

# High-fidelity Aerostructural Optimization of Flexible Wings with Flutter Constraints

by

Eirikur Jonsson

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
(Aerospace Engineering)  
in the University of Michigan  
2020

Doctoral Committee:

Professor Joaquim R. R. A. Martins, Co-Chair  
Assistant Research Scientist Charles A. Mader, Co-Chair  
Professor Carlos E. S. Cesnik  
Professor Bogdan I. Epureanu

Eirikur Jonsson

@umich.edu

ORCID iD: 0000-0002-5166-3889

© Eirikur Jonsson 2020

To my dear wife Bryhildur, whose love, patience, and support has made my dreams possible.

## Acknowledgments

The completion of this work would not have been possible without the support of many people and I wish I could thank them all by name here. First, I would like to thank my advisor Prof. Joaquim R.R.A. Martins for giving me this great opportunity to join and study in the MDO Lab. His guidance, support, research freedom, and enthusiasm for engineering design and optimization (including life optimization) has not only made me a better researcher, but also a better person. For this I am sincerely grateful. Thank you. I would like to extend my gratitude to Dr. Charles (Sandy) Mader. His willingness to provide continuous support and knowledge was indispensable in the completion of this work. His systematic approach addressing challenging problems and troubleshooting is truly inspiring. I also want to acknowledge Prof. Carlos Cesnik and Prof. Bogdan Epureanu for their time and consideration, valuable suggestions, and insights, improving the quality of this work.

The members of the MDO Lab have also been an integral part in the success of this work. The work presented here leverages efforts of the previous MDO Lab alumni, whose contributions provide the foundation for this work. I especially like to thank Dr. Gaetan K. W. Kenway, whom I shared an office with and learned much from, for his help and support during the initial development of this work. I would also like to thank Professor Graeme J. Kennedy whose help, support, and very capable finite-element solver made many of the developments in this work possible. I would also like to thank my office mates whom I shared an office with for the longest—Anil, Neil, and Sham—for great technical discussions, all the fun times, and for putting up with me and making the office a welcoming place to be. Other members of the lab with whom I worked directly and would like to thank include: Ali, Ben, David, Gustavo, Jason, John J., John H., Josh, Justin, Marco, Mohamed, Ney, Nick, Ping, Sabet, Shugo, Sicheng, Tim, Xiaosong, Yingqian, Zhoujie (Peter). I have made many friends, whose friendship will extend well in to the future. I will certainly miss Ann Arbor and the FXB building. The Aerospace Department at the University of Michigan has some truly talented students and amazing faculty and supporting staff. I would especially like to thank Denise Phelps for all her efforts and assistance throughout my stay, in particular for assisting my wife, providing a place to sit and

proctoring her Reykjavik University exams.

I would not be here without the support of my friends and family. I am thankful for my parents and their continuous encouragement and believe in me. Their frequent visit to Ann Arbor, along with my in-laws, to help and stay with grandchildren, was invaluable and certainly made us feel closer to home. Finally, to my wife, Bryhildur, I owe the greatest dept of gratitude. Thank you for being patient, optimistic, and giving me strength to overcome difficulties during the PhD. Your encouragement and love is truly unbounded. During our stay in Ann Arbor we welcomed our son, Jon, into the family, joining our daughter Kristin. Without you all, this work would not have been possible.

# Table of Contents

Dedication . . . . .	ii
Acknowledgments . . . . .	iii
List of Figures . . . . .	viii
List of Tables . . . . .	xii
List of Appendices . . . . .	xiv
Abstract . . . . .	xv
<b>Chapter</b>	
<b>1 Introduction, Background and Objectives . . . . .</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Background on Multidisciplinary Design Optimization . . . . .	5
1.3 Background on Flutter Modeling . . . . .	7
1.4 Flutter Analysis in Aircraft Design Optimization . . . . .	17
1.4.1 Prediction Methods . . . . .	18
1.4.2 Application to Optimization . . . . .	19
1.4.3 Open Problems . . . . .	31
1.5 Thesis Objectives . . . . .	33
1.6 Thesis Outline . . . . .	34
<b>2 Existing Computational Tools . . . . .</b>	<b>36</b>
2.1 Geometric Parametrization . . . . .	38
2.2 CSM Solver . . . . .	38
2.3 CFD Solver . . . . .	39
2.4 Mesh Deformation . . . . .	39
2.5 Aerostructural Solver . . . . .	40
2.6 Optimizer . . . . .	40
<b>3 Flutter Prediction and Constraint Formulation . . . . .</b>	<b>41</b>
3.1 Flutter Analysis . . . . .	42
3.1.1 Load and Displacement Transfer . . . . .	43
3.1.2 Generalized Structural Matrices . . . . .	44
3.1.3 Generalized Aerodynamic Loads . . . . .	46
3.1.4 Generalized Equations of Motion . . . . .	50

3.2	Non-iterative Flutter Solution Method . . . . .	53
3.3	Mode Tracking . . . . .	56
3.3.1	Existing Methods . . . . .	56
3.3.2	Mode Tracking Algorithm . . . . .	58
3.4	Flutter Constraint Formulation . . . . .	59
3.5	Flutter Constraint Aggregation . . . . .	64
3.6	Integration into MACH . . . . .	67
<b>4</b>	<b>Flutter Constraint Derivative Implementation . . . . .</b>	<b>70</b>
4.1	Automatic Differentiation . . . . .	71
4.2	Component Derivative Implementation . . . . .	73
4.3	Derivative Verification . . . . .	75
4.3.1	Intermediate Derivatives . . . . .	77
4.3.2	Total Derivatives . . . . .	77
<b>5</b>	<b>Aerostructural Optimization of Plate . . . . .</b>	<b>79</b>
5.1	Baseline Model Description . . . . .	79
5.1.1	Baseline Model Flutter Analysis . . . . .	82
5.2	Problem Statement . . . . .	82
5.3	Design Space Analysis . . . . .	86
5.4	Optimization Results . . . . .	90
5.5	Summary . . . . .	94
<b>6</b>	<b>Aerostructural Optimization of Wingbox . . . . .</b>	<b>95</b>
6.1	Wing Model Description . . . . .	96
6.1.1	Aerodynamic Models . . . . .	96
6.1.2	Structural Model . . . . .	97
6.1.3	Structural Parametrization . . . . .	98
6.2	Structural Pre-Optimization . . . . .	102
6.2.1	Problem Statement . . . . .	102
6.2.2	Design Variables . . . . .	103
6.2.3	Constraint Definition . . . . .	104
6.2.4	Results . . . . .	105
6.3	Aerostructural Optimization . . . . .	109
6.3.1	Objective Function and Flight Conditions . . . . .	109
6.3.2	Design Variables and Bounds . . . . .	111
6.3.3	Constraints . . . . .	111
6.4	Results . . . . .	113
6.4.1	Modal Analysis . . . . .	116
6.4.2	Flutter Analysis . . . . .	118
6.5	Summary . . . . .	120
<b>7</b>	<b>Aerostructural Optimization of XRF1 . . . . .</b>	<b>121</b>
7.1	XRF1 Model . . . . .	122
7.1.1	Geometry . . . . .	122
7.1.2	Aerodynamic Model . . . . .	123

7.1.3	Structural Model . . . . .	124
7.1.4	Structural Parametrization . . . . .	125
7.2	Mass Element Implementation . . . . .	126
7.2.1	Element Definition . . . . .	126
7.2.2	Implementation Verification . . . . .	130
7.2.3	Fuel and Engine Mass Implementation . . . . .	130
7.3	Optimization Problem . . . . .	136
7.3.1	Objective Function and Flight Conditions . . . . .	136
7.3.2	Design Variables . . . . .	139
7.3.3	Constraint Definition . . . . .	144
7.4	Optimization Results . . . . .	148
7.4.1	Design Differences . . . . .	150
7.4.2	Modal Differences . . . . .	155
7.4.3	Flutter Differences . . . . .	157
7.5	Summary . . . . .	159
<b>8</b>	<b>Conclusion . . . . .</b>	<b>162</b>
8.1	Conclusions and Final Remarks . . . . .	162
8.2	Contribution . . . . .	166
8.3	Recommendations for Future Work . . . . .	167
	<b>Appendices . . . . .</b>	<b>171</b>
	<b>Bibliography . . . . .</b>	<b>188</b>



## List of Figures

### FIGURE

1.1	Aeroelastic system response before and past the flutter point. Prior to reaching the flutter point, the aeroelastic response is damped. At the flutter point, the system response is an oscillation with a small constant amplitude. Past the flutter point, a linear system response diverges, while a system with structural or aerodynamic nonlinearities develops a stable response with finite amplitude that remains constant in time, known as LCO. . . . .	8
1.2	$V - g$ diagrams for different types of flutter. Soft flutter is a gradual loss of damping with increased speed while hard flutter occurs abruptly and violently. A hump mode manifests itself as a damping decrease followed by an increase, which may result in considerably lower flutter speed. . .	10
1.3	Characteristic transonic dip of a transport wing. Aerodynamic nonlinearities due to shock waves and flow separation have a significant impact on the flutter speed, which may decrease dramatically. Linear theory (e.g., DLM) is non-conservative when compared to nonlinear viscous theory (e.g., RANS). Nonlinear inviscid theory (e.g., Euler) predicts highly conservative flutter speed at the dip, but it is generally closer to viscous theory predictions. . . . .	12
2.1	XDSM [1] overview of the aerostructural optimization process using MACH.	37
3.1	XDSM [1] of the proposed flutter analysis process and constraint formulation. . . . .	42
3.2	Collars triangle. . . . .	43
3.3	Hypothetical system with two modes. Black dots represent a valid solution, i.e., where the modes intersect the diagonal line, $\Im(p) = k$ , depicted in black. . . . .	55
3.4	Possible sources of discontinuities in the flutter constraint. Mode 1, critical at design $\mathbf{x}^i$ , switches with mode 2 as the critical mode in design $\mathbf{x}^{i+1}$ [2]. . . . .	61
3.5	Two possible methods to prevent discontinuities in the flutter constraint: frequency-separation method (left) and damping boundary (right, solid black) [2]. . . . .	61

3.6	Bounding curve and configurable constants applied to mitigate issues with discontinuities due to hump and mode swaps. The intersection of the bounding curve and the zero axis implicitly defines the minimum flutter speed $q_F$ . . . . .	63
3.7	KS aggregation of vectors containing only zeros gives a positive value. It is important to compensate for this effect if in-plane modes are present by shifting the bounding curve $G(q)$ . . . . .	66
3.8	XDSM [1] of the proposed flutter analysis process and constraint formulation integrated into the high-fidelity aerostructural optimization MACH framework. . . . .	68
4.1	XDSM [1] of derivatives required for the proposed flutter analysis and constraint formulation presented in reverse mode. The derivatives are computed by a combination of analytic adjoint and automatic differentiation (AD) in reverse. . . . .	74
5.1	Flat plate structural and aerodynamic mesh shown in red and black respectively. The plate is cantilevered at the left edge. A Free-Form-Deformation (FFD) volume is also shown with 8 control points which are depicted as red spheres connected by solid black lines. . . . .	80
5.2	First 4 modes shapes for the baseline geometry. . . . .	81
5.3	Flutter analysis of the flat plate baseline geometry. The unstable area is highlighted with a pink color in a), b). Frequency migration is shown in c). . . . .	83
5.4	XDSM [1] showing the flutter constraint as applied in the optimization. . . . .	86
5.5	Contour plot of the objective function, range, shown with the flutter constraint applied to the contour plot. Blanked out regions represent values of where the constraint is violated or $KS_{flutter} > 0$ . To generate the contour, the design space is sampled using 32 points in both variables for a total of 1024 design points. . . . .	87
5.6	Aggregated damping value, $KS_{flutter}$ , for a slice through the design space at $t = 0.001619$ m. Small change in the design, increasing the aspect ratio from 2.176 to 2.196, results in a discontinuity. . . . .	88
5.7	Small change in the design from aspect ratio from 2.176 to 2.196, shown in solid and dashed lines respectively, results in a large change in the damping. Frequency changes minimally as expected. . . . .	89
5.8	First five mode shapes for designs at aspect ratio 2.169 and 2.196. Mode 4 and 5 swap between the two designs. . . . .	89
5.9	Damping and frequency plots for designs with aspect ratio 5.235 (solid) and 5.706 (dashed) for a fixed thickness. Increasing the aspect ratio shifts the bifurcation of mode 1 to a lower velocity. Frequency changes minimally as expected. . . . .	91
5.10	Baseline (dashed) and optimized (solid) frequency and damping characteristics. . . . .	92
5.11	First 4 modes shapes for the optimized geometry. . . . .	92

5.12	Optimization results showing the major iterations (left) taken by the optimizer and the initial and optimized wing planform (right). Optimum aspect ratio and thickness are $AR^* = 6, t^* = 0.00166$ m, respectively . . .	93
5.13	Optimization convergence history. . . . .	94
6.1	CFD and DLM surface meshes used in this work. . . . .	97
6.2	Wingbox structural model. . . . .	98
6.3	Panel geometry and the design variables available in the smeared stiffness model. . . . .	99
6.4	Modal analysis of a panel with low stiffness, demonstrating the need to increase panel thickness. A smeared stiffness approach is used to avoid excessively thick panels. . . . .	104
6.5	Cumulative convergence history of the 5 structural optimizations. The orange markers represent a start of a new structural optimization. The buckling constraints prove to be more difficult satisfy than the failure constraints. . . . .	107
6.6	First eight natural mode shapes of the structural optimum. . . . .	108
6.7	Flutter damping and frequency characteristics of the structural optimum. This structural design is not flutter critical, since none of the modes cross into the region defined by the bounding curve $G(q)$ . . . . .	109
6.8	Planform comparison of the baseline and the two aerostructural optimal designs, with and without the flutter constraint. . . . .	114
6.9	Stiffener thickness, panel thickness, local failure, and buckling values shown for the two optimal designs, without and with flutter constraint. A diverging color map highlights the difference between the designs. . .	117
6.10	Tip deflection comparison between the two optimal solutions under the cruise and maneuver flight conditions. The deflection is normalized with respect to the semi-span. . . . .	118
6.11	First eight natural mode shapes from the two optimal aerostructural solutions obtained without (a) and with (b) the flutter constraint. . . . .	118
6.12	Flutter damping and frequency characteristics of the two aerostructural optimal designs without and with a flutter constraint. . . . .	119
7.1	XRF1 configuration. The wingbox is shown on the left and the aerodynamic surface on the right. The outline shown is the undeformed shape.	122
7.2	XRF1 configuration. . . . .	125
7.3	Point mass in space relative to global coordinate system and a grid node reference frame. . . . .	127
7.4	Rectangular wingbox. All bays are defined to hold fuel. User configured distribution scheme shown with different colors. Each fuel bay mass has user defined components. . . . .	133
7.5	Examples of the engine or leading edge devices (red sphere) applied to components defined by the user. . . . .	135
7.6	Flutter flight conditions. . . . .	140
7.7	Geometric and structural design variables overview. . . . .	141

7.8	Fuel mass for each bay split into multiple point masses that are then translated onto lower skins, show in red. . . . .	144
7.9	Optimization convergence history. . . . .	149
7.10	Equivalent thickness shown for the two optimal designs, without and with flutter constraint. A diverging color map highlights the difference between the designs. . . . .	152
7.11	Wing span and tip deflection comparison. . . . .	154
7.12	Aerodynamic and geometric properties of the two optimized designs. . .	156
7.13	First twelve natural mode shapes from the two optimal aerostructural solutions obtained without (a) and with (b) the flutter constraint. . . . .	158
7.14	Flutter damping and frequency characteristics of the two aerostructural optimal designs without and with a flutter constraint. . . . .	160
C.1	Structural thicknesses of the optimized designs with and without the flutter constraint. A diverging color map highlights the difference between the designs. . . . .	183
C.2	Structural thicknesses of the optimized designs with and without the flutter constraint. A diverging color map highlights the difference between the designs. . . . .	184
C.3	Buckling and failure comparison of the optimized designs with and without the flutter constraint, for the 2.5g maneuver condition. A diverging color map highlights the difference between the designs. . . . .	185
C.4	Buckling and failure comparison of the optimized designs with and without the flutter constraint, for the -1g maneuver condition. A diverging color map highlights the difference between the designs. . . . .	186
C.5	Buckling and failure comparison of the optimized designs with and without the flutter constraint, for the 1g maneuver condition. A diverging color map highlights the difference between the designs. . . . .	187

## List of Tables

### TABLE

1.1	Summary of efforts on optimization or derivative computation considering flutter [2]. . . . .	21
4.1	Intermediate sensitivities of the aggregated flutter constraint, $KS$ , with respect to a single value in the reduced stiffness $\mathbf{K}_r$ matrix, aerodynamic mesh points $\mathbf{X}^A$ matrix and the reduced aerodynamic mode shapes $\mathbf{Q}_r^A$ . Finite difference step size of $h = 10^{-6}$ gave overall the best results. . . .	77
4.2	Sensitivities of flutter constraint, $KS$ , with respect to design variables, chord, span and material thickness. Finite difference step size of $h = 10^{-3}$ for geometric variables (chord, span) and $h = 10^{-6}$ for structure variables gave overall the best results. . . . .	78
5.1	Flat plate mechanical properties, dimensions, and discretization of the structure and the aerodynamic surface. . . . .	80
5.2	Flat plate operating conditions under investigation used in the baseline analysis. . . . .	81
5.3	Optimization formulation of the flat plate problem. . . . .	85
5.4	Numerical comparison of baseline and optimized wing. . . . .	93
6.1	Aerodynamic geometry, CFD, and DLM mesh summary. . . . .	97
6.2	Mechanical properties, dimensions, and discretization of the wingbox structure. . . . .	98
6.3	A summary of the critical load computations for the overall and skin buckling constraints. Note that $A_n$ and $C_n$ are the modulus-weighted zeroth and first moments of area of the panel and stiffener, and $z_n$ is the modulus-weighted centroid. The bending stiffness $EI_s$ can then be used to determine the critical buckling load. . . . .	102
6.4	Total of 4 failure and 3 buckling constraints are used in this work. . . .	105
6.5	Optimization problem formulation for the structural mass minimization.	106
6.6	Structural optimum mass distribution of the final design based on component group. . . . .	108
6.7	Operating conditions considered in this work. . . . .	110
6.8	Aerostructural optimization problem statement. . . . .	113
6.9	Aerostructural range maximization final designs, with and without the flutter constraint (FC) compared to the baseline shape. . . . .	114

7.1	XRF1 Specifications . . . . .	123
7.2	Structure mechanical properties. . . . .	125
7.3	Mass element implementation verification results. Flat plate modal analysis results from TACS compare well with NASTRAN. . . . .	131
7.4	Flat plate frequency analysis using NASTRAN and TACS with weights computed using the inverse distance. . . . .	135
7.5	Flight conditions used in the multipoint aerostructural optimization. . .	138
7.6	Structural design variables and their grouping. Note that components in a given group, e.g. upper skin stiffener pitch, is controlled with a single design variable . . . . .	143
7.7	Buckling and failure KS aggregation constraints used for each maneuver flight condition. . . . .	146
7.8	Aerostructural optimization formulation. . . . .	148
7.9	Comparisons of the two optimized designs, without and with the flutter constraint active. . . . .	150
7.10	Structural mass by groups of components for both optimized designs. All values are given in kg. . . . .	151
7.11	The state of the buckling and failure KS aggregation constraints for each maneuver flight conditions. Both designs demonstrate the same state, thus only one dot shown for both cases. Active and inactive constraints are shown with filled or empty circle, respectively. . . . .	153

## List of Appendices

### APPENDIX

A Doublet Lattice Method . . . . .	171
B Lanczos Method . . . . .	179
C XRF1 . . . . .	182

## Abstract

High-fidelity optimization of aircraft has the potential to produce more efficient designs and to further reduce the risk of late design changes. For the optimization to generate a useful design, all the relevant constraints must be considered, including flutter. This is especially important in the high-fidelity aerostructural optimization of commercial aircraft, which is likely to result in high-aspect-ratio wing designs that are prone to flutter. To address this issue, we develop a flutter constraint formulation suitable for gradient-based aerostructural optimization with accurate and efficient adjoint derivatives. This approach scales well with the number of design variables and considers both structural sizing and aerodynamic planform variables. An effective bounding curve defines the flutter-free flight envelope, prevents discontinuities in the flutter constraint, and allows for minimum flutter speed to be specified implicitly. The flutter constraint formulation utilizes an efficient non-iterative  $p$ - $k$  method, an effective bounding curve, and an aggregation technique that results in a single constraint in the optimization problem. Accurate and efficient derivatives of the flutter constraint value with respect to structural sizing variables and aerodynamic planform variables are computed. Furthermore, to enable changes in planform, derivatives of the mode shapes are also computed efficiently. The derivatives are computed using a combination of analytic and automatic differentiation methods in reverse mode (adjoint) and rigorously validated using the complex-step method. We perform a multipoint, high-fidelity aerostructural optimization of a wing and full configuration aircraft with and without the flutter constraint, subject to stress and buckling con-



straints. With the flutter constraint, we obtain a stiffer, lower aspect ratio wing with stark differences in structural sizing, but without a significant reduction in objective. These results demonstrate the importance of including flutter constraints in wing design optimization. The proposed approach can be used to enforce such constraints in other applications and could be adapted to constrain other types of phenomena with the same form.

# CHAPTER 1

## Introduction, Background and Objectives

### 1.1 Introduction

Flutter is a dynamic aeroelastic instability that causes divergent oscillatory vibrations [3]. It is an undesirable phenomenon in aircraft because it can cause structural damage or failure, performance and ride comfort degradation, or loss of control. Flutter computations are typically performed only after an initial detailed design of the aircraft is completed, because they require the vehicle stiffness, mass, and aerodynamic models to be available [4]. If the design does not satisfy the flutter requirements at this stage, a redesign is necessary, which adds costs and delays to the aircraft development cycle. Thus, it is desirable to consider flutter concurrently with the aircraft design and the wing design in particular. Such a process would not only shorten the design cycle, but also allow for advantageous design trade-offs between the flutter requirements, the other constraints, and the aircraft performance.

Performing multidisciplinary design optimization (MDO) that considers both aerodynamic shape and structural sizing simultaneously while enforcing flutter constraints is a way to address this issue [5, 6]. Structural optimization alone, even if including aerostructural analyses for enforcing flutter constraints, yields design solutions with suboptimal performance compared to the optimal designs resulting from MDO, where structural and aerodynamic sizing variables are optimized simultaneously [6, 7].

MDO can minimize structural weight, fuel consumption, or a combination of these two objectives with respect to wing shape, internal structure arrangements, and sizing, while accounting for the interactions between aerodynamics, structures, and other disciplines, and satisfying various constraints. MDO with flutter constraints results in designs with optimal aeroelastic tailoring. Omitting flutter constraints in the MDO process when minimizing fuel consumption tends to yield light-weight, high-aspect-ratio wing (HARW) designs that despite being highly efficient may not be feasible [8, 9].

After the aircraft has been designed and a prototype has been built, certification requires flight tests to demonstrate that the aircraft be free from flutter in the flight envelope with a 15% safety margin beyond the dive speed. If flutter is discovered at the flight test certification stage, it requires redesign to address it, incurring additional costs. The redesign effort typically increases the structural weight, reducing the performance originally anticipated for the aircraft.

The trend towards HARW aircraft is driven by better fuel efficiency, but their increased flexibility makes it all the more important to consider flutter accurately and early in the design process [10]. Another recent trend is the increasing use of control surfaces to suppress flutter. Active flutter suppression systems can be incorporated late in the design process when aeroelastic instabilities are encountered and a passive solution such as redesign is impractical and expensive [11, 12]. Alternatively, MDO provides a way to obtain the best possible configuration by co-designing the wing shape and internal structure, which contribute to passive flutter suppression that can then be augmented with an active flutter control system.

While there has been extensive work in methods for flutter analysis, integrating flutter constraints into design optimization requires additional considerations. Models used for flutter prediction should capture the relevant physics with adequate accuracy to correctly drive the optimizer, and inevitably there is a compromise between model

fidelity and computational cost. To include flutter analysis in a numerical optimization cycle, speed of execution is particularly important to make sure that the overall optimization process is tractable.

Another important characteristic for integrating flutter analysis into the optimization process is the robustness of the flutter prediction method. Since the optimization process automatically samples the design space, it is likely to request for the analysis of designs that would normally not be chosen by a human designer. Thus, it is important that the flutter analysis converges for such designs so that the overall optimization process is not interrupted.

Gradient-based optimization algorithms are needed to optimize practical aircraft configurations parameterized with a large number of design variables [13]. When using gradient-based algorithms, it is important to consider the smoothness of the objective and constraint functions, as well as the accuracy and efficiency of the derivative computations.

Optimizing HARW configurations subject to flutter constraints is even more challenging because it requires capturing couplings between aeroelasticity and flight dynamics along with geometric nonlinearities that arise in the presence of low natural vibration frequencies and significant structural flexibility [14, 15].

Nonlinearities in the structure (large deflections, free-play of control surfaces, follower loading) or the aerodynamics (shock waves and flow separation) can cause self-sustained oscillations of limited amplitude that remain constant in time, known as limit cycle oscillations (LCOs). For certain types of nonlinearities, LCOs may exist at flight conditions below the flutter point [16]. When nonlinear effects become important, post-flutter analysis should be integrated into the design process in the form of constraints to make sure that the optimal design is feasible.

There have been several review papers and textbooks on flutter and post-flutter analysis. Livne [17, 18] reviewed the state-of-the-art and future challenges in aeroe-

lasticity of conventional and unconventional vehicles. A recent review by the same author focused on active flutter suppression control systems [12]. Friedmann [19] reviewed the general challenges in nonlinear aeroelasticity, where the applications focused on rotary wings. Dowell et al. [16] classified nonlinear aeroelastic behaviors and discussed theoretical, computational, and experimental analysis efforts. de C. Henshaw et al. [20] discussed traditional industrial linear flutter prediction and recent efforts for including nonlinear effects, particularly due to transonic flows. More recently, Afonso et al. [10] reviewed nonlinear aeroelasticity of HARWs. Dimitriadis [21] discussed nonlinear post-flutter behaviors in aeroelastic systems and the related analysis methods. Beran et al. [22] reviewed methods for uncertainty quantification in aircraft aeroelasticity and their application to formulate nondeterministic optimization problems. However, the field lacks a review on the integration of flutter and post-flutter analysis as constraints in aircraft design optimization.

In this chapter, we address this shortcoming by reviewing methods for flutter prediction, and we discuss their advantages and disadvantages in the context of aircraft design optimization. We refer the reader to Jonsson et al. [2] for discussion on structural nonlinearities in flutter analysis and post-flutter analysis and their integration into design optimization.

First, we provide a brief background on multidisciplinary design optimization in Section 1.2 and on flutter modeling in Section 1.3. These sections emphasize the key aspects relevant to aircraft design and are beneficial for readers not familiar with either of these topics. These methods and examples of their application in aircraft design optimization problems are then discussed in Section 1.4. The chapter concludes with remarks on the state of this field and the open challenges to be addressed for integrating flutter considerations into the optimal design of aircraft configurations.

## 1.2 Background on Multidisciplinary Design Optimization

Multidisciplinary design optimization couples the relevant disciplines of an engineering system and performing a numerical optimization to aid the design of that system [23]. MDO considers several disciplines simultaneously such that their interactions can be leveraged, resulting in a better optimum than if each discipline were optimized sequentially [24]. Thus, considering MDO early in the design process allows engineers not only to improve the design, but also to minimize development time and cost of the overall design.

Performing MDO of aircraft configurations by describing its outer mold line (OML) and structural sizing with high fidelity requires a large number of design variables. Detailed aerodynamic optimization of wings requires hundreds of shape variables [25] and structural sizing based on a detailed finite-element wingbox model that is best utilized with an equally large number of sizing variables [26]. Gradient-based optimization methods are the feasible way to solve for high-dimensional problems within a reasonable computational time, especially when using high-fidelity analyses [13, 27]. Gradient-based methods require derivatives of the objective and constraint functions with respect to the design variables to help the optimization algorithm find the most promising search directions and establish rigorous optimality conditions.

While gradient-free algorithms are typically more robust and some of them explore the design space more widely, their cost is prohibitive when the number of design variables is large. Although gradient-based methods only guarantee convergence to a local optimum, this can be mitigated by using a multi-start technique [28]. Furthermore, recent studies failed to find multiple local minima (multimodality) in airfoil and wing shape design optimization [13, 25], except the case of planform optimizations where expected local minima were found related to choices such as upwards or

downwards winglets [29].

The efficacy of gradient-based algorithms relies on accurate and efficient gradient computations. The gradient accuracy directly affects the ability to converge to the optimum with a specified tolerance and the order of convergence of the optimization. In the best case, inaccurate gradients increase the number of iterations required for convergence and in the worst case cause early stopping due to numerical issues. Efficiency gradient computation is also important because this computation is sometimes the bottleneck in the optimization cycle.

When it comes to methods for computing gradients, the finite-difference method is a popular choice because it is easy to implement and can always be used, even with black-box codes. The major drawbacks of the finite-difference method is that it is inaccurate and its computational cost scales poorly with the number of design variables. Unlike the finite-difference method, the complex-step method [30] is accurate, but its cost still scales unfavorably with the number of design variables, making it prohibitive for wing design applications. Another option for computing gradients is automatic differentiation (AD), which uses a software tool to parse the code of an analysis to produce a new code that computes derivatives of that analysis [31, 32]. Although AD can scale well with the number of variables, it does not handle iterative simulations efficiently in general. Finally, analytic methods are the most desirable because they are both accurate and efficient, especially for iterative simulations [33]. However, they require significant implementation effort. There are two main approaches within the analytic methods: the direct approach and the adjoint approach. The adjoint approach is attractive because the computational cost is dependent on the number of outputs of interest (objectives and constraints) but independent of the number of design variables [33, 34]. A coupled-adjoint approach can be solved for solve static aeroelastic problems [7, 8, 35] and can be generalized to any multidisciplinary problem [36, 37].

In the context of aircraft design optimization subject to flutter or post-flutter constraints, most of the early efforts used gradient-based optimization with gradients computed with either finite differences or the direct analytic method. However, recent efforts implemented the more efficient adjoint approach, and some also used AD techniques. These applications are further discussed in Section 1.4.2.

### 1.3 Background on Flutter Modeling

For aircraft designs to be useful and practical, the underlying models used in the flutter and post-flutter analysis need to capture the correct physics involved. However, a simplification of the phenomena is often necessary to make problems tractable to solve. Therefore, the choice of model should balance the fidelity needed to obtain accurate predictions and the mathematical or computational tractability for design applications.

This section highlights the modeling aspects to be considered in flutter analysis, which is discussed in more detail in Section 1.4 for obtaining meaningful results in a design optimization. By flutter, we mean the onset of divergent oscillations as the flight conditions of aircraft cross the critical stability boundary (flutter boundary).

Mathematically, flutter occurs at a Hopf bifurcation point [38] beyond which the system is in the post-flutter regime. Several post-flutter behaviors are possible, as discussed in detail by Dimitriadis [21] for a two-dimensional aeroelastic system with stiffness and damping nonlinearities. Among these behaviors, we are particularly interested in self-sustained oscillations with limited amplitude that remains constant in time, known as LCOs.

LCOs typically develop beyond the flutter boundary; however, for certain types of nonlinearities, they can also occur before reaching the flutter boundary [14]. Integrating post-flutter analyses into the design process can prevent this undesirable situation.



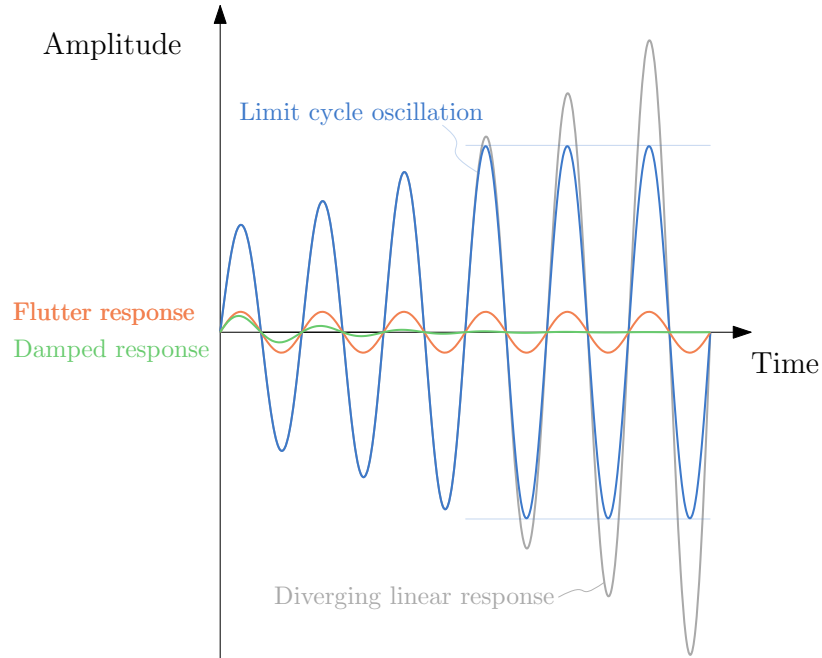


Figure 1.1: Aeroelastic system response before and past the flutter point. Prior to reaching the flutter point, the aeroelastic response is damped. At the flutter point, the system response is an oscillation with a small constant amplitude. Past the flutter point, a linear system response diverges, while a system with structural or aerodynamic nonlinearities develops a stable response with finite amplitude that remains constant in time, known as LCO.

For details on post-flutter analysis methods and their application in optimization we refer the reader to Jonsson et al. [2].

Flutter is defined as a self-exciting dynamic instability that is associated with the interaction of inertial, elastic, and aerodynamic forces [3]. At the onset of flutter, this aeroelastic instability can be physically described as an oscillation with a small amplitude that is constant in time triggered by a small-amplitude disturbance, as shown in Fig. 1.1. The flight condition in which the system damping vanishes, resulting in this self-sustained oscillation, represents the *flutter point* (or flutter boundary). For linear systems, the flutter point is defined as the minimum dynamic pressure at which at least one of the modes becomes unstable [3]. The dynamic pressure can be replaced by equivalent airspeed, and is a function of altitude and Mach number.

Past the flutter point, in the absence of restraining nonlinearities from the aerody-

namics, the structure, or both, the amplitude of the oscillations grow exponentially. Fluid-structure interactions may also result in a static instability called divergence [3], which is not associated with oscillations. As for flutter, the structural response grows unbounded past the onset point, eventually reaching a limited-amplitude oscillation if restraining nonlinearities are present.

In the following discussion, we focus mainly on flutter phenomena, because for many practical configurations flutter occurs before divergence. However, accounting for divergence and the associated post-critical response in the design process shares many of the modeling and analysis aspects associated with flutter. Furthermore, some of the analysis methods and constraints discussed in Section 1.4 are applicable to divergence as well as flutter. Moreover, in this work we focus on global wing or component flutter rather than localized effects such as panel flutter that typically occurs at supersonic flow conditions.

The possible flutter characteristics are illustrated in Fig. 1.2, which shows the variation of the modal damping with flight speed at a fixed altitude. This is known as  $V-g$  diagram, which is a classical tool used in linear flutter analyses for determining the flutter point and interpreting the flutter characteristics. A similar representation can be obtained by varying dynamic pressure at a fixed Mach number.

Damping changes with flight speed in different ways among different designs, leading to different flutter behaviors. *Soft flutter* occurs when the damping decreases gradually with increasing flight speed, while *hard flutter* occurs when this decrease is abrupt. Another possibility is that there is a gradual decrease in damping with increasing flight speed, all the way to cross the zero value and beyond, followed by a damping increase. This phenomenon is known as a *hump mode*. These concepts are important when considering how to formulate a smooth and continuous flutter constraint for gradient-based optimization and are discussed in Section 1.4.

In flutter analysis, the physics described above is often represented by less expen-

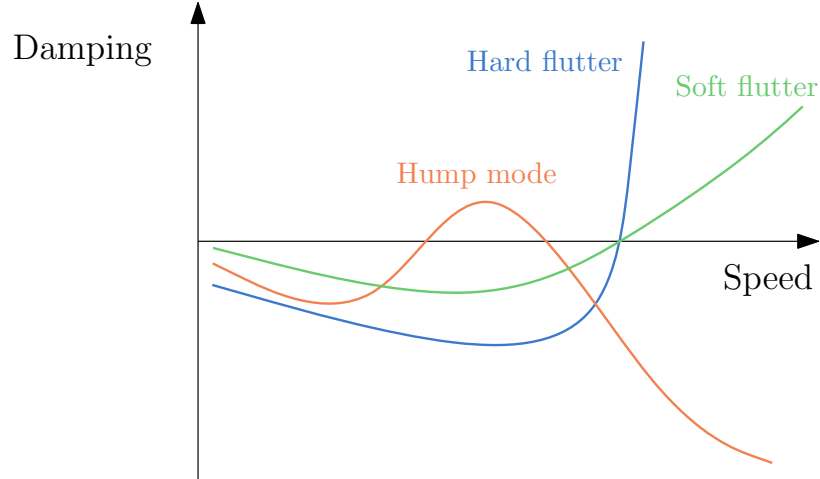


Figure 1.2:  $V - g$  diagrams for different types of flutter. Soft flutter is a gradual loss of damping with increased speed while hard flutter occurs abruptly and violently. A hump mode manifests itself as a damping decrease followed by an increase, which may result in considerably lower flutter speed.

sive linear models due to the number of conditions to be considered for certification. However, nonlinear structural and aerodynamic effects or the interaction between elastic and rigid-body degrees of freedom (DOF), which become important in the presence of low structural vibration frequencies, may significantly impact the flutter point. Therefore, the flutter prediction accuracy depends on the appropriate modeling of nonlinear effects and boundary conditions.

Furthermore, nonlinear effects impact not only the models used in flutter analysis, but also the analysis methods themselves. For linear systems, flutter characteristics do not depend on the deformation state. Therefore, flutter is typically analyzed by considering the unloaded and undeformed structure. For nonlinear systems, stability characteristics vary with the deformation configuration. Therefore, flutter analysis must be performed by computing the eigenvalues of the aeroelastic system linearized around equilibrium points for each flight condition to identify at what point the damping vanishes [14]. The eigenvalues can be computed by considering both the elastic and rigid-body DOFs (flutter in free flight) or by retaining only the elastic DOFs (traditional flutter) or the rigid-body DOFs (flight dynamic stability).

Many possible sources of nonlinearities can be present simultaneously in aeroelastic systems [10, 16, 21]. Here, we focus on aerodynamic nonlinearities due to transonic flow regimes and geometric structural nonlinearities due to large deflections, both of which are critical in the design of next-generation transport aircraft.

Aerodynamic nonlinearities due to shock waves and flow separation significantly impact the flutter speed. This decreases dramatically in the transonic regime, a phenomenon known as the *transonic dip* [39–43] illustrated in Fig. 1.3. Low-order, linear unsteady aerodynamic models commonly used in flutter analysis are in general accurate enough for subsonic and supersonic flows, but they severely overestimate the flutter speed for transonic conditions [44–46].

As shown in Fig. 1.3 for a hypothetical wing, linear theory is non-conservative when compared to nonlinear viscous models. Nonlinear inviscid models based on Euler equations can capture shock waves but they still fail to accurately predict the flutter boundary [47]. In many cases, the nonlinear inviscid theory predicts a highly conservative flutter speed at the dip, even though it is generally closer to viscous theory predictions. Depending on the severity of shocks, models based on Navier–Stokes equations (which include viscous and turbulence effects like boundary layer thickening, flow separation, and interactions between shocks and regions of separated flow) are necessary to obtain accurate flutter points [45]. Studies on various geometries demonstrated that taking into account viscous phenomena in the transonic flow regime improves the numerical prediction of transonic dip [48–51].

A common approach to improve the accuracy of transonic flutter computations while minimizing the increase in computational cost is to use numerical or experimental corrections applied to potential-flow linear models [52]. However, the correction data may not be available for optimization, either because it requires high-fidelity computations that are too expensive or because it is obtained from wind-tunnel measurements. This problem can be addressed by analyzing flutter using time-accurate

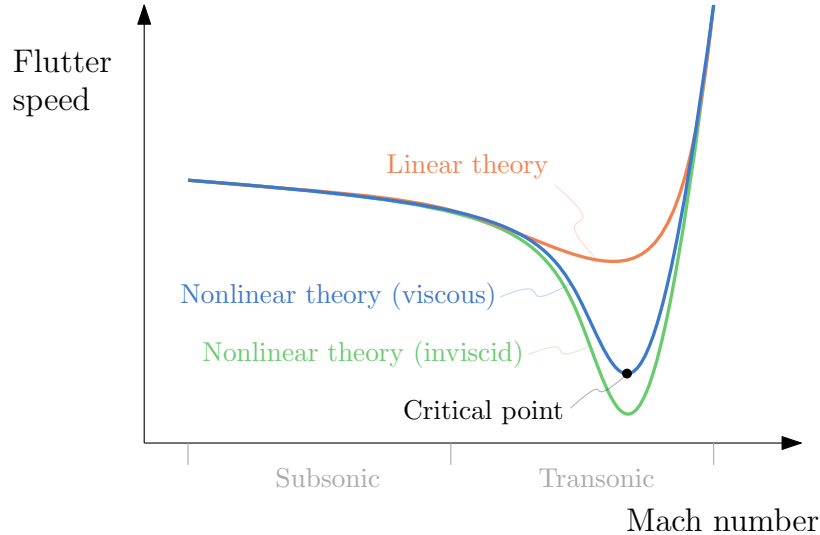


Figure 1.3: Characteristic transonic dip of a transport wing. Aerodynamic nonlinearities due to shock waves and flow separation have a significant impact on the flutter speed, which may decrease dramatically. Linear theory (e.g., DLM) is non-conservative when compared to nonlinear viscous theory (e.g., RANS). Nonlinear inviscid theory (e.g., Euler) predicts highly conservative flutter speed at the dip, but it is generally closer to viscous theory predictions.

dynamic simulations and higher-fidelity aerodynamic models. On the other hand, flutter prediction based on time-accurate computational fluid dynamics (CFD) is a challenge even for just analyzing the final configuration and is currently prohibitive for design space exploration.

Methods exist that try to preserve the computational efficiency of lower-fidelity methods while retaining the nonlinear physics modeled by the higher-fidelity CFD methods. One possibility is to use time-linearized transonic small disturbance (TSD) equations. Linear small-disturbance theory is inadequate for capturing strong transonic shocks, but small-disturbance solutions about the steady nonlinear background flow computed using high-fidelity CFD can provide acceptable performance and accuracy [53]. Another possibility is to use the transonic equivalent strip (TES) theory [54] and a provided pressure distribution from either experimental data or a high-fidelity CFD code to compute the small-disturbance transonic aerodynamic loads for flutter analysis [55]. Furthermore, several efforts have applied the time-linearization directly

to CFD solvers. This approach is appealing since it retains nonlinear effects, provides accurate results in the transonic regime, and intrinsically accounts for geometric properties of the body such as thickness and camber.

Kreiselmaier and Laschka [56] developed an unsteady method based on the small-disturbance Euler equations, which was later extended to small disturbance Navier–Stokes equations to include viscous effects [57]. The proposed method produced good results in the transonic flow regime [58, 59]. Thormann and Winghalm [60] developed a linear frequency domain (LFD) solver taking advantage of preconditioned Krylov GMRES [61] solution method. Later, Widhalm and Thormann [62] improved the algorithm and provided the analytic derivatives needed in the solution, improving the solver efficiency. The method was shown to be accurate when compared to full unsteady time-marching solutions.

These approaches consider unsteady aerodynamic models linearized about nonlinear equilibrium states and thus can capture the impact of static nonlinear effects on flutter. Moreover, they demonstrate computational savings well beyond an order of magnitude compared to fully unsteady time-marching solutions [56, 60]. However, computing derivatives of such methods for optimization is challenging due to the need for second-order derivative information.

Motivated by the interest in capturing key transonic flow physics with low computational cost, recent efforts also developed low-order unsteady transonic aerodynamic models suitable for integrating transonic flutter analyses into aircraft design.

Skujins and Cesnik [63] proposed a reduced-order unsteady aerodynamic model for multiple Mach regimes based on linear convolution with a nonlinear static correction. The methodology included error estimation capabilities based on the newly developed method of segments, which represents a flexible wing as a collection of rigid spanwise segments subject to local angle of attack and Mach number conditions. The method of segments was also applied to transonic flutter analysis of a transport vehicle by

Kitson and Cesnik [64].

Mallik et al. [65] developed a reduced-order model for HARW configurations by combining time-linearized Leishman indicial functions [66] with lift-curve and moment-curve slopes obtained by solving the RANS equations around airfoils at various Mach number and angle of attack conditions and for various thickness ratios. They obtained a state-space formulation for the airfoil unsteady aerodynamics to be used for eigenvalue analysis, which was extended to three-dimensional HARW discretized in spanwise strips by accounting for sweep correction. Flutter results were compared with wind-tunnel experiments for a truss-braced wing (TBW) configuration. The low-order model captured the transonic dip that was not predicted by potential-flow theory and presented good agreement with experiments at significantly lower computational cost compared to unsteady RANS simulations. These results showed the method suitability for conceptual HARW aircraft design including transonic flutter constraints.

Opgenoord et al. [67] developed a physics-based two-dimensional low-order model for transonic airfoils using the perturbations of the lowest-order volume-source and vorticity moments with respect to a known nonlinear background flow solution as the states. Evolution equations for these perturbations were derived and calibrated using data from high-fidelity Euler CFD simulations. A state-space unsteady aerodynamic model was obtained for airfoil flutter analysis which was later extended to three-dimensional HARW configurations [68] using strip theory and sweep correction, as done by Mallik et al. [65]. The method was applied in conceptual design and optimization problems including transonic flutter considerations [68, 69].

In addition to capturing transonic effects, a more recent flutter modeling need is to take into account geometric structural nonlinearities. These are particularly important in the analysis of HARW configurations, which achieve higher energy efficiency at the cost of increased structural flexibility and thus experience large deflections un-

der normal operating load conditions. The changes in geometric shape and stiffness properties due to these deflections significantly affect the flutter boundary [15]. When deformations are large, traditional flutter analysis based on the vehicle undeformed shape does not capture the actual behavior of the aircraft during flight.

Studies on isolated HARWs [70], high-altitude long-endurance (HALE) flying-wing configurations [14, 71, 72], and commercial transport vehicles [64] pointed out the need to analyze very flexible aircraft in statically deformed configuration at trim, which varies with the flight condition. Including structural nonlinearities in flutter prediction is challenging for both analysis and design due to the high computational cost of nonlinear aeroelastic simulations and the flutter boundary dependency on the deformation state, which is not considered in linear approaches.

Finally, classical wing flutter analyses typically assume the vehicle to be clamped at the wing root. While this may be an acceptable simplification for some vehicles, it does not reflect the vehicle behavior in free flight [17, 18]. For some configurations, simply including rigid-body DOFs influences the flutter solution substantially [14]. This occurs due to the coupling between rigid-body motion and structural dynamics that arise in the presence of low natural vibration frequencies. These interactions usually result in lower flutter points than the cantilevered configurations or different flutter mechanisms like body-freedom flutter (BFF) [73]. Therefore, it is imperative to understand the effect of boundary conditions and state variables on flutter prediction.

Mazidi et al. [74] investigated the effect of engine placement and roll maneuver on flutter results. They observed that the roll maneuver has a destabilizing effect on the flutter boundary dependent on the bank angle. Additionally, the location of the engine or external store greatly affects the flutter boundary and the roll-induced effects. Nearly all vehicles perform roll maneuvers during turns, making the inclusion of these conditions relevant to the aircraft design process.

There has been further work on the effect of rigid-body DOFs on the flutter



problem. Niblett [73] investigated the causes of BFF for a conventional wing-fuselage configuration using a linear analytical flutter method. Su and Cesnik [14] investigated the flutter behavior of a blended wing-body (BWB) aircraft both for cantilevered and free-flying conditions using the University of Michigan’s Nonlinear Aeroelastic Simulation Toolbox (UM/NAST). They observed a reduction in the flutter speed when the rigid body DOFs were included compared to the cantilever case. Moreover, the flutter mode changed to include pitch and plunge motions, resulting in BFF. Similarly, Jones and Cesnik [75] investigated the BFF characteristics of the X-56A experimental aircraft, describing the entire modeling process used for the flutter prediction. Cesnik and Su [76] analyzed the University of Michigan’s X-HALE very flexible aeroelastic testbed [77] and observed that significant wing deformations can drive lateral BFF due to the coupling of the Dutch roll and asymmetric wing bending modes. Su and Cesnik [72] investigated the stability and dynamic response of a highly flexible flying wing for different payload configurations and gust disturbances. They found that wing deflections can lead to an unstable phugoid mode and an aperiodic short-period mode. Similar behaviors were observed by Patil and Hodges [71] and Patil and Taylor [78]. Richards et al. [79] also analyzed the coupled flight dynamics and aeroelasticity of flying wings. They noted that BFF occurred due to a coupling of the short period pitching mode and the first elastic bending mode. They compared different inertial configurations of the aircraft and noted that BFF depends largely on the inertia about the pitch axis. They found a boundary value for pitch inertia that uncoupled the pitch and bending modes, thus replacing BFF with a more conventional flutter. After parameter studies, they concluded that BFF was caused by low fuselage inertia, which could be mitigated by redistributing the fuselage mass.

BFF is not exclusive to flying-wing configurations; conventional tube-and-wing aircraft can also encounter this type of instability. Cavallaro et al. [80] investigated the flutter behavior of a Prandtl Plane (boxed wing aircraft) including rigid-body

DOFs using MSC Nastran. Similarly to Richards et al. [79], they observed a dependence on the fuselage mass in causing BFF: while the baseline configuration encountered flutter without rigid-body contributions, increasing the fuselage mass resulted in BFF. Therefore, capturing these phenomena in flutter analyses is advisable even for conventional configurations and imperative when investigating non-conventional ones [17, 18].

## 1.4 Flutter Analysis in Aircraft Design Optimization

Because flutter is a safety-critical phenomenon, analyses and experimental investigations are required for vehicle certification. Analyzing a configuration for flutter late in the development cycle is likely to result in an inefficient design solution or pose challenges mitigating unexpected instabilities, resulting in performance decrease, financial losses, or both. For this reason, flutter should be integrated into the design process early in the form of a constraint.

In this section, we review flutter prediction methods and previous research that addressed optimization subject to flutter constraints. Past work has focused primarily on flutter analyses using linear structures and linear aerodynamic models. Further work considered transport aircraft operating in the transonic regime and thus used linear structures and nonlinear aerodynamics. The recent trend towards more flexible aircraft has led to research on constraining flutter for geometrically nonlinear structures.

Due to the sheer number of design variables typically used in practical aircraft optimization, most of the previous work reviewed here used gradient-based methods. However, studies on geometrically nonlinear configurations were limited to simple structures parametrized by few design variables. For this reason, efforts that consid-

ered gradient-free optimizations are also included.

### 1.4.1 Prediction Methods

Flutter computations are typically performed in the frequency domain by solving an eigenvalue problem. Well-established eigenvalue-analysis methods exist for flutter analysis of linear aeroelastic systems (e.g.,  $k$ -,  $p$ -,  $pk$ - and  $g$ -method) [81–85]. These methods are also applicable to nonlinear systems by linearizing the equations of motion about the nonlinear equilibrium configuration at each flight condition for capturing static nonlinearities due to the structure (large deflections) or the aerodynamics (background transonic flow). Direct methods, based on the Hopf-bifurcation theory, can also be used to predict the flutter point of nonlinear aeroelastic systems directly in the frequency domain [20].

It is also possible to predict flutter in the time domain, but this incurs a much higher computational cost. When analyzing the stability of a system in the time domain, the flutter point is typically evaluated by perturbing equilibrium configurations at different flight conditions and by time-marching the equations of motion in order to verify the decay or growth of the response, or to extract damping values in a post-processing stage [86–93].

The computational cost of time-domain flutter analysis based on transient simulations is currently prohibitive for optimization, particularly in the presence of aerodynamic or structural nonlinearities. The high computational cost is due to the large number of computations required to evaluate the flutter speed by means of flight speed (or dynamic pressure) sweep or bisection. Additionally, ascertaining the stability close to the flutter point requires long integration times due to small damping values.

For gradient-based optimization, another challenge is the efficient computation of derivatives for time-marched systems. Adjoint methods are advantageous for opti-

mizations with many design variables, but they are computationally expensive when applied to time marching solvers [94]. This is because the adjoint solution requires a reverse time integration following the forward integration for computing the system response [95], which results in large computational times, storage, and memory requirements [96]. Due to the above limitations, frequency-domain flutter prediction methods are more prevalent in literature, but some fully coupled unsteady aeroelastic adjoint implementations are found in the literature [97–102].

For a more detailed and general discussion on eigenvalue-based and direct methods we refer the reader to Jonsson et al. [2].

### 1.4.2 Application to Optimization

Despite early work by researchers such as Haftka [5, 103] and Hajela [104] optimization subject to flutter constraints is still not a standard design practice. More recently, several authors have integrated flutter constraints into design and investigated their effect on the optimal solutions. These efforts are summarized in Table 1.1 and reviewed below. They differ in the use of eigenvalue analysis (EV), direct (Hopf), or time-domain (TD) prediction methods for flutter analysis, the fidelity of structural and aerodynamic models, and the optimization problem formulation in terms of objective, type and number of design variables, and use of gradient-based (GB) or gradient-free (GF) solution algorithms. Efforts that used gradient-based algorithms also differ in the methods used for computing derivatives with respect to design variables. Finally, previous work shown in Table 1.1 differs in the types and number of design variables that were considered in the optimization problem. Most of the efforts included only structural sizing variables without considering changes in the aerodynamic shape. Fewer efforts optimized the airfoil or planform shapes. Including planform shape variables is challenging because changes in the mode shapes and the corresponding natural frequencies need to be considered when computing derivatives,

incurring additional computational cost.

Table 1.1: Summary of efforts on optimization or derivative computation considering flutter [2].

Effort	Method*	Models		Objective <sup>§</sup>	Design variables			$N$	Algorithm <sup>  </sup>
		Aerodynamics <sup>†</sup>	Structure <sup>‡</sup>		Structural	Shape			
						Airfoil	Planform		
Turner [105]	EV	Strip theory	Panel	Min mass	•			3	GB-Analytic
Bhatia and Rudisill [106]	EV	–	Beam	Min mass	•			12	GB-Analytic
Rudisill and Bhatia [107]	EV	–	Beam	Min mass	•			12	GB-Analytic
Gwin and Taylor [108]	EV	DLM	Beam	Min mass	•			33	GB-Analytic
Ringertz [109]	EV	DLM	Composite	Min mass	•			9	GB-Analytic
Mallik et al. [110]	EV	Strip theory	Beam	Min FB, Max TOGW	•	•	•	19	GF-Genetic
Stanford et al. [111]	EV	TSD+Euler	Shell	Min mass	•			92000	GB-Adjoint
Chen et al. [112]	EV	Euler	Shell	–	Derivative computation only			–	GB-Complex-step
Bartels and Stanford [113]	EV	RANS	Shell	Min mass	•			711	GB-Adjoint
Variyar et al. [9]	EV	Lifting line	NL beam	Min FB	•		•	12	GB-Finite difference
Xie et al. [114]	EV	DLM	NL beam	Min mass	•			44	GF-Direct
Bhatia and Beran [115]	EV	Euler	NL beam	Min mass	•			8	GB-Analytic
Lupp and Cesnik [116]	EV	Strip theory	NL beam	Min FB	•		•	5	GB-Adjoint
Kennedy et al. [117]	Direct	Panel	Shell	–	Derivative computation only			–	GB-Adjoint
Beran et al. [118]	Direct	ONERA stall	NL beam	–	Derivative computation only			–	GB-Adjoint
Mani and Mavriplis [97]	TD	Euler	Mass-spring	Max fl. speed		•		32	GB-Adjoint
Zhang et al. [99]	TD	RANS	Mass-spring	Max fl. speed		•		48	GB-Adjoint
Zhang et al. [100]	TD	RANS	FEM	Max fl. speed	•	•		120	GB-Adjoint

\* Methods: EV—Eigenvalue method, TD—Time-domain

† Aerodynamics: DLM—Doublet lattice method, TSD—Transonic small disturbance, RANS—Reynolds averaged Navier–Stokes

‡ Structures: NL—Nonlinear

§ Objective: FB—Fuel burn, TOGW—Takeoff gross weight

|| Algorithm: GB—Gradient based, GF—Gradient free

In one of the early efforts to constrain flutter, Turner [105] formulated a mass minimization problem by considering distribution of material rather than the structure topology to meet a specified flutter speed. Later, Bhatia and Rudisill [106] developed a numerical procedure to minimize wing mass while satisfying a flutter constraint. They applied the procedure to a uniform cross-section box beam consisting of three bays in order to minimize the mass while maintaining the flutter speed. Rudisill and Bhatia [107] improved the rate of convergence of this procedure by computing second-order derivatives of the eigenvalues and of the flutter speed with respect to the design variables. Gwin and Taylor [108] developed the method of feasible directions for the mass minimization of a structure subject to a minimum flutter speed. They were able to handle up to 33 structural design variables in a supersonic problem.

