Experimental and Numerical Aeroelastic Study of Wings

Ivo Miguel Delgado Rocha

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor: Prof. André Calado Marta

Examination Committee

Chairperson: Prof. Filipe Szolnoky Ramos Pinto Cunha
Supervisor: Prof. André Calado Marta
Member of the Committee: Dr. José Manuel Vieira Antunes

July 2019

Dedicated to my family

Acknowledgments

I would like to thank my supervisor , Professor André Calado Marta, for his dedication and guidance through this work,as his counsel and knowledge was fundamental to finish this task. I also would like to thank Professors André Rui Dantas Carvalho and Éder Luiz Oliveira for providing the equipment and guidance necessary to perform the experimental aeroelastic tests. Finally, I would also like to thank Alexandre Cruz for providing documentation and assistance during the wing model construction.

Resumo

Desde os primórdios da aviação que os problemas aeroelásticos apresentam os maiores desafios no que toca à sua resolução. Com o advento dos métodos numéricos, o estudo de estruturas aeronáuticas e a sua interação com o ar nas diferentes condições de voo tornou-se acessível, levando a que agora seja obrigatório na fase de projeto de qualquer aeronave.

Este trabalho tem como foco o desenvolvimento de uma ferramenta numérica que permita a análise da interação entre a estrutura de uma asa e o fluido em seu redor e o teste em túnel de vento de modelos de meia asa para ajudar na validação da ferramenta numérica. A análise aerodinâmica é tem como base um método de painéis, enquanto que para a análise estrutural foi implementado um modelo de elementos finitos que usa elementos viga. Ambos módulos foram programados usando MATLAB®. A forma da asa foi parametrizada usando a sua área, perfil aerodinâmico, razão de aspeto, afilamento, ângulo de flecha e ângulo de diedro. Cada módulo computacional foi verificado com sucesso recorrendo a outras fontes bibliográficas e foram unidas recorrendo a um módulo de interface fluido-estrutura. Um estudo paramétrico foi feito para ilustrar a influência da razão de aspeto sobre a velocidade de flutter.

A ferramenta aeroelástica desenvolvida foi usada em conjunto com uma ferramenta de otimização numérica com a finalidade de obter três asas ótimas cuja funcionalidade era a maximização da razão de sustentação-arrasto, a minimização da massa da asa e a maximização da velocidade de flutter, respetivamente. Cada processo de otimização garante também que a performance da nova asa não é inferior à configuração inicial.

Palavras-chave: Projeto de Aeronaves, flutter, velocidade de divergência, interacção fluido-estrutura, túnel de vento, optimização

Abstract

Since the early days of aviation, aeroelastic problems have shown to be some of the most challenging to solve. With the development of numerical methods, the study of aircraft structures and their interaction with the surrounding air flow at different flight conditions has become easily accessible and, thus, is now mandatory in the design phase of an aircraft.

This work focuses on the development of a numerical tool for aircraft wing fluid-structure interaction (FSI) analyses, in which the external airflow and the internal structure interact, as well as the wind tunnel testing of two half wing prototypes to help validate the accuracy of the numerical tool developed. A panel method was implemented for the aerodynamic analysis and a finite-element model using equivalent beam elements was implemented for the structural analysis, both coded in MATLAB® language. The wing shape was parametrized using area, airfoil cross-section shape, aspect ratio, taper ratio, sweep angle and dihedral angle. Each analysis models were successfully individually verified against other bibliographic sources and then the two disciplines were coupled into the FSI numerical tool. A parametric study was also conducted to study the influence of the wing aspect ratio on flutter speed.

The validated FSI tool was then used in an optimization framework to obtain three separate optimized wing shapes with the objectives of maximizing the lift-to-drag ratio, minimizing wing mass and maximizing wing flutter velocity respectively, whilst guaranteeing that the new wing performance is not worse than that of the baseline wing.

Keywords: Aircraft design, flutter, divergence speed, fluid-structure interaction, wind tunnel, optimization

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α	Angle of attack.
Γ	Dihedral angle.
η	Generalized coordinates.
θ	Pitch angle.
θ_{root}	Wing root twist.
θ_{tip}	Wing tip twist.
Λ	Sweep angle.
λ	Taper ratio.
μ	Doublet strength.
ν	Kinematic viscosity.
ρ	Density.
σ	source strength.
σ_r	Ratio of uncoupled plunge and pitch frequencies.
τ	Shear stress.
Φ	Velocity potential.
ω	Eigenfrequencies.

Roman symbols

A	Wing cross section area.
AR	Aspect Ratio.
b	Wing half-chord.
c	Wing chord.

C_D	Drag coefficient.
C_L	Lift coefficient.
C_M	Moment coefficient.
C_p	Pressure coefficient.
E	Young Modulus.
E_k	Kinetic energy.
E_p	Potential energy.
F	External forces vector.
f	Frequency.
G	Shear Modulus.
g	Damping ratio.
h	Plunge displacement.
I	Moment of Inertia.
K	Stiffness matrix.
k	Reduced frequency.
M	Mass matrix.
m	Wing mass.
Ma	Mach number.
m_r	Mass ratio.
Q	Generalized forces.
q_∞	Dynamic pressure.
r	Dimensionless radius of gyration.
Re	Reynolds number.
s	Wing span.
T	Temperature.
t	Time step.
U	Airspeed.
U_D	Divergence speed.

U_F	Flutter speed.
V	Reduced velocity.
v	Velocity vector.
V_F	Flutter Speed Index.

Subscripts

∞	Free-stream condition.
l, m, n	Panel components.
n	Normal component.
ref	Reference condition.

Glossary

AAW	Active Aeroelastic Wing
AEL	Aerospace Engineering Laboratory
AGARD	Advisory Group for Aerospace Research and Development
B&K	Brüel & Kjaer
CAE	Computational AeroElasticity
CFD	Computational Fluid Dynamics
CSD	Computational Structural Dynamics
DLM	Distributed Lagrange Multiplier
DNS	Direct Navier-Stokes
DPM	Dispersed Phase Method
FEM	Finite Element Method
FE	Finite Element
FFT	Fast Fourier Transform
FSI	Fluid-Structure Interaction.
HALE	High-Altitude Long Endurance
LCO	Limit Cycle Oscillations
MDO	Multidisciplinary Design Optimization
NACA	National Advisory Committee for Aeronautics
RANS	Reynolds Averaged Navier-Stokes
SQP	Sequential Quadratic Programming
TSD	Transonic Small Disturbances
UAV	Unmanned Aircraft Vehicle
VLM	Vortex Lattice Method

Chapter 1

Introduction

The usage of numeric tools to aid aircraft design can be seen as one of the greatest advances in the aircraft industry, as it enables the preliminary testing of different aircraft configurations without the costs associated with wind tunnel testing. Despite these advancements, experimental tests are still required to achieve the final aircraft configuration, as numeric results cannot be considered valid without verification and if no benchmark cases exist for a specific set of conditions, work must be done to define new benchmarks for the new conditions, so as to enable the usage of numeric tools for those specific conditions.

1.1 Aircraft Design

Aircraft design as always been a balancing act between four fundamental forces, as shown in Fig. 1.1. While at a first glance, they may seem independent, these fundamental forces are actually interdependent, meaning a change to any one of them will have consequences to the other three. Since the main objective of aircraft is to fly, it must generate enough lift to balance its weight. However, by increasing the lift force, the drag produced also increases, which leads to a higher required thrust. By this logic, instead of attempting to increase the aircraft lift, one should attempt to minimize its weight, either by reducing the payload and other items that turn the aircraft profitable, or by reducing its structural weight.

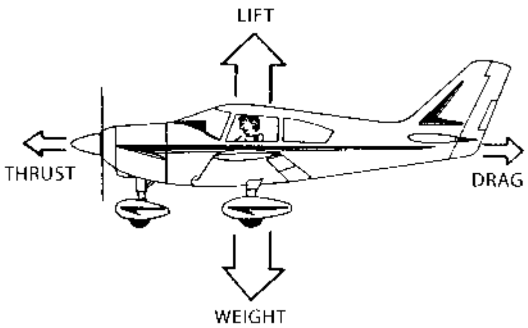


Figure 1.1: Forces acting on an aircraft

The first production aircraft, due to the reduced power that primitive engines produced [1], could not carry large numbers of passengers or cargo, due to low operating speeds limiting the maximum weight of the aircraft, as lift is also dependent on airflow speed. With the increase in power of the engines however, higher speeds started to be achievable but, at the same time, it was verified that structural problems started to appear, particularly an increase in vibrations and other dynamic phenomena.

At first these phenomena were not fully understood, and led to a series of mishaps and accidents that gave aviation a dangerous reputation [2]. By studying these incidents, the first aeroelastic testing of aircraft begun and, since then, this topic has become one of the most studied subject in the aviation industry. The implication of aeroelasticity for the increase of safety of flight but its difficult accurate modelling mean that care must be devoted in determining the aeroelastic behaviour of every aircraft design.

While the basic mechanics behind flight have been understood for centuries, the interaction between seemingly unrelated topics like structural dynamics and unsteady aerodynamics constitutes the basis for aeroelastic analysis.

1.2 Analysis and Design Tools

To attempt to model the interaction between the topics of structural dynamics and unsteady aerodynamics, the first methods based on modal analysis and 2-D aerodynamic theory appeared, such as the k-method, that are still used for preliminary design phases as they allow for reasonably accurate results [3]. With the advances in computing power and new Computational Fluid Dynamics (CFD) algorithms, it was possible to produce an accurate aeroelastic tool that could simulate things like transonic flow, stall influence on wing dynamic behaviour and other non-linear phenomena that effect real aircraft performance [4].

To achieve a correct simulation of aeroelastic behaviour, however, benchmark experimental cases would have to be performed, as any computational tool cannot be stated to give accurate results without proper validation. Initially, the benchmark for transonic regime aeroelastic behaviour was the AGARD 445.6 wing [5], as seen in Fig. 1.2, which, to this day constitutes the basis for any validation attempt on transonic aeroelastic tools.

Most of the validation works developed using this baseline wing are centered around the determination of the Flutter Speed Index, an non dimensional parameter that relates a wings' dynamic behaviour with the variation of both the freestream velocity and the first torsional mode frequency, as shown in Fig. 1.3. This non dimensional parameter constitutes the boundary of the stable and unstable zones for flutter, as for the same Mach number, a value of the flutter speed index greater than the determined indicates that the flutter phenomenon is likely to occur [8] and, as such, it is a great tool to predict aeroelastic behaviour for a wide array of Mach values.

For the specific case of the AGARD 445.6, this parameter displays a dip in value for the transonic region that some CFD models cannot model correctly [9], and, due to this, it continues to be a benchmark for advancements in current aeroelastic tools. Despite being able to define the boundaries of the occur-

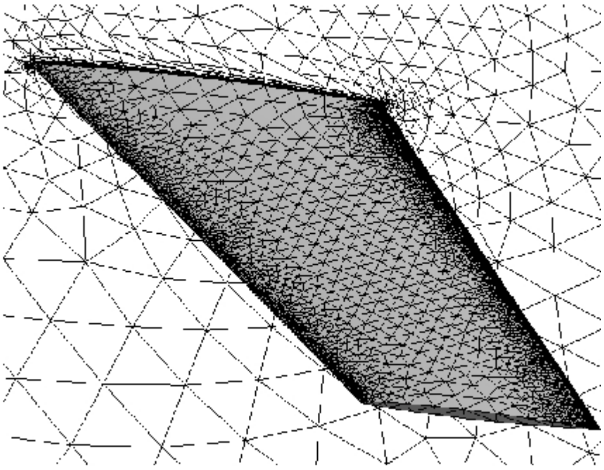


Figure 1.2: Discretization of AGARD 445.6 wing[6]

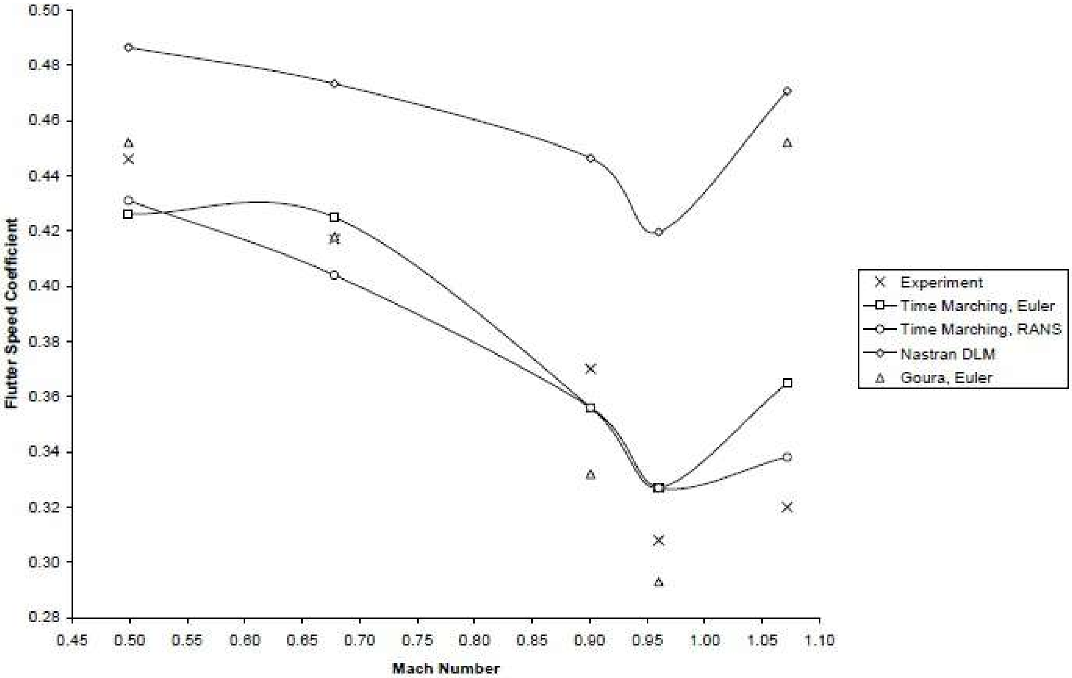


Figure 1.3: Flutter speed index variation for the AGARD 445.6 wing ($\alpha = 0°$)[7]

rence of divergent behaviour, the flutter speed index cannot define the type of dynamic phenomena that occurs in the post flutter region, as it can range from Limit Cycle Oscillations (LCO) of small amplitude to buffeting [8].

The determination of the Flutter Speed Index is common in most numeric and experimental aeroelastic applications, but it is also useful from a design standpoint to determine both the wing structural frequency(f) spectra and damping ratio (g) variation with speed, as by computing the variation of both the damping and frequencies of each mode with the speed, the approximate flutter speed can be obtained for the current wing configuration, as shown in Fig. 1.4. This is the method that legacy aeroelastic tools use, based on numeric methods like the p and p-k methods [3].

While the advances in aeroelastic tools have been remarkable, the most recent developments in wing

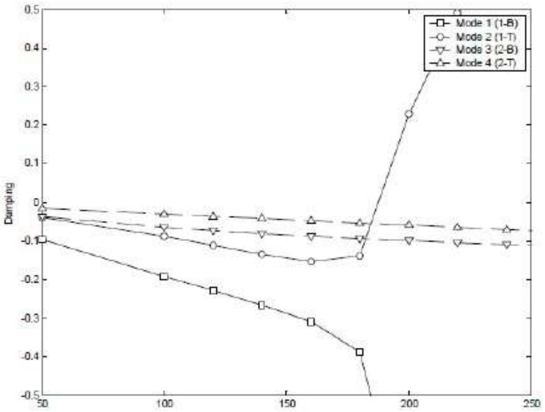


Figure 1.4: U-g plot for the AGARD 445.6 wing ($Ma = 0.499$) [7]

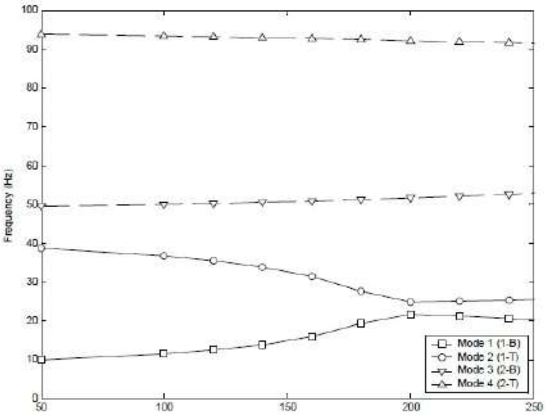


Figure 1.5: U-f plot for the AGARD 445.6 wing ($Ma = 0.499$) [7]

design have furthered the need for reliable prediction of aeroelastic phenomena, as the introduction of new generation wings that are more flexible (such as the Active Aeroelastic Wing(AAW) in Fig. 1.6 [10]); have higher aspect ratios (AR) and even have morphing shapes during flight [11, 12]. As such, the flight envelope defining characteristics like flutter can now occur for speeds much lower than those on legacy aircraft, which leads to a new design challenge: maintain flight envelopes similar to legacy aircraft whilst making sure that the new aircraft generation is not plagued by aeroelastic instabilities.

Figure 1.6: Boeing F/A-18 active aeroelastic wing research aircraft[10]

1.3 Motivation

The increase in flexibility and aspect ratio is not very pronounced in civil aviation due to the inherent risk averse nature of the industry, but on Unmanned Air Vehicles (UAV) these changes have been quickly adopted.For example, the High Altitude Long Endurance (HALE) UAV Airbus Zephyr shown in Fig. 1.7, explores the correlation between larger aspect ratio and reduced induced drag, that leads to an increase of the overall lift-to-drag ratio (L/D) and, therefore, to an increase of range.

Fixed wing UAV aircraft are then built with extremely slender wings, which potentiate the occurrence of aeroelastic phenomena at much lower speeds, well within the incompressible flow regimen. The

Figure 1.7: Airbus Zephyr HALE aircraft [13]

aerodynamic behaviour of an aircraft subject to these conditions can be accurately modelled by lower complexity models, as an approximation with linearised flow is reasonably accurate for the incompressible flow range.

Although the aerodynamic behaviour is easier to model, there is a lack of readily available aeroelastic experimental data for these speed ranges, as most studies are performed at the transonic level [14, 9, 7]. There are some attempts to improve data for experimental confirmation, particularly for the case of geometric non-linearities [15], but for the most part, there is a need for a broad range of aeroelastic testing data cases [16], specially with the recent numeric developments concerning the simulation of geometric non-linear behaviour and Limit Cycle Oscillations [17, 15, 18].

Besides the introduction of more complex geometric definitions, there is interest in analysing several possible interface methods between the aerodynamic and structural solvers and its tradeoffs [19, 9], to improve accuracy of current aeroelastic tools. Another advantage of the increase in accuracy of aeroelastic tools is the possibility of incorporating them in optimization frameworks to allow design refining around the expected aeroelastic behaviour of an aircraft. By doing this, the typical cycle of manual design iteration that characterised most legacy projects is dropped in favour of an automated alternative, which gives cost and time savings, at the expense of introducing another layer to the existing analysis framework.

1.4 Objectives and Deliverables

The first objective of this work is to perform a series of experimental aeroelastic tests, performed at the Aerospace Engineering Laboratory (AEL) Wind Tunnel at Instituto Superior Técnico, a low speed wind tunnel, to produce low-speed aeroelastic data and define methodologies for similar future tests at the same facility.

At the same time, a modular numeric aeroelastic tool is to be implemented, developed using MATLAB® software [20], based on work by Almeida [21] and Cardeira [22], so that all validation work performed previously is still valid on the new numeric tool, but also improving on its architecture and adding some useful functionalities. This framework is to be validated with results from the experimental tests performed and other available bibliographic data.

Another objective is to show the reliability of the aeroelastic tool by performing a series of parametric studies based on typical aircraft design variables.

The final expected deliverables include:

- A reliable aeroelastic analysis tool with a modular architecture that allows for future additions to its architecture and that can be both used for calculations with aircraft in the low speed regime and also to perform aeroelastic design optimization;
- A test bench for future aeroelastic testing at the AEL Wind Tunnel, including documentation for all equipments and procedures to be used.

1.5 Thesis Outline

Chapter 2 presents the subject of aeroelasticity, focusing both on the static approach, with the definition of divergence velocity and the dynamic approach, which is the basis for flutter speed determination. From the dynamic aeroelastic definitions, an overview of legacy flutter speed determination methods is given. Lastly, the topic of computational aeroelasticity (CAE) is introduced, together with different framework methodologies and discipline models.

Chapter 3 introduces the experimental methodology, describing the material selection for the wing model construction, the configuration of the wing mount, the selected instrumentation and the experimental procedure. It concludes with the experimental data processing and analysis.

Chapter 4 focuses on the numerical implementation of the aeroelastic framework, starting with the both aerodynamic and structural models selection and implementation, validation of section properties estimation by the structural module, parametrization of a wing structure, implementation of the fluid-structure interface module and ending with remarks on program architecture and computational implementation.

Chapter 5 displays the numeric results obtained from the aeroelastic framework, beginning with convergence and validation studies of selected wing configurations, leading to the flutter speed calculation, together with a comparison with experimental data obtained previously. It ends with the parametric study of the variation of the flutter speed with aspect ratio.

Chapter 6 presents a brief introduction to numerical optimization, detailing the selection of a constrained gradient-based optimizing tool to perform three optimization runs, consisting of a wing lift to drag ratio maximization problem, a wing mass minimization problem and a wing flutter speed maximization problem, ending with a summary of computational cost of the three optimization tests.

Chapter 7 presents the thesis concluding remarks, together with some future ideas based on the work developed.

Chapter 2

Aeroelasticity Principles

Although often studied as separate subjects, the aerodynamic and structural response of a lifting body are, in reality, connected . Since aircraft wings can be quite flexible, they are prone to elastic deformation under load and, since they are the primary lifting surface, they are always subjected to loads that, at a minimum, are equal to the weight of the aircraft. These loads are enough to induce elastic deformation to the wing, which then influences the wing's angle of incidence and overall shape. A typical example of the high flexibility of aircraft wings is the Boeing 787 wing shape at takeoff, as shown in Fig. 2.1.

Figure 2.1: Boeing 787 Dreamliner at takeoff

Since the wing shape also influences the overall lift, the result is a coupled aerodynamic-structural system, or aeroelastic system. The Collar diagram, seen in Fig. 2.2, illustrates the interaction of forces that dictate the aeroelastic behaviour of an aircraft.

The interaction between aerodynamic forces and inertial forces results into the aerodynamic stability characteristics of the aircraft, while the interaction between inertial forces and elastic forces gives the structural vibration behaviour. Finally, the coupling of aerodynamic forces and elastic forces defines static aeroelasticity. By coupling all of these interactions, we obtain the dynamic aeroelastic behaviour of an aircraft, responsible for the definition of dynamic behaviour such as flutter and low cycle oscillations(LCO).

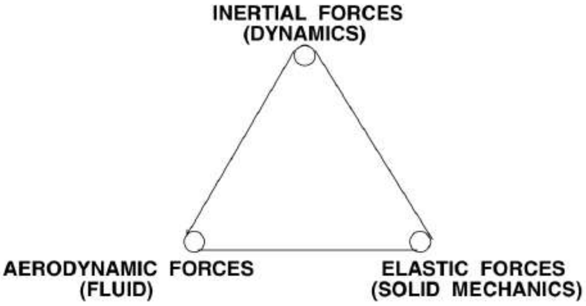


Figure 2.2: Collar diagram (adapted from [23])

2.1 Static Aeroelasticity

When considering problems of static aeroelasticity one is referring to the interaction of aerodynamic loading caused by steady flow over a wing and causing elastic deformation of the structure. These deformations have effects on flight stability, handling qualities and structural-load distribution. Steady state systems of aeroelastic forces produce phenomena such as divergence and control reversal. The most common divergence problem in static aeroelasticity is the torsional divergence of a wing, as stated by Megson [24].

To determine the wing torsional divergence, it is considered the case of a wing of area S without ailerons in a two-dimensional flow, where the torsional stiffness of the wing is represented by a spring of stiffness, as shown in Fig. 2.3.

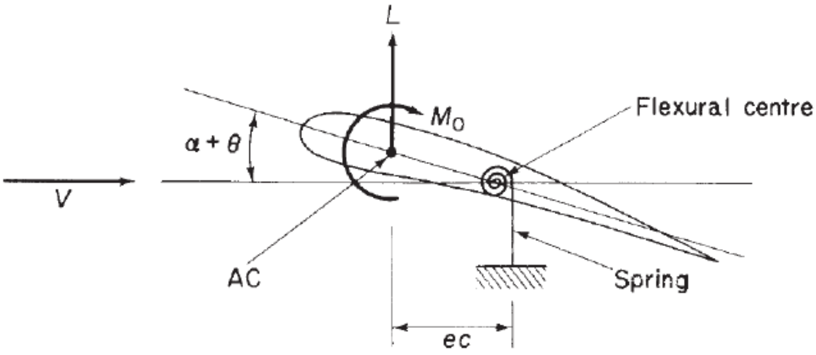


Figure 2.3: Determination of wing divergence speed (two-dimensional case)[24]

Performing a moment equilibrium of a wing section about the aerodynamic centre results in,

$$M_0 + Lec = K\theta \ , \tag{2.1}$$

With K as the torsional stiffness of the wing, L the lift vector,d M_0 wing pitching moment about the aerodynamic centre (AC), ec is the distance of the aerodynamic centre forward of the flexural centre expressed in terms of their wing chord c and θ is the elastic twist of the wing. From aerodynamic theory [25], M_0, L, and C_L are defined as

$$M_0 = \frac{1}{2}\rho S c U^2 C_{M,0} \ , \tag{2.2a}$$

$$L = \frac{1}{2}\rho S U^2 C_L \ , \tag{2.2b}$$

$$C_L = C_{L,0} + \frac{\partial C_L}{\partial \alpha}(\alpha + \theta) \ , \tag{2.2c}$$

where α is the wing angle of attack, ρ is the air density, S is the wing area, V is the airspeed and C_{L_0} is the wing's lift coefficient for $\alpha = 0$. Substituting Eqs.(2.2) in Eq.(2.1), and solving with respect to the angle of twist yields

$$\theta = \frac{\frac{1}{2}\rho S c U^2 (C_{M,0} + eC_{L,0} + e\frac{\partial C_L}{\partial \alpha}\alpha)}{K - \frac{1}{2}\rho S e c U^2 \frac{\partial C_L}{\partial \alpha}} \ . \tag{2.3}$$

From Eq. (2.3), it is concluded that divergence (when θ becomes infinite) occurs for

$$U_d = \sqrt{\frac{2K}{\rho S e c \frac{\partial C_L}{\partial \alpha}}} \ , \tag{2.4}$$

that is called the divergence speed U_d. This speed is the limit from which an increase in lift produces a positive feedback effect that further increases the wing angle of attack α, which also increases the lift further, making the system unstable.

