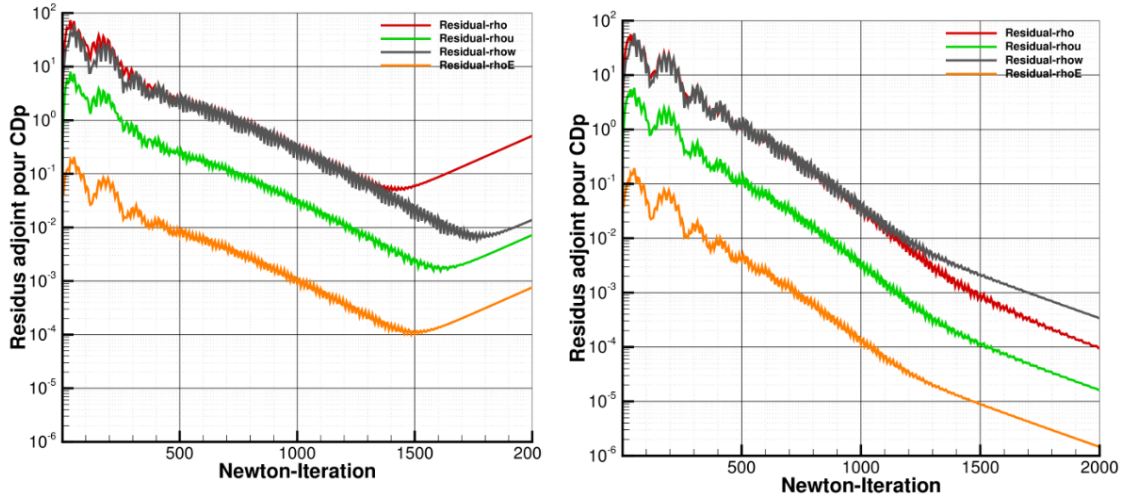


FIGURE 2.2 : Illustration du manque de robustesse du solveur de Newton inexact pour le problème adjoint. À droite, restauration de la convergence par ajout de dissipation artificielle au second membre.

après réduction de plus de 3 ordres de grandeur. La solution classique pour restaurer la convergence consiste en l'ajout d'une dissipation artificielle au second membre du système linéaire (à droite de la figure). Malheureusement, ceci a pour conséquence d'altérer la solution du système adjoint puisqu'on résout une version modifiée du problème initial.

Ce manque de robustesse fait qu'il est nécessaire d'exploiter d'autres types de solveurs plus robustes et ayant de bonnes propriétés de convergence. Les solveurs de Krylov respectent ces critères et sont connus pour leur robustesse. À ce stade, on cherchera à résoudre le système adjoint par une méthode itérative de Krylov préconditionnée :

$$\left( \frac{\partial \mathbf{R}}{\partial \mathbf{W}} \right)^T \lambda = - \left( \frac{\partial J}{\partial \mathbf{W}} \right)^T \quad (2.35)$$

## 2.4 Résolution d'un problème adjoint aérodynamique par méthode de Krylov

### 2.4.1 Contribution des méthodes de Krylov par bloc

Il est courant de traiter plusieurs fonctions d'intérêt lors d'un processus d'optimisation aérodynamique. Ceci implique la résolution du système (2.36) pour différents seconds membres. Le système adjoint par bloc s'écrit alors

$$\mathbf{A}\mathbf{\Lambda} = \mathbf{B} \quad (2.36)$$

où  $\mathbf{\Lambda} \in \mathbb{R}^{N \times p}$  représente le vecteur adjoint par bloc,  $\mathbf{B} \in \mathbb{R}^{N \times p}$  l'ensemble des  $p$  fonctions d'intérêt aérodynamiques et  $\mathbf{A}$  la matrice jacobienne des flux convectifs et diffusifs. Dans le cadre de l'utilisation des méthodes de Krylov, le système (2.36) se résout classiquement au

moyen de deux approches. Il y a l'approche séquentielle qui consiste à résoudre  $p$  systèmes linéaires successifs via la méthode GMRES. La deuxième approche utilise la version par bloc via la méthode BGMRES et qui résout tous les seconds membres simultanément. L'intérêt de la formulation par bloc est double : (i) L'espace de recherche (somme d'espaces de Krylov) est plus large, donc plus riche en information (ii) les opérations algébriques traitées par blocs autorisent un accès plus rapide à la mémoire.

Dans le cadre d'une optimisation aérodynamique, Pinel et Montagnac [9] ont réalisé une étude comparative des solveurs de Krylov afin de démontrer les performances de la version bloc par rapport à la version séquentielle. Dans leurs expériences numériques, ils utilisent les équations RANS comme modèle fluide ainsi que le modèle de turbulence  $k-\omega$  de Wilcox [21].

$$\frac{\partial \mathbf{W}}{\partial t} + \nabla \cdot (\mathbf{F}_c - \mathbf{F}_d) = \mathbf{T} \quad (2.37)$$

Le vecteur d'état  $\mathbf{W}$ , le flux convectif  $\mathbf{F}_c$ , le flux diffusif  $\mathbf{F}_d$  ainsi que le terme source  $\mathbf{T}$  sont donnés dans [9]. Le schéma spatial employé pour les flux convectifs du champ moyen est le schéma de Roe décentré avec correction entropique de Harten et associé à une extrapolation MUSCL (Monotone Upstream-centered Scheme for Conservative Law) [22] et un limiteur Van Albada [22]. Les flux convectifs turbulents sont discrétisés avec le schéma de Roe à l'ordre 1. Enfin, les flux diffusifs sont discrétisés avec un schéma centré du second ordre.

Pinel et Montagnac [9] ont considéré cinq stratégies de résolution. La première consiste à résoudre un système linéaire issu du schéma d'Euler rétrograde introduit par Beam et Warming [23]. Ce schéma est aussi appelé méthode de Newton inexacte. En reprenant les notations de [9], le système à résoudre s'écrit :

$$(\mathbf{D}_t + \tilde{\mathbf{A}})(\lambda^{l+1} - \lambda^l) = -\mathbf{A}\lambda^l - \mathbf{b} \quad (2.38)$$

où  $\mathbf{D}_t = \frac{V}{\Delta t} \mathbf{I}_n \in \mathbb{R}^{N \times N}$ ,  $V$  le volume de la cellule courante,  $\Delta t$  le pas de temps local et  $\tilde{\mathbf{A}} \in \mathbb{R}^{N \times N}$  une approximation de  $\mathbf{A}$ . Le terme  $\mathbf{D}_t$  a pour rôle d'augmenter la dominance diagonale de la phase implicite de (2.38) et rendre ainsi la méthode plus robuste car c'est bien la phase implicite qui contrôle la vitesse de convergence. La matrice  $\tilde{\mathbf{A}}$  est construite à partir de la linéarisation décentrée d'ordre 1 de Steger et Warming des flux convectifs et de la linéarisation centrée d'ordre 1 des flux diffusifs en négligeant les dérivées croisées. Le fait de négliger les dérivées croisées est connu sous le nom de l'hypothèse de "couche mince" [11]. La phase explicite (membre de droite) fait intervenir la matrice du système  $\mathbf{A}$  qui contrôle la précision de la solution. Pour résoudre le système (2.38), les auteurs emploient une méthode de relaxation de type LU-SSOR par bloc (Lower-Upper Successive Symmetric Over Relaxation) [17].

Les deuxième et troisième stratégies consistent à résoudre directement le système ad-

joint (2.36) via un GMRES flexible (FGMRES( $m$ )) et un GMRES flexible avec stratégie de déflation (FGMRES-DR( $m,k$ )) respectivement. La stratégie de déflation consiste à éliminer l'effet indésirable de certaines valeurs propres  $\omega_i$  de l'opérateur préconditionné de sorte à accélérer la convergence. Ces valeurs propres sont choisies selon deux critères. Le premier consiste à sélectionner les plus petites valeurs propres en valeur absolue (c'est-à-dire les plus petites valeurs de  $|\omega_i|$ ) et le deuxième consiste à choisir les valeurs propres extrêmes qui se situent à l'extérieur du cercle centré en 1 (c'est-à-dire les plus grandes valeurs de  $|1 - \omega_i|$ ). Les auteurs ont choisi la deuxième option qui s'avère plus performante pour les cas étudiés dans [24]. Les paramètres  $m$  et  $k$  correspondent respectivement à la taille maximale de l'espace de Krylov pour un cycle de GMRES et au nombre de vecteurs propres déflatés. Dans [9], le paramètre  $k$  a été fixé à 50 % de la taille de l'espace de Krylov  $m$ .

Enfin les quatrième et cinquième stratégies font appel à la version bloc de FGMRES ( $m$ ) et de FGMRES-DR( $m,k$ ) notées respectivement BFGMRES( $m$ ) et BFGMRES-DR( $m,k$ ). Le GMRES(5) est utilisé comme préconditionneur de premier niveau tandis qu'un LU-SSOR par bloc est utilisé comme préconditionneur de deuxième niveau dans le GMRES(5). On rappelle que la stratégie FGMRES par bloc est déjà disponible dans le code elsA [25].

#### 2.4.1.1 Configuration NACA 64212

Pinel et Montagnac [9] considèrent tout d'abord un profil NACA 64212 en régime transsonique dont le domaine fluide est discrétisé par un maillage de type C structuré et constitué de 16 000 cellules. Le nombre de Mach est de 0.71 et le nombre de Reynolds est fixé à  $2 \times 10^6$ . L'angle d'attaque est de 2.5 degrés.

La Table 2.1 présente les performances de la méthode de Krylov flexible par rapport à la méthode de Newton inexacte pour la résolution du problème adjoint (2.36) pour trois seconds membres. Ces derniers sont liés respectivement au coefficient de portance, au coefficient de traînée en champ proche paroi (*near-field drag coefficient*) et au coefficient de traînée en champ lointain (*far-field drag coefficient*).

La Table 2.1 montre un gain de performance des méthodes de Krylov flexibles par rapport au solveur de Newton inexact. Plus précisément, toutes les variantes du FGMRES réalisent au moins un gain d'un facteur 2 en temps de calcul pour converger à la tolérance de  $10^{-8}$ . Pour le solveur BFGMRES-DR(20,10) et le solveur Full FGMRES, nous obtenons un facteur 4. Nous rappelons que la stratégie Full FGMRES consiste à générer un espace de Krylov sans redémarrage. Nous constatons également que la stratégie de déflation permet de réduire le nombre d'itérations pour la version bloc et la version standard pour une empreinte mémoire similaire. Cette réduction est de plus observée au niveau du temps de calcul avec un gain de 15 % pour  $m = 10, 20$  et un gain de 5 % pour  $m = 50$ . L'autre constat est la capacité de la version bloc BFGMRES-DR à surclasser la version standard FGMRES-DR en nombre d'itérations et en temps de calcul mais pour un coût mémoire

Methods	$m$	$k$	Its	T, s	M, MB
Approximate Newton	-	-	14,540	2166	120
FGMRES( $m$ )	10	-	928	997	221
	20	-	710	772	292
	50	-	580	645	504
	$\infty$	-	463	568	1275
FGMRES-DR( $m,k$ )	10	5	825	905	221
	20	10	561	632	292
	50	25	514	616	504
BFGMRES( $m$ )	10	-	774	856	376
	20	-	597	681	589
	50	-	495	613	1225
BFGMRES-DR( $m,k$ )	10	5	567	670	376
	20	10	450	569	589
	50	25	405	601	1225

TABLE 2.1 : Performance des méthodes de Krylov emboîtées pour la résolution du système adjoint (Figure reproduite à partir de [9]).

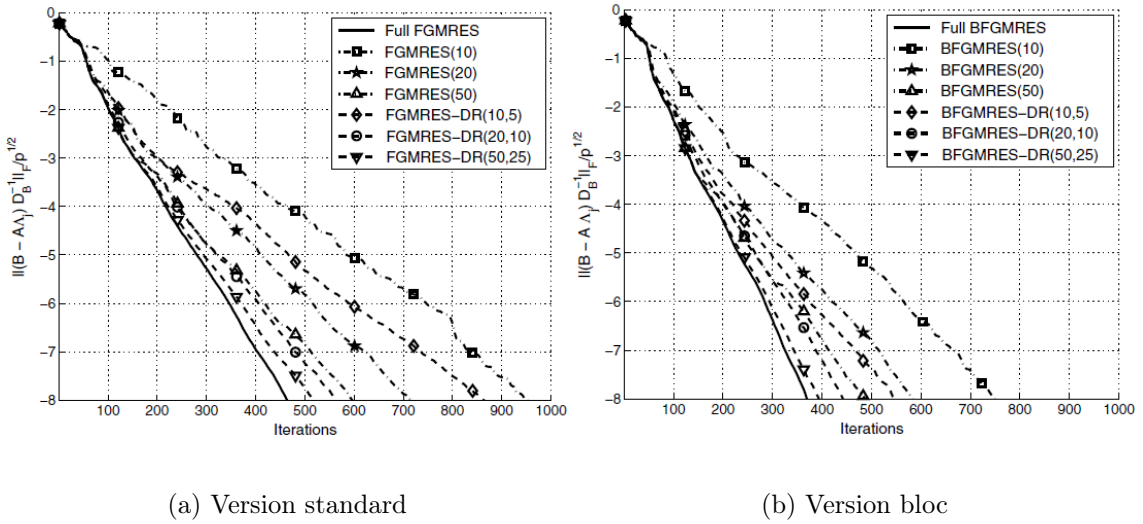


FIGURE 2.3 : Convergence des solveurs FGMRES et BFGMRES avec et sans déflation pour différentes valeurs de  $m$  (Figures extraites de [9]).

deux fois plus élevé. Enfin, le BFGMRES-DR est capable d'imiter le comportement du Full BFGMRES tout en économisant 50 % de la mémoire (voir la Table 2.1). Néanmoins, cette version a aussi quelques limitations dès que la taille de l'espace de Krylov externe devient importante. Il suffit de remarquer que les performances du Full BFGMRES deviennent moins bonnes par rapport au BFGMRES-DR(20,10). Cette contre-performance est due au coût de l'orthogonalisation qui est nettement plus important pour la version bloc.

Nous présentons sur les Figures 2.3a et 2.3b la convergence du résidu relatif pour les solveurs FGMRES-DR et BFGMRES-DR. Nous attirons l'attention sur le fait que la norme de Frobenius du résidu relatif par bloc  $\frac{\|R_j D_B^{-1}\|_F}{\sqrt{p}}$  est tracée sur les deux Figures avec  $\mathbf{R}_j = \mathbf{B} - \mathbf{A}\mathbf{A}_j$  et  $\mathbf{D}_B \in \mathbb{R}^{N \times p}$  la matrice diagonale dont chaque entrée correspond à

la norme euclidienne des colonnes de  $\mathbf{B}$  ( $\mathbf{D}_B = \text{diag}(\|\mathbf{B}(:,l)\|_2)$ ), où  $l = 1, \dots, p$ .

Blondeau a mené une étude comparative des formulations séquentielle et bloc du solveur GMRES-DR dans le cadre de la résolution d'un système adjoint aérodynamique et couplé [26]. Les résultats numériques ont été obtenus pour le cas test aile M6 de l'ONERA pour un modèle fluide RANS couplé au modèle de turbulence à une équation Spalart-Allmaras. L'auteur considère deux fonctions d'intérêt, soit  $p = 2$ . Plus précisément, l'auteur a résolu un système adjoint fluide au moyen d'un GMRES emboîté. La convergence a été réalisée selon 3 cas de figures en fonction du choix des paramètres du FGMRES-DR. Il a notamment testé le FGMRES-DR(200,10), le FMGRES(30,10) et enfin le FGMRES-DR(30,10,15). La Table 2.2 résume les performances du solveur GMRES emboîté dans le cas séquentiel et dans le cas par bloc.

Case	GMRES	Nrelax	size of Krylov space		$k$	Nbr. its Séq/Bloc	CPU time (s)
1	Nested	6	200	10, 0.7	0	(75,74)/152	4197/4500
2	Nested	6	30	10, 0.7	0	(89,86)/174	5017/4988
3	Nested	6	30	10, 0.7	15	(80,80)/168	4628/4877

TABLE 2.2 : Performances du GMRES emboîté. Tolérance du résidu =  $10^{-4}$ .

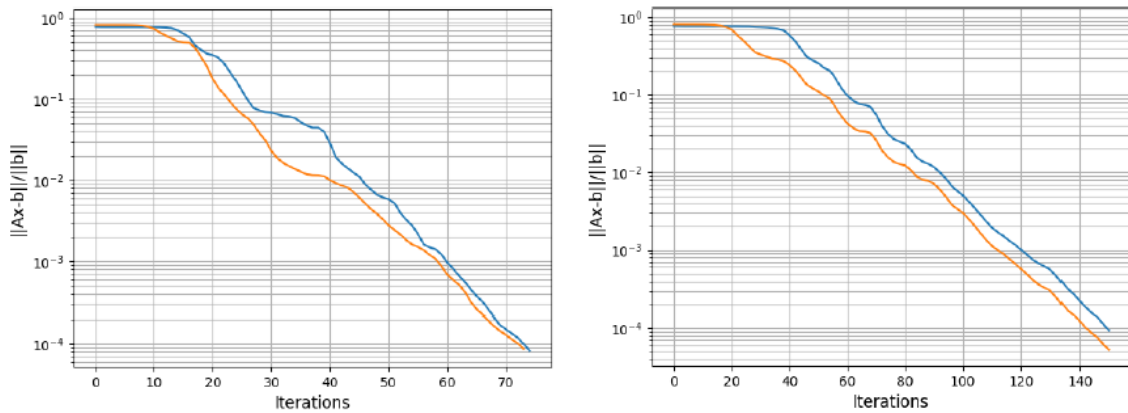


FIGURE 2.4 : Historique de convergence de la norme du résidu relative pour deux fonctions d'intérêt. Mode séquentiel (à gauche) vs mode par bloc (à droite).

Au vu des résultats et de la Figure 2.4, l'auteur conclut que les deux formulations aboutissent au même nombre d'itérations avec un avantage en temps de calcul pour la formulation séquentielle. De plus, l'efficacité de la formulation par bloc peut parfois être limitée par le fait que les résidus associés à chaque second membre ne convergent pas forcément à la même vitesse.

A. Bissuel a récemment mené une étude sur les méthodes de Krylov par bloc dans le cadre d'une résolution d'un problème adjoint aérodynamique. Le calcul a été effectué sur le cas Falcon générique tridimensionnel [27]. La figure 2.5 présente la courbe de convergence



du solveur GMRES par bloc avec un espace de Krylov par bloc de 50, soit un espace de Krylov de 5 pour chaque second membre (total de 10 seconds membres). Les courbes fines correspondent à la convergence du résidu associé à un second membre unique tandis que la courbe épaisse en bleue correspond à la convergence du résidu global. On remarque que le résidu a diminué de 2.5 ordres de grandeur au bout de 4000 itérations. À l'opposé, la formulation séquentielle du solveur GMRES demande moins d'itérations pour converger à la même précision. La conclusion est que la formulation par bloc n'est pas satisfaisante en terme d'efficacité. A. Bissuel explique que cela est principalement due au coût d'orthonormalisation qui est proportionnel au carré de la taille de l'espace de Krylov. Il est donc impératif de garder un espace de petite taille pour espérer une résolution rapide. Cela impose de limiter le nombre de seconds membres résolus simultanément, ainsi que la taille de l'espace de Krylov  $m$ . La conséquence de cela est d'une part la faible précision des petites valeurs propres de  $\mathbf{A}$  et d'autre part le nombre limité de seconds membres lors de la résolution par bloc. Il conclut donc que la formulation par bloc n'est pas compétitive comparée à la formulation séquentielle pour des problèmes difficiles. D'ailleurs Morgan aboutit à la même conclusion dans [28].

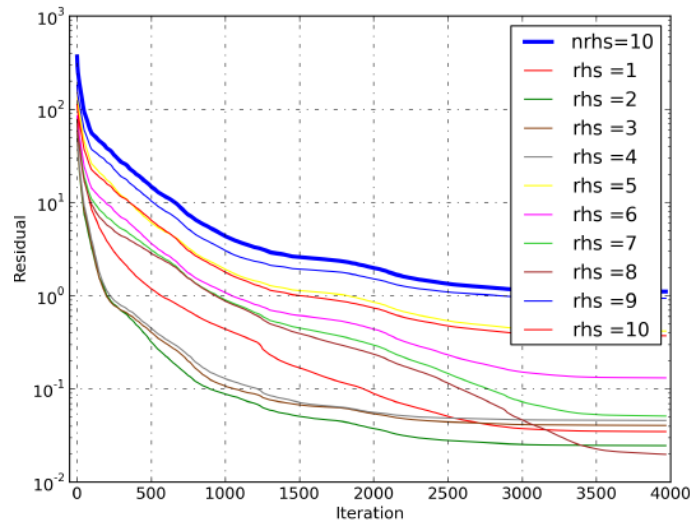


FIGURE 2.5 : Historique de convergence de la norme du résidu pour 10 seconds membres. 50 vecteurs de Krylov. Trait épais : résidu global. Trait fin : résidu partiel de chaque second membre (Figure extraite de [27]).

## 2.5 Résolution du problème adjoint aéro-structure par méthode de Krylov

Soit le système adjoint couplé suivant

$$\mathbf{A}\lambda = \mathbf{b}, \tag{2.39}$$

où  $\mathbf{A} \in \mathbb{R}^{N \times N}$  représente la transposée de la matrice jacobienne du second ordre  $\partial \mathbf{R} / \partial \mathbf{Y}$  où  $\mathbf{Y} = (\mathbf{W}, \mathbf{U})$  est le vecteur d'état aéro-structure,  $\mathbf{b} \in \mathbb{R}^N$  le second membre qui correspond à la dérivée partielle de la fonction objectif par rapport aux variables d'états et  $\lambda = (\lambda_a, \lambda_s)$  le vecteur adjoint couplé.

### 2.5.1 Approche monolithique et approche partitionnée

Kenway et al. [7] comparent l'approche partitionnée et l'approche monolithique pour résoudre le système adjoint couplé (2.39). La première approche se base sur une résolution itérative de type Gauss-Seidel linéaire par blocs (LBGS) [7] dont le principe est de résoudre à chaque itération les équations fluide et structure en mettant à jour les seconds membres. Ces derniers sont issus des termes croisés entre le fluide et la structure. L'avantage de cette méthode est qu'elle réutilise les solveurs existants de chacune des disciplines sans exiger un effort d'implémentation supplémentaire. La contrepartie de cette méthode est la nécessité d'introduire un paramètre de relaxation  $\theta$  pour préserver la stabilité du système surtout pour des problèmes à interaction forte. La partie fluide est résolue de manière approchée à l'aide du GMRES dont la taille maximale de l'espace de Krylov est de 75.

La deuxième approche se base sur une résolution itérative de type Krylov dont le principe est de résoudre le système adjoint global (2.39) en cherchant une approximation de la solution  $\lambda = \mathbf{x}_m$  dans un espace de Krylov  $\mathcal{K}_m$  par minimisation du résidu  $\mathbf{r}_m = \mathbf{b} - \mathbf{A}\mathbf{x}_m$  à l'itération  $m$ . Ces méthodes exigent cependant des techniques de préconditionnement robustes. Dans [7], les auteurs ont choisi de préconditionner le système (2.39) au moyen de la méthode de Jacobi par bloc dans l'idée de réutiliser les préconditionneurs propres au problème fluide et structure et de les appliquer dans un contexte parallèle. En terme d'occupation mémoire, seuls les termes blocs diagonaux de  $\mathbf{A}$  qui correspondent à la matrice jacobienne exacte des flux  $\partial \mathbf{R}_a / \partial \mathbf{W}$  pour la partie fluide et à la matrice de rigidité  $\mathbf{K}$  pour la partie structure sont stockées.

Ainsi, le préconditionnement du système (2.39) consiste à résoudre approximativement le problème fluide par une méthode de Krylov et à effectuer une factorisation incomplète de Cholesky de  $\mathbf{K}$ . Pour la partie fluide, les auteurs font appel à un préconditionneur GMRES( $s$ ) (où  $s$  est la taille maximale de l'espace de Krylov et  $10 \leq s \leq 20$ ) préconditionné par un ILU(2) (Incomplete Lower-Upper Factorization) et couplé à une méthode de Schwarz additive restrictive (RAS). La résolution du système adjoint couplé (2.39) est effectuée au moyen du solveur FGMRES( $\lfloor \frac{75-s}{2} \rfloor$ ) où  $\lfloor \cdot \rfloor$  représente la partie entière. Ce type de solveur correspond à un GMRES flexible (FGMRES) dans le sens où le préconditionneur varie d'une itération à une autre. Dans ce cas précis, le FGMRES fait référence à un GMRES emboîté dont la taille de l'espace externe est de  $\lfloor \frac{75-s}{2} \rfloor$  et la taille de l'espace interne est de  $s$ . Le choix particulier de ces deux derniers paramètres a été effectué dans le but de comparer l'approche monolithique et l'approche partitionnée pour une contrainte

mémoire identique.

Leurs expériences numériques se basent essentiellement sur un cas test numérique 3D représentatif, le CRM (Common Research Model). Ce modèle a été conçu par la NASA en 2009 dans le but de répondre à une demande de développer de nouvelles bases de données expérimentales pour valider des applications spécifiques CFD [29]. Ces bases de données permettent une meilleure compréhension des simulations numériques, notamment dans le but d'améliorer les prédictions des coefficients aérodynamiques. La NASA a donc développé une configuration d'avion transsonique à voilure supercritique munie d'un fuselage, d'un empennage, d'une nacelle et d'un pylône. Le CRM est conçu pour voler à Mach 0.85 pour un coefficient de portance de 0.5 à une altitude nominale de 37000 pieds. Kenway et al. [7] développent en 2014 un modèle structure du CRM dédié à l'optimisation aéro-structure. La Table 2.3 regroupe l'essentiel des paramètres qui caractérisent le CRM.

Parameter	Value	Units
Cruise Mach number	0.85	-
Cruise lift coefficient	0.5	-
Span	58.6	m
Aspect ratio	9.0	-
Reference wing area	383.7	m <sup>2</sup>
Sweep (leading edge)	37.4	deg
Maximum takeoff weight	298.000	kg
Operational empty weight	138.100	kg

TABLE 2.3 : Paramètres caractéristiques du CRM (Table extraite de [7]).

Deux cas de charges sont considérés : le premier cas est une déformation élastique modérée de l'aile sous un facteur de charge de 1g, tandis que le deuxième cas correspond à une déformation plus significative de l'aile avec un facteur de charge de 2.5g. Ce dernier cas est plus intéressant car il correspond à une forte interaction entre le fluide et la structure et rend le problème numériquement raide. Les tailles des problèmes fluide et structure sont de 10 550 480 et de 324 180, respectivement. Concernant le fluide, les équations d'Euler ont été utilisées.

La Figure 2.6 présente les résultats de convergence de (2.39) pour l'approche partitionnée (LBGS) et pour l'approche monolithique (CK). La résolution a été réalisée à une tolérance de  $\epsilon_{AS} = 10^{-8}$ .

Le solveur (LBGS) dépend essentiellement de deux paramètres, le facteur de relaxation  $\theta^0$  et la tolérance du problème adjoint aérodynamique  $\epsilon_A$ . Pour  $\theta^0$ , deux cas de figure ont été considérés ( $\theta^{(0)} = 0.5, 1.0$ ) tandis que pour  $\epsilon_A$ , nous avons trois cas de figure ( $\epsilon_A = 0.10, 0.25, 0.50$ ). Le solveur de Krylov (CK) dépend de deux paramètres, la taille de l'espace de Krylov interne  $10 \leq s \leq 20$  et la taille de l'espace de Krylov externe  $\lfloor \frac{75-s}{2} \rfloor$ . Le fait de varier les paramètres numériques permet de sélectionner les bons paramètres afin

Load factor	Solver	$\theta^{(0)}$	$\epsilon_A$	Inner size $s$	Outer size $m$	$N_{iter}$	Time, s			
							CFD	Mesh	CSM	Total
1.0	LBGS	0.5	0.10	—	—	21	542.6	9.1	2.1	554.0
1.0	LBGS	0.5	0.25	—	—	22	304.2	9.5	2.3	316.1
1.0	LBGS	0.5	0.50	—	—	23	141.7	10.0	2.3	154.1
1.0	LBGS	1.0	0.10	—	—	17	375.9	7.4	1.7	385.0
1.0	LBGS	1.0	0.25	—	—	16	158.9	7.0	1.6	167.6
1.0	LBGS	1.0	0.50	—	—	21	111.6	9.1	2.1	123.0
1.0	CK	—	—	5	35	120	78.2	53.4	10.6	136.3
1.0	CK	—	—	10	32	58	71.3	25.1	5.1	99.6
1.0	CK	—	—	20	27	40	99.8	17.3	3.5	118.5
2.5	LBGS	0.5	0.10	—	—	21	552.4	9.1	2.1	563.8
2.5	LBGS	0.5	0.25	—	—	21	311.1	9.2	2.1	322.5
2.5	LBGS	0.5	0.50	—	—	24	139.2	10.5	2.4	152.2
2.5	LBGS	1.0	0.10	—	—	79	4518.4	34.5	7.8	4561.3
2.5	LBGS	1.0	0.25	—	—	42	308.1	18.2	4.2	330.1
2.5	LBGS	1.0	0.50	—	—	46	175.9	20.0	4.5	200.8
2.5	CK	—	—	5	35	125	81.2	55.5	11.0	141.8
2.5	CK	—	—	10	32	53	77.6	27.4	5.6	107.4
2.5	CK	—	—	20	27	47	115.9	20.4	4.1	137.9

FIGURE 2.6 : Comparaison des deux approches pour la résolution du système couplé adjoint (Figure extraite de [7]).

d’obtenir la meilleure performance du solveur.

Dans le cas où l’interaction fluide-structure est faible (charge 1g), le solveur LBGS converge sans trop de problème si ce n’est que la convergence est meilleure sans relaxation ( $\theta = 1.0$ ). En particulier, le choix de  $\theta^{(0)} = 1.0$  et  $\epsilon_A = 0.5$  offre la meilleure performance avec un gain de 68 % par rapport à une tolérance  $\epsilon_A = 0.1$ . Dans le cas d’une interaction forte (charge 2.5g), le facteur  $\theta^{(0)} = 0.5$  est plus efficace en temps de calcul. En choisissant une tolérance  $\epsilon_A = 0.50$ , les auteurs obtiennent la meilleure performance avec un gain de 53 % par rapport au cas  $\theta^{(0)} = 0.5$  et  $\epsilon_A = 0.25$ .

Pour le solveur CK, on observe une stabilité en termes de performance quelque soit le degré d’interaction. Si on compare la meilleure performance du solveur CK à celle du solveur LBGS, on observe un gain de 19 % en coût CPU dans le cas 1g et un gain de 29 % dans le cas 2.5g en faveur du solveur CK. Ce qui prouve l’efficacité des méthodes de Krylov flexibles couplées à une approche monolithique pour des systèmes adjoints fortement couplés.

Zhang et Zingg [4] proposent dans le même esprit une étude comparative entre les deux approches pour résoudre le système couplé (2.39) en utilisant une formulation à trois champs dans laquelle ils expriment le système aéro-structure comme trois systèmes d’équations couplées, c’est-à-dire un bloc d’équations par discipline plus un autre dédié à la déformation de maillage fluide. Une comparaison des performances entre le préconditionneur de Jacobi par bloc et le préconditionneur Gauss-Seidel par bloc est réalisée. Dans [4], le système (2.39) s’écrit

$$\underbrace{\begin{bmatrix} \frac{\partial \mathbf{R}_a T}{\partial \mathbf{q}} & 0 & \frac{\partial \mathbf{R}_s T}{\partial \mathbf{q}} \\ \frac{\partial \mathbf{R}_a T}{\partial \mathbf{b}_\Delta} & \frac{\partial \mathbf{R}_{M\Delta} T}{\partial \mathbf{b}_\Delta} & \frac{\partial \mathbf{R}_s T}{\partial \mathbf{b}_\Delta} \\ 0 & \frac{\partial \mathbf{R}_{M\Delta} T}{\partial \mathbf{u}} & \frac{\partial \mathbf{R}_s T}{\partial \mathbf{u}} \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} \lambda_a \\ \lambda_{M\Delta} \\ \lambda_s \end{bmatrix} = \begin{bmatrix} -\frac{\partial \mathcal{J}^T}{\partial \mathbf{q}} \\ -\frac{\partial \mathcal{J}^T}{\partial \mathbf{b}_\Delta} \\ -\frac{\partial \mathcal{J}^T}{\partial \mathbf{u}} \end{bmatrix} \quad (2.40)$$

où  $\mathbf{R}_A$ ,  $\mathbf{R}_{M\Delta}$  et  $\mathbf{R}_S$  représentent les résidus discrets fluide, maillage et structure tandis que  $\mathbf{q}$ ,  $\mathbf{b}_\Delta$  et  $\mathbf{u}$  correspondent aux variables d'états fluide, maillage et structure respectivement. Les auteurs procèdent également à une stratégie de mise à l'échelle afin d'améliorer le conditionnement de  $\mathbf{A}$ . On verra dans la suite que cette stratégie aura un impact sur la précision du gradient selon le choix de  $\mathcal{J}$ .

Contrairement à Kenway et al [7], Zhang et Zingg font appel au solveur de Krylov FGCROT( $m,k$ ) (Generalized Conjugate Residual with Orthogonalization and Outer Truncation) [30] qui est une version flexible du GCROT [5]. Plus précisément, ce solveur utilise deux méthodes de Krylov emboîtées avec GCRO( $k$ ) comme méthode externe et un FGMRES( $m$ ) comme méthode interne. Le point intéressant à souligner est que le solveur GCROT( $m,k$ ) fait partie des méthodes de Krylov conçues pour améliorer la résolution des séquences de systèmes linéaires de type  $\mathbf{A}\lambda_i = \mathbf{b}_i$  où  $\mathbf{b}_i$  est un second membre qui varie. Ce type de système apparaît naturellement dans la formulation de l'approche partitionnée décrite ci-dessous :

$$\frac{\partial \mathbf{R}_A^T}{\partial \mathbf{q}} \lambda_a^{(n+1)} = -\frac{\partial \mathcal{J}^T}{\partial \mathbf{q}} - \frac{\partial \mathbf{R}_S^T}{\partial \mathbf{q}} \lambda_s^{(n)} \quad (2.41)$$

$$\frac{\partial \mathbf{R}_{M\Delta}^T}{\partial \mathbf{b}_\Delta} \lambda_{M\Delta}^{(n+1)} = -\frac{\partial \mathcal{J}^T}{\partial \mathbf{b}_\Delta} - \frac{\partial \mathbf{R}_A^T}{\partial \mathbf{b}_\Delta} \lambda_a^{(n+1)} - \frac{\partial \mathbf{R}_S^T}{\partial \mathbf{b}_\Delta} \lambda_s^{(n)} \quad (2.42)$$

$$\mathbf{K}_S \lambda_s^{(n+1)} = -\frac{\partial \mathcal{J}^T}{\partial \mathbf{u}} - \frac{\partial \mathbf{R}_{M\Delta}^T}{\partial \mathbf{u}} \lambda_{M\Delta}^{(n+1)} \quad (2.43)$$

où  $\mathbf{K}_S$  et l'indice  $n$  représentent respectivement la matrice de rigidité et l'itération courante.

Pour améliorer la résolution d'une séquence de systèmes linéaires provenant d'un système aéro-structure couplé, l'idée serait de recycler une information spectrale du système précédant afin d'accélérer la résolution du système suivant et des méthodes du type GCRO sont justement capables de réaliser cela. Cela fera l'objet du Chapitre 4.

Concernant le cas test, les auteurs étudient une configuration voilure de Boeing 737-900 basée sur le profil RAE 2822. Le nombre de Mach est de 0.798 à une altitude de 35,000 ft et l'angle d'attaque est de 2 degrés. Le maillage fluide est composé de 458,752 nœuds et découpé en 112 blocs.

Les Figures 2.7a et 2.7b illustrent la convergence du solveur GCROT( $m,k$ ) monolithique pour la résolution de (2.40) selon le cas où une stratégie de mise à l'échelle est appliquée ou

non. Nous précisons que la mise à l'échelle consiste à normaliser les lignes et les colonnes de la matrice du système afin d'obtenir un meilleur conditionnement. Ce solveur utilise un préconditionneur de type Gauss-Seidel par bloc de sorte que la partie fluide soit résolue par un FGMRES. La Figure 2.7a présente les résultats de convergence pour le coefficient de portance tandis que la Figure 2.7b présente ceux pour la fonction d'agrégation des contraintes de Kreisselmeier-Steinhauser (KS2). Les paramètres du GCROT ont été fixés à  $m = 20$  et  $k = 1$ .

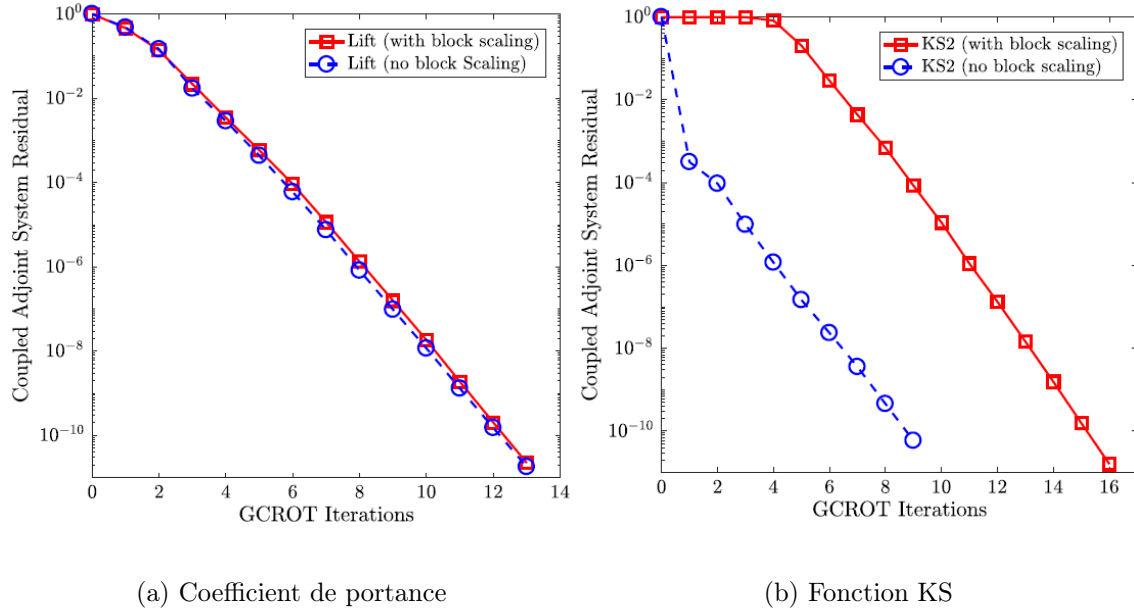


FIGURE 2.7 : Historique de convergence de la norme du résidu relatif pour le solveur monolithique GCROT (Figure extraite de [4]).

Les courbes de la Figure 2.7a montrent que la mise à l'échelle n'apporte pas d'amélioration tandis que celles de la Figure 2.7b montrent une accélération de convergence dans le cas où le second membre correspond à la fonction d'agrégation KS2 mais sans mise à l'échelle. On s'attendrait logiquement à ce que la mise à l'échelle améliore la convergence du système adjoint. En analysant de plus près, le système non mis à l'échelle fournit une solution moins précise que dans le cas d'une mise à l'échelle. Dans sa thèse, Zhang démontre que pour obtenir une précision équivalente à celle obtenue avec une mise à l'échelle, il est nécessaire de faire converger le système adjoint à une tolérance de  $10^{-13}$  (voir la partie 6.2.1 de [31]).

Zhang et Zingg étudient les performances du préconditionneur Gauss-Seidel par bloc qui a été choisi pour la résolution du problème adjoint. Plus précisément, un gain de 72 % en coût CPU est observé en faveur du solveur monolithique dans le cas où  $\theta = 0.5$  et un gain de 61 % dans le cas où  $\theta = 0.75$  et où  $\theta$  est le coefficient de relaxation sur le vecteur adjoint structure.

Zhang et Zingg [4] approfondissent leurs expériences numériques en menant une étude paramétrique sur le degré d'interaction fluide-structure. Ce degré d'interaction est quantifié par le ratio  $q_\infty/E$  où  $q_\infty$  et  $E$  représentent respectivement la pression dynamique du fluide et le module d'Young de la structure. Une série de 10 valeurs réparties uniformément est considérée comme le montre la Table 2.4.

Data point	1	2	3	4	5	6	7	8	9	10
$q_\infty/E$ ( $1 \times 10^{-8}$ )	1.46	7.76	14.1	20.4	26.7	33.0	39.3	45.6	51.9	58.2
Tip deflection, % (of semispan)	0.86	3.81	5.82	7.2	8.15	8.79	9.18	9.39	9.46	9.41

TABLE 2.4 : Série de valeurs de  $q_\infty/E$  pour l'analyse de performance (Figure extraite de [4]).

Le jeu de données 3 correspond à la forme croisière sous 1g avec un nombre de Mach de 0.785 à une altitude de 35,000 ft. Le jeu de données 7 correspond à la condition de charge de 2.5g pour un nombre de Mach de 0.798 et à une altitude de 12,000 ft. Le reste des données est généré à partir de ces deux données par simple extrapolation. Pour les petites valeurs de  $q_\infty/E$ , le solveur monolithique est environ 42 % plus efficace que le solveur partitionné pour un facteur de relaxation  $\theta = 0.75$ . Cependant, ce facteur de relaxation est associé à des limitations numériques et devient inefficace pour un système fortement couplé (courbe rouge en continue sur la Figure 2.8). Le choix de  $\theta = 0.5$  semble être plus robuste que  $\theta = 0.75$  (courbe rouge en pointillé). Malgré cela, le solveur reste plus performant avec un gain allant à 70 % pour un fort degré d'interaction.

Pour tester les limites des deux solveurs, Zhang et Zingg [4] proposent d'augmenter le degré d'interaction jusqu'à un facteur 4 comme indiqué dans la Table 2.5. Typiquement, pour  $q_\infty/E = 2.47 \times 10^{-6}$ , le solveur partitionné ne converge pas et ce malgré l'utilisation d'une accélération de Aitken qui permet de choisir dynamiquement  $\theta$ . À l'inverse, le solveur monolithique paraît quasiment stable quelque soit la valeur de  $q_\infty/E$ .

$q_\infty/E$ ( $1 \times 10^{-8}$ )	Tip deflection, %	Monolithic analysis, s	Monolithic adjoint, s	Partitioned analysis, s	Partitioned adjoint, s
153	4.48	187	73.6	983	694 ( $\theta = 0.25$ )
184	2.72	256	81.4	9060	2922 ( $\theta = 0.1$ )
247	0.24	356	116	Failed	Failed ( $\theta = 0.1$ )

TABLE 2.5 : Comparaison des performances du solveur monolithique et partitionné pour des valeurs critiques de  $q_\infty/E$  (Figure extraite de [4]).