Due to the limited computational capabilities of the time, these early efforts used simple aerodynamic and structural models and employed a similar strategy to enforce the flutter constraint. They all formulated the flutter problem in the frequency domain as an eigenvalue problem, which was then differentiated with respect to the design variables. The derivatives of the eigenvalues were obtained using the left and right eigenvectors and one of these efforts also considered second-order derivatives to better guide the optimization process and improve the rate of convergence [107].

Ringertz [109] applied the  $k$ -method to minimize the weight of a cantilevered wing in incompressible flow subject to flutter and divergence constraints. The structure was modeled as a composite FEM model, while the unsteady aerodynamic loads were computed using the doublet lattice method (DLM) [119, 120]. The eigenvalue problem was analytically differentiated in the modal space. A continuous flutter constraint was formulated using a boundary to constrain the damping values, which resulted in a large number of constraints. The method was demonstrated on a rectangular wing and on a swept wing with taper, where the objective was to minimize weight with respect to element group thicknesses subject to flutter and divergence constraints. In

both cases, a considerably lighter design was achieved.

Mallik et al. [110] investigated the impact of a previously developed flutter constraint [121] on the MDO of a TBW aircraft. The flutter speed was computed using the  $k$ -method applied to a linear pre-stressed structural model. The unsteady aerodynamic loads were computed using Theodorsen theory [122] with a Prandtl–Glauert compressibility correction. They implemented an iterative procedure to ensure consistency of the flutter speed, Mach number, and altitude, and optimized a representative TBW configuration for minimum takeoff weight and fuel burn using a genetic algorithm. The flutter constraint was formulated by enforcing a minimum flutter Mach number and was added to several other mission constraints. Comparing the optimization results with those obtained by removing the flutter constraint, they showed that this is necessary to obtain a flutter-free optimal solution.

Several authors have focused on including aerodynamic nonlinearities in flutter analysis to optimize transonic configurations. Stanford et al. [111] evaluated six different novel tailoring schemes employed in mass minimization optimization. They analyzed the flutter characteristics using the  $pk$  method and the commercial software ZTRAN [53] to retain aerodynamic nonlinearities. The nonlinear higher-fidelity Euler code, ZEUS, was used to compute steady background flow at multiple transonic Mach numbers for a fixed cruise shape. Using these steady-state CFD solutions as an input, the linearized unsteady loads were computed for a range of reduced frequencies using time-linearized transonic small disturbance (TSD) analyses about the equilibrium solutions.

The system damping values were forced to be under a stability boundary, similarly to the approach by Ringertz [109]. The transonic aerodynamic loads were computed offline before the optimization and their derivatives were obtained by differentiating the eigenvalue problem [123]. Flutter, stress, and buckling constraints computed in this study were all aggregated using a KS function [124]. They considered the



fixed-mode derivatives to improve computational efficiency, an approach that does not allow for shape changes.

They studied the undeflected Common Research Model (uCRM) wing [125] and obtained six different optimal wing structures corresponding to different tailoring schemes, all for the same operating condition and setup. The six tailoring schemes considered for the structural design were metallic thickness variations, functionally graded materials, balanced or unbalanced composite laminates, curvilinear tow steering, and distributed trailing edge control surfaces. While there was a structural wing mass reduction for every case, many of the lighter designs had an active flutter constraint, while the buckling constraint was active for the heavier cases.

Chen et al. [112] extended their previous work [126] by computing derivatives of flutter constraints with respect to shape variables using ZEUS coupled with a boundary layer code. The derivatives with respect to shape were computed using the complex-step approach [30, 33], which is numerically exact. The flutter constraint was formulated using the  $g$ -flutter method [85], which was differentiated analytically with respect to the design variables. The approach was verified for a cantilevered planform, similar to the F-5 geometry [127, 128], where a structure consisting of 10 spars, 10 ribs, and upper and lower skins was modeled using MSC Nastran. While no optimization results were presented, the flutter derivatives were verified against exact results.

Bartels and Stanford [113] proposed an approach to enforce a CFD-based flutter constraint for gradient-based structural optimization of transonic vehicles. The flutter analysis was performed as a standard eigenvalue analysis on the state-space representation of the aeroelastic system obtained via RFA [129]. The generalized aerodynamic force (GAF) matrix of the baseline structure was identified from time-linearized unsteady RANS simulations about nonlinear steady-state solutions [130]. The GAF matrix of the updated design was then computed by projecting the updated

modes onto the baseline modes. The methodology was used to minimize the uCRM mass subject to structural and aeroelastic constraints. The flutter was constrained by requiring that the real part of the system eigenvalues be below a bounding curve [104]. The optimization assumed a fixed-mode approximation. The authors compared the optimal solutions obtained using unsteady RANS and DLM aerodynamics in the optimization loop. They showed that the DLM-based solution was not conservative and had a significantly different thickness distribution compared to the CFD-based optimal design.

Opgenoord et al. [68] extended a low-order two-dimensional transonic flutter prediction model [67] to wings and implemented the model into a conceptual aircraft design tool to investigate the impact of geometric parameters and Mach number on the flutter boundary. Furthermore, they optimized the D8 configuration by minimizing the maximum take-off weight and fuel burn with and without a transonic flutter constraint. Enforcing the flutter constraints resulted in lower optimal aspect ratio and a weight penalty or lower fuel burn reduction compared to the case when the flutter constraint was omitted. Opgenoord et al. [69] also used the developed flutter model to optimize the internal lattice structure of a wing by minimizing weight with and without enforcing a flutter constraint in addition to stress and buckling constraints. The optimal design including the flutter constraint showed only a slight mass increase thanks to the appropriate aeroelastic tailoring of the lattice structure.

While several authors have performed flutter-constrained optimizations using nonlinear aerodynamic models, examples of flutter constraints considering nonlinear structures are more rare due to the more recent interest in optimizing very flexible aircraft. Variyar et al. [9] developed a framework for MDO of geometrically nonlinear aircraft subject to flutter constraints by coupling the SUAVE design tool [131] with the ASWING nonlinear aeroelastic solver [132]. They developed an interface to convert the arbitrary aircraft designs output by SUAVE into equivalent nonlinear beam

models for ASWING to drive structural and aeroelastic analyses and to post-process the results. The flutter speed was obtained iteratively by computing the eigenvalues of the statically deformed aircraft at different flight conditions.

This MDO framework was used to optimize the Sugar VOLT strut-braced aircraft [133] for minimum fuel burn subject to mission, maneuver, gust, and flutter constraints. The flutter constraint was implemented by imposing a minimum flutter speed, and derivatives with respect to the design variables were obtained via finite differences. The authors performed three MDO cycles by adding the maneuver, gust, and flutter constraints to the mission constraints. Despite having better performance, the optimal solution achieved with only maneuver and gust constraints experienced flutter within the flight envelope, highlighting the need for a flutter constraint in the design process.

Lupp and Cesnik [116] studied the effect of a flutter constraint including geometrical nonlinearities on the design of a BWB. They extended the University of Michigan's Nonlinear Aeroelastic Simulation Toolbox (UM/NAST) to utilize AD in reverse mode to determine coupled aeroelastic derivatives including geometrical nonlinearities. They proposed an algorithm to increase the computational efficiency of the gradient evaluation for a geometrically nonlinear aeroelastic analysis. The sample optimization formulation included a geometrically nonlinear beam-based flutter constraint using a KS aggregation [134] to obtain a scalar constraint for the entire flight envelope. The authors ran three fuel burn minimization cases: with a strength constraint, with a linear flutter constraint, and with a geometrically nonlinear flutter constraint with the wing chord distribution, wing box size, and wing box thickness as design variables. While the linear flutter constraint became active over the strength constraint, it was not conservative compared to the geometrically nonlinear constraint. They concluded that a geometrically nonlinear flutter constraint is needed for very flexible aircraft.

Xie et al. [114] used a flutter constraint to minimize the weight of a very flexible wind tunnel model. The purpose of this constraint was to ensure flutter within the wind tunnel speed range. The optimization problem coupled a geometrically nonlinear beam solver with a vortex lattice code for the static aeroelastic analysis, while a doublet lattice code was used to compute the unsteady loads. Since they use a gradient-free algorithm, no derivatives were computed. They compared optimization results based on linear and geometric nonlinear beam models subject to a flutter constraint. In addition to flutter, tip displacement and torsion angle constraints were also enforced. The linearly optimized configuration resulted in a wing lighter than the optimal solution obtained with the nonlinear flutter constraint. Furthermore, the flutter and displacements constraints were violated when the linear optimized configuration was analyzed considering geometric nonlinear effects. The difference in the results highlighted the need for a flutter constraint when optimizing very flexible vehicles and the importance of using nonlinear flutter prediction methods not only for analysis, but in design optimization as well.

Bhatia and Beran [115] developed a framework to optimize thermally stressed nonlinear structures subject to transonic flutter constraints. They showed that including aerothermoelastic static nonlinearities in the flutter analysis impacts the optimal design. These effects are particularly important for high-speed vehicles, which are subject to significant thermoelastic stresses when flying through the transonic regime during reentry [135]. The authors optimized a skin panel with respect to the thickness and density distributions to minimize the mass subject to a flutter constraint. The structure was modeled using a Timoshenko beam finite element with nonlinear von Krmn strain, while the transonic flow was solved via a finite-element discretization of the Euler equations. The structure was linearized around the static thermoelastic response and the aerodynamics was linearized around the background steady transonic flow past the baseline geometry. Flutter was analyzed in the frequency domain as an

iterative  $V$ - $g$  solution [3] considered the system linearized around the nonlinear equilibrium configuration for each operating condition. The flutter speed was constrained directly. The optimal solution obtained by linearizing around the thermally stressed equilibrium configuration showed a mass lower than the result using an unstressed analysis.

The Hopf-bifurcation method has also been applied in design optimization. Kennedy et al. [117] proposed a variant of previous bifurcation approaches [136, 137] to optimize an aeroelastic system subject to a flutter constraint, which was formulated in terms of flight speed rather than damping. The proposed method had the benefit of not requiring a search of the flutter point (which may be located well outside of the flight envelope) at each design iteration. The KS function [124] was used to smooth the effect of mode switching in the constraint value by aggregating the less damped modes, which yielded smooth gradients. The authors presented preliminary results for a medium fidelity three-dimensional aerodynamic panel code coupled with the structural finite-element code TACS [138]. Although no detailed optimization was performed, they performed a preliminary study on the uCRM benchmark [125].

Beran et al. [118] developed a fast adjoint method to compute derivatives of flutter points computed via the Hopf-bifurcation method for gradient-based MDO. The approach was applied to the highly flexible cantilevered wing studied by Tang and Dowell [139, 140]. Both geometric nonlinearities due to the structure and aerodynamic effects described using the ONERA stall model [140] were considered when computing the flutter point and its derivatives with respect to aerodynamic and structural design variables. While no optimization study was presented, the authors outlined the future work required to apply the methodology: validate the derivatives, assess the computational cost compared to alternate time- and frequency-domain flutter prediction methods, and develop the handling of mode switching to avoid discontinuous flutter points.

Some rare examples of optimizations with flutter constraints based on time-accurate analyses are also found in the literature, but they have been restricted to simple problems. In an early work, Holden [141] applied a collocation method to constrain the aeroelastic response envelope for a wing optimization problem. More recently, Mani and Mavriplis [97] were among the first to present a fully coupled unsteady adjoint for aeroelastic optimization. They demonstrated their method successfully in a shape optimization of two-dimensional airfoil section to suppress flutter, using a total of 32 design variables in the form of Hicks–Henne bump functions [142]. Later, Mishra et al. [98] extended previous work and presented the fully coupled unsteady adjoint for three-dimensional aeroelastic problems, which was demonstrated in a shape optimization of a flexible rotorcraft configuration.

Zhang et al. [99] developed a coupled adjoint method for unsteady aerostructural problems solved via time simulations. The method was applied to an airfoil shape optimization problem with the goal of suppressing flutter. The aerodynamics was computed with an Euler CFD code coupled with a boundary layer code to account for viscous effects. Both the continuous and discrete coupled adjoint were developed for steady-state analyses. The discrete approach proved more promising and only this version was developed for unsteady cases. A damping objective function was proposed that used a Hilbert transform [90] of the nonlinear unsteady time-history. To achieve the required flutter boundary, the damping objective function was minimized to obtain a neutral response, indicating the flutter point. The authors applied the methodology to the optimization of the two-dimensional (2D) Isogai airfoil [40, 41] to suppress flutter. Only derivatives with respect to shape variables were computed and the airfoil shape was parametrized using 48 Hicks–Henne bump functions [142]. A neutrally stable (zero-damping) configuration was obtained for a given flight condition.

Zhang et al. [100] extended their previous work [99], where the coupled adjoint

was developed for time-marching simulations. Two objective functions were used: one that maximized the flutter boundary and another that matched a given flutter boundary. To maximize the flutter boundary, they minimized the squared and time averaged history of the lift coefficient. To match a given flutter boundary, the damping value of a given time history was minimized to obtain a neutral response. B-spline curves were chosen as a parametrization method due the large number of Hicks–Henne functions that were previously needed. The authors presented steady-state optimization results for a 2D airfoil optimized to match a given pressure distribution and for the 3D ONERA M6 case [143], where the objective was a composite function of lift and drag. Configurations considered in the unsteady optimization consisted of the 2D Isogai airfoil and the Goland wing as modeled by Kurdi et al. [144]. For the unsteady 2D airfoil, two optimization cases were considered: flutter margin maximization and a flutter boundary matching (i.e., a neutral response for the given flight condition). The design variables were the plunge and pitch stiffness values. For the Goland wing, they optimized the aerodynamic shape to maximize the flutter speed with respect to 120 shape variables. A structural optimization of the Goland wing was also performed to maximize the flutter speed with respect to the skin thickness. No optimization was performed using simultaneously structural sizing and aerodynamic shape variables.

Kiviaho et al. [145] developed a flutter constraint using their previously developed unsteady aeroelastic framework with adjoint sensitives [101, 102]. The flutter constraint is based on a matrix pencil method [88] applied to a time-history, which estimates the damping based on most critical aeroelastic modes. Two methods are proposed, a direct flutter point evaluation which finds the flutter point where the dynamic pressure lower bound is specified as the design flight condition, and a flutter margin or clearance approach where the dynamic pressure times some tolerance is fixed. The direct method was demonstrated in a single design variable optimization of an airfoil where the dynamic pressure is minimized subject to the flutter constraint

in order to identify the flutter point.

### 1.4.3 Open Problems

Now that we reviewed methods for flutter prediction and their applications to optimization (see Table 1.1), we can summarize the open problems and challenges of integrating flutter constraints into aircraft design optimization.

Gradient-based optimization is the preferred choice for optimizations with respect to large numbers of design variables. When enforcing a flutter constraint in a gradient-based optimization, a serious challenge is varying structural and aerodynamic design variables simultaneously to optimize the aircraft external shape, planform, and internal sizing. Most of previous work optimized only structural sizing, and only a few efforts included airfoil shape and wing planform design variables as well. Furthermore, simplifying assumptions like the fixed-mode approximation were frequently used when computing derivatives to limit computational cost [113, 146, 147]. These assumptions are adequate for a structural optimization, but they can lead to inaccurate results when varying aerodynamic properties because this can cause significant changes in the mode shapes at each design iteration. Therefore, gradient-based optimizations with respect to structural, planform, and shape variables need approaches that take into account the derivatives of the mode shapes when computing the derivatives of the flutter constraints. Few examples of these approaches applied to simplified configurations or using lower-fidelity models are available in the literature [9, 116, 134]. However, they still have to be demonstrated on practical configurations parametrized by large number of structural and aerodynamic design variables.

A second major challenge is developing efficient flutter analysis models and methods applicable in the presence of aerodynamic or structural nonlinearities. In previous aircraft optimizations including linear flutter constraints, the natural choice was to analyze flutter in the frequency-domain due to the availability of well-established



and computationally efficient eigenvalue-analysis methods. The computational cost of these methods is a big challenge when including aerodynamic or structural nonlinearities because the flutter point depends on the equilibrium state. For each flight condition considered in the flutter search, the steady background flow solution (in the case of aerodynamic nonlinearities) or the coupled aerostructural equilibrium (in the case of both aerodynamic and structural nonlinearities) needs to be determined first. Next, the linearized model about each equilibrium point must be identified for computing the aeroelastic eigenvalues and determine at which point modal damping vanished. This process must be repeated for each flight condition, while linear methods analyze flutter by considering the undeformed configuration at zero angle of attack for each flutter search point. Moreover, in the presence of nonlinear effects multiple equilibrium points may also exist for each flight condition, which further increases complexity and computational cost. Stability must be analyzed about all equilibrium points, otherwise critical constraint values may be missed.

For large high-fidelity models with both structural and aerodynamic nonlinearities, computing the aerostructural equilibrium points and the corresponding linearized systems may be computationally prohibitive. For moderately flexible configurations, structural nonlinearities can be neglected, so eliminating the need to solve a nonlinear static aeroelastic problem at each flight condition. However, transonic aerodynamic nonlinearities still require identifying a linearized unsteady aerodynamic model for each steady background flow solution. Computing and retaining aerodynamic nonlinearities to accurately predict the flutter point in the transonic regime remains a big challenge due to the large computational cost associated with CFD. To address this problem, some efforts tried to preserve the computational efficiency of lower-fidelity methods while retaining the nonlinear physics modeled by the higher-fidelity CFD methods [53, 55]. Other works proposed time-linearized CFD solvers [56, 60] or reduced- or low-order models calibrated using CFD [65, 67]. Despite these progresses,

optimizing aircraft subject to transonic flutter constraints is still an open problem, particularly when using gradient-based methods and when seeking both structural sizing and aerodynamic shape changes.

Due to these challenges, previous work mainly optimized linear aircraft configurations or included only aerodynamic nonlinearities while limiting to structural sizing. Aerostructural optimizations considering both aerodynamic and geometric structural nonlinearities used low-order models or optimized simple configurations to limit the computational cost [9, 116]. No previous work considered both structural and aerodynamic nonlinearities in a high-fidelity MDO setting. Additionally, rigid-body DOFs were never included in the flutter constraints, which may lead to unfeasible designs for configurations prone to coupled rigid-elastic instabilities.

Few studies used alternatives to frequency-domain eigenvalue analysis methods for nonlinear flutter prediction, like the Hopf-bifurcation method or time-domain simulations. Such studies are rare because the high computational cost of time-accurate analyses makes them prohibitive to optimize complex configurations. For this reason, these applications were limited to simple configurations and frequently two-dimensional problems.

## 1.5 Thesis Objectives

In the previous section we reviewed the state-of-the-art in flutter analysis, its application in design optimization, and finally providing a summary of open challenges. In this work we focus on the first major challenge identified, developing a flutter constraint that is suitable for large scale high-fidelity aerostructural optimization, considering both structural and aerodynamic design variable.

We address the above in this work with the following intermediate milestones:

1. Develop a robust, efficient, and continuous flutter analysis methodology that

can be used to constrain flutter in a large scale high-fidelity aerostructural optimization, considering both structural sizing and aerodynamic design variables.

2. Compute derivatives of the proposed flutter constraint using methods that are accurate, efficient, and independent of the number of design variables. No simplifications or approximations, such as the fixed mode approximation, should be applied.
3. Demonstrate the proposed flutter constraint in a large scale high-fidelity aerostructural optimizations, and study its effect on representative wing geometries, demonstrating the need for such constraint.

## 1.6 Thesis Outline

To achieve the stated objectives, the multidisciplinary design optimization for aircraft configurations with high fidelity (MACH) framework is expanded by implementing several new components to enforce a flutter constraint that can be applied in high-fidelity design optimization.

A brief introduction of existing components in MACH framework that are used in this work are discussed in Chapter 2. To address objective 1, the flutter analysis methodology and constraint formulations is discussed in detail in Chapter 3. This includes theory, algorithms, and integration considerations necessary for a successful flutter constraint suitable for high-fidelity gradient based design. The subsequent chapter, Chapter 4, discusses the derivative strategy in order to compute the derivatives efficiently and accurately, addressing objective 2. In particular, we discuss algorithmic differentiation (AD) in reverse as well as analytic approaches that are suitable for AD, and verify the accuracy of the implementation.

Finally, objective 3 is addressed in a series of chapters, increasing in problem size and complexity, demonstrating the need for including a flutter constraint in high-

fidelity design. Chapter 5 demonstrates the constraint on an idealized wing (plate). A design space study is performed, demonstrating the continuity of the constraint that is finally demonstrated in an optimization without CFD. In Chapter 6 the proposed flutter constraint is applied and demonstrated using a rectangular subsonic wing. Two multipoint high-fidelity aerostructural optimizations are conducted, with and without the flutter constraint, maximizing the range. Finally, in Chapter 7 the flutter constraint is applied in a large scale multipoint aerostructural optimization on a full configuration aircraft, namely the XRF1. Here,  $\mathcal{O}(1000)$  design variables and constraints are applied demonstrating the capabilities of the constraint.

Chapter 8 concludes the thesis summarizing the key results and contributions. Suggestions for possible future work are then discussed. This includes discussion on improving the proposed methodology and capabilities, and including larger set of problems such as transonic aerodynamics and geometrically nonlinear effects.

## CHAPTER 2

# Existing Computational Tools

In this section, we briefly describe existing components from the multidisciplinary design optimization for aircraft configurations with high fidelity (MACH) framework [7] that are employed in static aerostructural optimization. An overview of the process is detailed in Fig. 2.1. This framework has been extensively used in aerodynamic [25, 148–153] and aerostructural [7, 8, 125, 154–156] optimization.

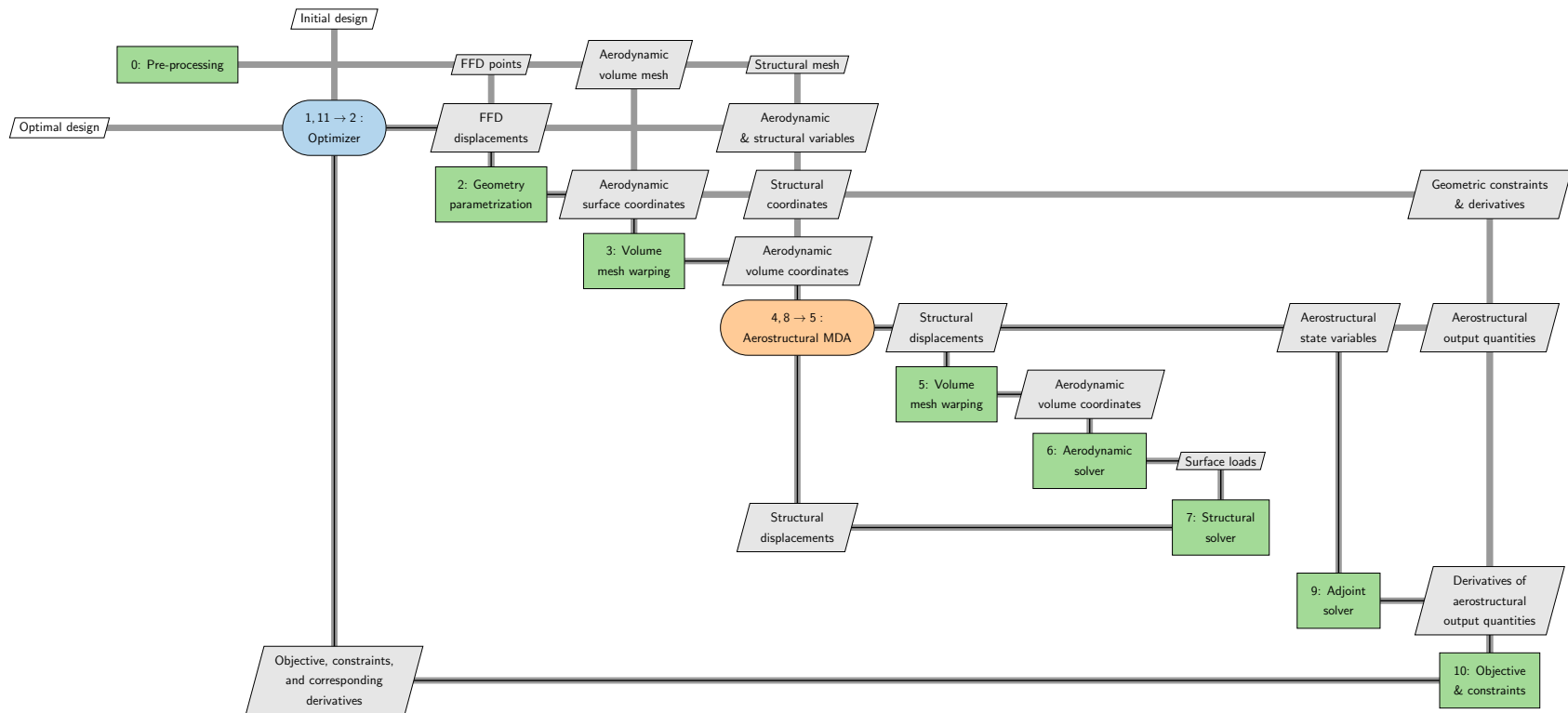


Figure 2.1: XDSM [1] overview of the aerostructural optimization process using MACH.

## 2.1 Geometric Parametrization

The geometric parametrization of the aerodynamic surfaces and structure is done using a free form deformation volume (FFD) approach [157], which has its origins in computer graphics applications [158].

Using this approach, the geometry of interest is embedded within an FFD volume that can be thought of as a flexible volume. The FFD volume is then deformed using a number of B-spline control points. As the control points are moved in space, the embedded coordinates are deformed in a continuous fashion. This method is well suited for aerostructural optimization because it provides a unified way of treating aerodynamic surfaces and the internal structure. Global design variables, such as span or sweep, are implemented by moving multiple points according to suitable functions. Airfoil cross-sectional shapes are controlled by individual B-spline control points.

## 2.2 CSM Solver

The computation structural mechanics (CSM) solver used in this work is the Toolkit for the Analysis of Composite Structures (TACS) [138].<sup>1</sup> TACS is an open-source parallel finite-element (FEM) solver that can handle poorly conditioned problems, which is common in the thin-walled structures found in transport aircraft. For such cases, the stiffness matrix condition numbers may exceed  $\mathcal{O}(10^9)$ , but through the use of a Schur-complement based parallel direct solver, TACS is able to effectively solve these poorly conditioned problems. Sensitivities of structural functions of interest with respect to structural and geometric design parameters are computed efficiently using the adjoint method [138]. The load and displacement transfer scheme used here is the rigid link approach [7], which follows the work of Brown [159]. In this approach, rigid links extrapolate the displacements from the structural surface to the

---

<sup>1</sup><https://github.com/gjkennedy/tacs>

aerodynamic surface. These rigid links, which are suitable for non-matching surfaces, are constructed between the aerodynamic surface mesh points and the points on the structural model lying closest to this set of points.

## 2.3 CFD Solver

In this work, we use the open-source CFD solver ADflow, a parallel, finite-volume, cell-centered, multi-block solver, which solves the Euler and the Reynolds averaged Navier–Stokes (RANS) equations in either steady, unsteady, or time spectral modes [35].<sup>2</sup> The mean flow equations and the one-equation Spalart–Allmaras turbulence model are solved in a coupled fashion by an approximate Newton–Krylov method [160]. A discrete adjoint method for the Euler and RANS equations is implemented within ADflow, enabling the efficient computation of gradients of functions of interest with respect to any number of design variables [35]. This solver has been extensively verified and validated against experimental results [161, 162].

## 2.4 Mesh Deformation

The mesh warping scheme used here is a robust inverse distance method, which has been used extensively in aerodynamic and aerostructural optimization [156, 163, 164]. Given an updated set of surface mesh nodes, the warping scheme uses an inexact explicit interpolation scheme to update the volume mesh nodes [165]. The derivative computation is implemented using the reverse mode AD method.

---

<sup>2</sup><https://github.com/mdolab/adflow>



## 2.5 Aerostructural Solver

The aerostructural solver facilitates the solution of the nonlinear aerostructural equations, and the linear adjoint equations. For the nonlinear system a block Gauss–Seidel (NLBGS) with Aitken acceleration scheme is employed. The adjoint system is solved using a coupled-Krylov (CK) method, which has been shown to be significantly faster than a commonly used linear block Gauss–Seidel (LBGS) approach [7]. The above solution approach has proven sufficiently robust for a range of aerostructural problems at various flight conditions [7, 8, 125, 154–156].

## 2.6 Optimizer

The optimization package used in this work is SNOPT (Sparse Nonlinear OPTimizer) [166]. SNOPT is a gradient-based optimizer that implements a sequential quadratic programming (SQP) algorithm. SNOPT uses an augmented Lagrangian merit function and the Hessian of the Lagrangian is approximated using a quasi-Newton approach. This optimizer is designed to perform well for optimization problems featuring many sparse nonlinear constraints. SNOPT is wrapped with pyOptSparse, a sparse implementation of pyOpt [167].<sup>3</sup>

---

<sup>3</sup><https://github.com/mdolab/pyoptsparse>

## CHAPTER 3

# Flutter Prediction and Constraint Formulation

As discussed in the introduction, the goal is to develop a flutter constraint that is suitable for high-fidelity aerostructural optimization. In this chapter, we describe the techniques and components necessary to enable such a flutter constraint. Figure 3.1 gives a high-level overview of the overall flutter analysis process developed in this work. Numerous new components are developed as part of this work. This includes the Lanczos eigenvalue solution method, doublet lattice method (DLM), flutter analysis, and constraint aggregation strategy suitable for gradient based optimization. All new components developed in this work are implemented in Python and Fortran with communication done efficiently in memory. In the following sections, we outline the characteristics of these components that are needed to enable flutter analysis and present a flutter constraint applicable to gradient-based optimization with structural and aerodynamic changes. Additionally, a component that handles communication between these components is also implemented. The methodology described here is independent of specific components or methods, i.e., different components can be used and substituted in, without any need for updating the formulation, nor the usage of the constraint in an aerostructural optimization. For example, to obtain higher fidelity generalized aerodynamic forces that capture transonic effects the aerodynamic component, in this case the DLM, can be replaced with CFD. This is due to

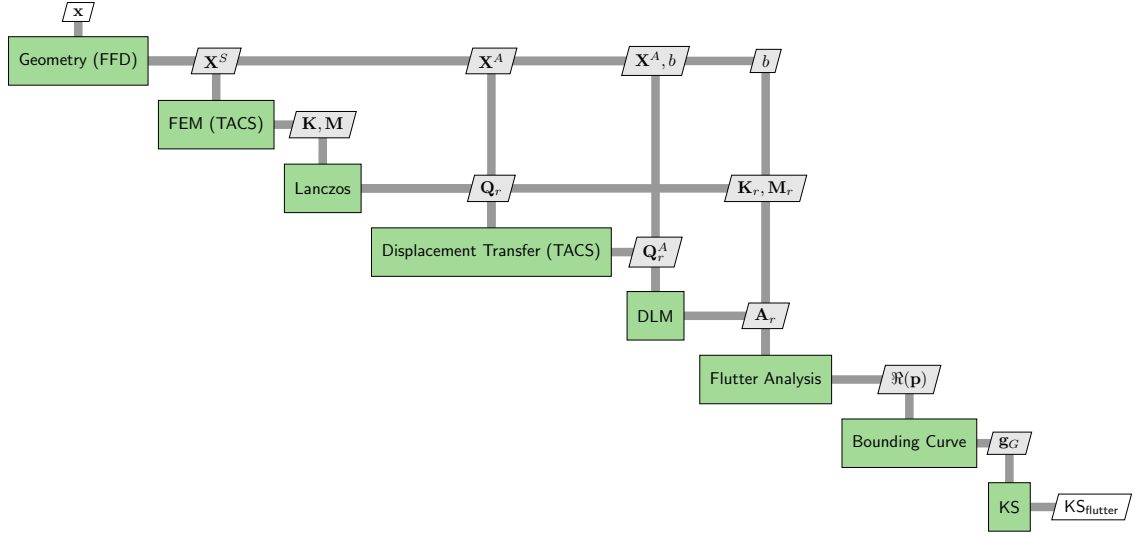


Figure 3.1: XDSM [1] of the proposed flutter analysis process and constraint formulation.

the fact that inputs and outputs have been clearly defined and identified. Similarly, derivatives needed are also defined.

The chapter is structured as follows, Section 3.1 discusses theory and equations used in this work to predict the flutter characteristics. To solve the flutter equations, a robust non-iterative solution strategy is introduced in Sections 3.2 and 3.3. Finally, Sections 3.4 and 3.5 describes the constraint formulation and aggregation strategy used which is suitable for high-fidelity gradient based optimization.

### 3.1 Flutter Analysis

Aeroelasticity consists of the interaction of aerodynamics, structures and inertial forces and are often depicted by the *Collars triangle* Fig. 3.2.

The discrete equation of motion for a generic linear aeroelastic system can be written as:

$$\mathbf{M}(\mathbf{x})\ddot{\mathbf{u}}(t) + \mathbf{C}(\mathbf{x})\dot{\mathbf{u}}(t) + \mathbf{K}(\mathbf{x})\mathbf{u}(t) = \mathbf{F}(\mathbf{x}, t), \quad (3.1)$$

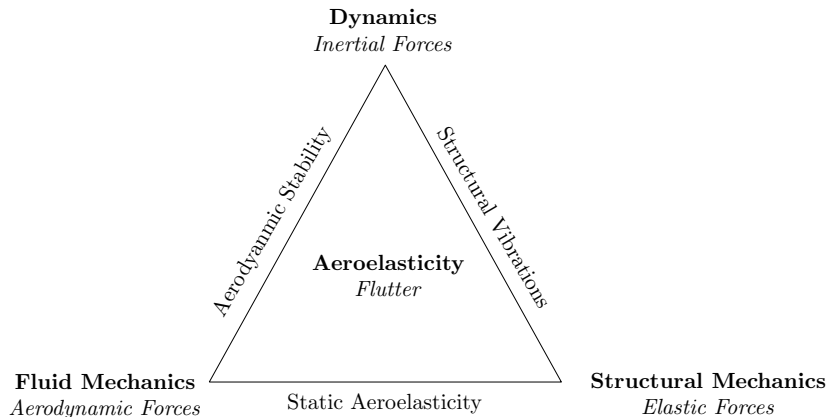


Figure 3.2: Collars triangle.

where  $N_s$  is the number of structural DOFs,  $\mathbf{M}(\mathbf{x}), \mathbf{C}(\mathbf{x}), \mathbf{K}(\mathbf{x}) \in \mathbb{R}^{N_s \times N_s}$  are the mass, damping, and stiffness matrices resulting from the finite-element discretization and are a function of the design variables  $\mathbf{x} \in \mathbb{R}^{N_x}$ .  $\mathbf{u}(t) \in \mathbb{R}^{N_s}$  is the displacement vector including all degrees of freedom of the system, translations and rotations. In general the forces  $\mathbf{F}(\mathbf{x}, t) \in \mathbb{R}^{N_s}$  can be split into external forces  $\mathbf{F}_{ext}(\mathbf{x}, t)$ , which we assume to be zero in this work, and the unsteady aerodynamic forces  $\mathbf{F}_{aero}(\mathbf{u}(t))$ . The equations of motions can then be written as,

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) - \mathbf{F}_{aero}(t) = 0. \quad (3.2)$$

For the sake of conciseness, Eq. (3.2) omits the dependency of the structural matrices and aerodynamic loads on the design variables  $\mathbf{x}$ . Therefore, flutter analysis methods are presented for a fixed design and all the structural and aerodynamic quantities are updated at each optimization step based on the current values of the design variables.

### 3.1.1 Load and Displacement Transfer

The aerodynamic forces presented Eq. (3.2) are given in the structural degrees of freedom. An aeroelastic interface is required that is capable of transferring aerodynamic loads and then structural displacement using a suitable interpolation strategy.

Furthermore, in practice, the aerodynamic mesh does typically not coincide with the structural mesh. For the displacement transfer, the structural DOF  $\mathbf{u}$  can be related to the aerodynamic DOF  $\mathbf{h} \in \mathbb{R}^{N_a}$  by

$$\mathbf{h} = \mathbf{T}\mathbf{u}, \quad (3.3)$$

where  $\mathbf{T} \in \mathbb{R}^{N_a \times N_s}$ , is a linear operator, and a displacement transfer interpolation matrix and  $N_a$  is the number of aerodynamic DOF.

The aerodynamic forces on the aerodynamic mesh,  $\mathbf{F}_a \in \mathbb{C}^{N_a}$ , can be interpolated using

$$\mathbf{F}_{aero} = \mathbf{T}^T \mathbf{F}_a \quad (3.4)$$

Equation (3.4) can be derived using the principle of virtual work. By imposing the equivalence of the virtual work done by the aerodynamic forces  $\mathbf{F}_a$  and the equivalent structural forces we can write,

$$\delta \mathcal{L} = \delta \mathbf{h}^T \mathbf{F}_a = \delta \mathbf{u}^T \mathbf{F}_{aero}, \quad (3.5)$$

where  $\delta \mathbf{h}, \delta \mathbf{u}$  are virtual displacements. By substituting Eq. (3.3) into Eq. (3.5) we obtain

$$\delta \mathbf{u}^T (\mathbf{T}^T \mathbf{F}_a - \mathbf{F}_{aero}) = 0. \quad (3.6)$$

Since the virtual deformations  $\delta \mathbf{u} \neq 0$  are arbitrary, the terms inside the parenthesis must equal to zero, therefore giving Eq. (3.4).

### 3.1.2 Generalized Structural Matrices

Due to the sheer number of structural DOFs, it is common to reduce the computational effort by rewriting Eq. (3.2) in terms of a reduced set of  $N_r$  generalized coor-

dinates (modal amplitudes) where  $N_r \ll N_s$ . This is referred as a modal approach. The displacement field can be then approximated by

$$\mathbf{u}(x, y, z; t) \approx \mathbf{Q}_r(x, y, z)\mathbf{q}(t) , \quad (3.7)$$

where  $\mathbf{q} \in \mathbb{R}^{N_r}$  is the vector of retained generalized coordinates and  $\mathbf{Q}_r \in \mathbb{R}^{N_s \times N_r}$  is a matrix whose columns contain the corresponding eigenvectors (mode shapes).

The natural mode shapes are obtained by setting any loads to zero, assume harmonic motion substituting  $\mathbf{u} = \bar{\mathbf{u}}e^{i\omega t}$  in Eq. (3.2) and solve the following eigenvalue problem,

$$[\mathbf{K} - \omega_i^2\mathbf{M}] \bar{\mathbf{u}}_i = 0, \quad (3.8)$$

where  $\omega_i$  is the natural frequency.

To obtain the lowest natural modes and mode shapes, Eq. (3.8) is solved using a shift-and-invert Lanczos method [168]. The Lanczos algorithm, discussed in Appendix B.1, uses an  $\mathbf{M}$ -orthonormal subspace, written as  $\mathbf{V}_m \in \mathbb{R}^{N_s \times N_m}$ , such that  $\mathbf{V}_m^T\mathbf{M}\mathbf{V}_m = \mathbf{I}_m$ , where  $N_s$  is the size of the square mass and stiffness matrices, and  $N_m$  the size of the subspace chosen. We use an expensive, but effective, full-orthonormalization procedure (Gram–Schmidt) that enforces  $\mathbf{M}$ -orthonormality. The Lanczos implementation is shown to produce good approximation and has been verified against the commercial software MSC/Nastran [169].

The eigenvectors (mode shapes)  $\bar{\mathbf{u}}_i$  are then computed from the constructed subspace  $\mathbf{V}_m$ . The  $\bar{\mathbf{u}}_i$  for  $i = 1, \dots, N_r < N_m$  are collected in the matrix  $\mathbf{Q}_r \in \mathbb{R}^{N_s \times N_r}$ ,

$$\mathbf{Q}_r = \begin{bmatrix} | & | & & | \\ \bar{\mathbf{u}}_1 & \bar{\mathbf{u}}_2 & \cdots & \bar{\mathbf{u}}_{N_r} \\ | & | & & | \end{bmatrix} \quad (3.9)$$

where  $N_r$  is the number of column vectors in the reduced mode shape matrix. These eigenvectors are  $\mathbf{M}$ -orthonormal, such that  $\mathbf{Q}_r^T \mathbf{M} \mathbf{Q}_r = \mathbf{I}_r$ . To obtain good approximations for the first  $N_r$  number of natural modes and mode shapes, the subspace size  $N_m$  is chosen to be at least  $N_m > 2N_r$ .

Substituting Eq. (3.7) into Eq. (3.2) and pre-multiplying by  $\mathbf{Q}_r^T$  we obtain the EoM in a generalized form,

$$\mathbf{M}_r \ddot{\mathbf{q}}(t) + \mathbf{C}_r \dot{\mathbf{q}}(t) + \mathbf{K}_r \mathbf{q}(t) - \mathbf{Q}_r^T \mathbf{F}_{\text{aero}}(t) = 0. \quad (3.10)$$

The eigenvectors are  $\mathbf{M}$ -orthonormal, such that  $\mathbf{Q}_r^T \mathbf{M} \mathbf{Q}_r = \mathbf{I}_r$ . The reduced generalized mass, damping, and stiffness matrices take the form:

$$\begin{aligned} \mathbf{M}_r &= \mathbf{Q}_r^T \mathbf{M} \mathbf{Q}_r = \mathbf{I}_r \in \mathbb{R}^{N_r \times N_r}, \\ \mathbf{C}_r &= \mathbf{Q}_r^T \mathbf{C} \mathbf{Q}_r \in \mathbb{R}^{N_r \times N_r}, \\ \mathbf{K}_r &= \mathbf{Q}_r^T \mathbf{K} \mathbf{Q}_r = \text{diag}\{\omega_i^2\} \in \mathbb{R}^{N_r \times N_r} \end{aligned} \quad (3.11)$$

The final expression of  $\mathbf{C}_r$  depends on how the damping matrix is constructed.

### 3.1.3 Generalized Aerodynamic Loads

From Eq. (3.10) the generalized aerodynamic forces (GAFs) on the structure are written as

$$\mathbf{F}_{\text{aero},r} = \mathbf{Q}_r^T \mathbf{F}_{\text{aero}} \in \mathbb{R}^{N_r} \quad (3.12)$$

or in terms of the scalar modal forces for the  $i$ -th mode

$$F_{\text{aero},r,i} = \bar{\mathbf{u}}_i \cdot \mathbf{F}_{\text{aero}} \quad (3.13)$$

Using the pressure coefficient  $C_p$  and the dynamic pressure  $q_\infty$ , the forces in the

time-domain can be obtained by integrating the surface area, scaled by a mode shape

$$F_{\text{aero},r,i} = q_\infty \int_S C_p(t) \bar{\mathbf{u}}_i \cdot d\mathbf{S} \quad (3.14)$$

By assuming we have a linear time invariant system (LTI) then an arbitrary input and output signal can be described by the convolution integral,

$$\mathbf{F}_{\text{aero},r} = q_\infty \int_0^t \mathbf{H}(t - \tau) \mathbf{q}(\tau) d\tau = q_\infty (\mathbf{H} * \mathbf{q})(t), \quad (3.15)$$

where the  $\mathbf{H}$  is a transfer matrix and is the system response due to an input impulse. This is under the assumptions that linear relations describe the generalized aerodynamic forces due to a deflection in one of the mode shapes. Now transforming into the frequency domain we obtain,

$$\mathbf{F}_{\text{aero},r}(s) = q_\infty \mathbf{H}(s) \bar{\mathbf{q}}(s) \quad (3.16)$$

where  $\mathbf{H}(s)$  is aerodynamic transfer function and  $\bar{\mathbf{q}}(s)$  is the Laplace transform of  $\mathbf{q}(t)$  and  $s$  is the Laplace variable.

### 3.1.3.1 Aerodynamic Loads

The aerodynamic forces on the aerodynamic mesh can be written as

$$\mathbf{F}_a = q_\infty \mathbf{A}_{\text{IC}}(ik) \mathbf{h}, \quad (3.17)$$

where  $q_\infty$  is the dynamic pressure,  $\mathbf{A}_{\text{IC}}(ik) \in \mathbb{C}^{N_a \times N_a}$  is the aerodynamic influence coefficient load matrix, where  $k = \omega b/U$  is the reduced frequency,  $\omega$  is angular frequency,  $b$  is the reference semi-chord, and  $U$  is the freestream velocity. Combing



equations Eqs. (3.4) and (3.17), the forces in the structural DoFs become,

$$\mathbf{F}_{\text{aero}} = q_{\infty} \mathbf{T}^T \mathbf{A}_{\text{IC}}(ik) \mathbf{T} \mathbf{u}. \quad (3.18)$$

Pre-multiplying the mode shapes,  $\mathbf{Q}_r^T$ , and using Eq. (3.7) we obtain the reduced generalized forces

$$\mathbf{F}_{\text{aero,gen}}(ik) = \mathbf{Q}_r^T \mathbf{F}_{\text{aero}} \quad (3.19)$$

$$= q_{\infty} \mathbf{Q}_r^T \mathbf{T}^T \mathbf{A}_{\text{IC}}(ik) \mathbf{T} \mathbf{Q}_r \mathbf{q} \quad (3.20)$$

$$= q_{\infty} \mathbf{A}_r(ik) \mathbf{q} \quad (3.21)$$

where we have defined

$$\mathbf{A}_r(ik) = \mathbf{Q}_r^T \mathbf{T}^T \mathbf{A}_{\text{IC}}(ik) \mathbf{T} \mathbf{Q}_r \in \mathbb{C}^{N_r \times N_r}. \quad (3.22)$$

The aerodynamic mode shapes obtained by interpolating the structural mode shapes using

$$\mathbf{Q}_{ra} = \mathbf{T} \mathbf{Q}_r \in \mathbb{R}^{N_a \times N_r}. \quad (3.23)$$

and thus we can write the generalized forces matrix in terms of the aerodynamic mode shapes as

$$\mathbf{A}_r(ik) = \mathbf{Q}_{ra}^T \mathbf{A}_{\text{IC}}(ik) \mathbf{Q}_{ra} \quad (3.24)$$

### 3.1.3.2 Aerodynamic Loads - DLM

In this work, we use the doublet-lattice method (DLM) [119, 120] to generate the unsteady aerodynamic loads. It consists of a lifting surface method that is formulated in the frequency domain. A substantial body of literature exists on the DLM. An excellent reference worth mentioning is the work done by Blair [170]. The DLM has been widely adopted in the aeroelastic community and has been a valuable tool for

the flutter analysis of subsonic aircraft. Commercial software tools such as Nastran have adopted the DLM [171]. The implementation is based in part on the method of Albano and Rodden [119], and the extension by Rodden *et al.* [120]. For more details we refer interested readers to the Appendix A.

Using the Doublet-Lattice Method (DLM) the unsteady pressure coefficient can be computed as follows,

$$\mathbf{C}_p = \frac{\Delta \mathbf{p}}{q_\infty} = \mathbf{D}^{-1} \bar{\mathbf{w}} \in \mathbb{C}^{N_a} \quad (3.25)$$

where the  $\mathbf{D}^{-1} \in \mathbb{C}^{N_a \times N_a}$  is the inverted aerodynamic influence coefficient (AIC) matrix computed by the DLM, and  $\mathbf{w} \in \mathbb{C}^{N_a}$  is the normalwash.  $N_a$  is the number of panels used in the aerodynamic mesh. Recalling from Appendix A.1.4 and repeating Eq. (A.22) in vector form the normalwash in the frequency domain is written as

$$\bar{\mathbf{w}} = i\hat{k}\bar{\mathbf{h}} + \bar{\mathbf{h}}_x \quad (3.26)$$

The displacement  $\bar{\mathbf{h}}$  is then written in terms of generalized coordinates as

$$\bar{\mathbf{h}} = \mathbf{T}\bar{\mathbf{u}} \quad (3.27)$$

$$= \mathbf{T}\mathbf{Q}_r\bar{\mathbf{q}} \quad (3.28)$$

$$= \mathbf{Q}_{ra}\bar{\mathbf{q}} \quad (3.29)$$

where we have used Eqs. (3.3), (3.7) and (3.23). Substituting we obtain

$$\mathbf{w} = \left( i\hat{k}\mathbf{Q}_{ra} + \frac{\partial \mathbf{Q}_{ra}}{\partial x} \right) \bar{\mathbf{q}} \quad (3.30)$$

The aerodynamic forces can then be written as

$$\mathbf{F}_a = q_\infty \mathbf{S} \mathbf{C}_p \quad (3.31)$$

$$= q_\infty \mathbf{S} \mathbf{D}^{-1} \mathbf{w} \quad (3.32)$$

$$= q_\infty \mathbf{S} \mathbf{D}^{-1} \left( i\hat{k} \mathbf{Q}_{ra} + \frac{\partial \mathbf{Q}_{ra}}{\partial x} \right) \bar{\mathbf{q}} \quad (3.33)$$

where  $\mathbf{S} \in \mathbb{R}^{N_a \times N_a}$  is the area of individual panels and distributes the forces from the collocation point to the nodes of the panel. The generalized forces matrix is then

$$\mathbf{A}_r(ik) = \mathbf{Q}_{ra}^T \mathbf{S} \mathbf{D}^{-1} \left( i\hat{k} \mathbf{Q}_{ra} + \frac{\partial \mathbf{Q}_{ra}}{\partial x} \right) \quad (3.34)$$

where the generalized forces are written as

$$\mathbf{F}_{\text{aero},r}(ik) = q_\infty \mathbf{A}_r(ik) \bar{\mathbf{q}} \quad (3.35)$$

### 3.1.4 Generalized Equations of Motion

All matrices have now been defined in generalized coordinates using a reduced set of mode shapes. Taking the Laplace transform of the equations of motion Eq. (3.10) yields,

$$\left[ s^2 \mathbf{M}_r + s \mathbf{C}_r + \mathbf{K}_r - q_\infty \mathbf{A}_r(s) \right] \bar{\mathbf{q}}(s) = 0, \quad (3.36)$$

by assuming motion of  $\mathbf{q}(t) = \bar{\mathbf{q}} e^{st}$  where  $s = \gamma + i\omega$ . This equation can also be written in terms of non-dimensional Laplace parameter  $p = sb/U = g + ik$ ,

$$\left[ \left( \frac{U}{b} \right)^2 p^2 \mathbf{M}_r + \frac{U}{b} p \mathbf{C}_r + \mathbf{K}_r - q_\infty \mathbf{A}_r(p) \right] \bar{\mathbf{q}}(p) = 0, \quad (3.37)$$

where,  $b$  is the reference half chord,  $g$  is the nondimensional damping, and  $k$  is the reduced frequency [81].

Equation (3.37) requires computing the GAF in the Laplace domain. However, the GAF is typically given as a transcendental function,  $\mathbf{A}_r(ik)$ , of the reduced frequency. Two approaches are used for overcoming this problem [12]. One approach is to approximate  $\mathbf{A}_r(p) \approx \mathbf{A}_r(ik)$  and solve the flutter equation, Eq. (3.36), or an equivalent form by computing the GAF matrix in the reduced frequency domain while enforcing the condition  $\Im(p) = k$ . An alternate approach is to obtain a rational function approximation (RFA) of  $\mathbf{A}_r(ik)$  and use analytic continuation [172] to extend its domain from the imaginary axis (reduced frequency) to the entire complex plane (nondimensional Laplace variable). The aeroelastic system can be recast in state-space form by introducing additional aerodynamic states, such that flutter can be analyzed using a standard eigenvalue analysis.

When the GAF is represented as a transcendental function of  $k$ , flutter analysis is performed using iterative or non-iterative methods that either compute the true damping only at the flutter point ( $k$ -method) or at all flight speeds or dynamic pressure values (root locus,  $p$ -,  $pk$ -, and  $g$ -methods) [81–83, 85, 173]. Here, we use the  $pk$ -method, assuming purely oscillatory aerodynamic forces  $\mathbf{A}_r(p) \approx \mathbf{A}_r(ik)$ , then Eq. (3.37) is rewritten as [81],

$$\left[ \left( \frac{U}{b} \right)^2 p^2 \mathbf{M}_r + \frac{U}{b} p \mathbf{C}_r + \mathbf{K}_r - q_\infty \mathbf{A}_r(ik) \right] \bar{\mathbf{q}}(p) = 0 \quad (3.38)$$

$$\mathbf{F}_r(p, k) \bar{\mathbf{q}}(p) = 0 .$$

This is a second-order nonlinear eigenvalue problem, where the nonlinearity stems from the dependency of  $\mathbf{A}_r$  on the imaginary part of  $p$ .

Several decompositions of the GAF  $\mathbf{A}_r$  in Eq. (3.38) exist in the literature. Stanford [174] summarized these decompositions and showed that they predict the same

flutter speed, but that the mode migration and characteristics may be very different. This directly impacts the optimization, resulting in different optimal designs.

The GAF are split into its real and imaginary parts as  $\mathbf{A}_r = \mathbf{A}_r^R + i\mathbf{A}_r^I$  order to improve the approximation of the damping. Assuming small damping,  $p/(ik) \approx 1$  [169], the forces can be written as,

$$\mathbf{A}_r = \mathbf{A}_r^R + p/k\mathbf{A}_r^I. \quad (3.39)$$

Finally, Eq. (3.38) can be rewritten in first-order form as the generalized nonlinear eigenvalue problem [82, 134, 147],

$$p \begin{bmatrix} \mathbf{I}_r & 0 \\ 0 & (\frac{U}{b})^2 \mathbf{M}_r \end{bmatrix} \begin{Bmatrix} \bar{\mathbf{q}} \\ p\bar{\mathbf{q}} \end{Bmatrix} - \begin{bmatrix} 0 & \mathbf{I}_r \\ -(\mathbf{K}_r - q_\infty \mathbf{A}_r^R) & -(\frac{U}{b} \mathbf{C} - \frac{q_\infty}{k} \mathbf{A}_r^I) \end{bmatrix} \begin{Bmatrix} \bar{\mathbf{q}} \\ p\bar{\mathbf{q}} \end{Bmatrix} = 0, \quad (3.40)$$

where  $\mathbf{I}_r \in \mathbb{R}^{N_r \times N_r}$  is an identity matrix. All matrices are real but due to the non-symmetric nature of the aerodynamic loads, the eigenvalues and eigenvectors are complex. The real part of the eigenvalue,  $p$ , dictates the stability of the system, where a positive value corresponds to an unstable system, a zero value represents a neutrally stable system, and a negative value represents a stable one.

The matrix  $\mathbf{A}_r$  is dense in general, while  $\mathbf{M}_r$  and  $\mathbf{K}_r$  are diagonal. The structural viscous damping can be approximated by Rayleigh damping,  $\mathbf{C}_r = \alpha\mathbf{M}_r + \beta\mathbf{K}_r$  [175], as proposed by Stanford and Dunning [146], but is omitted here for simplicity. While rigid-body modes are not considered in this study, the above system of equations could be extended without affecting the proposed flutter aggregation strategy [2].

The reduced generalized eigenvalue problem, Eq. (3.40), of size  $2N_r \times 2N_r$  is solved with LAPACK [176]. However, valid roots need to satisfy the equivalence  $\Im(p) = k$ . A robust flutter solution strategy is discussed in the next section.

Note that the computational cost scales with double the number of mode shapes, i.e., the eigenvalue problem size is doubled. Industrial applications for the analysis of full configuration aircraft may consider up to 100 natural modes, thus placing an upper bound on the problem size. While LAPACK is capable of solving such problems, parallel eigenvalue solution methods, such as SLEPc [?] could be considered to improve performance.

## 3.2 Non-iterative Flutter Solution Method

In flutter analysis, when solving Eq. (3.40) (or Eq. (3.38)), iterative procedures are usually applied because the aerodynamic matrix depends on the reduced frequency,  $k$ , the imaginary part of the eigenvalue,  $p$ . Valid roots need to satisfy the equivalence  $\Im(p) = k$ .

One such procedure is the determinant iteration proposed by Hassig [81]. This method is a secant method applied to the determinant of Eq. (3.38),

$$\Delta(p) = \det(\mathbf{F}_r(p, k)). \quad (3.41)$$

Given initial guesses  $p_1$ , and  $p_2$ , the method computes  $p_{k+2}$  as follows,

$$p_{k+2} = \frac{p_{k+1}\Delta(p_k) - p_k\Delta(p_{k+1})}{\Delta(p_k) - \Delta(p_{k+1})}. \quad (3.42)$$

The iteration is continued until  $|\Delta(p_{k+2})| \leq \epsilon_{tol}$  for some specified tolerance.

