Titre: Title:	Harmonic Balance Vortex Lattice Method with Numerical Continuation for Nonlinear Aeroelastic Analysis
Auteur: Author:	Samuel Ayala
Date:	2025
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Ayala, S. (2025). Harmonic Balance Vortex Lattice Method with Numerical Continuation for Nonlinear Aeroelastic Analysis [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/71958/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/71958/
Directeurs de recherche: Advisors:	Éric Laurendeau, & Roberto Paoli
Programme: Program:	Génie aérospatial

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Harmonic Balance Vortex Lattice Method with Numerical Continuation for Nonlinear Aeroelastic Analysis

SAMUEL AYALA
Département de génie mécanique

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie aérospatial

Decembre 2025

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Harmonic Balance Vortex Lattice Method with Numerical Continuation for Nonlinear Aeroelastic Analysis

présenté par **Samuel AYALA**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

David SAUSSIÉ, président
Eric LAURENDEAU, membre et directeur de recherche
Roberto PAOLI, membre et codirecteur de recherche
Alain BATAILLY, membre

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Prof. Éric Laurendeau, for giving me the opportunity to work on this very interesting topic and for his guidance and availability throughout the duration of this research. I'm grateful for him for giving me the freedom to explore various directions while granting me full autonomy over the work.

I would also like to thank Matthieu Parenteau and Gabriel Saiz from Bombardier for being interested in this project and providing useful insights from an industrial perspective. Since my work was partially based on Matthieu's previous research, I would like to thank him for sharing his knowledge every week and helping me navigate through the complexities of aeroelasticity and frequency domain methods.

Finally, I would like to thank my family and friends for all their support during my studies. It has been much more enjoyable thanks to their encouragements and day-to-day support.

RÉSUMÉ

Les progrès récents de l'industrie aérospatiale vers des structures plus flexibles ainsi qu'avec des exigences de certification en constante évolution, ont introduit d'importants défis aéroélastiques non linéaires que les méthodes d'analyse traditionnelles ne peuvent pas résoudre de manière adaptée. Le flottement et les oscillations périodiques (LCO), qui peuvent émerger des non-linéarités structurelles telles que du jeu mécanique ou encore un amortissement non linéaire, présentent des risques critiques qui doivent être évalués minutieusement sur l'ensemble de l'enveloppe de vol. Bien que les méthodes temporelles sont capables de capturer ces phénomènes, le coût de calcul associé les rendent peu pratiques pour les phases de conception préliminaire et de certification.

Ce mémoire présente le développement et l'implémentation de la méthode *Harmonic Balance Vortex Lattice Method (HBVLM)* accélérée par GPU et intégrée au sein d'un solveur de continuation numérique pour l'analyse aéroélastique non linéaire. L'approche dans le domaine fréquentiel est applicable aux problèmes de dynamique non linéaire avec la même précision que sa variante temporelle: *Unsteady Vortex Lattice Method (UVLM)*, tout en solvant pour la réponse périodique en régime permanent directement, contournant ainsi toute réponse transitoire dans la solution. Des améliorations à la formulation HBVLM originale sont introduites pour améliorer l'efficacité de convergence de la méthode, et une accélération substantielle est obtenue en GPU.

Le solveur de continuation utilise une approche prédicteur-correcteur et intègre un pas de continuation adaptatif avec une paramétrisation en longueur d'arc, pour permettre un suivi robuste des branches de solutions à travers les points de retournement et les bifurcations complexes. Avec des fonctionnalités comme l'analyse de stabilité directement dans le domaine fréquentiel et la détection de bifurcations, la réponse non linéaire peut être caractérisée efficacement. Toutes les méthodes sont rigoureusement vérifiées contre des problèmes académiques établis avant d'être appliquées à une section d'aile à trois degrés de liberté avec plusieurs types de non-linéarités au niveau de la surface de contrôle.

Les résultats démontrent un bon accord avec la théorie analytique de Theodorsen tout en révélant des branches LCO secondaires précédemment non documentées et des structures de bifurcation détaillées. L'analyse a tracé avec succès la réponse non linéaire complète, incluant les cycles limites stables et instables, les sauts d'amplitude, les comportements hystérétiques et les transitions vers le chaos. Ces résultats démontrent l'applicabilité de ces méthodes pour comprendre les phénomènes de flottement sous-critique qui émergent en dessous de la vitesse

de flottement linéaire.

Ce travail constitue une avancée significative dans le développement d'outils de fidélité moyenne efficaces pour l'analyse aéroélastique non linéaire. Bien que la solution actuelle soit limitée aux conditions incompressibles et aux modèles structurels simplifiés, elle sert de base pour des avancées futures où des extensions peuvent être rajoutées pour augmenter la fidélité du modèle telles que la méthode *Non-Linear Vortex Lattice Method (NLVLM)*.

ABSTRACT

The aerospace industry's progress towards more flexible aircraft structures along with ever-changing certification demands have introduced significant nonlinear aeroelastic challenges that traditional analysis methods struggle to address. Flutter and limit cycle oscillations (LCOs), which can emerge from structural nonlinearities such as control surface freeplay and nonlinear damping, pose critical risks that must be thoroughly evaluated across the entire flight envelope. While time-domain methods can capture these phenomena, their computational demands makes them impractical for early design phases and certification.

This Masters thesis presents the development and implementation of a GPU-accelerated Harmonic Balance Vortex Lattice Method (HBVLM) integrated within a numerical continuation framework for nonlinear aeroelastic analysis. The frequency-domain approach is applicable to nonlinear dynamics with the same accuracy as time-domain methods such as the Unsteady Vortex Lattice Method (UVLM), all while computing the steady-state periodic response directly, effectively bypassing any transients in the solution. Improvements to the original HBVLM formulation are introduced to significantly improve the convergence efficiency of the method, along with substantial speedups through GPU acceleration.

The numerical continuation framework uses a predictor-corrector architecture and integrates adaptive step-size control with arc-length parameterization, enabling robust tracking of solution branches through turning points and complex bifurcations. Additional capabilities such as frequency domain stability analysis and bifurcation detection are incorporated for characterization of the nonlinear responses. All methods were thoroughly verified against established academic problems before being applied to a three-degree-of-freedom wing section exhibiting multiple kinds of nonlinearities at the control surface hinge.

Results demonstrate good agreement with analytical Theodorsen theory while revealing previously undocumented secondary LCO branches and detailed bifurcation structures. The analysis successfully mapped the complete nonlinear response, including stable and unstable limit cycles, amplitude jumps, hysteretic behaviors, and transitions to chaos. These findings show the applicability of these methods for understanding subcritical flutter phenomena that emerge below the linear flutter boundary.

This work is a significant step forward in providing efficient medium-fidelity tools for nonlinear aeroelastic analysis. While the current solution is limited to incompressible flow and simplified structural models, it serves as a useful foundation for future enhancements where higher-fidelity components can be incorporated such as the Non-Linear Vortex Lattice Method.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ACRONYMS

BEM	Boundary Element Method
VLM	Vortex Lattice Method
UVLM	Unsteady Vortex Lattice Method
HB-VLM	Harmonic Balance Vortex Lattice Method
NL-VLM	Nonlinear Vortex Lattice Method
DLM	Doublet Lattice Method
HB	Harmonic Balance
NS	Neimark-Sacker
BP	Branch Point
LP	Limit Point
PD	Period Doubling
FFT	Fast Fourier Transform
DFT	Discrete Fourier Transform
AFT	Alternating Frequency Time
RANS	Reynolds-Averaged Navier Stokes
LES	Large Eddy Simulation
DNS	Direct Numerical Simulation
LCO	Limit Cycle Oscillation
CFD	Computational Fluid Dynamics
CSD	Computational Structural Dynamics
FSI	Fluid-Structure Interaction
DOF	Degree Of Freedom
TE	Trailing Edge
LE	Leading Edge
LHS	Left Hand Side
RHS	Right Hand Side
CPU	Central Processing Unit
GPU	Graphics Processing Unit
MPI	Message Passing Interface

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

The aerospace industry stands at a critical juncture where environmental pressures, economic constraints, and technological advancement converge to demand changes in aircraft design philosophy. This has pushed a fundamental shift in how engineers approach aircraft design, pushing the boundaries of structural efficiency through the adoption of lighter materials, higher aspect ratio wings, and increasingly flexible structures. These design choices, while promising significant improvements in fuel efficiency and operational performance, introduce complex aeroelastic challenges that traditional analysis methods struggle to address adequately.

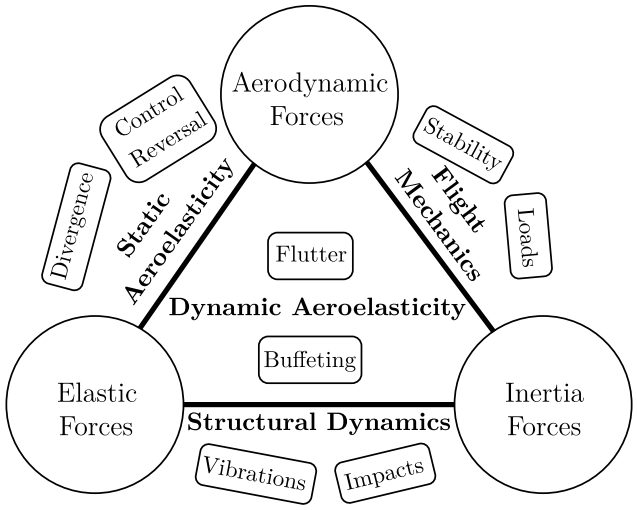


Figure 1.1 Aeroelasticity Collar's triangle of forces [2]

Flutter, perhaps the most critical of these aeroelastic phenomena, represents a self-excited oscillation arising from the coupling between aerodynamic forces and structural dynamics [7]. When an aircraft structure vibrates, it modifies the surrounding airflow, which in turn generates aerodynamic forces that can either damp or amplify the structural motion. Beyond a critical flight speed, this interaction can lead to oscillatory growth, potentially resulting in catastrophic structural failure within seconds. The complexity of flutter prediction increases dramatically when considering that modern aircraft must operate safely across an extensive flight envelope, encompassing variations in altitude, Mach number, fuel loading, and countless other parameters that influence both the structural and aerodynamic characteristics of the vehicle.

1.1 Motivation

The traditional approach to flutter analysis in the aerospace industry has relied heavily on linear methods, which assume small perturbations around an equilibrium state. These methods, while computationally efficient and well-established in certification procedures, fail to capture the rich nonlinear dynamics that emerge in modern aircraft structures. The reality is that structural nonlinearities are ubiquitous in aircraft systems, arising from various sources including large deflections of flexible wings [8], freeplay in control surface hinges [9], nonlinear stiffness characteristics of composite materials [10], and the inherently nonlinear behavior of hydraulic actuators and control systems. These nonlinearities can fundamentally alter the flutter characteristics of an aircraft, potentially leading to limit cycle oscillations (LCOs) that persist at flight speeds below the linear flutter boundary, a phenomenon better known as subcritical flutter [11].

The identification and characterization of subcritical flutter represents one of the most challenging aspects of modern aeroelastic analysis. Unlike classical flutter, which exhibits a clear stability boundary, subcritical flutter can emerge suddenly when the aircraft encounters a sufficiently large disturbance, such as atmospheric turbulence or aggressive control inputs. Once initiated, these oscillations can persist even when the flight conditions return to their nominal values, creating a hysteretic behavior that significantly complicates the aircraft certification process. The amplitude and frequency of these limit cycle oscillations depend sensitively on the specific nonlinear characteristics of the system, necessitating analysis methods capable of capturing both the global dynamics and local stability properties of the nonlinear response [12].

The evolving landscape of aviation certification requirements further compounds these challenges. Regulatory authorities increasingly demand comprehensive demonstration of flutter-free operation not only under nominal conditions but across a wide spectrum of potential failure modes and off-nominal scenarios. This includes consideration of degraded actuator performance, control surface freeplay development over the aircraft's service life, and the effects of structural modifications or repairs. Each of these scenarios potentially introduces or modifies nonlinear behaviors that must be thoroughly understood and documented. The traditional approach of relying on extensive flight testing to clear the flutter envelope becomes prohibitively expensive and potentially dangerous when dealing with nonlinear phenomena that may exhibit sensitive dependence on initial conditions or parameter variations.

Within this context, the role of control surface actuators emerges as particularly critical. Modern fly-by-wire aircraft rely on sophisticated actuator systems that must provide precise

control authority while minimizing weight and power consumption. These actuators invariably exhibit nonlinear damping characteristics that vary with oscillation amplitude and frequency. Understanding how these nonlinear damping properties influence the onset and characteristics of limit cycle oscillations is essential for both design optimization and certification compliance. The parametric sensitivity of LCO characteristics to actuator properties necessitates analysis methods capable of efficiently exploring vast parameter spaces while maintaining sufficient fidelity to capture the essential physics.

The high computational cost associated with nonlinear aeroelastic analysis suggests that leveraging accelerated computing platforms, such as Graphics Processing Units (GPUs), could provide significant performance improvements. GPUs, with their massively parallel architecture, are particularly well-suited for both fluid and structural methods. However, effectively harnessing this computational power requires careful algorithm design and implementation strategies that take into account the underlying hardware architecture.

As a result, this work is motivated by the need to extend current medium fidelity methods to nonlinear aeroelastic analysis and tackles the following problem statement:

> *Modern aircraft design demands robust and efficient methods for analyzing complex nonlinear aeroelastic phenomena with sufficient fidelity to ensure accurate enough predictions across the flight envelope. Key challenges in certification programs include full characterization of limit cycle oscillations under various forms of nonlinearities.*

1.2 Literature Review

Nonlinear aeroelasticity is traditionally studied using time-domain methods that involve the coupling of a Computational Structural Dynamics (CSD) model with a Computational Fluid Dynamics (CFD) method.

The computational demands of nonlinear aeroelastic analysis using time-domain methods quickly become intractable as the integration is performed over very long time periods to ensure steady-state limit cycle behavior, cleared of any transients. As time domain steady state responses depend on initial conditions, the identification of the full set of possible stable solutions requires multiple types of initial conditions to be tested, further increasing the computational cost.

This led to the neglect of nonlinear analysis in early design phases where multiple configurations must be evaluated rapidly [13].

Frequency domain methods offer a compelling alternative as the steady periodic response is

solved directly, along with being able to solve for unstable solution branches using continuation techniques.

The Harmonic Balance Method (HBM) has seen widespread adoption in the turbomachinery community, where it is used for forced response analysis with complex wall friction models [3]. However, its adoption in the aerospace industry remains limited. The industry standard aeroelastic practice is based on linearized approaches such as the Rational Function Approximation (RFA) of compressible Doublet Lattice Methods [14] to handle nonlinearities expressed in the time domain.

Multiple approaches exist for the treatment of nonlinear forces. The simplest continuous nonlinearities can be performed using analytical polynomial development [15] of the Taylor series of the nonlinearity. Piecewise polynomial approximations are an alternative that can handle discontinuous nonlinearities like freeplay and friction [16]. The Alternating Frequency-Time (AFT) method however, is a general approach that is capable of dealing with any kind of nonlinearity expressed in time domain, as such it has become the method of choice in most practical HBM implementations [17]. Shooting methods are another approach to computing periodic solutions by solving a boundary value problem in time domain [18]. However, they tend to be less computationally efficient for large scale systems which limits their applicability for industrial problems [5].

The current landscape of harmonic balance implementations is diverse with numerous variations and formulations existing across different research groups. Many of these variations are mathematically equivalent and simply differ in their numerical implementation. A comprehensive overview of these variants and their relationships has been recently presented by Yan et al. [19], providing useful guidance for researchers entering this field.

The application of harmonic balance methods to CFD methods represents an active area of research. While significant progress has been made in developing time-spectral formulations for Euler and Reynolds-Averaged Navier-Stokes (RANS) methods [20, 21], the integration of these methods within a complete aeroelastic framework remains difficult. Embedding both into a numerical continuation framework remains in early developmental stages [22] due to the sheer computational cost associated to using high fidelity CFD methods.

Even in time-domain simulations, the substantial computational cost associated with volumetric CFD methods is compounded by additional complexities related to mesh deformation and force transfers. While alternatives such as the immersed boundary method are being actively investigated to circumvent these issues [23], these approaches introduce their own challenges and have not yet reached maturity.

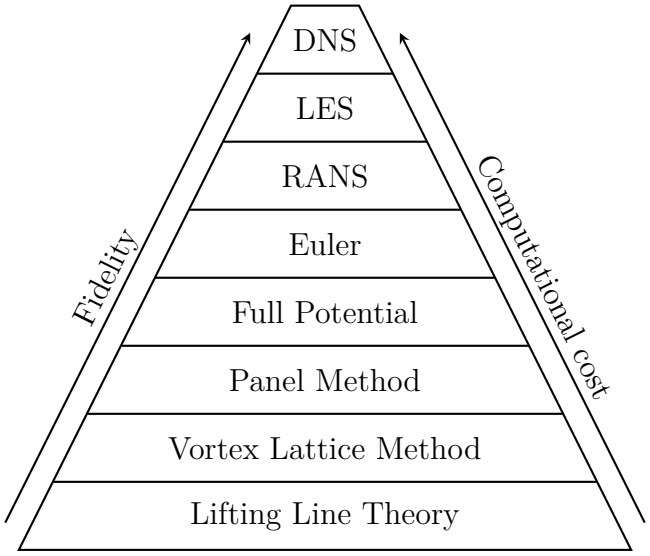


Figure 1.2 Hierarchy of CFD models

This represents a critical gap that must be addressed in order to provide valuable tools for early design phases.

Figure 1.2 illustrates the hierarchy of CFD methods ranging from basic lifting line theory to ultra high-fidelity Direct Numerical Simulation (DNS). Each step up in fidelity brings a substantial increase in computational cost, often approaching an order of magnitude jump.

In this context, boundary element CFD methods present an attractive compromise between computational efficiency and physical fidelity. Methods such as the Vortex Lattice Method (VLM) and its unsteady variant, the Unsteady Vortex Lattice Method (UVLM) [24], offer significant advantages for fluid-structure interaction (FSI) problems. The absence of a volumetric mesh eliminates the need for remeshing entirely, while their computational efficiency makes them well suited for parametric studies. Recent developments have showed that by incorporating sectional corrections derived from RANS computations, the NL-VLM can provide reasonable predictions in transonic and compressible flow regimes [25]. This significantly broadens the applicability of these methods while maintaining their computational efficiency.

Recent efforts have demonstrated significant speedups through GPU implementations of time domain vortex panel methods [26], suggesting that similar performance gains could be achieved for their frequency domain counterparts.

1.3 Research objectives

Following the problem statement and the literature review, the goal of this work is defined as:

> *Develop an aeroelastic framework based on the High Order Harmonic Balance Vortex Lattice Method (HBVLM) for the analysis of nonlinear aeroelastic phenomena of a wing section including stability analysis and bifurcation identification, all while evaluating the performance gains offered by GPU acceleration.*

The following objectives are defined to achieve this goal:

1. **Frequency-Domain Framework**: Implement the HBVLM and embed it within a numerical continuation framework for tracking LCOs and their stability.
2. **Nonlinear Analysis Capability**: Apply the integrated framework to analyze a wing section subject to various structural nonlinearities.
3. **Computational Efficiency**: Develop GPU-accelerated implementations of VLM, UVLM and HBVLM methods.

and the following methodology is proposed:

1. Reimplement the existing VLM and UVLM methods in Prof. Laurendeau's lab with a focus on performance and GPU acceleration.
2. Apply the UVLM to time domain aeroelastic problems and verify the implementation against Theodorsen theory.
3. Implement the iterative HBVLM method and verify against UVLM time domain results.
4. Develop a continuation framework and verify for non autonomous and autonomous nonlinear systems.
5. Integrate the HBVLM method within the continuation framework and compare against Theodorsen theory.

1.4 Thesis Outline

The remainder of this thesis is organized as follows:

Chapter 2 presents the details of the solution developed in this work. Theory details are given for the aerodynamic method used, the aeroelastic coupling, the harmonic balance formulation and the continuation framework.

Chapter 3 presents the results obtained using the developed framework. Every segregated module is thoroughly verified against analytical theory and validated against experimental data when available. Finally, the modules are integrated into a complete framework and results are presented for complete academic aeroelastic test cases.

The last chapter, Chapter 4, summarizes the contributions of this work, addresses its limitations and proposes avenues for future research.

1.5 Softwares

With the exception of linear algebra routines and the MINPACK [27] library, all code developed during this thesis has been written from scratch. The VLM, UVLM and HBVLM implementations have been written in C++17 with performance as a primary concern. The code has two backends: CPU and GPU. CPU kernels have been written in ISPC [28] to leverage the vector units of modern CPU ISAs and parallelized using Taskflow [29] for data and task parallelism on shared memory. The GPU backend was written in CUDA [30] and leverages NVIDIA's cuBLAS and cuSOLVER libraries for best performance. The HBVLM library also features a Python interface through pybind11 [31]. Meanwhile, the continuation framework, harmonic balance residuals and Theodorsen equations have been implemented in Python.

CHAPTER 2 DETAILS OF THE SOLUTION

This chapter presents the theoretical foundation and numerical methods that underpin the aeroelastic analyses conducted in this thesis.

Section 2.1 establishes the aerodynamic foundation through the Vortex Lattice Method and its unsteady extension. These potential flow-based methods provide the aerodynamic forces and moments necessary for fluid-structure interaction while being computationally efficient.

Section 2.2 presents the aeroelastic coupling, exposing how the interface between aerodynamic and structural domains is formulated. The section addresses how the aeroelastic equations are assembled and solved in the time domain using different time integration schemes.

Section 2.3 introduces the Harmonic Balance method for efficient computation of periodic solutions in dynamical systems. This frequency-domain approach transforms the time-dependent problem into an algebraic system, which directly computes limit cycles without expensive time integration.

Section 2.4 concludes with the description of numerical continuation methods that are capable of tracking solution branches through parameter space. Stability analysis and bifurcation detection are presented for characterization of the solution branches and identification of state transitions.

2.1 Aerodynamics

2.1.1 Vortex Lattice Method

Description

The aerodynamic model employed in this thesis is the Vortex Lattice Method (VLM). For preliminary design studies, it provides reasonable accuracy while being computationally inexpensive.

Under the assumptions of an inviscid, irrotational, and incompressible flow, the velocity field can be expressed as the gradient of a scalar potential ϕ, which satisfies Laplace's equation:

$$\nabla^2 \phi = 0 \tag{2.1}$$

Through Green's theorem, the solution can be solved by the boundary element method, thus avoiding discretization of the flow field. The linearity of the equation enables the construction of a solution through superposition of elementary solutions. The numerical implementation discretizes each lifting surface into quadrilateral panels arranged in a structured grid [24]. Following the fundamental principle established by Pistolesi, vortex filaments are positioned along the quarter-chord line of each panel, while the flow tangency boundary condition is enforced at the three-quarter chord position. This spatial arrangement, known as the quarter-three-quarter rule, ensures consistency with classical thin airfoil theory [32].
According to Bio-Savart law, the induced velocity of an infinitesimal vortex segment $d\boldsymbol{l}$ with a circulation Γ is:

$$d\boldsymbol{v} = \frac{\Gamma}{4\pi} \frac{d\boldsymbol{l} \times \boldsymbol{r}}{r^3} \tag{2.2}$$

After integration over a finite vortex segment, the induced velocity at an arbitrary field point $\boldsymbol{c}$ is given by:

$$\boldsymbol{q}(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{c}) = \frac{\Gamma}{4\pi} \frac{\boldsymbol{r}_1 \times \boldsymbol{r}_2}{|\boldsymbol{r}_1 \times \boldsymbol{r}_2|^2} \boldsymbol{r}_0 \cdot \left(\frac{\boldsymbol{r}_1}{|\boldsymbol{r}_1|} - \frac{\boldsymbol{r}_2}{|\boldsymbol{r}_2|} \right) \tag{2.3}$$

where $\boldsymbol{r}_0 = \boldsymbol{x}_2 - \boldsymbol{x}_1$ represents the vortex segment vector, $\boldsymbol{r}_1 = \boldsymbol{x}_1 - \boldsymbol{c}$ and $\boldsymbol{r}_2 = \boldsymbol{x}_2 - \boldsymbol{c}$ denote the position vectors from the evaluation point to the segment endpoints, see Figure 2.1. Therefore, the influence of a panel at an arbitrary point is simply the sum of the four vortex segment's induced velocities.

The Kutta condition is enforced at the trailing edge, requiring that the circulation of the trailing vortex rings matches the circulation of the bound vortices at the trailing edge panels. This ensures smooth flow departure from the trailing edge and extends the vortex system into the wake with constant circulation strength, consistent with Helmholtz's theorem [33].

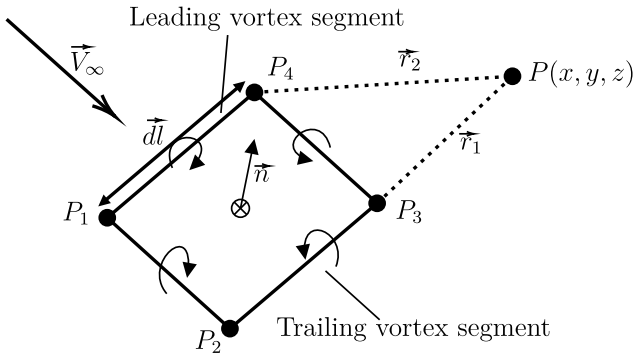


Figure 2.1 Topology of a vortex ring panel [2]

The no-penetration boundary condition is applied at the three-quarter chord location of each panel, also called the collocation point [24]:

$$(\Delta\phi_i + \boldsymbol{V}_i) \cdot \boldsymbol{n}_i = 0 \tag{2.4}$$

where $\boldsymbol{V}_i$ is the local velocity of panel i in global coordinates. The application of the boundary condition to the whole lifting body leads to the assembly of the VLM linear system:

$$\boldsymbol{A}\boldsymbol{\Gamma} = \boldsymbol{b} \quad \text{with} \quad \boldsymbol{A} \in \mathbb{R}^{n\times n} \tag{2.5}$$

Where n is the number of panels. The influence matrix $\boldsymbol{A}$ is defined as:

$$\boldsymbol{A}_{ij} = \boldsymbol{q}(\boldsymbol{x}_{j1}, \boldsymbol{x}_{j2}, \boldsymbol{c}_i) + \boldsymbol{q}(\boldsymbol{x}_{j2}, \boldsymbol{x}_{j3}, \boldsymbol{c}_i) + \boldsymbol{q}(\boldsymbol{x}_{j3}, \boldsymbol{x}_{j4}, \boldsymbol{c}_i) + \boldsymbol{q}(\boldsymbol{x}_{j4}, \boldsymbol{x}_{j1}, \boldsymbol{c}_i) \tag{2.6}$$

where $\boldsymbol{x}_{jk}$ are the four vertices of panel j with linearized indices. The exact index formulas are left out as an implementation detail since they depend on the chosen memory layout. The right hand side is defined as:

$$\boldsymbol{b}_i = -\boldsymbol{V}_i \cdot \boldsymbol{n}_i \tag{2.7}$$

And Γ are the panels circulation to solve for. Through testing, it was determined that the influence matrix should be stored in column-major order and panel data should be in span-wise order as it improves cache efficiency when assembling the matrix. The dense nature of the influence matrix makes the VLM consume $\mathcal{O}(n^2)$ memory. Dense matrices are best solved using direct factoring methods such as the partial pivoting LU decomposition. Modern high-performance implementations employ blocked LU algorithms that leverage highly optimized BLAS Level 3 operations, particularly matrix-matrix multiplication (GEMM) routines. These blocked algorithms exhibit excellent data locality and are particularly well-suited for

GPU acceleration, where the high arithmetic intensity of GEMM operations can fully utilize the computational throughput of modern accelerators.

Postprocessing

Once the circulation distribution is determined, aerodynamic lift is computed using the Kutta-Joukowski theorem. For steady flow, the force per panel is given by:

$$\Delta \boldsymbol{F}_{steady} = \rho \bar{\Gamma} (\boldsymbol{V} \times \boldsymbol{l}) \tag{2.8}$$

where $\bar{\Gamma}_{i,j} = \Gamma_{i,j} - \Gamma_{i-1,j}$ is the circulation difference between chordwise consecutive panels, $\boldsymbol{V}$ is the local panel velocity and $\boldsymbol{l}$ is the bound vortex segment.

The total lift coefficient is obtained by projecting the panel forces onto the unit length lift axis $\boldsymbol{e}_L$ and normalizing by the dynamic pressure and reference area:

$$C_L = \frac{1}{\frac{1}{2}\rho V_\infty^2 S} \sum_i \sum_j \Delta \boldsymbol{F}_{ij} \cdot \boldsymbol{e}_L \tag{2.9}$$

Similarly, the moment coefficient about a reference point is computed as:

$$C_M = \frac{1}{\frac{1}{2}\rho V_\infty^2 S c_{ref}} \sum_i \sum_j (\boldsymbol{r}_{ij} - \boldsymbol{r}_{ref}) \times \Delta \boldsymbol{F}_{ij} \tag{2.10}$$

where $\boldsymbol{r}_{ij}$ denotes the force application point (typically the center of the leading vortex segment) and c_{ref} is the reference length that is usually the mean aerodynamic chord (MAC).

$$c_{ref} = \frac{2}{S} \int_0^{b/2} c(y)^2 dy \tag{2.11}$$

where the integration is performed using trapezoidal rule.
While induced drag can be computed from the VLM solution, the assumptions inherent in the method limit its accuracy for drag prediction. When drag estimation is required, the Trefftz plane analysis provides a more consistent approach by evaluating the trailing vorticity in the far wake. However, for the aeroelastic applications considered in this work, drag forces are of secondary importance compared to lift and moment predictions.