Le développement des solveurs de Krylov à l'ONERA (Office National d'Etudes et de Recherches Aérospatiales) a été motivé par la nécessité d'obtenir un calcul plus précis et

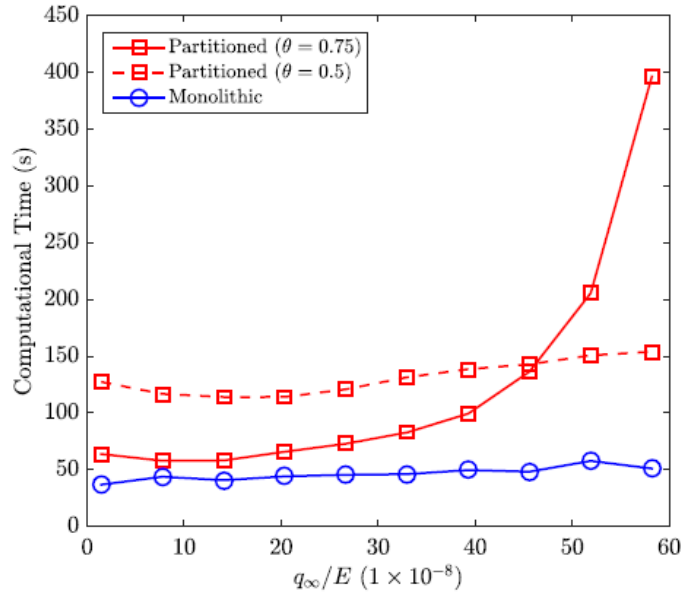


FIGURE 2.8 : Performance des deux approches (partitionnée et monolithique) en fonction des valeurs de  $q_\infty/E$  et pour  $M_\infty = 0.785$  (Figure extraite de [4]).

plus robuste des gradients aéro-structures. Comme expliqué dans l'introduction, la qualité de ce calcul joue un rôle essentiel dans la phase de conception d'un avion lors du processus d'optimisation. Comme le calcul de ce gradient nécessite l'inversion d'un système linéaire, les numériciens de l'ONERA ont développé le solveur GMRES emboîté au sein du solveur aérodynamique *elsA* pour la résolution des systèmes linéaires adjoints aérodynamiques et couplés aéro-structures. [26, 25, 32, 33]. Des techniques de préconditionnement ont également été mises en place pour améliorer considérablement la robustesse du solveur GMRES emboîté. Un préconditionneur typique est la méthode de relaxation LU-SGS utilisé au sein du solveur historique BWE-Newton pour la résolution par avance en temps fictif. La Figure 2.9 illustre la convergence d'un système adjoint couplé aéro-structure au moyen de l'approche partitionnée. La résolution est effectuée en appliquant un solveur GMRES emboîté préconditionné par un LU-SGS (solveur GMRES préconditionné par un GMRES lui-même préconditionné par un LU-SGS). La méthode LUSGS dépend essentiellement de deux paramètres : le CFL qui contrôle le pseudo-pas de temps  $\delta\tau$  et le nombre de cycles de relaxation. Chaque cycle de relaxation est constitué de deux étapes descente-remontées. On observe clairement que la convergence est ralentie par l'apparition d'une succession de plateaux au début de chaque nouveau couplage fluide-structure. Ceci est dû essentiellement à la mise à jour du second membre structure. Nous observons à la fin de la convergence la disparition des plateaux et une convergence typique d'un solveur GMRES. En effet, du fait que le second membre converge vers une valeur constante, on se retrouve à résoudre un système adjoint purement fluide, c'est-à-dire un seul système linéaire. L'objectif des chapitres suivants est d'accélérer la convergence en éliminant les effets plateaux.



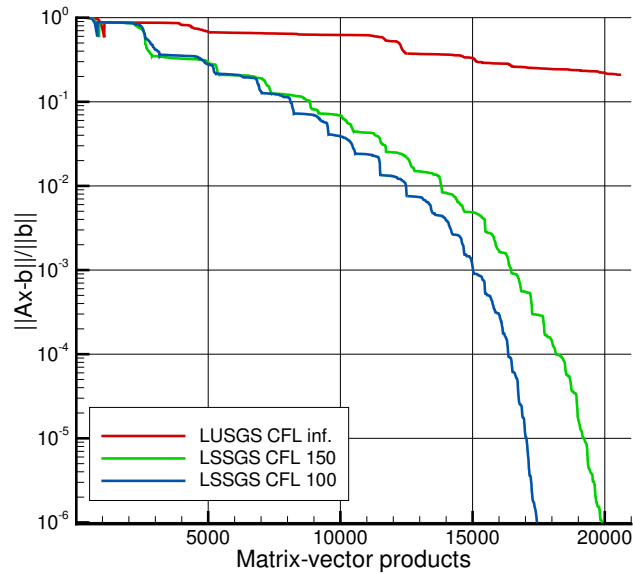


FIGURE 2.9 : Historique de convergence de la norme du résidu fluide pour l’approche partitionné. GMRES emboîté préconditionné par un LUSGS et impact du CFL sur la convergence du système adjoint.

## Bilan :

Dans ce chapitre, nous avons mené une étude bibliographique qui fait ressortir l’intérêt des méthodes de Krylov emboîtées à résoudre de manière robuste des systèmes adjoints couplés aéro-structures. Plusieurs études numériques ont effectivement démontré la capacité des méthodes GMRES emboîtées à traiter efficacement des systèmes adjoints fortement couplés. En outre, nous avons démontré l’intérêt de l’approche monolithique à résoudre de tels systèmes grâce à leurs propriétés remarquables à éliminer du problème la contrainte du couplage fort entre le fluide et la structure dont souffre justement l’approche partitionnée. Ces études ont démontré également le rôle majeur des stratégies de préconditionnement robustes dans les performances du solveur GMRES. Des préconditionneurs tels que la méthode de relaxation LUSGS et la factorisation LU incomplète par bloc ont été mises en œuvre dans les études abordées dans ce chapitre.

## Chapitre 3

# Développement de solveurs de Krylov emboîtés associés à des stratégies de préconditionnement robustes

### Sommaire

---

<b>3.1</b>	<b>Méthode de projection - Méthode de Krylov</b> . . . . .	<b>51</b>
3.1.1	La méthode généralisée du résidu minimal (GMRES) . . . . .	51
3.1.2	Procédure d'orthogonalisation - Méthode d'Arnoldi . . . . .	52
<b>3.2</b>	<b>Stratégies de préconditionnement</b> . . . . .	<b>53</b>
3.2.1	Méthode LU-SGS couplée à la méthode de Schwarz additive . . . . .	54
3.2.2	Factorisation LU incomplète . . . . .	56
3.2.3	La méthode GMRES flexible (FGMRES) . . . . .	56
3.2.4	Matrices de préconditionnement . . . . .	58
<b>3.3</b>	<b>Approche de déflation spectrale</b> . . . . .	<b>60</b>
3.3.1	Calcul des vecteurs de Ritz harmoniques . . . . .	61
3.3.2	Description de l'algorithme FGMRES-DR . . . . .	63
<b>3.4</b>	<b>Résultats numériques</b> . . . . .	<b>64</b>
3.4.1	Profil OAT15A . . . . .	65
3.4.2	Aile M6 ONERA . . . . .	69

---

## Introduction

L'objectif de ce chapitre est d'améliorer la résolution des systèmes adjoints fluides pour des problèmes numériques raides. Pour cela, nous avons mis en place un solveur GMRES modulaire associé à des stratégies de préconditionnement robustes. Une attention particulière est portée sur le solveur GMRES emboîté qui fait partie de la classe des méthodes de Krylov flexibles. Ces méthodes possèdent en effet des propriétés de convergence remarquables par rapport au GMRES standard. Dans la section 3.1, nous exposons succinctement les principes fondamentaux sur lesquels se basent les méthodes de Krylov en général et le GMRES en particulier. La section 3.2 présente les différentes stratégies de préconditionnement. Une analyse de convergence est effectuée pour le cas du GMRES emboîté. La section 3.3 expose la technique de déflation spectrale par augmentation qui a l'avantage d'accélérer la convergence des systèmes linéaires mal conditionnés. Pour finir, une analyse de performance du solveur est réalisée dans la section 3.4. Deux cas tests développés par l'ONERA sont considérés : le profil supercritique OAT15A bidimensionnel et l'aile M6 tridimensionnel. Dans les deux cas, l'écoulement stationnaire est transsonique visqueux.

### 3.1 Méthode de projection - Méthode de Krylov

On souhaite résoudre le système linéaire suivant :

$$\mathbf{Ax} = \mathbf{b}, \quad \mathbf{A} \in \mathbb{R}^{N \times N}, \quad \mathbf{b}, \mathbf{x} \in \mathbb{R}^N \quad (3.1)$$

Les méthodes de Krylov font partie d'une classe de méthodes bien connue dont l'objectif est de calculer une correction  $\mathbf{t}_m$  de l'erreur initiale, notée  $\mathbf{t}$ , d'un système linéaire donné. Pour une solution initiale  $\mathbf{x}_0$  donnée, la solution de (3.1) satisfait la relation  $\mathbf{x} = \mathbf{x}_0 + \mathbf{t} \approx \mathbf{x}_0 + \mathbf{t}_m$ , c'est-à-dire, nous cherchons à approximer  $\mathbf{t}$  qui vérifie

$$\mathbf{A}(\mathbf{x}_0 + \mathbf{t}) = \mathbf{b} \quad (3.2)$$

Pour résoudre le système linéaire (3.2) avec  $\mathbf{r}_0 = \mathbf{b} - \mathbf{Ax}_0$ , on construit un espace de correction, appelé espace de Krylov, de manière itérative

$$\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0) = \text{Vect}\{\mathbf{r}_0, \mathbf{Ar}_0, \dots, \mathbf{A}^{m-1}\mathbf{r}_0\} \quad (3.3)$$

À l'itération  $m$ ,  $\mathbf{t}_m \in \mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$  est calculé en imposant une contrainte sur le résidu  $\mathbf{r}_m$  et  $\mathbf{x}_m = \mathbf{x}_0 + \mathbf{t}_m$  est l'approximation de la solution dans l'espace affine  $\mathbf{x}_0 + \mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$ . La contrainte imposée sur le résidu est une contrainte d'orthogonalité appelée condition de *Petrov-Galerkin*. Parmi ces méthodes, nous porterons notre attention sur la méthode GMRES [8] du fait de ses bonnes propriétés de convergence.

#### 3.1.1 La méthode généralisée du résidu minimal (GMRES)

La méthode GMRES [8] permet de résoudre des systèmes linéaires non symétriques vérifiant les deux contraintes suivantes :

$$\mathbf{x}_m \in \mathbf{x}_0 + \mathcal{K}_m(\mathbf{A}, \mathbf{r}_0), \quad (3.4)$$

$$\mathbf{r}_m \perp \mathbf{A}\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0) \quad (3.5)$$

Toutes les méthodes vérifiant l'équation (3.5) sont appelées des méthodes de projection résiduelle. En effet, l'équation (3.5) définit la projection (oblique ou orthogonale) du résidu sur l'espace  $\mathbf{A}\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$  qui est appelé espace de contrainte. Cette contrainte équivaut à résoudre le problème de minimisation résiduel :

$$\mathbf{t}_m = \underset{\delta \in \mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)}{\text{argmin}} \|\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \delta)\|, \quad (3.6)$$

impliquant les approximations

$$\mathbf{t}_m = \mathbf{P}_{\mathcal{K}_m} \mathbf{t} \quad \text{et} \quad \mathbf{r}_m = (\mathbf{I} - \mathbf{Q}_{\mathcal{K}_m})\mathbf{r}_0 \quad (3.7)$$

### 3. DÉVELOPPEMENT DE SOLVEURS DE KRYLOV EMBOÎTÉS ASSOCIÉS À DES STRATÉGIES DE PRÉCONDITIONNEMENT ROBUSTES

où  $\mathbf{P}_{\mathcal{K}_m}$  est la projection  $(\mathbf{A}^t \mathbf{A})$ -orthogonale sur  $\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$  et  $\mathbf{Q}_{\mathcal{K}_m}$  est la projection orthogonale sur  $\mathbf{A}\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$ .

Les opérateurs  $\mathbf{P}_{\mathcal{K}_m}$  et  $\mathbf{Q}_{\mathcal{K}_m}$  sont définis par les expressions suivantes :

$$\mathbf{P}_{\mathcal{K}_m} = \mathbf{V}_m (\mathbf{V}_m^t \mathbf{A}^t \mathbf{A} \mathbf{V}_m)^{-1} \mathbf{V}_m^t \mathbf{A}^t \mathbf{A} \quad (3.8)$$

$$\mathbf{Q}_{\mathcal{K}_m} = \mathbf{A} \mathbf{V}_m (\mathbf{V}_m^t \mathbf{A}^t \mathbf{A} \mathbf{V}_m)^{-1} \mathbf{V}_m^t \mathbf{A}^t \quad (3.9)$$

Une base naturelle de  $\mathcal{K}_m$  pour le calcul de  $\mathbf{x}_m$  est définie par (3.3). Or, cette base n'est pratiquement pas utilisable car les vecteurs  $\mathbf{A}^m \mathbf{r}_0$  deviennent rapidement colinéaires ce qui donne lieu à une dégénérescence numérique. Pour éviter cela, une solution consiste à implémenter une procédure d'orthonormalisation, appelée procédure d'Arnoldi, afin d'obtenir une base orthonormale de  $\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$  [34] qui est représentée par la matrice  $\mathbf{V}_m$ .

#### 3.1.2 Procédure d'orthogonalisation - Méthode d'Arnoldi

Cette procédure construit itérativement une base orthonormale de  $\mathcal{K}_{m+1}(\mathbf{A}, \mathbf{r}_0)$ . Elle fait appel notamment à la procédure de Gram-Schmidt modifiée [35]. À l'itération  $m$ , on calcule

$$\mathbf{v}_{m+1} h_{m+1,m} = \mathbf{A} \mathbf{v}_m - \sum_{i=1}^m \mathbf{v}_i h_{i,m} \quad \text{avec} \quad h_{i,m} = \mathbf{v}_i^t \mathbf{A} \mathbf{v}_m \quad \text{et} \quad h_{m+1,m} = \|\mathbf{A} \mathbf{v}_m - \sum_{i=1}^m \mathbf{v}_i h_{i,m}\|, \quad (3.10)$$

tel que  $\mathbf{v}_{m+1}$  soit de norme unitaire. La procédure construit la matrice

$$\mathbf{V}_{m+1} = [\mathbf{v}_1 \quad \mathbf{v}_2 \quad \cdots \quad \mathbf{v}_{m+1}] \in \mathbb{R}^{N \times (m+1)} \quad \text{et} \quad \bar{\mathbf{H}}_m \in \mathbb{R}^{(m+1) \times m} \quad (3.11)$$

tel que  $\mathbf{V}_{m+1}^t \mathbf{V}_{m+1} = \mathbf{I}_{m+1}$ .  $\bar{\mathbf{H}}_m$  est une matrice de Hessenberg supérieure avec comme composantes  $h_{i,m}$ . Elle vérifie la relation d'Arnoldi fondamentale :

$$\mathbf{A} \mathbf{V}_m = \mathbf{V}_{m+1} \bar{\mathbf{H}}_m = \mathbf{V}_m \mathbf{H}_m + h_{m+1,m} \mathbf{v}_{m+1} \mathbf{e}_m^t, \quad (3.12)$$

où  $\mathbf{H}_m \in \mathbb{R}^{m \times m}$  est la sous-matrice constituée des  $m$  premières lignes de  $\bar{\mathbf{H}}_m$  et  $\mathbf{e}_m$  le vecteur  $m$  de la base canonique de  $\mathbb{R}^N$ . Cette relation permet de réduire le problème de minimisation (3.6) en un problème aux moindres carrés de taille réduite  $(m+1) \times m$  :

$$\mathbf{y}_m = \underset{\mathbf{y} \in \mathbb{R}^m}{\operatorname{argmin}} \|\|\mathbf{r}_0\| \mathbf{e}_1 - \bar{\mathbf{H}}_m \mathbf{y}\|, \quad \text{où} \quad \mathbf{t}_m = \mathbf{V}_m \mathbf{y}_m \quad (3.13)$$

Nous pouvons facilement vérifier à travers les équations ci-dessous que la méthode

GMRES vérifie bien la condition d'orthogonalité (3.5). En effet,

$$\mathbf{r}_i = \mathbf{b} - \mathbf{A}\mathbf{x}_i \quad (3.14)$$

$$= \mathbf{V}_{m+1}(\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}_m) \quad \text{avec} \quad \mathbf{c} = \mathbf{V}_{m+1}^T \mathbf{r}_i \quad (3.15)$$

et

$$\mathbf{y}_m = \underset{\mathbf{y}}{\operatorname{argmin}} \|\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}\| \quad (3.16)$$

$$= \bar{\mathbf{H}}_m^\dagger \mathbf{c} \quad \text{avec} \quad \bar{\mathbf{H}}_m^\dagger = (\bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m)^{-1} \bar{\mathbf{H}}_m^T \quad (3.17)$$

En utilisant la relation d'Arnoldi (3.12), on a

$$\begin{aligned} (\mathbf{A}\mathbf{V}_m)^T \mathbf{r}_i &= (\mathbf{A}\mathbf{V}_m)^T \mathbf{V}_{m+1}(\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}_m) \\ &= \bar{\mathbf{H}}_m^T \mathbf{V}_{m+1}^T \mathbf{V}_{m+1}(\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}_m) \\ &= \bar{\mathbf{H}}_m^T (\mathbf{c} - \bar{\mathbf{H}}_m \bar{\mathbf{H}}_m^\dagger \mathbf{c}) \\ &= \mathbf{0} \end{aligned}$$

## 3.2 Stratégies de préconditionnement

La version standard du GMRES est pratiquement inutilisable essentiellement à cause de la limitation mémoire due au stockage des vecteurs de Krylov au cours des itérations. L'approche classique est d'adopter une stratégie de redémarrage du GMRES qui consiste à fixer une taille maximale  $m$  de l'espace de Krylov  $\mathcal{K}_m$ , puis de redémarrer le GMRES à partir de la solution  $\mathbf{x}_m$  du cycle précédent. La méthode est dorénavant notée GMRES( $m$ ).

Bien que la méthode GMRES( $m$ ) ait des propriétés de convergence satisfaisantes, elle souffre d'un manque de robustesse qui se traduit par l'apparition d'un plateau (stagnation de convergence) suite au redémarrage. Cette stagnation est liée à la distribution spectrale de l'opérateur  $\mathbf{A}$ . Dans le domaine de la CFD, les systèmes linéaires sont généralement mal conditionnés. Les expériences numériques ont montré que les plus petites valeurs propres sont très souvent responsables de cette stagnation [36]. Cette constatation numérique repose sur une justification théorique en ce qui concerne les matrices symétriques définies positives. Pour les matrices plus générales, un facteur important est le degré de non-normalité. Selon la nature du problème, il est tout à fait envisageable que d'autres valeurs propres aient un effet plus considérable sur la convergence. L'inconvénient avec ces valeurs propres est qu'elles sont difficilement captées par les méthodes de Krylov. Le lecteur peut se référer à [37]. Ces dernières ont tendance à capter plus facilement les plus grandes valeurs propres de l'opérateur comme nous l'expliquerons dans la section suivante.

L'objectif principal du préconditionneur est de "décaler" les valeurs propres en question autour de 1, ce qui aura pour effet d'améliorer le conditionnement de l'opérateur  $\mathbf{A}$ .

### 3. DÉVELOPPEMENT DE SOLVEURS DE KRYLOV EMBOÎTÉS ASSOCIÉS À DES STRATÉGIES DE PRÉCONDITIONNEMENT ROBUSTES

---

Elle sert également à créer des clusters de valeurs propres. Le cas idéal serait de placer le spectre entier sur la droite réelle égale à 1, ce qui revient à choisir  $\mathbf{M}^{-1} = \mathbf{A}^{-1}$ . Cette dernière correspond à l'utilisation d'une méthode directe pour inverser  $\mathbf{A}$ . La méthode GMRES converge dans ce cas en une seule itération. Dans de telles conditions, le préconditionneur serait extrêmement coûteux en termes de ressource mémoire et de calcul. Cette option n'est évidemment pas envisageable puisqu'au final, on ne stocke jamais la matrice  $\mathbf{A}$ . Soulignons que l'opérateur de préconditionnement est appliqué avant chaque produit matrice-vecteur. Les résolutions avec l'algorithme GMRES nécessitent des centaines voire des milliers d'itérations. Il est donc essentiel de trouver un compromis entre robustesse et coût mémoire/temps de calcul.

Pour répondre à un tel compromis, l'idée consiste d'une part à construire un préconditionneur qui soit une bonne approximation de la matrice  $\mathbf{A}$ , et d'autre part à faire un choix stratégique de la méthode d'inversion du préconditionneur. Le premier critère permet d'assurer l'aspect robustesse/coût mémoire tandis que le deuxième garantit l'aspect robustesse/coût CPU. Parmi ces méthodes, il y a les méthodes de relaxation de type LU-SGS et les méthodes de factorisation incomplète version scalaire ILU et version par bloc BILU.

#### 3.2.1 Méthode LU-SGS couplée à la méthode de Schwarz additive

C'est une méthode itérative combinant une factorisation  $\mathbf{LU}$  avec une méthode de relaxation de type SSOR[37] qui dépend d'un paramètre de relaxation  $\omega$ . La méthode LU-SGS (*Lower Upper Symmetric Gauss-Seidel*) est obtenue en posant  $\omega = 1$ . Dans un contexte de préconditionnement, nous cherchons une approximation de la solution du système linéaire  $\mathbf{M}\mathbf{x} = \mathbf{v}$  où  $\mathbf{v}$  est un vecteur de Krylov. Dans la phase de factorisation, l'opérateur implicite  $\mathbf{M}$  est décomposé en une matrice triangulaire inférieur  $\mathbf{L}$ , une matrice diagonale  $\mathbf{D}$  et une matrice triangulaire supérieure  $\mathbf{U}$  :  $\mathbf{M} = \mathbf{L} + \mathbf{D} + \mathbf{U}$ . La solution du système est alors approchée par la méthode de relaxation [17, 11] qui consiste à effectuer un balayage amont et aval. Chaque cycle de relaxation est constitué de deux étapes :

$$(\mathbf{L} + \mathbf{D})\mathbf{x}^{(p+\frac{1}{2})} = \mathbf{b} - \mathbf{U}\mathbf{x}^{(p)} \quad (3.18)$$

$$(\mathbf{U} + \mathbf{D})\mathbf{x}^{(p+1)} = \mathbf{b} - \mathbf{L}\mathbf{x}^{(p+\frac{1}{2})} \quad (3.19)$$

où  $p$  indique le numéro du cycle LU-SGS, en démarrant par  $\mathbf{x}^{(0)}=0$ . Dans un contexte de préconditionnement, 2 à 4 cycles de relaxation sont suffisants pour obtenir une solution approchée acceptable. Pour un cycle de relaxation, il est possible d'explicitier l'opérateur de préconditionnement correspondant  $\mathbf{M}_{SGS} = (\mathbf{L} + \mathbf{D})\mathbf{D}^{-1}(\mathbf{U} + \mathbf{D})$ . Comme nous traitons un problème structuré par bloc, l'algorithme LU-SGS par bloc a été implémenté donnant ainsi une matrice  $\mathbf{D}$  diagonale par bloc. Nous soulignons que le caractère diagonal dominant de  $\mathbf{M}_{SGS}$  est une condition nécessaire et suffisante pour assurer la convergence de la méthode LU-SGS. Néanmoins, dans le cas d'une linéarisation du modèle de turbulence, les résultats numériques montrent qu'il est bénéfique d'ajouter un terme diagonal de la

forme  $\mathbf{I}/\Delta\tau$ , où  $\mathbf{I}$  est la matrice identité et où  $\Delta\tau$  est homogène à un pseudo-pas de temps local intervenant typiquement lors d'une phase implicite d'un solveur de Newton inexact non linéaire. Le pseudo-pas de temps local est calculé à partir de la condition de stabilité Courant-Friedrichs-Lewy (CFL) et du champ fluide stationnaire. En choisissant la bonne valeur de CFL, il est ainsi possible d'accélérer de manière drastique la convergence du solveur adjoint.

Dans un contexte de décomposition de domaine, la partie du second membre de (3.18, 3.19) associée aux cellules de recouvrement est mise à jour à chaque cycle afin de prendre en compte les informations issues des blocs voisins. Nous avons en fait combiné la méthode LU-SGS à une méthode de Schwarz Additive Restrictive (RAS). Selon l'ordre des blocs, le second membre est construit à partir de la solution à l'étape précédente ou à partir des blocs dont la solution est déjà à jour. Dans la suite, nous noterons LU-SGS( $N_s$ ,  $CFL$ ) où  $N_s$  correspond au nombre de cycles de relaxation. L'application de la méthode RAS permet au LU-SGS d'être nativement parallèle. La Figure 3.1 illustre comment l'échange d'information se réalise entre deux blocs partageant la même interface. Dans cet exemple, nous avons considéré deux rangées de cellules fictives. Le nombre de rangées dépend de l'ordre de discrétisation des flux convectifs. Dans le cas du LU-SGS, nous appliquerons une discrétisation d'ordre un des flux convectifs, donc une seule rangée sera considérée.

Dans le cas du GMRES, le préconditionneur LU-SGS couplé à une méthode de Schwarz additive effectue la résolution suivante :

$$\begin{cases} \mathbf{z}_j^{(p)} = \mathbf{M}^{-1}\mathbf{v}_j^{(p)} & \text{dans } \Omega_p, \\ \mathbf{z}_{j|\Gamma_p}^{(p)} = \mathbf{z}_j^{(q)} \end{cases} \quad \begin{cases} \mathbf{z}_j^{(q)} = \mathbf{M}^{-1}\mathbf{v}_j^{(q)} & \text{dans } \Omega_q, \\ \mathbf{z}_{j|\Gamma_p}^{(q)} = \mathbf{z}_j^{(p)} \end{cases}$$

avec  $\Omega_{p,q}$  deux blocs ayant une interface commune  $\Gamma = \Gamma_p = \Gamma_q$ .

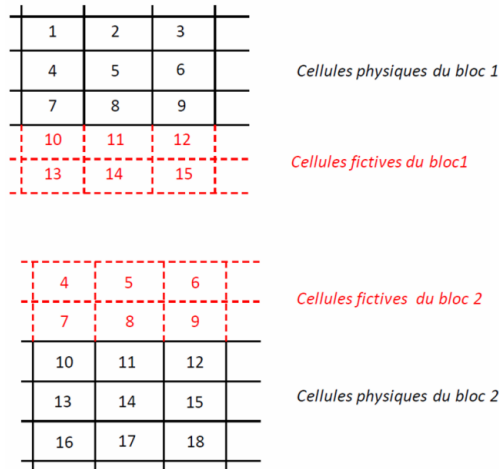


FIGURE 3.1 : Échange d'incrément de la solution entre le Bloc 1 et le Bloc 2.



### 3.2.2 Factorisation LU incomplète

Une factorisation incomplète [38] consiste à calculer une approximation de la factorisation de  $\mathbf{A}$ . Cette approximation est basée sur une stratégie d'élimination qui varie selon différents critères tels que la position des coefficients, leur valeur absolue, ou la combinaison des deux.

L'approche standard, connue sous le nom de ILU(0), consiste à ne calculer que les termes qui sont aux emplacements des termes non nuls de  $\mathbf{A}$ . Autrement dit, les matrices  $\bar{\mathbf{L}}$  et  $\bar{\mathbf{U}}$  de la factorisation incomplète possèdent respectivement la partie triangulaire inférieure et supérieure de  $\mathbf{A}$ . L'inconvénient majeur du ILU(0) est son incapacité à détecter d'éventuels pivots nuls, ce qui rend cette méthode impraticable dans certaines situations. Le bloc ILU(0) (BILU(0)) résout justement ce problème en inversant une matrice de petite taille. Le BILU(0) est également connu pour être plus robuste et stable numériquement. Il existe également le BILU( $p$ ) qui permet un certain remplissage de  $\bar{\mathbf{L}}$  et  $\bar{\mathbf{U}}$  et où  $p$  est un paramètre de contrôle du niveau de remplissage.

Une autre stratégie consiste à éliminer les termes dont l'amplitude est inférieure à un certain seuil noté  $\tau$ . Cette stratégie est notée ILUT( $\tau$ ) [39]. L'élimination des termes dans la factorisation ILUT( $\tau$ ) se fait à partir du critère suivant :

$$|\mathbf{A}_{i,j}| < \tau \|\mathbf{A}_{i,:}\| \Rightarrow \mathbf{A}_{i,j} = 0,$$

avec  $\mathbf{A}_{i,:}$  la  $i^{\text{ème}}$  colonne de  $\mathbf{A}$ .

Le ILUT( $\tau, p$ ) est une version améliorée du ILUT( $\tau$ ) qui prend en compte un paramètre supplémentaire. En effet, en plus d'éliminer les termes dont la valeur est inférieure à  $\tau$ , les  $p$  plus grandes valeurs absolues de la ligne courante de  $\bar{\mathbf{L}}$  et  $\bar{\mathbf{U}}$  résultant de ILUT( $\tau$ ) sont préservées. Ceci permet un contrôle du coût mémoire associé à la factorisation.

### 3.2.3 La méthode GMRES flexible (FGMRES)

Le principe du GMRES flexible (FGMRES) [40] consiste à faire varier le préconditionneur à chaque itération  $j$  du processus d'Arnoldi en considérant un opérateur non linéaire variable  $\mathcal{M}_j : \mathbb{R}^N \rightarrow \mathbb{R}^N$ . La récurrence d'Arnoldi s'écrit alors :

$$\mathbf{AZ}_m = \mathbf{V}_{m+1} \bar{\mathbf{H}}_m \tag{3.20}$$

où  $\mathbf{Z}_m = [\mathbf{z}_1, \dots, \mathbf{z}_m] = [\mathcal{M}_1 \mathbf{v}_1, \dots, \mathcal{M}_m \mathbf{v}_m]$  correspond à un deuxième espace qui sera stocké en plus de  $\mathbf{V}_m$ . La solution de l'itération courante  $\mathbf{x}_j$  s'exprime alors comme combinaison linéaire des vecteurs  $\mathbf{z}_j$  par la relation  $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{Z}_j \mathbf{y}_j$  minimisant le résidu  $\mathbf{r}_j = \mathbf{b} - \mathbf{A} \mathbf{x}_j$  sur l'espace  $\mathbf{x}_0 + \text{Vect}\{\mathbf{Z}_j\}$ .

L'opérateur  $\mathcal{M}_j$  se construit généralement au moyen des méthodes algébriques itératives. Parmi celles-ci, on trouve la méthode Jacobi par bloc [37] et la méthode multigrille

algébrique [41]. Bien que cette dernière soit connue pour la résolution des systèmes linéaires de grande taille, elle est également connue pour être particulièrement robuste en tant que préconditionneur. Plusieurs études ont été réalisées à ce sujet et les expériences numériques ont démontré sa capacité à résoudre efficacement des systèmes linéaires pour des problèmes numériques raides [42, 43]. Le principe du multigrille est d'éliminer les composantes associées aux basses fréquences de l'erreur de la solution numérique permettant d'accélérer la convergence.

Dans le cadre des méthodes flexibles, il existe un cas particulier qui est le solveur GMRES emboîté qui consiste à utiliser un solveur de Krylov comme préconditionneur. Les propriétés remarquables de convergence ont été démontrées dans un cadre théorique clair. En effet, V. Simoncini démontre dans [6] que l'espace de Krylov dans lequel est calculé la solution est largement plus grand que celui généré par un GMRES standard. Elle démontre plus spécifiquement que cet espace augmente à chaque itération du GMRES externe mais sous certaines conditions (voir théorème 1 dans [6]). En effet, à chaque itération  $j$  externe du GMRES, chaque vecteur  $\mathbf{z}_j$  est une approximation de la solution du système

$$\mathbf{A}\mathbf{z}_j = \mathbf{w}_j, \quad (3.21)$$

où  $\mathbf{w}_j$  correspond au  $j^{\text{ème}}$  vecteur de Krylov externe. Ceci génère un espace de Krylov interne noté  $\mathcal{K}_{m_j}(\mathbf{A}, \mathbf{w}_j)$  générant ainsi la base orthonormale  $\mathbf{V}_{m_j}^{(j)} = [\mathbf{v}_1^{(j)}, \dots, \mathbf{v}_{m_j}^{(j)}]$ . La matrice  $\mathbf{V}_{m_j}^{(j)}$  satisfait la récurrence d'Arnoldi interne

$$\mathbf{A}\mathbf{V}_{m_j}^{(j)} = \mathbf{V}_{m_j+1}^{(j)}\bar{\mathbf{H}}_{m_j}^{(j)}, \quad (3.22)$$

$$\mathbf{z}_j = \mathbf{V}_{m_j}^{(j)}\mathbf{y}_j, \quad \mathbf{y}_j \in \mathbb{R}^{m_j} \quad (3.23)$$

Sachant que  $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{Z}_j\mathbf{u}_j$  pour un certain  $\mathbf{u}_j \in \mathbb{R}^{m_j}$  et d'après (3.23), on conclut que la solution  $\mathbf{x}_j$  s'exprime comme combinaison linéaire de l'ensemble des  $j$  bases des espaces de Krylov internes. Si on collecte toutes les bases internes dans une seule matrice notée  $\mathcal{B}_j \in \mathbb{R}^{N \times p}$ , on obtient

$$\mathbf{x}_j = \mathbf{x}_0 + \mathcal{B}_j\mathbf{Y}_j\mathbf{u}_j, \quad (3.24)$$

où  $p = \sum_{i=1}^j m_i$  et  $\mathbf{Y}_j = \text{diag}(\mathbf{y}_1, \dots, \mathbf{y}_j) \in \mathbb{R}^{p \times j}$ . Ces propriétés de convergence justifient clairement la robustesse des méthodes emboîtées afin de traiter des problèmes numériques particulièrement raides. Un point important à souligner est que les vecteurs des bases  $\mathcal{B}_j$  ne sont pas linéairement indépendants. Il faut donc un nombre minimal de vecteurs dans la base interne pour assurer une bonne capacité d'exploitation du GMRES emboîté.

Une autre propriété remarquable du GMRES emboîté est sa capacité à résoudre globalement un problème décomposé en sous-domaines. Dans le cas d'un GMRES standard,

l'action du préconditionneur est local au sous-domaine auquel il appartient. Néanmoins, il est possible de globaliser l'effet du préconditionneur sur l'ensemble du domaine de calcul en appliquant la méthode RAS. Dans notre cas, cette méthode a été appliquée au préconditionneur LUSGS. Malgré tout, cette amélioration ne suffit pas pour mettre en concurrence le GMRES emboîté grâce à ses propriétés remarquables. Cette propriété de globalité vient du fait que le préconditionneur résout approximativement le système (3.21) et non  $\mathbf{Mz}_j = \mathbf{w}_j$  comme c'est le cas pour le GMRES standard.

Nous avons mentionné dans l'introduction de ce chapitre l'implémentation modulaire du solveur GMRES. La stratégie d'implémentation a consisté à faire appel à plusieurs routines bas niveau C++ existantes dans le solveur elsA puis de les importer dans un langage de haut niveau tel que le langage Python. Ces routines concernent essentiellement les opérations algébriques telles que le produit matrice-vecteur, le produit scalaire, la phase de préconditionnement et en l'occurrence l'accès aux Jacobiennes approchées et à la relaxation LU-SGS, et le calcul de la norme. Le lecteur pourra se référer à [44] pour plus de détails sur ce type d'implémentation modulaire. Le rôle de l'utilisateur sera ensuite de fournir des arguments d'entrées qui correspondent à des adresses mémoire stockant les vecteurs de Krylov dont la taille correspond à la taille du sous-domaine auquel ils appartiennent. L'avantage de cette approche est le fait de réutiliser des routines existantes qui gèrent toute la complexité du parallélisme et des échanges MPI entre les processeurs. Ceci nous a donc permis de se concentrer plus sur la méthodologie des solveurs de Krylov.

#### 3.2.4 Matrices de préconditionnement

Dans une phase de préconditionnement, deux aspects importants sont à prendre en compte. Le premier est le choix d'un préconditionneur efficace et robuste. Le deuxième aspect est le choix de la méthode de résolution qui s'applique au préconditionneur. Pour le deuxième aspect, le LU-SGS couplé à une méthode de Schwarz additive et le BILU(0) ont été sélectionnés. Il reste à présent la construction d'un préconditionneur. Comme mentionné précédemment, la construction d'un préconditionneur est fortement dépendante du problème physique de départ. L'idée est donc de partir d'une approximation de la matrice du système tout en respectant le compromis robustesse/coût. Pour cela, nous avons deux options selon le type d'approximation qu'on effectue. La Figure 3.2 illustre les différentes discrétisations des flux convectifs possibles en 3D, c'est-à-dire, dans les directions  $i$ ,  $j$  et  $k$  du maillage. La figure de gauche correspond à la discrétisation exacte d'ordre deux des flux convectifs qui correspond au stencil de la matrice du système adjoint. La figure du milieu correspond à une discrétisation d'ordre un et la figure de droite correspond à une discrétisation d'ordre un dans laquelle une approximation a été effectuée sur les flux visqueux.

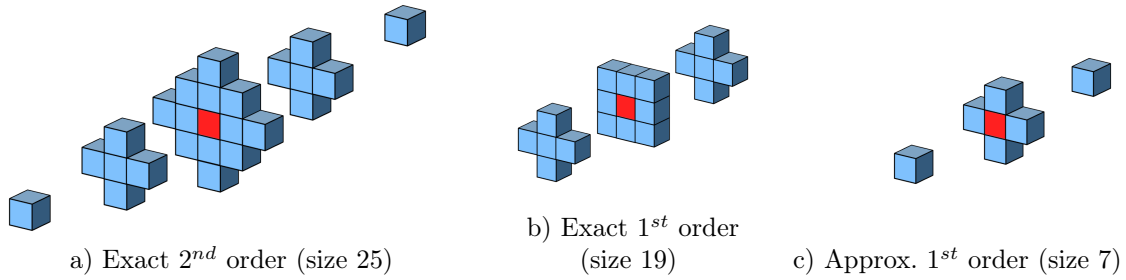


FIGURE 3.2 : Stencils des différentes matrices Jacobiennes.

### 3.2.4.1 Matrice jacobienne approchée d'ordre un

La matrice jacobienne approchée d'ordre un est définie à partir des linéarisations approchées des flux convectifs et visqueux du champ moyen et du champ turbulent définies dans [11] et rappelées ci-dessous :

- flux convectifs moyens : linéarisation décentrée d'ordre 1 de Steger et Warming :

$$d(\mathbf{F}_{i+\frac{1}{2}}^c - \mathbf{F}_{i-\frac{1}{2}}^c) = \mathbf{A}^+(\mathbf{W}_i)d\mathbf{W}_i + \mathbf{A}^-(\mathbf{W}_{i+1})d\mathbf{W}_{i+1}; \quad (3.25)$$

- flux visqueux moyens : linéarisation centrée avec approximation de type "couche mince" [11] et gradients évalués aux centres des cellules :

$$d(\mathbf{F}_{i+\frac{1}{2}}^v - \mathbf{F}_{i-\frac{1}{2}}^v) = \mathbf{M}^v(\mathbf{W}_{i+1})d\mathbf{W}_{i+1} - \mathbf{M}^v(\mathbf{W}_i)d\mathbf{W}_i; \quad (3.26)$$

Les expressions de  $\mathbf{A}^+$  et  $\mathbf{A}^-$  peuvent être trouvées dans [45] ainsi que les expressions pour les flux turbulents. Pour l'expression explicite de  $\mathbf{M}^v$  voir l'article de Peter et al. [11].

Pour l'approche physique "couche mince", des hypothèses doivent être vérifiées : les effets visqueux doivent être prépondérants en proche paroi (nombre de Reynolds élevé) et on doit se situer au voisinage d'une paroi dans la couche limite.

Dans les régions où l'hypothèse de "couche mince" est valide, la partie diffusive des équations de Navier-Stokes vérifient les équations de couche limite : les dérivées transversale et latérale sont négligées et seule la dérivée suivant la direction normale à la face est calculée.

La Figure 3.3 décrit la structure de la matrice Jacobienne approchée d'ordre un. Plus précisément, si on considère une cellule courante d'un maillage structuré d'indice  $(i, j)$ , alors la matrice jacobienne approchée d'ordre un possède un stencil à 5 points : 3 points en  $(i-1, j)$ ,  $(i, j)$  et  $(i+1, j)$  pour la diagonale principale puis 2 points en  $(i, j-1)$  et  $(i, j+1)$  pour la partie extra diagonale. Dans toute la suite, nous noterons  $\mathbf{J}_{O1}^{APP}$  la matrice Jacobienne d'ordre 1 approchée.



Un exemple typique est la méthode du gradient conjugué qui s'applique aux problèmes symétriques définis positifs. L'analyse de convergence de cette méthode montre qu'une borne supérieure qui dépend du conditionnement spectral noté  $\kappa = \lambda_{max}/\lambda_{min}$ , où  $\lambda_{max}$  et  $\lambda_{min}$  sont respectivement la plus grande et la plus petite valeur propre de la matrice. La borne supérieure sur la convergence est  $(\sqrt{\kappa} - 1)/(\sqrt{\kappa} + 1)$ . La borne de convergence du GMRES est un peu plus complexe et dépend des propriétés de l'opérateur. Dans le cas particulier des matrices normales, on peut obtenir des expressions des bornes de convergence qui ne dépendent que des valeurs propres. Quant aux opérateurs non normaux, la borne de convergence ne dépend plus seulement des valeurs propres. [49]. L'effet de cette non normalité se traduit par l'apparition de plateaux de convergence.

Les expérimentations numériques ont démontré que les plus petites valeurs propres sont généralement responsables des problèmes de convergence du GMRES( $m$ ) [36]. Pour comprendre pourquoi ces valeurs propres sont si critiques pour la convergence, il faut revenir à aux fondements de la méthode d'Arnoldi. Cette méthode a été introduite au départ dans le but de transformer une matrice dense en une matrice de Hessenberg. Arnoldi remarque également que le spectre de la matrice de Hessenberg est une bonne approximation des plus grandes valeurs propres de la matrice d'origine. L'espace de Krylov  $\mathcal{K}_m$  contiendra donc naturellement les vecteurs propres de  $\mathbf{A}\mathbf{M}^{-1}$  associés aux plus grandes valeurs propres. Il est donc naturel de recycler les vecteurs propres associés aux plus petites valeurs propres qui sont difficilement captées par la méthode d'Arnoldi.

### 3.3.1 Calcul des vecteurs de Ritz harmoniques

L'étape importante dans une stratégie de déflation est le calcul des vecteurs propres associés aux valeurs propres extrêmes de l'opérateur  $\mathbf{A}$ . D'après [50], les vecteurs de Ritz harmoniques sont une bonne approximation de ces vecteurs propres et se calculent comme combinaison linéaire des vecteurs de Krylov de  $\mathcal{K}_m$  dans le cas de la méthode GMRES standard. Dans le cas flexible, il est possible de choisir une autre base ce qui fera l'objet du chapitre suivant. Les coordonnées associées sont calculées à partir de la résolution d'un problème aux valeurs propres issu de la méthode de Rayleigh-Ritz [50].

#### 3.3.1.1 Méthode de Rayleigh-Ritz

La méthode de Rayleigh-Ritz consiste à résoudre le problème aux valeurs propres de l'opérateur  $\mathbf{A}$  tel que son résidu soit orthogonal à un espace de Krylov généré par  $\mathbf{A}$ . Dans ce qui suit, nous postulons l'existence d'une relation d'Arnoldi de la forme (3.12) avec  $\mathbf{V}_m$  une base orthonormale de  $\mathcal{K}_m$ .