Another popular method is found in commercial software such as MSC/Nastran [169, 171], where the eigenvalue problem is solved based on an assumed reduced frequency  $k_i$ . The resulting computed eigenvalue,  $p(k_i)$ , for the mode under study is identified and its imaginary part is compared to the assumed reduced frequency. If the difference,  $|\Im(p(k_i)) - k_i| \leq \epsilon_{tol}$ , exceeds some predefined tolerance, the iteration

continues using the imaginary part of the new eigenvalue as the reduced frequency,  $k_{i+1} = \Im(p(k_i))$ , computing new aerodynamic loads repeating the process.

Iterative methods however may in general experience convergence issues, such as slow convergence rate, non convergence, and convergence to incorrect values. For the basic iterative algorithm,  $k_{i+1} = \Im(p(k_i))$ , to converge locally, the following must hold,

$$\left| \frac{d\Im(p)}{dk}(k^*) \right| < 1, \quad (3.43)$$

where  $k^*$  is solution to  $k - \Im(p) = 0$  [177]. It can be shown that the asymptotic convergence rate is at best linear. Convergence rate is slow for magnitudes close to one, requiring numerous small dynamic pressure increments. For values greater than one, it does not converge. Further, if the aeroelastic eigenvalue changes rapidly with dynamic pressure, the convergence rate can be slow because of a large number of dynamic pressure increments. In addition, an incorrect eigenvalue can be picked up if two eigenvalues are close to each other in frequency. This results in *mode hopping*, such that a discontinuity appears in the damping, which is detrimental for gradient-based optimization.

Basic iterative methods do not distinguish aerodynamic lag roots from structural modes. If an aerodynamic lag root becomes unstable, these methods may converge to the lag root over the structural mode or not converge at all. Further, neither of the aforementioned iterative methods are able to add or remove aerodynamic roots. A more sophisticated and robust root finding method is thus needed for integrating the flutter process in an optimization process. In addition, a robust mode tracking algorithm is also needed to avoid mode hopping.

To improve the convergence rate of flutter analysis, a Newton method with safeguards was proposed by Back and Ringertz [177]. While such method offers superior convergence rate, it may still suffer from poor initial conditions and failure to converge. In contrast, a non-iterative method is not subject to such limitations. During

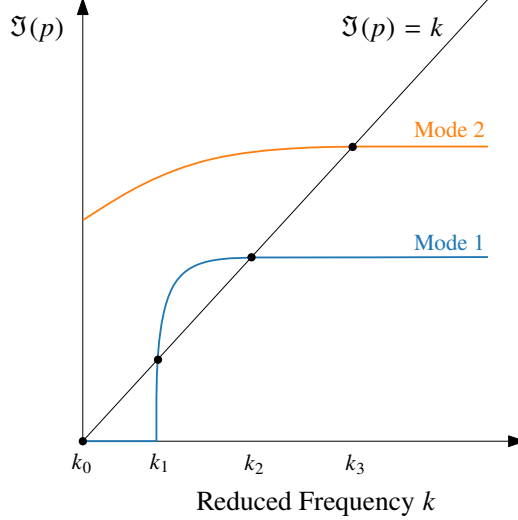


Figure 3.3: Hypothetical system with two modes. Black dots represent a valid solution, i.e., where the modes intersect the diagonal line,  $\Im(p) = k$ , depicted in black.

gradient-based optimization a wide range of designs and operating conditions may be analyzed. Thus, for a successful optimization robust convergence properties of the analysis methods is vital. In addition, it is important that the objective and constraint functions be continuous and smooth. Several methods have been proposed in the literature [85, 178, 179]. The proposed flutter solution method used in this work aims to satisfy these requirements.

In this work we implement a similar non-iterative method as proposed by van Zyl [179]. The method is as follows. At each dynamic pressure increment,  $q_i$ , Eq. (3.40) is solved for a range of reduced frequencies,  $k$ . The eigenvalues are valid roots of the flutter equation if the imaginary part of the eigenvalue equals the assumed  $k$  value, i.e., a matched point solution where  $\Im(p) = k$ . A change in sign of the difference  $\Im(p) - k$  thus signifies the existence of a valid root. The root is determined by a linear interpolation. This non-iterative method places no restriction on the number of roots that can be found.

For a hypothetical system with two modes, Fig. 3.3 shows qualitatively the reduced frequency sweep for a single dynamic pressure  $q_i$ . The black dots represent



an intersection of a mode with the black diagonal line,  $\Im(p) = k$ , which is found by linear interpolation. There are five valid roots for this particular system, four from the first mode and one from second mode. For the first mode, there are two real roots at  $k = k_0 = 0$ , and two complex roots at  $k = k_1$  and  $k = k_2$ . The second mode is complex throughout the reduced frequency sweep, and has only one valid root at  $k = k_3$ . An iterative method would have issues converging to  $k = k_1$  because Eq. (3.43) is not satisfied and would converge to  $k = k_2$ . Similar behavior is noted by Rodden and Bellinger [180] where the iterative method converges to a real root when it should converge to an oscillatory complex root.

### 3.3 Mode Tracking

Mode tracking is an important component in the success of a  $p$ - $k$ -type flutter analysis. The primary function of this tracking is to provide a correlation functionality between two consecutive iterations during mode migration (e.g. with increasing dynamic pressure), and reduced frequency sweep, to prevent mode hopping. Such tracking is typically performed by utilizing the mode shapes (eigenvectors) computed.

#### 3.3.1 Existing Methods

Several mode tracking methods have been proposed and one of the early and popular methods is the modal assurance criterion (MAC) proposed by Allemang and Brown [181]. Later, due to its popularity, Allemang [182] discusses its original development and various extensions to the method. The general form of the MAC correlation metric for real and complex vectors is:

$$C_{ij} = \frac{|\bar{\mathbf{q}}_i^* \bar{\mathbf{q}}_j|^2}{(\bar{\mathbf{q}}_i^* \bar{\mathbf{q}}_j)(\bar{\mathbf{q}}_i^* \bar{\mathbf{q}}_j)}, \quad (3.44)$$

where  $\bar{\mathbf{q}}_i, \bar{\mathbf{q}}_j$  are the  $i$ -th previous and  $j$ -th current eigenvectors, and  $*$  represents the Hermitian for complex numbers and transpose for real numbers. Equation (3.44) computes the  $C_{ij}$  entry in the correlation matrix  $\mathbf{C}$ .

van Zyl [183] proposed a similar version correlating modes based on complex inner products between current and previous eigenvalues. The scalar product of two complex eigenvector must be defined to be independent of scaling and phase. A definition that satisfies these conditions is

$$\mathbf{X} \cdot \mathbf{Y} = \frac{\sqrt{S_1^2 + S_2^2}}{\sqrt{S_3 S_4}} = C_{ij} \quad (3.45)$$

where

$$S_1 = \sum_{k=1}^n \Re(X_k)\Re(Y_k) - \Im(X_k)\Im(Y_k) \quad (3.46)$$

$$S_2 = \sum_{k=1}^n \Re(X_k)\Im(Y_k) - \Im(X_k)\Re(Y_k) \quad (3.47)$$

$$S_3 = \sum_{k=1}^n \|X_k\|^2 \quad (3.48)$$

$$S_4 = \sum_{k=1}^n \|Y_k\|^2 \quad (3.49)$$

Using this method a full correlation matrix  $\mathbf{C}$  is constructed between two sets of mode shapes from the previous and current iteration. The matrix is then searched for the largest elements in which its position then relates previous iteration (line) with current iteration (column). Once a mode has been selected the corresponding line and column are excluded from the selection process and this process continues until all the current modes have been correlated with the previous set of modes.

Eldred et al. [184] proposed a mode tracking method applicable for real eigenvalue problems, demonstrated in structural optimization with free vibration constraints. Later Eldred et al. [185] proposed two mode tracking methods for complex eigenvalue

problems and demonstrated its usefulness in flutter analysis.

Other mode tracking methods exist that do not use the eigenvector information for eigenvalue tracking. One such method proposed by Chen [85] is a predictor-corrector scheme for the eigenvalue tracking in the  $g$  method [85]. This method uses a linear extrapolation to predict the new eigenvalue using the current eigenvalue and gradient. Another class of methods builds on the piecewise quadratic interpolation (PQI) method proposed by Eller [186]. Huang et al. [187] later improved the method using shape-preserving cubic spline for mode tracking that uses available data to compute polynomial coefficients.

However, many of these methods require small increments to successfully track modes. Furthermore, if multiple modes interact within a small frequency range, methods utilizing spline or interpolation techniques may have difficulties tracking the modes. More recently, Hang et al. [188] proposed a mode tracking technique utilizing the orthogonality of left hand eigenvectors, demonstrating the method on both linear and geometrically nonlinear flutter analyses.

### 3.3.2 Mode Tracking Algorithm

In this work we implement a variation of the mode tracking method proposed by van Zyl [183]. As mentioned previously, to prevent mode hopping, the modes must be tracked at two stages in the analysis process: 1) During the reduced frequency sweep, which finds all the roots (eigenvalues) for a given dynamic pressure,  $q_i$ , and 2) during the migration of the modes between dynamic pressure increments,  $q_i$  and  $q_{i+1}$ .

The tracking of the modes during the first stage, the frequency sweep, is simpler to handle and implement rather than the second stage. This is because the number of eigenvectors are the same for the previous and current reduced frequency, resulting in a one-to-one mapping of eigenvectors, or a square correlation matrix. The tracking of the modes during the second stage, the mode migration, is more challenging and

requires extra care. This is due to the fact that any number of new modes can show up, as well as disappear, resulting in a rectangular correlation matrix. In the case of new modes appearing, for example in the case of a bifurcation, some modes are not correlated to any previous mode. Thus, the mode tracking algorithm need to handle such events.

To address this, a correlation metric is implemented to determine if the computed set eigenvalues are “too far” away from the previously computed values (at  $q_i$ ) for the current dynamic pressure increment,  $q_{i+1} = \Delta q_{\text{step}} + q_i$ . Based on this value the eigenvalues and eigenvectors from the frequency sweep computed at  $q_{i+1}$  are either accepted or rejected. If accepted the process continues to the next dynamic pressure. If rejected and the dynamic pressure step is halved and the reduced frequency sweep is rerun computing new roots. The halving process is controlled by a minimum allowed for  $\Delta q_{\text{min}}$  increment and if reached the current roots are accepted and process continues to the next dynamic pressure. In case of failure where minimum increment is reached, the roots are accepted and the dynamic pressure is incremented using a normal step size.

### 3.4 Flutter Constraint Formulation

For effective gradient-based optimization the constraint function should be at least  $C^1$  continuous. Jonsson *et al.* [2] discuss many of the considerations necessary to formulate an efficient and continuous flutter constraint suitable for a gradient-based optimization framework.

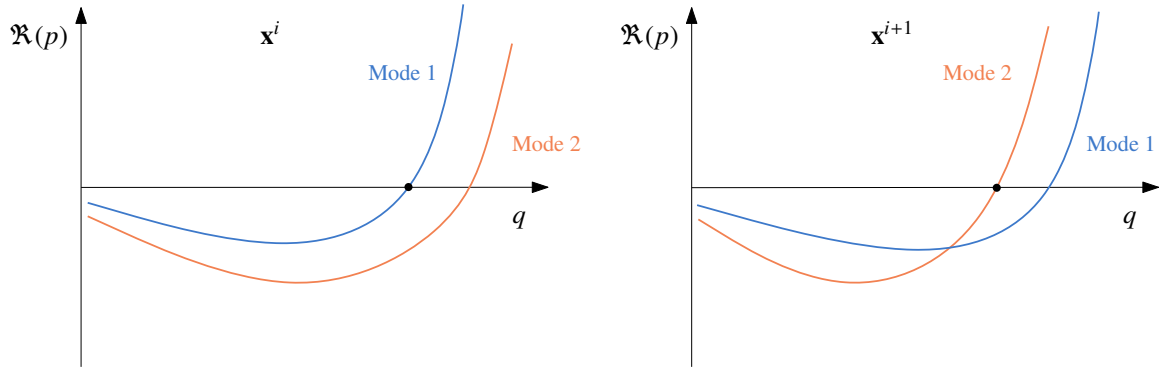
For linear systems, the flutter point is defined as the minimum dynamic pressure where at least one of the modes becomes unstable [189]. A naive approach to constrain flutter is to specify the flutter point directly. However, this may introduce discontinuities in the constraint value between two consecutive design iterations,  $\mathbf{x}^i$

and  $\mathbf{x}^{i+1}$ , due to mode switching or to a hump mode becoming active at a significantly lower dynamic pressure [5]. Therefore, additional steps are required to ensure a continuous constraint.

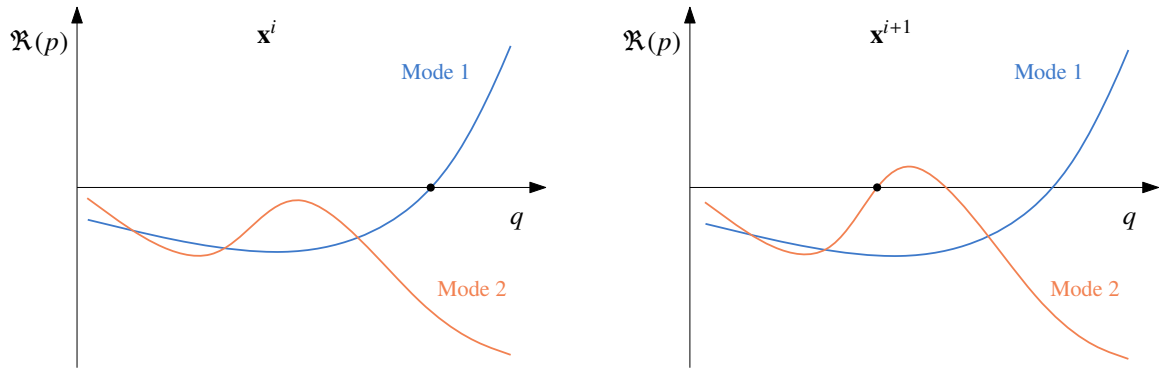
Mode switching occurs when the mode that first becomes unstable (that is, at the lower dynamic pressure) changes between two consecutive design iterations  $\mathbf{x}^i$  and  $\mathbf{x}^{i+1}$ . An example of mode switching is shown in Fig. 3.4a, where the hypothetical damping of a system with two modes is plotted with respect to speed. Mode switching causes a  $C^1$  discontinuity of the flutter point, which poses a challenge to gradient-based optimizers [109]. The imaginary part of the eigenvalue (frequency) also switches, and in many cases the frequencies coalesce, causing a mode to become unstable. A more serious problem is when a hump mode is present in design ( $\mathbf{x}^i$ ) and it becomes the critical mode in the new design ( $\mathbf{x}^{i+1}$ ), as shown in Fig. 3.4b. The constraint demonstrates  $C^0$  discontinuity, which is even more challenging for gradient-based optimizers.

Techniques exist to mitigate these problems and they are summarized by Stanford et al. [146]. Frequency-separation constraints proposed by Langthjem and Sugiyama [190] and also by Odaka and Furuya [191] can prevent mode switching by enforcing the critical mode to remain the same. This approach is illustrated in Fig. 3.5 (left) for a hypothetical case. The disadvantage of this method is that  $N_m - 2$  constraints are needed for a case with  $N_m$  modes (with the expectation that two modes coalesce, and hence no constraint is needed for two modes). Furthermore, specifying the unstable mode could over-constrain the optimization process. A more serious flaw with this approach is that a hump mode is still possible.

Other techniques exist to handle both mode switching and hump modes. One approach is to enforce the real part of each eigenvalue—the damping—to remain below a preset bounding curve. The area above this bounding curve, also known as flutter margin [113], is now defined as the unstable region were no modes should



(a) Critical flutter mode switching with small changes in the design causing a  $C^1$  discontinuity in the flutter point value.



(b) Hump mode becoming active with small changes in the design, causing a  $C^0$  discontinuity in the flutter point value.

Figure 3.4: Possible sources of discontinuities in the flutter constraint. Mode 1, critical at design  $\mathbf{x}^i$ , switches with mode 2 as the critical mode in design  $\mathbf{x}^{i+1}$  [2].

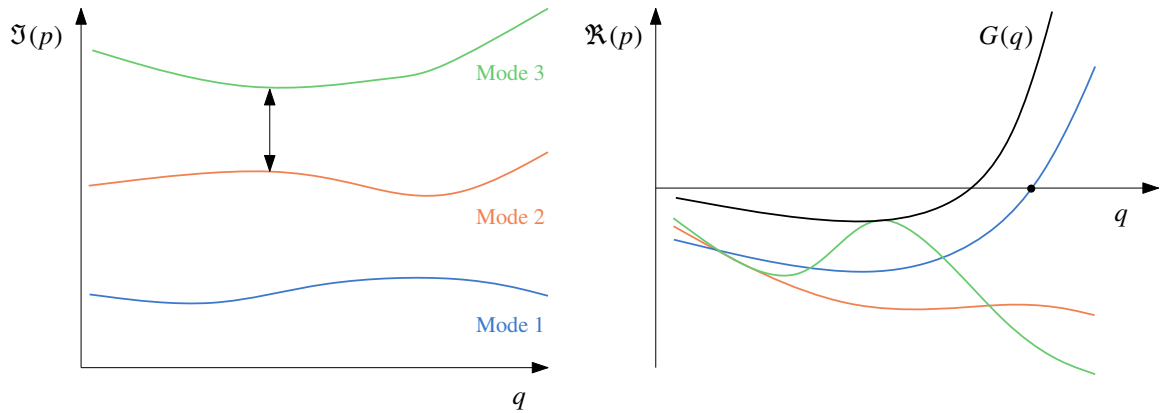


Figure 3.5: Two possible methods to prevent discontinuities in the flutter constraint: frequency-separation method (left) and damping boundary (right, solid black) [2].

cross into. Several variations of such a bounding curve have been proposed and employed [109, 113, 174, 192, 193], and implemented in commercial software [194]. This curve,  $G(q)$  in Fig. 3.6, spans the operating conditions of interest from wind-off to some maximum speed. This approach mitigates the aforementioned issues and has the following additional benefits:

1. No constraint is placed on the flutter point itself. Thus, there is no need to compute it explicitly because it is implicitly set by the intersection of the curve with the dynamic pressure axis.
2. Modes that become unstable abruptly (hard flutter) and have steeper slope than the boundary are handled.
3. Hump modes can be pushed into the negative damping to make the design more robust.

In this work, we use a modified version of the bounding curve proposed by Stanford et al. [113, 193], where we introduce a new shifting parameter  $g^+$ . The curve,  $G(q)$ , is a piecewise cubic and quadratic function and is defined as,

$$G(q) = \begin{cases} g^* (3q^2q^* - 2q^3)/(q^*)^3 + g^+ & 0 \leq q < q^* \\ \beta(q - q^*)^2 + g^* + g^+ & q \geq q^* \end{cases} \quad (3.50)$$

where  $g^*$ ,  $g^+$ ,  $q^*$ , and  $\beta$  are constants that control the shape and position of the curve and are chosen based on the criteria of the problem at hand. The  $g^*$  parameter controls the depth of the transition location of cubic and quadratic functions and is primarily used to configure the boundary for marginally stable hump modes. The  $g^+$  parameter shifts the entire boundary along the damping axis. For instance,  $g^+$  could be used to compensate or shift the boundary appropriately when structural damping is included in Eq. (3.40) or dealing with marginally stable in-plane modes as discussed

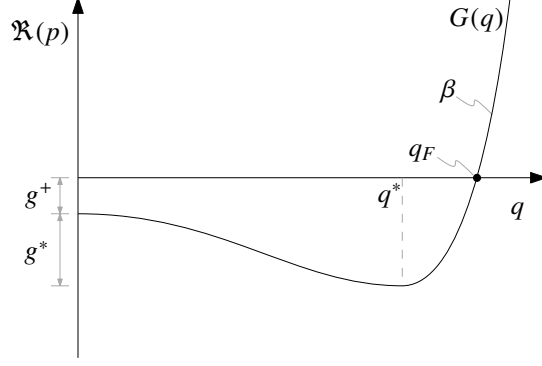


Figure 3.6: Bounding curve and configurable constants applied to mitigate issues with discontinuities due to hump and mode swaps. The intersection of the bounding curve and the zero axis implicitly defines the minimum flutter speed  $q_F$ .

in the next section. The  $q^*$  parameter sets the location of the implicit flutter point. Assuming that  $g^* + g^+ < 0$ , the implicit flutter point can then be computed as,

$$q_F = q^* + \sqrt{\frac{-(g^* + g^+)}{\beta}}. \quad (3.51)$$

Finally, the  $\beta$  parameter controls the slope of the quadratic function, limiting how hard modes can flutter. These parameters help improve the overall safety and robustness of the design. Figure 3.6 shows qualitatively the bounding curve and its configurable parameters.

A system is considered stable only if the damping,  $g_{ij} = \Re(p_{ij})$ , is below  $G(q_i)$  for every mode  $j$  at every dynamic pressure of interest  $i$ , i.e.,

$$g_{ij} \leq G(q_i) \quad i = 1, \dots, N_q \quad j = 1, \dots, N_m \quad (3.52)$$

where,  $N_q, N_m$  are the total number of dynamic pressure increments and modes, respectively. In other words, for system stability we require,

$$g_{G,ij} = g_{ij} - G(q_i) \leq 0 \quad \forall i, j. \quad (3.53)$$



If the bounding curve is not used, then  $G(q_i) = 0$ , such that,  $g_{G,ij} = g_{ij} \leq 0 \quad \forall i, j$ , which simply enforces all damping values to be less than zero.

### 3.5 Flutter Constraint Aggregation

Although the bounding curve approach mitigates discontinuity issues, it requires a constraint for each velocity increment and mode as Eq. (3.53) suggests. Thus,  $N_q$  dynamic pressure increments and  $N_m$  modes results in  $N_m \times N_q$  constraints, increasing the total number of constraints dramatically. In these circumstances, the active set method could be used. This consists of considering the full set of points but, based on the constraint value, reduces them to a smaller set before evaluating derivatives [109]. This approach was applied by Ringertz [109] and was also applied in the time domain by Kang et al. [195]. An alternate approach was proposed by Haftka [103], who suggested replacing parametric constraints by minimum-value constraints. This reduces the number of constraints to the total number of modes for the entire flight envelope. Using an active set method would reduce the cost of gradient computation for such constraints even further, as mentioned above.

To reduce the number of constraints resulting from the proposed bounding curve to a single scalar constraint that can be used in an optimization, we apply a constraint aggregation approach. We employ the discrete Kreisselmeier–Steinhauser (KS) function [124, 196–198] to aggregate the constraints into a single composite function. While alternative aggregation functions exist that are more accurate than the KS function, such as the induced exponential function [198], we take advantage of the KS function conservative property. The KS function is  $C_1$  continuous and gives a conservative estimate of the maximum for a given set of constraints. However, the KS function does not come without limitations, as discussed later. The KS function

can be defined as,

$$\text{KS}(\mathbf{g}(\mathbf{x})) = \frac{1}{\rho_{\text{KS}}} \ln \left( \sum_{j=1}^m e^{\rho_{\text{KS}} g_j(\mathbf{x})} \right). \quad (3.54)$$

To avoid numerical difficulties due to overflow, we use an alternate form of the KS function,

$$\text{KS}(\mathbf{g}(\mathbf{x})) = g_{\max}(\mathbf{x}) + \frac{1}{\rho_{\text{KS}}} \ln \left( \sum_{j=1}^m e^{\rho_{\text{KS}} [g_j(\mathbf{x}) - g_{\max}(\mathbf{x})]} \right), \quad (3.55)$$

where  $\mathbf{g}(\mathbf{x})$  is a set of constraints,  $g_{\max}(\mathbf{x})$  is the maximum of all constraints evaluated at the current design at a design point  $\mathbf{x}$ , and  $\rho_{\text{KS}}$  is the KS parameter. This KS parameter is used as a margin and is analogous to a penalty parameter used in constrained optimization. As  $\rho_{\text{KS}} \rightarrow \infty$ , the KS function approaches the true maximum. However, large values of  $\rho_{\text{KS}}$  results in a high curvature in the constraint function, which can make optimization more challenging.

Here, we apply the KS function twice in sequence: once for each mode over all dynamic pressures resulting in  $N_m$  KS constraints values, and then again over all the modes. The end result is the single constraint,

$$\text{KS}(g_{G,ij}, \rho_{\text{KS}})_{\text{flutter}} = \text{KS}([\text{KS}(g_{G,i1}, \rho_{\text{KS}}), \text{KS}(g_{G,i2}, \rho_{\text{KS}}), \dots, \text{KS}(g_{G,iN_m}, \rho_{\text{KS}})], \rho_{\text{KS}}) \leq 0. \quad (3.56)$$

The KS function is indifferent to the order of aggregation if the aggregation parameter  $\rho_{\text{KS}}$  is chosen to be the same for both aggregation steps. Although aggregating all damping values into a single constraint hides information from the optimizer, this allows for any number of modes to show up in the analysis process. In general, we only need to know if any one mode violates the constraint boundary during optimization. For a given design to be flutter free, the double aggregated KS value has to be less than zero, as defined in Eq. (3.56), indicating that the constraint is inactive.

Due to the conservative behavior of the KS function, it is sometimes found necessary to provide a positive  $g^+$  value such that the boundary does not adversely interfere

with any marginally stable in-plane modes during optimization. This is because when an in-plane mode (for which all damping values are zero) is aggregated using a KS aggregation, the aggregated value is greater than zero. Figure 3.7 demonstrates this issue: When vectors of different sizes containing only zeros are aggregated, the aggregation yields a positive value for practical KS parameter choices.

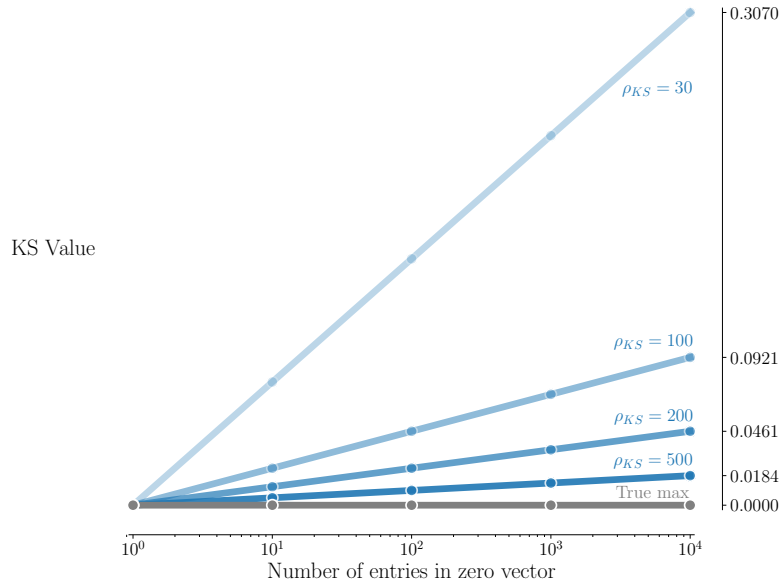


Figure 3.7: KS aggregation of vectors containing only zeros gives a positive value. It is important to compensate for this effect if in-plane modes are present by shifting the bounding curve  $G(q)$ .

Given this conservative behavior of the KS function, the flutter constraint is violated, even though no mode intersects or crosses into the area defined by the bounding curve. This causes problems for the optimizer because it may not be able to alter the design such that these in-plane modes are sufficiently affected, resulting in a premature termination of the optimization due to numerical difficulties. Thus, it is important to compensate for in-plane modes if they are present by choosing an appropriate KS parameter value and by shifting the boundary. This behavior is analogous to the mesh-dependency behavior of a KS or  $p$ -norm function, as discussed by Kennedy and Hicken [198].

## 3.6 Integration into MACH

The proposed flutter analysis is integrated into MACH as shown in Fig. 3.8. The flutter analysis is run in parallel to the static aerostructural computation. Thus, similar to the static aerostructural analysis a component that handles communications of data between different components such as the structural code and aerodynamic code is also implemented. The same component is also responsible for propagating reverse mode derivatives and computing the total derivative of the flutter constraint with respect to the design variables.

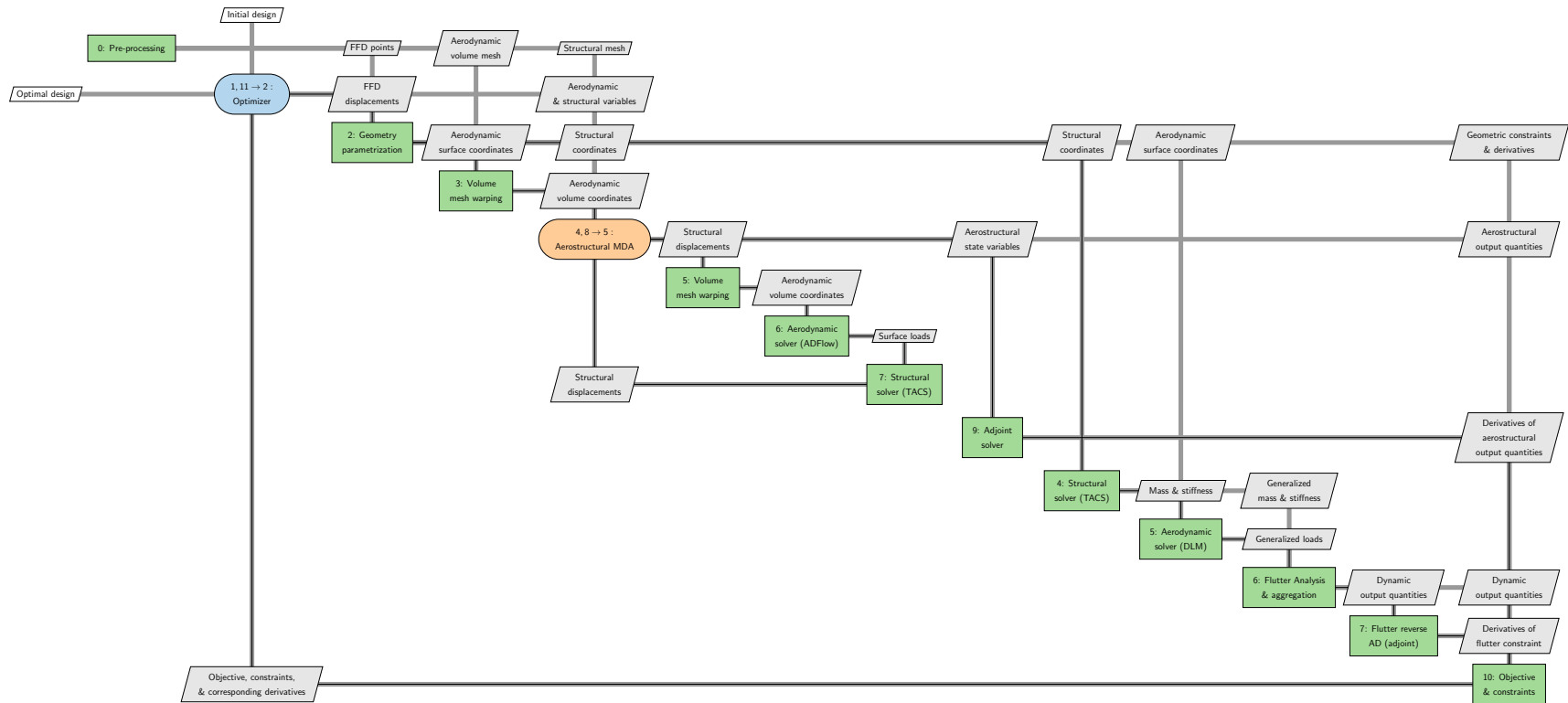


Figure 3.8: XDSM [1] of the proposed flutter analysis process and constraint formulation integrated into the high-fidelity aerostructural optimization MACH framework.

While MACH utilizes the Message Passing Interface (MPI) library that enables parallelization, the current implementation is not parallel. This means that the flutter analysis and the associated derivatives are executed on a single processor during the optimization. However, this is not found to be a limitation in an optimization as the CFD is the most expensive component in terms of computational cost.

## CHAPTER 4

# Flutter Constraint Derivative Implementation

To apply the flutter constraint described in Chapter 3 in a high-fidelity gradient-based optimization, efficient and accurate gradients are needed. The finite-difference method is a popular choice because it is easy to implement and can always be used, even with black-box codes. For a function of interest  $I(x)$ , the second order finite-difference stencil is

$$\frac{dI}{dx} = \frac{I(x+h) - I(x-h)}{2h} + O(h^2), \quad (4.1)$$

with a step-size ranging from  $h = 10^{-3}$  to  $h = 10^{-6}$  determined from a step-size study, depending on the function under consideration. However, finite differences are inaccurate and their computational cost scales poorly with the number of design variables.

Unlike the finite-difference method, the complex-step method [30] is accurate. For the complex-step method the derivative of a function is computed as,

$$\frac{dI}{dx} = \frac{\Im[I(x+ih)]}{h} + O(h^2), \quad (4.2)$$

where  $i = \sqrt{-1}$ . The step-size can be made very small, e.g.  $h = 10^{-40}$ , hence the  $O(h^2)$  truncation error becomes negligible. The complex-step method does not suffer

from a subtractive cancellation errors unlike the finite-difference method. However, its cost still scales poorly with the number of design variables, making it prohibitive for detailed design optimization. Furthermore, the complex-step method cannot be directly applied to programs that already use complex numbers. Such programs need to be made complex-step compatible by modifying them so that the real and imaginary part of the complex number are represented as two real numbers.

Analytic methods are desirable because they are both accurate and efficient, especially for iterative simulations [33]. However, they require significant implementation effort. There are two main approaches within the analytic methods: the direct approach and the adjoint approach. The adjoint approach is especially attractive because the computational cost is independent of the number of design variables [33, 34]. Finally, automatic differentiation (AD) can be applied to analysis codes to produce a new code that computes derivatives of that analysis [31, 32]. Two modes exist (forward and reverse) that are equivalent to the direct and adjoint in terms of efficiency and accuracy [33].

Typically, for an aerostructural optimization the functions of interest are very few compared to number of design variables such as, panel thickness. For a realistic aircraft configuration there are tens to thousands of design variables. This yields a high ratio of input variables to output variables and thus strongly favors the to use the adjoint equations or reverse-mode AD approach, where the latter is applied in this work.

## 4.1 Automatic Differentiation

Automatic differentiation, also known as algorithmic differentiation, is a well established method that systematically applies the differentiation chain rule to source code. This method uses source transformation tools that takes in the original computer pro-



gram, augments it, and generates a new code, such that it computes the analytical derivatives along with the original program [199, 200]. Two modes exist, the forward mode and the reverse mode. For a generic system with scalar input  $x$  and output  $y$  we can write it as:

$$\begin{aligned} \text{System} \quad x &\rightarrow \boxed{F(x)} \rightarrow y \\ \text{Forward AD} \quad \dot{x} &\rightarrow \boxed{F'(x)} \rightarrow \dot{y} \\ \text{Reverse AD} \quad \bar{x} &\leftarrow \boxed{F'^*(x)} \leftarrow \bar{y} \end{aligned}$$

where the arrows represent the flow of information, the box represents the system or a function. The forward mode, known as the tangent, is denoted with a dot ( $\dot{\phantom{x}}$ ) over the variable. Given some small variations on the input (independent) variables  $\mathbf{x}$  we can compute the resulting variations of the output (dependent) variables  $\mathbf{y}$ . The Jacobian matrix  $\mathbf{J}$  contains the partial derivatives of each dependent variable  $y_j$  with respect to each independent variable  $x_i$ . The forward mode thus computes  $d\mathbf{y} = \mathbf{J}d\mathbf{x}$  for each given  $d\mathbf{x}$  or

$$d\mathbf{y} = \mathbf{J}d\mathbf{x}$$

$$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \vdots \\ \dot{y}_n \end{bmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \dots & \frac{\partial y_1}{\partial x_m} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & \dots & \frac{\partial y_2}{\partial x_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial y_n}{\partial x_1} & \frac{\partial y_n}{\partial x_2} & \dots & \frac{\partial y_n}{\partial x_m} \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_m \end{bmatrix}.$$

Conversely, the reverse mode, known as the adjoint, is denoted by a bar ( $\bar{\phantom{x}}$ ) over the variable. The order of operations reverses and we compute the transposed

Jacobian product  $d\mathbf{x} = \mathbf{J}^T d\mathbf{y}$  for each given  $d\mathbf{y}$  or

$$d\mathbf{x} = \mathbf{J}^T d\mathbf{y}$$

$$\begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \vdots \\ \bar{x}_n \end{bmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_2}{\partial x_1} & \dots & \frac{\partial y_n}{\partial x_1} \\ \frac{\partial y_1}{\partial x_2} & \frac{\partial y_2}{\partial x_2} & \dots & \frac{\partial y_n}{\partial x_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial y_1}{\partial x_m} & \frac{\partial y_2}{\partial x_m} & \dots & \frac{\partial y_n}{\partial x_m} \end{bmatrix} \begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \\ \vdots \\ \bar{y}_m \end{bmatrix}.$$

In other words, the gradient of the independent variable is a linear combination of the variation in the dependent variable. This is a very important observation, particularly in the case with fewer output variables than input variables. Since many aerodynamic optimization formulations contain many more design variables than outputs of interest (or  $n_x \gg n_I$ ) we use the reverse mode or the adjoint mode in order to obtain the derivatives as efficiently as possible. The AD source-transformation tool, Tapenade [201, 202] is used in this work.

## 4.2 Component Derivative Implementation

Derivatives of the entire flutter constraint process presented in Fig. 3.1 are required. Since we allow change in planform through the chord and span design variables, the natural modes and mode shapes are impacted. This means that the fixed-mode derivative approximation [123] is not valid, and thus we require derivatives of the modes and mode shapes ( $\mathbf{K}_r, \mathbf{Q}_r$ ), the interpolated aerodynamic mode shapes ( $\mathbf{Q}_r^A$ ), the aerodynamic and structures meshes ( $\mathbf{X}^A, \mathbf{X}^S$ ), and the reference semi-chord ( $b$ ). Since we have only one output (the aggregated flutter constraint value) and a large number of design variables, we use a combination of analytic and reverse mode AD to obtain the total derivative  $dKS_{\text{flutter}}/d\mathbf{x}$  as efficiently as possible.

A high-level overview of derivatives implemented and computed is presented in

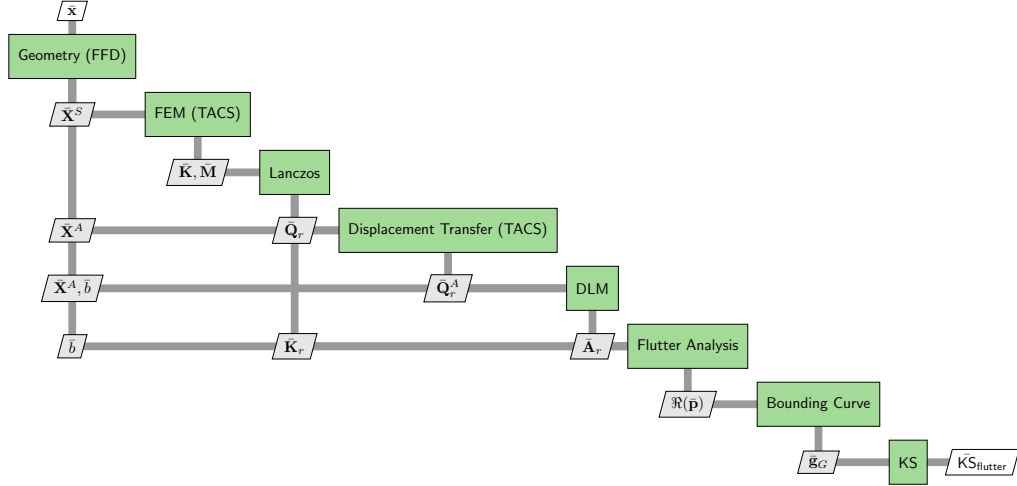


Figure 4.1: XDMSM [1] of derivatives required for the proposed flutter analysis and constraint formulation presented in reverse mode. The derivatives are computed by a combination of analytic adjoint and automatic differentiation (AD) in reverse.

Fig. 4.1. Reverse-mode compatible derivatives are required for all components. The bar (  $\bar{\phantom{x}}$  ) over the variable represent reverse derivative seed. Since the derivatives are back-propagated, the initial seed is  $\bar{K}S_{\text{flutter}}$ .

All new components implemented in this work, namely the Lanczos algorithm, DLM, flutter analysis algorithm, bounding curve, and aggregation functions are all implemented using a combination of AD and analytically differentiated code in reverse [203]. Matrix operation such as LAPACK's [176] complex and real linear solvers, matrix inversion, and eigenvalue solvers are not automatically differentiated, but must instead differentiated analytically in reverse. For instance, for a linear solver of the form  $\mathbf{Ax} = \mathbf{b}$ , the analytic derivative in reverse mode is obtained in a two-step process [203],

$$\bar{\mathbf{b}} = \mathbf{A}^{-T} \bar{\mathbf{x}} , \quad (4.3)$$

$$\bar{\mathbf{A}} = -\bar{\mathbf{b}}\mathbf{x}^T . \quad (4.4)$$

Equation (4.3) is first solved using the input seed  $\bar{\mathbf{x}}$ , and then Eq. (4.4) is solved to obtain the output seed.

The two reference semi-chord seeds,  $\bar{b}$ , are aggregated before they are used in the geometry FFD module. For increased efficiency, the  $\bar{\mathbf{K}}, \bar{\mathbf{M}}$  matrices are not explicitly formed. The resulting derivatives are propagated from the reverse mode Lanczos method through the FEM solver and directly accumulated onto the design variables.

### 4.3 Derivative Verification

To verify that the reverse derivatives are consistent and accurate we perform a rigorous verification. Each function is unit tested, where derivatives are computed and compared using a second-order central finite-difference stencil, the complex-step method [30], forward, and reverse mode AD. A similar procedure is performed for all analytically differentiated code in reverse.

For the finite difference verification we apply the second order stencil defined in Eq. (4.1). Note that in order to get a reasonable prediction with finite-difference, a step-size study was performed to get the most accurate gradient possible. Since the finite difference method requires no implementation it is used here as a reference for the complex-step and forward mode derivative implementation.

To accommodate and use the complex-step approach to verify the reverse derivatives the underlying implementation is modified such that the programs are complex-step safe. This is done by representing the complex numbers as two real numbers, one for the real part and one for the imaginary part. Complex arithmetic is performed using manually defined functions since it cannot be done using intrinsic functions. While this adds complexity and increases development time the benefit of accurate derivative outweighs the development cost. Here, all codes that use complex numbers, for instance the DLM and flutter analysis method, are modified such that only real numbers are utilized for complex calculations. In general a step size of  $h = 10^{-40}$  in Eq. (4.2) is used in all derivative tests in this work.

While not used in optimization, the forward mode derivative implementation serves an important role for derivative verification and debugging the reverse mode. Forward mode AD is simpler to implement and easier to debug and in code, can be compared line by line with values computed by finite difference or the complex-step approach. The forward AD computed derivative is compared to the complex-step method, where it should match to machine precision,

$$\frac{\Im[I(\mathbf{x} + ih\dot{\mathbf{x}})]}{h} = \mathbf{J}\dot{\mathbf{x}} = \dot{\mathbf{y}}, \quad (4.5)$$

where  $\dot{\mathbf{x}}$  is the derivative seed.

Furthermore, the forward mode is used to verify the reverse mode AD implementation through a *dot product test*. The dot product test [204] can be written as:

$$\begin{aligned} \bar{\mathbf{x}}^* \dot{\mathbf{x}} &= (\mathbf{J}^* \bar{\mathbf{y}})^* \dot{\mathbf{x}} \\ &= \bar{\mathbf{y}}^* (\mathbf{J}\dot{\mathbf{x}}) \\ &= \bar{\mathbf{y}}^* \dot{\mathbf{y}} \quad \text{set } \bar{\mathbf{y}} = \dot{\mathbf{y}} \\ &= \dot{\mathbf{y}}^* \dot{\mathbf{y}} \end{aligned} \quad (4.6)$$

This equality should match to machine precision. This demonstrates that the reverse mode implementation is consistent with the forward mode implementation, but this does not prove that the derivatives are correct. To verify the correctness of the derivatives, the forward mode should be compared against a method that provides accurate reference values, such as the complex-step method. Derivative verification is presented in the following sections. While derivative verification is done on a much lower level, here we perform collective verification involving a sequence of derivative computations.

Table 4.1: Intermediate sensitivities of the aggregated flutter constraint,  $KS$ , with respect to a single value in the reduced stiffness  $\mathbf{K}_r$  matrix, aerodynamic mesh points  $\mathbf{X}^A$  matrix and the reduced aerodynamic mode shapes  $\mathbf{Q}_r^A$ . Finite difference step size of  $h = 10^{-6}$  gave overall the best results.

	$\frac{\partial KS_{\text{flutter}}}{\partial \mathbf{K}_r}$	$\frac{\partial KS_{\text{flutter}}}{\partial \mathbf{X}^A}$	$\frac{\partial KS_{\text{flutter}}}{\partial \mathbf{Q}_r^A}$
Finite Difference	0.00220121232353	0.0921885550120	-0.000394003940585
Complex-step	0.00220122045797	0.0921885466706	-0.000394004582749
AD (Reverse)	0.00220122045797	0.0921885466706	-0.000394004582749

### 4.3.1 Intermediate Derivatives

Here we present derivative results of the flutter constraint with respect to the reduced stiffness matrix  $\mathbf{K}_r$ , the aerodynamic mesh nodes  $\mathbf{X}^A$  and  $\mathbf{Q}_r^A$  which are the reduced mode shapes,  $\mathbf{Q}_r$ , transferred on to the aerodynamic mesh. This includes derivatives computation of the generalized loads in the DLM, the flutter analysis implementation, and aggregation. This includes most of the new code implemented in this work. As shown in Table 4.1 the reverse AD sensitivities developed match to machine precision when compared to complex-step. As expected, the second order finite difference method does not perform as well, and is sensitive to variation in step-size. In order to get the best finite difference derivative the step-size was varied from  $h = 10^{-3}$  to  $h = 10^{-7}$  depending on which derivative was being calculated.

### 4.3.2 Total Derivatives

Total sensitivities of the flutter constraint with respect to the design variables are presented in Table 4.2. These are sensitivities of the entire analysis process. As before, finite difference offers less accuracy compared to AD or complex-step despite performing a step-size study using values from  $h = 10^{-3}$  to  $h = 10^{-7}$ . The number of matching digits for the full derivative chain is somewhat less than what is

Table 4.2: Sensitivities of flutter constraint,  $KS$ , with respect to design variables, chord, span and material thickness. Finite difference step size of  $h = 10^{-3}$  for geometric variables (chord, span) and  $h = 10^{-6}$  for structure variables gave overall the best results.

	$\frac{dKS_{\text{flutter}}}{dx_{\text{chord}}}$	$\frac{dKS_{\text{flutter}}}{dx_{\text{span}}}$	$\frac{dKS_{\text{flutter}}}{dx_{\text{thickness}}}$
Finite Difference	0.53698 <b>6799217</b>	-1.0462 <b>5065236</b>	177. <b>140039513</b>
Complex-step	0.536985 <b>041094</b>	-1.04624 <b>172821</b>	177.2002 <b>99680</b>
AD (Reverse)	0.536985 <b>160000</b>	-1.04624 <b>205000</b>	177.2002 <b>09304</b>

presented in Table 4.1. Once passed through the reverse implementation of TACS accuracy appears to be lost. However, the same behavior is reported in literature for aerostructural optimization (c.f. Table 2 in [7]). This stems from the large condition numbers of the structural matrices when employing thin shell structures. Despite this reduction, the accuracy of the implemented adjoint sensitivities are sufficient for high-fidelity aerostructural optimization.

## CHAPTER 5

# Aerostructural Optimization of Plate

The purpose of this chapter is to demonstrate the flutter constraint formulation capabilities in an multidisciplinary optimization and verify its continuity. The geometry used here is an idealized wing, a rectangular flat plate, which is chosen specifically for verification purposes due to its simplicity and short optimization turnaround. The proposed constraint is demonstrated in a simple but representative optimization, formulated to verify the constraint behavior in a gradient based optimization, including structural sizing and aerodynamic planform variables.

### 5.1 Baseline Model Description

The geometry, shown in Fig. 5.1, consists of a flat plate structure, shown in red, that is embedded within a larger flat aerodynamic mesh, shown in gray. The red spheres are control points of the FFD volume, which both meshes have been embedded in. The black lines connecting the spheres show the outer edges of the FFD.

The structural model consists of 12 elements in the streamwise direction and 40 elements in the spanwise direction a total of 480 finite MITC shell elements. Initial element thickness is chosen to be  $t = 0.0012$  m. The aerodynamic model consists of 12 elements in the streamwise direction and 20 elements in the spanwise direction. Material properties, dimensions, and discretization for the baseline flat plate are summarized in Table 5.1.



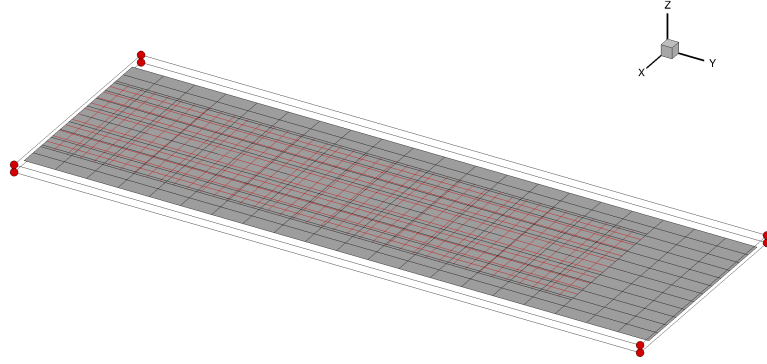


Figure 5.1: Flat plate structural and aerodynamic mesh shown in red and black respectively. The plate is cantilevered at the left edge. A Free-Form-Deformation (FFD) volume is also shown with 8 control points which are depicted as red spheres connected by solid black lines.

Table 5.1: Flat plate mechanical properties, dimensions, and discretization of the structure and the aerodynamic surface.

	Variable	Symbol	Value
Mechanical properties	Density	$\rho_s$	2800 kg/m <sup>3</sup>
	Modulus of elasticity	$E$	70 GPa
	Poisson ratio	$\nu$	0.3
	Yield stress	$\sigma_y$	400 MPa
CSM	Thickness	$t$	0.0012 m
	Structure span	$b_s$	0.85 m
	Structure chord	$c_s$	0.21 m
	Finite elements, streamwise	$n_x^{\text{FEM}}$	12
	Finite elements, spanwise	$n_y^{\text{FEM}}$	40
DLM	Span	$b$	1.0 m
	Chord	$c$	0.3 m
	DLM elements, streamwise	$n_x^{\text{DLM}}$	12
	DLM elements, spanwise	$n_y^{\text{DLM}}$	20
	Planform area	$A_{\text{init}}$	0.3 m <sup>2</sup>

Table 5.2: Flat plate operating conditions under investigation used in the baseline analysis.

	Variable	Symbol	Value
Operating conditions	Mach	$M_\infty$	0
	Lift coefficient	$C_L$	0.5
	Air density	$\rho_\infty$	1.225 kg/m <sup>3</sup>
	Air speed range	$U_\infty$	2 – 15 m/s
	Reduced frequency range	$k$	0 – 20

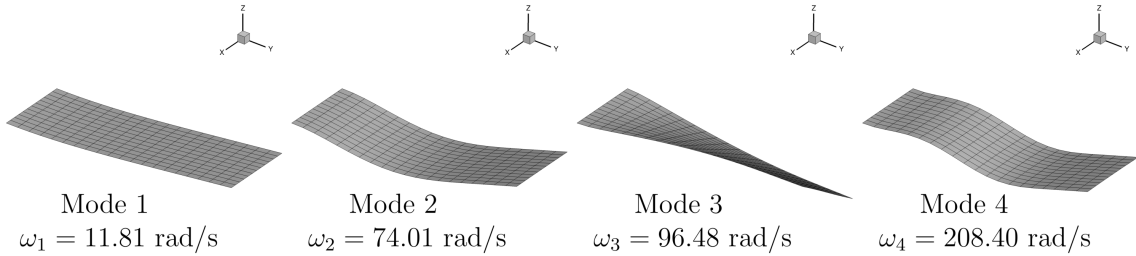


Figure 5.2: First 4 modes shapes for the baseline geometry.

For the flat plate analysis and optimization the air density is kept fixed and the Mach number is set to zero (incompressible flow). The velocity range to be analyzed is from 2 to 15 m/s. The flight conditions are summarized in Table 5.2. The aerodynamic loads generated by the DLM are generated using 50 values for the reduced frequency,  $k$ . These values are sampled non-uniformly using a quadratic or cubic stencil. In order to be consistent with previous work [168], we apply the constraint aggregation presented in Section 3.4 to the first 4 flutter modes. The first  $r = 6$  natural modes and mode shapes are computed using the Lanczos algorithm, with a subspace of size  $m = 20$ , and are used as a basis for the flutter solution. A large subspace size ensures that the first 6 modes are accurate and orthonormal modes. Figure 5.2 shows the first 4 natural mode shapes.

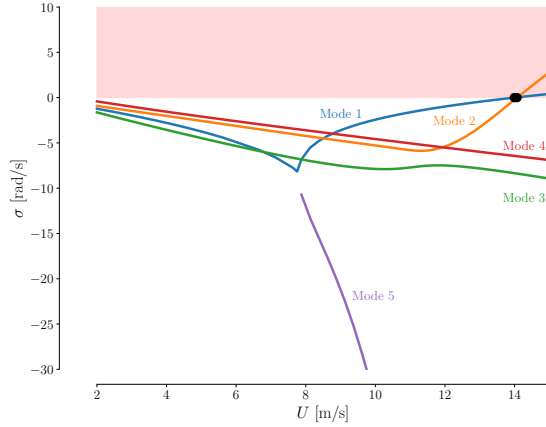
### 5.1.1 Baseline Model Flutter Analysis

We investigate the flutter and divergence characteristics of the baseline geometry under the given operating conditions. Figure 5.3 shows the flutter and divergence characteristics for the baseline geometry where damping and frequency are given in dimensional units. The unstable region is highlighted in a faded red color. In Fig. 5.3a the unstable region is the entire positive damping region. Aeroelastic divergence and flutter occur at  $U_d = 13.99$  and  $U_f = 14.07$  m/s, respectively.

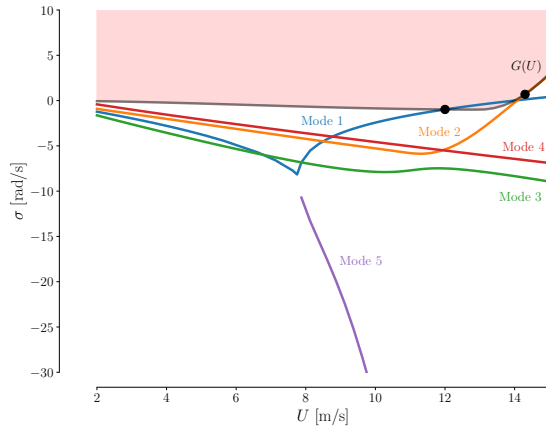
In Fig. 5.3b the boundary in Eq. (3.50) has been applied with  $g^* = -1$ ,  $g^+ = 0$  rad/s,  $U^* = 13$  m/s, and  $\beta = 1$  rad  $\cdot$  s/m<sup>2</sup>. Note that Eq. (3.50) assumes dynamic pressure, but it can be equivalently written in terms of speed. The boundary is shown in light gray and defines the beginning of the unstable damping region. The selection of current parameters sets the minimum implicit allowable flutter speed to 14 m/s. Due to  $0 > g^*$  the boundary extends into the negative damping area, creating different mode intersection points. The boundary divergence speed occurs at  $U_{G,d} = 12.00$  m/s, but the boundary flutter speed has increased and is  $U_{G,f} = 14.30$  m/s. Note that these are not the physical divergence or flutter velocities, but the values where modes intersect the constraint boundary  $G(U)$ . The physical flutter and divergence speeds  $U_f, U_d$  are the ones reported above. Frequency migration is shown in Fig. 5.3c. The aeroelastic mode emerging from the first 1<sup>st</sup> bending mode diverges whereas the fluttering mode emerges from the 2<sup>nd</sup> bending mode.