2.2 Dynamic Aeroelasticity

While static aeroelasticity deals with the interactions of elastic and steady aerodynamic forces, dynamic aeroelasticity encompasses the interactions between aerodynamic, elastic and inertial forces. These interactions in particular differ from static aeroelasticity as the equilibrium equations now include the representation of the unsteady aerodynamic behaviour in terms of the elastic deformation of the wing, so that dynamic phenomena like flutter can be estimated.

2.2.1 Equations of Motion of a Linear Aeroelastic System

To perform a flutter analysis of a linear aeroelastic system, it is required to formulate the equations of motion for the system. To achieve this purpose, Lagrange's equations are used for the deduction, specialized here for the case that the kinetic energy E_k depends only on generalized coordinates $\dot{\eta}_1\dot{\eta}_2$,... yielding

$$\frac{d}{dt}(\frac{\partial E_k}{\partial \dot{\eta}_i}) + \frac{\partial E_p}{\partial \eta_i} = Q_i \ , \tag{2.5}$$

where E_p represents the systems' potential energy, η_i is the generalized coordinate and Q_i are the forces applied on the system.

This means that both the potential and kinetic energy are needed , as well as the generalized forces resulting from aerodynamic loading. Figure 2.4 shows the aeroelastic model used, as well as the distances required for the calculations:

The Potential energy considering the linear and torsional springs is given by

$$E_p = \frac{1}{2}k_h h^2 + \frac{1}{2}k_\theta \theta^2 \ . \tag{2.6}$$

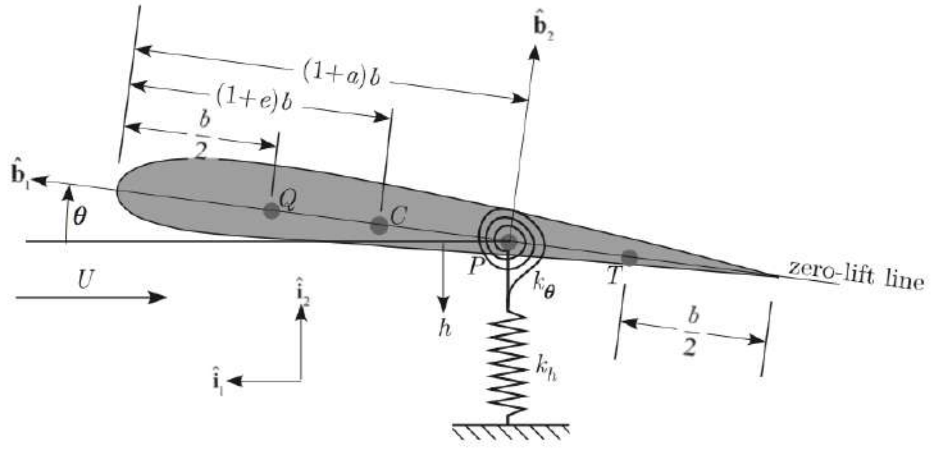


Figure 2.4: Geometry of the wing section with pitch and plunge spring restraints[26]

where k_h is the wing structural bending stiffness, h is the plunge displacement, k_θ is the the wing structural torsional stiffness and θ is the wing model pitch displacement. The kinetic energy is given by

$$\begin{aligned} E_k &= \frac{1}{2}m(\dot{h}^2 + b^2x_\theta^2\dot{\theta}^2 + 2bx_\theta\dot{h}\dot{\theta}) + \frac{1}{2}I_C\dot{\theta}^2 \\ &= \frac{1}{2}m(\dot{h}^2 + 2bx_\theta\dot{h}\dot{\theta}) + \frac{1}{2}I_P\dot{\theta}^2 \end{aligned} \tag{2.7}$$

where m is the wing model mass, b is the wing semi chord, x_θ is the chordwise offset of the centre of mass from the reference point, I_C is the moment of inertia about point C and I_P is the moment of inertia about point P, defined as $I_P = I_C + mb^2x_\theta^2$.

For the aerodynamic loading, the generalized forces become

$$Q_h = -L \tag{2.8a}$$

and

$$Q_\theta = M_{\frac{c}{4}} + b(\frac{1}{2} + a)L. \tag{2.8b}$$

where $M_{\frac{c}{4}}$ is the wing moment about the quarter-chord point from the leading edge and $b(\frac{1}{2} + a)$ is the distance between the quarter-chord point from the leading edge and point P.

Since the analysis is for a 2-D problem, we have that $n = 2$, $q_1 = h$ and $q_2 = \theta$ and the equations of motion become

$$m(\ddot{h} + bx_\theta\ddot{\theta}) + k_hh = -L \tag{2.9a}$$

and

$$I_P\ddot{\theta} + mbx_\theta\ddot{h} + k_\theta = M_{\frac{c}{4}} + b(\frac{1}{2} + a)L. \tag{2.9b}$$

Assuming that the airfoil is symmetric, from thin-airfoil aerodynamic theory [26], $C_L = C_{L_\alpha}\alpha$ and $C_{L_\alpha} = 2\pi$, and the aerodynamic centre is located at the quarter-chord point from the leading edge, leading to

$$L = 2\pi\rho_\infty bU^2\theta \tag{2.10a}$$

and

$$M_{\frac{c}{4}} = 0 \tag{2.10b}$$

Th Introducing the uncoupled, natural frequencies at zero airspeed, defined by

$$\omega_h = \sqrt{\frac{k_h}{m}} \tag{2.11a}$$

and

$$\omega_\theta = \sqrt{\frac{k_\theta}{I_P}} \tag{2.11b}$$

and substituting Eqs. (2.10) into Eqs. (2.9) and rearranging the equations into matrix form yields

$$\begin{bmatrix} mb^2 & mb^2 x_\theta \\ mb^2 x_\theta & I_P \end{bmatrix} \begin{bmatrix} \frac{\ddot{h}}{b} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} mb^2\omega_h^2 & 2\pi\rho_\infty b^2 U^2 \\ 0 & I_P\omega_\theta^2 - 2(\frac{1}{2}+a)\pi \end{bmatrix} \begin{bmatrix} \frac{h}{b} \\ \theta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{2.12}$$

Eq. 2.12 allows reduction of the equations of motion to an eigenvalue problem, as by defining the plunge and pitch variables as exponential functions of time a general solution can be found, with complex roots, which enable the easy definition of the types of motion and stability characteristics.

2.2.2 Flutter

Typical flutter analysis uses one of 3 distinct methods: the p-method [24], the k method [27] or V-g method and the p-k method [3]. The most accurate solution is obtained from the p-method, as it comes from the original equations of motion Eq. (2.12), and it will be the method used for flutter calculations. For completeness, all three methods are briefly described next.

p method

From Eq. (2.12), introducing r as the dimensionless radius of gyration of the airfoil cross section, σ_r the ratio of uncoupled plunge and pitch frequencies, m_r the mass-ratio parameter, V the dimensionless free stream speed, p as the unknown dimensionless complex eigenvalue, plunge h as an exponential function of time with amplitude $\bar{h}$and frequency $\nu = \frac{pU}{b}$ and the pitch θ as an exponential function of time with amplitude $\bar{h}$ and frequency $\nu = \frac{pU}{b}$,

$$r^2 = \frac{I_P}{mb^2}, \tag{2.13a}$$

$$\sigma_r = \frac{\omega_h}{\omega_\theta}, \tag{2.13b}$$

$$m_r = \frac{m}{\rho_\infty \pi b^2}, \tag{2.13c}$$

$$V = \frac{U}{b\omega_\theta}, \tag{2.13d}$$

$$h = \bar{h}exp(\nu t), \tag{2.13e}$$

$$\theta = \bar{\theta}exp(\nu t), \tag{2.13f}$$

leads to

$$\begin{bmatrix} p^2 + \frac{\sigma_r^2}{V^2} & x_\theta p^2 + \frac{2}{m_r} \\ x_\theta p^2 & r^2 p^2 + \frac{r^2}{V^2} - \frac{2}{m_r}(a + \frac{1}{2}) \end{bmatrix} \begin{bmatrix} \frac{\bar{h}}{b} \\ \bar{\theta} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{2.14}$$

To find a non-trivial solution for Eq. (2.14), the determinant of the coefficient matrix must be set to zero. There are typically two complex conjugate pairs of roots, usually specified in order to the system parameter ω_θ:

$$Vp_1 = \frac{\Gamma_1}{\omega_\theta} \pm i\frac{\Omega_1}{\omega_\theta} \tag{2.15a}$$

$$Vp_2 = \frac{\Gamma_2}{\omega_\theta} \pm i\frac{\Omega_1}{\omega_\theta} \tag{2.15b}$$

where the negative of Γ_k is the modal damping and Ω_k is the modal frequency. The behaviour of the solution depends on the value of the real and imaginary parts of p. Their influence on the stability of the system is viewed in Tab. 2.1:

Table 2.1: Types of motion and stability characteristics for various values of Γ_k and Ω_k [26]

Γ_k	Ω_k	Type of Motion	Stability characteristic
< 0	$\neq 0$	Convergent Oscillations	Stable
$= 0$	$\neq 0$	Simple Harmonic	Stability Boundary
> 0	$\neq 0$	Divergent Oscillations	Unstable
< 0	$= 0$	Continuous Convergence	Stable
$= 0$	$= 0$	Time Independent	Stability Boundary
> 0	$= 0$	Continuous Divergence	Unstable

For a given configuration and altitude, it is required an analysis of the complex roots as functions of V in order to find the smallest value to give divergent oscillations in accordance to Tab. 2.1. This value is $V_F = U_F/(b\omega_\theta)$, with U_F the flutter speed. The variation of both the modal frequency and the modal damping with the reduced is shown in Figs. 2.5 and 2.6, for a case where $a = -\frac{1}{5}$, $e = -\frac{1}{10}$, $m_r = 20$, $r^2 = \frac{6}{25}$ and $\sigma_r = \frac{2}{5}$.

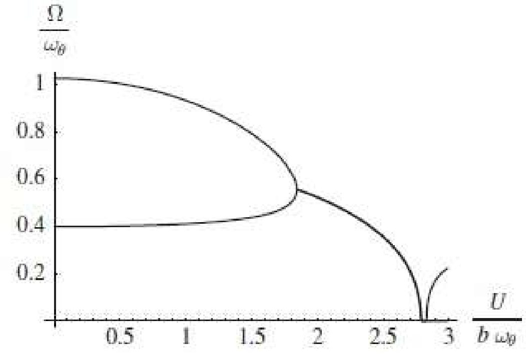


Figure 2.5: Modal frequency versus reduced velocity[26]

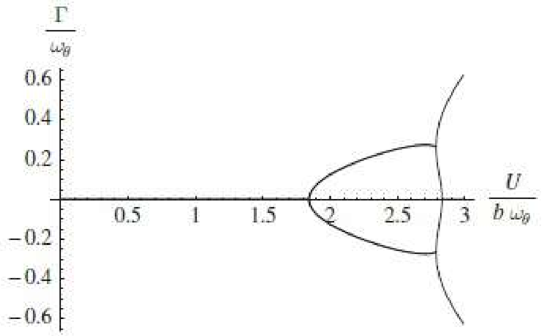


Figure 2.6: Modal damping versus reduced velocity[26]

It is also possible to compute the divergence speed by setting $p = 0$ in Eq. (2.14), making $\bar{\theta}$ zero and solving for V, giving the divergence speed

$$V_D = \frac{U_D}{b\omega_\theta} = r\sqrt{\frac{m_r}{1+2a}}. \tag{2.16}$$

The figures shown also identify the two methods typically used to calculate flutter and divergence points for any aeroelastic method, as well as for experimental testing. The evolution shown for both the damping and the modal frequency is different when departing from a steady-state aerodynamic analysis, as experimental testing shows that damping for all modes below the flutter point is not zero, and there is no coalescence of roots to the same value at the exact flutter point. These factors make the determination of the actual flutter point more complicated, but for a preliminary approach it can be assumed that the flutter point will occur when there is damping ratio greater than zero.

While the p method produces the most accurate solution, it is difficult to implement computationally, and, therefore, most engineering applications of 2-D wing flutter use either the k or the p-k methods.

k method

The k method was the first computational method to allow flutter speed determination. It resulted from observations made that indicated that the energy removed per cycle during a simple harmonic oscillation was nearly proportional to the square of the amplitude but independent of the frequency, which can be characterized by a damping force that is proportional to the displacement but in phase with velocity. The computational strategy for solving a flutter problem using the K- method is shown schematically in Fig. 2.7 and it includes eight steps:

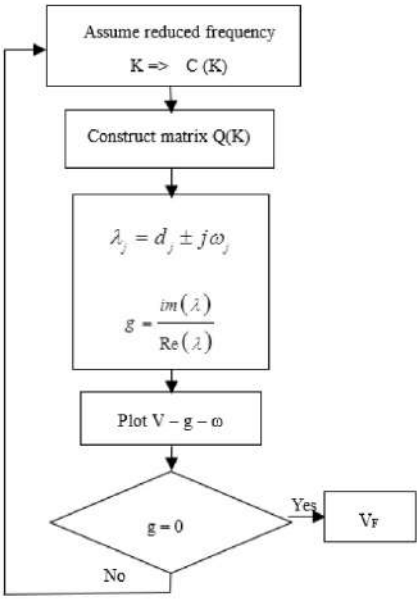


Figure 2.7: k-method algorithm [27]

1. Assume purely harmonic response $p = ik$, where k is reduced frequency,
2. Set up the eigenvalue problem from the equations of motions such that: $[K]^{-1}([M]+\frac{1}{2}\rho U^2[A(k)/k^2])\hat{q} =$

$\frac{1+ig}{\omega^2}\hat{q}$ or $[B(k)]\hat{q} = \lambda\hat{q}$, where $\lambda = \lambda' + i\lambda''$, $\lambda = \frac{1+ig}{\omega^2}$ and $B(k) = [K]^{-1}([M] + \frac{1}{2}\rho U^2[A(k)/k^2]);$

3. Choose an altitude, which results in a value for ρ;
4. Choose k and calculate $[B(k)]$, which will include aerodynamic data at frequency k;
5. Compute the eigenvalues for each mode $\lambda_1, \ldots \lambda_N$;
6. Compute the frequency of each mode $\omega_i = 1/\sqrt{\lambda_i'}$;
7. Compute the flight speed $U_i = \omega_i b/k$ and fictitious damping $g_i = \omega_i^2 \lambda_i''$ corresponding to each mode;
8. Compute the flutter speed.

p-k method

From [3], and using the p-k method for a linear aeroelastic analysis, the equilibrium equation is

$$M\ddot{\eta} + D\dot{\eta} + K\eta + q_\infty Q(k, \textit{Mach})\eta = 0 \tag{2.17}$$

where M is the mass matrix, D is the damping matrix, K is the stiffness matrix, η are the generalized coordinates, q_∞ is the dynamic pressure $q_\infty = \frac{1}{2}\rho V^2$, k is the reduced frequency, $Mach$ is the relation between the airspeed and the speed of sound and Q is the modal generalized aerodynamic forces matrix, usually complex. Its real part, denoted by Q_R is called aerodynamic stiffness and is in phase with the vibration displacement, and the imaginary part of Q, denoted by Q_I, is called aerodynamic damping and is in phase with the vibration velocity.

Equation (2.17) is a second degree non-linear equation, with the non-linearity coming from the fact that the generalized aerodynamic forces matrix Q is a a function of reduced frequency k , which depends on ω, as seen in Eq. (2.11).

If the generalized coordinates vector has dimension n, as the equation of aeroelastic dynamics is a second degree equation, the vector of eigenvalues has dimension $2n$:

$$\lambda = [\lambda_1 \ldots \lambda_i \ldots \lambda_{2n}]^T \tag{2.18}$$

with each eigenvalue written as $\lambda_i = d_i + j\omega_i$, where ω_i is the imaginary part of the eigenvalue representing the frequency, and d_i is the real part representing the damping. Equation (2.17) can be put in matrix form as

$$\begin{bmatrix} \dot{\eta} \\ \eta \end{bmatrix} = \begin{bmatrix} 0 & I \\ -M^{-1}(K + q_d Q) & -M^{-1}D \end{bmatrix} \begin{bmatrix} \eta \\ \dot{\eta} \end{bmatrix} = A \begin{bmatrix} \eta \\ \dot{\eta} \end{bmatrix} \tag{2.19}$$

From here, the p-k algorithm can be applied, as schematically shown in Fig. 2.8 and it comprises five steps:

1. Choose initial speed V

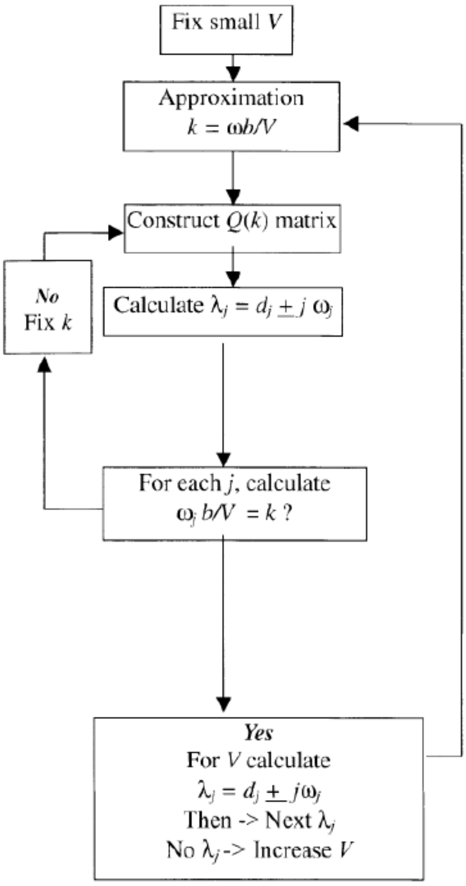


Figure 2.8: p-k method algorithm[3]

2. Approximate k by $k = \omega b/V$;
3. From the initial estimation, construct modal generalized aerodynamic forces matrix Q;
4. Solve Eq. 2.19 to obtain the eigenvalues;
5. For each eigenvalue, check if $\omega_j b/V = k$. If this condition is not fulfilled, a the new value of k is estimated with the obtained eigenvalues and a new computation of Q and eigenvalues is performed;
6. If the condition $\omega_j b/V = k$ is fulfilled, after all eigenvalues have been estimated, the velocity is incremented and used for a new estimation of k. Flutter speed is achieved when $d_i = 0$.

The p-k method is still widely used as it produces accurate results and it is easy to implement. It can be found in many commercial engineering tools such as MSC NASTRAN® software.

2.3 Computational Aeroelasticity

While the p,k and p-k methods can be also considered computational aeroelastic (CAE) methods, Computational Aeroelasticity specifically refers to the coupling of high-fidelity Computational Fluid Dynamics

(CFD) methods with Computational Structural Dynamics (CSD) methods to perform aeroelastic analysis [9]. When the other methods were being developed, computing power simply did not allow for higher-complexity models to be used, however, since then, large breakthroughs have happened and it is possible to choose from a large array of aerodynamic and structural analysis tools. Like previous aeroelastic analysis, the basis for any CAE methodology is the coupled equations of motion

$$[M]\ddot{\eta}(t) + [D]\dot{\eta}(t) + [K]\eta(t) = F(t), \tag{2.20}$$

Where M,D and K are generalized mass, damping and stiffness matrices respectively, and $F(t)$ is generalized force vector, where the aerodynamic loads are accounted for.

Having the baseline defined, it is necessary to define the type of methodology to be followed concerning the coupling model to allow a true aeroelastic problem to be solved.

2.3.1 Coupling Models

A typical structure of an aeroelastic analysis framework is shown in Fig. 2.9.

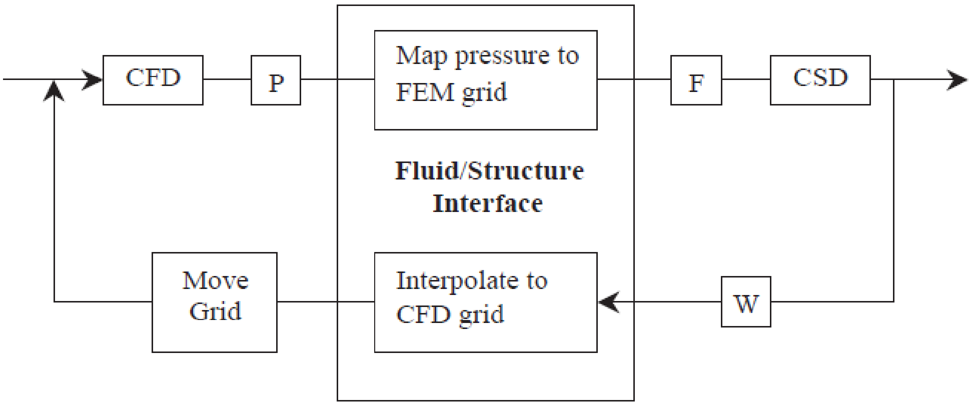


Figure 2.9: Structure of a typical coupled aeroelastic framework [9]

As shown, Fluid-Structure Interface (FSI) is paramount to connect the separate modules of the aeroelastic framework, and can be done in different ways. Some of the possible models include the fully-coupled model, the loosely coupled model and the closely coupled model, as briefly described next.

Fully Coupled

In the fully coupled model, the governing equations are rearranged by combining fluid and structural equations of motion, which are then solved and integrated in time simultaneously. Using this procedure, the fluid equation are set on an Eulerian reference system, while the structural equations are on a Lagrangian system, which leads to the matrices being orders of magnitude stiffer for structure systems as compared to fluid systems, making it almost impossible to solve using a monolithic computational scheme for large scale projects. Due to this, most fully coupled methods are only used to solve 2D problems [9].

Loosely coupled model

For a loosely coupled model, the structural and aerodynamic equations are solved using two separate solvers, and that can lead to two different computational grids that are not likely to coincide at the boundary, which calls for an interfacing technique to be developed, to exchange information between the structural and the aerodynamic modules. One advantage of this method is that it gives the flexibility in choosing different solvers for each module, but also leads to a loss in accuracy as the modules are updated only after partial or complete convergence, and so, loosely coupled approaches are usually limited to small perturbations and problems with moderate non-linearity [9].

Closely coupled model

In this model, the fluid and the structure equations are solved separately using different solvers but are coupled into one single module with exchange of information taking place at the boundary via an interface module which makes the entire CAE model tightly coupled. The information exchanged are surface loads, which are part of the solution of the CFD methods are are required as input for most dynamic structural analysis methods. This requires a deformation of the CFD surface mesh, and this call for a moving boundary technique that enables re-meshing the entire CFD domain as the solution marches in time [9].

2.3.2 Discipline Models

By selecting a loosely coupled or a closely coupled model, it is possible to have two separate solvers for both structure and aerodynamic models computations, and this allows for easier verification and/or validation of solver results and it also reduces the complexity of the implementation.

As far as aerodynamic solvers go, there are several possible to choose, as illustrated in Fig, 2.10.

For most engineering applications 2D effects models are not used as wing structures are not infinite and suffer from effects such as induced drag that a 2D based model cannot predict.

3D effects models allow the computation of aerodynamic loads on a full scale wing, and, for inviscid models, they range between lifting surface models, such as the Distributed Lagrange Multiplier method (DLM)[29],Dispersed Phase Method (DPM) the Vortex Lattice Method (VLM), which uses potential flow equations and the Transonic Small Disturbances (TSD), that allows transonic computations without the need of an Euler equation based solver. Panel methods are also based on potential flow equations and as such cannot accurately predict transonic behaviour as it is highly non-linear, but they simulate accurately the effect of the airfoil thickness on the wing. To compute aerodynamic loads at transonic regimen, solvers based on the Euler equation are commonly used [30], as well as interactions with shockwaves and other compressible phenomena. When it is necessary to compute viscous behaviour

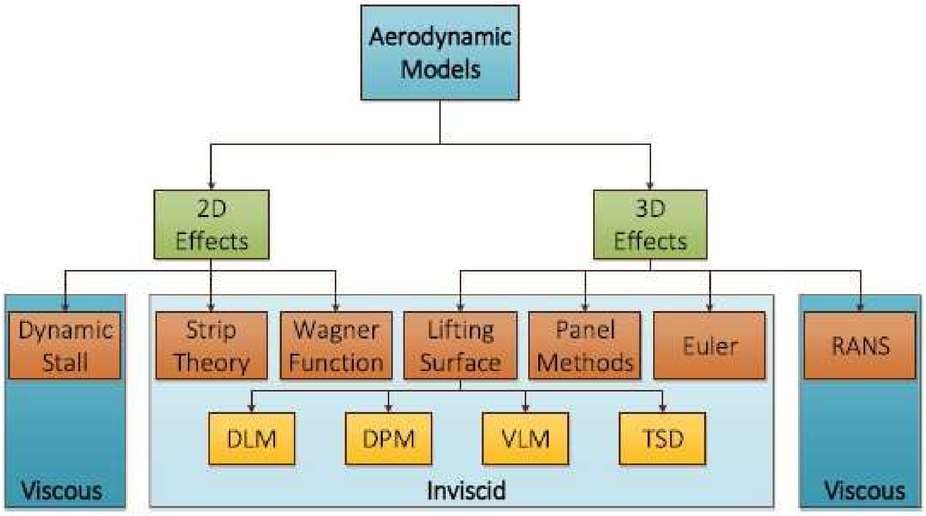


Figure 2.10: Aerodynamic models [28]

of flow around a wing, higher complexity solvers based on the Navier-Stokes equations are used, such as the Reynolds-Averaged Navier-Stokes (RANS) model, or the Direct Navier-Stokes model (DNS).

