

# Three-Dimensional Aeroelastic Solutions via the Nonlinear Frequency Domain Method

Pierre-Olivier Tardif

Master of Engineering

Department of Mechanical Engineering

McGill University

Montréal, Québec

August 2015

A thesis submitted to McGill University in partial fulfillment of the requirements of  
the degree of Master of Engineering

© Pierre-Olivier Tardif 2015  
All rights reserved

## ACKNOWLEDGEMENTS

First of all, I want to thank my advisor, professor Siva Nadarajah, for his indispensable guidance throughout this research. His vast technical insight was an essential asset in the realization of this work, and his constant encouragement and patience were key factors in the success of this research. Second, I want to thank my colleagues and friends from McGill's Computational Aerodynamics Group, who did not only provide me with technical assistance but also contributed in making my journey to graduate studies far more pleasant. I also definitely want to thank the National Sciences and Engineering Research Council of Canada for its financial support through grant 464871/2014 and Hydro-Quebec for its generous Masters Fellowship in Engineering. Furthermore, I want to express my deep gratitude to my family, in which I include my family-in-law, for being such loving, encouraging and comprehensive people. The moments of rejoice that we have been able to have throughout this research were most welcomed when motivation seemed to lack. I am in addition profoundly thankful to my parents for pushing me to the limit of what I can accomplish and making me believe in myself. Knowing their pride towards me fuelled my ambitions and encouraged me to envision bold goals. Finally, I am eternally grateful to my wife, Tamara, who encouraged and loved me all along the road. Her abundant love does fill me with happiness every day that goes by.

## ABSTRACT

A three-dimensional aeroelastic solver using the Nonlinear Frequency Domain (NLFD) method in conjunction with a novel dynamic mesh deformation approach using Radial Basis Functions (RBF) is developed. A structured multiblock finite-volume flow solver coupled to a plate bending finite-element model is employed for aeroelastic computations. The novelty of the developed dynamic mesh deformation method consists in the computation of the mesh velocities using the RBF matrix operator for use in the Arbitrary Lagrangian-Eulerian formulation of the flow governing equations. The approach is validated for a pitching two-dimensional NACA 0012 airfoil and a pitching three-dimensional LANN wing against the rigid grid motion method, the Fast Fourier Transform (FFT) mesh velocity computation technique, as well as experimental results. Both the accuracy and the efficiency of the method are validated. A methodology for the assessment of flutter and limit cycle oscillations (LCO) using the NLFD approach is developed. The NLFD/LCO aeroelastic solver is validated based on previous work and experimental results for the AGARD I.-Wing 445.6 Weakened Model 3 in air and Solid Model 2 in R-12. The flutter boundary and LCO trends are assessed for both cases. The method is estimated to perform aeroelastic computations an order of magnitude faster than typical time-marching methods.

## ABRÉGÉ

Un solveur aéroélastique tridimensionnel employant la méthode Non Linéaire du Domaine Fréquentiel (NLDF) ainsi qu'une approche novatrice de déformation dynamique de maillage par Fonctions de Base Radiales (FBR) sont développés. Un solveur fluide multibloc à maillage structuré couplé à un modèle d'éléments finis de plaques en flexion est employé pour les calculs aéroélastiques. La nouveauté de la méthode de déformation dynamique de maillage réside dans le calcul des vitesses nodales du maillage à partir de l'opérateur des FBR, dans le but de les utiliser par la suite dans la formulation Arbitrairement Lagrangienne-Eulérienne (ALE) des équations gouvernant l'écoulement. Cette approche est validée pour un profil d'aile NACA 0012 et une aile LANN tridimensionnelle soumis à une oscillation angulaire, et est comparée aux méthodes du mouvement rigide de maillage et du calcul de vitesses de maillage par Transformée de Fourier Rapide (TFR), ainsi qu'à des résultats expérimentaux. La précision et l'efficacité de la méthode sont validées. Une méthodologie pour la détermination des conditions de flottement et d'oscillations de cycle limite (OCL) est développée en employant la méthode NLDF. Le solveur aéroélastique NLDF/OCL résultant est validé en fonction de travaux antérieurs et de résultats expérimentaux pour l'aile I-Wing 445.6 de l'AGARD; le modèle affaibli 3 est testé dans l'air, et le modèle solide 2 est étudié dans le R12. Les conditions de flottement linéaire et d'OCL de l'aile sont déterminées. Il est estimé que la méthode proposée soit en mesure d'exécuter des calculs aéroélastiques environ un ordre de grandeur plus rapidement que les techniques usuelles d'intégration en temps.

## TABLE OF CONTENTS

ACKNOWLEDGEMENTS . . . . .	ii
ABSTRACT . . . . .	iii
ABRÉGÉ . . . . .	iv
LIST OF TABLES . . . . .	viii
LIST OF FIGURES . . . . .	ix
1 Introduction . . . . .	1
1.1 Motivation . . . . .	1
1.2 Frequency-Domain Solution Methods . . . . .	5
1.3 Aeroelastic Solutions using Frequency-Domain Methods . . . . .	9
1.4 Dynamic Mesh Deformation for Aeroelastic Solutions . . . . .	13
1.5 Research Objectives . . . . .	16
2 Flow Solver . . . . .	17
2.1 Governing Equations . . . . .	17
2.2 Spatial Discretization . . . . .	19
2.3 Temporal Discretization . . . . .	20
2.3.1 Nonlinear Frequency Domain Method . . . . .	20
2.3.2 Pseudotime-Stepping Scheme . . . . .	23
3 Structural Solver . . . . .	27
3.1 Governing Equations . . . . .	27
3.2 Spatial Discretization . . . . .	28
3.3 Temporal Discretization . . . . .	33
4 Dynamic Mesh Deformation . . . . .	36
4.1 Radial Basis Function Framework . . . . .	36

4.1.1	Mesh Deformation . . . . .	36
4.1.2	Mesh Velocities . . . . .	41
4.2	Treatment of the Geometric Conservation Law . . . . .	45
5	Aeroelastic Framework . . . . .	51
5.1	Fluid-Structure Coupling . . . . .	51
5.2	Interpolation of Aerodynamic Forces and Displacements . . . . .	52
5.2.1	Spatial Interpolation . . . . .	52
5.2.2	Temporal Discretization Interface . . . . .	53
5.3	Limit Cycle Oscillation Determination Methodology . . . . .	55
6	Results - Dynamic Mesh Deformation . . . . .	62
6.1	NACA 0012 Airfoil - <i>Run 81</i> of the CT1 . . . . .	62
6.1.1	Accuracy of Mesh Velocities using the RBFV and FFTV Methods . . . . .	63
6.1.2	Solution Accuracy using the RBFV Method . . . . .	63
6.1.3	Efficiency of the RBFV and FFTV Methods . . . . .	65
6.2	LANN Wing - <i>Run 73</i> of the CT5 . . . . .	69
6.2.1	Solution Accuracy using the RBFV Method . . . . .	70
6.2.2	Efficiency of the RBFV and FFTV Methods . . . . .	72
7	Results - Aeroelastic Solutions . . . . .	79
7.1	AGARD I.-Wing 445.6 Weakened Model 3 . . . . .	79
7.1.1	Structural Solver Validation . . . . .	79
7.1.2	Spatial and Temporal Accuracy Analyses . . . . .	84
7.1.3	Flutter Results in Air . . . . .	87
7.1.4	Limit Cycle Oscillation Results in Air . . . . .	92
7.2	AGARD I.-Wing 445.6 Solid Model 2 . . . . .	104
7.2.1	Flutter Results in R-12 . . . . .	105
7.2.2	Limit Cycle Oscillation Results in R-12 . . . . .	106
7.3	Estimated Acceleration . . . . .	109
8	Conclusions . . . . .	112
8.1	Summary . . . . .	112
8.2	Limitations . . . . .	113
8.3	Future Work . . . . .	115

REFERENCES . . . . . 116

LIST OF TABLES

Table	page
2-1 Stage and Blending Coefficients of the Hybrid Multistage Pseudotime-Stepping Scheme . . . . .	25
6-1 Computational Cost of the FFTV and RBFV Methods for <i>Run 81</i> using 4 Modes . . . . .	69
6-2 Fourier Coefficients of Lift and Drag Coefficients using the Rigid Grid Motion and RBFV Methods for <i>Run 73</i> . . . . .	73
6-3 Computational Cost of the FFTV and RBFV Methods for <i>Run 73</i> using 2 Modes . . . . .	76
6-4 Computational Cost of the FFTV Method for Various Accuracy Levels for <i>Run 73</i> . . . . .	78
7-1 First Four Natural Frequencies of the AGARD I.-Wing 445.6 Weakened Model 3 . . . . .	81
7-2 Fourier Coefficients of Lift and Drag Coefficients on Various Grids for the AGARD I.-Wing 445.6 Undergoing Prescribed Oscillatory Deformation, $M = 0.960$ , $\omega_r = 0.07892$ , $\delta = 377.0 \times 10^{-5}$ . . . . .	86
7-3 First Four Natural Frequencies of the AGARD I.-Wing 445.6 Solid Model 2 . . . . .	105
7-4 Comparison of the Estimated Computational Cost of the NLFD/LCO Method and of a Time-Marching Approach for the Determination of Limit Cycle Oscillation Flow Conditions . . . . .	111

## LIST OF FIGURES

<u>Figure</u>	<u>page</u>
1–1 Benign and Detrimental Effects of Nonlinearities on Limit Cycle Oscillations . . . . .	4
2–1 Simplified Dataflow Diagram of the Nonlinear Frequency Domain Method Using a Modified Runge-Kutta Technique . . . . .	26
3–1 Arbitrary Structural Element and Example of Three-Point Quadrature Points ( $q_i$ ) . . . . .	29
4–1 Deformation of the Mesh around a NACA 0012 Airfoil using the RBF Method . . . . .	41
5–1 Dataflow Diagram of the Temporal Discretization Interface of the Fluid-Structure Coupling . . . . .	54
6–1 NACA 0012 Undeformed and Deformed Meshes . . . . .	64
6–2 Convergence of Mesh Velocity Errors in $L^\infty$ and $L^2$ for the FFTV and RBFV Methods at $t = 0$ , <i>Run 81</i> . . . . .	64
6–3 Flow Solver Convergence using Rigid Grid Motion, <i>Run 81</i> . . . . .	66
6–4 Drag Coefficient Hysteresis using Rigid Grid Motion, <i>Run 81</i> . . . . .	66
6–5 Comparison of the Rigid Grid Motion and RBFV Methods for the Computation of Lift and Drag Coefficient Hysteresis, <i>Run 81</i> . . . . .	67
6–6 Comparison of the Rigid Grid Motion and RBFV Methods for the Computation of the Pressure Coefficient Distribution along the Airfoil Chord, <i>Run 81</i> . . . . .	68
6–7 LANN Wing Mesh . . . . .	70
6–8 Flow Solver Convergence using Rigid Grid Motion, <i>Run 73</i> . . . . .	71

6-9	Drag Coefficient Hysteresis using Rigid Grid Motion, <i>Run 73</i> . . . . .	71
6-10	Flow Solver Convergence using the RBFV Method, <i>Run 73</i> . . . . .	73
6-11	Comparison of the Rigid Grid Motion and RBFV Methods for the Computation of Lift and Drag Coefficient Hysteresis, <i>Run 73</i> . . .	74
6-12	Comparison of the Rigid Grid Motion and RBFV Methods for the Computation of the Pressure Coefficient Distribution at Two Spanwise Stations, $t = 0$ , <i>Run 73</i> . . . . .	75
7-1	Structural Mesh for the AGARD I.-Wing 445.6 . . . . .	81
7-2	Mode Shapes of the AGARD I.-Wing 445.6 Weakened Model 3 as Calculated by Yates [90] . . . . .	82
7-3	Calculated Mode Shapes of the AGARD I.-Wing 445.6 Weakened Model 3 . . . . .	83
7-4	AGARD I.-Wing 445.6 Flow Solver Mesh . . . . .	86
7-5	Flow Solver Convergence for the AGARD I.-Wing 445.6 Weakened Model 3 Undergoing Prescribed Oscillatory Deformation, $M =$ $0.960$ , $\omega_r = 0.07892$ , $\delta = 377.0 \times 10^{-5}$ . . . . .	87
7-6	Lift and Drag Coefficient Hysteresis for the AGARD I.-Wing 445.6 Weakened Model 3 Undergoing Prescribed Oscillatory Deformation, $M = 0.960$ , $\omega_r = 0.07892$ , $\delta = 377.0 \times 10^{-5}$ . . . . .	88
7-7	Pressure Coefficient Distribution at Two Spanwise Stations on the AGARD I.-Wing 445.6 Weakened Model 3 Undergoing Prescribed Oscillatory Deformation, $M = 0.960$ , $\omega_r = 0.07892$ , $\delta = 377.0 \times$ $10^{-5}$ , $t = 0$ . . . . .	89
7-8	Flutter Boundary of the AGARD I.-Wing 445.6 Weakened Model 3 in Air: Calculated, Thomas et al. [82], and Experimental [91] . . .	91
7-9	LCO Behavior Trends of the AGARD I.-Wing 445.6 Weakened Model 3 in Air . . . . .	94
7-10	Flow Solver Convergence for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method, $\delta_i = 301.6 \times 10^{-5}$ . . . . .	97

7-11	Convergence of Reduced Frequency and Speed Index for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method, $\delta_i = 301.6 \times 10^{-5}$ , $M = 0.960$	98
7-12	Convergence of Reduced Frequency and Speed Index for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method, $\delta_i = 301.6 \times 10^{-5}$ , $M = 1.072$	99
7-13	Truncated Time History of the Lift and Drag Coefficients for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method, $\delta_i = 301.6 \times 10^{-5}$ , $M = 0.960$	100
7-14	Truncated Time History of the Lift and Drag Coefficients for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method, $\delta_i = 301.6 \times 10^{-5}$ , $M = 1.072$	101
7-15	Contour Plots for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method, $\delta_i = 301.6 \times 10^{-5}$ , $M = 1.072$ , $t/T = 0.25$	102
7-16	Pressure Coefficient Distribution at Two Spanwise Stations for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method, $\delta_i = 301.6 \times 10^{-5}$ , $M = 1.072$ , $t/T = 0.25$	103
7-17	Flutter Boundary of the AGARD I.-Wing 445.6 Solid Model 2 in R-12: Calculated and Experimental [91]	107
7-18	LCO Behavior Trends of the AGARD I.-Wing 445.6 Solid Model 2 in R-12	108

## CHAPTER 1

### Introduction

#### 1.1 Motivation

Over the past several decades, engineering disciplines have greatly converged towards the employment of computational software due to the enormous efficiency advantages computers are able to provide. As computer calculations became increasingly fast, problems of greater complexity could be tackled by engineers within reasonable time frames. Refined computer hardware technologies and increasing computing capacities gave rise to novel engineering fields which took advantage of those new possibilities. Among them stands the Computational Fluid Dynamics (CFD) discipline, of which goal is to predict the behavior of fluid flows subject to predefined conditions through the use of numerical simulations relying on the governing equations of fluid flows. The need for this discipline originates from the difficulty to predict the flow behavior around even only slightly complex geometries such as two-dimensional airfoils, using solely analytical methods. Although such methods have been known to engineers for a long time, their very restrictive limitations motivated the development of numerical techniques such as CFD. The latter is nowadays widely employed in the industry. For instance, aircraft manufacturers make use of it to characterize aircraft components, in view of producing optimal designs. Its utilization directly translates into cost savings, as it shortens design duration and reduces

the need for experimentation, such as wind tunnel and flight testing, of which costs are orders of magnitude greater than CFD computations.

Despite the improvement in computing techniques, some complex problems remain very time-intensive due to the high amount of calculations they require. This is the case, among others, of unsteady flow simulations, where the flow behavior varies in time. Although steady-state problems already require a substantial amount of calculations due to the nonlinear and hyperbolic characters of the governing equations, unsteady flows necessitate additional effort, primarily because of two factors. First, the solution typically needs to be marched in time by constructing the temporal history of the flow at multiple time instances, thus requiring more computational effort than steady-state problems, which only need the solution at a single time instance. Second, a lot of effort is deployed to eliminate the initial transient effects that are due to the initialization of the flow domain to the free-stream values, which renders the first few steps of the solution totally inaccurate. Despite these drawbacks, unsteady flow simulations are widely employed in the industry since they are essential to the aircraft design process, particularly in the assessment of the aeroelastic behavior of aircraft components.

Aeroelasticity is the discipline studying the interaction between a structure and the fluid flow surrounding it. It is primordial in the design process of aircraft since undesired aeroelastic instabilities, be they static or dynamic, often lead to catastrophic structural failure or severe fatigue damage. For example, divergence is a static instability characterized by aerodynamic loads exceeding the structural resisting forces, leading to excessive deformation of structural components and, ultimately,

to their failure. On the other hand, dynamic instabilities are characterized by a dynamic interaction between the body and the fluid flow. Flutter is defined as the dynamic instability caused by a positive feedback between the aerodynamic loads and the deformation of the structure, producing a negative global damping. This negative damping results in diverging oscillations of the body, ultimately leading to its structural failure. For linear systems, the flutter point is defined as the exact stability limit between diverging and damped oscillations of the body, corresponding to the point where the net damping is exactly zero. Therefore, a linear stability analysis is sufficient to assess accurately the flutter point of a structure.

Limit cycle oscillations (LCO), on the other hand, consist of self-sustained oscillations of constant energy that are caused by either aerodynamic or structural nonlinearities. Hence, they cannot be modelled using linear stability analyses, and require a nonlinear approach. Dynamic nonlinearities may be caused, among others, by the movement of a shock wave on the surface of the body, flow separation, or any structural nonlinearity. It is therefore common to encounter LCO in transonic aeroelastic analyses, due to the presence of a moving shock wave. As regards to the flutter point, it is characterized by an LCO of which amplitude is sufficiently small that the nonlinear effects in the flow and the structure are negligible. In practice, aircraft engineers are concerned by the lowest flight speed at which either flutter or an LCO occurs, because none are desirable. In some cases, flutter will occur before LCO of higher amplitudes. Such cases are deemed to possess benign LCO nonlinearities [15], since the flight speed at which LCO occur for a constant Mach number increases as the LCO amplitude increases, as shown in Figure 1–1(a). Flutter is

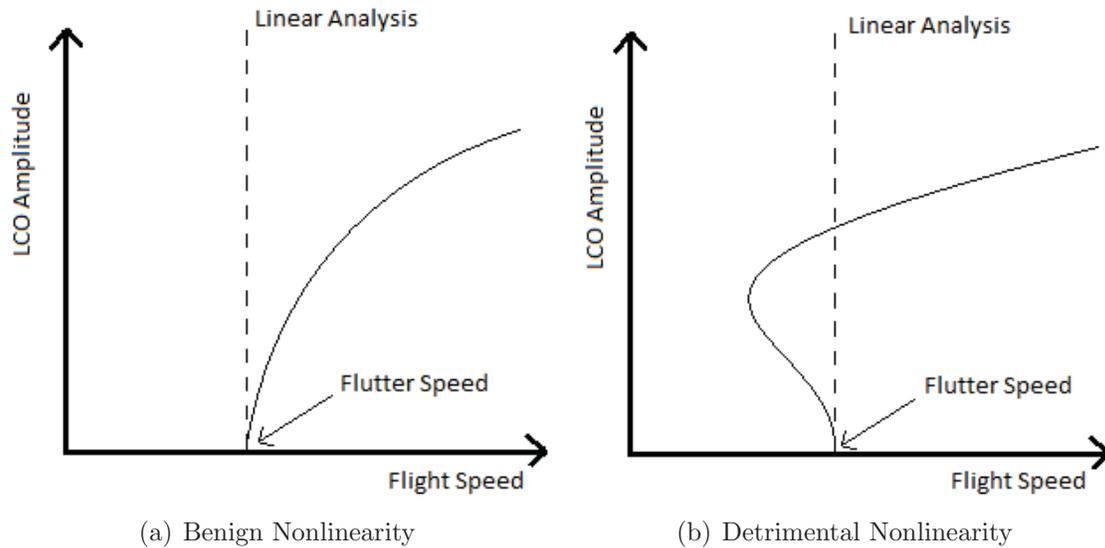


Figure 1-1: Benign and Detrimental Effects of Nonlinearities on Limit Cycle Oscillations (reproduced and adapted from [15])

thus the sole phenomenon that needs to be assessed, since it is the most critical, by occurring at the lowest flight speed. However, in other cases, so-called detrimental nonlinearities [15] may cause LCO to occur at lower speeds than the flutter point, as depicted in Figure 1-1(b), provided that the initial perturbation is sufficiently large. In such cases, a strict assessment of the flutter speed would lead to dangerous designs, since the occurrence of LCO below the flutter speed could dramatically reduce the desired safety margins or the performance of the aircraft. This behavior is partly responsible for the transonic dip phenomenon, which is characterized by a drop in the flutter speed at transonic Mach numbers. Although it does not correspond to flutter in the proper sense since it results from dynamic nonlinearities, the transonic dip is said to reduce the flutter speed because LCO can occur before linear flutter.

Because of the safety issues that dynamic instabilities engender, manufacturers must ensure that they do not occur within the flight envelope of the designed aircraft. However, modelling them requires a very high computational effort because of their nonlinear unsteady character. For that reason, the aeroelastic behavior of aircraft components is commonly assessed using low-fidelity or linear methods along with generous safety margins to ensure a large stability region. As a consequence, designs are generally suboptimal as far as aeroelastic behavior is concerned. High-fidelity methods would allow design optimization based on aeroelastic constraints to be included within the design loop, but current methods are computationally expensive and render this approach impractical.

This research is therefore motivated by the need for high-efficiency high-fidelity aeroelastic simulation techniques, such that they are viable for industrial applications. For example, a high-fidelity aeroelastic solver could be embedded in an aerodynamic shape optimization framework, which is now of common usage in the industry, as it allows for numerical rather than manual optimization of designs. Optimal designs, for their part, procure undeniable competitive advantages to aircraft manufacturers, since they directly result in cost savings for airlines by reducing the drag, increasing the range, and improving other desirable properties of the aircraft. The need for efficient high-fidelity aeroelastic solvers is thereby highly justified.

## **1.2 Frequency-Domain Solution Methods**

Until lately, the most popular approach for solving unsteady flows was Jameson's dual time-stepping technique [38], which nowadays has the recognition of the entire CFD community. The appellation stems from the technique's distinction between

physical (or real) time and pseudotime. The unsteadiness of the flow is accounted for by using a second-order backward finite-difference representation of the temporal derivative of the solution in physical time, whereas the solution at each physical time step is converged to a steady state using a pseudotime-marching approach. This allows for acceleration techniques such as local time-stepping and multigrid strategies [36, 37] to be employed. Since the solution is marched in time from the initial free-stream values to the final solution by constructing the time history of the flow, the method is deemed time-accurate (or time-marching). Unfortunately, despite its benefits, the dual time-stepping method remains computationally costly due to its need to converge to a solution at numerous time steps as well as because of the lengthy initial transient effects.

Fortunately, some great advancements were made in the field of unsteady flow computations in the last two decades, especially for periodic flows, characterized by continually repeated properties over time. In such flows, the solution is repeated once every period  $T$ , such that any property  $\phi$  of the flow at time  $t$  is equal one period later;  $\phi(t) = \phi(t + T)$ . Since a large amount of unsteady flows studied by engineers are indeed periodic, many of the aforementioned advancements were directed towards the efficient solving of such flows. In that respect, various frequency-domain flow solution methods emerged, in which the periodicity of the flow is exploited in order to accelerate the convergence to the final solution, based on the representation of the flow using a chosen number of harmonics.

At first, periodic time-linearized techniques [27] were welcomed by the aerospace industry due to their high computational efficiency. Such techniques use a temporal

linearization of the flow by assuming small periodic perturbations to the nonlinear steady background flow. A steady flow is solved using a conventional solver, and a system of wave equations is then solved for all the desired harmonics of the unsteady part of the flow. However, the required linearization translates directly into a lack of ability to model nonlinearities, as occurs in unsteady transonic flows. Semi-nonlinear methods, or deterministic stress [1] methods, such as the Nonlinear Harmonic (NLH) [32, 66] or phase solution [30] techniques, were derived in order to improve the capture of dynamically nonlinear flow patterns. These techniques are very similar to linearized frequency-domain approaches, but retain from the equations some additional stress terms that account partially for the effect of the unsteady perturbations to the steady background flow. The original approach could only be used to represent the flow using its fundamental frequency and a number of its first multiples, but He et al. [31] extended it such that various frequencies could be used without being multiples of one another, and employed it in turbomachinery applications. Both versions allow for an accurate modelling of weak nonlinear patterns, but still lack accuracy for flows containing strong dynamic nonlinearities.

This shortcoming motivated the development of fully nonlinear frequency-domain techniques such as the Harmonic Balance (HB) [28] and Time Spectral (TS) [25] methods. The former was thereafter modified by McMullen [54], giving rise to the Nonlinear Frequency Domain (NLFD) method [52]. These methods take advantage of the periodicity of the flow and use trigonometric interpolants to represent the solution accurately with very few time instances. Therefore, the flow is decomposed into a predefined number of harmonics (multiples of the fundamental frequency)

while preserving entirely the nonlinear character of the equations. Accordingly, the accuracy of the solution is dependent on the number of harmonics used as well as the level of unsteadiness in the flow. Further work demonstrated thoroughly the increased efficiency and excellent accuracy provided by frequency-domain methods in comparison with typical time-accurate approaches [43, 51, 53, 63]. In addition to the reduced number of time instances, the computation of the initial transient flow behavior that is usually necessary in a typical time-accurate framework is discarded due to the direct convergence of frequency-domain methods to the final periodic solution.

Later, the HB method was extended to a multi-frequential formulation where the employed frequencies need not be multiples of the fundamental frequency [18, 26]. An adaptive version of the method was developed by Maple [50], in which the number of harmonics vary throughout the domain proportionally to the local level of unsteadiness of the flow. As flows often contain both regions of low and high unsteadiness, this advancement allows for greater computational savings in regions where only a few harmonics are necessary. Mosahebi extended the method for nonlinear problems and developed an adapted approach for two-dimensional viscous flows using the NLFD approach [57], and demonstrated its application on deformable grids [59]. Originally developed using explicit solvers such as Runge-Kutta methods for the advancement of pseudotime, the convergence speed of frequency-domain methods was improved by the use of implicit algorithms such as block-Jacobi for the TS method [75] and Lower-Upper Symmetric Gauss-Seidel (LU-SGS) for the adaptive NLFD method [58].

### 1.3 Aeroelastic Solutions using Frequency-Domain Methods

As emphasized earlier, engineers grant high importance to aeroelastic problems such as flutter or LCO due to their potentially dangerous character. Those phenomena are characterized by two parameters: the air speed and the oscillation frequency. For flutter, they are referred to as the flutter speed and flutter frequency, which define the flutter point. The V-g [15] and p-k [29] methods are commonly used by engineers in order to compute iteratively both parameters and thus assess the conditions under which the structure begins to flutter. However, these methods cannot model LCO since they are unable to model aerodynamic nonlinearities and, as a consequence, cannot predict the existence of detrimental nonlinearities.

Due to the failure of the V-g and p-k methods in that respect, researchers first turned to time-marching methods for nonlinear aeroelastic computations; time-accurate techniques in which the structural equations of motion are integrated in time at each time step based on the aerodynamic loads, such that the body displacement can be updated for the computation of the flow solution at the next time instance. For a fixed air speed, the associated oscillation frequency thus automatically sets to the appropriate value and it can thereafter be observed whether the selected flow conditions lead to damped, neutral, or diverging oscillations. By iteration, the flutter speed and frequency can be assessed when neutral oscillations are observed. This method was employed with efficient convergence accelerators by Alonso and Jameson [2] for a two-dimensional airfoil undergoing pitching and plunging motions, and showed great accuracy in the prediction of the flutter boundary. Nevertheless, this technique requires the expensive computation of the solution during the decay

of the initial transients, as well as a high number of time instances in order for the solution to be of sufficient accuracy.

Fortunately, in many practical cases, flutter and LCO are characterized by periodic oscillations of constant amplitude. The previously described frequency-domain methods are therefore attractive approaches to solve aeroelastic problems in an efficient fashion. However, an additional difficulty arises when using frequency-domain methods: whereas time-accurate approaches naturally allow for the oscillation frequency to set automatically and therefore leave the air speed as the only independent variable for given flow conditions, frequency-domain methods require the frequency to be determined *a priori* in order to perform the appropriate trigonometric interpolation. Therefore, while time-accurate methods may iterate over the air speed only, frequency-domain methods must iterate over both the air speed and the frequency until a pair leads to neutral oscillations, regarded as LCO.

Thomas et al. [80, 81] first showed the possibility to employ the HB technique in order to compute LCO of pitching and plunging airfoils using a linear structural solver. By imposing the amplitude of one degree of freedom as an independent displacement (in this case, the angle of attack), the HB/LCO method applies a Newton-Raphson technique to a system of four equations to compute the real and imaginary parts of the plunging motion as well as the air speed and frequency that satisfy the conditions for neutral stability. This method not only allows for the computation of flutter when the imposed independent displacement is small such that nonlinearities are negligible, but also provides a method to assess nonlinear LCO of higher amplitudes by forcing the independent displacement to higher values.

The same authors later extended the HB/LCO method to three dimensions [16, 79, 83], in which the structural mode shapes of the body are assessed beforehand, and the two-dimensional technique is repeated using the first mode shape as the independent displacement. Then, using  $N_{ms}$  mode shapes for the displacement basis of the body,  $2(N_{ms} - 1)$  equations are solved for the real and imaginary parts of every modal displacement numbered from 2 to  $N_{ms}$ , and 2 additional equations are solved for the LCO frequency and air speed, for a total of  $2N_{ms}$  equations in the global system. The structural modal displacements are represented using one harmonic only, whereas the flow solution can be represented using more modes. This approach leads to good results, but require the evaluation of numerous finite-difference derivatives to build the  $2N_{ms} \times 2N_{ms}$  Jacobian matrix of the Newton-Raphson method at each iteration. Moreover, the computed values of frequency and airspeed exhibit convergence with respect to  $N_{ms}$ , the number of retained structural mode shapes. Therefore, this method imposes a convergence study over the employed number of mode shapes, which may become tedious when added to the traditional spatial and temporal accuracy studies.

Kachra and Nadarajah [43] first coupled an NLFD flow solver to a fully nonlinear two-degree-of-freedom structural solver using multiple structural harmonics, in order to assess the aeroelastic behavior of a two-dimensional airfoil. The structural displacements and aerodynamic loads were decomposed via Fourier transforms between each period of the flow solution, and a  $2 \times 2$  system of structural equations was obtained and solved for each harmonic separately. The issue of the *a priori* unknown frequency was however left unresolved since the oscillation frequency was

fixed to the value obtained from a previously executed time-accurate solution of the same test case, therefore defeating the purpose of using the NLFD technique. Ekici and Hall [19] afterwards revisited the HB/LCO technique using a nonlinear one-degree-of-freedom structural model, in which multiple harmonics are considered in the structural displacements and the frequency of oscillation is determined iteratively for a fixed air speed. They also developed a one-shot frequency-determination approach, based on the fully coupled fluid-structure system, which proved more efficient than the Newton-Raphson technique developed by Thomas et al. [81]. Nevertheless, the cost associated with the method was only assessed for a one-degree-of-freedom structural solver, and extending it to more degrees of freedom would require new computational cost analyses to ensure its efficiency.