## 5.2 Problem Statement

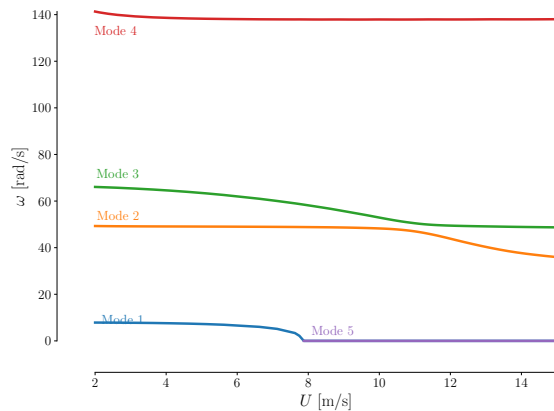
To maintain simplicity for this flat plate problem, the design variables chosen are as follows: plate thickness of the entire plate  $x_{\text{thickness}}$ , span  $x_{\text{span}}$  and chord length  $x_{\text{chord}}$ . While no sweep, taper or dihedral are considered here, the developed constraint and derivatives support such design variables. The objective is to maximize range using



(a) Damping shown without constraint boundary. Divergence and flutter occur at  $U_d = 13.99$  and  $U_f = 14.07$  m/s, respectively.



(b) Damping shown with constraint boundary applied on figure (shown in gray). Divergence and flutter (boundary intersection) occur at  $U_{G,d} = 12.00$  and  $U_{G,f} = 14.30$  m/s, respectively.



(c) Frequency migration

Figure 5.3: Flutter analysis of the flat plate baseline geometry. The unstable area is highlighted with a pink color in a), b). Frequency migration is shown in c).

the Breguet range equation,

$$R = \frac{V}{c_T} \frac{C_L}{C_D} \ln \left( \frac{W_{\text{init}}}{W_{\text{final}}} \right), \quad (5.1)$$

where  $R$  is range,  $V/c_T$  is the flight speed to thrust-specific fuel consumption ratio,  $C_L/C_D$  is the lift to drag ratio, and  $W_{\text{init}}/W_{\text{final}}$  is the initial to final cruise weight ratio. For simplicity we assume that  $V/c_T = 1$ , and we define the cruise weights as

$$\begin{aligned} W_{\text{final}} &= W_{\text{fixed}} + W_{\text{plate}}, \\ W_{\text{init}} &= W_{\text{final}} + W_{\text{fuel}}, \end{aligned} \quad (5.2)$$

where the  $W_{\text{fixed}}$  is a fixed weight and  $W_{\text{fuel}}$  is the fuel weight. Here we choose,  $W_{\text{fixed}} = 1.0$  kg and  $W_{\text{fuel}} = 0.25$  kg.

Since we do not compute any aerodynamic lift, we choose and set the target lift coefficient to  $C_L = 0.5$ . The drag coefficient computed here consists only of the lift induced drag,

$$C_D = \frac{C_L^2}{\pi e AR}, \quad (5.3)$$

where the wing span efficiency factor is set to  $e = 1$  for simplicity.  $AR$  is the aspect ratio defined as  $AR = b^2/S$  where  $b$  is the span and  $S$  is the planform area. Thus, the objective is maximized by reducing the drag coefficient or the thickness of the plate. The drag coefficient is reduced by increasing the aspect ratio as other parameters are fixed.

The chord and span directly affect the aspect ratio so we want to formulate the problem in terms of the aspect ratio rather than directly the chord and the span. By adding an area equality constraint we ensure that there is a link between the chord and span, allowing us to instead look at the trade-off between thickness and aspect ratio. Further, this allows for visualization, which can provide valuable information

Table 5.3: Optimization formulation of the flat plate problem.

	Function/variable	Description	Quantity
maximize	Range	Breguet equation	
with respect to	$x_{\text{span}}$	Plate span	1
	$x_{\text{chord}}$	Plate chord	1
	$x_{\text{thickness}}$	Plate thickness	1
		<b>Total design variables</b>	<b>3</b>
subject to	$A - A_{\text{init}} = 0.0$	Fixed plate area	1
	$\text{KS}(g_{G,ij})_{\text{flutter}} \leq 0$	KS aggregate of damping values for all modes	1
		<b>Total constraints</b>	<b>2</b>

about the design space . The initial area is given in Table 5.1.

We consider two scenarios to evaluate the proposed flutter constraint, one without applying the constraint curve, and a second one where the constraint curve is active, using the same parameters as presented in the baseline analysis. In the former we require that no flutter or divergence must occur for the entire velocity range of 2-15 m/s. For the latter case, minimum flutter or divergence speed is set implicitly by the parameters chosen for the constraint curve in the baseline analysis. Hence, the range 2-14 m/s must then be flutter and divergence free. In addition to pushing modes further into the design space making for a more robust design, it also controls how rapidly modes can flutter, further improving the safety. The flutter KS aggregation parameter is set to  $\rho_{\text{KS}_{\text{flutter}}} = 100$ .

The design variable bounds are as follows. Chord and span are specified such that the aspect ratio is allowed to vary from  $1 \leq AR \leq 6$ , and the material thickness of the plate is allowed to vary from  $0.0012 \leq t \leq 0.0025$  m. The optimization problem is summarized in Table 5.3 and Fig. 5.4 shows the flutter analysis implementation as it is applied in the optimization.

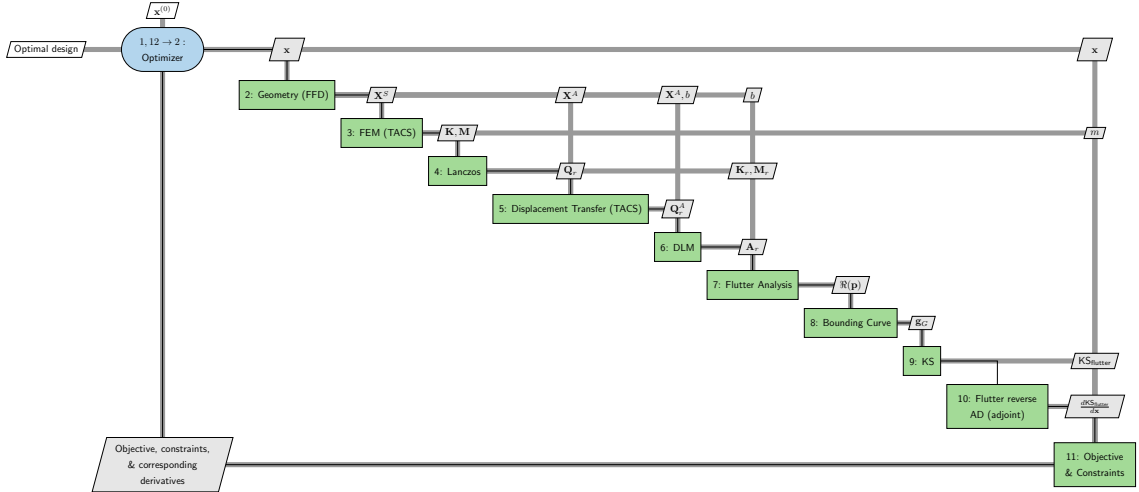


Figure 5.4: XDSM [1] showing the flutter constraint as applied in the optimization.

### 5.3 Design Space Analysis

Before running an optimization, its valuable to investigate the design space. A contour plot is generated by sweeping over both the aspect ratio range  $1 \leq AR \leq 6$  and the thickness range  $0.0012 \leq t \leq 0.0025$  m. A grid of 32 steps in each variable is used giving a total of 1024 analysis points. Figure 5.5 shows contour plots of the objective function in the feasible design space with and without the constraint curve  $G(U)$  active. Where the aggregated flutter constraint value is greater than zero,  $KS_{flutter} > 0$ , the objective function value has been blanked out, as this part of the design space is infeasible.

Comparing Figs. 5.5a and 5.5b it is evident that with the constraint curve active the feasible region is smaller. Furthermore, the constraint curve results in a smoother feasible design space, demonstrating the continuity property of the constraint curve formulation. The objective function is smooth and continuous with a clear maximum. By visual inspection the optimal aspect ratio is  $AR^* \approx 6$ , with a plate thickness of  $t^* \approx 0.00153$  m and  $t^* \approx 0.00165$  m without and with the constraint curve active, respectively. Note that here the range is presented as a negative quantity, due to

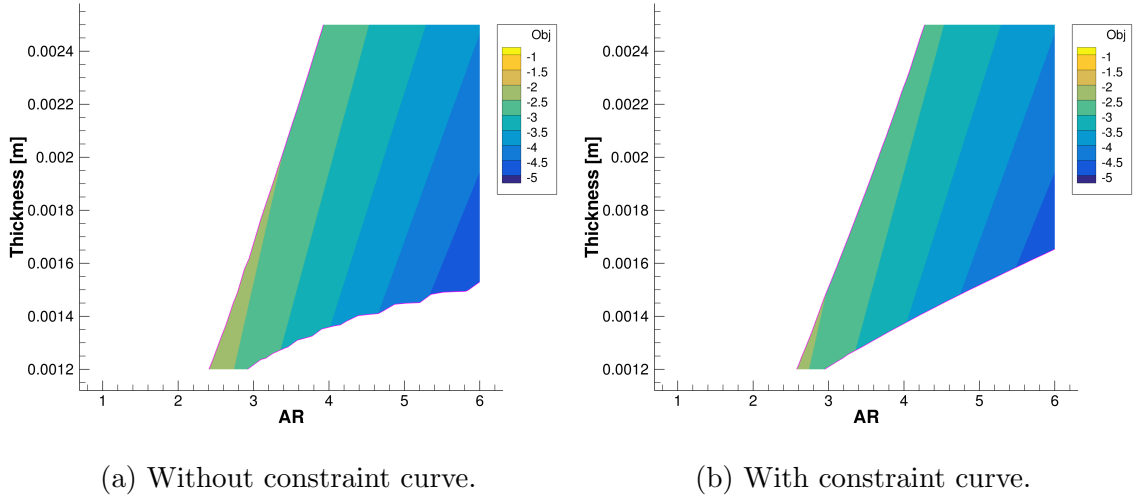


Figure 5.5: Contour plot of the objective function, range, shown with the flutter constraint applied to the contour plot. Blanked out regions represent values of where the constraint is violated or  $KS_{\text{flutter}} > 0$ . To generate the contour, the design space is sampled using 32 points in both variables for a total of 1024 design points.

objective being a maximization. Without and with the constraint curve active, the maximum range is approximately -5.0 and -4.8, respectively.

In both figures, the infeasible design space consists of two disjoint regions, one with low aspect ratio, spanning the entire thickness bounds, and the second one at higher aspect ratio with relatively thin plate thickness. These regions indicate that either a mode has fluttered or bifurcated and diverged. In order to investigate the infeasible regions further and determine the cause of these regions being infeasible, we fix the thickness of the plate to  $t = 0.001619$  m and perform sweep over the aspect ratio  $1 \leq AR \leq 6$ , using 256 points. This thickness value is chosen since it is close to the optimum thickness value. Here we perform the analysis without the constraint curve active.

Inspecting Fig. 5.6 we observe that the aggregated constraint value is smooth and continuous, except when the wing design changes from aspect ratio 2.176 to 2.196. At low aspect ratio the damping is positive, thus unstable. As aspect ratio grows



the wing becomes stable at  $AR \approx 3$ , remaining stable throughout. At  $AR \approx 5$  the damping increases again, tending towards zero, but this increment is investigated later. At the discontinuity, Figs. 5.7a and 5.7b show the damping and frequency migration for each design. Solid and dashed lines represent the wing design at aspect ratio 2.176 and 2.196 respectively. We observe that the damping characteristics of mode 4 seem to have changed significantly despite marginal changes in frequency for all modes. Other damping modes change marginally between designs, as expected.

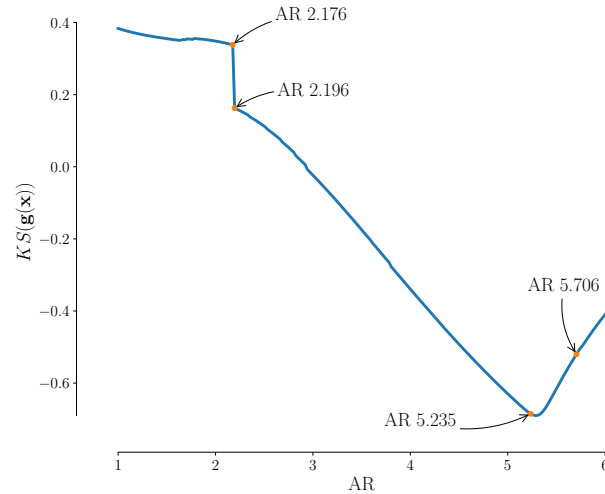


Figure 5.6: Aggregated damping value,  $KS_{\text{flutter}}$ , for a slice through the design space at  $t = 0.001619$  m. Small change in the design, increasing the aspect ratio from 2.176 to 2.196, results in a discontinuity.

This discontinuity is caused by a swap in natural mode shapes computed for each design. As the wing design changes, a higher frequency natural mode approaches and then becomes lower than natural mode 4, i.e. natural mode 5 and mode 4 trade places between these two designs. This becomes evident in Fig. 5.8 where the first five natural mode shapes are shown for these two designs. Since the natural modes are applied in the computation of the generalized matrices, this affects the flutter modes directly such that a different damping behavior may appear. However, in Fig. 5.7 mode 4 has fluttered at a very low velocity, which is non-physical. This is believed

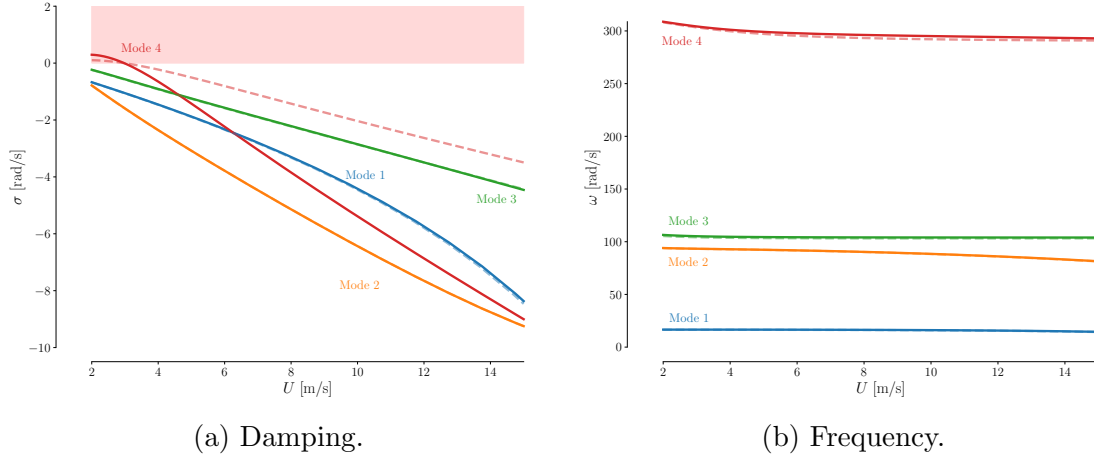


Figure 5.7: Small change in the design from aspect ratio from 2.176 to 2.196, shown in solid and dashed lines respectively, results in a large change in the damping. Frequency changes minimally as expected.

to be caused by the insufficient DLM resolution as the reduced frequency at such low flow velocities can be very large for the higher natural modes. This results in a non-physical behavior. With well behaved aerodynamics the continuity of the constraint is not compromised.

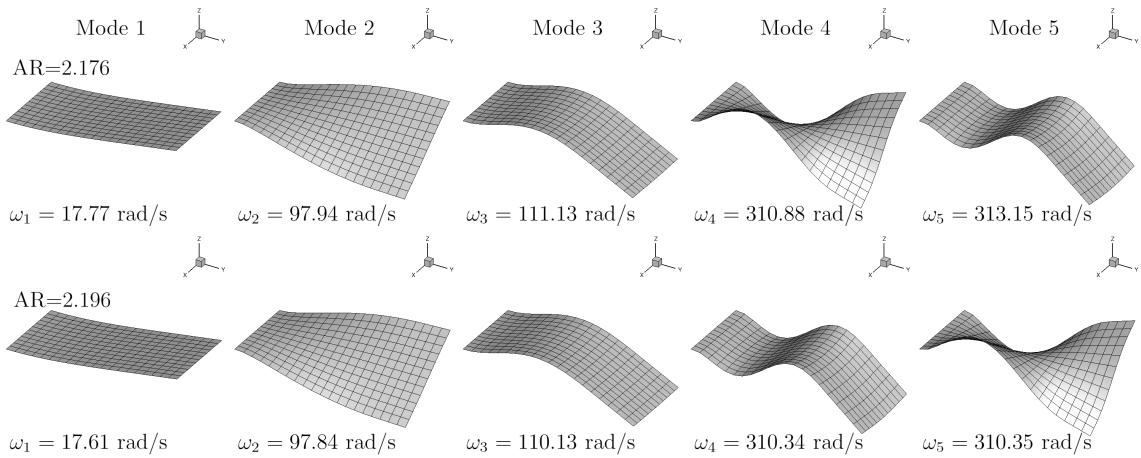


Figure 5.8: First five mode shapes for designs at aspect ratio 2.169 and 2.196. Mode 4 and 5 swap between the two designs.

To investigate the increase in the aggregated damping value at higher aspect ratio in Fig. 5.6 we select two designs at aspect ratio 5.235 and 5.706. Damping and

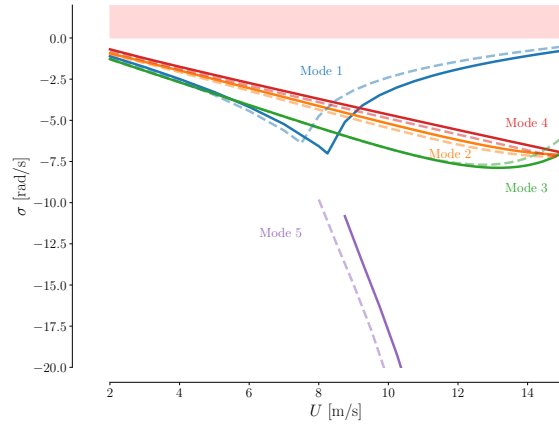
frequency plots for these designs are shown in Figs. 5.9a and 5.9c, respectively. The increase in damping is caused by the bifurcated mode 1 where, for the higher aspect ratio design, the bifurcation occurs at a lower velocity. The constraint curve is shown in Fig. 5.9b, where the bifurcated mode 1, at the higher aspect ratio design, is close to crossing into the unstable region. This is why the blanked out region in Fig. 5.5b at higher aspect ratio is larger than the one without it active. The effect is that the constraint curve pushes the bifurcated mode further into the damping region. Thus, in order to be flutter free (as indicated by Fig. 5.5b) a thicker plate is needed, which will increase the wing stiffness resulting in a more robust design.

## 5.4 Optimization Results

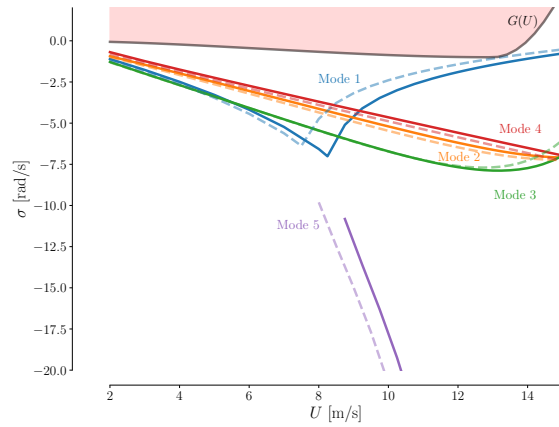
In this section we present the results from the optimization problem presented in Section 5.2. The baseline and optimized plate flutter and divergence characteristics are presented in Fig. 5.10 and the optimized modes and mode shapes are shown in Fig. 5.11.

Inspecting the mode migration in Fig. 5.10a we see that mode 1 is the critical mode, the active mode. However, no mode is close to being active for the chosen flight conditions, and has been pushed out of the range considered. This indicates that the diverging mode is driving the design in the optimization. Similarly, Fig. 5.10b shows that the optimized wing has better overall frequency separation than the baseline. The optimized mode shapes, given in Fig. 5.11, are found to be the same as the baseline i.e. in order, 1<sup>st</sup> bending, 2<sup>nd</sup> bending, 1<sup>st</sup> torsion and 3<sup>rd</sup> bending.

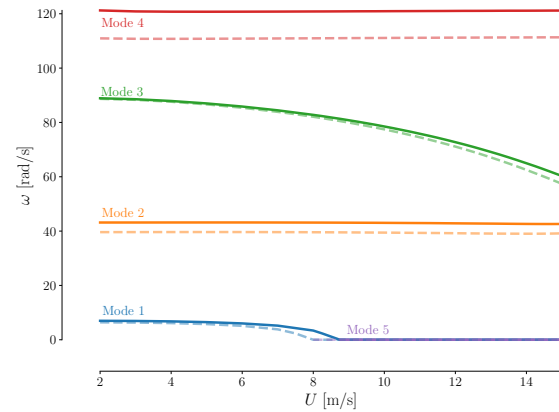
Figure 5.12 shows the major iterations that the optimizer superimposed on the design space and the baseline and optimized planform. The optimum aspect ratio and thickness is  $AR^* = 6$  and  $t^* = 0.00166$  m, respectively. The maximum range is found have increased by 60%, 38% thicker structure, and 82% increase in aspect ratio.



(a) Damping.



(b) Damping shown with constraint boundary, expanding the unstable region into the negative damping region.



(c) Frequency.

Figure 5.9: Damping and frequency plots for designs with aspect ratio 5.235 (solid) and 5.706 (dashed) for a fixed thickness. Increasing the aspect ratio shifts the bifurcation of mode 1 to a lower velocity. Frequency changes minimally as expected.

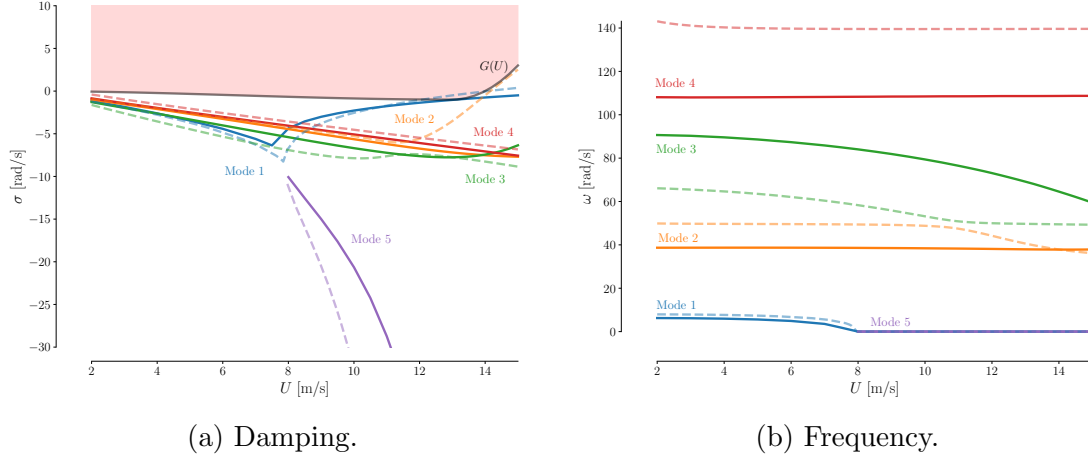


Figure 5.10: Baseline (dashed) and optimized (solid) frequency and damping characteristics.

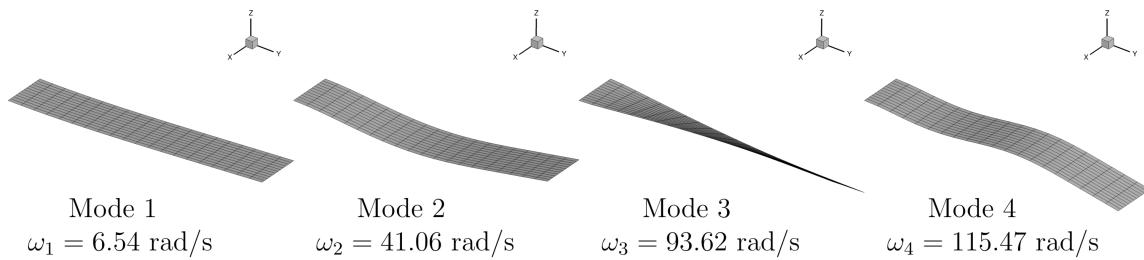


Figure 5.11: First 4 modes shapes for the optimized geometry.

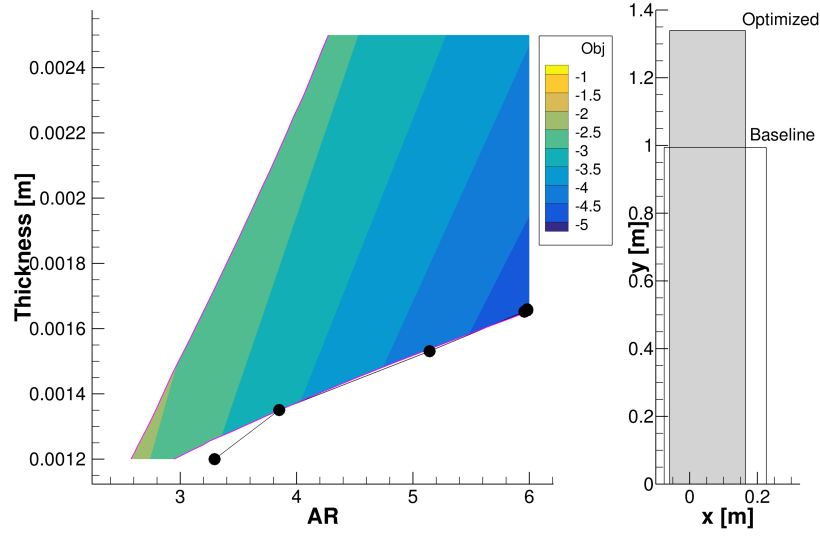


Figure 5.12: Optimization results showing the major iterations (left) taken by the optimizer and the initial and optimized wing planform (right). Optimum aspect ratio and thickness are  $AR^* = 6$ ,  $t^* = 0.00166$  m, respectively

Table 5.4: Numerical comparison of baseline and optimized wing.

	Baseline	Optimized	Rel. Diff. [%]
Aspect ratio [-]	3.29522	6.00000	+82.1
Thickness [m]	0.00120	0.00166	+38.3
Range [m]	-3.00636	-4.81924	+60.3

Table 5.4 compares the numerical values of the thickness, aspect ratio, and range. Figure 5.13 shows the convergence history for of the optimization. Observer that a quadratic convergence is obtained for this smooth problem. Minimum feasibility and optimality was specified as  $< 10^{-6}$  and was reached in both cases. Exceptional feasibility is demonstrated as it reaches machine zero.

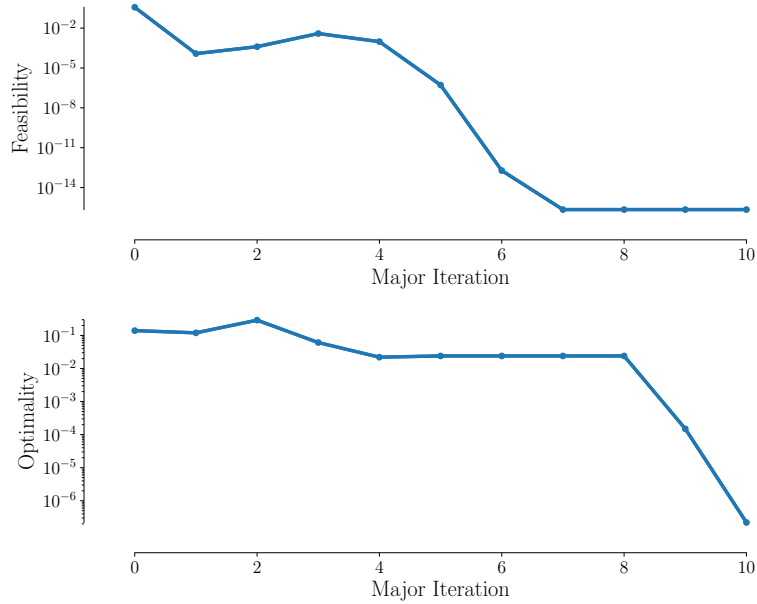


Figure 5.13: Optimization convergence history.

## 5.5 Summary

In this chapter the proposed flutter constraint formulation was successfully demonstrated on an idealized wing using a multidisciplinary objective, namely the Breguet range equation. The problem is specifically tailored to include both structural sizing and aerodynamic planform design variables. The bounds are specifically chosen to demonstrate a substantial change in planform, highlighting capabilities of the proposed constraint. Furthermore, the simplicity of the problem allows for a design space study, which demonstrates that the proposed constraint formulation is smooth and continuous. While an exception to this is observed at very low velocities, it is determined that this behavior is non-physical and numerical in nature. An optimized design is obtained efficiently, finishing successfully in only few steps. Compared to the baseline, the range is increased by 60% while maintaining a divergence and flutter free design. While the final optimized range has limited significance, the capabilities and continuity of the constraint are demonstrated successfully.

## CHAPTER 6

# Aerostructural Optimization of Wingbox

In this chapter, we demonstrate the proposed flutter constraint in a multipoint high-fidelity aerostructural optimization. We perform a multipoint high-fidelity aerostructural optimization of a wing with and without the flutter constraint subject to failure (stress) and buckling constraints. The objective is to maximize range with respect to structural sizing and aerodynamic planform variables. The geometry under study is a high-fidelity representation of a subsonic rectangular wing. First, the baseline aerodynamic and structural model parametrization is introduced, followed by a baseline characteristics analysis. Then, we describe the problem in detail (objective, multipoint flight conditions, design variables, and constraints). Finally, we compare two optimal aerostructural design results, obtained with and without the flutter constraint.

This chapter is structured as follows. Section 6.1 describes the aerodynamic and structural models and their parametrization. Following that Section 6.2 describes a structural pre-optimization to obtain a baseline sizing prior to the aerostructural optimization. Section 6.3 describes the optimization problem formulation, flight conditions, design variables, and constraints. We then demonstrate the proposed methodology in Section 6.4, where we compare high-fidelity aerostructural optimization results with and without the flutter constraint. Section 6.5 summarizes our findings.



## 6.1 Wing Model Description

### 6.1.1 Aerodynamic Models

The geometry under consideration is a rectangular wing with NACA 0012 cross-section, no dihedral, with a semi-span of 4 m and a chord of 0.765 m. This corresponds to an aspect ratio of 10.45 and a planform area of  $A_{\text{init}} = 3.06 \text{ m}^2$ .

The CFD mesh is generated using an in-house developed open-source hyperbolic volume mesh generator *pyHyp*, which is extruded from the surface mesh shown in Fig. 6.1a.<sup>1</sup> In general a mesh size study should be conducted such that aerodynamic loads are predicted accurately and are mesh independent. However, the goal in this work is to demonstrate the flutter constraint in a high-fidelity design optimization, and the effect such constraint has on the optimized design, so this study is omitted here. Furthermore, CFD computational cost is commonly the most computationally expensive component in an aerostructural optimization. Therefore, we choose a relatively coarse CFD volume mesh of 47,177 elements, which suffices for demonstrating the methods developed in this work. The computational domain is a semisphere farfield, which extends approximately 100 spans away from the wing and a symmetry plane boundary conditions defined at the wing root.

The DLM aerodynamic surface has the same dimensions as the CFD mesh. The mesh consists of 20 spanwise and 12 chordwise elements, for a total of 240 elements, and is shown in Fig. 6.1b. Similarly, as with the CFD mesh, a symmetry boundary condition is defined at the wing root. The CFD and DLM meshes and their physical dimension is summarized in Table 6.1.

---

<sup>1</sup><https://github.com/mdolab/pyhyp>

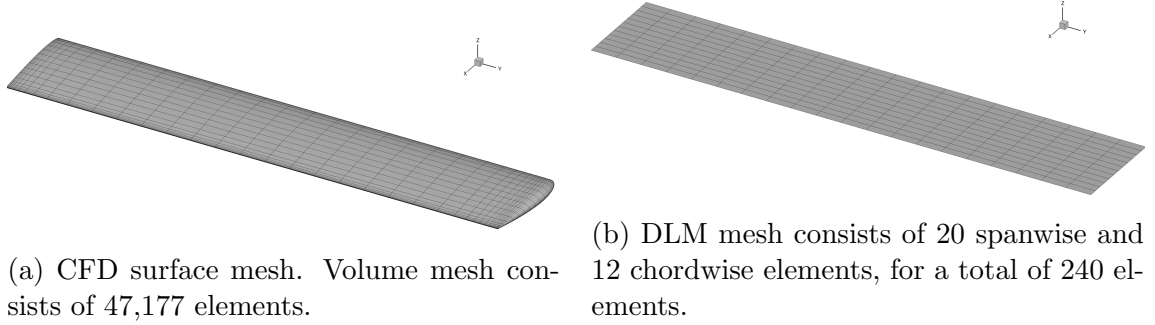


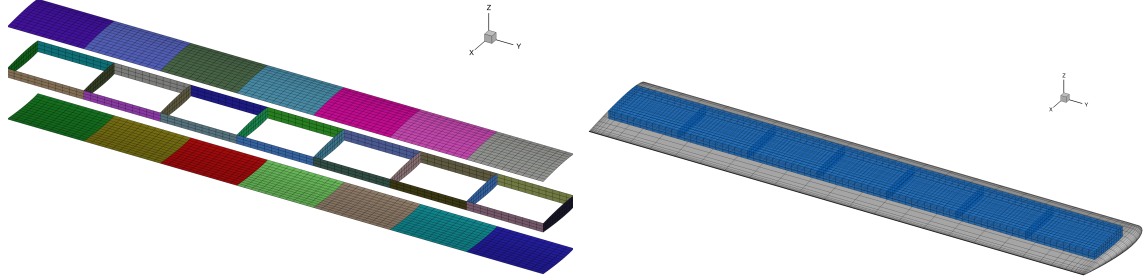
Figure 6.1: CFD and DLM surface meshes used in this work.

Table 6.1: Aerodynamic geometry, CFD, and DLM mesh summary.

	Variable	Symbol	Value	Unit
Geometry	Semi-span	$b_a$	4	m
	Chord	$c_a$	0.765	m
	Planform area	$S_{\text{ref}}$	3.06	m <sup>2</sup>
	Aspect ratio	$AR_{\text{init}}$	10.45	-
CFD mesh	# Elements	-	47177	-
DLM mesh	# elements, streamwise	$n_x^{\text{DLM}}$	12	-
	# elements, spanwise	$n_y^{\text{DLM}}$	20	-
	# elements, total	$n_{\text{total}}^{\text{DLM}}$	240	-

### 6.1.2 Structural Model

The wingbox geometry used in this work is generated using an in-house code pyLayout and is shown in Fig. 6.2a. It has a rectangular planform, it is fitted to the NACA 0012 airfoil cross-section, has 8 ribs, 2 spars (front and rear), and 14 skin panels (7 upper and 7 lower). The chord and span are 0.3825 m and 3.89 m, respectively. The structural model is discretized using MITC4 second-order shell elements, consisting of 2,688 nodes (16,128 DOFs) for the half-span model. The wingbox material is aluminum AL2024, whose mechanical properties are summarized in Table 6.2.



(a) Exploded view showing the upper and lower skins, the two spars, and the eight ribs. Each colored panel represents a design variable group. (b) Wingbox shown with the CFD surface mesh. Wingbox cross-sectional shape is fitted to the NACA 0012 airfoil.

Figure 6.2: Wingbox structural model.

Table 6.2: Mechanical properties, dimensions, and discretization of the wingbox structure.

	Variable	Symbol	Value	Unit
Mechanical properties (AL2024)	Density	$\rho_s$	2780	kg/m <sup>3</sup>
	Modulus of elasticity	$E$	72.4	GPa
	Poisson ratio	$\nu$	0.33	-
	Yield stress	$\sigma_y$	420	MPa
Geometry (fitted NACA0012)	Semi-span	$b_s$	3.89	m
	Chord	$c_s$	0.3825	m
FEM	# Nodes	-	2688	-
	# DOFs	-	16128	-

### 6.1.3 Structural Parametrization

Including stiffeners in wingbox design, is essential to provide lightweight but stiff structure. In this work we use a smeared stiffness approach instead of modeling the stiffeners explicitly [205]. Explicit stiffeners can be difficult to model accurately due to their small size, requiring often large number of elements, which increases computational cost. In contrast, using the smeared stiffener approach the effect of the implicit stiffeners on a given panel stiffness is modeled directly through the material

properties. This approach is computationally efficient and removes the need to model discrete stiffeners explicitly. Furthermore, it can model stiffeners in any direction and works well when panel length to stiffener pitch ratio is reasonably large, implying large number of stiffeners [206]. All components in the wingbox are modeled using a smeared stiffness approach. A cross-section of such panel geometry is shown in Fig. 6.3.

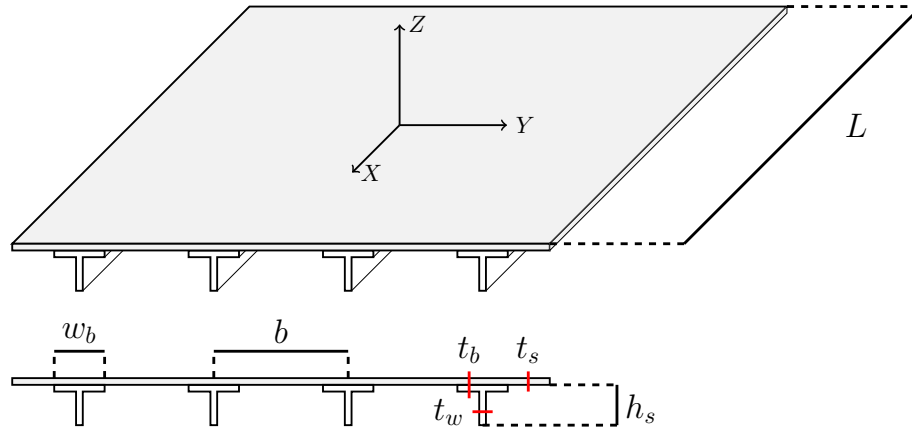


Figure 6.3: Panel geometry and the design variables available in the smeared stiffness model.

The design variables available in this smeared approach are, the panel thickness  $t_s$ , the stiffener height  $h_s$ , the stiffener thickness  $t_w$ , the stiffener pitch  $b$ , stiffener base width  $w_b$ , stiffener base thickness  $t_b$ , and panel length  $L$ . Each panel shown in Fig. 6.2a is modeled using its own set of design variables as depicted in Fig. 6.3. These variables can be used to compute an *equivalent thickness* value. This value is a measure on how thick a panel needs to be (if modeled only using a thickness value) to have the same stiffness properties as the smeared approach. Note that the panel length is made a design variable to simplify buckling constraint computations. This is done since each panel geometry changes during the optimization. Additionally, a nonlinear geometric consistency constraint is needed that ensures the physical panel length matches the panel length variable of each component.

Using this parametrization, each panel is assumed to be approximately flat and can be analyzed and designed based on stress states resulting from a collection of different loading conditions. Over the span of the wingbox the loading range can be large, being the smallest at the tip and largest at the root. This large variation in loading influences panel sizing, where the most heavily loaded components have the greatest impact on wingbox mass.

Finally, strength and buckling constraints can be enforced, as well as adjacency constraints that impose limits on the variation of the design variables between adjacent panels. The sizing of the panel is based on the simplified stiffness calculations and buckling criteria proposed by Stroud and Agranoff [206].

### 6.1.3.1 Smearred Panel Stiffness Calculations

The assumed overall layout of each panel is shown in Fig. 6.3. The stiffness of the overall panel is determined by accounting for the effect of the discrete stiffeners by adding additional bending and shear stiffness to the skin stiffness. The panel stiffness can be determined based on the panel thickness  $t_s$ , the stiffener height  $h_s$ , the stiffener width  $t_w$ , the stiffener pitch  $b$ , and the stiffness of the skin and stiffener  $\mathbf{Q}^{(p)}$  and  $\mathbf{Q}^{(s)}$ . The smeared panel stiffness matrices are:

$$\begin{aligned} \mathbf{A} &= t_s \mathbf{Q}^{(p)} + \mathbf{A}^{(s)}, & \mathbf{B} &= \mathbf{B}^{(s)}, \\ \mathbf{D} &= \frac{t_s^3}{12} \mathbf{Q}^{(p)} + \mathbf{D}^{(s)}, & \mathbf{A}_s &= \mathbf{A}_s^{(p)} + \mathbf{A}_s^{(s)}. \end{aligned} \tag{6.1}$$

The non-zero components of the matrices  $\mathbf{A}^{(s)}$ ,  $\mathbf{B}^{(s)}$ ,  $\mathbf{D}^{(s)}$ , and  $\mathbf{A}_s^{(s)}$  are:

$$\begin{aligned} \mathbf{A}_{11}^{(s)} &= \frac{E_s A_s}{s_p}, & \mathbf{B}_{11}^{(s)} &= -\frac{h_s}{2s_p} E_s A_s, \\ \mathbf{D}_{11}^{(s)} &= \frac{E_s (h_s^2 A_s + 4I_s)}{4b}, & \mathbf{A}_s^{(s)}{}_{11} &= \frac{5G_s A_s}{6s_p}, \end{aligned}$$

where,  $E_s = Q_{11}^s - \frac{Q_{21}^s Q_{12}^s}{Q_{66}^s}$  is the extension modulus of the stiffener,  $A_s$  is the area of the stiffener, and  $I_s$  is the second moment of area of the stiffener.

### 6.1.3.2 Panel-Level Buckling and Failure Analysis

The panel-level buckling and failure (stress) analysis is commonly conducted under the maneuver loading conditions, which are used to enforce buckling and failure constraints during optimization. Every panel in the structure is constrained such that under a given loading condition, its response is within a predefined operating envelope. For metallic structures the failure constraint is based on von Mises stress failure criterion.

The buckling constraint considers several independent modes: buckling of the panel section between stiffeners, buckling of the stiffeners, and overall panel buckling (including stiffeners and skins). The overall critical buckling loads are determined based on the approach of Stroud and Agranoff [206]. The skin and stiffener buckling loads are determined as follows. Panel ends are assumed to be simply supported along the lines of attachment with adjacent structural components. The critical loads are computed assuming a nearly flat panel, ignoring any curvature effects.

We assume that the interaction between the longitudinal and shear buckling modes collapses into the following buckling envelope:

$$B(N_1, N_{12}) = \frac{N_{12}^2}{N_{12,cr}^2} + \frac{N_1}{N_{1,cr}} \leq 1, \quad (6.2)$$

where  $N_1$  and  $N_{12}$  are the longitudinal and shear loads respectively, and  $N_{1,cr}$  and  $N_{12,cr}$  are the critical longitudinal and shear buckling loads. Note that  $N_1$  and  $N_{12}$  are computed in a locally aligned panel axis. Equation (6.2) is applied separately to the overall panel buckling, stiffener buckling and inter-stiffener skin buckling. The critical loads,  $N_{1,cr}$  and  $N_{12,cr}$  are determined based on the equations given in Table 6.3.

	Overall buckling	Skin buckling
$N_{1,cr}$	$\frac{\pi^2 EI_s}{bL_x^2}$	$\frac{2\pi^2}{b^2} \left( \sqrt{D_{11}D_{22}} + D_{12} + 2D_{66} \right)$
$EI_s = z_n^2(t_s b E_p + t_s h_s f_b E_s) + E_s \left( t_s \frac{h_s^3}{12} + t_s h_s \left( z_n - \frac{h_s}{2} \right)^2 \right)$		
$z_n = \frac{C_n}{A_n}$	$C_n = E_s \frac{h_s^2}{2} t_s$	$A_n = E_p b t_s + E_s h_s t_s (1 + f_b)$
$N_{12,cr}$	$\xi = \frac{\sqrt{D_1 D_2}}{D_3}$	$\xi = \frac{\sqrt{D_{11} D_{22}}}{D_{12} + 2D_{66}}$
If $\xi > 1$	$\frac{4}{L_x^2} (D_1^3 D_2)^{0.25} \left( 8.125 + \frac{5.045}{\xi} \right)$	$\frac{4}{b^2} (D_{11} D_{22}^3)^{0.25} \left( 8.125 + \frac{5.045}{\xi} \right)$
If $\xi \leq 1$	$\frac{4}{L_x^2} \sqrt{D_1 D_3} (11.7 + 0.532\xi + 0.938\xi^2)$	$\frac{4}{b^2} \sqrt{D_{22} (D_{12} + 2D_{66})} (11.7 + 0.532\xi + 0.938\xi^2)$

Table 6.3: A summary of the critical load computations for the overall and skin buckling constraints. Note that  $A_n$  and  $C_n$  are the modulus-weighted zeroth and first moments of area of the panel and stiffener, and  $z_n$  is the modulus-weighted centroid. The bending stiffness  $EI_s$  can then be used to determine the critical buckling load.

## 6.2 Structural Pre-Optimization

### 6.2.1 Problem Statement

Before performing an aerostructural optimization, we perform a structural sizing optimization of the wingbox. The objective is to minimize mass, operated under maneuver flight conditions, to determine its sizing, mass, and stiffness distribution. Such structural optimization has the following benefits: 1) The starting point in terms of mass and stiffness is closer to the aerostructural optimum; 2) The structural constraints start feasible and are more easily satisfied during the aerostructural optimization; 3) It accelerates the optimization and reduces the possibility of optimizer exiting prematurely.

The loads are computed from an aerostructural analysis at the maneuver flight conditions for a given structural sizing. This is because as the wingbox mass and stiffness properties change during the optimization the aerodynamic loads are in turn affected. Therefore, the structural optimization alone is multidisciplinary, requiring

the aerostructural loads to be updated at every design iterations. However, in order to reduce the computational cost we apply a lagged strategy to limit the number of costly aerostructural analysis. The structural optimization algorithm used here is described in Section 6.2.1.

---

```

1: Lagged structural optimization
2: Initialize structure sizing:  $x_1$ 
3: for  $k \leftarrow 1, k_{\max}$  do
4:   Perform aerostructural analysis using the maneuver load case with structural sizing  $x_k$ 
5:   Extract and save the aerostructural loads
6:   Perform the structural optimization described in Table 6.5 using the fixed loads computed
   in previous step until convergence
7:   Update the structural sizing  $x_{k+1}$  for next iteration
8: end for

```

---

The number of aerostructural analysis iteration needed for the structural optimization to converge is generally small and is set to  $k_{\max} = 5$  in this work. During each iteration  $k$  a structural optimization is performed where the objective is to minimize the mass of the structure.

## 6.2.2 Design Variables

The structural design variables for the smeared stiffener approach need to be chosen carefully. Thin panels with low stiffness may exhibit local behavior (as opposed to global) in a modal analysis. This is illustrated in Fig. 6.4, where local panel modes are dominant, which is an unwanted behavior in the context of this study.

The stiffener pitch and stiffener height are kept fixed and set to  $b = 0.15$  m and  $h_s = 0.05$  m, respectively. These values are chosen in conjunction with lower bounds on the remaining design variables, such that local behavior is minimized at the panel level during optimization. The remaining design variables are panel thickness  $x_{\text{panel thick}}$ , stiffener thickness  $x_{\text{stiff thick}}$ , and panel length  $L$ . As previously mentioned, the panel length is needed in the smeared stiffener approach due the change in geometry. The bounds on these design variables are large enough to allow



the optimizer to freely size the structure without the local behavior as previously described. The lower and upper bounds for the structural design variables are set to 0.0015 m and 0.25 m, respectively.

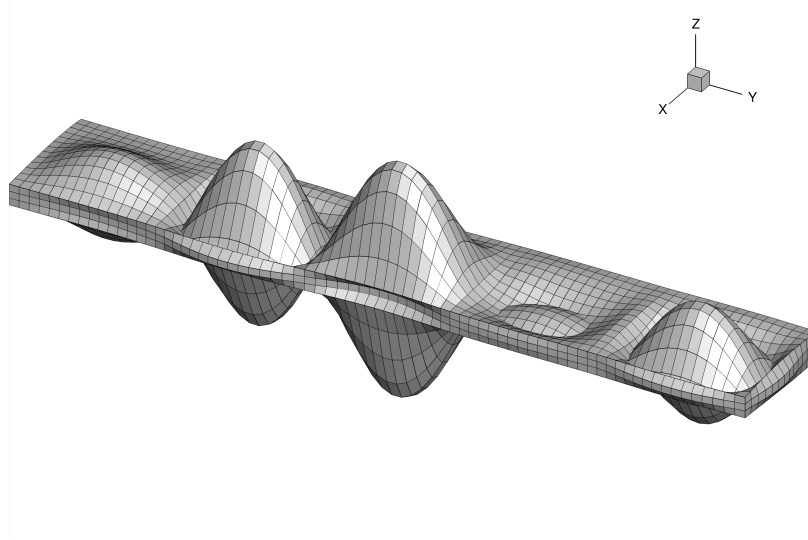


Figure 6.4: Modal analysis of a panel with low stiffness, demonstrating the need to increase panel thickness. A smeared stiffness approach is used to avoid excessively thick panels.

### 6.2.3 Constraint Definition

Each panel is constrained not to exceed the failure or buckling criterion discussed in Section 6.1.3.2, but this results in large number of constraints. To minimize the number of constraints used in the optimization, an aggregation technique is adopted. Here, a KS aggregation function is used to reduce the constraints by aggregating multiple panel failure and buckling constraints into groups. The KS aggregation parameter is set to a recommended value of  $\rho_{\text{KS}_{\text{stress}}} = \rho_{\text{KS}_{\text{buckling}}} = 100$  [26].

Several KS failure and buckling constraints are enforced and are summarized in Table 6.4. Three buckling constraints groups are used, one for the upper skins, one for both spars, and one for the ribs. Four failure constraints groups are used, one for the upper skins, one for the lower skins, one for both spars, and one for the ribs.

These KS constraints are generally difficult for the optimizer to satisfy due to their highly nonlinear behavior.

In addition to the maneuver loading, a safety factor of 1.5 is applied in the failure and buckling computation.

Table 6.4: Total of 4 failure and 3 buckling constraints are used in this work.

	Failure	Buckling
Upper skins	●	●
Lower skins	●	
Spars	●	●
Ribs	●	●
# Constraints	4	3

Linear adjacency constraints are enforced to limit large variation in the design variables between two adjacent panels. This constraint is applied on the panel and stiffener thickness design variables for the spars and the skins. Furthermore, panel length is added as a constraint that the optimizer matches based on the panel length design variable computed.

The optimization problem statement is summarized in Table 6.5.

## 6.2.4 Results

The cumulative convergence history of these 5 sequential structural optimizations is shown in Fig. 6.5. The orange markers indicate where aerostructural loads are updated. Convergence tolerances for feasibility and optimality are set to  $10^{-4}$ . As shown, only 2 outer iterations are needed as the later 3 iterations converge immediately. Note that both major and minor iterations are shown. Furthermore, the large spikes in the KS history is due the very nonlinear nature of this type of constraint. The optimized wingbox mass is 37.4 kg, where the mass breakdown by component is

Table 6.5: Optimization problem formulation for the structural mass minimization.

	Function/variable	Description	Quantity
minimize	mass	Structure mass (kg)	
with respect to	$x_{\text{panel thick}}$	Panel thickness Skins/Spars/Ribs	36
	$x_{\text{stiff thick}}$	Panel stiffener thickness Skins/Spars/Ribs	36
	$x_{\text{panel length}}$	Panel length Skin/Spars/Ribs	36
		<b>Total design variables</b>	<b>108</b>
subject to	$KS_{\text{stress}} \leq 1$	2.5 g Yield stress	4
	$KS_{\text{buckling}} \leq 1$	2.5 g Buckling	3
	$ x_{\text{stiff height}_i} - x_{\text{stiff height}_{i+1}}  \leq 0.001$	Skin stiffener height adjacency	12
	$ x_{\text{stiff thick}_i} - x_{\text{stiff thick}_{i+1}}  \leq 0.001$	Skin stiffener thickness adjacency	12
	$ x_{\text{panel thick}_i} - x_{\text{panel thick}_{i+1}}  \leq 0.001$	Skin thickness adjacency	12
	$ x_{\text{panel thick}_i} - x_{\text{panel thick}_{i+1}}  \leq 0.0015$	Spar thickness adjacency	12
	$ x_{\text{stiff thick}} - x_{\text{panel thick}}  < 0.001$	Maximum stiffener-skin difference	36
	$L_{\text{panel}} - x_{\text{panel length}} = 0$	Target panel length	36
		<b>Total constraints</b>	<b>128</b>

given in Table 6.6. All the buckling constraints for the specified groups, ribs, upper skins, and spars are active (at 1.0). These constraints are the most difficult constraints for the optimizer to satisfy. The lower skins group failure constraint is active, whereas all others are not close to being active.

#### 6.2.4.1 Modal and Flutter Characteristics

The first eight modes of the optimized structure are shown in Fig. 6.6. Natural mode shapes 1, 3, 5, and 7 are the 1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>rd</sup>, and 4<sup>th</sup> out-of-plane bending modes, respectively. Natural mode shapes 2 and 6 are the 1<sup>st</sup> and 2<sup>nd</sup> in-plane bending modes, respectively. Finally, natural mode shapes 4 and 8 are the 1<sup>st</sup> and 2<sup>nd</sup> torsion modes, respectively.

As shown in Fig. 6.6 some higher mode shapes, such as mode eight, show some local effects as discussed in Section 6.2.2. However, these effects are sufficiently small compared to the global behavior, such that they do not adversely affect the flutter analysis or optimization.

The flutter analysis for the flight conditions given in Table 6.7 is shown in Fig. 6.7.

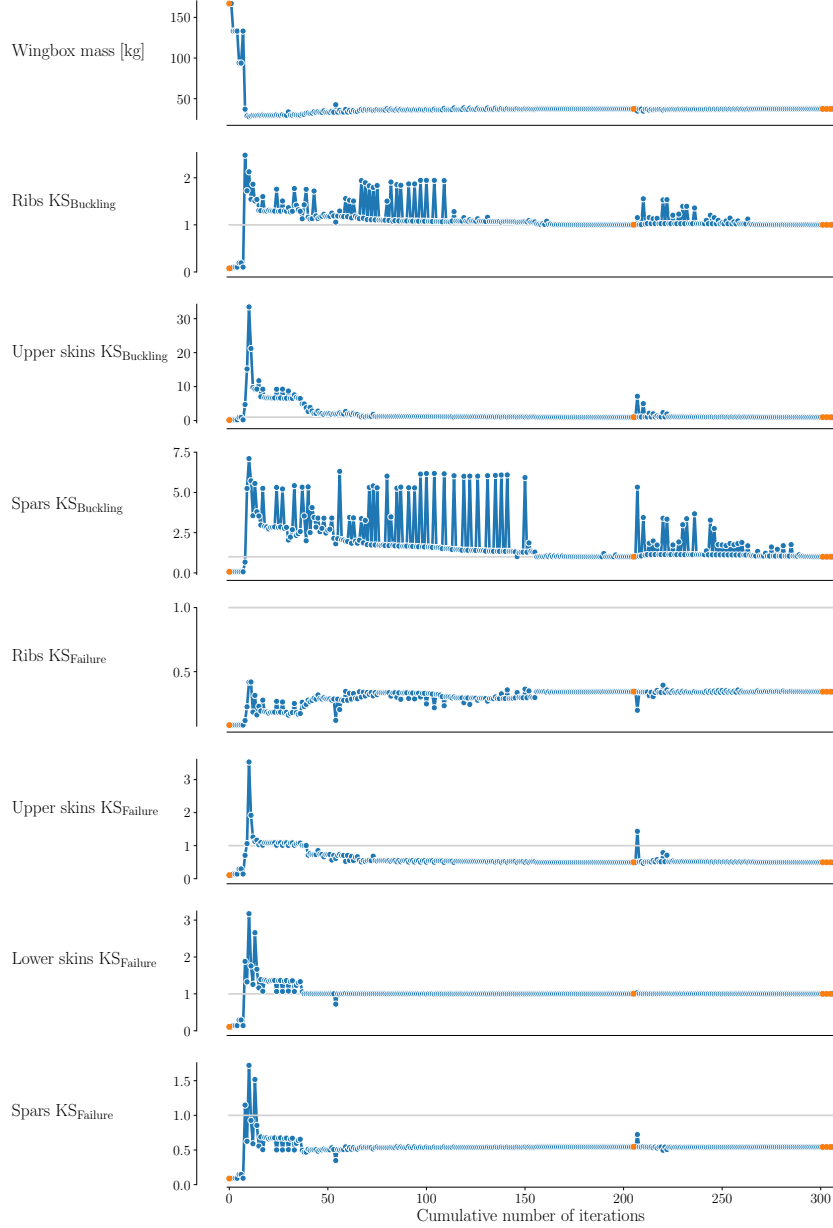


Figure 6.5: Cumulative convergence history of the 5 structural optimizations. The orange markers represent a start of a new structural optimization. The buckling constraints prove to be more difficult satisfy than the failure constraints.

The bounding curve parameters for  $G(q)$  presented here are the same as discussed in Section 6.3.3, resulting in a minimum flutter dynamic pressure of 98 kPa, show in Fig. 6.7a. This analysis demonstrates that no flutter or divergence occurs, since no modes cross into the positive damping space resulting in an inactive flutter constraint.