Nonlinear Extension

To extend the VLM applicability beyond its strict theoretical limitations, nonlinear section corrections can be incorporated to account for compressibility and viscous effects. A simple

approach involves the modified Van-Dam α-coupling method [34], where the effective angle of attack at each section is iteratively adjusted based on two-dimensional airfoil data that includes compressibility and thickness effects. The method is very accurate on constant-

Algorithm 1 Van Dam modified α method

1: VLM: Cl_{inviscid}
2: **for** Every Span-Wise Section i **do**
3: $\quad \alpha_e(i) = \dfrac{Cl_{\text{inviscid}}(i)}{2\pi} - \alpha_{2D}(i) + \alpha_{3D}$
4: $\quad \alpha_e(i) \Rightarrow Cl_{\text{visc}}(\alpha_e(i))$
5: $\quad \alpha_{2D}(i) = \alpha_{2D}(i) + \dfrac{Cl_{\text{visc}}(\alpha_e(i)) - Cl_{\text{inviscid}}(i)}{2\pi}$
6: **end for**
7: Repeat Steps until $|Cl_{\text{visc}} - Cl_{\text{inviscid}}| < \epsilon$

chord unswept or swept wings, free from wing-body or wall interference. For more complex geometries it provides sufficient fidelity for preliminary design and steady-state aeroelastic analysis, where the primary concern is accurate prediction of lift distribution and pitching moments.

2.1.2 Unsteady Vortex Lattice Method

Description

The Unsteady Vortex Lattice Method (UVLM) extends the steady-state formulation to account for time-dependent effects through the inclusion of a freely convecting wake. Unlike the steady VLM where the wake extends to infinity with constant circulation, the UVLM is capable of modelling the temporal evolution of shed vorticity, capturing unsteady aerodynamic phenomena essential for dynamic aeroelastic analysis.

In accordance with Kelvin's theorem, the total circulation around a closed material contour must remain constant in inviscid flow [24]. At each time step, new wake panels are shed from the trailing edge with the same circulation strength as the trailing edge panels, see Figure 2.2:

$$\Gamma^{w}_{n+1,j} = \Gamma_{-1,j} \tag{2.12}$$

where $\Gamma_{-1,j}$ denotes the trailing edge panel of spanwise section j.

The wake can be treated either as a *static boundary* where wake vertices move through space according to the motion kinematics, while the wing remains fixed, or as a *dynamic boundary* where the wing moves relative to a stationary reference frame. The dynamic boundary approach is adopted in this work as it maintains constant computational complexity for the

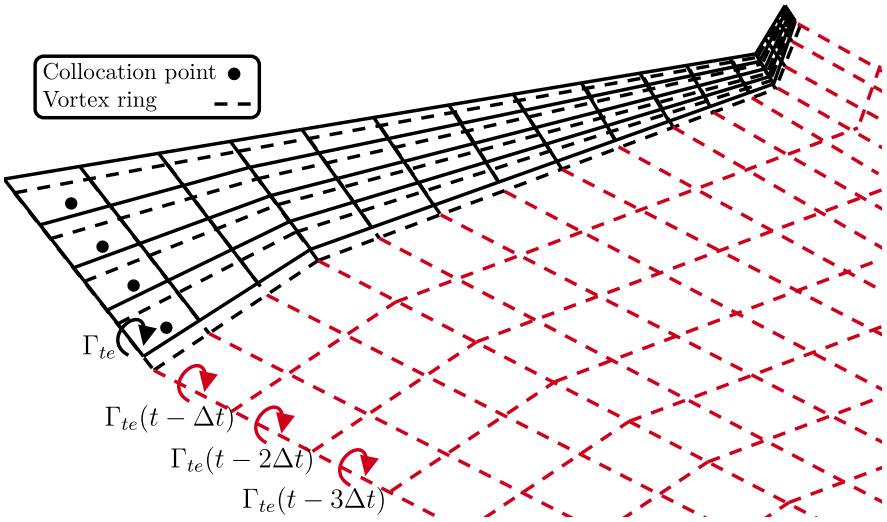


Figure 2.2 Illustration of the wake shedding procedure in the UVLM [2]

forward dynamics since the every growing wake is kept frozen in space without rollup. The UVLM equation is assembled similarly to the steady VLM, but now includes the influence of the wake panels:

$$\boldsymbol{A\Gamma} + \boldsymbol{B\Gamma}^w + (\boldsymbol{V} \cdot \boldsymbol{n}) = \boldsymbol{0} \quad \text{with} \quad \boldsymbol{B} \in \mathbb{R}^{n \times n_W} \tag{2.13}$$

with $\boldsymbol{B}$ the rectangle matrix containing the influence coefficients of the wake on the wing, n_W is the number of wake panels, which increases by the number of shed trailing edge panels at every time step. For better performance, the matrix-vector product should be performed in a matrix free way, that is without actually creating the $\boldsymbol{B}$ matrix as otherwise it will become extremely costly both in terms of compute and memory usage. In our cases, since the wing is rigid, the influence on itself is constant, so the matrix $\boldsymbol{A}$ only has to be assembled and factored once, effectively reducing the compute cost of a UVLM timestep from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$ as only a forward and backward triangular solve is needed.

Wake rollup effects can be incorporated by computing the induced velocity at each wake vertex from all wing and wake vortex rings:

$$\boldsymbol{v}_{wake} = \sum_{bodies} \left(\sum_{wing} \boldsymbol{q}_{induced} + \sum_{wakes} \boldsymbol{q}_{induced} \right) \tag{2.14}$$

The wake vertex positions are then updated using an explicit Euler scheme

$$\boldsymbol{x}_{wake}^{n+1} = \boldsymbol{x}_{wake}^{n} + \Delta t \cdot \boldsymbol{v}_{wake}^{n} \tag{2.15}$$

While wake rollup is implemented in the current framework, it was not employed in the present analyses as the additional computational cost did not justify the marginal accuracy improvements for the aeroelastic problems considered.
For true circulation conservation, the size of the vortex ring must stay constant. Consequently the time step Δt should be defined as

$$\Delta t = \frac{c_{te}}{V_\infty} \tag{2.16}$$

where c_{te} is the chord length of the trailing edge panel. It can be observed that this is an approximation since the velocity induced from the actual motion is ignored. This assumption is generally valid as the freestream velocity is usually dominant compared to the motion-induced velocities.

Kinematic Framework

The computation of boundary conditions requires accurate evaluation of local panel velocities arising from arbitrary body motion. To maintain generality and computational efficiency, the rigid body kinematics are represented using a graph-based kinematic chain formulation employing screw theory [35].

The motion of each body is decomposed into a series of joint transformations represented by homogeneous transformation matrices. For a kinematic chain with n joints, the position of a point on body i in the global frame is:

$$\boldsymbol{x}_g = \boldsymbol{T}_0^1(t)\boldsymbol{T}_1^2(t)\cdots\boldsymbol{T}_{i-1}^i(t)\boldsymbol{x}_b \tag{2.17}$$

where $\boldsymbol{T}_j^{j+1}(t)$ represents the time-dependent transformation from joint j to joint $j+1$, and $\boldsymbol{x}_b$ denotes the point coordinates in the body-fixed frame.

Each transformation matrix combines a rotation and translation component:

$$\boldsymbol{T}_j^{j+1}(t) = \begin{bmatrix} \boldsymbol{R}_j^{j+1}(t) & \boldsymbol{t}_j^{j+1}(t) \\ \boldsymbol{0}^T & 1 \end{bmatrix} \tag{2.18}$$

where $\boldsymbol{R}_j^{j+1}(t) \in SO(3)$ is the rotation matrix and $\boldsymbol{t}_j^{j+1}(t) \in \mathbb{R}^3$ is the translation vector.

The rotation matrix is constructed using the Rodrigues formula [36], which provides an efficient representation for finite rotations about an arbitrary axis. For a rotation by angle

$\theta(t)$ about a unit axis $\boldsymbol{k}$, the rotation matrix is:

$$\boldsymbol{R}(\theta, \boldsymbol{k}) = \boldsymbol{I} + \sin(\theta)\boldsymbol{K} + (1 - \cos(\theta))\boldsymbol{K}^2 \tag{2.19}$$

where $\boldsymbol{I}$ is the identity matrix and $\boldsymbol{K}$ is the skew-symmetric matrix associated with the axis vector:

$$\boldsymbol{K} = \begin{bmatrix} 0 & -k_z & k_y \\ k_z & 0 & -k_x \\ -k_y & k_x & 0 \end{bmatrix} \tag{2.20}$$

This formulation allows for compact representation of arbitrary rotational joints, including revolute joints ($\theta(t)$ varies, $\boldsymbol{k}$ fixed) and spherical joints (both $\theta(t)$ and $\boldsymbol{k}(t)$ vary). The time-dependent joint parameters $\theta(t)$ and $\boldsymbol{t}(t)$ are prescribed by the analytical motion trajectory or determined through the coupled aeroelastic equations of motion.

For prescribed motions, the velocity of panel collocation points is obtained through time differentiation of the kinematic chain. To minimize numerical errors inherent in finite difference schemes, particularly when operating in single precision, automatic differentiation is employed [37]. This approach provides machine-precision derivatives while maintaining computational efficiency comparable to hand-coded analytical expressions.

For a single rigid body with m panels, the vertex updates can be efficiently computed using a single matrix-matrix multiplication:

$$\boldsymbol{X}_g = \boldsymbol{T}\boldsymbol{X}_b \tag{2.21}$$

where $\boldsymbol{X}_b \in \mathbb{R}^{4\times 4m}$ contains the homogeneous coordinates of all vertices and $\boldsymbol{T} \in \mathbb{R}^{4\times 4}$ is the cumulative transformation matrix. Thanks to the dynamic boundary approach, the shed wake panels naturally take the form of the prescribed motion and the normals don't need special updating as they can be directly computed from the transformed vertices.

Unsteady Force Computation

The aerodynamic forces in unsteady flow include an additional contribution from the time rate of change of circulation. Following the formulation proposed by Simpson et al. [38], the total force per panel is:

$$\Delta\boldsymbol{F}_{ij} = \Delta\boldsymbol{F}_{steady} + \Delta\boldsymbol{F}_{unsteady} \tag{2.22}$$

where the unsteady contribution arises from the local acceleration of the fluid

$$\Delta \boldsymbol{F}_{unsteady} = \rho \frac{\partial \Gamma_{ij}}{\partial t} \Delta A_{ij} \boldsymbol{n}_{ij} \tag{2.23}$$

Here, ΔA_{ij} represents the panel area and $\boldsymbol{n}_{ij}$ is the panel normal vector. The time derivative of circulation is approximated using a backward difference scheme:

$$\frac{\partial \Gamma_{ij}}{\partial t} \approx \frac{\Gamma_{ij}^{n} - \Gamma_{ij}^{n-1}}{\Delta t} \tag{2.24}$$

This unsteady force contribution captures the added mass effects and other time-dependent phenomena like multi-body interations through motion.

2.2 Aeroelasticity

2.2.1 Governing Equations

Aeroelasticity concerns the interaction between aerodynamic forces and structural deformations, where the mutual coupling can lead to phenomena such as flutter, divergence, and limit cycle oscillations. The dynamic behavior of an aeroelastic system is governed by the following autonomous equation of motion:

$$\boldsymbol{M}\ddot{\boldsymbol{x}} + \boldsymbol{C}\dot{\boldsymbol{x}} + \boldsymbol{K}\boldsymbol{x} = \boldsymbol{f}(\ddot{\boldsymbol{x}}, \dot{\boldsymbol{x}}, \boldsymbol{x}) \tag{2.25}$$

where $\boldsymbol{M}$, $\boldsymbol{C}$, and $\boldsymbol{K}$ represent the mass, damping, and stiffness matrices respectively, $\boldsymbol{x}$ denotes the generalized displacement vector, and $\boldsymbol{f}$ represents the nonlinear aerodynamic forces that depend on the structural accelerations, velocities, and displacements.

The inherent nonlinearity of the aerodynamic forces, particularly when using methods such as UVLM, prevents direct modal analysis and necessitates time-domain integration. Two primary approaches exist for solving this coupled system: monolithic and partitioned schemes.

2.2.2 Monolithic Approach

The monolithic approach treats the aeroelastic system as a single entity by reformulating equation 2.25 into a first-order system. This requires the aerodynamic model to be expressible as a set of ordinary differential equations. Introducing the state vector $\boldsymbol{y} = [\boldsymbol{x}, \dot{\boldsymbol{x}}]^T$, the

second-order system becomes:

$$\dot{\boldsymbol{y}} = \begin{bmatrix} \dot{\boldsymbol{x}} \\ \ddot{\boldsymbol{x}} \end{bmatrix} = \begin{bmatrix} \dot{\boldsymbol{x}} \\ \boldsymbol{M}^{-1}(\boldsymbol{f} - \boldsymbol{C}\dot{\boldsymbol{x}} - \boldsymbol{K}\boldsymbol{x}) \end{bmatrix} \tag{2.26}$$

This formulation enables the use of standard ODE integration schemes. In my experience, for non-stiff systems, the Tsitouras 5(4) Runge-Kutta pair [39] offers excellent performance compared to other alternatives. When stiffness is present, implicit methods become more efficient, with the ESDIRK4 scheme [40] being the method of choice, having excellent L-stable properties while being computationally efficient.

The monolithic approach achieves optimal computational efficiency by avoiding iterative coupling between solvers. However, it requires tight integration between the structural and aerodynamic models, which is sometimes not possible with the use of commercial structural solvers like NASTRAN [41].

2.2.3 Partitioned Approach

The partitioned approach maintains separate structural and aerodynamic solvers, exchanging information at each time step. This modularity allows for the use of specialized solvers optimized for their respective physics and facilitates the integration of existing codes.
Two coupling strategies are commonly employed: loose (explicit) and tight (implicit) coupling [42]. In loose coupling, information is exchanged once per time step without iteration:

$$\boldsymbol{f}^{n+1} = \mathcal{A}(\boldsymbol{x}^n, \dot{\boldsymbol{x}}^n, \ddot{\boldsymbol{x}}^n) \tag{2.27}$$

$$\boldsymbol{x}^{n+1} = \mathcal{S}(\boldsymbol{f}^{n+1}, \boldsymbol{x}^n, \dot{\boldsymbol{x}}^n) \tag{2.28}$$

where $\mathcal{A}$ and $\mathcal{S}$ represent the aerodynamic and structural solver operators respectively. While computationally efficient, loose coupling can suffer from numerical instability, particularly for systems with strong fluid-structure interaction or when added mass effects are significant.

Tight coupling addresses these stability issues through sub-iterations within each time step until convergence is achieved:

$$\boldsymbol{f}^{n+1,k+1} = \mathcal{A}(\boldsymbol{x}^{n+1,k}, \dot{\boldsymbol{x}}^{n+1,k}, \ddot{\boldsymbol{x}}^{n+1,k}) \tag{2.29}$$

$$\boldsymbol{x}^{n+1,k+1} = \mathcal{S}(\boldsymbol{f}^{n+1,k+1}, \boldsymbol{x}^n, \dot{\boldsymbol{x}}^n) \tag{2.30}$$

where k denotes the sub-iteration index. Convergence is typically assessed using the relative

change in displacements:

$$\epsilon = \frac{\|\boldsymbol{x}^{n+1,k+1} - \boldsymbol{x}^{n+1,k}\|}{\|\boldsymbol{x}^{n+1,k+1}\|} < \text{tol} \tag{2.31}$$

Leveraging convergence acceleration techniques such as Anderson's Acceleration [43] is highly recommended in order to significantly reduce the number of required sub-iterations.

Since the partitioned approach doesn't change the form of the equations of motion, the Newmark-β [44] family of integrators is widely employed due to their favorable unconditional stability properties and second-order accuracy. For complex cases with high frequency modes, the HHT-α method [45] is preferred, as it introduces numerical damping to filter spurious modes while maintaining second-order accuracy. The implementation of these methods in this work closely follows the course notes by Gavin [46].

2.3 Harmonic Balance Method

The harmonic balance method (including its higher-order variants) is a straightforward approach for analyzing free or forced vibration problems in nonlinear systems exhibiting limit-cycle oscillations [47]. The core idea involves directly determining the coefficients of a truncated Fourier series that approximates the periodic solution.

$$\boldsymbol{x}(t) \simeq \frac{1}{2}\boldsymbol{a}_0 + \sum_{j=1}^{n_H} \left[\boldsymbol{a}_j \cos(j\omega t) + \boldsymbol{b}_j \sin(j\omega t)\right] \quad \boldsymbol{x} \in \mathbb{R}^{n_D} \tag{2.32}$$

with n_D the number of degrees of freedom/unknowns. The frequency domain coefficients are be grouped in the following vectorial and matrix form:

$$\tilde{\boldsymbol{x}} = \left[\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{b}_1, \ldots, \boldsymbol{a}_{n_H}, \boldsymbol{b}_{n_H}\right]^\top \quad \tilde{\boldsymbol{x}} \in \mathbb{R}^{n_D n_C}$$

$$\tilde{\boldsymbol{X}} = \left[\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{b}_1, \ldots, \boldsymbol{a}_{n_H}, \boldsymbol{b}_{n_H}\right] \quad \tilde{\boldsymbol{x}} \in \mathbb{R}^{n_D \times n_C}$$

with n_H the number of harmonics and $n_C = 2n_H + 1$ the number of fourier coefficients per dof. We denote $\mathbb{T}$ the Fourier basis vector:

$$\mathbb{T} = \left[\frac{1}{2}, \cos(\omega t), \sin(\omega t), \ldots, \cos(n_H \omega t), \sin(n_H \omega t)\right]^T \tag{2.33}$$

The solution and its time derivatives can now be expressed in a compact notation:

$$\begin{aligned} \boldsymbol{x} &= (\mathbb{T} \otimes \boldsymbol{I}_{n_C})\, \tilde{\boldsymbol{x}} \\ \dot{\boldsymbol{x}} &= \omega\, ((\mathbb{T}\boldsymbol{\nabla}) \otimes \boldsymbol{I}_{n_C})\, \tilde{\boldsymbol{x}} \\ \ddot{\boldsymbol{x}} &= \omega^2 \left(\left(\mathbb{T}\boldsymbol{\nabla}^2 \right) \otimes \boldsymbol{I}_{n_C} \right) \tilde{\boldsymbol{x}} \end{aligned} \tag{2.34}$$

where $\otimes$ denotes the Kronecker product, $\boldsymbol{I}_{n_C}$ is the identity matrix of size n_C and $\boldsymbol{\nabla}$ is the global differential operator of $\boldsymbol{b}$ defined by:

$$\boldsymbol{\nabla} = \text{diagblock}\,(0, \boldsymbol{\nabla}_1, \ldots, \boldsymbol{\nabla}_j, \ldots, \boldsymbol{\nabla}_{n_H}) \quad \text{and} \quad \boldsymbol{\nabla}^2 = \boldsymbol{\nabla}\boldsymbol{\nabla} \tag{2.35}$$

with the elementary first-order derivative matrix $\boldsymbol{\nabla}_j$:

$$\boldsymbol{\nabla}_j = j \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \text{ for } j \in [1..n_H] \tag{2.36}$$

By substituting (2.34) into the time-domain equations of motion (2.25) we obtain:

$$\boldsymbol{M}\omega^2 \left(\left(\mathbb{T}\boldsymbol{\nabla}^2 \right) \otimes \boldsymbol{I}_{n_C} \right) \tilde{\boldsymbol{x}} + \boldsymbol{C}\omega\, ((\mathbb{T}\boldsymbol{\nabla}) \otimes \boldsymbol{I}_{n_C})\, \tilde{\boldsymbol{x}} + \boldsymbol{K}\, (\mathbb{T} \otimes \boldsymbol{I}_{n_C})\, \tilde{\boldsymbol{x}} + \simeq (\mathbb{T}_H \otimes \boldsymbol{I}_n)\, \boldsymbol{f} \tag{2.37}$$

Using the properties of the mixed Kronecker product $(\boldsymbol{A} \otimes \boldsymbol{B})(\boldsymbol{C} \otimes \boldsymbol{D}) = (\boldsymbol{AC}) \otimes (\boldsymbol{BD})$, we rearrange the equation as follows:

$$\boldsymbol{R}(\tilde{\boldsymbol{x}}, t, \omega) = \left[\omega^2 \left(\left(\mathbb{T}\boldsymbol{\nabla}^2 \right) \otimes \boldsymbol{M} \right) + \omega\, ((\mathbb{T}\boldsymbol{\nabla}) \otimes \boldsymbol{C}) + (\mathbb{T} \otimes \boldsymbol{K}) \right] \tilde{\boldsymbol{x}} - (\mathbb{T} \otimes \boldsymbol{I}_{n_C})\, \boldsymbol{f}(\omega, t) \tag{2.38}$$

In order to minimize the error of the residual, we use Galerkin's method which projects the residual onto the vector subspace spanned by the Fourier basis $\mathbb{T}$ [47]. This is equivalent to requiring the orthogonality of the residual with respect to each basis function over one period T. This procedure is also called *Harmonic Balancing* [22].

$$\boldsymbol{R}(\tilde{\boldsymbol{x}}, \omega) = \boldsymbol{Z}(\omega)\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{f}}(\tilde{\boldsymbol{x}}, \omega) \tag{2.39}$$

where $\boldsymbol{Z}$ is the *global harmonic stiffness* block-diagonal matrix

$$\boldsymbol{Z}(\omega) = \omega^2 \boldsymbol{\nabla}^2 \otimes \boldsymbol{M} + \omega \boldsymbol{\nabla} \otimes \boldsymbol{C} + \boldsymbol{I}_{n_C} \otimes \boldsymbol{K} \tag{2.40}$$

partial derivatives, needed for the jacobian, are defined as follows:

$$\begin{aligned}
\frac{\partial \boldsymbol{R}}{\partial \tilde{\boldsymbol{x}}} &= \boldsymbol{Z}(\omega) - \frac{\partial \tilde{\boldsymbol{f}}}{\partial \tilde{\boldsymbol{x}}} \\
\frac{\partial \boldsymbol{R}}{\partial U} &= \left(\omega^2 \boldsymbol{\nabla}^2 \otimes \frac{\partial \boldsymbol{M}}{\partial U} + \omega \boldsymbol{\nabla} \otimes \frac{\partial \boldsymbol{C}}{\partial U} + \boldsymbol{I}_{n_C} \otimes \frac{\partial \boldsymbol{K}}{\partial U}\right) \tilde{\boldsymbol{x}} - \frac{\partial \tilde{\boldsymbol{f}}}{\partial U} \\
\frac{\partial \boldsymbol{R}}{\partial \omega} &= \left(2\omega \boldsymbol{\nabla}^2 \otimes \boldsymbol{M} + \boldsymbol{\nabla} \otimes \boldsymbol{C}\right) \tilde{\boldsymbol{x}} - \frac{\partial \tilde{\boldsymbol{f}}}{\partial \omega}
\end{aligned} \tag{2.41}$$

where $\frac{\partial \tilde{\boldsymbol{f}}}{\partial \tilde{\boldsymbol{x}}}$, $\frac{\partial \tilde{\boldsymbol{f}}}{\partial U}$ and $\frac{\partial \tilde{\boldsymbol{f}}}{\partial \omega}$ are evaluated with central finite differences with a tuned step size h that balances truncation and round-off error [48].

$$h(x) = \sqrt[3]{\epsilon}\, \max(|x|, 1.0) \tag{2.42}$$

2.3.1 Nonlinear forces evaluation

The vector function $\tilde{\boldsymbol{f}}$ represents the nonlinear forces in the frequency domain. Deriving a closed-form high-order harmonic balance formulation for these forces is feasible only for the simplest cases. The Alternating Frequency-Time (AFT) scheme [49] is a general approach, as illustrated in Fig 2.3,which computes the nonlinear forces at n_S discrete time samples over the oscillation period and then transforms them back to the frequency domain via a Fourier transform, circumventing the need for explicit analytical expressions.

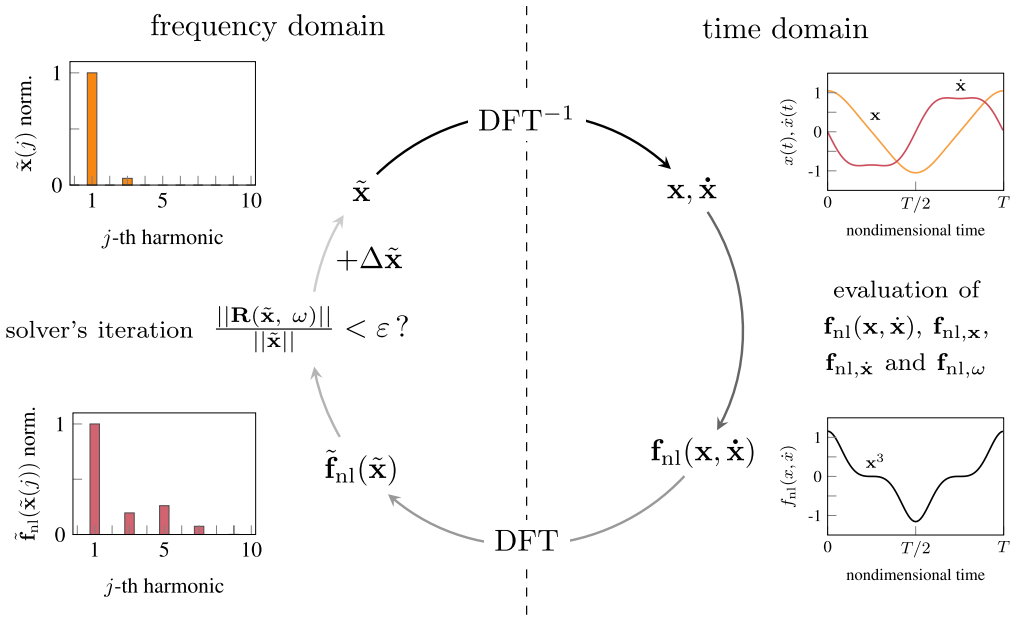


Figure 2.3 Illustration of the Alternating Frequency Time scheme. Reproduced from [3]

The scheme requires careful handling of aliasing, which occurs when high-frequency components are incorrectly represented as lower frequencies due to insufficient sampling. This is particularly critical for discontinuous nonlinearities like freeplay, where sharp transitions at contact points generate high-frequency content that can contaminate the solution if undersampled, see Figure 2.4.

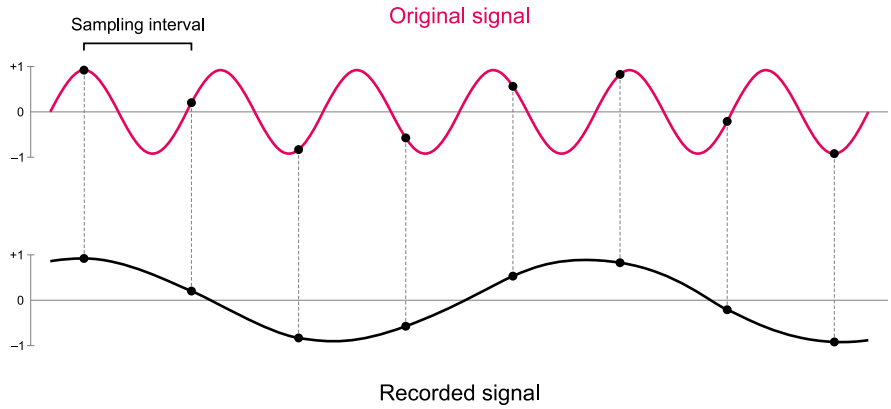


Figure 2.4 Illustration of signal aliasing due to insufficient sampling rate [4]

The Nyquist-Shannon sampling theorem [50] provides a theoretical minimum: to accurately represent a signal containing frequencies up to the n_H-th harmonic, we need at least $n_S \geq 2n_H + 1$ samples per period. However, practical implementations require significantly more samples to capture sharp transitions accurately. In this study, we use $n_S = (n_H + 1) \times 2^8$ samples.

For better performance and numerical accuracy, as shown in Table 2.1, the transformations from and to frequency-domain are performed using the Fast Fourier Transform (FFT) instead of a Discrete Fourier Transform (DFT) [51]. As the HBM operates purely on real values, the efficiency is further improved by using the rFFT variant of the algorithm.

Algorithm	Time Complexity	Roundoff Error
FFT	$O(n \log n)$	$O(\varepsilon \log n)$
DFT	$O(n^2)$	$O(\varepsilon n)$

Table 2.1 Comparison of FFT and DFT in time complexity and numerical accuracy [1]

We define the following linear maps:

$$\begin{aligned} \mathcal{C} &: \mathbb{R}^{n_D \times n_C} \to \mathbb{C}^{n_D \times (n_H+1)}, \hat{\boldsymbol{X}} = \mathcal{C}(\tilde{\boldsymbol{X}}) \\ \mathcal{R} &: \mathbb{C}^{n_D \times (n_H+1)} \to \mathbb{R}^{n_D \times n_C}, \tilde{\boldsymbol{X}} = \mathcal{R}(\hat{\boldsymbol{X}}) \end{aligned} \tag{2.43}$$

Efficient conversion is done in matrix form as:

$$\begin{aligned}
\mathcal{C}: \quad & \hat{\boldsymbol{X}}_{i,1} = \tilde{\boldsymbol{X}}_{i,1},\ \hat{\boldsymbol{X}}_{i,k+1} = \frac{1}{2}\left(\tilde{\boldsymbol{X}}_{i,2k} - i\tilde{\boldsymbol{X}}_{i,2k+1}\right), \quad k = 1, \ldots, n_H \\
\mathcal{R}: \quad & \tilde{\boldsymbol{X}}_{i,1} = \Re\left\{\hat{\boldsymbol{X}}_{i,1}\right\},\ \tilde{\boldsymbol{X}}_{i,2k} = 2\Re\left\{\hat{\boldsymbol{X}}_{i,k+1}\right\},\ \tilde{\boldsymbol{X}}_{i,2k+1} = -2\Im\left\{\hat{\boldsymbol{X}}_{i,k+1}\right\}, \quad k = 1, \ldots, N-1
\end{aligned} \tag{2.44}$$

Then the 2D inverse Fourier transform is applied to obtain the time-domain variables at each discrete time sample for every degree of freedom:

$$\boldsymbol{x} = \mathcal{F}_{n_S}^{-1}(\hat{\boldsymbol{X}}), \quad \dot{\boldsymbol{x}} = \mathcal{F}_{n_S}^{-1}(i\Omega\hat{\boldsymbol{X}}\boldsymbol{D}), \quad \ddot{\boldsymbol{x}} = \mathcal{F}_{n_S}^{-1}\left(-\Omega^2\hat{\boldsymbol{X}}\boldsymbol{D}^2\right),$$

where $\boldsymbol{D} = \operatorname{diag}(0, 1, \ldots, n_H)$, and $\mathcal{F}_{n_S}^{-1}$ denotes the real inverse discrete Fourier transform operator that maps one-sided spectra of length $n_H + 1$ to real time series of length n_S (with implicit Hermitian completion/zero-padding and without normalization).