Mundis and Mavriplis [61] later performed two-dimensional quasi-periodic aeroelastic computations using a hybrid Backward Differentiation Formula/Time-Spectral (BDFTS) solver along with a Generalized Minimal Residual (GMRES) method, in which they assumed the flutter frequency to be known beforehand and applied a fluid-structure coupling between each oscillation period. They later developed a fully-coupled TS aeroelastic solver, in which the structural equations of motion are solved along with the flow at each iteration by the mean of a single system of equations [62]. However, this method still assumes an *a priori* known frequency since a forced pitching motion is prescribed to the airfoil, while still allowing it to move aeroelastically in pitch and plunge directions. Although this allows for a somewhat free motion of the airfoil, it is still supposed that the airfoil is to oscillate at the same frequency as the forced pitching motion.

The HB method was afterwards adapted for simulating aeroelastic flows in turbomachinery using a predefined frequency [34] or set of frequencies [73, 74] that were deduced from the blade passage pattern. In both cases, however, since the frequencies are fixed, the aeroelastic analysis is unidirectional: the motion is prescribed based on the structural mode shapes with a given amplitude, and combined structural and aerodynamic damping is assessed afterwards and employed in an aeroelastic stability analysis. The same is true for the work of Rahmati et al. [68]. In the case of flutter and LCO computation, this approach cannot be employed, since the vibration frequency is *a priori* unknown, and thus the requirement for a novel approach is justified.

#### 1.4 Dynamic Mesh Deformation for Aeroelastic Solutions

By definition, aeroelastic solutions do not only involve unsteady flows but also moving boundaries, induced by body motion. In two dimensions, such movements may often be modelled by a rigid translation and rotation of the entire computational grid along with the body, since the assumption that the body is undeformable is generally acceptable. For instance, two-dimensional aeroelastic simulations often involve two-degree-of-freedom rigid airfoils that can freely pitch and plunge about a specific pivot point, but of which shape remains constant in time. However, this is not the case for most three-dimensional aeroelastic simulations, since three-dimensional bodies are often prone to more complex deformation, such as bending and torsion, that cannot be modelled by rigid translation and rotation only, but rather require a deformable grid approach.

Mesh deformation has been extensively studied in the past and, as a consequence, various algorithms have been developed in that respect. Spring analogy [6, 9, 14, 23, 24, 92], Laplacian smoothing [7, 20, 49], linear elasticity [5, 42, 76, 86], and Delaunay mapping [48] techniques are as many famous mesh deformation methods. Whereas spring analogy and linear-elasticity methods may be computationally expensive, Laplacian smoothing and Delaunay mapping are best suited for unstructured grids, yet structured meshes are employed in the present research. Interpolation-based techniques, such as algebraic mesh movement [10, 11, 21] and Radial Basis Functions (RBF) [13] deformation schemes, represent a possible alternative. In this work, the RBF approach is employed due to its ability to model large displacements in a robust fashion that is suitable for structured meshes [22]. It consists in a meshless approach as it acts only on grid vertices regardless of their connectivity, which makes it particularly attractive due to its simplicity of implementation. Although the original method was computationally expensive, many have contributed to improve its efficiency [35, 55, 56, 67, 71, 72]. Despite the author's knowledge of these improvements, the method used in this work is the original approach, modified by Walther [88] for parallel computations. Since the objective of this research is not to compare the efficiency of the RBF deformation technique with other schemes, the usage of the full-cost RBF is justified.

One direct effect of dynamic grid movement is the necessity to modify the evaluation of the fluxes at the grid cell interfaces in finite-volume schemes. A common approach is to employ the Arbitrary Lagrangian-Eulerian (ALE) formulation [33] of the governing equations to solve modified flow equations in a moving referential,

which requires the evaluation of the velocity of all cell faces at all time instances of the solution. The method is applied very easily for rigid grid movement, as the velocity of each mesh point is only based on the analytical derivative of the grid position with respect to time, provided that a known function describes the movement of the grid. However, the computation of mesh velocities may prove difficult for deforming grids, since there might not exist a simple function describing the movement of all grid points, therefore not allowing the necessary time differentiation. For this type of grid movement, time-accurate solvers have the possibility of computing mesh velocities through the use of a backward finite-difference method at each time instance for all grid points. However, frequency-domain techniques use a very low number of time instances per period, thus preventing the use of a finite-difference approach due to the enormous error that would be introduced by the large time steps. Since the movement of the grid is assumed periodic in such techniques, one possible option is to use the same differentiation operator for the computation of the mesh velocities as that employed for the differentiation of the flow variables in the frequency-domain solver [17, 60]. Nevertheless, the accuracy of this method depends on the ability of the position of each grid point to be accurately represented by Fourier series using a finite number of harmonics. As a consequence, the accuracy of this approach depends on the number of modes that is chosen to represent the solution, which is not desirable. Following that rationale, frequency-domain solvers would benefit from a mesh-velocity computation technique of which accuracy does not depend on the number of harmonics.

## 1.5 Research Objectives

The main objective of this research is to further extend the use of the NLFD method to three-dimensional aeroelastic solutions, such that it allows for the efficient assessment of flutter and limit cycle oscillation conditions. This objective is to be achieved through the steps listed below.

- Develop an alternative approach to that of Mosahebi and Nadarajah [60] and Dufour et al. [17] for the computation of the mesh velocities for the NLFD method using deforming grids, of which accuracy is not dependent on the employed number of harmonics.
- Develop a novel and efficient method for the determination of flutter and LCO conditions (air speed and frequency) using a three-dimensional NLFD aeroelastic solver.
- Validate the developed methods using experimental results and previously published numerical data.

The results obtained from the proposed methods are to be discussed in detail, and their limitations are to be addressed in the conclusions of this document.

## CHAPTER 2

### Flow Solver

The following sections present the employed formulation for the flow field along with its numerical discretization. The two-dimensional flow solver employed in section 6.1 is the single-block single-processor code used in the work of Kachra and Nadarajah [43]. The three-dimensional implementation is based on the work of Nadarajah et al. [65] and Nadarajah and Jameson [64], and consists in a parallel multiblock finite-volume NLFD flow solver.

#### 2.1 Governing Equations

This research employs the Euler equations, therefore assuming the flow as inviscid. The three-dimensional Euler equations in integral form for an element  $\mathcal{V}$  of volume  $\Omega$ , delimited by boundary  $\partial\mathcal{V}$  of which surface element is  $dS$ , are

$$\frac{\partial}{\partial t} \int_{\mathcal{V}} \mathbf{w} d\Omega + \oint_{\partial\mathcal{V}} \mathbf{F}_c dS = 0, \quad (2.1)$$

where  $\mathbf{w}$  is the state vector and  $\mathbf{F}_c$  is the convective flux vector. On moving meshes, they are defined respectively in ALE form as

$$\mathbf{w} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix}, \quad \mathbf{F}_c = \begin{pmatrix} \rho V_r \\ \rho u V_r + n_x p \\ \rho v V_r + n_y p \\ \rho w V_r + n_z p \\ \rho H V_r + p V_t \end{pmatrix},$$

where  $\rho$ ,  $u$ ,  $v$ ,  $w$ ,  $E$ ,  $H$  and  $p$  represent the fluid density, cartesian velocity components, total energy per unit mass, total enthalpy per unit mass, and pressure, respectively. Pressure is evaluated through the equation of state for ideal gases,

$$p = (\gamma - 1)\rho \left( E - \frac{u^2 + v^2 + w^2}{2} \right),$$

$\gamma$  being the specific heat ratio of the fluid, and total enthalpy is expressed as

$$H = E + \frac{p}{\rho}.$$

Also,  $n_x$ ,  $n_y$  and  $n_z$  are the components of the outward facing unit normal vector  $\mathbf{n}$  of surface  $\partial\mathcal{V}$ .  $V_r$  is the contravariant velocity of the flow relative to the motion of the boundary, and  $V_t$  is the velocity of the control volume boundary. Those velocities can be expressed as [8]

$$\begin{aligned} V &= \mathbf{n} \cdot \mathbf{V} = n_x u + n_y v + n_z w, \\ V_t &= \mathbf{n} \cdot \mathbf{V}_g = n_x \frac{dx_g}{dt} + n_y \frac{dy_g}{dt} + n_z \frac{dz_g}{dt}, \\ V_r &= \mathbf{n} \cdot (\mathbf{V} - \mathbf{V}_g) = V - V_t, \end{aligned}$$

where  $\mathbf{V}$  is the flow velocity vector and  $\mathbf{V}_g$  is the control volume boundary velocity vector.

## 2.2 Spatial Discretization

The Euler equations are spatially discretized using a second-order finite-volume method. The flow field is divided into the desired number of control volumes (cells), and the solution is evaluated at the cell centroids. Assuming a three-dimensional structured hexahedral grid with  $i$ ,  $j$  and  $k$  indices for the three computational-domain directions, equation (2.1) is rewritten in semi-discrete form for an arbitrary cell of indices  $(i, j, k)$  as

$$\frac{\partial (\Omega_{i,j,k} \mathbf{w}_{i,j,k})}{\partial t} + \mathbf{R}(\mathbf{w}_{i,j,k}) = 0 \quad \text{in } \mathcal{V}_{i,j,k}, \quad (2.2)$$

where  $\Omega_{i,j,k}$  is the cell volume and  $\mathbf{R}(\mathbf{w}_{i,j,k})$  is the discrete residual vector.  $\mathbf{R}(\mathbf{w}_{i,j,k})$  is evaluated through the summation of the fluxes through all the faces of the control volume. Defining the flux vectors to be pointing away from the control volume and collecting terms from each face leads to

$$\mathbf{R}(\mathbf{w}_{i,j,k}) = \sum_{m=1}^{N_f} (\mathbf{F}_c - \mathbf{F}_d)_m S_m, \quad (2.3)$$

where index  $m$  denotes each of the faces of  $\Omega_{i,j,k}$  from 1 to its total number of faces  $N_f$  (equal to 6 in this work),  $S_m$  is the surface area of the  $m^{\text{th}}$  face of the control volume, and  $\mathbf{F}_d$  represents the vector of dissipative fluxes. Artificial dissipation is added in order to prevent solution oscillations near discontinuities such as shocks, as well as to avoid odd-even decoupling of the solution. In this work, both the Jameson-Schmidt-Turkel [41] (JST) and Convective Upwind Split Pressure [39] (CUSP) schemes are

employed for the evaluation of artificial dissipation. The fluxes at each face of the control volume are evaluated through an average of the fluxes at the center of adjacent cells. For example, the fluxes at the face between computational cells  $\Omega_{i,j,k}$  and  $\Omega_{i+1,j,k}$  are computed from the following relation,

$$((\mathbf{F}_c - \mathbf{F}_d) S)_{i+\frac{1}{2},j,k} = \frac{1}{2} (\mathbf{F}_{c_{i,j,k}} - \mathbf{F}_{d_{i,j,k}} + \mathbf{F}_{c_{i+1,j,k}} - \mathbf{F}_{d_{i+1,j,k}}) S_{i+\frac{1}{2},j,k} , \quad (2.4)$$

where indices  $(i+\frac{1}{2}, j, k)$  indicate that the value is computed at the interface between cells  $(i, j, k)$  and  $(i+1, j, k)$ .

### 2.3 Temporal Discretization

The next subsections describe the approach for discretizing the governing equations in physical time, as well as for converging the discretized equations in pseudo-time.

#### 2.3.1 Nonlinear Frequency Domain Method

As mentioned previously, the present work employs an NLFD solver, and therefore the temporal discretization is the same as that demonstrated by McMullen et al. [52]. In order to solve the unsteady flow field, the semi-discrete form of the governing equations presented in equation (2.2) is discretized with respect to time in two steps: first, it is discretized with respect to real (or physical) time; second, it is discretized with respect to pseudotime. The discretization of the physical time accounts for the flow solution unsteadiness, while that of the pseudotime is needed for iterative convergence to a periodic steady-state solution.

Rewriting equation (2.2) without cell indices for an arbitrary control volume as

$$\frac{\partial(\Omega\mathbf{w})}{\partial t} + \mathbf{R}(\mathbf{w}) = 0 \quad \text{in } \mathcal{V}, \quad (2.5)$$

the physical time discretization is performed by substituting the volume-modified state vector  $\bar{\mathbf{w}} = \Omega\mathbf{w}$  and the residual vector  $\mathbf{R}(\mathbf{w}(t))$  by their corresponding discrete Fourier representations using a finite number of harmonics or modes. This is accomplished through the following equations,

$$\bar{\mathbf{w}}(t) = \sum_{k=-N}^N \hat{\mathbf{w}}_k e^{i\frac{2\pi k}{T}t}, \quad (2.6)$$

$$\mathbf{R}(\mathbf{w}(t)) = \sum_{k=-N}^N \hat{\mathbf{R}}_k e^{i\frac{2\pi k}{T}t}, \quad (2.7)$$

where  $i = \sqrt{-1}$ ,  $N$  is the number of modes employed in the Discrete Fourier Transform (DFT),  $k$  is the wave number,  $T$  is the time period, and  $\hat{\mathbf{w}}_k$  and  $\hat{\mathbf{R}}_k$  are the  $k^{\text{th}}$  Fourier coefficients of the state and residual vectors, respectively. For computations on deformable grids, it is essential to include the cell volume  $\Omega$  inside the time derivative in order to account for the variation of the volume in time. To perform the appropriate transforms, the solution and residual need to be sampled at  $N_{ts}$  equally spaced time instances, such that  $N_{ts} = 2N + 1$ . The Fourier coefficients are therefore readily found by

$$\hat{\mathbf{w}}_k = \sum_{n=0}^{2N} \Omega(t_n) \mathbf{w}(t_n) e^{i\frac{2\pi k}{T}t_n}, \quad \text{for } -N \leq k \leq N, \quad (2.8)$$

$$\hat{\mathbf{R}}_k = \sum_{n=0}^{2N} \mathbf{R}(\mathbf{w}(t_n)) e^{i\frac{2\pi k}{T}t_n}, \quad \text{for } -N \leq k \leq N, \quad (2.9)$$

where subscript  $n$  denotes values computed at the  $n^{\text{th}}$  time sample  $t_n$ , such that  $t_n = \frac{n}{2N+1}T$ . It is important to note that the residuals are evaluated in the time

domain at all time instances, and only then transferred into the frequency-domain using a DFT. The Fourier representation of the vectors are then substituted into equation (2.5), thus yielding

$$\frac{d}{dt} \left( \sum_{k=-N}^N \hat{\mathbf{w}}_k e^{i\frac{2\pi k}{T}t} \right) + \sum_{k=-N}^N \hat{\mathbf{R}}_k e^{i\frac{2\pi k}{T}t} = 0. \quad (2.10)$$

Calculating the temporal derivative of the leftmost term of the equation leads to

$$\sum_{k=-N}^N i\frac{2\pi k}{T} \hat{\mathbf{w}}_k e^{i\frac{2\pi k}{T}t} + \sum_{k=-N}^N \hat{\mathbf{R}}_k e^{i\frac{2\pi k}{T}t} = 0. \quad (2.11)$$

Due to the orthogonality of the Fourier basis functions, the above equation can be rewritten as a system of  $2N + 1$  independent equations, each of which is associated with a wave number  $k$  as

$$i\frac{2\pi k}{T} \hat{\mathbf{w}}_k + \hat{\mathbf{R}}_k = 0, \quad \text{for } -N \leq k \leq N. \quad (2.12)$$

As there is no evident representation of  $\hat{\mathbf{R}}_k$  as a function of  $\hat{\mathbf{w}}_k$ , equation (2.12) cannot be solved easily and a numerical method must be employed. An unsteady residual  $\hat{\mathbf{R}}_k^*$  is consequently defined as the residual of equation (2.12), such that

$$i\frac{2\pi k}{T} \hat{\mathbf{w}}_k + \hat{\mathbf{R}}_k = \hat{\mathbf{R}}_k^*, \quad \text{for } -N \leq k \leq N, \quad (2.13)$$

and  $\hat{\mathbf{R}}_k^*$  is driven to zero using a pseudotime-marching approach expressed as

$$\frac{d\hat{\mathbf{w}}_k}{dt^*} + \hat{\mathbf{R}}_k^* = 0, \quad \text{for } -N \leq k \leq N, \quad (2.14)$$

where  $t^*$  represents pseudotime. At convergence, as  $\hat{\mathbf{R}}_k^* = 0$  and equation (2.14) is satisfied, equation (2.12) is also satisfied and the final periodic flow solution is

obtained. Equation (2.14) can be solved iteratively using any time-stepping scheme, such as Euler or Runge-Kutta methods, in which the flow solution is initialized using the free-stream values. Between each pseudotime step, the solution is transformed back into the time domain using an Inverse Discrete Fourier Transform (IDFT), and the time-domain solution is obtained at each time instance  $t_n$  through a division by the cell volume, as follows,

$$\mathbf{w}(t_n) = \frac{\bar{\mathbf{w}}(t_n)}{\Omega(t_n)}, \quad \text{for } n = 0, 1, 2, \dots, 2N.$$

The boundary conditions are also evaluated in the time domain before another pseudotime cycle is accomplished. It is important to note that, in practice, since the solution  $\mathbf{w}$  and the residual  $\mathbf{R}(\mathbf{w})$  are always real-valued, the Fourier coefficients associated with their negative wave numbers are only the complex conjugates of the coefficients associated with their corresponding positive wave numbers. Therefore, the equations associated with negative wave numbers can be saved from computation.

### 2.3.2 Pseudotime-Stepping Scheme

The NLFD method is similar to the computation of a steady-state solution in pseudotime for each wave number  $k$ . In this work, the pseudotime discretization is performed using a modified Runge-Kutta scheme, which is an explicit multistage method that was first implemented for the solution of the Euler equations by Jameson [41]. In order to update the solution from pseudotime instance  $m$  to  $m + 1$  using an  $M$ -stage scheme, the procedure is as follows [8] for each wave number  $k$ :

$$\begin{aligned}
\hat{\mathbf{w}}_k^{(0)} &= \hat{\mathbf{w}}_k^m \\
\hat{\mathbf{w}}_k^{(1)} &= \hat{\mathbf{w}}_k^{(0)} - \alpha_1 \Delta t^* \hat{\mathbf{R}}_k^{*(0)} \\
\hat{\mathbf{w}}_k^{(2)} &= \hat{\mathbf{w}}_k^{(0)} - \alpha_2 \Delta t^* \hat{\mathbf{R}}_k^{*(1)} \\
&\vdots \\
\hat{\mathbf{w}}_k^{(p)} &= \hat{\mathbf{w}}_k^{(0)} - \alpha_p \Delta t^* \hat{\mathbf{R}}_k^{*(p-1)} && \text{for } p = 1, 2, \dots, M \\
&\vdots \\
\hat{\mathbf{w}}_k^{m+1} &= \hat{\mathbf{w}}_k^{(M)}, \tag{2.15}
\end{aligned}$$

where  $\hat{\mathbf{w}}_k^m$  and  $\hat{\mathbf{w}}_k^{m+1}$  are the solution vectors at pseudotime steps  $m$  and  $m + 1$  respectively,  $\hat{\mathbf{w}}_k^{(p)}$  and  $\hat{\mathbf{R}}_k^{*(p)}$  are the intermediary solution and unsteady residual vectors at stage  $p$ , and  $\alpha_p$  are the stage coefficients. The main difference between this method and the classical Runge-Kutta technique is that the solution only needs to be known at the beginning of the time interval ( $\hat{\mathbf{w}}_k^{(0)}$ ), and only the residual from the previous stage ( $\hat{\mathbf{R}}_k^{*(p-1)}$ ) is required, so that memory requirements are reduced. In order to increase the stability region of the scheme, blending coefficients  $\beta_p$  are included at each stage to control the amount of artificial dissipation. Combining equations (2.3), (2.7) and (2.13), and introducing parameter  $\beta_p$ , the unsteady residual at each stage is expressed as

$$\hat{\mathbf{R}}_k^{*(p)} = i \frac{2\pi k}{T} \hat{\mathbf{w}}_k^{(p)} + \hat{\mathbf{R}}_{c,k}^{(p)} - \left( \beta_p \hat{\mathbf{R}}_{d,k}^{(p)} + (1 - \beta_p) \hat{\mathbf{R}}_{d,k}^{(p-1)} \right), \tag{2.16}$$

where  $\hat{\mathbf{R}}_{c,k}^{(p)}$  and  $\hat{\mathbf{R}}_{d,k}^{(p)}$  are the respective contributions of the convective and dissipative fluxes to the Fourier coefficients of the residual vector associated with wave number  $k$

at stage  $p$ . It can therefore be seen that the dissipation at stage  $p$  is a combination of the computed dissipative fluxes at stages  $p$  and  $p-1$ . For simplification, the blending is applied to  $\mathbf{F}_d^{(p)}$  and  $\mathbf{F}_d^{(p-1)}$  before the application of the DFT to the residual vector, which is equivalent to the above formulation. This is called a hybrid scheme since the blending coefficients  $\beta_p$  allow the dissipative fluxes not to be calculated at each stage in order to reduce the associated computational cost. The parameters for  $\alpha_p$  and  $\beta_p$  are chosen as listed in Table 2–1, in order to be optimized for multigrid computations [8].

Table 2–1: Stage and Blending Coefficients of the Hybrid Multistage Pseudotime-Stepping Scheme

Stage	$\alpha$	$\beta$
1	0.2500	1.0000
2	0.1667	0.0000
3	0.3750	0.5600
4	0.5000	0.0000
5	1.0000	0.4400

Between each Runge-Kutta stage, the updated solution is transferred back into the time domain using an IDFT, and the wall and far-field boundary conditions are applied. Figure 2–1 schematizes one stage of the employed multistage technique in the NLFD framework. In this work, all DFT and IDFT calculations are performed using the FFTW library [3], which is a Fast Fourier Transform (FFT) algorithm scaling  $N_{ts} \log(N_{ts})$  in terms of computational effort, rather than  $N_{ts}^2$  as would a non optimized DFT, where  $N_{ts}$  is the number of time instances used in the transform.

In order to further accelerate the convergence to the steady-state periodic solution, a W-cycle multigrid strategy [36, 37] is employed in conjunction with a local

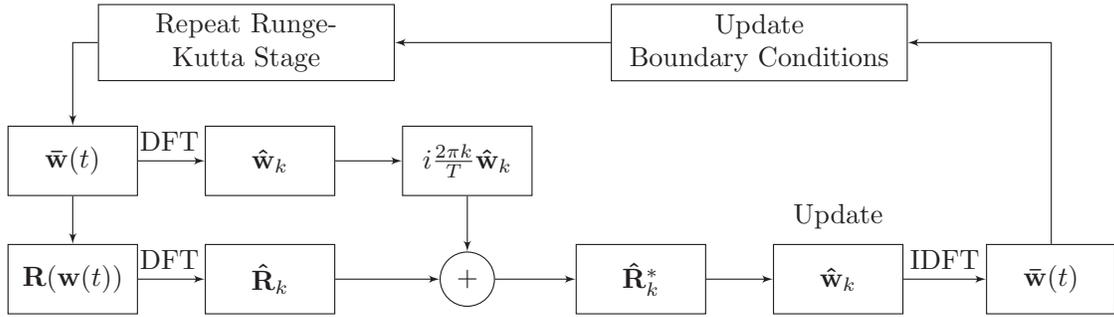


Figure 2–1: Simplified Dataflow Diagram of the Nonlinear Frequency Domain Method Using a Modified Runge-Kutta Technique

time-stepping technique and an implicit residual averaging method [40]. Characteristic boundary conditions using one-dimensional Riemann invariants are used at the far-field boundary [40] whereas, at the body surface, the normal flow velocity is required to be zero and the pressure is determined through a linear extrapolation of the flow field pressure from the adjacent cells.

## CHAPTER 3 Structural Solver

The following sections present the structural model employed for aeroelastic computations along with its numerical discretization. A plate bending finite-element method is employed for the spatial discretization, whereas the temporal discretization is carried out in the frequency domain in a similar fashion as the flow solver.

### 3.1 Governing Equations

Since this work is mainly focused on assessing the aeroelastic behavior of aircraft wings, which are commonly sufficiently thin to be represented as plates, the structural solver derivation is based on the classical small-deflection theory of thin plates and supposes that all studied bodies satisfy the following assumptions [77]:

- the body geometry allows for a thin-plate representation, that is, its thickness is small compared to its other dimensions;
- the constituting material of the plate is isotropic, homogeneous and linearly elastic;
- the middle plane of the plate remains unstrained during bending.

Since the transverse aerodynamic loads (perpendicular to the wing planform) are in general much higher than the streamwise loads, only bending was considered, whereas membrane deformation was neglected. The governing partial differential equation for the transverse motion of bending plates can be expressed as [77]:

$$\frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} + \frac{\bar{m}}{D} \frac{\partial^2 w}{\partial t^2} = \frac{p_z}{D}, \quad (3.1)$$

where  $w(x, y, t)$  is the local instantaneous deflection of the plate,  $p_z(x, y, t)$  is the local instantaneous load per unit area in the transverse direction  $z$  (perpendicular to the plate),  $\bar{m}$  is the plate mass per unit area, and  $D$  is defined as

$$D = \frac{Eh^3}{12(1 - \nu^2)},$$

where  $E$  and  $\nu$  are respectively Young's modulus and Poisson's ratio of the material, and  $h$  is the thickness of the plate. For the sake of brevity and because the present work is focused primarily on the computational fluid dynamics discipline, the procedure for deriving the finite element formulation of the above equation will not be presented here. The interested reader is referred to the work of Szilard [77] for a comprehensive review of plate bending analysis. Therefore, since a finite element method for plate bending analysis is employed in this work, only the relevant equations are provided herein.

### 3.2 Spatial Discretization

The procedure detailed in this section is based primarily on references [69] and [77]. The governing equations for a linear undamped dynamic structural system can be expressed in discretized matrix form as

$$[\mathbf{M}] \ddot{\mathbf{Q}} + [\mathbf{K}] \mathbf{Q} = \mathbf{P}, \quad (3.2)$$

where  $[\mathbf{M}]$  and  $[\mathbf{K}]$  are the mass and stiffness matrices,  $\mathbf{Q} = \mathbf{Q}(t)$  is the discrete displacement vector,  $\ddot{\mathbf{Q}} = \ddot{\mathbf{Q}}(t)$  is its second time derivative, and  $\mathbf{P} = \mathbf{P}(t)$  is the

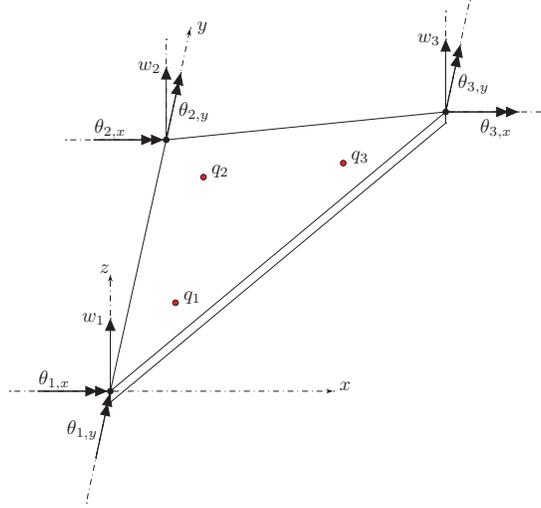


Figure 3–1: Arbitrary Structural Element and Example of Three-Point Quadrature Points ( $q_i$ )

exciting force vector. Note that the upper-case notation is adopted for vectors and matrices evaluated in global coordinates, while lower-case notation is used for values computed in local (elemental) coordinates. In order to define these matrices and vectors precisely, the structure is first discretized using the nonconforming straight-sided triangular element developed by Tocher [85], which has nine degrees of freedom: one translation and two rotations per node. An arbitrary element is displayed in Figure 3–1 along with its degrees of freedom and examples of quadrature points that can be employed for the integration of the mass and stiffness matrices, described subsequently in equations (3.6) and (3.7).