Table 6.6: Structural optimum mass distribution of the final design based on component group.

Component	Mass [kg]
Upper skins	20.06
Lower skins	11.48
Front spar	2.31
Rear spar	1.72
Ribs	1.83
Total	37.40

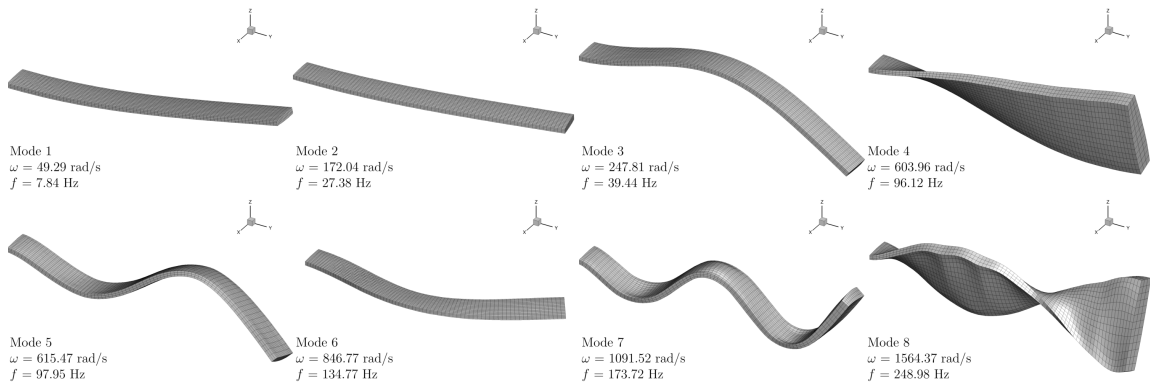


Figure 6.6: First eight natural mode shapes of the structural optimum.

A bifurcation the 1<sup>st</sup> bending mode (mode 1) appears at around 40 kPa, but does not lead to divergence. The in-plane bending modes do demonstrate some positive damping at the upper end of the dynamic pressure range, indicating that they are not perfectly in-plane. The optimized baseline model is thus neither flutter nor divergence critical.

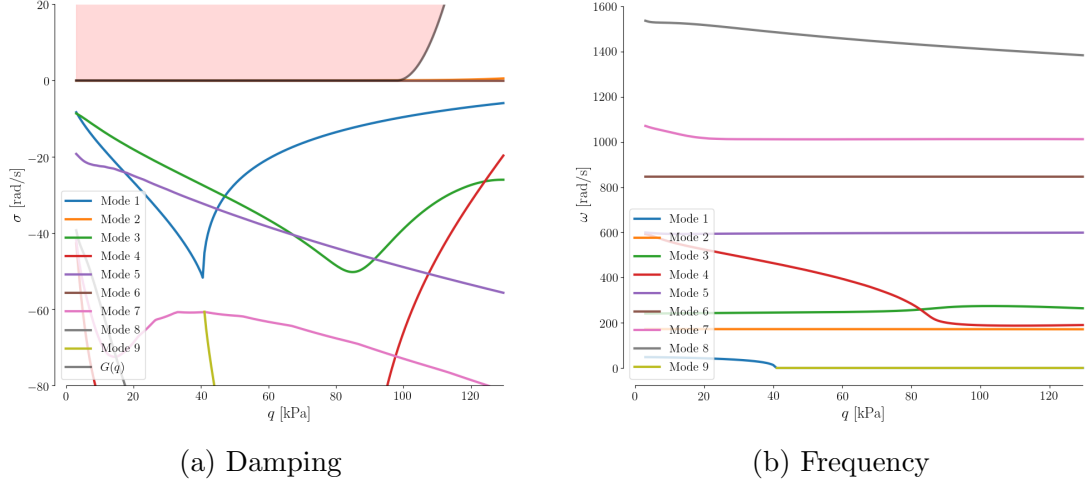


Figure 6.7: Flutter damping and frequency characteristics of the structural optimum. This structural design is not flutter critical, since none of the modes cross into the region defined by the bounding curve  $G(q)$ .

## 6.3 Aerostructural Optimization

### 6.3.1 Objective Function and Flight Conditions

To demonstrate the proposed flutter constraint in a high-fidelity multipoint aerostructural optimization, we need a multidisciplinary objective that is influenced both by aerodynamic and structural properties. As before we use the Breguet range equation,

$$R = \frac{V}{c_T} \frac{C_L}{C_D} \ln \left( \frac{W_{\text{init}}}{W_{\text{final}}} \right), \quad (6.3)$$

where  $R$  is range,  $V/c_T$  is the flight speed to thrust-specific fuel consumption ratio,  $C_L/C_D$  is the lift to drag ratio at cruise, and  $W_{\text{init}}/W_{\text{final}}$  is the initial to final cruise mass ratio. In this work we assume that  $V/c_T = 170.0/(0.5/3600)$ , and we define the cruise mass as

$$W_{\text{final}} = W_{\text{fixed}} + W_{\text{wingbox}}, \quad (6.4)$$

$$W_{\text{init}} = W_{\text{final}} + W_{\text{fuel}},$$

where the fixed and fuel mass are  $W_{\text{fixed}} = 400$  kg and  $W_{\text{fuel}} = 75$  kg, respectively. In the static aeroelastic analyses performed in this work, self-weight loads are included. The objective is to maximize the range with respect to aerodynamic planform and structural sizing variables.

Three flight conditions are defined for these two optimizations: cruise, maneuver, and flutter. The operating Mach number is  $M_\infty = 0.5$  for all flight conditions. The cruise altitude is 10,000 m and the maneuver conditions is run at sea level. Cruise lift and drag coefficients are used in the objective function, while the maneuver case is used to size the structure. For both conditions, a lift coefficient is estimated based on the total mass and is given in Table 6.7. For the cruise and maneuver conditions a load factor of  $n_c = 1.0g$  and  $n_m = 2.5g$  are used, respectively. Prior to the optimization, a coupled aerostructural  $C_L$  solve is performed to ensure the specified lift is obtained. This process will perform a coupled multidisciplinary analysis (MDA) by solving the coupled aerostructural problem for a given  $C_L$  by changing the angle of attack. For the flutter flight condition, the density is fixed at sea level standard atmosphere value,  $\rho_f = 1.225$  kg/m<sup>3</sup>, and the velocity is varied, resulting in a operating dynamic pressure range of 3–130 kPa. This is done since the dynamic pressure from the matched point sweep through altitudes is not great enough to incur flutter or divergence for the given design. Thus, in order to demonstrate the flutter constraint in a high-fidelity optimization a large enough dynamic pressure range is thus chosen such that flutter or divergence occurs. All flight conditions are summarized in Table 6.7.

Table 6.7: Operating conditions considered in this work.

Name	Mach	Altitude [m]	$q_\infty$ [kPa]	Target $C_{L,i}$	$\alpha$	Load factor
Cruise	0.5	10000	4.6	0.45	Design <sub>1</sub>	1.0
Maneuver	0.5	0	17.7	0.30	Design <sub>2</sub>	2.5
Flutter	0.5	0	3.0 – 130	-	-	1.0

### 6.3.2 Design Variables and Bounds

The design variables can be split into three types: geometric, aerodynamic, and structural design variables. The geometric parametrization influences simultaneously all aerodynamic surfaces and the structure directly. The aerodynamic and structural variables influence its discipline directly, and indirectly through coupling effects.  $\mathcal{O}(100)$  design variables are used in this multipoint aerostructural optimization and they are summarized in Table 6.8.

The wing planform is controlled through the movement of FFD control points, as explained in Section 2.1. Two global design variables are used: chord and span. Since both aerodynamic and the structural meshes are embedded within an FFD volume, these two variables influence both models. The bounds on these variables are large enough that they are not active at the optimum.

The aerodynamic variables are the angle of attack,  $\alpha$ , at each flight condition. This allows the design to meet the specified cruise and maneuver lift constraints.

The structural parametrization and design variables and bounds are the same as used in the structural optimization. We refer the reader to the discussion in Section 6.2.2 for details.

### 6.3.3 Constraints

For the optimization to give a design that is both physical and meaningful, a number of constraints need to be defined. In this section we describe all the constraints used in this work.  $\mathcal{O}(100)$  constraints are used in this multipoint aerostructural optimization and they are summarized in Table 6.8.

A planform area equality constraint is enforced, which maintains the initial baseline planform area during the optimization. Due to the planform area constraint, the chord and span design variables are linked, effectively reducing the two design variables to an aspect ratio variable. A lift coefficient equality constraint is enforced



for both the cruise and maneuver flight conditions.

The structural constraints used here are identical are the same as used in the structural optimization. Three buckling constraints groups are used, one for the upper skins, one for both spars, and one for the ribs. Four failure constraints groups are used, one for the upper skins, one for the lower skins, one for both spars, and one for the ribs. Linear adjacency constraints are enforced to limit large variation in the design variables between two adjacent panels. We refer the reader to the discussion in Section 6.2.3 for further details.

In addition to the constraints above, the previously described flutter constraint is enforced only in one of the two aerostructural optimization to study its effect on the wing design. For the bounding curve parameters used in this work we set  $g^* = 0$  rad/s,  $g^+ = 0.08$  rad/s,  $q^* = 98$  kPa, and  $\beta = 10^{-7}$  rad/(Pa<sup>2</sup> · s). The value of  $q^*$  is chosen by scaling the flutter dynamic pressure of the optimum design obtained without the flutter constraint,  $q_{f,w/o\ FC}$ , by a factor of 1.3. This is approximately 75% of the considered dynamic pressure range (see Table 6.7). The value of  $\beta$  is chosen such that hard flutter modes are limited. The  $g^+ = 0.08$  value shifts the entire boundary vertically into the positive damping space. The flutter KS aggregation parameter is set to  $\rho_{KS_{flutter}} = 500$ . These values are chosen based on the discussion in Sections 3.4 and 3.5 to compensate for the conservative behavior of the KS function. In particular, these values are chosen such that in-plane modes do not affect the optimization. No further attention is required regarding in-plane modes beyond what was previously defined. The optimizer should size the structure such that the in-plane modes remain in-plane for the flight envelope of interest.

The complete problem consists of 112 design variables and 131 constraints, including the flutter constraint. The optimization problem statement is summarized in Table 6.8. As mentioned previously, we perform two aerostructural optimizations, one with and one without the flutter constraint active, in an otherwise identical setup.

Table 6.8: Aerostructural optimization problem statement.

	Function/variable	Description	Quantity
maximize	Range	Breguet equation	
with respect to	$x_{\text{panel thick}}$	Panel thickness (skin, spars, ribs)	36
	$x_{\text{stiff thick}}$	Panel stiffener thickness (skin, spars, ribs)	36
	$x_{\text{panel length}}$	Panel length (skin, spars, ribs)	36
	$x_{\text{span}}$	Wing span	1
	$x_{\text{chord}}$	Wing chord	1
	$x_{\text{alpha}_i}$	Angle of attack at each flight condition	2
		<b>Total design variables</b>	<b>112</b>
subject to	$\text{KS}_{\text{flutter}} \leq 0$	KS aggregate of $g_{G,ij}$	1
	$\text{KS}_{\text{stress}} \leq 1$	2.5 g yield stress	4
	$\text{KS}_{\text{buckling}} \leq 1$	2.5 g buckling	3
	$C_L = C_{L,i}$	Lift constraint at each flight condition	2
	$A - A_{\text{init}} = 0.0$	Fixed planform area	1
	$ x_{\text{stiff height}_i} - x_{\text{stiff height}_{i+1}}  \leq 0.001$	Skin stiffener height adjacency	12
	$ x_{\text{stiff thick}_i} - x_{\text{stiff thick}_{i+1}}  \leq 0.001$	Skin stiffener thickness adjacency	12
	$ x_{\text{panel thick}_i} - x_{\text{panel thick}_{i+1}}  \leq 0.001$	Skin thickness adjacency	12
	$ x_{\text{panel thick}_i} - x_{\text{panel thick}_{i+1}}  \leq 0.0015$	Spar thickness adjacency	12
	$ x_{\text{stiff thick}} - x_{\text{panel thick}}  < 0.001$	Maximum stiffener-skin difference	36
	$L_{\text{panel}} - x_{\text{panel length}} = 0$	Target panel length	36
		<b>Total constraints</b>	<b>131</b>

## 6.4 Results

The two optimal designs and the baseline are compared in Table 6.9. Without any significant reduction in range or mass, we obtain a stiffer, lower aspect ratio wing design, with stark differences structural sizing, highlighting the importance of including such a constraint in the design optimization process. In both cases, the optimizer tries to minimize the induced drag by increasing the aspect ratio, which in turn increases range. However, while reducing the induced drag, the higher aspect ratio causes larger bending moments, resulting in a structural mass increase to satisfy failure and buckling constraints. With the flutter constraint, a lower aspect ratio design is obtained, resulting in a higher drag, which in turn will impact the objective. Planform comparison between the two optimum designs and the baseline is shown in Fig. 6.8. While the final mass of both structures is similar, with the flutter constrained optimal

Table 6.9: Aerostructural range maximization final designs, with and without the flutter constraint (FC) compared to the baseline shape.

	Baseline (B)	Opt. w/o FC (O1)	Opt. w/ FC (O2)	(O1-B)/B [%]	(O2-B)/B [%]	(O2-O1)/O1 [%]
Range [km]	4393.9	4552.9	4537.2	+3.62	+3.26	-0.34
Semi-span [m]	4.00	4.85	4.75	+21.25	+18.75	-2.06
Chord [m]	0.765	0.631	0.644	-17.52	-15.82	+2.06
Aspect ratio	10.45	15.39	14.78	+47.27	+41.44	-3.96
Upper skins [kg]	20.06	20.09	19.33	+0.17	-3.64	-3.80
Lower skins [kg]	11.48	13.03	13.12	+13.52	+14.26	+0.65
Front spar [kg]	2.31	2.99	3.45	+29.43	+49.21	+15.29
Rear spar [kg]	1.72	1.81	1.99	+5.44	+15.63	+9.66
Ribs [kg]	1.83	1.24	1.33	-32.07	-27.53	+6.68
Total [kg]	37.40	39.17	39.21	+4.74	+4.83	+0.09

design being slightly heavier, the internal mass distribution is different between the two designs. To understand where this difference is coming from, we can compare the structural sizing of those two optimal designs.

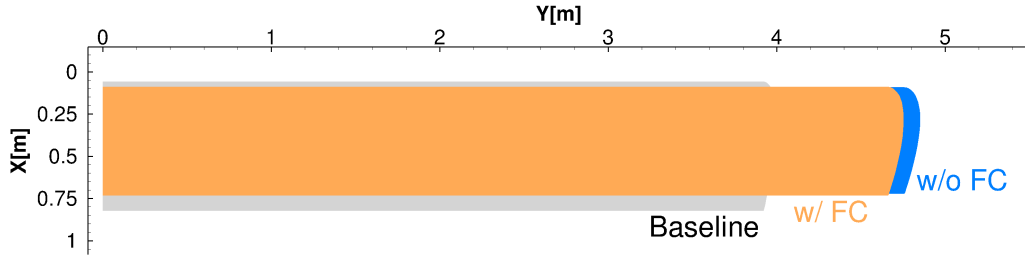


Figure 6.8: Planform comparison of the baseline and the two aerostructural optimal designs, with and without the flutter constraint.

The structural design variables and the failure and buckling element values is shown in Fig. 6.9. Each figure shows three contour plots, where the top and mid are the optimal aerostructural solutions obtained without and with the flutter constraint, respectively. Using a diverging color map, the bottom contour plot shows the difference of those two solutions computed using,  $(\mathbf{x}_{w/FC}^* - \mathbf{x}_{w/oFC}^*)/\mathbf{x}_{w/oFC}^*$  for the stiffener and panel thickness and  $\mathbf{x}_{w/FC}^* - \mathbf{x}_{w/oFC}^*$  for the failure and buckling. The difference in stiffener and panel thickness shown in Figs. 6.9a and 6.9b reveal the following insights for the spar, skins, and rib components.

The front spar optimal stiffener and panel thickness without the flutter constraint

is close to the lower bound for many of the elements, which is not the case for the optimal design with the flutter constraint. In particular, the front spar stiffener and panel thickness is reduced from the mid span towards the root, while from mid span towards the tip it is found to increase. The thickness at the tip is more than double compared to the optimal design obtained without the flutter constraint. However, the increase is not monotonic, as seen on the thickness at the second to last panel at the tip, where the optimizer increases the thickness slightly, compared to the adjacent panels. The rear spar stiffener thickness shows less change between the two optimal designs overall than the front spar. In particular, most elements are close to the lower bound except the root panel, which demonstrates increase in thickness between the designs. For the rear spar panel thickness, the three panels closest to the root increase in thickness, with no other change present.

The skin stiffener and panel thickness is greatest at the root, as expected, and then decreases along the span, reaching the lower bound at the tip. A small reduction in both stiffener and panel thickness is visible for upper and lower skins, except for the skin at the tip, which demonstrates an increase in thickness. Additionally, panel thickness increases on the lower skins at mid span.

All the rib design variables converge to the lower bound, except for the second and third rib from the root, and the one at the tip. Overall the ribs show the smallest change between the two optimal designs. While the ribs close to the root show a slight decrease in thickness, the rib at the tip shows a significant increase in thickness.

The change in thickness distribution between these two optimum designs is solely due to the addition of the flutter constraint. It is clear from the thickness changes, that the optimizer is shifting the center of gravity of the structure forward and outboard by increasing the thickness and thus the mass. The main contribution comes from the front spar, as indicated in Table 6.9, with an increase of over 15% between the two designs. Shifting the center of gravity outboard lowers the natural modes, which may

adversely influence divergence and flutter characteristics. However, shifting the center of gravity of the structure upstream has well-known aeroelastic benefits: It provides stabilizing effects, which help satisfy the flutter constraint [189]. While these design tradeoffs are non intuitive they overall result in an improved divergence and flutter characteristics.

The failure and buckling values for each element of the two optimal designs are shown in Figs. 6.9c and 6.9d. As expected, the lower skins are failure critical and the upper skins, spars, and ribs are buckling critical. The failure values between the two designs increase on the front spar and decrease on the lower skins at the mid span. This is directly linked to the change in thickness of the stiffeners and panels at those locations. Similarly, the difference in buckling is directly related to the change in thickness between the two designs.

The change in thickness between the two optimal solutions is further reflected in the deflection of the aerodynamic surfaces shown in Figure 6.10, for both the cruise and maneuver flight conditions. While the optimal design obtained with the flutter constraints is slightly shorter, it is stiffer and thus deflects less, as shown by the reduced tip deflection. This reduction in deflection is attributed to the increase in thickness of the lower skins and spar thickness changes. Furthermore, these thickness changes affect how mass is distributed in the structure, affecting the gravity loads, which are added in the static aeroelastic analysis.

### **6.4.1 Modal Analysis**

The first eight natural mode shapes of the two optimal aerostructural solutions are shown in Fig. 6.11. The natural mode shape order is the same for the two solutions, except for mode shapes 5 and 6, which trade places. Overall, the natural frequencies are slightly higher for the optimal solution obtained when enforcing the flutter constraint. Since both optimal structures have similar total mass, this is likely due to a

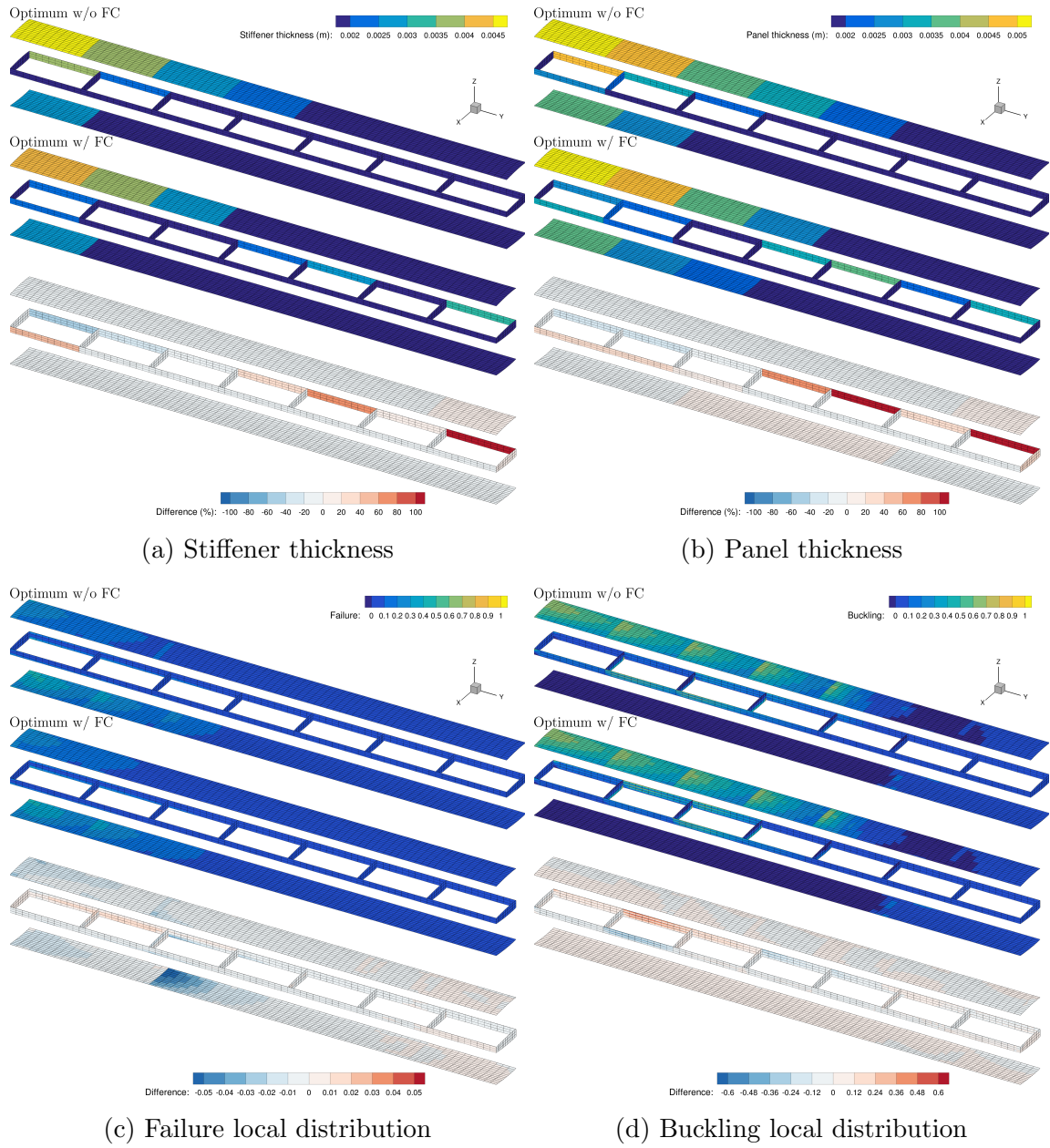


Figure 6.9: Stiffener thickness, panel thickness, local failure, and buckling values shown for the two optimal designs, without and with flutter constraint. A diverging color map highlights the difference between the designs.

combination of a smaller aspect ratio, how mass is distributed, and a stiffer structure.

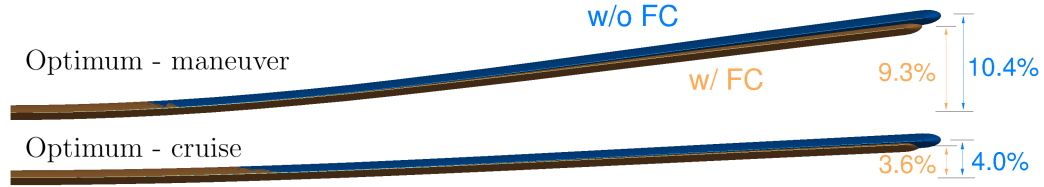
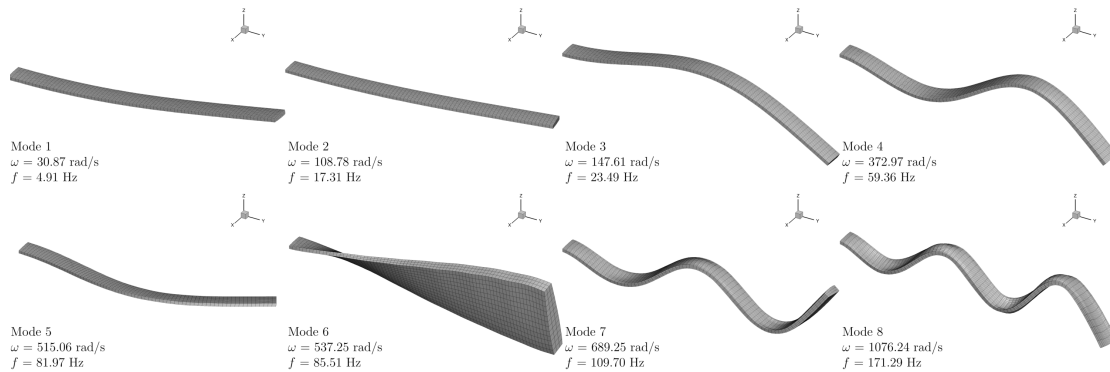
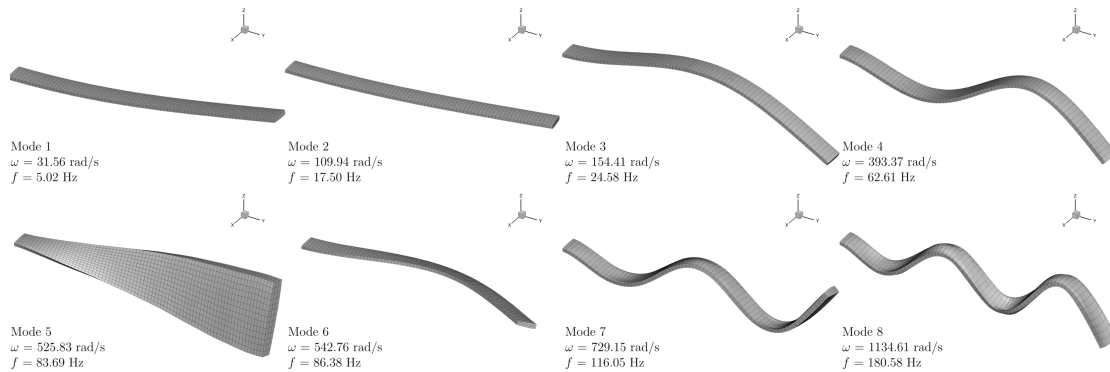


Figure 6.10: Tip deflection comparison between the two optimal solutions under the cruise and maneuver flight conditions. The deflection is normalized with respect to the semi-span.



(a) Optimal natural mode shapes obtained without the flutter constraint.



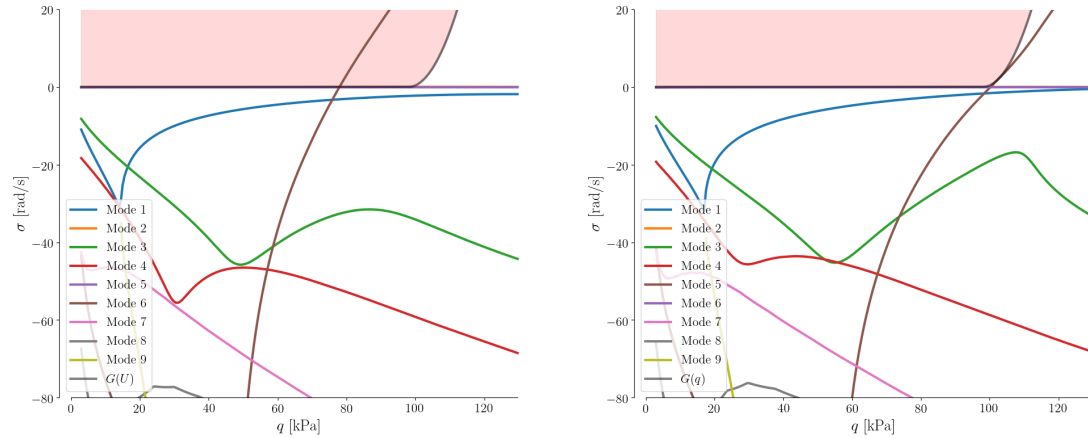
(b) Optimal natural mode shapes obtained with the flutter constraint.

Figure 6.11: First eight natural mode shapes from the two optimal aerostructural solutions obtained without (a) and with (b) the flutter constraint.

## 6.4.2 Flutter Analysis

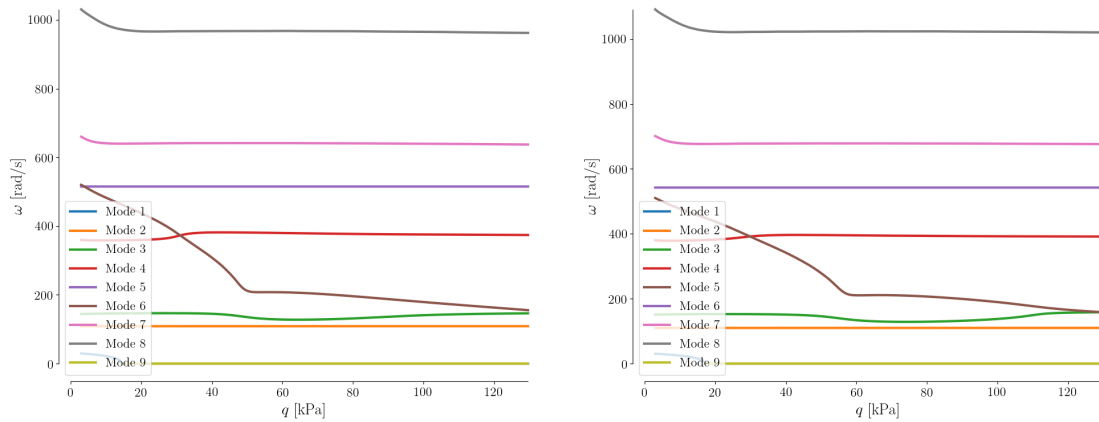
The flutter characteristics of the two optimal aerostructural designs are presented in Fig. 6.12. The optimizer successfully satisfies the flutter constraint by moving the critical mode out of the infeasible space, as shown in Figs. 6.12a and 6.12b. In both

cases, the critical flutter mode is the 1<sup>st</sup> torsion mode, and overall other characteristics are similar. When comparing the frequency characteristics (Figs. 6.12c and 6.12d), we can see that the torsion mode frequency approaches that of the 2<sup>nd</sup> bending mode (mode 3).



(a) Damping without flutter constraints.

(b) Damping with flutter constraints.



(c) Frequency without flutter constraints.

(d) Frequency with flutter constraints.

Figure 6.12: Flutter damping and frequency characteristics of the two aerostructural optimal designs without and with a flutter constraint.



## 6.5 Summary

The proposed flutter constraint is demonstrated successfully in a high-fidelity aerostructural optimization of a representative geometry, and compared against an optimal solution obtained without the flutter constraint. With the flutter constraint we obtain a stiffer, less slender wing, without any significant reduction in range (less than 1% between the two designs). The wing aspect ratio decreases by 4%, and structural design variables show significant differences. The optimizer redistributes the mass by changing the panel and stiffener thicknesses. This changes the flutter characteristics while maintaining roughly the same total mass (less than 0.1% between the two designs). The center of gravity of the structure is moved upstream, which is known to have a stabilizing aeroelastic effect. Finally, the increase in stiffness of the flutter-constrained structure is evident from the reduced tip deflection. Overall, we obtain a feasible and optimal design without significant compromise in range and structural mass.

This wing aerostructural optimization demonstrates the necessity of adding and enforcing a flutter constraint in the high-fidelity aerostructural design process. The flutter-constrained optimum exhibits different shape and sizing features, which alter and improve the wing aeroelastic characteristics. The proposed flutter constraint complements commonly enforced structural constraints such as failure, buckling, and adjacency constraints, making the overall optimal design more robust.

## CHAPTER 7

# Aerostructural Optimization of XRF1

In previous chapter the flutter constraint formulation was demonstrated in a high-fidelity aerostructural optimization on a rectangular idealized wing planform. We showed that such constraint can be applied in the design of a simplified geometry with  $\mathcal{O}(100)$  design variables and  $\mathcal{O}(100)$  constraints. With the flutter constraint, we obtained a stiffer, lower aspect ratio wing with stark differences structural sizing, without a significant reduction in range (objective). These results demonstrated the importance of including flutter constraints in wing design optimization.

The goal of this work is to apply the proposed constraint to a large scale aerostructural optimization of the XRF1 geometry with  $\mathcal{O}(1000)$  design variables, including the aerodynamic shape variables. To do so, two aerostructural optimization are performed, without and with the flutter constraint. The baseline geometry used here is based on the work done previously by Kenway and Martins [207].

This chapter is outlined as follows: Section 7.1 describes the structural and aerodynamic geometry and models used in this study. Following that Section 7.2 describes the point mass implementation and development necessary to model masses such as fuel and engine. Section 7.3 describes the optimization problem formulation, flight conditions, design variables, and constraints. Finally, in Section 7.4 we describe the results and compare and contrasts the design differences between the two optimized designs. Section 7.5 concludes the report with a summary and discussion of the effect

of adding a flutter constraint has on aerostructural optimization.

## 7.1 XRF1 Model

### 7.1.1 Geometry

The XRF1 geometry is representative of a modern transport aircraft configuration, provided by Airbus to its research collaborators. In this work we use a modified wing-body-tail configuration that has been previously developed. These modifications are minimal, but necessary, to accommodate the MACH computational tools, both in terms of structural wingbox and aerodynamic mesh. An in-house wingbox is constructed as well as multiblock RANS CFD mesh. Full set of modifications to the structural and aerodynamic models are detailed in a previous report [207]. The XRF1

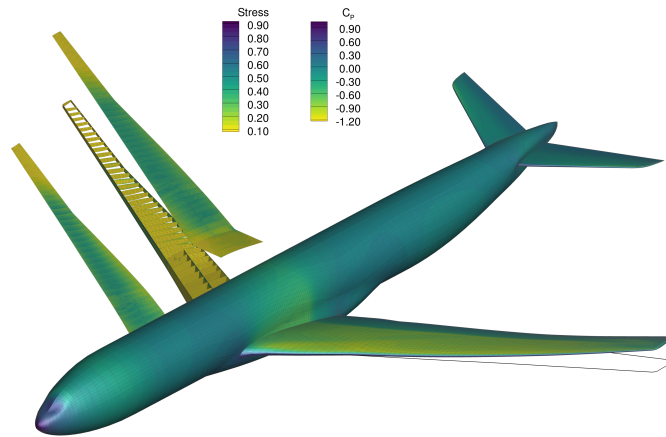


Figure 7.1: XRF1 configuration. The wingbox is shown on the left and the aerodynamic surface on the right. The outline shown is the undeformed shape.

baseline geometry, nominal flight conditions, and weight is summarized in Table 7.1.

Table 7.1: XRF1 Specifications

Parameter	Value	Units
Cruise Mach number	0.83	-
Cruise lift coefficient (nominal)	0.5	-
Cruise altitude	35000	ft
Span	58.0	m
Aspect ratio	9.12	-
Reference wing area	368.8	m <sup>2</sup>
Sweep (1/4 chord)	30.0	°
Design range	8000	nm
Thrust-specific fuel consumption	15.57	g/kN-s
Design payload	23950	kg
Reserve fuel	8000	kg
Fixed weight	97300	kg

### 7.1.2 Aerodynamic Model

Two CFD meshes are employed in this work that are approximately 908k and 282k cells for the cruise and maneuver conditions, respectively. These meshes are coarser than used by Kenway and Martins [207]. The cruise mesh is denoted as L2 and the maneuver mesh is generated specifically for this study and is denoted as L2.5. This L2.5 is a coarsened version of the L1.5 mesh referenced in Kenway and Martins [207]. While the meshes are coarser than used in the original aerostructural optimization, the goal here is not to predict accurate drag estimates. The goal is to demonstrate the recently developed flutter constraint in a large scale high-fidelity design optimization and the effect that such a constraint may have on the optimized design. Furthermore, the CFD computational cost is the most expensive component in an aerostructural optimization, thus using reduced sized meshes shortens turnaround time. The computational domain is a semi-sphere with a symmetry plane defined at the center fuselage.

The DLM mesh location and size is carefully chosen. While the DLM mesh can be used to model the fuselage of the aircraft, we are only interested in the unsteady effects of the wing. This means that any mesh elements inside the fuselage may create unwanted or artificial unsteady effects. Thus, the DLM mesh is set to start at the side of body. The DLM mesh is flat, i.e. parallel to the flow, but follows in general the centerline of the wingbox and aerodynamic surface. The mesh is composed of 3 panels that all have 12 elements in the chordwise direction with 6, 11, and 11 elements in the spanwise direction, a total of 336 elements. The first panel starts at the side of body and ends at the break in the wing. The second panel starts at the break and ends at the outboard break in the wingbox, and the third panel extends to the tip. All panels have a slight dihedral, with the first panel having with the most.

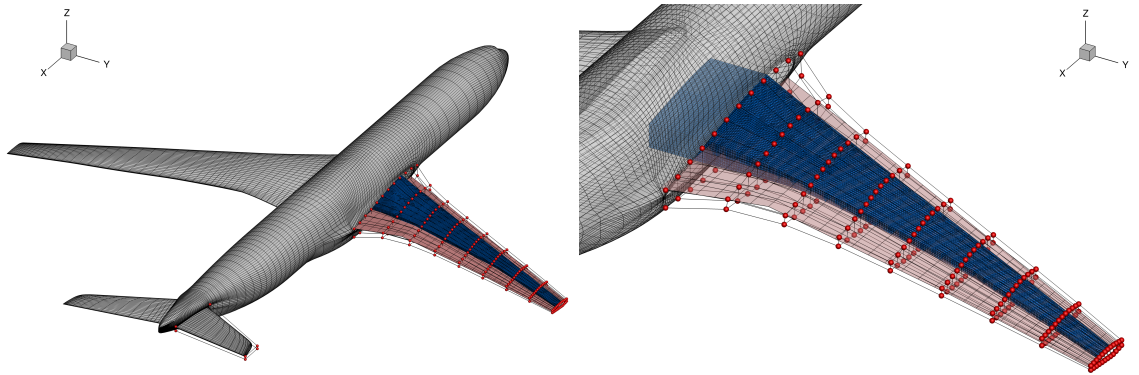
### 7.1.3 Structural Model

The structural model is in-house generated and is the same as used in previous study by Kenway and Martins [207]. The wingbox is discretized using approximately 23k 2<sup>nd</sup> order MITC-shell elements, resulting in approximately 131k DOFs. For improved accuracy a 3<sup>rd</sup> order shell elements could be employed for the same number of elements. This however increases the DOFs approximately by a factor of 4, increasing the overall computational cost, both in terms of aerostructural solutions and modal analysis used for the flutter analysis and is not done here.

While TACS is capable of modeling composites, we use a metallic wingbox, where the material chosen is representative for aluminum AL7075. Material properties, dimensions, and element discretization for the wingbox structure are summarized in Table 7.2.

Table 7.2: Structure mechanical properties.

Variable	Symbol	Value	Unit
Density	$\rho_s$	2780	kg/m <sup>3</sup>
Modulus of elasticity	$E$	70.0	GPa
Poisson ratio	$\nu$	0.3	-
Yield stress	$\sigma_y$	420	MPa



(a) Cruise CFD surface mesh shown on the left, and DLM and wingbox on the right. (b) Closeup of DLM mesh and the wingbox. Note the DLM mesh starting at the side of body.

Figure 7.2: XRF1 configuration.

#### 7.1.4 Structural Parametrization

The structural parametrization applied here is the smeared stiffness approach discussed in Section 6.1.3. Using the smeared stiffener approach the effect of the implicit stiffeners on a given panel stiffness is modeled directly through the material properties. This approach is computationally efficient and removes the need to model discrete stiffeners explicitly. All components in the wingbox are modeled using a smeared stiffness approach. This includes skins, spars, ribs, and engine mount. For more details, see Section 6.1.3.

## 7.2 Mass Element Implementation

In many applications non-structural masses are important to consider due to their impact on the dynamics of the system. In aircraft design such masses could include engine mass, leading and trailing edge devices, and fuel mass. Modeling these non-structural masses can be achieved either by explicit models, or by an approximation. For example, explicit detailed FEM model of the engine and pylon can be approximated and replaced by its rigid body inertial properties and a point mass. The resulting point mass is then connected to the wingbox by either other simplified elements, such as beam elements, or more frequently, rigid body elements (e.g. RBE3s in NASTRAN). Despite the name, the RBE3 does not add any stiffness to the structure and behaves like an interpolation element.

To address this need of modeling non-structural masses in TACS, a mass element in TACS is implemented. A brief element definition is first presented, followed by verification against NASTRAN. Then fuel and engine mass modeling capabilities and implementation is discussed. Finally, we discuss and demonstrate this new capability in TACS with simple use cases.

### 7.2.1 Element Definition

A point mass is placed at an arbitrary location in space as shown in Fig. 7.3. The center of gravity of this point mass is located at  $CG$  and has a mass of  $m$ . This point mass could represent a rigid body with mass moments of inertia tensor  $\mathbf{I}_{CG}$ , which is computed in the  $x' - y' - z'$  coordinate system. Defining  $g$  as a grid point in the FEM mesh, we compute the delta displacement from the center of gravity as  $\mathbf{d} = \mathbf{r}_{CG} - \mathbf{r}_g$ . The distance from the node  $g$  to the  $CG$  is constant meaning that derivatives are zero and thus not computed. For 6 DOF the local mass matrix for the point mass can be

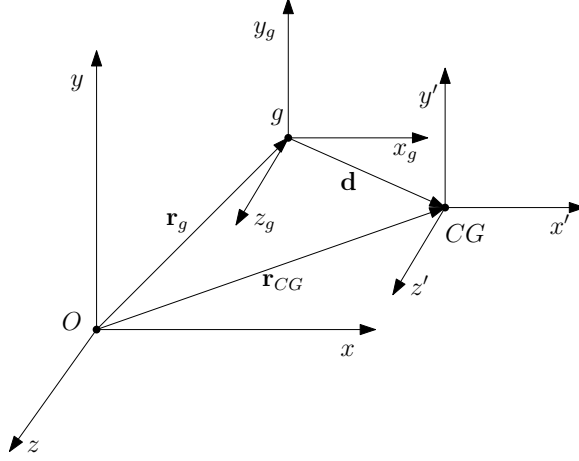


Figure 7.3: Point mass in space relative to global coordinate system and a grid node reference frame.

written as,

$$\mathbf{M}_l = \left[ \begin{array}{c|c} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \hline \mathbf{M}_{21} & \mathbf{M}_{22} \end{array} \right], \quad (7.1)$$

where  $\mathbf{M}_{21} = \mathbf{M}_{12}^T$  due to symmetry. The first matrix is simply the acceleration at  $g$ ,

$$\mathbf{M}_{11} = m \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (7.2)$$

The sum of moments about point  $g$  can be written as the time derivative of angular momentum

$$\Sigma \mathbf{M}_g = \dot{\mathbf{H}}_g, \quad (7.3)$$

where angular momentum can be written as,

$$\mathbf{H}_g = \mathbf{d} \times m\mathbf{v} = \mathbf{d} \times m\dot{\mathbf{d}}. \quad (7.4)$$



Then by differentiating we have

$$\Sigma \mathbf{M}_g = \dot{\mathbf{H}}_g = \dot{\mathbf{d}} \times m \dot{\mathbf{d}} + \mathbf{d} \times m \ddot{\mathbf{d}} \quad (7.5)$$

$$= \mathbf{d} \times m \ddot{\mathbf{d}}, \quad (7.6)$$

where we have used that  $\dot{\mathbf{d}} \times m \dot{\mathbf{d}} = m(\dot{\mathbf{d}} \times \dot{\mathbf{d}}) = 0$ . We can write the cross product as a skew symmetric matrix,

$$\Sigma \mathbf{M}_g = \dot{\mathbf{H}}_g = m \begin{vmatrix} \hat{\mathbf{i}} & \hat{\mathbf{j}} & \hat{\mathbf{k}} \\ d_1 & d_2 & d_3 \\ \ddot{d}_1 & \ddot{d}_2 & \ddot{d}_3 \end{vmatrix} \quad (7.7)$$

$$= m \left[ (d_2 \ddot{d}_3 - d_3 \ddot{d}_2) \hat{\mathbf{i}} - (d_1 \ddot{d}_3 - d_3 \ddot{d}_1) \hat{\mathbf{j}} + (d_1 \ddot{d}_2 - d_2 \ddot{d}_1) \hat{\mathbf{k}} \right] \quad (7.8)$$

$$= m \begin{bmatrix} 0 & d_3 & -d_2 \\ -d_3 & 0 & d_1 \\ d_2 & -d_1 & 0 \end{bmatrix} \begin{bmatrix} \ddot{d}_1 \\ \ddot{d}_2 \\ \ddot{d}_3 \end{bmatrix}, \quad (7.9)$$

where  $d_1, d_2, d_3$  are the  $x, y, z$  components of  $\mathbf{d}$ . We thus obtain the mass matrix as,

$$\mathbf{M}_{21} = m \begin{bmatrix} 0 & d_3 & -d_2 \\ -d_3 & 0 & d_1 \\ d_2 & -d_1 & 0 \end{bmatrix}, \quad (7.10)$$

and since  $\mathbf{M}_{21} = \mathbf{M}_{12}^T$  we have,

$$\mathbf{M}_{12} = m \begin{bmatrix} 0 & -d_3 & d_2 \\ d_3 & 0 & -d_1 \\ -d_2 & d_1 & 0 \end{bmatrix}. \quad (7.11)$$

The angular momentum of a rigid body in 3D space about point  $g$  can be written

as,

$$\mathbf{H}_g = \mathbf{I}_g \boldsymbol{\omega}, \quad (7.12)$$

where  $\mathbf{I}_g$  is mass moment of inertia tensor, written in matrix form. We can write the last mass matrix as,

$$\mathbf{M}_{22} = \begin{bmatrix} (I_{xx})_g & -(I_{xy})_g & -(I_{xz})_g \\ & (I_{yy})_g & -(I_{yz})_g \\ sym. & & (I_{zz})_g \end{bmatrix}, \quad (7.13)$$

where the diagonal terms are defined as,

$$(I_{xx})_g = I_{xx} + m(d_2^2 + d_3^2) \quad (7.14)$$

$$(I_{yy})_g = I_{yy} + m(d_1^2 + d_3^2) \quad (7.15)$$

$$(I_{zz})_g = I_{zz} + m(d_1^2 + d_2^2) \quad (7.16)$$

$$(7.17)$$

and the off diagonal terms are defined as,

$$(I_{xy})_g = (I_{yx})_g = I_{xy} + md_1d_2 \quad (7.18)$$

$$(I_{xz})_g = (I_{zx})_g = I_{xz} + md_1d_3 \quad (7.19)$$

$$(I_{yz})_g = (I_{zy})_g = I_{yz} + md_2d_3 \quad (7.20)$$

$$(7.21)$$

Here  $I_{xx}, I_{yy}, I_{zz}, I_{xy}, I_{xz}, I_{yz}$  are the components of  $\mathbf{I}_{CG}$  the mass moment of inertia of the rigid body computed in the  $x' - y' - z'$  coordinate system.

The full local mass matrix is thus defined as,

$$\mathbf{M}_l = \left[ \begin{array}{c|c} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{array} \right] = \left[ \begin{array}{ccc|ccc} m & 0 & 0 & 0 & -md_3 & md_2 \\ & m & 0 & md_3 & 0 & -md_1 \\ & & m & -md_2 & md_1 & 0 \\ \hline & & & (I_{xx})_g & -(I_{xy})_g & -(I_{xz})_g \\ & & & & (I_{yy})_g & -(I_{yz})_g \\ & & & & & (I_{zz})_g \end{array} \right]. \quad (7.22)$$

## 7.2.2 Implementation Verification

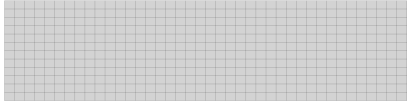


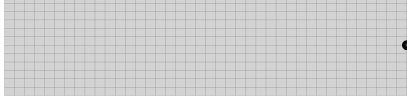
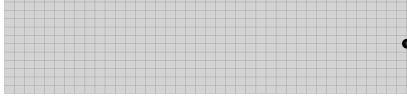
The verification geometry selected is a cantilevered (left side) flat plate. The structure span is 0.85 m and chord 0.21 m and is discretized with 480 2<sup>nd</sup> order MITC shell elements. Material is aluminum with a uniform thickness of 1 mm.

We perform 4 modal analyses, where the mass is placed either on a node or away from the plate. In all cases the mass is 1 kg and we assume this to be a point mass, i.e. we assume  $\mathbf{I}_{CG} = 0$  in the mass moment of inertia tensor  $\mathbf{I}_g$ . In addition, we run one modal analysis without any mass added. The purpose of this is to establish a baseline difference between the solvers. Results are compared with a reference output obtained by NASTRAN in Table 7.3. The point mass is shown with a red sphere. The black sphere represents the node which the mass is associated with and the delta  $\mathbf{d}$  is computed from. Regardless of the placement of the mass, either directly on a node or away from the plate, results agree well with NASTRAN and show similar differences as without a mass.

## 7.2.3 Fuel and Engine Mass Implementation

In the aerostructural optimization we may be interested in including the fuel or engine masses, or both. In the case of the fuel mass, each bay allocated as fuel bay can hold

Table 7.3: Mass element implementation verification results. Flat plate modal analysis results from TACS compare well with NASTRAN.

Test case	#	NASTRAN [Hz]	TACS [Hz]	Rel. Diff. [%]
No mass 	1	1.13	1.14	0.06
	2	7.09	7.11	0.26
	3	9.18	9.27	0.98
	4	19.91	20.03	0.61
	5	28.32	28.64	1.14
	6	39.13	39.58	1.13
	7	49.63	50.36	1.47
	8	64.85	66.03	1.82
Mass placed at node, at wingtip 	1	0.37	0.37	0.03
	2	5.09	5.10	0.15
	3	9.18	9.27	0.98
	4	16.08	16.15	0.43
	5	28.32	28.64	1.14
	6	33.06	33.34	0.85
	7	49.63	50.36	1.47
	8	55.40	56.18	1.40
Mass away from node, $\mathbf{d} = (0.1, 0, 0)$ 	1	0.37	0.37	0.05
	2	4.22	4.24	0.41
	3	8.03	8.08	0.61
	4	15.42	15.59	1.16
	5	22.64	22.88	1.08
	6	32.34	32.85	1.56
	7	41.08	41.66	1.40
	8	53.15	54.15	1.87
Mass away from node, $\mathbf{d} = (0, 0.1, 0)$ 	1	0.32	0.32	0.05
	2	4.01	4.03	0.35
	3	9.18	9.27	0.98
	4	12.43	12.54	0.91
	5	25.78	26.13	1.36
	6	28.32	28.64	1.14
	7	44.90	45.68	1.75
	8	49.63	50.36	1.47
Mass away from node, $\mathbf{d} = (0.1, 0.1, 0)$ 	1	0.32	0.32	0.07
	2	3.62	3.64	0.46
	3	8.39	8.45	0.72
	4	13.10	13.26	1.21
	5	23.49	23.77	1.16
	6	29.98	30.39	1.37
	7	42.66	43.32	1.54
	8	51.09	51.94	1.66

a certain amount of fuel mass based on its volume. This allocation is determined by the user who defines this fuel mass distribution. Using this fuel mass distribution the local fuel volumes are computed for each bay as well as the total fuel volume capacity for the wingbox. In this volume computation, each bay is approximated as a quadrilateral with flat surfaces, as opposed following the curvature of the wingbox, such as the skins. The fuel mass can then be approximated as a point mass at the center of each fuel bay quadrilateral. The local fuel mass is computed based on the local-to-total fuel volume times total fuel mass or,

$$m_{\text{Bay}} = \frac{V_{\text{Bay}}}{V_{\text{Total}}} m_{\text{Total}}. \quad (7.23)$$

Similarly, the engine mass can be approximated by a point mass either with or without mass moment of inertia given with the point mass.

To apply either a fuel or an engine mass to the structure, RBE3s may be used to connect the mass and the nodes or components on the structure. In many situations this is the preferred option. An alternative solution implemented in this work where we divide and split the larger point mass based on some weighting function,  $w_i$ , into smaller, coinciding point masses. Resulting masses are then associated with a single node in the structure. The masses are computed as follows,

$$m_i = w_i m_{\text{object}}, \quad (7.24)$$

where the  $\sum_i m_i$  recovers the mass of the object being modeled. In this case the object could be a fuel or an engine. The weighting function and the split of the mass can be performed in two ways, depending on the application.

### 7.2.3.1 Equal Weights

The first and simplest option is to split the mass equally, i.e.  $w_i = 1/N$ , where  $N$  is the total number of nodes in all components associated with this mass defined by the user. This option is ideal for applying the mass of the fuel to the structure. The user can choose to include any number of components for a given fuel bay, splitting and distribution the larger mass further onto those components. The resulting smaller equal sized masses are then translated directly onto the nodes (i.e.  $\mathbf{d} = \mathbf{0}$ ) of the specified components. By default, only nodes from the lower skin component are used for each bay. This approach is similar to how fuel loads are modeled in a static aeroelastic optimization. For instance, for a 1g cruise condition the fuel loads are applied to the lower skins as traction. Examples of this capability is shown in figure Fig. 7.4, where we assume all bays hold fuel that needs to modeled. Each bay can be treated differently depending on the desired configuration. Verification of this method was already covered in Section 7.2.2.

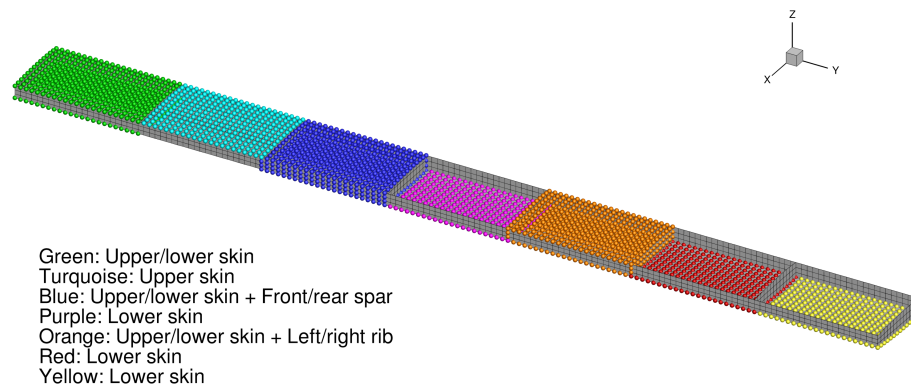


Figure 7.4: Rectangular wingbox. All bays are defined to hold fuel. User configured distribution scheme shown with different colors. Each fuel bay mass has user defined components.

### 7.2.3.2 Inverse Distance Weights

For the second option the weights are computed based on the inverse distance of the objects point mass to each node for the given components. The weights are computed as,

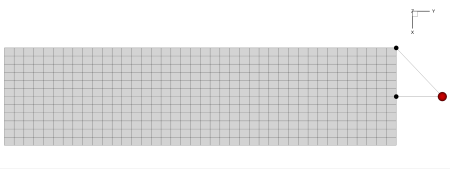
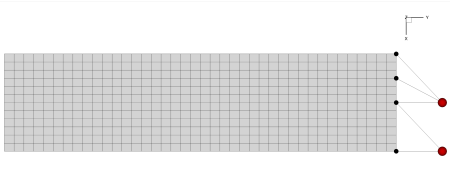
$$w_i = \frac{1}{N-1} \frac{\sum_j d_j - d_i}{\sum_j d_j} = \frac{1}{N-1} \left( 1 - \frac{d_i}{\sum_j d_j} \right), \quad (7.25)$$

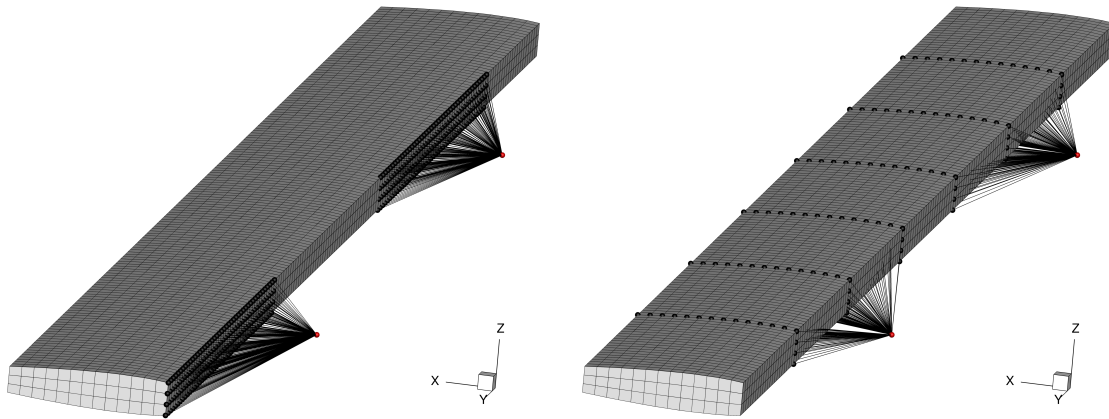
where  $d_i$  is the  $L_2$  norm of  $\mathbf{d}_i$  which is the distance from the mass to each node  $i$ , and  $N$  is the number of nodes the mass is connected to. This means that larger weights are associated with nodes with a smaller distance from the mass. Similar operation can be used to split the moment of inertia tensor if it is provided. This option is more suitable for modeling engine masses away from the structure than fuel mass. As before, the user can choose any number of components that the mass is associated to and influences.