Soit  $\mathcal{S}$  est un sous-espace de  $\mathbb{R}^n$ . On dit que  $(\theta, \mathbf{y}) \in \mathbb{R} \times \mathcal{S}$  est une paire de Ritz de l'opérateur  $\mathbf{A}$  par rapport à  $\mathcal{S}$  si

$$(\mathbf{A}\mathbf{y} - \theta\mathbf{y}) \perp \mathcal{S}, \quad \mathbf{y} \in \mathcal{S} \quad (3.27)$$

Si on pose  $\mathcal{S} = \mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$  et  $\mathbf{y} = \mathbf{V}_m\mathbf{u}$ , tel que  $\mathbf{u} \in \mathbb{R}^m$ , alors on a

$$\begin{aligned} \mathbf{V}_m^T(\mathbf{A}\mathbf{V}_m\mathbf{u} - \theta\mathbf{V}_m\mathbf{u}) &= \mathbf{0} \\ \Leftrightarrow \quad \bar{\mathbf{H}}_m\mathbf{u} &= \theta\mathbf{u} \end{aligned} \quad (3.28)$$

D'après (3.28), on remarque que les valeurs propres de la matrice de Hessenberg  $\bar{\mathbf{H}}_m$  sont des approximations des valeurs propres de l'opérateur  $\mathbf{A}$  appelées aussi valeurs de Ritz. Elles ont la particularité de bien approximer les plus grandes valeurs propres de  $\mathbf{A}$  [51]. Pour obtenir une bonne approximation des plus petites valeurs propres de  $\mathbf{A}$ , on identifie les plus grandes valeurs propres de  $\mathbf{A}^{-1}$ . On définit donc les valeurs de Ritz harmoniques comme étant l'inverse des valeurs de Ritz de  $\mathbf{A}^{-1}$  tel que son résidu soit orthogonal à  $\mathbf{A}\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$ . Autrement dit on a,

$$(\mathbf{A}^{-1}\mathbf{y} - \theta\mathbf{y}) \perp \mathbf{A}\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0), \quad \mathbf{y} \in \mathbf{A}\mathcal{K}_m \quad (3.29)$$

où  $\theta^{-1}$  est défini comme étant une valeur de Ritz harmonique de  $\mathbf{A}$  et  $\mathbf{y} = \mathbf{A}\mathbf{V}_m\mathbf{u}$  étant le vecteur de Ritz harmonique associé à  $\theta^{-1}$ . Ainsi, la relation d'orthogonalité s'écrit

$$\begin{aligned} (\mathbf{A}^{-1}\mathbf{y} - \theta\mathbf{y}) &\perp \mathbf{A}\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0) \\ \Leftrightarrow \quad \mathbf{V}_m^T\mathbf{A}^T(\mathbf{A}^{-1}\mathbf{y} - \theta\mathbf{y}) &= \mathbf{0} \\ \Leftrightarrow \quad \mathbf{V}_m^T\mathbf{A}^T(\mathbf{A}^{-1}\mathbf{A}\mathbf{V}_m\mathbf{u} - \theta\mathbf{A}\mathbf{V}_m\mathbf{u}) &= \mathbf{0} \\ \Leftrightarrow \quad \mathbf{V}_m^T\mathbf{A}^T(\mathbf{V}_m\mathbf{u} - \theta\mathbf{A}\mathbf{V}_m\mathbf{u}) &= \mathbf{0} \end{aligned}$$

On utilisant la relation d'Arnoldi (3.12) et sachant que  $\mathbf{H}_m = \mathbf{V}_m^T\mathbf{V}_{m+1}\bar{\mathbf{H}}_m$ , nous obtenons

$$\begin{aligned} \bar{\mathbf{H}}_m^T\mathbf{V}_{m+1}^T\mathbf{V}_m\mathbf{u} - \theta\bar{\mathbf{H}}_m^T\mathbf{V}_{m+1}^T\mathbf{A}\mathbf{V}_m\mathbf{u} &= \mathbf{0} \\ \Leftrightarrow \quad \mathbf{H}_m^T\mathbf{u} - \theta\bar{\mathbf{H}}_m^T\bar{\mathbf{H}}_m\mathbf{u} &= \mathbf{0} \\ \Leftrightarrow \quad \mathbf{H}_m^T\mathbf{u} &= \theta\bar{\mathbf{H}}_m^T\bar{\mathbf{H}}_m\mathbf{u} \end{aligned} \quad (3.30)$$

La relation (3.30) peut s'écrire

$$\bar{\mathbf{H}}_m^T\bar{\mathbf{H}}_m\mathbf{u} = \theta^{-1}\mathbf{H}_m^T\mathbf{u} \quad (3.31)$$

### 3.3.1.2 Traitement numérique des valeurs propres

Théoriquement, résoudre le problème aux valeurs propres généralisé (3.31) de taille  $m \ll n$  ne pose pas particulièrement de problème. Or, pour des cas numériques raides (ce qui se traduit par un très mauvais conditionnement du système), l'approximation des vecteurs propres devient peu précise introduisant ainsi des erreurs numériques dans l'espace de déflation qui risquent de se propager au cours des cycles du GMRES. En effet, le conditionnement de  $\bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m$  est égal au carré de celui de la matrice de Hessenberg  $\mathbf{H}_m$ .

L'idée est donc de simplifier l'expression (3.31) afin d'améliorer le conditionnement. Cette idée a été introduite pour la première fois par Morgan [36]. Nous obtenons ainsi un problème aux valeurs propres standard :

$$(\mathbf{H}_m + h_{m+1,m}^2 \mathbf{f}_m \mathbf{e}_m^T) \mathbf{u} = \theta \mathbf{u} \quad (3.32)$$

où  $\mathbf{f}_m = \mathbf{H}_m^{-T} \mathbf{e}_m$  représente la dernière colonne de  $\mathbf{H}_m^{-T}$  et  $\mathbf{e}_m$  est le  $m$ -ème vecteur de la base canonique de  $\mathbb{R}^m$ . Le passage de (3.31) à (3.32) est démontré dans l'annexe A.

### 3.3.2 Description de l'algorithme FGMRES-DR

Le premier cycle du FGMRES-DR commence par un cycle complet de FGMRES donnant en sortie le résidu  $\mathbf{r}_m$  (étape 3 de l'algorithme 1). À la fin du premier cycle, on résout le problème aux valeurs propres standard (3.32) donnant en sortie les  $m$  paires de valeurs-vecteurs propres  $(\theta_i, \mathbf{p}_i)$  (étape 6 de l'algorithme 1). On sélectionne ensuite les  $k$  paires associées aux  $k$  plus petites valeurs de Ritz harmoniques  $\theta_i$ . On construit la matrice  $\mathbf{P}_{k+1} \in \mathbb{R}^{m \times (k+1)}$  constituée des  $k$  vecteurs propres  $\mathbf{p}_1, \dots, \mathbf{p}_k$  auxquels on ajoute le résidu projeté  $\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}$  en prenant soin d'ajouter une ligne composée de zéros (étape 7 de l'algorithme 1). On orthonormalise les colonnes de  $\mathbf{P}_{k+1}$  pour donner  $\bar{\mathbf{P}}_{k+1}$  (étape 8 de l'algorithme 1). À ce stade, les vecteurs de Ritz harmoniques sont construits par combinaison linéaire des vecteurs de Krylov du cycle précédant (étape 9 de l'algorithme 1). Le second cycle s'achève en complétant la base de Krylov par l'application de  $(m - k)$  étapes d'Arnoldi (étape 11 de l'algorithme 1). Nous économisons ainsi  $k$  produits par  $\mathbf{A}$  et par  $\mathcal{M}_j$  qui sont remplacés par l'étape de déflation qui est moins coûteuse. L'espace de déflation  $\mathbf{V}_{k+1}^{new}$  a la propriété remarquable de vérifier une relation d'Arnoldi flexible de taille  $k$  [24] :

$$\mathbf{A} \mathbf{Z}_k^{new} = \mathbf{V}_{k+1}^{new} \bar{\mathbf{H}}_k^{new} \quad (3.33)$$

L'algorithme FGMRES-DR est présenté dans l'algorithme ci-dessous. L'algorithme GMRES-DR se déduit facilement en considérant un préconditionneur stationnaire  $\mathcal{M}_j = \mathbf{M}^{-1}$ .



### 3. DÉVELOPPEMENT DE SOLVEURS DE KRYLOV EMBOÎTÉS ASSOCIÉS À DES STRATÉGIES DE PRÉCONDITIONNEMENT ROBUSTES

---



---

**Algorithm 1** FGMRES-DR( $m, m_i, k$ )

---

- 1: **Start** :  $\mathbf{x}_0, \mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, tol, i = 0$ .
  - 2: Soit  $\mathbf{v}_1 = \mathbf{r}_0 / \|\mathbf{r}_0\|$  et  $\beta = \|\mathbf{r}_0\|$ .
  - 3: Appliquer un cycle complet FGMRES( $m, m_i$ ) et construire  $\mathbf{V}_{m+1}, \mathbf{Z}_m, \bar{\mathbf{H}}_m$  et  $\mathbf{r}_m^{(1)}$ .
  - 4: **while**  $\|\mathbf{r}_m^{(i)}\| > tol$  **do**
  - 5:      $i = i + 1$
  - 6:     Calculer  $k$  vecteurs propres  $\mathbf{p}_i$  de  $\mathbf{H}_m + h_{m+1, m}^2 \mathbf{H}_m^{-T} \mathbf{e}_m \mathbf{e}_m^T$  pour les plus petites valeurs propres  $\theta_i$ .
  - 7:     Construire  $\mathbf{P}_k = [\mathbf{p}_1, \dots, \mathbf{p}_k]$  et  $\mathbf{P}_{k+1} = \begin{bmatrix} \mathbf{P}_k \\ \mathbf{0}_{1 \times k} \end{bmatrix}, \mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}$
  - 8:     Calculer la décomposition QR de  $\mathbf{P}_{k+1}$  :  $\mathbf{P}_{k+1} = \bar{\mathbf{P}}_{k+1} \Gamma_{k+1}$
  - 9:      $\mathbf{V}_{k+1}^{new} = \mathbf{V}_{m+1} \bar{\mathbf{P}}_{k+1}, \mathbf{Z}_k^{new} = \mathbf{Z}_m \bar{\mathbf{P}}_k$  and  $\bar{\mathbf{H}}_k^{new} = \bar{\mathbf{P}}_{k+1}^T \bar{\mathbf{H}}_m \bar{\mathbf{P}}_k$ .
  - 10:     Soit  $\bar{\mathbf{H}}_k = \bar{\mathbf{H}}_k^{new}, \mathbf{V}_{k+1} = \mathbf{V}_{k+1}^{new}$  et  $\mathbf{Z}_k = \mathbf{Z}_k^{new}$ .
  - 11:     Appliquer  $(m - k)$  itérations d'Arnoldi à partir de  $\mathbf{V}_{k+1}^{new}, \mathbf{Z}_k^{new}$ , et  $\bar{\mathbf{H}}_k^{new}$  pour construire  $\mathbf{V}_{m+1}, \mathbf{Z}_m$  et  $\bar{\mathbf{H}}_m$ .
  - 12:      $\mathbf{c} = \mathbf{V}_{m+1}^T \mathbf{r}_m^{(i)}$
  - 13:     Résoudre le problème aux moindres carrés  $\min_{\mathbf{y}} \|\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}\|$  pour  $\mathbf{y}^*$
  - 14:     Calculer  $\mathbf{x}_m^{(i)} = \mathbf{x}_m^{(i-1)} + \mathbf{Z}_m \mathbf{y}^*$
  - 15:      $\mathbf{r}_m^{(i)} = \mathbf{r}_m^{(i-1)} - \mathbf{V}_{m+1} \bar{\mathbf{H}}_m \mathbf{y}^*$
  - 16: **end while**
- 

### 3.4 Résultats numériques

Afin d'évaluer la capacité du solveur modulaire GMRES à résoudre efficacement le système adjoint aérodynamique, nous considérons deux cas tests numériques. Le premier correspond au profil d'aile supercritique OAT15A en écoulement stationnaire transsonique visqueux. Le nombre de Mach est fixé à 0.734, le nombre de Reynolds vaut  $6.5 \times 10^6$  et l'angle d'incidence est égal à 1.15 degrés. Afin d'éviter un biais numérique dans le calcul du champ adjoint et du gradient associé, le calcul du champ stationnaire est convergé jusqu'à des résidus numériquement très faibles (réduction de la norme de 10 ordres de grandeur). Le deuxième cas test correspond à l'aile M6 ONERA pour le même type d'écoulement. Le nombre de Mach est fixé à 0.84 pour un nombre de Reynolds qui vaut  $11.7 \times 10^6$  et un angle d'incidence égal à 3.06 degrés. La Table 3.1 décrit les paramètres numériques du solveur FGMRES-DR. Nous soulignons que le critère d'arrêt du GMRES interne est basé sur deux paramètres qui sont la taille de l'espace de Krylov interne  $m_i$  et la tolérance interne qu'on a fixé à 0.5, suffisant pour la phase de préconditionnement. Le GMRES interne s'arrête dès que l'un des deux critères est vérifié.

$m$	60	Taille de l'espace de Krylov externe
$k$	20	Nombre de vecteurs déflatés
tol outer	1.e-9	Tolérance de convergence relative
$m_i$	20	Taille de l'espace de Krylov interne
tol inner	0.5	Tolérance de convergence du GMRES interne

TABLE 3.1 : Paramètres numériques du solveur FGMRES-DR

### 3.4.1 Profil OAT15A

Le profil OAT15A présente un bord de fuite épais imposant un maillage initial structuré d'au moins deux blocs. Le nombre correspondant de cellules est de 158 208. Pour une étude de scalabilité, le maillage est découpé en 16 blocs (cf. Figure 3.5a). Un schéma numérique décentré de type Roe est sélectionné avec une reconstruction des flux convectifs à l'ordre 2 de type MUSCL associé à un limiteur de Van Albada. Le modèle de fluide est de type RANS et la turbulence est modélisée par le modèle à une équation de transport de Spalart Allmaras.

La Figure 3.5b présente le champ stationnaire de la densité volumique. La Figure 3.6 présente la convergence en norme  $L_2$  pour le champ de densité et le champ turbulent. Une décroissance de 10 ordres de grandeur est observée pour la norme  $L_2$  des deux variables.

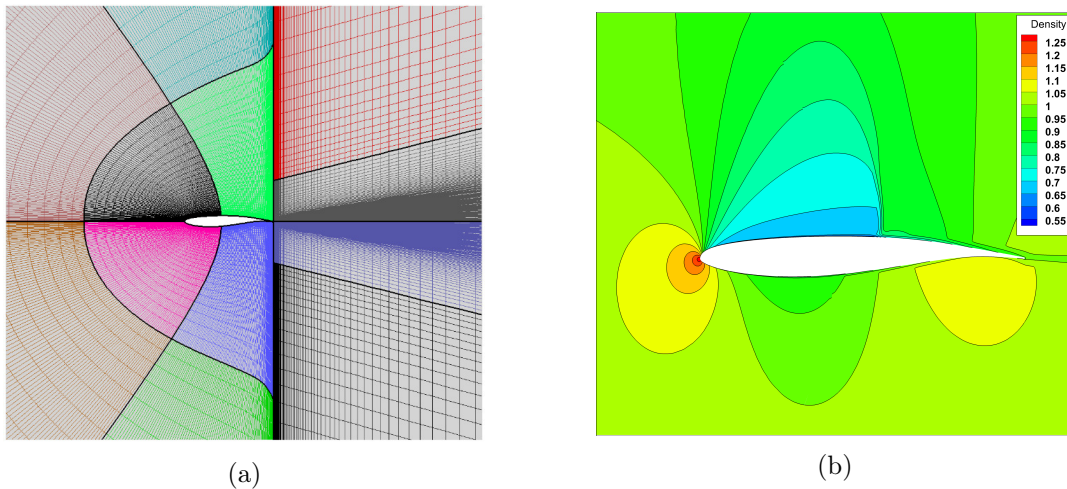


FIGURE 3.5 : Profil OAT15A pour un écoulement fluide transsonique et turbulent : (a) illustre le maillage structuré découpé en 16 blocs tandis que (b) illustre la densité volumique à l'état stationnaire.

#### 3.4.1.1 Résolution du système adjoint

Cette section a pour objectif d'améliorer la résolution du système adjoint aérodynamique (2.19) au moyen du solveur GMRES emboîté associé à la déflation spectrale par augmentation qui est noté par la suite FGMRES-DR( $m, m_i, k$ ). Nous attirons l'attention sur le fait que la résolution de tous les systèmes adjoints sera effectuée en considérant une linéarisation exacte de l'équation turbulente. L'ensemble des calculs est effectué sur 16

### 3. DÉVELOPPEMENT DE SOLVEURS DE KRYLOV EMBOÎTÉS ASSOCIÉS À DES STRATÉGIES DE PRÉCONDITIONNEMENT ROBUSTES

---

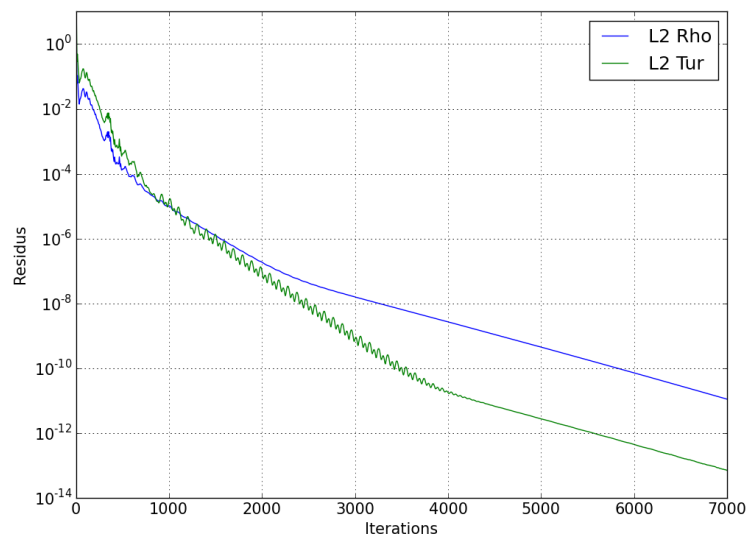


FIGURE 3.6 : Courbe de convergence de la densité et de l'énergie cinétique turbulente.

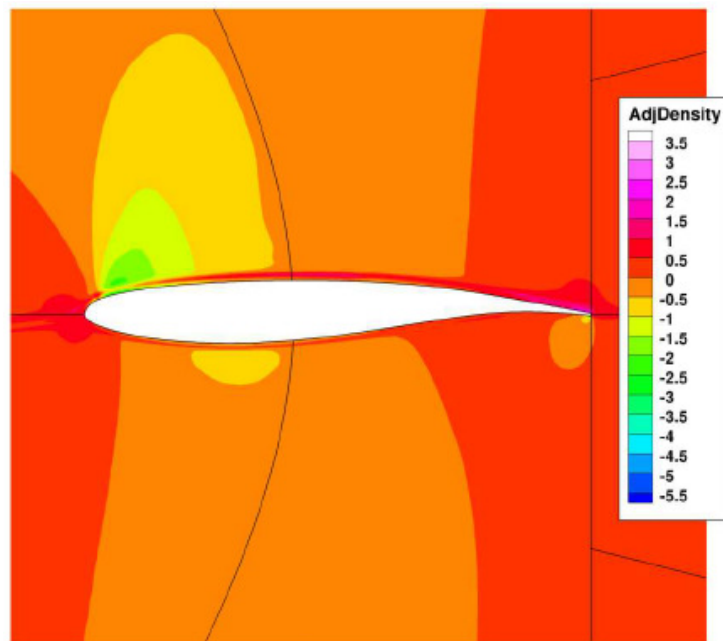


FIGURE 3.7 : Solution adjointe pour la variable densité.

processeurs de sorte que chaque bloc est associé à un seul processeur. La taille du système est d'environ 1 million et le nombre de termes non nuls est estimé à 86 millions.

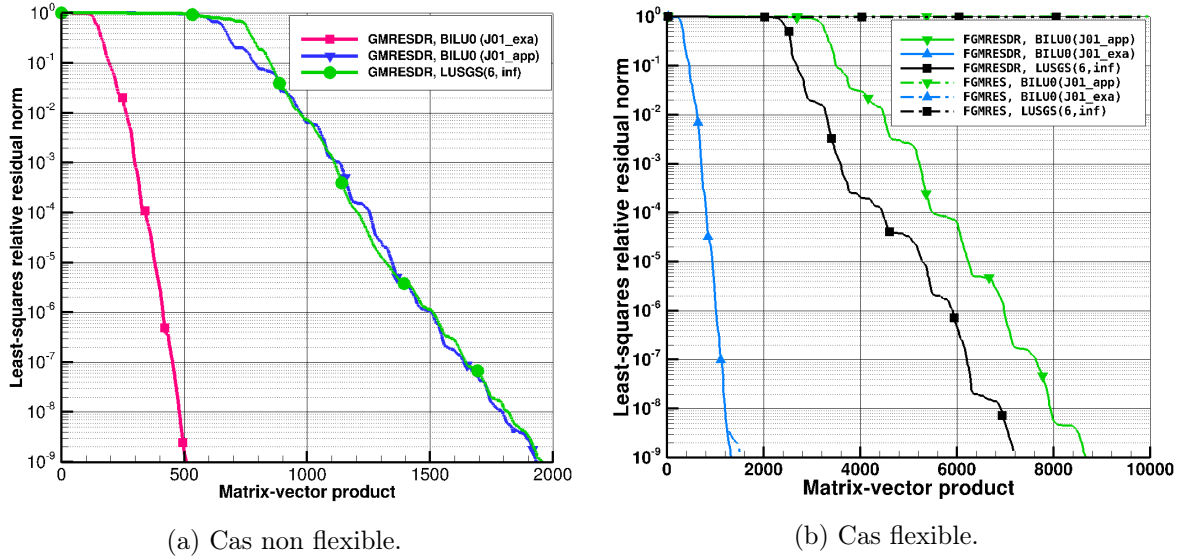


FIGURE 3.8 : Historique de convergence du résidu en fonction du nombre de produits matrices-vecteurs. L’impact de la déflation améliore drastiquement la convergence pour le LU-SGS et le BILU(0) appliqués à  $\mathbf{J}_{O1}^{APP}$ . L’effet du préconditionneur BILU(0) appliqué à  $\mathbf{J}_{O1}^{APP}$  améliore également la robustesse du solveur.

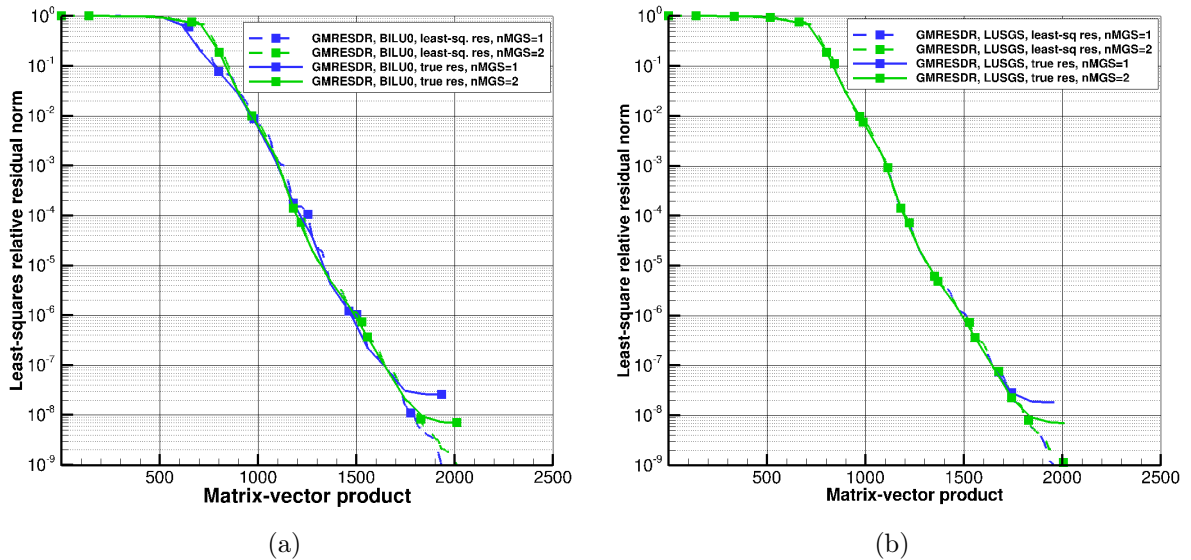


FIGURE 3.9 : Impact de la réorthogonalisation sur la convergence du GMRES-DR. (a) courbe de convergence pour le BILU(0) et (b) courbe de convergence pour le LUSGS.

La Figure 3.8 présente l’historique de convergence de la norme du résidu projeté  $\|\mathbf{c} - \overline{\mathbf{H}}_m \mathbf{y}_m\|$  dans le cas non flexible (Figure 3.8a) et dans le cas flexible (3.8b). Nous remarquons que le solveur FGMRES(60) sans déflation stagne pour le préconditionneur LU-SGS et le BILU(0) appliqué à la Jacobienne approchée d’ordre un  $\mathbf{J}_{O1}^{APP}$ . Une solution naturelle consiste simplement à augmenter la base orthonormale de l’espace de Krylov  $\mathcal{K}_m$ , de dimension  $m$ , afin d’obtenir de nouvelles directions de recherche. Cette solution n’est pas envisageable puisque le coût mémoire devient rapidement important. En appliquant la stratégie de déflation par augmentation, on récupère la convergence du résidu pour un coût

### 3. DÉVELOPPEMENT DE SOLVEURS DE KRYLOV EMBOÎTÉS ASSOCIÉS À DES STRATÉGIES DE PRÉCONDITIONNEMENT ROBUSTES

de calcul raisonnable. Une déflation de 30 % de  $m$  est suffisante pour restaurer la convergence du solveur FMGRES-DR pour ces deux préconditionneurs. On note la convergence rapide du FMGRES pour le BILU(0) appliqué à la Jacobienne exacte d'ordre 1  $\mathbf{J}_{O_1}^{EX}$  qui atteste de sa robustesse à tel point que la déflation n'apporte aucune amélioration supplémentaire. La contrepartie est le coût mémoire de  $\mathbf{J}_{O_1}^{EX}$  qui vaut environ le double de celui de  $\mathbf{J}_{O_1}^{APP}$ . Plus précisément,  $\mathbf{J}_{O_1}^{APP}$  et  $\mathbf{J}_{O_1}^{EX}$  comportent respectivement 25 et 65 millions de termes non nuls. Le tableau 3.2 montre l'impacte de la déflation sur le nombre d'itérations et le nombre de produits par  $\mathbf{A}$  selon le choix du préconditionneur.

Une autre observation numérique intéressante est la décroissance du résidu exact  $\mathbf{r}_m = \mathbf{b} - \mathbf{A}\mathbf{x}_m$ . Celui-ci est calculé à la fin de chaque cycle et est comparé au résidu projeté. Côté performance, le calcul de  $\mathbf{r}_m$  est déconseillé car il coûte un produit par  $\mathbf{A}$ , ce qui peut devenir pénalisant pour un grand nombre de cycles. Le but de son calcul est juste de s'assurer de la convergence réelle du système adjoint. En effet, la décroissance du résidu projeté vers la tolérance prescrite ne garantit pas forcément la convergence du système. Il existe des situations où les deux résidus diffèrent de quelques ordres de grandeur surtout pour des cas numériques raides comme c'est le cas pour le profil OAT15A. L'un des facteurs qui est à l'origine de cette différence est la perte d'orthogonalité de la base de Krylov. La Figure 3.8a illustre bien ce biais numérique. On remarque que le résidu projeté a atteint la tolérance de  $10^{-9}$  tandis que le résidu exact stagne à  $10^{-8}$  pour le LUSGS et le BILU(0) appliqués à  $\mathbf{J}_{O_1}^{APP}$ . Cette différence entre le résidu exact et le résidu projeté s'estompe dans le cas du GMRES emboîté. Ceci est dû en l'occurrence à la propriété de globalisation des méthodes emboîtées comme le montre l'équation (3.21). Afin de comprendre ce phénomène de stagnation, Röllin et al. [52] ont mené une étude sur le comportement du GMRES-DR à tolérance très faible. Les auteurs expliquent que la décroissance du résidu exact est liée à l'erreur commise sur la relation d'Arnoldi (3.12). Cette erreur provient plus précisément de l'approximation des vecteurs de Ritz harmoniques via la relation  $\mathbf{V}_{k+1}^{new} = \mathbf{V}_{m+1}\mathbf{P}_{k+1}$ . L'orthogonalité de l'espace de déflation  $\mathbf{V}_{k+1}^{new}$  est moins précise que si  $\mathbf{V}_{m+1}$  était calculé avec redémarrage classique sans déflation. Dans le cas de la déflation, l'erreur a tendance à se propager d'un cycle à l'autre. Cette erreur peut être quantifiée par l'erreur d'Arnoldi qui vérifie, sous certaines hypothèses, la relation suivante

$$\|\mathbf{E}_{m,k}\| = \|\mathbf{A}\mathbf{V}_k^{new} - \mathbf{V}_{k+1}^{new}\bar{\mathbf{H}}_k^{new}\| \quad (3.34)$$

Trois stratégies sont mises en œuvre pour corriger cette propagation d'erreur : la première, proposée par Morgan [36], consiste à réorthogonaliser le vecteur  $\mathbf{v}_{k+1}$  par rapport aux  $k$  premiers vecteurs. La deuxième consiste à réorthogonaliser entièrement l'espace  $\mathbf{V}_{k+1}^{new}$  et la dernière consiste à faire une double orthogonalisation au cours du processus d'Arnoldi. Les auteurs concluent que la double orthogonalisation est la meilleure option pour améliorer la précision du GMRES-DR. En appliquant cette stratégie, la convergence du

résidu exact est effectivement améliorée (voir Figure 3.9).

$\mathcal{M} = \text{BILU}(0)$	FGMRES(60,20)		FGMRES-DR(60,20,20)	
Matrice Jacobienne	$\mathbf{J}_{O1}^{EX}$	$\mathbf{J}_{O1}^{APP}$	$\mathbf{J}_{O1}^{EX}$	$\mathbf{J}_{O1}^{APP}$
its	74	-	74	546
Mvps	1498	-	1321	8600

TABLE 3.2 : Performance de FGMRES-DR

### 3.4.1.2 Étude de la scalabilité forte

Nous avons démontré dans la section précédente la capacité du FGMRES-DR à résoudre des problèmes raides. Dans un contexte de calcul parallèle, il arrive souvent que les méthodes développées soient certes robustes mais souffrent d'un problème de passage à l'échelle. La scalabilité désigne la capacité d'une méthode à ne pas se dégrader quand le nombre de processeurs augmente. On distingue la scalabilité forte et la scalabilité faible. La première se définit par la variation du temps de résolution d'un problème de taille fixe en fonction du nombre de domaines. La scalabilité forte est parfaite quand le temps de résolution est inversement proportionnel au nombre de domaines. La scalabilité faible se définit par la variation du temps de calcul lorsque la taille du problème augmente linéairement avec le nombre de domaines. La scalabilité faible est atteinte lorsque le temps de résolution est constant. Nous évaluons ici la scalabilité forte du solveur FGMRES-DR.

On observe sur la Figure 3.10 que l'efficacité parallèle atteint jusqu'à 75 % pour le LU-SGS appliqué à  $\mathbf{J}_{O1}^{APP}$  tandis que celle du BILU(0) appliqué à  $\mathbf{J}_{O1}^{EX}$  atteint jusqu'à 80 %. Soulignons que la nature locale du BILU(0) n'a pas d'effet réel sur le comportement du GMRES emboîté (propriété de globalité du préconditionneur externe) ce qui n'est pas le cas du GMRES standard.

## 3.4.2 Aile M6 ONERA

Ce cas test correspond à l'aile M6 ONERA pour un fluide en écoulement stationnaire transsonique et turbulent. Le maillage structuré est composé de  $3.6 \times 10^6$  cellules hexaédriques. La Figure 3.11 présente le maillage et le champ de pression à l'état stationnaire. La taille du problème algébrique est de 21 millions avec un nombre de termes non nuls égal à 2 milliards.

### 3.4.2.1 Résolution du système adjoint

Avant d'illustrer la robustesse du solveur flexible sur ce cas test, nous tentons d'appliquer tout d'abord le GMRES-DR standard. Cet aspect est crucial à mettre en avant car la plupart des auteurs utilisent le GMRES standard dans leurs expériences numériques. Nous

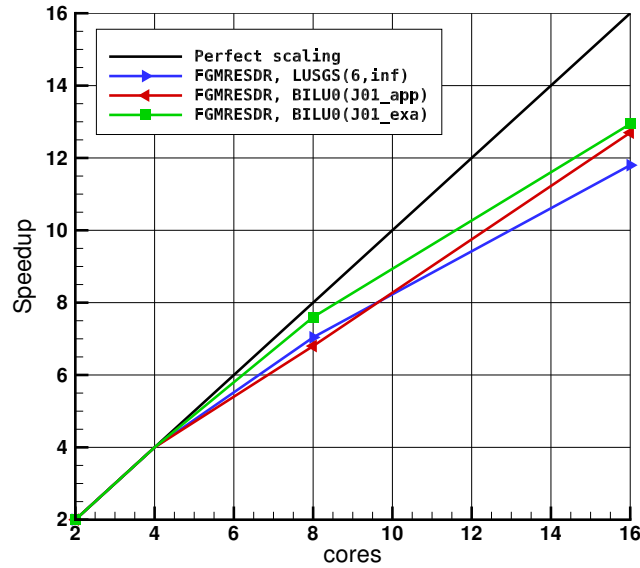


FIGURE 3.10 : Courbe de scalabilité du FGMRES-DR(60,20,20) sur le système adjoint du profil OAT15A.

choisissons les mêmes paramètres numériques que précédemment (Table 3.3). La comparaison des deux solveurs se fera à iso-taille mémoire. Sachant qu'il faut stocker 140 vecteurs de Krylov pour le solveur FGMRES-DR, nous fixons un espace de Krylov de 140 pour le GMRES-DR avec un taux de déflation de 33 %, soit 46 vecteurs.

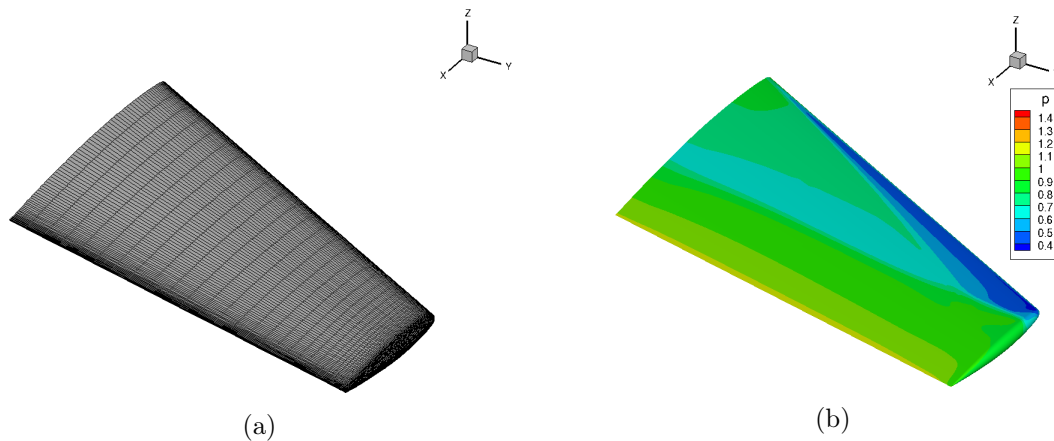


FIGURE 3.11 : Vue tridimensionnelle de l'aile M6 ONERA pour un écoulement fluide transsonique turbulent : (a) et (b) représentent respectivement la peau de l'aile pour un maillage structuré et le champ de pression à l'état stationnaire.

Comme souligné pour le cas OAT15A, le processus de Gram-Schmidt modifiée (MGS) souffre de limitations numériques. Une étape de réorthogonalisation est donc nécessaire au prix d'un coût de calcul plus important. La Figure 3.12 présente l'historique de convergence de la norme des deux résidus pour les préconditionneurs BILU(0) et BILU(1) appliqués à  $\mathbf{J}_{O1}^{EXA}$ . En analysant les courbes, l'application d'une seule étape d'orthogonalisation a

pour effet de faire converger le résidu projeté (courbe pointillée rouge) à la tolérance souhaitée tandis que le résidu exact stagne à  $10^{-6}$  (courbe en trait plein rouge). Encore une fois, ce phénomène s'explique par la propagation des erreurs numériques qui proviennent essentiellement de la stratégie de déflation. En appliquant une double orthogonalisation, on gagne deux ordres de grandeur sur la convergence du résidu exact atteignant donc la tolérance de  $10^{-8}$  (courbe pointillée noire). Le résidu projeté quant à lui converge plus rapidement (courbe noire en trait plein). Il est néanmoins possible d'accélérer la convergence en améliorant la qualité du préconditionneur. Le lecteur pourra se référer à [53] dans lequel une étude numérique comparative de solveurs GMRES est réalisée pour des maillages structurés et non-structurés d'ordre élevé. Toujours dans l'idée d'améliorer la vitesse de convergence du système adjoint, nous avons proposé d'augmenter le taux de remplissage dans la factorisation BILU. Nous avons testé le préconditionneur BILU(1) et les courbes bleues montre finalement un gain marginal.

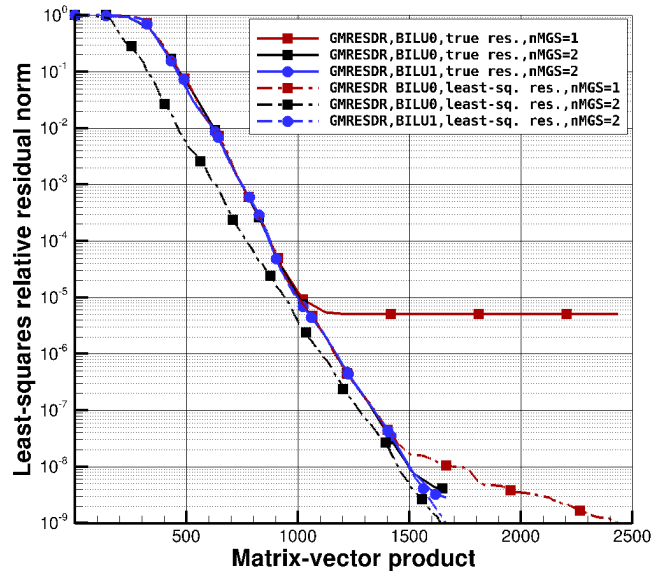


FIGURE 3.12 : Historique de convergence de la norme du résidu projeté (least-sq. res) et du résidu exact (true res.). Impact de la double orthogonalisation de Gram-Schmidt sur le comportement du GMRES-DR. Nous rappelons les paramètres numériques du GMRES-DR :  $m = 140$  et  $k = 46$ .

Pour le GMRES-DR emboîté (FGMRES-DR), les problèmes d'erreurs numériques disparaissent. C'est l'un des intérêts de ce solveur qui, rappelons le, a un comportement global et construit un espace de recherche plus large. Ces propriétés font que le GMRES emboîté est beaucoup moins sensible à ces types d'erreurs. Contrairement au cas OAT15A, le solveur FMRES(60,20) préconditionné par LUSGS ou le BILU(0) appliqués à  $\mathbf{J}_{O1}^{APP}$  converge en 150 itérations pour le BILU(0) contre 377 pour le LUSGS(6,inf). En revanche, l'effet de la déflation est moins sensible avec un gain de 22 itérations pour le LUSGS(6,inf) et de 15 itérations pour le BILU(0). Même si la matrice  $\mathbf{J}_{O1}^{APP}$  est moins robuste que  $\mathbf{J}_{O1}^{EXA}$ , il reste intéressant d'un point de vue ressource mémoire surtout pour des problèmes tridi-



### 3. DÉVELOPPEMENT DE SOLVEURS DE KRYLOV EMBOÎTÉS ASSOCIÉS À DES STRATÉGIES DE PRÉCONDITIONNEMENT ROBUSTES

mensionnels. En effet, la matrice  $\mathbf{J}_{O1}^{APP}$  a un stencil de 7 points en 3D contre un stencil de 9 points pour  $\mathbf{J}_{O1}^{EXA}$ , soit un ratio de 2.7. Ces observations numériques sont résumées dans la Figure 3.13.

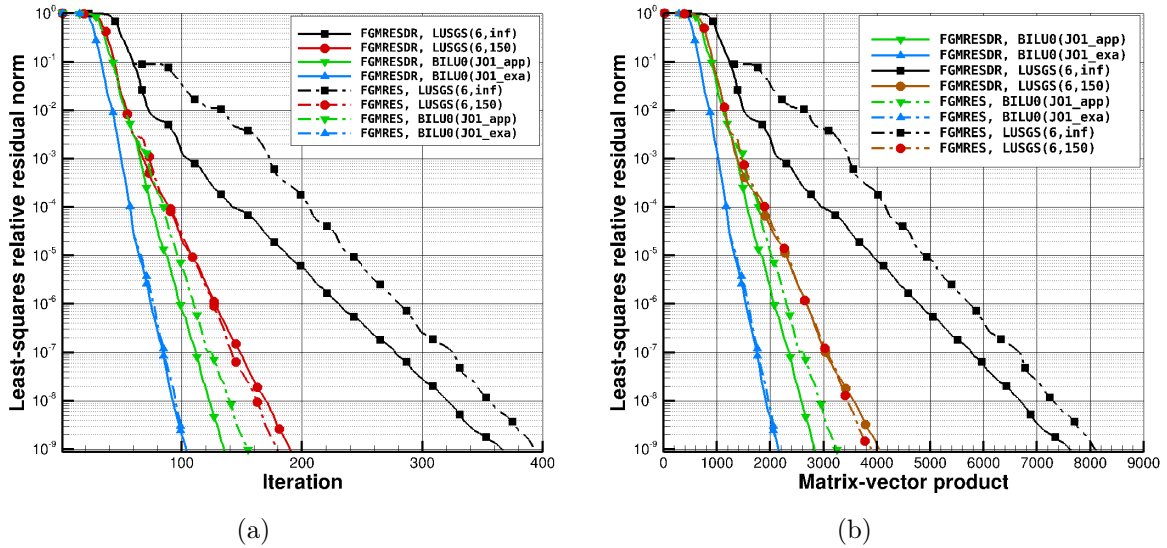


FIGURE 3.13 : Historique de convergence de la norme relative du résidu en nombre d'itérations. Impact de la déflation pour les différents préconditionneurs. L'augmentation de la dominance diagonale du préconditionneur LUSGS en ajoutant une diagonale scalaire  $I/\Delta\tau$  semble être prometteur mais reste moins performante que le BILU. L'impact de la déflation n'est pas concluant pour le BILU appliqué à la jacobienne d'ordre 1 exacte puisqu'un seul cycle est effectué. Nous rappelons les paramètres numériques pour le FGMRES-DR :  $m = 60$ ,  $m_i = 20$  et  $k = 20$ .

Afin de mettre en valeur la stratégie de déflation, nous avons réalisé d'autres simulations en faisant varier la taille de l'espace de Krylov interne avec 6, 10, 15 et 20 vecteurs pour le préconditionneur BILU. La Figure 3.14 montre que la stratégie de déflation devient intéressante en nombre d'itérations lorsque le solveur effectue au moins deux redémarrages. Par contre, le gain en termes de produits matrices-vecteurs est marginal excepté pour la plus petite taille de l'espace de Krylov interne  $m_i = 6$ . Le meilleur compromis pour ce cas spécifique en termes de coût mémoire/performance serait de prendre un espace de Krylov interne de taille 15.

#### 3.4.2.2 Étude de la scalabilité forte

Nous avons montré pour le cas 2D OAT15A à quel point le solveur GMRES emboîté était extensible au sens fort. Cette section a pour but de montrer les mêmes performances sur le cas aile M6. Pour cette étude, nous avons décidé de découper le maillage initial constitué de 42 blocs en 60 blocs. En effet, avec 42 blocs, la répartition de charge entre les différents processeurs est mauvaise (plus de 50 %) ce qui a pour conséquence de biaiser les résultats de scalabilité du solveur. En effectuant plusieurs re-découpages, nous obtenons une répartition quasi-parfaite pour le cas 60 blocs. La Figure 3.15 illustre l'efficacité parallèle du solveur FMGRES-DR. Plus spécifiquement, le solveur atteint 80% en efficacité parallèle

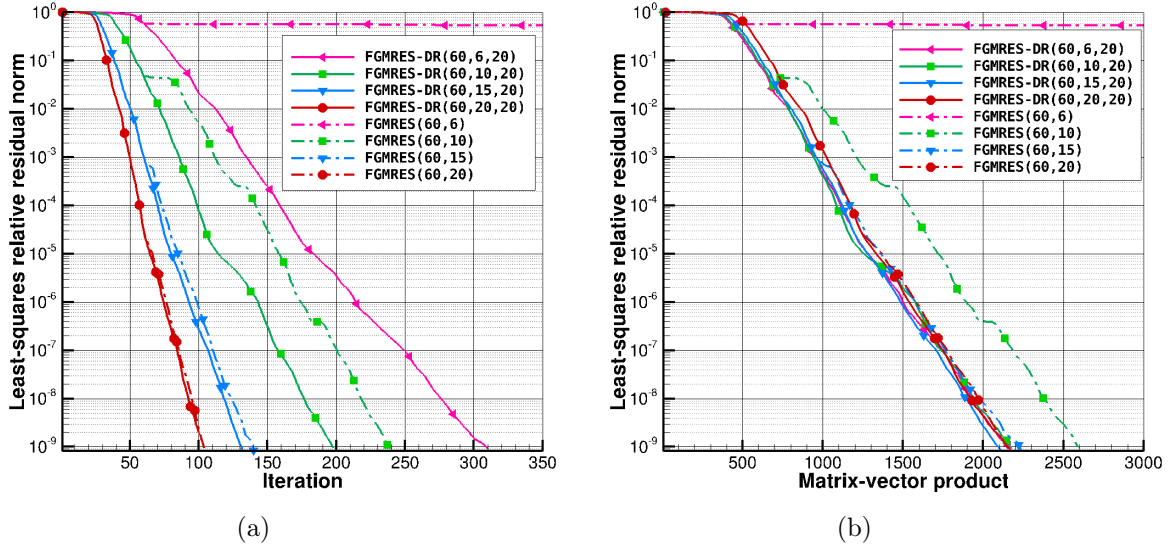


FIGURE 3.14 : Impact de la déflation pour différentes tailles de l'espace de Krylov interne. Historique de convergence de la norme relative du résidu en nombre d'itérations (a) et en nombre de produits matrices-vecteurs (b) pour le solveur FGMRES(60, $m_i$ ) et FGMRES-DR(60, $m_i$ ,20) avec  $m_i = 6, 10, 15, 20$ .

pour le préconditionneur BILU(0) et 70% pour le LUSGS. Nous avons reporté dans la Table 3.3 la variation du nombre d'itérations du FGMRES-DR en fonction du nombre de sous-domaines.