According to the formulation of the triangular elements, formula (3.2) thus represents a system of  $N_{\text{DOF}}$  equations, where  $N_{\text{DOF}}$  is the total number of degrees of freedom of the discretized structure and is equal to nine times the number of

nodes,  $N_{\text{nodes}}$ . Now considering an arbitrary triangular element, it is assumed that the solution  $w(x, y, t)$  over the element at time  $t$  can be expressed by a nine-term polynomial as

$$w(x, y, t) = \mathbf{u}^T \boldsymbol{\alpha}(t), \quad (3.3)$$

with

$$\mathbf{u}^T = \left\{ 1 \quad x \quad y \quad x^2 \quad xy \quad y^2 \quad x^3 \quad (x^2y + xy^2) \quad y^3 \right\},$$

$$\boldsymbol{\alpha}(t)^T = \left\{ \alpha_1(t) \quad \alpha_2(t) \quad \alpha_3(t) \quad \alpha_4(t) \quad \alpha_5(t) \quad \alpha_6(t) \quad \alpha_7(t) \quad \alpha_8(t) \quad \alpha_9(t) \right\},$$

where  $\mathbf{u}$  is called the shape function vector and  $\boldsymbol{\alpha}(t)$  is the vector of polynomial coefficients. Note that variable  $t$  is omitted onward in order to alleviate the upcoming expressions, although the presented derivation retains its time-dependency. It can be seen that a complete third-order polynomial cannot be obtained since it would require ten terms, but only nine are available due to the number of degrees of freedom; therefore, two terms are combined in forming the eighth term. This causes the element to lack invariance, and thus the solution to depend on the local coordinates orientation. Nevertheless, despite its nonconformity and its lack of invariance, this element was chosen due of its simplicity of implementation and because it has been demonstrated to provide reasonably good results and has been widely used in the past [77]. Note that cartesian coordinates  $x$  and  $y$ , employed in the definition of the shape functions, are computed in a local coordinate system where the  $x$  and  $y$  axes are in the plane of the plate element, while the local  $z$  axis is perpendicular to the element. Considering the transverse displacement  $w$  and the rotations  $\theta_x$  and

$\theta_y$  about the  $x$  and  $y$  axes at each node as the degrees of freedom, the elemental displacement vector is expressed as

$$\mathbf{q}_{(e)} = \begin{Bmatrix} w_1 \\ \theta_{x,1} \\ \theta_{y,1} \\ w_2 \\ \theta_{x,2} \\ \theta_{y,2} \\ w_3 \\ \theta_{x,3} \\ \theta_{y,3} \end{Bmatrix} = \begin{Bmatrix} w_1 \\ \frac{\partial w_1}{\partial y} \\ -\frac{\partial w_1}{\partial x} \\ w_2 \\ \frac{\partial w_2}{\partial y} \\ -\frac{\partial w_2}{\partial x} \\ w_3 \\ \frac{\partial w_3}{\partial y} \\ -\frac{\partial w_3}{\partial x} \end{Bmatrix},$$

where subscript numbers 1, 2 and 3 denote the elemental node at which the value is computed, and the subscript ( $e$ ) signifies that the vector is evaluated for the noted element only, as a subset of the displacement vector of the entire model introduced in equation (3.2). Proceeding with the common method, detailed in [77], a matrix  $[\mathbf{A}]$  can be defined to relate the local displacement vector  $\mathbf{q}_{(e)}$  to the polynomial coefficient vector  $\boldsymbol{\alpha}$ , such that

$$\mathbf{q}_{(e)} = [\mathbf{A}]\boldsymbol{\alpha}. \quad (3.4)$$

From the strain-displacement relations, defined as follows,

$$\boldsymbol{\epsilon} = \begin{Bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \gamma_{xy} \end{Bmatrix} = -z \begin{Bmatrix} \frac{\partial w}{\partial x^2} \\ \frac{\partial w}{\partial y^2} \\ 2 \frac{\partial w}{\partial y \partial x} \end{Bmatrix},$$

matrix  $[\mathbf{B}]$  can be defined such that

$$\boldsymbol{\epsilon} = [\mathbf{B}] \boldsymbol{\alpha}, \quad (3.5)$$

where  $\boldsymbol{\epsilon}$  is the strain vector. Then, the stiffness matrix for an arbitrary element of volume  $\Omega_{(e)}$  can be obtained from

$$[\mathbf{k}_{(e)}] = ([\mathbf{A}]^{-1})^T \left( \int_{\Omega_{(e)}} [\mathbf{B}]^T [\mathbf{D}] [\mathbf{B}] d\Omega \right) [\mathbf{A}]^{-1}, \quad (3.6)$$

where  $[\mathbf{D}]$  is obtained from the stress-strain relations and is expressed as

$$[\mathbf{D}] = \frac{E}{(1 - \nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1 - \nu}{2} \end{bmatrix}.$$

The same procedure is employed for orthotropic plates, except for the evaluation of  $[\mathbf{D}]$ , which is instead calculated as

$$[\mathbf{D}] = \frac{1}{(1 - \nu_{xy}\nu_{yx})} \begin{bmatrix} E_{xx} & \nu_{yx}E_{xx} & 0 \\ \nu_{xy}E_{yy} & E_{yy} & 0 \\ 0 & 0 & G_{xy}(1 - \nu_{xy}\nu_{yx}) \end{bmatrix},$$

where subscripts account for the directionality of the material properties, and  $G_{xy}$  is the shear modulus of the material. In addition, the relation  $E_{xx}\nu_{yx} = E_{yy}\nu_{xy}$  is

assumed to be always satisfied. For both isotropic and orthotropic materials, the elemental mass matrix is obtained from

$$[\mathbf{m}_{(e)}] = ([\mathbf{A}]^{-1})^T \left( \int_{\Omega_{(e)}} \rho [\mathbf{u}]^T [\mathbf{u}] d\Omega \right) [\mathbf{A}]^{-1}, \quad (3.7)$$

where  $\rho$  is the density of the plate material. Both the mass and stiffness matrices require the evaluation of an integral over the volume of the element, which can be simplified to an area integral. Both integrals are evaluated numerically using a 12-point Gaussian quadrature providing exact integrals for polynomials of degree 6 or less [12]. All matrices and vectors can then be transferred to the global coordinate system using a simple rotation matrix in order to obtain the elemental stiffness and mass matrices in global coordinates,  $[\mathbf{K}_{(e)}]$  and  $[\mathbf{M}_{(e)}]$ , and the elemental displacement vector in global coordinates,  $\mathbf{Q}_{(e)}$ . Then, collecting all elemental matrices and vectors and sorting them properly according to the node-numbering scheme, global matrices  $[\mathbf{K}]$  and  $[\mathbf{M}]$ , as well as global displacement vector  $\mathbf{Q}$ , can be calculated and used in equation (3.2). The same technique is applied to the force vectors  $\mathbf{p}_{(e)}$ ,  $\mathbf{P}_{(e)}$  and  $\mathbf{P}$ , in the case where the excitation forces are provided in local coordinates. Once the global mass and stiffness matrices are computed, the structural natural frequencies and mode shapes can be determined via a common eigenvalue and eigenvector calculation procedure.

### 3.3 Temporal Discretization

For generality, the equations for the temporal discretization detailed herein are an extension of those demonstrated by Kachra and Nadarajah [43], and yield a fully nonlinear structural solver. Assuming that both the flow solution and the structure

behave in a periodic fashion, and considering that the mass and stiffness matrices are constant in time, the displacement and load vectors of equation (3.2) can be expressed using their DFT representation as

$$\mathbf{Q}(t) = \sum_{k=-N_s}^{N_s} \hat{\mathbf{Q}}_k e^{i\frac{2\pi k}{T}t}, \quad (3.8)$$

$$\mathbf{P}(t) = \sum_{k=-N_s}^{N_s} \hat{\mathbf{P}}_k e^{i\frac{2\pi k}{T}t}, \quad (3.9)$$

where  $N_s$  is the number of harmonics employed in the discretization of the structural equations of motion, and is not necessarily equal to  $N$ , the number of harmonics used for the flow solution. Differentiating the displacement vector twice with respect to time therefore leads to

$$\ddot{\mathbf{Q}}(t) = \sum_{k=-N_s}^{N_s} -\left(\frac{2\pi k}{T}\right)^2 \hat{\mathbf{Q}}_k e^{i\frac{2\pi k}{T}t}. \quad (3.10)$$

Substituting these expressions into equation (3.2) and again taking advantage of the orthogonality of the Fourier basis functions, a system of  $2N_s + 1$  equations is obtained, each associated with a wave number  $k$ :

$$\left( [\mathbf{K}] - \left(\frac{2\pi k}{T}\right)^2 [\mathbf{M}] \right) \hat{\mathbf{Q}}_k = \hat{\mathbf{P}}_k, \quad \text{for } -N_s \leq k \leq N_s. \quad (3.11)$$

Equation (3.11) thus only consists of a linear problem, of which size is equal to the total number of degrees of freedom in the structure ( $N_{\text{DOF}}$ ), and is easily solved either by a direct or iterative method, such that

$$\hat{\mathbf{Q}}_k = \left( [\mathbf{K}] - \left(\frac{2\pi k}{T}\right)^2 [\mathbf{M}] \right)^{-1} \hat{\mathbf{P}}_k, \quad \text{for } -N_s \leq k \leq N_s. \quad (3.12)$$

The displacements at each time instance are then retrieved via an IDFT, and can be employed as desired in the flow solver. However, since the system is at first under-constrained due to possible rigid body translations and rotations, boundary conditions have to be imposed. In all studied cases, the structure (wing) is clamped at its root; the associated boundary conditions require that no displacement nor rotation occur at the nodes located at the root. This is accomplished by removing lines and columns of the corresponding degrees of freedom from matrices  $[\mathbf{K}]$  and  $[\mathbf{M}]$  and vectors  $\mathbf{Q}$  and  $\mathbf{P}$ , and solving equation (3.11) for a reduced set of well-constrained equations.

As explained for a similar method in [19], the nonlinear character of the detailed approach is due to the fact that the load vector is discretized using multiple harmonics, which are coupled in the time domain at each flow solver iteration (see chapter 5). The structural solution is therefore fully coupled to the nonlinear flow solution, and is thus also deemed nonlinear. However, although the temporal discretization was developed for generality in a fully nonlinear fashion, a linear version is employed in this work. Indeed, the employed number of structural harmonics is always  $N_s = 1$ , such that no coupling can occur between the higher harmonics of the flow and the sole mode of the structure, except for the effect of the higher harmonics of the flow on its own first mode, which in turn influences the harmonic of the structure. Nevertheless, this linear structural solver is equivalent to that employed by Thomas et al. [79–81] and Dowell et al. [16], which showed the capability of assessing flutter and LCO.

## CHAPTER 4

### Dynamic Mesh Deformation

As emphasized previously, a dynamic mesh deformation framework must be adopted for three-dimensional aeroelastic computations. This chapter presents the developed approach and the accuracy implications related to the Geometric Conservation Law (GCL).

#### 4.1 Radial Basis Function Framework

A Radial Basis Function (RBF) algorithm is employed for mesh deformation since it provides grids of good quality, while remaining robust and easily implementable. The following subsections describe the procedure for the mesh deformation as well as the computation of the mesh velocities, resulting in a complete dynamic mesh deformation framework.

##### 4.1.1 Mesh Deformation

The RBF deformation method first makes a distinction between RBF points and volume points: RBF points are grid points of which displacements are known *a priori*, that is, before the mesh deformation; volume points are grid points of which displacements are unknown *a priori*, and for which the post-deformation position is desired. Here, displacement refers to the variation of the position of a point from the undeformed mesh to the deformed mesh. In this work, the RBF points are always chosen to be all or a subset of the body surface points; therefore, the position of the body must be known before and after deformation. This does not pose any problem

since, in the case of prescribed body motion, an analytical function describes the position of the body in time, whereas in the case of aeroelastic solutions, the motion of the body is provided by the structural solver. It is important to note that, since the NLFD method requires the evaluation of the fluxes in the time domain, as denoted by equation (2.3), a grid must be generated at all  $2N + 1$  time instances, where  $N$  is the number of harmonics employed in the computation of the flow.

Here, the RBF technique will be detailed for displacements in the  $x$ -direction only, but the procedure is exactly identical for the  $y$ - and  $z$ -directions. The RBF method relies on the assumption that the displacement of any grid point  $p$  may be calculated via a weighted sum of its distance to all RBF points as follows,

$$s_x(\mathbf{x}_p, t) = \sum_{i=1}^{N_{rp}} \alpha_i(t) \phi(\|\mathbf{x}_p - \mathbf{x}_i\|_2), \quad (4.1)$$

where  $s_x(\mathbf{x}_p, t)$  is the  $x$ -direction displacement of point  $p$  during deformation,  $N_{rp}$  is the number of RBF points,  $\alpha_i$  are the interpolating coefficients, and  $\mathbf{x}_p$  and  $\mathbf{x}_i$  are the position vectors of noted point  $p$  and RBF point  $i$  in the undeformed grid, respectively.  $\phi$  is the basis function and depends on  $\|\mathbf{x}_p - \mathbf{x}_i\|_2$ , the distance between points  $\mathbf{x}_p$  and  $\mathbf{x}_i$  in the undeformed grid. Various basis functions are available for this purpose, but this work considers only Wendland's C2 function [89] for two-dimensional cases,

$$\phi(\xi) = \begin{cases} (1 - \xi)^4(4\xi + 1) & \text{if } \xi < 1 \\ 0 & \text{if } \xi \geq 1 \end{cases}, \quad \xi = \frac{\|\mathbf{x}_p - \mathbf{x}_i\|_2}{R}, \quad (4.2)$$

and Wendland's C0 function [89] for three-dimensional cases,

$$\phi(\xi) = \begin{cases} (1 - \xi)^2 & \text{if } \xi < 1 \\ 0 & \text{if } \xi \geq 1 \end{cases}, \quad \xi = \frac{\|\mathbf{x}_p - \mathbf{x}_i\|_2}{R}, \quad (4.3)$$

where, in both cases,  $R$  is a user-defined support radius defining the radial distance from the RBF points within which the remainder of the grid is to be deformed. In other words, if a volume point is farther than  $R$  from all RBF points, its position will not change during deformation. It is important to note that, in equation (4.1),  $s$  and  $\alpha_i$  are defined as a function of time  $t$ , because their value differ from one deformation to another, that is, from one time instance to another in the NLFD scheme. Therefore, as stated previously, since  $2N + 1$  deformed meshes are required for an  $N$ -mode solution, equation (4.1) is repeated for each time instance  $t_n = \frac{n}{2N+1}T$ ,  $n = 0, 1, \dots, 2N$ , and thus  $s$  and  $\alpha_i$  are reputed to be functions of time. The time variable  $t$  will however be dropped from the subsequent equations for clarity, since the process is identical regardless of the time instance at which it is performed. It should also be stressed that  $\mathbf{x}_p$  denotes the position of any arbitrary point  $p$ , be it an RBF point  $\mathbf{x}_r$  or a volume point  $\mathbf{x}_v$ , in the undeformed grid. Regrouping all RBF-point displacements in vector  $\Delta\mathbf{x}_r$ , equation (4.1) may be rewritten in matrix form as

$$s_x(\mathbf{x}_r) = \Delta\mathbf{x}_r = \begin{bmatrix} \phi_{r_1 r_1} & \phi_{r_1 r_2} & \cdots \\ \phi_{r_2 r_1} & & \\ \vdots & \ddots & \end{bmatrix}_{N_{rp} \times N_{rp}} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \end{bmatrix}_{N_{rp}} = [\mathbf{M}] \boldsymbol{\alpha}, \quad (4.4)$$

where  $\phi_{r_1 r_2} = \phi(\|\mathbf{x}_{r_1} - \mathbf{x}_{r_2}\|_2)$  is the value of  $\phi$  as a function of the distance between RBF points  $r_1$  and  $r_2$ , and  $\boldsymbol{\alpha}$  is the vector of interpolating coefficients  $\alpha_i$ . Since the

displacement of the RBF points is known, the vector of coefficients is easily computed by

$$\boldsymbol{\alpha} = [\mathbf{M}]^{-1} \Delta \mathbf{x}_r. \quad (4.5)$$

Similarly, regrouping the displacement in the  $x$ -direction of all volume points in vector  $\Delta \mathbf{x}_v$ , equation (4.1) is rewritten as

$$s_x(\mathbf{x}_v) = \Delta \mathbf{x}_v = \begin{bmatrix} \phi_{v_1 r_1} & \phi_{v_1 r_2} & \cdots \\ \phi_{v_2 r_1} & & \\ \vdots & & \ddots \end{bmatrix}_{N_{vp} \times N_{rp}} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \end{bmatrix}_{N_{rp}} = [\mathbf{A}] \boldsymbol{\alpha}, \quad (4.6)$$

where  $N_{vp}$  is the number of volume points in the mesh. Substituting equation (4.5) into equation (4.6) yields

$$\Delta \mathbf{x}_v = [\mathbf{A}] [\mathbf{M}]^{-1} \Delta \mathbf{x}_r, \quad (4.7)$$

and the displacement of all volume points is hence defined as a function of the displacement of all RBF points through matrices  $[\mathbf{A}]$  and  $[\mathbf{M}]^{-1}$ . The displacements are afterwards readily used as position increments from the original mesh in order to yield the deformed mesh. The procedure is repeated for all three directions,  $x$ ,  $y$  and  $z$ , and comes down to evaluating equation (4.7) three times for each time instance. In the case where a symmetry plane is imposed at some specific location in the mesh, it is essential that the grid points contained in the plane be imposed a null out-of-plane displacement in order to preserve symmetry. In the case of a wing of which root is located on the  $z = 0$  plane, a scaling function is used for the deformation of the mesh in the spanwise ( $z$ ) direction such that the spanwise displacements are exponentially

decreasing and are exactly zero at the wing root [35, 67]. The displacement in the spanwise direction of an arbitrary point is therefore expressed as

$$s_z(\mathbf{x}_p) = \kappa(z, \gamma) \sum_{i=1}^{N_{rp}} \alpha_i \phi(\|\mathbf{x}_p - \mathbf{x}_i\|_2), \quad (4.8)$$

with

$$\kappa(z, \gamma) = 1 - \exp\left(-\gamma \frac{z}{z_{\text{tip}}}\right), \quad (4.9)$$

where  $\kappa(z, y)$  is the scaling function,  $\gamma$  is a tuning parameter,  $z$  is the spanwise coordinate of the point, and  $z_{\text{tip}}$  is the spanwise coordinate of the wing tip.

Besides, since the employed solver for three-dimensional problems handles multi-block grids, the deformation is performed in two steps, as proposed by Walther [88]. In the first step, the body points are used as RBF points and only the block faces are deformed on a single processor. In the second step, the displacements of the points located on the block faces are passed to all processors, and the face points are used as RBF points for the parallel computation of the displacement of the interior points of the blocks. It is also extremely important to note that, in order to minimize the computational cost of the method, deformation at all time instances is performed based on the same undeformed mesh, and only one evaluation of matrices  $[\mathbf{M}]^{-1}$  and  $[\mathbf{A}]$  is therefore required. However, in practice, matrix  $[\mathbf{A}]$  is very memory-intensive, and would require excessive space if stored entirely. Therefore, only the inverse of  $[\mathbf{M}]$  is computed during preprocessing, and  $[\mathbf{A}]$  is computed at execution without being assembled. In this work, all matrix computations are carried out using the LAPACK library [4]. An example of a mesh deformation using the RBF method in

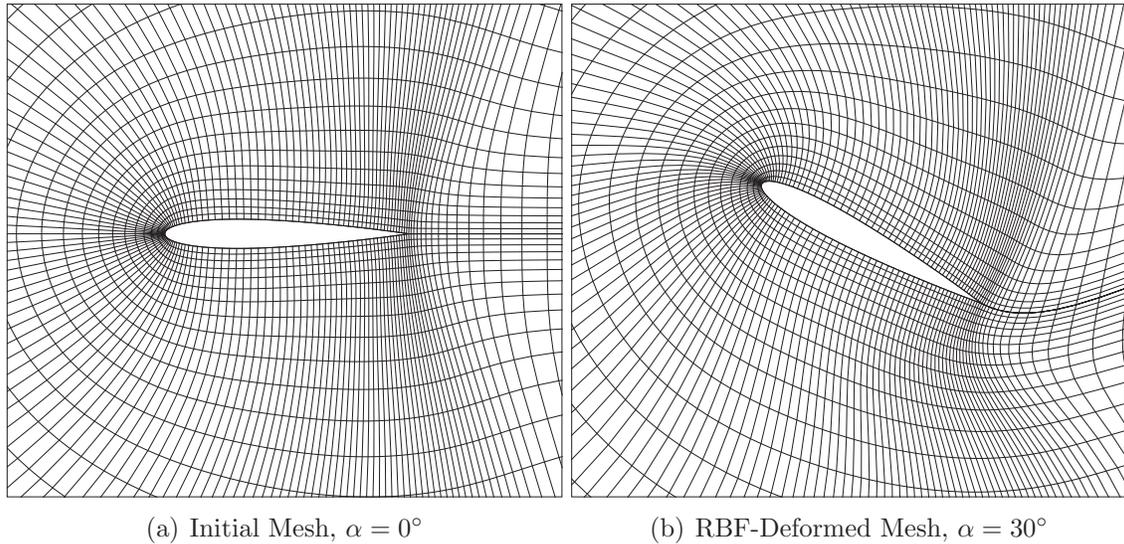


Figure 4-1: Deformation of the Mesh around a NACA 0012 Airfoil using the RBF Method

two dimensions is illustrated in Figure 4-1 for a NACA 0012 airfoil pitched at a  $30^\circ$  angle of attack.

#### 4.1.2 Mesh Velocities

As mentioned in section 2.1, the ALE formulation of the flow governing equations require the computation of the mesh velocities in order to modify the value of the convective fluxes at the cell faces. The method employed herein for the computation of the mesh velocities was previously developed by the present author and his advisor [78]. First, it is convenient to consider the displacement vector of an arbitrary mesh point, defined in section 4.1.1, as a position vector having its origin on the point in the undeformed grid and its other end on the point in the deformed grid. This consideration is perfectly admissible since the computed displacement corresponds to the subtraction of the initial position vector of the point from its

final post-deformation position. Therefore, for a deformation performed at time  $t$ , the displacement of an arbitrary point  $p$  may be expressed as

$$s(\mathbf{x}_p, t) = \mathbf{x}_f(t) - \mathbf{x}_p, \quad (4.10)$$

where  $\mathbf{x}_f(t)$  is the position of point  $p$  after deformation at time  $t$ . It should be stressed that  $\mathbf{x}_f$  is a function of time, since it corresponds to the position of the point after deformation (which varies according to the time instance at which the deformation is carried out), whereas  $\mathbf{x}_p$  is always taken as the position of the point in the fixed, original grid.

Equation (4.10) may also be rewritten by defining the final position of the point as the sum of its initial position and its displacement,

$$\mathbf{x}_f(t) = \mathbf{x}_p + s(\mathbf{x}_p, t), \quad (4.11)$$

and afterwards be differentiated with respect to time, such that

$$\frac{d(\mathbf{x}_f(t))}{dt} = \frac{d(\mathbf{x}_p)}{dt} + \frac{d(s(\mathbf{x}_p, t))}{dt}. \quad (4.12)$$

It can be seen that the left-hand side of the equation represents the velocity of point  $p$  after deformation at time  $t$ , which is thus rewritten as  $\mathbf{v}_f(t)$  or, equivalently, as  $v(\mathbf{x}_p, t)$ . On the right-hand side, the first term vanishes since the position of the point in the undeformed grid is the same regardless of the time at which the deformation is performed. Equation (4.12) is therefore rewritten as

$$v(\mathbf{x}_p, t) = \frac{d(s(\mathbf{x}_p, t))}{dt}. \quad (4.13)$$

After substitution of equation (4.1) in equation (4.13) and evident simplifications, the  $x$ -direction velocity of the grid point is expressed as

$$v_x(\mathbf{x}_p, t) = \sum_{i=1}^{N_{rp}} \beta_i(t) \phi(\|\mathbf{x}_p - \mathbf{x}_i\|_2), \quad \beta_i(t) = \frac{d}{dt} (\alpha_i(t)) \quad \forall i. \quad (4.14)$$

This equation is very similar to equation (4.1), and the remainder of the procedure is thus very straightforward. Regrouping all RBF-point  $x$ -direction velocities in vector  $\mathbf{v}_r$  and following the same procedure as for the computation of the  $\alpha_i$  coefficients, the vector  $\boldsymbol{\beta}$  of  $\beta_i$  coefficients is evaluated through

$$\boldsymbol{\beta} = [\mathbf{M}]^{-1} \mathbf{v}_r. \quad (4.15)$$

Then, regrouping all volume-point  $x$ -direction velocities in vector  $\mathbf{v}_v$ , those velocities are readily computed using the same procedure as for displacements, such that

$$\mathbf{v}_v = [\mathbf{A}] \boldsymbol{\beta}, \quad (4.16)$$

which, after substitution of equation (4.15), simplifies to

$$\mathbf{v}_v = [\mathbf{A}] [\mathbf{M}]^{-1} \mathbf{v}_r. \quad (4.17)$$

By evaluating equation (4.17) in all three directions, the volume-point velocities are related to the RBF-point velocities in an identical fashion as their displacements. Consequently, the developed formulation assumes that RBF-point velocities are known *a priori* and are used as the basis for the computation of volume-point velocities. This in general does not pose any problem since, similarly to displacements, the RBF-point velocities may either be provided by analytical functions in

the case of prescribed body motion, or by the structural solver in the case of aeroelastic simulations. Besides, a similar modification is applied to the method to prevent out-of-plane velocities in the symmetry plane, such that

$$v_z(\mathbf{x}_p, t) = \kappa(z, \gamma) \sum_{i=1}^{N_{rp}} \beta_i(t) \phi(\|\mathbf{x}_p - \mathbf{x}_i\|_2), \quad (4.18)$$

where  $\kappa(z, \gamma)$  is given in equation (4.9).

The approach described by equation (4.17) for computing the mesh velocities will be referred to as the Radial Basis Function method for Velocities (RBFV), in reference to its usage of the same RBF operator as is employed for the deformation of the mesh. The cell face velocities, required in the ALE formulation of the flow governing equations, are afterwards easily computed as the average of the velocity of the four vertices defining each face.

In contrast, the developed RBFV method will be compared to what will be referred to as the Fast Fourier Transform method for Velocities (FFTV), introduced by Mosahebi [60] for the NLFD method in a fashion analogous to Dufour [17] for the HB method, where the time spectral derivative operator is applied to the position of the mesh points in order to yield the velocities. By assuming that the position of an arbitrary grid point after mesh deformation at any time  $t$  can be represented via its  $N$ -mode Fourier series as follows,

$$\mathbf{x}_f(t) = \sum_{k=-N}^N \hat{\mathbf{x}}_k e^{i\frac{2\pi k}{T}t}, \quad (4.19)$$

its velocity at time  $t$  is then expressed as

$$\mathbf{v}_f(t) = \sum_{k=-N}^N i \frac{2\pi k}{T} \hat{\mathbf{x}}_k e^{i \frac{2\pi k}{T} t}. \quad (4.20)$$

The FFTV approach has the advantage of being very simple to implement and computationally efficient, but its accuracy is dependent on the number of modes  $N$  employed to represent the grid position in time [78]. By opposition, since the RBFV method relies directly on the mathematics of the deformation, the computed mesh velocities are exact. The developed RBFV method is employed in the present work, and its accuracy and efficiency is compared to the FFTV method in chapter 6. It is very important to note that, regardless of the approach employed for the computation of the mesh velocities (RBFV or FFTV), the RBF method is always employed for the mesh deformation. When applicable, results obtained using the RBFV method are also be compared to those obtained from a rigid grid movement technique, in which the entire computational grid is translated and rotated rigidly with the body.

## 4.2 Treatment of the Geometric Conservation Law

As first demonstrated by Thomas and Lombard [84], an additional conservation law, called the Geometric Conservation Law (GCL), must be solved concurrently with the other governing equations on deforming grids in order to avoid errors induced by the deformation of the control volumes. Such errors have been reported by Lesoinne and Farhat [46], among others, who illustrated the impact of the non-satisfaction of the GCL on aeroelastic solutions. In fact, the GCL states that the rate of change of the volume of a computational cell must equal the rate of change of the volumetric increment caused by the motion of its boundary, such that

$$\frac{\partial}{\partial t} \int_{\mathcal{V}} d\Omega - \oint_{\partial\mathcal{V}} (\mathbf{V} \cdot \mathbf{n}) dS = 0. \quad (4.21)$$

In continuous form, this equation is always satisfied since it relies purely on geometrical considerations, as can be demonstrated for an arbitrary control volume  $\mathcal{V}$ . It is also always satisfied for meshes where the shape of the control volumes does not change in time, that is, for rigid grid motion. The satisfaction of the GCL ensures that the state of a uniform flow is preserved, regardless of the movement of the control volumes. However, in order to yield a consistent solution method, the GCL must be discretized using the same numerical scheme as employed for the solution of the other conservation laws [8]. In the case of the present work, this translates to a finite-volume spatial discretization and an NLFD temporal discretization of the GCL. Applying the spatial discretization, equation (4.21) is expressed in semi-discrete form for an arbitrary control volume as

$$\frac{d\Omega}{dt} - \sum_{m=1}^{N_f} (\mathbf{V} \cdot \mathbf{n})_m S_m = 0, \quad \text{in } \mathcal{V}. \quad (4.22)$$

Previous work has considered satisfying the GCL separately for each face of the discretized control volumes, such that the increase in the cell volume due to the movement of one face is equal to the volume swept by this same face. Although equation (4.22) does not require this methodology to be employed, the approach is more intuitive and has been shown to provide good results [87]. Nevertheless, this method is easily applicable to linear or quadratic temporal discretizations, commonly employed in time-marching unsteady flow solvers, but its extension to frequency-domain solution techniques is not straightforward. Effectively, since the GCL needs

to be discretized in the same fashion as are the governing equations, the discretization procedure is similar to that detailed in section 2.3.1, such that

$$\Omega(t) = \sum_{k=-N}^N \hat{\Omega}_k e^{i\frac{2\pi k}{T}t}, \quad (4.23)$$

$$G(t) = \sum_{k=-N}^N \hat{G}_k e^{i\frac{2\pi k}{T}t}, \quad (4.24)$$

with

$$G(t) = \sum_{m=1}^{N_f} (\mathbf{V} \cdot \mathbf{n})_m S_m. \quad (4.25)$$

Substituting the above Fourier representations in equation (4.22) yields

$$\frac{d}{dt} \left( \sum_{k=-N}^N \hat{\Omega}_k e^{i\frac{2\pi k}{T}t} \right) - \sum_{k=-N}^N \hat{G}_k e^{i\frac{2\pi k}{T}t} = 0. \quad (4.26)$$

Evaluating the temporal derivative, the above equation is rewritten as

$$\sum_{k=-N}^N i\frac{2\pi k}{T} \hat{\Omega}_k e^{i\frac{2\pi k}{T}t} - \sum_{k=-N}^N \hat{G}_k e^{i\frac{2\pi k}{T}t} = 0. \quad (4.27)$$

Taking advantage of the orthogonality of the Fourier series, this equation is decomposed in a system of  $2N + 1$  equations, each of which corresponds to a wave number  $k$  as follows:

$$i\frac{2\pi k}{T} \hat{\Omega}_k - \hat{G}_k = 0, \quad \text{for } -N \leq k \leq N. \quad (4.28)$$

The criterion for the satisfaction of the GCL using the NLFD method is therefore the satisfaction of equation (4.28) for all wave numbers. It is not satisfied automatically in general and must therefore be enforced through the modification of either

the cell volume or the cell face velocities. However, many obstacles prevent this condition from being fulfilled in a straightforward fashion. First, it seems particularly attractive to split the GCL in separate parts for each face of the control volume as was shown in previous work for time-marching methods [87, 93]. In fact, the exact volume of the cell is known at all time instances since it is only a function of the position of the cell vertices, which is known from the mesh deformation algorithm. It would therefore be straightforward to consider the volumetric increment due to the movement of each face separately, based on this analytical function. However, this analytical function  $\Omega(t)$  is known only in continuous form; this information is lost during the temporal discretization of the volume described by equation (4.23), and only the spectral representation of the volume is known in equation (4.28). A separation of the volumetric increments induced by each face is therefore not feasible in the discrete form of the GCL.

A second approach would consist in modifying the value of  $\hat{\Omega}_k$  for all wave numbers such that the GCL is satisfied. This would result in

$$\hat{\Omega}_k = \frac{T}{i2\pi k} \hat{G}_k, \quad \text{for } -N \leq k \leq N, \quad k \neq 0, \quad (4.29)$$

where the exception to the zeroth mode would be required to avoid a division by zero. The average volume of the cell would therefore remain unchanged, while its Fourier coefficients would be modified to satisfy the GCL. The issue with this approach is that the computed volume,  $\Omega(t)$ , retrieved using an IDFT after the satisfaction of the GCL, would not correspond to the volume computed from the actual metrics at

each time instance. Such inconsistency could lead to solution errors by rendering the discretization non conservative.

The inverse approach would consist in modifying the value of  $\hat{G}_k$  such that it equals  $i\frac{2\pi k}{T}\hat{\Omega}_k$  for all wavenumbers. However, this would lead to non unique solutions since  $G(t)$  comprises a sum and scalar products, which are irreversible operations. A decision would therefore need to be taken regarding the manner in which the face velocities are modified individually. Moreover, great care would need to be taken in order to ensure that the GCL is satisfied for every volume cell; indeed, modifying the face velocities for a specific cell would result in the non-satisfaction of the GCL in the adjacent cells, since they are coupled through the requirement that the face velocity be equal for the two cells adjacent to that face. One possible solution to this issue would consist in solving implicitly the cell face velocities simultaneously via the construction of a global linear system regrouping the velocities of all cell faces in the domain. Since every cell has six faces and as the GCL must be satisfied for every cell, this would yield a 6-banded matrix, of size equal to the total number of cell faces in the domain. The system would need to be solved every time the grid is deformed, and thus an efficient approach would need to be adopted to minimize the computational cost of the procedure. This seems to be the most promising approach to the author. However, it was not implemented in the present work due to time constraints, and therefore the GCL was not enforced for deforming grids. Nevertheless, since the employed temporal discretization provides results close to spectral accuracy, it is expected that the error induced by the omission of the GCL will be negligible in comparison with typical time-marching methods which

use low-order integration techniques. This assumption seems to be confirmed by the results obtained in chapter 6, where solutions obtained using deforming grids compare extremely well with rigid grid solutions, for which the GCL is automatically satisfied.