Equivalently, in indexed form, for $i = 1, \ldots, n_D$ and $m = 0, \ldots, n_S - 1$,

$$\begin{aligned}
\boldsymbol{X}_{i,m} &= \hat{\boldsymbol{X}}_{i,0} + 2\Re\left\{\sum_{k=1}^{n_H} \hat{\boldsymbol{X}}_{i,k} e^{\frac{2\pi i k m}{n_S}}\right\}, \\
\dot{\boldsymbol{X}}_{i,m} &= 2\Re\left\{\sum_{k=1}^{n_H} (i\Omega k)\hat{\boldsymbol{X}}_{i,k} e^{\frac{2\pi i k m}{n_S}}\right\}, \\
\ddot{\boldsymbol{X}}_{i,m} &= 2\Re\left\{\sum_{k=1}^{n_H} \left(-(\Omega k)^2\right)\hat{\boldsymbol{X}}_{i,k} e^{\frac{2\pi i k m}{n_S}}\right\}.
\end{aligned} \tag{2.45}$$

Then the nonlinear forces are evaluated in the time domain at each time sample. Finally the forces are converted back to frequency domain using the normalized forward fourier transform.

$$\hat{\boldsymbol{F}}_{i,k} = \frac{1}{n_S}\sum_{m=0}^{n_S-1} \boldsymbol{f}_{i,m} e^{-\frac{2\pi i k m}{n_S}}, \quad k = 0, \ldots, n_H, \tag{2.46}$$

The five step procedure can be summarized as follows:

$$\tilde{\boldsymbol{X}} \xrightarrow{\mathcal{C}} \hat{\boldsymbol{X}} \xrightarrow{\mathcal{F}^{-1}} \ddot{\boldsymbol{X}}_s, \dot{\boldsymbol{X}}_s, \boldsymbol{X}_s \xrightarrow{\boldsymbol{f}} \boldsymbol{F} \xrightarrow{\mathcal{F}} \hat{\boldsymbol{F}} \xrightarrow{\mathcal{R}} \tilde{\boldsymbol{F}} \tag{2.47}$$

Where $\boldsymbol{F} \in \mathbb{R}^{n_D \times n_S}$ is the matrix of nonlinear forces evaluated at each time sample, $\hat{\boldsymbol{F}} \in \mathbb{C}^{n_D \times (\frac{n_S}{2}+1)}$ is the one-sided spectrum of the nonlinear forces and $\tilde{\boldsymbol{F}} \in \mathbb{R}^{n_D \times n_C}$ is the frequency domain matrix with real coefficients.

2.3.2 Frequency filtering

To further mitigate Gibbs phenomenon effects [52], which manifest as slow-decaying Fourier coefficients due to discontinuities, filtering techniques are applied. Common approaches include raised cosine filters, exponential filters, Fejér averaging, and second-order Lanczos filtering [3]. In this work, the Lanczos filter is used as it is found to provide strong spurious oscillations mitigation without much waveform deformation or phase distortions in the vicinity of discontinuous areas.

$$\boldsymbol{f}(\ddot{\boldsymbol{x}}, \dot{\boldsymbol{x}}, \boldsymbol{x}) \simeq \frac{1}{2}\hat{\boldsymbol{F}}_0 + \sum_{j=1}^{H} \operatorname{sinc}\left(\mathcal{X}_j\right)^m \left[\hat{\boldsymbol{F}}_{2j-1}\cos(j\omega t) + \hat{\boldsymbol{F}}_{2j}\sin(j\omega t)\right] \tag{2.48}$$

where $\operatorname{sinc}(x) = \sin(\pi x)/\pi x$ is the normalized cardinal sine function, $\mathcal{X}_j$ is the base harmonic decay factor, and m is the decay exponent which is set between $[0.1, 2.0]$ with higher values meaning stronger damping.

$$\mathcal{X}_j = \frac{\rho_j}{H+1} \quad \forall j\{1, \ldots, H\} \tag{2.49}$$

with ρ_j the parameter that controls the start of the filtering window:

$$\rho_j = \begin{cases} 0 & \text{for } j < C_H \\ j & \text{for } C_H \leq j \leq H \end{cases} \tag{2.50}$$

C_H being the cutoff harmonic from which the filtering will be applied onward. Based on the assumption that the first harmonic dominates the overall system response [53], it is set to 1 in this study.

This regularization not only improves the fidelity of the frequency-domain representation but also enhances the robustness of nonlinear continuation solvers, particularly for challenging cases where unfiltered harmonics would otherwise destabilize the iterative process in the corrector phase [3].

2.3.3 Harmonic Balance Vortex Lattice Method

The Harmonic Balance Vortex Lattice Method (HBVLM) is a reformulation of the classical UVLM in the frequency domain without being linearized around a frozen geometry and isn't limited to small amplitude oscillations [2].

The circulation is formulated as a truncated Fourier series:

$$\boldsymbol{\Gamma}_j(t) = \sum_{k=0}^{N} \hat{\boldsymbol{\Gamma}}_{jk} \cdot e^{i\omega k t} \tag{2.51}$$

The substitution of (2.51) into the UVLM equation 2.13 gives:

$$\boldsymbol{A} \sum_{k=0}^{N} \hat{\boldsymbol{\Gamma}}_k \cdot e^{i\omega k t} + \boldsymbol{B} \sum_{k=0}^{N} \hat{\boldsymbol{\Gamma}}_{TE,k} \cdot e^{i\omega k (t-(j+1)\Delta t} + \sum_{k=0}^{N} \hat{\boldsymbol{V}}_k \cdot e^{i\omega k t} \cdot \sum_{k=0}^{N} \hat{\boldsymbol{N}}_k \cdot e^{i\omega k t} = 0 \tag{2.52}$$

with $\boldsymbol{A}$ the wing-wing panel influence matrix, $\boldsymbol{B}$ the wing-wake influence matrix and $\boldsymbol{\Gamma}_{TE,k}$ denotes the circulation coefficients at the wing trailing edges.

While this equation can be solved directly, it would involve assembling and solving a size $(n_P \cdot n_C) \times (n_P \cdot n_C)$ matrix, which will inherently be extremely large. For memory efficiency, an alternative formulation: the iterative HB-VLM method [54], is selected. This approach couples the wake circulation through a Discrete Fourier Transform (DFT) and allows the system to be reformulated as a fixed point residual that we can solve using iterative methods. Doing this reduces the memory usage by a factor of n_C^2.

$$\hat{\boldsymbol{\Gamma}}^{k+1} = \boldsymbol{A}^{-1} \boldsymbol{Q}(\hat{\boldsymbol{\Gamma}}^k) \boldsymbol{D} \tag{2.53}$$

where D is the orthonormal real DFT transform matrix.

$$\boldsymbol{D} = \sqrt{\frac{2}{n}} \begin{bmatrix} \frac{1}{\sqrt{2}} & \cos(\omega t_1) & \sin(\omega t_1) & \cdots & \cos(\omega k t_1) & \sin(\omega k t_1) \\ \frac{1}{\sqrt{2}} & \cos(\omega t_2) & \sin(\omega t_2) & \cdots & \cos(\omega k t_2) & \sin(\omega k t_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{1}{\sqrt{2}} & \cos(\omega t_n) & \sin(\omega t_n) & \cdots & \cos(\omega k t_n) & \sin(\omega k t_n) \end{bmatrix} \tag{2.54}$$

$\boldsymbol{Q}$ contains the non-penetration condition and the wake influence. For optimal computational efficiency, the matrix vector product $\boldsymbol{B}\boldsymbol{\Gamma}_W$ should be performed in a matrix-free fashion, otherwise the matrix of coefficients $\boldsymbol{B}$ will become the memory bottleneck.

$$\boldsymbol{Q}_{ij} = -\left[\boldsymbol{V}_i \boldsymbol{N}_i + \boldsymbol{B}\boldsymbol{\Gamma}_W\right]_{t_j} \tag{2.55}$$

where $\boldsymbol{V}$ and $\boldsymbol{N}$ are matrices that store the air flow velocity for each panel in global coordinates and the unitary normal vector of each panel in global coordinates.

The aerodynamic forces are then computed at each time sample using the same UVLM

equations, detailed in section 2.1.2. However, the time derivatives of the circulation are now computed using the Fourier coefficients as a compact matrix-matrix product with the DFT derivative matrix:

$$\frac{\partial \boldsymbol{\Gamma}}{\partial t} = \hat{\boldsymbol{\Gamma}} \left(\frac{\partial \boldsymbol{D}}{\partial t} \right)^{\top} \tag{2.56}$$

The HBVLM fixed point equation 2.53 is solved iteratively using $\hat{\boldsymbol{\Gamma}}^0 = 0$ as initial guess until convergence is reached. To improve on the linear rate of convergence of the simple fixed point iteration, acceleration techniques such as Anderson Acceleration (AA) [43] is employed.

AA works by combining information from several past iterates and residuals instead of using only the most recent one. At each step, it computes an optimal linear combination of recent updates (in a least-squares sense) to construct a corrected next iterate that typically converges much faster than the original fixed-point method. It is a highly scalable method as it doesn't require computation of Jacobians or Hessians and every step is parallelizable. A short description of the algorithm is provided in Appendix A.

2.4 Numerical Continuation

Coexisting stable solutions, such as distinct limit cycle amplitudes, may arise at specific operating conditions, evidenced by amplitude jumps during freestream velocity sweeps [11]. To systematically trace these solution branches, the Harmonic Balance Method is combined with continuation techniques, enabling the prediction a series of steady state responses. Continuation methods leverage prior solutions at adjacent parameter values to generate initial estimates for subsequent computations, progressively advancing through the parameter space. To solve for aeroelastic problems, the fundamental frequency ω and the reduced velocity parameter U are treated as unknowns. While multi-parameter continuation is possible [55], it is more common to vary a single parameter and add an additional constraint for the other parameter.

$$\boldsymbol{Y} = [\tilde{\boldsymbol{x}}, \omega, U]^T \tag{2.57}$$

Solving such extended system involves using a predictor-corrector algorithm with proper parametrisation.

2.4.1 Predictor

The predictor step is responsible for estimating the next point on the solution branch and using it as initial solution for the nonlinear solver. Two broad predictor families are commonly used:

- ODE-type predictors, built from the residual $\boldsymbol{R}$ and its derivatives (tangent-based).
- Polynomial extrapolation predictors, built only from past solutions.

The simplest predictor is a 1st order polynomial extrapolation, also called *Secant Predictor*:

$$\boldsymbol{Y}_{i+1}^{0} = \boldsymbol{Y}_i + \Delta s \frac{\boldsymbol{Y}_i - \boldsymbol{Y}_{i-1}}{\|\boldsymbol{Y}_i - \boldsymbol{Y}_{i-1}\|}$$

Δs is the step size along the solution branch. In practical applications, this predictor is not very robust and can fail to converge. A better alternative is the *Tangent Predictor*, which is also the one used throughout this work. It estimates the direction of the solution branch by computing the local tangent vector to the equilibrium manifold at the current solution. This vector, derived from the null space of the augmented system's Jacobian matrix, provides a first-order approximation of the trajectory in the extended space.

$$\boldsymbol{Y}_{i+1}^{0} = \boldsymbol{Y}_i + \boldsymbol{z}_i \Delta s$$

where $\boldsymbol{z}$ is a unit length vector tangent to the solution branch, determined by solving the linear algebraic equation system using a least-squares solver for stability [56]:

$$\begin{bmatrix} \frac{\partial \tilde{\boldsymbol{R}}^i}{\partial \tilde{\boldsymbol{x}}^i} \\ \boldsymbol{c}^\top \end{bmatrix} \boldsymbol{z}_i = \begin{bmatrix} 0_{(n_E-1)\times 1} \\ 1 \end{bmatrix} \tag{2.58}$$

$\boldsymbol{c}$ is a unit vector defined as:

$$\boldsymbol{c} = \boldsymbol{e}_{i^*}, \quad \text{where} \quad i^* = \arg\max_i \frac{|\boldsymbol{z}_i^{j-1}|}{\max(|\tilde{\boldsymbol{x}}_i^{j-1}|, \epsilon)} \tag{2.59}$$

with $\boldsymbol{e}_{i^*}$ the canonical basis vector (all zeros except 1 at index i^*) and $\epsilon = 10^{-4}$ ensures numerical stability. Additionally, the sign of the predictor may need to be flipped in case it points in the wrong direction [17].

Higher accuracy can be obtained by using a *Polynomial predictor* of degree d (higher-order extrapolation). Let's define the local arclength abscissa s with origin at the last converged point $\boldsymbol{Y}_i$:

$$s_i = 0, \quad s_{i-j} = s_{i-j+1} - \Delta s_{i-j+1} = s_{i-j+1} - \|\boldsymbol{Y}_{i-j+1} - \boldsymbol{Y}_{i-j}\| < 0, \quad j \in \{1, \ldots, d\}.$$

The Lagrange interpolation polynomial $\boldsymbol{P}_d(s)$ satisfying $\boldsymbol{P}_d(s_{i-j}) = \boldsymbol{Y}_{i-j}$ for $j = 0, \ldots, d$ yields

$$\boldsymbol{Y}_{i+1}^0 = \boldsymbol{P}_d(s) = \sum_{j=i-d}^{i} L_j(s)\boldsymbol{Y}_j, \quad L_j(s) := \prod_{\substack{k=i-d \\ k\neq j}}^{i} \frac{s - s_k}{s_j - s_k}, \quad s = s_{i+1} = \Delta p$$

In practice, to mitigate Runge's phenomenon (growing oscillations with equispaced nodes), one typically limits the order to $d \leq 3$.

2.4.2 Corrector

The corrector step is responsible for iteratively solving the nonlinear system starting from the predictor estimate until convergence. A common option is to use a Newton-Raphson solver, which has quadratic convergence under standard assumptions, defined by the recurrent relation:

$$\boldsymbol{Y}^{k+1} = \boldsymbol{Y}^k + \Delta\boldsymbol{Y}^k \quad \text{with} \quad \boldsymbol{J}^k \Delta\boldsymbol{Y}^k = -\boldsymbol{R}\left(\boldsymbol{Y}^k\right) \tag{2.60}$$

where $\boldsymbol{J}^k = \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{Y}}$ is the Jacobian matrix of the residual $\boldsymbol{R}$ evaluated at the current iteration k.

For large-scale systems this method can be computationally demanding because each inner iteration of the corrector requires assembling and factorizing the Jacobian. If an LU factorization with partial pivoting is used, factorization of a dense $(n \times n)$ block costs $\mathcal{O}(n^3)$ floating-point operations, and the triangular solves cost $\mathcal{O}(n^2)$.

To reduce Jacobian recomputation while preserving fast local convergence, a quasi-Newton approach based on Broyden's "good" update can be employed [57]. The idea is to maintain an approximation $B^k \approx J^k$ that is updated by imposing the secant condition along the taken step while changing B^k as little as possible in Frobenius norm. Let

$$s^k = \boldsymbol{Y}^{k+1} - \boldsymbol{Y}^k, \qquad y^k = R(\boldsymbol{Y}^{k+1}) - R(\boldsymbol{Y}^k). \tag{5}$$

The secant equation is $B^{k+1}s^k = y^k$. Broyden's first method enforces this with a rank-one update

$$B^{k+1} = B^k + \frac{\left(y^k - B^k s^k\right)(s^k)^\top}{(s^k)^\top s^k}. \tag{6}$$

At iteration k the step is computed by solving the linear system

$$B^k \Delta \boldsymbol{Y}^k = -R(\boldsymbol{Y}^k), \qquad \boldsymbol{Y}^{k+1} = \boldsymbol{Y}^k + \Delta \boldsymbol{Y}^k, \tag{7}$$

followed by evaluation of $R(Y^{k+1})$ and the rank-one update above. In practice B^0 is initialized with the exact Jacobian J^0. Subsequent linear systems can be solved efficiently using a rank-one update of the factorization, or by applying the Sherman-Morrison formula [58] to update an existing inverse or preconditioned solve. Each Broyden update, as well as the application of B^k to vectors, is $\mathcal{O}(n^2)$, which is quite cheaper than forming and refactorizing a new Jacobian.

These gains come at the cost of robustness. Since B^k is only an approximation to J^k, the linear model can become poor when nonlinearity is strong, when the Jacobian changes rapidly, or near singularities such as folds, which is a frequent occurrence when solving HB systems. The rank-one updates can also degrade conditioning, leading to erratic steps or stagnation. In practical HB problems, this loss of robustness can outweigh the savings from fewer factorizations.

To improve robustness while retaining most of the efficiency, this work adopts Powell's *Hybrid Dogleg method*, implemented in the MINPACK library [27]. The method is "hybrid" in two senses: it uses a trust-region globalization around a Gauss-Newton/Newton step, and it periodically updates or recomputes the Jacobian.

At each iteration, the method constructs a quadratic model of the objective function and seeks a step within a trust region radius. This radius adapts dynamically based on how well the model predicts the actual function reduction. When far from the solution or near singularities, the trust region automatically constrains the step size and ensures descent, preventing the divergence issues that affect Newton and Quasi-Newton methods. Near the solution, the quadratic model becomes accurate, letting the method take full Newton steps and achieving fast local convergence.

From a computational standpoint, MINPACK's implementation achieves efficiency comparable to Broyden methods by maintaining and updating a QR factorization of the Jacobian between iterations. The Jacobian is recomputed only when the trust-region mechanism detects poor agreement between predicted and actual reductions. This significantly reduces the number of Jacobian evaluations while maintaining stability when working with outdated Jacobians thanks to the trust-region method.

2.4.3 Parameterization

Parameter

The parameterization equation constrains the continuation solver by defining a manifold of possible solutions and enforcing the corrector to find a solution on a it.

The simplest form of parameterization is the *Natural/Local Parameterization*, which consists in fixing the value of the continuation parameter at each continuation step.

$$\mathcal{P}\left(\boldsymbol{Y}_{i+1}^{k}\right) = \lambda - \lambda_{i+1}^{k} = 0 \tag{2.61}$$

Where λ is the continuation parameter, k is the current corrector iteration within the continuation step i. This approach is straightforward but will fail to capture complex solution branches with folds or bifurcations. *Arc-Length Parameterization*, illustrated in Fig 2.5, overcomes this limitation by constraining the solution to lie on a hyper-sphere centered on the previous solution [17].

$$\mathcal{P}\left(\boldsymbol{Y}_{i+1}^{k}\right) = \left\|\boldsymbol{Y}_{i+1}^{k} - \boldsymbol{Y}_{i}\right\|_{2}^{2} - \Delta s_{i}^{2} = 0 \tag{2.62}$$

with its derivative:

$$\frac{\partial \mathcal{P}}{\partial \boldsymbol{Y}} = 2\left(\boldsymbol{Y}_{i+1}^{k} - \boldsymbol{Y}_{i}\right) \tag{2.63}$$

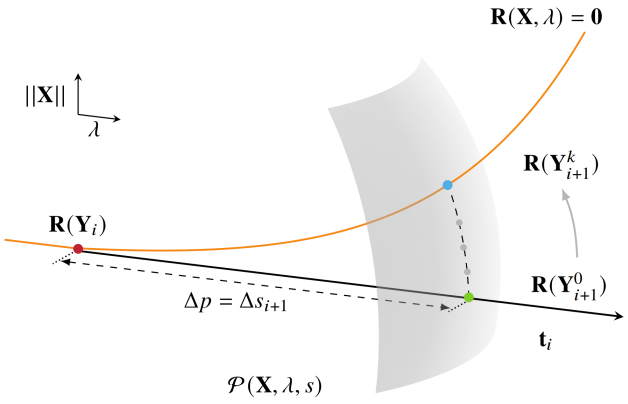


Figure 2.5 Illustration of the Arc-Length Parameterization constraint. Reproduced from [5]

Since this parameterization necessitates a previous solution, a common practice is to use natural parameterization for the first continuation step instead.

Frequency

To resolve the inherent phase ambiguity in autonomous systems arising from time-translation invariance, an integral orthogonal phase condition is introduced [22]. This condition is derived by enforcing orthogonality between the periodic solution and its time derivative, effectively minimizing phase drift relative to a reference trajectory.

$$\mathcal{G}(t) = \int_0^T \boldsymbol{y}_{i+1}(t)^\top \dot{\boldsymbol{y}}_i(t)\,\mathrm{d}t = 0 \tag{2.64}$$

By substituting in the fourier series expressions we obtain the discrete equation:

$$\mathcal{G}_{i+1}^k = \sum_{j=1}^{n_H} j \left(\tilde{\boldsymbol{X}}_{i,2j} \tilde{\boldsymbol{X}}^k_{i+1,2j-1} - \tilde{\boldsymbol{X}}_{i,2k-1} \tilde{\boldsymbol{X}}^k_{i+1,2j} \right) = 0 \tag{2.65}$$

where $\tilde{\boldsymbol{X}}_{i,2j}$ means the $2j$-th column of the coefficient matrix of the i-th continuation step. For $d \in \{1, ..., n_H\}$ we define the column index intervals:

$$\mathcal{I}_d^{\text{odd}} = \Big\{(2d{-}1)n,\ (2d{-}1)n{+}1,\ \ldots,\ 2dn{-}1\Big\} \quad \text{and} \quad \mathcal{I}_d^{\text{even}} = \Big\{2dn,\ 2dn{+}1,\ \ldots,\ (2d{+}1)n{-}1\Big\} \tag{2.66}$$

the partial derivative $\frac{\partial \mathcal{G}}{\partial \tilde{\boldsymbol{Y}}}$ can now be defined in parts:

$$\begin{aligned} \frac{\partial \mathcal{G}}{\partial \tilde{\boldsymbol{x}}}\bigg|^{j+1,k}_{\mathcal{I}_d^{\text{odd}}} &= d\,\tilde{\boldsymbol{X}}_{2d} & \frac{\partial \mathcal{G}}{\partial \lambda} &= 0 \\ \frac{\partial \mathcal{G}}{\partial \tilde{\boldsymbol{x}}}\bigg|^{j+1,k}_{\mathcal{I}_d^{\text{even}}} &= -d\,\tilde{\boldsymbol{X}}_{2d-1} & \frac{\partial \mathcal{G}}{\partial \omega} &= 0 \end{aligned} \tag{2.67}$$

2.4.4 Step Size Control

The efficiency and robustness of the continuation solver depends on the step size along the solution branch. A step size that is too large may cause the corrector iterations to outright fail or converge slowly. Conversely, a very small step size leads to unnecessary computations where the solution barely changes along the branch. Therefore, an adaptive step size control strategy is essential for maintaining an optimal balance between computational efficiency and numerical reliability throughout the continuation process. A simple yet robust strategy featured extensively in the literature is based on adapting the step size in function of the amount of corrector steps it took to converge [59]. By utilizing the fact that we are using a quasi-Newton method as corrector, it is better suited to track a different, but closely related metric: the number of residual function evaluations. In this work the target number is defined

as:

$$n_{\text{opt}} = 6 + n_U \tag{2.68}$$

This target value reflects the expected computational cost for a well-conditioned problem with an appropriately sized predictor step: 6 function evaluation and 1 jacobian evaluation. After each successful continuation step, the actual computational cost is evaluated as

$$n_f = n_{\text{fev}} + n_U \cdot n_{\text{jev}} \tag{2.69}$$

Where n_fev and n_jev are the number of function and jacobian evaluations respectively, returned by the *hybrd* solver upon successful convergence. This formulation accounts for the total computational expense, recognizing that each Jacobian evaluation effectively requires n_J function evaluations when using finite differences. The step size adjustment factor is then computed as the ratio between the optimal and actual number of function evaluations:

$$\xi = \frac{n_{\text{opt}}}{n_f} \tag{2.70}$$

To ensure numerical stability and prevent excessive step size variations, this factor is constrained to lie within reasonable bounds [60]:

$$\xi_{\text{bounded}} = \min\left(\max(\xi, 0.5), 2.0\right) \tag{2.71}$$

This constraint ensures that the step size can at most double or halve between consecutive continuation steps, preventing overly aggressive adjustments that might destabilize the continuation process.

The new step size for the subsequent continuation step is then determined by

$$\Delta s_{k+1} = \Delta s_k \cdot \xi_{\text{bounded}} \tag{2.72}$$

subject to the global constraints

$$\Delta s_{\min} \leq \Delta s_{k+1} \leq \Delta s_{\max} \tag{2.73}$$

where $\Delta s_{\min} = \Delta s_0/10$ and $\Delta s_{\max} = 10\Delta s_0$ represent the minimum and maximum allowable step sizes, with Δs_0 being the initial step size.

In addition, the algorithm incorporates a failure recovery mechanism. When the nonlinear solver fails to converge within the prescribed tolerance, the step size is immediately halved and the predictor-corrector step is reattempted. This process continues until either convergence

is achieved or the minimum step size threshold is reached, at which point the continuation terminates. This dual approach ensures both efficient progress along smooth portions of the solution branch and robust navigation through regions of high curvature or near singular points.

2.4.5 Stability Analysis

The computation of periodic solutions through numerical continuation procedures does not guarantee that it will be observed in practice. A crucial aspect that determines whether a periodic solution is physically realizable is its stability. Without stability analysis, continuation methods may trace unstable solution branches that, while mathematically valid, cannot manifest in real systems due to their sensitivity to infinitesimal perturbations.

The concept of stability for periodic solutions extends the classical Lyapunov stability definition to trajectories that form closed orbits in phase space [61]. A periodic solution $\boldsymbol{\chi}(t)$ with period T is considered stable in the local sense if trajectories starting sufficiently close to it remain bounded and nearby for all future times. More precisely, if we consider a perturbed trajectory $\boldsymbol{x}(t) = \boldsymbol{\chi}(t)+\boldsymbol{\psi}(t)$ where $\boldsymbol{\psi}(t)$ represents a small perturbation, the periodic solution is asymptotically stable when $\|\boldsymbol{\psi}(t)\| \to 0$ as $t \to \infty$. Conversely, the solution is unstable if perturbations grow unbounded over time, causing trajectories to diverge from the limit cycle.

Floquet Theory

Floquet theory [62]provides the mathematical framework for analyzing the stability of periodic solutions in nonlinear dynamical systems. Consider a general nonlinear system described by

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) \tag{2.74}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{\chi}(t)$ represents a periodic solution with period T. To investigate the stability of this periodic solution, we examine the behavior of small perturbations around it. Introducing a perturbed trajectory $\boldsymbol{x}(t) = \boldsymbol{\chi}(t) + \boldsymbol{\psi}(t)$ and substituting into the governing equations yields, after linearization [63], the variational equation

$$\dot{\boldsymbol{\psi}} = \boldsymbol{A}(t)\boldsymbol{\psi} \tag{2.75}$$

where the time-periodic matrix $\boldsymbol{A}(t) = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{\chi}(t)}$ represents the Jacobian evaluated along the periodic orbit. Since $\boldsymbol{\chi}(t)$ is periodic with period T, the matrix $\boldsymbol{A}(t)$ inherits the same periodicity, satisfying $\boldsymbol{A}(t + T) = \boldsymbol{A}(t)$.

The fundamental result of Floquet theory states that the solutions of the linear time-periodic system can be expressed in a special form. If $\boldsymbol{\Psi}(t)$ denotes a fundamental matrix solution whose columns are n linearly independent solutions of the variational equation, then according to Floquet's theorem, this fundamental matrix can be decomposed as

$$\boldsymbol{\Psi}(t) = \boldsymbol{P}(t)e^{\boldsymbol{B}t} \tag{2.76}$$

where $\boldsymbol{P}(t)$ is a periodic matrix with period T and $\boldsymbol{B}$ is a constant matrix. This decomposition reveals that solutions of the variational equation consist of a periodic component modulated by exponential growth or decay, with the stability characteristics entirely determined by the eigenvalues of $\boldsymbol{B}$.

The monodromy matrix $\boldsymbol{M}$ emerges as the central object in practical stability computations. This matrix captures the cumulative effect of the time-periodic dynamics over one complete period and is defined through the fundamental matrix evolution as

$$\boldsymbol{M} = \boldsymbol{\Psi}(0)^{-1}\boldsymbol{\Psi}(T) \tag{2.77}$$

The monodromy matrix satisfies the important property $\boldsymbol{\Psi}(t+T) = \boldsymbol{\Psi}(t)\boldsymbol{M}$, indicating that it maps the state of perturbations from one period to the next. In practical computations, the monodromy matrix is obtained through numerical integration of the variational equation over one period. By choosing the initial condition $\boldsymbol{\Psi}(0) = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix, the numerical integration of

$$\dot{\boldsymbol{\Psi}} = \boldsymbol{A}(t)\boldsymbol{\Psi} \tag{2.78}$$

from $t = 0$ to $t = T$ directly yields $\boldsymbol{M} = \boldsymbol{\Psi}(T)$. This approach requires simultaneous integration of n independent initial conditions, effectively computing the state transition matrix over one period of the limit cycle.