To verify the implementation we use the same flat plate problem as previously used and perform a frequency analysis that we compare with NASTRAN. Two cases are run, one mass connected to two nodes, and two masses, connected to three and two nodes. All masses are 1 kg. The NASTRAN BDF is generated by hand by manually computing the weights and splitting the masses. The results are shown in Table 7.4, and compare favorably with NASTRAN. However, few modes have more error than others, e.g. mode 2 of case 1.

A possible application of the masses is shown in Fig. 7.5. Two masses, engine or leading edge masses, are connected to components selected by the user, here the front spar only and ribs only.

Table 7.4: Flat plate frequency analysis using NASTRAN and TACS with weights computed using the inverse distance.

	#	NASTRAN [Hz]	TACS [Hz]	Rel. Diff. [%]
Case 1: Mass, 1 kg, divided onto 2 nodes 	1	0.32	0.32	0.05
	2	3.53	3.65	3.30
	3	4.12	4.15	0.86
	4	9.19	9.28	0.98
	5	12.64	12.74	0.77
	6	26.31	26.63	1.23
	7	28.36	28.69	1.16
	8	45.72	46.49	1.68
Case 2: Masses, 1 kg each, divided to 3 and 2 nodes 	1	0.23	0.23	0.04
	2	2.20	2.23	1.19
	3	3.53	3.65	3.53
	4	3.98	4.00	0.53
	5	4.87	5.05	3.69
	6	7.47	7.77	3.94
	7	13.05	13.13	0.60
	8	16.45	16.86	2.50



(a) Two engine masses assigned to two front spar elements. (b) Two engine masses assigned to nodes on three ribs each.

Figure 7.5: Examples of the engine or leading edge devices (red sphere) applied to components defined by the user.



## 7.3 Optimization Problem

In this section we describe the optimization problem setup, flight conditions, design variables, and constraints. The optimization problem used here is based on Case 3 in the report by Kenway and Martins [207], which is a multipoint aerostructural wing design optimization problem including planform. The goal of this large scale optimization is to provide a robust design, maximizing its performance for the operational range of interest, while still providing useful performance over a range of other flight conditions.

To evaluate the effect on the optimal design by including and enforcing a flutter constraint, we perform two optimizations, one without and one with the flutter constraint enforced. The details of the optimization formulation, design variables, and constraints are presented below.

### 7.3.1 Objective Function and Flight Conditions

The multipoint optimization objective is to minimize fuel burn for the nominal 8000 nm flight. Fuel burn from taxi, take-off climb, and descent of the flight is ignored. The fuel burn and takeoff-gross-weight (TOGW) are evaluated at the given  $L/D$  for a range of flight conditions. Here we assume the mission is a single cruise-climb stage, but a more correct approach would be to model multiple cruise-climb stages. By rearranging the Breguet range equation, we can use it to compute the fuel burn for each cruise conditions,

$$\begin{aligned} \text{TOGW}_i &= \text{LGW} \exp\left(\frac{R \text{TSFC}}{V_i (L/D)_i}\right) \\ \text{FB}_i &= \text{TOGW}_i - \text{LGW}, \end{aligned} \tag{7.26}$$

where  $\text{FB}_i$  is the fuel burn,  $\text{LGW}$  is the aircraft landing weight,  $R$  is the design mission range,  $\text{TSFC}$  is the thrust-specific fuel consumption,  $V_i$  is the cruise speed,

and  $(L/D)_i$  is the lift-to-drag ratio computed at the  $i$ -th cruise design point. To account for drag from components that are not explicitly modeled, such as engine nacelle, pylon, and vertical tail, a correction of 30 drag counts is added to the drag value computed by the CFD model.

The optimization objective is then a weighted average of all the cruise points,

$$\text{FB}_{\text{obj}} = \sum_{i=1}^{N_{\text{cruise}}} p_i \text{FB}_i, \quad (7.27)$$

where  $p_i$  are the weights for the  $i$ -th cruise condition. In this work we use equal weights  $p_i = 1/5$ . The landing weight in the objective is computed using,

$$\begin{aligned} \text{LGW} = 1.25 \times W_{\text{wing}} + \text{Area Weight} \frac{\text{Wing Area}}{\text{Ref Wing Area}} \\ + \text{Fixed Weight} + \text{Payload} + \text{Reserve Fuel Weight}, \end{aligned} \quad (7.28)$$

where  $W_{\text{wing}}$  is the weight computed by the structural finite-element model. The factor of 1.25 accounts for additional weight associated with fasteners, overlaps, and other components not modeled in the idealized wingbox. The ‘‘Area Weight’’ is the additional mass associated with the leading and trailing edges, as well as the necessary actuation equipment. Since we are varying the planform, this value is linearly scaled by the exposed wing area. The value for area weight is set to 6000 kg.

All operating conditions used in the multipoint aerostructural optimization are given in Table 7.5. They consist of a 5 point cruise stencil, 2 buffet conditions, 3 maneuver conditions, and finally 1 flutter condition.

The objective is computed based on the cruise flight conditions 1-5 only. The nominal design altitude is not specified here since it is made a design variable in the optimization. This allows for the fulfillment of the physical lift constraint independently of the lift coefficient. Cruise flight condition 4 and 5, are specified  $\pm 3000$  ft relative to the design altitude. By including these design conditions the aircraft is

Table 7.5: Flight conditions used in the multipoint aerostructural optimization.

Operating condition	#	Mach	Altitude	$\alpha$	Fuel fraction	Load factor
Cruise	1	0.83	Design	Design <sub>1</sub>	0.50	1.0
	2	0.83	Design	Design <sub>1</sub> + 0.25 deg	0.50	1.0
	3	0.83	Design	Design <sub>1</sub> - 0.25 deg	0.50	1.0
	4	0.82	Design-3000 ft	Design <sub>2</sub>	0.95	1.0
	5	0.84	Design+3000 ft	Design <sub>3</sub>	0.10	1.0
Buffet	6	0.83	Design	Design <sub>4</sub>	0.50	1.3
	7	0.87	Design	Design <sub>5</sub>	0.50	1.0
Maneuver	1	0.78	16000 ft	Design <sub>6</sub>	1.0	2.5
	2	0.78	22000 ft	Design <sub>7</sub>	1.0	-1.0
	3	0.84	27000 ft	Design <sub>8</sub>	1.0	1.0
Flutter	1	0.80	(range)	-	1.0	1.0

designed in a  $C_L$ -Mach space rather than at a single flight condition. This strategy has been shown to produce more robust overall designs, with fuel burn that is less sensitive to varying aircraft weight and flight conditions [154, 208].

The angle of attack,  $\alpha$ , is also made a design variable in order to satisfy the lift constraints. This applies to all flight conditions except for cruise 2 and 3, which are analyzed at angles of attack  $\pm 0.25$  degrees relative to cruise condition 1. This is done to compute the static margin constraint and to estimate  $C_{L_\alpha}$  of the aircraft for the gust computation. Flight conditions 1-3 use a fuel fraction of 0.5, indicating that the analysis point is done approximately half way through the trip, while 4 and 5 use 0.95 and 0.1, signifying the early and late stages of the cruise-climb segment, respectively.

The FAA regulation require that transonic transport aircraft meet buffet onset minimums. Thus, flight conditions 6 and 7 are used to compute the buffet onset, but are not included in the objective computation. Condition 6 is a high loading condition, with load factor of 1.3g at cruise, and conditions 7 is a high Mach condition. The buffet onset is computed base on a separation sensor which estimates separated flow on the wing surface. Computing and enforcing buffet constraints has been shown to be important in improving design robustness [152].

In addition to the design flight conditions, three maneuver flight conditions are specified. The primary function of these conditions is to facilitate sizing the structure such that failure and buckling loads are not exceeded. The conditions used here are a 2.5g pull up maneuver and a -1g push over maneuver. The third maneuver condition is a 1g cruise pseudo gust load case for the FAA Part 23.343/Part 25.341 gust criteria. The main purpose of a gust condition is to size the structure based on a load, which occurs over a short amount of time, such that passive load alleviation is not possible.

Finally, the flutter condition is specified over a range of dynamic pressures, effectively an altitude sweep, at Mach 0.8. As with buffet, the FAA regulation also require that transonic transport aircraft meet flutter requirements. Using the standard atmospheric model the matched points computed here reflect a sweep through altitudes, which corresponds to changes in the speed and density as a function of dynamic pressure. These relations are show in Fig. 7.6. The dynamic pressure range is specified approximately from 0.4 – 35 kPa, and as it increases the altitude reduces, density increases, and speed varies for this fixed Mach number.

### 7.3.2 Design Variables

The design parametrization of the optimization problem defines the design variables that are exposed to the optimization algorithm. The design variables can be split into three types: geometric, aerodynamic, and structural design variables. The geometric parametrization influences simultaneously all aerodynamic surfaces and the structure directly. The aerodynamic and structural variables influence its discipline directly, and indirectly though coupling effects.  $\mathcal{O}(1000)$  design variables are used in this multipoint aerostructural optimization and they are summarized in Table 7.8.

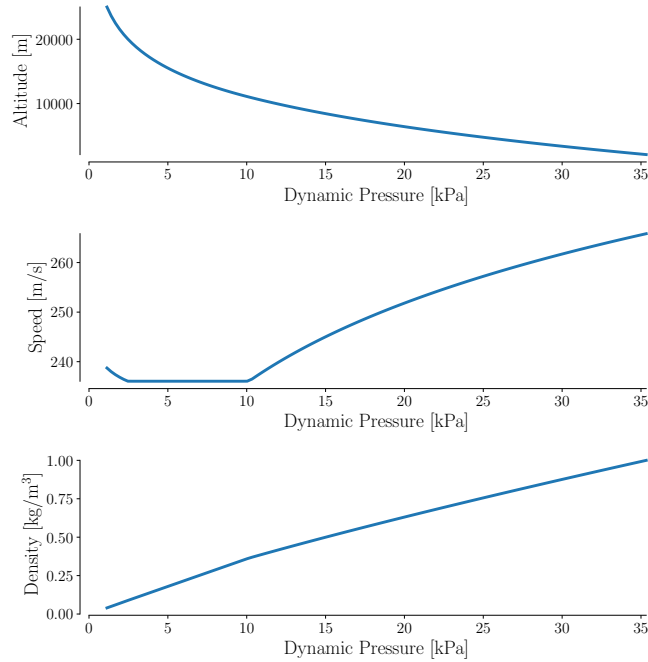


Figure 7.6: Flutter flight conditions.

### 7.3.2.1 Geometric Design Variables

The geometric design variables manipulate all aerodynamic surfaces and the structure directly which have been embedded inside an FFD. These design variables are the FFD b-spline control points and moving them individually, or collectively, we can manipulate the shape and planform, continuously, in great detail.

The wing shape is parametrized using 9 sections of 24 control points, which can be controlled independently, a total of 216 shape design variables. These numbers have been shown provide a good balance between number of shape design variables and optimized design performance [209].

To control the spanwise twist distribution, each of the 9 sections define a local twist design variable. Each twist variable rotates its entire section by simultaneously and rigidly rotating the FFD control points about a central reference axis. For the wing planform we define 4 design variables, chord, span, sweep, and dihedral. These planform variables are applied though the same central reference axis that manipu-

lates all the sections at once. The horizontal tail has its own FFD and reference axis. We define one design variable for the tail rotation, for each flight conditions, which rotates the entire tail. This is necessary to satisfy the trim constraint at a given flight condition during the optimization. Finally, the reference point, about which the pitch moment is computed, is also made a design variable.

Figure 7.7 shows the geometric and structural parameterizations.

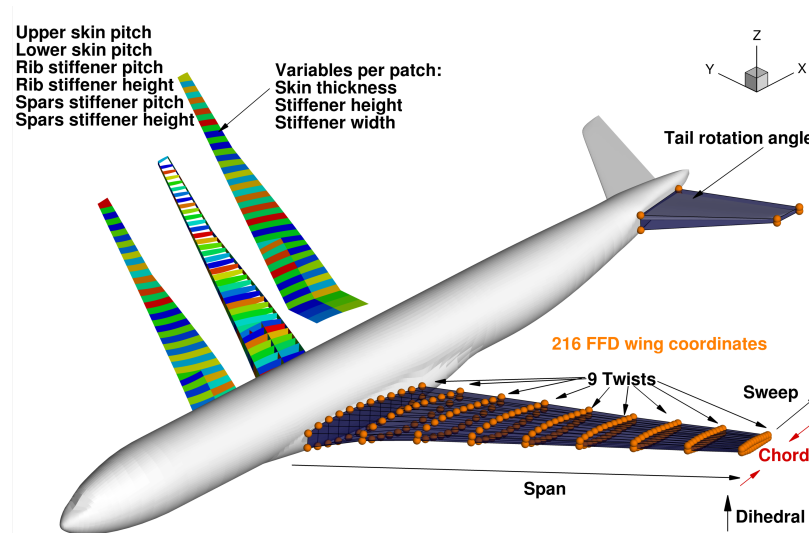


Figure 7.7: Geometric and structural design variables overview.

### 7.3.2.2 Aerodynamic Design Variables

The aerodynamic design variables are the angle of attack  $\alpha$  for each flight condition, and a design altitude  $h$ . The angle of attack is allowed to change and is used to satisfy the lift constraint at a given operating condition, except for cruise points 2 and 3, which are shifted around cruise point 1. Additionally, the tail angle is allowed to change in order to trim the aircraft, enforced through a moment constraint.

For the altitude, only a single master design variable is specified for all cruise conditions. This master altitude influences the on the entire cruise point stencil. All cruise points operate on the same altitude, with cruise points 4 and 5 being offset

by a constant from this master altitude. A Mach number is specified for each flight conditions and is fixed during the optimization.

### 7.3.2.3 Structural Design Variables

The structural parametrization for each component was discussed in detail in Section 7.1.4. In structural design, generally by adding more design variables gives the optimizer greater freedom to size the structure precisely where it is needed and thus reduce the overall weight. However, large number of design variables may increase total optimization solution time. For simplicity and in order to reduce the total number of design variables, we set  $t_b = t_w$ , and  $w_b = h_s$  for all structural components. The remaining design variables, panel thickness  $t_s$ , stiffener height  $h_s$ , stiffener thickness  $t_w$ , stiffener pitch  $b$ , and panel length  $L$  are exposed to the optimizer. However, to further reduce the design variables, some components are grouped together and controlled by a single design variable per group. For instance, one stiffener pitch design variable is used for all components in the upper skins group, and one for the components in the lower skins group. This number of design variables per components grouping is illustrated in Table 7.6. Note that the engine mount is a single component thus is fully parametrized using the 5 design variables. Total number of sizing structural design variables is 934. The skin stiffeners are oriented parallel to the sweep angle, while the rib and spar stiffeners are oriented parallel to the  $z$ -axis. The reference axis direction is fixed throughout the optimization, but should ideally be updated to match the changes in sweep angle.

Fuel loads resulting from the volume in each fuel bay are also added as design variables. The fuel loads are modeled as tractions on the lower surface, and are updated as the volume changes with geometric changes. The fuel bay distribution is shown in Fig. 7.8a. For the flutter analysis the fuel mass in each bay is modeled as point masses which are split and translated onto the lower skin of the structure

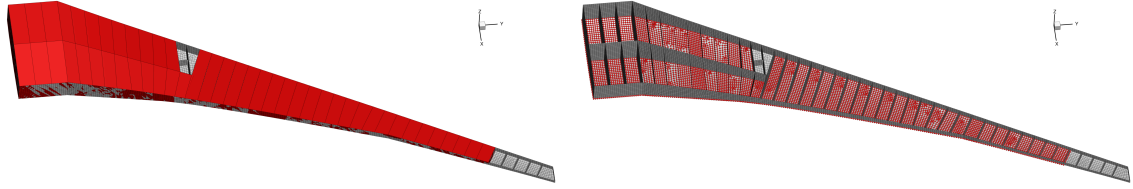
Table 7.6: Structural design variables and their grouping. Note that components in a given group, e.g. upper skin stiffener pitch, is controlled with a single design variable

	Symbol	U/L Skins	F/M/R Spars	Ribs	Engine Mount	Total
Length	$L$	56/56	43/14/42	58	1	270
Stiffener Pitch	$b$	1/1	1/1/1	1	1	7
Stiffener Height	$h_s$	56/56	1/1/1	1	1	117
Stiffener Thickness	$t_w$	56/56	43/14/42	58	1	270
Panel Thickness	$t_s$	56/56	43/14/42	58	1	270
<b>Total</b>		<b>450</b>	<b>303</b>	<b>176</b>	<b>5</b>	<b>934</b>

as shown in Fig. 7.8b. This process is further described in Section 7.2.3. The total fuel mass considered in the optimization is 48750 kg, and in the dynamic analysis is fixed throughout the optimization. While there will be some discrepancy between the final optimized fuel mass and the fixed value used for the dynamic analysis in the optimization, it is usually small such that this a reasonable approximation.

In the static aerostructural analysis the engine loads are included and added directly on the one member of the front spar and the engine mount. These loads remain fixed throughout the optimization. For the flutter analysis considerable effort was spent on trying to include the engine mass using the techniques discussed in Section 7.2.3. However, this resulted in a number of local modes appearing, making this hard to apply in optimization. The engine mass is thus omitted in the flutter computations. However, in terms of total mass, while not insignificant, the engine is an order of magnitude smaller than the fuel mass. Including a mass that is offset from the main body, such as the engine mass, is expected to impact the modal analysis by lowering the torsional natural mode. How significantly this will impact the dynamic analysis and ultimately the final design is a topic of future study.





(a) Fuel volumes used to compute the fuel bay mass. (b) Fuel masses split for each bay translated onto lower skins.

Figure 7.8: Fuel mass for each bay split into multiple point masses that are then translated onto lower skins, show in red.

### 7.3.3 Constraint Definition

For the optimization to give a design that is both physical and meaningful, a number of constraints need to be defined. In this section we describe all the constraints used in this work.  $\mathcal{O}(1000)$  constraints are used in this multipoint aerostructural optimization and they are summarized in Table 7.8.

#### 7.3.3.1 Geometric Constraints

A number of geometric constraints are added to ensure the final shape is realistic and lends itself to manufacturing processes. A constraint is placed on the leading edge radius that ensures no reduction beyond the initial value. Omitting such constraint has been shown to result in unrealistic, sharp leading edges [25]. Furthermore, for low-speed high-lift conditions, such as landing, sharp leading edges may compromise performance. The trailing edge thickness is constrained in a similar manner, preventing any reduction from the initial value. This is done to prevent very thin trailing edges, which may lead to manufacturing and structural issues.

Finally, the leading and trailing edge FFD control points are constrained to move in an equal and opposite  $z$  direction. As mentioned in the previous section a twist variable is defined at each station, which performs twist by a nonlinear rotation. These linear constraints are necessary to prevent the shape variables emulating a shearing

twist and guarantee that these the shape and twist variables are independent.

### 7.3.3.2 Aerodynamic Constraints

We apply lift constraints for all cruise and maneuver conditions, except for cruise points 2 and 3. For the enforced lift constraints we require the physical lift to be equal to the aircraft weight times a given flight condition load factor. The aircraft weight is computed using,

$$W = LGW + \gamma_i FB_1, \quad (7.29)$$

where the  $\gamma_i$  is the fuel fraction for the  $i$ -th flight condition and  $FB_1$  is the fuel burn from the first cruise condition. Furthermore, we require that the aircraft is trimmed for the same cruise and maneuver points. Separation based buffet constraints are added to ensure that flow remains attached for both the buffet and maneuver flight conditions. Here we constrain the separation to be less than 4% of the reference area.

Longitudinal stability constraint is added based on static margin. The static margin is computed for each constraint using,

$$\text{sm} = -\frac{C_{M_\alpha}}{C_{L_\alpha}} = -\frac{\partial C_M / \partial \alpha}{\partial C_L / \partial \alpha} \quad (7.30)$$

where the derivatives  $C_{M_\alpha}, C_{L_\alpha}$  are estimated with finite differences. Since we have two perturbed flight conditions, we can form two constraints using data from the nominal flight condition 1 and then the perturbed angle of attack cruise conditions 2 and 3. A lower limit of 10% is enforced for both constraints, which is based on the cruise CG position of 35% MAC.

### 7.3.3.3 Structural Constraints

Number of constraints are applied to the structure, linear and nonlinear, which apply to performance or manufacturing, or both. During optimization, panel-level buckling

and failure (stress) KS aggregation constraints are enforced under a given maneuver loading conditions. The KS aggregation technique is important to minimize the number of constraints used in the optimization. For the 2.5g maneuver and 1g cruise conditions we apply four KS material failure constraints, one for each group: ribs, spars, upper skins, and lower skins. Three buckling KS aggregation constraints are applied for these flight condition, one for each group: ribs, spars, and upper skins. A buckling constraint for the lower skins is not necessary as these components are always in tension for the respective flight conditions. For the remaining flight condition, -1g, we enforce three buckling constraints, one for each group: ribs, spars, and finally the lower skins. No failure constraints are necessary for this flight conditions since this condition is not critical for sizing. The KS aggregation parameter for all functions is set to a recommended value of  $\rho_{\text{KS}_{\text{stress}}} = \rho_{\text{KS}_{\text{buckling}}} = 100$  [26]. These KS constraints are generally difficult for the optimizer to satisfy due to their highly nonlinear behavior. In addition to the maneuver load factor, a safety factor of 1.5 is applied in the failure computation as required by regulations. For the -1g case an additional safety factor of 1.95 is used.

Table 7.7: Buckling and failure KS aggregation constraints used for each maneuver flight condition.

	2.5g		-1g		1g	
	Buckling	Failure	Buckling	Failure	Buckling	Failure
Ribs	•	•	•		•	•
Spars	•	•	•		•	•
Upper Skins	•	•			•	•
Lower Skins		•	•			•

Linear adjacency constraints are enforced to limit large variation in the design variables between two adjacent panels. This constraint is applied on the panel and stiffener thickness design variables for the spars and the skins. Finally, as discussed in

Section 7.1.4, nonlinear consistency constraint is added for the panel length to match geometric length.

Fuel constraints are included to ensure that the structure internal volume is sufficient to store the necessary fuel, including reserve fuel, for the mission. Note the interesting implications of such constraint. By reducing the fuel burn, less fuel volume is needed, reducing the wing thickness and drag, which in turn reduces the fuel burn. The fuel bays are shown in Fig. 7.8a.

Finally, the previously described flutter constraint is enforced only in one of the two aerostructural optimization to study its effect on the wing design.

For the bounding curve parameters used in this work we set  $g^* = 0$  rad/s,  $g^+ = 0.02$  rad/s,  $q^* = 20$  kPa, and  $\beta = 10^{-7}$  rad/(Pa<sup>2</sup> · s). The combination of parameters set the implicit minimum flutter speed to 20kPa and limit any hard flutter modes governed by the small parabolic growth parameter  $\beta$ . The  $g^+ = 0.02$  parameter shifts the entire boundary vertically into the positive damping space. In particular, these values are chosen such that in-plane modes do not affect the optimization. No further attention is required regarding in-plane modes beyond what was previously defined. The optimizer should size the structure such that the in-plane modes remain in-plane for the flight envelope of interest. The flutter KS aggregation parameter is set to  $\rho_{\text{KS}_{\text{flutter}}} = 500$ . These values are chosen based on the discussion in Sections 3.4 and 3.5 to compensate for the conservative behavior of the KS function.

The complete problem consists of around 1200 design variables and 1200 constraints. While we have numerous constraints, the majority are linear constraints. The optimization problem statement is summarized in Table 7.8. As mentioned previously, we perform two aerostructural optimizations, one without and one with the flutter constraint active, in an otherwise identical setup.

Table 7.8: Aerostructural optimization formulation.

	Function/variable	Description	Quantity	
minimize with respect to	Fuel Burn			
	$x_{\text{airfoil}}$	FFD control points	216	
	$x_{\text{twist}}$	Wing twist	9	
	$x_{\text{alpha}_i}$	Angle of attack for each flight condition	8	
	$x_{\eta_i}$	Tail rotation angle for each flight condition	8	
	$x_{x_{\text{ref}}}$	Reference point in $x$ about which pitch moment is computed	1	
	$x_{\text{altitude}}$	Nominal design altitude	1	
	$x_{\text{span}}$	Span	1	
	$x_{\text{sweep}}$	Sweep	1	
	$x_{\text{dihedral}}$	Dihedral	1	
	$x_{\text{chord}}$	Full chord scaling	1	
	$x_{\text{panel length}}$	Panel length skins/spars/ribs/engine	270	
	$x_{\text{stiff pitch}}$	Panel stiffener pitch skins/spars/ribs/engine	7	
	$x_{\text{stiff height}}$	Panel stiffener height skins/spars/ribs/engine	117	
	$x_{\text{stiff thick}}$	Panel stiffener thickness skins/spars/ribs/engine	270	
	$x_{\text{panel thick}}$	Panel thickness skins/spars/ribs/engine	270	
	$x_{\text{fuel traction}}$	Fuel loads/traction per bay	49	
	$x_{\text{fuel mass}}$	Total fuel mass	1	
	subject to	$L_i = n_i W$	Lift constraint	8
		$C_{M_{y_i}} = 0.0$	Trim constraint	8
		$\text{sep} \leq 4\% A_{\text{ref}}$	Amount of separation at buffet or maneuver	5
Static margin $\geq 10\%$		Sufficient static margins	2	
$x_{x_{\text{ref}}} - (x_{LE} + 0.35\text{MAC}) = 0$		Reference trim point location in $x$ direction	1	
$t_{LE}/t_{LE_{\text{init}}} \geq 1.0$		Leading edge radius	20	
$t_{TE}/t_{TE_{\text{init}}} \geq 1.0$		Trailing edge thickness	20	
$\Delta z_{TE, \text{upper}} = -\Delta z_{TE, \text{lower}}$		Fixed trailing edge	8	
$\Delta z_{LE, \text{upper}} = -\Delta z_{LE, \text{lower}}$		Fixed leading edge	8	
$L_{\text{panel}} - x_{\text{panel length}} = 0$		Target panel length	270	
$\text{KS}_{\text{buckling}} \leq 1$		2.5 g buckling	3	
$\text{KS}_{\text{stress}} \leq 1$		2.5 g yield stress	4	
$\text{KS}_{\text{buckling}} \leq 1$		1.0 g gust buckling	3	
$\text{KS}_{\text{stress}} \leq 1$		1.0 g gust yield stress	4	
$\text{KS}_{\text{buckling}} \leq 1$		-1.0 g buckling	3	
$ x_{\text{panel thick}_i} - x_{\text{panel thick}_{i+1}}  \leq 0.0005$		Skin thickness adjacency	217	
$ x_{\text{stiff thick}_i} - x_{\text{stiff thick}_{i+1}}  \leq 0.0005$		Stiffener thickness adjacency	217	
$ x_{\text{stiff height}_i} - x_{\text{stiff height}_{i+1}}  \leq 0.0005$		Stiffener height adjacency	217	
$x_{\text{stiff thick}} - x_{\text{panel thick}} < 0.0025$		Maximum stiffener-skin difference	186	
$\mathcal{V}_{\text{wing}} > \mathcal{V}_{\text{fuel}}$		Minimum fuel volume	1	
$\text{KS}_{\text{flutter}} \leq 0$		KS aggregate of $g_{G,ij}$	1	

## 7.4 Optimization Results

The initial design, i.e. shape, aerodynamic, and structural sizing is based on Case 2 presented by Kenway and Martins [207]. First an aerostructural optimization with only one cruise and maneuver condition is performed. This strategy provides a good

initial high-aspect ratio design for the full stencil optimization, and is computationally cheaper than the full stencil. The full stencil aerostructural optimization without the flutter constraint then starts from this design, reaching the optimized design in approximately 17 hours using 648 processors. Each cruise and maneuver condition use 72 and 48 cores, respectively. The optimization with the flutter constraint is then started from the previous optimized design and completes in approximately 14 hours, with the same number of processors used as before.

The optimization history for both cases is presented in figure Fig. 7.9. The ob-

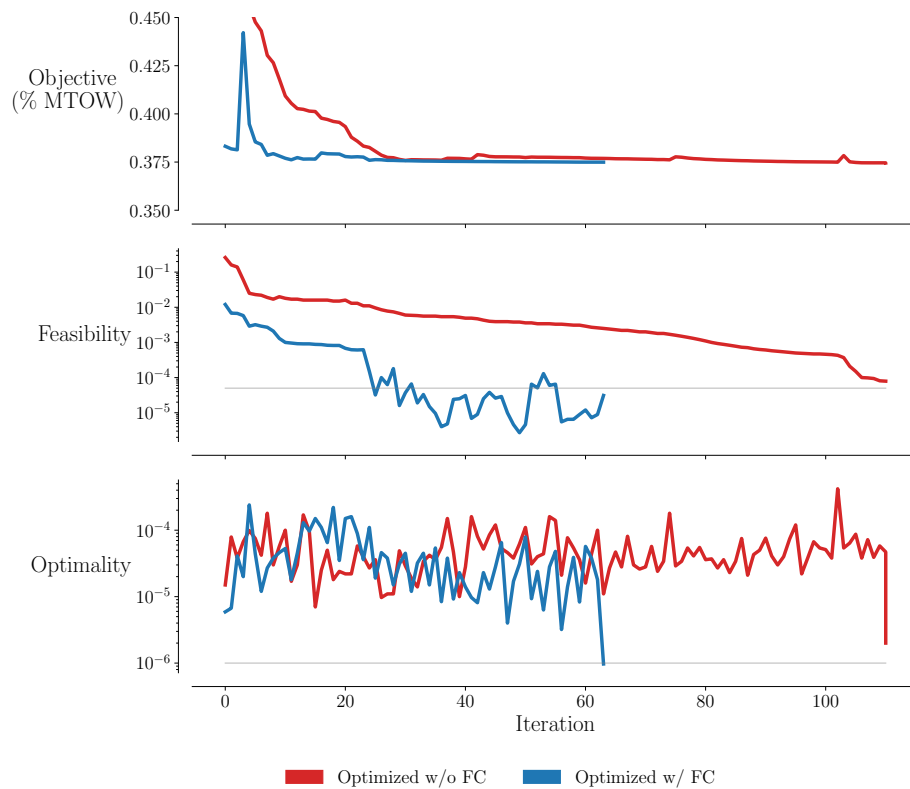


Figure 7.9: Optimization convergence history.

jective (fuel burn) is scaled and presented here as a fraction of the MTOW. The feasibility is a measurement on the constraint violation and the optimality indicates how well the first-order KKT condition is satisfied. The feasibility and optimality for each case is set to  $5 \times 10^{-5}$  and  $10^{-6}$ , respectively, shown with a gray line in the figure. It can be observed that in both cases the fuel burn is reduced while making

the constraints feasible. In the case of the optimization without the flutter constraint the optimizer comes close to the specified tolerances, but is unable to reach them and exits. The optimization with the flutter constraint meets the specified tolerances and exits successfully.

The two optimal designs and the baseline are compared in Table 7.9. Without any significant increase in fuel burn, we obtain a stiffer, heavier, lower aspect ratio wing design, with the structural sizing showing the largest differences between the two designs. While some of these differences are not substantial, they nonetheless highlight the importance of including a flutter constraint in the design optimization process.

Table 7.9: Comparisons of the two optimized designs, without and with the flutter constraint active.

	w/o FC	w/ FC	Diff.	Rel Diff. %
Fuel burn [kg]	92494.09	92602.41	108.33	0.12
AR	11.97	11.90	-0.07	-0.61
Semi span [m]	34.47	34.41	-0.06	-0.18
MAC [m]	6.40	6.43	0.03	0.41
Projected Area [m <sup>2</sup> ]	178.42	178.82	0.41	0.23
Wimpress Area [m <sup>2</sup> ]	198.47	198.97	0.51	0.25
Trapezoidal Area [m <sup>2</sup> ]	196.75	197.25	0.50	0.25
Taper ratio	0.26	0.26	0.00	0.31
1/4 chord Sweep [°]	34.73	34.68	-0.05	-0.14
LE Sweep [°]	36.78	36.74	-0.04	-0.10
Average cruise $L/D$	23.71	23.79	0.07	0.30
Fuel mass [kg]	49920.35	49980.84	60.49	0.12
Wing mass (total) [kg]	43246.61	43915.19	668.58	1.55

### 7.4.1 Design Differences

Overall planform changes for both optimized design are similar and show minor differences. In both cases, the optimizer increases the aspect ratio from the initial design, to minimize the induced drag, which in turn decreases the fuel burn. For the flutter

constrained design a lower aspect ratio is obtained. This design has a smaller span, but a larger MAC, resulting in an overall larger area.

Through these geometric differences a marginally larger  $L/D$  ratio is obtained for the design where the flutter constraint is active. While a larger  $L/D$  impacts the objective favorably, the final fuel burn is larger, indicating that the fuel burn is penalized by the increased mass of the structure. Fuel mass is thus increased to compensate for the increased fuel burn of the mission. The reduced wing sweep is also an indication that the wing is heavier and the optimizer unsweeps the wing in order to reduce structural weight.

The increase in fuel burn is primarily attributed to the increase in wingbox mass. The wingbox mass distribution, per group of components is given in Table 7.10. The

Table 7.10: Structural mass by groups of components for both optimized designs. All values are given in kg.

	w/o FC	w/ FC	Diff.	Rel Diff. %
Upper skin	6111.61	6222.83	111.22	1.82
Lower skin	4966.69	5075.53	108.84	2.19
Ribs	1666.91	1684.73	17.82	1.07
Front spar	602.34	616.16	13.82	2.29
Mid spar	862.74	868.89	6.15	0.71
Rear spar	280.15	283.34	3.19	1.14
Engine Mount	8.73	8.72	-0.02	-0.20
Total	14499.18	14760.20	261.02	1.80

flutter constrained optimized design is 1.8% heavier, with each component group showing an increase in structural mass, except for the engine mount. The group with the largest increase in structural mass is the front spar, which increases by approximately 2.3%.

By investigating the structural design variables directly we gain a better understanding and insights into where the optimizer chooses to increase material thickness.



Figure 7.10 shows the equivalent thickness, where the top and mid are the optimal aerostructural solutions obtained without and with the flutter constraint, respectively. Using a diverging color map, the bottom contour plot shows the difference of those two solutions computed using,  $(\mathbf{x}_{w/FC}^* - \mathbf{x}_{w/oFC}^*)/\mathbf{x}_{w/oFC}^* * 100$ . By investigating the difference plot we see that the largest change occurs in the outboard section of the wing, where the optimizer doubles the lower skin thickness. The upper skin is increased substantially also in a similar location. Other panels show less drastic changes. Plots for individual design variable, the stiffener pitch, stiffener height, stiffener thickness, and panel thickness, are provided in the in Figs. C.1 and C.2 in Appendix C.

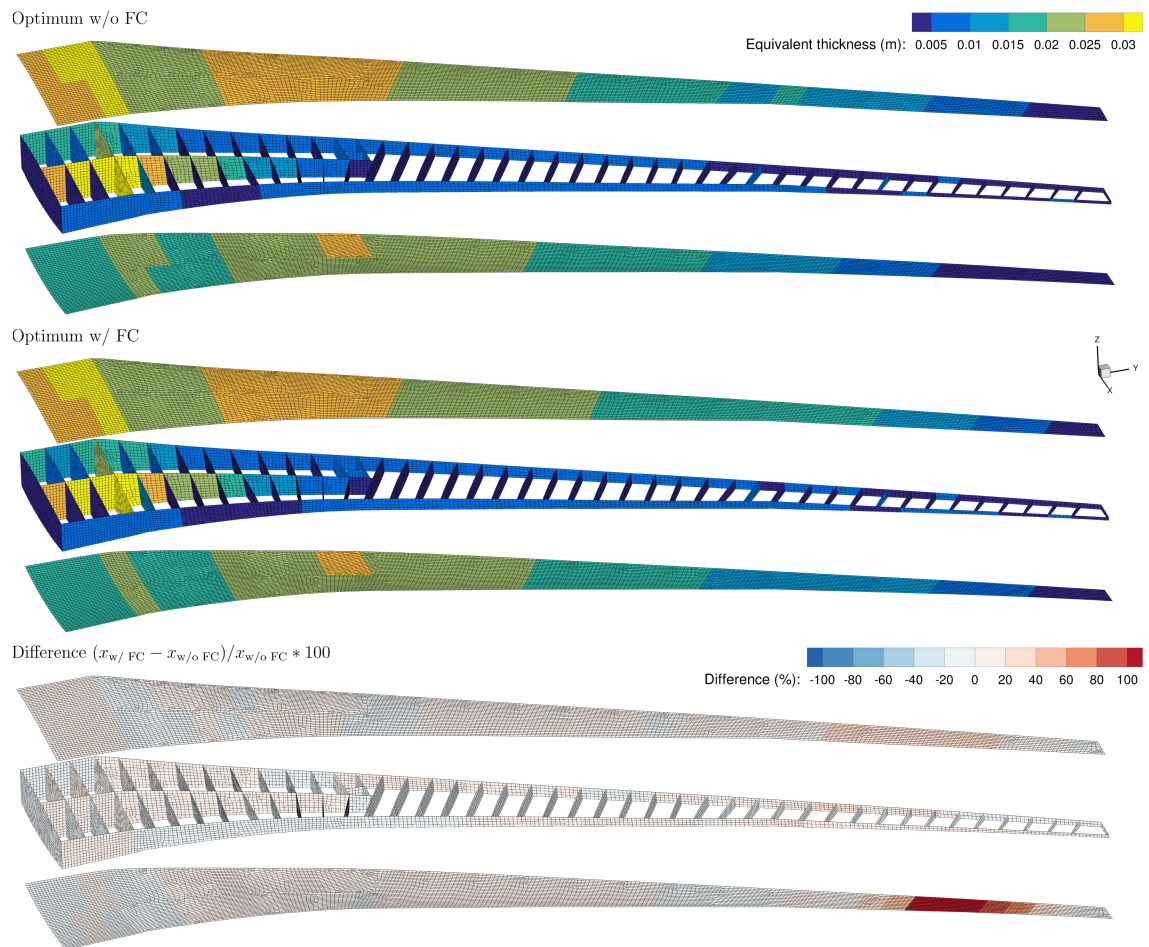


Figure 7.10: Equivalent thickness shown for the two optimal designs, without and with flutter constraint. A diverging color map highlights the difference between the designs.

The change in thickness distribution between these two optimized designs is solely due to the addition of the flutter constraint. It is clear from these thickness changes, that the center of gravity of the structure is shifted forward and outboard, which directly impacts the wing flutter characteristics. For instance, shifting the center of gravity outboard lowers the natural modes, which may adversely influence divergence and flutter characteristics. However, shifting the center of gravity of the structure upstream has well-known aeroelastic benefits: It provides stabilizing effects, which help satisfy the flutter constraint [189]. The increased thickness results in a stiffer structure that may further improve stabilizing effects. While some of these design tradeoffs are non-intuitive, they ultimately result in a flutter free design.

The failure and buckling for each component group is reported in Table 7.11. Active constraints are shown with solid circle and inactive with an empty circle. As expected for the 2.5g maneuver case, all buckling and failure constraints are active. For the -1g case only the lower skins are buckling critical, while the ribs and spars are inactive. Finally, the 1g case the ribs, spars, and uppers skins are buckling critical, and the spars, upper and lower skins failure critical. The ribs are inactive, which is somewhat expected from the 2.5g maneuver sizing.

Table 7.11: The state of the buckling and failure KS aggregation constraints for each maneuver flight conditions. Both designs demonstrate the same state, thus only one dot shown for both cases. Active and inactive constraints are shown with filled or empty circle, respectively.

	2.5g		-1g		1g	
	Buckling	Failure	Buckling	Failure	Buckling	Failure
Ribs	●	●	○		●	○
Spars	●	●	○		●	●
Upper Skins	●	●			●	●
Lower Skins		●	●			●

A detailed buckling and failure map for each maneuver flight conditions 2.5g, -1g,

and 1g is presented in Figs. C.3 to C.5 in the Appendix C. The buckling and failure values between the two designs show the largest increase in the outboard section of the wingbox, particularly the upper and lower skins and front spar. This is directly linked to the change in the panel variable thickness at those locations.

This difference in thickness impacts the stiffness of the structure. This is evident from the deflection for the nominal cruise, 2.5g, and the -1g maneuver cases shown in Fig. 7.11. Additionally, the DLM mesh is shown for reference. Both designs show similar deflection for the inboard section, but for the outboard section the flutter constrained design has a smaller deflection, indicating a stiffer structure. Note that the DLM mesh does not deflect and follows the jig shape.

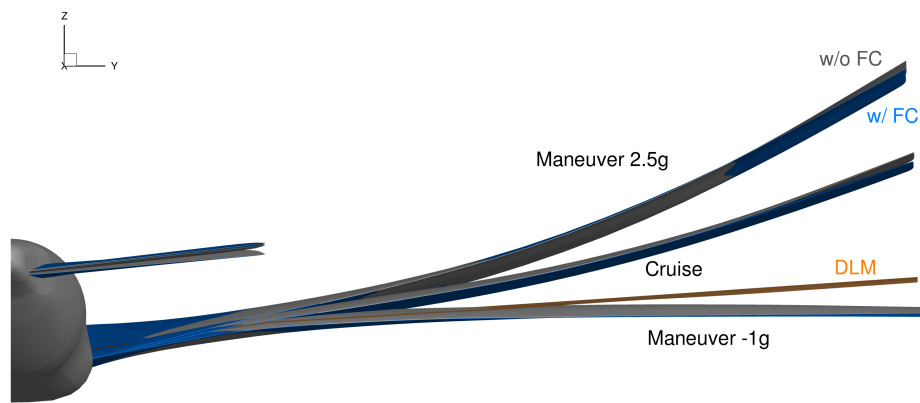


Figure 7.11: Wing span and tip deflection comparison.

The spanwise lift, twist, and thickness distributions are presented in Fig. 7.12a for both optimal designs. The normalized lift does not show any significant differences between the two designs for the nominal cruise, nor the 2.5g maneuver condition. The largest difference occurs outboard for the -1g maneuver conditions, where the flutter constrained design demonstrates a larger lift. For both designs a non-elliptical lift distribution is obtained, which will impact the induced drag detrimentally. The outboard section produces less lift, meaning that the inboard section tries to compensate by increasing lift.

The twist distribution of the flying shape does overall show minimal differences

between the two optimized designs. The largest difference is observed in the outboard section of the wing for the -1g maneuver condition, where the optimized design with the flutter constraint demonstrates less tip twist. Similarly, for the 2.5g case less twist is observed. This reduced twist can be attributed to the optimizer increasing structural thickness, hence stiffness in the outboard section of the wingbox.

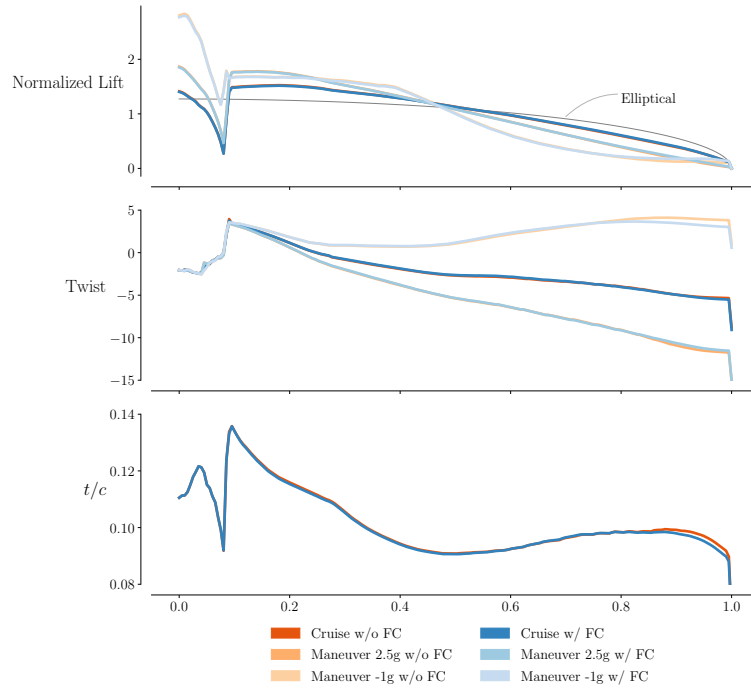
The  $t/c$  distribution shows minor difference between the two designs, except for the outboard section. The optimizer reduces the airfoil thickness of the flutter constrained design, which is likely due to increased material thickness, stiffening the structure, allowing for a more slender profile. In turn smaller  $t/c$  will decrease the drag, improving the fuel burn through an improved  $L/D$ . Furthermore, a lower  $t/c$  also mitigates buffet onset.

The airfoil cross-section and nominal cruise pressure coefficient, shown for several span-stations  $y/b$ , for both optimized designs is shown in Fig. 7.12b. As expected, the airfoil shape shows minor differences between the two designs. This is because airfoil shape changes are not captured properly with the flat mesh of the DLM or the small mesh deformation changes in the FFD. Shape changes thus have very little impact on the dynamic solution, hence the constraint.

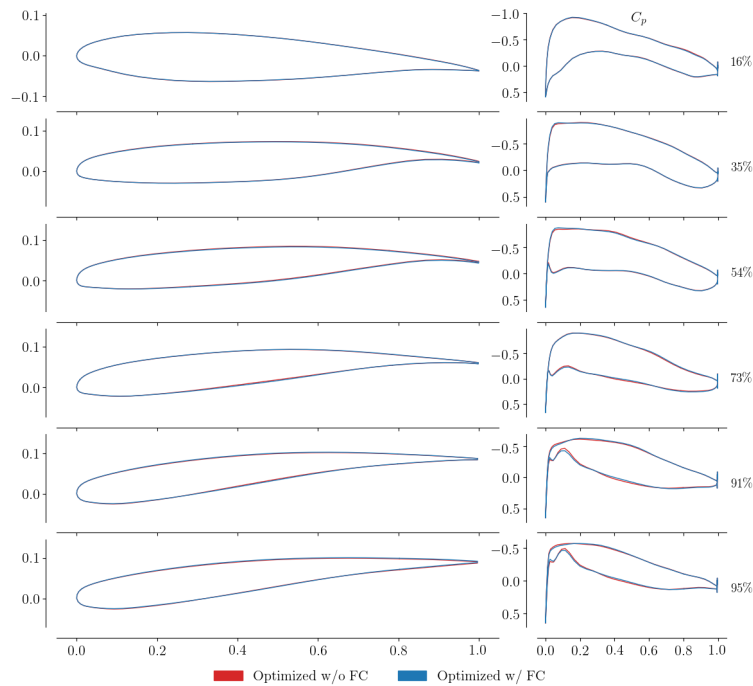
The optimizer is able to eliminate any shock on the wing surface by manipulating the airfoil shape, removing the wave drag, which improves the fuel burn. Furthermore, separation constraint for the buffet flight conditions are active, indicating a separation of 4% of the reference area,  $A_{\text{ref}}$ . However, the separation constraints for all maneuver conditions are inactive.

## 7.4.2 Modal Differences

The first twelve natural mode shapes of the two optimized aerostructural designs are shown in Fig. 7.13. The natural mode shape order is the same for the two solutions, except for mode shape 12, which demonstrates local panel modes. This local mode



(a) Spanwise lift and twist distributions are similar, except for the outboard section where the largest change in thickness occurs.



(b) Normalized airfoil section comparison (left) and nominal cruise pressure coefficient (right) at several span-stations  $y/b$ . The flutter constraint does not have a significant impact on the shape.

Figure 7.12: Aerodynamic and geometric properties of the two optimized designs.

however does not influence the optimization in any adverse way, highlighting the robustness of the proposed flutter constraint formulation. Local panel modes are in general unwanted in the detailed design optimization process.

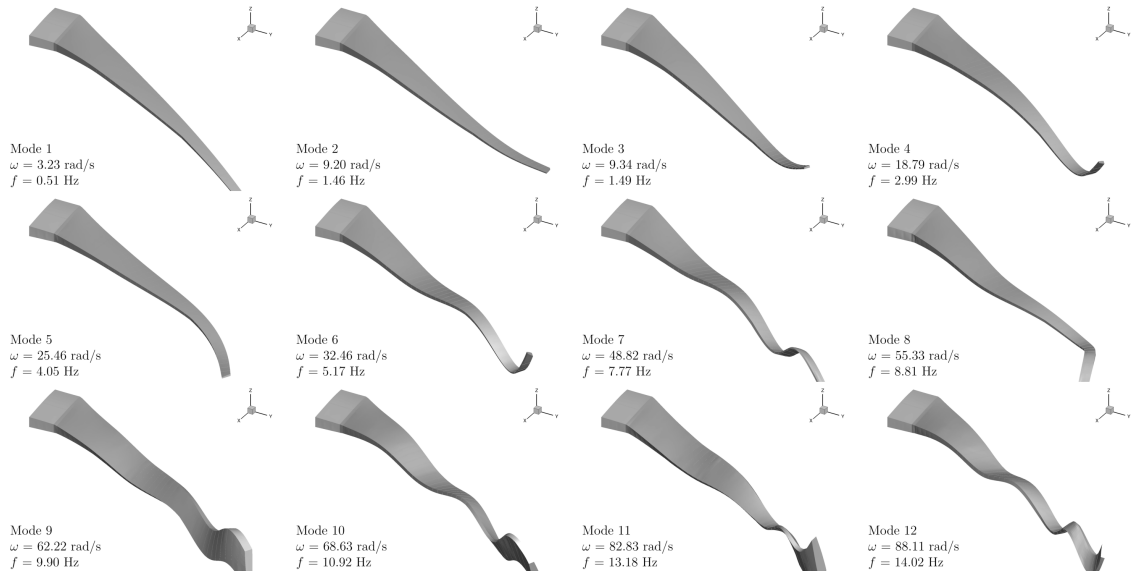
Overall, the natural frequencies are slightly higher for the optimized design obtained when enforcing the flutter constraint. This is despite of the heavier structure, in particular the outboard section, where the upper and lower skins thickness increased. This increase is likely due to a combination of a smaller aspect ratio, mass distribution, and a stiffer structure.

### 7.4.3 Flutter Differences

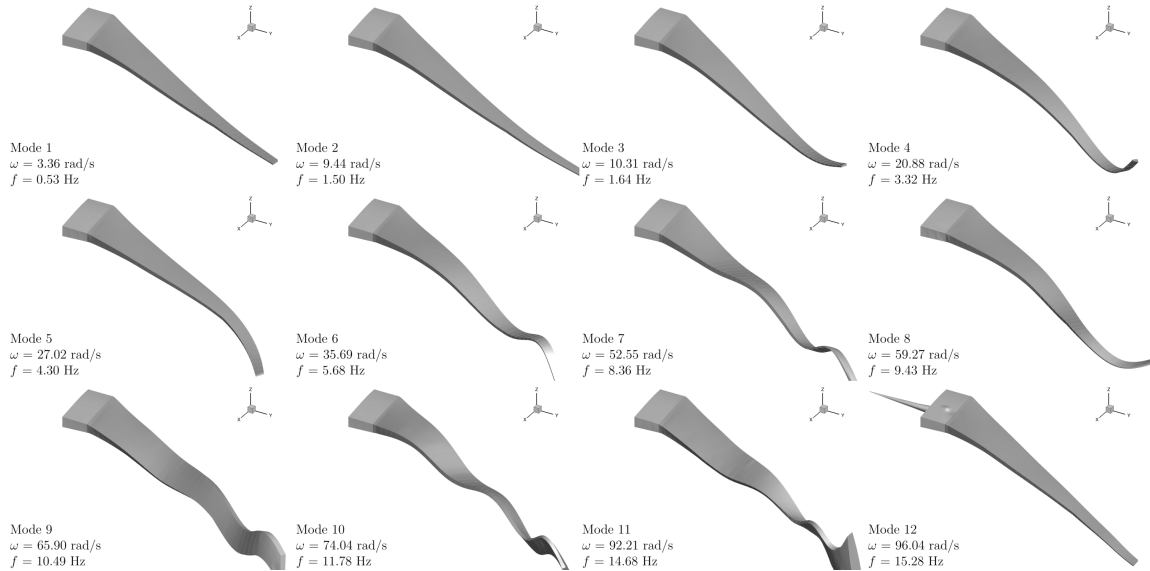
The flutter characteristics of the two optimal aerostructural designs are presented in Fig. 7.14. The optimized design without the flutter constraint, shown in Fig. 7.14a, has the 1<sup>st</sup> mode as the active flutter mode, with a flutter dynamic pressure at approximately 15 kPa. Reading Fig. 7.6 this translates to a flutter speed of approximately 240 m/s, at the specified Mach number 0.8.

For the design with the flutter constraint the optimizer successfully finds a flutter free design by moving any critical modes out of the infeasible space as shown in Fig. 7.14b. The critical mode which is driving the optimization is the 2<sup>nd</sup> mode, which is a hump mode. The 2<sup>nd</sup> natural mode shown in Fig. 7.13b is a primarily an in-plane mode with some out-of-plane bending. By inspecting the flutter frequency in Fig. 7.14d we observe that, while no mode coalescence occurs, the first three modes have similar frequencies and influence each other. This is better observed in the root-loci Figs. 7.14e and 7.14f. This indicates that the flutter mode shape is a coupled in-plane, out-of-plane bending mode.

The previously critical flutter mode, the 1<sup>st</sup> mode, has been pushed outside the flutter margin, to higher dynamic pressures, and now becomes unstable at a dynamic pressure of approximately 24 kPa. Again, referring to Fig. 7.6, this dynamic pressure



(a) Natural mode shapes of the optimized design without the flutter constraint.



(b) Natural mode shapes of the optimized design with the flutter constraint.

Figure 7.13: First twelve natural mode shapes from the two optimal aerostructural solutions obtained without (a) and with (b) the flutter constraint.

translates to a flutter speed of approximately 250 m/s, an increase of 10 m/s for this mode between the two designs. Overall, the optimizer successfully finds a flutter free design, in addition to pushing other flutter modes to higher dynamics pressures, without significant impact on the objective.

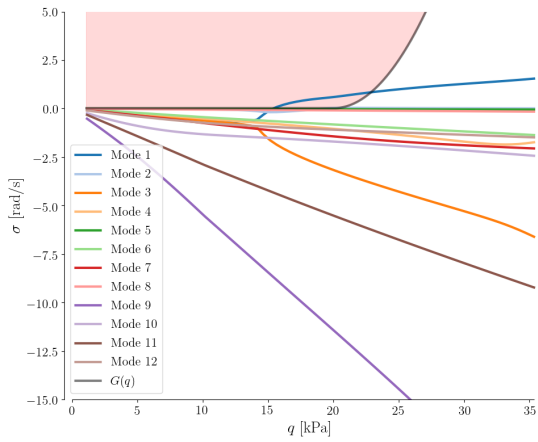
## 7.5 Summary

The proposed flutter constraint is demonstrated successfully in a high-fidelity aerostructural optimization and compared against an optimal solution obtained without the flutter constraint. Without any significant increase in fuel burn (less than 1% between the two designs), we obtain a flutter free design that is heavier and stiffer when the flutter constraint is enforced.

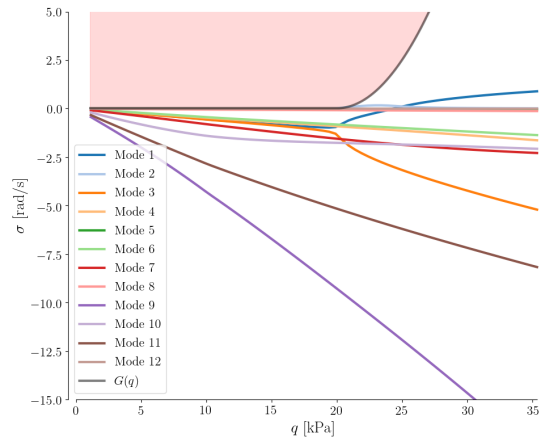
Geometric changes and aerodynamic characteristics demonstrate within 1% difference between the two optimized designs. The span and aspect ratio are found to be smaller for the flutter constrained design with an increased MAC and surface area. Similarly, the aerodynamic performance of the two optimized designs are similar, with the flutter constrained design slightly more efficient. However, these design differences are small and close to the function accuracy and tolerances set in the optimization. This makes it difficult to interpret and evaluate if such differences are solely due to the inclusion of the flutter constraint. Despite these changes being small they are not insignificant. In particular, the geometric changes are worthy of attention in the context of manufacturing, such as the span, which shortens by 6 cm.