## CHAPTER 5

### Aeroelastic Framework

This chapter describes the employed technique for the coupling of the flow and structural solvers, and details the proposed NLFD/LCO method for the determination of flow conditions leading to limit cycle oscillations.

#### 5.1 Fluid-Structure Coupling

Typical time-accurate aeroelastic solvers [2] couple the fluid and structural solutions at each time step; after the flow solution has converged satisfactorily in pseudotime, the aerodynamic loads are transferred to the structural solver, which integrates the structural equations in time in order to provide the body displacement and velocity at the next time step to the flow solver. The process is repeated until the transient effects due to the initialization of the flow field to the free-stream values have vanished, such that the obtained solution does not depend on the initial conditions. As the main advantage of the NLFD method is its direct convergence to a periodic steady-state solution, the fluid and structural solutions cannot be coupled at each physical time step, and are instead coupled for an entire period at once.

Indeed, the coupling method employed in this work is based on the work of Kachra and Nadarajah [43]. The flow and structural solvers are coupled every  $N_{mg}$  multigrid cycles, where  $N_{mg}$  is a user-defined parameter chosen such that the convergence of the flow solution is satisfactory between each coupling. To initiate the motion, the first oscillation is prescribed to the body at a predefined frequency in the

vicinity of that of the final motion. Then, after each block of  $N_{mg}$  multigrid cycles, the aerodynamic loads obtained from the flow solution are provided as an exciting force vector to the structural solver, which may therefore solve the equations of motion to calculate the displacements and velocities of the body points. These values are transferred back to the flow solver and are used in the dynamic mesh deformation framework detailed in chapter 4. A new flow solution is then computed using the modified displacements and velocities. The flow conditions are also updated during the fluid-structure coupling, based on the Newton-Raphson method described in section 5.3, and the process is repeated until the solution has converged to a limit cycle oscillation.

## **5.2 Interpolation of Aerodynamic Forces and Displacements**

As the spatial and temporal discretizations may differ from the flow to the structural solver, interpolation approaches are employed for the fluid-structure coupling, as detailed hereafter.

### **5.2.1 Spatial Interpolation**

Since the structural grid employed for the discretization of the body is in general not coincident with the fluid grid employed for the discretization of the fluid domain, an interpolation algorithm has to be employed in order to transfer aerodynamic loads from the flow solver to the structural solver. Here, a very simple interpolation procedure is employed for the sake of simplicity. Effectively, the forces resulting from the integration of the pressure at the face of each fluid cell are assumed to be acting exactly through the face center. Accordingly, for an arbitrary cell face located on the body surface in the fluid mesh, the acting point is first located on the structural

grid, and the corresponding element containing the point is determined. Knowing the force vector and the acting point on the element, the nodal forces and moments for the noted element are computed based on the elemental shape functions defined in equation (3.3), following the procedure described in [69]. The so-called consistent load vector is thus determined by superimposing the discrete representation of the distributed forces and moments acting at the body surface for all elements.

Since the displacements obtained from the structural solver are expressed as structural-model nodal displacements, a similar procedure must be employed for the computation of the fluid-model body displacements. Effectively, each body point of the fluid mesh is mapped to the structural model and is associated with the structural element containing it. When the body displacements need to be mapped back from the structural to the fluid model, the nodal displacements from the structural model are used along with the shape functions defined in equation (3.3) for the interpolation of the displacement of the fluid-mesh body point, based on its position inside the structural element. The same approach is employed using the nodal velocities, which therefore translate into body point velocities in the fluid mesh. The computed displacements and velocities can then be used in the mesh deformation algorithm described in chapter 4.

### **5.2.2 Temporal Discretization Interface**

Because the harmonics of the structural displacements are directly computed from the harmonics of the aerodynamic loads, as expressed by equation 3.12, the number of harmonics employed in the flow solution ( $N$ ) must be greater or equal to that employed in the structural solution ( $N_s$ ). Therefore, an interpolation approach

has to be used for the transfer of the aerodynamic loads to the structural solver, and of the structural displacements to the flow solver. This is done identically to the work of Thomas et al. [79], in which the highest harmonics of the aerodynamic loads are dropped during the transfer to the structural solver, such that only the first  $N_s$  harmonics are passed. During the inverse operation, the highest harmonics of the displacements, for which  $|k| > N_s$ , are zeroed out for the transfer to the flow solver, such that all  $N$  harmonics of the displacements are defined. Figure 5–1 schematizes the procedure for the transfer of data between the flow and structural solvers.

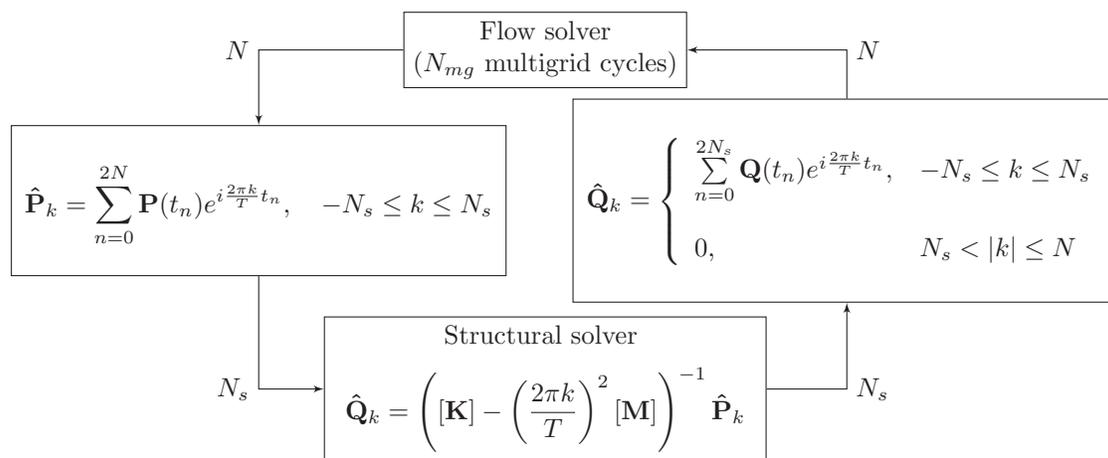


Figure 5–1: Dataflow Diagram of the Temporal Discretization Interface of the Fluid-Structure Coupling

It is important to note that, to the author’s knowledge, this approach does not guarantee that the transfer is conservative, on the point of view of the work done by the aerodynamic forces on the body over a period. Effectively, if  $N$  differs from  $N_s$ , information about the highest harmonics is lost during the transfer of the aerodynamic loads. Moreover, the value of the aerodynamic loads in the time

domain at all solution points may not be preserved, and therefore the approach cannot be referred to as an interpolation method. The only solution to this issue would consist in the employment of the same discretization for both solvers, such that  $N_s = N$ . However, the method proposed in section 5.3 for the computation of LCO flow conditions was observed to behave improperly using  $N_s = N$ . This behavior is still under investigation, in hope that a fully nonlinear structural solver may be employed in the future. To prevent this issue, the discretization of the structural solver is set to  $N_s = 1$  for all cases of the present work, yielding a linear structural solver. This also signifies that only the first harmonic of the aerodynamic loads is employed during the fluid-structure coupling. However, the first harmonic still contains the nonlinear effects of the flow since it is computed as part of a multiple-harmonic solution, as explained by Ekici and Hall [19].

### 5.3 Limit Cycle Oscillation Determination Methodology

Since the NLFD method is only able to model purely periodic flows, the extent of the present work is limited to the study of periodic aeroelastic motion, such as periodic LCO, which can also be used to model flutter onset if the amplitude of the motion is sufficiently small that the nonlinearities are negligible. In fact, periodic LCO are primarily characterized by two independent parameters; frequency and airspeed. For a wing, they are represented in nondimensional form by the reduced frequency and the speed index, defined respectively as

$$\omega_r = \frac{\omega c}{2V_\infty}, \quad (5.1)$$

$$V_f = \frac{V_\infty}{b_s \omega_\alpha \sqrt{\mu}}, \quad (5.2)$$

where  $\omega$  is the angular frequency of the motion,  $c$  is the reference chord,  $V_\infty$  is the modulus of the free-stream flow velocity,  $b_s$  is one half of the reference chord,  $\omega_\alpha$  is the angular frequency of the first torsional mode of the wing, and  $\bar{\mu}$  is the mass ratio of the wing, defined as

$$\bar{\mu} = \frac{m_b}{\rho_b \Omega_{\text{frus}}}. \quad (5.3)$$

In this equation,  $m_b$  is the total mass of the wing,  $\rho_b$  is the density of the wing material, and  $\Omega_{\text{frus}}$  is the volume of the conical frustum having the streamwise root chord of the wing as lower-base diameter, the streamwise tip chord as upper-base diameter, and the wing span as height.

In time-accurate aeroelastic computations, the speed index is the only independent variable, since the frequency tends to set naturally while the solution is marched in time. Therefore, one may choose a given speed index, and observe the flow behavior after the initial transients have vanished in order to assess whether the flow conditions lead to damped, neutral, or diverging oscillations. However, the NLFD method requires that a reduced frequency be provided *a priori* to the flow solver such that the flow can be decomposed in Fourier harmonics accordingly. The proposed NLFD method for aeroelastic computations thus possesses both the reduced frequency and speed index as independent variables. For example, one cannot assess uniquely the aeroelastic behavior of a body under a given speed index, since different reduced frequencies could lead to different results for this single speed index. Nevertheless, these different results may not all be physically sensible, as explained hereafter.

Firstly, a solution of which amplitude varies from period to period has a questionable physical meaning for the NLFD method since the latter assumes that the solution is periodic in time. Therefore, for a chosen pair of reduced frequency and speed index, if the fluid-structure coupling tends to increase or decrease the amplitude of the motion from period to period, the pair is then reputed to be physically meaningless from an NLFD point of view. This thus justifies the requirement according to which the amplitude of the motion needs to be constant from period to period in NLFD-based aeroelastic computations. Secondly, a physically sensible flow solution must be continuous in time and space from one period to the next. Therefore, a solution experiencing phase jumps between periods due to the fluid-structure coupling must be considered nonphysical, since it would represent either space or time discontinuities in the flow or structural solution. This concept was also exploited by Kielb et al. [44] for vortex shedding frequency determination using the HB method.

Based on these two concepts, a methodology is developed for the computation of periodic aeroelastic phenomena using the NLFD technique, called the NLFD/LCO method. Assuming that the reduced frequency and speed index are the only two unknown parameters, they are employed as the independent variables of the problem. Since lift provides a good measure of the global behavior of the flow around the structure, it is used as a basis for the definition of the two dependent variables of the system. As the amplitude and phase lag of the first mode of the lift coefficient are defined respectively as

$$A_1 = 2 \sqrt{\left(\operatorname{Re}(\hat{C}_{L,1})\right)^2 + \left(\operatorname{Im}(\hat{C}_{L,1})\right)^2},$$

$$\psi_1 = \arctan\left(\frac{\operatorname{Im}(\hat{C}_{L,1})}{\operatorname{Re}(\hat{C}_{L,1})}\right),$$

it can be seen that they are both only functions of the real and imaginary parts of the first mode of the lift coefficient. The two latter are therefore employed as dependent variables. The objective, in order to get a constant-amplitude and constant-phase solution, is to find the pair of reduced frequency and speed index that yields constant  $\operatorname{Re}(\hat{C}_{L,1})$  and  $\operatorname{Im}(\hat{C}_{L,1})$  from period to period. To achieve this objective, two functions  $\phi_1$  and  $\phi_2$  are defined as the variation of those parameters over a given number of periods  $M_p$ , such that

$$\begin{aligned}\phi_1 &= \Delta\operatorname{Re}(\hat{C}_{L,1}) = \operatorname{Re}(\hat{C}_{L,1})_{m+M_p} - \operatorname{Re}(\hat{C}_{L,1})_m, \\ \phi_2 &= \Delta\operatorname{Im}(\hat{C}_{L,1}) = \operatorname{Im}(\hat{C}_{L,1})_{m+M_p} - \operatorname{Im}(\hat{C}_{L,1})_m,\end{aligned}\tag{5.4}$$

where the subscript  $m$  denotes the index of the period at which the lift coefficient is computed. In addition,  $\phi_1$  and  $\phi_2$  are assumed to be functions of the reduced frequency and speed index, such that

$$\begin{aligned}\phi_1 &= f(\omega_r, V_f), \\ \phi_2 &= g(\omega_r, V_f).\end{aligned}$$

A root-finding Newton-Raphson technique is employed in order to drive the two dependent variables to zero as follows,

$$\begin{Bmatrix} \omega_r \\ V_f \end{Bmatrix}_{n+1} = \begin{Bmatrix} \omega_r \\ V_f \end{Bmatrix}_n - \alpha [\mathbf{J}]_n^{-1} \begin{Bmatrix} \phi_1 \\ \phi_2 \end{Bmatrix}_n, \quad (5.5)$$

where subscripts  $n$  and  $n + 1$  denote the Newton-Raphson iteration number,  $\alpha$  is a relaxation factor, and  $[\mathbf{J}]$  is the Jacobian of the system, defined as

$$[\mathbf{J}] = \begin{bmatrix} \frac{\partial \phi_1}{\partial \omega_r} & \frac{\partial \phi_1}{\partial V_f} \\ \frac{\partial \phi_2}{\partial \omega_r} & \frac{\partial \phi_2}{\partial V_f} \end{bmatrix}. \quad (5.6)$$

In this work, the Jacobian is evaluated using a first-order forward finite-differencing technique, and the relaxation factor is defined as

$$\alpha = 1 - \frac{1}{2}e^{-n}.$$

This iterative method is employed until satisfactory values are obtained for the dependent variables  $\phi_1$  and  $\phi_2$ , that is, when they are both sufficiently close to zero for engineering purposes. In theory, the dependent variables could be defined using  $M_p = 1$ , that is, always comparing two successive solution periods. However, using a greater number of periods (increased  $M_p$ ) may provide a better approximation of the variation of  $\text{Re}(\hat{C}_{L,1})$  and  $\text{Im}(\hat{C}_{L,1})$  in the solution due to a longer sample length and a better fluid-structure coupling. The sequential approach is detailed below.

1. At the beginning of period  $m$ , which begins the  $n^{\text{th}}$  iteration of the Newton-Raphson method, the flow solution is stored, along with  $\text{Re}(\hat{C}_{L,1})_m$ ,  $\text{Im}(\hat{C}_{L,1})_m$ ,  $\omega_r$ , and  $V_f$ .
2. The flow solution is computed for  $M_p$  consecutive periods, between each of which the flow and structural solvers are coupled.

3. After  $M_p$  periods,  $\text{Re}(\hat{C}_{L,1})_{m+M_p}$  and  $\text{Im}(\hat{C}_{L,1})_{m+M_p}$  are obtained, and reference values for  $\phi_1$  and  $\phi_2$  are computed via equation (5.4).
4. The solution is reinitialized to that of step 1. The reduced frequency is incremented by  $\Delta\omega_r$ , determined based on the level of convergence of the solution.
5. The solution is computed for  $M_p$  consecutive periods, with a reduced frequency equal to  $\omega_r + \Delta\omega_r$ .
6. After  $M_p$  periods,  $\phi_1^*$  and  $\phi_2^*$  are computed via equation (5.4), but differ from  $\phi_1$  and  $\phi_2$  because of the increment in reduced frequency.
7. The solution is reinitialized to that of step 1. The reduced frequency is set back to its initial value  $\omega_r$ , and the speed index is incremented by  $\Delta V_f$ , based on the level of convergence of the solution.
8. The solution is computed for  $M_p$  consecutive periods, with a speed index equal to  $V_f + \Delta V_f$ .
9. After  $M_p$  periods,  $\phi_1^{**}$  and  $\phi_2^{**}$  are computed via equation (5.4), but differ from  $\phi_1$  and  $\phi_2$  because of the increment in speed index.
10. The Jacobian of the method is constructed through the computation of the derivatives as follows,

$$\begin{aligned} \frac{\partial\phi_1}{\partial\omega_r} &= \frac{\phi_1^* - \phi_1}{\Delta\omega_r}, & \frac{\partial\phi_1}{\partial V_f} &= \frac{\phi_1^{**} - \phi_1}{\Delta V_f}, \\ \frac{\partial\phi_2}{\partial\omega_r} &= \frac{\phi_2^* - \phi_2}{\Delta\omega_r}, & \frac{\partial\phi_2}{\partial V_f} &= \frac{\phi_2^{**} - \phi_2}{\Delta V_f}, \end{aligned}$$

and the reduced frequency and speed index are updated through equation (5.5) to yield their value at iteration  $n + 1$ .

11. The process is repeated from step 1 until satisfactory convergence is achieved.

Using the NLFD/LCO method, it is possible to find limit cycle oscillations using the NLFD method. Unfortunately, the amplitude of the resulting LCO is undetermined *a priori* since the employed approach converges to constant amplitude and phase lag, but does not impose the final value they reach. It is therefore impossible to determine the reduced frequency and speed index values leading to an LCO of a given amplitude using this method, although this feature would be useful in the assessment of LCO parameters over a predefined range of LCO amplitudes. Further development on the technique is thus needed in order to allow the calculation of LCO of predefined amplitudes. However, it is assumed that the amplitude of LCO resulting from the use of the described Newton-Raphson method is strongly dependent on the amplitude of the forced oscillation prescribed to the body to initiate the solution. Therefore, pairs of reduced frequency and speed index leading to LCO can be found for various *a priori* unknown LCO amplitudes.

It is also important to note that the described Newton-Raphson method is carried out using the real and imaginary parts of the first mode of the lift coefficient only, and not the higher harmonics. However, since the structural solver employs only one harmonic, driving solely the first mode of the lift coefficient is sufficient for the convergence of the structural solution. Further development would be necessary for a structural solver where  $N_s > 1$ , since the higher harmonics of the structural solution could cause the higher harmonics of the flow solution to diverge while the first harmonic converges. A more involved study should therefore be carried out in order to transpose the presented method to a structural solver where  $N_s > 1$ .

## CHAPTER 6

### Results - Dynamic Mesh Deformation

The objective of this chapter is to validate the employed dynamic mesh deformation framework for the NLFD method, and more specifically the developed RBFV method for computing the mesh velocities, described in section 4.1.2. Results are obtained for a two-dimensional pitching NACA 0012 airfoil and a three-dimensional pitching LANN wing. Parts of the results presented at the 53<sup>rd</sup> AIAA Aerospace Sciences Meeting by the present author [78] are integrally reproduced herein.

#### 6.1 NACA 0012 Airfoil - *Run 81* of the CT1

The dynamic mesh deformation framework is first tested on a two-dimensional pitching NACA 0012 airfoil. The flow conditions are those described by *Run 81* of the Central Transonic test case CT1, carried out by Landon [45] at the Aircraft Research Association (ARA). The airfoil is subject to a sinusoidal pitching motion described by

$$\alpha(t) = \alpha_m + \alpha_0 \sin(\omega t + \psi), \quad (6.1)$$

where  $\alpha(t)$  is the angle of attack of the airfoil at time  $t$ ,  $\alpha_m$  and  $\alpha_0$  are respectively the mean angle of attack and the amplitude of the motion,  $\omega$  is the oscillation frequency and  $\psi$  is the phase lag of the motion. The reduced frequency of the motion is  $\omega_r = 0.0808$ , the mean angle of attack is  $\alpha_m = 2.89^\circ$ , the pitching amplitude is  $\alpha_0 = 2.41^\circ$ , and the free-stream Mach number is  $M = 0.60$ . The pivot point is

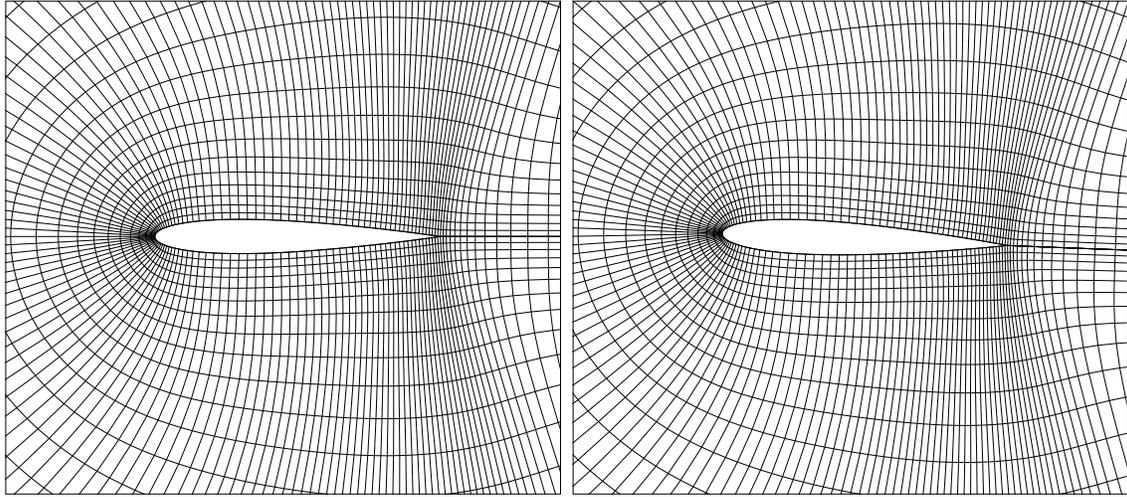
located one quarter-chord aft of the leading edge of the airfoil. The flow solution is computed on a  $192 \times 32$  C-mesh.

### 6.1.1 Accuracy of Mesh Velocities using the RBFV and FFTV Methods

Before computing flow solutions for the presented test case, the accuracy of the RBFV method for computing mesh velocities is compared to that of the FFTV method. Accordingly, the mesh is deformed dynamically to model the displacement of the airfoil over a period, and the mesh velocities are compared at the beginning of the period ( $t = 0$ ) for the two methods using various numbers of harmonics. Figure 6–1 depicts examples of undeformed and deformed meshes at angles of attack  $\alpha = 0^\circ$  and  $\alpha = 2.41^\circ$ , which correspond to the position of the airfoil at times  $t/T = 0$  and  $t/T = 1/4$ . A reference solution for the mesh velocities at time  $t = 0$  is computed using the FFTV method with 17 modes. The  $L^\infty$ - and  $L^2$ -norms of the error with respect to the reference solution are reported in Figure 6–2 as a function of the employed number of modes for both the RBFV and FFTV methods. It can be observed that, as expected, whereas the accuracy of the FFTV method depends on the number of harmonics, it is not the case for the RBFV method, which introduces a machine-precision error regardless of the number of modes. The mathematical approach developed in section 4.1.2 is therefore validated.

### 6.1.2 Solution Accuracy using the RBFV Method

In order to validate the accuracy provided by the RBFV method in the computation of unsteady flows using deformable grids, the solution obtained for *Run 81* using the RBFV method is compared to that obtained using rigid grid motion. To ensure that the flow is correctly converged, the rigid grid motion solution is first



(a) Undeformed Mesh,  $\alpha = 0^\circ$

(b) Deformed Mesh,  $\alpha = \alpha_0 = 2.41^\circ$

Figure 6-1: NACA 0012 Undeformed and Deformed Meshes

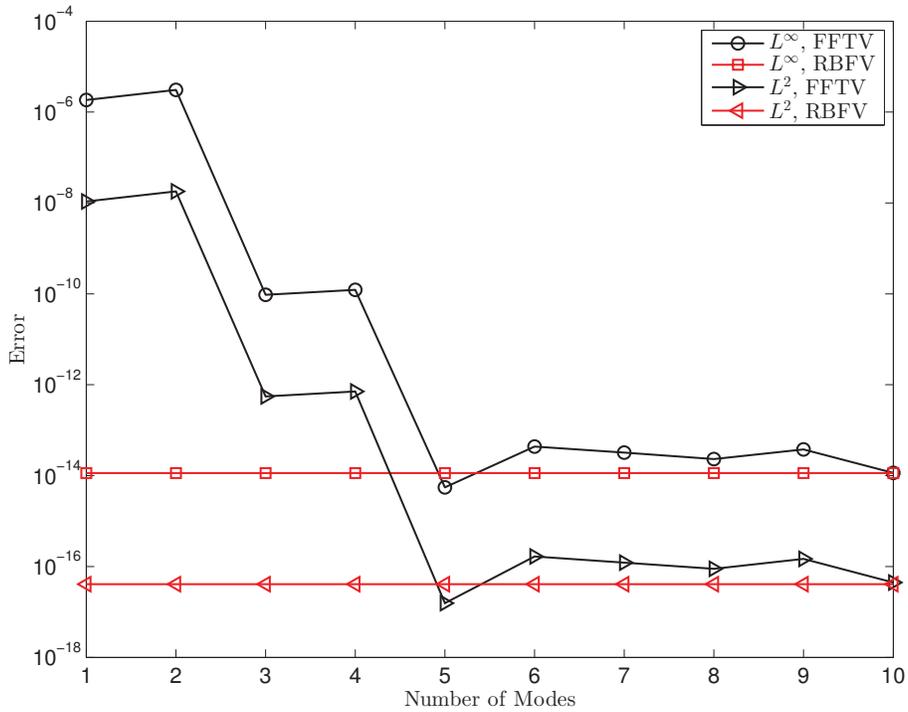


Figure 6-2: Convergence of Mesh Velocity Errors in  $L^\infty$  and  $L^2$  for the FFTV and RBFV Methods at  $t = 0$ , Run 81

calculated over 400 multigrid cycles for various numbers of harmonics. Figure 6–3 shows the flow solver convergence for all solutions, where “2 Modes – 1” in the legend indicates the convergence curve of the first harmonic of a 2-mode solution. As can be observed in Figure 6–4, the drag coefficient hysteresis loop is very similar from the 3-mode solution to the 4-mode solution. Therefore, 3 modes are employed for the comparison of the RBFV and rigid grid motion solutions.

Figure 6–5 displays the superposition of both the lift and drag coefficient hysteresis computed via the rigid grid motion and RBFV methods using 3 modes. The two solutions agree very well for both parameters. The pressure coefficient distribution along the chord of the airfoil is also presented in Figure 6–6, and it can again be seen that the rigid grid motion and RBFV approaches agree perfectly. The figure also shows that the pressure distribution is very close to experimental results for the two presented time instances, except at the leading edge for  $t/T = 0.314$ , where the pressure is slightly overpredicted.