The relationship between the monodromy matrix $\boldsymbol{M}$ and the matrix $\boldsymbol{B}$ in the Floquet decomposition is established through the matrix exponential relation $e^{\boldsymbol{B}T} = \boldsymbol{M}$. The eigenvalues of these matrices, known respectively as Floquet multipliers and Floquet exponents, are fundamentally connected. If $\lambda_{\boldsymbol{M}_i}$ denotes the i-th eigenvalue of the monodromy matrix (Floquet multiplier), then the corresponding eigenvalue of $\boldsymbol{B}$ (Floquet exponent) is given by

$$\lambda_{\boldsymbol{B}_i} = \frac{1}{T}\ln \lambda_{\boldsymbol{M}_i} \tag{2.79}$$

This logarithmic relationship implies that the Floquet exponents are generally complex-valued even when the original system is real, with the real parts determining the growth or

decay rates of perturbations and the imaginary parts characterizing oscillatory behavior of the perturbations around the limit cycle.

The periodic solution is deemed stable if

$$\forall i, \quad \Re(\lambda_{\boldsymbol{B}_i}) < 0 \quad \text{or} \quad |\lambda_{\boldsymbol{M}_i}| < 1 \tag{2.80}$$

In this case, perturbations decay exponentially and nearby trajectories are attracted to the limit cycle. The solution is unstable if at least one Floquet multiplier satisfies $|\lambda_{\boldsymbol{M}_i}| > 1$, equivalently when at least one Floquet exponent has a positive real part, causing perturbations to grow exponentially.

Hill's Method

Hill's method provides a frequency-domain approach to Floquet stability analysis, making it particularly well-suited for use with a HBM solver.

The method begins by perturbing the periodic solution $\boldsymbol{x}^*(t)$ with a disturbance term:

$$\boldsymbol{x}(t) = \boldsymbol{x}^*(t) + e^{\Lambda t}\boldsymbol{s}(t) \tag{2.81}$$

where $\boldsymbol{s}(t) \in \mathbb{R}^{n\times 1}$ is a periodic perturbation that can be approximated using a Fourier series truncated to order H:

$$\boldsymbol{s}(t) = \sum_{k=1}^{2H+1} \boldsymbol{s}_k \boldsymbol{T}_k(\omega t)$$

After substitution into the equations of motion and some lengthy algebra [64], the stability analysis transforms into a quadratic eigenvalue problem (QEP):

$$\left(\boldsymbol{\Delta}_2\Lambda^2 + \boldsymbol{\Delta}_1\Lambda + \boldsymbol{J}\right) e^{\Lambda t}\boldsymbol{S} = \boldsymbol{0} \tag{2.82}$$

where $\boldsymbol{J} = \frac{\partial \boldsymbol{R}}{\partial \tilde{\boldsymbol{x}}}$, Λ and S are the complex eigenvalues and eigenvectors of the truncated problem, and the matrices $\boldsymbol{\Delta}_2$ and $\boldsymbol{\Delta}_1$ are constructed from the mass and damping matrices:

$$\begin{aligned} \boldsymbol{\Delta}_2 &= \boldsymbol{I}_{n_C} \otimes \boldsymbol{M} \\ \boldsymbol{\Delta}_1 &= 2\omega\nabla \otimes \boldsymbol{M} + \boldsymbol{I}_{n_C} \otimes \boldsymbol{C} \end{aligned} \tag{2.83}$$

The QEP is converted to a standard eigenvalue problem through the companion matrix formulation:

$$(\boldsymbol{H} - \Lambda \boldsymbol{I}_{2n_U})\boldsymbol{Q} = \boldsymbol{0} \tag{2.84}$$

where the Hill matrix $\boldsymbol{H} \in \mathbb{R}^{2n_U \times 2n_U}$ is given by:

$$\boldsymbol{H} = \begin{bmatrix} \mathbf{0} & \boldsymbol{I}_{n_U} \\ -\boldsymbol{\Delta}_2^{-1}\boldsymbol{J} & -\boldsymbol{\Delta}_2^{-1}\boldsymbol{\Delta}_1 \end{bmatrix} \tag{2.85}$$

For first-order systems, $\boldsymbol{H} \in \mathbb{R}^{n_U \times n_U}$ reduces to:

$$\boldsymbol{H} = -\boldsymbol{\Delta}_1^{-1}\boldsymbol{J} \tag{2.86}$$

The eigenvalue problem yields $2n_U$ eigenvalues, but only $2n_D$ have physical meaning as approximations to the Floquet exponents. The remaining eigenvalues are artifacts of the harmonic truncation. The filtering approach implemented here selects eigenvalues based on the magnitude of their imaginary parts. This method is robust and has been formally proven to converge with a sufficient number of harmonics [65].

$$\begin{aligned} \boldsymbol{s} &= \operatorname{argsort}(|\Im(\boldsymbol{\Lambda})|) \\ (\lambda_{B_1}, \ldots, \lambda_{B_k}) &= (\Lambda_{s_1}, \ldots, \Lambda_{s_k}) \end{aligned} \tag{2.87}$$

One observed downside of Hill's method as observed in practice, is the large number of harmonics needed (more than 15) until bifurcation identification is performed accurately.

Bifurcations

In numerical continuation, bifurcations represent changes in the dynamical system behavior as parameters vary. At bifurcation points, the system becomes singular and exhibits specific eigenvalue configurations signaling the emergence, disappearance, or stability change of solution branches.

Test functions provide a way of detecting bifurcations during the continuation process. These scalar-valued functions $\psi : \mathbb{R}^n \to \mathbb{R}$ are designed to change sign when the system crosses a bifurcation point. A bifurcation is detected between continuation steps $i-1$ and i when:

$$\operatorname{sign}(\psi_{i-1}) \neq \operatorname{sign}(\psi_i) \tag{2.88}$$

This condition is efficiently evaluated using the copysign function to avoid numerical overflow issues that may arise in alternative formulations. The specific form of the test function depends on the bifurcation type being monitored.

Limit Point (LP). The limit point bifurcation occurs when the solution curve becomes

vertical with respect to the continuation parameter, marking where two solution branches with different stability properties meet. The test function monitors the derivative of the continuation parameter ω along the arc-length parameterized path:

$$\psi_{LP} = \frac{\omega_i - \omega_{i-1}}{|\omega_i - \omega_{i-1}|} \tag{2.89}$$

This function changes sign when the solution curve reverses direction, indicating a fold in the solution manifold.

Branch Point (BP). Branch points mark locations where distinct solution branches intersect. At these points, the extended Jacobian matrix becomes singular, reflecting the non-uniqueness of the tangent direction. The test function evaluates:

$$\psi_{BP} = \det(\boldsymbol{J}) \tag{2.90}$$

where $\boldsymbol{J}$ is the Jacobian of the extended continuation system. The determinant sign switch signals a branch intersection, and additional computations can be performed to determine the tangent directions of intersecting branches.

Neimark-Sacker (NS). For periodic solutions, the Neimark-Sacker bifurcation occurs when a complex conjugate pair of Floquet multipliers crosses the unit circle, leading to quasi-periodic solutions. The test function employs the bialternate product:

$$\psi_{NS} = \det(2(\tilde{\boldsymbol{A}} \odot \boldsymbol{I})) \tag{2.91}$$

where $\tilde{\boldsymbol{A}} = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ contains the Floquet exponents and $\boldsymbol{I}$ is the identity matrix. The bialternate product $\boldsymbol{C} = \boldsymbol{A} \odot \boldsymbol{B}$ of two $n \times n$ matrices produces an $m \times m$ matrix with $m = n(n-1)/2$, where each element is constructed as follows:

$$[\boldsymbol{A} \odot \boldsymbol{B}]_{(p,q),(r,s)} = \frac{1}{2}(a_{pr}b_{qs} - a_{ps}b_{qr} + a_{qs}b_{pr} - a_{qr}b_{ps}) \tag{2.92}$$

with indices (p, q) and (r, s) representing ordered pairs where $p > q$ and $r > s$.

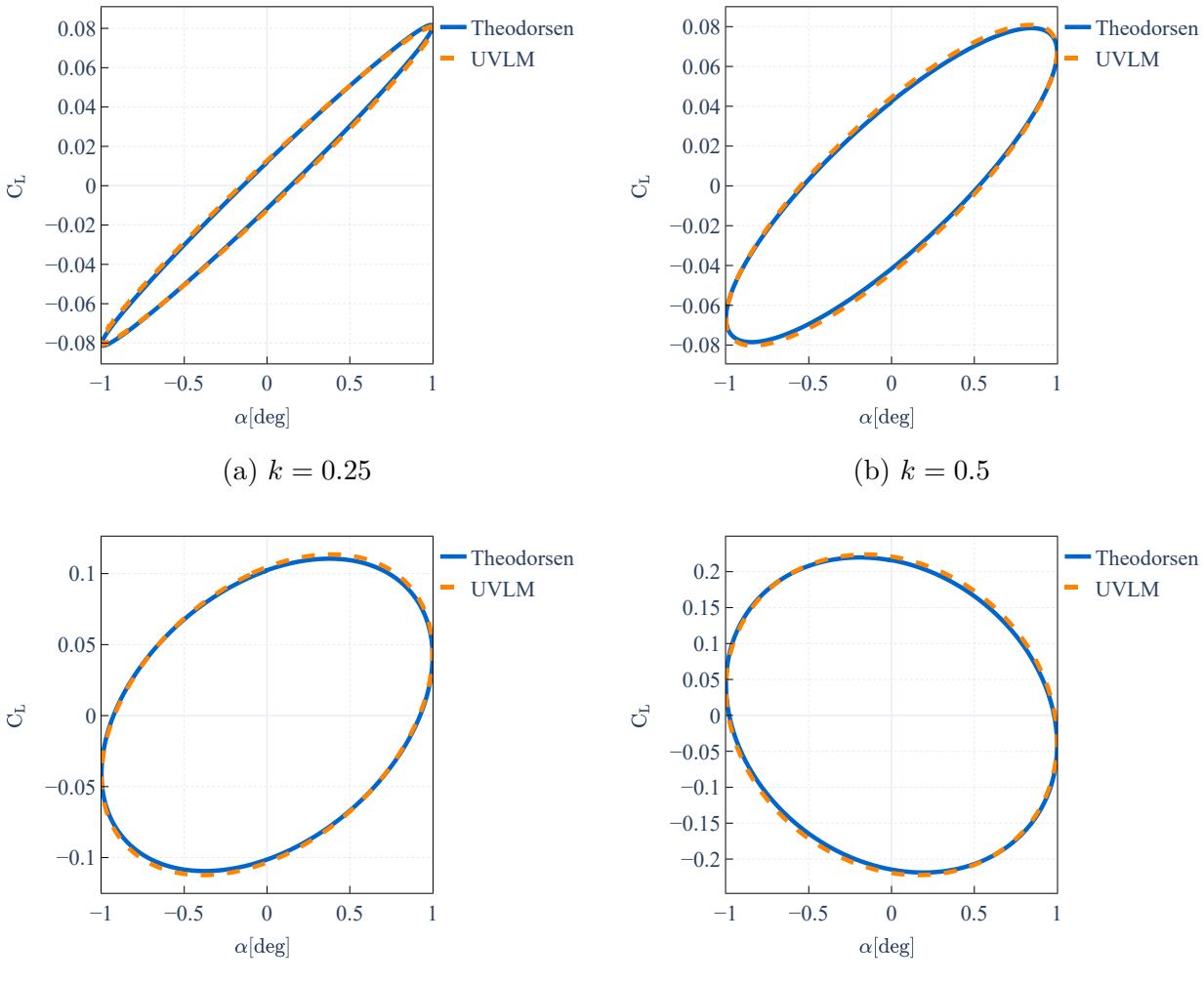
Period-Doubling (PD). Period-doubling bifurcations occur when a Floquet multiplier crosses through -1, typically leading to stable periodic orbits with twice the original period. The test function examines the dominant Floquet multiplier:

$$\psi_{PD} = \mathrm{sign}(\Re(\max \lambda_M)) \cdot |\max \lambda_M| + 1 \tag{2.93}$$

CHAPTER 3 VERIFICATION AND RESULTS

3.1 UVLM verification

The UVLM implementation is verified for 2-DOF plunge-pitch rigid wing formulated in the time domain for general motions. The Theodorsen state-space formulation provides an analytical reference for unsteady aerodynamic forces and is detailed in Appendix B. Full verification was performed for both plunging and pitching motions; here we present results for pitching motions only.



(a) $k = 0.25$ (b) $k = 0.5$

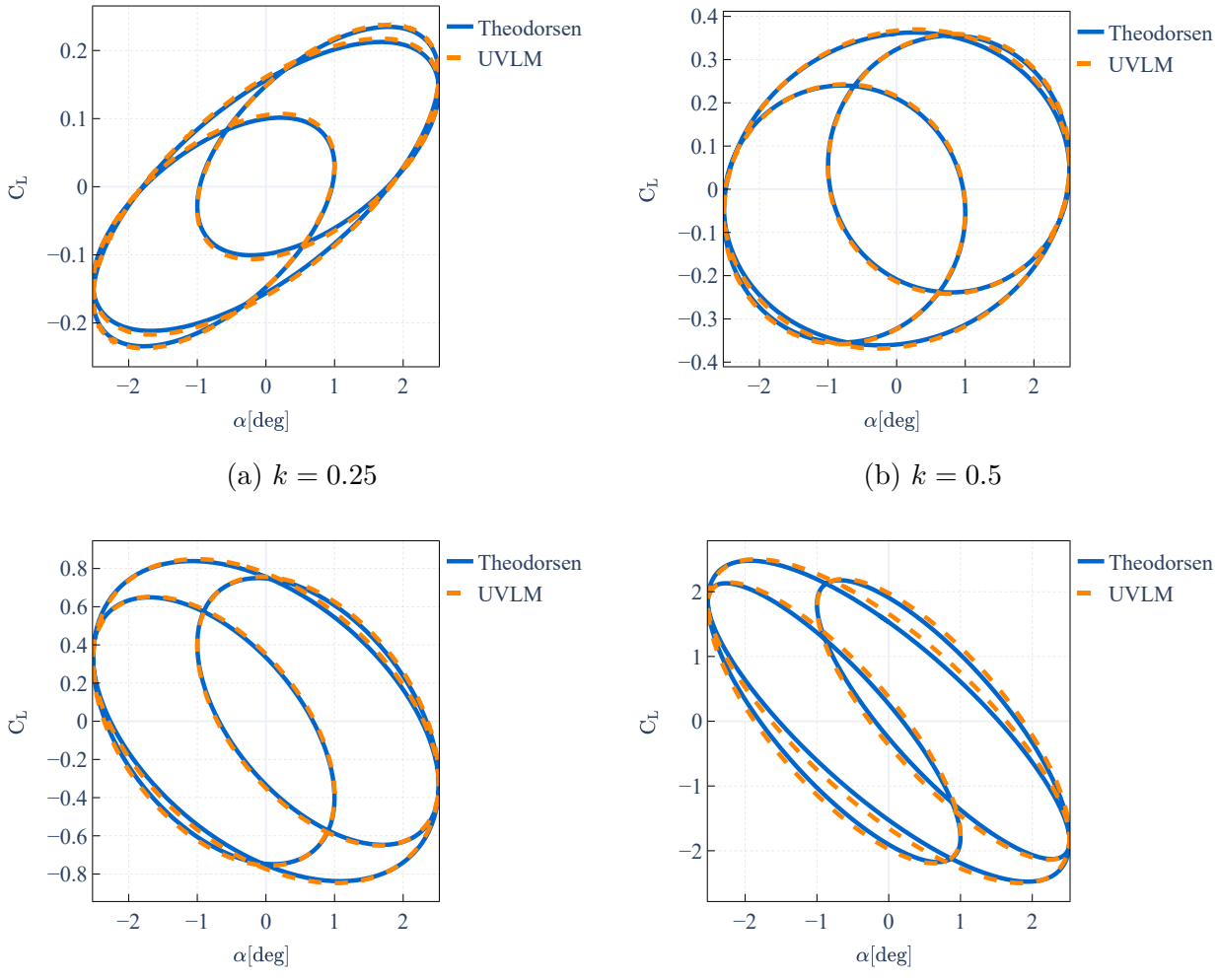
(c) $k = 1.0$ (d) $k = 2.0$

Figure 3.1 Lift coefficient for monoharmonic pitching airfoil at various reduced frequencies

Starting with monoharmonic periodic motions with small amplitudes, the pitching motion is prescribed as $\alpha(t) = \sin(\omega t)$ [deg] with the pitch axis located at the quarter-chord. The reduced frequency k is varied from 0.25 to 2.0.

To represent a 2D wing in 3D space, a rectangular wing with aspect ratio 20000 is used, discretized with 80 panels chordwise and 5 panels spanwise for a total of 400 panels. Simulations are run for three complete motion cycles to ensure solution convergence.

Figure 3.1 shows good agreement between the UVLM results and the Theodorsen reference across all reduced frequencies with a solution error very similar to that found in other UVLM implementations [14]. Additionally, these results are consistent with those reported by Gao et al. [66].



(a) $k = 0.25$

(b) $k = 0.5$

(c) $k = 1.0$

(d) $k = 2.0$

Figure 3.2 Lift coefficient for biharmonic pitching airfoil at various reduced frequencies

Aeroelastic LCO typically exhibits significant higher harmonic content in the response. Therefore, the UVLM implementation must be validated for multiharmonic motions to ensure accurate representation of nonlinear aeroelastic phenomena. This validation also tests the kinematics module's ability to compute positions and surface velocities for complex motion profiles.

A biharmonic pitching motion is prescribed as $\alpha(t) = \sin(\omega t) + 2\sin(3\omega t)$ [deg], combining a fundamental frequency component with its third harmonic. The mesh resolution is increased to 160 panels chordwise while maintaining 5 panels spanwise (800 panels total) to adequately capture the higher frequency content. The aspect ratio and reduced frequency range remain unchanged.

The results in Figure 3.2 demonstrate good agreement with the Theodorsen reference for multiharmonic motions. The UVLM accurately captures both the amplitude and phase relationships between harmonic components, confirming its suitability for nonlinear aeroelastic analysis where higher harmonic content is present. For brevity, mesh convergence studies are not shown here but can be found in [14].

3.2 HBVLM Verification

The HBVLM implementation can be verified against both the Theodorsen model and the previously verified UVLM implementation for periodic motions. Since the HBVLM is a direct reformulation of the UVLM in frequency domain, it should yield identical results for the same motion profiles.

For the biharmonic pitching motion $\alpha(t) = 3\sin(\omega t) + 2\sin(3\omega t)$ [deg] with $\omega = k\frac{U_\infty}{b}$, Figure 3.3 compares the lift and moment coefficients along with the wing circulation values between the HBVLM and UVLM implementations at a reduced frequency of $k = 0.1$ on a 400 panels wing and nondimensionalized by its half-chord $b = 0.5$.

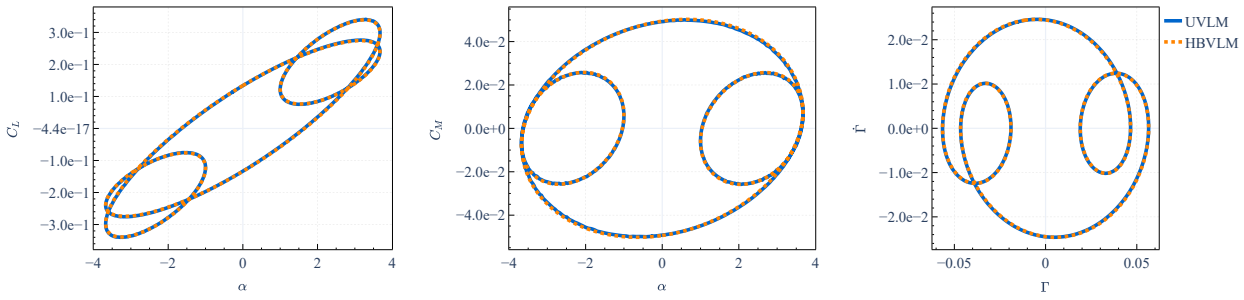


Figure 3.3 Coefficients and circulation for a biharmonic pitching motion at $k = 0.1$

From the motion equation, it is clear that 3 harmonics are required to exactly represent the motion. Therefore, the HBVLM simulation is done with 3 harmonics up to a iterative convergence tolerance of 10^{-8}. Double precision arithmetic is strongly recommended for HBVLM simulations as single precision yields significant numerical errors.

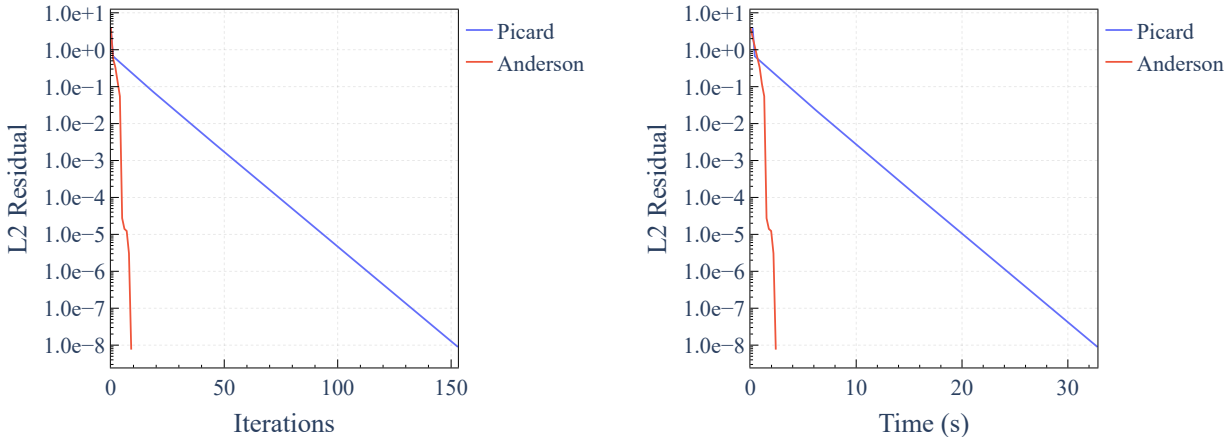
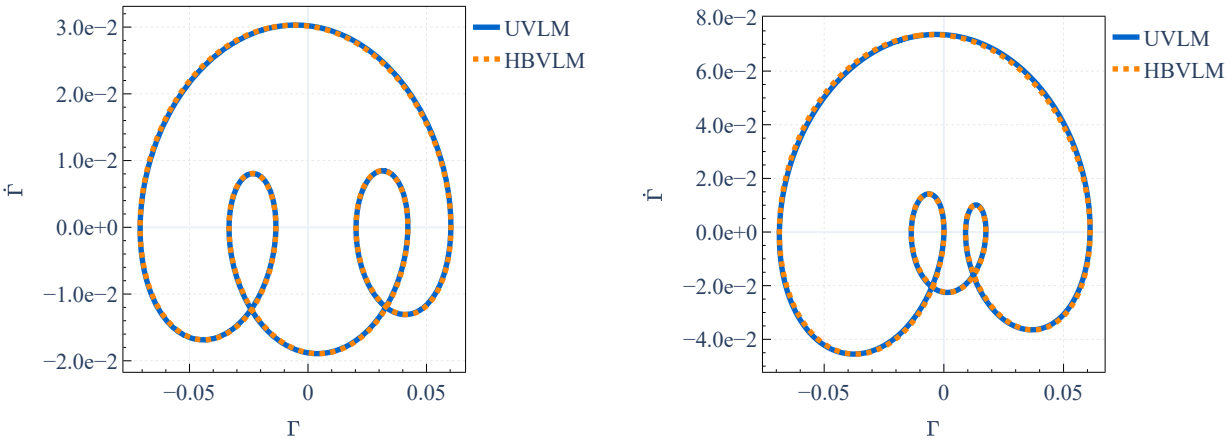


Figure 3.4 Convergence of HBVLM solution with 3 harmonics

Figure 3.4 illustrates the substantial convergence acceleration achieved by the fixed-point formulated HBVLM when employing Anderson Acceleration with a memory depth of $m = 5$, compared to standard fixed-point (Picard) iterations. This amounts to speedups around $10\times$ in practical applications, while requiring only modest additional memory overhead for storing the m previous iterates and residuals.



(a) $k = 0.1$ (b) $k = 0.25$

Figure 3.5 Circulation Poincaré sections for mixed heave and pitching motion

Aeroelastic LCOs involve multiple if not all of the degrees of freedom simultaneously, therefore the HBVLM is also verified for mixed heave and pitching motions. The pitch motion is identical to the previous biharmonic case, and a periodic heave motion is added as $h(t) = 0.05\cos(2\omega t)$. Results are obtained using the same wing discretized with 40 panels chordwise. Figure 3.5 shows excellent agreement between the HBVLM and UVLM implementations.

3.3 Coupled Duffing oscillators

The 2-DOF coupled Duffing oscillator [67] is a classic nonlinear dynamics academic problem that is very well suited for verification of the multi-dof continuation solver and bifurcation detection and identification. The non autonomous system, represented on Figure 3.6, consists of two masses connected to ground with linear and cubic springs, and to each other with a linear spring. The first mass is forced with a harmonic force while the second mass is unforced.

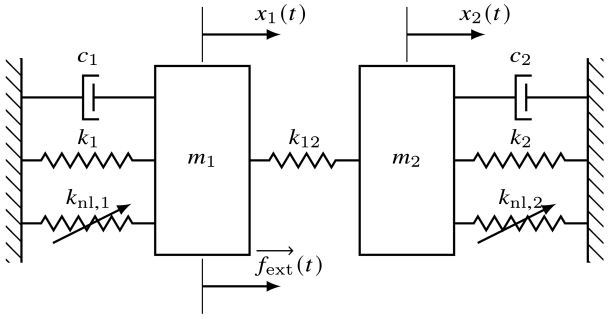


Figure 3.6 2-DOF coupled Duffing oscillator with nonlinear stiffness

The equations take the form of 2.25 with the following matrices:

$$\boldsymbol{M} = \begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix}, \quad \boldsymbol{C} = \begin{bmatrix} c_1 & 0 \\ 0 & c_2 \end{bmatrix} \quad \text{and} \quad \boldsymbol{K} = \begin{bmatrix} k_1 + k_{12} & -k_{12} \\ -k_{12} & k_2 + k_{12} \end{bmatrix} \tag{3.1}$$

and the nonlinear force vector that contains both the excitation term and the cubic stiffness is given by:

$$\boldsymbol{f} = \begin{bmatrix} F_0\cos(\omega t) - k_{\mathrm{nl},1}x_1^3 \\ -k_{\mathrm{nl},2}x_1^3 \end{bmatrix} \tag{3.2}$$

Numerical continuation is performed on this system with the forcing frequency ω as the continuation parameter over the range $\omega \in [0.05, 6.0]$ rad/s. Solution was initialized from a FFT approximation of the time domain solution obtained from direct time integration at $\omega = 0.05$ rad/s. Results were obtained using 15 harmonics and with an initial step size $\Delta s = 0.02$. The LCO branch is plotted as a spectral frequency response curve (FRC) in

terms of the root mean square (RMS) of the displacement of each mass as a function of the forcing frequency ω. Parseval's Theorem states that the average power in the time domain equals the sum of powers in the frequency domain [68]:

$$\mathrm{RMS}(x(t)) = \sqrt{\frac{1}{T}\int_0^T x^2(t)dt} = \sqrt{a_0^2 + \frac{1}{2}\sum_{j=1}^{H}\left(a_j^2 + b_j^2\right)} \tag{3.3}$$

The first parameter set (Table 3.1) represents a classic hardening Duffing oscillator with strong linear coupling, which produces isolated resonance curves. This is a relatively straightforward case for a continuation solver, requiring only 758 iterations to complete.

Table 3.1 Parameters of the dynamical system of Figure 3.6

Parameter	DOF 1	DOF 2
m_i	$1\,\mathrm{kg}$	$1\,\mathrm{kg}$
c_i	$0.1\,\mathrm{N\,m^{-1}\,s}$	$0.1\,\mathrm{N\,m^{-1}\,s}$
k_i	$1\,\mathrm{N\,m^{-1}}$	$1\,\mathrm{N\,m^{-1}}$
$k_{\mathrm{nl},i}$	$1\,\mathrm{N\,m^{-3}}$	$1\,\mathrm{N\,m^{-3}}$
k_{12}	$5\,\mathrm{N\,m^{-1}}$	
F_0	$2\,\mathrm{N}$	

The results are shown in Figure 3.7 and demonstrate excellent agreement with the reference solution from Colaïtis [5].

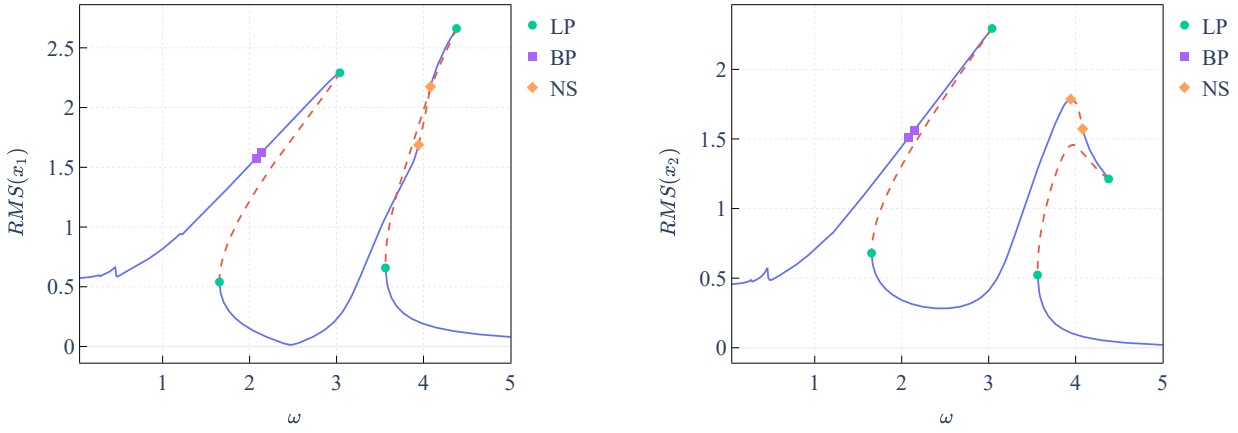


Figure 3.7 LCO branch for the 2-DOF coupled Duffing oscillator 3.1

A more challenging case is considered next with parameters given in Table 3.2.