For each of the referred models, there are advantages and disadvantages, as for instance, the 2D models have low engineering application but allow to verify results with exact solutions, and solvers based on lifting surface or panel methods do not have accurate results for transonic flows, but for each application a careful thought process must be done, as with increasing complexity of the solver comes increasing computational cost.

As for structural models, they are typically based on Finite Element Analysis (FEM), which allows for several types of shapes to be discretized and so simplifies the interfacing model. Some of the possible structural models are shown in Fig. 2.11.

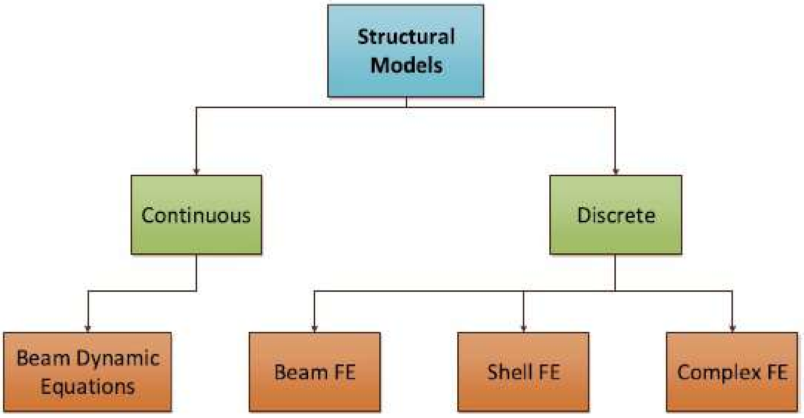


Figure 2.11: Structural models [28]

While it is possible to choose between continuous and discrete models, implementing a solver that utilizes Beam Dynamic Equations is much more complex than using Finite Elements(FE), and so it is seldom used. As for the three types of FE's shown,they can be seen as three different complexity stages: Beam FE is the simplest model possible, and should be used for low and medium fidelity applications, such as simulating a solid structure wing or a spar, while a Shell FE allows to compute the skin of a wing box and a Complex FE should be used for medium and high fidelity applications, such as the full scale computation of a wing structure.

Chapter 3

Experimental Testing

To perform any experimental test, a series of tasks must be accomplished before the actual testing begins. At first, the wing model must be constructed or repurposed from previous tests. Next, the instrumentation required to the tests must be procured and mounted on the wing model. After these steps are accomplished, an experimental procedure must be chosen, either from previous tests and other bibliographic sources [31]. After the testing is complete, the data obtained might require processing before any actual results are achieved, adding another step to the experimental methodology.

Nonetheless, experimental testing is paramount in validating any implemented numeric tool, as well as establishing dynamic behaviour of models to real-world conditions in a safe and controlled environment, as established in Sec. 1.2.

3.1 Wing Models

Two wing models where built, with the only differing measure between the two being the wing span. This is due to the second model being created by cutting part of the initial wing model, to provide test data for two different wing aspect ratios. The tested wings are shown in Fig. 3.1 installed in the wind tunnel.

Figure 3.1: Reduced span wing

Figure 3.2: Baseline wing

3.1.1 Model Geometric and Physical Properties

The experimental test model had geometric and physical properties as shown in Tab. 3.1.

Table 3.1: Initial wing geometrical and physical properties

Geometric Properties	
Airfoil	NACA 0015
Half-span	$0.75\ m$
Root chord	$0.25\ m$
Twist	$0°$
Taper ratio	1
Sweep angle	$0°$
Dihedral angle	$0°$
Material Properties	
Young's modulus(E)	$23.92\ MPa$
Shear modulus(G)	$9.14\ MPa$
Poison ratio	0.2018
Density(ρ)	$31.453\ kg/m^3$

The wing's geometric properties were selected due to both construction and analysis constraints. The root chord is $0.25\ m$ as this is the largest possible chord that allows two separate wings to be machined from the same polystyrene block, while the half-span is $0.75\ m$ so that the wing's aspect ratio is not greater than 6, as for higher values non-linear geometric effects become non-negligible and, since the numeric model cannot compute non-linearities, the results would deviate from the estimated solutions.

Also worth noting, due to the wind tunnel's test section diameter, wings with half-span over $0.90\ m$ would suffer from contaminated airflow near its extremities, which would again cause deviations from the numerical model. Concerning the model's overall shape, a rectangular wing with no dihedral and sweep was selected so that a basic validation case was defined, which also to simplified the building process.

The process of selecting the material to build the model was as follows:

1. easily machined material, preferably using an hot wire technique;
2. low resistance to bending or torsion, so that aeroelastic effects can be observed at low speeds;
3. cheap and easily obtainable.

Having these three guidelines in mind, an extruded polystyrene block was selected, made by IberFRAN, SA, a material that is usually used in thermal insulation of buildings, but it is also commonly used to build lifting surfaces of aero-models using a hot wire cut machine. The material properties were obtained through three separate experimental tests performed by Almeida [21], and are shown in detail in Appendix B.

For the second model, the geometrical properties remain the same, only the wing half-span reduces from $0.75\ m$ to $0.625\ m$.

3.2 Wind Tunnel Apparatus

3.2.1 General Description

The tunnel used was the Wind Tunnel in the AEL (Aerospace Engineering Laboratory), which was originally an open-section tunnel, and later an anechoic chamber was built around the test section to allow aeroacoustic testing.

There is no direct control of the flow velocity on the test section, instead the user controls the fan motor working frequency, which can be correlated to flow velocity after calibration work. For the testing made, the calibration work gave the following relation between fan motor frequency f and flow velocity U,

$$U = 1.437f + 0.03037 \ . \tag{3.1}$$

Although no other aeroelastic testing was ever performed at the before mentioned wind tunnel, the anechoic chamber allows for a great reduction of outside sound sources which, for the sensors used, would be a significant source of noise introduced to the signal. As for the wind tunnel test section properties, they are shown in Tab. 3.2.

Table 3.2: Wind tunnel characterization and operating conditions at test date

Test section dimensions	$1.2\ m \times 1.2\ m$
Maximum allowed velocity	$60\ m/s$
Maximum power output	$165\ kW$
Air temperature (T)	$15\ ^\circ C$
Air density (ρ)	$1.225\ kg/m^3$
Kinematic viscosity (ν)	$1.461e-005\ m^2/s$

3.2.2 Model Construction and Mount

The model was cut from a single block of extruded polystyrene, using the procedure shown in Appendix A. Since the machined wing was longer than the expected test span, a block from the same material type was machined to encase the extra length that remained. This was done purposefully, as it would be very difficult to support the wing vertically with only its section area.

The block and wing mount was then glued to a wood plaque, so that it could be easily secured to the testing table using clamps.

The testing table was a stainless steel table that was built to stand on existing supports inside the tunnel. The whole montage was made so that the wing model could be easily replaced in case it was damaged in any tests. A picture of the wing mount is shown in Fig. 3.3.

3.2.3 Instrumentation

The wing model was instrumented with three accelerometers equal to the one shown in Fig. 3.4.

The accelerometers where placed in three positions on the wing mount:

1. Leading edge- $B\&K$ type 24508B-2199113 : measures wing transverse vibration response;

Figure 3.3: Baseline instrumented wing mount

Figure 3.4: $B\&K$ type 24507 accelerometer [32].

2. Trailing edge- $B\&K$ type 24508-30915 : measures wing transverse vibration response, and with the data from the leading edge accelerometer, the torsional behaviour can be measured;

3. Table- $B\&K$ type 24507-2054330 : used to improve accuracy of the other measures by eliminating the tables' influence on the wing's elastic behaviour.

All sensors were connected to the OROS OR34 spectral analyser [33] seen in Fig. 3.5. Data from the spectral analyser was then fed into a computer with OROS Nvgate 7.1 software for vibration analysis.

The schematic of the entire montage is illustrated with Fig. 3.6.

3.2.4 Experimental Testing Procedure

Each test started with securing the wing mount to the testing table with clamps at a predetermined angle to simulate different angle of attack conditions. After initial tests, it was decided not to exceed an angle of 4° as for higher values the wing static deflection was too high and there was a risk of damage to the

Figure 3.5: OROS OR34 spectral analyser [33]

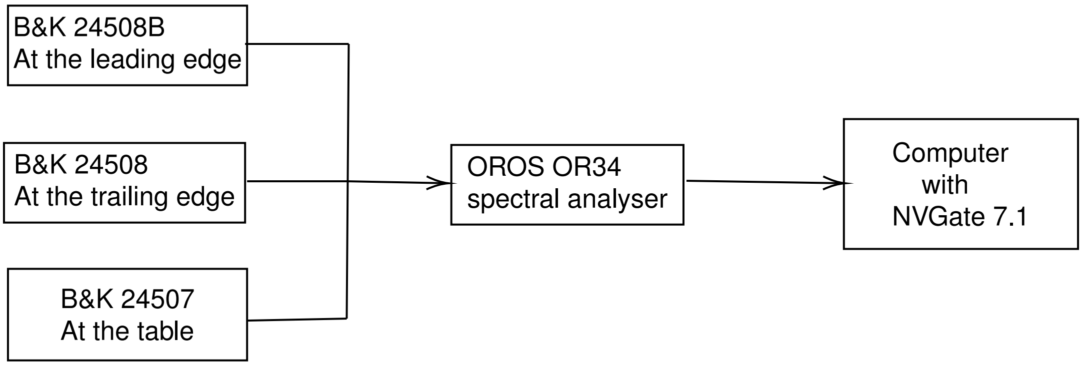


Figure 3.6: Wing instrumentation flowchart

model. The experimental procedure for both tests was as follows:

1. Secure the wing mount;
2. Start the tunnel, always at an initial fan frequency of $5\ Hz$, as the calibration performed was not accurate for lower frequencies;
3. verify that the upstream velocity was constant and the model was not in risk of damage or other hazards, a period of 90 seconds was started where sensor data was extracted;
4. After concluding the measurement a decision is made to either increase the frequency or stop the testing. The latter only occurred if doubts remained either the model could safely cope with the increased wind speed due the increase in wing deflection.

For both tests the tunnel fan frequency did not exceed $12\ Hz$ for the aforementioned reasons.

3.2.5 Experimental Data and Calculations

The experimental testing performed featured an airflow speed variation between $7.91\ m/s$ and $18.92\ m/s$, which translates to a Reynolds Number(Re) between $132,903$ and $317,958$. Since Reynolds number for the performed testing is well bellow 5×10^5, which is commonly defined as the transition Reynolds number, it is safe to assume that the flow is mainly laminar over the wing, as turbulent flow over the wing would induce undesired aeroelastic behaviour that could not be modelled using the numeric code's inviscid aerodynamic module.

It was not possible to verify if there was vortex shedding happening during the testing period, as there was no method to visualize this phenomenon, which can occur for lower Reynolds numbers, and, if its frequency is close enough to that of the structure's natural frequencies, can have implications on the aeroelastic behaviour of the model, specially on the torsional response [34]. However, since it was verified that the model presented minimal torsional oscillation, it can be safe to assume that this phenomenon does not occur or if it is occurring, does not have any measurable effect on the model.

While using accelerometers to gather the wing dynamic performance simplifies the instrumentation, the obtained data cannot be directly analysed, due to the noisy nature of the extracted signal, as seen in Fig. 3.7.

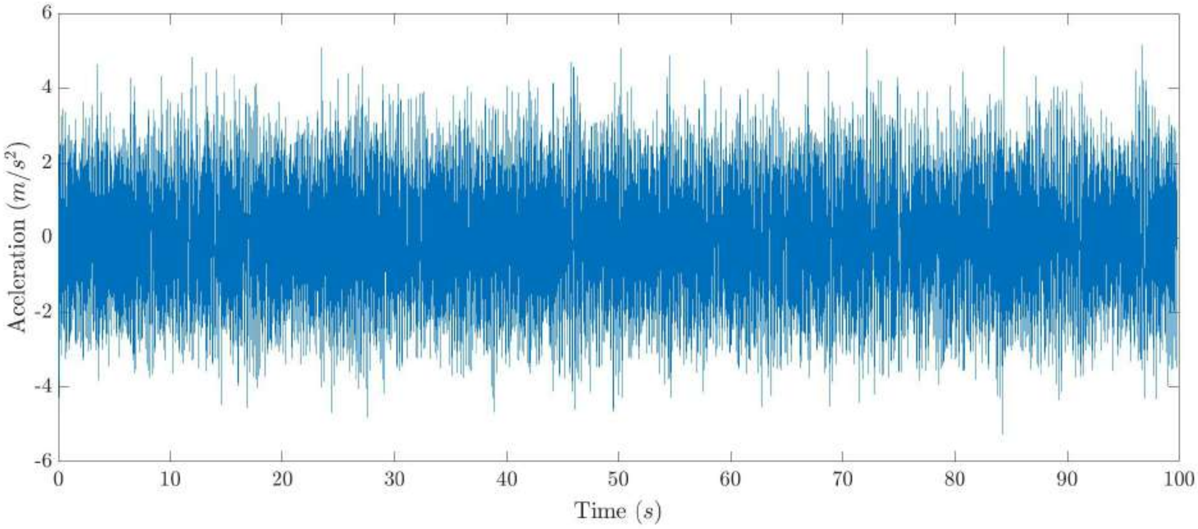


Figure 3.7: Trailing edge accelerometer unprocessed data for motor frequency of $5\ Hz$

In order to process the obtained data, a Fast-Fourier transform (FFT) is performed to distinguish the structural frequencies from those associated with noise. This calculation is performed automatically by the NVGate 7.1 software, and the result for the same motor frequency is shown in Fig. 3.8, showing both the leading edge and trailing edge data.

Using data from both accelerometers is necessary to identify structural modes, since the only value of acceleration measured is the vertical component. To distinguish flapwise from torsion modes, a comparison of peak values is made between the frequency spectra of both accelerometers. For flapwise modes, the peak values occur at the same frequency, albeit with smaller magnitude on the leading edge accelerometer due to the greater material thickness at that location. As for the torsion modes, peak values display an offset between the two sets of data, leading to a torsion effect occurring on the wing.

As previously stated, two wing models were analysed, a baseline $s = 1.5\ m$ model, and a reduced span version of the same wing with $s = 1.25\ m$.

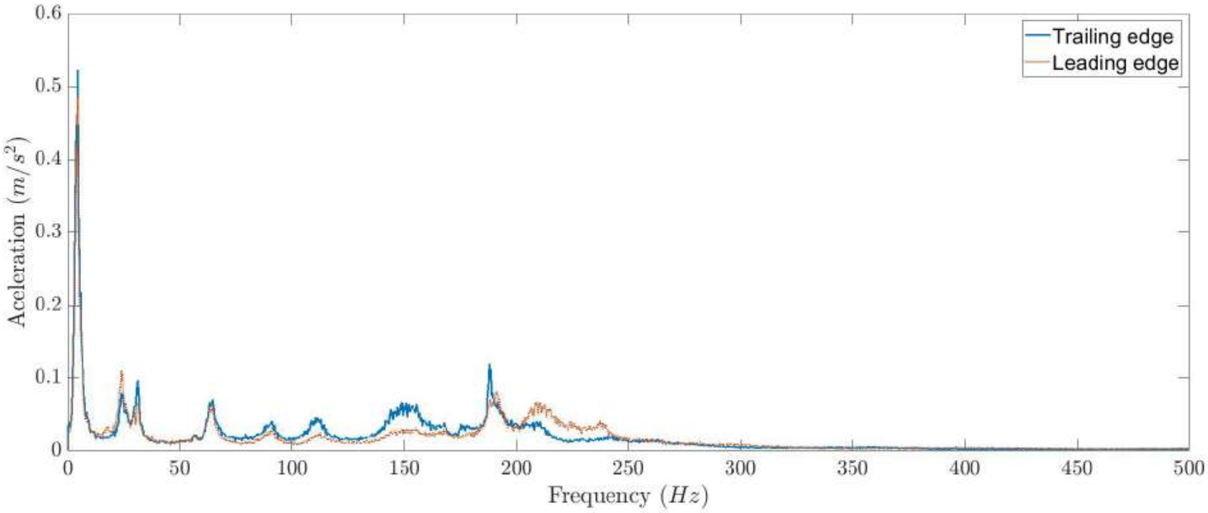


Figure 3.8: Trailing edge accelerometer processed data for motor frequency of $5\ Hz$

Baseline Wing

The evolution of the wing's frequency spectra with the flow speed for the wing with $s = 1.5\ m$ is shown in Fig. 3.9:

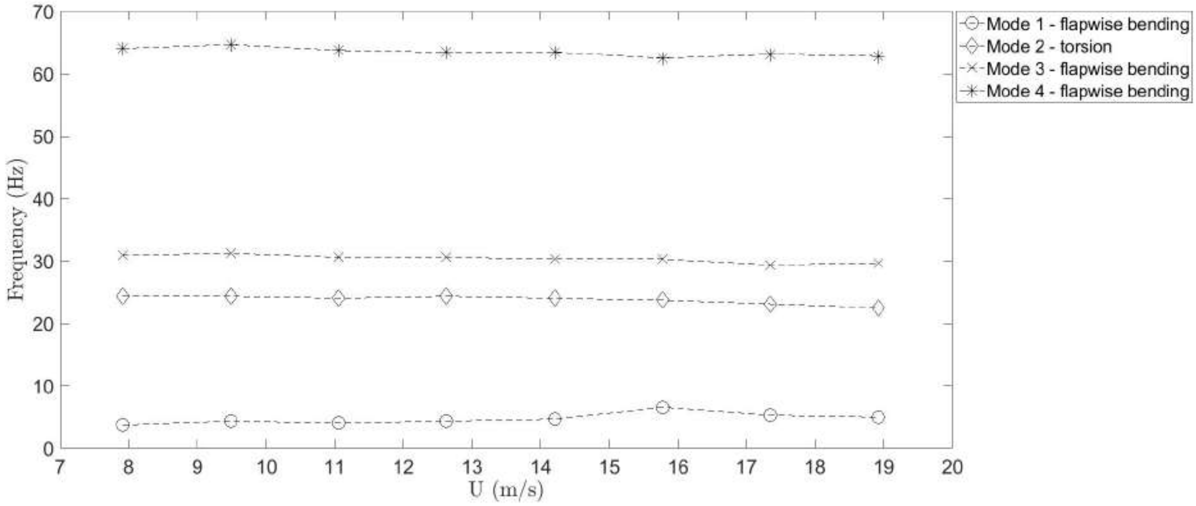


Figure 3.9: f-U graph for the experimental baseline case

From the analysis of the frequency spectra, it cannot be concluded that the tested wing experiences flutter behaviour, as the frequencies remain stable throughout the entire test. However, further incrementing the test velocity could be hazardous as the model was displaying a very high deflection and, as such, a call was made to not increase the flow speed any further.

Reduced Span Wing

The evolution of the wing's frequency spectra with the flow velocity for the wing with $s = 1.25\ m$ is shown in Fig. 3.10:

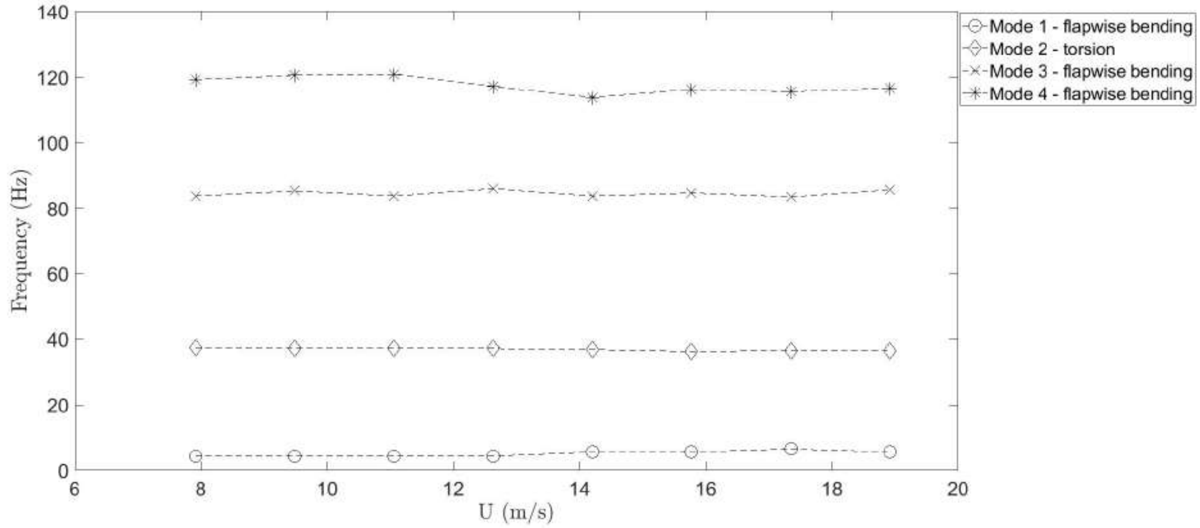


Figure 3.10: f-U graph for the experimental reduced span wing case

As expected, this model also does not display flutter behaviour, as it has a higher structural rigidity virtue of the lower wing span. Also expected is the higher values of frequency for the four modes selected in this analysis, as there is a tendency to structures with higher rigidity to display higher structural frequencies.

The test was also stopped at the same maximum flow speed to allow a better comparison between both tests, and due to the fact that, considering the previous test, the wing would display a very high deflection before any divergent behaviour could be observed.

Overall, despite not being able to determine the flutter point for the tested configuration, the test was considered to be successful as important aeroelastic data was gathered to help validate the numerical tool developed, and to define procedures and methodologies for future aeroelastic tests using the same equipment.

Chapter 4

Numerical Implementation

Any numerical implementation can be split into two components, the mathematical model and the discretization model. The selection of the mathematical model has to take into account the target application of the numerical solution, as designing a general purpose solution method is impractical and difficult to validate for all cases. The next step in a numerical solution is the discretization model, which is a method of approximating the governing equations of the mathematical model by a system of algebraic equations. By performing this approximation, the obtained results will now depend on both the mathematical model and the type of discretization chosen, which can lead to different sets of results compared to experimental cases. Therefore, it is paramount the verification of each step of the numerical implementation of an analysis tool.

In order to implement an aeroelastic tool, it is required an aerodynamic model, a structural model and the interface model, as seen in Sec. 2.3. Besides the model definition, it is also required that the design variables are selected and their influence considered. Finally, due to the multidisciplinary nature of an aeroelastic tool, special attention is required with the code structure, as modular approaches improve the ease of verification and enable future developments without major changes to its structure.

4.1 Aerodynamic Model

The process of selecting a aerodynamic model requires balancing required complexity with available computing power, as with increasing complexity the computing power required is also greatly increased, as shown in Fig. 4.1. Since the objective is to demonstrate the feasibility of the framework as a whole, and to compare with low-speed wind tunnel testing, a low complexity model was established, derived from the potential flow equations. Among the potential flow models available, as shown in Sec. 2.3.2, a 3D panel method was selected. It produces sufficiently accurate results[36] and most importantly, it facilitates the interaction between the aerodynamic and structural computational meshes, as they are both based on the lifting body surface and thus require minimal re-meshing between time-steps and reduce the complexity of the Fluid-Structure Interaction model. The methodology followed is similar to the defined by Katz [25].

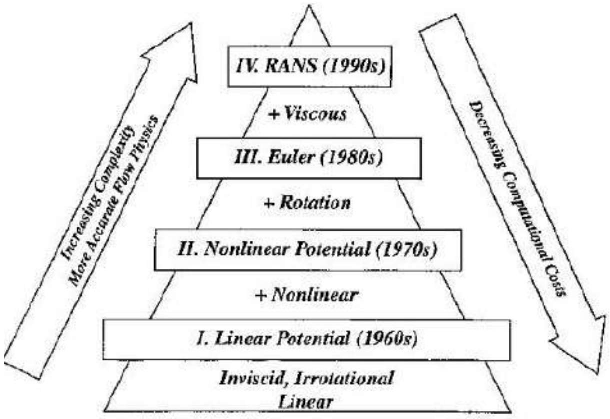


Figure 4.1: Levels of approximation for fluid flow[35]

4.1.1 3D Panel Method

As previously established, the aerodynamic model is based on a first order 3D Panel method, which itself is based on the potential flow equation,

$$\nabla^2\Phi^* = 0 \; , \tag{4.1}$$

where Φ^* is the total velocity potential. This equation is applied to a body with known boundaries S_B, as seen in Fig. 4.2, and the flow of interest is, by definition of the potential flow, incompressible, inviscid and irrotacional. Applying Green's Identity to Eq.(4.1), a general solution can be found by a sum of source (σ) and doublet (μ) distributions placed on the S_B boundary,

$$\Phi^*(x,y,z) = -\frac{1}{4\pi}\int_{S_B}\left[\sigma\left(\frac{1}{r}\right) - \mu\mathbf{n}\cdot\nabla\left(\frac{1}{r}\right)\right]dS + \Phi_\infty \; , \tag{4.2}$$

where r is the distance to a point outside the S_B boundary and vector $\mathbf{n}$ points in the direction of potential jump μ.