### 6.1.3 Efficiency of the RBFV and FFTV Methods

To assess the computational efficiency of the RBFV method, it is compared to the FFTV method, since the latter is the only other method known to the author for the computation of mesh velocities in deforming grid cases using frequency-domain schemes. The flow conditions of *Run 81* are once again used, and the solution is computed on a  $384 \times 64$  C-mesh using 4 harmonics for 400 multigrid cycles. The finer mesh is intended to provide a better precision in the determination of the computational cost of the methods. Table 6–1 reports the computational time results for the two approaches, as well as the associated percentage of the total computational

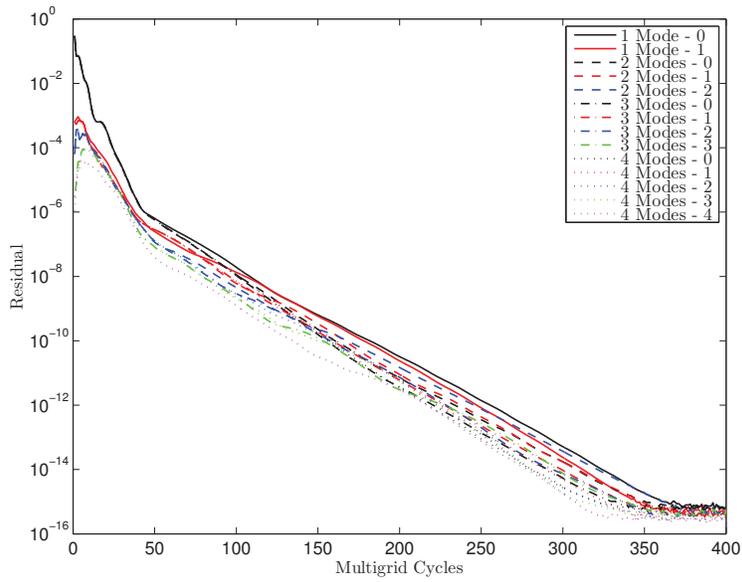


Figure 6-3: Flow Solver Convergence using Rigid Grid Motion, *Run 81*

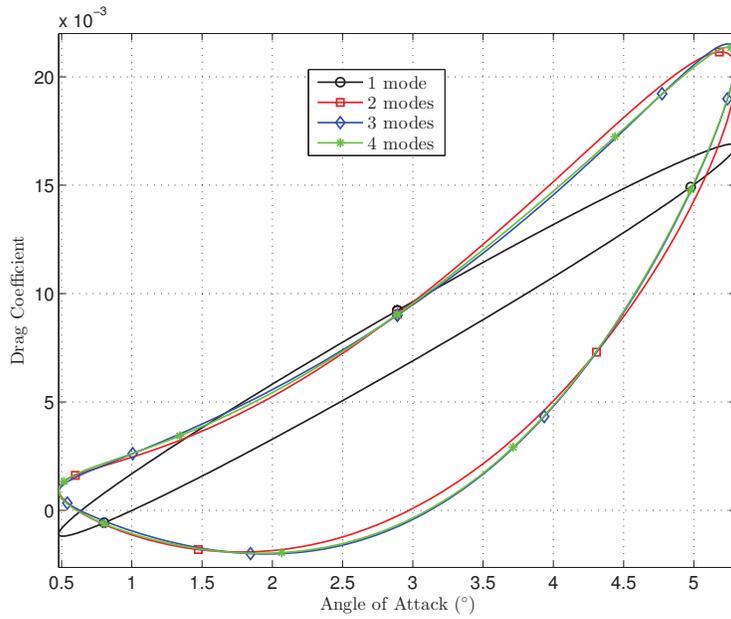
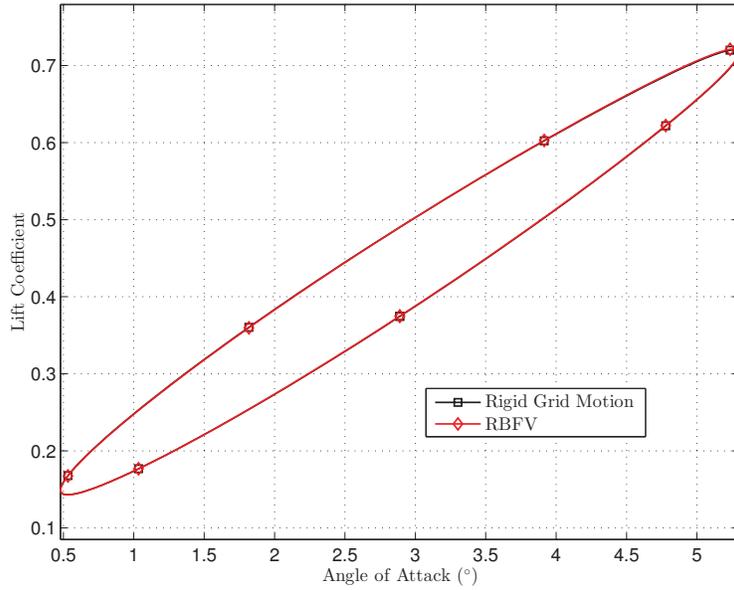
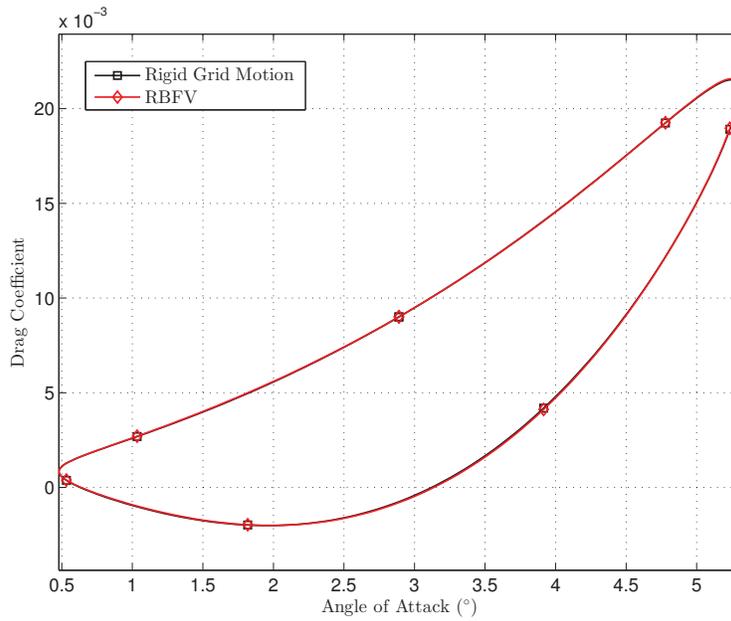


Figure 6-4: Drag Coefficient Hysteresis using Rigid Grid Motion, *Run 81*

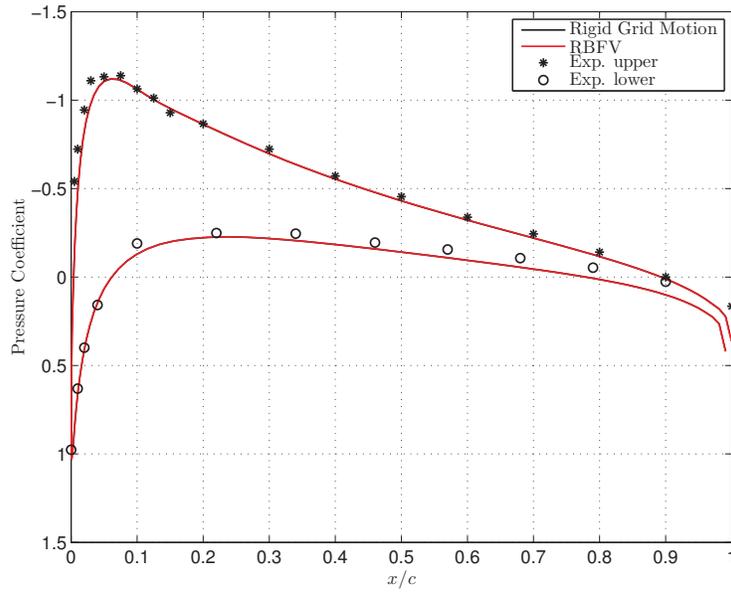


(a) Lift Coefficient Hysteresis

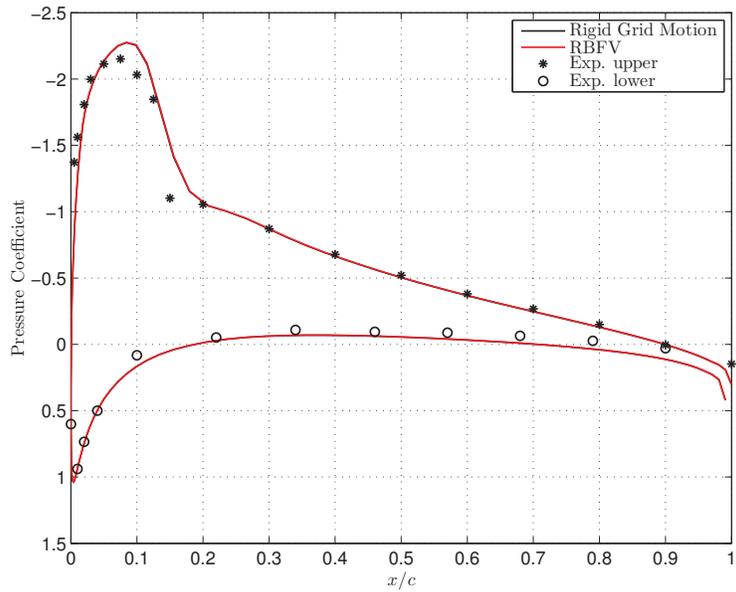


(b) Drag Coefficient Hysteresis

Figure 6–5: Comparison of the Rigid Grid Motion and RBFV Methods for the Computation of Lift and Drag Coefficient Hysteresis, *Run 81*



(a)  $t/T = 0.005, \alpha = 2.97^\circ$



(b)  $t/T = 0.314, \alpha = 5.11^\circ$

Figure 6-6: Comparison of the Rigid Grid Motion and RBFV Methods for the Computation of the Pressure Coefficient Distribution along the Airfoil Chord, *Run 81*

Table 6–1: Computational Cost of the FFTV and RBFV Methods for *Run 81* using 4 Modes

	CPU time (s)	CPU time (% of complete flow solution time)
FFTV method	0.06	0.02
RBFV method	0.11	0.04
Complete flow solution	291.88	100.00

time of the flow solution. From the results, it can be seen that the FFTV approach is roughly twice as fast as the RBFV method. This is due to the extremely low cost associated with the FFT computations, in comparison to the additional matrix operations required for the RBFV method. Nevertheless, the computational time for both approaches represents a very low percentage of that required to compute the entire solution. For that reason, the RBFV method is still an attractive alternative to the FFTV method.

## 6.2 LANN Wing - *Run 73* of the CT5

Tests similar to those applied to the two-dimensional NACA 0012 are carried out for a wing in order to validate the RBFV method in three dimensions. The flow conditions are those described by *Run 73* of the central transonic test case CT5, conducted by Zwaan [94] at the National Aerospace Laboratory (NLR). It consists of a LANN wing subject to a sinusoidal pitching motion described by equation (6.1), where the wing is pitched rigidly around an axis located 0.621 root chord lengths aft of the root leading edge. The oscillatory parameters are set to  $\alpha_m = 0.59^\circ$ ,  $\alpha_0 = 0.25^\circ$  and  $\psi = 0$ . The reduced frequency, based on the root chord, is set to  $\omega_r = 0.102$ , and the Mach number is  $M = 0.822$ . As it was shown to provide good results in previous work [65], a multiblock  $192 \times 32 \times 96$  C-mesh, consisting of 12

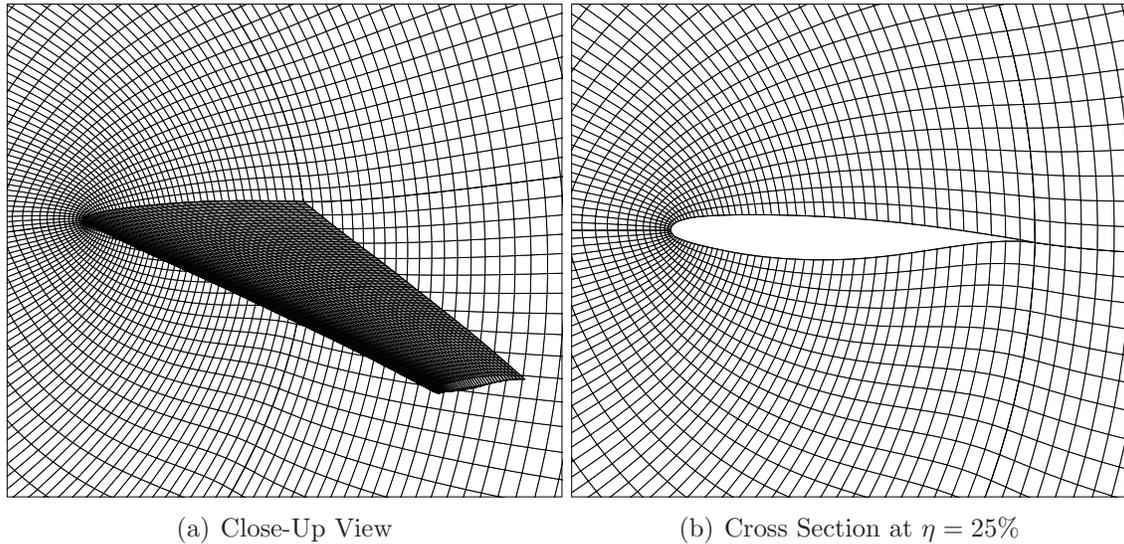


Figure 6–7: LANN Wing Mesh

blocks of  $48 \times 32 \times 32$  cells each, is employed for the computations. Figure 6–7 shows a close-up view of the wing mesh, as well as a cross section of the mesh at  $\eta = 25\%$ , where  $\eta$  is the spanwise location. In order to determine the number of harmonics to use for the solution, the calculations are first carried out with 1, 2, and 3 modes using a rigid grid motion over 2500 multigrid cycles. The flow solver convergence and drag coefficient hysteresis are displayed in Figures 6–8 and 6–9 for all solutions. As can be observed, the 2- and 3-mode drag coefficient hysteresis agree very well together, and 2 modes are therefore employed onward for the validation of the RBFV method for *Run 73*.

### 6.2.1 Solution Accuracy using the RBFV Method

The same procedure is employed for the assessment of the accuracy of the RBFV method in three dimensions as in two dimensions: the flow solution is computed using both the rigid grid motion and RBFV methods and the obtained results are

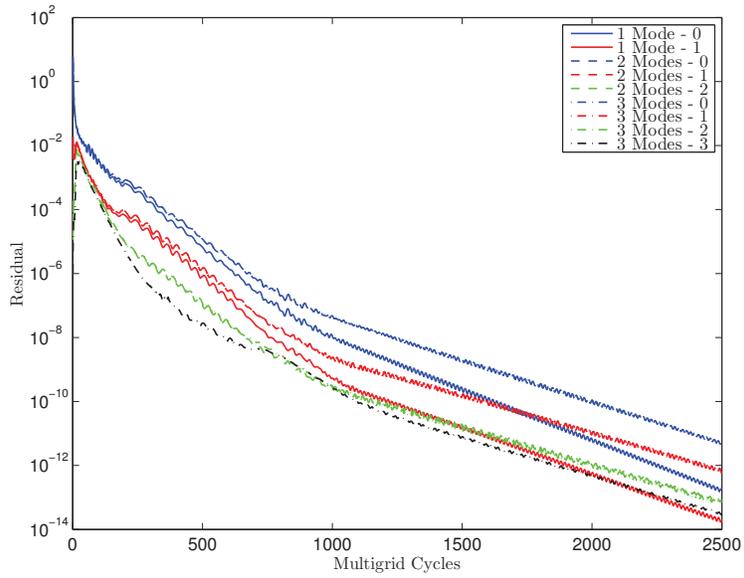


Figure 6-8: Flow Solver Convergence using Rigid Grid Motion, *Run 73*

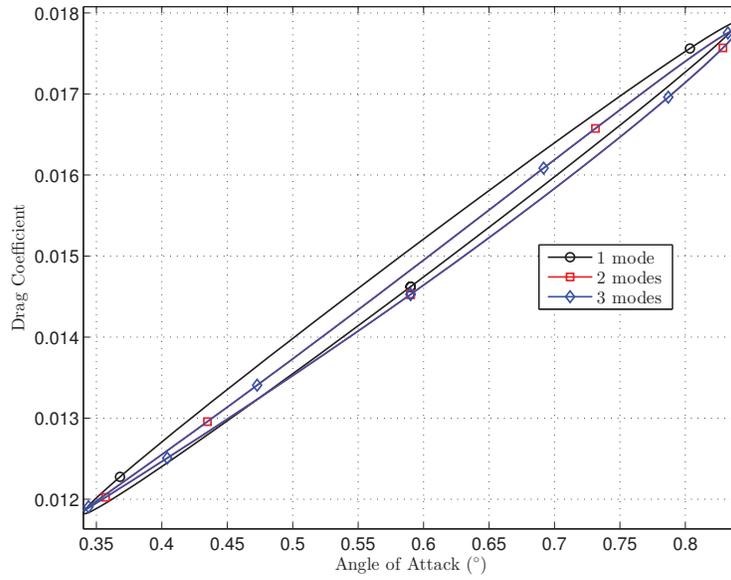


Figure 6-9: Drag Coefficient Hysteresis using Rigid Grid Motion, *Run 73*

compared together and to experimental values. The flow solver convergence using the RBFV method with 2 modes for *Run 73* is shown in Figure 6–10. As reported in Table 6–2, the Fourier representations of the lift and drag coefficients using the RBFV method show very good comparison with those of the rigid grid motion approach. In all cases, values agree up to at least the fourth digit, which is sufficient for engineering purposes. Comparison of the lift and drag coefficient hysteresis confirms the strong agreement between the two solutions, as depicted in Figure 6–11. The pressure distribution at spanwise positions  $\eta = 20\%$  and  $\eta = 65\%$  is also illustrated in Figure 6–12, and the agreement between the two methods is once again excellent. The numerical solutions predict the leading edge pressure peak accurately for both positions, but clearly mispredict the position of the shock on the upper surface of the wing. This discrepancy may partly be due to the absence of viscous effects in the flow solver. Nevertheless, the agreement between rigid- and deformable-grid solutions is excellent. This is consistent with the two-dimensional results, and the RBFV method is therefore validated. This also partly confirms the hypothesis according to which the non-satisfaction of the GCL, detailed in section 4.2, does not introduce significant errors in the solution using frequency-domain methods, at least for the selected cases.

### 6.2.2 Efficiency of the RBFV and FFTV Methods

In order to evaluate the efficiency of the RBFV method in comparison to the FFTV method in three dimensions, *Run 73* flow conditions are employed, again using 2 harmonics. Table 6–3 reports the noted computational time for both methods to execute the calculation of mesh velocities. For an equal number of modes, the

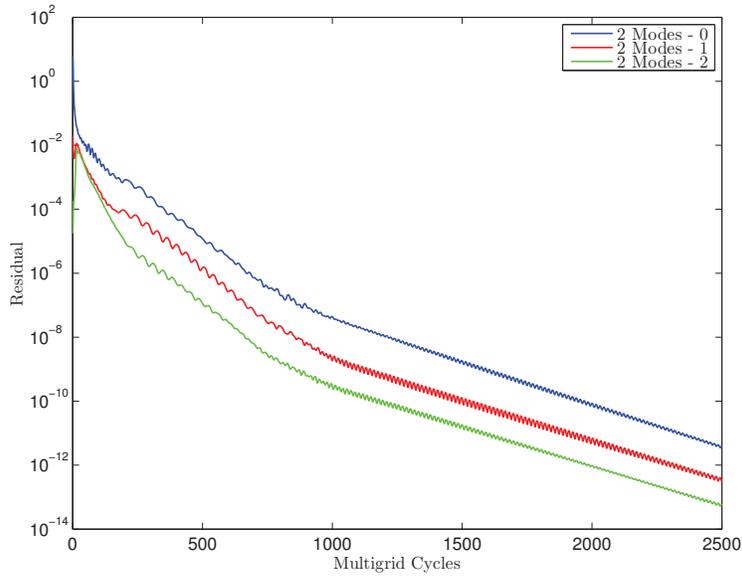
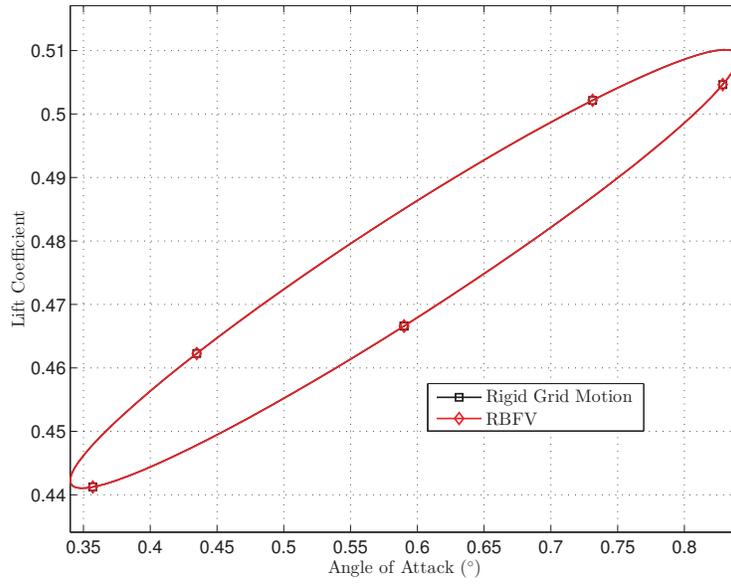


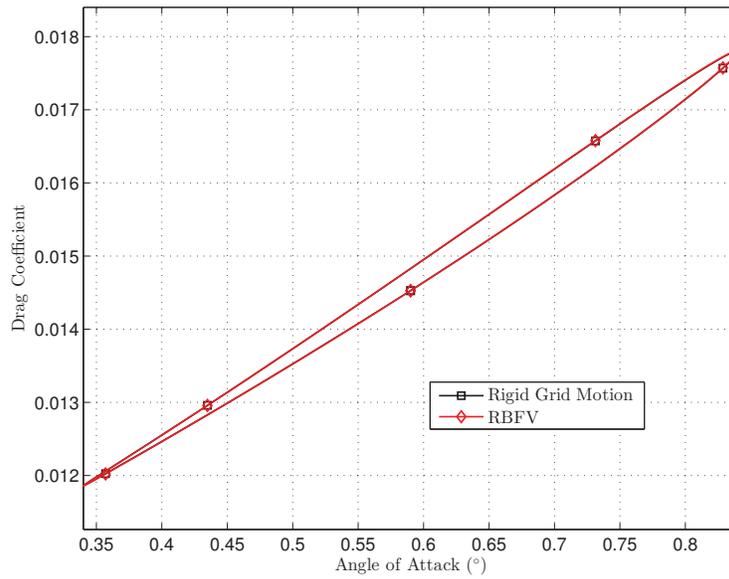
Figure 6–10: Flow Solver Convergence using the RBFV Method, *Run 73*

Table 6–2: Fourier Coefficients of Lift and Drag Coefficients using the Rigid Grid Motion and RBFV Methods for *Run 73*

Mode	Method	Lift Coefficient		Drag Coefficient	
		Real	Imaginary	Real	Imaginary
0	Rigid Grid Motion	0.475690	0.00000	0.0014751	0.00000
	RBFV	0.475691	0.00000	0.0014752	0.00000
1	Rigid Grid Motion	-0.004608	-0.016648	-0.000075	-0.001483
	RBFV	-0.004615	-0.016651	-0.000075	-0.001485
2	Rigid Grid Motion	0.000061	0.000004	-0.000037	0.000027
	RBFV	0.000062	0.000004	-0.000037	0.000027

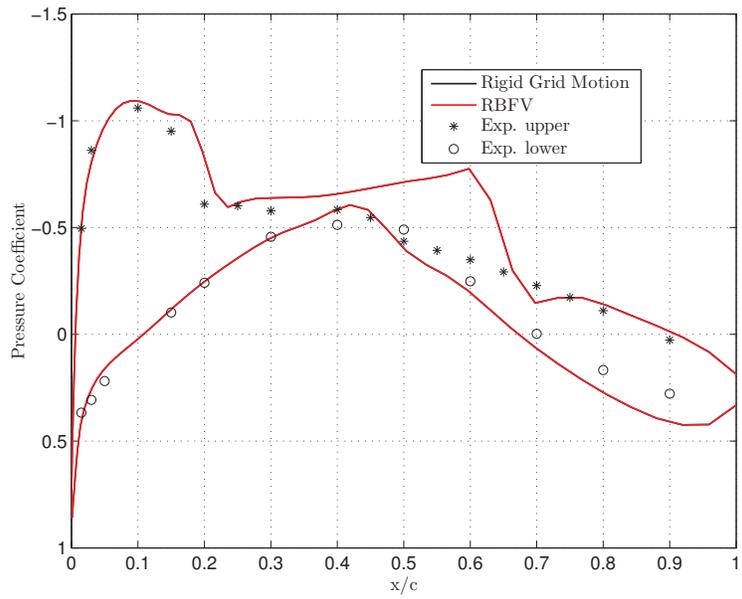


(a) Lift Coefficient Hysteresis

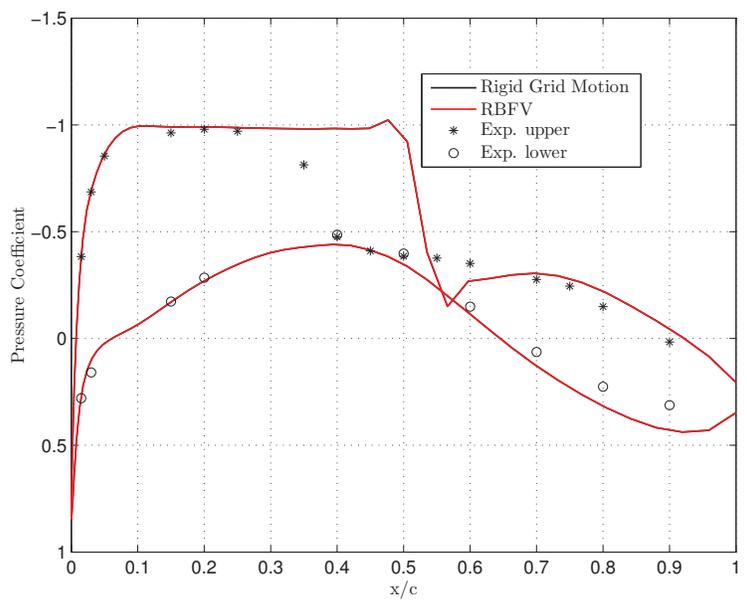


(b) Drag Coefficient Hysteresis

Figure 6–11: Comparison of the Rigid Grid Motion and RBFV Methods for the Computation of Lift and Drag Coefficient Hysteresis, *Run 73*



(a)  $\eta = 20\%$



(b)  $\eta = 65\%$

Figure 6–12: Comparison of the Rigid Grid Motion and RBFV Methods for the Computation of the Pressure Coefficient Distribution at Two Spanwise Stations,  $t = 0$ , *Run 73*

Table 6–3: Computational Cost of the FFTV and RBFV Methods for *Run 73* using 2 Modes

	CPU time (s)	CPU time (% of complete flow solution time)
FFTV method	0.04	0.0005
RBFV method	2.35	0.0288
Complete flow solution	8147.35	100.00

RBFV method is more expensive than the FFTV method by roughly two orders of magnitude. Still, both methods only require a small fraction of the total computational time for the whole flow solution, as can be seen in the rightmost column of the table. In comparison with the time required for the deformation of the mesh only (and no computation of the mesh velocities), which is approximately 60 seconds, the time required for computing the velocities is very small (2.35 seconds). This is due to the fact that the RBF operators, matrices  $[\mathbf{A}]$  and  $[\mathbf{M}]^{-1}$ , are already set up during the mesh deformation part, regardless of the method employed for the computation of the mesh velocities, be it the RBFV or the FFTV approach. Therefore, the RBFV method takes advantage of the work accomplished beforehand for the RBF deformation and is thus equivalent to a matrix-vector product only. This explains why the cost of the RBFV method is so low in comparison with the cost of the RBF deformation.

To evaluate the efficiency of the RBFV method from another point of view, it is compared to the FFTV method by taking into account the accuracy of the mesh velocity computation. Since it was shown mathematically (section 4.1.2) and numerically (Figure 6–2) that the RBFV method provides exact mesh velocities, the values computed using the FFTV technique are compared to the reference values of

the RBFV approach. Since the flow solution for *Run 73* requires 2 modes (5 time steps), it is assumed that only 5 mesh deformations are absolutely necessary. The mesh velocities are therefore evaluated at  $t = 0$  using the RBFV method with 5 time steps as a reference, and are again evaluated at  $t = 0$  using the FFTV method with 2, 3, 4, and 5 modes (5, 7, 9, and 11 time steps). The first line of Table 6–4 reports the actual time for the computation of the mesh velocities only using the FFTV method. Since as many mesh deformations as time steps are needed, the FFTV method with 7, 9, and 11 time steps requires more mesh deformations than the RBFV method, which uses only 5 time steps, since it is the minimum required for the accuracy of the flow solution. The second line of Table 6–4 therefore lists the computational time required for the additional mesh deformations for the FFTV approach that are not required for the RBFV method. The value in the first column of this line is zero since, at that point, the RBFV and FFTV require the same number of mesh deformations, that is, 5. The total time, in the third line of the table, is to be compared with the total time of the RBFV approach, whereas the fourth line lists the order of the  $L^\infty$ -norm of the error induced in the mesh velocities by the FFTV method.

Since the total computational time required for the computations of the mesh velocities using the RBFV approach is 2.35 seconds, as reported in Table 6–3, the FFTV method is faster than the RBFV method for the 2-mode deformation only, which induces an error on the order of  $\mathcal{O}(10^{-9})$ . If the flow solution was computed using 3 modes instead of two, then the 2- and 3-mode deformations would be faster for the FFTV approach than for the RBFV. It can also be seen that a minimum of 4 modes in the computation of the mesh velocities are required for the FFTV

Table 6–4: Computational Cost of the FFTV Method for Various Accuracy Levels for *Run 73*

	Number of modes			
	2	3	4	5
Pure FFTV method CPU time (s)	0.04	0.06	0.08	0.09
Required additional deformation time (s)	0.00	25.39	51.35	80.91
Total velocity computation time (s)	0.04	25.45	51.43	81.00
Order of mesh velocity error ( $L^\infty$ -norm)	$10^{-9}$	$10^{-12}$	$10^{-15}$	$10^{-15}$

method to provide results of machine-precision accuracy. In that case, the RBFV method becomes far less expensive due to the additional mesh deformations saved from computation. Nevertheless, the 2-mode FFTV shows good accuracy ( $\mathcal{O}(10^{-9})$ ) and high efficiency. Therefore, it would be needed to verify if the level of error induced in the computation of the mesh velocities using the FFTV method with 2 modes is of sufficient importance to impact the final flow solution. In conclusion, the RBFV method is an attractive alternative to the FFTV approach due to its exactitude, and because its superior requirements in terms of computational effort are not significant in comparison with the computational cost of an entire flow solution.

## CHAPTER 7

### Results - Aeroelastic Solutions

This chapter presents aeroelastic results obtained via the NLFD/LCO framework described in chapter 5, using the flow and structural solvers detailed in chapters 2 and 3, respectively.

#### 7.1 AGARD I.-Wing 445.6 Weakened Model 3

The first analyzed aeroelastic case is based on experimental results obtained by Yates et al. [91] regarding the aeroelastic response of the three-dimensional AGARD I.-Wing 445.6 Weakened Model 3. The geometry of the wing is defined as follows:

- Root chord:  $c_r = 1.833$  ft;
- Taper ratio:  $\lambda = 0.66$ ;
- Semi-span:  $\frac{b}{2} = 2.500$  ft;
- Quarter-chord sweep angle:  $\Lambda_{c/4} = 45^\circ$ ;
- Streamwise section: NACA 65A004 airfoil.

All aeroelastic computations are carried out using the root chord  $c_r$  as the reference chord for the evaluation of the reduced frequency and speed index.

##### 7.1.1 Structural Solver Validation

In order to produce unbiased aeroelastic results, the structural solver is first validated against available experimental and numerical data. The wing is discretized using 15 node stations in the spanwise and streamwise directions, for a total of 225 nodes and 392 triangular elements, as portrayed in Figure 7-1. The thickness of

each element is dictated by the wing geometry, and the density is adjusted such that the total wing mass is  $m_b = 0.12764$  slugs, as is the case in the experimental model. The orthotropic properties of the wing are modified such that its first and second natural frequencies match those of the experimental model. As a result, the material properties are set to:

- $E_{xx} = 8.690 \times 10^6$  lb/ft<sup>2</sup> (across fibre);
- $E_{yy} = 7.070 \times 10^7$  lb/ft<sup>2</sup> (along fibre);
- $G_{xy} = 8.680 \times 10^6$  lb/ft<sup>2</sup>;
- $\nu_{yx} = 0.310$ .

The wing is considered to be clamped at its root, and the fibres are oriented along the wing quarter-chord ( $y$ -direction). The first four resulting natural frequencies are listed in Table 7-1, along with those obtained from the experimental model and from Yates' numerical model [90]. Small discrepancies can be noted in the third and fourth natural frequencies, but similar differences are observed in Yates'. The structural mode shapes obtained by Yates are illustrated in Figure 7-2, whereas those obtained in the present work are displayed in Figure 7-3. Note that the non-dimensionalization of the eigenvectors is not the same in the two cases, and therefore only the contour line patterns should be compared, regardless of the contour values. The oscillations in the contour lines of the calculated mode shapes are only due to the nonconformity of the plate elements, which causes the slope of the solution to be discontinuous across element boundaries. However, it can be seen that the overall shapes agree very well with those computed by Yates, and this structural model is therefore retained for aeroelastic computations.