Table 3.2 Parameters of the dynamical system of Figure 3.6

Parameter	DOF 1	DOF 2
m_i	$1\,\mathrm{kg}$	$1\,\mathrm{kg}$
c_i	$0.1\,\mathrm{N\,m^{-1}\,s}$	$0.1\,\mathrm{N\,m^{-1}\,s}$
k_i	$1\,\mathrm{N\,m^{-1}}$	$1\,\mathrm{N\,m^{-1}}$
$k_{\mathrm{nl},i}$	$1\,\mathrm{N\,m^{-3}}$	$1\,\mathrm{N\,m^{-3}}$
k_{12}	$1\,\mathrm{N\,m^{-1}}$	
F_0	$2\,\mathrm{N}$	

The linear coupling is severely weakened which leads to more complex phenomena such as modal energy transfer. The results are shown in Figure 3.8.

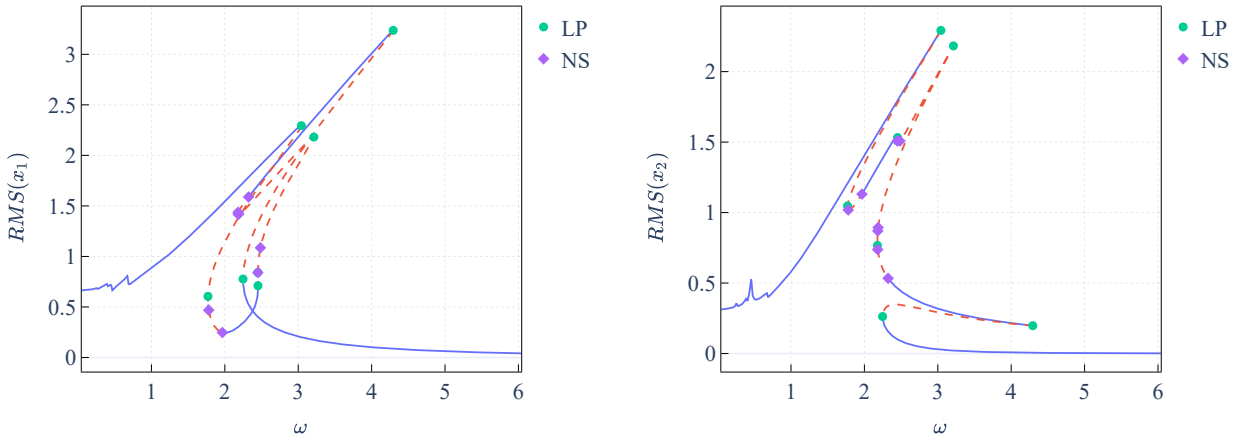


Figure 3.8 LCO branch for the 2-DOF coupled Duffing oscillator 3.2

The continuation solver completed the branch in 1158 iterations, requiring two restarts with a reduced step size due to convergence failures. The results demonstrate that the Hill method successfully captures the system's overall stability but produces spurious Neimark-Sacker (NS) bifurcation points. This is a known artifact of the method, as it approximates Floquet multipliers using a truncated Fourier series. While 15 harmonics were sufficient to model the nonlinear response, they proved to be insufficient for precise stability analysis. A test with 25 harmonics yielded no improvements, indicating that the solution likely requires a more advanced eigenvalue filtering technique, such as FEF-RP [69] or a sorting-free method

[70], rather than simply increasing the number of harmonics. Other test cases, omitted here for brevity, confirmed the method's robustness in different scenarios such as systems with asymmetric cubic-quadratic springs, which successfully validated the correct handling of mixed odd and even harmonics under more severe nonlinearity.

3.4 Van der Pol oscillator

The Van der Pol oscillator represents a fundamental nonlinear dynamical system that exhibits self-sustained oscillations through limit cycle behavior, this makes it an ideal test case to verify numerical continuation with the response frequency as an unknown. Originally developed to model vacuum tube oscillator circuits, it describes a triode with a nonlinear resistance characterized by a cubic current-voltage relationship [71].

While single degree-of-freedom Van der Pol oscillators solved using high-order harmonic balance methods have been extensively studied in the literature [72], this work examines a coupled two degree-of-freedom system.

The system's equations take the form of 2.25 with the following matrices:

$$\boldsymbol{M} = \begin{bmatrix} \gamma & 0 \\ 0 & \gamma \end{bmatrix}, \quad \boldsymbol{C} = \begin{bmatrix} -\mu & 0 \\ 0 & -\mu \end{bmatrix} \quad \text{and} \quad \boldsymbol{K} = \begin{bmatrix} 1+\kappa & -\kappa \\ -\kappa & 1+\kappa \end{bmatrix} \tag{3.4}$$

The force vector, which introduces the characteristic nonlinear damping term, takes the form:

$$\boldsymbol{f} = \begin{bmatrix} -\mu x_1^2 \dot{x}_1 \\ -\mu x_2^2 \dot{x}_2 \end{bmatrix} \tag{3.5}$$

The following parameters were used: $\gamma = 1.0$ and $\kappa = 1.0$, with the nonlinear damping coefficient μ serving as the continuation parameter over the range $[0.1, 5.0]$.

The continuation procedure was initialized using a Fourier approximation of the steady-state solution obtained through direct time integration at $\mu = 0.1$. The time integration employed initial conditions $x_1(0) = x_2(0) = 3.0$ and $\dot{x}_1(0) = \dot{x}_2(0) = 0.0$, ensuring convergence to the upper limit cycle branch (the system has two solution branches). The harmonic balance solution was obtained using 15 harmonics with an initial continuation step size of $\Delta s = 0.02$.

The continuation solver required 2381 steps to trace the complete solution branch over the specified parameter range. Figure 3.9 presents the evolution of both the root-mean-square (RMS) amplitude and the fundamental frequency as functions of the continuation parameter. The HBM results are validated against discrete points obtained from time integration (TI).

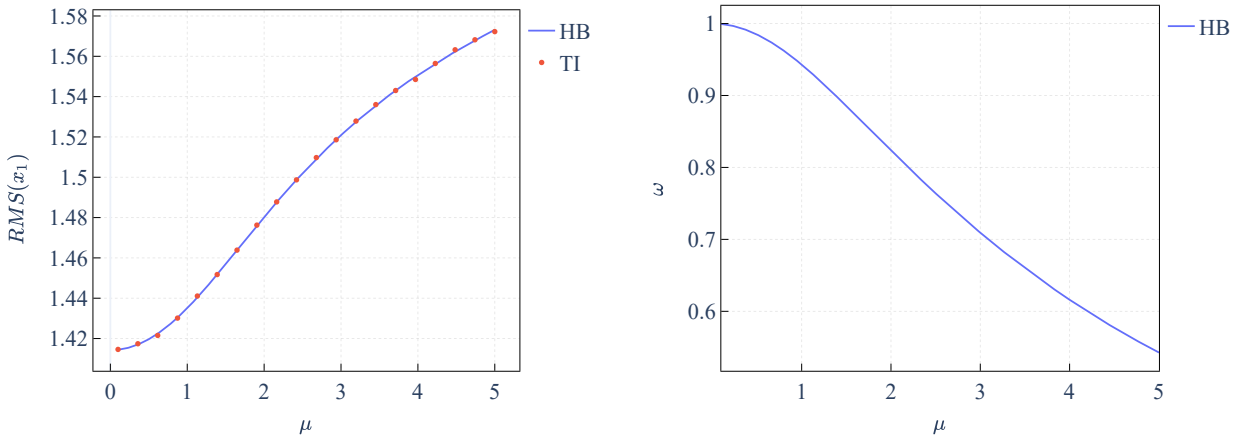
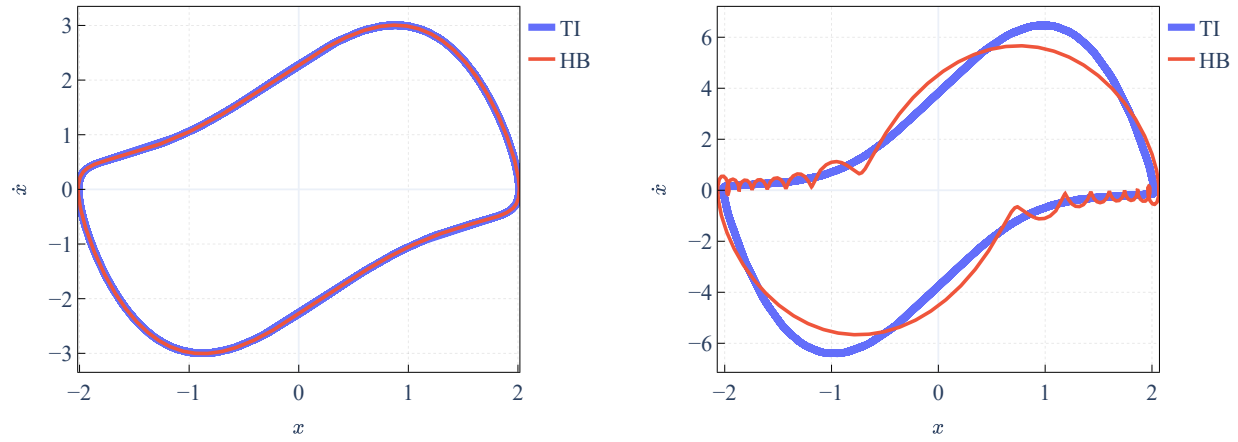


Figure 3.9 LCO characteristics for the 2-DOF coupled Van der Pol oscillator

For the time-domain solutions, the RMS values were computed over sufficiently long time windows to ensure convergence of the average power, approximating the ideal requirement of integration over an integer number of periods.

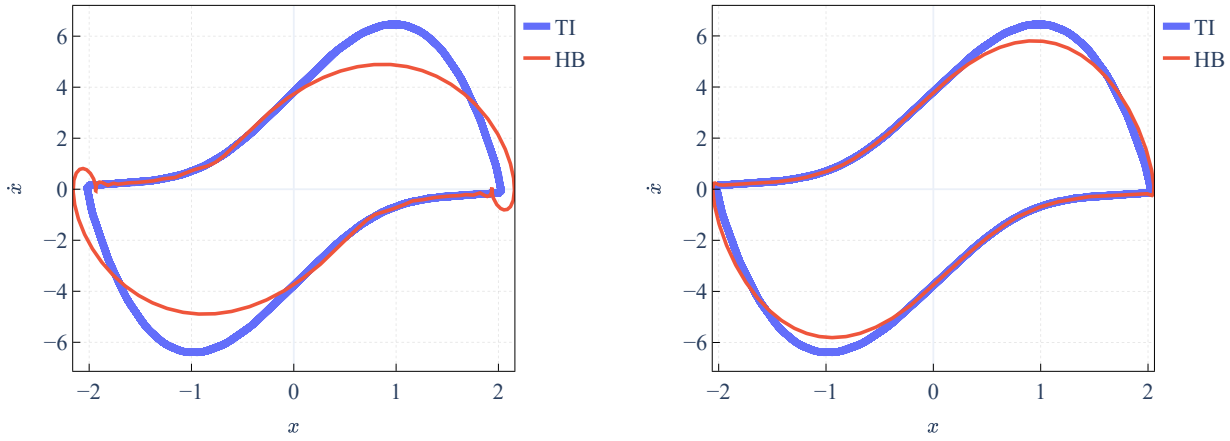


(a) $H = 15, \quad \mu = 1.3$ (b) $H = 15, \quad \mu = 4.0$

Figure 3.10 Poincaré sections without frequency filtering.

Poincaré sections provide an effective visualization tool for autonomous systems, as they capture the periodic orbit characteristics independent of phase. Figure 3.10 shows excellent agreement between HBM and time integration for moderate values of μ. However, as μ increases, the limit cycle transitions from nearly sinusoidal to relaxation oscillations characterized by sharp transitions. This non-smooth behavior manifests as Gibbs phenomenon [73], introducing spurious high-frequency oscillations that distort the reconstructed time signal.

These artifacts become particularly pronounced in the Poincaré sections at high μ values, where accurate representation would require an impractically large number of harmonics.



(a) $H = 15, \quad \mu = 4.0$ (b) $H = 30, \quad \mu = 4.0$

Figure 3.11 Poincaré sections with Lanczos filtering ($m = 0.5$).

To address these numerical artifacts, a filtering technique such as Lanczos's filter (detailed in Section 2.3.2) can be applied. Figure 3.11 illustrates the effectiveness of this approach. With 15 harmonics and a filter parameter $m = 0.5$, the spurious oscillations are successfully suppressed at the expense of a heavily reduced peak amplitude. Increasing the harmonic content to 30 recovers most the of signal and yields good agreement with the time integration reference solution.

3.5 UVLM 3-DOF Wing

To establish a foundation for implementing HB-VLM in multi-lifting surface configurations, we first verify the UVLM implementation for aeroelastic applications. The UVLM is coupled with a three-degree-of-freedom (3-DOF) rigid wing model incorporating plunge, pitch, and control surface deflection.

This standard model has been extensively studied in the literature, particularly in its linear configuration [74]. Linear stability analysis (omitted for brevity) identifies the flutter speed as $U_f = 23.96 \text{ m s}^{-1}$ [6]. The system's equations of motion and corresponding Theodorsen aerodynamic matrices are provided in Appendix D, while all system parameters are listed in Appendix C.

The aerodynamic forces from the 3D UVLM are mapped to the 2D wing through the following

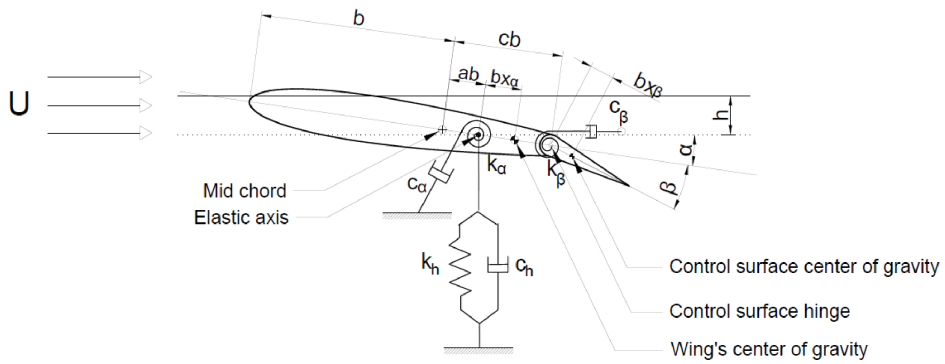


Figure 3.12 Schematic of a 2D airfoil with control surface [6].

transformation:

$$\boldsymbol{f_a} = \frac{V^2}{\pi} \begin{bmatrix} -CL \\ 2\,CM_\alpha \\ \frac{(1-c)^2}{2}\,CM_\beta \end{bmatrix} \tag{3.6}$$

For verification, we introduce nonlinearity through the control surface stiffness:

$$\boldsymbol{f_s} = \begin{bmatrix} 0 \\ 0 \\ -\left(\frac{\omega_\beta}{\omega_\alpha}\right)^2 r_\beta^2 \mathcal{M}(\beta) \end{bmatrix} \tag{3.7}$$

We analyze the system's limit cycle oscillations (LCOs) using bifurcation diagrams and Poincaré sections across various flow speeds. Bifurcation diagrams track the evolution of LCO peak amplitudes as functions of flow speed $U \in [0.2, 22]$ m s^{-1}, revealing transitions between stable period-n motions and chaotic regions. Poincaré sections visualize the periodic orbit structure while remaining phase-independent, which is particularly useful for this autonomous system.

Two representative nonlinearities commonly found in the literature are investigated: cubic stiffness and freeplay. Theodorsen solutions are obtained using a state-space formulation integrated with an adaptive Runge-Kutta 5(4) scheme. The UVLM discretization employs 45 chordwise panels for the main wing and 15 for the control surface, with 5 spanwise panels for both surfaces. The coupled system is integrated using an implicit Newmark-β scheme [44] with iterative solver coupling until achieving convergence to 10^{-8}. Each simulation spans 10 seconds to ensure LCO convergence, initialized with a small heave displacement of 0.01 m. Time series data is presented for reference in Appendix E for both configurations.

3.5.1 Cubic Nonlinearity

The cubic stiffness is modeled as a polynomial function of control surface deflection:

$$\mathcal{M}_p(\beta) = \beta_0 + \beta_1\beta + \beta_2\beta^2 + \beta_3\beta^3 \tag{3.8}$$

where we set $\beta_0 = \beta_1 = \beta_2 = 0$ and $\beta_3 = 1$ for this study.

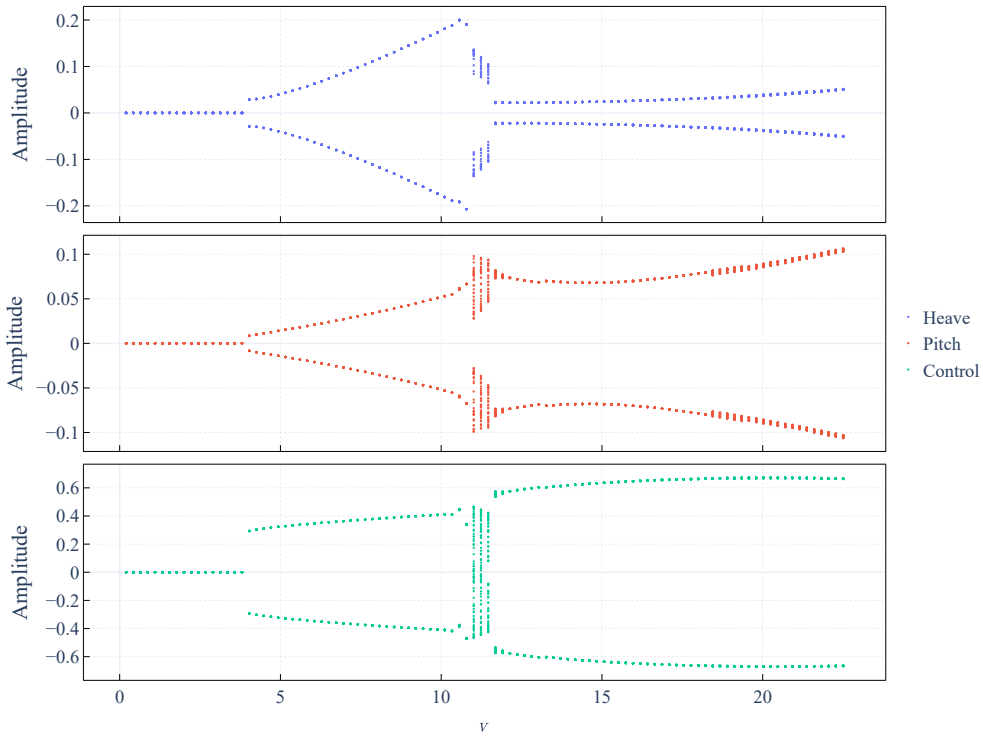


Figure 3.13 Bifurcation diagram for the 3-DOF wing with cubic control surface stiffness.

Figure 3.13 reveals a period-1 LCO emerging at approximately 4 $\mathrm{m\,s^{-1}}$, with amplitude increasing until 11 $\mathrm{m\,s^{-1}}$, where the system transitions to chaos. At 12 $\mathrm{m\,s^{-1}}$, the system returns to period-1 motion with continuously increasing amplitude beyond 22 $\mathrm{m\,s^{-1}}$. Note that time-domain bifurcation diagrams depend on initial conditions and are therefore non-unique.

The Poincaré sections, shown in Figure 3.14,demonstrate excellent agreement between the UVLM and Theodorsen methods.

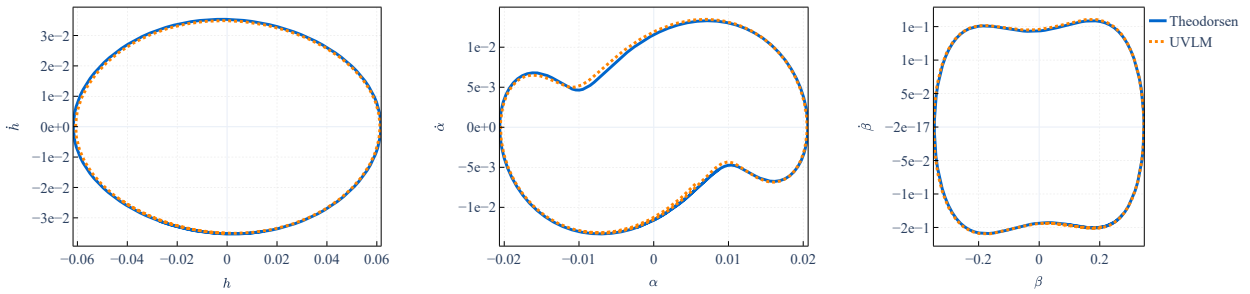


Figure 3.14 Poincaré sections for the 3-DOF wing with cubic stiffness at $U = 6\ \mathrm{m\,s^{-1}}$.

3.5.2 Freeplay Nonlinearity

Following Conner's experimental study [75], the freeplay nonlinearity is defined as:

$$\mathcal{M}_f(\beta) = M_0 + \begin{cases} \beta - a_f & \text{if } \beta < a_f \\ M_f(\beta - a_f) & \text{if } a_f \leq \beta \leq a_f + \delta \\ \beta - a_f + \delta(M_f - 1) & \text{if } \beta > a_f + \delta \end{cases} \tag{3.9}$$

where $a_f = -2.12°$ is the freeplay offset, $\delta = 4.24°$ is the freeplay range, $M_f = 0$ is the freeplay stiffness, and $M_0 = 0$ is the preload. All angles are converted to radians for simulation. The freeplay discontinuity requires careful treatment in the time integration scheme. For the adaptive Runge-Kutta method, the time step will naturally be reduced to avoid prediction errors in the stage values. In the UVLM case, the mesh resolution has to be high enough as it dictates the time step used.

Figure 3.15 shows a period-1 LCO appearing at approximately 4 $\mathrm{m\,s^{-1}}$, transitioning to an asymmetric period-2 motion near 10 $\mathrm{m\,s^{-1}}$. The system undergoes two additional transitions before entering a chaotic region at 21.5 $\mathrm{m\,s^{-1}}$, ultimately leading to divergent flutter.

The Poincaré sections, shown in Figure 3.16, again show good agreement between methods, with some deviations in the pitch degree of freedom attributed to accumulated numerical errors from UVLM discretization, time integration, and solver coupling. At higher flow speeds, the LCO frequency can increase significantly, necessitating finer UVLM meshes to accurately model the wake and reduce the time step to capture the transitions through the freeplay region.

Figure 3.17 compares the RMS of the steady-state LCO amplitudes from Conner's experiments with both Theodorsen and UVLM results. The values are computed as in the Conner study; the plunge is converted to centimetres, the wing and flap pitch are divided by the

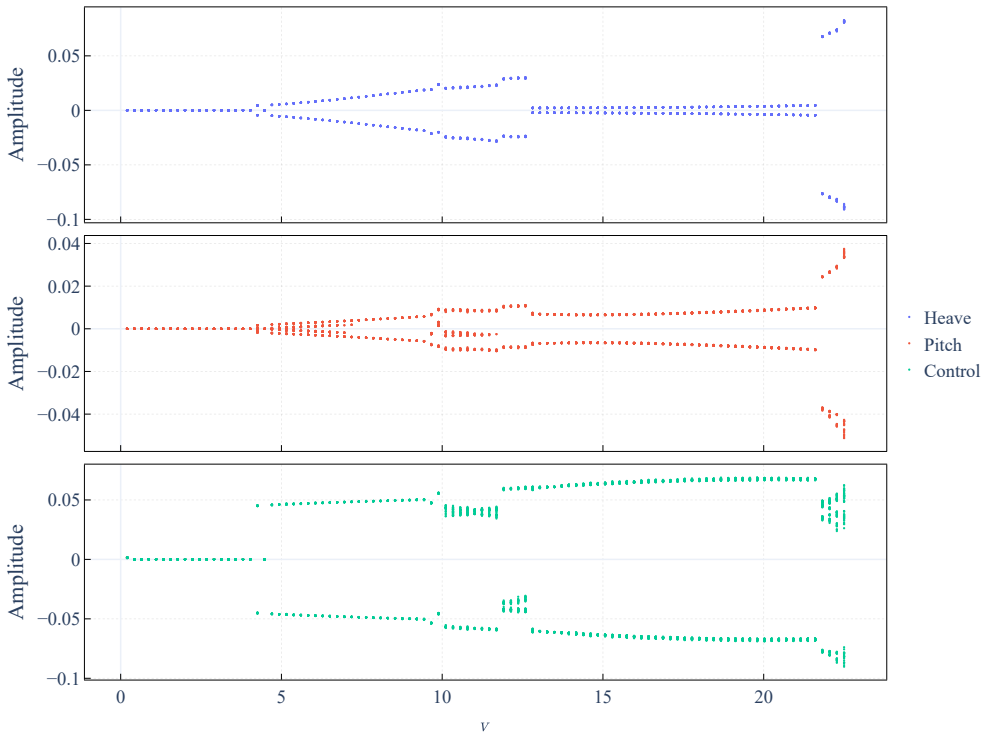


Figure 3.15 Bifurcation diagram for the 3-DOF wing with freeplay control surface stiffness.

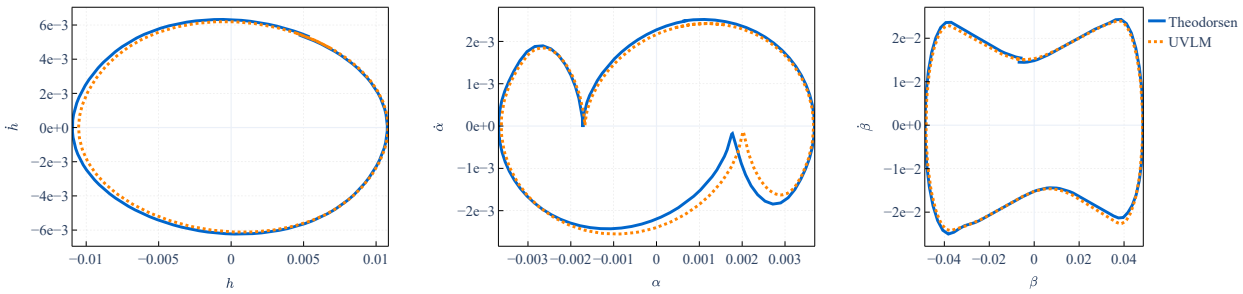


Figure 3.16 Poincaré sections for the 3-DOF wing with freeplay stiffness at $U = 7\ \mathrm{m\,s^{-1}}$.

freeplay range $\delta = 4.24°$ in degrees.

Conner's study included Theodorsen results for comparison, which aligns well with the Theodorsen implemented in this work. Across both computational methods, the conclusion is similar to Conner's findings: while there is some overestimation of amplitudes at lower flow speeds, the overall trend aligns well with the experimental data. Stable LCOs are predicted and amplitude jumps are captured with sufficient accuracy.

This verification establishes confidence in the UVLM implementation for aeroelastic applications as it is capable of predicting, both qualitatively and quantitatively, the nonlinear

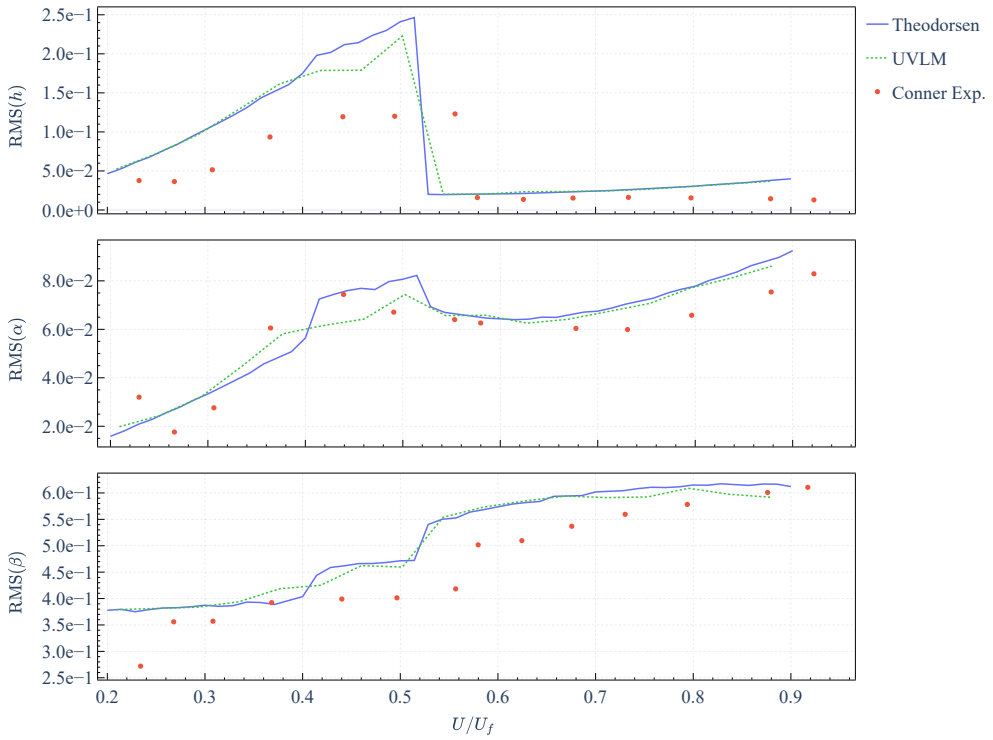


Figure 3.17 RMS Amplitudes from Conner's experiments compared to Theodorsen and UVLM results.

behavior of a 3-DOF wing with freeplay control surface stiffness.