The structural sizing differences are however significant, where the optimized design with the flutter constraint is 1.8% heavier. The largest increase is found at the front spar, followed by the lower and upper skin. This difference influences the flutter characteristics favorably, effectively moving the center of gravity upstream, which has known aeroelastic benefits, making the final design flutter free. The increase in

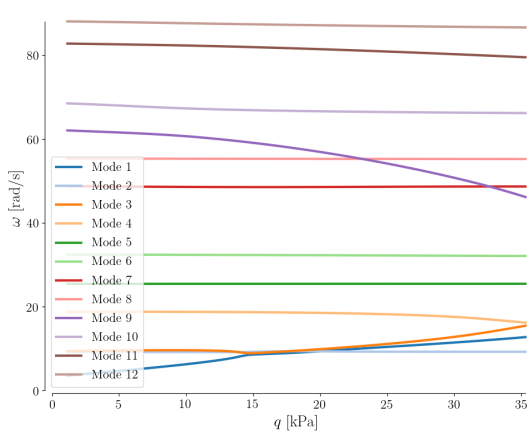




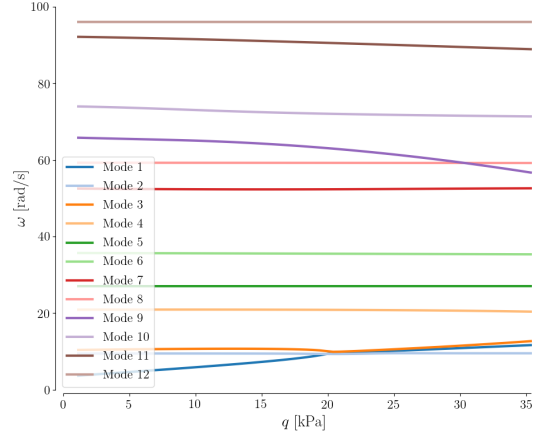
(a) Damping without flutter constraint.



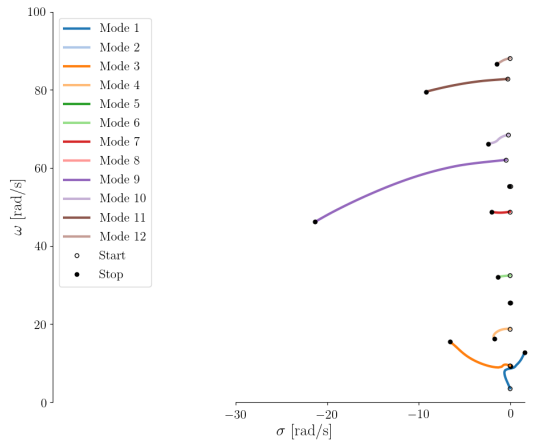
(b) Damping with flutter constraint.



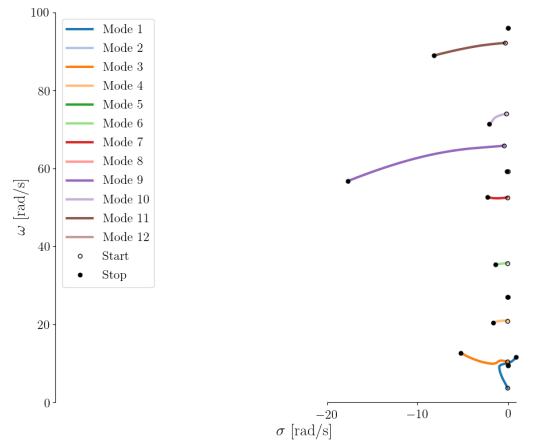
(c) Frequency without flutter constraint.



(d) Frequency with flutter constraint.



(e) Root locus without flutter constraint.



(f) Root locus with flutter constraint.

Figure 7.14: Flutter damping and frequency characteristics of the two aerostructural optimal designs without and with a flutter constraint.

stiffness of the flutter-constrained structure is evident from the reduced tip deflection.

Overall, we obtain a feasible flutter free design without significant compromise in fuel burn and structural mass. This large scale aerostructural optimization demonstrates the necessity of adding and enforcing a flutter constraint in the high-fidelity aerostructural design process. The flutter-constrained optimum exhibits small differences in planform and shape, but more significant sizing features are observed, which alter and improve the wing aeroelastic characteristics. The proposed flutter constraint complements commonly enforced structural constraints such as failure, buckling, and adjacency constraints, making the overall optimal design more robust.

## CHAPTER 8

# Conclusion

The results

This chapter gives an overview of the main conclusions of this thesis, contributions, and finally suggests directions for future work building on the work herein.

### 8.1 Conclusions and Final Remarks

The trend towards higher aspect ratio aircraft designs is driven by better fuel efficiency. Furthermore, with new materials and manufacturing techniques, their increased flexibility makes it all the more important to consider flutter accurately and early in the design process [10]. Conservative design approaches may lead to excessively stiff and hence heavy designs, while optimization approaches without flutter constraints may lead to excessively flexible wings that are infeasible because they flutter. Thus, it is desirable to consider flutter concurrently with the aircraft design and the wing design in particular. To facilitate this, MDO can be employed to obtain the best possible configuration by simultaneously design the wing shape and internal structure, subject to a flutter constraint.

While there has been extensive work in methods for flutter analysis, integrating flutter constraints into design optimization, in particular gradient-based optimization, requires additional considerations. Ideally models used for flutter prediction should capture the relevant physics with adequate accuracy to correctly drive the optimizer

and inevitably there is a compromise between model fidelity and computational cost. However, to include flutter analysis in a numerical optimization cycle, speed of execution is particularly important to make sure that the overall optimization process is tractable. In Chapter 1 a detailed literature survey of flutter modeling, analysis methods, and their application in optimization is presented. Furthermore, several open problems are identified, one being the need for a flutter methodology able to form a constraint in high-fidelity aerostructural design optimization, considering structural and aerodynamic design variables simultaneously.

To address this, Chapter 3 presents a robust, efficient, and continuous flutter analysis methodology satisfying these requirements. The methodology described is independent of specific components in which higher-fidelity methods could be used without any need for updating the formulation, nor the usage of the constraint in an aerostructural optimization. Robustness and efficiency is paramount due to the fact that each function evaluations in a high-fidelity gradient based design can be computationally expensive. Thus failing to compute the constraint would halt and potentially terminate the optimization. For this reason, a non-iterative strategy, coupled with mode tracking, is chosen to solve the nonlinear aeroelastic eigenvalue problem. To formulate a continuous flutter constraint, a bounding curve, or a flutter margin, is proposed. This bounding curve, implicitly sets the minimum flutter dynamic pressure, prevents hump modes, controls hard flutter modes, and is fully configurable based on the aircraft requirements. A conservative KS aggregation strategy is then employed, twice in series, aggregating all damping values for all modes over all dynamic pressures to obtain a scalar value, namely the flutter constraint value.

In Chapter 4, the derivative strategy of the flutter analysis process is discussed. The sensitivity of the flutter constraint with respect to all the design variables is computed accurately and efficiently using a combination of analytic and reverse mode AD. In fact, to ensure derivative are accurate, all new components developed as

part of this work are differentiated using both forward and reverse mode AD, and verified rigorously using both finite difference and complex step. A dot product test is also conducted to ensure the forward and reverse mode implementations are consistent. Derivatives are demonstrated to be accurate to machine precision for all newly implemented components.

To demonstrate the proposed flutter constraint in practical aerostructural design optimization of a HARW, a series of results are presented in Chapters 5 to 7, increasing in optimization problem size and complexity. The primary purpose of 5 is to demonstrate flutter constraint and its characteristics in a multidisciplinary optimization. The constraint was successfully applied to an idealized rectangular wing, a flat plate, in a range maximization. This problem was specifically tailored to demonstrate the continuity of the constraint and its ability to use structural and aerodynamic planform design variables in a multidisciplinary design optimization.

In Chapter 6 the flutter constraint is demonstrated in an aerostructural optimization of a rectangular wing geometry considering both structural sizing and aerodynamic planform shape. The optimized design is compared and contrasted with an optimized solution obtained without the flutter constraint. The results indicate that with the flutter constraint a stiffer, less slender wing is obtained, without any significant reduction in range (less than 1% between the two designs). The wing aspect ratio decreases by 4%, but the total mass remains roughly the same (less than 0.1% between the two designs). However, a significant difference is observed in the structural sizing variables, namely the panel and stiffener thicknesses. With the flutter constraint, the optimizer moves favors thicker panels upstream and outboard, such as leading edge spar, while reducing the upper skin thickness. The net effect is that center of gravity of the structure is moved upstream, which is known to have a stabilizing aeroelastic effect. The increase in stiffness of the flutter-constrained structure is evident from the reduced tip deflection. Most importantly, a feasible design is

obtained without significant compromise in range and structural mass.

Building on these results, in Chapter 7 the flutter constraint is applied in a large scale multipoint aerostructural optimization of the XRF1 aircraft, minimizing fuel burn considering structural, aerodynamic shape, and planform design variables. As before, the results are compared against an optimal solution obtained without the flutter constraint to identify the effect of including such constraints on the design. Due to the more complex and complete optimization formulation, additional development in TACS was necessary to account for the large fuel mass contained in the wing. To address this, a point masses element in TACS was developed that is capable of modeling the fuel mass appropriately.

Without any significant increase in fuel burn (less than 1% between the two designs), a heavier and stiffer design is obtained when the flutter constraint is enforced. While a smaller aspect ratio design is obtained with the flutter constraint, geometric differences and aerodynamic differences are overall small, and possibly close to the function accuracy and tolerances set in the optimization. This makes it difficult to interpret or draw decisive conclusions whether such differences are solely due to the inclusion of the flutter constraint. Despite these differences being small they are not insignificant, in particular in terms of manufacturing.

However, the structural sizing differences are significant, where the optimized design with the flutter constraint is 1.8% heavier. The largest increase is found at the front spar, followed by the lower and upper skin. These characteristics are similar as identified in Chapter 6, namely the optimizer takes advantage of aeroelastic benefits by effectively moving the center of gravity upstream, resulting in flutter free final design. This increase in stiffness of the flutter-constrained structure is evident from the reduced tip deflection. Overall, a feasible and optimal, flutter free design is obtained without significant compromise in fuel burn and structural mass.

The results presented in Chapters 6 and 7 demonstrate the necessity of adding

and enforcing a flutter constraint in the high-fidelity aerostructural design process. Depending on the configuration, the flutter-constrained optimum exhibits differences in planform and shape, but more significant sizing features are observed. Collectively, these differences alter and improve the wing aeroelastic characteristics ultimately obtaining a feasible and optimal flutter free design. The proposed flutter constraint complements commonly enforced structural constraints such as failure, buckling, and adjacency constraints, making the overall optimal design more robust.

## 8.2 Contribution

A large portion of the work in this dissertation is concerned with development and implementation of the proposed flutter constraint and the associated derivatives. However, original contributions are also concerned with application and analysis. The contributions of this thesis are the following:

1. Developed and implemented a robust flutter analysis methodology, and formulated a continuous flutter constraint applicable for gradient based optimization. Demonstrated the continuity of the constraint formulation resulting in a smooth feasible region for design optimization. This work made it possible to include a flutter constraint in high-fidelity aerostructural optimization, which was demonstrated using MACH [134, 210].
2. Developed and implemented efficient and accurate derivative computation of the flutter constraint using analytic and reverse mode AD. The derivative implementation is rigorously verified and for all new components shown to be accurate to machine precision. Reverse mode derivative implementation of the Lanczos eigenvalue method to compute derivatives of modes and mode shapes with reduced basis. This enables computation of varying modes shapes, which is important when considering geometric design variables.

3. Performed the first known high-fidelity aerostructural optimization subject to flutter constraint considering aerodynamic, planform, and structural sizing design variables.
4. Performed the first known multipoint high-fidelity aerostructural optimization of a full configuration commercial aircraft subject to flutter constraint. This optimization problems considers over  $\mathcal{O}(1000)$  aerodynamic, shape, planform, and structural sizing design variables together with  $\mathcal{O}(1000)$  aerodynamic, geometric, stress, buckling, manufacturing, and flutter constraints.
5. Extensive literature survey conducted for flutter and post-flutter analysis methods in aircraft design optimization [2].

### 8.3 Recommendations for Future Work

Throughout this work, numerous possible avenues for future work have been identified that can continue to advance the proposed capabilities and the state of the art of high-fidelity aerostructural design optimization. These suggestions include not only applications, but also new methods that can be incorporated.

1. *Multipoint flutter constraint.*

In this work a single Mach number is specified and the dynamic pressure is varied (using matched density and speeds) effectively simulating change in altitude. By including more Mach numbers a greater portion of the flight envelope could be considered to provide a more robust design. Two possible approaches are suggested here. Leveraging the MACH framework multipoint component, a multipoint stencil could be generated including the most critical Mach numbers similarly as done for the cruise conditions. Another approach is to adopt a sampling strategy that samples the entire flight envelope over multiple Mach



and altitudes. A similar strategy is proposed by [116]. The second approach is more intrusive and requires a more fundamental development and refactoring of the current code. Both approaches are embarrassingly parallel, thus analysis points can be run in parallel, minimizing the impact on computational cost.

2. *Full configuration with fuel and engine mass.*

Including the engine mass is known to lower the torsion mode, thus directly affecting the flutter characteristics. As part of this work, the necessary point mass element and weighting strategy were implemented to model the engine mass. However, including the engine mass in the dynamic analysis resulted in unwanted local modes appearing when changing the design. While it may be difficult to completely get rid of such behavior, it is possible that more sophisticated element is needed, in particular the RBE3. Such element was found not attainable to implement in the version of TACS used in this work. Further investigation and implementation is suggestion in a more recent version of TACS.

3. *Visualize the critical and active aeroelastic modes.*

In this work the natural mode shapes are visualized to determine which type of mode shapes, e.g., bending, are used in the aeroelastic analysis. The aeroelastic modes then emanate from these modes at wind-off. By using the aeroelastic frequency characteristics (the imaginary part of the eigenvalues), one can estimate the aeroelastic mode shape based on how close the frequencies of other modes are to the critical mode. However, visualizing the actual aeroelastic modes can provide additional insight and aid understanding interpreting results and design tradeoffs. Implementing such visualization requires storing the aeroelastic mode shapes and a post-processing step such that the reduced aeroelastic mode shape can be represented on the structure. Storing these modes is not necessary

during optimization and this analysis could be performed when analyzing the optimized design.

4. *Parallel implementation of the flutter solution method.*

The current implementation of the flutter analysis is executed on a single core. While this is not found to be a limitation for the cases considered in this work, solving the generalized nonlinear eigenvalue problem for large application problems considering 100 natural modes or more, may become a bottleneck. Utilizing eigenvalue libraries to solve such large problems in parallel could alleviate that. Similarly, during the computation of the correlation matrix for the mode tracking is independent of each other and is embarrassingly parallel. More generally, using libraries such as MPI could further improve performance in various places.

5. *Implement aerodynamic solver directly representing the aerodynamic shape.*

In this work the aerodynamic method chosen is a DLM, which models the aerodynamic surface directly as a flat surface, parallel to the stream of flow. Aerodynamic shape changes are thus not directly modeled in the dynamic analysis. Other aerodynamic methods are possible as the flutter methodology presented here is general.

6. *Flutter analysis considering aerodynamic nonlinearities.*

A major challenge is developing efficient flutter analysis models and methods applicable in the presence of aerodynamic nonlinearities. Aerodynamic nonlinearities, such as shock waves and flow separation, significantly impact the flutter speed. The linear doublet lattice method is not capable of modeling such nonlinearities, and thus may give inaccurate predictions in the transonic flow regime. However, the computational cost is a big challenge because the flutter point de-

depends on the equilibrium state, requiring at minimum, the steady background flow solution. Several paths are possible here that are part of this topic.

Correction methods exist that corrects linear responses using higher-fidelity data to better match nonlinear CFD methods can be applied. However, despite many of the correction methods achieving good agreement with reference CFD data, they all depend on high-fidelity reference data. In fact, many of the more recent methods employ time-linearized frequency domain or small perturbation CFD methods, to compute and provide the reference data. Such CFD methods could also be applied directly, but developing derivatives for such methods needs care due to the second order derivative information needed on an already linearized approach. Furthermore, ROMs based on high-fidelity CFD are becoming increasingly popular choice in aeroelastic analysis. Finally, despite the high computational cost, full nonlinear frequency domain CFD approaches, such as harmonic-balance or time-spectral can also be considered and applied.

7. *Geometrically nonlinear configurations.* For moderately flexible configurations, structural nonlinearities can be neglected, so eliminating the need to solve a nonlinear static aeroelastic problem at each flight condition. However, for high aspect ratio wings this assumption may not be applicable. Investigating structural nonlinearities in flutter prediction is challenging for both analysis and design and is an active field of research.

# APPENDIX A

## Doublet Lattice Method

### A.1 Unsteady Transonic Flow

Here we provide a brief review unsteady transonic flow equations ranging from linear to nonlinear that are found in literature and applied in flutter analysis.

#### A.1.1 Transonic Small Disturbance (TSD)

The TSD equations are a popular linearization of the full potential equations (FPE) where the velocity potential is assumed have a uniform flow in the  $x$  direction and composed of

$$\Phi = U_\infty x + \phi, \quad (\text{A.1})$$

where  $U_\infty$  is the freestream velocity, and  $\phi$  is the perturbation velocity potential, where perturbations are assumed to be small. Unlike the FPE which is valid for arbitrary bodies, due the small perturbations the, TSD equation is only valid for slender thin bodies. It can be shown, neglecting higher order effects, that the unsteady TSD can be written as [211]

$$\beta^2 \phi_{xx} + M^2(\gamma + 1)\phi_x \phi_{xx} + \phi_{yy} + \phi_{zz} - \frac{2U}{a^2} \phi_{xt} - \frac{1}{a^2} \phi_{tt} = 0 \quad (\text{A.2})$$

where  $\beta^2 = 1 - M^2$ . This equation is nonlinear, due to the  $\phi_x \phi_{xx}$  term, and is capable of modeling shockwaves in the direction of the flow or  $x$ . Neglecting  $\phi_x \phi_{xx}$  will make Eq. (A.2) valid in the subsonic regime, but will reduce accuracy in the transonic regime. Thus, neglecting the nonlinear term the pressure potential equation can be written as follows:

$$\beta^2 p_{xx} + p_{yy} + p_{zz} - \frac{2U}{a^2} p_{xt} - \frac{1}{a^2} p_{tt} = 0 \quad (\text{A.3})$$

where  $p$  is both the pressure and the pressure potential.

### A.1.2 Doublet Lattice Method (DLM)

Here we provide a short discussion on boundary element methods which are also referred to as lifting surface methods. They have clear advantages over other methods in terms of efficiency as only the boundary of the domain needs to be discretized. In terms of lifting surfaces the flow field around a body is fully described by elements such as sources and doublets. Depending on the type of element used the flow field can be described as a multiplication of the element strength and a kernel function which is defined based on the properties of the element. Using the superposition principle the body can then be divided into multiple elements and integrated over in order to compute the flow field.

The pressure potential (A.3) is the basis for deriving the DLM [119, 120]. An excellent document on the mathematics leading up to the DLM is given by Blair [170], but here we briefly outline the procedure. Solutions to the pressure potential equation (A.3) can be written as follows:

$$p = \frac{1}{R} f(t - \tau)$$

where  $f$  is a general function and  $\tau$  (retarded time) and  $R$  are defined as follows:

$$\begin{aligned} R &= \sqrt{(x - \xi)^2 + \beta^2((y - \eta)^2 + (z - \zeta)^2)} \\ &= \sqrt{x_0^2 + \beta^2(y_0^2 + z_0^2)} \\ \tau &= \frac{R - M(x - \xi)}{a\beta^2} \end{aligned}$$

Now, specializing this general expression to harmonic behavior in time, we can arrive at the following expression:

$$\begin{aligned} p &= \frac{1}{R} e^{\left[ \frac{i\omega}{a\beta^2} (M(x-\xi) - R) \right]} e^{i\omega t} \\ \psi &= \frac{\partial}{\partial z} \left[ \frac{1}{R} e^{\left[ \frac{i\omega}{a\beta^2} (M(x-\xi) - R) \right]} \right] \\ &= \beta^2 (z - \zeta) \left[ \frac{1}{R^2} - \frac{1}{R^3} \right] e^{\left[ \frac{i\omega}{a\beta^2} (M(x-\xi) - R) \right]} \end{aligned}$$

This is the formula for a single pressure doublet. It describes the pressure  $\psi$  at coordinates  $(x, y, z)$  due to a pressure doublet at coordinates  $(\xi, \eta, \zeta)$ .

$$\phi = -\frac{1}{U} e^{-\frac{i\omega x}{U}} \int_{-\infty}^x e^{\frac{i\omega x}{U}} \psi dx$$

$$\bar{w} = -\frac{1}{U} \frac{\partial \phi}{\partial z} = \frac{1}{U^2} e^{-\frac{i\omega x}{U}} \int_{-\infty}^x e^{\frac{i\omega x}{U}} \psi dx$$

$$\bar{w} = \frac{1}{4\pi\rho U^2} \int_S \Delta p e^{-\frac{i\omega(x-\xi)}{U}} \frac{\partial^2}{\partial z^2} \left[ \int_{-\infty}^{x-\xi} \frac{1}{R} e^{i\omega \left( \frac{x}{U} - \frac{R-Mx}{a\beta^2} \right)} dx \right] dS$$

The doublet lattice method (DLM) is based on the following expression for the nondimensional normal wash,  $\bar{w}(x, s)$  on a wing in an oscillating flow where the  $x$ -

direction is parallel to the free-stream direction:

$$\bar{w}(x, s) = \frac{1}{8\pi} \int_S \Delta C_p(\xi, \eta) K(x - \xi, s - \eta) d\xi d\eta \quad (\text{A.4})$$

where the kernel function  $K$ , is given by the following expression:

$$K = e^{-i\omega x_0/U} \left[ \frac{K_1 T_1}{r_1^2} + \frac{K_2 T_2}{r_1^4} \right] \quad (\text{A.5})$$

where:

$$x_0 = x - \xi$$

$$y_0 = y - \eta$$

$$z_0 = z - \zeta$$

$$r_1 = \sqrt{y_0^2 + z_0^2}$$

The kernel function is derived from the doublet acceleration potential and for the DLM given by [212].

The terms  $T_1$  and  $T_2$  are the normal and transverse contributions to the normal wash defined as follows:

$$T_1 = \cos(\gamma_r - \gamma_s)$$

$$T_2 = (z_0 \cos \gamma_r - y_0 \sin \gamma_r)(z_0 \cos \gamma_r - y_0 \sin \gamma_r)$$

The terms  $K_1$  and  $K_2$  are the planar and nonplanar contributions to the kernel function given as follows:

$$K_1 = I_1 + \frac{Mr_1}{R} \frac{e^{-ik_1 u_1}}{\sqrt{1 + u_1^2}}$$

$$K_2 = -3I_2 - i \frac{k_1 M^2 r_1^2}{R^2} \frac{e^{-ik_1 u_1}}{\sqrt{1 + u_1^2}} - \frac{Mr_1}{R} \left[ (1 + u_1^2) \frac{\beta^2 r_1^2}{R^2} + 2 + \frac{Mr_1 u_1}{R} \right] \frac{e^{-ik_1 u_1}}{(1 + u_1^2)^{3/2}}$$

where the following definitions are used:

$$\begin{aligned}\beta^2 &= 1 - M^2 \\ k_1 &= \frac{\omega r_1}{U} \\ R^2 &= x_0^2 + \beta^2 r_1^2 \\ u_1 &= \frac{MR - x_0}{\beta^2 r_1}\end{aligned}$$

Finally,  $I_1$  and  $I_2$  are integrals that are given as follows:

$$\begin{aligned}I_1 &= \int_{u_1}^{\infty} \frac{e^{-ik_1 u}}{(1 + u^2)^{3/2}} du \\ I_2 &= \int_{u_1}^{\infty} \frac{e^{-ik_1 u}}{(1 + u^2)^{5/2}} du\end{aligned}\tag{A.6}$$

The doublet lattice method contains two essential components:

1. The integral expressions (A.6) are approximated using an approximation of the function  $1 - u/\sqrt{1 + u^2}$
2. The kernel function itself is approximated using a quartic expression and integrated across the panel length

The body under investigation is discretized into panels and Eq. (A.4) is evaluated for each panel, at the 3/4 chords (the collocation point), against all other panels on the body. Note that panels are required to be parallel to the streamwise direction in order for the theory to hold. This approach leads to a system of equations where the resulting matrix relates pressure coefficient to the downwash on all panels

$$\bar{\mathbf{w}} = \mathbf{D}\Delta\mathbf{C}_p\tag{A.7}$$

where the matrix  $\mathbf{D}$  elements are each solution of the Eq. (A.4). For further details please refer to Ref. [119, 120]



### A.1.3 Boundary Conditions

Given the appropriate boundary condition it can be shown that the linearized small disturbance equation has a unique solution. A time varying surface in space can be given by the equation

$$F(x, y, z, t) = 0 \quad (\text{A.8})$$

Due to the inviscid nature of the problem, the boundary conditions requires the flow to be tangential to the surface at a given time instance. This is equivalent to saying that the normal flow component should be zero. This can be written as

$$\frac{\partial F}{\partial t} + \mathbf{V} \cdot \nabla F = 0 \quad (\text{A.9})$$

$$\frac{\partial F}{\partial t} + V_i \frac{\partial F}{\partial x_i} = 0 \quad (\text{A.10})$$

where the latter equation is given in index notation. Here we linearize the above equation about an undisturbed and uniform flow. This limits us to thin wings. The surface of the wing is described as  $F = F_w(x, y, z, t)$  and given by two uncoupled components,  $h_m, h_t$  which are the deformations of the midplane and the thickness envelope, respectively, about the undeformed midplane, i.e.,  $z = 0$ . This can be written as

$$F_w(x, y, z, t) = z - h(x, y, t) = 0 \quad (\text{A.11})$$

where

$$h(x, y, t) = h_m(x, y, t) \pm h_t(x, y, t) \quad (\text{A.12})$$

The linearized flow about uniform freestream  $\mathbf{V} = U\hat{i}$  is given as

$$\mathbf{V} = (U + u)\hat{i} + v\hat{j} + w\hat{k} \quad (\text{A.13})$$

where  $u, v, w$  are the small disturbances from the uniform stream. By substituting Eq. (A.11) in Eq. (A.9) we get

$$-\frac{\partial h}{\partial t} - (U + u)\frac{\partial h}{\partial x} - v\frac{\partial h}{\partial y} + w = 0 \quad (\text{A.14})$$

dropping nonlinear terms and rearranging we obtain

$$w = \frac{\partial h}{\partial t} + U\frac{\partial h}{\partial x} \quad (\text{A.15})$$

We can further nondimensionalize the equation by dividing with the freestream velocity  $U$

$$\hat{w} = \frac{1}{U}\frac{\partial h}{\partial t} + \frac{\partial h}{\partial x}. \quad (\text{A.16})$$

#### A.1.4 Boundary Conditions in Frequency Domain

The normal wash  $w$  is computed from the boundary conditions (BCs). Impenetrability boundary condition is expressed as

$$\phi_z(x, y, 0, t) = w = h_t + Uh_x \quad (\text{A.17})$$

where subscripts are partial differentiation and  $h(x, y, t)$  is the disturbance function describing deviation in  $z$ . Transform to frequency domain using harmonic motion, let

$$\phi_z(x, y, z, t) = \bar{\phi}_z(x, y, z)e^{i\omega t} \quad (\text{A.18})$$

$$h(x, y, t) = \bar{h}(x, y)e^{i\omega t} \quad (\text{A.19})$$

and substituting into Eq. (A.17), differentiating and simplifying, we get

$$\bar{\phi}_z = i\omega\bar{h} + U\bar{h}_x. \quad (\text{A.20})$$

In terms of the nondimensional Eq. (A.16) and using the reduced frequency  $k = b\omega/U$  we obtain the nondimensional normal wash

$$\bar{w} = \frac{\bar{\phi}_z}{U} = \frac{ik}{b}\bar{h} + \bar{h}_x \quad (\text{A.21})$$

We can further write the nondimensional normal wash using the reduced frequency that is used in the DLM, namely,  $\hat{k} = k/b = \omega/U$  or

$$\bar{w} = \frac{\bar{\phi}_z}{U} = i\hat{k}\bar{h} + \bar{h}_x \quad (\text{A.22})$$

Assuming a general motion (as is possible when using the determinant iterative method)

$$\phi_z(x, y, z, t) = \bar{\phi}_z(x, y, z)e^{st} \quad (\text{A.23})$$

$$h(x, y, t) = \bar{h}(x, y)e^{st} \quad (\text{A.24})$$

where  $s = \gamma + i\omega$  is the Laplace parameter we obtain

$$\bar{\phi}_z = s\bar{h} + U\bar{h}_x \quad (\text{A.25})$$

$$= (\gamma + i\omega)\bar{h} + U\bar{h}_x \quad (\text{A.26})$$

Expanding and normalize by dividing with  $U$  we obtain

$$\bar{w} = \frac{\bar{\phi}_z}{U} = i\hat{k}\bar{h} + \left(\frac{\gamma}{U}\bar{h} + \bar{h}_x\right). \quad (\text{A.27})$$

## APPENDIX B

# Lanczos Method

### B.1 Lanczos Algorithm

The Lanczos algorithm extracts eigenvalues for symmetric generalized eigenvalue problems. Here, we use this algorithm to solve for the natural frequencies of the structural problem without aerodynamic loads:

$$\mathbf{K}\mathbf{u} = \lambda\mathbf{M}\mathbf{u}.$$

Instead of solving this problem directly, we use a shift and invert strategy to zero-in on the desired spectrum to reduce the number of iterations required. This shift and invert technique produces the following eigenproblem that has the same eigenvectors but different eigenvalues:

$$\mathbf{M}(\mathbf{K} - \sigma\mathbf{M})^{-1}\mathbf{M}\mathbf{u} = \mu\mathbf{M}\mathbf{u},$$

where the transformed eigenvalue  $\mu$  is related to the original eigenvalue  $\lambda$  through the relationship:

$$\mu = \frac{1}{\lambda - \sigma}.$$

When  $\sigma$  is chosen such that it lies close to the desired  $\lambda$ , the corresponding transformed eigenvalues,  $\mu$ , become well separated, making the Lanczos algorithm more efficient.

The Lanczos algorithm uses an  $\mathbf{M}$ -orthonormal subspace, written as  $\mathbf{V}_m \in \mathbb{R}^{n \times m}$ , such that  $\mathbf{V}_m^T \mathbf{M} \mathbf{V}_m = \mathbf{I}_m$ . In exact arithmetic, this subspace can be formed directly from the Lanczos three-term recurrence. However, the resulting subspace loses orthogonality as the algorithm converges to an eigenvalue due to numerical truncation errors. Instead, we use an expensive, but effective, full-orthonormalization procedure (Gram–Schmidt) that enforces  $\mathbf{M}$ -orthonormality.

---

```

1: Lanczos method for computing eigenvalues/eigenvectors of  $\mathbf{K}\mathbf{u} = \lambda\mathbf{M}\mathbf{u}$ 
2: Given:  $m, \hat{\mathbf{v}}_1, \sigma, \epsilon_{tol}$ 
3: Factor the matrix  $(\mathbf{K} - \sigma\mathbf{M})$ 
4: Set  $i = 1$ 
5: while  $i \leq m$  do
6:    $\hat{\mathbf{v}}_{i+1} = (\mathbf{K} - \sigma\mathbf{M})^{-1}\mathbf{M}\mathbf{v}_i$ 
7:   Set  $j = 1$ 
8:   while  $j \leq i$  do ▷ Full  $\mathbf{M}$ -orthonormalization
9:      $h_{ji} = \mathbf{v}_j^T \mathbf{M} \hat{\mathbf{v}}_{i+1}$ 
10:     $\hat{\mathbf{v}}_{i+1} \leftarrow \hat{\mathbf{v}}_{i+1} - h_{ji} \mathbf{v}_j$ 
11:     $j \leftarrow j + 1$ 
12:  end while
13:   $\alpha_i \leftarrow h_{ii}$ 
14:   $\beta_i = \sqrt{\hat{\mathbf{v}}_{i+1}^T \mathbf{M} \hat{\mathbf{v}}_{i+1}}$ 
15:   $\mathbf{v}_{i+1} = \hat{\mathbf{v}}_{i+1} / \beta_i$ 
16:   $\mathbf{T}_i = \text{tridiag}_k\{\beta_k, \alpha_k, \beta_{k-1}\}$  ▷ Solve the reduced eigenproblem
17:  Solve  $\mathbf{T}_i \mathbf{y}_i = \theta \mathbf{y}_i$  for  $(\theta, \mathbf{y}_i)$ 
18:  if  $\beta_i \mathbf{y}_i^T \mathbf{e}_i < \epsilon_{tol}$  then ▷ Test for convergence
19:     $\mathbf{u} = \mathbf{V}_i \mathbf{y}_i$ 
20:     $\lambda = \frac{1}{\theta} + \sigma$ 
21:    break
22:  end if
23:   $i \leftarrow i + 1$ 
24: end while

```

---

The Lanczos method can be easily extended to find multiple eigenpairs  $(\lambda_i, \mathbf{u})$ . A byproduct of the Lanczos method is the  $\mathbf{M}$ -orthonormal subspace. Instead of discarding this subspace, we use these vectors to enhance the flutter prediction and

eigenvector computation.

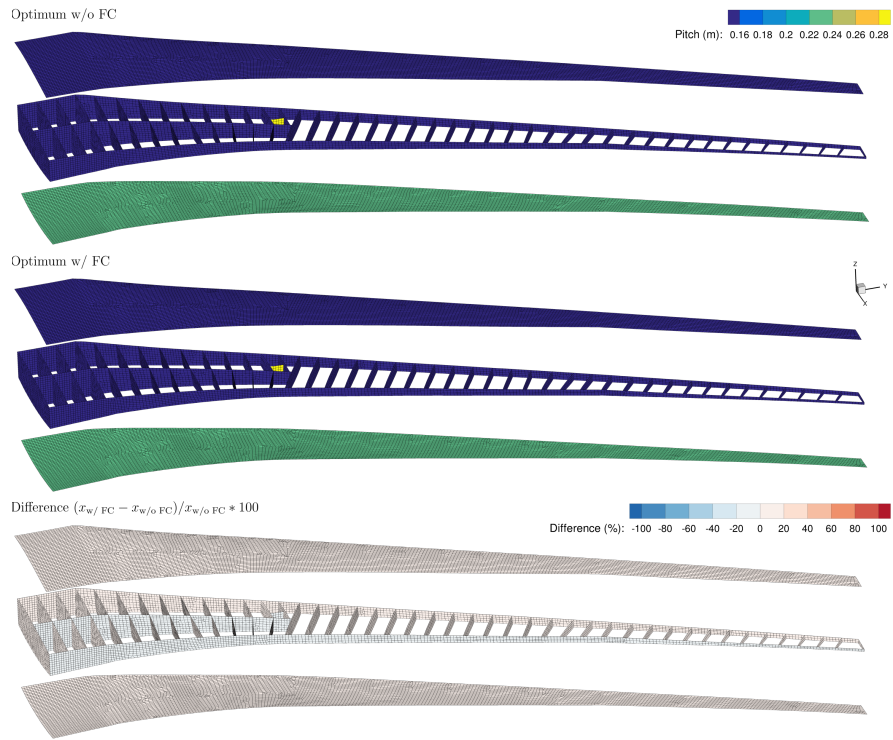
## APPENDIX C

### XRF1

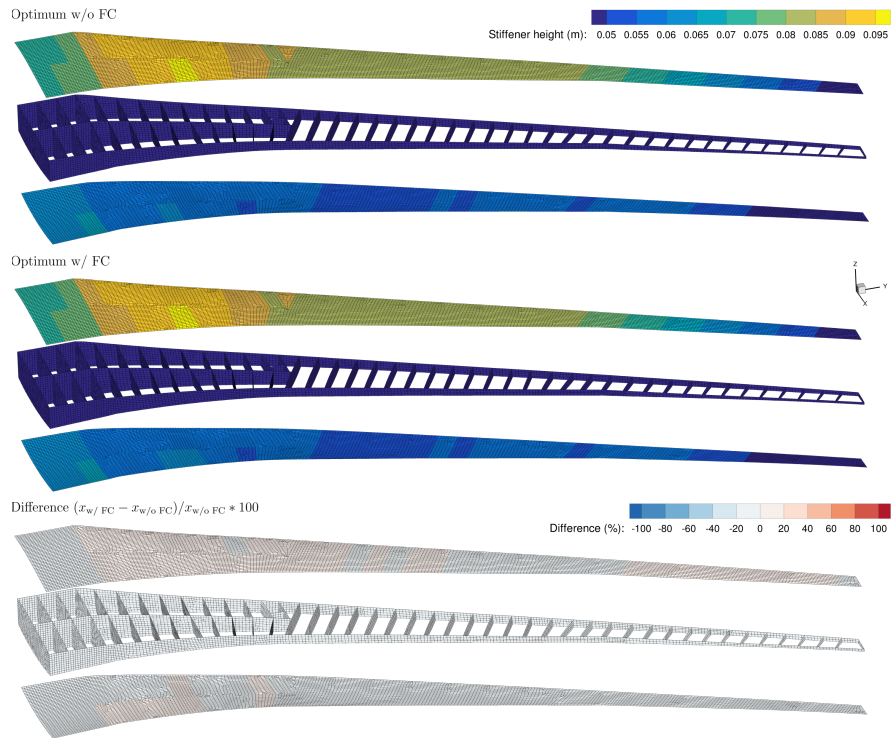
#### C.1 Additional Optimization Results

Plots for individual design variable, the stiffener pitch, stiffener height, stiffener thickness, and panel thickness, are provided in the in Figs. C.1 and C.2.

A detailed buckling and failure map for each maneuver flight conditions 2.5g, -1g, and 1g is presented in Figs. C.3 to C.5.



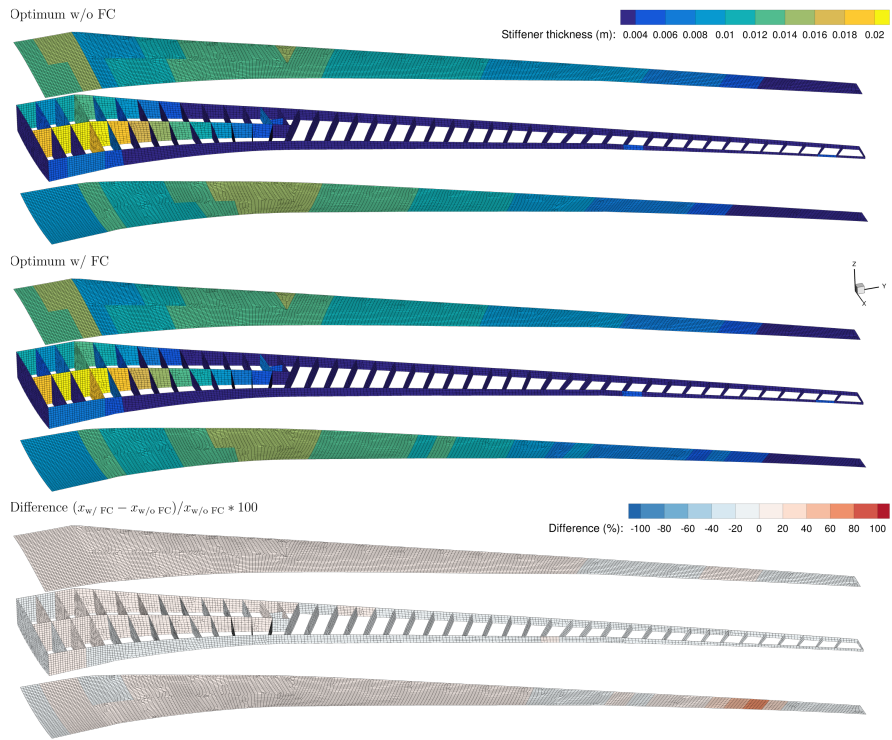
(a) Pitch



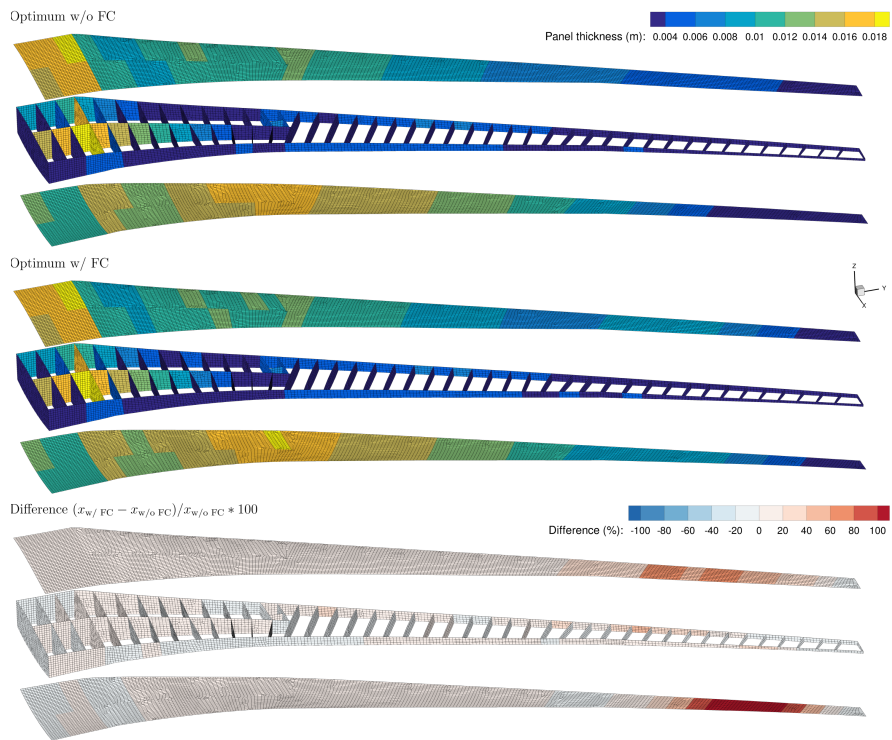
(b) Stiffener height

Figure C.1: Structural thicknesses of the optimized designs with and without the flutter constraint. A diverging color map highlights the difference between the designs.



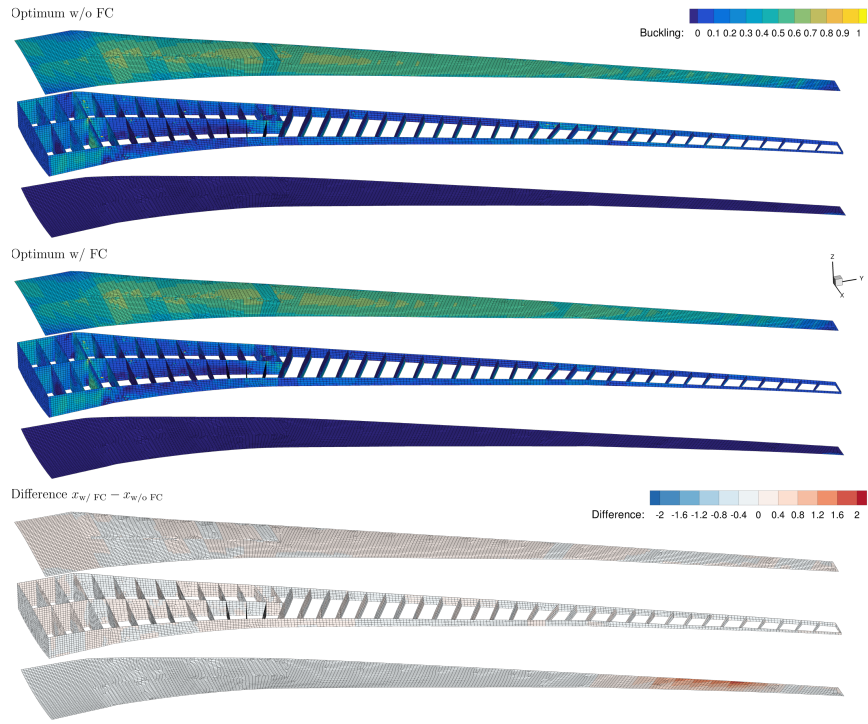


(a) Stiffener thickness

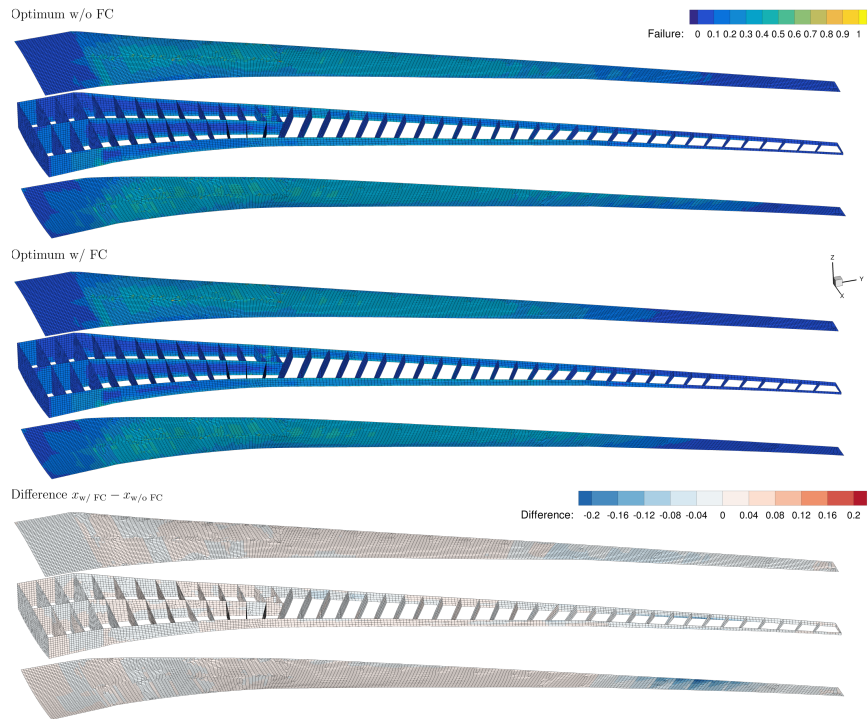


(b) Panel thickness

Figure C.2: Structural thicknesses of the optimized designs with and without the flutter constraint. A diverging color map highlights the difference between the designs.

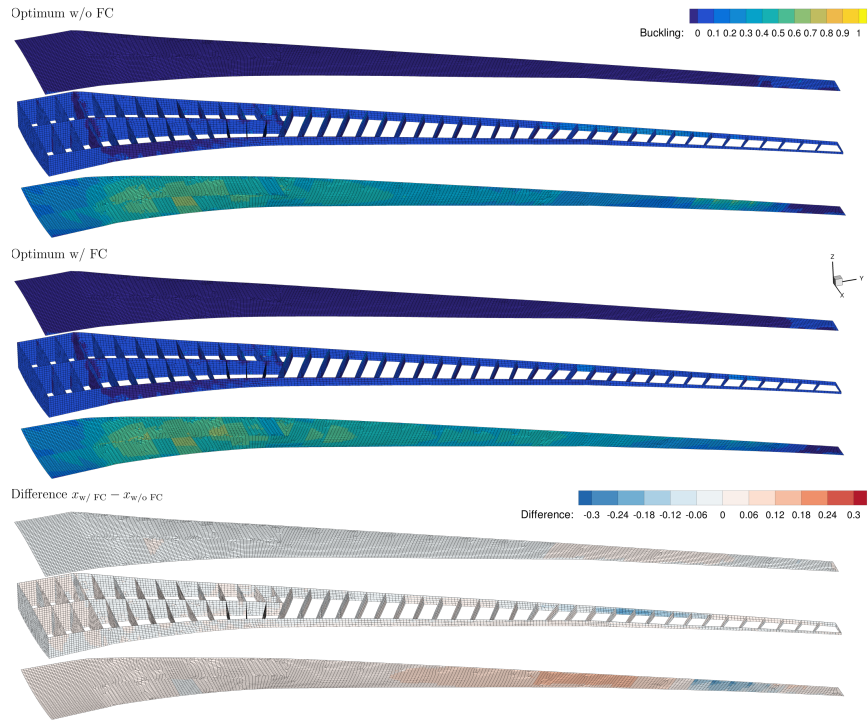


(a) Buckling

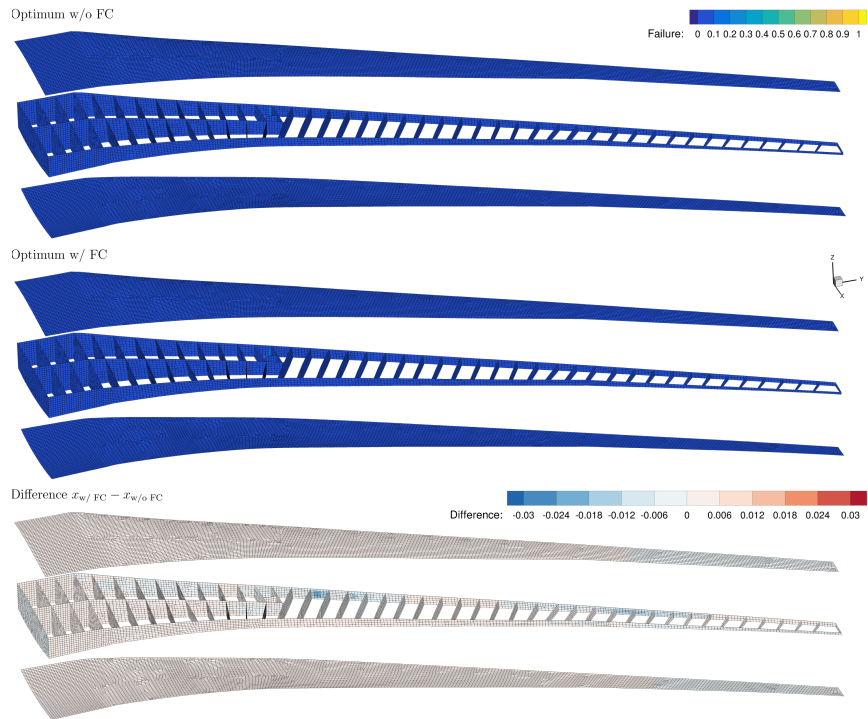


(b) Failure

Figure C.3: Buckling and failure comparison of the optimized designs with and without the flutter constraint, for the 2.5g maneuver condition. A diverging color map highlights the difference between the designs.

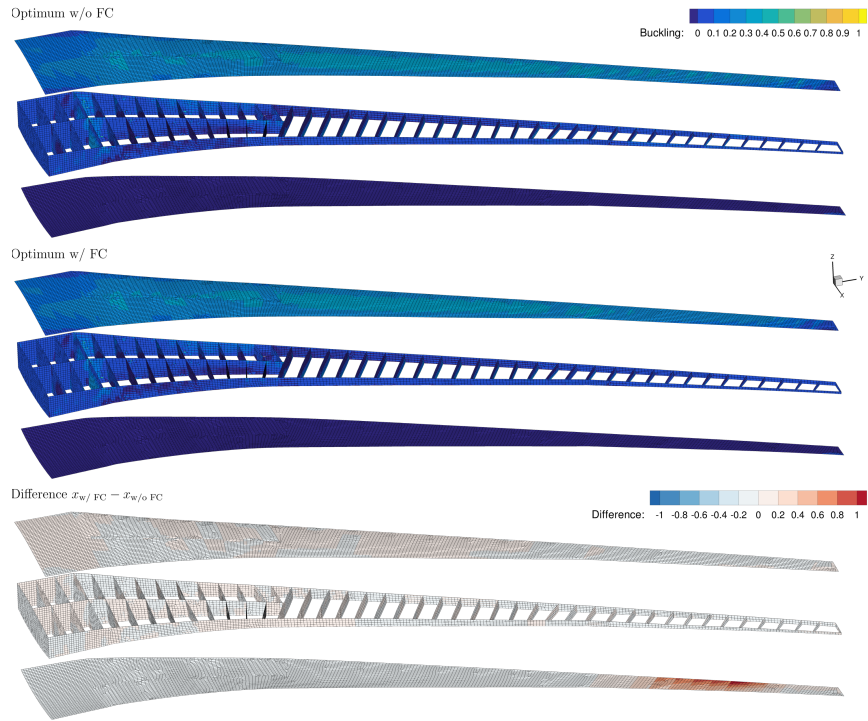


(a) Buckling

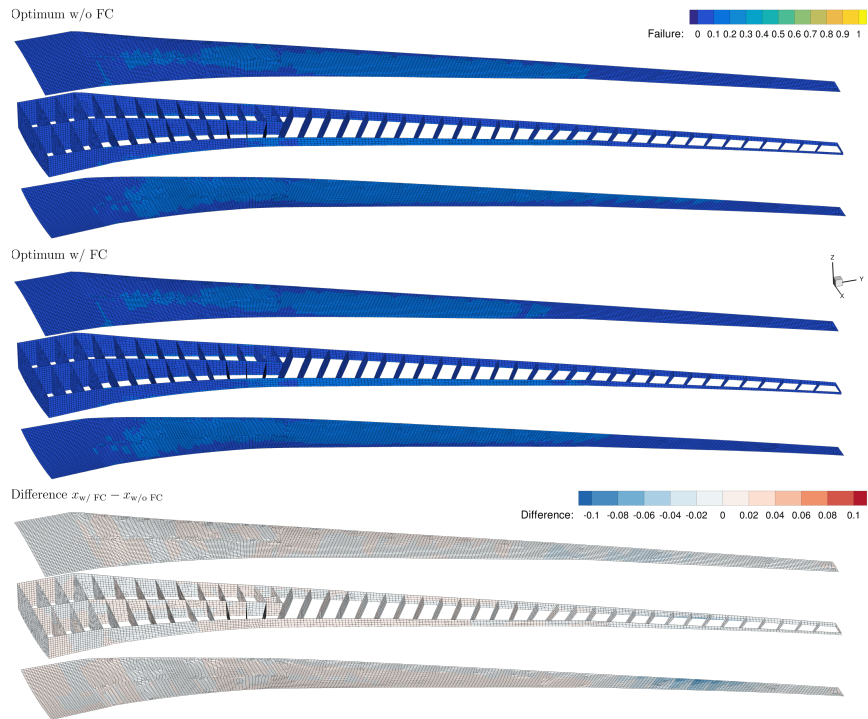


(b) Failure

Figure C.4: Buckling and failure comparison of the optimized designs with and without the flutter constraint, for the -1g maneuver condition. A diverging color map highlights the difference between the designs.



(a) Buckling



(b) Failure

Figure C.5: Buckling and failure comparison of the optimized designs with and without the flutter constraint, for the 1g maneuver condition. A diverging color map highlights the difference between the designs.

## Bibliography

- [1] Lambe, A. B., and Martins, J. R. R. A., “Extensions to the Design Structure Matrix for the Description of Multidisciplinary Design, Analysis, and Optimization Processes,” *Structural and Multidisciplinary Optimization*, Vol. 46, 2012, pp. 273–284. doi:10.1007/s00158-012-0763-y.
- [2] Jonsson, E., Riso, C., Lupp, C. A., Cesnik, C. E., Martins, J. R., and Epureanu, B. I., “Flutter and Post-Flutter Constraints in Aircraft Design Optimization,” *Progress in Aerospace Sciences*, Vol. 109, 2019, p. 100537. doi:10.1016/j.paerosci.2019.04.001, URL <http://www.sciencedirect.com/science/article/pii/S0376042118301520>.
- [3] Bisplinghoff, R. L., Ashley, H., and Halfman, R. L., *Aeroelasticity*, 2<sup>nd</sup> ed., Dover Publications, Mineola, NY, 1996.
- [4] Garrigues, E., “A Review of Industrial Aeroelasticity Practices at Dassault Aviation for Military Aircraft and Business Jets,” *AerospaceLab Journal*, , No. 14, 2018, pp. 1–34. doi:10.12762/2018.AL14-09, URL <https://hal.archives-ouvertes.fr/hal-01935273>.
- [5] Haftka, R. T., “Automated procedure for design of wing structures to satisfy strength and flutter requirements,” Tech. Rep. TN D-7264, NASA, July 1973.
- [6] Sobieszczanski-Sobieski, J., and Haftka, R. T., “Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments,” *Structural Optimization*, Vol. 14, No. 1, 1997, pp. 1–23. doi:10.1007/BF01197554.
- [7] Kenway, G. K. W., Kennedy, G. J., and Martins, J. R. R. A., “Scalable Parallel Approach for High-Fidelity Steady-State Aeroelastic Analysis and Derivative Computations,” *AIAA Journal*, Vol. 52, No. 5, 2014, pp. 935–951. doi:10.2514/1.J052255.
- [8] Kenway, G. K. W., and Martins, J. R. R. A., “Multipoint High-Fidelity Aerostructural Optimization of a Transport Aircraft Configuration,” *Journal of Aircraft*, Vol. 51, No. 1, 2014, pp. 144–160. doi:10.2514/1.C032150.
- [9] Variyar, A., Economou, T. D., and Alonso, J. J., “Design and Optimization of Unconventional Aircraft Configurations with Aeroelastic Constraints,” Grapevine, TX, 2017. doi:10.2514/6.2017-0463, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2017-0463>.