The formulation presented does not uniquely describe a solution since a large number of source and doublet distributions will satisfy a given set of boundary conditions [25], and so a specific combination of source and doublet combinations must be chosen. Considering typical examples, it is defined that the

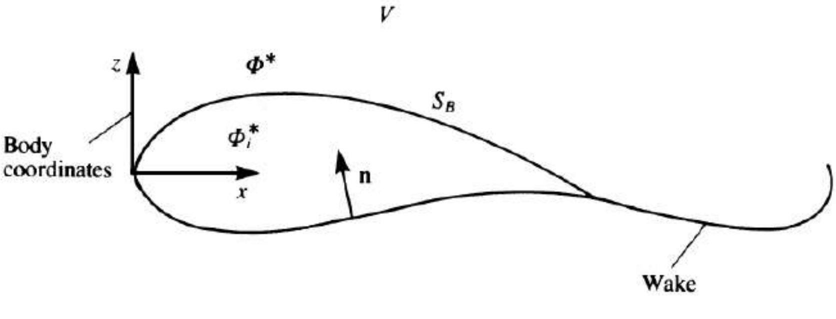


Figure 4.2: Potential flow over a closed body [25].

wake will be modelled by thin doublet sheets, and so Eq. (4.2) becomes

$$\Phi^*(x,y,z) = \frac{1}{4\pi}\int_{body+wake} \mu \mathbf{n}\cdot\nabla\left(\frac{1}{r}\right) dS - \frac{1}{4\pi}\int_{body} \sigma\left(\frac{1}{r}\right) dS + \Phi_\infty \; . \tag{4.3}$$

With the basic formulation defined, it is required to define boundary conditions. For most physical problems, they can be split into two types:

- Neumann boundary condition — The normal velocity component is zero at boundary S_B, which means $\frac{\delta\Phi^*}{\delta n} = 0$;
- Dirichlet boundary condition — Φ^* is specified at the boundary, so that zero normal flow condition is indirectly met.

The implemented boundary condition is a Dirichlet condition, which implies that the perturbation potential Φ has to be specified on the entire S_B surface. To define the condition $\nabla\left(\Phi + \Phi_\infty\right)\cdot\mathbf{n} = 0$ in terms of the velocity potential, we have

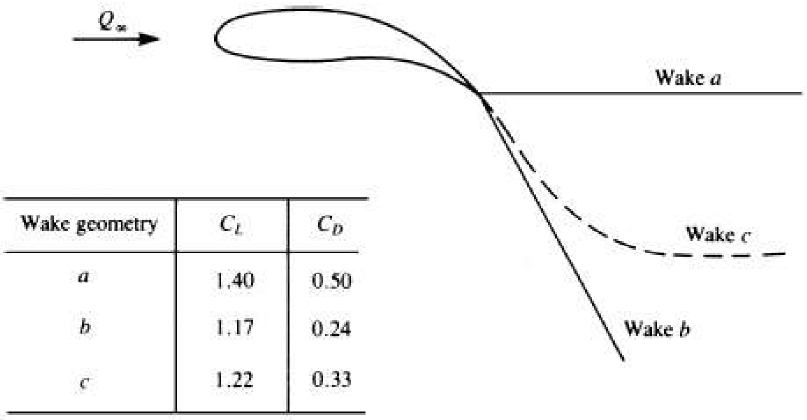
$$\Phi^*(x,y,z) = \frac{1}{4\pi}\int_{body+wake} \mu\frac{\delta}{\delta n}\left(\frac{1}{r}\right) dS - \frac{1}{4\pi}\int_{body} \sigma\left(\frac{1}{r}\right) dS + \Phi_\infty = cte. \; , \tag{4.4}$$

with the boundary conditions inserted into the problem formulation and defining the source strength as

$$\sigma = \mathbf{n}\, U_\infty \; , \tag{4.5}$$

The problem can be reduced to a set of algebraic equations, with doublet distribution μ as the unknowns to the problem.

Another important definition that affects the accuracy of the method is the definition of the wake geometry, as seen in Fig.4.3. The wake geometry c gives the result closest to experimental results [25], but it is also the most difficult to define computationally, as such it is not considered in the formulation developed. Instead, the wake geometry defined resembles that of the wake geometry a, as it allows fewer wake panels to be defined, which decreases significantly the computational times. By selecting the



Wake geometry	C_L	C_D
a	1.40	0.50
b	1.17	0.24
c	1.22	0.33

Figure 4.3: Effect of predefined wake geometry on the aerodynamics of an $AR = 1.5$ wing [25].

wake geometry a, it is noted that both C_L and C_D are overestimated, comparing with the experimental results. By overestimating the aerodynamic forces, it can also be inferred that the elastic behaviour will be overestimated, which means that dynamic instabilities will appear earlier than in experimental cases. By overestimating the aerodynamic parameters an added safety factor is introduced, leading to reduction of potential hazards in experimental testing of the same design.

With the wake geometry defined by discretizing the body's surface into N panels and the wake in N_W panels, Eq. (4.4) is rewritten as

$$\sum_{k=1}^{N}\frac{1}{4\pi}\int_{body\ panel}\mu\mathbf{n}\cdot\nabla\left(\frac{1}{r}\right)dS+\sum_{l=1}^{N_W}\frac{1}{4\pi}\int_{wake\ panel}\mu\mathbf{n}\cdot\nabla\left(\frac{1}{r}\right)dS-\sum_{k=1}^{N}\frac{1}{4\pi}\int_{body\ panel}\sigma\left(\frac{1}{r}\right)dS=0\ , \tag{4.6}$$

Defining a collocation point P at the centre point of each panel, with four vortices of a panel k, as shown in Fig. 4.4, and assuming constant source strength σ and doublet strength μ for each panel, Eq. (4.6) can be further simplified into

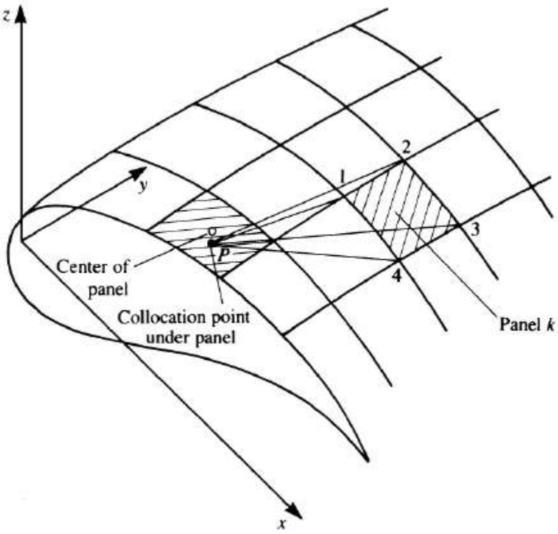


Figure 4.4: Influence of panel k on point P [25].

$$\sum_{k=1}^{N}C_k\mu_k+\sum_{l=1}^{N_W}C_l\mu_l+\sum_{k=1}^{N}B_k\sigma_k=0 \quad \text{for each internal point } P\ , \tag{4.7}$$

where

$$C_k=\frac{1}{4\pi}\int_{1,2,3,4}\frac{\delta}{\delta n}\left(\frac{1}{r}\right)dS\bigg|_k \tag{4.8a}$$

and

$$B_k=-\frac{1}{4\pi}\int_{1,2,3,4}\left(\frac{1}{r}\right)dS\bigg|_k\ . \tag{4.8b}$$

Equation (4.7) is the numerical equivalent of the boundary condition. By using the Kutta condition [25], the wake doublets can be defined in terms of the unknown surface doublets μ_k. This leads to the algebraic relation that can be substituted into de C_k coefficients of the unknown surface doublet such that

$$A(k) = \begin{cases} C_k & \text{if panel is not at the trailing edge} \\ C_k \pm C_t & \text{if panel is at the trailing edge ,} \end{cases} \tag{4.9}$$

where the ± signal depends on if the is at the upper or lower side of the trailing edge. Consequently, for each collocation point P, a linear algebraic equation containing N unknown singularity variables μ_k can be derived,

$$\sum_{k=1}^{N} A_k \mu_k = -\sum_{k=1}^{N} B_k \sigma_k \; . \tag{4.10}$$

Evaluating Eq. (4.10) at each of the N collocation points results in a linear algebraic system, of equations of size N. Since the value of σ_k is known, the right-hand side(RHS) of the equation can be computed, leaving the system as

$$\begin{bmatrix} a_{11}, & a_{12}, & \cdots, & a_{1N} \\ a_{21}, & a_{22}, & \cdots, & a_{2N} \\ \vdots & \vdots & & \vdots \\ a_{N1}, & a_{N2}, & \cdots, & a_{NN} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_N \end{bmatrix} = \begin{bmatrix} RHS_1 \\ RHS_2 \\ \vdots \\ RHS_N \end{bmatrix} \; . \tag{4.11}$$

The coefficients a_{ij} are known as the aerodynamic induced coefficients.

4.1.2 Aerodynamic Loads

After solving Eq.(4.11), the unknown singularity values are obtained , and so the the velocity components can be evaluated. Using panel coordinates (l, m, n) as shown in Fig. 4.5,

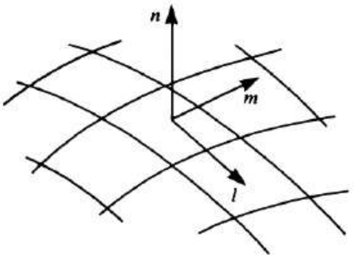


Figure 4.5: Panel coordinate system for evaluating the tangential velocity components [25].

the components are

$$v_l = -\frac{\delta \mu}{\delta l} \; , \tag{4.12a}$$

$$v_m = -\frac{\delta \mu}{\delta m} \tag{4.12b}$$

and

$$v_n = -\sigma \; . \tag{4.12c}$$

These perturbation velocities are related with the local velocity by $V_k = (U_{\infty_l} U_{\infty_m} U_{\infty_n}) + (v_l, v_m v_n)_k$. The perturbation velocities can be evaluated numerically with central differences, by knowing the distance between two collocation points and the singularity values on both panels. By defining the local velocity on each panel, the pressure coefficient C_p can be computed on a panel basis,

$$C_{p_k} = 1 - \frac{V_k^2}{U_\infty^2} \ . \tag{4.13}$$

Finally, the aerodynamic force F_k for each panel can be computed from

$$F_k(l,m,n) = -q_\infty \, C_{p_k} \cdot S_k(l,m,n)_k \ , \tag{4.14}$$

where S_k is the panel area vector projected onto the panel's mean plane and q_∞ is the dynamic pressure.

4.1.3 Quasi-Unsteady Panel Method Implementation

For a potential equation type of problem, the equation itself does not include time dependent terms directly, so the time dependency must be introduced through the boundary conditions, which means that a steady panel method solver can be used to model unsteady flows with small modifications [25], leading to a quasi-unsteady model. The main differences are the aforementioned boundary conditions, the wake implementation, and the pressure computation.

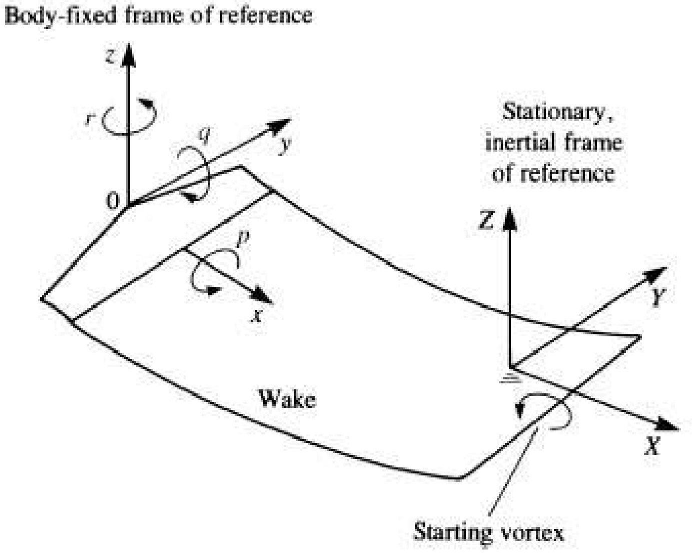


Figure 4.6: Inertial and body coordinates used to describe the motion of the body [25].

For an unsteady panel method implementation, the wake is defined on a time step basis so, for each time step, a new wake panel is added to the pre-existing wake panels. Considering a constant flow of speed U_∞ in the positive x direction shown in Fig. 4.6, for each time step a translation is applied to the body frame of reference defined as $(X_0, Y_0, Z_0) = (-U_\infty t, 0, 0)$. This translation is then used to define the new wake panel, with one extremity on the previous wake panel and the other at a X_0 distance from the other extremity. As for the wake geometry, since it now depends on the trailing edge shape, if oscillations or other movement types are induced onto the wing, they will also be verified on the wake panels. The new boundary condition, that replaces the one established in Eq. (4.5), is defined as

$$\sigma = -\mathbf{n} \cdot (\mathbf{V_0} + \mathbf{v_{rel}} + \mathbf{\Omega} \times \mathbf{r}) \; . \tag{4.15}$$

where $\mathbf{V_0} = (\dot{X}_0, \dot{Y}_0, \dot{Z}_0)$ is the velocity of the (x, y, z) system's origin, $\mathbf{v_{rel}} = (\dot{x}, \dot{y}, \dot{z})$ is the relative velocity of the body fixed frame of reference relative to the inertial frame of reference, $\mathbf{\Omega}$ is the rate of rotation of the body's frame of reference, as shown in Fig. 4.6, and $\mathbf{r}$ is the position vector .

For the specific case of a flat plate at an angle of attack α, which is the closest theoretical approximation to a wing subject ot the same conditions, $\mathbf{V_0} = (-U_\infty, 0, 0)$, $\mathbf{\Omega} = 0$, which translates to

$$\sigma = -\mathbf{n} \cdot \mathbf{V_0} \; . \tag{4.16}$$

This case has a boundary solution equal to that of the steady case, thus requiring no major rewrite to the formulation.

For the pressure computation, considering that the perturbation velocities have identical definitions to the steady case, the pressure coefficient on each body panel has a similar definition to the steady case, only with an extra time-stepping term,

$$C_{p_k} = 1 - \frac{V_k^2}{U_\infty^2} - \frac{2}{U_\infty^2}\frac{\delta\phi}{\delta t} \; . \tag{4.17}$$

To determine the pressure coefficient at time $t + \Delta t$ a Backward Euler method is used [37], yielding

$$C_{p_k}^{t+\Delta t} = 1 - \frac{V_{t+\Delta t}^2}{U_\infty^2} - \frac{2}{U_\infty^2}\frac{\phi^{t+\Delta t} - \phi^t}{\Delta t} \; . \tag{4.18}$$

The main advantage of using a Backward Euler method is that it is an implicit scheme, making the solution unconditionally stable and, as such, enabling the use of larger time steps than with explicit schemes [38].

The aerodynamic forces computation is identical to the steady case.

4.2 Structural Model

There are several structural models capable of simulating the dynamic response of a body subject to time-dependent forces, as seen in Sec. 2.3.2. Considering the type of solution desired, a discrete model is required, specifically a Beam Finite Element model, as it is the simplest FEM model while also maintaining reasonable accuracy for the selected application [39].

By determining geometric and aerodynamic parameters on an airfoil section basis, two types of wing designs are supported, solid wing and hollowed wing, with the user selecting either design according to its needs. The solid wing definition is a new addition the pre-existing module and it is the design used for the numerical studies performed as it is easier to construct a wing model with this design, as opposition to a hollow wing model with spars, that would require extensive machining time. Both wing sections are shown in Fig. 4.7.

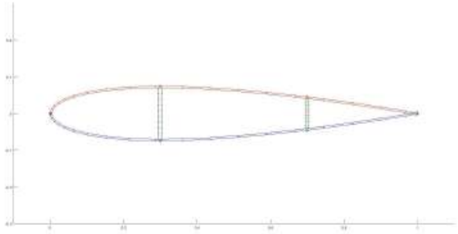


Figure 4.7: Hollow wing box

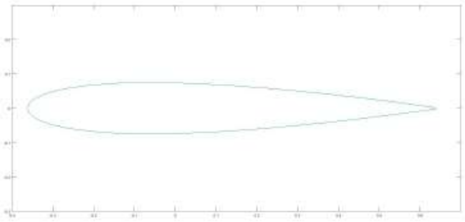


Figure 4.8: Solid wing box

4.2.1 Modal Analysis

Considering a system with n-degrees of freedom and no damping, the fundamental equation is [40]

$$[M]\ddot{\boldsymbol{x}} + [K]\boldsymbol{x} = \{\boldsymbol{F}(t)\} \ , \tag{4.19}$$

where $[K]$ and $[M]$ are, respectively, the stiffness and Mass matrices, $\boldsymbol{x}$ is the systems' displacements and rotations and $\{F(t)\}$ is the external force vector, which, for this module,consists of the aerodynamic forces. The reasoning for the exclusion of damping in this particular application is that damping is not easily estimated theoretically. Although there are models to estimate the damping, such as Rayleigh damping matrix [41], this methodology requires experimental determination of damping coefficients, and as such, it is off the scope of the work developed. Another important factor is that a damped system would most likely display divergent behaviour for a higher airspeed than the undamped system and so, by having an undamped system, a first estimation of the divergence speed is achieved that will be lower than the real divergence speed. Since this is an eigenvalue problem, Eq. (4.19) is rearranged into

$$([M] - \omega^2[K])\boldsymbol{x} = 0, \tag{4.20}$$

where ω is the systems' angular eigenfrequencies. With this relation, the systems' frequencies can be obtained, which allows for better prediction of wing behaviour and also to adjust the ideal time step to perform computational calculations.

As for the selection of the time step, by having the system's predicted frequencies, and by the Nyquist-Shannon sampling theorem [42]

$$t_s = \frac{1}{2f_{max}} \ , \tag{4.21}$$

where f_{max} is the maximum frequency that is to be observed by the structural solver. By adjusting the sample time, the correct structural frequency spectra can be obtained.

4.2.2 3D Beam Finite Element Implementation

A typical FEM analysis has three main features [39]:

- The geometry domain is discretized into a set of simple sub-domains, the so called finite elements. Their discretization is called a mesh of initial elements;
- The physical process is evaluated at each element and approximated by functions(shape functions) and algebraic equations are developed at each element corners, called nodes, relating physical quantities;
- The element equations are assembled using continuity and the solution is obtained for every node.

For a beam type FE, a wing divided in a number of spanwise sections, and, to facilitate the Fluid-Structure Interaction, these sections are the same as the sections produced for the aerodynamic panel method.By matching the aerodynamic and structural meshes, the complexity of the interface mechanism is drastically reduced, and by reducing its complexity, the computing times are also greatly reduced [9]. The wing's geometric properties are assessed on a section basis, as well as the aerodynamic forces applied on the section.

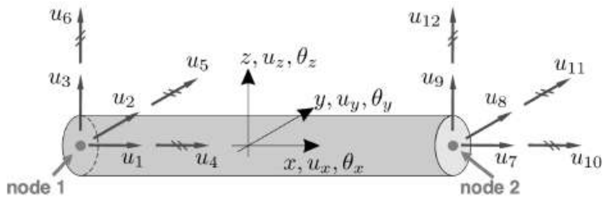


Figure 4.9: 3D beam element [21]

The selected 3D beam element, shown in Fig.4.9, is based on the Euler-Bernoulli beam theory [43], so that the bending and torsional displacements are uncoupled and thus the element formulation can be split into three governing equations, each concerning bending deformation, axial deformation and free torsion. Starting with bending deformation, given by

$$\frac{d^2}{dx}\left(EI_{yy}\frac{d^2u_z}{dx}\right) - q(x) = 0 \ , \tag{4.22}$$

where E is the Young's modulus, I_{yy} is the moment of inertia about the y axis, , u_z is the vertical displacement and $q(x)$ is a distributed force in the z direction along the x axis. Since Eq. (4.22) is a fourth order differential equation, it is required to have four boundary conditions to solve it, that can be considered as one of three types:

- Free end: shear force and moments are zero, $V = M = 0$;
- Simple Support: moment and vertical displacement are zero, $M = u_z = 0$;
- Fixed Support: vertical displacement and rotation are null,$u_z = \frac{du_z}{dx} = 0$.

To achieve the weak form of Eq. (4.22), it is multiplied with an arbitrary field $v(x)$, which leads to

$$[vV]_0^L - \left[\frac{dv}{dx}M\right]_0^L - \int_0^L \left(\frac{d^2}{dx^2}EI_{yy}\frac{d^2u_z}{dx^2} - vq\right) dx = 0 \ , \tag{4.23}$$

Where L is the beam's length. For axial deformation, the equation is given by

$$\frac{d}{dx}\left(EA\frac{du_x}{dx}\right)+b(x)=0\ , \tag{4.24}$$

where A is the cross-sectional area of the beam, u_x is the axial displacement along the x axis and $b(x)$ is the axial applied force per unit length. As Eq. (4.24) is a second order differential equation, it requires two boundary conditions to be solved. Such conditions can be displacement and applied forces at the boundary nodes, $e.g. u(x_1)=u_1$ and $F(x_2)=AE\frac{du}{dx}|_{x_2}=F_2$. Following the same methodology as for the bending deformation, the weak formulation for Eq. (4.24) is

$$\left[vAE\frac{du_x}{dx}\right]_0^L-\int_0^L\left(\frac{dv}{dx}AE\frac{du_x}{dx}-vb\right)dx=0\ . \tag{4.25}$$

The free torsion of a beam subject to a twisting load is given by

$$\frac{d}{dx}\left(GJ\frac{d\theta_x}{dx}\right)+m_x=0\ , \tag{4.26}$$

where G is the material's shear modulus, J is the torsional moment of inertia, θ_x is the torsion and m_x is the distributed twisting load. Since Eq.(4.26) is a second order differential equation, two boundary conditions are required, such as

- Applied twist θ_1 at point x_1 $\theta(x_1)=\theta_1$;
- Applied torque T_2 at point x_2 $T(x_2)=GJ\frac{d\theta_x}{dx}|_{x_2}=T_2$

Finally the weak formulation is

$$\left[vGJ\frac{d\theta_x}{dx}\right]_0^L-\int_0^L\left(\frac{d_v}{dx}GJ\frac{d\theta_x}{dx}-vm_x\right)dx=0\ . \tag{4.27}$$

By combining the three weak formulations and solving for each beam section, the mass and stiffness matrices are obtained, starting with the stiffness matrix K_e

$$[K]_e=\begin{bmatrix}
X & 0 & 0 & 0 & 0 & 0 & -X & 0 & 0 & 0 & 0 & 0\\
 & Y_1 & 0 & 0 & 0 & Y_2 & 0 & -Y_1 & 0 & 0 & 0 & Y_2\\
 & & Z_1 & 0 & -Z_2 & 0 & 0 & 0 & -Z_1 & 0 & -Z_2 & 0\\
 & & & S & 0 & 0 & 0 & 0 & 0 & -S & 0 & 0\\
 & & & & Z_3 & 0 & 0 & 0 & Z_2 & 0 & Z_4 & 0\\
 & & & & & Y_3 & 0 & -Y_2 & 0 & 0 & 0 & Y_4\\
 & & & & & & X & 0 & 0 & 0 & 0 & 0\\
 & & & & & & & Y_1 & 0 & 0 & 0 & -Y_2\\
 & & & & & & & & Z_1 & 0 & Z_2 & 0\\
 & & & & & & & & & S & 0 & 0\\
 & & & & & & & & & & Z_3 & 0\\
 & & & & & & & & & & & Y_3
\end{bmatrix} \tag{4.28}$$

where

$$X = \frac{AE}{L}, \qquad Y_1 = \frac{12EI_z}{L^3}, \qquad Y_2 = \frac{6EI_z}{L^2}, \qquad Y_3 = \frac{4EI_z}{L}, \qquad Y_4 = \frac{2EI_z}{L},$$

$$S = \frac{GI_x}{L}, \qquad Z_1 = \frac{12EI_y}{L^3}, \qquad Z_2 = \frac{6EI_y}{L^2}, \qquad Z_3 = \frac{4EI_y}{L}, \qquad Z_4 = \frac{2EI_y}{L}.$$

The stiffness matrix is required by all FEM analysis, be it static or dynamic. However, in order to perform a dynamic structural analysis, it is also required the mass matrix M_e,

$$[M]_e = \begin{vmatrix}
X_1 & 0 & 0 & 0 & 0 & 0 & X_2 & 0 & 0 & 0 & 0 & 0 \\
 & Y_1 & 0 & 0 & 0 & Y_2 & 0 & -Y_1 & 0 & 0 & 0 & Y_2 \\
 & & Z_1 & 0 & -Z_2 & 0 & 0 & 0 & -Z_1 & 0 & -Z_2 & 0 \\
 & & & S_1 & 0 & 0 & 0 & 0 & 0 & S_2 & 0 & 0 \\
 & & & & Z_3 & 0 & 0 & 0 & Z_2 & 0 & Z_4 & 0 \\
 & & & & & Y_3 & 0 & -Y_2 & 0 & 0 & 0 & Y_4 \\
 & & & & & & X_1 & 0 & 0 & 0 & 0 & 0 \\
 & & & & & & & Y_1 & 0 & 0 & 0 & -Y_2 \\
 & & & & & & & & Z_1 & 0 & Z_2 & 0 \\
 & & & & & & & & & S_1 & 0 & 0 \\
 & & & & & & & & & & Z_3 & 0 \\
 & & & & & & & & & & & Y_3
\end{vmatrix} \tag{4.29}$$

where

$$X_1 = \frac{AL\rho}{3}, \qquad X_2 = \frac{AL\rho}{6}, \qquad Y_1 = \frac{13AL\rho}{35}, \qquad Y_2 = \frac{11AL^2\rho}{210}, \qquad Y_3 = \frac{AL^3\rho}{105}, \; Y_4 = -\frac{AL^3\rho}{140},$$

$$S_1 = \frac{I_xL\rho}{3}, \qquad S_2 = \frac{I_xL\rho}{6}, \qquad Z_1 = \frac{13AL\rho}{35}, \qquad Z_2 = -\frac{13AL^2\rho}{420}, \qquad Z_3 = \frac{AL^3\rho}{105}, \; Z_4 = -\frac{AL^3\rho}{140}.$$

Both matrices are based on the stiffness and mass matrices for the BEAM4 3D elastic beam used in ANSYS APDL [44].