3.6 HB-Continuation 3-DOF Wing

This section presents the culmination of this work: flutter LCO analysis using HB-formulated aeroelastic methods combined with numerical continuation. The analysis employs the same 3-DOF wing model introduced in Section 3.5, maintaining all parameters detailed in Appendix C. The structural nonlinearities remain unchanged, incorporating both cubic stiffness and freeplay characteristics as previously described in Section 3.5.

The key methodological advancement lies in replacing traditional time integration approaches with numerical continuation to trace LCO branches directly from the frequency domain formulation. This approach enables more efficient exploration of the solution space and provides comprehensive mapping of the system's nonlinear behavior.

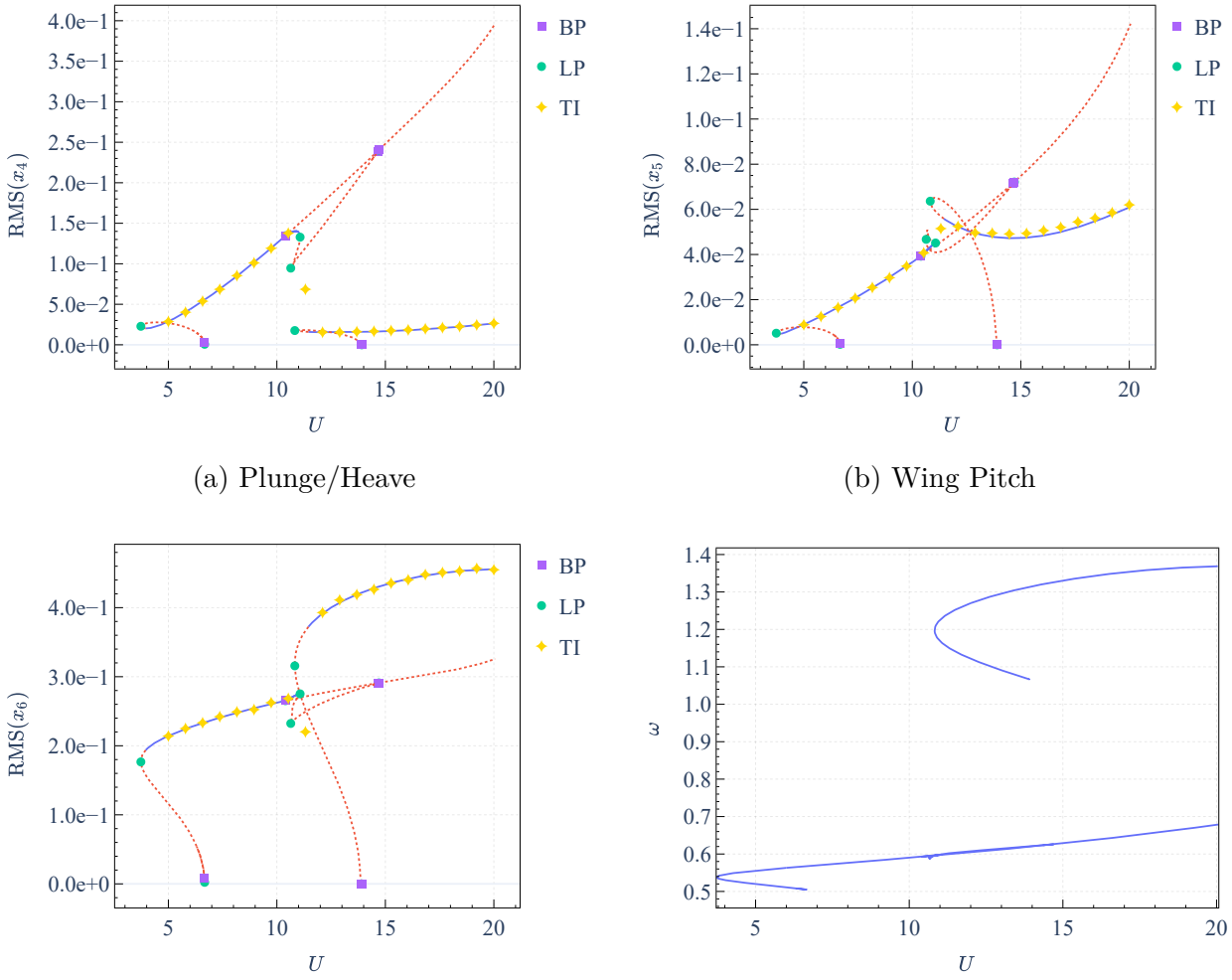
To establish baseline results, we first apply the continuation method to an HBM formulation of the Theodorsen model (detailed in Appendix D). These results serve as a benchmark against which both time integration solutions and more advanced aerodynamic models can be compared. Subsequently, we extend the analysis to incorporate the HB-VLM approach, where aerodynamic forces are computed using the methodology developed in Section 2.3.3 and validated in Section 3.2. To our knowledge, these HB-VLM continuation results represent a novel contribution to the literature.

Both the HB-Theodorsen and HB-VLM analyses employ 5 harmonics with 1536 sampling points per period for the AFT procedure. The HB-VLM implementation utilizes a refined mesh configuration consisting of 45 chordwise panels for the main wing surface and 15 chordwise panels for the flap, with both surfaces discretized using 2 spanwise panels. The wake extends over a length corresponding to three periods of periodic motion, ensuring adequate capture of unsteady aerodynamic effects.

Rather than employing the AFT procedure, the HB-VLM directly processes Fourier coefficients of the motion as input and generates corresponding Fourier coefficients of the aerodynamic forces as output. The iterative solution process within the HB-VLM occurs at each evaluation of the continuation residual. While full integration of the HB-VLM system within the continuation framework would create a coupled aerodynamic-continuation solver, such an approach would dramatically expand the system size and necessitate evaluation of numerous Jacobian terms. The segregated approach adopted here is hypothesized to offer superior memory efficiency and reduced computational cost.

3.6.1 Cubic Stiffness Nonlinearity

While continuation results for a pitching cubic stiffness nonlinearity for a 2-DOF wing have been presented very recently in the literature [76], to the authors' knowledge, no results for the numerical continuation of a 3-DOF wing with cubic stiffness nonlinearity have been published to date.



(a) Plunge/Heave

(b) Wing Pitch

(c) Flap Pitch

(d) LCO Frequency

Figure 3.18 HB-Theodorsen results for the 3-DOF wing with flap cubic stiffness nonlinearity

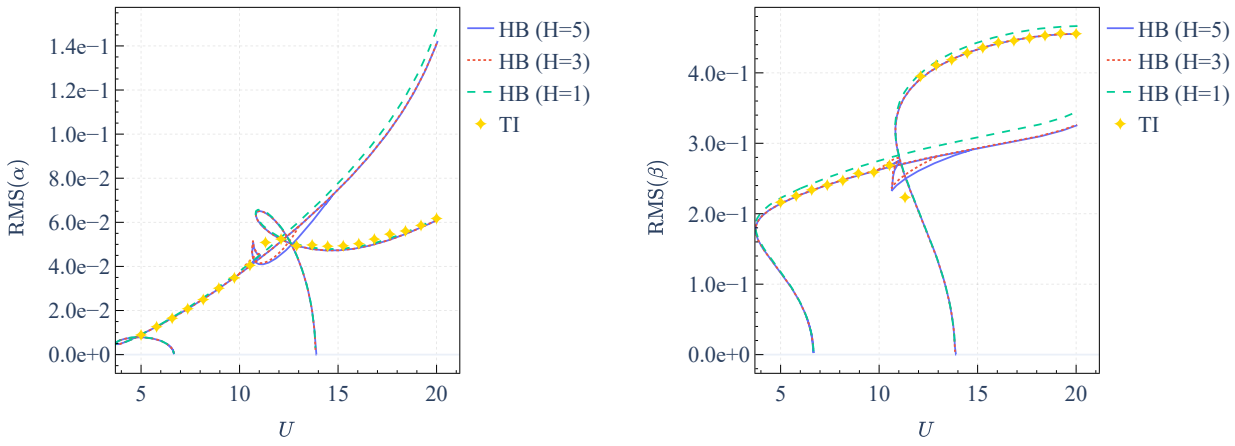
The results are presented with stable regions indicated by solid lines and unstable regions by dashed lines. Branch Points (BP), Limit Points (LP), and Time Integration (TI) results are marked on the figure for reference.

The figures reveals a rich nonlinear behavior with two distinct primary LCO branches emerg-

ing from the trivial solution branch. The first branch detaches at approximately $U = 6.7$, while the second emerges at $U = 13.9$. Notably, the upper branch exhibits additional complexity through a secondary branch that detaches and subsequently reattaches between $U = 10.5$ and $U = 14.5$. These primary branches are characterized by different base frequencies and possess distinct stability regions, demonstrating the system's multi-modal response characteristics.

The time integration results provide excellent validation, matching perfectly with the HB-Continuation predictions along the stable branches. Of particular interest is the isolated TI point near $U = 11.5$ that lies on neither primary branch. Cross-referencing with the bifurcation plot in Figure 3.13 shows that this point corresponds to the small chaotic region between the two stable zones.

To construct the complete diagram, multiple continuation runs were necessary due to the absence of branch switching capabilities in the current solver implementation. The figures were generated by superimposing three separate continuation analyses, each performed in both forward and backward directions. These analyses were started from time integration solutions obtained at $U = 6$, $U = 11$, and $U = 12$, ensuring comprehensive coverage of the solution space and proper capture of all bifurcation branches.



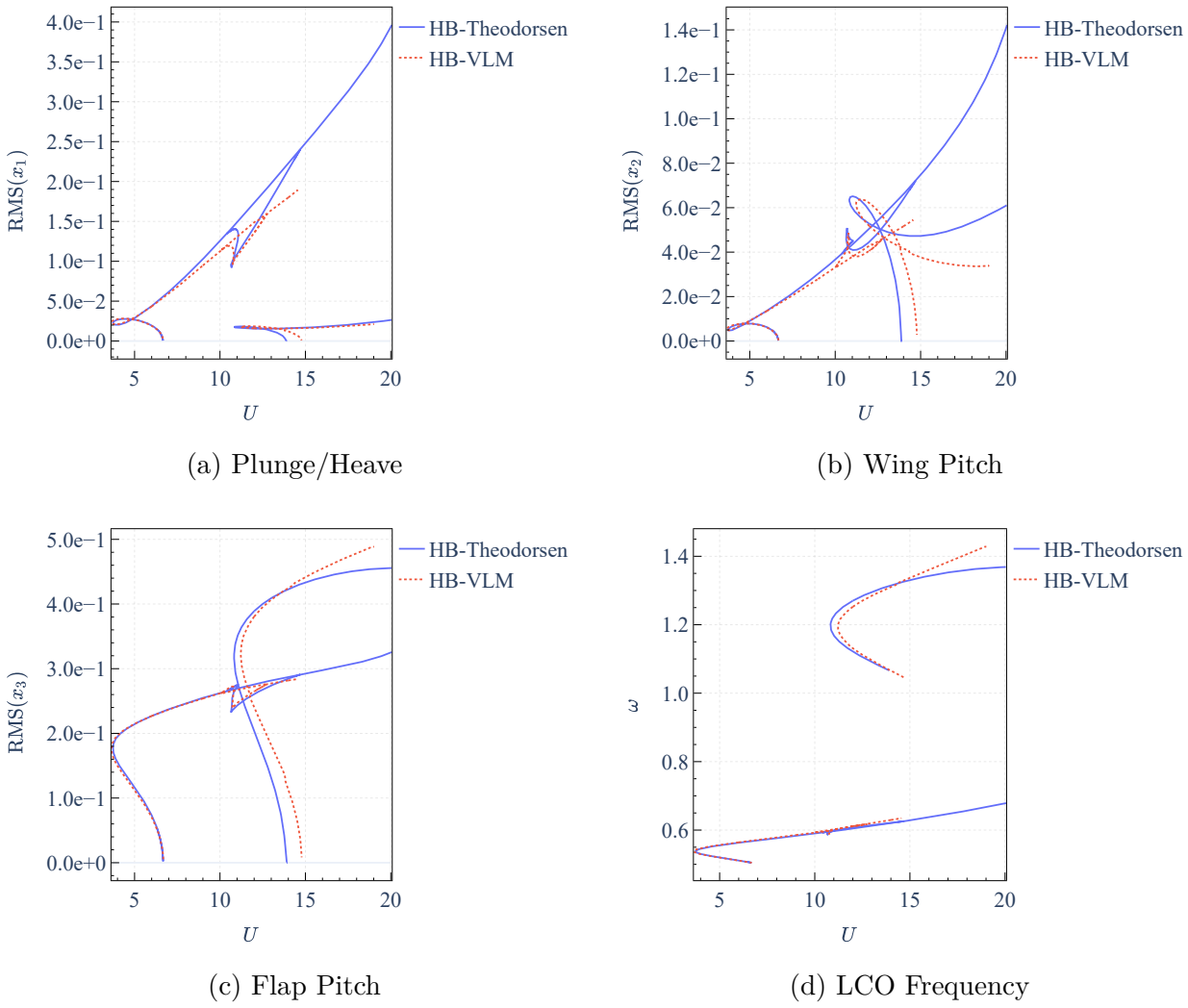
(a) Wing Pitch

(b) Flap Pitch

Figure 3.19 3-DOF wing HB Continuation results with different number of harmonics

Figure 3.19 presents a comparison of HB-Theodorsen results obtained using 1, 3 and 5 harmonics. The results demonstrate that the LCO harmonic content cannot be accurately captured with only one harmonic, leading to some discrepancies.

Figure 3.20 presents a comparison between the HB-Theodorsen and HB-VLM results. For



(a) Plunge/Heave

(b) Wing Pitch

(c) Flap Pitch

(d) LCO Frequency

Figure 3.20 HB-Theodorsen and HB-VLM Comparison for Cubic Stiffness Nonlinearity

clarity, stability and bifurcation features have been omitted from the visualization to facilitate direct comparison of the solution branches.

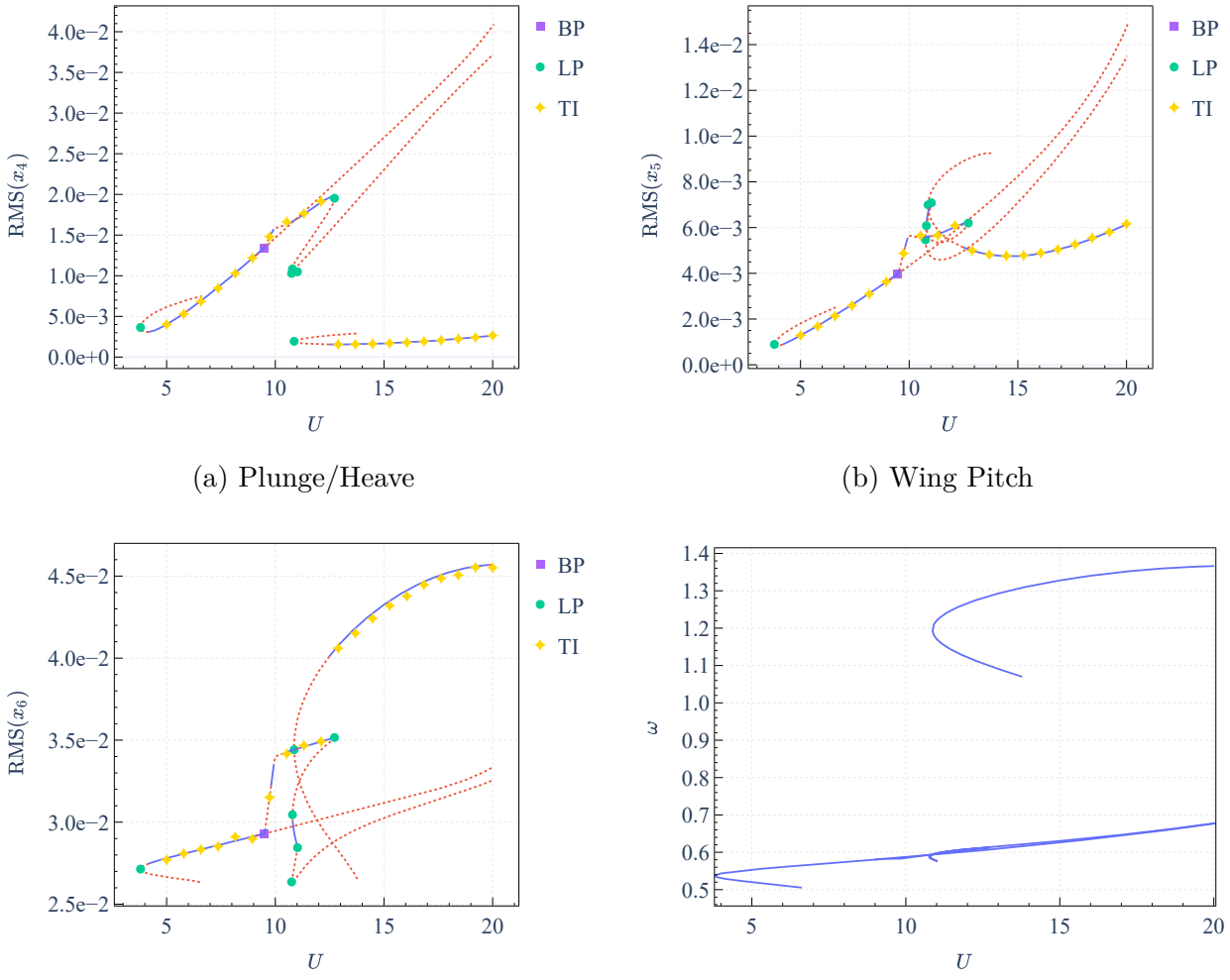
While both methods show good agreement in response amplitudes for low-frequency oscillations ($\omega < 0.6$), significant discrepancies emerge at higher frequencies. This divergence is expected, as the Theodorsen model's assumptions become less valid in these regimes. Nonetheless, the HBVLM still captures the overall branch topology, including the presence of multiple branches.

A second observation concerns the premature termination of solution branches, which results from convergence failure in the continuation residual computation. This behavior highlights the inherent sensitivity of the continuation procedure to aerodynamic forces accuracy. Small

errors accumulated during the iterative HB-VLM solution process propagate through the continuation algorithm, creating convergence difficulties.

The relationship between HB-VLM solver tolerance and continuation performance is particularly revealing. When the HB-VLM convergence tolerance is relaxed from $\epsilon = 10^{-12}$ to $\epsilon = 10^{-8}$, the continuation nonlinear solver requires approximately three times more iterations to achieve convergence. This suggests that using the alternative HB-VLM formulation, the monolithic approach proposed by [2], would be more robust as it delivers machine-precision accuracy, albeit at substantially increased computational and memory costs.

3.6.2 Freeplay Stiffness Nonlinearity



(a) Plunge/Heave

(b) Wing Pitch

(c) Flap Pitch

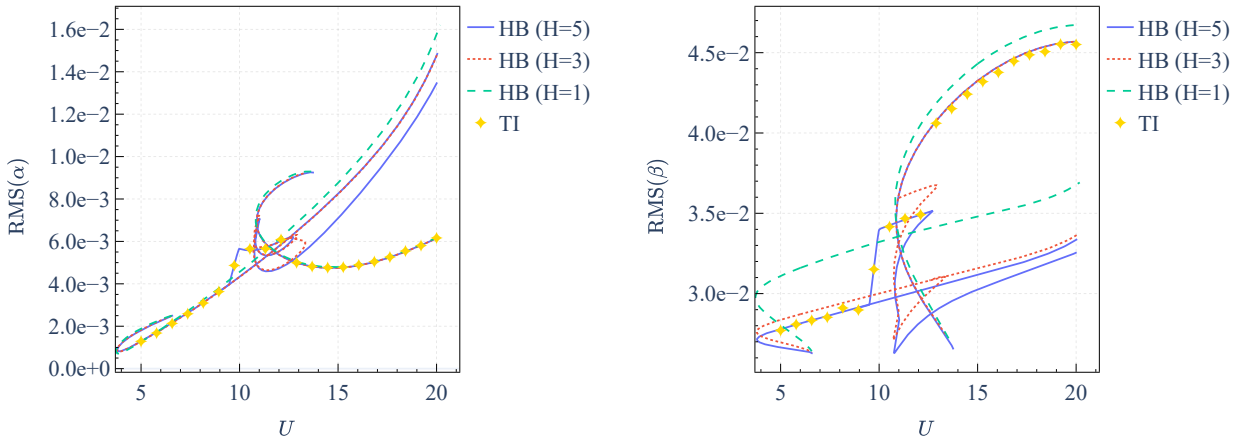
(d) LCO Frequency

Figure 3.21 Continuation results for the 3-DOF wing with flap freeplay stiffness nonlinearity

HB-Theodorsen continuation results for a 3-DOF wing with flap freeplay stiffness have been previously reported in the literature [77]. However, these earlier analyses were incomplete, lacking identification of a critical secondary LCO branch and omitting essential bifurcation information and stability characteristics.

The results, depicted in Figure 3.21, reveal two primary LCO branches that start abruptly by a jump from the trivial solution branch. The branches emerge at the same freestream velocities as in the cubic stiffness case as the onset of instability is governed by the same underlying linearized system [11]. The symmetry breaking pitchfork secondary bifurcation is observed again on the upper branch with the same detachment point but a different reattachment point.

Again, time integration results (TI) align perfectly with the HB-Continuation predictions along the stable branch sections, validating the continuation results. Similar to the cubic stiffness case, multiple continuation runs were necessary to construct the complete bifurcation diagram. Separate continuation analyses were performed starting from solutions at $U = 6$, $U = 11$, and $U = 15$.

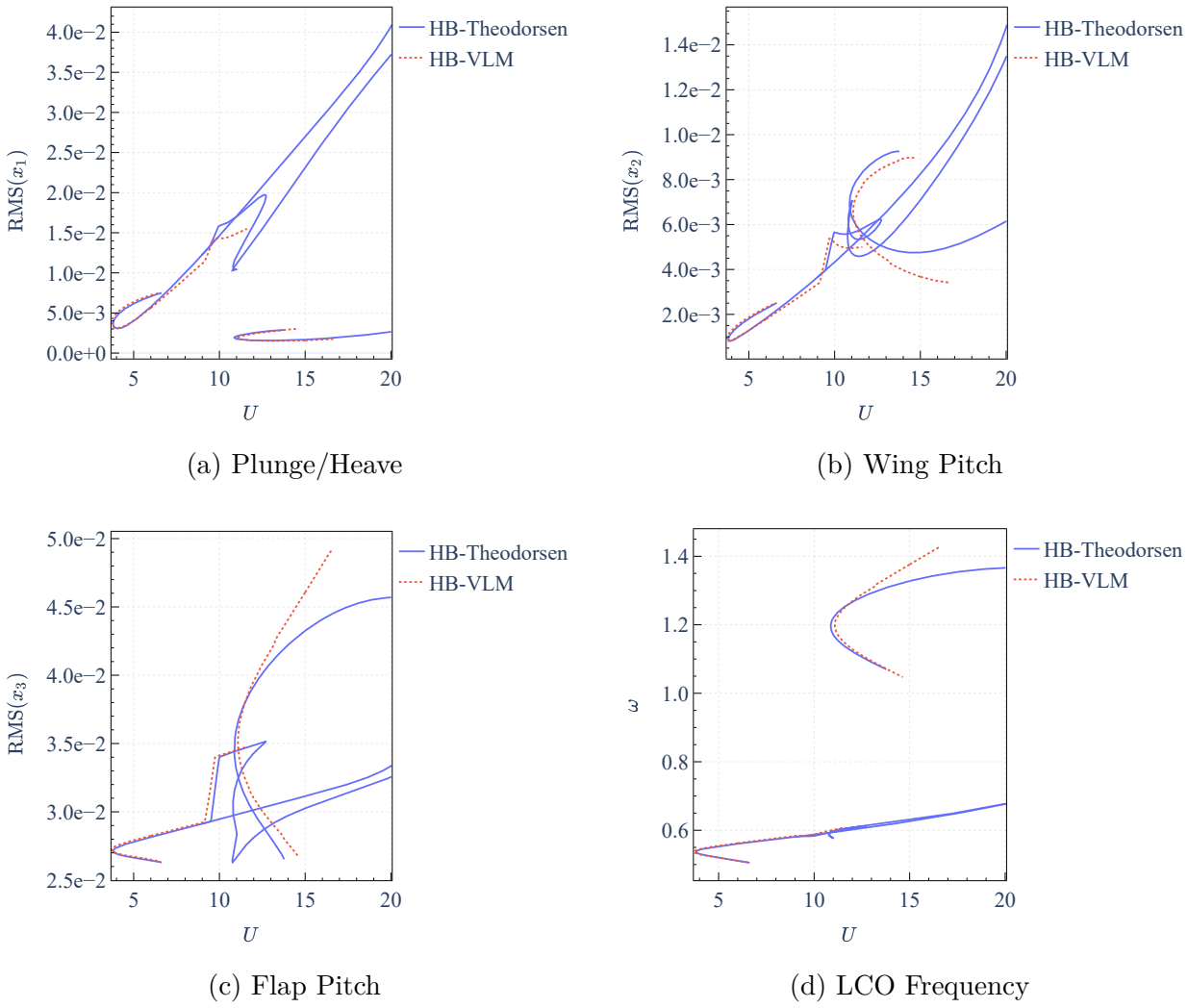


(a) Wing Pitch

(b) Flap Pitch

Figure 3.22 3-DOF wing HB Continuation results with different number of harmonics

Figure 3.22 presents a comparison of HB-Theodorsen results obtained using 1, 3 and 5 harmonics. The response contains significant higher-order harmonic content due to the discontinuous nature of the freeplay nonlinearity, making it essential to include multiple harmonics for accurate representation. The one-harmonic solution deviates substantially from the time integration ground-truth, especially for the degree of freedom directly affected by the nonlinearity.



(a) Plunge/Heave

(b) Wing Pitch

(c) Flap Pitch

(d) LCO Frequency

Figure 3.23 HB-Theodorsen and HB-VLM Comparison for the 3-DOF Wing with Flap Freeplay Stiffness Nonlinearity

Figure 3.23 shows similar trends to those observed in the cubic stiffness case when comparing HB-Theodorsen and HB-VLM results. Correct agreement observed at low frequencies with strong discrepancies at higher frequencies. Due to the lack of branch switching capabilities in the continuation solver, the lower branch could not be fully captured with the HB-VLM approach. Early termination of both primary branches is, once again, caused by convergence issues in the continuation residual.

3.7 Performance Benchmarking

Performance benchmarks were performed to evaluate the computational efficiency of both the UVLM and HB-VLM implementations on CPU and GPU architectures, both in single precision. Testing was conducted on a desktop workstation equipped with an Intel i7-13700K processor with 8 performance cores and 8 effiency cores, 64 GB of DDR5-5600 RAM and an NVIDIA RTX 4070 GPU with 16 GB of GDDR6X memory. Benchmarks were measured on Windows 11 using the MSVC 14.44 compiler, ISPC 1.27 compiler and the CUDA 12.9 toolkit. Measured time is end-end wall time and is calculated as the average of 3 runs. L2 cache was cleared between runs to ensure consistent results. Both the CPU and GPU were allowed to boost to their maximum clock frequencies during the benchmarks.

3.7.1 UVLM

The UVLM benchmarks were performed on the same pitching airfoil case used for verification in section 3.1. Simulations were run for a total of 3 oscillation cycles with a mesh of 100 spanwise panels and chordwise panels varying from 5 to 160.

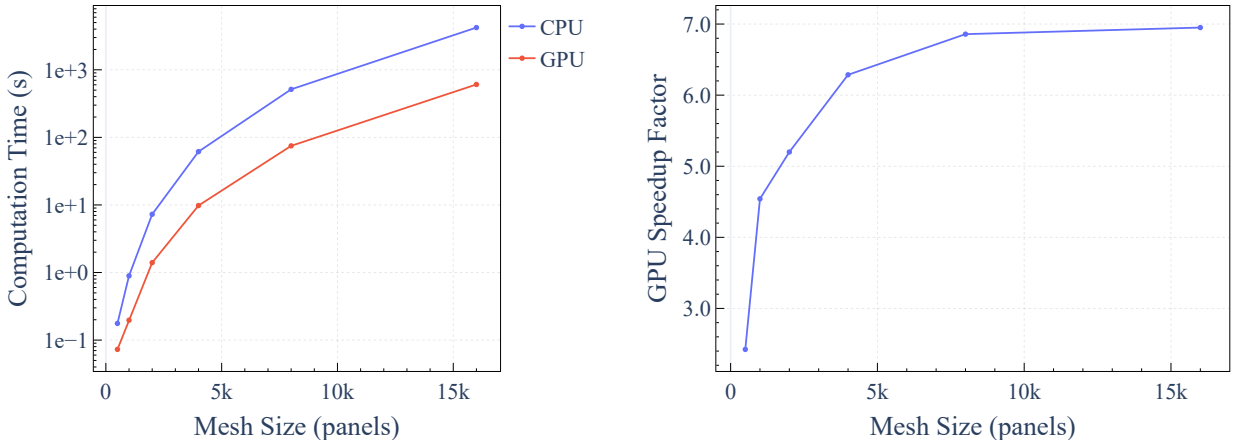


Figure 3.24 Performance comparison for UVLM simulations

The increasing number of chordwise panels directly increases the number of time steps required: it takes 74 time steps for 5 chordwise panels and 2996 time steps for 160 chordwise panels. Coupled with the fact that UVLM time complexity for rigid wings is $O(n^2)$, the overall time complexity of this benchmark is $O(n^3)$. Figure 3.24 shows strong acceleration when using the GPU implementation with speedups reaching close to 7×. As the specific GPU used here has around 7× more memory bandwidth than the CPU, this indicates that the implementation is memory-bound and effectively utilizes all available memory bandwidth.

3.7.2 HBVLM

The HBVLM benchmarks were done on the same biharmonic case used for verification in section 3.2. Simulations used 5 harmonics, a pregenerated wake length of 3 cycles and the same meshes as in the UVLM benchmarks.