- [10] Afonso, F., Vale, J., Oliveira, É., Lau, F., and Suleman, A., “A review on Non-Linear Aeroelasticity of High Aspect-Ratio Wings,” *Progress in Aerospace Sciences*, Vol. 89, No. Supplement C, 2017, pp. 40–57. doi:10.1016/j.paerosci.2016.12.004.
- [11] Livne, E., “Integrated Aeroservoelastic Optimization: Status and Direction,” *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 122–145. doi:10.2514/2.2419.
- [12] Livne, E., “Aircraft Active Flutter Suppression: State of the Art and Technology Maturation Needs,” *Journal of Aircraft*, Vol. 55, No. 1, 2017, pp. 410–452. doi:10.2514/1.C034442, URL <https://doi.org/10.2514/1.C034442>.
- [13] Yu, Y., Lyu, Z., Xu, Z., and Martins, J. R. R. A., “On the Influence of Optimization Algorithm and Starting Design on Wing Aerodynamic Shape Optimization,” *Aerospace Science and Technology*, Vol. 75, 2018, pp. 183–199. doi:10.1016/j.ast.2018.01.016.
- [14] Su, W., and Cesnik, C. E. S., “Nonlinear Aeroelasticity of a Very Flexible Blended-Wing-Body Aircraft,” *Journal of Aircraft*, Vol. 47, No. 5, 2010, pp. 1539–1553. doi:10.2514/1.47317.
- [15] Cesnik, C. E., Palacios, R., and Reichenbach, E. Y., “Reexamined Structural Design Procedures for Very Flexible Aircraft,” *Journal of Aircraft*, Vol. 51, No. 5, 2014, pp. 1580–1591. doi:10.2514/1.C032464.
- [16] Dowell, E., Edwards, J., and Strganac, T., “Nonlinear Aeroelasticity,” *Journal of Aircraft*, Vol. 40, No. 5, 2003, pp. 857 – 874. doi:10.2514/2.6876.
- [17] Livne, E., and Weisshaar, T. A., “Aeroelasticity of Nonconventional Airplane Configurations-Past and Future,” *Journal of Aircraft*, Vol. 40, No. 6, 2003, pp. 1047–1065. doi:10.2514/2.7217, URL <https://doi.org/10.2514/2.7217>.
- [18] Livne, E., “Future of Airplane Aeroelasticity,” *Journal of Aircraft*, Vol. 40, No. 6, 2003, pp. 1066–1092.
- [19] Friedmann, P. P., “Renaissance of Aeroelasticity and Its Future,” *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 105–121. doi:10.2514/2.2418.
- [20] de C. Henshaw, M. J., Badcock, K. J., Vio, G. A., Allen, C. B., Chamberlain, J., Kaynes, I., Dimitriadis, G., Cooper, J. E., Woodgate, M. A., Rampurawala, A. M., Jones, D., Fenwick, C., Gaitonde, A. L., Taylor, N. V., Amor, D. S., Eccles, T. A., and Denley, C. J., “Non-linear aeroelastic prediction for aircraft applications,” *Progress in Aerospace Sciences*, Vol. 43, No. 4, 2007, pp. 65–137. doi:<https://doi.org/10.1016/j.paerosci.2007.05.002>, URL <http://www.sciencedirect.com/science/article/pii/S0376042107000462>.
- [21] Dimitriadis, G., *Introduction to Nonlinear Aeroelasticity*, John Wiley and Sons, 2017. doi:10.1002/9781118756478.

- [22] Beran, P., Stanford, B., and Schrock, C., “Uncertainty Quantification in Aeroelasticity,” *Annual Review of Fluid Mechanics*, Vol. 49, No. 1, 2017, pp. 361–386. doi:10.1146/annurev-fluid-122414-034441.
- [23] Martins, J. R. R. A., and Lambe, A. B., “Multidisciplinary Design Optimization: A Survey of Architectures,” *AIAA Journal*, Vol. 51, No. 9, 2013, pp. 2049–2075. doi:10.2514/1.J051895.
- [24] Chittick, I. R., and Martins, J. R. R. A., “An Asymmetric Suboptimization Approach to Aerostructural Optimization,” *Optimization and Engineering*, Vol. 10, No. 1, 2009, pp. 133–152. doi:10.1007/s11081-008-9046-2.
- [25] Lyu, Z., Kenway, G. K. W., and Martins, J. R. R. A., “Aerodynamic Shape Optimization Investigations of the Common Research Model Wing Benchmark,” *AIAA Journal*, Vol. 53, No. 4, 2015, pp. 968–985. doi:10.2514/1.J053318.
- [26] Lambe, A. B., Martins, J. R. R. A., and Kennedy, G. J., “An Evaluation of Constraint Aggregation Strategies for Wing Box Mass Minimization,” *Structural and Multidisciplinary Optimization*, Vol. 55, No. 1, 2017, pp. 257–277. doi:10.1007/s00158-016-1495-1.
- [27] Lyu, Z., Xu, Z., and Martins, J. R. R. A., “Benchmarking Optimization Algorithms for Wing Aerodynamic Design Optimization,” *Proceedings of the 8th International Conference on Computational Fluid Dynamics*, Chengdu, Sichuan, China, 2014. ICCFD8-2014-0203.
- [28] Cox, S. E., Haftka, R. T., Baker, C. A., Grossman, B., Mason, W. H., and Watson, L. T., “A Comparison of Global Optimization Methods for the Design of a High-speed Civil Transport,” *Journal of Global Optimization*, Vol. 21, No. 4, 2001, pp. 415–432. doi:10.1023/A:1012782825166, URL <https://doi.org/10.1023/A:1012782825166>.
- [29] Bons, N. P., He, X., Mader, C. A., and Martins, J. R. R. A., “Multimodality in Aerodynamic Wing Design Optimization,” *AIAA Journal*, Vol. 57, No. 3, 2019, pp. 1004–1018. doi:10.2514/1.J057294.
- [30] Martins, J. R. R. A., Sturdza, P., and Alonso, J. J., “The Complex-Step Derivative Approximation,” *ACM Transactions on Mathematical Software*, Vol. 29, No. 3, 2003, pp. 245–262. doi:10.1145/838250.838251.
- [31] Griewank, A., *Evaluating Derivatives*, SIAM, Philadelphia, 2000.
- [32] Naumann, U., *The Art of Differentiating Computer Programs—An Introduction to Algorithmic Differentiation*, SIAM, 2011.
- [33] Martins, J. R. R. A., and Hwang, J. T., “Review and Unification of Methods for Computing Derivatives of Multidisciplinary Computational Models,” *AIAA Journal*, Vol. 51, No. 11, 2013, pp. 2582–2599. doi:10.2514/1.J052184.

- [34] Jameson, A., “Aerodynamic Design via Control Theory,” *Journal of Scientific Computing*, Vol. 3, No. 3, 1988, pp. 233–260. doi:10.1007/BF01061285.
- [35] Kenway, G. K. W., Mader, C. A., He, P., and Martins, J. R. R. A., “Effective Adjoint Approaches for Computational Fluid Dynamics,” *Progress in Aerospace Sciences*, Vol. 110, 2019, p. 100542. doi:10.1016/j.paerosci.2019.05.002.
- [36] Hwang, J. T., and Martins, J. R. R. A., “A computational architecture for coupling heterogeneous numerical models and computing coupled derivatives,” *ACM Transactions on Mathematical Software*, Vol. 44, No. 4, 2018, p. Article 37. doi:10.1145/3182393.
- [37] Gray, J. S., Hwang, J. T., Martins, J. R. R. A., Moore, K. T., and Naylor, B. A., “OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization,” *Structural and Multidisciplinary Optimization*, Vol. 59, No. 4, 2019, pp. 1075–1104. doi:10.1007/s00158-019-02211-z.
- [38] Marsden, J. E., and McCracken, M., *The Hopf Bifurcation and Its Applications*, Springer-Verlag, 1976.
- [39] Farmer, M., and Hanson, P., “Comparison of Supercritical and Conventional Wing Flutter Characteristics,” *Structures, Structural Dynamics, and Materials and Co-located Conferences*, American Institute of Aeronautics and Astronautics, 1976, pp. –. doi:10.2514/6.1976-1560, URL <https://doi.org/10.2514/6.1976-1560>.
- [40] Isogai, K., “On the Transonic-Dip Mechanism of Flutter of a Sweptback Wing,” *AIAA Journal*, Vol. 17, No. 7, 1979, pp. 793–795. doi:10.2514/3.61226, URL <https://doi.org/10.2514/3.61226>.
- [41] Isogai, K., “Transonic dip mechanism of flutter of a sweptback wing. II,” *AIAA Journal*, Vol. 19, No. 9, 1981, pp. 1240–1242. doi:10.2514/3.7853, URL <https://doi.org/10.2514/3.7853>.
- [42] Kholodar, D. B., Thomas, J. P., Dowell, E. H., and Hall, K. C., “Parametric Study of Flutter for an Airfoil in Inviscid Transonic Flow,” *Journal of Aircraft*, Vol. 40, No. 2, 2003, pp. 303–313. doi:10.2514/2.3094, URL <https://doi.org/10.2514/2.3094>.
- [43] Kholodar, D. B., Dowell, E. H., Thomas, J. P., and Hall, K. C., “Improved Understanding of Transonic Flutter: A Three-Parameter Flutter Surface,” *Journal of Aircraft*, Vol. 41, No. 4, 2004, pp. 911–917. doi:10.2514/1.467, URL <https://doi.org/10.2514/1.467>.
- [44] Bennett, R. M., Batina, J. T., and Cunningham, H. J., “Wing-flutter calculations with the CAP-TSD unsteady transonic small-disturbance program,” *Journal of Aircraft*, Vol. 26, No. 9, 1989, pp. 876–882. doi:10.2514/3.45854, URL <https://doi.org/10.2514/3.45854>.



- [45] Schuster, D. M., Liu, D. D., and Huttzell, L. J., “Computational Aeroelasticity: Success, Progress, Challenge,” *Journal of Aircraft*, Vol. 40, No. 5, 2003, pp. 843–856. doi:10.2514/2.6875.
- [46] Taylor, N. V., Allen, C. B., Gaitonde, A. L., Jones, D. P., Vio, G. A., Cooper, J. E., Rampurawala, A. M., Badcock, K. J., Woodgate, M. A., Henshaw, M. J. d. C., and et al., “Aeroelastic analysis through linear and non-linear methods: a summary of flutter prediction in the PUMA DARP,” *The Aeronautical Journal*, Vol. 110, No. 1107, 2006, pp. 333–343. doi:10.1017/S0001924000013208.
- [47] Lee-Rausch, E. M., and Batina, J. T., “Wing flutter boundary prediction using unsteady Euler aerodynamic method,” *Journal of Aircraft*, Vol. 32, No. 2, 1995, pp. 416–422. doi:10.2514/3.46732.
- [48] Robinson, B. A., Yang, H. T., and Batina, J. T., “Aeroelastic Analysis of Wings using the Euler Equations with a Deforming Mesh,” *Journal of Aircraft*, Vol. 28, No. 11, 1991, pp. 781–788. doi:10.2514/3.46096.
- [49] Gibbons, M. D., “Aeroelastic Calculations Using CFD for a Typical Business Jet Model,” Tech. rep., NASA Contractor Report 4753, 1996.
- [50] Lee-Rausch, E., and Baitina, J. T., “Wing flutter computations using an aerodynamic model based on the Navier-Stokes equations,” *Journal of Aircraft*, Vol. 33, No. 6, 1996, pp. 1139–1147. doi:10.2514/3.47068.
- [51] Gordnier, R. E., and Melville, R. B., “Transonic Flutter Simulations Using an Implicit Aeroelastic Solver,” *Journal of Aircraft*, Vol. 37, No. 5, 2000, pp. 872–879. doi:10.2514/2.2683.
- [52] Palacios, R., Climent, H., Karlsson, A., and Winzell, B., “Assessment of strategies for correcting linear unsteady aerodynamics using CFD or experimental results,” *International Forum for Aeroelasticity and Structural Dynamics*, 2001.
- [53] Chen, P. C., Gao, X. W., and Tang, L., “Overset Field-Panel Method for Unsteady Transonic Aerodynamic Influence Coefficient Matrix Generation,” *AIAA Journal*, Vol. 42, No. 9, 2004. doi:10.2514/1.4390, URL <https://arc.aiaa.org/doi/pdf/10.2514/1.4390>.
- [54] Liu, D., Kao, Y. F., and Fung, K. Y., “An efficient method for computing unsteady transonic aerodynamics of swept wings with control surfaces,” *Journal of Aircraft*, Vol. 25, No. 1, 1988, pp. 25–31. doi:10.2514/3.45536, URL <https://doi.org/10.2514/3.45536>.
- [55] Chen, P. C., Sarhaddi, D., and Liu, D. D., “Transonic-Aerodynamic-Influence-Coefficient Approach for Aeroelastic and MDO Applications,” *Journal of Aircraft*, Vol. 37, No. 1, 2000, pp. 85–94. doi:10.2514/2.2565, URL <https://doi.org/10.2514/2.2565>.

- [56] Kreiselmaier, E., and Laschka, B., “Small Disturbance Euler Equations: Efficient and Accurate Tool for Unsteady Load Prediction,” *Journal of Aircraft*, Vol. 37, No. 5, 2000, pp. 770–778. doi:10.2514/2.2699, URL <https://doi.org/10.2514/2.2699>.
- [57] Pechloff, A. N., and Laschka, B., “Small Disturbance Navier–Stokes Method: Efficient Tool for Predicting Unsteady Air Loads,” *Journal of Aircraft*, Vol. 43, No. 1, 2006, pp. 17–29. doi:10.2514/1.14350, URL <https://doi.org/10.2514/1.14350>.
- [58] Fleischer, D., and Breitsamter, C., *CFD Based Methods for the Computation of Generalized Aerodynamic Forces*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 331–338. doi:10.1007/978-3-642-35680-3\_40, URL [https://doi.org/10.1007/978-3-642-35680-3\\_40](https://doi.org/10.1007/978-3-642-35680-3_40).
- [59] Fleischer, D., and Breitsamter, C., “Efficient Computation of Unsteady Aerodynamic Loads Using Computational-Fluid-Dynamics Linearized Methods,” *Journal of Aircraft*, Vol. 50, No. 2, 2013, pp. 425–440. doi:10.2514/1.C031851, URL <https://doi.org/10.2514/1.C031851>.
- [60] Thormann, R., and Widhalm, M., “Linear-Frequency-Domain Predictions of Dynamic-Response Data for Viscous Transonic Flows,” *AIAA Journal*, Vol. 51, No. 11, 2013, pp. 2540–2557. doi:10.2514/1.J051896, URL <http://arc.aiaa.org/doi/10.2514/1.J051896>.
- [61] Saad, Y., *Iterative Methods for Sparse Linear Systems*, 2<sup>nd</sup> ed., SIAM, 2003.
- [62] Widhalm, M., and Thormann, R., “Efficient Evaluation of Dynamic Response Data with a Linearized Frequency Domain Solver at Transonic Separated Flow Condition,” *AIAA Aviation Forum*, American Institute of Aeronautics and Astronautics, 2017, pp. –. doi:10.2514/6.2017-3905, URL <https://doi.org/10.2514/6.2017-3905>.
- [63] Skujins, T., and Cesnik, C. E. S., “Reduced-Order Modeling of Unsteady Aerodynamics Across Multiple Mach Regimes,” *Journal of Aircraft*, Vol. 51, No. 6, 2014, pp. 1681–1704. doi:10.2514/1.c032222.
- [64] Kitson, R., Lupp, C. A., and Cesnik, C. E. S., “Modeling and Simulation of Flexible Jet Transport Aircraft with High-Aspect-Ratio Wings,” *15th Dynamics Specialists Conference*, San Diego, CA, 2016. doi:10.2514/6.2016-2046.
- [65] Mallik, W., Schetz, J. A., and Kapania, R. K., “Rapid Transonic Flutter Analysis for Aircraft Conceptual Design Applications,” *AIAA Journal*, Vol. 56, No. 6, 2018, pp. 2389–2402. doi:10.2514/1.J056218.
- [66] Leishman, J., “Validation of Approximate Indicial Aerodynamic Functions for Two-Dimensional Subsonic Flow,” *Journal of Aircraft*, Vol. 25, No. 10, 1988, pp. 914–922. doi:10.2514/3.45680.

- [67] Opgenoord, M. M. J., Drela, M., and Willcox, K. E., “Physics-Based Low-Order Model for Transonic Flutter Prediction,” *AIAA Journal*, Vol. 56, No. 4, 2018, pp. 1519–1531. doi:10.2514/1.J056710, URL <https://doi.org/10.2514/1.J056710>.
- [68] Opgenoord, M. M. J., Drela, M., and Willcox, K. E., “Influence of Transonic Flutter on the Conceptual Design of Next-Generation Transport Aircraft,” *AIAA Journal*, 2019, pp. 1–15. doi:10.2514/1.j057302.
- [69] Opgenoord, M. M. J., and Willcox, K. E., “Design Methodology for Aeroelastic Tailoring of Additively Manufactured Lattice Structures Using Low-Order Methods,” *AIAA Journal*, Vol. 57, No. 11, 2019, pp. 4903–4914. doi:10.2514/1.J058169, URL <https://doi.org/10.2514/1.J058169>.
- [70] Patil, M. J., Hodges, D. H., and Cesnik, C. E. S., “Nonlinear Aeroelasticity and Flight Dynamics of High-Altitude Long-Endurance Aircraft,” *Journal of Aircraft*, Vol. 38, No. 1, 2001, pp. 88–94. doi:10.2514/2.2738.
- [71] Patil, M. J., and Hodges, D. H., “Flight Dynamics of Highly Flexible Flying Wings,” *Journal of Aircraft*, Vol. 43, No. 6, 2006, pp. 1790–1799. doi:10.2514/1.17640.
- [72] Su, W., and Cesnik, C. E. S., “Dynamic Response of Highly Flexible Flying Wings,” *AIAA Journal*, Vol. 49, No. 2, 2011, pp. 324–339. doi:10.2514/1.J050496, uM/NAST reference.
- [73] Niblett, L. T., “The Fundamentals of Body-Freedom Flutter,” *Aeronautical Journal*, Vol. 90, No. 899, 1986, pp. 373–377. doi:10.1017/S0001924000015979.
- [74] Mazidi, A., Fazelzadeh, S. A., and Marzocca, P., “Flutter of Aircraft Wings Carrying a Powered Engine Under Roll Maneuver,” *Journal of Aircraft*, Vol. 48, No. 3, 2011, pp. 874–883. doi:10.2514/1.C031080.
- [75] Jones, J. R., and Cesnik, C. E. S., “Nonlinear Aeroelastic Analysis of the X-56A Multi-Utility Aeroelastic Demonstrator,” *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, San Diego, CA, 2016. doi:10.2514/6.2016-1799.
- [76] Cesnik, C., and Su, W., “Nonlinear Aeroelastic Simulation of X-HALE: a Very Flexible UAV,” *49th AIAA Aerospace Sciences Meeting*, American Institute of Aeronautics and Astronautics, Orlando, FL, 2011. doi:10.2514/6.2011-1226.
- [77] S. Cesnik, C. E., Senatore, P. J., Atkins, E. M., Su, W., and Shearer, C. M., “X-HALE: A Very Flexible Unmanned Aerial Vehicle for Nonlinear Aeroelastic Tests,” *AIAA Journal*, Vol. 50, No. 12, 2012, pp. 2820–2833. doi:10.2514/1.j051392.

- [78] Patil, M., “Nonlinear Gust Response of Highly Flexible Aircraft,” *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials and Co-located Conferences*, Honolulu, HI, 2007. doi:10.2514/6.2007-2103.
- [79] Richards, P. W., Yao, Y., Herd, R. A., Hodges, D. H., and Mardanpour, P., “Effect of Inertial and Constitutive Properties on Body-Freedom Flutter for Flying Wings,” *Journal of Aircraft*, Vol. 53, No. 3, 2016, pp. 756–767. doi: 10.2514/1.C033435.
- [80] Cavallaro, R., Bombardieri, R., Demasi, L., and Iannelli, A., “PrandtlPlane Joined Wing: Body Freedom Flutter, Limit Cycle Oscillation and Freeplay Studies,” *Journal of Fluids and Structures*, Vol. 59, No. January, 2015, pp. 57–84. doi:10.1016/j.jfluidstructs.2015.08.016.
- [81] Hassig, H. J., “An approximate true damping solution of the flutter equation by determinant iteration.” *Journal of Aircraft*, Vol. 8, No. 11, 1971, pp. 885–889. doi:10.2514/3.44311.
- [82] Wright, J. R., and Jonathan, E., *Introduction to Aircraft Aeroelasticity and Loads*, Wiley, 2007. doi:10.1002/9781118700440.
- [83] Hodges, D. H., and Pierce, G. A., *Introduction to Structural Dynamics and Aeroelasticity*, 2<sup>nd</sup> ed., Vol. 15, Cambridge University Press, 2011. doi:10.1017/CBO9780511997112.
- [84] Rodden, W. P., *Theoretical and computational aeroelasticity*, Crest Pub., 2011.
- [85] Chen, P., “Damping perturbation method for flutter solution: the g-method,” *AIAA journal*, Vol. 38, No. 9, 2000, pp. 1519–1524. doi:10.2514/2.1171.
- [86] Bennett, R. M., and Desmarais, R. N., “Curve Fitting of Aeroelastic Transient Response Data with Exponential Functions,” Tech. rep., NASA Langley Research Center, 1976.
- [87] Bousman, W., and Winkler, D., “Application on the moving-block analysis,” *Structures, Structural Dynamics, and Materials and Co-located Conferences*, American Institute of Aeronautics and Astronautics, 1981, pp. –. doi:10.2514/6.1981-653, URL <https://doi.org/10.2514/6.1981-653>.
- [88] Hua, Y., and Sarkar, T. K., “Matrix Pencil Method for Estimating Parameters of Exponentially Damped/Undamped Sinusoids in Noise,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 38, No. 5, 1990, pp. 814–824. doi:10.1109/29.56027.
- [89] Pak, C.-G., and Friedmann, P. P., “New Time-Domain Technique for Flutter Boundary Identification,” *33rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Dallas, TX, 1992. doi: 10.2514/6.1992-2102.

- [90] Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N.-C., Tung, C. C., and Liu, H. H., “The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis,” *Proceedings of the Royal Society of London A: mathematical, physical and engineering sciences*, Vol. 454, The Royal Society, 1998, pp. 903–995. doi:10.1098/rspa.1998.0193.
- [91] McNamara, J. J., and Friedmann, P. P., “Flutter Boundary Identification for Time-Domain Computational Aeroelasticity,” *AIAA Journal*, Vol. 45, No. 7, 2007, pp. 1546–1555. doi:10.2514/1.26706, URL <https://doi.org/10.2514/1.26706>.
- [92] Hallissy, B., and Cesnik, C., “High-fidelity Aeroelastic Analysis of Very Flexible Aircraft,” *Structures, Structural Dynamics, and Materials and Co-located Conferences*, American Institute of Aeronautics and Astronautics, 2011, pp. –. doi:10.2514/6.2011-1914, URL <https://doi.org/10.2514/6.2011-1914>.
- [93] Jacobson, K. E., Kiviaho, J. F., Kennedy, G. J., and Smith, M. J., “Evaluation of time-domain damping identification methods for flutter-constrained optimization,” *Journal of Fluids and Structures*, Vol. 87, 2019, pp. 174–188. doi:10.1016/j.jfluidstructs.2019.03.011.
- [94] Kim, N., and Choi, K., “Design Sensitivity Analysis and Optimization of Nonlinear Transient Dynamics,” *8th Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, 2000. doi:10.2514/6.2000-4905.
- [95] Trier, A., S. Marthinsen, and Sivertsen, O., “Design Sensitivities by the Adjoint Variable Method in Nonlinear Structural Dynamics,” *SIMS Simulation Conference*, Trondheim, Norway, 1996.
- [96] Beran, B., P. Stanford, and Kurdi, M., “Sensitivity Analysis and Optimization of Dynamic Systems with Reduced Order Modeling,” *AIAA Aerospace Sciences Meeting*, Orlando, FL, 2010. doi:10.2514/6.2010-1503.
- [97] Mani, K., and Mavriplis, D. J., “Adjoint-Based Sensitivity Formulation for Fully Coupled Unsteady Aeroelasticity Problems,” *AIAA Journal*, Vol. 47, No. 8, 2009, pp. 1902–1915. doi:10.2514/1.40582.
- [98] Mishra, A., Mani, K., Mavriplis, D., and Sitaraman, J., “Time dependent adjoint-based optimization for coupled fluidstructure problems,” *Journal of Computational Physics*, Vol. 292, 2015, pp. 253 – 271. doi:10.1016/j.jcp.2015.03.010.
- [99] Zhang, Z., Chen, P. C., Yang, S., Wang, Z., and Wang, Q., “Unsteady Aerostructure Coupled Adjoint Method for Flutter Suppression,” *AIAA Journal*, Vol. 53, No. 8, 2015, pp. 2121–2129. doi:10.2514/1.J053495, URL <https://doi.org/10.2514/1.J053495>.

- [100] Zhang, Z., Chen, P.-C., Wang, Q., Zhou, Z., Yang, S., and Wang, Z., “Adjoint Based Structure and Shape Optimization with Flutter Constraints,” *AIAA SciTech Forum*, American Institute of Aeronautics and Astronautics, 2016, pp. –. doi:10.2514/6.2016-1176, URL <https://doi.org/10.2514/6.2016-1176>.
- [101] Kiviaho, J. F., Jacobson, K., Smith, M. J., and Kennedy, G., “A Robust and Flexible Coupling Framework for Aeroelastic Analysis and Optimization,” *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Denver, CO, 2017. doi:10.2514/6.2017-4144.
- [102] Jacobson, K., Kiviaho, J. F., Smith, M. J., and Kennedy, G., “An Aeroelastic Coupling Framework for Time-accurate Analysis and Optimization,” *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Kissimmee, FL, 2018. doi:10.2514/6.2018-0100.
- [103] Haftka, R. T., “Parametric Constraints with Application to Optimization for Flutter Using a Continuous Flutter Constraint,” *AIAA Journal*, Vol. 13, No. 4, 1975, pp. 471–475. doi:10.2514/3.49733, URL <http://dx.doi.org/10.2514/3.49733>.
- [104] Hajel, P., “A Root Locus-Based Flutter Synthesis Procedure,” *Journal of Aircraft*, Vol. 20, No. 12, 1983, pp. 1021–1027. doi:10.2514/3.48206.
- [105] Turner, M. J., “Optimization of structures to satisfy flutter requirements.” *AIAA Journal*, Vol. 7, No. 5, 1969, pp. 945–951. doi:10.2514/3.5248, URL <https://doi.org/10.2514/3.5248>.
- [106] Bhatia, K. G., and Rudisill, C. S., “Optimization of complex structures to satisfy flutter requirements,” *AIAA Journal*, Vol. 9, No. 8, 1971, pp. 1487–1491. doi:10.2514/3.6389, URL <https://doi.org/10.2514/3.6389>.
- [107] Rudisill, C. S., and Bhatia, K. G., “Second derivatives of the flutter velocity and the optimization of aircraft structures.” *AIAA Journal*, Vol. 10, No. 12, 1972, pp. 1569–1572. doi:10.2514/3.6690, URL <https://doi.org/10.2514/3.6690>.
- [108] Gwin, L., and Taylor, R., “A General Method for Flutter Optimization,” *AIAA Journal*, Vol. 11, No. 12, 1973, pp. 1613–1617. doi:10.2514/3.50657, URL <https://doi.org/10.2514/3.50657>.
- [109] Ringertz, U., “On structural optimization with aeroelasticity constraints,” *Structural optimization*, Vol. 8, No. 1, 1994, pp. 16–23. doi:10.1007/BF01742928.
- [110] Mallik, W., Kapania, R. K., and Schetz, J. A., “Effect of Flutter on the Multidisciplinary Design Optimization of Truss-Braced-Wing Aircraft,” *Journal of Aircraft*, Vol. 52, No. 6, 2015, pp. 1858–1872. doi:10.2514/1.C033096.

- [111] Stanford, B. K., Jutte, C. V., and Wieseman, C. D., “Trim and Structural Optimization of Subsonic Transport Wings Using Nonconventional Aeroelastic Tailoring,” *AIAA Journal*, Vol. 54, No. 1, 2015, pp. 293–309. doi:10.2514/1.J054244.
- [112] Chen, P. C., Zhang, Z., and Livne, E., “Design-Oriented Computational Fluid Dynamics-Based Unsteady Aerodynamics for Flight-Vehicle Aeroelastic Shape Optimization,” *AIAA Journal*, Vol. 53, No. 12, 2015, pp. 3603–3619. doi:10.2514/1.J054024, URL <http://arc.aiaa.org/doi/10.2514/1.J054024>.
- [113] Bartels, R. E., and Stanford, B. K., “Aeroelastic Optimization with an Economical Transonic Flutter Constraint Using Navier-Stokes Aerodynamics,” *Journal of Aircraft*, Vol. 55, No. 4, 2018, pp. 1522–1530. doi:10.2514/1.C034675, URL <https://arc.aiaa.org/doi/pdf/10.2514/1.C034675><https://arc.aiaa.org/doi/10.2514/1.C034675>.
- [114] Xie, G., Liu, X., and Yan, H., “Film cooling performance and flow characteristics of internal cooling channels with continuous/truncated ribs,” *International Journal of Heat and Mass Transfer*, Vol. 105, 2017, pp. 67–75.
- [115] Bhatia, M., and Beran, P., “Design of Thermally Stressed Panels Subject to Transonic Flutter Constraints,” *Journal of Aircraft*, Vol. 54, No. 6, 2017, pp. 2340–2349. doi:10.2514/1.C034301.
- [116] Lupp, C. A., and Cesnik, C. E., “A Gradient-Based Flutter Constraint Including Geometrically Nonlinear Deformations,” *AIAA SciTech Forum*, American Institute of Aeronautics and Astronautics, 2019, pp. –. doi:10.2514/6.2019-1212, URL <https://doi.org/10.2514/6.2019-1212>.
- [117] Kennedy, G. J., Kenway, G. K. W., and Martins, J. R. R. A., “Towards Gradient-Based Design Optimization of Flexible Transport Aircraft with Flutter Constraints,” *Proceedings of the 15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Atlanta, GA, 2014. doi:10.2514/6.2014-2726, aIAA 2014-2726.
- [118] Beran, P., Stanford, B. K., and Wang, K. G., “Fast Prediction of Flutter and Flutter Sensitivities,” *17th International Forum on Aeroelasticity and Structural Dynamics (IFASD)*, Como, Italy, 2017. doi:10.2514/6.2017-1350.
- [119] Albano, E., and Rodden, W. P., “A doublet-lattice method for calculating lift distributions on oscillating surfaces in subsonic flows.” *AIAA Journal*, Vol. 7, No. 2, 1969, pp. 279–285. doi:10.2514/3.5086, URL <https://arc.aiaa.org/doi/abs/10.2514/3.5086>.
- [120] Rodden, W. P., Taylor, P. F., and McIntosh, S. C., “Further Refinement of the Subsonic Doublet-Lattice Method,” *Journal of Aircraft*, Vol. 35, No. 5, 1998, pp. 720–727. doi:10.2514/2.2382, URL <http://dx.doi.org/10.2514/2.2382>.

- [121] Bhatia, M., Kapania, R., Gur, O., Schetz, J., Mason, W., and Haftka, R., “Progress Towards Multidisciplinary Design Optimization of Truss Braced Wing Aircraft With Flutter Constraints,” *13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Fort Worth, TX, 2010. doi:10.2514/6.2010-9077.
- [122] Theodorsen, T., “General Theory of Aerodynamic Instability and the Mechanism of Flutter,” Tech. rep., NACA Report 496, 1935.
- [123] Haftka, R. T., Sobieszczanski-Sobieski, J., and Padula, S. L., “On Options for Interdisciplinary Analysis and Design Optimization,” *Structural Optimization*, Vol. 4, 1992, pp. 65–74. doi:10.1007/BF01759919.
- [124] Kreisselmeier, G., and Steinhauser, R., “Systematic Control Design by Optimizing a Vector Performance Index,” *International Federation of Active Controls Symposium on Computer-Aided Design of Control Systems, Zurich, Switzerland*, 1979. doi:10.1016/S1474-6670(17)65584-8.
- [125] Brooks, T. R., Kenway, G. K. W., and Martins, J. R. R. A., “Benchmark Aerostructural Models for the Study of Transonic Aircraft Wings,” *AIAA Journal*, Vol. 56, No. 7, 2018, pp. 2840–2855. doi:10.2514/1.J056603.
- [126] Chen, P., Liu, D., and Livne, E., “Unsteady-aerodynamic shape sensitivities for airplane aeroservoelastic configuration optimization,” *Journal of Aircraft*, Vol. 43, No. 2, 2006, pp. 471–481. doi:10.2514/1.10007.
- [127] Van Nunen, J. W. G., Tijdeman, H., and N., K. A., “Results of transonic wind tunnel measurements on an oscillating wing with external store (data report),” Tech. rep., DTIC Document, NLR TR 78030 U, 1978.
- [128] Tijdeman, H., Van Nunen, J. W. G., and N., K. A., “Transonic wind tunnel tests of an oscillating wing with external store, Parts I-IV,” Tech. rep., DTIC Document, NLR TR 78106 U, 1978.
- [129] Roger, K. L., “Airplane Math Modeling Methods For Active Control Design,” *AGARD-CP-288*, , No. 228, 1977, pp. 4. 1–4. 11.
- [130] Bartels, R., “Flexible Launch Vehicle Stability Analysis Using Steady and Unsteady Computational Fluid Dynamics,” *Journal of Spacecraft and Rockets*, Vol. 49, No. 4, 2012, pp. 644–650. doi:10.2514/1.A32082.
- [131] Lukaczyk, T., Wendor, A., Boteroz, E., Macdonaldz, T., Momosez, T., Variyarz, A., Veghz, J., Colonnnox, M., Economon, T., Alonsok, J., Orra, T., and Da Silvayy, C., “SUAVE: An Open-Source Environment for Multi-Fidelity Conceptual Vehicle Design,” *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Dallas, TX, 2015. doi:10.2514/6.2015-3087.



- [132] Drela, M., “Integrated Simulation Model for Preliminary Aerodynamic, Structural, and Control-Law Design of Aircraft,” *40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, St. Louis, MO, 1999. doi:10.2514/6.1999-1394.
- [133] Bradley, M. K., and K., D. C., “Subsonic Ultra Green Aircraft Research Phase II: N+4 Advanced Concept Development,” Tech. rep., NASA Contract Report 2012-217556, 2012.
- [134] Jonsson, E., Mader, C. A., Kennedy, G., and Martins, J., “Computational Modeling of Flutter Constraint for High-Fidelity Aerostructural Optimization,” *AIAA SciTech Forum*, American Institute of Aeronautics and Astronautics, 2019, pp. –. doi:10.2514/6.2019-2354, URL <https://doi.org/10.2514/6.2019-2354>.
- [135] Klock, R. J., and Cesnik, C. E. S., “Aerothermoelastic Simulation of Air-Breathing Hypersonic Vehicle,” *55th AIAA/ASME/ASCE/AHS/SC Structures, Structural Dynamics, and Materials Conference*, National Harbor, MD, 2014. doi:10.2514/6.2014-0149.
- [136] Badcock, K. J., Woodgate, M. a., and Richards, B. E., “Direct Aeroelastic Bifurcation Analysis of a Symmetric Wing Based on the Euler Equations,” *Journal of Aircraft*, Vol. 42, No. 3, 2005, pp. 731–737. doi:10.2514/1.5323, URL <http://arc.aiaa.org/doi/abs/10.2514/1.5323>.
- [137] Badcock, K. J., and Woodgate, M. A., “Bifurcation Prediction of Large-Order Aeroelastic Models,” *AIAA Journal*, Vol. 48, No. 6, 2010, pp. 1037–1046. doi:10.2514/1.40961, URL <http://arc.aiaa.org/doi/10.2514/1.40961>.
- [138] Kennedy, G. J., and Martins, J. R. R. A., “A parallel aerostructural optimization framework for aircraft design studies,” *Structural and Multidisciplinary Optimization*, Vol. 50, No. 6, 2014, pp. 1079–1101. doi:10.1007/s00158-014-1108-9.
- [139] Tang, D., and Dowell, E. H., “Experimental and Theoretical Study on Aeroelastic Response of High-Aspect-Ratio Wings,” *AIAA Journal*, Vol. 39, 2001, pp. 1430–1441. doi:10.2514/2.1484.
- [140] Tang, D., and Dowell, E. H., “Limit-Cycle Hysteresis Response for a High-Aspect-Ratio Wing Model,” *Journal of Aircraft*, Vol. 39, 2002, pp. 885–888. doi:10.2514/2.3009.
- [141] Holden, M. E., “Optimization of Dynamic Systems Using Collocation Methods,” Ph.D. thesis, Stanford University, Stanford, CA, May 1999.
- [142] Hicks, R. M., and Henne, P. A., “Wing Design by Numerical Optimization,” *Journal of Aircraft*, Vol. 15, No. 7, 1978, pp. 407–412. doi:10.2514/3.58379, URL <https://doi.org/10.2514/3.58379>.

- [143] Schmitt, V., and Charpin, F., “Pressure distributions on the ONERA-M6-wing at transonic mach numbers,” Tech. rep., Office National d’Etudes et Recherches Aérospatiales, 92320, Chatillon, France, 1979.
- [144] Kurdi, M., Lindsley, N., and Beran, P., “Uncertainty quantification of the Goland wings flutter boundary,” *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, AIAA Hilton Head, SC, 2007, pp. 20–23. doi:10.2514/6.2007-6309.
- [145] Kiviaho, J. F., Jacobson, K., Smith, M. J., and Kennedy, G., “Application of a Time-Accurate Aeroelastic Coupling Framework to Flutter-Constrained Design Optimization,” *2018 Multidisciplinary Analysis and Optimization Conference*, 2018, p. 2932.
- [146] Stanford, B. K., and Dunning, P. D., “Optimal Topology of Aircraft Rib and Spar Structures Under Aeroelastic Loads,” *Journal of Aircraft*, Vol. 52, No. 4, 2014, pp. 1298–1311. doi:10.2514/1.C032913, URL <http://dx.doi.org/10.2514/1.C032913>.
- [147] Dunning, P. D., Stanford, B. K., Kim, H. A., and Jutte, C. V., “Aeroelastic tailoring of a plate wing with functionally graded materials,” *Journal of Fluids and Structures*, Vol. 51, No. Supplement C, 2014, pp. 292–312. doi:https://doi.org/10.1016/j.jfluidstructs.2014.09.008, URL <http://www.sciencedirect.com/science/article/pii/S0889974614002096>.
- [148] Mader, C. A., and Martins, J. R. R. A., “Stability-Constrained Aerodynamic Shape Optimization of Flying Wings,” *Journal of Aircraft*, Vol. 50, No. 5, 2013, pp. 1431–1449. doi:10.2514/1.C031956.
- [149] Lyu, Z., and Martins, J. R. R. A., “Aerodynamic Design Optimization Studies of a Blended-Wing-Body Aircraft,” *Journal of Aircraft*, Vol. 51, No. 5, 2014, pp. 1604–1617. doi:10.2514/1.C032491.
- [150] Mader, C. A., and Martins, J. R. R. A., “Computing Stability Derivatives and their Gradients for Aerodynamic Shape Optimization,” *AIAA Journal*, Vol. 52, No. 11, 2014, pp. 2533–2546. doi:10.2514/1.J052922.
- [151] Chen, S., Lyu, Z., Kenway, G. K. W., and Martins, J. R. R. A., “Aerodynamic Shape Optimization of the Common Research Model Wing-Body-Tail Configuration,” *Journal of Aircraft*, Vol. 53, No. 1, 2016, pp. 276–293. doi:10.2514/1.C033328.
- [152] Kenway, G. K. W., and Martins, J. R. R. A., “Buffet Onset Constraint Formulation for Aerodynamic Shape Optimization,” *AIAA Journal*, Vol. 55, No. 6, 2017, pp. 1930–1947. doi:10.2514/1.J055172.
- [153] He, X., Li, J., Mader, C. A., Yildirim, A., and Martins, J. R. R. A., “Robust aerodynamic shape optimization—from a circle to an airfoil,” *Aerospace Science and Technology*, Vol. 87, 2019, pp. 48–61. doi:10.1016/j.ast.2019.01.051.

- [154] Liem, R. P., Kenway, G. K. W., and Martins, J. R. R. A., “Multimission Aircraft Fuel Burn Minimization via Multipoint Aerostructural Optimization,” *AIAA Journal*, Vol. 53, No. 1, 2015, pp. 104–122. doi:10.2514/1.J052940.
- [155] Burdette, D. A., and Martins, J. R. R. A., “Design of a Transonic Wing with an Adaptive Morphing Trailing Edge via Aerostructural Optimization,” *Aerospace Science and Technology*, Vol. 81, 2018, pp. 192–203. doi:10.1016/j.ast.2018.08.004.
- [156] Brooks, T. R., Martins, J. R. R. A., and Kennedy, G. J., “High-fidelity Aerostructural Optimization of Tow-steered Composite Wings,” *Journal of Fluids and Structures*, Vol. 88, 2019, pp. 122–147. doi:10.1016/j.jfluidstructs.2019.04.005.
- [157] Kenway, G. K., Kennedy, G. J., and Martins, J. R. R. A., “A CAD-Free Approach to High-Fidelity Aerostructural Optimization,” *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, Fort Worth, TX, 2010. doi:10.2514/6.2010-9231.
- [158] Sederberg, T. W., and Parry, S. R., “Free-form Deformation of Solid Geometric Models,” *SIGGRAPH Comput. Graph.*, Vol. 20, No. 4, 1986, pp. 151–160. doi:10.1145/15886.15903.
- [159] Brown, S. A., “Displacement Extrapolation for CFD+CSM Aeroelastic Analysis,” *Proceedings of the 35th AIAA Aerospace Sciences Meeting*, Reno, NV, 1997. AIAA 1997-1090.
- [160] Yildirim, A., Kenway, G. K. W., Mader, C. A., and Martins, J. R. R. A., “A Jacobian-free approximate Newton–Krylov startup strategy for RANS simulations,” *Journal of Computational Physics*, Vol. 397, 2019, p. 108741. doi:10.1016/j.jcp.2019.06.018.
- [161] Coder, J. G., Pulliam, T. H., Hue, D., Kenway, G. K. W., and Sclafani, A. J., “Contributions to the 6th AIAA CFD Drag Prediction Workshop Using Structured Grid Methods,” *AIAA SciTech Forum*, American Institute of Aeronautics and Astronautics, 2017. doi:10.2514/6.2017-0960.
- [162] Kenway, G. K. W., Secco, N., Martins, J. R. R. A., Mishra, A., and Duraisamy, K., “An Efficient Parallel Overset Method for Aerodynamic Shape Optimization,” *Proceedings of the 58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA SciTech Forum*, Grapevine, TX, 2017. doi:10.2514/6.2017-0357.
- [163] Burdette, D. A., and Martins, J. R. R. A., “Impact of Morphing Trailing Edge on Mission Performance for the Common Research Model,” *Journal of Aircraft*, Vol. 56, No. 1, 2019, pp. 369–384. doi:10.2514/1.C034967.

- [164] Secco, N. R., and Martins, J. R. R. A., “RANS-based Aerodynamic Shape Optimization of a Strut-braced Wing with Overset Meshes,” *Journal of Aircraft*, Vol. 56, No. 1, 2019, pp. 217–227. doi:10.2514/1.C034934.
- [165] Luke, E., Collins, E., and Blades, E., “A Fast Mesh Deformation Method Using Explicit Interpolation,” *Journal of Computational Physics*, Vol. 231, No. 2, 2012, pp. 586–601. doi:10.1016/j.jcp.2011.09.021.
- [166] Gill, P. E., Murray, W., and Saunders, M. A., “SNOPT: An SQP algorithm for large-scale constrained optimization,” *SIAM Journal of Optimization*, Vol. 12, No. 4, 2002, pp. 979–1006. doi:10.1137/S1052623499350013.
- [167] Perez, R. E., Jansen, P. W., and Martins, J. R. R. A., “pyOpt: A Python-Based Object-Oriented Framework for Nonlinear Constrained Optimization,” *Structural and Multidisciplinary Optimization*, Vol. 45, No. 1, 2012, pp. 101–118. doi:10.1007/s00158-011-0666-3.
- [168] Jonsson, E., Kenway, G., and Martins, J. R. R. A., “Development of Flutter Constraints for High-fidelity Aerostructural Optimization,” *35th AIAA Applied Aerodynamics Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2017, p. 26. doi:10.2514/6.2017-4455, URL <https://arc.aiaa.org/doi/10.2514/6.2017-4455>.
- [169] Rodden, W. P., Bellinger, E. D., Harder, R. L., and Center, L. R., *Aeroelastic addition to NASTRAN*, National Aeronautics and Space Administration, Scientific and Technical Information Branch, 1979. URL <http://hdl.handle.net/2027/uiug.30112106862490>.
- [170] Blair, M., “A compilation of the mathematics leading to the doublet lattice method,” Tech. rep., DTIC Document, 1992.
- [171] *MSC Nastran 2012 Release Guide*, MSC Software Corp, 2012. URL <http://www.mssoftware.com/product/msc-nastran>.
- [172] Ablowitz, M. J., and Fokas, A. S., *Complex Variables: Introduction and Applications*, 2<sup>nd</sup> ed., Cambridge University Press, Cambridge, UK, 2003. doi:10.1017/CBO9780511791246.
- [173] Edwards, J. W., and Wieseman, C. D., “Flutter and divergence analysis using the generalized aeroelastic analysis method,” *Journal of Aircraft*, Vol. 45, No. 3, 2008, pp. 906–915. doi:10.2514/1.30078.
- [174] Stanford, B., Wieseman, C. D., and Jutte, C., “Aeroelastic Tailoring of Transport Wings Including Transonic Flutter Constraints,” *AIAA SciTech*, American Institute of Aeronautics and Astronautics, 2015, pp. –. doi:10.2514/6.2015-1127, URL <http://dx.doi.org/10.2514/6.2015-1127>.
- [175] Bathe, K.-J., *Finite element procedures*, Klaus-Jurgen Bathe, 2006.

- [176] Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., and Sorensen, D., *LAPACK Users' Guide*, 3<sup>rd</sup> ed., Society for Industrial and Applied Mathematics, Philadelphia, PA, 1999.
- [177] Back, P., and Ringertz, U., “Convergence of Methods for Nonlinear Eigenvalue Problems,” *AIAA Journal*, Vol. 35, No. 6, 1997, pp. 1084–1087. doi:10.2514/2.200, URL <https://doi.org/10.2514/2.200>.
- [178] Pitt, D., “A new non-iterative P-K match point flutter solution,” *40th Structures, Structural Dynamics, and Materials Conference and Exhibit*, American Institute of Aeronautics and Astronautics, 1999. doi:10.2514/6.1999-1353, URL <https://doi.org/10.2514/6.1999-1353>.
- [179] van Zyl, L. H., “Aeroelastic Divergence and Aerodynamic Lag Roots,” *Journal of Aircraft*, Vol. 38, No. 3, 2001, pp. 586–588. doi:10.2514/2.2806, URL <http://dx.doi.org/10.2514/2.2806>.
- [180] Rodden, W. P., and Bellinger, E. D., “Aerodynamic lag functions, divergence, and the British flutter method,” *Journal of Aircraft*, Vol. 19, No. 7, 1982, pp. 596–598. doi:10.2514/3.44772, URL <http://dx.doi.org/10.2514/3.44772>.
- [181] Allemang, R. J., and Brown, D. L., “A correlation coefficient for modal vector analysis,” *Proceedings of the 1st international modal analysis conference*, Vol. 1, SEM Orlando, 1982, pp. 110–116.
- [182] J. Allemang, R., “The Modal Assurance Criterion (MAC): Twenty Years of Use and Abuse,” *Sound and Vibration*, Vol. 37, 2003, pp. 14–21.
- [183] van Zyl, L. H., “Use of eigenvectors in the solution of the flutter equation,” *Journal of Aircraft*, Vol. 30, No. 4, 1993, pp. 553–554. doi:10.2514/3.46380, URL <http://dx.doi.org/10.2514/3.46380>.
- [184] Eldred, M. S., Venkayya, V. B., and Anderson, W. J., “Mode tracking issues in structural optimization,” *AIAA Journal*, Vol. 33, No. 10, 1995, pp. 1926–1933. doi:10.2514/3.12747, URL <https://doi.org/10.2514/3.12747>.
- [185] Eldred, M. S., Venkayya, V. B., and Anderson, W. J., “New mode tracking methods in aeroelastic analysis,” *AIAA Journal*, Vol. 33, No. 7, 1995, pp. 1292–1299. doi:10.2514/3.12552, URL <https://doi.org/10.2514/3.12552>.
- [186] Eller, D., “Flutter Equation as a Piecewise Quadratic Eigenvalue Problem,” *Journal of Aircraft*, Vol. 46, No. 3, 2009, pp. 1068–1070. doi:10.2514/1.40653, URL <https://doi.org/10.2514/1.40653>.
- [187] Huang, R., Qian, W. M., Zhao, Y. H., and Hu, H. Y., “Flutter Analysis: Using Piecewise Quadratic Interpolation with Mode Tracking and Wind-Tunnel Tests,” *Journal of Aircraft*, Vol. 47, No. 4, 2010, pp. 1447–1451. doi:10.2514/1.47687, URL <https://doi.org/10.2514/1.47687>.

- [188] Hang, X., Fei, Q., and Su, W., “A Mode Tracking Method in Aeroelastic Stability Analysis Using Left Eigenvectors,” *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, 2018. doi:10.2514/6.2018-1204, URL <https://doi.org/10.2514/6.2018-1204>.
- [189] Bisplinghoff, R. L., Ashley, H., and Halfman, R. L., *Aeroelasticity*, Courier Corporation, 2013.
- [190] Langthjem, M. A., and Sugiyama, Y., “Optimum Shape Design Against Flutter Of A Cantilevered Column With An End-mass Of Finite Size Subjected To A Non-conservative Load,” *Journal of Sound and Vibration*, Vol. 226, No. 1, 1999, pp. 1–23. doi:<https://doi.org/10.1006/jsvi.1999.2211>, URL <http://www.sciencedirect.com/science/article/pii/S0022460X99922113>.
- [191] Odaka, Y., and Furuya, H., “Robust structural optimization of plate wing corresponding to bifurcation in higher mode flutter,” *Structural and Multidisciplinary Optimization*, Vol. 30, No. 6, 2005, pp. 437–446. doi:10.1007/s00158-005-0538-9, URL <https://doi.org/10.1007/s00158-005-0538-9>.
- [192] Hajela, P., “A root locus-based flutter synthesis procedure,” *Journal of Aircraft*, Vol. 20, No. 12, 1983, pp. 1021–1027. doi:10.2514/3.48206, URL <http://arc.aiaa.org/doi/10.2514/3.48206>.
- [193] Stanford, B. K., “Role of Unsteady Aerodynamics During Aeroelastic Optimization,” *AIAA Journal*, Vol. 53, No. 12, 2015, pp. 3826–3831. doi:10.2514/1.J054314, URL <https://doi.org/10.2514/1.J054314>.
- [194] Neill, D. J., Johnson, E. H., and Canfield, R., “ASTROS - A multidisciplinary automated structural design tool,” *Journal of Aircraft*, Vol. 27, No. 12, 1990, pp. 1021–1027. doi:10.2514/3.45976, URL <http://dx.doi.org/10.2514/3.45976>.
- [195] Kang, B. S., Park, G. J., and Arora, J. S., “A Review of Optimization of Structures Subjected to Transient Loads,” *Structural and Multidisciplinary Optimization*, Vol. 31, No. 2, 2006, pp. 81–95. doi:10.1007/s00158-005-0575-4.
- [196] Wrenn, G. A., “An Indirect Method for Numerical Optimization Using the Kreisselmeier–Steinhauser Function,” Tech. Rep. CR-4220, NASA Langley Research Center, Hampton, VA, 1989.
- [197] Poon, N. M. K., and Martins, J. R. R. A., “An Adaptive Approach to Constraint Aggregation Using Adjoint Sensitivity Analysis,” *Structural and Multidisciplinary Optimization*, Vol. 34, No. 1, 2007, pp. 61–73. doi:10.1007/s00158-006-0061-7.
- [198] Kennedy, G. J., and Hicken, J. E., “Improved constraint-aggregation methods,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 289, 2015, pp. 332–354. doi:10.1016/j.cma.2015.02.017.

- [199] Griewank, A., and Walther, A., *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, 2<sup>nd</sup> ed., Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2008.
- [200] Mader, C. A., and Martins, J. R. R. A., “Derivatives for Time-Spectral Computational Fluid Dynamics Using an Automatic Differentiation Adjoint,” *AIAA Journal*, Vol. 50, No. 12, 2012, pp. 2809–2819. doi:10.2514/1.J051658.
- [201] Hascoët, L., and Pascual, V., “TAPENADE 2.1 User’s Guide,” Technical report 300, INRIA, 2004. URL <http://www.inria.fr/rrrt/rt-0300.html>.
- [202] Pascual, V., and Hascoët, L., “Extension of TAPENADE Towards Fortran 95,” *Automatic Differentiation: Applications, Theory, and Tools*, edited by H. M. Bücker, G. Corliss, P. Hovland, U. Naumann, and B. Norris, Lecture Notes in Computational Science and Engineering, Springer, Berlin, Germany, 2005.
- [203] Giles, M. B., “Collected matrix derivative results for forward and reverse mode algorithmic differentiation,” *Advances in Automatic Differentiation*, Springer, 2008, pp. 35–44.
- [204] Hascoët, L., and Pascual, V., “The Tapenade automatic differentiation tool: Principles, model, and specification,” *ACM Transactions on Mathematical Software*, Vol. 39, No. 3, 2013, pp. 20:1–20:43. doi:10.1145/2450153.2450158.
- [205] Kennedy, G. J., Kenway, G. K. W., and Martins, J. R. R. A., “High Aspect Ratio Wing Design: Optimal Aerostructural Tradeoffs for the Next Generation of Materials,” *Proceedings of the AIAA Science and Technology Forum and Exposition (SciTech)*, National Harbor, MD, 2014. doi:10.2514/6.2014-0596.
- [206] Stroud, W. J., and Agranoff, N., “Minimum-mass design of filamentary composite panels under combined loads: Design procedure based on simplified buckling equations,” Tech. Rep. TN-D-8257, NASA Langley Research Center, Hampton, VA, 1976.
- [207] Kenway, G. K., and Martins, J. R. R. A., “High-Fidelity Aerostructural Optimization of the Airbus XRF1 Aircraft Configuration,” Tech. rep., Multidisciplinary Design Optimization Laboratory, June 2016.
- [208] Kenway, G. K. W., and Martins, J. R. R. A., “Multipoint Aerodynamic Shape Optimization Investigations of the Common Research Model Wing,” *AIAA Journal*, Vol. 54, No. 1, 2016, pp. 113–128. doi:10.2514/1.J054154.
- [209] Lyu, Z., Kenway, G. K. W., and Martins, J. R. R. A., “RANS-based Aerodynamic Shape Optimization Investigations of the Common Research Model Wing,” *Proceedings of the AIAA Science and Technology Forum and Exposition (SciTech)*, National Harbor, MD, 2014. doi:10.2514/6.2014-0567, aIAA 2014-0567.

- [210] Jonsson, E., Mader, C. A., Kennedy, G., and Martins, J., “Flutter Constraint Formulation for High-Fidelity Aerostructural Optimization,” *AIAA Journal*, Vol. –, 2020. Under revisions.
- [211] Landahl, M. T., and Landahl, M., *Unsteady transonic flow*, CUP Archive, 1989.
- [212] Landahl, M. T., “Kernel Function for Nonplanar Oscillating Surfaces in a Subsonic Flow,” *AIAA Journal*, Vol. 5, No. 5, 1967, pp. 1045–1046. doi: 10.2514/3.55319, URL <https://doi.org/10.2514/3.55319>.