4.2.3 Dynamic Structural Behaviour and Implementation

To implement the dynamic structural response a Newmark - β time integration scheme was chosen [45] as, with careful selection of parameters, the method is implicit and unconditionally stable, and so the time step can be chosen independently from any stability issues. While this method is reasonably accurate for the computation of displacements for all time steps [46], the values of node velocity and acceleration tend to be poorly predicted as, for the chosen values of the Newmark time integration parameters, the estimated accelerations and velocities are average values for the current time-step [47]. As a result, any numerical analysis made shall consider the nodal displacements in preference to the nodal velocities and acceleration. The procedure is described as follows:

1. Define first acceleration estimation $\ddot{x}^i = M^{-1}(F - K\ x^i)$;
2. Define Newmark time integration parameters $\beta = 0.5$, $\gamma = 0.25$ and time step Δt;
3. Calculate the integration constants: $a_0 = \frac{1}{\beta \Delta t^2}$, $a_1 = \frac{1}{\beta \Delta t}$, $a_2 = \frac{1}{2\beta} - 1$, $a_3 = \Delta t(1-\gamma)$ and $a_4 = \gamma \Delta t$;
4. Obtain the effective stiffness matrix $K_{eff} = K + a_0 M$;
5. Define the R_{eff} matrix $R_{eff}^{i+1} = F + M\left(a_0 x^i + a_1 \dot{x}^i + a_2 \ddot{x}^i\right)$;
6. Find the displacement, velocity and acceleration values for the next time-step: $x^{i+1} = K_{eff}^{-1} R_{eff}^{i+1}$, $\ddot{x}^{i+1} = a_0\left(x^{i+1} - x^i\right) - a_1 \dot{x}^i - a_2 \ddot{x}^i$ and $\dot{x}^{i+1} = \dot{x}^i + a_3 \ddot{x}^i + a_4 \ddot{x}^{i+1}$.

Where K and M are, respectively the stiffness and mass matrices defined previously and F is the external loads vector.

4.2.4 Solid wing section implementation and verification

In order to perform the comparison between experimental and numerical results, due to constraints in wing size and building material, the test wing was constructed as a solid section wing. This facilitates the building and testing procedures, while also reducing complexity of the section properties estimation.

Since, originally, the framework only allowed for wing discretization as a wing-box model, a new wing discretization module was developed to further improve program functionally and allow the use of solid wings in experimental testing for direct comparison with the numerical results. This new discretization method requires fewer parameters than the wing box discretization [21], as no skin and web thickness are required and no spars are used. The required parameters to define the solid wing section are shown in Tab. 4.1.

Table 4.1: Solid wing section parameters

Wing geometric properties
Span
Airfoil shape
Chord
Sweep,dihedral and twist angles
Angle of attack
Wing material properties
Elastic modulus
Shear modulus
Material density

The new section properties are estimated using the same base as the existing in Almeida [21] framework and, to validate the new properties, the same wing was defined in the Aeroelastic Framework and on ANSYS APDL software, defined in Tab 4.2 and in Fig. 4.10.

Using the methodology defined in the ANSYS reference manual [44], the section properties are estimated as shown in Tab. 4.3.

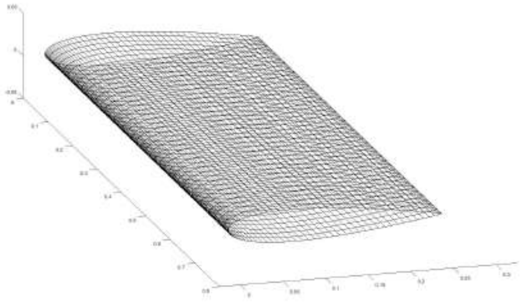


Figure 4.10: Computational mesh for the section properties verification.

Table 4.2: Test wing for section properties comparison

Parameter	Value
Airfoil	NACA 0015
Span	$1.5\ m$
Chord	$0.25\ m$
Taper Ratio	1
Sweep	$0°$
Dihedral	$0°$
Twist	$0°$

Table 4.3: Comparison of section properties obtained with the aeroelastic framework and ANSYS APDL.

	Aeroelastic Framework	ANSYS APDL	Difference
A	$0.0064\ m^2$	$0.0064\ m^2$	0%
I_{xx}	$2.2573e-0.05\ m^4$	$2.268e-0.05\ m^4$	0.47%
I_{yy}	$5.1438e-007\ m^4$	$5.2e-007\ m^4$	1.08%
I_{zz}	$2.2058e-005\ m^4$	$2.216e-005\ m^4$	0.46%

Since the values are similar to those computed on ANSYS APDL, it can be verified that section properties are being correctly computed.

4.3 Wing Parametrization

The process of designing a wing usually consists of defining a number of wing characteristics:

- wing area S;
- airfoil cross-section shape;
- wing aspect ratio $AR = s^2/S$;
- wing taper ratio $\lambda = \frac{c_{tip}}{c_{root}}$;
- wing sweep angle Λ;
- wing dihedral angle Γ.

Wing area S is the first design parameter to be defined, usually obtained from empiric relations for wing loading [48], and relates directly to C_L definition, as it is the reference area for most non-dimensional aerodynamic calculations, and, together with a wing's aspect ratio, it defines basic wing dimensions, the span (s) and average chord $(\bar{c})$.

The airfoil's cross-section shape essentially defines the pressure distribution on the wing, with the airfoil type selection depending on design specifications [35], as well as its maximum thickness. For initial wing design, it is normal to select pre-existing airfoils, such as the NACA 4,5 and 6 series.

The aspect ratio AR is defined by the wingspan versus its area, and it mostly relates with the L/D ratio, in a way that a higher aspect ratio means a higher L/D, which itself means that the aircraft can have a higher payload or range [48].

A typical simple rectangular wing with an aspect ratio of 7 is shown in Fig. 4.11.

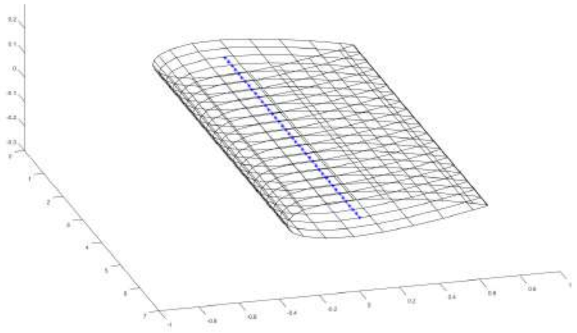


Figure 4.11: Rectangular wing

The wing taper ratio is defined by both tip and root chord values, c_{tip} and c_{root}, so a rectangular wing has a unit taper ratio. The key effect of the taper ratio is the minimization of lift-induced drag, as from lifting line theory, an elliptical wing has the minimum lift-induced drag, so approximating this shape with a trapezoidal wing form gives the best results, and leads to a taper ratio of 0.4. In the program used, taper ratio is indirectly defined by both c_{tip} and c_{root}, and a wing with taper ratio of 0.4 is shown in Fig.4.12.

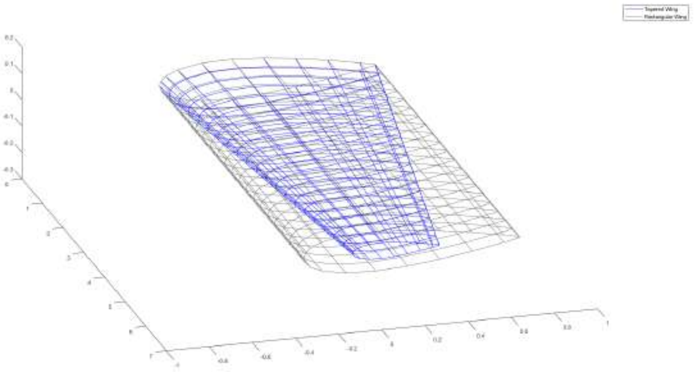


Figure 4.12: Tapered wing

Sweep angle Λ is defined as the angle between a line perpendicular to the aircraft centreline and a line parallel to the leading edge. Its primary use is to increase a wing section's critical Mach by reducing the effective Mach number at which the wing is operating. Since its main task is to minimize transonic effects, it is only applied to wings flying at a cruise speed greater than Mach 0.5. In the aeroelastic framework used, sweep angle is defined by the wing leading edge tip chordwise coordinate, x_{tip}, and a wing with sweep of $15°$ is shown in Fig. 4.13.

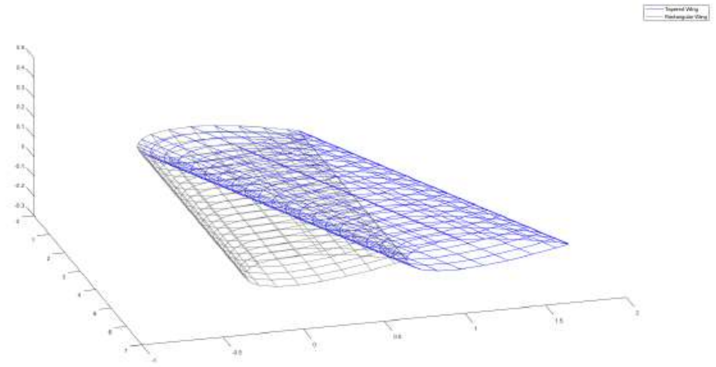


Figure 4.13: Swept wing

The wing dihedral angle Γ is defined as the upward angle between the wing tip and the wing root, and its main influence is concerning the aircraft's roll stability, having little effect on the wing actual performance, although wings with negative dihedral angle have slightly lower induced drag, at the expense of decreased lateral stability. A wing with dihedral angle of $7°$ is shown in Fig. 4.14.

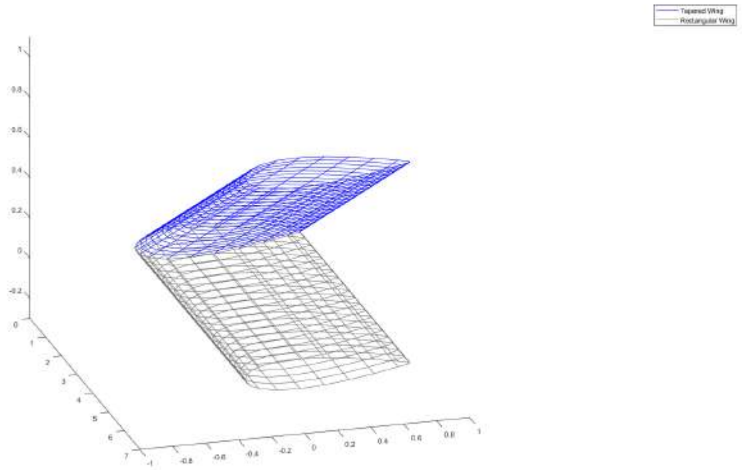


Figure 4.14: Dihedral wing

4.4 Fluid-Structure Interaction

As previously stated, the interface between aerodynamic and structural solvers in an closely coupled aeroelastic tool is one of the most difficult modules to implement, in part due to the difference between both solvers' coordinate systems. This problem is non-existent for the chosen solvers, as they both use an Lagrangian frame of reference which greatly reduces the complexity of the interface module. Nonetheless, for each time iteration, there is a change in shape of the wing and this shape is dictated by the displacements produced by the structural module. So, in order to produce the wing shape for the next time-iteration, the implemented interface model does as follows:

1. Wing displacements and twist are determined by the structural solver using the force and moment field from the aerodynamic module at t_n;
2. From the displacements and twist and also the mass and stiffness matrices, the structure's dynamic behaviour (velocities, accelerations) is computed using the Newmark - β time integration scheme as seen in Sec 4.2.3;
3. Using the structures dynamic behaviour, the mesh is changed using one of four interface algorithms (described next);
4. Finally,a 3D rigid body transformation is applied to the body to update the aerodynamic solver mesh for computations at $t_{n+1} = t_n + \Delta t$.

The interface algorithms, as initially implemented by Almeida [21], are:

- $CSS1$: Conventional Serial Staggered Algorithm;
- $CSS2$: Serial Staggered Algorithm with First Order Structural Predictor ;
- $CSS3$: Serial Staggered Algorithm with Second Order Structural Predictor;
- $CSS4$: Improved Serial Staggered Algorithm.

They estimate the new CFD mesh points in different manners, as shown in Tab. 4.4.

Table 4.4: Comparison of FSI algorithm displacement estimation

Algorithm	Displacement calculation
$CSS1$	$x_{n+1} = u(n)$
$CSS2$	$x_{n+1} = u(n) + \Delta t\ v(n)$
$CSS3$	$x_{n+1} = u(n) + \Delta t(1.5v(n) - 0.5v(n-1))$
$CSS4$	$x_{n+1} = u(n) + \frac{\Delta t}{2} v(n)$

Excluding the Conventional Serial Staggered Algorithm, all other methods use the structure's velocity to improve the computation of the new mesh points. As seen in Sec. 4.2.3, the method of obtaining both the structural velocities and accelerations is not very accurate for the selected Newmark - β integration parameters. To study further the effects of this accuracy loss on flutter speed computation, a comparison was made between the four FSI algorithms, using the test wing shown in Tab. 4.5.

The results are shown in Tab. 4.6.

From an analysis standpoint, the obtained values between $CSS1$ and $CSS3$ are quite similar, as for $CSS2$ and $CSS4$, the registered value was measurably higher than the previous algorithms. For previous aeroelastic analysis performed by Almeida [21], the $CSS3$ algorithm was deemed the optimal algorithm [21]. However, for the tested wing, it was verified that $CSS1$ displayed the best aeroelastic behaviour

Table 4.5: Initial wing geometrical and physical properties

Geometric Properties	
Airfoil	NACA 0015
Half-span	$0.75\ m$
Root chord	$0.25\ m$
Twist	$0°$
Taper ratio	1
Sweep angle	$0°$
Dihedral angle	$0°$
Material Properties	
Young's modulus(E)	$23.92\ MPa$
Shear modulus(G)	$9.14\ MPa$
Poison ratio	0.2018
Density(ρ)	$31.453\ kg/m^3$

Table 4.6: Comparison of FSI algorithm predicted flutter speed

Algorithm	Predicted flutter speed
$CSS1$	$16.66\ m/s$
$CSS2$	$17.35\ m/s$
$CSS3$	$16.25\ m/s$
$CSS4$	$18.14\ m/s$

transition from a non flutter condition to a flutter condition and, as such, its results can be more reliable to determine the flutter speed.

Since the difference in flutter speed between both the $CSS1$ and $CSS3$ is small comparing with the other algorithms and as the program is stated to underestimate aeroelastic divergent behaviour due to the overestimation of aerodynamic forces, the chosen algorithm for all numerical computations made using the aeroelastic tool will be the $CSS1$ algorithm.

4.5 Framework Architecture

As seen in Section 2.3, the typical aeroelastic analysis tool is made from the coupling of at least two different modules, and the original framework developed by Almeida [21] is no exception. While this original version does have both aerodynamic and a structural modules, it cannot be defined as a modular software, as these two modules are not clearly separated from the program's mainframe, giving it a monolithic structure.

Monolithic programs are simpler and more straightforward to write, and the code can be seen as one large block [49], as seen on the schematic shown in Fig. 4.15.

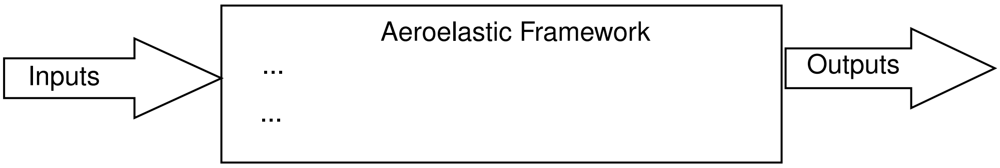


Figure 4.15: typical Monolithic code structure

Monolithic frameworks are normally used as experimental or one-off programs, where its structure is secondary to the task. For our particular case, since further developments of the framework were required, the monolithic structure makes any improvements harder or impossible to implement without major re-writes of the program [50].

Modular frameworks can be seen as a series of building blocks which are connected by the framework [49], which means they can be implemented in phases. While the initial implementation of a modular framework is harder to implement, it also has some advantages, namely:

- ease of use: it makes it easier to debug results and to understand the program, as it can be easily divided;
- reusability: it allows for modules to be exchanged or added without requiring large changes to the code structure;
- ease of maintenance: reduces time required to check all connections between modules and to use each module separately.

As such, a new version of the framework was implemented with a modular approach, using the same theoretical bases and also making sure that there are no differences between program outputs for both versions, so that all previous verification work is still valid. Besides changing the code structure, improvements were made to the solvers themselves, in an attempt to improve readability and potentially reduce computing time.

By modularizing the program, it can be now truly divided into the following five modules:

- Steady aerodynamic module: starts the aerodynamic computations required, defines initial aerodynamic mesh;
- Unsteady aerodynamic module: performs the aerodynamic computations for any $t > 0$;
- Structural module: computes mass and stiffness matrices,defines structural mesh and also calculates forces for each structural mesh node;
- Newmark time integration module: performs the structural time integration from time step t_n to t_{n+1};
- Fluid-Structure Interaction module: couples the structural and aerodynamic meshes, advances the aerodynamic mesh from t_n to t_{n+1}.

The new modular structure can be easily seen as a schematic in Fig. 4.16.

4.6 Code Improvements and Benchmark

After the changes were made to program structure, a comparison of the time elapsed for each aeroelastic computation for the new version versus the aeroelastic framework from Almeida [21] was performed, using a computer with an Intel ®Core$^{\text{TM}}$i7-2630QM, with 8.00 Gb of RAM memory. As the time measurement depends on either the computer used was performing other tasks that the user cannot control

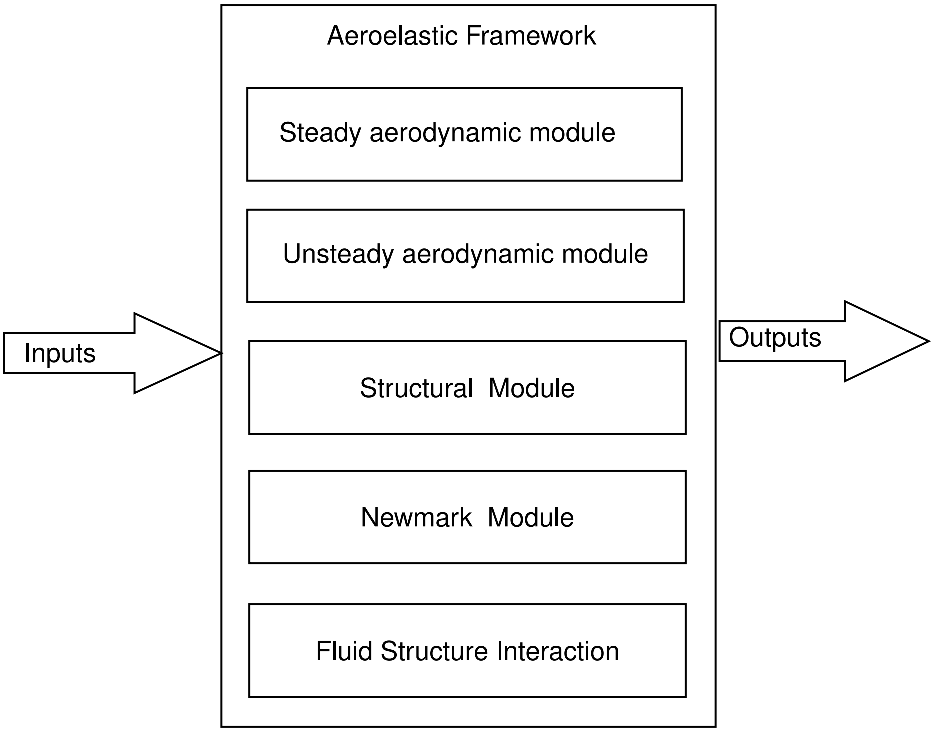


Figure 4.16: Typical modular code structure

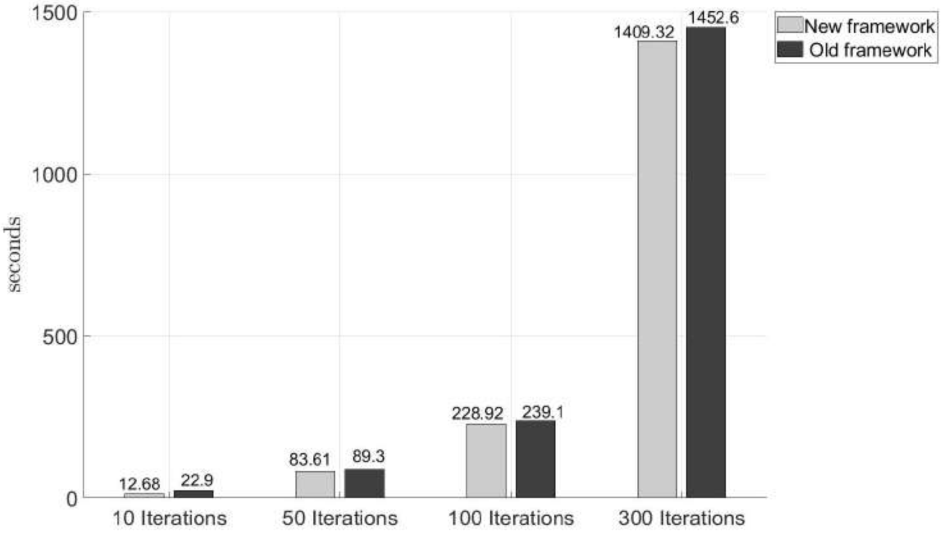


Figure 4.17: Benchmark between code versions

or terminate, the results shown in Fig. 4.17 are average values of 5 measurements: As it can be seen, the changes between the old and the new version show diminishing returns with the increase of the iteration number, as there is a noticeable difference for the lower iteration number that reduces with the increase in iteration number. This is mainly due to two factors:

1. The architecture of the unsteady aerodynamic model was not changed between both versions and, as such, the number of wake panels grows larger with each iteration, causing an increase in computing time for each additional wake panel [25];

2. The steady aerodynamic computation that starts the program is the main source of computational improvement but, as the number of iterations increases, this improvement is diluted in the total computing time.

Another analysis was made for the computing time for each module, using the new version, for the 300 iteration case. The results are shown in Tab. 4.7.

Table 4.7: Computational time per module

Module	Time (s)
Fluid solver	1403.64
Structural solver	3.30
Fluid Structure Interaction	1.45
Other sources	0.93
Total	1409.32

Unsurprisingly, most of the computing time is spent on the fluid solver module, constituting up to 99.6% of the total computing time. This is mostly due to the aerodynamic influence calculation routine, as each panel must be compared to every other panel in the wing for each time iteration, resulting in the large computing time.

Another important conclusion is that the time gain from using additional time steps in the fluid solver instead of defining the same time step for all solvers is negligible, and so, for all further analysis, the same time step is used for both solvers.

The other sources shown in the table are defines and small computations necessary to all modules and are not included in their time measurement, while the structural solver also includes the Newmark time integration module.

Chapter 5

Numerical results

Before the main aeroelastic analysis was started, a modal analysis was performed, using functionalities present in the Aeroelastic Framework developed using the methodology inscribed in Sec. 4.2.1. As previously stated, this procedure was required to define the correct time step for the analysis. The first 8 natural frequencies are shown in Tab. 5.1.

Table 5.1: Modes and frequencies for the tested wing

Mode	Frequency(Hz)
1^{st} flapwise bending	7.856
2^{nd} flapwise bending	48.446
1^{st} torsion	58.889
3^{rd} flapwise bending	132.19
2^{nd} torsion	176.75
1^{st} chordwise bending	244.16
4^{th} flapwise bending	248.03
5^{th} flapwise bending	291.37

With the estimation of the natural frequencies of the wing structure and considering that, due to program constraints, time step values lower than $0.005\ s$ are not feasible to use, the time step chosen is the lowest value possible. By selecting this value of time step, from the Nyquist-Shannon sampling theorem defined in Sec. 4.2.1, frequencies up to $f_{max} = \frac{1}{2t_s}$ can be correctly sampled, leading to

$$f = \frac{1}{2 \times 0.005} = 100\ Hz\ . \tag{5.1}$$

This value allows the capture of both flapwise bending and torsion modes, which were shown be the major components in achieving divergent behaviour, from previous testing.

5.1 Convergence Studies

As with all numerical simulations, a convergence study is required before actual results are obtained. For that, two main parameterswere used: number of spanwise and chordwise points. In the convergence study of thr number of spanwise and chordwise points, a wing was defined with parameters shown in

Tab.5.2.

Table 5.2: Baseline numeric test wing parameters

Fluid and Structural Solver Options	
Time step	$0.005\,s$
Total time	$1.5\,s$
Fluid structure interaction algorithm	$CSS1$
Structural subiterations	0
Wing Geometric Properties	
Root chord	$0.25\,m$
Half span	$0.75\,m$
Taper ratio	1
Sweep angle	$0°$
Dihedral angle	$0°$
Angle of attack	$4°$
Material Properties	
Young's modulus (E)	$23.92\,MPa$
Shear modulus (G)	$9.14\,MPa$
Material density	$31.453\,kg/m^3$
Flight Condition	
Freestream velocity	$10.00\,m/s$
Altitude	$0\,m$
Air density	$1.225\,kg/m^3$

Since the numeric case studied is an aeroelastic behaviour analysis, the main focus is on the computation of the aerodynamic forces, specifically vertical lift, as this is the primary source of wing loading. To select the best mesh for the aeroelastic analysis, a comparison of lift,moment and drag coefficients is performed while also checking the wing tip displacement using four different meshes, while also showing the computational time for each mesh.

Each mesh is defined by the number of chordwise points nc versus the number of spanwise points ns . While the number of chordwise points affects mainly the aerodynamic component, the spanwise points also affect the structural module, and, as such, should not be lower than 10 points. The results are shown in Tab. 5.3.

Table 5.3: Convergence test results

$nc \times ns$	$20 \times 10\ Mesh$	$40 \times 20\ Mesh$	$64 \times 30\ Mesh$	$100 \times 40\ Mesh$
C_L	0.2947	0.3041	0.3075	0.3092
C_M	-0.0720	-0.0723	-0.0731	-0.0735
C_D	0.0101	0.0060	0.0044	0.0032
Wing tip displacement	$0.00113\,m$	$0.00115\,m$	$0.00118\,m$	$0.00122\,m$
Computing time	$0.3020\,s$	$1.2930\,s$	$6.3170\,s$	$26.4830\,s$

By checking the aerodynamic coefficients, it is clear that there is a low variation of the lift and moment coefficients between meshes, which allows for the usage of a coarser mesh without compromising the accuracy of the results. However, the coarser mesh grossly overestimates the induced drag and, as such, it is excluded from the analysis.