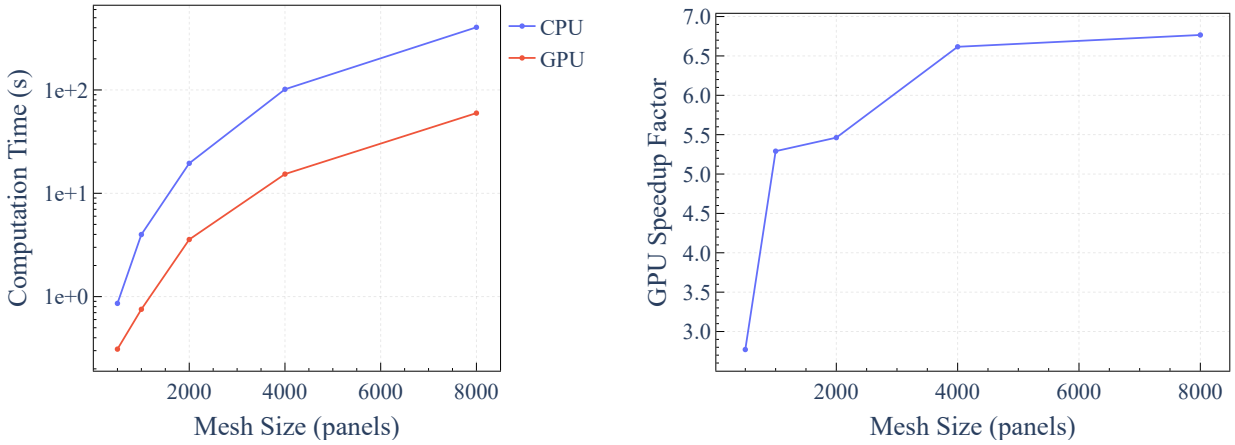


Figure 3.25 Performance comparison for HB-VLM simulations

Again, Figure 3.25 shows very good acceleration when using the GPU implementation with speedups reaching 7×. As with the UVLM, once the main influence matrix is pre-factorized, the iterative loop of the HBVLM is mostly memory-bound, leading to similar speedup values as the UVLM benchmarks.

It is worth noting that the CPU implementation is quite efficient and uses AVX2 vectorization in all of the computational sections thanks to the ISPC compiler. Changing the target to SSE2 leads to a 1.6× slowdown and fully scalar code leads to a 6.7× slowdown for large mesh sizes. This highlights the importance of vectorization for achieving high throughput on modern CPU architectures.

CHAPTER 4 CONCLUSION

4.1 Summary of Works

This thesis addressed the pressing need for robust and efficient methods to analyze nonlinear aeroelastic phenomena on aircraft lifting surfaces. Traditional linear analyses often fail to capture these complex dynamics, and rely on time domain reformulations which are computationally expensive and impractical for early design phases. To overcome these challenges, this research focused on developing a fully frequency-domain approach capable of modelling nonlinear aeroelastic responses.

The core contribution of this work is the development and implementation of a GPU-accelerated Harmonic Balance Vortex Lattice Method (HBVLM) integrated within a sophisticated numerical continuation framework for nonlinear aeroelastic analysis.

The key accomplishments include:

- A high performance reimplementation of the iterative HBVLM using a fixed point formulation that achieves substantial speedups using GPUs and reduces the number of required iterations by an order of magnitude using Anderson Acceleration. The HBVLM was verified against the UVLM for a wing undergoing forced oscillations and demonstrated excellent agreement in predicting aerodynamic loads.

- A comprehensive numerical continuation framework, incorporating predictor-corrector algorithms with adaptive step-size control, arc-length parameterization, frequency domain stability analysis via the Floquet-Hill method and bifurcation detection. Harmonic Balance residuals and the Jacobian were implemented using a hybrid-analytical formulation with a precise finite difference scheme, along with frequency filtering techniques to ensure numerical stability and accuracy. The integrated framework was thoroughly verified against established academic problems that include both forced and autonomous systems.

- Successful application of the HBVLM with numerical continuation to a three-degree-of-freedom (3-DOF) wing section with significant structural nonlinearities: cubic stiffness and freeplay at the flap joint. The results showed detailed mapping of complex LCO bifurcations, with good agreement with Theodorsen results which revealed multiple stable and unstable solution branches, amplitude jumps, and transitions to chaotic behavior. This provided a comprehensive understanding of the system's nonlinear

response characteristics and stands as a novel and more comprehensive contribution to the literature, identifying secondary LCO branches and detailed bifurcation information previously omitted.

This work marks a significant step towards providing robust and efficient tools for the design and certification of next-generation aircraft by enabling the possiblity of extending analysis to more complex configuration using higher fidelity extensions in the future.

4.2 Limitations

The current implementation and underlying models present several limitations that constrain its applicability and practicality. However, most of these limitations can be addressed through future research and not fundamental to the methods themselves.

4.2.1 Aerodynamics

The VLM itself is limited to incompressible and inviscid flows, as such it cannot accurately model high amplitude oscillations where the viscous effects become significant. Additionally, the current wake formulation in frequency-domain is prescribed and does not account for wake rollup, which restricts it's applicability to scenarios where wake rollup does not significantly affect the aerodynamics, like in the case of rotor-wake interactions. The inexact nature of the iterative HBVLM can lead to convergence issues within the continuation solver, especially at higher reduced frequencies.

4.2.2 Structure

The structural model demonstrated in this work is extremely simple and limited to fixed mass, damping and stiffness matrices. It assumes small deformations and linear structural dynamics with added nonlinearities. This restricts the analysis to specific simplified aeroelastic systems and currently cannot take into account flexible structures.

4.2.3 Continuation

The current numerical continuation solver lacks the capability of branch point localization and branch switching. This prevents the full mapping of complex LCO profiles with multiple branches without manual intervention by restarting from a time domain stable solution. The stability analysis using Hill's method can be quite noisy and inaccurate for a small num-

ber of harmonics and is generally computationally expensive as it requires a full eigenvalue decomposition of the truncated Hill matrix.

4.2.4 Software

As numerical continuation is inherently extremely sensitive to numerical inaccuracies, this has forced the use of double-precision arithmetic throughout the implementation. This significantly reduces the potential speedups that can be obtained using consumer-grade GPUs as their double-precision ALUs are often emulated at a ratio of 1/32 or worse compared to single-precision. Since the current implementation is a hybrid Python and C++ codebase, there is non-negligible overhead and it prevents the use of automatic differentiation across the numerical models.

4.3 Future Research

4.3.1 Aerodynamics

In order to mitigate the limitations of the VLM and expand its applicability into compressible flow regimes, sectional corrections with steady state 2.5D RANS databases [25] can be performed to correct the relative flow angle of attack. While this approach offers computational efficiency, it's worth noting that it relies on quasi-steady assumptions and may not fully capture unsteady transonic phenomena.

Another alternative that would tackle the problem with a richer theoretical foundation would consist in implementing a harmonic balance formulation of the compressible source and doublet panel method with retarded time [78]. Along with modelling thickness and fuselage effects, the method would be able to model weakly compressible flow with reasonable accuracy. Nevertheless, potential flow methods will ultimately remain unsuitable for high-amplitude oscillations where viscous effects dominate, suggesting that the use of external viscous databases or coupling with CFD solvers may be unavoidable.

4.3.2 Structure

The structural modeling framework presents many opportunities for improvements. A natural progression would involve implementing a corotational beam solver [79] formulated within the HBM framework to model structures undergoing large deformations [17].

Furthermore, this would enable the use of nonlinear material models and advanced beam formulations, such as Timoshenko-Ehrenfest beams with hardening or softening nonlinearities

[80]. When dealing with thickness models in the aerodynamic solver, efficient force transfer between the structural and aerodynamic domains can be accomplished using methods like thin plate splines [54].

4.3.3 Continuation

Ideally, the entire continuation process should be fully automated and started from the trivial branch with a reduced velocity close to zero. The solver would then be able to follow the trivial branch until it encounters a branch point where it will automatically switch to the bifurcating branch.

Branch switching techniques [81] are still an active research area and most popular continuation software packages don't support it yet. It requires solving an augmented system to precisely locate branch points followed by controlled perturbations to trace bifurcating branches. This would enable tracing complete LCO profiles without manual intervention or parameter tuning, significantly improving the method's practical utility for design applications and removing the need for time domain based initial guesses.

4.3.4 Software

Development of future numerical software have a few promising directions. While the accelerator programming landscape is still quite fragmented and vendor-specific, emerging languages and libraries are beginning to offer compelling abstractions that could simplify development and maintenance of high-performance scientific codes.

One such option is the Chapel [82] language which contains parallelism constructs built-in the language itself. Along with simplifying code maintenance, it is the most accessible approach to writing distributed heterogeneous applications without resorting to low level libraries like MPI.

Another approach which was heavily considered at the start of this thesis, was to use autodifferentiable tensor libraries like JAX [83] commonly used in the machine learning community. These libraries are at the frontier of automatic scheduling, kernel fusion and JIT compilation for accelerators. Autodifferentiability is still overlooked in the scientific community despite the tremendous potential it offers for integration in adjoint based optimization, sensitivity analysis and jacobian free computations. However, the ecosystem is still quite experimental, sparse formats are missing and double precision support is often lacking. Nevertheless, it is likely that in the next 1-3 years these libraries will mature and become a viable alternative for scientific computing.

REFERENCES

[1] M. Tasche and H. Zeuner, "Worst and average case roundoff error analysis for FFT," *BIT Numerical Mathematics*, vol. 41, pp. 563–581, 2001.

[2] M. Parenteau, "A Non-Linear Frequency Domain Potential Flow Model for Compressible, Transonic and Viscous Aeroelastic Analyses," phd, Polytechnique Montréal, May 2021. [Online]. Available: https://publications.polymtl.ca/6629/

[3] Y. Colaïtis and A. Batailly, "The harmonic balance method with arc-length continuation in blade-tip/casing contact problems," *Journal of Sound and Vibration*, p. 116070, Mar. 2021, publisher: Elsevier. [Online]. Available: https://hal.science/hal-03163560

[4] L. R. Krol, "Signal aliasing demonstration," 2020. [Online]. Available: https://commons.wikimedia.org/wiki/File:Signal_aliasing_demonstration.svg

[5] Y. Colaïtis, "Stratégie numérique pour l'analyse qualitative des interactions aube/carter," phd, Polytechnique Montréal, Jul. 2021. [Online]. Available: https://publications.polymtl.ca/9096/

[6] T. T. Darabseh, "Nonlinear State Dependent Riccati Equation Controller for 3-DOF Airfoil with Cubic Structural Nonlinearity," *International Review of Aerospace Engineering (IREASE)*, vol. 15, no. 2, pp. 97–111, 2022, publisher: Praise Worthy Prize. [Online]. Available: https://www.praiseworthyprize.org/jsm/index.php?journal=irease&page=article&op=view&path[]=26230

[7] E. Dowell, *A Modern Course in Aeroelasticity: Fifth Revised and Enlarged Edition*, 5th ed., ser. Solid Mechanics and Its Applications. Springer International Publishing, 2015. [Online]. Available: https://www.springer.com/gp/book/9783319094526

[8] M. J. Patil and D. H. Hodges, "On the importance of aerodynamic and structural geometrical nonlinearities in aeroelastic behavior of high-aspect-ratio wings," *Journal of Fluids and Structures*, vol. 19, no. 7, pp. 905–915, 2004.

[9] M. Padmanabhan, E. Dowell, and C. Pasiliao, "Computational study of aeroelastic limit cycles due to localized structural nonlinearities," *Journal of Aircraft*, vol. 55, no. 4, pp. 1531–1541, 2018.

[10] A. Gilioli, A. Manes, U. Ringertz, and M. Giglio, "Investigation about the structural nonlinearities of an aircraft pylon," *Journal of Aircraft*, vol. 56, no. 1, pp. 273–283, 2019.

[11] G. Dimitriadis, *Introduction to Nonlinear Aeroelasticity*. Wiley, 2017, google-Books-ID: z8NlswEACAAJ.

[12] S. T. Trickey, L. Virgin, and E. Dowell, "The stability of limit cycle oscillations in a nonlinear aeroelastic system," *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 458, pp. 2203–2226, 2002.

[13] C. R. Mair, "Stability analysis of whirl flutter in rotor-nacelle systems with structural nonlinearities," Ph.D. dissertation, University of Bristol, 2021.

[14] J. Murua, "Flexible Aircraft Dynamics with a Geometrically-Nonlinear Description of the Unsteady Aerodynamics," Ph.D. dissertation, Jun. 2012.

[15] L. Liu, E. H. Dowell, and J. P. Thomas, "A high dimensional harmonic balance approach for an aeroelastic airfoil with cubic restoring forces," *Journal of Fluids and Structures*, vol. 23, no. 3, pp. 351–363, Apr. 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0889974606001095

[16] M. Krack, L. P.-v. Scheidt, and J. Wallaschek, "A High-Order Harmonic Balance Method for Systems With Distinct States," *Journal of Sound and Vibration*, vol. 332, no. 21, pp. 5476–5488, Oct. 2013, arXiv:2101.01806 [math]. [Online]. Available: http://arxiv.org/abs/2101.01806

[17] M. Krack and J. Gross, *Harmonic Balance for Nonlinear Vibration Problems*, ser. Mathematical Engineering. Cham: Springer International Publishing, 2019. [Online]. Available: http://link.springer.com/10.1007/978-3-030-14023-6

[18] S. Stoykov and S. Margenov, "Scalability of Shooting Method for Nonlinear Dynamical Systems," in *Large-Scale Scientific Computing*, I. Lirkov, S. D. Margenov, and J. Waśniewski, Eds. Cham: Springer International Publishing, 2015, pp. 401–408.

[19] Z. Yan, H. Dai, Q. Wang, and S. Atluri, "Harmonic Balance Methods: A Review and Recent Developments," *Computer Modeling in Engineering & Sciences*, vol. 137, no. 2, pp. 1419–1459, 2023, publisher: Tech Science Press. [Online]. Available: https://www.techscience.com/CMES/v137n2/53365

[20] J. Thomas, C. Custer, E. Dowell, and K. Hall, "Unsteady Flow Computation Using a Harmonic Balance Approach Implemented About the OVERFLOW 2 Flow Solver," in *19th AIAA Computational Fluid Dynamics*, ser. Fluid Dynamics and Co-located Conferences. American Institute of Aeronautics and Astronautics, Jun. 2009. [Online]. Available: https://arc.aiaa.org/doi/10.2514/6.2009-4270

[21] J. Thomas and E. H. Dowell, "Aeroelastic Flap Free-Play Analysis Using a Harmonic Balance Approach," in *AIAA SCITECH 2025 Forum*, ser. AIAA SciTech Forum. American Institute of Aeronautics and Astronautics, Jan. 2025. [Online]. Available: https://arc.aiaa.org/doi/10.2514/6.2025-1200

[22] G. Dimitriadis, "Continuation of Higher-Order Harmonic Balance Solutions for Nonlinear Aeroelastic Systems," *Journal of Aircraft*, vol. 45, no. 2, pp. 523–537, Mar. 2008. [Online]. Available: https://arc.aiaa.org/doi/10.2514/1.30472

[23] D. T. K. Hoang, S. V. Pham, K. N. Tran, C. D. Nguyen, and K. P. Nguyen, "Aeroelastic Analysis on Wing Structure Using Immersed Boundary Method," in *Proceedings of the International Conference on Advances in Computational Mechanics 2017*, H. Nguyen-Xuan, P. Phung-Van, and T. Rabczuk, Eds. Singapore: Springer, 2018, pp. 783–792.

[24] J. Katz and A. Plotkin, *Low-Speed Aerodynamics*, 2nd ed., ser. Cambridge Aerospace Series. Cambridge: Cambridge University Press, 2001. [Online]. Available: https://www.cambridge.org/core/books/lowspeed-aerodynamics/077FAF851C4582F1B7593809752C44AE

[25] V. Liguori, E. Laurendeau, F. Moens, and J. Peter, "Appropriate Lift Slope Model for Nonlinear Vortex Lattice Method," *Journal of Aircraft*, vol. 0, no. 0, pp. 1–9, 2025, publisher: American Institute of Aeronautics and Astronautics _eprint: https://doi.org/10.2514/1.C038299. [Online]. Available: https://doi.org/10.2514/1.C038299

[26] S. F. Pullin, M. Groom, B. Zhou, and M. Azarpeyvand, "A GPU-Accelerated Mid-Fidelity Aerodynamic and Aeroacoustic Simulation Framework: 30th AIAA/CEAS Aeroacoustics Conference," Jun. 2024.

[27] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, "User guide for MINPACK-1," Argonne National Laboratory, Argonne, Ill., Tech. Rep. ANL-80-74, 1980.

[28] M. Pharr and W. R. Mark, "ispc: A spmd compiler for high-performance cpu programming," in *2012 Innovative Parallel Computing (InPar)*, 2012, pp. 1–13.

[29] T.-W. Huang, D.-L. Lin, C.-X. Lin, and Y. Lin, "Taskflow: A Lightweight Parallel and Heterogeneous Task Graph Computing System," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 6, pp. 1303–1320, Jun. 2022, arXiv:2004.10908 [cs]. [Online]. Available: http://arxiv.org/abs/2004.10908

[30] NVIDIA, P. Vingelmann, and F. H. Fitzek, "Cuda, release: 12.6," 2024. [Online]. Available: https://developer.nvidia.com/cuda-toolkit

[31] W. Jakob, J. Rhinelander, and D. Moldovan, "pybind11 — seamless operability between c++11 and python," 2016, https://github.com/pybind/pybind11.

[32] G. Chiocchia, D. Tordella, and S. Prössdorf, "The Lifting Line Equation for a Curved Wing in Oscillatory Motion," *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 77, no. 4, pp. 295–315, Jan. 1997. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/zamm.19970770419

[33] C. W. P. Ford and H. Babinsky, "Lift and the leading-edge vortex," *Journal of Fluid Mechanics*, vol. 720, pp. 280–313, Apr. 2013. [Online]. Available: https://www.cambridge.org/core/journals/journal-of-fluid-mechanics/article/abs/lift-and-the-leadingedge-vortex/E734299FFBFF5048AD33C579C195BB44

[34] M. Parenteau, "Aerodynamic Optimization of Aircraft Wings Using a Coupled VLM-2.5D RANS Approach," Master's thesis, École Polytechnique de Montréal, May 2017. [Online]. Available: https://publications.polymtl.ca/2555/

[35] R. Featherstone, *Robot Dynamics Algorithms*. Kluwer, Jan. 1987, google-Books-ID: c6yz7f_jpqsC.

[36] O. Rodrigues, "Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace, et de la variation des coordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire," *Journal de Mathématiques Pures et Appliquées*, vol. 5, pp. 380–440, 1840. [Online]. Available: https://www.numdam.org/item/?id=JMPA_1840_1_5__380_0

[37] L. B. Rall, *Automatic Differentiation: Techniques and Applications*. Springer-Verlag, 1981, google-Books-ID: HFRqQgAACAAJ.

[38] R. J. S. Simpson, R. Palacios, and J. Murua, "Induced-Drag Calculations in the Unsteady Vortex Lattice Method," *AIAA Journal*, vol. 51, no. 7, pp. 1775–1779, Jul. 2013, publisher: American Institute of Aeronautics and Astronautics. [Online]. Available: https://arc.aiaa.org/doi/10.2514/1.J052136

[39] C. Tsitouras, "Runge-Kutta pairs of order 5(4) satisfying only the first column simplifying assumption," *Computers & Mathematics with Applications*, vol. 62, no. 2,

pp. 770–775, Jul. 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0898122111004706

[40] C. A. Kennedy and M. H. Carpenter, "Diagonally Implicit Runge-Kutta Methods for Ordinary Differential Equations. A Review," 2016.

[41] S. Horne, "MSC/NASTRAN," in *Finite Element Systems: A Handbook*, C. A. Brebbia, Ed. Berlin, Heidelberg: Springer, 1982, pp. 287–294. [Online]. Available: https://doi.org/10.1007/978-3-662-07229-5_19

[42] C. Farhat and M. Lesoinne, "Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 182, no. 3, pp. 499–515, Feb. 2000. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782599002066

[43] H. F. Walker and P. Ni, "Anderson Acceleration for Fixed-Point Iterations," *SIAM Journal on Numerical Analysis*, vol. 49, no. 4, pp. 1715–1735, Jan. 2011. [Online]. Available: http://epubs.siam.org/doi/10.1137/10078356X

[44] N. M. Newmark, "A method of computation for structural dynamics," *Journal of the Engineering Mechanics Division*, vol. 85, no. EM3, pp. 67–94, 1959.

[45] T. J. R. Hughes, "Analysis of transient algorithms with particular reference to stability behavior." 1983. [Online]. Available: https://api.semanticscholar.org/CorpusID:115315298

[46] H. P. Gavin, "Numerical integration in structural dynamics," CEE 541. Structural Dynamics, Department of Civil & Environmental Engineering, Duke University, 2020. [Online]. Available: https://people.duke.edu/~hpgavin/StructuralDynamics/NumericalIntegration.pdf

[47] M. Urabe, "Galerkin's procedure for nonlinear periodic systems," *Archive for Rational Mechanics and Analysis*, vol. 20, no. 2, pp. 120–152, Jan. 1965. [Online]. Available: https://doi.org/10.1007/BF00284614

[48] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 2002.

[49] T. M. Cameron and J. H. Griffin, "An alternating frequency/time domain method for calculating the steady-state response of nonlinear dynamic systems," *Journal of Applied Mechanics*, vol. 56, no. 1, pp. 149–154, 1989.

[50] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the Institute of Radio Engineers*, vol. 37, no. 1, pp. 10–21, 1949.

[51] F. H. Ling and X. X. Wu, "Fast galerkin method and its application to determine periodic solutions of non-linear oscillators," *International Journal of Non-Linear Mechanics*, vol. 22, no. 2, pp. 89–98, 1987.

[52] A. J. Jerri, *The Gibbs Phenomenon in Fourier Analysis, Splines and Wavelet Approximations*. Boston, MA: Springer US, 1998. [Online]. Available: http://link.springer.com/10.1007/978-1-4757-2847-7

[53] R. Djeddi and K. Ekici, "Resolution of Gibbs phenomenon using a modified pseudo-spectral operator in harmonic balance CFD solvers," *International Journal of Computational Fluid Dynamics*, vol. 30, no. 7-10, pp. 495–515, 2016.

[54] M. Parenteau and E. Laurendeau, "A general modal frequency-domain vortex lattice method for aeroelastic analyses," *Journal of Fluids and Structures*, vol. 99, p. 103146, Nov. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0889974620306150

[55] M. E. Henderson, "MULTIPLE PARAMETER CONTINUATION: COMPUTING IMPLICITLY DEFINED k-MANIFOLDS," *International Journal of Bifurcation and Chaos*, vol. 12, no. 03, pp. 451–476, Mar. 2002, publisher: World Scientific Publishing Co. [Online]. Available: https://www.worldscientific.com/doi/abs/10.1142/S0218127402004498

[56] R. Seydel, "Principles of Continuation," in *Practical Bifurcation and Stability Analysis*, R. Seydel, Ed. New York, NY: Springer, 2010, pp. 169–198. [Online]. Available: https://doi.org/10.1007/978-1-4419-1740-9_4

[57] C. G. Broyden, "A class of methods for solving nonlinear simultaneous equations," *Mathematics of Computation*, vol. 19, no. 92, pp. 577–593, 1965.

[58] J. Sherman and W. J. Morrison, "Adjustment of an inverse matrix corresponding to changes in the elements of a given column or a given row of the original matrix (abstract)," *Annals of Mathematical Statistics*, vol. 20, p. 621, 1949.

[59] R. Seydel, "A Continuation Algorithm with Step Control," in *Numerical Methods for Bifurcation Problems: Proceedings of the Conference at the University of Dortmund, August 22–26, 1983*, T. Küpper, H. D. Mittelmann, and

H. Weber, Eds. Basel: Birkhäuser, 1984, pp. 480–494. [Online]. Available: https://doi.org/10.1007/978-3-0348-6256-1_33

[60] M. Krack and J. Groß, "Nlvib," 2020. [Online]. Available: https://www.ila.uni-stuttgart.de/nlvib/

[61] A. H. Nayfeh, *Applied Nonlinear Dynamics: Analytical, Computational and Experimental Methods*. Wiley, Feb. 1995, google-Books-ID: MposnwEACAAJ.

[62] G. Moore, "Floquet theory as a computational tool," *SIAM J. Numer. Anal.*, vol. 42, no. 6, pp. 2522–2568, 2005.

[63] F. Brauer and J. A. Nohel, *The Qualitative Theory of Ordinary Differential Equations: An Introduction*. Courier Corporation, Jan. 1989, google-Books-ID: KqKN2vOBpZ4C.

[64] C. Delbé, "Application d'une stratégie numérique de suivi de bifurcations à l'analyse d'interactions aube/carter dans les moteurs d'avions," Master's thesis, Polytechnique Montréal, Jul. 2023. [Online]. Available: https://publications.polymtl.ca/54150/

[65] J. Zhou, T. Hagiwara, and M. Araki, "Spectral characteristics and eigenvalues computation of the harmonic state operators in continuous-time periodic systems," *Systems & Control Letters*, vol. 53, no. 2, pp. 141–155, Oct. 2004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167691104000532

[66] W. Gao, Y. Liu, Q. Li, and B. Lu, "Aerodynamic Modeling and Simulation of Multi-Lifting Surfaces Based on the Unsteady Vortex Lattice Method," *Aerospace*, vol. 10, no. 2, p. 203, Feb. 2023, number: 2 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2226-4310/10/2/203

[67] J. M. T. Thompson and H. B. Stewart, *Nonlinear Dynamics and Chaos*. John Wiley & Sons, 2002.

[68] G. P. Tolstov, *Fourier Series*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1962.

[69] J. Wu, L. Hong, and J. Jiang, "A robust and efficient stability analysis of periodic solutions based on harmonic balance method and Floquet-Hill formulation," *Mechanical Systems and Signal Processing*, vol. 173, p. 109057, Jul. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0888327022002291

[70] F. Bayer and R. I. Leine, "Sorting-free Hill-based stability analysis of periodic solutions through Koopman analysis," *Nonlinear Dynamics*, vol. 111, no. 9, pp. 8439–8466, May 2023. [Online]. Available: https://doi.org/10.1007/s11071-023-08247-7

[71] W. M. H. Greaves, "On the Stability of the Periodic States of the Triode Oscillator," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 22, no. 1, pp. 16–23, Feb. 1924. [Online]. Available: https://www.cambridge.org/core/product/identifier/S0305004100000050/type/journal_article

[72] L. Liu, E. H. Dowell, and K. C. Hall, "A novel harmonic balance analysis for the Van Der Pol oscillator," *International Journal of Non-Linear Mechanics*, vol. 42, no. 1, pp. 2–12, Jan. 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020746207000200

[73] M. A. Pinsky, *Introduction to Fourier Analysis and Wavelets*. United States of America: Brooks/Cole, 2002, page 27.

[74] T. Theodorsen, "General Theory of Aerodynamic Instability and the Mechanism of Flutter," Tech. Rep. NACA-TR-496, Jan. 1949, nTRS Author Affiliations: National Advisory Committee for Aeronautics. Langley Aeronautical Lab. NTRS Document ID: 19930090935 NTRS Research Center: Legacy CDMS (CDMS). [Online]. Available: https://ntrs.nasa.gov/citations/19930090935

[75] M. Conner, D. Tang, E. Dowell, and L. Virgin, "NONLINEAR BEHAVIOR OF A TYPICAL AIRFOIL SECTION WITH CONTROL SURFACE FREEPLAY: A NUMERICAL AND EXPERIMENTAL STUDY," *Journal of Fluids and Structures*, vol. 11, no. 1, pp. 89–109, Jan. 1997. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0889974696900687

[76] M. McGurk and J. Yuan, "Prediction and validation of aeroelastic limit cycle oscillations using harmonic balance methods and Koopman operator," *Nonlinear Dynamics*, Mar. 2025. [Online]. Available: https://doi.org/10.1007/s11071-025-11065-8

[77] L. Liu and E. H. Dowell, "High Dimensional Harmonic Balance Analysis for Dynamic Piecewise Aeroelastic Systems." American Society of Mechanical Engineers Digital Collection, Aug. 2009, pp. 659–669. [Online]. Available: https://dx.doi.org/10.1115/IMECE2008-66681

[78] G. Dimitriadis, *Unsteady Aerodynamics: Potential and Vortex Methods*. John Wiley & Sons, Ltd, 2024.

[79] G. Wang, Z. Qi, and J. Xu, "A high-precision co-rotational formulation of 3D beam elements for dynamic analysis of flexible multibody systems," *Computer Methods in*

Applied Mechanics and Engineering, vol. 360, p. 112701, Mar. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782519305894

[80] X. Zhou, "Investigation and improvement of a co-rotational 3d beam element including plasticity and non-linear local element formulation," M.Sc. thesis, Technical University of Munich, Munich, 2021, supervised by Klaus B. Sautter and Prof. Dr.-Ing. Kai-Uwe Bletzinger.

[81] A. Dhooge, W. Govaerts, and Y. A. Kuznetsov, "Numerical Continuation of Branch Points of Limit Cycles in MATCONT," in *Computational Science - ICCS 2004*, M. Bubak, G. D. van Albada, P. M. A. Sloot, and J. Dongarra, Eds. Berlin, Heidelberg: Springer, 2004, pp. 42–49.

[82] "Chapel: Productive parallel programming." [Online]. Available: https://chapel-lang.org/

[83] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, "JAX: composable transformations of Python+NumPy programs," 2025. [Online]. Available: http://github.com/jax-ml/jax

[84] H. Wagner, "Über die entstehung des dynamischen auftriebes von tragflügeln," *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 5, no. 1, pp. 17–35, January 1925. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/zamm.19250050103/abstract

[85] R. T. Jones, "Operational treatment of the nonuniform-lift theory in airplane dynamics," NASA, Tech. Rep., October 1938. [Online]. Available: http://ntrs.nasa.gov/search.jsp?R=19930081472

APPENDIX A ANDERSON ACCELERATION METHOD

The implementation used in this work follows the description by Walker and Ni [43].