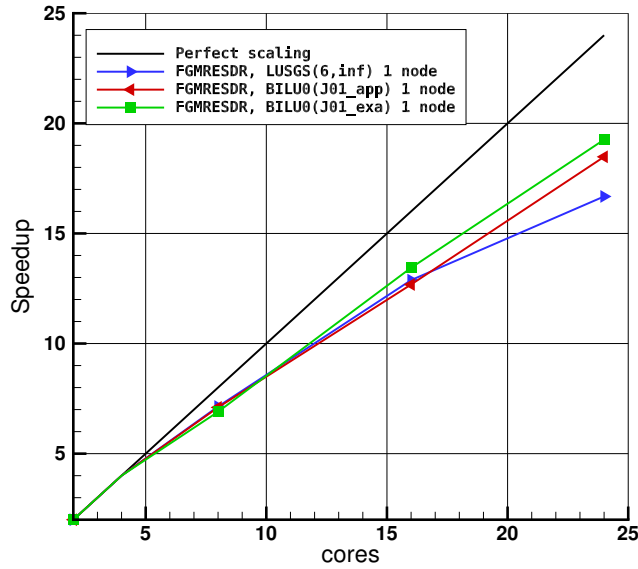


FIGURE 3.15 : Courbe de scalabilité du FGMRES-DR(60,20,20) sur le système adjoint aile M6.

### 3. DÉVELOPPEMENT DE SOLVEURS DE KRYLOV EMBOÎTÉS ASSOCIÉS À DES STRATÉGIES DE PRÉCONDITIONNEMENT ROBUSTES

# its ( $\mathcal{M}$ )	# sous-domaines				
	2	4	8	16	24
# its (LUSGS(6,150))	190	189	190	188	189
# its (LUSGS(6,inf))	348	348	349	349	348
# its (BILU(0), $\mathbf{J}_{O_1}^{APP}$ )	134	135	134	134	135
# its (BILU(0), $\mathbf{J}_{O_1}^{EXA}$ )	99	98	99	99	98

TABLE 3.3 : Impact du nombre de sous-domaines sur le nombre d'itérations pour le solveur FMGRES-DR(60,20,20) et selon le type du préconditionneur.

## Bilan

Nous avons présenté dans ce chapitre l'intérêt des solveurs GMRES emboîtés à accélérer drastiquement la résolution des systèmes linéaires mal conditionnés. Plus précisément, des stratégies de préconditionnement robustes ont été développées telles que la méthode de factorisation par bloc incomplète appliquée à la matrice jacobienne exacte d'ordre 1. Nous avons aussi mis en oeuvre la technique de déflation spectrale par augmentation qui a montré tout son intérêt dans les situations où le solveur GMRES emboîté ne convergait pas. Les résultats numériques ont démontré des gains significatifs allant jusqu'à 50 % en temps de calcul. Bien que ces stratégies aient déjà été étudiées auparavant, elles ont montré leur réel intérêt pour des cas industriels numériquement raides.

# Chapitre 4

## Stratégie de recyclage d'espace invariant approché pour le solveur partitionné

### Sommaire

---

<b>4.1</b>	<b>Historique des méthodes de recyclage</b>	<b>77</b>
4.1.1	Correction de rang 1	77
4.1.2	Méthode généralisée des résidus conjugués - GCR	79
4.1.3	Méthode GMRES Récursive - GMRESR	80
4.1.4	Stratégies de troncature pour les méthodes de Krylov - GCROT	81
4.1.5	Méthode de recyclage GCRO-DR	83
4.1.6	Cadre général des méthodes de recyclage	90
4.1.7	Loose GCRO-DR	93
4.1.8	Loose FGCRO-DR	96
4.1.9	Critère d'activation pour le recyclage d'espace invariant approché	96
<b>4.2</b>	<b>Résultats numériques</b>	<b>98</b>
4.2.1	Aile M6 Analyse aéroélastique	98
4.2.2	Évaluation des performances dans le cas d'un système adjoint aéro-dynamique	100
4.2.3	Effet de l'approximation des erreurs sur le système adjoint aéro-dynamique	103
4.2.4	Évaluation des performances du solveur partitionné dans le cas d'un système adjoint couplé	106

---

## Introduction

L'objectif de ce chapitre est d'accélérer la résolution des systèmes adjoints couplés dans le cadre d'une approche partitionnée. La démarche consiste dans un premier temps à intégrer naturellement dans le solveur partitionné les contributions des préconditionneurs développés dans le cas d'un solveur adjoint fluide. Dans un deuxième temps, nous mettons en place une stratégie de recyclage qui consiste à recycler l'information spectrale du système adjoint à chaque couplage fluide-structure. Nous verrons que le solveur GMRES-DR basé sur la technique de déflation par augmentation ne permet pas de recycler une telle information. Pour cela, nous proposons d'implémenter un algorithme de Krylov adapté à ce type de recyclage, le GCRO-DR. La section 4.1 décrit l'historique du développement des méthodes de recyclage. Dans la section 4.2, nous démontrons numériquement l'intérêt du recyclage à travers une étude comparative entre le GMRES-DR et le GCRO-DR. Cette démonstration s'effectuera également dans le cas des solveurs de Krylov flexibles.

## 4.1 Historique des méthodes de recyclage

### 4.1.1 Correction de rang 1

Dans le but d'accélérer la résolution des systèmes linéaires, Eirola et Nevanlinna ont développé une méthode itérative de type point fixe [54]. Pour cela, ils ont proposé de construire à chaque itération un préconditionneur en lui ajoutant une correction matricielle de rang 1. Cette correction engendre une suite de matrices qui converge vers l'opérateur  $\mathbf{A}^{-1}$ .

La méthode suit le schéma classique de décomposition suivant :

$$\mathbf{x}_{k+1} = \mathbf{M}^{-1}\mathbf{N}\mathbf{x}_k + \mathbf{M}^{-1}\mathbf{b} \quad (4.1)$$

avec  $\mathbf{M}, \mathbf{N}$  qui vérifient  $\mathbf{A} = \mathbf{M} - \mathbf{N}$  et où  $\mathbf{M}^{-1}$  est une approximation de  $\mathbf{A}^{-1}$ . Dans la suite, nous noterons  $\mathbf{M} = \mathbf{H}_k^{-1}$  le préconditionneur mis à jour à l'itération  $k$  et  $\mathbf{N} = \mathbf{R}_k$ . On peut réécrire l'équation (4.1) sous la forme :

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{H}_{k+1}\mathbf{r}_k, \quad \mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k \quad (4.2)$$

En termes de résidu, nous obtenons

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \mathbf{A}\mathbf{H}_{k+1}\mathbf{r}_k \quad (4.3)$$

La correction de rang 1 consiste à mettre à jour  $\mathbf{H}_k$  par une matrice de rang 1 de la forme :

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \mathbf{u}_k\mathbf{v}_k^T \quad (4.4)$$

où  $\mathbf{u}_k, \mathbf{v}_k \in \mathbb{R}^n$  sont des vecteurs à déterminer.

En injectant (4.4) dans (4.3), on obtient :

$$\mathbf{r}_{k+1} = (\mathbf{I} - \mathbf{A}\mathbf{H}_k)\mathbf{r}_k - (\mathbf{v}_k^T\mathbf{r}_k)\mathbf{A}\mathbf{u}_k \quad (4.5)$$

$$= (\mathbf{I} - \mathbf{A}\mathbf{H}_k)\mathbf{r}_k - \mu_k\mathbf{A}\mathbf{u}_k \quad (4.6)$$

où  $\mu_k = \mathbf{v}_k^T\mathbf{r}_k$ . La détermination des vecteurs  $\mathbf{u}_k$  et  $\mathbf{v}_k$  s'effectue de sorte que  $\mathbf{r}_{k+1}$  soit nul, ce qui donne

$$\mu_k\mathbf{A}\mathbf{u}_k = (\mathbf{I} - \mathbf{A}\mathbf{H}_k)\mathbf{r}_k \quad (4.7)$$

Le paramètre  $\mu_k$  est déterminé en minimisant  $\|\mathbf{r}_{k+1}\|_2$ , soit pour  $\mathbf{r}_{k+1} = 0$ . Donc, d'après (4.7) on obtient

$$\mu_k = \frac{1}{\|\mathbf{A}\mathbf{u}_k\|_2^2} (\mathbf{A}\mathbf{u}_k)^T (\mathbf{I} - \mathbf{A}\mathbf{H}_k) \mathbf{r}_k \quad (4.8)$$

Le vecteur  $\mathbf{v}_k$  est ensuite déterminé grâce à la relation  $\mu_k = \mathbf{v}_k^T \mathbf{r}_k$  ce qui nous donne :

$$\mathbf{v}_k = \frac{1}{\|\mathbf{A}\mathbf{u}_k\|_2^2} (\mathbf{I} - \mathbf{A}\mathbf{H}_k)^T \mathbf{A}\mathbf{u}_k \quad (4.9)$$

Le lecteur pourra consulter l'Algorithme 4 (voir Annexe B p. 129).

L'opérateur  $\mathbf{E}_k$  défini dans l'Algorithme 4 vérifie la propriété suivante : si on pose  $\mathbf{c}_k = \frac{\mathbf{A}\mathbf{u}_k}{\|\mathbf{A}\mathbf{u}_k\|}$  le vecteur unitaire conjugué à  $\mathbf{u}_k$ , il s'ensuit d'après (4.9) que :

$$\begin{aligned} \mathbf{E}_{k+1} &:= (\mathbf{I} - \mathbf{A}\mathbf{H}_{k+1}) \\ &= (\mathbf{I} - \mathbf{A}\mathbf{H}_k) - \mathbf{A}\mathbf{u}_k \mathbf{v}_k^T \\ &= (\mathbf{I} - \mathbf{A}\mathbf{H}_k) - \frac{\mathbf{A}\mathbf{u}_k}{\|\mathbf{A}\mathbf{u}_k\|_2^2} (\mathbf{A}\mathbf{u}_k)^T (\mathbf{I} - \mathbf{A}\mathbf{H}_k) \\ &= (\mathbf{I} - \mathbf{c}_k \mathbf{c}_k^T) \mathbf{E}_k \end{aligned}$$

où  $\mathbf{c}_k \mathbf{c}_k^T$  est la projection de rang 1. Par récurrence sur  $k$ , on en déduit que l'opérateur  $\mathbf{E}_{k+1}$  n'est autre que la projection orthogonale sur  $\text{Vect}\{\mathbf{A}\mathbf{u}_0, \mathbf{A}\mathbf{u}_1, \dots, \mathbf{A}\mathbf{u}_k\}$ . Autrement dit,  $\mathbf{E}_{k+1}$  vérifie :

$$\mathbf{E}_{k+1} = (\mathbf{I} - \sum_{i=0}^k \mathbf{c}_i \mathbf{c}_i^T) \mathbf{E}_0 \quad (4.10)$$

Dans la pratique, l'Algorithme 4 est trop coûteux notamment à cause des produits par  $\mathbf{A}$ . Il est possible de réduire ce coût en utilisant la relation (4.10) qui est une combinaison linéaire des vecteurs  $\mathbf{c}_0, \dots, \mathbf{c}_k$  et qui aboutit à l'Algorithme 5 (Annexe B p. 129).

Dans les lignes qui suivent, nous allons expliquer le passage de l'Algorithme 4 à l'Algorithme 5. En posant  $\xi^{(0)} := \mathbf{E}_0 \mathbf{r}_k$ , on a

$$\begin{aligned} \xi^{(k)} &:= \mathbf{E}_k \mathbf{r}_k = (\mathbf{I} - \sum_{i=0}^{k-1} \mathbf{c}_i \mathbf{c}_i^T) \xi^{(0)} \quad \text{d'après (4.10)} \\ &= \xi^{(0)} - \sum_{i=0}^{k-1} (\mathbf{c}_i^T \xi^{(0)}) \mathbf{c}_i \\ &= \xi^{(0)} - \sum_{i=0}^{k-1} \alpha_i \mathbf{c}_i, \quad \alpha_i = \langle \mathbf{c}_i, \xi^{(0)} \rangle \end{aligned}$$

Sachant que  $\mathbf{u}_k = \mathbf{H}_k \mathbf{E}_k \mathbf{r}_k$  (d'après (4.7)), on a

$$\begin{aligned}
\mathbf{u}_k &= \mathbf{H}_k \xi^{(k)} \\
&= (\mathbf{H}_0 + \sum_{i=0}^{k-1} \mathbf{u}_i \mathbf{v}_i^T) \xi^{(k)}, \quad \text{d'après (4.4)} \\
&= \mathbf{u}_k^{(0)} + \sum_{i=0}^{k-1} (\mathbf{v}_i^T \xi^{(k)}) \mathbf{u}_i \\
&= \mathbf{u}_k^{(0)} + \sum_{i=0}^{k-1} (\mathbf{c}_i^T \mathbf{E}_i \xi^{(k)}) \mathbf{u}_i, \quad \text{d'après (4.9)} \\
&= \mathbf{u}_k^{(0)} + \sum_{i=0}^{k-1} (\mathbf{c}_i^T (\mathbf{I} - \sum_{j=0}^{i-1} \mathbf{c}_j \mathbf{c}_j^T) \mathbf{E}_0 \xi^{(k)}) \mathbf{u}_i, \quad \text{d'après (4.10)} \\
&= \mathbf{u}_k^{(0)} + \sum_{i=0}^{k-1} (\mathbf{c}_i^T \mathbf{E}_0 \xi^{(k)}) \mathbf{u}_i, \quad \mathbf{c}_i \perp \mathbf{c}_0, \dots, \mathbf{c}_{i-1} \\
&= \mathbf{u}_k^{(0)} + \sum_{i=0}^{k-1} (\mathbf{c}_i^T \mathbf{E}_0 \xi^{(k)}) \mathbf{u}_i \\
&= \mathbf{u}_k^{(0)} + \sum_{i=0}^{k-1} \beta_i \mathbf{u}_i, \quad \beta_i = \langle \mathbf{c}_i, \mathbf{E}_0 \xi^{(k)} \rangle
\end{aligned}$$

L'intérêt du développement ci-dessus est la capacité à calculer efficacement le vecteur  $\mathbf{u}_k$  au moyen d'une récurrence. Les expressions de  $\mathbf{x}_{k+1}$  et  $\mathbf{r}_{k+1}$  se déduisent de la même façon en utilisant les relations (4.4), (4.9) et (4.10). Un point important à souligner est qu'il n'y a pas de contrainte sur le choix du vecteur  $\mathbf{u}_k$ . À titre d'exemple, Van der Vorst et Vuik ont démontré dans [51] qu'en choisissant  $\mathbf{u}_k = \mathbf{H}_k \mathbf{r}_k$  au lieu de  $\mathbf{H}_k \mathbf{E}_k \mathbf{r}_k$ , l'Algorithme 5 est algébriquement équivalent au GMRES. En choisissant  $\mathbf{u}_k = \mathbf{r}_k$ , nous obtenons la méthode généralisée des résidus conjugués (GCR). L'Algorithme 6 (Annexe B p. 130) est une généralisation de l'Algorithme 5. Cet algorithme, baptisé GMRES $\star$ , a été présenté dans [51].

La question qui se pose est celle du choix optimal de  $\mathbf{u}_k$ . Ce choix est tel que le résidu  $\mathbf{r}_{k+1}$  soit nul, ce qui implique de choisir idéalement  $\mathbf{u}_k$  comme direction de recherche égale à  $\mathbf{u}_k = \mathbf{A}^{-1} \mathbf{r}_k = \mathbf{e}_k$  où  $\mathbf{e}_k = \mathbf{x} - \mathbf{x}_k$  est l'erreur sur la solution. La détermination de  $\mathbf{u}_k$  s'effectue en résolvant l'équation  $\mathbf{A} \mathbf{u}_k = \mathbf{r}_k$  au moyen d'une méthode itérative, en l'occurrence la méthode GMRES( $m$ ). À chaque itération externe  $k$ , nous générons une direction de recherche égale à  $\mathcal{P}_m(\mathbf{A}) \mathbf{r}_k$  où  $\mathcal{P}_m$  représente un polynôme de degré  $\leq m$ . Nous obtenons ainsi la méthode GMRES Récursive (GMRESR). Nous exposons à présent les différentes méthodes qui découlent du GMRES $\star$ .

#### 4.1.2 Méthode généralisée des résidus conjugués - GCR

La méthode généralisée des résidus conjugués (GCR) fait partie des méthodes de Krylov (Algorithme 7, Annexe B p.130) dont le principe consiste à construire deux espaces



conjugués  $\mathbf{U}_k$  et  $\mathbf{C}_k$  qui vérifient :

$$\text{range}(\mathbf{U}_k) = \mathcal{K}_m(\mathbf{A}, \mathbf{r}_0) \quad (4.11)$$

$$\mathbf{A}\mathbf{U}_k = \mathbf{C}_k \quad (4.12)$$

$$\mathbf{C}_k^T \mathbf{C}_k = \mathbf{I}_k. \quad (4.13)$$

Comme dans le cas du GMRES, la méthode calcule une approximation de la solution par minimisation du résidu. Autrement dit, on a

$$\begin{aligned} \mathbf{x}_k &= \underset{\mathbf{x} \in \mathbf{x}_0 + \text{range}(\mathbf{U}_k)}{\text{argmin}} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2 \Leftrightarrow \mathbf{y}_k = \underset{\mathbf{y} \in \mathbb{R}^k}{\text{argmin}} \|\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{U}_k\mathbf{y})\|_2 \\ &\Leftrightarrow \mathbf{y}_k = \underset{\mathbf{y} \in \mathbb{R}^k}{\text{argmin}} \|\mathbf{r}_0 - \mathbf{C}_k\mathbf{y}\|_2. \end{aligned} \quad (4.14)$$

Par orthonormalité de  $\mathbf{C}_k$ , la solution du problème (4.14) est donnée par  $\mathbf{y}_k = \mathbf{C}_k^T \mathbf{r}_0$  et par conséquent,

$$\mathbf{x}_k = \mathbf{x}_0 + \mathbf{U}_k \mathbf{C}_k^T \mathbf{r}_0 \quad (4.15)$$

$$\mathbf{r}_k = \mathbf{r}_0 - \mathbf{C}_k \mathbf{C}_k^T \mathbf{r}_0, \quad \mathbf{r}_k \perp \text{range}(\mathbf{C}_k) \quad (4.16)$$

### 4.1.3 Méthode GMRES Récursive - GMRESR

Comme souligné précédemment, l'algorithme GMRES<sub>\*</sub> est général au sens où il n'y a aucune contrainte dans le choix  $\mathbf{u}_k$ . Sachant que  $\mathbf{u}_k = \mathbf{A}^{-1} \mathbf{r}_k = \mathbf{e}_k$  est le choix optimal, l'idée est de calculer une bonne approximation de  $\mathbf{e}_k$  en résolvant l'équation résiduelle (4.17) par un solveur GMRES

$$\mathbf{A}\mathbf{e}_k = \mathbf{r}_k \quad (4.17)$$

Nous obtenons ainsi une méthode emboîtée nommée GMRESR (Algorithme 8, Annexe B p.131). La méthode externe GCR calcule la solution optimale  $\mathbf{x}_k$  dans l'espace  $\text{range}(\mathbf{U}_k)$  au sens de la minimisation du résidu  $\mathbf{r}_k$ . La méthode interne GMRES calcule une approximation de la solution de l'équation résiduelle (4.17) fournissant une nouvelle direction de recherche pour l'espace externe. L'inconvénient de cette méthode est que l'approximation de la solution est sous-optimale au sens où l'espace dans lequel le problème (4.14) est résolu n'est pas global. En effet, le solveur GMRES interne résout

$$\min_{\mathbf{y}_m \in \mathbb{R}^m} \|\mathbf{r}_{k-1} - \mathbf{A}\mathbf{V}_m\mathbf{y}_m\|_2 \quad (4.18)$$

où  $\mathbf{V}_m$  correspond à une base orthonormale de l'espace de Krylov  $\mathcal{K}_m(\mathbf{A}, \mathbf{r}_{k-1})$ . Par consé-

quent, les vecteurs externes  $\mathbf{u}_k$  et  $\mathbf{c}_k$  s'expriment :

$$\begin{aligned}\mathbf{u}_k &= (\mathbf{V}_m \mathbf{y}_m - \mathbf{U}_{k-1} \mathbf{C}_{k-1}^T \mathbf{A} \mathbf{V}_m \mathbf{y}_m) / \|(\mathbf{I} - \mathbf{C}_{k-1}^T \mathbf{C}_{k-1}) \mathbf{A} \mathbf{V}_m \mathbf{y}_m\|_2 \\ \mathbf{c}_k &= (\mathbf{I} - \mathbf{C}_{k-1} \mathbf{C}_{k-1}^T) \mathbf{A} \mathbf{V}_m \mathbf{y}_m / \|(\mathbf{I} - \mathbf{C}_{k-1} \mathbf{C}_{k-1}^T) \mathbf{A} \mathbf{V}_m \mathbf{y}_m\|_2 \\ \mathbf{x}_k &= \mathbf{x}_{k-1} + \mathbf{u}_k \mathbf{C}_k^T \mathbf{r}_{k-1} \\ \mathbf{r}_k &= \mathbf{r}_{k-1} - \mathbf{c}_k \mathbf{C}_k^T \mathbf{r}_{k-1}\end{aligned}$$

À chaque boucle externe, le vecteur  $\mathbf{c}_k$  est orthogonalisé par rapport à l'espace des résidus  $\text{range}(\mathbf{C}_{k-1})$ . Cette condition d'orthogonalité n'est pas prise en compte dans la boucle interne et cela entraîne un calcul sous-optimal de la solution. Le théorème suivant démontre qu'en maintenant cette relation d'orthogonalité entre l'espace externe et l'espace interne (ce qui revient à appliquer un GMRES( $m$ ) sur l'opérateur  $\mathbf{A}_{\mathbf{C}_k} = (\mathbf{I} - \mathbf{C}_k \mathbf{C}_k^T \mathbf{A})$ ), l'approximation de la solution  $\mathbf{x}_k$  vérifie un problème de minimisation global.

**Théorème 1** *Soient  $\mathbf{A}$ ,  $\mathbf{U}_k$ ,  $\mathbf{C}_k$ ,  $\mathbf{r}_k$ ,  $\mathbf{x}_k$ ,  $\mathbf{A}_{\mathbf{C}_k}$  définis à l'itération  $k$ . Supposons que les vecteurs  $\mathbf{r}_k, \mathbf{A}_{\mathbf{C}_k} \mathbf{r}_k, \mathbf{A}_{\mathbf{C}_k}^2 \mathbf{r}_k, \dots, \mathbf{A}_{\mathbf{C}_k}^m \mathbf{r}_k$  soient indépendants et que  $\{\mathbf{v}_1, \dots, \mathbf{v}_{m+1}\}$  soit une base orthonormale de  $\mathcal{K}_{m+1}(\mathbf{A}_{\mathbf{C}_k}, \mathbf{r}_k)$  générée par la méthode GMRES( $m$ ) et où  $\mathbf{v}_1 = \mathbf{r}_k / \|\mathbf{r}_k\|$ . Soit également  $\mathbf{y} \in \mathbb{R}^m$  la solution du problème aux moindres carrés*

$$\mathbf{y} = \underset{\tilde{\mathbf{y}} \in \mathbb{R}^m}{\text{argmin}} \|\mathbf{r}_k - \mathbf{A}_{\mathbf{C}_k} \mathbf{V}_m \tilde{\mathbf{y}}\|_2 = \underset{\tilde{\mathbf{y}} \in \mathbb{R}^m}{\text{argmin}} \|\mathbf{r}_k - \mathbf{V}_{m+1} \bar{\mathbf{H}}_m \tilde{\mathbf{y}}\|_2 \quad (4.19)$$

Alors, l'approximation de la solution  $\mathbf{x}_{k+1}$  s'écrit

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{A}^{-1} \mathbf{A}_{\mathbf{C}_k} \mathbf{V}_m \mathbf{y}, \quad (4.20)$$

et vérifie le problème de minimisation globale suivant

$$\min\{\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2 : \mathbf{x} \in \text{range}(\mathbf{U}_k) \oplus \text{range}(\mathbf{V}_m)\} \quad (4.21)$$

$$\min_{\mathbf{y} \in \mathbb{R}^m} \|\mathbf{r}_{k-1} - (\mathbf{I} - \mathbf{C}_{k-1} \mathbf{C}_{k-1}^T) \mathbf{A} \mathbf{V}_m \mathbf{y}\|_2 = \min_{\mathbf{y} \in \mathbb{R}^m} \|(\mathbf{I} - \mathbf{C}_{k-1} \mathbf{C}_{k-1}^T) (\mathbf{r}_{k-1} - \mathbf{A} \mathbf{V}_m \mathbf{y})\|_2 \quad (4.22)$$

Cette relation d'orthogonalité nous donne ainsi l'algorithme GCRO (Generalized Conjugate Residuals with inner Orthogonalization) (Algorithme 9, Annexe B p. 131).

#### 4.1.4 Stratégies de troncature pour les méthodes de Krylov - GCROT

Nous avons vu que l'algorithme GCRO permet de calculer la solution optimale dans l'espace  $\text{range}(\mathbf{U}_k) \oplus \text{range}(\mathbf{V}_m)$ . Malgré cette propriété satisfaisante, le coût mémoire augmente linéairement avec les itérations externes. En effet, si on fixe  $k_{max}$  le nombre

#### 4. STRATÉGIE DE RECYCLAGE D'ESPACE INVARIANT APPROCHÉ POUR LE SOLVEUR PARTITIONNÉ

maximal de vecteurs externes retenus dans  $\mathbf{C}_k$  et  $\mathbf{U}_k$  et  $m$  la taille maximale de l'espace de Krylov interne, le coût mémoire est évalué à  $m + 2k_{max}$ . Afin d'éviter les problèmes liés aux contraintes mémoire, il est nécessaire de procéder à une troncature de l'espace  $\text{range}(\mathbf{C}_k)$ .

La conséquence d'une telle troncature est que la solution calculée devient sous-optimale car la condition d'orthogonalité  $\mathbf{r}_i \perp \mathbf{A}\mathcal{K}_i(\mathbf{A}, \mathbf{b})$  n'est satisfaite que sur un espace de taille inférieure à  $m$ . Cela introduit une erreur sur le résidu optimal  $\mathbf{r}_i$  calculé sur l'espace de Krylov de taille  $m$ . De Sturler appelle cette erreur "*l'erreur résiduelle*" définie comme étant la différence entre le résidu optimal et le résidu calculé sur l'espace de Krylov après troncature. Pour ne pas dégrader la robustesse de l'algorithme GCRO, De Sturler propose une stratégie de troncature qui consiste à minimiser l'erreur résiduelle [5]. Cette méthode s'appelle le GCROT.

La question qui se pose est : quel sous-espace de  $\text{range}(\mathbf{C}_k)$  devrait-on recycler afin d'améliorer la convergence du GMRES interne lors des cycles suivants. Dans le cadre du GCRO, le GMRES interne génère la récurrence d'Arnoldi :

$$\mathbf{A}\mathbf{V}_m = \mathbf{C}_k\mathbf{B} + \mathbf{V}_{m+1}\bar{\mathbf{H}}_m. \quad (4.23)$$

où  $\mathbf{B} = \mathbf{C}_k^T \mathbf{A}\mathbf{V}_m \in \mathbb{R}^{k \times m}$ . En introduisant la factorisation QR de  $\bar{\mathbf{H}}_m$  et en négligeant la dernière ligne de  $\mathbf{R}$ , nous obtenons

$$\mathbf{A}\mathbf{V}_m = \mathbf{C}_k\mathbf{B} + (\mathbf{V}_{m+1}\mathbf{Q}_m)\mathbf{R}_m, \quad (4.24)$$

Si on pose  $\mathbf{Q} = \mathbf{V}_{m+1}\mathbf{Q}_m$ , on obtient la relation

$$\mathbf{A}\mathbf{V}_m = \mathbf{C}_k\mathbf{B} + \mathbf{Q}\mathbf{R}_m. \quad (4.25)$$

Une propriété intéressante de l'espace  $\mathbf{V}_{m+1}$  est qu'il vérifie par construction la relation d'orthogonalité  $\mathbf{C}_k^T \mathbf{V}_{m+1} = 0$ , ce qui implique  $\mathbf{C}_k^T \mathbf{Q} = 0$ . Par conséquent, l'équation (4.25) fournit une base orthonormale de l'espace  $\text{range}(\mathbf{A}\mathbf{V}_m)$ . À présent, si on néglige l'orthogonalité sur l'espace  $\text{range}(\mathbf{C}_k)$ , alors la factorisation QR de  $\mathbf{A}\mathbf{V}_m$  se réduit à :

$$\mathbf{A}\mathbf{V}_m = \mathbf{W}_m\mathbf{S}, \quad (4.26)$$

où  $\mathbf{S} \in \mathbb{R}^{m \times m}$  est triangulaire supérieure et  $\mathbf{W}_m^T \mathbf{W}_m = \mathbf{I}_m$ . Il s'ensuit d'après (4.25) et (4.26) que la meilleure approximation du résidu  $\mathbf{r}_k$  dans  $(\text{range}(\mathbf{C}_k) \oplus \text{range}(\mathbf{A}\mathbf{V}_m))^\perp$  est définie par le résidu  $\mathbf{r}_k^{(1)}$  et s'exprime :

$$\begin{aligned} \mathbf{r}_k^{(1)} &= (\mathbf{I} - \mathbf{C}_k\mathbf{C}_k^T - \mathbf{Q}\mathbf{Q}^T)\mathbf{r}_k \\ &= (\mathbf{I} - \mathbf{Q}\mathbf{Q}^T)\mathbf{r}_k \quad \text{car} \quad \mathbf{C}_k^T \mathbf{r}_k = 0 \end{aligned}$$

De même, on définit le résidu  $\mathbf{r}_k^{(2)}$  par la meilleure approximation de  $\mathbf{r}_k$  dans l'espace  $\text{range}(\mathbf{A}\mathbf{V}_m)^\perp$  en négligeant l'orthogonalité sur l'espace  $\text{range}(\mathbf{C}_k)$  et qui s'exprime :

$$\mathbf{r}_k^{(2)} = (\mathbf{I} - \mathbf{W}_m \mathbf{W}_m^T) \mathbf{r}_k$$

L'erreur résiduelle est donc définie par :

$$\epsilon = \mathbf{r}_k^{(2)} - \mathbf{r}_k^{(1)}. \quad (4.27)$$

De Sturler montre que l'erreur  $\epsilon$  s'exprime comme combinaison linéaire des vecteurs singuliers de la matrice  $\mathbf{B}\mathbf{R}_m^{-1}$  et dont les coefficients s'expriment en fonction des valeurs singulières associées (voir [5, Théorème 2.2]). Considérons ci-dessous la décomposition en valeur singulière de  $\mathbf{D} = \mathbf{B}\mathbf{R}_m^{-1}$  :

$$\mathbf{D} = \Lambda_{\mathbf{D}} \Sigma_{\mathbf{D}} \Gamma_{\mathbf{D}}^T \quad (4.28)$$

où les matrices unitaires  $\Lambda_{\mathbf{D}} = [\lambda_1, \dots, \lambda_k] \in \mathbb{R}^{k \times k}$  et  $\Gamma_{\mathbf{D}} = [\gamma_1, \dots, \gamma_k] \in \mathbb{R}^{m \times m}$  sont ordonnées telles que les valeurs singulières  $\Sigma_{\mathbf{D}}$  satisfont :

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p, \quad p = \min(k, m) \quad (4.29)$$

Par conséquent, l'erreur satisfait

$$\epsilon = \sum_{i=1}^p \epsilon_i, \quad \text{où} \quad \epsilon_i = \left( \frac{\nu_i \sigma_i^2}{1 + \sigma_i^2} \mathbf{Q} \gamma_i - \frac{\nu_i \sigma_i}{1 + \sigma_i^2} \mathbf{C}_k \lambda_i \right), \quad (4.30)$$

et  $\nu_i = \gamma_i^T \mathbf{Q}^T \mathbf{r}_k$ . La norme de l'erreur est donnée par

$$\|\epsilon\|_2 = \left( \sum_i^p \|\epsilon_i\|_2 \right)^{\frac{1}{2}} \quad \text{où} \quad \|\epsilon_i\|_2 = \frac{\nu_i^2 \sigma_i^2}{1 + \sigma_i^2}. \quad (4.31)$$

En analysant le comportement de la norme de l'erreur, on remarque que pour les plus petites valeurs de  $\sigma_i$ , l'erreur associée  $\epsilon_i$  est faible tandis que pour les plus grandes valeurs de  $\sigma_i$ ,  $\epsilon_i$  converge vers la valeur prépondérante  $\nu_i^2$ . À travers ses expériences numériques, De Sturler souligne le comportement insatisfaisant du GCROT en fonction des valeurs singulières et propose de tronquer les colonnes de  $\mathbf{C}_k$  associées aux plus grandes valeurs singulières.

#### 4.1.5 Méthode de recyclage GCRO-DR

Une propriété remarquable de la famille GCRO est leur capacité à résoudre des séquences de systèmes linéaires en recyclant un sous-espace invariant approché de l'opérateur  $\mathbf{A}$ . Par exemple, De Sturler a mis en oeuvre un solveur GCROT adapté à la résolution d'une

#### 4. STRATÉGIE DE RECYCLAGE D'ESPACE INVARIANT APPROCHÉ POUR LE SOLVEUR PARTITIONNÉ

---

séquence de systèmes linéaires dans laquelle il fait varier le second membre [55]. Le solveur GMRES-DR utilisant la déflation par augmentation [36] ne permet malheureusement pas de recycler d'un système à un autre même si seul le second membre varie. En effet, le changement de l'opérateur ou même du second membre ne permet pas de redémarrer une récurrence d'Arnoldi générant une base de Krylov. Pour comprendre pourquoi le GMRES-DR ne permet pas le recyclage spectral, nous revisitons les relations fondamentales du GMRES-DR.

Nous rappelons que le principe de la déflation spectrale est de recycler, à chaque cycle, un sous espace invariant approché de  $\mathbf{A}$  engendré par les vecteurs propres associés aux plus petites valeurs propres dans le but d'accélérer la convergence. À la fin du cycle, le GMRES-DR retient  $k$  vecteurs de Ritz harmoniques représentés par la matrice  $\tilde{\mathbf{Y}}_k \in \mathbb{R}^{n \times k}$ . Après avoir orthonormalisé  $\tilde{\mathbf{Y}}_k$  pour donner  $\tilde{\mathbf{Y}}_k$ , le GMRES-DR complète le reste de la base en appliquant  $m - k$  itérations d'Arnoldi à partir de  $\mathbf{v}_1 = \mathbf{r}/\|\mathbf{r}\|$ , tout en maintenant l'orthogonalité avec  $\tilde{\mathbf{Y}}_k$ . Une des propriétés remarquables de la matrice  $\tilde{\mathbf{Y}}_k$  est qu'elle vérifie les propriétés d'un espace de Krylov. Ceci permet à la technique de déflation développée par Morgan d'être cohérente avec la relation fondamentale d'Arnoldi. Autrement dit, nous avons

$$\mathbf{A}[\tilde{\mathbf{Y}}_k \quad \mathbf{V}_{m-k}] = [\tilde{\mathbf{Y}}_k \quad \mathbf{V}_{m-k+1}]\bar{\mathbf{H}}_m \quad (4.32)$$

où  $\bar{\mathbf{H}}_m$  est une matrice de Hessenberg supérieure sauf pour le premier bloc dense de dimension  $(k + 1) \times (k + 1)$ .

En changeant l'opérateur  $\mathbf{A}$  ou même le second membre, la matrice de déflation  $\tilde{\mathbf{Y}}_k$  ne constitue plus une base de Krylov pour le nouvel opérateur et donc la relation (4.32) n'est plus vérifiée pour le système suivant.

Parks et al. [13] ont mis en place une version généralisée du GMRES-DR permettant le recyclage d'espace invariant approché d'un système à un autre. Elle combine les propriétés du GMRES-DR et du GCRO, d'où la dénomination GCRO-DR. Ce solveur est plus flexible que le GMRES-DR dans le sens où un *espace quelconque* peut être recyclé pour la résolution du système suivant. Cette flexibilité est possible grâce à la propriété du solveur GCRO qui consiste à maintenir un espace de correction  $\mathbf{U}$  et son espace conjugué  $\mathbf{C}$  tels que  $\mathbf{AU} = \mathbf{C}$  et que  $\mathbf{C}$  soit orthonormal. Nous soulignons aussi que les deux solveurs GCRO-DR et GMRES-DR sont algébriquement équivalents dans le cadre d'une résolution d'un seul système linéaire. Nous illustrerons cette propriété dans les résultats numériques.

#### 4.1.5.1 Calcul des espaces de correction et de recyclage dans le cadre du GCRO

Cette section se concentre sur le calcul des espaces  $\mathbf{U}_k$  et  $\mathbf{C}_k$  dans le cadre du GCRO. Soit  $\mathbf{x}_0$  une solution initiale et  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$  le résidu associé. L'application de  $m$  itérations de GCRO nous donne

$$\mathbf{x}_m = \mathbf{x}_0 + \mathbf{U}_m \mathbf{C}_m^T \mathbf{r}_0, \quad (4.33)$$

$$\mathbf{r}_m = (\mathbf{I} - \mathbf{C}_m \mathbf{C}_m^T) \mathbf{r}_0 \quad (4.34)$$

avec  $\mathbf{U}_m$  et  $\mathbf{C}_m$  vérifiant les relations (4.12) et (4.13).

À l'itération  $m + 1$ , l'algorithme GCRO calcule  $\mathbf{c}_{m+1}$  et  $\mathbf{u}_{m+1}$  de la façon suivante :

$$\mathbf{c}_{m+1} = (\mathbf{I} - \mathbf{C}_m \mathbf{C}_m^T) \mathbf{A} \mathbf{r}_m / \|(\mathbf{I} - \mathbf{C}_m \mathbf{C}_m^T) \mathbf{A} \mathbf{r}_m\|_2, \quad (4.35)$$

$$\mathbf{u}_{m+1} = (\mathbf{r}_m - \mathbf{U}_m \mathbf{C}_m^T \mathbf{A} \mathbf{r}_m) / \|(\mathbf{I} - \mathbf{C}_m \mathbf{C}_m^T) \mathbf{A} \mathbf{r}_m\|_2, \quad (4.36)$$

$$\mathbf{r}_{m+1} = \mathbf{r}_m - \mathbf{c}_{m+1} \mathbf{c}_{m+1}^T \mathbf{r}_m, \quad (4.37)$$

$$\mathbf{x}_{m+1} = \mathbf{x}_m + \mathbf{u}_{m+1} \mathbf{c}_{m+1}^T \mathbf{r}_m. \quad (4.38)$$

Par ailleurs, l'application de  $m$  itérations de GMRES va générer la relation d'Arnoldi

$$\mathbf{A} \mathbf{W}_m = \mathbf{W}_{m+1} \bar{\mathbf{H}}_m, \quad \mathbf{w}_1 = \mathbf{r}_0 / \|\mathbf{r}_0\|_2 \quad (4.39)$$

$$\mathbf{W}_{m+1}^T \mathbf{W}_{m+1} = \mathbf{I}_{m+1} \quad (4.40)$$

$$\bar{\mathbf{H}}_m = \mathbf{Q}_m \bar{\mathbf{R}}_m \quad (4.41)$$

où  $\mathbf{Q}_m \in \mathbb{R}^{(m+1) \times (m+1)}$  est le résultat du produit des  $m$  rotations de Givens et  $\bar{\mathbf{R}}_m \in \mathbb{R}^{(m+1) \times m}$  la matrice triangulaire supérieure dont la dernière ligne est constituée de zéros. En éliminant la dernière colonne de  $\mathbf{Q}_m$  et la dernière ligne de  $\bar{\mathbf{R}}_m$ , on obtient la relation

$$\bar{\mathbf{H}}_m = \bar{\mathbf{Q}}_m \mathbf{R}_m \quad (4.42)$$

La solution produite par le GMRES s'écrit :

$$\begin{aligned} \mathbf{x}_m &= \mathbf{x}_0 + \mathbf{W}_m \mathbf{y}, \quad \mathbf{y} \in \mathbb{R}^m \\ &= \mathbf{x}_0 + \mathbf{W}_m \mathbf{R}_m^{-1} \bar{\mathbf{Q}}_m^T \|\mathbf{r}_0\|_2 \mathbf{e}_1 \end{aligned} \quad (4.43)$$

Grâce au caractère orthonormal de  $\bar{\mathbf{Q}}_m$  et à la relation (4.42), le problème aux moindres carrés s'exprime de la façon suivante :

$$\mathbf{y} = \operatorname{argmin}_{\mathbf{y} \in \mathbb{R}^m} \|\bar{\mathbf{Q}}_m^T \|\mathbf{r}_0\|_2 \mathbf{e}_1 - \mathbf{R}_m \mathbf{y}\|_2 \quad (4.44)$$

#### 4. STRATÉGIE DE RECYCLAGE D'ESPACE INVARIANT APPROCHÉ POUR LE SOLVEUR PARTITIONNÉ

La solution est donc donnée par  $\mathbf{y} = \mathbf{R}_m^{-1} \bar{\mathbf{Q}}_m^T \|\mathbf{r}_0\| \mathbf{e}_1$ , d'où l'expression de  $\mathbf{x}_m$ . Le résidu a pour expression :

$$\mathbf{r}_m = \mathbf{b} - \mathbf{A}\mathbf{x}_m \quad (4.45)$$

$$\begin{aligned} &= \mathbf{r}_0 - \mathbf{W}_{m+1} \bar{\mathbf{H}}_m \mathbf{R}_m^{-1} \bar{\mathbf{Q}}_m^T \|\mathbf{r}_0\| \mathbf{e}_1 \\ &= \mathbf{r}_0 - \mathbf{W}_{m+1} \bar{\mathbf{Q}}_m \bar{\mathbf{Q}}_m^T \|\mathbf{r}_0\| \mathbf{e}_1 \\ &= \mathbf{r}_0 - \mathbf{W}_{m+1} \bar{\mathbf{Q}}_m \bar{\mathbf{Q}}_m^T \mathbf{W}_{m+1}^T \mathbf{r}_0 \quad \text{car} \quad \mathbf{W}_{m+1}^T \mathbf{r}_0 = \mathbf{W}_{m+1}^T \|\mathbf{r}_0\| \mathbf{w}_1 = \|\mathbf{r}_0\| \mathbf{e}_1 \\ &= (\mathbf{I} - (\mathbf{W}_{m+1} \bar{\mathbf{Q}}_m)(\mathbf{W}_{m+1} \bar{\mathbf{Q}}_m)^T) \mathbf{r}_0 \end{aligned} \quad (4.46)$$

En comparant la relation (4.34) et (4.46), on déduit que :

$$\mathbf{C}_m = \mathbf{W}_{m+1} \bar{\mathbf{Q}}_m, \quad (4.47)$$

$$\mathbf{U}_m = \mathbf{W}_{m+1} \mathbf{R}_m^{-1} \quad (4.48)$$

Nous venons de fournir une expression de  $\mathbf{C}_m$  et  $\mathbf{U}_m$  en fonction de la base de Krylov  $\mathbf{W}_m$  et de la factorisation QR de  $\bar{\mathbf{H}}_m$ . Il est important de considérer ces expressions car elles sont numériquement plus robustes que les expressions (4.35)-(4.36). Cette robustesse est due à l'emploi d'une orthogonalisation implicite de l'espace de Krylov généré par la récurrence d'Arnoldi du GMRES. Nous allons à présent fournir les expressions de ces deux espaces dans le cadre du GCRO-DR, c'est-à-dire dans le cas d'une déflation spectrale.