The wing tip displacement does not exhibit a significant change with mesh sizes but it increases with the increase in element number. This is expected as finite element models overestimate the systems' rigidity, and by increasing the element number this estimation tends to the theoretical behaviour [39].

Another important value is the computational time, as the value shown is for only one aerodynamic iteration, and each numeric aeroelastic test performed is expected to require 300 iterations per flow velocity, it is important to minimize the computing time. For this reason, the finer mesh of 100×40 points is excluded from the analysis, as the gain in accuracy is minimal compared to the increase in computing time. Between the remaining two meshes, there is a slight increase of the lift coefficient, but it is not significant enough to compensate for the increase in computing time. As such, the numerical test will be performed using the 40×20 mesh, which was also previously used in the numeric case study shown in Almeida [21].

5.2 Verification Studies

Since the aerodynamic module is comprised of both a static aerodynamic solver and a unsteady aerodynamic solver that are of similar structure, a verification of the static aerodynamic solver is performed to check the evaluation of the aerodynamic forces, as they constitute the main output of the aerodynamic module, using the open-source software XFLR-5 [51]. Although previous verification work was performed by Almeida [21], the wing used in the numerical and experimental tests is different from the original verification test wing. Therefore a new verification test is required to eliminate any errors due to the different wing type implemented and tested.

5.2.1 Static Aerodynamic Model

To perform this verification, two identical wings are defined, one in the aerodynamic framework developed and another in the open-source software XFLR-5 [51]. This software was chosen due to user familiarity and the fact that it is one of the few available programs that also have a potential flow panel method implemented, which allows for direct comparison between two nearly identical solvers.

The wing geometric and computational parameters are summarized in Tab. 5.4.

Table 5.4: Test wing for XFLR-5 comparison

Parameter	Value
Airfoil	NACA 0015
Span	$1.5\ m$
Chord	$0.25\ m$
Taper Ratio	1
Sweep	0°
Dihedral	0°
Twist	0°
Angle of Attack	4°
Mesh Type	Uniform
Number of chordwise points	100
Number of spanwise points	40
Number of Panels	4000

The wing computational mesh is shown in Fig. 5.1. The wing dimensions are not arbitrary, they are also the dimensions of the test case wing for the aeroelastic experimental and numerical study. The

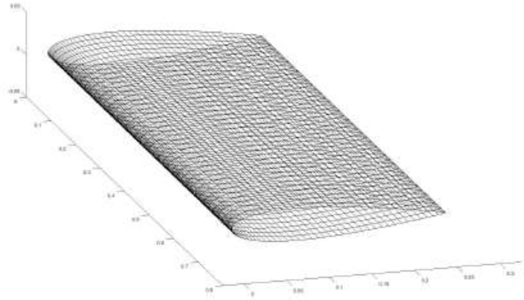


Figure 5.1: Computational mesh for the aerodynamic verification.

reasoning for the actual dimensions is further explained in Sec. 3.1. As for the mesh parameters, the developed solver has the possibility of uniform grid or non-uniform grid, but since the verification software offers the possibility of using an uniform grid, this was the selected option as it simplifies the meshing methodology and was verified that the increase in accuracy was minimal.The number of panels was set by the developed framework, as it requires a file with airfoil coordinates and, for the selected airfoil, the largest number of points available is 100 in the chordwise direction. The 40 spanwise points were selected due to the structural component of the framework, as it was verified that for the aerodynamic solver, any number of spanwise points greater than 10 does not produce a verifiable increase in result accuracy that justifies the increase in computing time. The results obtained are shown in Tab. 5.5.

Table 5.5: Comparison of aerodynamic coefficients with between the framework used and XFLR-5.

	Aeroelastic Framework	XFLR-5	Difference
C_L	0.3092	0.3137	1.4%
C_D	0.0032	0.00517	37.3%
C_M	-0.07353	-0.07506	1.4%

Analysing the obtained values, it is clear that the lift coefficient C_L presents a nearly null difference between both softwares, which implies that the vertical forces are being correctly computed.

As for the drag coefficient C_D, there are some disparities in the obtained values, with the developed aerodynamic module underestimating the drag component by almost 38%, which indicates that the drag computation is not very reliable. Since the aerodynamic module goal is to determine aerodynamic forces for the aeroelastic framework, and the horizontal aerodynamic forces are not impactful on the aeroelastic response of a wing in a normal flight condition, this value disparity is ignored for its lack of relevance. Also worth noting that both solvers are operating under inviscid conditions, the drag value obtained is only the induced drag component, which, for a panel method solver can fluctuate depending on the method used for wake shape estimation [25]. Since the source code of the XFLR-5 software is not easily accessible, it cannot be assured that both solvers are using the same wake shape estimation method.

The pitching moment coefficient C_M has a similar value deviation to the lift coefficient, which is expected as the main force behind the moment computation is the lift. Due to the low differences for both lift and pitching moment, which are the main parameters behind the expected aeroelastic behaviour of the wing, it can be assumed that the static values are correctly computed by the developed aerodynamic solver.

5.2.2 Static Structural Model

To perform this verification, two identical wings are defined, one in the static structural module of the framework, and another on ANSYS® Workbench software. In both cases, the wing is discretized with a beam finite element with cross section properties of the airfoil, while the material properties are that of the polystyrene block used in experimental testing. As for the computational mesh, both wings have a mesh size of 20 spanwise elements, as shown in Fig. 5.2.

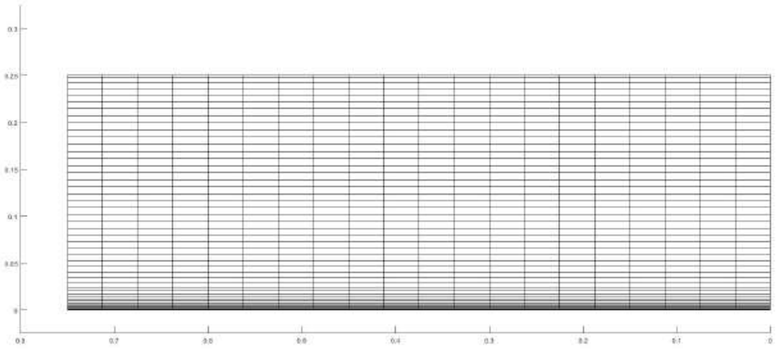


Figure 5.2: Computational mesh for the structural verification.

The verification consists in applying a vertical force of $5\ N$ on the wing tip and considering the wing root as a fixed support. With this test procedure, the maximum wing tip displacement is computed. The test wing parameters are shown in Tab. 5.6, while the results are shown in Tab. 5.7.

Table 5.6: Test Wing for ANSYS® Workbench comparison

Parameter	Value
Airfoil	NACA 0015
Number of spanwise elements	20
Span	$1.5\ m$
Chord	$0.25\ m$
Taper Ratio	1
Sweep	$0°$
Dihedral	$0°$
Twist	$0°$
Young modulus	$23.92\ MPa$
Shear modulus	$9.14\ MPa$

Table 5.7: Maximum wing tip displacement comparison

	Aeroelastic Framework	ANSYS® Workbench
Maximum wing tip displacement	$0.0123\ m$	$0.0139\ m$

The wing tip displacement displays a difference of 11.5% between the Aeroelastic Framework and

ANSYS® Workbench. While this difference is not negligible, due to the lower value obtained by the Aeroelastic Framework, this can be considered as an additional safety factor for the developed model.

5.3 Flutter Speed Calculation

Since most structural vibration phenomena can be characterized as a damped harmonic movement, the damping ratio g was estimated to find the flutter speed, as the transition of the damping from positive values to negative values yields the Flutter speed [26] . The damping ratio can be obtained from a quantity known as the logarithmic increment δ_n [40], defined as

$$\delta_n = \frac{1}{n} \ln \frac{X_i}{X_{i+n}} = \frac{2\pi g}{\sqrt{1-g^2}} \quad , \tag{5.2}$$

where X_i is the peak value at peak i, X_{i+n} is the peak value at peak $i+n$ and n is the number of peaks between X_i and X_{i+n}.

The damping ratio was computed for different airspeeds, and the results are shown in Fig. 5.3. At the same time, a Fast-Fourier transform (FFT) is performed on the obtained wing tip displacement behaviour, to check the frequency evolution with speed, also shown in Fig. 5.3.

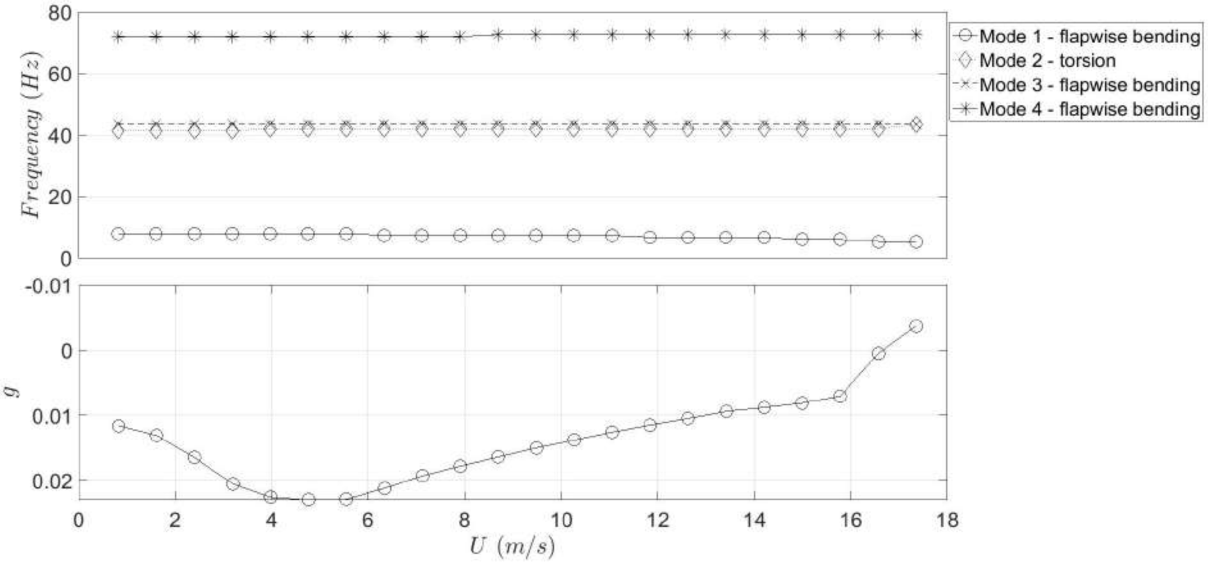


Figure 5.3: f-U and g-U graphs for the baseline numerical case

As previously stated, the flutter point is where the damping ratio transitions from a negative value to a positive one and, for the tested wing, this point occurs at $U = 16.66\ m/s$. This is considered the primary method to find the flutter speed. By analysing the frequency spectra, an approximate estimation can also be found by checking when two separate frequencies coalesce into a single value, as shown in the previous figure, where for a speed of $17.35\ m/s$, modes 2 and 3 have the same frequency value, implying that the structure is experiencing divergent behaviour.

The evolution of the wing tip displacement clearly shows the evolution of the wing's behaviour when transitioning from a non-flutter condition to a flutter condition and, as such, it is the preferred variable to compute the damping ratios.

To further illustrate the time variation of the wing tip displacement, two cases are shown, one illustrative of a pre-flutter condition and another of a clear flutter behaviour. In Fig. 5.4, the wing is in a pre-flutter condition, where the vibration amplitude decreases with time, and the points considered for the calculation of the damping ratio are identified. The dashed line represents the average wing tip displacement for the time duration considered, with a value of $0.0025\ m$.

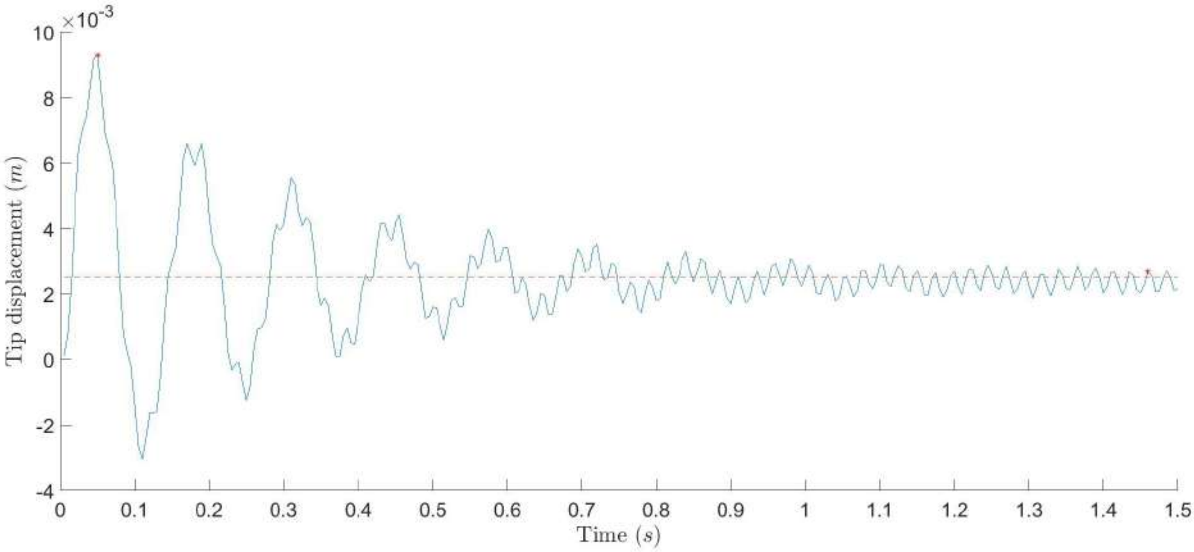


Figure 5.4: Wing tip displacement time variation for $U = 7.9104\ m/s$

As for Fig. 5.5, there is a clear increase of the wing tip displacement with time that is expected of a flutter condition and the points selected for the damping ratio are also shown. The dashed line also represents the average wing tip displacement for the time duration considered, with a value of $0.0417\ m$.

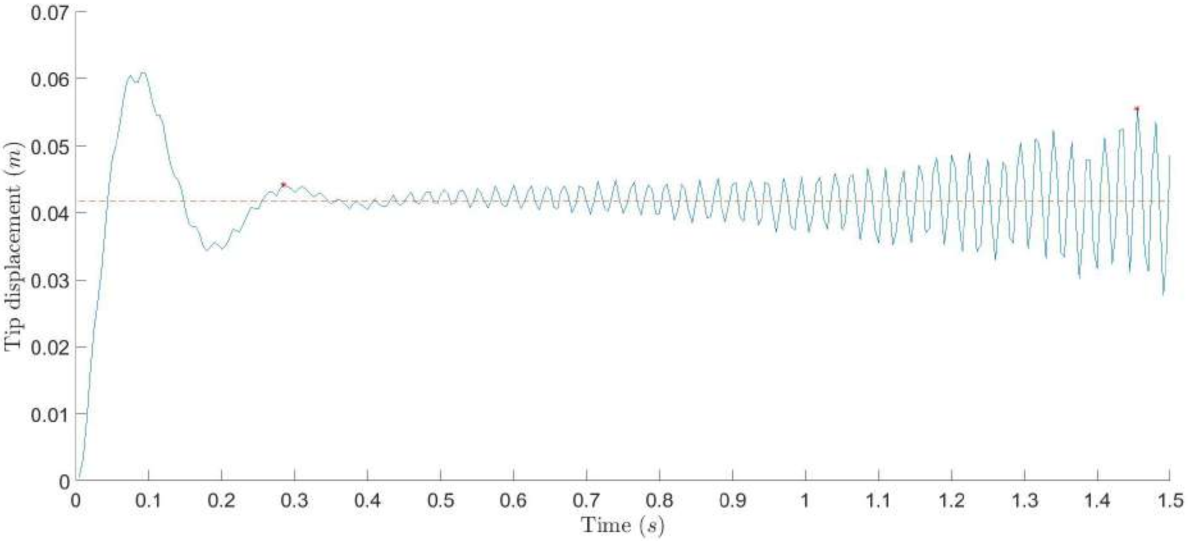


Figure 5.5: Wing tip displacement time variation for $U = 17.351\ m/s$

Although the time span shown is short, it was verified that it is the minimum time span required to observe the divergence of the wing tip displacement. As for the selection of this particular speed, it was the maximum value at which results where obtainable, as for greater values the solution diverges and the program is forced to terminate in order to not crash.

Besides the increase in vibration amplitude verified between the two illustrated cases, the mean tip displacement also increases. This is expected, as for a higher airspeed the lift produced by the wing will also increase, leading to higher stress exerted on the wing, which results in an higher static deformation.

5.4 Flutter Speed Index Comparison

The Flutter Speed Index [9] is defined as

$$V_f = \frac{U_\infty}{s\omega_a\sqrt{m_r}} \, , \tag{5.3}$$

where U_∞ is the flow velocity, s is the wing span, ω_a is the first torsional mode frequency and m_r is the mass ratio of the wing [9]. The definition of the mass ratio of the wing comes from stability theory [52],

$$m_r = \frac{m}{\frac{1}{2}\rho_{air} S\bar{c}} \, , \tag{5.4}$$

where m is the wing mass, ρ_{air} is the air density, S the aerodynamic wing area and $\bar{c}$ the mean chord of the wing.

A comparison between the flutter speed index obtained for the numeric analysis and the experimental test is shown in Fig. 5.6.

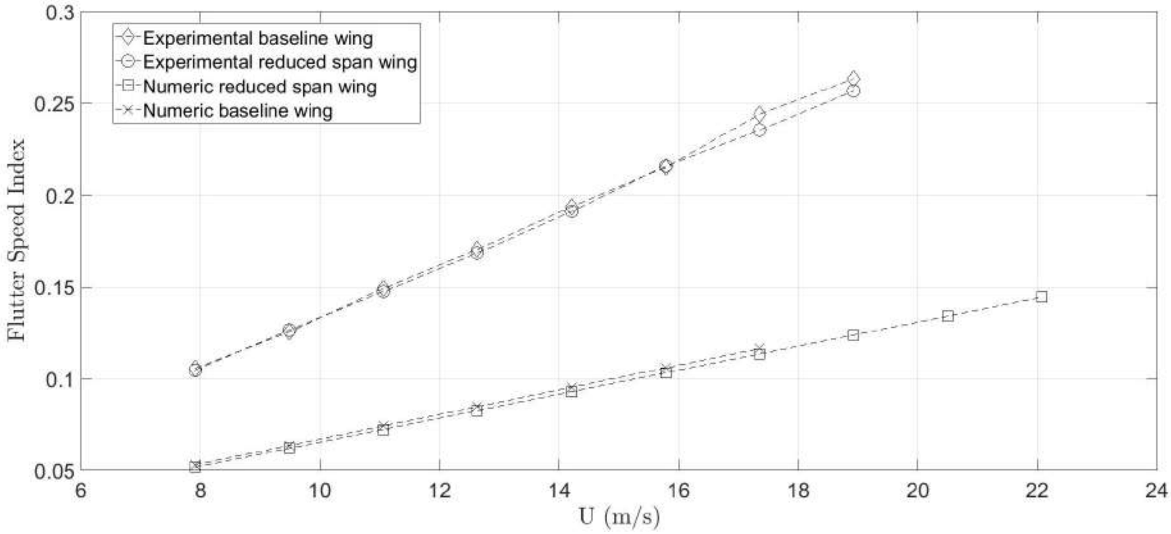


Figure 5.6: Flutter speed index variation with speed

For both the experimental and the numerical cases, the flutter speed index remain close between the two different tested wings as, while they have dissimilar spans and torsional behaviour, the difference is not pronounced enough to make a large difference between curves.

The main factor is the difference between the values of the experimental and the numeric tests. This is uniquely attributed to the difference in the first torsional mode observed as all other parameters are equal.

Also worth noting, for the numerical case, no values of the flutter speed index are computed on the baseline wing for a speed greater than $17.3514\ m/s$ due to the presence of highly divergent behaviour of the wing, consistent with the expected post-flutter behaviour.

Comparing the results from the baseline experimental wing with those by the numerical analysis, there are some disparities that can be explained by:

- Overestimation of aerodynamic forces, as stated in Sec. 4.1.1;
- Inexistence of damping in the numeric model;
- Errors in the experimental estimation of the materials elastic properties;
- Parasite vibrations of the experimental wing mount model, that contribute to the damping of the wing natural vibrations.

5.5 Aspect ratio parametrization

As the experimental testing showed, there is a substantial change in the wing's aeroelastic behaviour with the decrease in aspect ratio, mainly due to the increase in wing rigidity and corresponding increase in modal frequency values.

To further study the aeroelastic behaviour, a parametric analysis of the flutter speed variation with wing aspect ratio was performed using the numerical model developed. To perform this parametrization, the wing chord was frozen at $0.25\ m$ and the span was changed for each case. The variation of the flutter speed with the wing's aspect ratio is shown in Fig. 5.7.

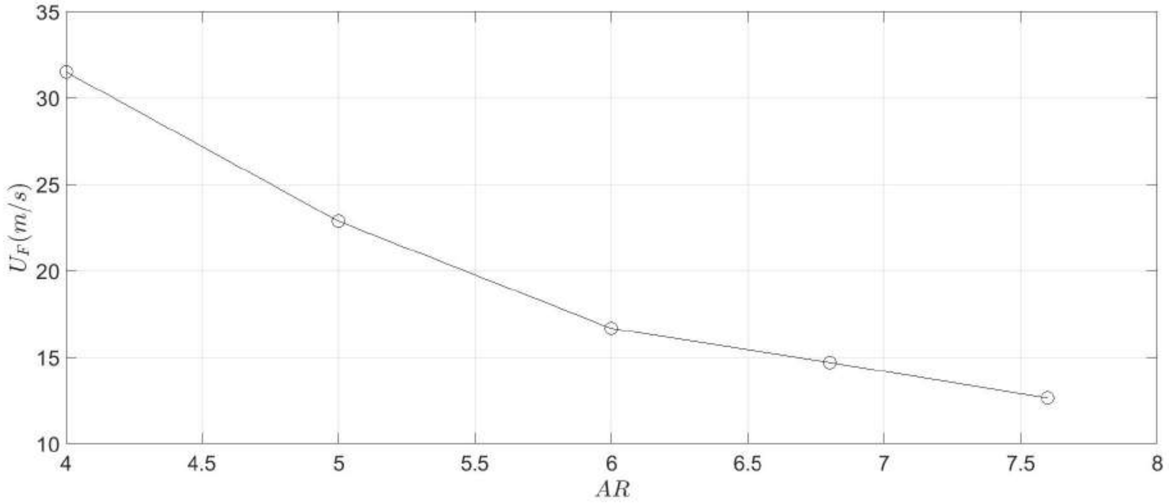


Figure 5.7: Flutter speed variation with aspect ratio

As expected, there is an increase of the flutter speed with the decrease of the wing aspect ratio, effectively doubling the expected flutter speed between aspect ratio values of 4 and 6. The evolution for values greater than 6 is lower and for aspect ratio values greater than 8 no correct results can be computed due to the presence of non-linear behaviour not computed by the numerical model used. As stated, the increase of flutter speed by decreasing the aspect ratio is mainly due to the increase of the wing's rigidity.

Chapter 6

Numerical Optimization

Most aerospace optimization problems are of multidisciplinary nature, as the physics involved per discipline are coupled to one another. As with all optimization problems, the objective is to maximize or minimize a desired objective function with respect to prevailing constraints. An objective function is the value or values that enables a comparison between two designs, while the design variables are the design parameters that are prone to being changed in the optimization process, and are subject to bounds and constraints. In a pure mathematical form, an optimization problem can be seen as a minimization or maximization of a function subjected to constraints, as

$$\begin{aligned} \text{minimize} \quad & f(\boldsymbol{x}) \\ w.r.t. \quad & \boldsymbol{x} \in N_x \,, \\ \text{subject to} \quad & h_p(\boldsymbol{x}) = 0, p = 1, 2, \dots, N_h \\ & g_m(\boldsymbol{x}) \geq 0, m = 1, 2, \dots, N_g \,, \end{aligned} \tag{6.1}$$

where

- f : objective function (output)
- $\boldsymbol{x}$: vector of design variables (input) bounds can be set on those variables;
- $\boldsymbol{h}$: vector of equality constraints; in general these are non-linear functions of the design variables;
- $\boldsymbol{g}$: vector of inequality constraints: may also be non-linear and implicit;
- N_x: bounds of the design variables;
- N_h: total number of equality constraints;
- N_g: total number of inequality constraints;

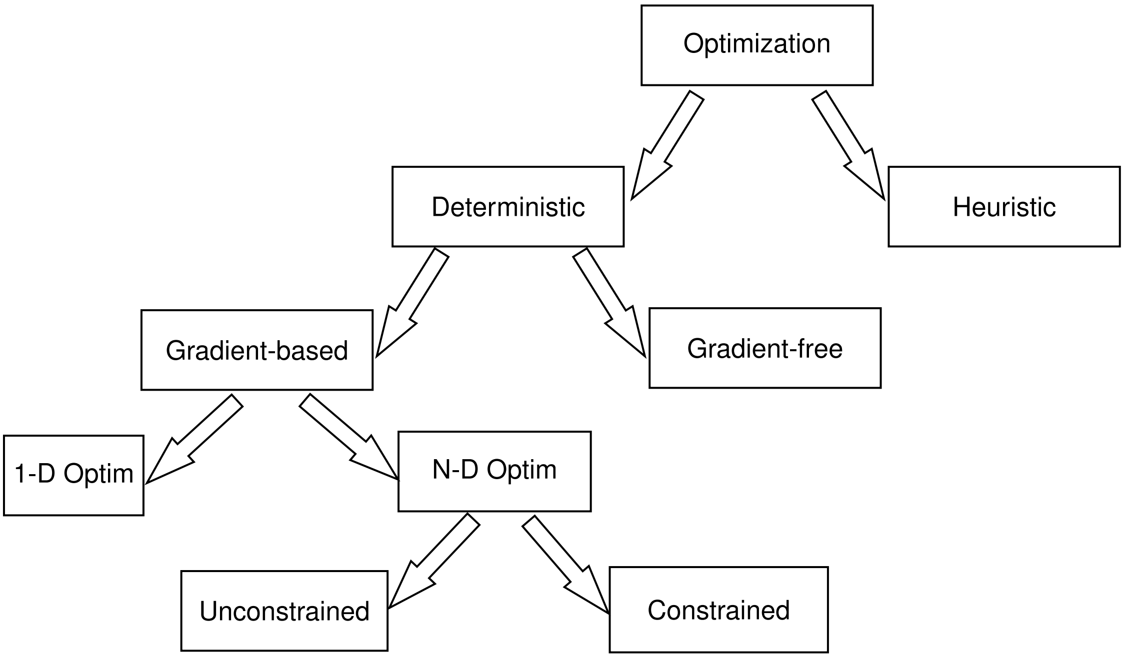


Figure 6.1: Overview of optimization methods

6.1 Overview of Optimization Methods

Although optimization methods can be classified into various categories, as seen in Fig. 6.1, only deterministic methods are considered due to the scope of the work developed.