Given a function $f : \mathbb{R}^n \to \mathbb{R}^n$, the method seeks to find a fixed point $\boldsymbol{x}^*$ satisfying

$$\boldsymbol{x}^* = f(\boldsymbol{x}^*). \tag{A.1}$$

While simple fixed-point iteration $\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k)$ can converge slowly or fail to converge for many problems, Anderson acceleration improves convergence by utilizing information from m previous iterates to compute an improved update at each step.

The residual at iteration k is defined as

$$\boldsymbol{g}_k = f(\boldsymbol{x}_k) - \boldsymbol{x}_k. \tag{A.2}$$

At iteration k, the method considers the set of previous iterates $\{\boldsymbol{x}_{k-m_k}, \ldots, \boldsymbol{x}_k\}$ and residuals $\{\boldsymbol{g}_{k-m_k}, \ldots, \boldsymbol{g}_k\}$, where $m_k = \min(k, m)$ ensures we use at most m previous iterates. The algorithm constructs increment matrices:

$$\boldsymbol{X}_k = [\boldsymbol{x}_{k-m_k+1} - \boldsymbol{x}_{k-m_k}, \ldots, \boldsymbol{x}_k - \boldsymbol{x}_{k-1}] \tag{A.3}$$

$$\boldsymbol{G}_k = [\boldsymbol{g}_{k-m_k+1} - \boldsymbol{g}_{k-m_k}, \ldots, \boldsymbol{g}_k - \boldsymbol{g}_{k-1}] \tag{A.4}$$

The goal is now to find coefficients $\boldsymbol{\gamma}_k \in \mathbb{R}^{m_k}$ that minimize the norm of a linear combination of the residuals. This is achieved by solving the least squares problem:

$$\boldsymbol{\gamma}_k = \arg\min_{\boldsymbol{\gamma}} \|\boldsymbol{g}_k - \boldsymbol{G}_k\boldsymbol{\gamma}\|_2. \tag{A.5}$$

This optimization problem can be efficiently solved using a numerically stable QR decomposition. With $\boldsymbol{G}_k = \boldsymbol{Q}\boldsymbol{R}$ the QR factorization of $\boldsymbol{G}_k$, then

$$\boldsymbol{\gamma}_k = \boldsymbol{R}^{-1}\boldsymbol{Q}^T\boldsymbol{g}_k. \tag{A.6}$$

Once $\boldsymbol{\gamma}_k$ is computed, the next iterate is obtained via:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{g}_k - (\boldsymbol{X}_k + \boldsymbol{G}_k)\boldsymbol{\gamma}_k. \tag{A.7}$$

This update formula can be interpreted as applying a standard fixed-point step $\boldsymbol{x}_k + \boldsymbol{g}_k = f(\boldsymbol{x}_k)$, followed by a correction term $-(\boldsymbol{X}_k + \boldsymbol{G}_k)\boldsymbol{\gamma}_k$ that incorporates information from previous iterations.

The parameter m controls the depth of the history used in the acceleration. While larger m leads to faster convergence, it also increases memory usage and raises the computational cost. In this work, it was found that values between 5 and 10 provide the best results.

APPENDIX B THEODORSEN EQUATIONS

The Theodorsen model [74] provides a theoretical model for computing unsteady aerodynamic forces on thin airfoils undergoing harmonic oscillations. The total lift force is decomposed into two distinct contributions: the circulatory lift arising from bound circulation, and the apparent mass (or added mass) effects due to fluid acceleration:

$$L = L_m + L_c \tag{B.1}$$

where L_m represents the apparent mass contribution and L_c the circulatory contribution. The complete lift formulation is:

$$L = \underbrace{\rho b^2 \left(\pi U_\infty \dot{\alpha} + \pi \ddot{h} - \pi b a \ddot{\alpha}\right)}_{L_m} + \underbrace{2\pi\rho U_\infty b C(k) \left(U_\infty \alpha + \dot{h} + b\left(\frac{1}{2} - a\right)\dot{\alpha}\right)}_{L_c} \tag{B.2}$$

where:

- ρ is the fluid density
- b is the semi-chord length
- U_∞ is the freestream velocity
- α is the angle of attack
- h is the plunge displacement (positive downward)
- a is the pitch axis location normalized by semi-chord (measured from mid-chord)
- $k = \omega b / U_\infty$ is the reduced frequency
- $C(k)$ is the complex-valued Theodorsen function

The Theodorsen function accounts for the phase lag and amplitude attenuation of circulatory forces due to wake vorticity. It is expressed in terms of Hankel functions of the second kind:

$$C(k) = F(k) + iG(k) = \frac{H_1^{(2)}(k)}{H_1^{(2)}(k) + iH_0^{(2)}(k)} \tag{B.3}$$

where $H_n^{(2)}$ denotes the Hankel function of the second kind of order n. The real part $F(k)$ represents the in-phase component while the imaginary part $G(k)$ represents the out-of-phase component of the circulatory lift.

The classical Theodorsen formulation is limited to harmonic motion due to its frequency-domain nature. To extend the theory to arbitrary time-varying motion, we employ Wagner's indicial function $\phi(s)$ [84], which represents the circulatory lift response to a unit step change in angle of attack. The circulatory lift for arbitrary motion is expressed through the Duhamel superposition integral:

$$L_c(s) = 2\pi\rho U_\infty b \left[w(0)\phi(s) + \int_0^s \frac{dw(\sigma)}{d\sigma}\phi(s-\sigma)d\sigma \right] \tag{B.4}$$

where $s = U_\infty t/b$ is the non-dimensional time (semi-chords traveled), and the downwash is:

$$w(s) = U_\infty \alpha + \dot{h} + b\left(\frac{1}{2} - a\right)\dot{\alpha} \tag{B.5}$$

Wagner's function is related to the Theodorsen function through the inverse Fourier transform:

$$\phi(s) = \frac{1}{2\pi i}\int_{-\infty}^{\infty} \frac{C(k)}{k} e^{iks} dk \tag{B.6}$$

Direct evaluation of the Duhamel integral is computationally expensive. Jones [85] proposed an exponential approximation of Wagner's function:

$$\phi(s) \approx b_0 + b_1 e^{-\beta_1 s} + b_2 e^{-\beta_2 s} \tag{B.7}$$

with coefficients: $b_0 = 1$, $b_1 = -0.165$, $b_2 = -0.335$, $\beta_1 = 0.0455$, $\beta_2 = 0.3$.

To derive efficient ordinary differential equations for the circulatory lift, we take the Laplace transform of the Duhamel integral. Substituting Jones's approximation into the circulatory lift expression:

$$L_c(s) = 2\pi\rho U_\infty b \left[b_0 w(s) + \int_0^s b_1 e^{-\beta_1(s-\sigma)} \frac{dw(\sigma)}{d\sigma} d\sigma + \int_0^s b_2 e^{-\beta_2(s-\sigma)} \frac{dw(\sigma)}{d\sigma} d\sigma \right] \tag{B.8}$$

Let's define the lag states:

$$x_1(s) = b_1\beta_1 \int_0^s e^{-\beta_1(s-\sigma)} w(\sigma) d\sigma \tag{B.9}$$

$$x_2(s) = b_2\beta_2 \int_0^s e^{-\beta_2(s-\sigma)} w(\sigma) d\sigma \tag{B.10}$$

Taking derivatives with respect to s and applying Leibniz's rule:

$$\frac{dx_1}{ds} = b_1\beta_1 w(s) - \beta_1 x_1(s) \tag{B.11}$$

$$\frac{dx_2}{ds} = b_2\beta_2 w(s) - \beta_2 x_2(s) \tag{B.12}$$

Converting to dimensional time ($t = sb/U_\infty$):

$$\frac{dx_1}{dt} = \frac{U_\infty}{b} \left(b_1\beta_1 w(t) - \beta_1 x_1(t)\right) \tag{B.13}$$

$$\frac{dx_2}{dt} = \frac{U_\infty}{b} \left(b_2\beta_2 w(t) - \beta_2 x_2(t)\right) \tag{B.14}$$

The complete lift force becomes:

$$L(t) = L_m(t) + L_c(t) \tag{B.15}$$

where:

$$L_m(t) = \rho b^2 \pi \left(U_\infty \dot{\alpha} + \ddot{h} - ba\ddot{\alpha}\right) \tag{B.16}$$

$$L_c(t) = 2\pi\rho U_\infty b \left[(b_0 + b_1 + b_2) w(t) - x_1(t) - x_2(t)\right] \tag{B.17}$$

The lift coefficient is then:

$$C_L(t) = \frac{L(t)}{\frac{1}{2}\rho U_\infty^2 c} \tag{B.18}$$

where $c = 2b$ is the chord length. This state-space formulation allows for numerical integration of the unsteady aerodynamic forces for arbitrary airfoil motion, requiring only the solution of two first-order ODEs for the lag states x_1 and x_2.

APPENDIX C 3-DOF PARAMETERS

This appendix presents the complete set of parameters used in the aeroelastic analysis of the three-degree-of-freedom wing-flap system.

Table C.1 System parameters for the three-degree-of-freedom aeroelastic model

Symbol	Description	Value	Unit
Geometric Parameters			
b	Semi-chord	0.127	m
a	Distance between mid-chord and elastic axis	-0.5	–
c	Distance between flap hinge and mid-chord	0.5	–
x_α	Distance between EA and center of gravity	0.4340	–
x_β	Distance between flap CG and flap hinge	0.02	–
Mass and Inertia Properties			
m	Mass of wing-aileron per unit span	1.5666	kg/m
m_t	Mass of wing-aileron and supports per unit span	3.39298	kg/m
I_α	Mass moment of inertia about wing EA per unit span	0.01347	kg·m
I_β	Mass moment of inertia about hinge per unit span	0.0003264	kg·m
μ	Mass ratio	31.8846	–
S_α	Static mass moment about wing EA per unit span	0.08587	kg
S_β	Static mass moment of flap about hinge per unit span	0.00395	kg
r_α	Radius of gyration around EA	0.7321	–
r_β	Radius of gyration around flap hinge	0.1140	–
Structural Properties			
k_h	Plunge stiffness coefficient	2818.8	kg/s^2
k_α	Pitch stiffness coefficient	37.34	kg·m^2/s^2
k_β	Flap stiffness coefficient	3.9	kg·m^2/s^2
ω_h	Uncoupled plunge natural frequency	42.5352	Hz
ω_α	Uncoupled pitch natural frequency	52.6506	Hz
ω_β	Uncoupled flap natural frequency	109.3093	Hz
σ	Frequency ratio	0.8080	–
ζ_h	Plunge damping ratio	0.0113	–
ζ_α	Pitch damping ratio	0.01626	–
ζ_β	Flap damping ratio	0.0115	–
Flow Properties			
ρ	Air density	1.225	kg/m^3
U	Freestream velocity	Variable	m/s
V	Reduced velocity	Variable	–

The values are taken from an experimental study by Conner et al. [75] and have been adapted to the chosen non-dimensional form of the equations of motion.

Some of the parameters listed in Table C.1 are related through the following equations:

$$
\begin{aligned}
\mu &= \frac{m}{\pi\,\rho_\infty\,b^2}, & r_\alpha^2 &= \frac{I_\alpha}{m\,b^2}, & r_\beta^2 &= \frac{I_\beta}{m\,b^2},\\
k_h &= m\,\omega_h^2, & k_\alpha &= I_\alpha\,\omega_\alpha^2, & k_\beta &= I_\beta\,\omega_\beta^2,\\
\sigma &= \frac{\omega_h}{\omega_\alpha}, & V &= \frac{U}{b\,\omega_\alpha}, & C_h &= 2\,m\,\omega_h\,\zeta_h,\\
C_\alpha &= 2\,I_\alpha\,\omega_\alpha\,\zeta_\alpha, & C_\beta &= 2\,I_\beta\,\omega_\beta\,\zeta_\beta, & \tau &= \omega_\alpha\,t.
\end{aligned}
\tag{C.1}
$$

APPENDIX D 3-DOF WING EQUATIONS

This appendix presents the equations of motion for the 3-DOF wing-flap system including the state space formulation of Theodorsen's unsteady aerodynamic model. A detailed list of the parameters and their description is provided in Appendix C. Given the dimensionless equation of motion:

$$\boldsymbol{M}_s\ddot{\boldsymbol{q}} + \boldsymbol{C}_s\dot{\boldsymbol{q}} + \boldsymbol{K}_s\boldsymbol{q} = \boldsymbol{f} \tag{D.1}$$

where $\boldsymbol{q} = [h/b, \alpha, \beta]^T$ are the variables for the plunge, pitch and flap deflections respectively, and $\boldsymbol{f} = \boldsymbol{f_s} + \boldsymbol{f_a}$ is the total force vector including nonlinear structural and aerodynamic forces.

The dimensionless matrices are given by [6]:

$$\begin{aligned}
\boldsymbol{M}_s &= \mu \begin{bmatrix} m_T/m & x_\alpha & x_\beta \\ x_\alpha & r_\alpha^2 & \left[(c-a)x_\beta + r_\beta^2\right] \\ x_\beta & \left[(c-a)x_\beta + r_\beta^2\right] & r_\beta^2 \end{bmatrix} \\
\boldsymbol{C}_s &= 2\mu \begin{bmatrix} \sigma\zeta_h & 0 & 0 \\ 0 & r_\alpha^2\zeta_\alpha & 0 \\ 0 & 0 & \frac{\omega_\beta}{\omega_\alpha} r_\beta^2 \zeta_\beta \end{bmatrix} \\
\boldsymbol{K}_s &= \mu \begin{bmatrix} \sigma^2 & 0 & 0 \\ 0 & r_\alpha^2 & 0 \\ 0 & 0 & \left(\frac{\omega_\beta}{\omega_\alpha}\right)^2 r_\beta^2 \end{bmatrix}
\end{aligned} \tag{D.2}$$

Note that in the case of a nonlinear degree of freedom, the stiffness matrix K_s must be modified accordingly to remove the linear stiffness term for that degree of freedom. In the case of a nonlinearity in the control surface, the structural force vector becomes:

$$\boldsymbol{f_s} = \begin{bmatrix} 0 \\ 0 \\ -\left(\frac{\omega_\beta}{\omega_\alpha}\right)^2 r_\beta^2 \mathcal{M}(\beta) \end{bmatrix} \tag{D.3}$$

where $\mathcal{M}(\beta)$ is a nonlinear moment function defined by the user.

The aerodynamic forces are obtained from either the UVLM or Theodorsen's model, the latter is briefly presented in Appendix B for a 2-DOF wing. For a 3-DOF wing, the aerodynamic

forces are given by:

$$\boldsymbol{f_a} = \boldsymbol{M}_a\ddot{\boldsymbol{q}} + \boldsymbol{C}_a\dot{\boldsymbol{q}} + \boldsymbol{K}_a\boldsymbol{q} + \boldsymbol{L_\delta}\boldsymbol{b} \tag{D.4}$$

By doing a mathematical development similar to the 2-DOF case, the generalized Theodorsen function is represented by Wagner's function $\Phi(t)$ [84] and computed using Jones' approximation [85].

$$\Phi(t) = 1 - \delta_1 e^{-\lambda_1 \frac{U}{b} t} - \delta_2 e^{-\lambda_2 \frac{U}{b} t} \tag{D.5}$$

with the coefficients $\delta_1 = 0.165$, $\delta_2 = 0.335$, $\lambda_1 = 0.0455$, $\lambda_2 = 0.3$. This yields two additional lag states $\boldsymbol{b} = [b_1, b_2]^T$ associated to the unsteady aerodynamics along with their differential equation definition.

$$\dot{\boldsymbol{b}} = \boldsymbol{Q}_a\ddot{\boldsymbol{q}} + \boldsymbol{Q}_v\dot{\boldsymbol{q}} + \boldsymbol{L}_\lambda\boldsymbol{b} \tag{D.6}$$

All the matrices are defined below:

$$\boldsymbol{M}_a = \begin{bmatrix} -1 & a & \frac{T_1}{\pi} \\ a & -\left(\frac{1}{8} + a^2\right) & -\frac{2T_{13}}{\pi} \\ \frac{T_1}{\pi} & -\frac{2T_{13}}{\pi} & \frac{T_3}{\pi^2} \end{bmatrix}$$

$$\boldsymbol{C}_a = V \begin{bmatrix} -2 & -2(1-a) & \frac{T_4 - T_{11}}{\pi} \\ 1+2a & a(1-2a) & \frac{1}{\pi}\left(T_8 - T_1 + (c-a)T_4 + aT_{11}\right) \\ -\frac{T_{12}}{\pi} & \frac{1}{\pi}\left(2T_9 + T_1 + (T_{12} - T_4)\left(a - \frac{1}{2}\right)\right) & \frac{T_{11}}{2\pi^2}\left(T_4 - T_{12}\right) \end{bmatrix}$$

$$\boldsymbol{K}_a = V^2 \begin{bmatrix} 0 & -2 & -\frac{2T_{10}}{\pi} \\ 0 & 1+2a & \frac{1}{\pi}\left(2aT_{10} - T_4\right) \\ 0 & -\frac{T_{12}}{\pi} & -\frac{1}{\pi^2}\left(T_5 - T_{10}\left(T_4 - T_{12}\right)\right) \end{bmatrix}$$

$$\boldsymbol{L}_\delta = 2V \begin{bmatrix} \delta_1 & \delta_2 \\ -\left(\frac{1}{2} + a\right)\delta_1 & -\left(\frac{1}{2} + a\right)\delta_2 \\ \frac{T_{12}\delta_1}{2\pi} & \frac{T_{12}\delta_2}{2\pi} \end{bmatrix}$$

$$\boldsymbol{Q}_a = \begin{bmatrix} 1 & \frac{1}{2} - a & \frac{T_{11}}{2\pi} \\ 1 & \frac{1}{2} - a & \frac{T_{11}}{2\pi} \end{bmatrix}$$

$$\boldsymbol{Q}_v = V \begin{bmatrix} 0 & 1 & \frac{T_{10}}{\pi} \\ 0 & 1 & \frac{T_{10}}{\pi} \end{bmatrix}$$

$$\boldsymbol{L}_\lambda = V \begin{bmatrix} -\lambda_1 & 0 \\ 0 & -\lambda_2 \end{bmatrix} \tag{D.7}$$

With the auxiliary terms given by Theodorsen [74]:

$$
\begin{aligned}
T_1 &= -\frac{1}{3}\sqrt{1-c^2}\left(2+c^2\right) + c\cos^{-1}(c) \\
T_2 &= c\left(1-c^2\right) - \sqrt{1-c^2}\left(1+c^2\right)\cos^{-1}(c) + c\left[\cos^{-1}(c)\right]^2 \\
T_3 &= -\left(\frac{1}{8}+c^2\right)\left[\cos^{-1}(c)\right]^2 + \frac{1}{4}c\sqrt{1-c^2}\cos^{-1}(c)\left(7+2c^2\right) - \frac{1}{8}\left(1-c^2\right)\left(5c^2+4\right) \\
T_4 &= -\cos^{-1}(c) + c\sqrt{1-c^2} \\
T_5 &= -\left(1-c^2\right) - \left[\cos^{-1}(c)\right]^2 + 2c\sqrt{1-c^2}\cos^{-1}(c) \\
T_6 &= T_2 \\
T_7 &= -\left(\frac{1}{8}+c^2\right)\cos^{-1}(c) + \frac{1}{8}c\sqrt{1-c^2}\left(7+2c^2\right) \\
T_8 &= -\frac{1}{3}\sqrt{1-c^2}\left(2c^2+1\right) + c\cos^{-1}(c) \\
T_9 &= \frac{1}{2}\left[\frac{1}{3}\left(1-c^2\right)^{\frac{3}{2}} + aT_4\right] \\
T_{10} &= \sqrt{1-c^2} + \cos^{-1}(c) \\
T_{11} &= \cos^{-1}(c)(1-2c) + \sqrt{1-c^2}(2-c) \\
T_{12} &= \sqrt{1-c^2}(2+c) - \cos^{-1}(c)(1+2c) \\
T_{13} &= \frac{1}{2}\left[-T_7 - (c-a)T_1\right] \\
T_{14} &= \frac{1}{16} + \frac{1}{2}ac
\end{aligned}
\tag{D.8}
$$

To be able to integrate the coupled equations of motion and lag states, they are rewritten in state-space form as:

$$
\dot{\boldsymbol{y}} = \boldsymbol{B}^{-1}\left[\boldsymbol{A}\boldsymbol{y} + \boldsymbol{f}(\boldsymbol{y})\right] \tag{D.9}
$$

with the state vector $\boldsymbol{y} = [\dot{\boldsymbol{q}}^T, \boldsymbol{q}^T, \boldsymbol{b}^T]^T$.

The matrix $\boldsymbol{A}$ is defined as:

$$
\boldsymbol{A} = \begin{bmatrix} \boldsymbol{C}_a - \boldsymbol{C}_s & \boldsymbol{K}_a - \boldsymbol{K}_s & \boldsymbol{L}_\delta \\ \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times2} \\ \boldsymbol{Q}_v & \boldsymbol{0}_{2\times3} & \boldsymbol{L}_\lambda \end{bmatrix} \tag{D.10}
$$

the matrix $\boldsymbol{B}$ is defined as:

$$
\boldsymbol{B} = \begin{bmatrix} \boldsymbol{M}_s - \boldsymbol{M}_a & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times2} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times2} \\ -\boldsymbol{Q}_a & \boldsymbol{0}_{2\times3} & \boldsymbol{I}_{2\times2} \end{bmatrix} \tag{D.11}
$$

and finally the extended force vector is defined as:

$$\boldsymbol{f}(\boldsymbol{y}) = \begin{bmatrix} \boldsymbol{f_s} \\ 0_{3\times 1} \end{bmatrix} \tag{D.12}$$

APPENDIX E 3-DOF WING TIME SERIES RESULTS

This appendix presents the time series results for the 3-DOF wing time domain simulations discussed in Section 3.5. The following figures illustrate the temporal evolution of key aerodynamic and structural parameters over the course of the simulations.

These visualizations proved useful during code development and debugging, offering immediate insight into system dynamics and numerical behavior. Such detailed time series representations remain notably scarce in the literature, making their inclusion here particularly valuable for reference and comparison purposes.

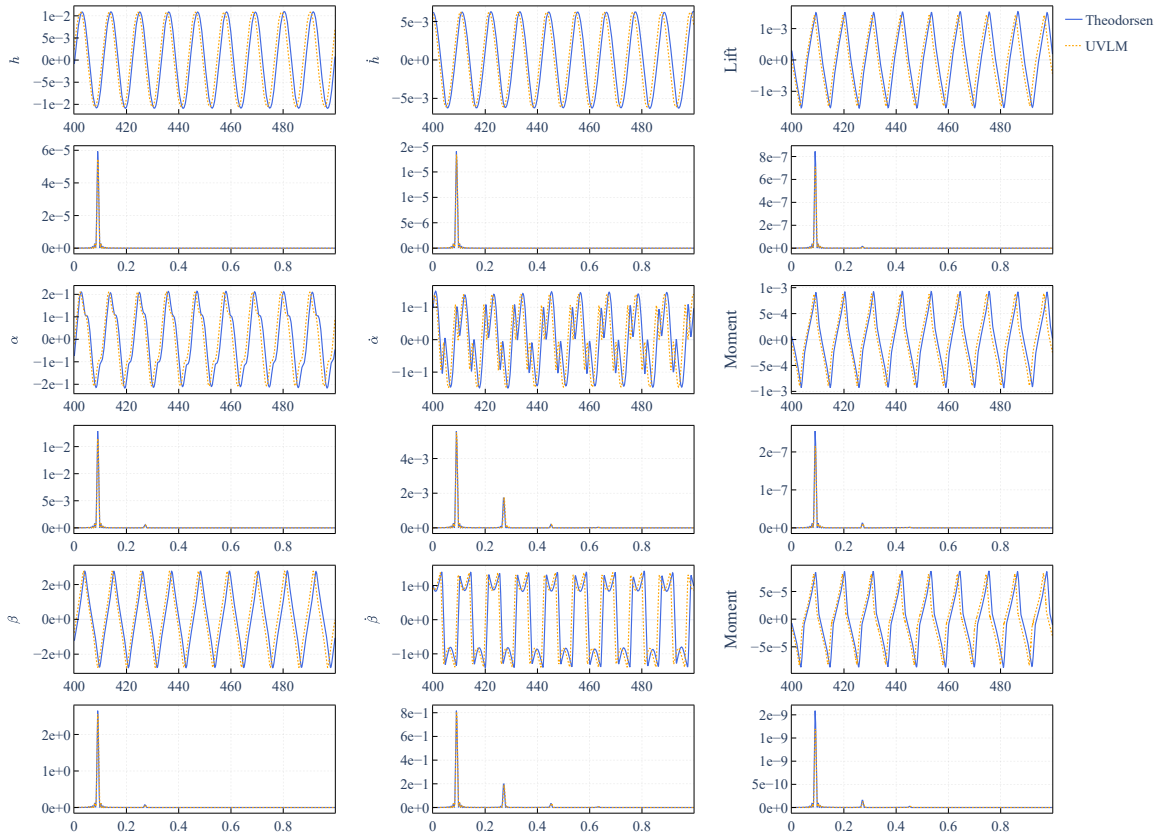


Figure E.1 Time series results for a 3DOF wing with a control surface with freeplay stiffness at $U = 7$ m/s

Both time series data and corresponding power spectral density (PSD) plots are presented for each parameter. The time series show the evolution of variables over non-dimensional time, while the PSD analysis reveals the frequency content of the responses. This dual

representation serves multiple purposes: it enables direct validation of response frequencies between UVLM and Theodorsen implementations independent of amplitude variations, and provides information about the harmonic content required for frequency-domain HBM analysis. The PSD results inform the selection of harmonic truncation levels necessary for accurate representation of the nonlinear dynamics.

Systems with freeplay nonlinearity usually exhibit extended transient behavior before reaching steady-state LCO conditions. It is therefore recommended to perform the simulation for longer durations than those presented here. The observed phase lag between UVLM and Theodorsen results comes from spatial discretization inaccuracies. The difference can be substantially reduced through mesh refinement, though at increased computational cost.

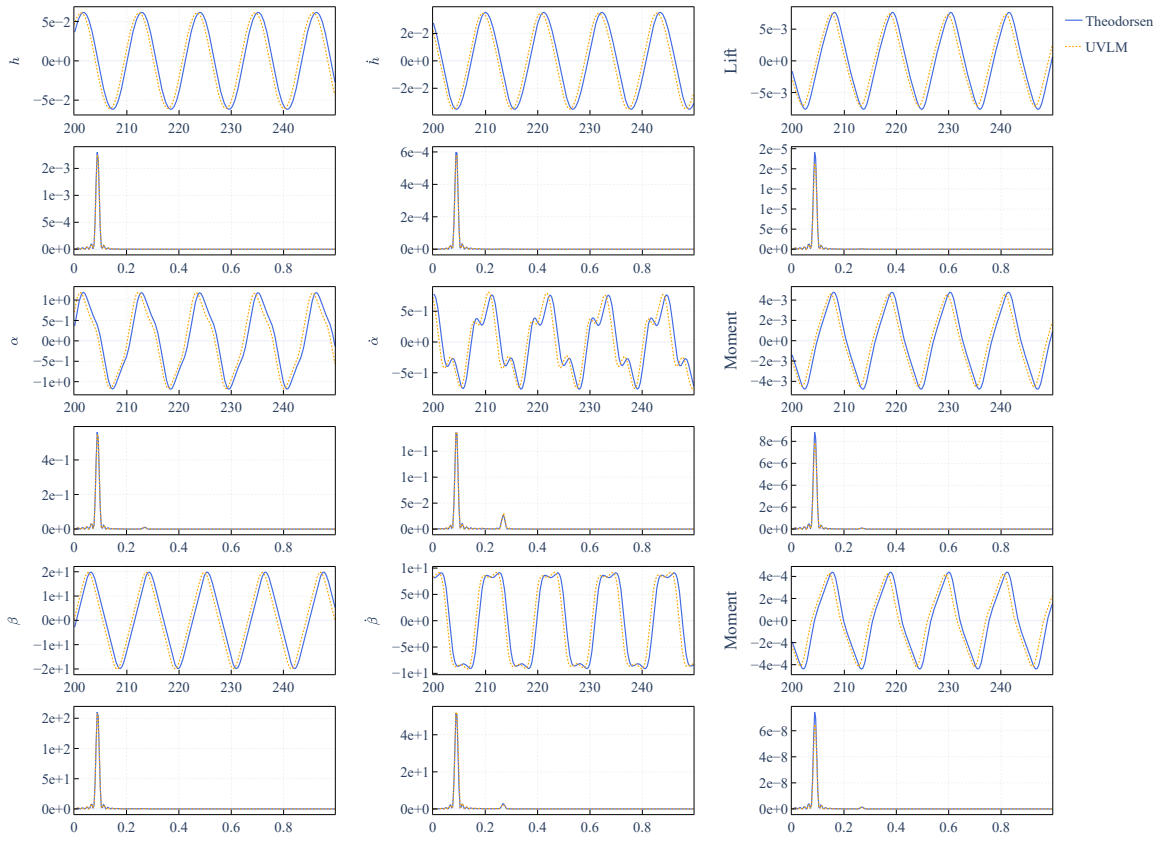


Figure E.2 Time series results for a 3DOF wing with a control surface with cubic stiffness at $U = 6$ m/s

Additional discrepancies come from numerical artefacts in the single precision kinematics calculations which accumulate progressively over extended simulation periods. Generally, the phase error growth rate correlates directly with simulation duration, becoming particularly pronounced in long-time integrations.