##### 4.1.5.2 Description de l'algorithme GCRO-DR flexible

Considérons la séquence de systèmes linéaires suivante :

$$\mathbf{A}\mathbf{x}^{(s)} = \mathbf{b}^{(s)} \quad s = 1, 2, \dots, \quad (4.49)$$

où  $\mathbf{b}^{(s)} \in \mathbb{R}^n$  varie d'un système à un autre et où l'indice  $s$  correspond au  $s^{\text{ème}}$  système linéaire.

Dans le cadre de cette thèse, l'opérateur  $\mathbf{A}$  est constant et seul le second membre varie. Pour fixer les idées, on notera  $\mathbf{r}_i^{(s)}$  le résidu à la fin du cycle  $i$  pour le système linéaire  $s$ . Nous soulignons que la relation (4.12) devient dans le cas flexible :

$$\mathbf{A}\mathbf{Z}_k = \mathbf{C}_k \quad (4.50)$$

##### Premier cycle

- Le premier cycle du FGCRO-DR correspond à un cycle complet d'un FGMRES-DR suivi du calcul de la matrice  $\mathbf{P}_k$  nécessaire à l'obtention des vecteurs de Ritz harmoniques. Les espaces  $\mathbf{U}_k$ ,  $\mathbf{Z}_k$  et  $\mathbf{C}_k$  sont construits à partir de l'information spectrale contenue dans  $\bar{\mathbf{H}}_m \mathbf{P}_k$ . Plus précisément, en calculant la décomposition QR de  $\bar{\mathbf{H}}_m \mathbf{P}_k$  et en appliquant la relation d'Arnoldi flexible (3.20) on a

$$\mathbf{AZ}_m \mathbf{P}_k = \mathbf{V}_{m+1} \bar{\mathbf{H}}_m \mathbf{P}_k = (\mathbf{V}_{m+1} \mathbf{Q}) \mathbf{R} \quad (4.51)$$

Comme  $\mathbf{V}_{m+1}$  et  $\mathbf{Q}$  sont deux matrices orthonormales et qu'il faut vérifier (4.50), on déduit de (4.51) les relations ci-dessous

$$\mathbf{C}_k = \mathbf{V}_{m+1} \mathbf{Q} \quad (4.52)$$

$$\mathbf{Z}_k = \mathbf{Z}_m \mathbf{P}_k \mathbf{R}^{-1} \quad (4.53)$$

Dans le cas non flexible, on a la relation  $\mathbf{U}_k = \mathbf{V}_m \mathbf{P}_k \mathbf{R}^{-1}$ .

Soulignons que le résidu calculé à la fin du premier cycle vérifie bien la condition d'optimalité sur l'espace  $\text{range}(\mathbf{C}_k)$ . En effet,

$$(\mathbf{AZ}_m)^T \mathbf{r}_1^{(s)} = 0 \quad (4.54)$$

$$\Leftrightarrow \mathbf{P}_k^T \mathbf{Z}_m^T \mathbf{A}^T \mathbf{r}_1^{(s)} = 0 \quad (4.55)$$

$$\Leftrightarrow (\mathbf{V}_{m+1} \bar{\mathbf{H}}_m \mathbf{P}_k)^T \mathbf{r}_1^{(s)} = 0 \quad (4.56)$$

$$\Leftrightarrow \mathbf{R}^T \mathbf{C}_k^T \mathbf{r}_1^{(s)} = 0 \quad (4.57)$$

$$\Leftrightarrow \mathbf{C}_k^T \mathbf{r}_1^{(s)} = 0 \quad (4.58)$$

### Autres cycles

- Les autres cycles, indexés par l'entier  $i$ , consistent à compléter la base de Krylov en appliquant  $(m - k)$  étapes d'Arnoldi sur le système projeté  $\mathbf{P} \mathbf{A} \mathbf{x} = \mathbf{P} \mathbf{b}$  où  $\mathbf{P} := \mathbf{I} - \mathbf{C}_k \mathbf{C}_k^T$  est le projecteur orthogonal sur  $\text{range}(\mathbf{C}_k)^\perp$ . Autrement dit, on résout le problème projeté suivant :

$$(\mathbf{I} - \mathbf{C}_k^T \mathbf{C}_k) \mathbf{A} \mathbf{z} = (\mathbf{I} - \mathbf{C}_k^T \mathbf{C}_k) \mathbf{r}_i \quad (4.59)$$

$$= \mathbf{r}_i \quad \text{car } \mathbf{r}_i \perp \text{range}(\mathbf{C}_k) \quad (4.60)$$

On rappelle que la résolution de l'équation ci-dessus au moyen d'une méthode de Krylov permet de minimiser le résidu sur la somme directe de l'espace externe et de l'espace interne. Ceci permet d'obtenir une relation d'Arnoldi généralisée de la forme :

$$\mathbf{AZ}_m = \mathbf{V}_{m+1} \bar{\mathbf{H}}_m \quad (4.61)$$

où

$$\mathbf{Z}_m = [\mathbf{Z}_k \quad \mathbf{Z}_{m-k}], \quad \mathbf{V}_{m+1} = [\mathbf{C}_k \quad \mathbf{V}_{m-k+1}] \quad \text{et} \quad \bar{\mathbf{H}}_m = \begin{bmatrix} \mathbf{I}_k & \mathbf{B}_{m-k} \\ \mathbf{0} & \bar{\mathbf{H}}_{m-k} \end{bmatrix} \quad (4.62)$$

La formulation GCRO-DR nous donne la relation d'Arnoldi généralisée suivante :

$$\mathbf{A} \hat{\mathbf{V}}_m = \mathbf{V}_{m+1} \bar{\mathbf{H}}_m \quad (4.63)$$



où

$$\hat{\mathbf{V}}_m = \begin{bmatrix} \tilde{\mathbf{U}}_k & \mathbf{V}_{m-k} \end{bmatrix}, \quad \mathbf{V}_{m+1} = [\mathbf{C}_k \quad \mathbf{V}_{m-k+1}] \quad \text{et} \quad \bar{\mathbf{H}}_m = \begin{bmatrix} \mathbf{D}_k & \mathbf{B}_{m-k} \\ \mathbf{0} & \bar{\mathbf{H}}_{m-k} \end{bmatrix} \quad (4.64)$$

où  $\mathbf{B}_{m-k} := \mathbf{C}_k^T \mathbf{A} \mathbf{Z}_{m-k}$  (on remplace  $\mathbf{Z}_{m-k}$  par  $\mathbf{V}_{m-k}$  dans le cas du GCRO-DR) qui contient les produits scalaires avec les colonnes de  $\mathbf{C}_k$ . La matrice  $\mathbf{D}_k$  est une matrice de mise à l'échelle introduite par Parks et al. [13] qui consiste à améliorer le comportement numérique des problèmes aux valeurs propres généralisés. Cette matrice est définie par  $\mathbf{D}_k = \text{diag}(\frac{1}{\|\mathbf{u}_1\|}, \frac{1}{\|\mathbf{u}_2\|}, \dots, \frac{1}{\|\mathbf{u}_k\|})$  telle que  $\tilde{\mathbf{U}}_k = \mathbf{U}_k \mathbf{D}_k$ .

Nous attirons l'attention sur le calcul de la matrice réduite  $\mathbf{B}_{m-k}$ . En effet, pour des raisons de performance, cette matrice ne doit pas se construire selon son expression directe  $\mathbf{C}_k^T \mathbf{A} \mathbf{Z}_{m-k}$ . Sachant que la matrice  $\bar{\mathbf{H}}_{m-k}$  est construite à partir des produits scalaires et des normes, il en est de même avec la sous-matrice  $\mathbf{B}_{m-k}$  qui contient des produits scalaires avec les colonnes de  $\mathbf{C}_k$ . Le remplissage des  $(m-k)$  dernières colonnes de  $\bar{\mathbf{H}}_m$  peut être vu comme la suite d'un processus d'Arnoldi, pour lequel les colonnes de  $\mathbf{C}_k$  forment les premiers vecteurs de la base de Krylov déjà construits. Côté implémentation, il suffira de faire un décalage sur le point de départ du processus d'Arnoldi.

La nouvelle solution  $\mathbf{x}_i$  optimale est de la forme  $\mathbf{x}_{i-1} + \mathbf{Z}_m \mathbf{y}$ , où  $\mathbf{y}$  est calculé en minimisant la norme du résidu associé  $\mathbf{r}_i^{(s)} = \mathbf{b}^{(s)} - \mathbf{A} \mathbf{x}_i^{(s)}$ . Ce dernier est tel que

$$\mathbf{r}_i^{(s)} = \mathbf{b}^{(s)} - \mathbf{A} \mathbf{x}_{i-1}^{(s)} - \mathbf{A} \mathbf{Z}_m \mathbf{y} = \mathbf{r}_{i-1}^{(s)} - \mathbf{V}_{m+1} \bar{\mathbf{H}}_m \mathbf{y} \quad (4.65)$$

grâce à la récurrence d'Arnoldi généralisée (4.61). On peut maintenant expliciter la norme de ce résidu

$$\|\mathbf{r}_i^{(s)}\| = \|\mathbf{r}_{i-1}^{(s)} - \mathbf{V}_{m+1} \bar{\mathbf{H}}_m \mathbf{y}\| = \|\mathbf{V}_{m+1}^T \mathbf{r}_{i-1}^{(s)} - \bar{\mathbf{H}}_m \mathbf{y}\| = \|\|\mathbf{r}_{i-1}^{(s)}\| \mathbf{e}_{k+1} - \bar{\mathbf{H}}_m \mathbf{y}\| \quad (4.66)$$

Nous sommes donc amenés à résoudre le problème aux moindres carrés :

$$\min \|\|\mathbf{r}_{i-1}^{(s)}\| \mathbf{e}_{k+1} - \bar{\mathbf{H}}_m \mathbf{y}\| \quad (4.67)$$

- La seconde étape consiste en une phase de déflation et de mise à jour de  $\mathbf{Z}_k$ ,  $\mathbf{U}_k$  et  $\mathbf{C}_k$ . Cette déflation est différente de celle du premier cycle au sens où on doit résoudre un problème aux valeurs propres généralisé. Commençons par rappeler une définition des vecteurs de Ritz harmoniques :

**Définition 1** Soit  $\mathcal{U}$  un sous-espace vectoriel de  $\mathbb{R}^N$ . Soient  $\mathbf{B} \in \mathbb{R}^{N \times N}$ ,  $\theta \in \mathbb{R}$  et  $\mathbf{u} \in \mathcal{U}$ . On dit que  $(\theta, \mathbf{u})$  est une paire de Ritz harmonique de l'opérateur  $\mathbf{B}$  par rapport à l'espace  $\mathcal{U}$  si et seulement si

$$\mathbf{B}\mathbf{u} - \theta\mathbf{u} \perp \mathbf{B}\mathbf{u}, \quad \forall \mathbf{u} \in \mathcal{U} \quad (4.68)$$

ou de manière équivalente

$$\mathbf{w}^T (\mathbf{B}\mathbf{u} - \theta\mathbf{u}) = \mathbf{0} \quad \forall \mathbf{w} \in \text{range}(\mathbf{B}\mathcal{U}) \quad (4.69)$$

On dit que  $\mathbf{u}$  est le vecteur de Ritz harmonique associé à la valeur de Ritz harmonique  $\theta$ .

D'après la définition ci-dessus, il existe différentes façons de calculer les vecteurs de Ritz selon le choix de l'opérateur  $\mathbf{B}$  et de l'espace  $\mathcal{U}$ . Carvalho et al. ont justement proposé 3 stratégies de déflation qui découlent de cette définition [56].

**Stratégie A** :  $\mathbf{B} := \mathbf{A}$  et  $\mathcal{U} := \text{range}(\mathbf{Z}_m)$

La formulation du problème aux valeurs propres généralisé dans le cas de la stratégie A s'écrit :

$$(\mathbf{A}\mathbf{Z}_m)^T (\mathbf{A}\mathbf{Z}_m\mathbf{y} - \theta\mathbf{Z}_m\mathbf{y}) = \mathbf{0} \quad (4.70)$$

où  $\mathbf{u} = \mathbf{Z}_m\mathbf{y}$ ,  $\mathbf{y} \in \mathbb{R}^m$  et  $\theta \in \mathbb{R}$ . En utilisant la relation d'Arnoldi (4.61), on obtient

$$\boxed{\bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m \mathbf{y} = \theta \bar{\mathbf{H}}_m^T \mathbf{V}_{m+1}^T \mathbf{Z}_m \mathbf{y}} \quad (4.71)$$

**Stratégie B** :  $\mathbf{B} := \mathbf{A}\mathbf{Z}_m\mathbf{V}_m^T$  et  $\mathcal{U} := \text{range}(\mathbf{V}_m)$

La formulation du problème aux valeurs propres généralisé dans le cas de la stratégie B s'écrit :

$$(\mathbf{A}\mathbf{Z}_m\mathbf{V}_m^T\mathbf{V}_m)^T (\mathbf{A}\mathbf{Z}_m\mathbf{V}_m^T\mathbf{V}_m\mathbf{y} - \theta\mathbf{V}_m\mathbf{y}) = \mathbf{0} \quad (4.72)$$

$$\Rightarrow (\mathbf{A}\mathbf{Z}_m)^T (\mathbf{A}\mathbf{Z}_m\mathbf{y} - \theta\mathbf{V}_m\mathbf{y}) = \mathbf{0} \quad (4.73)$$

où  $\mathbf{u} = \mathbf{V}_m\mathbf{y}$ ,  $\mathbf{y} \in \mathbb{R}^m$  et  $\theta \in \mathbb{R}$ . De même, on obtient d'après (4.61)

$$\boxed{\bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m \mathbf{y} = \theta \mathbf{H}_m^T \mathbf{y}} \quad (4.74)$$

**Stratégie C** :  $\mathbf{B} := \mathbf{A}\mathbf{Z}_m\mathbf{W}_m^\dagger$  et  $\mathcal{U} := \text{range}(\mathbf{W}_m)$  avec  $\mathbf{W}_m^\dagger := (\mathbf{W}_m^T\mathbf{W}_m)^{-1}\mathbf{W}_m^T$

$$(\mathbf{A}\mathbf{Z}_m\mathbf{W}_m^\dagger\mathbf{W}_m)(\mathbf{A}\mathbf{Z}_m\mathbf{W}_m^\dagger\mathbf{W}_m\mathbf{y} - \theta\mathbf{W}_m\mathbf{y}) = \mathbf{0} \quad (4.75)$$

$$\Rightarrow (\mathbf{A}\mathbf{Z}_m)^T (\mathbf{A}\mathbf{Z}_m\mathbf{y} - \theta\mathbf{W}_m\mathbf{y}) \quad (4.76)$$

où  $\mathbf{u} = \mathbf{W}_m\mathbf{y}$ ,  $\mathbf{y} \in \mathbb{R}^m$  et  $\theta \in \mathbb{R}$ . On obtient d'après (4.61)

$$\boxed{\bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m \mathbf{y} = \theta \bar{\mathbf{H}}_m^T \mathbf{V}_{m+1}^T \mathbf{W}_m \mathbf{y}} \quad (4.77)$$

Concernant la stratégie C, Carvalho et al. ont proposé une simplification du problème aux valeurs propres généralisé que nous détaillons dans l'Annexe A.

**Remarques :**

- La stratégie B correspond à la formulation qu'on retrouve dans la méthode GMRES-DR et sa version flexible.
- Dans le cas du FGMRES-DR, il est également possible de formuler le problème aux valeurs propres selon la stratégie A ou B. Ceci est possible puisqu'on a à notre disposition deux espaces  $\mathbf{V}_m$  et  $\mathbf{Z}_m$ .
- Dans le cas du GCRO-DR, on retrouve la stratégie A proposée initialement par Parks et al. dans [13] en considérant  $\mathcal{U} = \text{range}(\hat{\mathbf{V}}_m)$ , ce qui revient à remplacer  $\mathbf{Z}_m$  par  $\hat{\mathbf{V}}_m$  dans (4.71).
- Dans le cas du FGCRO-DR, il est possible de formuler le problème aux valeurs propres généralisé selon les trois stratégies car nous avons à notre disposition trois espaces qui sont  $\mathbf{V}_{m+1}$ ,  $\mathbf{Z}_m$  et  $\mathbf{W}_m$ . On retrouve la stratégie C proposée par Carvalho et al. [14].

Nous allons tester ces trois stratégies dans le cadre d'une résolution adjointe et voir si on obtient les mêmes conclusions numériques que dans [56].

#### 4.1.6 Cadre général des méthodes de recyclage

Les idées développées dans cette section sont essentiellement basées sur les travaux de Soodhalter et al. [57]. Dans cet article, les auteurs ont réalisé une investigation exhaustive des méthodes de recyclages au sens large. Dans notre cas, l'idée est de donner un cadre plus général des méthodes de recyclage d'espace de Krylov. Une propriété intéressante des méthodes de recyclage est leur capacité à combiner la déflation par *augmentation* et la déflation par *projection*. Contrairement à la déflation par augmentation, la déflation par projection permet à ces méthodes de recycler une information quelconque. Son principe consiste à projeter le problème initial orthogonalement à un sous-espace qui pénalise la convergence. Plus précisément, on résout le problème projeté  $\mathbf{P}\mathbf{A}\mathbf{x} = \mathbf{P}\mathbf{b}$  où  $\mathbf{P}$  est un opérateur de projection orthogonal ou oblique. Cette idée a été proposée par De Sturler [58] en proposant la méthode GCRO qui consiste à calculer la solution optimale sur la *somme directe* de l'espace de correction (ou l'espace de recyclage) et de l'espace de Krylov projeté en imposant une contrainte d'optimalité sur le résidu. Soulignons qu'en fonction de l'espace qu'on souhaite recycler, nous obtenons une méthode différente. Dans le cas du GCRO, l'espace de recyclage est engendré par les approximations d'erreur  $\mathbf{e}_k = \mathbf{x} - \mathbf{x}_k$ . De Sturler montre dans [58] que si on avait accès explicitement à  $\mathbf{e}_k$ , ce qui n'est jamais le cas, on obtiendrait la solution exacte. L'idée est de fournir une approximation de cette erreur à chaque itération  $k$  et de la recycler à chaque cycle du GCRO. Baker et al. [59] utilisent la

même idée qui consiste à recycler à chaque cycle du GMRES l'approximation de l'erreur, ce qui a donné naissance à la méthode LGMRES (Loose GMRES). Plus tard, De Sturler propose la méthode GCROT en recyclant un sous-espace de  $\text{range}(\mathbf{C}_k)$  qui minimise la norme de l'erreur résiduelle. Parks et al. [13] s'inspirent de Morgan [36] en recyclant le sous-espace invariant approché de l'opérateur  $\mathbf{A}$ , ce qui a donné naissance au solveur GCRO-DR. Le calcul de l'espace invariant approché s'effectue au moyen des vecteurs de Ritz harmoniques.

Considérons deux espaces vectoriels  $\mathcal{U} \subset \mathbb{R}^N$  de dimension  $k$  et  $\mathcal{V}_j \subset \mathbb{R}^N$  de dimension  $j$  tels que  $\mathcal{V}_j$  est généré par une méthode itérative. L'espace  $\mathcal{U} + \mathcal{V}_j$  est défini comme l'espace de correction. On définit également l'espace de contrainte  $\tilde{\mathcal{U}} + \tilde{\mathcal{V}}_j$  où  $\tilde{\mathcal{U}} \subset \mathbb{R}^N$  et  $\tilde{\mathcal{V}}_j \subset \mathbb{R}^N$ . Nous supposons que les quatre espaces sont en somme directe, c'est-à-dire que  $\dim(\mathcal{U} + \mathcal{V}_j) = \dim(\tilde{\mathcal{U}} + \tilde{\mathcal{V}}_j) = k + j$ . On définit alors la méthode par augmentation générale par la relation suivante :

$$\text{Choisir } \mathbf{s}_j \in \mathcal{U} \text{ et } \mathbf{t}_j \in \mathcal{V}_j \text{ tel que } \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{s}_j + \mathbf{t}_j) \perp (\tilde{\mathcal{U}} + \tilde{\mathcal{V}}_j). \quad (4.78)$$

Soient  $\mathbf{U}, \tilde{\mathbf{U}} \in \mathbb{R}^{N \times k}$  and  $\mathbf{V}_j, \tilde{\mathbf{V}}_j \in \mathbb{R}^{N \times j}$ , tels que  $\text{range}(\mathbf{U}) = \mathcal{U}$ ,  $\text{range}(\tilde{\mathbf{U}}) = \tilde{\mathcal{U}}$ . On définit également  $\mathbf{P} = \mathbf{U}(\tilde{\mathbf{U}}^T \mathbf{A} \mathbf{U})^{-1} \tilde{\mathbf{U}}^T \mathbf{A}$  le projecteur sur  $\mathcal{U}$  parallèlement à  $(\mathbf{A}^T \tilde{\mathcal{U}})^\perp$  et  $\mathbf{Q} = \mathbf{A} \mathbf{U}(\tilde{\mathbf{U}}^T \mathbf{A} \mathbf{U})^{-1} \tilde{\mathbf{U}}^T$  le projecteur sur  $\mathbf{A} \mathcal{U}$  parallèlement à  $\tilde{\mathcal{U}}^\perp$ . L'équation (4.78) peut être reformulée comme un problème projeté où  $\mathbf{V}_j \mathbf{y}_j$  est l'approximation de la solution associée

$$(\mathbf{I} - \mathbf{Q}) \mathbf{A} \mathbf{t} = (\mathbf{I} - \mathbf{Q}) \mathbf{r}_0 \quad (4.79)$$

Le résidu  $\hat{\mathbf{r}}_j$  vérifie la condition d'orthogonalité

$$\hat{\mathbf{r}}_j = (\mathbf{I} - \mathbf{Q})(\mathbf{r}_0 - \mathbf{A} \mathbf{V}_j \mathbf{y}_j) \perp \tilde{\mathcal{V}}_j \quad \text{et} \quad \mathbf{t}_j = \mathbf{V}_j \mathbf{y}_j, \quad (4.80)$$

et on utilise  $\mathbf{P}$  pour calculer

$$\mathbf{s}_j = \mathbf{P} \mathbf{t} - \mathbf{P} \mathbf{t}_j \quad \text{où} \quad \mathbf{P} \mathbf{t} = \mathbf{U}(\tilde{\mathbf{U}}^T \mathbf{A} \mathbf{U})^{-1} \tilde{\mathbf{U}}^T \mathbf{r}_0, \quad (4.81)$$

en substituant les expressions de  $\mathbf{P}$  et  $\mathbf{t}_j = \mathbf{V}_j \mathbf{y}_j$ , l'approximation de la solution à l'itération  $j$  s'exprime :

$$\mathbf{x}_j = \mathbf{x}_0 + \mathbf{U}(\tilde{\mathbf{U}}^T \mathbf{A} \mathbf{U})^{-1} \tilde{\mathbf{U}}^T \mathbf{r}_0 + \mathbf{V}_j \mathbf{y}_j - \mathbf{U} \mathbf{B}_j \mathbf{y}_j \quad \text{où} \quad \mathbf{B}_j = (\tilde{\mathbf{U}}^T \mathbf{A} \mathbf{U})^{-1} \tilde{\mathbf{U}}^T \mathbf{A} \mathbf{V}_j. \quad (4.82)$$

#### Remarques importantes :

- Il est important de souligner que le résidu du problème complet et du problème projeté sont égaux, cette remarque permet de s'affranchir du calcul complet du résidu et de gagner en performance. Le résidu complet est donné par

$$\mathbf{r}_j = \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{s}_j + \mathbf{t}_j) = \hat{\mathbf{r}}_j. \quad (4.83)$$

- Ce cadre général autorise un choix quelconque des espaces  $\mathcal{V}_j$  et  $\tilde{\mathcal{V}}_j$ . Néanmoins, le choix conditionne fortement les performances de la méthode. Il est donc important de procéder à un choix pertinent.

**Exemple 1 : méthode de recyclage pour l'algorithme FOM**

En prenant en compte la remarque précédente, il devient assez facile de mettre en oeuvre une méthode de recyclage selon le choix des espaces de correction et de contrainte qu'on impose. Par exemple, De Sturler et al. ont mis en place une méthode de recyclage basée sur la méthode FOM en choisissant  $\mathcal{V}_j = \mathcal{K}_j((\mathbf{I} - \mathbf{Q})\mathbf{A}, (\mathbf{I} - \mathbf{Q})\mathbf{r}_0)$  et où  $\mathbf{Q}$  est la projection oblique sur  $\mathbf{AU}$  parallèlement à  $\mathcal{U}^\perp$ . Dans ce cas, on a  $\mathbf{Q} = \mathbf{AU}(\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*$  et  $\mathbf{P} = \mathbf{U}(\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*\mathbf{A}$ . Selon le cadre décrit précédemment, on applique FOM au problème projeté (4.79) en prenant soin de stocker les coefficients de  $\mathbf{B}_j = (\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*\mathbf{AV}_j$  à chaque étape  $j$  du processus d'Arnoldi. Ensuite, on résout le problème linéaire

$$\mathbf{H}_j\mathbf{y}_j = \beta\mathbf{y}_j, \quad \beta = \|(\mathbf{I} - \mathbf{Q})\mathbf{r}_0\|, \quad (4.84)$$

Il suffit au final de mettre à jour la solution  $\mathbf{x}_j = (\mathbf{x}_0 + \mathbf{V}_j\mathbf{y}_j) + \mathbf{U}(\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*\mathbf{r}_0 - \mathbf{UB}_j\mathbf{y}_j$ .

**Exemple 2 : méthode de recyclage pour le GMRES-DR/GCRO-DR**

Cet exemple concerne spécifiquement les méthodes de recyclage qui minimisent le résidu sur l'espace augmenté  $\mathcal{U} + \mathcal{V}_j$  et qui se caractérisent par la condition d'optimalité suivante :

$$\text{Choisir } \mathbf{s}_j \in \mathcal{U} \text{ et } \mathbf{t}_j \in \mathcal{V}_j \text{ tels que } \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{s}_j + \mathbf{t}_j) \perp \mathbf{A}(\mathcal{U} + \mathcal{V}_j), \quad (4.85)$$

Le choix commun de toutes les méthodes basées sur le GCRO est  $\mathcal{V}_j = \mathcal{K}_j((\mathbf{I} - \mathbf{Q})\mathbf{A}, (\mathbf{I} - \mathbf{Q})\mathbf{r}_0)$  avec  $\mathbf{Q}$  le projecteur orthogonal sur l'espace  $\mathbf{AU}$ . Pratiquement, ces méthodes construisent une base orthonormale de  $\mathcal{K}_j((\mathbf{I} - \mathbf{Q})\mathbf{A}, (\mathbf{I} - \mathbf{Q})\mathbf{r}_0)$ . Soit  $\mathbf{C} \in \mathbb{R}^{N \times k}$  tel que  $\mathbf{C}^T\mathbf{C} = \mathbf{I}$ , et  $\text{range}(\mathbf{C}) = \mathbf{AU}$  tel que  $\mathbf{Q} = \mathbf{C}\mathbf{C}^T$ . La relation d'Arnoldi s'écrit alors :

$$(\mathbf{I} - \mathbf{Q})\mathbf{AV}_j = \mathbf{V}_{j+1}\bar{\mathbf{H}}_j \Leftrightarrow \mathbf{AV}_j = \mathbf{CB}_j + \mathbf{V}_{j+1}\bar{\mathbf{H}}_j \quad \text{où } \mathbf{B}_j = \mathbf{C}^T\mathbf{AV}_j, \quad (4.86)$$

Une écriture matricielle aboutit à

$$\mathbf{A} \begin{bmatrix} \mathbf{U} & \mathbf{V}_j \end{bmatrix} = \begin{bmatrix} \mathbf{C} & \mathbf{V}_{j+1} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{B}_j \\ \mathbf{0} & \bar{\mathbf{H}}_j \end{bmatrix} \quad (4.87)$$

Le calcul de la solution s'effectue en résolvant le problème de minimisation

$$(\mathbf{z}_j, \mathbf{y}_j) = \underset{\mathbf{z} \in \mathbb{R}^k, \mathbf{y} \in \mathbb{R}^j}{\operatorname{argmin}} \left\| \begin{bmatrix} \mathbf{I} & \mathbf{B}_j \\ \mathbf{0} & \overline{\mathbf{H}}_j \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{y} \end{bmatrix} - \begin{bmatrix} \mathbf{C}^T \mathbf{r}_0 \\ \beta \mathbf{e}_{k+j+1} \end{bmatrix} \right\| \quad \text{et} \quad \mathbf{s}_j = \mathbf{U} \mathbf{z}_j, \quad \mathbf{t}_j = \mathbf{V}_j \mathbf{y}_j, \quad (4.88)$$

où  $\beta = \|(\mathbf{I} - \mathbf{Q})\mathbf{r}_0\|$ . Il est suggéré dans [58] de résoudre (4.88) par bloc en résolvant d'abord l'équation en  $\mathbf{y}_j$  puis de déduire  $\mathbf{z}_j = \mathbf{C}^T \mathbf{r}_0 - \mathbf{B}_j \mathbf{y}_j$ . Cela équivaut à appliquer un GMRES directement sur le problème projeté (4.79) et de stocker les coefficients de  $\mathbf{B}_j$  obtenue par application de  $\mathbf{Q}$  lors du processus d'Arnoldi (4.86).

La mise à jour de la solution s'écrit alors

$$\mathbf{x}_j = \mathbf{x}_0 + \mathbf{U} \mathbf{C}^T \mathbf{r}_0 + \mathbf{V}_j \mathbf{y}_j - \mathbf{U}(\mathbf{B}_j \mathbf{y}_j), \quad (4.89)$$

Pour résumer, les méthodes de recyclage basées sur le GCRO vérifiant (4.85) dépendent du choix de  $\mathbf{U} + \mathbf{V}_j$ .

#### 4.1.7 Loose GCRO-DR

On rappelle que la méthode LGMRES consiste à recycler, à chaque cycle, les approximations de l'erreur définies comme étant l'écart entre la vraie solution et l'itéré courant. Ce recyclage est effectué en ajoutant dans l'espace de Krylov les  $k$  approximations de l'erreur

$$\mathbf{e}_i = \mathbf{x}_i - \mathbf{x}_{i-1}, \quad i=1, \dots, k \quad (4.90)$$

Il est important de rappeler que le choix de ce vecteur est pertinent car, dans le cadre du GCRO, il représente le choix optimal pour le calcul de la direction de recherche  $\mathbf{u}_i$ , au sens où il permet d'obtenir la solution exacte. De plus, le fait de recycler  $\mathbf{e}_i \in \mathcal{K}_m(\mathbf{A}, \mathbf{r}_{i-1})$  est pertinent car ce vecteur représente l'information provenant de l'espace de Krylov du cycle précédent. Par conséquent, recycler les  $k$  dernières approximations d'erreur de la forme (4.90) peut être un bon choix pour une méthode d'augmentation.

Le principe du LGMRES consiste à effectuer  $m - k$  étapes d'Arnoldi et à ajouter  $k$  approximations de l'erreur stockées dans la matrice  $\mathbf{E}_k = [\mathbf{e}_1, \dots, \mathbf{e}_k]$  aboutissant à une relation similaire à celle d'Arnoldi :

$$\mathbf{A} \mathbf{V}_m = \mathbf{V}_{m+1} \overline{\mathbf{H}}_m, \quad (4.91)$$

où  $\mathbf{V}_m = [\mathbf{V}_{m-k} \quad \mathbf{E}_k]$ . Les  $m - k + 1$  premières colonnes de  $\mathbf{V}_{m+1}$  correspondent aux vecteurs de Krylov orthonormalisés tandis que les  $k$  derniers vecteurs sont construits en orthogonalisant les vecteurs  $\mathbf{A} \mathbf{e}_i$ ,  $i = 1, \dots, k$  par rapport aux  $m - k + i$  premières colonnes de  $\mathbf{V}_{m+1}$ .

Comme pour la plupart des méthodes de recyclage, il est indispensable de déterminer

#### 4. STRATÉGIE DE RECYCLAGE D'ESPACE INVARIANT APPROCHÉ POUR LE SOLVEUR PARTITIONNÉ

un critère qui permet de décider ou non d'activer le recyclage. Il a été observé dans [59] que les vecteurs résidus alternaient de manière cyclique à la fin de chaque cycle GMRES. Notamment, on constate fréquemment une colinéarité des résidus du cycle  $i + 1$  et du cycle  $i - 1$  causant ainsi le ralentissement de la convergence. En recyclant l'approximation de l'erreur du cycle, il est possible d'éliminer des directions de recherche déjà exploitées dans les cycles précédents. Finalement, on comprend qu'il y a une relation entre l'angle des résidus et le comportement du GMRES, ce qui nous amène à introduire deux notions d'angle dans le cadre du LGMRES.

**Définition 2** *On définit l'angle séquentiel comme étant l'angle entre deux vecteurs résidus consécutifs, noté  $\angle(\mathbf{r}_{i+1}, \mathbf{r}_i)$ , et l'angle de saut l'angle entre le vecteurs résidu du cycle  $i - 1$  et celui du cycle  $i + 1$  qui est noté  $\angle(\mathbf{r}_{i+1}, \mathbf{r}_{i-1})$ .*

Les auteurs donnent un résultat qui permet de calculer ces angles de manière analytique pour le GMRES et le LGMRES. Le résultat suivant concerne l'angle séquentiel

$$\cos(\angle(\mathbf{r}_{i+1}, \mathbf{r}_i)) = \frac{\|\mathbf{r}_{i+1}\|}{\|\mathbf{r}_i\|}, \quad (4.92)$$

Cette formule nous dit qu'un angle séquentiel faible implique une convergence lente du solveur. Néanmoins, ce critère ne suffit pas pour caractériser la convergence du solveur. Il est important de considérer également l'angle de saut. En effet, une valeur importante de l'angle séquentiel (donc des résidus proches d'être orthogonaux) n'implique pas une amélioration de la convergence. Cela n'est vrai que si l'angle de saut est également important. On peut avoir deux résidus consécutifs qui sont proches d'être orthogonaux mais en même temps retomber sur une direction de recherche déjà exploitées. L'angle de saut nous indique finalement si le résidu pointe vers une nouvelle direction de recherche. L'angle de saut est défini par la formule ci-dessous

$$\cos(\angle(\mathbf{r}_{i+1}, \mathbf{r}_i)) = \frac{\|\mathbf{r}_{i+1}\|}{\|\mathbf{r}_i\|} - \frac{\langle \mathbf{A}\delta_{i+1}, \mathbf{A}\delta_i \rangle}{\|\mathbf{r}_{i+1}\| \|\mathbf{r}_{i-1}\|}, \quad (4.93)$$

où  $\mathbf{A}\delta_{i+1} = \mathbf{r}_i - \mathbf{r}_{i+1}$  et  $\mathbf{A}\delta_i = \mathbf{r}_{i-1} - \mathbf{r}_i$ .

Dans le cas du GCRO-DR, il est possible de simplifier la formule (4.93) grâce à une propriété remarquable des vecteurs  $\mathbf{e}_i$ . En effet, ces vecteurs sont  $\mathbf{A}^t \mathbf{A}$ -orthogonaux. Ceci permet d'éliminer le second terme de (4.93) pour obtenir la formule suivante

$$\cos(\angle(\mathbf{r}_{i+1}, \mathbf{r}_i)) = \frac{\|\mathbf{r}_{i+1}\|}{\|\mathbf{r}_i\|} \quad (4.94)$$

À ce stade, il est intéressant de recycler les approximations de l'erreur dans le cadre du GCRO-DR. Pour le GCRO-DR, Niu et al. ont observé un comportement des vecteurs résidu similaire à celui du GMRES-DR. Ils ont alors proposé la méthode LGCRO-DR. Par conséquent, il est possible d'améliorer la convergence du solveur GCRO-DR en recyclant

quelques approximations de l'erreur  $\mathbf{e}_i$ . Le LGCRO-DR (Algorithme 11, Annexe B p.133) a donc la faculté de combiner le recyclage d'un espace invariant approché et d'un espace engendré par quelques approximations de l'erreur. Niu et al. montrent qu'il existe une manière simple et économique d'implémenter cet algorithme.

Pour comprendre l'algorithme du LGCRO-DR, commençons par rappeler les relations importantes que vérifie le GCRO-DR. L'approximation de la solution s'écrit :

$$\mathbf{x}_i = \mathbf{x}_{i-1} + \mathbf{z}_i, \quad \text{avec} \quad \mathbf{z}_i = \hat{\mathbf{V}}_m \xi_i \quad (4.95)$$

où  $\mathbf{z}_i$  est la correction optimale de la solution  $\mathbf{x}_{i-1}$  du cycle précédent  $i - 1$ .

#### 4.1.7.1 Cas 1 : Recyclage d'une seule approximation d'erreur

Considérons d'abord le cas où on souhaite recycler une seule approximation de l'erreur. En reprenant la formule de calcul des vecteurs de Ritz harmoniques  $\tilde{\mathbf{Y}}_k = \hat{\mathbf{V}}_m \mathbf{P}_k$  et  $\mathbf{z}_i = \hat{\mathbf{V}}_m \xi_i$ , on remarque que le recyclage du vecteur  $\mathbf{z}_i$  équivaut à stocker le vecteur  $\xi_i$  dans la matrice  $\mathbf{P}_{k-1}$ . Autrement dit, il suffit de former la matrice augmentée  $\mathbf{P}_k = [\mathbf{P}_{k-1}, \xi_i]$ . À ce stade, l'ajustement des espaces  $\mathbf{C}_k$  et  $\mathbf{U}_k$  devient économique au sens des relations suivantes :

$$\mathbf{QR} = \bar{\mathbf{H}}_m \mathbf{P}_k, \quad (4.96)$$

$$\mathbf{C}_k = \mathbf{V}_{m+1} \mathbf{Q}, \quad (4.97)$$

$$\mathbf{U}_k = \tilde{\mathbf{Y}}_k \mathbf{R}^{-1}, \quad (4.98)$$

#### 4.1.7.2 Cas 2 : Recyclage de plusieurs approximations de l'erreur

Dans le cas où on souhaite recycler  $p$  approximations de l'erreur (avec  $p \geq 2$ ), l'astuce précédente ne s'applique pas directement sauf à introduire des hypothèses simplificatrices dans la formulation du recyclage de plusieurs erreurs. Il existe donc deux approches pour recycler plus d'un vecteur :

##### Approche directe

Cette approche consiste à augmenter l'espace de déflation  $\tilde{\mathbf{Y}}_{k-p}$  par les vecteurs  $\mathbf{z}_j \in \mathbb{R}^N$  pour  $j = i, i - 1, \dots, i - p + 1$  où l'indice  $i$  représente le cycle courant. Ensuite, on effectue une factorisation QR de cette matrice de la façon suivante :

$$\mathbf{A} \tilde{\mathbf{Y}}_k = \mathbf{QR}, \quad (4.99)$$

où  $\mathbf{Q} \in \mathbb{R}^{N \times k}$  et  $\mathbf{R} \in \mathbb{R}^{k \times k}$ . Au final, il ne reste plus qu'à effectuer (4.97) - (4.98). Bien que cette approche soit précise, elle reste assez coûteuse du fait de la factorisation  $\mathbf{QR}$  qui est indexée sur la taille du système.



### Approche économique

Pour contourner le problème du coût de la factorisation, on procède à une approximation afin d'obtenir une méthode économique mais inexacte. Le principe consiste à retenir  $\xi_j$  pour  $j = i, i - 1, \dots, i - p + 1$  et de les ajouter à la fin de la matrice  $\mathbf{P}_{k-p}$  tel que  $\mathbf{P}_k = [\mathbf{P}_{k-p}, \xi_i, \dots, \xi_{i-p+1}]$ . Ensuite, on applique (4.96) - (4.98). Dans notre étude du LGCRO-DR, nous retiendrons l'approche économique qui a aussi l'avantage d'avoir le même coût de calcul que le GCRO-DR.

#### 4.1.8 Loose FGCRO-DR

Cette section a pour but d'étendre l'étude du recyclage d'approximation d'erreur dans le cas du FGCRO-DR. À notre connaissance, aucune étude n'a été réalisée à ce sujet. Une possibilité à cela vient du fait que la formulation dans le cas flexible n'est pas aussi consistante que dans le cas non flexible. En effet, l'approximation de l'erreur s'écrit  $\mathbf{z}_i = \mathbf{Z}_m \xi_i$ . Si on considère la stratégie C adoptée par Carvalho et al. dans [14], il devient alors clair que l'ajout du vecteur  $\mathbf{z}_i$  dans l'espace  $\tilde{\mathbf{Y}}_k = \mathbf{W}_m \mathbf{P}_k$  n'a plus de sens puisque il y a un mélange entre les espaces  $\mathbf{W}_m$  et  $\mathbf{Z}_m$ . Par conséquent, aucune des approches décrites ci-dessus n'est applicable.

Une solution consiste à adopter la stratégie A pour la déflation, car dans ce cas, nous avons  $\tilde{\mathbf{Y}}_k = \mathbf{Z}_m \mathbf{P}_k$ . Il est donc à nouveau possible d'appliquer l'approche économique.

#### 4.1.9 Critère d'activation pour le recyclage d'espace invariant approché

Cette section a pour but de formuler un critère dynamique qui consiste à activer ou non la phase de recyclage de l'espace invariant approché  $\text{range}(\mathbf{C}_k)$ .

Pour obtenir un critère pertinent, il est important d'analyser la convergence du GCRO-DR. Parks et al. [13] ont proposé une analyse de convergence de ce solveur dans le but de quantifier la qualité d'approximation de l'espace  $\text{range}(\mathbf{C}_k)$  par rapport au vrai espace invariant de  $\mathbf{A}$ . Le résultat ci-dessous concerne la borne supérieure de la convergence des méthodes de recyclage et explique qu'il n'est pas nécessaire d'avoir une approximation précise pour améliorer le taux de convergence.

Soit  $\mathcal{Q}$  un espace invariant de dimension  $l$  de  $\mathbf{A}$  et soit  $\mathcal{C} = \text{range}(\mathbf{C}_k)$  un espace de dimension  $k$  ( $k \geq l$ ) qui approche  $\mathcal{Q}$ . On définit  $\Pi_{\mathcal{C}}$  et  $\Pi_{\mathcal{Q}}$  les projecteurs orthogonaux sur  $\mathcal{C}$  et  $\mathcal{Q}$  respectivement. On définit également  $\mathbf{P}_{\mathcal{Q}}$  le projecteur spectral sur  $\mathcal{Q}$ . Pour finir, on définit la distance dite *unilatérale* entre  $\mathcal{Q}$  et  $\mathcal{C}$  par :

$$\delta(\mathcal{Q}, \mathcal{C}) = \|(\mathbf{I} - \Pi_{\mathcal{C}})\Pi_{\mathcal{Q}}\|, \quad (4.100)$$

qui représente le sinus du plus grand angle principal entre  $\mathcal{Q}$  et  $\mathcal{C}$ . Une autre manière d'interpréter cette quantité est de dire que n'importe quel vecteur unitaire de  $\mathcal{Q}$  possède une composante qui est au plus égale à  $\delta$  dans la direction orthogonale à  $\mathcal{C}$ .