Table 7–1: First Four Natural Frequencies of the AGARD I.-Wing 445.6 Weakened Model 3

	Mode 1 (Hz)	Mode 2 (Hz)	Mode 3 (Hz)	Mode 4 (Hz)
Present work	9.60	38.10	49.13	91.60
Experiment [90]	9.60	38.10	50.70	98.50
Yates [90]	9.60	38.17	48.35	91.54

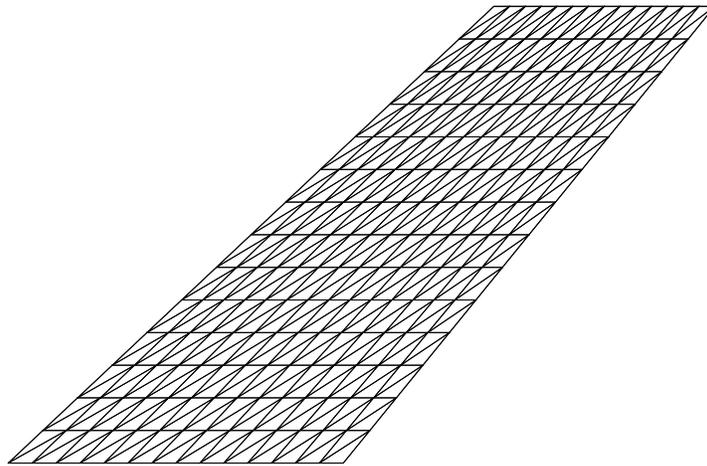


Figure 7–1: Structural Mesh for the AGARD I.-Wing 445.6

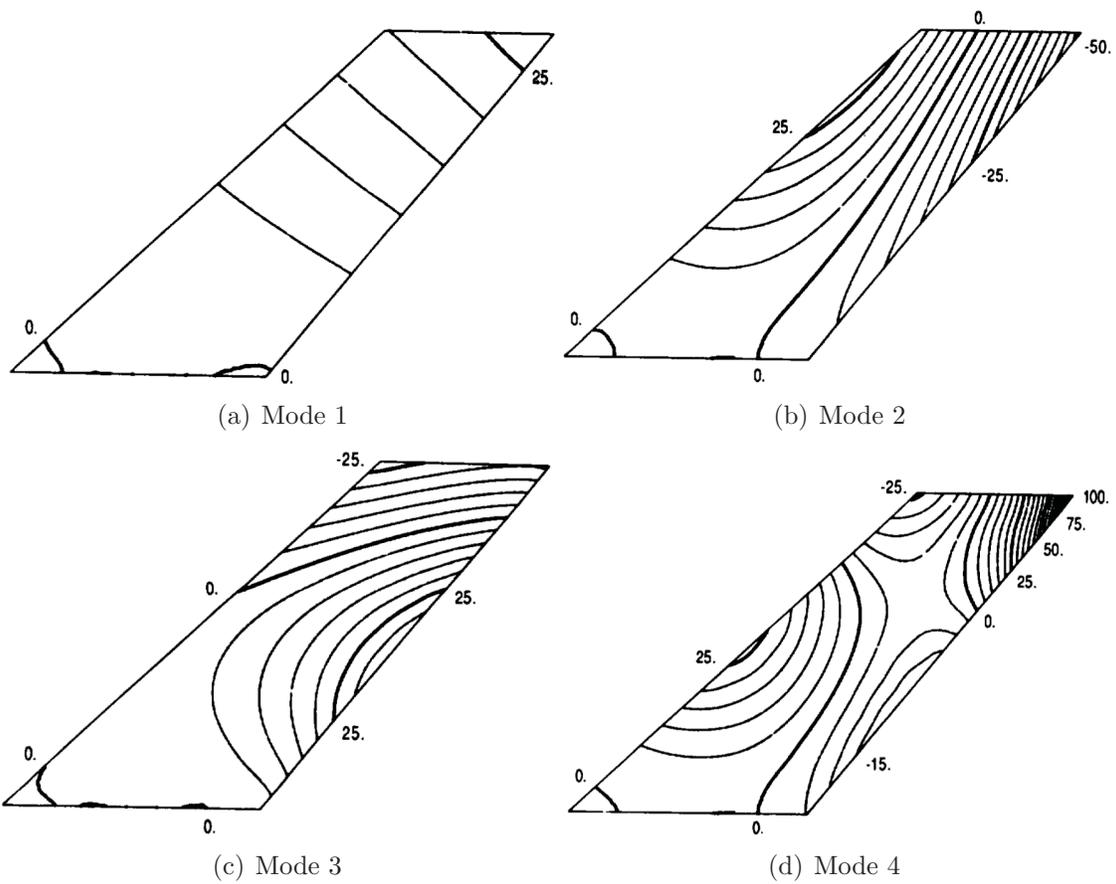
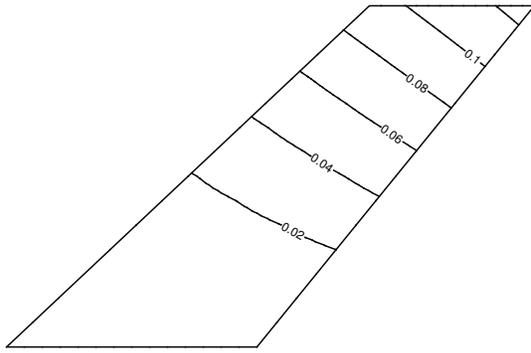
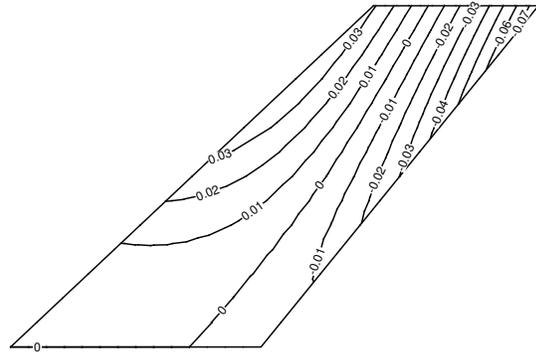


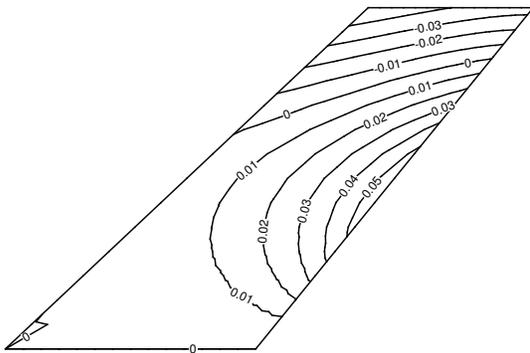
Figure 7-2: Mode Shapes of the AGARD I.-Wing 445.6 Weakened Model 3 as Calculated by Yates [90]



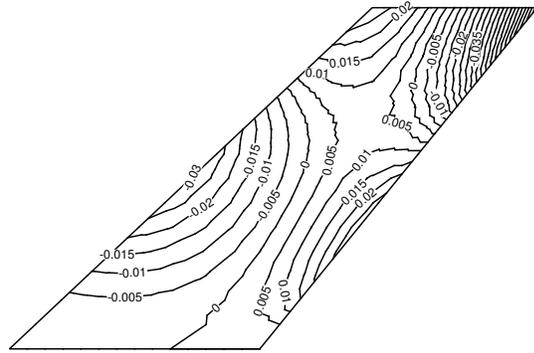
(a) Mode 1



(b) Mode 2



(c) Mode 3



(d) Mode 4

Figure 7-3: Calculated Mode Shapes of the AGARD I.-Wing 445.6 Weakened Model 3

### 7.1.2 Spatial and Temporal Accuracy Analyses

For the spatial and temporal accuracy analyses, a harmonic motion is prescribed to the wing, using the first structural mode shape as the displacement basis. It is known to the author that a thorough analysis would have required the calculation of aeroelastic results at various Mach numbers and amplitudes, but such study would have been extremely time-intensive. The wing is therefore deformed harmonically, and all structural nodes are moved in the same proportion as the first eigenvector of the structure. The fluid is air and the selected Mach number is  $M = 0.960$ , since it is the highest subsonic Mach number at which aeroelastic solutions are assessed in this work. For all cases, the amplitude of the motion is characterized by an amplitude parameter, which will be referred to as  $\delta$ . Its purpose is to quantify the displacement of the wing with solely one parameter, since detailing the amplitude of all structural degrees of freedom would severely reduce the conciseness of this work. The amplitude parameter is defined as the nondimensional average amplitude in root chord lengths of all vertical (transverse) displacements of the structural nodes, such that

$$\delta = \frac{2 \sum_{i=1}^{N_{\text{nodes}}} |\hat{Q}_{1,y}|_i}{cN_{\text{nodes}}},$$

with

$$|\hat{Q}_{1,y}|_i = \sqrt{\left(\text{Re}(\hat{Q}_{1,y})_i\right)^2 + \left(\text{Im}(\hat{Q}_{1,y})_i\right)^2},$$

where  $N_{\text{nodes}}$  is the number of nodes in the structural model,  $c$  is the reference chord, and  $\left(\hat{Q}_{1,y}\right)_i$  is the Fourier coefficient of the first mode of the vertical displacement of node  $i$ . The amplitude of any periodic wing motion can therefore be described by

this amplitude parameter,  $\delta$ . In aeroelastic cases, the amplitude parameter of the initial prescribed motion of the wing will be referred to as  $\delta_i$ .

For the accuracy studies, the amplitude parameter is set to  $\delta = 377.0 \times 10^{-5}$  which, as an indication, corresponds to a vertical displacement of the wing tip trailing edge of about 0.07 root chords. The reduced frequency is set to  $\omega_r = 0.07892$ , since it corresponds to the experimental frequency at which flutter occurs for the wing at  $M = 0.960$ . For the spatial accuracy analysis, the flow is computed on three grids: coarse ( $96 \times 32 \times 48$  cells), medium ( $192 \times 64 \times 96$  cells), and fine ( $384 \times 128 \times 192$  cells), each grid consisting of 12 blocks. To ensure that the error due to the temporal discretization is minimal, 4 modes are employed in the flow solution, which is computed over 2500 multigrid cycles. Table 7–2 reports the relevant Fourier coefficients of the lift and drag coefficients for the three grids. Since the wing is symmetric and its mean displacement is zero, the zeroth mode of the lift coefficient is very close to zero and is not reported in the table, nor is the second mode of the lift coefficient due to its very small value. On the other hand, since symmetric wings still produce drag and since two modes are necessary to accurately capture the drag coefficient hysteresis of the wing, as depicted in Figure 7–6(b), only the real parts of the zeroth and second modes of the drag coefficient are reported in the table. As a reference,  $\text{Re}(\hat{C}_{D,0})$  is synonymous with the real part of the zeroth mode of the drag coefficient. It can be seen that the medium grid provides results of good accuracy, since the difference with the fine grid for lift and drag coefficients is always within 0.0002, which is considered sufficient for aeroelastic computations. The  $192 \times 64 \times 96$

Table 7–2: Fourier Coefficients of Lift and Drag Coefficients on Various Grids for the AGARD I.-Wing 445.6 Undergoing Prescribed Oscillatory Deformation,  $M = 0.960$ ,  $\omega_r = 0.07892$ ,  $\delta = 377.0 \times 10^{-5}$

Grid	$\text{Re}(\hat{C}_{D,0})$	$\text{Re}(\hat{C}_{L,1})$	$\text{Im}(\hat{C}_{L,1})$	$\text{Re}(\hat{C}_{D,2})$
Coarse	0.0014	0.0005	0.0318	-0.0002
Medium	0.0010	0.0011	0.0309	-0.0002
Fine	0.0009	0.0013	0.0309	-0.0002

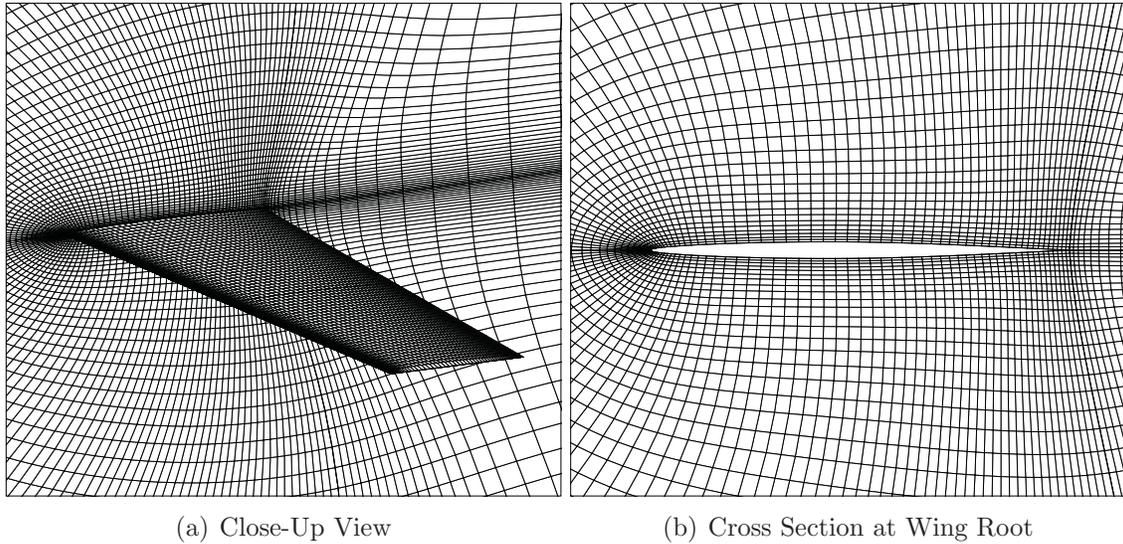


Figure 7–4: AGARD I.-Wing 445.6 Flow Solver Mesh

grid is therefore employed onward, and a close-up view as well as a cross section of the mesh are depicted in Figure 7–4.

For the temporal accuracy study, the same test case is analyzed using 1, 2, and 3 harmonics on the medium grid. The convergence of the solution for all numbers of harmonics is shown in Figure 7–5. The lift and drag coefficient hysteresis loops obtained using each number of harmonics are displayed in Figure 7–6. Pressure distributions are shown in Figure 7–7 for all cases at  $t = 0$ . It can be observed from these graphs that the 2-mode and 3-mode solutions produce very similar results, and

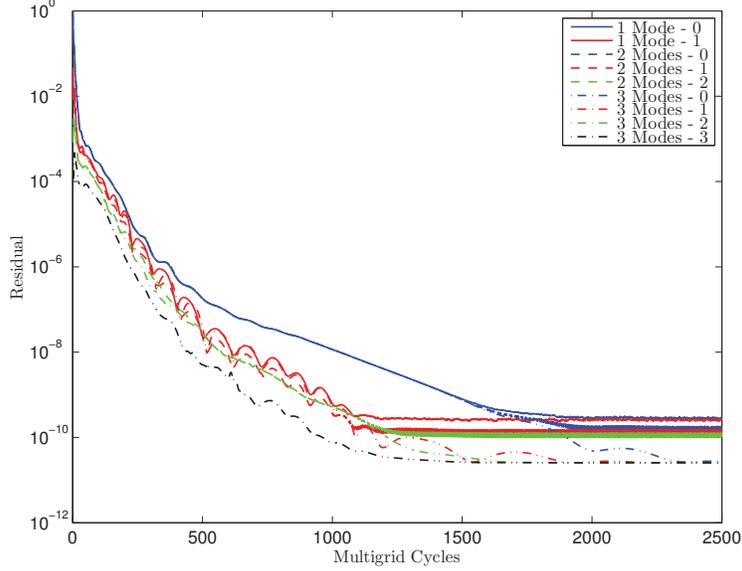
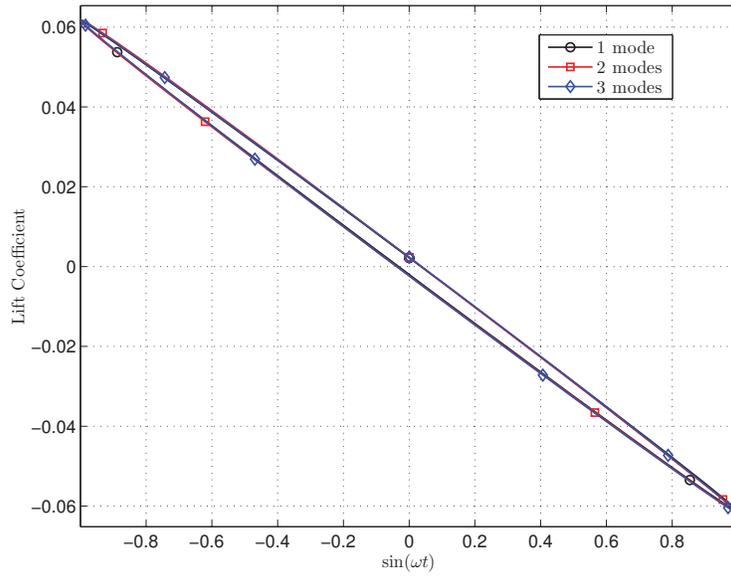


Figure 7–5: Flow Solver Convergence for the AGARD I.-Wing 445.6 Weakened Model 3 Undergoing Prescribed Oscillatory Deformation,  $M = 0.960$ ,  $\omega_r = 0.07892$ ,  $\delta = 377.0 \times 10^{-5}$

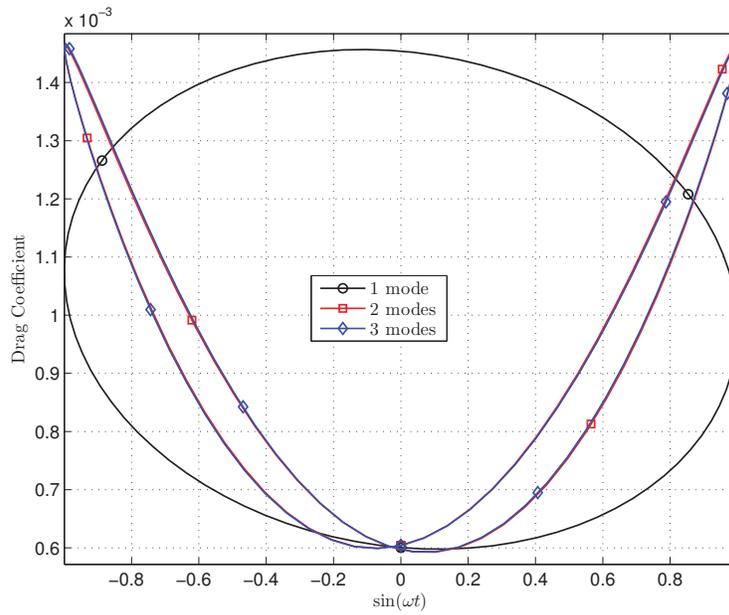
that it is thus sufficient to employ 2 modes for the flow solution in this case. It is thus assumed that 2 modes are sufficient for the entire set of aeroelastic solutions.

### 7.1.3 Flutter Results in Air

All aeroelastic solutions are initiated by imposing a sinusoidal motion to the wing proportionally to its first structural mode shape with a predefined speed index and reduced frequency, before letting it respond aeroelastically using the NLFD/LCO method described in section 5.3. Hence, to assess the flutter boundary of the Weakened Model 3 of the AGARD I.-Wing 445.6 in air, a very low amplitude parameter ( $\delta_i = 3.8 \times 10^{-5}$ ) is chosen for the initial motion of the wing, and only one mode is employed in the flow solution ( $N = 1$ ), such that the nonlinearities in the resulting aeroelastic solution are negligible. The calculations are performed at Mach

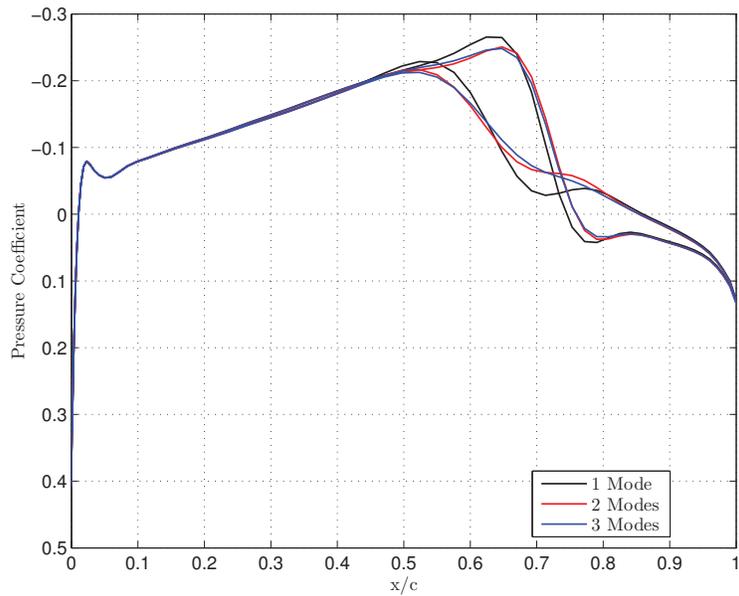


(a) Lift Coefficient Hysteresis

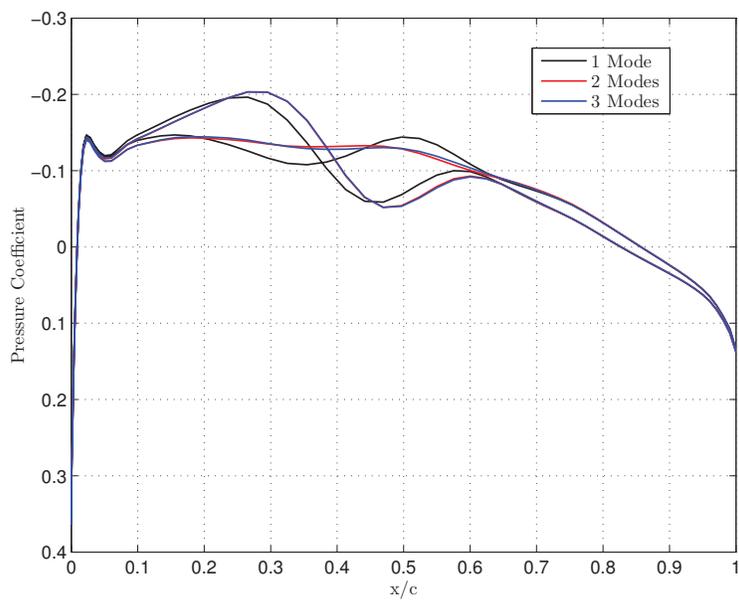


(b) Drag Coefficient Hysteresis

Figure 7–6: Lift and Drag Coefficient Hysteresis for the AGARD I.-Wing 445.6 Weakened Model 3 Undergoing Prescribed Oscillatory Deformation,  $M = 0.960$ ,  $\omega_r = 0.07892$ ,  $\delta = 377.0 \times 10^{-5}$



(a)  $\eta = 30\%$

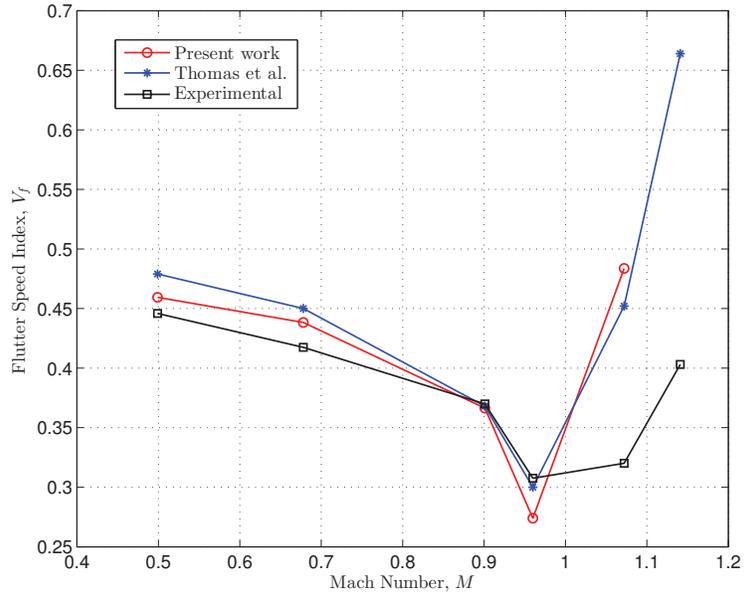


(b)  $\eta = 60\%$

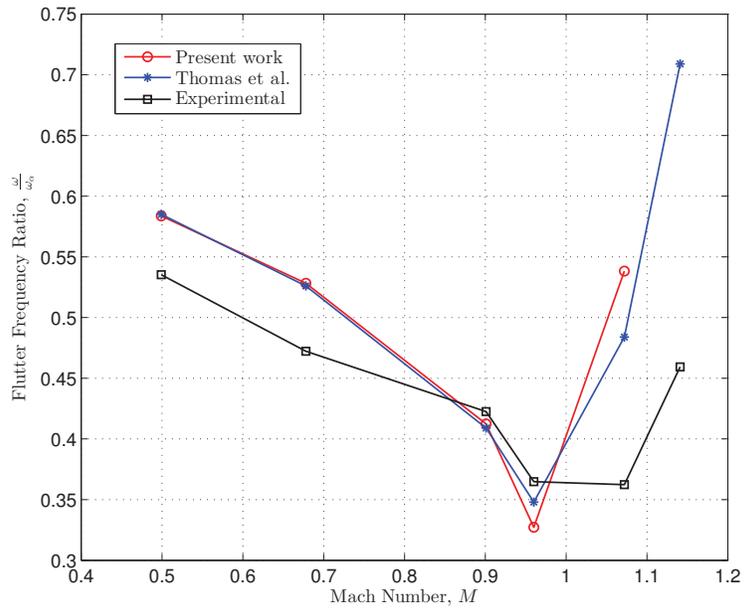
Figure 7-7: Pressure Coefficient Distribution at Two Spanwise Stations on the AGARD I.-Wing 445.6 Weakened Model 3 Undergoing Prescribed Oscillatory Deformation,  $M = 0.960$ ,  $\omega_r = 0.07892$ ,  $\delta = 377.0 \times 10^{-5}$ ,  $t = 0$

numbers 0.499, 0.678, 0.901, 0.960, and 1.072, in order to cover the entire range of transonic flow speeds. The mass ratios  $\bar{\mu}$  are set to their experimental values, reported by Yates [91]. For each case, except at  $M = 1.072$  where the initial speed index is increased to prevent the solution to fail, the initial speed indices and reduced frequencies are set to those obtained from the experimental flutter conditions. Unfortunately, in this work, no suitable solution is obtained at  $M = 1.141$  as was done by Yates et al. [91], because no stable convergence of the Newton-Raphson method could be achieved due to the sensitivity of the method to the initial guess of reduced frequency and speed index. This result is correlated by previous work, where strong discrepancies are observed at  $M = 1.141$  between the calculated solution and the experimental data [82]. A solution is nevertheless obtained for all other five Mach numbers.

All runs are carried out using  $M_p = 1$  period for the computation of the finite differences in the Newton-Raphson method, except for  $M = 0.499$ , where  $M_p = 2$  is employed for a better fluid-structure coupling ( $M_p$  being defined in section 5.3). The flow and structural solvers are coupled every  $N_{mg} = 50$  multigrid cycles. Although the wing is symmetrical, it is observed that the NLFD method can yield a nonzero value for the zeroth mode of the lift coefficient due to discretization errors, which could cause the structural solver to diverge. The zeroth mode of the body displacement vector is hence forced to zero in order to prevent such a problem. For an asymmetric wing, a static aeroelastic solution would have to be calculated first, and then the dynamic aeroelastic computations could be carried out.



(a) Flutter Speed Index Boundary



(b) Flutter Frequency Ratio Boundary

Figure 7–8: Flutter Boundary of the AGARD I.-Wing 445.6 Weakened Model 3 in Air: Calculated, Thomas et al. [82], and Experimental [91]

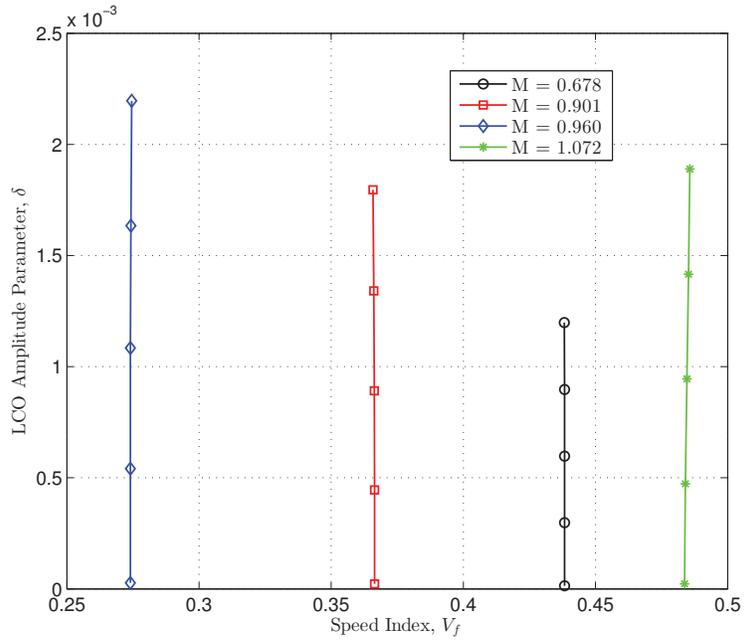
Using the proposed NLFD/LCO technique for all Mach numbers, the flutter speed index boundary is obtained, and is depicted in Figure 7–8(a) along with linear results of Thomas et al. [82] and experimental values [91]. The frequency ratio, which is the ratio of the flutter frequency to the natural frequency of the first torsional mode,  $\frac{\omega}{\omega_\alpha}$ , is also depicted in Figure 7–8(b) alongside experimental results and previous work. The results show good overall comparison against experimental values for subsonic Mach numbers, especially for the speed index, but a large discrepancy is observed at  $M = 1.072$ . This discrepancy has also been observed for Euler calculations in previous work [47, 70, 82] and is yet poorly understood by researchers. Dowell et al. [15], among others, supposed that the error could be due to wall interference or other uncertainties in the flutter test procedures. Considering viscous effects could also improve the accuracy of the numerical solutions.

#### 7.1.4 Limit Cycle Oscillation Results in Air

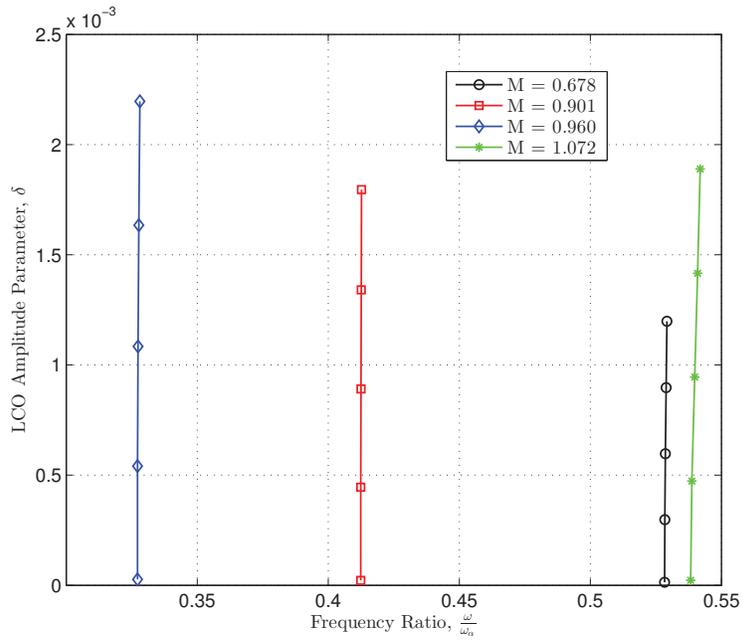
For the assessment of LCO in air, a very similar procedure to that of flutter is employed. The initial oscillation is imposed using experimental flutter reduced frequency and speed index, but the amplitude of the initial motion is varied in order to produce final LCO of different amplitudes. To achieve this goal, the initial amplitude parameter is sequentially varied such that  $\delta_i \times 10^5 = 3.8, 75.4, 150.8, 226.2,$  and  $301.6$ . Although the amplitude parameters of the converged LCO are not exactly equal to those imposed for the initial motion, it is noted that they are nevertheless proportional to it in general. All cases are run using 2 flow-solver modes and  $M_p = 1$ , except at  $M = 0.960$ , where  $M_p = 2$  is used. This allows to create the plots displayed on Figure 7–9, where the amplitude of the LCO is reported as a function of the speed

index (7–9(a)) and frequency ratio (7–9(b)) for Mach numbers  $M = 0.678, 0.901, 0.960,$  and  $1.072$ . It can be noticed that the lines of constant Mach number are almost vertical, which is consistent with the work of Thomas et al. [79]. These results denote the very low level of nonlinearity in the solution, which is expected since the wing is very thin and, therefore, no large shock motion is observed. It is important to note that the amplitude parameters are maintained low because viscous effects would need to be accounted for in oscillations of higher amplitude. Nevertheless, Thomas et al. [79] were able to obtain LCO of higher amplitude using an inviscid flow solver, and hence the LCO trends in [79] show slightly greater deviation than those presented herein. In any case, a small deviation of the  $M = 1.072$  curve to the right shows the ability of the method to model at least weak nonlinearities. It is expected that higher amplitudes and viscous effects would lead to stronger nonlinearities.