Deterministic problems can be either gradient-based or gradient-free. Gradient free methods are usually implemented for problems where the objective function is noisy or is discontinuous, while gradient based optimization methods constitute the preferred method for finding local minima for large dimensionality, non-linearity, convex search space problems.

Within the gradient-based optimization methods, further separation can be made concerning the number of dimensions the problem has. While most physically relevant problems are not one-dimensional, by performing a line search in which the optimization algorithm finds the best path to perform an optimization analysis, the problem is reduced to a one-dimensional type, which reduces the computational effort considerably.

Finally, further divisions can be made for N dimension optimization methods, concerning the existence of constraints that translate design requirements [53].

6.1.1 Constrained gradient-based optimization

Most aerospace problems involve constrained problems, as design often dictates minimum or maximum values for functions of interest, such as maximum wing drag, or maximum stress on a wing spar. Mathematically, a constrained problem is the typical optimization problem, as defined in Sec. 6.1, and its typical procedure is schematically shown in Fig. 6.2, consisting in a problem with a single objective and a vector of constraints, where separate components exist to compute the objective, the constraints and the gradients, assuming that they can be computed without knowledge of the objective and constraint function values.

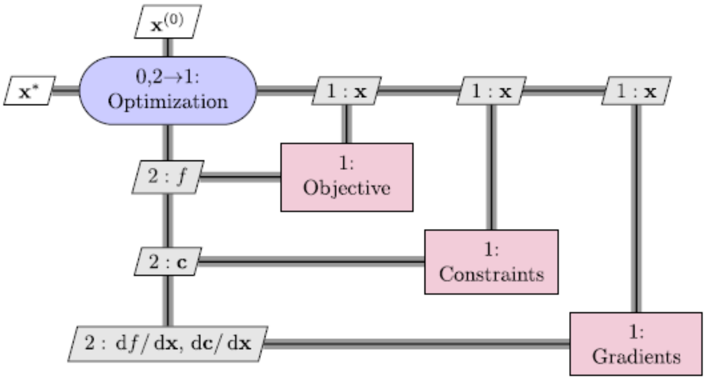


Figure 6.2: Constrained gradient-based optimization procedure [54]

6.1.2 Program Implementation

For the actual implementation of the optimization algorithm, the optimization toolbox for Mathworks' MATLAB® is used, specifically the $fmincon$ function [55] for the aeroelastic optimization test case and the $fgoalattain$ function [56] for the aerodynamic and structural optimization test cases. By using the MATLAB® optimization toolbox, any module of the developed aeroelastic framework can be set as an objective function, or even the entire framework, which reduces significantly the implementation time.

The $fmincon$ function allows for a selection of algorithm options, such as the interior-point algorithm, which is a constrained gradient based algorithm [57], the Sequential Quadratic Programming method (SQP) algorithm, that is also a constrained gradient based algorithm [53] or the active set algorithm, which is similar to the interior point algorithm but uses larger steps, which increases computing speed, but result accuracy is lower [53].

As for the $fgoalattain$ function, it has only one possible algorithm, and it is a SQP algorithm, while the formulation implemented is the goal attainment problem of Gembicki [58].

Another important component of a constrained gradient-based optimization procedure is the computation of the gradients which, for all optimization cases, defaulted to forward differences, as this is the default definition of MATLAB® optimization toolbox.

While the $fmincon$ function is more versatile and allows for greater control on algorithms, the $fgoalattain$ simplifies the introduction of constraints on the output of the objective function, which is important both for aerodynamic and structural optimization, as the aerodynamic coefficients are outputs of the objective function. Another advantage is the definition of weights for each output constraint, which enables greater control on what output variables are more important to the iterating procedure. As such, both the wing lift to drag ratio and the wing mass optimization problems use this function, while the flutter velocity optimization uses the $fmincon$ function.

6.2 Wing Lift to Drag Optimization

The problem pursued is a purely aerodynamic optimization problem, in which the L/D ratio is maximized with constraints in minimum lift and wing area.This is done to assure that both the original and the optimized wings produce equivalent amounts of lift, as would be required on a real aircraft design problem. The objective function in this case is an output of the static aerodynamic solver incorporated in the aeroelastic framework used on the numerical testing, and its goals and constraints are summarized in table 6.1.

Table 6.1: Static aerodynamic optimization goals and constraints

Goal	**Constraints**
Maximize L/D ratio	$S \geq 0.375\ m^2$
	$C_L \geq 0.3$
	$\alpha = 4^\circ$
	$U_\infty = 10\ m/s$
	$1.3 \leq s \leq 1.7\ m$
	$0.25\ m \leq c_{root} \leq 0.4\ m$
	$\Lambda = 0^\circ$
	$\Gamma = 0^\circ$
	$\lambda \geq 0.4$
	$-5^\circ \leq \theta_{root} \leq 5^\circ$
	$-5^\circ \leq \theta_{tip} \leq 5^\circ$

While the design variables vector $\boldsymbol{x}$ is defined by

$$x = \begin{bmatrix} c_{root} \\ \lambda \\ s \\ \theta_{root} \\ \theta_{tip} \end{bmatrix} \tag{6.2}$$

where c_{root} is the chord value at the wing root, λ is the taper ratio, Λ is the leading edge sweep angle, Γ is the dihedral angle, θ_{root} is the root twist angle and θ_{tip} is the wing tip twist angle.

As far as the chosen constraints are concerned, the value for the angle of attack is fixed due to this being the wing's designed cruise angle of attack. The dihedral angle Γ is fixed to 0° , as it is only important for dynamic stability analysis, which are out of scope for this test.

The sweep angle Λ is fixed at 0°, since it is only important for transonic wing design. For the taper ratio, the lower limit is the trapezoidal wing shape described in Sec.4.3, as it provides the best lift distribution and does not present a very large increase in the wing aspect ratio.

As for the wing span s, it must not be greater than $0.85\ m$ due to the very large aspect ratio it would have if this value was allowed to be greater, and root chord c_{root} cannot be smaller than its original value, as this would again increase the aspect ratio to very high values, which would compromise structural integrity.

Finally, both wing tip twist(θ_{tip}) and wing root twist (θ_{root}) are allowed to achieve values not greater than 5° due to the linear nature of the aerodynamic solver, as higher values would make induced angles of attack outside the linear regime.

Due to the usage of the $fgoalattain$ function on this optimization task, no alternatives are available for the algorithm selection and so the algorithm used is the SQP algorithm.

The evolution of the L/D ratio with each iteration is shown in Fig. 6.3. While the L/D ratio converges quickly, from iteration number 10 onwards the program is tuning the other goals for the optimization function so that all constraints are respected, leading to an increased iteration number. The first iterations also produce a higher value of the L/D ratio than the final result but, due to other constraints being violated, these results cannot be achieved once all constraints are being respected.

As for the stopping criteria, it is defined as the point when the magnitude of the search direction is less than the specified tolerance, defined as 1×10^{-06} ,and also no constraint is violated. The value assumed for the tolerance is the default value from MATLAB®, as using a larger tolerance resulted in worse final wing aerodynamic performance. To perform the optimization routine, the total number of function evaluation was 149, which is well below the maximum allowed value of 3000, also the default value set by MATLAB®.

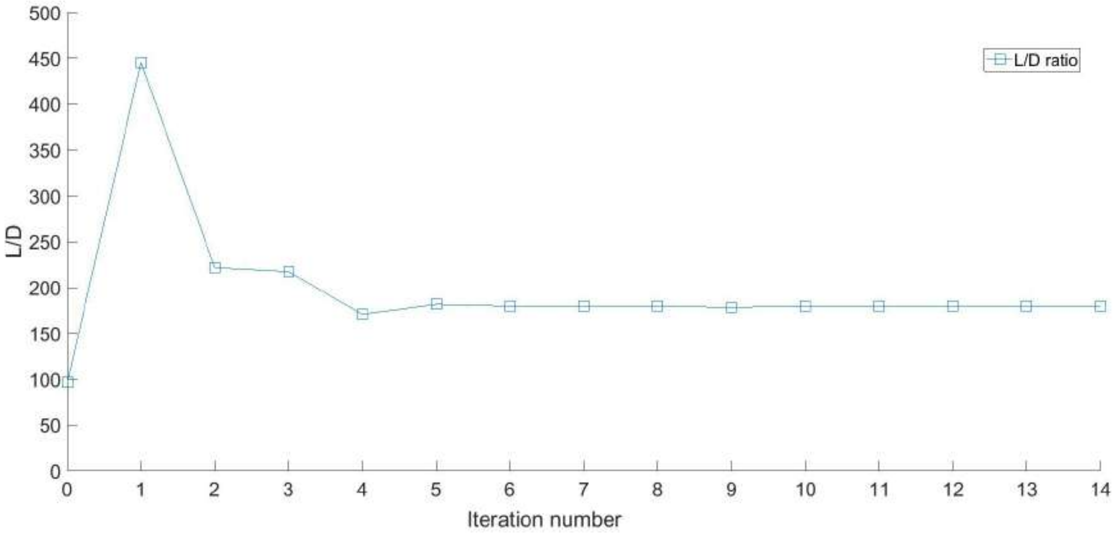


Figure 6.3: L/D ratio evolution with iteration number

The original and optimized wing parameters are shown in Tab.6.2.

Table 6.2: Wing lift to drag optimization geometrical properties

	Original Wing	Optimized wing
Airfoil	NACA 0015	NACA 0015
Number of chordwise points	100	100
Number of spanwise points	20	20
Half-span	$0.75\ m$	$0.85\ m$
Root chord	$0.25\ m$	$0.3180\ m$
Tapper ratio	1	0.4
Wing root twist	$0°$	$-0.9411°$
Wing tip twist	$0°$	$1.0769°$
Wing area	$0.375\ m^2$	$0.3784\ m^2$
Wing mass	$0.1510\ kg$	$0.1452\ kg$

Comparing both wings, it is verified that although there is a difference both in the span and root chord, the surface area remains almost equal, meaning that the optimizer respected the constraints imposed.

As for the aerodynamic parameters, its comparison is shown in Tab. 6.3:

Table 6.3: Aerodynamic parameters comparison

	C_L	C_D	C_M	L/D
Original wing	0.3097	0.0032	-0.0738	96.78
Optimized wing	0.3034	0.0017	-0.0790	180

The optimizer achieved every goal it was imposed, and while there is a slight decrease in lift coefficient, the constraint was respected. As for the drag coefficient value, it decreased as expected, due to a decrease in the wing taper ratio that improved the lift distribution while reducing the induced drag. The moment coefficient is not critical for the intended analysis, but it presents similar values for both wings and as such, the wing pitching behaviour remains the same. The main goal of the analysis, the increase of the L/D ratio was achieved, mostly due to the decrease of the induced drag. The final wing shape is shown in Fig. 6.4, with the original wing shape displayed as a dashed line.

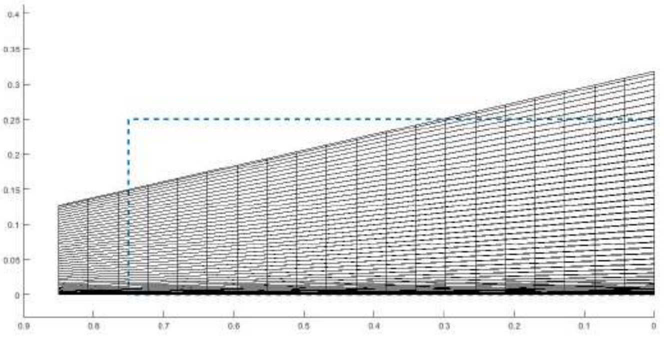


Figure 6.4: Optimized lift to drag wing discretization

6.3 Wing Mass Optimization

In this case, the main objective is the minimization of the wing mass, but also the aerodynamic properties must be respected. The constraints for this case are mostly equal to the static aerodynamic optimization problem 6.2 but an additional maximum shear stress constraint was considered to guarantee the wing structural integrity , as shown in Tab 6.4.

Due to the usage of the $fgoalattain$ function on this optimization task, no alternatives are available for the algorithm selection and so the algorithm used is the SQP algorithm.

The evolution of the wing mass with each iteration is shown in Fig. 6.5.

Comparing with the previous optimization case, the wing mass optimization converges quickly, despite using the stopping criteria as the wing lift to drag optimization case. This rapid convergence is explained due to the final wing being very similar in shape to the original wing, meaning that the optimizer required less function evaluations to achieve an optimized result, performing only 39 function

Table 6.4: Wing mass optimization goals and constraints

Goal	Constraints
Minimize Wing mass	$S \geq 0.375\ m^2$
	$C_L \geq 0.3$
	$\tau_{max} = 1828000 Pa$
	$\alpha = 4°$
	$U_\infty = 10\ m/s$
	$1.3 \leq b \leq 1.7\ m$
	$0.25\ m \leq c_{root} \leq 0.4\ m$
	$\Lambda = 0°$
	$\Gamma = 0°$
	$\lambda \geq 0.4$
	$-5°\ \leq \theta_{root} \leq 5°$
	$-5°\ \leq \theta_{tip} \leq 5°$

evaluations, comparing with the 149 from the previous case.

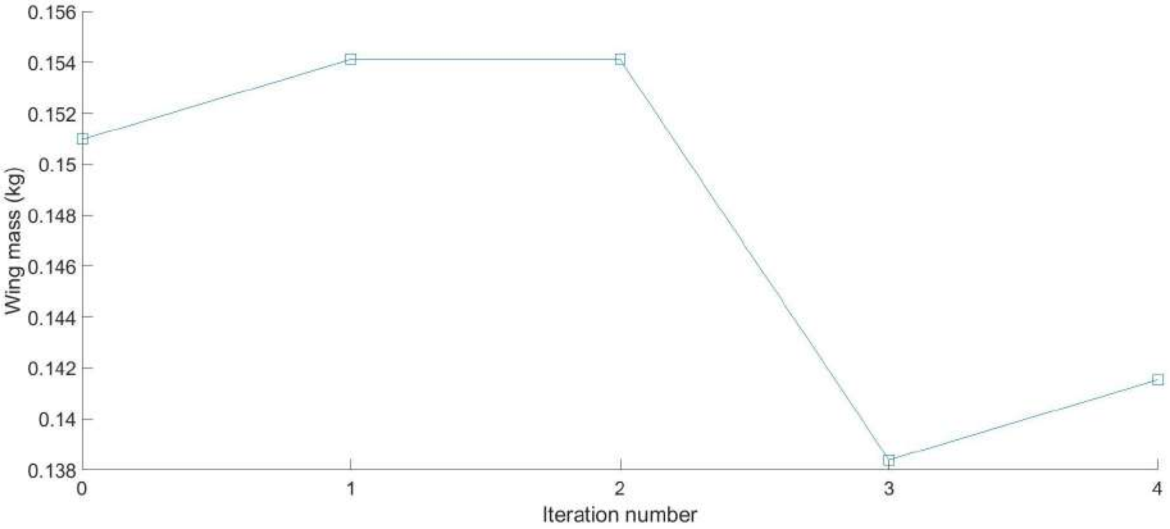


Figure 6.5: Wing mass evolution with iteration number

As for the maximum shear stress, this new constraint had a maximum value of $1828000\ Pa$, this value obtained using the offset yield point definition of $0.2G$ [59], as no publicly available data of the maximum shear stress was available for the selected material. The evolution of the wing maximum shear stress is shown in Fig. 6.6.

Although a large increase in the maximum shear stress is verified, the value is still well bellow the maximum allowed value and, as such, this constraint is not critical for the optimizer. The low values for the shear stress are explained by the low airspeed value and the low wing area, which lead to a low value of lift, the main vertical force that produces the shear stress.

After performing the optimization calculation, the obtained wing is shown in Tab. 6.5, as well as the original wing for comparison.

Analysing the results of the mass optimization problem we find that, as expected, the mass minimization problem achieved a smaller final mass and did not fail to respect any aerodynamic constraint. However, by comparing the results with the aerodynamic optimization wing we find that, although the

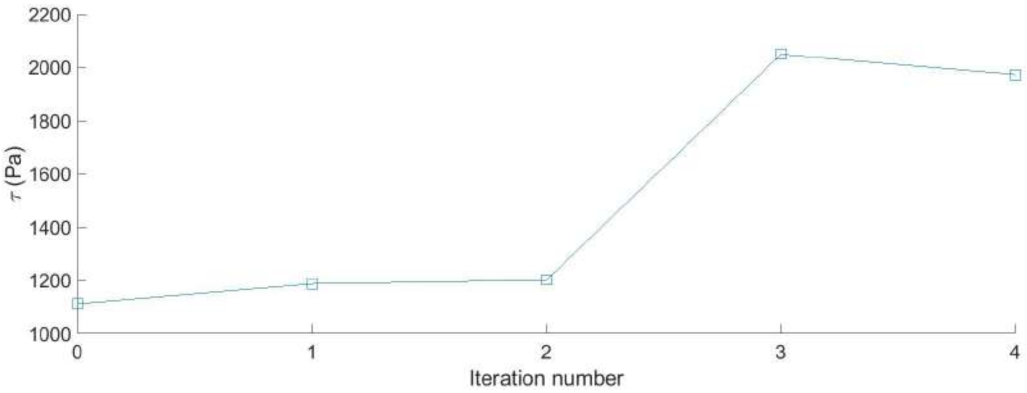


Figure 6.6: Shear stress evolution with iteration number

Table 6.5: Wing mass optimization geometrical properties

	Original Wing	Optimized Wing
Airfoil	NACA 0015	NACA 0015
Number of chordwise points	100	100
Number of spanwise points	20	20
Half-span	$0.75\ m$	$0.85\ m$
Root chord	$0.25\ m$	$0.25\ m$
Taper ratio	1	0.8
Wing root twist	$0°$	$0.101°$
Wing tip twist	$0°$	$0.0709°$
Wing Area	$0.375\ m^2$	$0.3826\ m^2$
τ_{max}	$1111.7\ Pa$	$1972.3\ Pa$
Wing Mass	$0.1510\ kg$	$0.1415\ kg$
C_L	0.31	0.35
L/D	96.78	129.22

final mass is still smaller, the difference is only of $0.0095\ kg$, and considering the difference in L/D between both analysis, which can translate into a greater payload/range, from a design perspective the preferred wing design is the wing from the aerodynamic optimization case. The optimized wing is shown in Fig. 6.7, with the original wing shape displayed as a dashed line.

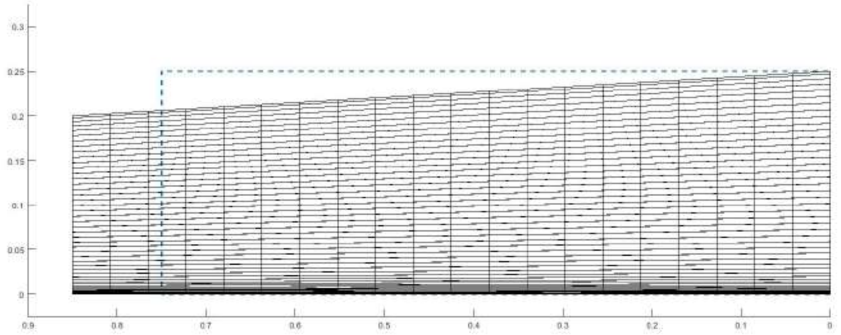


Figure 6.7: Optimized mass wing discretization

6.4 Wing Flutter Optimization

For this optimization problem, a function was defined to determine the speed for which the numeric solver achieves a numerical divergent solution. This divergent solution occurs for a speed greater than the flutter speed, as they are obtained with different methodologies. The flutter speed, as previously stated, is estimated with the logarithmic increment, while this numerical divergence speed is estimated by finding the point which produces wing tip deflections greater than $10b$. The value of the boundary was defined via the verification that, if tip displacement grows to such a large value, the program would crash before the last time iteration was computed.

This optimization uses the same mesh as numeric case study in Sec. 5.1, the 40×20 panels mesh, while the constraints remain mostly the same as stated in Sec. 6.2,excluding the speed constraint as it is not applicable in this case, as shown in Tab. 6.6.

Table 6.6: Wing flutter optimization goals and constraints

Goal	**Constraints**
Maximize U_F	$\alpha = 4^\circ$
	$C_L \geq 0.3$
	$1.3 \leq b \leq 1.7\ m$
	$0.25\ m \leq c_{root} \leq 0.4\ m$
	$\Lambda = 0^\circ$
	$\Gamma = 0^\circ$
	$\lambda \geq 0.4$
	$-5^\circ \ \leq \theta_{tip} \leq 5^\circ$

As for the optimization algorithm selection, a SQP algorithm was chosen, in part to maintain coherence with the previous optimization problems and also due to information existing documentation [20] that refers that it converges quicker than the $interior - point$ algorithm without the loss in accuracy of other algorithms, leading to a reduction in computing time. The stopping criteria remained the same as previous optimization problems, while the total number of function evaluations performed is 41.

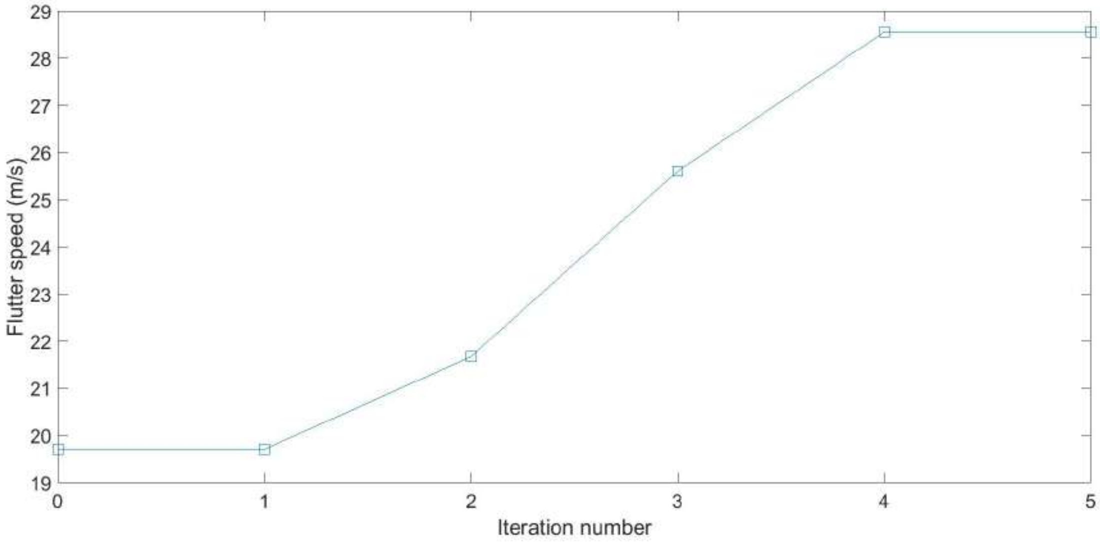


Figure 6.8: Wing flutter speed evolution with iteration number

The optimized wing parameters are shown in Tab. 6.7, as well as the original wing.

Table 6.7: Wing flutter speed optimization geometric properties

	Original Wing	Optimized Wing
Airfoil	NACA 0015	NACA 0015
Number of chordwise points	40	40
Number of spanwise points	20	20
Half-span	$0.75\ m$	$0.85\ m$
Root chord	$0.25\ m$	$0.4\ m$
Taper ratio	1	0.5848
Wing tip twist	$0°$	$5°$
Wing area	$0.375\ m^2$	$0.5388\ m^2$
Wing mass	$0.1510\ kg$	$0.2875\ kg$
Flutter speed	$16.66\ m/s$	$28.56\ m/s$
C_L	0.31	0.46

The optimized wing achieved a large increase in the flutter speed compared to the original wing and a greater base C_L at the expense of a large increase in wing mass, in part due to the increase in both root chord and wing tip twist. A large wing tip twist is usually not advisable due to the presence of control surfaces near the wing tip but, as this is a proof of concept analysis, this factor was disregarded. The large increase in mass was expected due to the necessity of increasing the wing structural rigidity to enable the maximization in flutter velocity. Overall, the analysis was successful, as it produced a wing with greater flutter speed while not compromising the aerodynamic properties of said wing.

The optimized wing is shown in Fig. 6.9, with the original wing shape displayed as a dashed line.