**Théorème 2** Soit  $\mathcal{C}$  un espace quelconque et  $\mathcal{V} = \text{range}(\mathbf{V}_{m-k}\bar{\mathbf{H}}_{m-k})$  l'espace de Krylov de dimension  $m - k$  généré par le GCRO-DR. Soit  $\mathbf{r}_0 \in \mathbb{R}^N$  et  $\mathbf{r}_1 = (\mathbf{I} - \Pi_{\mathcal{C}})\mathbf{r}_0$ . Alors, pour tout  $\mathcal{Q}$  tel que  $\delta(\mathcal{Q}, \mathcal{C}) < 1$  on a

$$\min_{\mathbf{d}_1 \in \mathcal{V} \oplus \mathcal{C}} \|\mathbf{r}_0 - \mathbf{d}_1\| \leq \min_{\mathbf{d}_2 \in (\mathbf{I} - \mathbf{P}_{\mathcal{Q}})\mathcal{V}} \|(\mathbf{I} - \mathbf{P}_{\mathcal{Q}})\mathbf{r}_1 - \mathbf{d}_2\| + \frac{\gamma}{1 - \delta} \|\mathbf{P}_{\mathcal{Q}}\| \cdot \|(\mathbf{I} - \Pi_{\mathcal{V}})\mathbf{r}_1\|, \quad (4.101)$$

où  $\gamma = \|(\mathbf{I} - \Pi_{\mathcal{C}})\mathbf{P}_{\mathcal{Q}}\|$ . Le membre de gauche représente la norme du résidu après  $m - k$  itérations du GCRO-DR avec recyclage de l'espace  $\mathcal{C}$ . Le premier terme du membre de droite représente la convergence du problème projeté où toutes les composantes dans  $\mathcal{Q}$  ont été éliminées, ce qui aboutit à une meilleure convergence. Le second terme représente la norme du résidu après  $m - k$  itérations du GCRO-DR pour le résidu  $\mathbf{r}_1$ .

Si l'espace de recyclage  $\mathcal{C}$  contient l'espace invariant  $\mathcal{Q}$ , alors  $\delta = \gamma = 0$  et le GCRO-DR converge au moins aussi vite que le problème projeté. Ceci reste valable tant que  $\|\mathbf{P}_{\mathcal{Q}}\|$  n'est pas trop importante. Parks et al. [13] remarquent que  $\delta = O(10^{-2})$  est facilement atteignable et suffit pour améliorer la convergence du GCRO-DR.

Un bon critère finalement serait d'estimer la distance entre l'espace de recyclage  $\text{range}(\mathbf{C}_k)$  et l'espace invariant  $\mathcal{Q}$  de  $\mathbf{A}$ . Comme nous n'avons pas accès à l'opérateur  $\mathbf{A}$ , il n'est pas possible de calculer l'espace invariant  $\mathcal{Q}$ . Une alternative consiste à calculer la distance entre  $\mathbf{C}_{k_i}^{(i)}$  et  $\mathbf{C}_{k_{i+1}}^{(i+1)}$  qui représentent respectivement les espaces de recyclage calculés au  $i^{\text{ème}}$  et au  $(i + 1)^{\text{ème}}$  couplage fluide-structure. Pour calculer la distance entre ces deux espaces, on utilise la distance de Grassmann définie par la formule suivante :

$$d_p(\mathbf{C}_{k_i}^{(i)}, \mathbf{C}_{k_{i+1}}^{(i+1)}) = \left( \sum_{i=1}^p \theta_i^2 \right)^{1/2} \quad p = \min(k_i, k_{i+1}), \quad (4.102)$$

où  $\theta_i$  constituent les angles principaux entre les colonnes de  $\mathbf{C}_{k_i}^{(i)}$  et  $\mathbf{C}_{k_{i+1}}^{(i+1)}$ . Plus précisément, l'angle principal est défini par :

$$\cos\theta_j = \langle \mathbf{c}_j^{(i)}, \mathbf{c}_j^{(i+1)} \rangle \quad \forall j = 1, \dots, p \quad (4.103)$$

Cette formule est valable pour des vecteurs unitaires ce qui est le cas pour les colonnes de  $\mathbf{C}_k$ . Il est possible de calculer les angles principaux grâce à la décomposition en valeurs singulières de la matrice  $(\mathbf{C}_{k_i}^{(i)})^T \mathbf{C}_{k_{i+1}}^{(i+1)}$

$$(\mathbf{C}_{k_i}^{(i)})^T \mathbf{C}_{k_{i+1}}^{(i+1)} = \mathbf{U}\Sigma\mathbf{V}^T \quad (4.104)$$

où  $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_p)$

Ainsi, les angles principaux vérifient

$$\cos\theta_i = \sigma_i \quad (4.105)$$

## 4.2 Résultats numériques

### 4.2.1 Aile M6 Analyse aéroélastique

Les expériences numériques ont été réalisées sur la même configuration de l'aile M6. Un maillage structuré multibloc composé d'une topologie C-H est utilisé (voir Figure 4.2). Nous conservons le même modèle de turbulence à savoir le modèle Spalart-Allmaras à une équation. Les distributions du coefficient de pression en configuration rigide et à l'équilibre aéroélastique sont présentées en Figure 4.1. On observe le double choc en forme de "Lambda" typique de la configuration de l'aile M6. Pour l'analyse aéroélastique, un modèle éléments finis simplifié a été conçu (voir Figure 4.2). Le déplacement vertical maximal est estimé à 0.14 mètres soit 11.7 % de l'envergure de l'aile, ce qui correspond à un fort couplage fluide-structure.

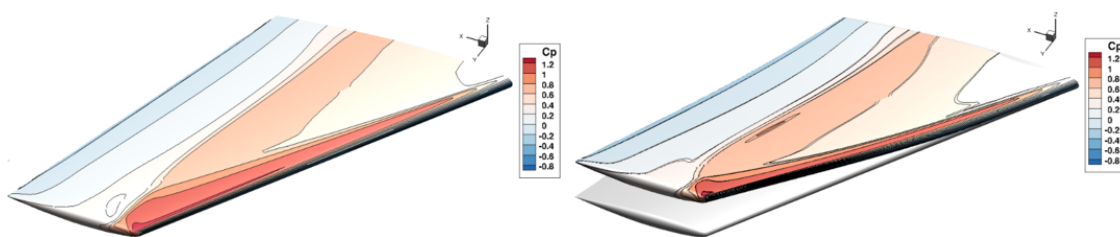


FIGURE 4.1 : Contours du coefficient de pression pour les écoulements stationnaires rigides et aéroélastiques.

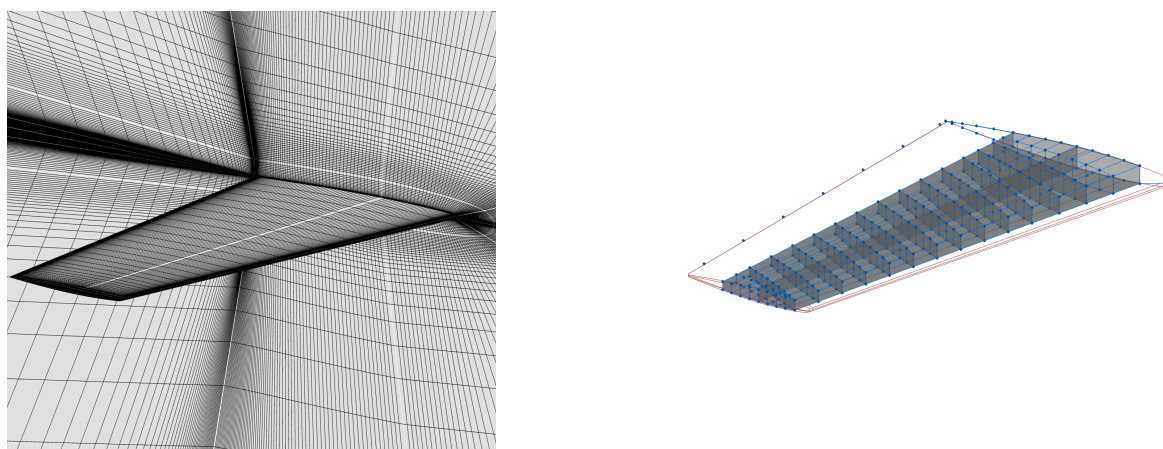


FIGURE 4.2 : Modèle aéroélastique de l'aile M6 : maillage RANS CFD à 42 blocs et schéma interne FEM.

Afin de mieux appréhender l'effet de la flexibilité sur la distribution de pression, nous avons reporté sur la Figure 4.3 la distribution de  $C_p$  pour deux sections : l'une à  $y = 0.60$  m et l'autre à  $y = 1.12$  m. Les distributions des déplacements verticaux associés aux longerons avant et arrière ainsi que la distribution des incréments de torsion sont tracées sur la Figure 4.4. L'analyse rigide donne un coefficient de portance  $C_L = 0.27$  tandis que l'analyse aéroélastique donne un coefficient de portance  $C_L = 0.23$  pour le même angle d'attaque.

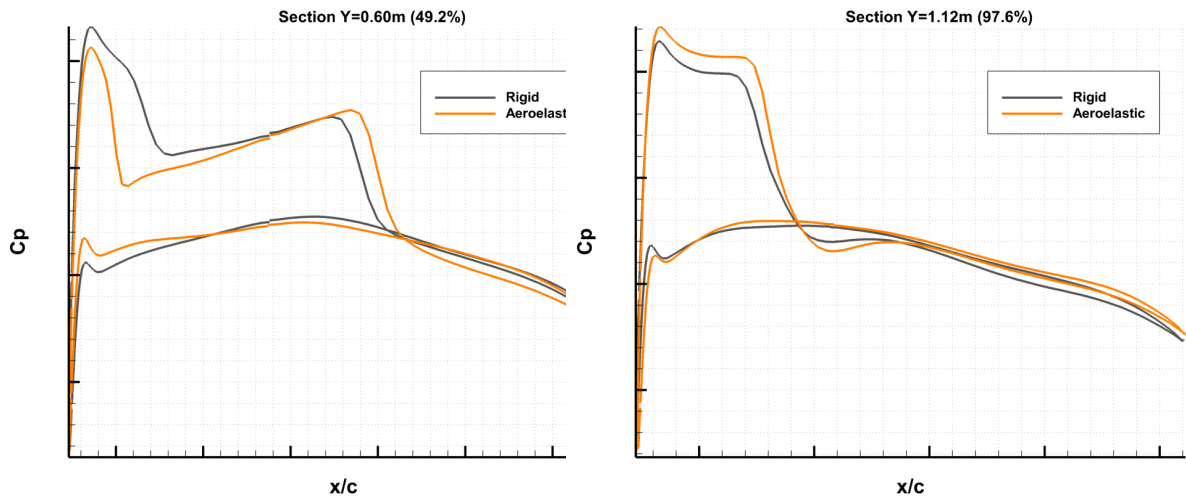


FIGURE 4.3 : Comparaison des tracés de section des coefficients de pression rigides et aéroélastiques à  $y=0.60$ m et  $y=1.12$ m.

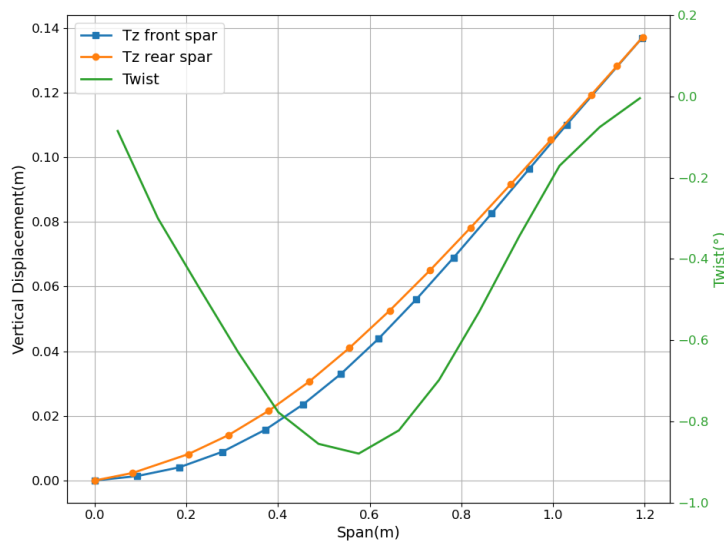


FIGURE 4.4 : Distribution du déplacement vertical et de l'incrément de torsion à l'équilibre aéroélastique.

### 4.2.2 Évaluation des performances dans le cas d'un système adjoint aérodynamique

L'objectif de cette section est de comparer les performances du solveur GCRO-DR standard et flexible avec celles du solveur GMRES-DR pour un problème adjoint fluide. Un résultat théorique montre que, sous une certaine condition de colinéarité, les deux solveurs sont algébriquement équivalents. Dans le cas standard, cette condition de colinéarité est systématiquement satisfaite. Pour aller plus loin, la démonstration du théorème se trouve dans [14]. Puisque nous résolvons un système adjoint fluide, nous ignorons l'étape de recyclage du FGCRO-DR (étape 2 - étape 4 de l'Algorithme 10, Annexe B p. 132).

La Figure 4.5 illustre la convergence des deux solveurs GMRES-DR et GCRO-DR pour un espace de Krylov de 120 vecteurs et un espace de déflation de 40 vecteurs. On constate l'équivalence numérique et algébrique des deux solveurs avec la superposition des deux courbes. On observe la même similitude dans le cas flexible (Figure 4.6). Nous soulignons cependant que la convergence s'est faite en deux cycles seulement. Sachant que le premier cycle est rigoureusement identique pour le FGMRES-DR et le FGCRO-DR, la différence n'est observée qu'à partir du deuxième cycle.

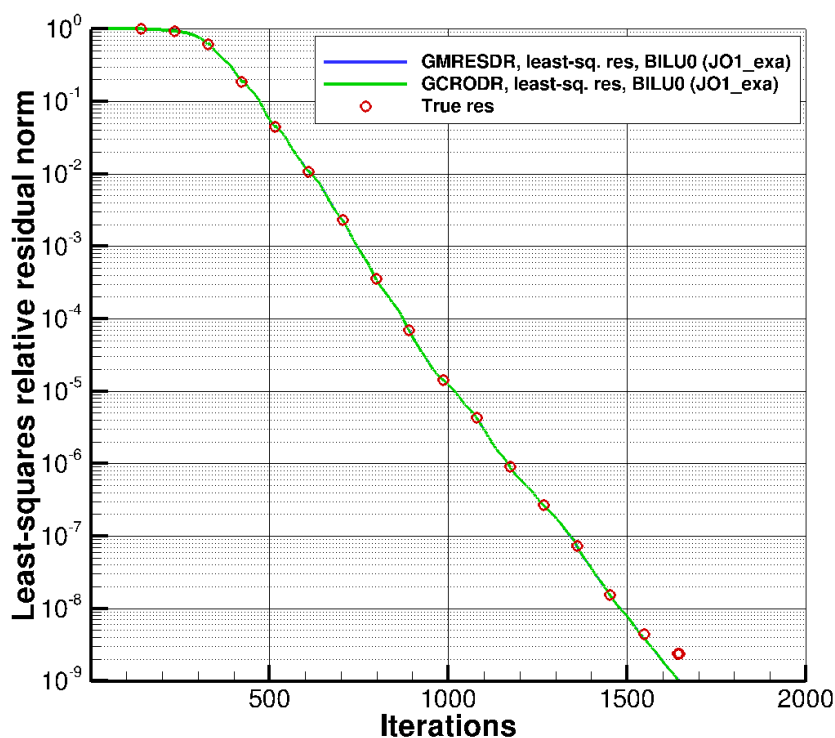


FIGURE 4.5 : Historique de convergence de la norme du résidu relatif. Équivalence algébrique et numérique des solveurs GMRES-DR et GCRO-DR préconditionnés par un BILU(0) appliqué à jacobienne exacte d'ordre 1. Nous rappelons les paramètres numériques :  $m = 120$ , et  $k = 40$ .

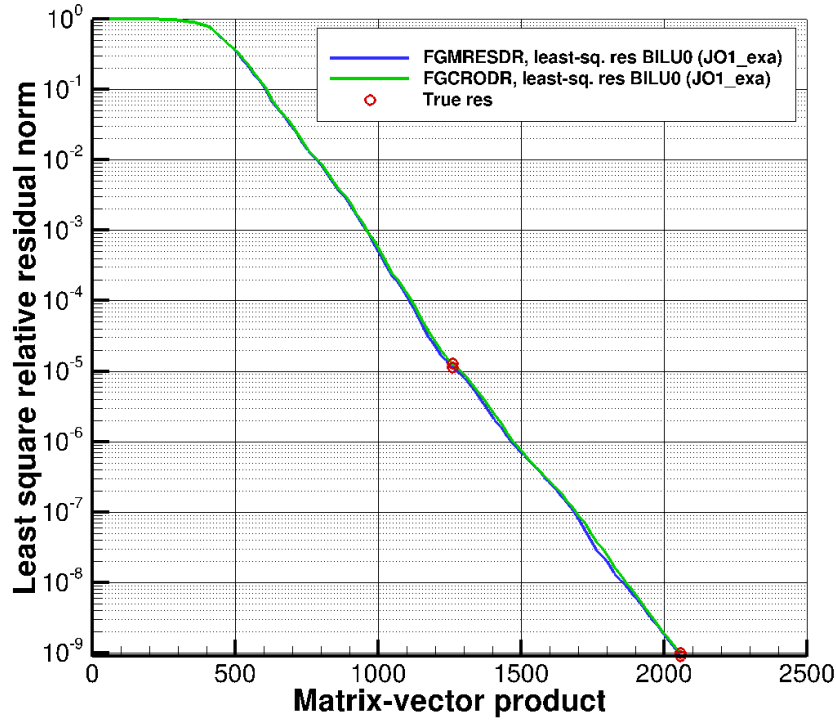


FIGURE 4.6 : Historique de convergence de la norme du résidu relative. L'équivalence algébrique et numérique des solveurs FGMRES-DR et FGCRO-DR n'est pas systématiquement vérifiée. Nous rappelons les paramètres numériques :  $m = 60$ ,  $m_i = 20$  et  $k = 20$ .

La Figure 4.7 illustre l'effet des trois stratégies de déflation décrites dans la sous-section 4.1.5.2 dans le cadre d'une résolution d'un système adjoint purement fluide. Les courbes montrent que les trois stratégies sont assez comparables. Les expériences numériques ont montré que la taille de l'espace de recyclage avait un impact non négligeable sur la convergence. Plus précisément, en prenant un espace de recyclage  $k$  qui vaut  $m/2$ , le comportement du FGCRO-DR était plus stable entre les 3 stratégies de déflation que si on considérait le cas classique  $k = m/3$ . Il faut garder à l'esprit que dans le cadre de la méthode GCRO flexible, nous avons à notre disposition trois espaces qui sont  $\mathbf{V}_m$ ,  $\mathbf{Z}_m$  et  $\mathbf{W}_m$ . Il n'y a aucune justification théorique qui montre qu'un espace est meilleure que l'autre. Par conséquent, les conclusions faites sur ces stratégies de déflation sont difficilement généralisables. Nous allons montrer dans le cas d'une résolution partitionnée que les conclusions ne sont pas les mêmes.

Pour se rendre compte du réel intérêt du solveur GCRO-DR par rapport au GMRES-DR, nous avons approfondi l'étude comparative des deux solveurs en considérant le LUSGS et le BILU(0) appliqués à  $\mathbf{J}_{O1}^{APP}$ . La Figure 4.8 illustre la convergence du GMRES-DR(120,40) pour le système adjoint fluide. Sur la Figure 4.8a, on remarque que le résidu exact stagne à  $10^{-7}$  pour le LUSGS et à  $10^{-5}$  pour le BILU appliqué à  $\mathbf{J}_{O1}^{APP}$ . Ce dernier semble être moins performant que le préconditionneur LUSGS du fait que le LUSGS a un effet global grâce aux échanges d'information entre les sous-domaines tandis que l'effet du BILU(0)

#### 4. STRATÉGIE DE RECYCLAGE D'ESPACE INVARIANT APPROCHÉ POUR LE SOLVEUR PARTITIONNÉ

demeure local. Pour éliminer cette stagnation, on applique simplement un redémarrage classique sans déflation (Figure 4.8b). On remarque que le solveur GCRO-DR(120,40) est numériquement plus robuste que le GMRES-DR puisque la norme du résidu exact et celle du résidu projeté coïncident parfaitement sauf dans le cas du BILU(0) appliqué à  $\mathbf{J}_{O1}^{APP}$  (voir Figure 4.9). Une façon de remédier à ce problème est d'effectuer un redémarrage standard sans déflation lorsque les résidus commencent à être significativement différents.

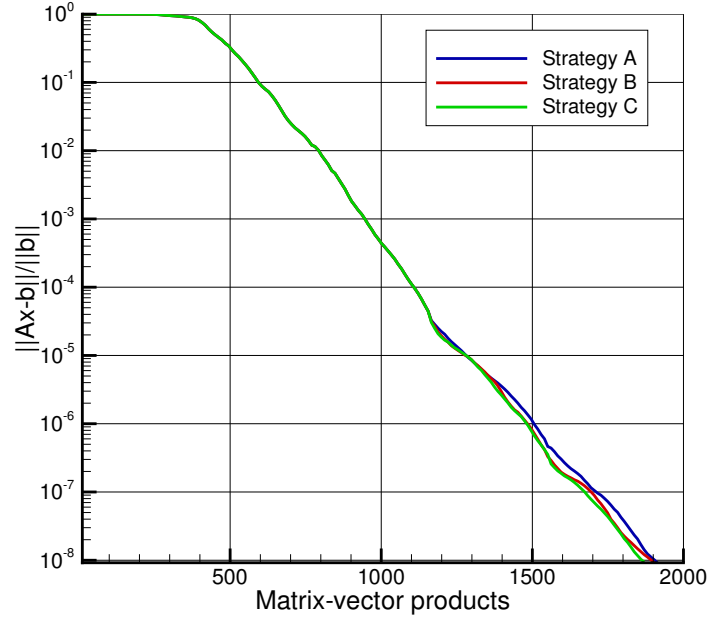


FIGURE 4.7 : Historique de convergence de la norme du résidu relative. Comparaison des trois stratégies de déflation.

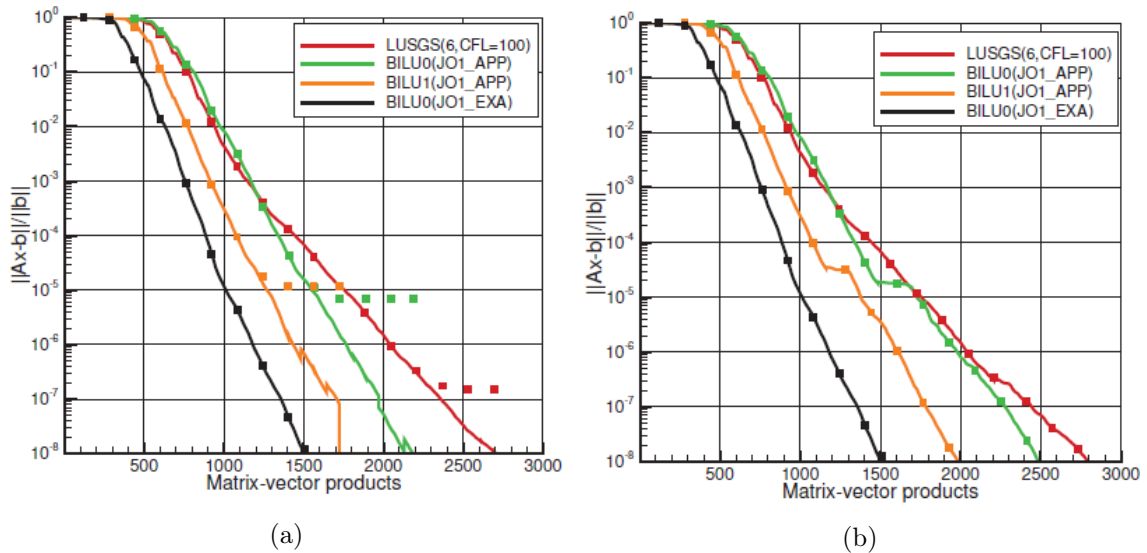


FIGURE 4.8 : Historique de convergence de la norme du résidu relative pour le solveur GMRES-DR(120,40). Impact des différents préconditionneurs. Dans (a), une certaine stagnation du résidu exact se produit en raison de la propagation des erreurs d'arrondi pendant le processus de redémarrage. Dans (b), un redémarrage sans déflation permet d'éliminer cette stagnation au prix d'un coût de calcul plus important.

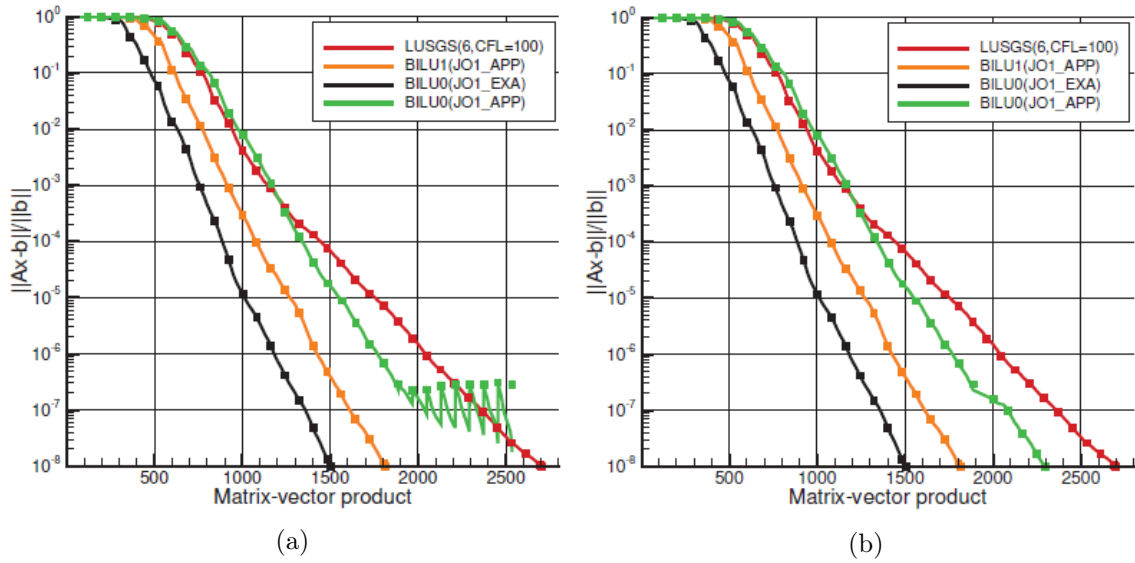


FIGURE 4.9 : Historique de convergence de la norme du résidu relatif pour le solveur GCRO-DR(120,40). Impact des différents préconditionneurs. Dans (a), une convergence erratique du résidu des moindres carrés associée à une stagnation du résidu exact se produit dans le cas du BILU(0) appliquée à la matrice jacobienne approchée du premier ordre. Dans (b), un redémarrage sans déflation permet d'éliminer cette stagnation au prix d'un coût de calcul plus important.

### 4.2.3 Effet de l'approximation des erreurs sur le système adjoint aérodynamique

Cette section a pour but de démontrer numériquement l'effet bénéfique du recyclage des approximations des erreurs  $\mathbf{e}_i = \mathbf{x} - \mathbf{x}_i$  à chaque cycle du GCRO-DR. Nous avons considéré le cas où on recycle une seule approximation de l'erreur et le cas où on recycle deux approximations de l'erreur en appliquant l'approche économique. Pour la résolution du système adjoint fluide, nous avons considéré le cas  $m = 30$ ,  $k = 10$  et  $p = 1, 2$  où  $p$  est le nombre de vecteurs  $\mathbf{e}_i$  qu'on recycle. La Figure 4.10 montre comment le recyclage de l'approximation d'erreur améliore la robustesse du solveur GCRO-DR. Plus précisément, le solveur GCRO-DR(60,20) converge en 1900 itérations. Même si un plateau apparaît au début, la convergence du GCOR-DR(60,20) semble suivre la propriété de convergence superlinéaire du GMRES sans redémarrage. La Figure 4.11 illustre la variation des angles séquentiels ainsi que des angles de saut associés au solveur GCRO-DR(60,20) tout au long des cycles. Nous observons au début des cycles des valeurs faibles pour les deux angles, ce qui justifie l'apparition du plateau. En revanche, les angles séquentiels et de saut augmentent progressivement au fur et à mesure des cycles. Plus précisément, l'angle de saut atteint une valeur de 80 degrés tandis que l'angle séquentiel atteint une valeur de 60 degrés. Dans le cas du solveur GCRO-DR(30,10) (Figure 4.12), nous observons une stagnation tout au long des cycles, ce qui se manifeste au niveau des angles par des valeurs très faibles ne dépassant pas les 2 degrés. Par contre, le recyclage d'une seule approximation de l'erreur permet de récupérer la convergence du système adjoint en 9000 itérations. La



#### 4. STRATÉGIE DE RECYCLAGE D'ESPACE INVARIANT APPROCHÉ POUR LE SOLVEUR PARTITIONNÉ

Figure 4.13 montre une variation des angles séquentiels. En moyenne, l'angle de saut est estimé à 30 degrés. En recyclant deux approximations de l'erreur (Figure 4.14), la courbe de convergence est légèrement améliorée avec un gain de 10 % par rapport à  $p = 1$ .

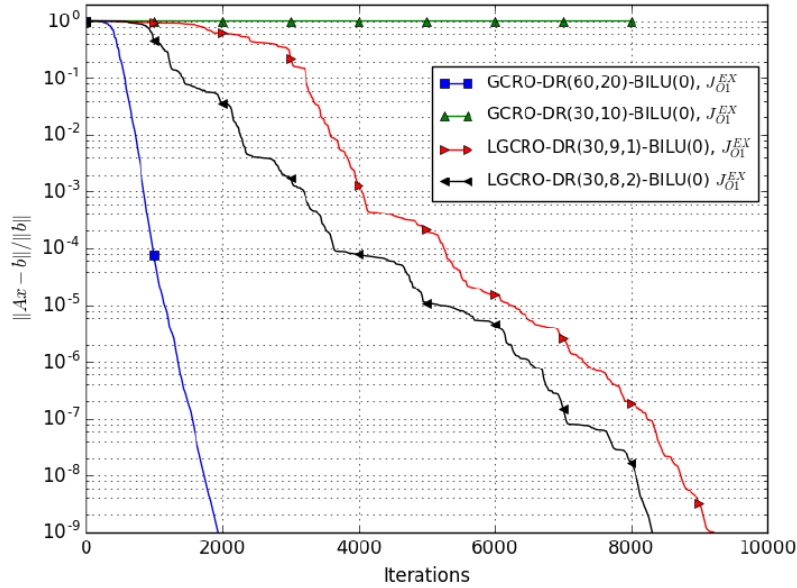


FIGURE 4.10 : Historique de convergence du résidu relatif des solveurs GCRO-DR(60,20) et LGCRO-DR(30,10, $p$ ) pour  $p = 1$  et  $p = 2$ . Impact du recyclage des approximations de l'erreur sur la convergence du système adjoint fluide.

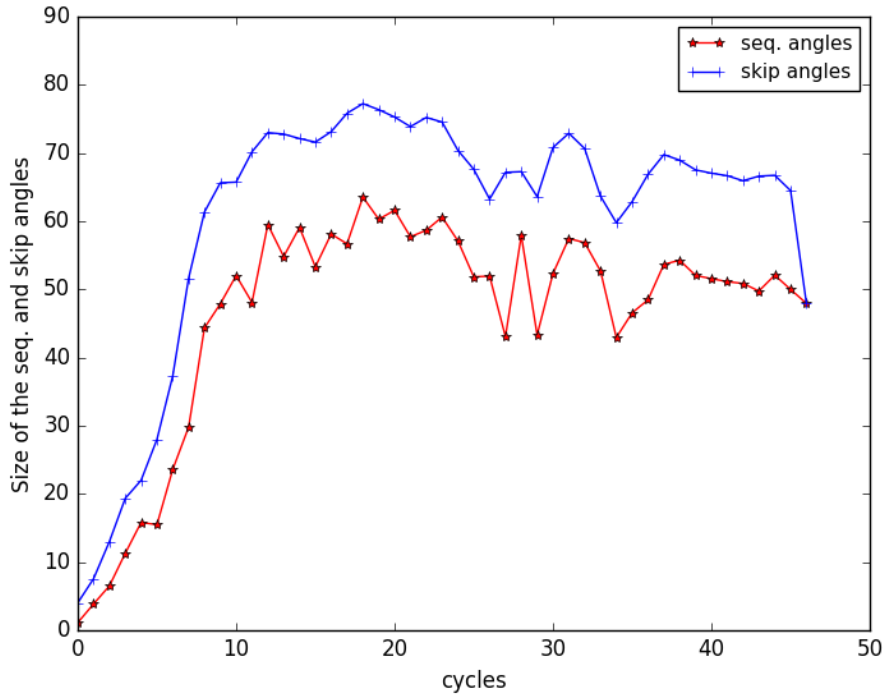


FIGURE 4.11 : Angles séquentiels et de saut pour le solveur GCRO-DR(60,20).

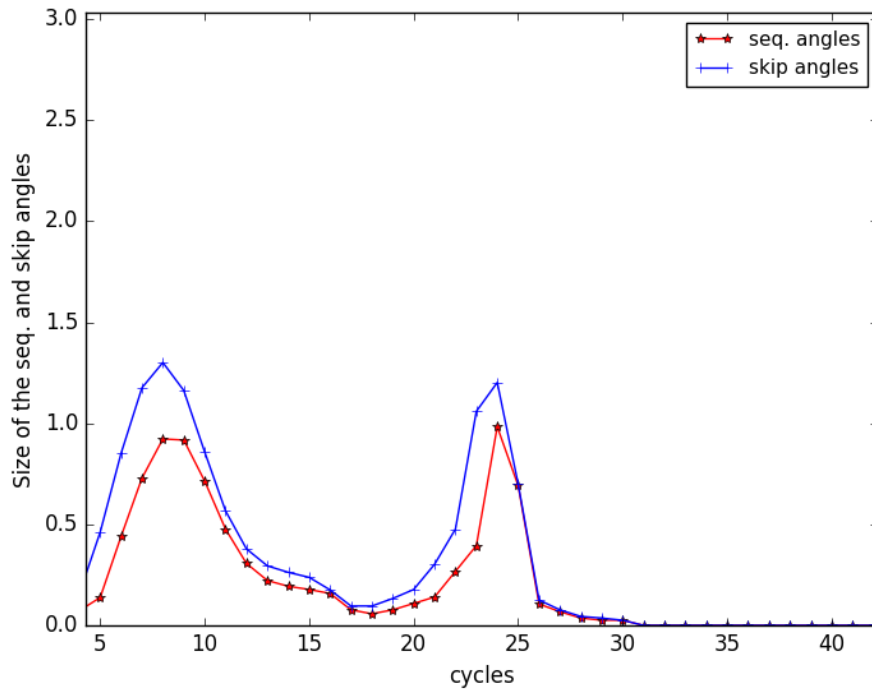


FIGURE 4.12 : Angles séquentiels et de saut pour le solveur GCRO-DR(30,10).

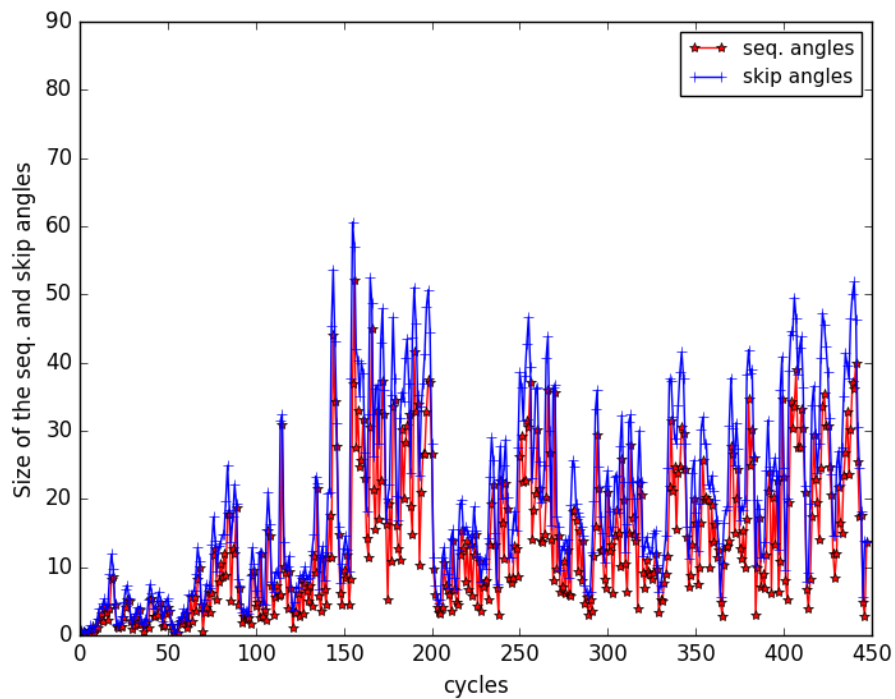


FIGURE 4.13 : Angles séquentiels et de saut pour le solveur GCRO-DR(30,9,1).

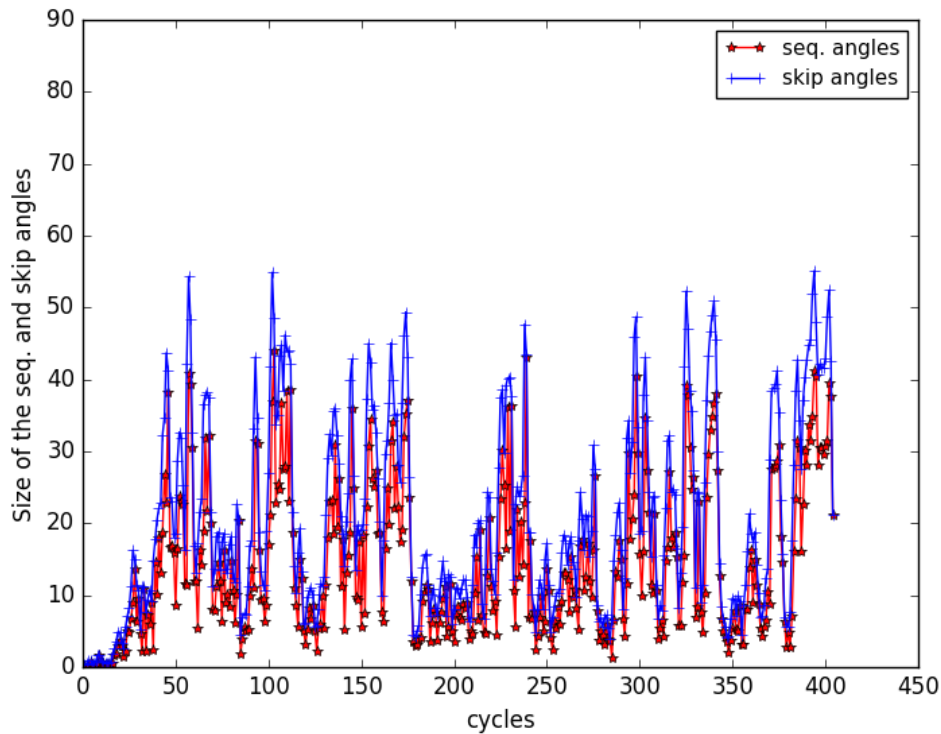


FIGURE 4.14 : Angles séquentiels et de saut pour le solveur GCRO-DR(30,8,2).

#### 4.2.4 Évaluation des performances du solveur partitionné dans le cas d'un système adjoint couplé

Nous rappelons que le solveur partitionné souffre d'un manque de robustesse lors de la résolution de systèmes adjoints fortement couplés et peut même diverger dans les cas extrêmes. Cette section a pour but de démontrer numériquement l'amélioration de la résolution partitionnée. Pour cela, nous allons dans un premier temps améliorer la robustesse du solveur partitionné en intégrant le préconditionneur BILU(0) appliqué aux matrices jacobiniennes  $\mathbf{J}_{O1}^{APP}$  et  $\mathbf{J}_{O1}^{EXA}$ . Ensuite, nous allons mettre en place le recyclage d'espace invariant approché entre chaque système linéaire.

La Figure 4.15 illustre la convergence du solveur partitionné au moyen du FGMRES-DR(60,20,20) en fonction du préconditionneur. La courbe bleue correspond à la courbe de référence qui fait appel au LU-SGS. L'application du BILU(0) sur  $\mathbf{J}_{O1}^{EXA}$  permet d'obtenir le meilleur gain en termes de coût de calcul avec un gain d'environ 79 % en nombre d'itérations. La Figure 4.16 illustre la convergence du solveur partitionné au moyen du GCRO-DR(140,42). La courbe noire correspond à la convergence du système adjoint sans recyclage inter-système qui est équivalente à celle du GMRES-DR(140,42). Pour le recyclage inter-système, nous avons réalisé une série de tests basée sur un critère simple qui consiste à recycler à différents moments du couplage fluide-structure. Plus précisément,

Stratégies étape du recyclage	Stratégie A	Stratégie B	Stratégie C
5	11750	<b>10500</b>	12500
6	11000	<b>9750</b>	10600
7	10000	9500	<b>9000</b>
8	9500	9000	<b>8250</b>
9	8750	<b>8500</b>	<b>8500</b>

TABLE 4.1 : Performances du solveur partitionné en fonction des stratégies de déflation. Les résultats sont rapportés en nombre de produits par **A**.

nous avons considéré le recyclage de l'espace invariant approché à partir du deuxième couplage fluide-structure jusqu'au sixième couplage fluide-structure. Nous observons que l'effet du recyclage améliore la convergence du solveur partitionné avec un gain maximal de 40 % en coût de calcul. Nous observons également l'élimination de l'effet plateau qui apparaît à chaque couplage fluide-structure.

L'effet du recyclage dans le cas flexible semble être moins concluant selon les résultats numériques. Nous avons observé lors des premiers tests que la taille de l'espace de déflation  $k$  avait un réel impact sur le comportement du FGCRO-DR. Typiquement, en respectant la règle des 30 % pour  $k$ , on observait une dégradation nette de la convergence lors du recyclage au cours des premiers couplages fluide-structure. Carvalho et al. ont démontré d'ailleurs dans leur rapport technique [56] qu'il était raisonnable d'augmenter le nombre de vecteurs propres déflatés. Dans leur cas, ils ont réalisé une étude comparative sur les différentes stratégies de déflation en faisant varier le paramètre  $k$  de 1 à 19 pour un espace  $m = 20$ . Il s'avère donc qu'il y a un réel intérêt à augmenter le nombre de vecteurs propres déflatés dans le cas flexible. Pour confirmer cette observation, nous avons comparé les trois stratégies A, B et C en prenant précisément  $m = 70$ ,  $m_i = 10$  et  $k = 35$  soit 50% de l'espace de Krylov externe  $m$ . Les Figures 4.17, 4.18 et 4.19 illustrent les résultats de convergence du FGMRES-DR(70,10,35) en fonction des trois stratégies de déflation. Ces résultats ont démontré une amélioration nette de la convergence avec un gain moyen de 2000 produits par **A** par rapport à une déflation de 30 %.

La Table 4.1 décrit les performances du solveur partitionné en fonction de la stratégie de déflation. On remarque que pour la stratégie C, le solveur partitionné réalise le meilleur gain en termes de produit par **A** lorsqu'on recycle à partir du 7<sup>ème</sup> couplage fluide-structure, soit lorsque le second membre a convergé. On se retrouve alors dans le cas d'une résolution adjointe purement fluide. La stratégie B donne les meilleurs résultats lors du 5<sup>ème</sup> et du 6<sup>ème</sup> couplage tandis que la stratégie A paraît la moins robuste.