To illustrate the convergence of the Newton-Raphson method, Figure 7–10 shows the convergence of the flow solution at  $M = 0.960$  and  $M = 1.072$  with  $\delta_i = 301.6 \times 10^{-5}$ , from which the multigrid cycles necessary for the evaluation of the Jacobian have been removed during post-processing. The small jumps in the residual are due to the update of the reduced frequency and speed index during the convergence to the LCO conditions. Figure 7–11 displays the convergence of the reduced frequency and speed index at  $M = 0.960$  using the proposed Newton-Raphson method, whereas Figure 7–12 shows analogous results at  $M = 1.072$ . The errors are computed based on the final values, at the end of the computations. It is observed that the solution at  $M = 0.960$  converges almost monotonously, whereas that at  $M = 1.072$  experiences more oscillations. This behavior may be attributed to the



(a) LCO Speed Index Trends



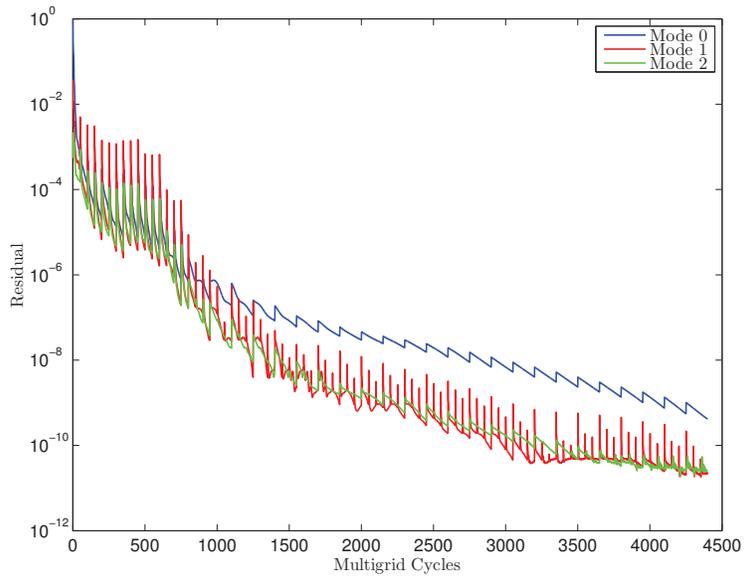
(b) LCO Frequency Ratio Trends

Figure 7-9: LCO Behavior Trends of the AGARD I.-Wing 445.6 Weakened Model 3 in Air

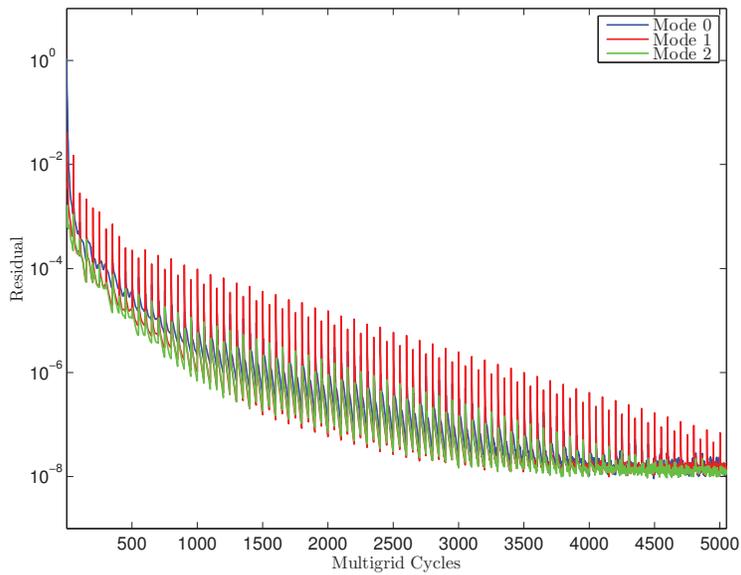
unoptimized relaxation coefficient  $\alpha$  employed in the algorithm (see equation 5.5). It may also be due to the fact that the fluid-structure coupling is better in the  $M = 0.960$  case, since it is performed with  $M_p = 2$ , therefore leading to a more accurate computation of the Jacobian. In any case, it is inferred from the error convergence plots that a line search algorithm is not necessary since the error tends to decrease at each iteration. It is also seen that the error decreases below engineering accuracy ( $10^{-4}$ ) within 5 iterations at  $M = 0.960$  and 11 iterations at  $M = 1.072$ . As an indication, 5 iterations with  $M_p = 2$  require around 3.6 hours of computations, whereas 11 iterations with  $M_p = 1$  require approximately 5 hours.

Figures 7–13 and 7–14 show examples of the time history of the lift and drag coefficients for LCO computed at  $M = 0.960$  and  $M = 1.072$ , respectively. It can be seen that both the lift and drag coefficients stabilize to steady periodic functions as the Newton-Raphson method converges, therefore confirming the presence of an LCO. Figure 7–15 displays the pressure coefficient and Mach number distributions on the surface of the wing at  $t/T = 0.25$  for the case at  $M = 1.072$ , whereas Figure 7–16 shows the streamwise pressure distribution at two different spanwise locations. Although a shock is found on the surface of the wing, its motion is observed to follow closely the wing movement; the absence of shock motion lag may therefore explain the low amount of flow nonlinearities. Although it was mentioned previously that the NLFD/LCO method using  $N_s < N$  does not guarantee energy conservation because of the different temporal discretizations of the structural and flow solvers, it is observed that the relative error in the work done by the aerodynamic loads on the wing over a period due to the removal of the highest modes from the structural

solution does not exceed 2% for all studied cases, thus confirming the validity of the approach for the analyzed cases.

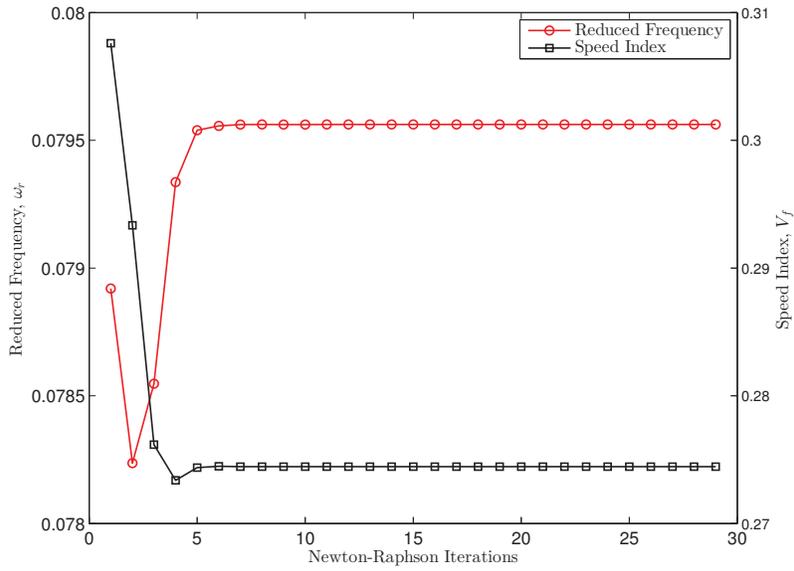


(a)  $M = 0.960$

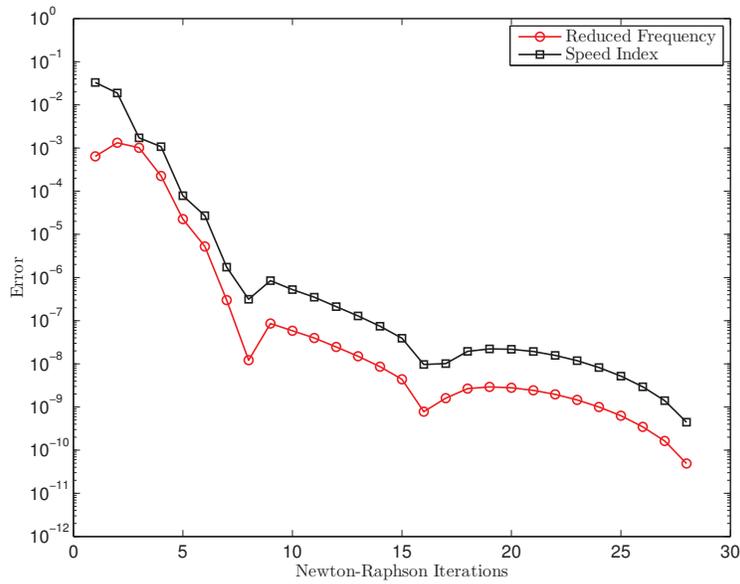


(b)  $M = 1.072$

Figure 7–10: Flow Solver Convergence for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method,  $\delta_i = 301.6 \times 10^{-5}$

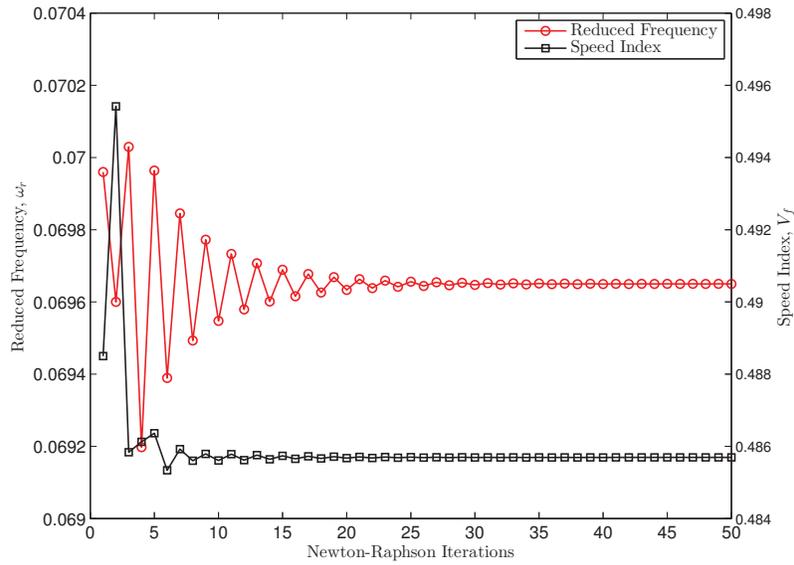


(a) Evolution of Reduced Frequency and Speed Index

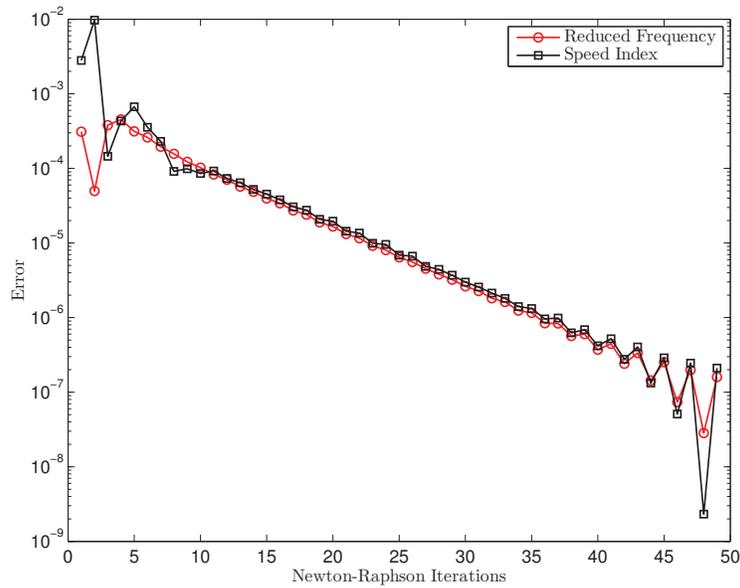


(b) Convergence of Reduced Frequency and Speed Index Error

Figure 7–11: Convergence of Reduced Frequency and Speed Index for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method,  $\delta_i = 301.6 \times 10^{-5}$ ,  $M = 0.960$

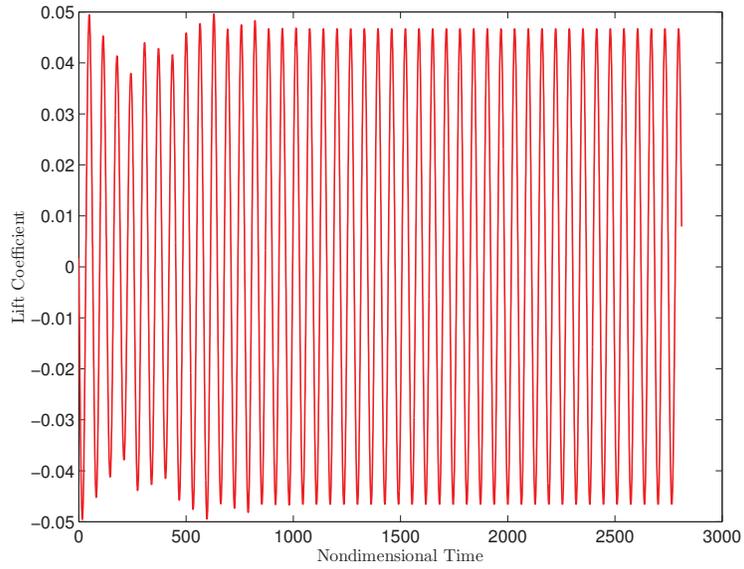


(a) Evolution of Reduced Frequency and Speed Index

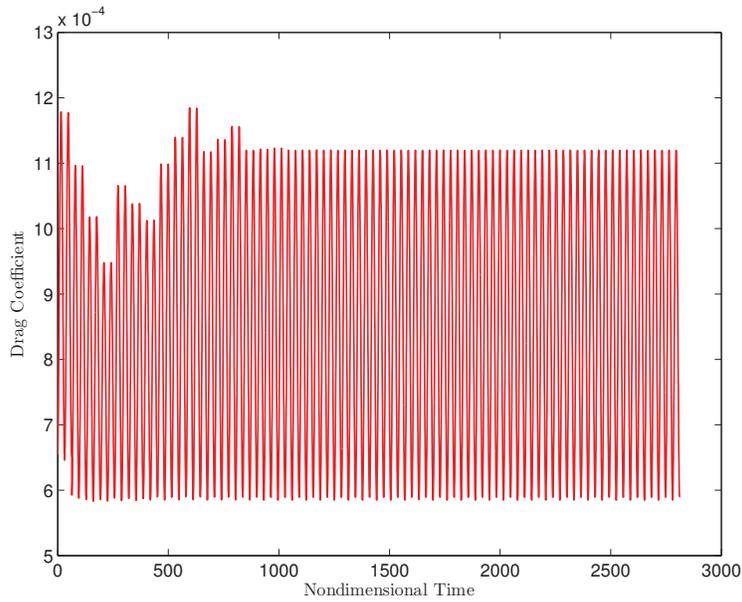


(b) Convergence of Reduced Frequency and Speed Index Error

Figure 7–12: Convergence of Reduced Frequency and Speed Index for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method,  $\delta_i = 301.6 \times 10^{-5}$ ,  $M = 1.072$

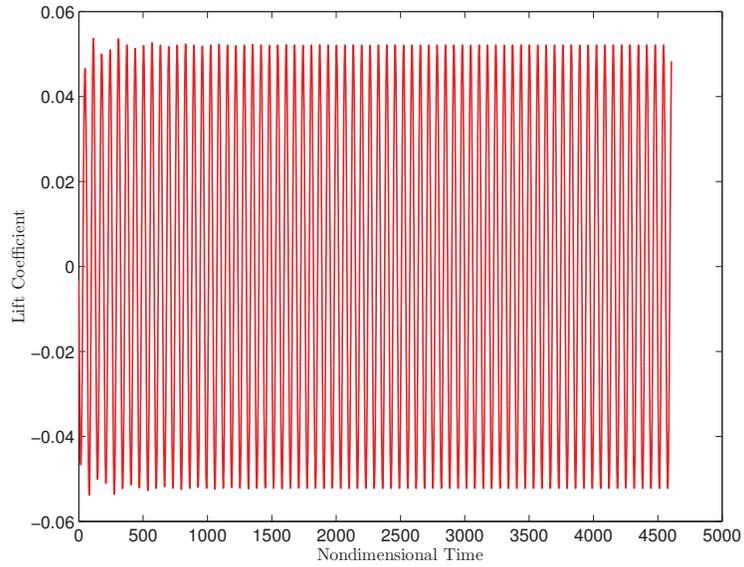


(a) Lift Coefficient Time History

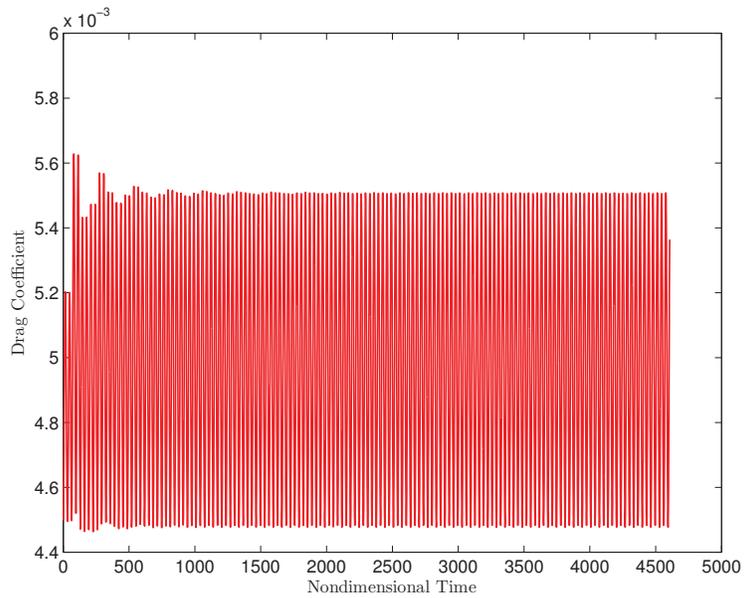


(b) Drag Coefficient Time History

Figure 7–13: Truncated Time History of the Lift and Drag Coefficients for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method,  $\delta_i = 301.6 \times 10^{-5}$ ,  $M = 0.960$

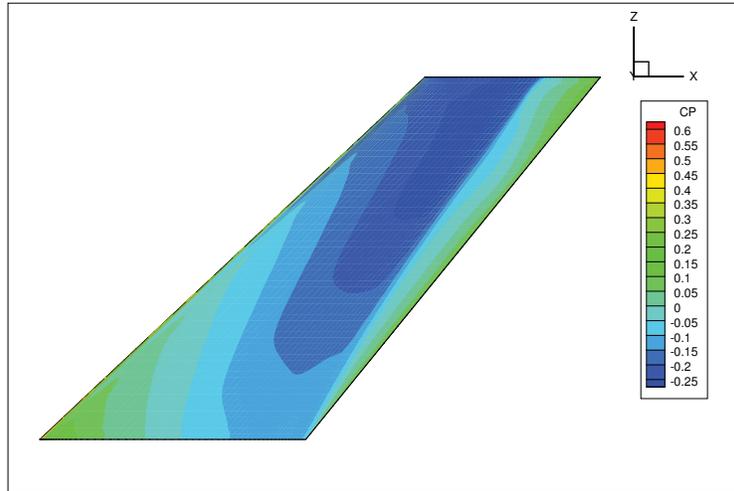


(a) Lift Coefficient Time History

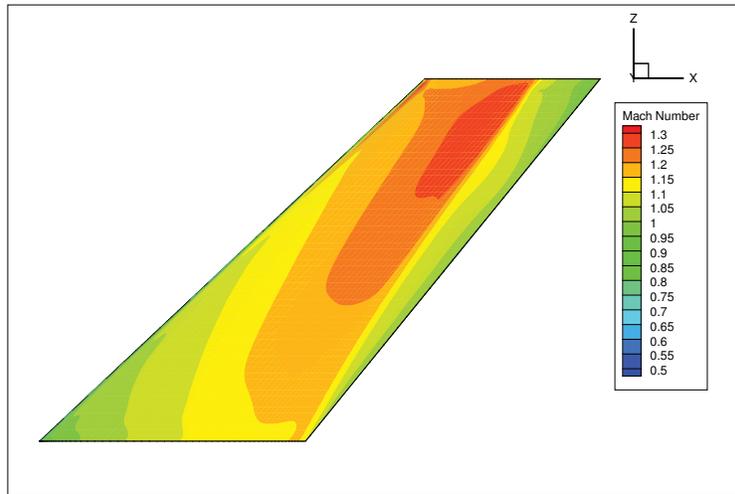


(b) Drag Coefficient Time History

Figure 7-14: Truncated Time History of the Lift and Drag Coefficients for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method,  $\delta_i = 301.6 \times 10^{-5}$ ,  $M = 1.072$

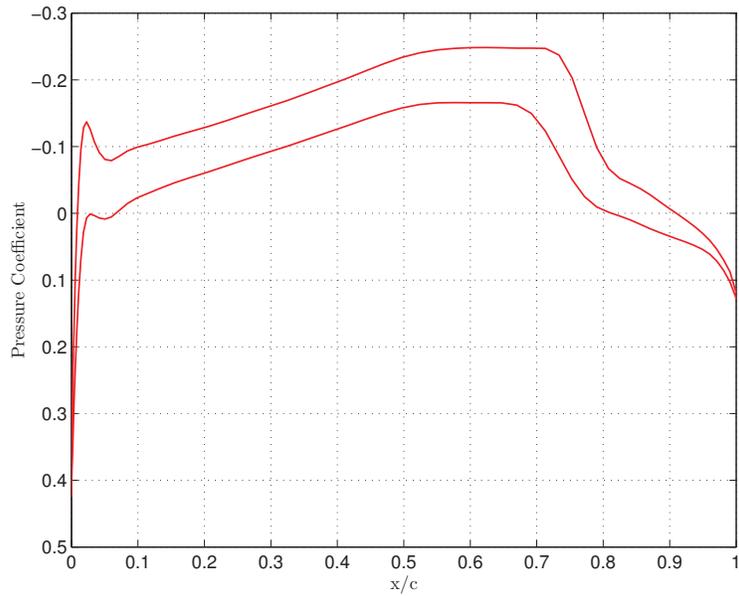


(a) Pressure Coefficient Distribution

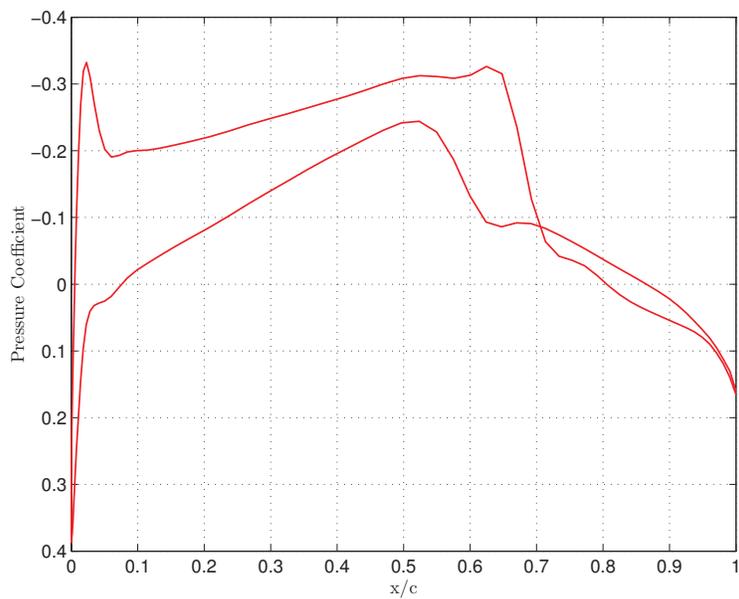


(b) Mach Number Distribution

Figure 7–15: Contour Plots for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLFD/LCO Method,  $\delta_i = 301.6 \times 10^{-5}$ ,  $M = 1.072$ ,  $t/T = 0.25$



(a)  $\eta = 60\%$



(b)  $\eta = 90\%$

Figure 7–16: Pressure Coefficient Distribution at Two Spanwise Stations for a Limit Cycle Oscillation of the AGARD I.-Wing 445.6 Weakened Model 3 in Air Obtained with the NLF/D/LCO Method,  $\delta_i = 301.6 \times 10^{-5}$ ,  $M = 1.072$ ,  $t/T = 0.25$

## 7.2 AGARD I.-Wing 445.6 Solid Model 2

In this section, the AGARD I.-Wing 445.6 Solid Model 2 [90] is considered. The geometry is identical to that of Weakened Model 3, but the material properties differ. For this model, the spatial discretization of the body is the same as that mentioned in section 7.1.1, and the density of the wing is adjusted to match the total mass of the experimental model, that is,  $m_b = 0.14658$  slugs. The structural properties of the wing are modified to enhance the match with the experimental frequencies, and are enumerated below:

- $E_{xx} = 8.680 \times 10^6$  lb/ft<sup>2</sup> (across fibre);
- $E_{yy} = 1.798 \times 10^8$  lb/ft<sup>2</sup> (along fibre);
- $G_{xy} = 1.642 \times 10^7$  lb/ft<sup>2</sup>;
- $\nu_{yx} = 0.310$ .

For conciseness, only the natural frequencies are compared to the experiment and to the calculations of Yates [90], without comparing the associated mode shapes. Considering the very good agreement observed for Weakened Model 3, it is assumed that the structural mode shapes of Solid Model 2 also agree very well with those of Yates. It is observed, from Table 7-3, that the natural frequencies match well those obtained experimentally.

It is known to the author that a complete spatial and temporal accuracy analysis would need to be carried out for Solid Model 2 for a thorough validation of the code. However, since the same geometry is employed, the previously performed analyses on Weakened Model 3 are considered sufficient for the purposes of this work. Therefore,

Table 7–3: First Four Natural Frequencies of the AGARD I.-Wing 445.6 Solid Model 2

	Mode 1 (Hz)	Mode 2 (Hz)	Mode 3 (Hz)	Mode 4 (Hz)
Present work	14.10	50.70	69.87	122.37
Experiment [90]	14.10	50.70	69.30	127.10
Yates [90]	14.12	50.91	68.94	122.26

a  $192 \times 64 \times 96$  mesh and 2 modes are used in the flow solver for all LCO computations concerning Solid Model 2, in accordance with the results obtained in section 7.1.2.

### 7.2.1 Flutter Results in R-12

The procedure employed for the flutter results of Solid Model 2 is identical in all points to that of Weakened Model 3. However, since the experimental tests are carried out in R-12<sup>1</sup>, the value of the specific heat ratio of the fluid is set to  $\gamma = 1.1389$ . Solutions are obtained at Mach numbers 0.870, 0.920, 0.960 and 1.020. All other parameters, such as the mass ratio, are set to their experimental values. The motion is again initiated using the shape of the first eigenvector of the wing, with an amplitude parameter of  $\delta_i = 3.8 \times 10^{-5}$ , at the experimental flutter reduced frequency and speed index. The NLFD/LCO technique is then employed to converge to neutral oscillations, using 1 mode for both the flow and structural solvers. The obtained flutter boundaries for the speed index and frequency ratio are depicted in Figure 7–17. Small discrepancies are observed between calculated and experimental results, part of which may be explained by the absence of viscous effects from the

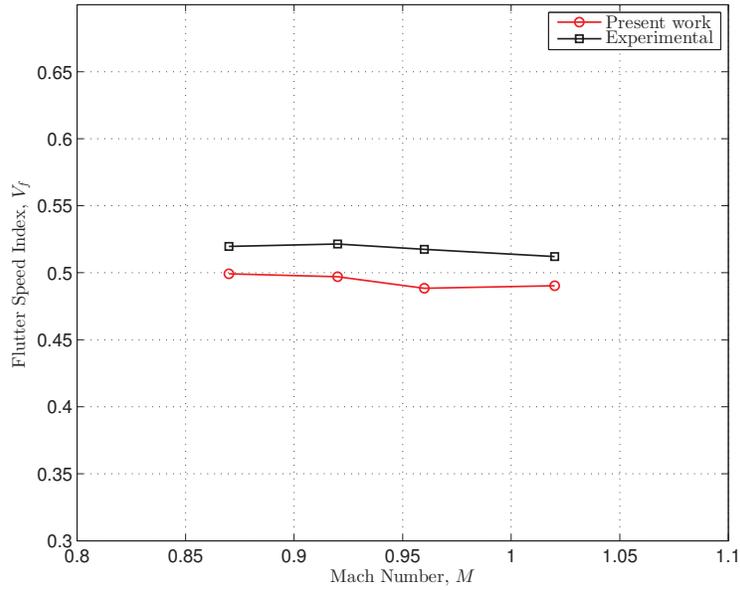
---

<sup>1</sup> Dichlorodifluoromethane, commonly referred to as Freon<sup>®</sup>-12, commercialized by DuPont<sup>®</sup>.

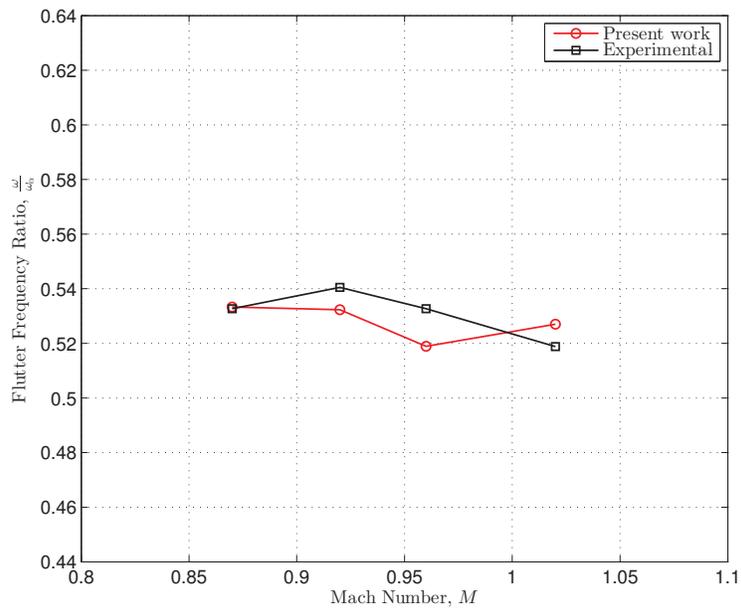
computations. Nevertheless, the general trends are consistent with those obtained experimentally.