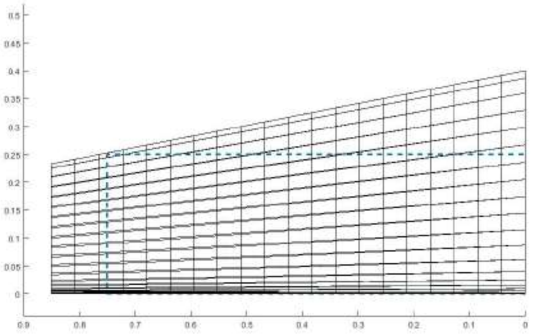


Figure 6.9: Optimized flutter speed wing discretization

6.5 Summary of Computational Cost

After performing the three optimization problems, a study was performed on the computational time spent by each optimization. Due to large fluctuations in total computing time and due the unfeasibility of performing several runs of the same optimization due to the large number of function evaluations it was chosen not to compare directly the absolute values of computing time. Instead, each optimization computing time is adimensionalized by the respective computing time of the used framework module:

- wing L/D optimization —only the steady aerodynamic module is considered, with a total computing time of $t = 6.7070\ s$;
- wing mass optimization — both the steady aerodynamic module and the structural module are considered, with a total computing time of $t = 7.0640\ s$;
- wing flutter speed optimization — the entire aeroelastic framework is used, with a total computing time of $t = 1409.32\ s$.

The results of the adimensionalization are shown in Tab. 6.8, in conjunction with the total number of function evaluations.

Table 6.8: Adimensionalized computing time for optimization problems

Optimization type	Adimensionalized time	Total function evaluations
Wing L/D optimization	140.2	149
Wing mass optimization	36.7	39
Wing flutter speed optimization	26.4	41

By adimensionalizing the total computing time with the computing time of each module, it is expected that this value should be similar to the total number of function evaluations, as they constitute most of the computing effort. This is verified on Tab. 6.8, as all results are within the same range as the total number of function evaluations, while also being lower in value. Also expected is the larger magnitude of the wing L/D optimization value, as for this specific case a greater number of both iterations and function evaluations are performed, leading to an higher computing time.

Another factor is the lower value of the wing flutter speed optimization adimensional time comparing with the total function evaluations, which is explained due to larger fluctuations in computing time for each function evaluation on the optimizer, as the aeroelastic framework is, by far, the most expensive computational module utilized.

Chapter 7

Conclusions

7.1 Achievements

The experimental testing performed was considered to be successful, as it was possible to produce a dynamic structural frequency spectra for two different wing configurations at a low speed regime, while also developing the methodology required for both the performed tests and all future testing of the same type. It also showed the viability of using linear accelerometers as a simple instrumentation basis to perform aeroelastic testing.

As for the numeric framework, the new modular implementation allowed to reduce program complexity and facilitate future add-ons or replacement of existing modules, while at the same time not losing the previous verification efforts produced by Almeida [21].

The aerodynamic module was verified using the open source software XFLR-5, for the new wing configuration that was to be tested both numerically and experimentally, and the results were within expected variation, so no corrective measures were required.

A new type of wing section discretization was added to the structural module, the solid wing section, much simpler than the wing box discretization that the original framework used. This was necessary since the experimental wing model was built from a solid polystyrene block. To check the accuracy of the new discretization, the wing section properties were compared with the values obtained using ANSYS® APDL and were found to have a very small difference in values.

The numeric framework was shown to be able to estimate the flutter speed both by computing the damping ratio associated to the wing's dynamic behaviour and the structural frequency spectra that results from this dynamic behaviour, while also producing aerodynamic data of the wing.

The comparison of numerical and experimental data showed a discrepancy between the measured frequency spectra for both cases, with the experimental results displaying a higher rigidity comparing to numeric results. While this variation cannot be dismissed, it can be seen as an extra safety margin, as the numeric model underestimates the wing's flutter velocity and thus experimental tests can be performed within safety limits.

A parametrization study of the effect of the wing aspect ratio on flutter speed was performed to show that wing rigidity plays a crucial role on the aeroelastic instabilities and further illustrating both the frameworks' capabilities and the major design problem when increasing the wing's aspect ratio to improve the L/D ratio.

The numerical optimization problems served both as additional illustration of the frameworks' versatility while also indirectly allowing to verify the results, as the static aerodynamic and structural solvers produced results that were within expected values, while also accomplishing all goals and constraints imposed. The flutter speed maximization test was also a way to check the interaction between all modules.

7.2 Future Work

During the course of the development of the numerical and experimental study of wings, several possible future additions emerged, relate to both to the numeric and the experimental methodologies developed:, namely:

- After the experimental aeroelastic testing showed some discrepancies with the numerical data, further verifications are required, such as the experimental determination of aerodynamic loads for the tested wings, to check for possible inconsistencies in the aerodynamic behaviour;
- Since the implemented wake shape overestimates the aerodynamic forces, a new wake shape implementation should be done, preferably using a model that gives results closer to those of the experimental tests;
- Although it is not its primordial purpose, the implemented panel method can be easily extended to account for compressibility effects and, therefore, be accurate for a large speed regime. Another potential improvement for the aerodynamic module is to add a viscous module similar to those of VSAERO [36];
- As another source of possible inconsistencies between experimental and numerical results, a structural damping model should be added to the developed framework to further increase its accuracy;
- Perform more experimental aeroelastic tests for different wing materials and configurations, as due to time and material constraints, only two wings were tested. By performing more tests, further improvements to the software's accuracy can be achieved, while also producing more low speed aeroelastic data that is seen to be scarce nowadays;
- All numerical optimization results were obtained using a ”black box” approach for each of the used modules, which reduces significantly the user's options when performing this type of optimization problems. By using the Aeroelastic Framework developed as a base to join the adjoint sensitivity analysis modules implemented by Rodrigues [60] and Freire [61], a more extensive study of aeroelastic wing optimization can be performed, thanks to greatly improved computational efficiency.

Bibliography

[1] R. V. Petrescu, R. Aversa, B. Akash, R. Bucinell, J. Corchado, F. Berto, M. Mirsayar, A. Apicella, and F. I. Petrescu. History of aviation-a short review. *Journal of Aircraft and Spacecraft Technology*, 1(1), 2017.

[2] R. Ricketts. Experimental aeroelasticity-history, status and future in brief. In *31st Structures, Structural Dynamics and Materials Conference*, Long Beach, United States of America, 1990.

[3] R. M. Botez, A. Doin, D. E. Biskri, I. Cotoi, D. Hamza, and P. Parvu. Method for flutter aeroservoelastic open loop analysis. *Canadian Aeronautics and Space Journal*, 49:179–190, Dec. 2003. DOI:10.5589/q03-016.

[4] J. Alonso and A. Jameson. Fully-implicit time-marching aeroelastic solutions. In *32nd Aerospace Sciences Meeting and Exhibit*, Reno, United States of America, 1994.

[5] E. C. Yates Jr. AGARD standard aeroelastic configurations for dynamic response. candidate configuration i.-wing 445.6. *Tech. Rep. 765*, 1987.

[6] D. Kholodar, S. A. Morton, and R. Cummings. Deformation of unstructured viscous grids. *Aerospace Engineering*, 01 2005. DOI:10.2514/6.2005-926.

[7] R. J. Beaubien, F. Nitzsche, and D. Feszty. Time and frequency domain flutter solutions for the agard 445.6 wing. *Paper No. IF-102, IFASD*, 2005.

[8] R. Voß, L. Tichy, and R. Thormann. A ROM based flutter prediction process and its validation with a new reference model. In *15th International Forum of Aeroelasticity and Structural Dynamics, IFASD*, volume 36, Paris, France, 2011.

[9] R. Kamakoti and W. Shyy. Fluid-structure interaction for aeroelastic applications. *Progress in Aerospace Sciences*, 40:535–558, Nov. 2004. DOI:10.1016/j.paerosci.2005.01.001.

[10] E. W. Pendleton, D. Bessette, P. B. Field, G. D. Miller, and K. E. Griffin. Active aeroelastic wing flight research program: Technical program and model analytical development. *Journal of Aircraft*, 37:554–561, Aug. 2000. DOI:10.2514/2.1484.

[11] G. Andersen, D. Cowan, and D. Piatak. Aeroelastic modeling, analysis and testing of a morphing wing structure. In *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Honolulu,United States of America, 2007.

[12] F. H. Gern, D. J. Inman, and R. K. Kapania. Structural and aeroelastic modeling of general planform wings with morphing airfoils. *AIAA Journal*, 40:628–637, Apr. 2002. DOI:10.2514/2.1719.

[13] A. Rapinett. Zephyr: A high altitude long endurance unmanned air vehicle. Master's thesis, University of Surrey, April 2009.

[14] W. E. Silva and R. E. Bartels. Development of reduced-order models for aeroelastic analysis and flutter prediction using the CFL3Dv6.0 code. *Journal of Fluids and Structures*, 19:729–745, Mar. 2004. DOI:10.1016/j.jfluidstructs.2004.03.004.

[15] M. Kämpchen, A. Dafnis, H. G. Reimerdes, G. Britten, and J. Ballman. Dynamic aero-structural response of an elastic wing model. *Journal of Fluids and Structures*, 18:63–77, July 2003. DOI:10.1016/S0889-9746(03)00090-2.

[16] R. Bennett and J. Edwards. An overview of recent developments in computational aeroelasticity. In *29th AIAA, Fluid Dynamics Conference*, Albuquerque, United States of America, 1998.

[17] E. Dowell, J. Edwards, and T. W. Strganac. Non-linear aeroelasticity. *Journal of Aircraft*, 40:857–874, Oct. 2003. DOI:10.2514/2.6876.

[18] D. Tang and E. H. Dowell. Experimental and theoretical study on aeroelastic response of high aspect-ratio wings. *AIAA Journal*, 39:1430–1441, Aug. 2001. DOI:10.2514/2.1484.

[19] E. H. Dowell and K. C. Hall. Modelling of fluid-structure interaction. *Annual Review of Fluid Mechanics*, 33:445–490, Jan. 2001. DOI:10.1146/annurev.fluid.33.1.445.

[20] T. A. Davis. *MATLAB® Primer*. MathWorks Inc., 28^{th} edition, March 2017. ISBN:978-0429152269.

[21] J. Almeida. Structural dynamics for aeroelastic analysis. Master's thesis, Instituto Superior Técnico, Universidade de Lisboa, November 2015.

[22] A. Cardeira. Aeroelastic analysis of aircraft wings. Master's thesis, Instituto Superior Técnico, Universidade de Lisboa, December 2014.

[23] R. Clark, D. Cox, H. Curtiss, J. Edwards, K. Hall, D. Peters, R. Scanlan, E. Simiu, F. Sisto, and W. Strganac Th. *A modern course in Aeroelasticity volume 116 of Solid Mechanics and its Applications*. Kluwer Academic, New York, 4^{th} edition, 2005. ISBN:9781402020391.

[24] T. H. G. Megson. *Aircraft Structures for Engineering Students*. Elsevier, 5^{th} edition, 2013. ISBN:978-0-08-096905-3.

[25] J. Katz and A. Plotkin. *Low Speed Aerodynamics*. Cambridge University Press, 2^{nd} edition, 2001. ISBN:978-0-521-66219-0.

[26] D. H. Hodges and G. A. Pierce. *Introduction to Structural Dynamics and Aeroleasticity*. Cambridge University Press, 2^{nd} edition, 2011. ISBN:978-0-521-19590-4.

[27] T. T. N. Le, C.-C. Nguyen, and H. N. Vang. Design of aeroelasticity bench test for NACA0012 wing model in the low speed wind tunnel: Influence of wing's parameters on flutter speed. *International Journal of Mechanical Engineering and Applications*, 3:35–40, Jan. 2015. DOI:10.11648/j.ijmea.s.2015030103.16.

[28] F. Afonso, J. Vale, Éder Oliveira, F. Lau, and A. Suleman. A review on non-linear aeroelasticity of high aspect-ratio wings. *Progress in Aerospace Sciences*, 89:40–57, Feb. 2017. DOI:10.1016/j.paerosci.2016.12.004.

[29] P. Singh, T. Hesla, and D. Joseph. Distributed lagrange multiplier method for particulate flows with collisions. *International Journal of Multiphase Flow*, 29(3):495–509, 2003. DOI:10.1016/S0301-9322(02)00164-7.

[30] J. D. Anderson. *Modern Compressible Flow*. McGrawHill, 3^{rd} edition, 2012. ISBN:978-1-25-902742.

[31] J. B. Barlow, J. William H. Rae, and A. Pope. *Low-Speed Wind Tunnel Testing*. John Willey & Sons, Inc., 3^{rd} edition, 1999. ISBN:0-471-55774-9.

[32] Piezoelectric accelerometer types 4507 and 4508 datasheet. `https://www.bksv.com/media/doc/Bp1841.pdf`. acessed on March 2019.

[33] Or34 compact spectral analyser manual. `https://sagetechnologies.com/attachments/article/35/OR34-VS%20datasheet.pdf`. acessed on March 2019.

[34] E. F. Sheta, V. J. Harrand, D. E. Thompson, and T. W. Strganac. Computational and experimental investigation of limit cycle oscillations of nonlinear aeroelastic systems. *Journal of Aircraft*, 39: 133–141, Feb. 2002. DOI:10.2514/2.2907.

[35] I. Kroo. AA 241b aircraft design: Synthesis and analysis. Stanford University, United States of America, 2004. Course Notes.

[36] B. Maskew. Program VSAERO theory document: a computer program for calculating nonlinear aerodynamic characteristics of arbitrary configurations. *NASA CR 4023*, Sept. 1987.

[37] J. H. Ferziger and M. Peric. *Computational Methods for Fluid Dynamics*. Springer, 3^{rd} edition, 2002. ISBN:978-3-540-42074-3.

[38] C. Hirsch. *Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics*. Elsevier, 2^{nd} edition, 2007. ISBN:978-0080550022.

[39] J. N. Reddy. *An Introduction to the Finite Element Method*. McGraw-Hill, 3^{rd} edition, 2010. ISBN:007-124473-5.

[40] S. S. Rao. *Mechanical Vibrations*. Prentice Hall, 5^{nd} edition, 2011. ISBN:978-0-13-212819-3.

[41] R. R. Craig and A. J. Kurdila. *Fundamentals of structural dynamics*. John Wiley & Sons, 2^{nd} edition, 2006. ISBN:978-1118174449.

[42] A. Yadav. *Analog Communication System*. Firewall Media, 1^{st} edition, 2008. ISBN:978-8131803196.

[43] O. Bauchau and J. Craig. *Euler-Bernoulli beam theory*. Springer, 2009. p. 173–221.

[44] I. ANSYS. *Theory Reference for the Mechanical APDL and Mechanical Applications*. SAS IP,Inc, release 12.0 edition, 2009.

[45] L. William and M. Paz. *Structural Dynamics: Theory and Computation*. Springer, 5^{th} edition, 2012. ISBN:978-1461504818.

[46] K.-J. Bathe and G. Noh. Insight into an implicit time integration scheme for structural dynamics. *Journal of Computers and Structures*, 98-99:1–6, May 2012. DOI:10.1016/j.compstruc.2012.01.009.

[47] D. Maghdid. *Stability and accuracy of Newmark's method*. PhD thesis, Lund University, Denmark, 2002.

[48] T. C. Corke. *Design of Aircraft*. Pearson College Division, 1^{st} edition, 2002. ISBN:978-0130892348.

[49] J. E. Cooling. *Software Design for Real-time Systems*. Springer, 1^{st} edition, 1991. ISBN:978-0442311742.

[50] H. Bedoya et al. *Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between*. International Technical Support Organization. IBM RED BOOKS, 1^{st} edition, 2014. ISBN:978-0738439860.

[51] M. Drela, H. Youngren, M. Scherrer, and A. Deperrois. XFLR 5. `http://www.xflr5.com/xflr5.htm`. acessed on November 2018.

[52] B. Etkin and L. D. Reid. *Dynamics of Flight: Stability and Control*. Wiley, 3^{rd} edition, 1995. ISBN:978-0471034186.

[53] J. N. S. J. Wright. *Numerical Optimization*. Springer, 2^{nd} edition, 2006. ISBN:978-0387-30303-1.

[54] J. R. R. A. Martins and A. B. Lambe. Multidisciplinary design optimization: A survey of architectures. *AIAA journal*, 51(9):2049–2075, Jan. 2013. DOI:10.2514/1.J051895.

[55] fmincon. `https://www.mathworks.com/help/optim/ug/fmincon.html`. acessed on January 2019.

[56] fgoalattain. `https://www.mathworks.com/help/optim/ug/fgoalattain.html`, 2018. acessed on January 2019.

[57] R. H. Byrd, J. C. Gilbert, and J. Nocedal. A trust region method based on interior point techniques for nonlinear programming. *Mathematical programming*, 89(1):149–185, 2000. DOI:10.1007/PL00011391.

[58] F. W. Gembicki. *Vector Optimization for Control with Performance and Parameter Sensitivity Indices*. PhD thesis, Case Western Reserve University, 1974.

[59] C. T. F. Ross. *Mechanics of Solids*. Elsevier, 1^{st} edition, 1999. ISBN:978-0857099716.

[60] P. Rodrigues. Efficient aerodynamic optimization of aircraft wings. Master's thesis, Instituto Superior Técnico, Universidade de Lisboa, April 2018.

[61] T. Freire. Efficient structural optimization of aircraft wings. Master's thesis, Instituto Superior Técnico, Universidade de Lisboa, February 2017.

[62] F. P. Beer, E. R. J. Jr., and J. T. DeWolf. *Mechanics of Materials*. McGraw Hill, 4^{th} edition, 2005. ISBN:978-0073107950.

Appendix A

Hot Wire

As explained in Chap. 3, the test wing was cut using a hot wire machine, controlled by Jedicut software, as shown in Fig. A.1.

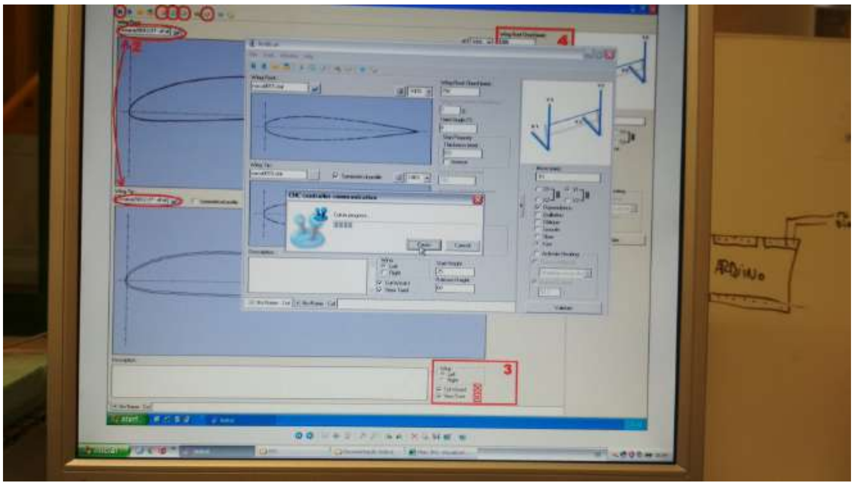


Figure A.1: Jedicut software interface

The motion system consists of 4 motors, two for each side of the wire, to enable vertical and horizontal movements. Each side can move independently, meaning that tapered parts can be made using this assembly. The blocks are cut by a nickel based wire that, by having a current go through it, heats by the Joule effect, melting the block into shape. The wire is kept tensioned by springs and its optimal cut temperature is controlled manually.

A.1 Calibration

To use the machine, calibration tests must be performed before each use. The calibration tests can be divided into two categories:

- Positional calibration;
- Temperature calibration.

For positional calibration, translations across the cut area are performed to see if there is any obstruction to the free movement of the hot wire assembly and, if any are found, they must be removed or additional lubricant should be added, as excess resistance in the moving parts will create defects in the produced part.

Temperature calibration consists of two separate procedures:

- Minimum Cut Temperature - As a default rule of operation, the wire is heated by driving an electric current through it, generated by a power source whose voltage is changed to find the optimal temperature. This is done by performing a series of ”L” shaped cuts in a block with the same length as the desired part to be produced. To find the ideal temperature, the shape of the cut must be uniform along the entire length of the block and, by changing the voltage, different temperatures are tested until an ideal voltage is achieved. Ideally this should be done only when changing the wire, however, since it also depends on the length of the cutting block, it is recommended that at least one test should be done to verify the optimal temperature. Also worth noting that this temperature also depends on the material of which the block is made, and since the machine is operated with at least three different types of materials, this test should be done for all three types, which potentially leads to three different optimal temperatures.
- Skin Thickness - After the optimal temperature is found, a real shape cut is performed, usually of an airfoil, in order to measure the distance between the airfoil’s surface and the original block, in order for the software to produce an offset so that the desired dimensions are obtained when performing the wing cut.

A.2 Cutting Procedure

To start the cutting procedure, firstly a file with the airfoil shape is required, in normalized coordinates. The software allows for different airfoils at wing root and tip and it also does the conversion to real coordinates with the root and tip chords that the user inputs. The wing twist is also a required for both the wing tip and root, as well as the skin thickness, so that the final shape has the exact dimensions established by the user.

The next step is to introduce wing span and sweep, with the sweep being defined as an offset between the root chord leading edge and the tip chord leading edge.

Lastly, the cutting procedure needs to be defined, as in the starting coordinate for the cut and the type of motion that the machine should perform to cut the desired shape. It was verified that to increase the smoothness of the cut surface, the process should start with the horizontal cut that ends at the airfoil’s trailing edge, as it has a smoother transition between both routines, as shown in Fig. A.2.

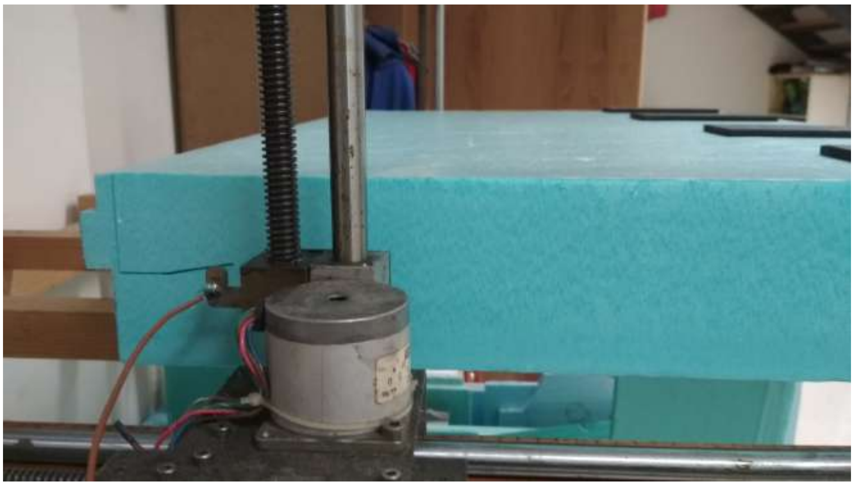


Figure A.2: Detail of the hot wire machine machining the polystyrene block

Appendix B

Experimental Characterization of Mechanical Properties

B.1 Density

The material's density was determined by cutting a rectangular block with known dimensions and weighting it to determine its mass m. Since the dimensions are known, the volume V is easily computed and, by definition, the density ρ is given by

$$\rho = \frac{m}{V} \, . \tag{B.1}$$

B.2 Young Modulus

The elastic modulus was obtained with a three point bending test as shown in Fig. B.1, which is normally used to determine the elastic flexural module. However, since for a large array of materials, the flexural module is nearly identical to the elastic module, they are considered to be equal in this case. By measuring the applied force P, the material displacement y_{max}, computing the mass moment of inertia

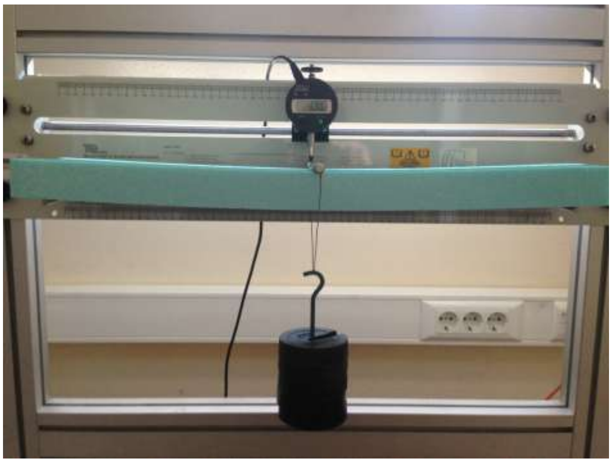


Figure B.1: Three point bending test

I and assuming a linear correlation between both properties, us the elastic modulus E can be computed obtained from [62]

$$y_{max} = \frac{L^3}{48EI} P \; . \tag{B.2}$$

To increase accuracy, the process was repeated six times to allow for an average of elastic module values to be computed, to increase accuracy.

B.3 Shear Modulus

The shear modulus G was obtained by a torsion test as shown in Fig. B.2, in which a test specimen is encased in one end and rotated in the other extremity.

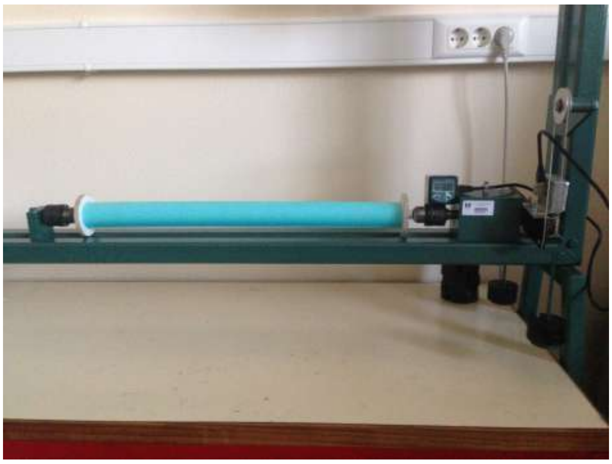


Figure B.2: Torsion test

By measuring the corresponding angle of twist φ caused by the rotation and knowing the applied torque T, the shear modulus is obtained using the relation [62]

$$T = \frac{J_T}{l} G\varphi \; , \tag{B.3}$$

where l is the length of the tested specimen and J_T is the torsion constant The latter is usually approximated by the second moment of area about the neutral axis (J), which for the circular section tested, is equal to

$$J = \frac{\pi d^4}{32} \; . \tag{B.4}$$