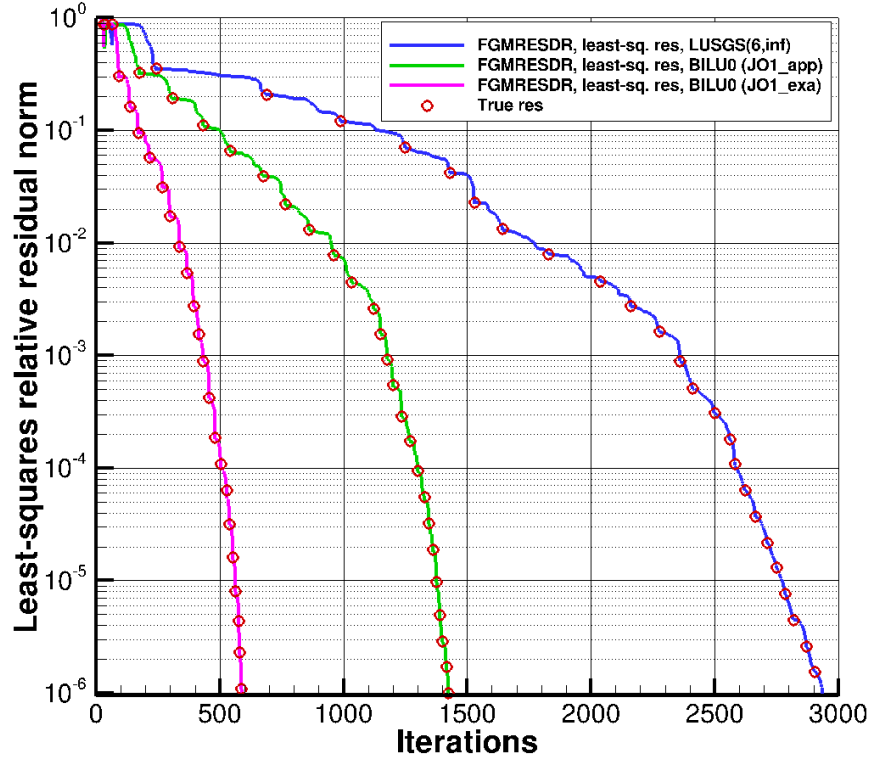


FIGURE 4.15 : Historique de convergence de la norme du résidu relatif. Comparaison des performances des préconditionneurs en partitionné. Nous rappelons les paramètres numériques :  $m = 60$ ,  $m_i = 20$  et  $k = 20$ .

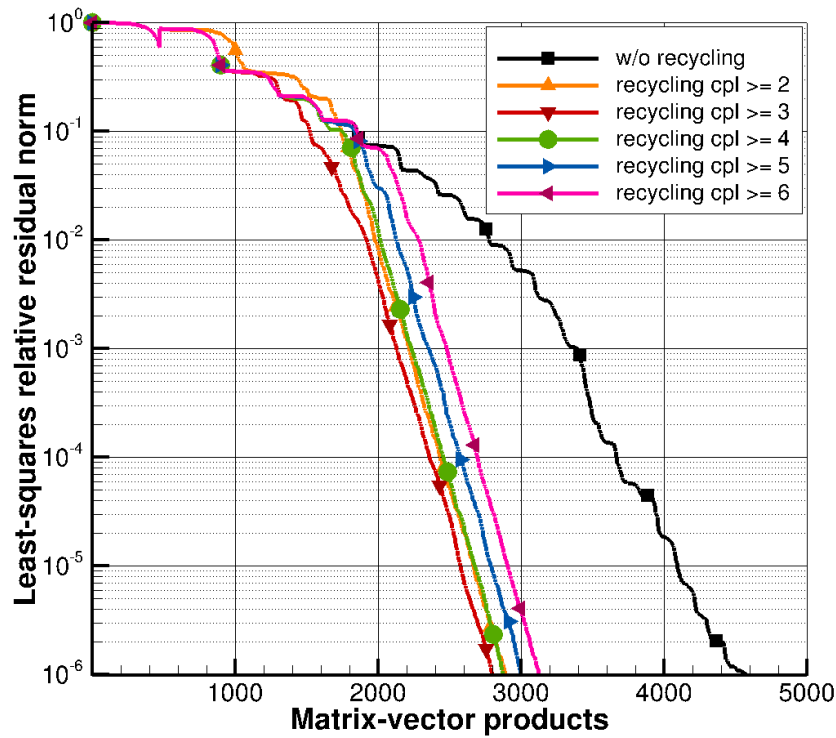


FIGURE 4.16 : Effet du recyclage inter-système du sous-espace invariant approché de l'opérateur  $\mathbf{A}$  sur la résolution partitionnée GCRO-DR(140,42).

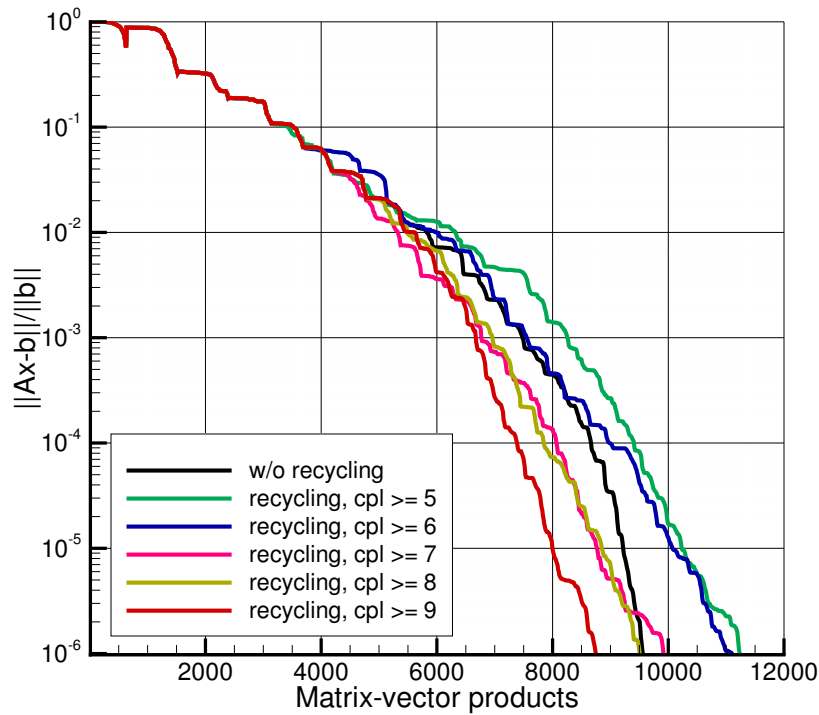


FIGURE 4.17 : Effet du recyclage inter-système du sous-espace invariant approché de l'opérateur  $\mathbf{A}$  sur la résolution partitionnée FGCRO-DR(70,10,35). Les vecteurs de Ritz sont calculés selon la stratégie A exploitant l'espace range( $\mathbf{Z}_m$ )

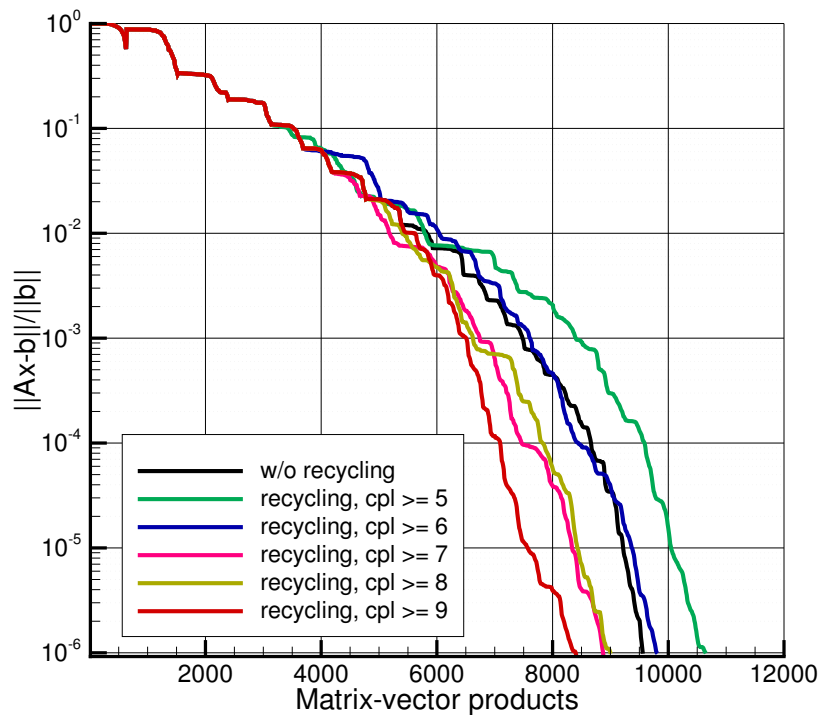


FIGURE 4.18 : Effet du recyclage inter-système du sous-espace invariant approché de l'opérateur  $\mathbf{A}$  sur la résolution partitionnée FGCRO-DR(70,10,35). Les vecteurs de Ritz sont calculés selon la stratégie B exploitant l'espace range( $\mathbf{V}_m$ ).

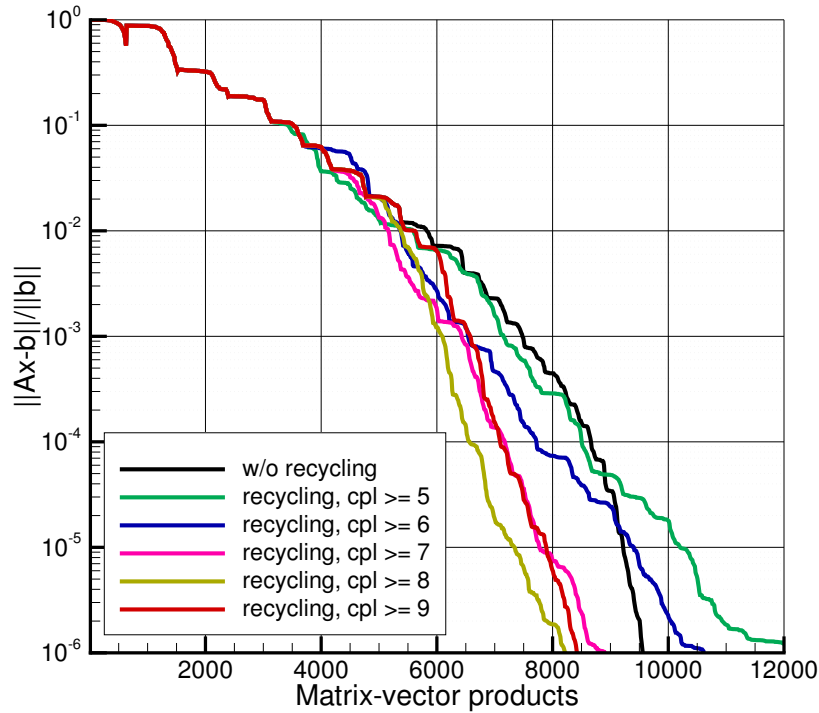


FIGURE 4.19 : Effet du recyclage inter-système du sous-espace invariant approché de l'opérateur  $\mathbf{A}$  sur la résolution partitionnée FGCRO-DR(70,10,35). Les vecteurs de Ritz sont calculés selon la stratégie C exploitant l'espace range( $\mathbf{W}_m$ ).

## Bilan

Nous avons montré dans ce chapitre l'amélioration de la résolution partitionnée pour les systèmes adjoints fortement couplés. Nous avons tout d'abord amélioré la robustesse du solveur partitionné en intégrant les préconditionneurs développés dans le Chapitre 3, c'est-à-dire le BILU appliqué à la matrice jacobienne exacte d'ordre 1. À la base de ce préconditionneur, nous avons mis en oeuvre une méthode de recyclage d'espace invariant approché de l'opérateur du système qui a permis non seulement d'éliminer les effets plateaux mais aussi les sauts apparents à chaque redémarrage lors de la résolution partitionnée. Nous avons notamment conclu que le recyclage permet au GCRO-DR d'obtenir une meilleure convergence même si le recyclage s'effectue au début des cycles fluide-structure caractérisés par une variation importante du second membre. Dans le cas flexible, le comportement du solveur partitionné était assez sensible au recyclage et nous avons conclu que plusieurs facteurs entraient en jeu. Notamment, nous avons remarqué que le nombre de vecteurs propres déflatés avait un impact non négligeable dans le processus de recyclage. Nous avons également conclu que le choix de la stratégie de déflation avait un impact non négligeable sur la convergence du solveur partitionné.

# Chapitre 5

## Mise en place d'un solveur monolithique couplé à des stratégies de préconditionnement robustes

### Sommaire

---

<b>5.1 Stratégies de préconditionnement dans le cadre d'un solveur monolithique</b>	<b>112</b>
5.1.1 Factorisation par bloc et complément de Schur	112
5.1.2 Stratégie de mise à l'échelle	115
<b>5.2 Étude des différentes approximations du complément de Schur</b>	<b>117</b>
5.2.1 Premier cas de figure :	117
5.2.2 Deuxième cas de figure :	117
5.2.3 Troisième cas de figure	117
<b>5.3 Résultats numériques</b>	<b>118</b>

---

### Introduction

Ce chapitre est dédié à l'étude de l'approche monolithique dans le cadre de la résolution des systèmes adjoints couplés aéro-structures. Nous allons, dans le même esprit que les chapitres précédents, comparer les performances de l'approche monolithique aux performances de l'approche partitionnée couplée à la stratégie de recyclage. L'étude comparative se concentre essentiellement sur l'analyse des performances des préconditionneurs du solveur monolithique à savoir le préconditionneur diagonal par bloc et l'approximation du complément Schur.



## 5.1 Stratégies de préconditionnement dans le cadre d'un solveur monolithique

Pour des raisons pratiques, nous allons reformuler le système adjoint couplé (2.31) en introduisant la matrice de flexibilité notée  $\mathbf{S}$  de sorte à ce qu'on obtienne :

$$\begin{bmatrix} \begin{bmatrix} \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \end{bmatrix}^T & -[\mathbf{C}]^T \mathbf{S}^T \\ [\mathbf{A}]^T \begin{bmatrix} \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \end{bmatrix}^T & \mathbf{I} - [\mathbf{D}]^T \mathbf{S}^T \end{bmatrix} \begin{bmatrix} \lambda_a \\ \lambda_s \end{bmatrix} = \begin{bmatrix} -\begin{bmatrix} \frac{\partial J}{\partial \mathbf{W}} \end{bmatrix}^T \\ -[\mathbf{A}]^T \begin{bmatrix} \frac{\partial J}{\partial \mathbf{X}_a} \end{bmatrix}^T \end{bmatrix} \quad (5.1)$$

La matrice de flexibilité  $\mathbf{S}$  est pratique au sens où elle représente une version "condensée" de l'inverse de la matrice de rigidité (pour plus de détails sur la construction de la matrice  $\mathbf{S}$ , voir [section 2.4.2, 20]). La résolution monolithique soulève des difficultés numériques différentes de celles de l'approche partitionnée. Tout d'abord, le conditionnement numérique du système couplé est un paramètre à prendre en compte systématiquement au sens où une mise à l'échelle est essentielle. En effet, le système est par construction constitué d'un bloc fluide et d'un bloc structure avec des discrétisations différentes. Cette disparité au niveau des discrétisations est le résultat d'un mauvais conditionnement de la matrice. La deuxième difficulté est de développer des stratégies de préconditionnement robustes pour le système couplé. Parmi ces stratégies, on cite le préconditionneur basé sur l'approximation du complément de Schur ou celui basé sur la méthode de Gauss-Seidel par bloc.

On rappelle que la construction d'un préconditionneur s'effectue à partir d'une approximation de l'opérateur du système. Le préconditionneur idéal serait évidemment l'opérateur inverse du système, mais comme souligné précédemment, il est pratiquement impossible d'avoir accès à l'inverse du système complet. Il est important de souligner que la plupart des préconditionneurs qui existent dans les problèmes d'algèbre linéaire numérique sont construits à partir de la factorisation par bloc de la matrice du système. L'étude de cette factorisation est cruciale puisqu'elle va nous permettre d'étudier les différentes possibilités qui s'offrent à nous et de décider *in fine* quel préconditionneur est le plus adapté à notre problème.

### 5.1.1 Factorisation par bloc et complément de Schur

Supposons que notre système adjoint couplé (5.1) s'écrive de la façon suivante :

$$\mathcal{A} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_1 \\ \mathbf{B}_2 & \mathbf{C} \end{bmatrix} \quad (5.2)$$

où  $\mathbf{A}$ ,  $\mathbf{B}_1$ ,  $\mathbf{B}_2$  et  $\mathbf{C}$  sont les blocs qui correspondent aux blocs de la matrice du système (5.1). La factorisation **LDU** par bloc de  $\mathcal{A}$  s'écrit alors :

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}_1 \\ \mathbf{B}_2 & \mathbf{C} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{O} \\ \mathbf{B}_2\mathbf{A}^{-1} & \mathbf{I} \end{bmatrix}}_{\mathbf{L}} \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{O} & \mathcal{S} \end{bmatrix}}_{\mathbf{D}} \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{A}^{-1}\mathbf{B}_1 \\ \mathbf{O} & \mathbf{I} \end{bmatrix}}_{\mathbf{U}} \quad (5.3)$$

Il est tout aussi intéressant de considérer la factorisation **LU** par bloc de  $\mathcal{A}$  qui s'exprime de deux façons :

$$\mathcal{A} = \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{O} \\ \mathbf{B}_2\mathbf{A}^{-1} & \mathbf{I} \end{bmatrix}}_{\mathbf{L}} \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{B}_1 \\ \mathbf{O} & \mathcal{S} \end{bmatrix}}_{\mathbf{U}} \quad (5.4)$$

ou

$$\mathcal{A} = \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{B}_2 & \mathcal{S} \end{bmatrix}}_{\mathbf{L}} \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{A}^{-1}\mathbf{B}_1 \\ \mathbf{O} & \mathbf{I} \end{bmatrix}}_{\mathbf{U}} \quad (5.5)$$

$\mathcal{S} = \mathbf{C} - \mathbf{B}_2\mathbf{A}^{-1}\mathbf{B}_1$  représente le complément de Schur de  $\mathbf{A}$  dans le système adjoint  $\mathcal{A}$ . Rappelons que le rôle du préconditionneur est d'améliorer les propriétés spectrales de  $\mathcal{A}$  de sorte à réduire le nombre d'itérations du solveur de Krylov. Benzi et al. [60] soulignent que le système  $\mathcal{M} = \mathbf{D}^{-1}\mathcal{A}$  possède trois valeurs propres distinctes, ce qui signifie qu'un solveur de Krylov de type GMRES appliqué au système préconditionné  $\mathcal{M}$  convergerait au plus en trois itérations. Il est démontré dans [61, 62] que la matrice  $\mathcal{M}$  satisfait la propriété suivante :

$$(\mathcal{M} - \mathbf{I})(\mathcal{M} + \frac{1}{2}(1 + \sqrt{5})\mathbf{I})(\mathcal{M} - \frac{1}{2}(1 + \sqrt{5})\mathbf{I}) = \mathbf{O}. \quad (5.6)$$

Il est clair que  $\mathcal{M}$  est diagonalisable et admet trois valeurs propres distinctes qui sont  $1$ ,  $\frac{1}{2}(1 + \sqrt{5})$  et  $\frac{1}{2}(1 - \sqrt{5})$ . Cette propriété spectrale de  $\mathcal{M}$  est intéressante pour le solveur GMRES qui converge théoriquement en trois itérations.

Dans la pratique, il n'est pas aisé de construire l'opérateur  $\mathbf{D}$ . En effet, si on remplace les opérateurs  $\mathbf{A}$ ,  $\mathbf{B}_1$ ,  $\mathbf{B}_2$  et  $\mathbf{C}$  dans le cas de notre système adjoint couplé (2.31), l'opérateur  $\mathbf{D}$  s'écrit :

$$\mathbf{D} = \begin{bmatrix} \left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right]^T & \mathbf{O} \\ \mathbf{O} & \mathbf{S} \end{bmatrix} \quad (5.7)$$

$$\text{où } \mathbf{S} = \left( \mathbf{I} - [\mathbf{D}]^T \mathbf{S}^T \right) - [\mathbf{A}]^T \left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \right]^T \left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right]^{-T} [\mathbf{C}]^T \mathbf{S}^T.$$

Le calcul du complément de Schur nécessite donc le produit par l'opérateur  $[\mathbf{C}]^T \mathbf{S}^T$  puis l'inversion de l'opérateur fluide  $\left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right]^{-T}$  puis le produit des deux opérateurs  $\left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \right]^T$  et  $[\mathbf{A}]^T$ . La résolution de ce dernier rend le coût de calcul trop important pour employer directement  $\mathbf{S}$  dans une phase de préconditionnement surtout pour un grand nombre d'itérations. Une manière de s'affranchir de ce problème est de remplacer (5.7) par une approximation du type :

$$\mathbf{D} = \begin{bmatrix} \left[ \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \right]^T & \mathbf{O} \\ \mathbf{O} & \hat{\mathbf{S}} \end{bmatrix} \quad (5.8)$$

où  $\left[ \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \right]^T$  et  $\hat{\mathbf{S}}$  sont des approximations de  $\left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right]^T$  et  $\mathbf{S}$  respectivement. Différentes approximations ont été considérées dans la littérature notamment pour le complément de Schur [60, 63, 64]. Dans notre cas, l'approximation de l'opérateur fluide repose essentiellement sur la discrétisation exacte d'ordre un des flux convectifs  $\mathbf{J}_{O_1}^{EX}$ .

Comme expliqué dans le Chapitre 1, Kenway et al. [7] ont mis en place un solveur GMRES monolithique pour améliorer la résolution du système adjoint couplé. Afin de respecter le compromis robustesse/coût d'inversion, les auteurs proposent comme préconditionneur la partie diagonale par blocs de l'opérateur du système (5.1) en négligeant les termes extra-diagonaux. L'avantage de ce type de préconditionneur est qu'il s'implémente facilement en ce sens que chaque bloc physique est traité de manière indépendante en réutilisant le solveur dédié. Pour la partie aérodynamique, les auteurs réutilisent le solveur GMRES pour un espace de Krylov de taille comprise entre 10 et 20. La partie structure est résolue de manière exacte au moyen d'une factorisation **LU**. La structure matricielle adoptée par Kenway et al. est donnée ci-dessous :

$$\mathcal{M} = \begin{bmatrix} \left[ \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \right]^T & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} \quad (5.9)$$

Soulignons également que les auteurs ont négligé le deuxième terme de la partie structure à savoir l'opérateur  $[\mathbf{D}]$  auquel cas la partie structure se restreint à la matrice identité.

Une raison éventuelle pour laquelle le terme  $[\mathbf{D}]$  a été négligé est son coût de calcul. En effet, ce terme est le produit de plusieurs termes faisant intervenir les opérateurs  $\mathbf{T}_{surf}^U$ ,  $\mathbf{T}_{surf}^Q$  et  $\mathbf{T}_{vol}$ . Le coût prépondérant vient de l'opérateur volumique de déformation de maillage  $\mathbf{T}_{vol}$ , ce qui peut devenir pénalisant sachant que ce terme est calculé à chaque itération du GMRES. A titre de rappel, Kenway et al. ont réalisé une étude comparative entre le solveur monolithique et le solveur partitionné et ont conclu que l'approche monolithique réalise un gain en coût CPU de 19 % dans le cas d'une charge de 1g et de 29% dans le cas d'une charge de 2.5g. Cette expérience nous donne une idée des performances que peut réaliser un solveur monolithique surtout pour un préconditionneur dans lequel plusieurs termes ont été négligés.

Dans le même esprit, Zhang et Zingg [4] ont développé un solveur GCROT monolithique en proposant le préconditionneur triangulaire par bloc (Gauss-Seidel) en plus du préconditionneur diagonal par bloc (Jacobi). Contrairement à Kenway et al., Zhang et Zingg utilisent une formulation à 3 champs en ajoutant dans le système le bloc maillage. Voici ci-dessous la structure matricielle du préconditionneur triangulaire inférieur par bloc :

$$\mathcal{M} = \begin{bmatrix} \begin{bmatrix} \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \end{bmatrix}^T & \mathbf{O} & \mathbf{O} \\ \begin{bmatrix} \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \end{bmatrix}^T & \begin{bmatrix} \frac{\partial \mathbf{R}_M}{\partial \mathbf{X}_a} \end{bmatrix}^T & \mathbf{O} \\ \mathbf{O} & \begin{bmatrix} \frac{\partial \mathbf{R}_M}{\partial \mathbf{U}} \end{bmatrix}^T & \mathbf{M}_s^T \end{bmatrix} \quad (5.10)$$

$\mathbf{M}_s$  correspond à la factorisation exacte LU de la matrice de rigidité  $\mathbf{K}$  tandis que  $\mathbf{R}_M$  correspond au résidu maillage.

Pour mettre en évidence cette amélioration, les auteurs comparent les performances des deux préconditionneurs lors d'un calcul stationnaire. La Figure 5.1 affiche le coût CPU de la phase de préconditionnement impliquée dans le processus de Krylov dans le cas du préconditionneur de Jacobi et de Gauss-Seidel. Nous remarquons effectivement que ce dernier est meilleur en coût CPU et a permis de diminuer le nombre total d'itérations pour une plage de tolérance qui varie entre 0.01 et 0.1. Maintenir un nombre d'itérations assez faible permet d'améliorer la robustesse de l'algorithme monolithique même pour des problèmes de taille et de complexité plus grande.

### 5.1.2 Stratégie de mise à l'échelle

La stratégie adoptée dans le cadre de cette thèse consiste à appliquer à droite et à gauche du système couplé (5.1) l'inverse de la diagonale de l'opérateur fluide. Plus précisément, si on note  $\mathbf{D}$  la diagonale de l'opérateur fluide, le système couplé mis à l'échelle s'écrit :

## 5. MISE EN PLACE D'UN SOLVEUR MONOLITHIQUE COUPLÉ À DES STRATÉGIES DE PRÉCONDITIONNEMENT ROBUSTES

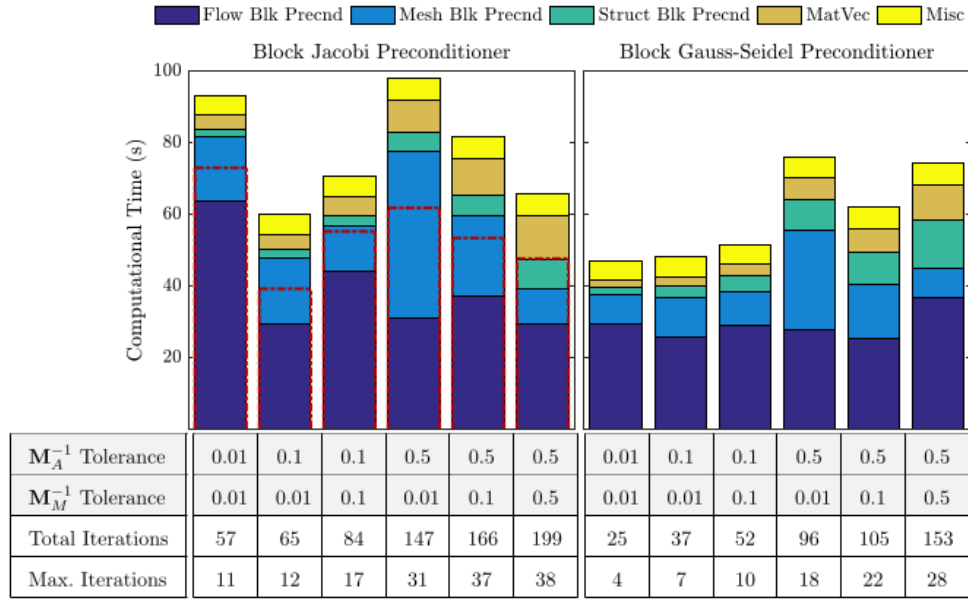


FIGURE 5.1 : Répartition du temps de calcul pour les préconditionneurs bloc Jacobi et bloc Gauss–Seidel ( $M_A^{-1}$ ) pour différentes tolérances. Les deux dernières lignes décrivent le nombre total et le nombre maximum d’itérations FGMRES lors du processus de Newton–Krylov.

$$D_L^{-1} A D_U^{-1} y = D_L^{-1} b \quad (5.11)$$

$$x = D_U^{-1} y \quad (5.12)$$

où  $D_L$  et  $D_U$  correspondent respectivement à la partie  $L$  et  $U$  de la factorisation  $LU$  par bloc auquel cas nous avons  $D_L D_U = D$ . De cette manière, une renormalisation des lignes et des colonnes du système couplé est réalisée. La figure illustre la factorisation  $LU$  par bloc de la diagonale de l’opérateur fluide.

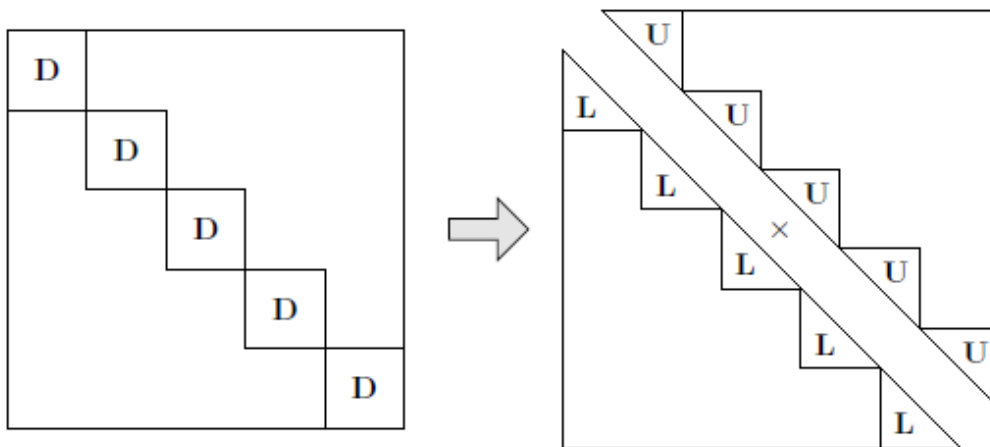


FIGURE 5.2 : Factorisation  $LU$  de la diagonale par bloc.

## 5.2 Étude des différentes approximations du complément de Schur

Comme évoqué précédemment, la robustesse du préconditionneur va fortement dépendre du type d'approximation apportée au complément de Schur et à l'opérateur fluide. Cette section est dédiée à l'analyse des différentes approximations du complément de Schur dans le cas d'un préconditionneur diagonal par bloc représenté par (5.13). Une étude comparative entre le solveur monolithique et le solveur partitionné est également prise en compte.

### 5.2.1 Premier cas de figure :

Ce cas de figure consiste à négliger l'opérateur  $[\mathbf{D}]\mathbf{S}^T$  afin de reproduire le préconditionneur diagonal par bloc  $\hat{\mathbf{D}}$  qui utilise la forme la plus simple de  $\mathbf{S}$ . Le préconditionneur s'écrit :

$$\mathcal{M} = \begin{bmatrix} \left[ \begin{array}{c} \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \end{array} \right]^T & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} \quad (5.13)$$

Bien qu'on ait négligé plusieurs termes, l'application de la mise à l'échelle devrait améliorer le conditionnement de la matrice d'une part mais aussi la précision de la solution. L'application des matrices  $\mathbf{D}_L^{-1}$  et  $\mathbf{D}_U^{-1}$  permet effectivement d'obtenir une matrice dont la diagonale est l'identité pour l'opérateur fluide.

### 5.2.2 Deuxième cas de figure :

Dans ce cas de figure, nous considérons la contribution du terme  $[\mathbf{D}]^T \mathbf{S}^T$  dans le but d'améliorer la robustesse mais au prix d'un coût de calcul plus important.

$$\mathcal{M} = \begin{bmatrix} \left[ \begin{array}{c} \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \end{array} \right]^T & \mathbf{O} \\ \mathbf{O} & \mathbf{I} - [\mathbf{D}]^T \mathbf{S}^T \end{bmatrix} \quad (5.14)$$

### 5.2.3 Troisième cas de figure

Pour finir, nous considérons l'approximation de Schur  $\hat{\mathbf{S}}$  dans le but d'améliorer la robustesse mais au prix d'un coût de calcul plus important.

$$\mathcal{M} = \begin{bmatrix} \left[ \begin{array}{c} \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \end{array} \right]^T & \mathbf{O} \\ \mathbf{O} & \hat{\mathbf{S}} \end{bmatrix} \quad (5.15)$$

L'approximation du complément de Schur est définie par :

$$\hat{\mathbf{S}} = \left( \mathbf{I} - [\mathbf{D}]^T \mathbf{S}^T \right) - [\mathbf{A}]^T \left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \right]^T \left[ \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \right]^{-T} [\mathbf{C}]^T \mathbf{S}^T \quad (5.16)$$

où  $\left[ \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \right]$  représente la matrice jacobienne exacte d'ordre un  $\mathbf{J}_{O1}^{EX}$ .

Soulignons que l'approximation de Schur  $\hat{\mathbf{S}}$  intervient dans la phase de préconditionnement. Nous cherchons donc à résoudre approximativement l'équation :

$$\hat{\mathbf{S}} \mathbf{x} = \mathbf{v} \quad (5.17)$$

où  $\mathbf{v}$  représente un vecteur de Krylov. Comme  $\hat{\mathbf{S}}$  n'est pas connu explicitement, nous procédons à une résolution itérative de (5.17). Ainsi, en fixant  $\mathbf{x}_0 = 0$ , on résout :

$$\left( \mathbf{I} - [\mathbf{D}]^T \mathbf{S}^T \right) \mathbf{x}_{k+1} = [\mathbf{A}]^T \left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \right]^T \left[ \frac{\partial \hat{\mathbf{R}}_a}{\partial \mathbf{W}} \right]^{-T} [\mathbf{C}]^T \mathbf{S}^T \mathbf{x}_k + \mathbf{v} \quad (5.18)$$

En effectuant  $k = 3$  ou  $k = 4$  itérations qui est largement suffisant dans ce cas.

### 5.3 Résultats numériques

La Figure 5.3 présente la comparaison de l'approche partitionnée et celle du monolithique. Pour le cas du solveur partitionné, nous avons sélectionné celui qui réalisait le meilleur gain. En l'occurrence, nous avons choisi la courbe pour laquelle nous avons activé le recyclage dès le deuxième couplage fluide-structure (courbe orange). Le solveur GMRES-DR monolithique est quant à lui préconditionné par une matrice diagonale par bloc. La courbe verte représente le cas où le préconditionneur est réduit à (5.13). La courbe bleue représente le cas où le préconditionneur est donné par (5.14). On constate que les courbes verte et bleue se superposent et que finalement le terme  $[\mathbf{D}]^T \mathbf{S}^T$  n'améliore pas la convergence. Une observation intéressante vient de la mise à l'échelle représentée par la courbe rose. On constate en effet une stagnation au début des itérations puis la courbe retrouve la même pente que dans le cas sans mise à l'échelle. En comparant les courbes bleue (sans mise à l'échelle) et rose (mise à l'échelle), cette dernière réalise plus d'itérations. Cette observation numérique ressemble fortement au résultat obtenu par Zhang et Zingg dans [4] (voir la Figure 2.7). Même si le nombre d'itérations est plus élevé, les résultats numériques ont montré dans notre cas que la précision de la solution est meilleure dans le cas d'une mise à l'échelle, ce qui est plus pertinent que le nombre d'itérations. Les tests numériques ont également démontré que l'approximation de Schur (5.15) n'a pas apporté de gain en nombre d'itérations par rapport à (5.13) et (5.14).

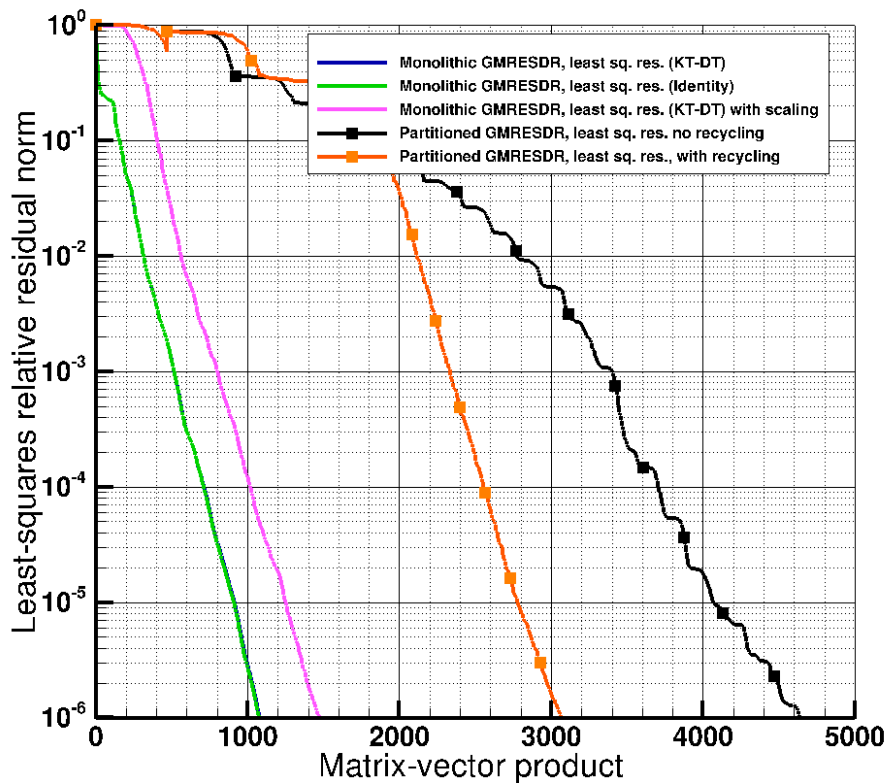


FIGURE 5.3 : Courbes de convergence du solveur GMRES-DR(120,40) monolithique et partitionné. Cas monolithique : Application d'un préconditionneur diagonal par bloc dont le deuxième bloc diagonal est représenté par l'opérateur  $\mathbf{I} - [\mathbf{D}]^T \mathbf{S}^T$ . Application d'une mise à l'échelle pour le cas monolithique (courbe rose). Cas partitionné : Application de la stratégie de recyclage dès le deuxième couplage fluide-structure (courbe orange).

## Bilan :

Nous avons démontré dans ce chapitre les performances de l'approche monolithique par rapport à l'approche partitionnée. Plus précisément, la forme la plus simple du préconditionneur diagonal par bloc a permis au solveur GMRES-DR monolithique de réaliser un gain de 66 % en coût CPU par rapport au solveur GMRES-DR partitionné couplé à la stratégie de recyclage. Également, la mise à l'échelle a permis au solveur monolithique de gagner en précision dans le calcul de la solution. Nous avons constaté que l'approximation de Schur n'as pas vraiment apporté d'amélioration au niveau de la convergence et que d'autres pistes seraient à exploiter pour pouvoir démontrer la robustesse de ce type de préconditionneur pour l'approche monolithique.





# Conclusion

Le travail au cours de ces trois années de thèse a porté essentiellement sur l'étude des solveurs de sous-espaces Krylov pour améliorer la résolution des systèmes adjoints couplés aéro-structure. Parmi ces solveurs, nous avons étudié la classe des méthodes de Krylov emboîtées et plus spécifiquement la méthode GMRES emboîtée connue pour ses propriétés de convergence robustes. La résolution de tels systèmes a été réalisée au moyen de deux approches antagonistes qui sont l'approche partitionnée et l'approche monolithique.

Tout d'abord, l'état de l'art réalisé au Chapitre 1 a montré l'intérêt des méthodes de Krylov à résoudre de manière robuste les systèmes adjoints couplés pour de forts interactions fluide-structure de sorte à obtenir des solutions adjointes plus précises. En outre, ce chapitre a montré que l'approche monolithique réalisait des performances meilleures que l'approche partitionnée surtout pour de forts couplages fluide-structure.

Dans le cadre de cette thèse, la première étape a consisté à mettre en place un solveur GMRES emboîté couplé à des stratégies de préconditionnement robustes dans le but d'accélérer la résolution des systèmes adjoints purement fluides dont le Chapitre 2 fait l'objet. Cette stratégie de préconditionnement a consisté à développer une méthode de factorisation LU incomplète par bloc qui a été appliquée à deux types de préconditionneur. Le premier repose sur la matrice jacobienne d'ordre un approchée au sens où une approximation est effectuée sur les flux visqueux. La deuxième repose sur la matrice jacobienne exacte d'ordre un. Nous avons également introduit la déflation spectrale par augmentation au sein du solveur GMRES, noté GMRES-DR, afin d'obtenir un gain supplémentaire lors de la convergence du système adjoint. Les résultats numériques ont montré de manière claire que la stratégie des méthodes emboîtées surpassait la stratégie des méthodes standards en termes de robustesse. Les courbes ont effectivement montré que le solveur GMRES emboîté réalisait des gains significatifs par rapport au solveur GMRES standard, qui convergait très lentement ou stagnait. En plus de l'intérêt des méthodes emboîtées, les expériences numériques ont démontré que la matrice jacobienne exacte d'ordre un réalisait également des gains satisfaisants par rapport à la version approchée.

L'objectif du Chapitre 3 a été d'améliorer la résolution du système adjoint couplé au

moyen du solveur partitionné. L'idée a été d'intégrer d'abord les contributions du Chapitre 2 dans le solveur partitionné. Ensuite, l'idée a été de mettre en place une stratégie de recyclage spectrale au sein du solveur partitionné qui a la particularité de recycler l'information spectrale dans le cadre d'une séquence de systèmes linéaires. Pour cela, nous avons développé une méthode de recyclage, le solveur GCRO-DR, qui permet justement de recycler d'un système à l'autre. Les résultats numériques ont démontré que le recyclage spectral apportait un gain significatif en nombre d'itérations mais aussi en coût CPU et permettait d'éliminer l'effet plateau et les sauts observés à chaque résolution d'un nouveau système linéaire. Il convient de souligner que l'effet de cette stratégie est différent selon qu'on considère le solveur GMRES emboîté ou le solveur GMRES standard. Même si une amélioration a été observée dans les deux cas, le cas flexible demeure sensible aux effets du recyclage. Plus précisément, le solveur partitionné a l'inconvénient de dépendre de plusieurs paramètres et il est nécessaire de mettre en place des critères de sélection des vecteurs propres afin de pallier ce problème.

Dans le dernier chapitre, nous avons développé un solveur GMRES-DR monolithique dans le but d'améliorer de manière drastique la résolution des systèmes adjoints fortement couplés. La stratégie de préconditionnement développée dans ce chapitre repose essentiellement sur la partie diagonale par bloc issue de la factorisation à trois facteurs de l'opérateur du système adjoint. Pour des raisons de performance, nous avons considéré différentes approximations du préconditionneur diagonal par bloc. Les résultats numériques ont démontré les performances du solveur monolithique et l'intérêt à adopter un tel solveur pour la résolution de problèmes adjoints fortement couplés. De plus, la stratégie de mise à l'échelle s'est révélée plus pertinente dans le cadre du solveur monolithique au sens où le calcul de la solution était plus précis. Néanmoins, les stratégies de préconditionnement basées sur l'approximation du complément de Schur n'ont pas amélioré la convergence.

# Perspectives

Nous avons démontré tout au long de ce travail de thèse les différentes démarches adoptées afin d'améliorer la résolution des systèmes linéaires adjoints pour des cas numériques raides. Nous avons démontré l'intérêt des stratégies de recyclage dans le cas de l'approche partitionnée d'une part et l'intérêt de l'approche monolithique couplée à des stratégies de préconditionnement robustes d'autre part. Néanmoins, nous proposons quelques pistes intéressantes afin d'améliorer la robustesse de ces approches.

Dans le cadre de l'approche partitionnée, il est intéressant d'utiliser les stratégies de recyclage d'espace invariant approché dans une résolution non-linéaire stationnaire qui peut être vue comme une séquence de systèmes linéaires qui varient à chaque itération fluide.

Également, une extension naturelle de ce travail de thèse est d'étudier une version multi-second membres (block GCRO-DR) dans le cadre des systèmes adjoints aéro-structure. A. Bissuel a étudié la version bloc du solveur GMRES-DR avec plusieurs seconds membres dans le cadre de la résolution des systèmes adjoints aérodynamiques [27]. En ce qui concerne les stratégies de préconditionnement, il est intéressant de considérer des stratégies hybrides qui combinent des solveurs itératifs et des solveurs directs. Un exemple typique est de combiner des méthodes de type multigrille avec des méthodes de factorisation LU incomplète par bloc. Aussi, il est possible d'améliorer la robustesse de la factorisation LU incomplète par bloc en appliquant la méthode d'*overlapping*. Elle consiste à inclure un certain degré de chevauchement entre les blocs pour capturer les dépendances entre eux. De cette façon, certains des couplages entre les équations sont pris en compte dans la factorisation.