### 7.2.2 Limit Cycle Oscillation Results in R-12

The procedure employed herein is identical to that described in section 7.1.4. The amplitude is set sequentially such that  $\delta_i \times 10^5 = 3.8, 75.5, 151.1, 226.6,$  and  $302.1$  to produce LCO of varying amplitude, and 2 modes are employed in the flow solver. Simulations are performed for  $M = 0.870, 0.920,$  and  $0.960$ . Figure 7–18 illustrates that nonlinearities are stronger in this case, as denoted by the slight deviation of the curves of constant Mach number from the vertical. This phenomenon is especially obvious in Figure 7–18(b), where all curves tend to bend to the right. Moreover, a detrimental nonlinearity is observed in Figure 7–18(a) at  $M = 0.920$ , where the curve tends to bend slightly to the left. This therefore demonstrates the ability of the NLFD/LCO method to model nonlinearities, and specifically justifies its employment for the computation of detrimental nonlinearities, which may lead to a reduction of design safety margins. It is again expected that the nonlinear behavior of the wing be more evident for LCO of higher amplitudes, and for cases where viscous effects are considered.

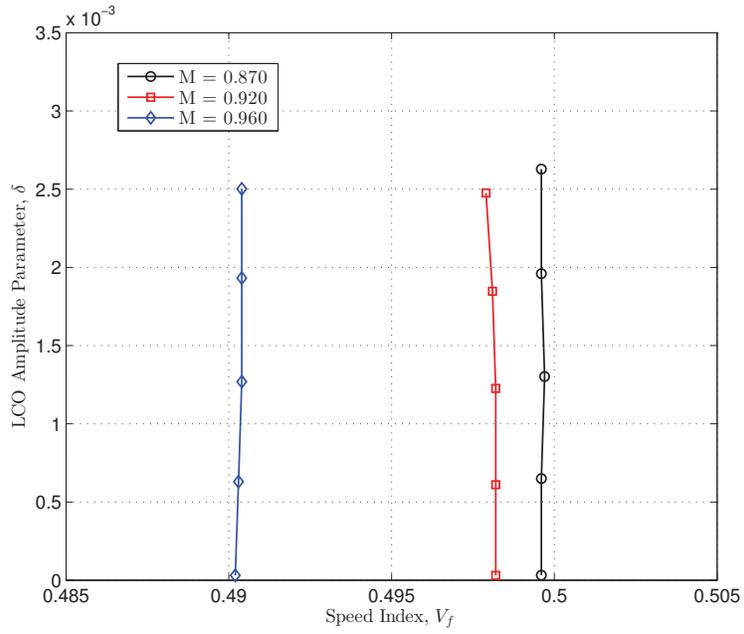


(a) Flutter Speed Index Boundary

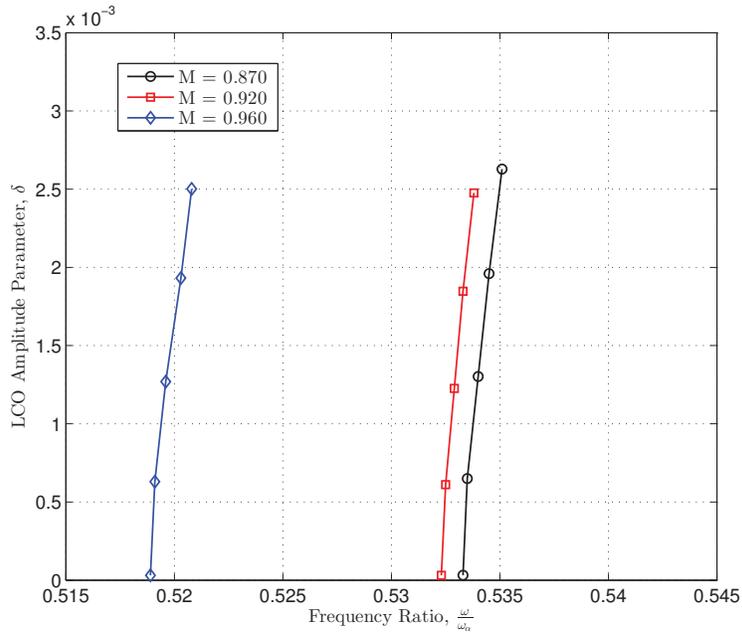


(b) Flutter Frequency Ratio Boundary

Figure 7–17: Flutter Boundary of the AGARD I.-Wing 445.6 Solid Model 2 in R-12: Calculated and Experimental [91]



(a) LCO Speed Index Trends



(b) LCO Frequency Ratio Trends

Figure 7-18: LCO Behavior Trends of the AGARD I.-Wing 445.6 Solid Model 2 in R-12

### 7.3 Estimated Acceleration

This section intends to provide an estimate of the computational savings resulting from the use of the proposed NLFD/LCO methodology in comparison with a typical time-marching technique. The comparison detailed herein assumes that the frequency of oscillations sets automatically in time-accurate aeroelastic computations, and therefore that a one-dimensional Newton-Raphson method is necessary for the determination of the LCO speed index only. In addition, it assumes that the convergence rate of the proposed two-dimensional Newton-Raphson technique is similar to that of its one-dimensional counterpart, and that a time-accurate flow solver converges at a similar rate to that of an NLFD flow solver. In order to get a conservative estimate, it is assumed that  $M_p = 2$  is employed for the computation of the Jacobian of the NLFD/LCO method. Therefore, 7 time periods are required for the computation of one Newton-Raphson iteration: 2 for the evaluation of the reference  $\phi_1$  and  $\phi_2$  (see section 5.3), 2 for the evaluation of  $\frac{\partial \phi_i}{\partial \omega_r}$ , 2 for the evaluation of  $\frac{\partial \phi_i}{\partial V_f}$ , and 1 additional period to allow a good fluid-structure coupling between iterations. Analogously, 12 time periods are required for one Newton-Raphson iteration in a time-marching framework: 6 for the reference  $\phi_1$  and  $\phi_2$ , and 6 for the computation of  $\frac{\partial \phi_i}{\partial V_f}$ . This number of periods (6) intends to be conservative as it stems from the computations of Kachra and Nadarajah [43], in which 8 periods were used for the assessment of the aeroelastic behavior of a two-dimensional airfoil. The comparison is done between a 2-mode (5-time-step) NLFD solution and a 36-time-step time-accurate solution, which was considered sufficient by Kachra and Nadarajah. The computational cost of grid deformation is not accounted for, since

the best-suited algorithm may differ from one approach to the other. As reported in Table 7–4, where NR is synonymous with Newton-Raphson, the total cost per Newton-Raphson iteration is more than 12 times higher in a time-accurate framework than using the proposed NLFD/LCO methodology. It can therefore be stated that the proposed methodology is about an order of magnitude faster than a typical time-marching technique.

No quantitative comparison is offered with the HB/LCO method proposed by Thomas et al. [79], since the difference between the two Newton-Raphson methods might lead to different convergence rates. Nevertheless, for equal Newton-Raphson convergence rates, it is supposed that the NLFD/LCO approach is computationally less costly, since it only requires the evaluation of a  $2 \times 2$  Jacobian, whereas the HB/LCO technique requires the construction of a  $2N_{ms} \times 2N_{ms}$  Jacobian, where  $N_{ms}$  is the number of structural mode shapes employed in the modal structural solver. Since the derivatives constituting the Jacobian are generally evaluated through finite-differencing, reducing the size of the Jacobian can lead to significant computational cost savings. Moreover, the method proposed in the present work does not require any convergence study to be performed with respect to  $N_{ms}$ , unlike the HB/LCO approach. It is therefore expected that the NLFD/LCO method is computationally advantageous.

Table 7–4: Comparison of the Estimated Computational Cost of the NLFD/LCO Method and of a Time-Marching Approach for the Determination of Limit Cycle Oscillation Flow Conditions

Element	NLFD/LCO Method	Time-Marching Method
Basic time-step cost	1	1
Time steps per period	5	36
Cost of FFT [43]	1.02	1
Periods per NR iteration	7	12
Total cost per NR iteration	35.7	432

## CHAPTER 8

### Conclusions

#### 8.1 Summary

The assessment of aeroelastic nonlinearities of aircraft designs is important due to the reduction of safety margins they can engender. Typical time-marching techniques enable such calculations but require intensive computational time, whereas frequency-domain approaches were shown to provide accurate results in an efficient fashion. However, frequency-domain approaches were either employed using fixed flow conditions, or using suboptimal methods for the computation of the conditions leading to periodic aeroelastic behavior. In addition, three-dimensional aeroelastic calculations require the use of some deformable grid approach, in which the efficient computation of accurate mesh velocities may be challenging for frequency-domain techniques. Based on this premise, following is a summary of the contributions and conclusions resulting from this research.

- A simple and accurate dynamic mesh deformation framework is developed for the NLFD method using an RBF algorithm. The accuracy of the proposed method for computing mesh velocities, that is, the RBFV method, is demonstrated to be independent of the number of harmonics employed in the NLFD solver, unlike the FFTV method. Flow solutions are shown for a pitching NACA 0012 airfoil and a pitching LANN wing.

- The RBFV method is observed to be roughly twice as computationally expensive as the FFTV method in two dimensions, and two orders of magnitude more expensive in three-dimensions, but nevertheless represents a very small portion of the computational time required for the entire flow solution.
- An NLFD/LCO methodology for the determination of flow conditions leading to LCO of an aircraft wing using a linear plate bending structural solver and an NLFD flow solver is proposed, using a root-finding Newton-Raphson approach in order to find the equilibrium point where LCO occur.
- The proposed methodology is validated for the AGARD I.-Wing 445.6 Weakened Model 3 in air and Solid Model 2 in R-12, where the flutter boundary and the LCO trends of the wing are calculated. Results show that the proposed technique generally converges to engineering precision within 11 Newton-Raphson iterations.
- The proposed NLFD/LCO method is estimated to perform LCO computations an order of magnitude faster than a typical aeroelastic time-marching approach. No quantitative comparison is offered with the HB/LCO method [79] but, according to a qualitative analysis, it is expected that the proposed technique is computationally more efficient.

## 8.2 Limitations

Below are listed the limitations regarded by the author as the most important of this research.

- The Geometric Conservation Law is not satisfied using the proposed dynamic mesh deformation framework, and may thus lead to a reduction in the order of accuracy of the method.
- Only LCO of small amplitude are studied in the context of this research due to the employed inviscid flow solver. Little nonlinear behavior is therefore obtained since higher amplitudes and viscous effects would account for flow nonlinearities.
- Dynamic structural nonlinearities may not yet be modelled using the employed technique since only one harmonic is employed in the discretization of the structural equations of motion.
- The temporal discretization interface between the flow and structural solvers is not conservative on the point of view of the work done by the aerodynamic forces over a period. Although the work loss is observed to be less than 2% for the analyzed cases, this assumption may not hold for other cases where the higher harmonics of the flow solution are stronger.
- The amplitude of the final LCO cannot be selected as an input to the method. The determination of LCO trends rely on the assumption that the amplitude of the final LCO is proportional to the amplitude of the initial oscillation prescribed to the structure. Trends may therefore be observed, but LCO of specified amplitudes are difficult to obtain.
- The method is only capable of assessing periodic LCO. Other types of LCO, in which the total energy of the system is constant but the motion is not periodic, cannot be modelled by the proposed approach.

### 8.3 Future Work

According to the objectives that could be reached in this research and the limitations imposed by the developed methods, following is a list of the envisioned future work related to this research.

- Develop an approach that guarantees the satisfaction of the GCL using the RBFV method for the computation of unsteady flows with the NLFD method on deforming grids, as discussed in section 4.2.
- Perform viscous calculations to allow oscillations of greater amplitude to be modelled and observe stronger nonlinear aeroelastic behavior.
- Employ a fully nonlinear structural solver, such that no assumption has to be made regarding the work lost by discarding the higher harmonics of the flow solution during the fluid-structure coupling.
- Extend the proposed LCO-determination method to a fully nonlinear structural solver, in which all modes of the solution converge to a constant value in order to obtain a perfect LCO.
- Include the aeroelastic solver inside an aerodynamic shape optimization framework in order to use the aeroelastic properties of aircraft components as design constraints.

## REFERENCES

- [1] J. J. Adamczyk. Model equation for simulating flows in multistage turbomachinery. Technical Report TM 86869, NASA, 1984.
- [2] J. J. Alonso and A. Jameson. Fully-implicit time-marching aeroelastic solutions. In *32<sup>nd</sup> AIAA Aerospace Sciences Meeting and Exhibit*, number 94-0056, Reno, NV, January 1994.
- [3] M. Frigo and. A Fast Fourier Transform compiler. In *Proceedings of the 1999 ACM SIGPLAN Conference on Programming Language Design and Implementation*, Atlanta, GA, May 1999.
- [4] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999.
- [5] P. Z. Bar-Yoseph, S. Mereu, S. Chippada, and V. J. Kalro. Automatic monitoring of element shape quality in 2-D and 3-D computational mesh dynamics. *Computational Mechanics*, 27(5):378–395, 2001.
- [6] J. Batina. Unsteady Euler airfoil solutions using unstructured dynamic meshes. *AIAA Journal*, 28(8):1381–1388, August 1990.
- [7] J. Batina. Unsteady Euler algorithm with unstructured dynamic mesh for complex-aircraft aerodynamic analysis. *AIAA Journal*, 29(3):327–333, 1991.
- [8] J. Blazek. *Computational Fluid Dynamics: Principles and Applications*. Elsevier, 2001.
- [9] C. L. Bottasso, D. Detomi, and R. Serra. The ball-vertex method: a new simple spring analogy method for unstructured dynamic meshes. *Computer Methods in Applied Mechanics and Engineering*, 194(39-41):4244 – 4264, 2005.

- [10] G. W. Burgreen and O. Baysal. Three-dimensional aerodynamics shape optimization using discrete sensitivity analysis. *AIAA Journal*, 34(9):1761–1770, 1996.
- [11] G. W. Burgreen, O. Baysal, and M. E. Eleshaky. Improving the efficiency of aerodynamic shape optimization. *AIAA Journal*, 32(1):69–76, 1996.
- [12] G. R. Cowper. Gaussian quadrature formulas for triangles. *International Journal for Numerical Methods in Engineering*, 7(3):405–408, June 2005.
- [13] A. de Boer, M. S. van der Schoot, and H. Bijl. Mesh deformation based on radial basis function interpolation. *Computers & Structures*, 85:784–795, March 2007.
- [14] C. Degand and C. Farhat. A three-dimensional torsional spring analogy method for unstructured dynamic meshes. *Computers and Structures*, 80(3-4):305–316, 2002.
- [15] E. H. Dowell, R. Clark, D. Cox, H. C. Curtiss Jr., J. W. Edwards, K. C. Hall, D. A. Peters, R. Scanlan, E. Simiu, F. Sisto, and T. W. Strganac. *A Modern Course in Aeroelasticity*. Kluwer Academic Publishers, 2005.
- [16] E. H. Dowell, J. P. Thomas, and K. C. Hall. Theoretical predictions of F-16 fighter limit cycle oscillations for flight flutter testing. *Journal of Aircraft*, 46(5):1667–1672, September-October 2009.
- [17] G. Dufour, F. Sicot, and G. Puigt. Contrasting the harmonic balance and linearized methods for oscillating-flap simulations. *AIAA Journal*, 48(4):788–797, April 2010.
- [18] K. Ekici and K. C. Hall. Nonlinear analysis of unsteady flows in multistage turbomachines using harmonic balance. *AIAA Journal*, 45(5):1047–1057, May 2007.
- [19] K. Ekici and K. C. Hall. Harmonic balance analysis of limit cycle oscillations in turbomachinery. *AIAA Journal*, 49(7):1478–1487, July 2011.
- [20] J. Elliot and J. Peraire. Aerodynamic design using unstructured meshes. In *27<sup>th</sup> AIAA Fluid Dynamics Conference*, number 1996-1941, New Orleans, LA, June 1996.

- [21] L. E. Eriksson. Generation of boundary-conforming grids around wing-body configurations using transfinite interpolation. *AIAA Journal*, 20(10):1313–1320, 1982.
- [22] O. Estruch, O. Lehmkuhl, R. Borrell, C. D. Pérez Segarra, and A. Oliva. A parallel radial basis function interpolation method for unstructured dynamic meshes. *Computers & Fluids*, 80:44–54, 2013. Selected contributions of the 23rd International Conference on Parallel Fluid Dynamics ParCFD2011.
- [23] C. Farhat. CFD on moving grids: from theory to realistic flutter, maneuvering, and multidisciplinary optimization. *International Journal of Computational Fluid Dynamics*, 19(8):595–603, 2005.
- [24] C. Farhat, C. Degand, B. Koobus, and M. Lesoinne. Torsional springs for two-dimensional dynamic unstructured fluid meshes. *Computer Methods in Applied Mechanics and Engineering*, 163:231–245, 1998.
- [25] A. Gopinath and A. Jameson. Time spectral method for periodic unsteady computations over two- and three-dimensional bodies. In *43<sup>rd</sup> AIAA Aerospace Sciences Meeting and Exhibit*, number 2005-1220, Reno, NV, January 2005.
- [26] A. K. Gopinath, E. van der Weide, J. J. Alonso, A. Jameson, K. Ekici, and K. C. Hall. Three-dimensional unsteady multi-stage turbomachinery simulations using the harmonic balance technique. In *45<sup>th</sup> AIAA Aerospace Sciences Meeting and Exhibit*, number 2007-892, Reno, NV, January 2007.
- [27] K. C. Hall and E. F. Crawley. Calculation of unstead flows in turbomachinery using the linearized Euler equations. *AIAA Journal*, 27(6):777–787, June 1989.
- [28] K. C. Hall, J. P. Thomas, and W. S. Clark. Computation of unsteady nonlinear flows in cascades using a harmonic balance technique. In *9<sup>th</sup> International Symposium on Unsteady Aerodynamics, Aeroacoustics and Aeroelasticity of Turbomachines*, Lyon, France, September 2000.
- [29] H. J. Hassig. An approximate true damping solution of the flutter equation by determinant iteration. *Journal of Aircraft*, 8(11):885–889, November 1971.
- [30] L. He. Harmonic solution of unsteady flow around blades with separation. *AIAA Journal*, 46(6):1299–1307, June 2008.

- [31] L. He, T. Chen, R. G. Wells, Y. S. Li, and W. Ning. Analysis of rotor-rotor and stator-rotor interferences in mutli-stage turbomachines. *Journal of Turbomachinery*, 124(4):564–571, November 2002.
- [32] L. He and W. Ning. Efficient approach for analysis of unsteady viscous flows in turbomachines. *AIAA Journal*, 36(11):2005–2012, November 1998.
- [33] C. W. Hirt, A. A. Amsden, and J. L. Cook. An Arbitrary Lagrangian-Eulerian computing method for all flow speeds. *Journal of Computational Physics*, 14:227–253, 1974.
- [34] H. Huang and K. Ekici. An efficient harmonic balance method for unsteady flows in cascades. *Aerospace Science and Technology*, 29(1):144–154, 2013.
- [35] S. Jakobsson and O. Amoignon. Mesh deformation using radial basis functions for gradient-based aerodynamic shape optimization. *Computers & Fluids*, 36:1119–1136, January 2007.
- [36] A. Jameson. Solution of the Euler equations for two dimensional transonic flow by a multigrid method. *Applied Mathematics and Computation*, 13(3-4):327–355, 1983.
- [37] A. Jameson. Multigrid algorithms for compressible flow calculations. In W. Hackbusch and U. Trottenberg, editors, *Multigrid Methods II*, volume 1228 of *Lecture Notes in Mathematics*, pages 166–201. Springer Berlin Heidelberg, 1986.
- [38] A. Jameson. Time-dependent calculations using multigrid with applications to unsteady flows past airfoils and wings. In *10<sup>th</sup> AIAA Computational Fluid Dynamics Conference*, number 1991-1596, Honolulu, HI, June 1991.
- [39] A. Jameson. Analysis and design of numerical schemes for gas dynamics 1 artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence. *International Journal of Computational Fluid Dynamics*, 4:171–218, 1995.
- [40] A. Jameson and T. J. Baker. Solution of the Euler equations for complex configurations. In *AIAA 6<sup>th</sup> Computational Fluid Dynamics Conference*, number 83-1929, pages 293–302, Danvers, MA, July 1983.
- [41] A. Jameson, W. Schmidt, and E. Turkel. Numerical solutions of the Euler equations by finite volume methods with Runge-Kutta time-stepping schemes.

In 14<sup>th</sup> *AIAA Fluid And Plasma Dynamics Conference*, number 81-1259, Palo Alto, CA, June 1981.

- [42] A. A. Johnson and T. E. Tezduyar. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Computer Methods in Applied Mechanics and Engineering*, 119(1-2):73–94, 1994.
- [43] F. Kachra and S. K. Nadarajah. Aeroelastic solutions using the nonlinear frequency-domain method. *AIAA Journal*, 46(9):2202–2210, September 2008.
- [44] R. E. Kielb, K. C. Hall, M. Spiker, and J. P. Thomas. Non-synchronous vibration of turbomachinery airfoils. Technical report, Duke University, Durham, NC, March 2006.
- [45] R. H. Landon. Data set 3, NACA 0012, oscillatory and transient pitching. Technical Report R-702, AGARD, 1985.
- [46] M. Lesoinne and C. Farhat. Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. *Computational Methods in Applied Mechanics and Engineering*, 134:71–90, 1996.
- [47] M. Lesoinne and C. Farhat. High-order subiteration-free staggered algorithm for nonlinear transient aeroelastic problems. *AIAA Journal*, 36(9):1754–1757, 1998.
- [48] Xueqiang Liu, Ning Qin, and Hao Xia. Fast dynamic grid deformation based on Delaunay graph mapping. *Journal of Computational Physics*, 211(2):405–423, 2006.
- [49] R. Lohner and C. Yang. Improved ALE mesh velocities for moving bodies. *Communications in Numerical Methods in Engineering*, 12:599–608, 1996.
- [50] R. C. Maple, P. I. King, P. D. Orkwis, and J. M. Wolff. Adaptive harmonic balance method for nonlinear time-periodic flows. *Journal of Computational Physics*, 193:620–641, 2004.
- [51] M. McMullen and A. Jameson. The computational efficiency of non-linear frequency domain methods. *Journal of Computational Physics*, 212:637–661, 2006.

- [52] M. McMullen, A. Jameson, and J. Alonso. Application of a non-linear frequency domain solver to the Euler and Navier-Stokes equations. In *40<sup>th</sup> AIAA Aerospace Sciences Meeting & Exhibit*, number 2002-0120, Reno, NV, January 2002.
- [53] M. McMullen, A. Jameson, and J. Alonso. Demonstration of non-linear frequency domain methods. *AIAA Journal*, 44(7):1428–1435, 2006.
- [54] M. McMullen, A. Jameson, and J. J. Alonso. Acceleration of convergence to a periodic steady state in turbomachinery flows. In *39<sup>th</sup> AIAA Aerospace Sciences Meeting & Exhibit*, number 2001-0152, Reno, NV, January 2001.
- [55] A. M. Morris, C. B. Allen, and T. C. S. Rendall. Domain-element method for aerodynamic shape optimization applied to a modern transport wing. *AIAA Journal*, 47(7):1647–1659, July 2009.
- [56] A. M. Morris, C. B. Allen, and T. C. S. Rendall. High-fidelity aerodynamic shape optimization of modern transport wing using efficient hierarchical parametrization. *International Journal for Numerical Methods in Fluids*, 63:297–312, 2010.
- [57] A. Mosahebi and S. Nadarajah. An adaptive non-linear frequency domain method for viscous flows. *Computers & Fluids*, 75:140–154, April 2013.
- [58] A. Mosahebi and S. Nadarajah. An implicit and adaptive nonlinear frequency domain approach for periodic viscous flows. *Journal of Computational Physics*, 278:92–116, December 2014.
- [59] A. Mosahebi and S. K. Nadarajah. An implicit adaptive non-linear frequency domain method (*p*NLFD) for viscous periodic steady state flows on deformable grids. In *49<sup>th</sup> AIAA Aerospace Sciences Meeting and Exhibit*, number 2011-775, Orlando, FL, January 2011.
- [60] A. Mosahebi and S. K. Nadarajah. Dynamic mesh deformation for implicit adaptive non-linear frequency domain method. In *7<sup>th</sup> AIAA International Conference on Computational Fluid Dynamics*, number 7-3804, Big Island, HI, July 2012.
- [61] N. L. Mundis and D. J. Mavriplis. Quasi-periodic time spectral method for aeroelastic flutter analysis. In *51<sup>st</sup> AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, number 2013-0638, Grapevine, TX, January 2013.

- [62] N. L. Mundis and D. J. Mavriplis. An efficient GMRES solver for the fully-coupled time-spectral aeroelastic system. In *52<sup>nd</sup> AIAA Aerospace Sciences Meeting*, number 2014-1427, National Harbor, MD, January 2014.
- [63] S. K. Nadarajah. Convergence studies of the time accurate and non-linear frequency domain methods for optimum shape design. *International Journal of Computational Fluid Dynamics*, 21(5-6):189–207, 2007.
- [64] S. K. Nadarajah and A. Jameson. Optimum shape design for unsteady three-dimensional viscous flows using a nonlinear frequency-domain method. *Journal of Aircraft*, 44(5):1513–1327, September-October 2007.
- [65] S. K. Nadarajah, M. McMullen, and A. Jameson. Non-linear frequency domain based optimum shape design for unsteady three-dimensional flows. In *44<sup>th</sup> AIAA Aerospace Sciences Meeting and Exhibit*, number 2006-1052, Reno, NV, January 2006.
- [66] W. Ning and L. He. Computation of unsteady flows around oscillating blades using linear and nonlinear harmonic Euler methods. *Journal of Turbomachinery*, 120(3):508–514, July 1998.
- [67] V. Poirier and S. K. Nadarajah. Efficient RBF mesh deformation within an adjoint-based aerodynamic optimization framework. In *50<sup>th</sup> AIAA Aerospace Sciences Meeting and Exhibit*, number 2012-0059, Nashville, TN, January 2012.
- [68] M. T. Rahmati, L. He, Y. S. Li, R. G. Wells, and S. K. Krishnababu. Nonlinear time and frequency domain methods for multirow aeromechanical analysis. *Journal of Turbomachinery*, 136(4):041010–1–10, September 2013.
- [69] S. S. Rao. *The Finite Element Method in Engineering*. Elsevier, Burlington, MA, 2011.
- [70] D. E. Raveh, Y. Levy, and M. Karpel. Efficient aeroelastic analysis using computational unsteady aerodynamics. *Journal of Aircraft*, 38(3):547–556, May-June 2001.
- [71] T. C. S. Rendall and C. B. Allen. Efficient mesh motion using radial basis functions with data reduction algorithms. *Journal of Computational Physics*, 228(17):6231–6249, 2009.

- [72] T. C. S. Rendall and C. B. Allen. Parallel efficient mesh motion using radial basis functions with application to multi-bladed rotors. *International Journal for Numerical Methods in Engineering*, 81:89–105, 2010.
- [73] F. Sicot, A. Gomar, G. Dufour, and A. Dugeai. Time-domain harmonic balance method for turbomachinery aeroelasticity. *AIAA Journal*, 52(1):62–71, January 2014.
- [74] F. Sicot, T. Guédeney, and G. Dufour. Time-domain harmonic balance method for aerodynamic and aeroelastic simulations of turbomachinery flows. *International Journal of Computational Fluid Dynamics*, 27(2):68–78, January 2013.
- [75] F. Sicot, G. Puigt, and M. Montagnac. Block-Jacobi implicit algorithms for the time spectral method. *AIAA Journal*, 46(12):3080–3089, December 2008.
- [76] K. Stein, T. Tezduyar, and R. Benney. Mesh moving techniques for fluid-structures interaction with large displacements. *Journal of Applied Mechanics*, 70(1):58–63, 2003.
- [77] R. Szilard. *Theories and Applications of Plate Analysis: Classical, Numerical and Engineering Methods*. John Wiley & Sons, Inc., Hoboken, NJ, 2004.
- [78] P.-O. Tardif and S. Nadarajah. Dynamic mesh deformation with radial basis functions for the non-linear frequency domain method. In *53<sup>rd</sup> AIAA Aerospace Sciences Meeting*, Kissimmee, FL, January 2015.
- [79] J. P. Thomas, E. H. Dowell, and K. C. Hall. A harmonic balance approach for modeling three-dimensional nonlinear unsteady aerodynamics and aeroelasticity. In *ASME 2002 International Mechanical Engineering Congress and Exposition*, number IMECE-2002-32532, pages 1323–1334, New Orleans, LA, November 2002. American Society of Mechanical Engineers.
- [80] J. P. Thomas, E. H. Dowell, and K. C. Hall. Modeling viscous transonic limit cycle oscillation behavior using a harmonic balance approach. In *43<sup>rd</sup> AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, number 2002-1414, Denver, CO, April 2002.
- [81] J. P. Thomas, E. H. Dowell, and K. C. Hall. Nonlinear inviscid aerodynamic effects on transonic divergence, flutter and limit-cycle oscillations. *AIAA Journal*, 40(4):638–646, April 2002.

- [82] J. P. Thomas, E. H. Dowell, and K. C. Hall. Three-dimensional transonic aeroelasticity using proper orthogonal decomposition-based reduced-order models. *Journal of Aircraft*, 40(3):544–551, May-June 2003.
- [83] J. P. Thomas, K. C. Hall, and E. H. Dowell. A harmonic balance approach for modeling nonlinear aeroelastic behavior of wings in transonic viscous flow. In *44<sup>th</sup> AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit*, number 2003-1924, Norfolk, VA, April 2003.
- [84] P. D. Thomas and C. K. Lombard. Geometric conservation law and its application to flow computations on moving grids. *AIAA Journal*, 17(10):1030–1037, 1979.
- [85] J. L. Tocher. *Analysis of Plate Bending Using Triangular Elements*. Department of civil engineering, University of California, Berkeley, 1962.
- [86] A. H. Truong, C. A. Oldfield, and D. W. Zingg. Mesh movement for a discrete-adjoint Newton-Krylov algorithm for aerodynamic optimization. *AIAA Journal*, 46:1695–1704, 2008.
- [87] M. Vinokur. An analysis of finite-difference and finite-volume formulations of the conservation laws. *Journal of Computational Physics*, 81(1):1–52, March 1989.
- [88] B. Walther and S. Nadarajah. Optimum shape design for multirow turbomachinery configurations using a discrete adjoint approach and an efficient radial basis function deformation scheme for complex multiblock grids. *Journal of Turbomachinery*, 137(8):081006–1–20, August 2015.
- [89] H. Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics*, 4:389–396, 1995.
- [90] E. C. Yates. AGARD standard aeroelastic configurations for dynamic response. Candidate configuration I.-wing 445.6. Technical Memorandum 100492, NASA, Hampton, VA, 1987.
- [91] E. C. Yates, N. S. Land, and J. T. Foughner. Measured and calculated subsonic and transonic flutter characteristics of a 45° sweptback wing planform in air and in Freon-12 in the Langley transonic dynamics tunnel. Technical Note D-1616, NASA, Hampton, VA, March 1963.

- [92] D. Zeng and C. R. Ethier. A semi-torsional spring analogy model for updating unstructured meshes in 3D moving domains. *Finite Elements in Analysis and Design*, 41(11-12):1118–1139, 2005.
- [93] H. Zhang, M. Reggio, J. Y. Trépanier, and R. Camarero. Discrete form of the GCL for moving meshes and its implementation in CFD schemes. *Computers & Fluids*, 22(1):9–23, 1993.
- [94] R. J. Zwaan. Data set 9, LANN wing, pitching oscillation. Technical Report R-702, Addendum No. 1, AGARD, 1985.