Dans le cadre de l'approche monolithique, il est crucial en plus des bonnes stratégies de mise à l'échelle et de préconditionnement, de disposer d'un outil de déformation de maillage en mode adjoint [4] très performant pour rendre le solveur couplé compétitif vis-à-vis du solveur partitionné.



## Annexe A

# Reformulation du problème aux valeurs propres

### A.1 Reformulation des problèmes aux valeurs propres

Nous allons démontrer de façon simple comment il est possible de transformer le problème aux valeurs propres généralisé en un problème aux valeurs propres standard. En reprenant les notations de la section 3.3.1.1, on a

$$\mathbf{H}_m^T \mathbf{u} = \theta \bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m \mathbf{u} \quad (\text{A.1})$$

$$\Leftrightarrow \mathbf{H}_m^{-T} \bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m \mathbf{u} = \theta^{-1} \mathbf{u} \quad (\text{A.2})$$

On remarque que  $\mathbf{H}_m^{-T} \bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m = (\bar{\mathbf{H}}_m \mathbf{H}_m^{-1})^T \bar{\mathbf{H}}_m$ . Cette relation nous permet d'écrire :

$$\bar{\mathbf{H}}_m \mathbf{H}_m^{-1} = \begin{pmatrix} \mathbf{H}_m & \\ 0 \dots 0 & h_{m+1,m} \end{pmatrix} \mathbf{H}_m^{-1} = \begin{pmatrix} \mathbf{I}_m & \\ h_{m+1,m} \mathbf{e}_m^T \mathbf{H}_m^{-1} \end{pmatrix} \quad (\text{A.3})$$

On en déduit

$$\begin{aligned} (\bar{\mathbf{H}}_m \mathbf{H}_m^{-1})^T \bar{\mathbf{H}}_m &= \begin{pmatrix} \mathbf{I}_m & h_{m+1,m} \mathbf{H}_m^{-T} \mathbf{e}_m \end{pmatrix} \begin{pmatrix} \mathbf{H}_m \\ h_{m+1,m} \mathbf{e}_m^T \end{pmatrix} \\ &= \mathbf{H}_m + h_{m+1,m}^2 \mathbf{H}_m^{-T} \mathbf{e}_m \mathbf{e}_m^T \end{aligned} \quad (\text{A.4})$$

Si on note  $\mathbf{f} = \mathbf{H}_m^{-T} \mathbf{e}_m$ , la dernière colonne de  $\mathbf{H}_m^{-T}$ , on obtient l'expression suivante :

$$\mathbf{H}_m^{-T} \bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m = \mathbf{H}_m + h_{m+1,m}^2 \mathbf{f} \mathbf{e}_m^T \quad (\text{A.5})$$

## A.2 Simplification du problème aux valeurs propres généralisé du FGCRO-DR

Le but est de reformuler le problème aux valeurs propres généralisé (4.77) en un problème aux valeurs propres plus simple. L'idée est de s'inspirer de la formulation de la section précédente en simplifiant les opérateurs. Si on note  $\mathbf{W}_m = [\mathbf{W}_{k+1}, \mathbf{v}_{k+2}, \dots, \mathbf{v}_m]$ , le produit  $\mathbf{V}_{m+1}^T \mathbf{W}_m$  peut être décomposé de la façon suivante :

$$\mathbf{V}_{m+1}^T \mathbf{W}_m = \begin{bmatrix} \mathbf{V}_{k+1} \mathbf{W}_{k+1} & \mathbf{0}_{(k+1) \times (m-k-1)} \\ \mathbf{0}_{(m-k-1) \times (k+1)} & \mathbf{I}_{m-k-1} \\ \mathbf{0}_{1 \times (k+1)} & \mathbf{0}_{1 \times (m-k-1)} \end{bmatrix} \quad (\text{A.6})$$

Par conséquent, le produit  $\bar{\mathbf{H}}_m \mathbf{V}_{m+1}^T \mathbf{W}_m$  s'exprime :

$$\bar{\mathbf{H}}_m \mathbf{V}_{m+1}^T \mathbf{W}_m = \begin{bmatrix} \mathbf{H}_m \\ \mathbf{0} \cdots h_{m+1,m} \end{bmatrix} \begin{bmatrix} \mathbf{V}_{k+1} \mathbf{W}_{k+1} & \mathbf{0}_{(k+1) \times (m-k-1)} \\ \mathbf{0}_{(m-k-1) \times (k+1)} & \mathbf{I}_{m-k-1} \\ \mathbf{0}_{1 \times (k+1)} & \mathbf{0}_{1 \times (m-k-1)} \end{bmatrix} \quad (\text{A.7})$$

Sachant que la dernière ligne de la matrice  $\mathbf{V}_{m+1}^T \mathbf{W}_m$  est composée de zéros, le produit  $\bar{\mathbf{H}}_m \mathbf{V}_{m+1}^T \mathbf{W}_m$  se réduit finalement à la matrice suivante :

$$\bar{\mathbf{H}}_m \mathbf{V}_{m+1}^T \mathbf{W}_m = \mathbf{H}_m \mathbf{V}_{m+1}^T \mathbf{W}_m [1 : m, :] \quad (\text{A.8})$$

où  $\mathbf{V}_{m+1}^T \mathbf{W}_m [1 : m, :]$  correspond à la sous-matrice de taille  $m \times m$  de l'opérateur  $\mathbf{V}_{m+1}^T \mathbf{W}_m$ .

En appliquant à présent les étapes de la section précédente, nous obtenons la problème aux valeurs propres suivant :

$$(\mathbf{H}_m + h_{m+1,m}^2 \mathbf{f} \mathbf{e}_m^T) \mathbf{y} = \mathbf{V}_{m+1}^T \mathbf{W}_m [1 : m, :] \mathbf{y} \quad (\text{A.9})$$

Nous obtenons ainsi un problème aux valeurs propres toujours généralisé mais dont les opérateurs ont été simplifiés. Ceci améliore grandement le conditionnement de l'opérateur et par conséquent la précision du calcul des vecteurs de Ritz harmoniques.





## Annexe B

# Algorithmes

---

**Algorithm 2** Schéma LBGS - Approche adjointe
 

---

- 1: Initialisation :  $\mathbf{U}, \mathbf{W}, \mathbf{X}_a, \mathbf{X}_s, \mathbf{X}_{a0}, \lambda_a^0, \lambda_s^0, \theta_s, tol$
- 2: **if**  $\lambda_s^0 \neq \mathbf{0}$  **then**
- 3:    $\mathbf{RHS}_{stru} \leftarrow \left[ \mathbf{T}_{surf}^Q \frac{\partial \mathbf{Q}_a}{\partial \mathbf{W}} \right]^T \mathbf{S}^T \lambda_s^0$
- 4: **else**
- 5:    $\mathbf{RHS}_{stru} \leftarrow \mathbf{0}$
- 6: **end if**
- 7: **if**  $\lambda_a^0 = \mathbf{0}$  **then**
- 8:    $\left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right]^T \lambda_a^0 = - \left[ \frac{\partial J}{\partial \mathbf{W}} \right]^T + \mathbf{RHS}_{stru}$    ▷ Résolution approchée du problème adjoint fluide
- 9: **end if**
- 10: **for**  $k \leftarrow 1, n_{cpl}$  **do**
- 11:    $\lambda_a^k = (1 - \theta_a) \lambda_a^{k-1} + \theta_a \lambda_a^k$    ▷ Relaxation du vecteur adjoint aérodynamique
- 12:    $A_{Xs, surf} \leftarrow \left[ \frac{\partial \mathbf{Q}_a}{\partial \mathbf{X}_{a, surf}} \right]^T \left[ \mathbf{T}_{surf}^Q \right]^T \mathbf{S}^T \lambda_s^{k-1}$    ▷ Calcul de l' adjoint géométrique structure
- 13:    $A_{Xa} \leftarrow - \left[ \lambda_a^{k-1} \right]^T \left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} + \frac{\partial J}{\partial \mathbf{X}_a} \right]^T$    ▷ Calcul de l' adjoint géométrique aérodynamique
- 14:    $A_{Xa, surf} \leftarrow (\mathbf{T}_{vol})^T \mathbf{A}_{Xa}$    ▷ Adjoint de la déformation de maillage
- 15:    $\lambda_s^k \leftarrow (\mathbf{T}_{surf}^U)^T (A_{Xs, surf} + A_{Xa, surf})$    ▷ Calcul du vecteur adjoint structure
- 16:    $\lambda_s^k \leftarrow (1 - \theta_s) \lambda_s^{k-1} + \theta_s \lambda_s^k$    ▷ Relaxation du vecteur adjoint structure
- 17:    $\mathbf{RHS}_{stru} \leftarrow \left[ \mathbf{T}_{surf}^Q \frac{\partial \mathbf{Q}_a}{\partial \mathbf{W}} \right]^T \mathbf{S}^T \lambda_s^k$    ▷ Mise à jour du second membre structure
- 18:    $\left[ \frac{\partial \mathbf{R}_a}{\partial \mathbf{W}} \right]^T \lambda_a^k = - \left[ \frac{\partial J}{\partial \mathbf{W}} \right]^T + \mathbf{RHS}_{stru}$    ▷ Résolution approchée du problème fluide
- 19: **end for**
- 20:  $\frac{dJ}{dp} \leftarrow \left[ \frac{\partial J}{\partial \mathbf{X}_a} + \lambda_a^T \frac{\partial \mathbf{R}_a}{\partial \mathbf{X}_a} \right] \mathbf{B} \frac{d\mathbf{X}_{a0}}{dp} - \lambda_s^T \mathbf{E} \frac{d\mathbf{X}_{a0}}{dp}$    ▷ Assemblage du gradient

---

**Algorithm 3** Méthode itérative associée à une correction de rang 1

---

```
1: Initialisation :  $\mathbf{x}_0, \mathbf{r}_0, \mathbf{H}_0, k = 0, \epsilon$ 
2:  $\mathbf{E}_k = (\mathbf{I} - \mathbf{A}\mathbf{H}_k)$ 
3:  $\mathbf{u}_k = \mathbf{H}_k \mathbf{E}_k \mathbf{r}_k$ 
4:  $\mathbf{v}_k = \frac{1}{\|\mathbf{A}\mathbf{u}_k\|_2} \mathbf{E}_k^T \mathbf{A}\mathbf{u}_k$ 
5:  $\mathbf{H}_{k+1} = \mathbf{H}_k + \mathbf{u}_k \mathbf{v}_k^T$  ▷ Correction de rang 1
6:  $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{H}_{k+1} \mathbf{r}_k$  ▷ Mise à jour de la solution
7:  $\mathbf{r}_{k+1} = \mathbf{b} - \mathbf{A}\mathbf{x}_{k+1}$  ▷ Mise à jour du résidu
8: if  $\|\mathbf{r}_{k+1}\| \leq \epsilon$  then ▷ Critère d'arrêt
9:   Stop
10: else
11:    $k = k + 1$ 
12:   Retour à l'étape 2
13: end if
```

---

---

**Algorithm 4** Méthode itérative associée à une correction de rang 1 : version améliorée

---

```
1: Input :  $\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_0, \dots, \mathbf{u}_{k-1}, \mathbf{c}_0, \mathbf{c}_1, \dots, \mathbf{c}_{k-1}$ 
2:  $\xi^{(0)} = \mathbf{E}_0 \mathbf{r}_k$ 
3: while  $\|\mathbf{r}_{k+1}\| \geq \epsilon$  do
4:    $k = k + 1$ 
5:   for  $i = 0, \dots, k - 1$  do
6:      $\alpha_i = \langle \mathbf{c}_i, \xi^{(i)} \rangle$ 
7:      $\xi^{(i+1)} = \xi^{(i)} - \alpha_i \mathbf{c}_i$  ▷ Orthogonalisation de  $\xi^{(0)}$  par rapport à  $\mathbf{c}_0, \dots, \mathbf{c}_{k-1}$ 
8:   end for
9:    $\mathbf{u}_k^{(0)} = \mathbf{H}_0 \xi^{(k)}, \mathbf{c}_k^{(0)} = \mathbf{A}\mathbf{u}_k^{(0)} = \mathbf{A}\mathbf{H}_0 \xi^{(k)}$ 
10:  for  $i = 0, \dots, k - 1$  do
11:     $\beta_i = \langle \mathbf{c}_i, \mathbf{E}_i \xi^{(k)} \rangle = -\langle \mathbf{c}_i, \mathbf{c}_k^{(i)} \rangle$  ▷ car  $\xi^{(k)} \perp \mathbf{c}_0, \dots, \mathbf{c}_{k-1}$ 
12:     $\mathbf{u}_k^{(i+1)} = \mathbf{u}_k^{(i)} + \beta_i \mathbf{u}_i$ 
13:     $\mathbf{c}_k^{(i+1)} = \mathbf{c}_k^{(i)} + \beta_i \mathbf{c}_i$ 
14:  end for
15:   $\mathbf{c}_k = \mathbf{c}_k^{(k)} / \|\mathbf{c}_k^{(k)}\|_2, \mathbf{u}_k = \mathbf{u}_k^{(k)} / \|\mathbf{c}_k^{(k)}\|_2,$ 
16:   $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{H}_0 \mathbf{r}_k + \sum_{i=0}^{k-1} \alpha_i \mathbf{u}_i + \mathbf{c}_k^T \xi^{(k)} \mathbf{u}_k$ 
17:   $\mathbf{r}_{k+1} = \xi^{(k)} - \mathbf{c}_k^T \xi^{(k)} \mathbf{c}_k$  ▷ Mise à jour du résidu
18: end while
```

---

---

**Algorithm 5** GMRES<sub>\*</sub>

---

```

1:  $\mathbf{x}_0, \epsilon$ ;
2:  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, k = -1$ ;
3: while  $\|\mathbf{r}_k\| \leq \epsilon$  do
4:    $k = k + 1$ ;
5:   Calcul de  $\mathbf{u}_k^{(0)}$ ;
6:   for  $i = 0, \dots, k - 1$  do
7:      $\alpha_i = \mathbf{c}_i^T \mathbf{c}_k^{(i)}$ ;
8:      $\mathbf{c}_k^{(i+1)} = \mathbf{c}_k^{(i)} - \alpha_i \mathbf{c}_i$ ;
9:      $\mathbf{u}_k^{(i+1)} = \mathbf{u}_k^{(i)} - \alpha_i \mathbf{u}_i$ 
10:  end for
11:   $\mathbf{c}_k = \mathbf{c}_k^{(k)} / \|\mathbf{c}_k^{(k)}\|$ ;  $\mathbf{u}_k = \mathbf{u}_k^{(k)} / \|\mathbf{c}_k^{(k)}\|$ ;
12:   $\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{u}_k \mathbf{c}_k^T \mathbf{r}_{k-1}$ ;
13:   $\mathbf{r}_k = \mathbf{r}_{k-1} - \mathbf{c}_k \mathbf{c}_k^T \mathbf{r}_{k-1}$ ;
14: end while

```

---



---

**Algorithm 6** GCR

---

```

1: Choisir  $\mathbf{x}_0, \epsilon$ ;
2:  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, k = 0$ ;
3: while  $\|\mathbf{r}_k\| \leq tol$  do
4:    $k = k + 1$ ;
5:    $\mathbf{u}_k = \mathbf{r}_{k-1}, \mathbf{c}_k = \mathbf{A}\mathbf{u}_k$ 
6:   for  $i = 0, \dots, k - 1$  do
7:      $\alpha_i = \mathbf{c}_i^T \mathbf{c}_k$ ;
8:      $\mathbf{c}_k = \mathbf{c}_k - \alpha_i \mathbf{c}_i$ ;
9:      $\mathbf{u}_k = \mathbf{u}_k - \alpha_i \mathbf{u}_i$ 
10:  end for
11:   $\mathbf{c}_k = \mathbf{c}_k / \|\mathbf{c}_k\|$ ;  $\mathbf{u}_k = \mathbf{u}_k / \|\mathbf{c}_k\|$ ;
12:   $\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{u}_k \mathbf{c}_k^T \mathbf{r}_{k-1}$ ;
13:   $\mathbf{r}_k = \mathbf{r}_{k-1} - \mathbf{c}_k \mathbf{c}_k^T \mathbf{r}_{k-1}$ ;
14: end while

```

---

---

**Algorithm 7** GMRESR

---

```
1: Choisir  $\mathbf{x}_0, \epsilon$  ;
2:  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, k = 0$  ;
3: while  $\|\mathbf{r}_k\| \leq tol$  do
4:    $k = k + 1$  ;
5:    $\mathbf{u}_k = \mathcal{P}_m(\mathbf{A})\mathbf{r}_{k-1}, \mathbf{c}_k = \mathbf{A}\mathbf{u}_k$  ▷ Solveur GMRES( $m$ ) appliqué à  $\mathbf{A}$ 
6:   for  $i = 0, \dots, k - 1$  do
7:      $\alpha_i = \mathbf{c}_i^T \mathbf{c}_k$  ;
8:      $\mathbf{c}_k = \mathbf{c}_k - \alpha_i \mathbf{c}_i$  ;
9:      $\mathbf{u}_k = \mathbf{u}_k - \alpha_i \mathbf{u}_i$  ;
10:  end for
11:   $\mathbf{c}_k = \mathbf{c}_k / \|\mathbf{c}_k\|$  ;  $\mathbf{u}_k = \mathbf{u}_k / \|\mathbf{c}_k\|$  ;
12:   $\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{u}_k \mathbf{c}_k^T \mathbf{r}_{k-1}$  ;
13:   $\mathbf{r}_k = \mathbf{r}_{k-1} - \mathbf{c}_k \mathbf{c}_k^T \mathbf{r}_{k-1}$  ;
14: end while
```

---

---

**Algorithm 8** GCRO

---

```
1: Choisir  $\mathbf{x}_0, \epsilon$  ;
2:  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, k = 0$  ;
3: while  $\|\mathbf{r}_k\| \leq tol$  do
4:    $k = k + 1$  ;
5:    $\mathbf{u}_k = \mathcal{P}_m(\mathbf{A}_{\mathbf{C}_k})\mathbf{r}_{k-1}, \mathbf{c}_k = \mathbf{A}\mathbf{u}_k$  ▷ Solveur GMRES( $m$ ) appliqué à  $\mathbf{A}_{\mathbf{C}_k}$ 
6:   for  $i = 0, \dots, k - 1$  do
7:      $\alpha_i = \mathbf{c}_i^T \mathbf{c}_k$  ;
8:      $\mathbf{c}_k = \mathbf{c}_k - \alpha_i \mathbf{c}_i$  ;
9:      $\mathbf{u}_k = \mathbf{u}_k - \alpha_i \mathbf{u}_i$  ;
10:  end for
11:   $\mathbf{c}_k = \mathbf{c}_k / \|\mathbf{c}_k\|$  ;  $\mathbf{u}_k = \mathbf{u}_k / \|\mathbf{c}_k\|$  ;
12:   $\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{u}_k \mathbf{c}_k^T \mathbf{r}_{k-1}$  ;
13:   $\mathbf{r}_k = \mathbf{r}_{k-1} - \mathbf{c}_k \mathbf{c}_k^T \mathbf{r}_{k-1}$  ;
14: end while
```

---

---

**Algorithm 9** FGCRO-DR( $m, m_i, k$ )
 

---

- 1: Choisir  $m$ , la taille maximale de l'espace de Krylov externe et  $k$  le nombre de vecteurs propres approchés à déflater. Soit  $tol$  la tolérance de convergence relative. Choisir  $\mathbf{x}_0$ . Calculer  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ , et poser  $i = 1$  et  $s = 1$ .  $s$  correspond au numéro du système linéaire et  $i$  au numéro du cycle au sein du système.
  - 2: **if**  $\mathbf{C}_k$  et  $\mathbf{Z}_k$  sont définis (issus d'un système linéaire précédent) **then**
  - 3:      $\mathbf{x}_1^{(s)} = \mathbf{x}_0^{(s)} + \mathbf{Z}_k \mathbf{C}_k^T \mathbf{r}_0^{(s)}$      ▷ Ne pas mettre à jour  $\mathbf{Z}_k$  and  $\mathbf{C}_k$  car seul le second membre  $\mathbf{b}$  varie
  - 4:      $\mathbf{r}_1^{(s)} = \mathbf{r}_0^{(s)} - \mathbf{C}_k \mathbf{C}_k^T \mathbf{r}_0^{(s)}$
  - 5: **else**
  - 6:      $\mathbf{v}_1 = \mathbf{r}_0^{(s)} / \|\mathbf{r}_0^{(s)}\|_2$
  - 7:      $\mathbf{c} = \|\mathbf{r}_0^{(s)}\|_2 \mathbf{e}_1$
  - 8:     Appliquer  $m$  étapes de FGMRES, résoudre  $\min \|\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}\|_2$  pour  $\mathbf{y}$  et générer  $\mathbf{V}_{m+1}$ ,  $\mathbf{Z}_m$  et  $\bar{\mathbf{H}}_m$ .
  - 9:      $\mathbf{x}_1^{(s)} = \mathbf{x}_0^{(s)} + \mathbf{Z}_m \mathbf{y}$
  - 10:      $\mathbf{r}_1^{(s)} = \mathbf{V}_{m+1} (\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y})$
  - 11:      $\hat{\mathbf{V}}_m = \mathbf{V}_m$
  - 12:     Calculer les  $k$  vecteurs propres  $\tilde{\mathbf{g}}_j$  de  $(\mathbf{H}_m + h_{m+1,m}^2 \mathbf{H}_m^{-T} \mathbf{e}_m \mathbf{e}_m^T) \tilde{\mathbf{g}}_j = \tilde{\theta}_j \tilde{\mathbf{g}}_j$  associés aux plus petites valeurs propres  $\tilde{\theta}_j$  et les stocker dans  $\mathbf{P}_k$ .
  - 13:     Soit [Q,R] la factorisation QR de  $\bar{\mathbf{H}}_m \mathbf{P}_k$ .
  - 14:      $\mathbf{C}_k = \mathbf{V}_{m+1} \mathbf{Q}$
  - 15:      $\mathbf{Z}_k = \mathbf{Z}_m \mathbf{P}_k \mathbf{R}^{-1}$
  - 16:      $\mathbf{U}_k = \hat{\mathbf{V}}_m \mathbf{P}_k \mathbf{R}^{-1}$
  - 17: **end if**
  - 18: **while**  $\|\mathbf{r}_i^{(s)}\|_2 > tol$  **do**
  - 19:      $i = i + 1$
  - 20:     Appliquer  $(m - k)$  étapes d'Arnoldi avec l'opérateur  $(\mathbf{I} - \mathbf{C}_k \mathbf{C}_k^T) \mathbf{A}$  et  $\mathbf{v}_{k+1} = \mathbf{r}_{i-1}^{(s)} / \|\mathbf{r}_{i-1}^{(s)}\|_2$  générant  $\mathbf{V}_{m-k+1}$ ,  $\mathbf{Z}_{m-k}$ ,  $\bar{\mathbf{H}}_{m-k}$ , et  $\mathbf{B}_{m-k}$ .
  - 21:      $\mathbf{V}_{m+1} = [\mathbf{C}_k, \mathbf{V}_{m-k+1}]$
  - 22:      $\mathbf{Z}_m = [\mathbf{Z}_k, \mathbf{Z}_{m-k}]$
  - 23:      $\hat{\mathbf{V}}_m = [\mathbf{U}_k, \mathbf{V}_{m-k}]$
  - 24:      $\bar{\mathbf{H}}_m = \begin{bmatrix} \mathbf{I}_k & \mathbf{B}_{m-k} \\ \mathbf{0} & \bar{\mathbf{H}}_{m-k} \end{bmatrix}$
  - 25:     Résoudre  $\min \|\mathbf{V}_{m+1}^T \mathbf{r}_{i-1}^{(s)} - \bar{\mathbf{H}}_m \mathbf{y}\|_2$  pour  $\mathbf{y}$ .
  - 26:      $\mathbf{x}_i^{(s)} = \mathbf{x}_{i-1}^{(s)} + \mathbf{Z}_m \mathbf{y}$
  - 27:      $\mathbf{r}_i^{(s)} = \mathbf{b} - \mathbf{A} \mathbf{x}_i^{(s)}$
  - 28:     Calculer les  $k$  vecteurs propres  $\tilde{\mathbf{g}}_j$  de  $\bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m \tilde{\mathbf{g}}_j = \tilde{\theta}_j \bar{\mathbf{H}}_m^T \mathbf{V}_{m+1}^T \hat{\mathbf{V}}_m \tilde{\mathbf{g}}_j$  associés aux plus petites valeurs propres  $\tilde{\theta}_j$  et les stocker dans  $\mathbf{P}_k$ .
  - 29:     Soit [Q,R] la factorisation QR de  $\bar{\mathbf{H}}_m \mathbf{P}_k$ .
  - 30:      $\mathbf{C}_k = \mathbf{V}_{m+1} \mathbf{Q}$
  - 31:      $\mathbf{Z}_k = \mathbf{Z}_m \mathbf{P}_k \mathbf{R}^{-1}$
  - 32:      $\mathbf{U}_k = \hat{\mathbf{V}}_m \mathbf{P}_k \mathbf{R}^{-1}$
  - 33: **end while**
  - 34:  $s = s + 1$
  - 35: Recyclage de  $\mathbf{Z}_k$  et  $\mathbf{C}_k$  (pour le système linéaire suivant)
-

---

**Algorithm 10** LGCRO-DR( $m, k - p, p$ )

---

- 1: Choisir  $m, k, \mathbf{x}_0$  et  $tol$ . Soit  $p$  le nombre d'approximation de l'erreur à recycler. Calculer  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ , et poser  $i = 1, s = 1$  et  $\Xi = [ \ ]$
  - 2: **if**  $\mathbf{C}_k$  et  $\mathbf{U}_k$  sont définis (issus d'un système linéaire précédant) **then**
  - 3:    $\mathbf{x}_1^{(s)} = \mathbf{x}_0^{(s)} + t b \mathbf{U}_k \mathbf{C}_k^T \mathbf{r}_0^{(s)}$    ▷ Ne pas mettre à jour  $\mathbf{U}_k$  and  $\mathbf{C}_k$  car seul  $b$  varie
  - 4:    $\mathbf{r}_1^{(s)} = \mathbf{r}_0^{(s)} - \mathbf{C}_k \mathbf{C}_k^T \mathbf{r}_0^{(s)}$
  - 5: **else**
  - 6:    $\mathbf{v}_1 = \mathbf{r}_0^{(s)} / \|\mathbf{r}_0^{(s)}\|_2$
  - 7:    $\mathbf{c} = \|\mathbf{r}_0^{(s)}\|_2 \mathbf{e}_1$
  - 8:   Appliquer GMRES( $m$ ), résoudre  $\min \|\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y}\|_2$  pour  $\mathbf{y}$  et générer  $\mathbf{V}_{m+1}$ , et  $\bar{\mathbf{H}}_m$ .
  - 9:    $\mathbf{x}_1^{(s)} = \mathbf{x}_0^{(s)} + \mathbf{V}_m \mathbf{y}$
  - 10:    $\mathbf{r}_1^{(s)} = \mathbf{V}_{m+1} (\mathbf{c} - \bar{\mathbf{H}}_m \mathbf{y})$
  - 11:    $\hat{\mathbf{V}}_m = \mathbf{V}_m$
  - 12:    $\Xi = [\mathbf{y} \ \Xi]$
  - 13:   Calculer les  $k$  vecteurs propres  $\tilde{\mathbf{g}}_j$  de  $(\mathbf{H}_m + h_{m+1,m}^2 \mathbf{H}_m^{-T} \mathbf{e}_m \mathbf{e}_m^T) \tilde{\mathbf{g}}_j = \tilde{\theta}_j \tilde{\mathbf{g}}_j$  associés aux plus petites valeurs propres  $\tilde{\theta}_j$  et les stocker dans  $\mathbf{P}_k$ .
  - 14:    $\mathbf{P}_k = [\mathbf{P}_{k-1} \ \Xi]$
  - 15:   Soit  $[\mathbf{Q}, \mathbf{R}]$  la factorisation QR de  $\bar{\mathbf{H}}_m \mathbf{P}_k$ .
  - 16:    $\mathbf{C}_k = \mathbf{V}_{m+1} \mathbf{Q}$
  - 17:    $\mathbf{U}_k = \hat{\mathbf{V}}_m \mathbf{P}_k \mathbf{R}^{-1}$
  - 18: **end if**
  - 19: **while**  $\|\mathbf{r}_i^{(s)}\|_2 > tol$  **do**
  - 20:    $i = i + 1$
  - 21:   Appliquer  $(m - k)$  étapes d'Arnoldi avec l'opérateur  $(\mathbf{I} - \mathbf{C}_k \mathbf{C}_k^T) \mathbf{A}$  et  $\mathbf{v}_{k+1} = \mathbf{r}_{i-1}^{(s)} / \|\mathbf{r}_{i-1}^{(s)}\|_2$  générant  $\mathbf{V}_{m-k+1}, \mathbf{Z}_{m-k}, \bar{\mathbf{H}}_{m-k}$ , et  $\mathbf{B}_{m-k}$ .
  - 22:    $\mathbf{V}_{m+1} = [\mathbf{C}_k, \mathbf{V}_{m-k+1}]$
  - 23:    $\hat{\mathbf{V}}_m = [\mathbf{U}_k, \mathbf{V}_{m-k}]$
  - 24:    $\bar{\mathbf{H}}_m = \begin{bmatrix} \mathbf{I}_k & \mathbf{B}_{m-k} \\ \mathbf{0} & \bar{\mathbf{H}}_{m-k} \end{bmatrix}$
  - 25:   Résoudre  $\min \|\mathbf{V}_{m+1}^T \mathbf{r}_{i-1} - \bar{\mathbf{H}}_m \mathbf{y}\|_2$  pour  $\mathbf{y}$ .
  - 26:    $\mathbf{x}_i^{(s)} = \mathbf{x}_{i-1}^{(s)} + \hat{\mathbf{V}}_m \mathbf{y}$
  - 27:    $\mathbf{r}_i^{(s)} = \mathbf{b} - \mathbf{A} \mathbf{x}_i^{(s)}$
  - 28:   Calculer les  $k$  vecteurs propres  $\tilde{\mathbf{g}}_j$  de  $\bar{\mathbf{H}}_m^T \bar{\mathbf{H}}_m \tilde{\mathbf{g}}_j = \tilde{\theta}_j \bar{\mathbf{H}}_m^T \mathbf{V}_{m+1}^T \hat{\mathbf{V}}_m \tilde{\mathbf{g}}_j$  associés aux plus petites valeurs propres  $\tilde{\theta}_j$  et les stocker dans  $\mathbf{P}_k$ .
  - 29:   **if**  $p < i$  **then**
  - 30:      $\Xi = [\mathbf{y} \ \Xi]$
  - 31:      $\mathbf{P}_k = [\mathbf{P}_{k-p} \ \Xi]$
  - 32:   **else**
  - 33:      $\Xi = [\mathbf{y} \ \Xi(:, 1 : i - 1)]$
  - 34:      $\mathbf{P}_k = [\mathbf{P}_{k-i} \ \Xi]$
  - 35:   **end if**
  - 36:   Soit  $[\mathbf{Q}, \mathbf{R}]$  la factorisation QR de  $\bar{\mathbf{H}}_m \mathbf{P}_k$ .
  - 37:    $\mathbf{C}_k = \mathbf{V}_{m+1} \mathbf{Q}$
  - 38:    $\mathbf{U}_k = \hat{\mathbf{V}}_m \mathbf{P}_k \mathbf{R}^{-1}$
  - 39: **end while**
  - 40:  $s = s + 1$
  - 41: Recyclage de  $\mathbf{U}_k$  et  $\mathbf{C}_k$  (pour le système linéaire suivant)
-



# Bibliographie

- [1] A. JAMESON, W. SCHMIDT et E. TURKEL. “Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes”. In : *14th fluid and plasma dynamics conference*. 1981, p. 1259.
- [2] T. ACHARD, C. BLONDEAU et R. OHAYON. “High-Fidelity Aerostructural Gradient Computation Techniques with Application to a Realistic Wing Sizing”. In : *AIAA Journal* 56.11 (2018), p. 4487-4499.
- [3] AC AITKEN. “On Bernulli’s numerical solution of algebraic equations. Proc. R”. In : *Soc. Edinb* 46 (1926), p. 623-634.
- [4] Z.J. ZHANG et D.W. ZINGG. “Efficient monolithic solution algorithm for high-fidelity aerostructural analysis and optimization”. In : *AIAA Journal* 56.3 (2018), p. 1251-1265.
- [5] E. DE STURLER. “Truncation strategies for optimal Krylov subspace methods”. In : *SIAM Journal on Numerical Analysis* 36.3 (1999), p. 864-889.
- [6] V. SIMONCINI et D.B. SZYLD. “Flexible inner-outer Krylov subspace methods”. In : *SIAM Journal on Numerical Analysis* 40.6 (2002), p. 2219-2239.
- [7] G.K.W KENWAY, G.J KENNEDY et J.R.R.A MARTINS. “Scalable parallel approach for high-fidelity steady-state aeroelastic analysis and adjoint derivative computations”. In : *AIAA journal* 52.5 (2014), p. 935-951.
- [8] Y. SAAD et M.H. SCHULTZ. “GMRES : A generalized minimal residual algorithm for solving nonsymmetric linear systems”. In : *SIAM Journal on scientific and statistical computing* 7.3 (1986), p. 856-869.
- [9] X. PINEL et M. MONTAGNAC. “Block Krylov methods to solve adjoint problems in aerodynamic design optimization”. In : *AIAA journal* 51.9 (2013), p. 2183-2191.
- [10] C-H. CHEN et S. NADARAJAH. “GCRO with dynamic deflated restarting for solving adjoint systems of equations for aerodynamic shape optimization”. In : *International Journal of Numerical Methods for Heat & Fluid Flow* (2019).
- [11] J. PETER et F. DRULLION. “Large stencil viscous flux linearization for the simulation of 3D compressible turbulent flows with backward-Euler schemes”. In : *Computers & fluids* 36.6 (2007), p. 1005-1027.



- [12] Zhouie LYU et al. “Automatic differentiation adjoint of the Reynolds-averaged Navier-Stokes equations with a turbulence model”. In : *21st AIAA Computational Fluid Dynamics Conference*. 2013, p. 2581.
- [13] M.L PARKS et al. “Recycling Krylov subspaces for sequences of linear systems”. In : *SIAM Journal on Scientific Computing* 28.5 (2006), p. 1651-1674.
- [14] L.M. CARVALHO et al. “A flexible generalized conjugate residual method with inner orthogonalization and deflated restarting”. In : *SIAM Journal on Matrix Analysis and Applications* 32.4 (2011), p. 1212-1235.
- [15] ONERA M6 Wing, <https://www.onera.fr/fr/actualites/laile-onera-m6-star-de-la-cfd>. Accessed : 2023-02-08.
- [16] L. CAMBIER, S. HEIB et S. PLOT. “The Onera elsA CFD software : input from research and feedback from industry”. In : *Mechanics & Industry* 14.3 (2013), p. 159-174.
- [17] S. YOON et A. JAMESON. “Lower-upper symmetric-Gauss-Seidel method for the Euler and Navier-Stokes equations”. In : *AIAA journal* 26.9 (1988), p. 1025-1026.
- [18] Joaquim RRA MARTINS, Peter STURDZA et Juan J ALONSO. “The complex-step derivative approximation”. In : *ACM Transactions on Mathematical Software (TOMS)* 29.3 (2003), p. 245-262.
- [19] Jacques EV PETER et Richard P DWIGHT. “Numerical sensitivity analysis for aerodynamic optimization : A survey of approaches”. In : *Computers & Fluids* 39.3 (2010), p. 373-391.
- [20] Timothée ACHARD. “Techniques de calcul de gradient aéro-structure haute-fidélité pour l’optimisation de voilures flexibles”. Thèse de doct. Paris, CNAM, 2017.
- [21] D.C. WILCOX. “Reassessment of the scale-determining equation for advanced turbulence models”. In : *AIAA journal* 26.11 (1988), p. 1299-1310.
- [22] B. VAN LEER. “Towards the ultimate conservative difference scheme”. In : *Journal of computational physics* 135.2 (1997), p. 229-248.
- [23] R.M. BEAM et R.F. WARMING. “An implicit finite-difference algorithm for hyperbolic systems in conservation-law form”. In : *Journal of computational physics* 22.1 (1976), p. 87-110.
- [24] L. GIRAUD et al. “Flexible GMRES with deflated restarting”. In : *SIAM Journal on Scientific Computing* 32.4 (2010), p. 1858-1878.
- [25] C.BLONDEAU. *Compatibility of GMRES solver with the elsA/Aoc aeroelastic shape gradient computation module*. Rapp. tech. ONERA, Juin 2018.
- [26] C.BLONDEAU. *elsAASO-2017 - AS-5.2 : GMRES solver compatibility for aeroelastic shape gradient*. Rapp. tech. ONERA, December 2017.

- 
- [27] A. BISSUEL. “Résolution des équations de Navier-Stokes linéarisées pour l’aéroélasticité, l’optimisation de forme et l’aéroacoustique”. Thèse de doct. Université Paris-Saclay (ComUE), 2018.
- [28] Ronald B MORGAN. “Restarted block-GMRES with deflation of eigenvalues”. In : *Applied Numerical Mathematics* 54.2 (2005), p. 222-236.
- [29] J. VASSBERG et al. “Development of a common research model for applied CFD validation studies”. In : *26th AIAA applied aerodynamics conference*. 2008, p. 6919.
- [30] J.E. HICKEN et D.W. ZINGG. “A simplified and flexible variant of GCROT for solving nonsymmetric linear systems”. In : *SIAM Journal on Scientific Computing* 32.3 (2010), p. 1672-1694.
- [31] J.Z. ZHANG. “Exploratory High-Fidelity Aerostructural Optimization Using an Efficient Monolithic Solution Method”. Thèse de doct. 2017.
- [32] C.BLONDEAU. *Méthode hybride directe/adjointe de calcul de sensibilités aéroélastiques. Mise en œuvre dans un environnement modulaire Python elsA*. Rapp. tech. ONERA, Janvier 2021.
- [33] P. RAUD. *elsAASO-2016-AS-5.1 : GMRES for adjoint method*. Rapp. tech. ONERA, July 2017.
- [34] W.E. ARNOLDI. “The principle of minimized iterations in the solution of the matrix eigenvalue problem”. In : *Quarterly of applied mathematics* 9.1 (1951), p. 17-29.
- [35] Charles F VAN LOAN et G GOLUB. “Matrix computations (Johns Hopkins studies in mathematical sciences)”. In : (1996).
- [36] R.B. MORGAN. “GMRES with deflated restarting”. In : *SIAM Journal on Scientific Computing* 24.1 (2002), p. 20-37.
- [37] Y. SAAD. *Iterative methods for sparse linear systems*. SIAM, 2003.
- [38] A. CHAPMAN, Y. SAAD et L. WIGTON. “High-order ILU preconditioners for CFD problems”. In : *International Journal for numerical methods in fluids* 33.6 (2000), p. 767-788.
- [39] Y. SAAD. “ILUT : A dual threshold incomplete LU factorization”. In : *Numerical linear algebra with applications* 1.4 (1994), p. 387-402.
- [40] Y. SAAD. “A flexible inner-outer preconditioned GMRES algorithm”. In : *SIAM Journal on Scientific Computing* 14.2 (1993), p. 461-469.
- [41] Wolfgang HACKBUSCH. *Multi-grid methods and applications*. T. 4. Springer Science & Business Media, 2013.
- [42] J ZHU et Wanchun TANG. “The multigrid preconditioned flexible GMRES solver for hierarchical TVFEM analysis”. In : *Microwave and Optical Technology Letters* 49.8 (2007), p. 2012-2018.

- [43] YT FENG, D PERIĆ et DRJ OWEN. “A multi-grid enhanced GMRES algorithm for elasto-plastic problems”. In : *International Journal for Numerical Methods in Engineering* 42.8 (1998), p. 1441-1462.
- [44] Christophe BLONDEAU et Cedric LIAUZUN. “A modular implementation of the time spectral method for aeroelastic analysis and optimization on structured meshes”. In : *IFASD 2019*. 2019.
- [45] C. HIRSCH. “Numerical computation of internal and external flows, Volume 1 : Fundamentals of numerical discretization”. In : *John Wiley and Sons* 9 (1988), p. 10.
- [46] A. RESMINI. “Sensitivity analysis for numerical simulation of compressible flows in external aerodynamics.” Thèse de doct. Université Pierre et Marie Curie Paris VI, 2015.
- [47] S. LE BRAS. “Modélisation de paroi et traitement aux interfaces des maillages non-conformes pour les simulations aéroacoustiques avec une approche numérique d’ordre élevé”. Thèse de doct. Université de Lyon, 2016.
- [48] Assefaw Hadish GEBREMEDHIN, Fredrik MANNE et Alex POTHEN. “What color is your Jacobian? Graph coloring for computing derivatives”. In : *SIAM review* 47.4 (2005), p. 629-705.
- [49] Mark EMBREE. “How descriptive are GMRES convergence bounds?” In : *arXiv preprint arXiv :2209.01231* (2022).
- [50] R.B. MORGAN et M. ZENG. “Harmonic projection methods for large non-symmetric eigenvalue problems”. In : *Numerical linear algebra with applications* 5.1 (1998), p. 33-55.
- [51] Cornelis VUIK et Henk A van der VORST. “A comparison of some GMRES-like methods”. In : *Linear Algebra and its Applications* 160 (1992), p. 131-162.
- [52] Stefan RÖLLIN et Wolfgang FICHTNER. “Improving the accuracy of GMRES with deflated restarting”. In : *SIAM Journal on Scientific Computing* 30.1 (2008), p. 232-245.
- [53] Mehdi JADOUI et al. “Comparative study of inner–outer Krylov solvers for linear systems in structured and high-order unstructured CFD problems”. In : *Computers & Fluids* 244 (2022), p. 105575.
- [54] Timo EIROLA et Olavi NEVANLINNA. “Accelerating with rank-one updates”. In : *Linear Algebra and its Applications* 121 (1989), p. 511-520.
- [55] E DE STURLER. “Inner-outer methods with deflation for linear systems with multiple right-hand sides”. In : *Householder Symposium XIII, Proceedings of the Householder Symposium on Numerical Algebra, Pontresina, Switzerland*. 1996, p. 193-196.
- [56] L. M. CARVALHO et al. *A Flexible Generalized Conjugate Residual Method with Inner Orthogonalization and Deflated Restarting*. Rapp. tech. CERFACS, 2011.

- 
- [57] Kirk M SOODHALTER, Eric de STURLER et Misha E KILMER. “A survey of subspace recycling iterative methods”. In : *GAMM-Mitteilungen* 43.4 (2020), e202000016.
- [58] E. de STURLER. “Nested Krylov methods based on GCR”. In : *Journal of Computational and Applied Mathematics* 67.1 (1996), p. 15-41.
- [59] A. BAKER, E.R JESSUP et Thomas MANTEUFFEL. “A technique for accelerating the convergence of restarted GMRES”. In : *SIAM Journal on Matrix Analysis and Applications* 26.4 (2005), p. 962-984.
- [60] Michele BENZI, Gene H GOLUB et Jörg LIESEN. “Numerical solution of saddle point problems”. In : *Acta numerica* 14 (2005), p. 1-137.
- [61] Yu A KUZNETSOV. “Efficient iterative solvers for elliptic finite element problems on nonmatching grids”. In : (1995).
- [62] Malcolm F MURPHY, Gene H GOLUB et Andrew J WATHEN. “A note on preconditioning for indefinite linear systems”. In : *SIAM Journal on Scientific Computing* 21.6 (2000), p. 1969-1972.
- [63] M. HEIL, A.L. HAZEL et J. BOYLE. “Solvers for large-displacement fluid–structure interaction problems : segregated versus monolithic approaches”. In : *Computational Mechanics* 43.1 (2008), p. 91-101.
- [64] M. HEIL. “An efficient solver for the fully coupled solution of large-displacement fluid–structure interaction problems”. In : *Computer Methods in Applied Mechanics and Engineering* 193.1-2 (2004), p. 1-23.