Summer 2024

Optimization of a Plate Beam System for Energy Harvesting Using a Piezoelectric Material

Jose Manuel Almendros Espantaleon
Embry-Riddle Aeronautical University, almendrj@my.erau.edu

OPTIMIZATION OF A PLATE BEAM SYSTEM FOR ENERGY HARVESTING USING A PIEZOELECTRIC MATERIAL

By

Jose Manuel Almendros Espantaleon

A Thesis Submitted to the Faculty of Embry-Riddle Aeronautical University

In Partial Fulfillment of the Requirements for the Degree of

Master of Science in Aerospace Engineering

July 2024

Embry-Riddle Aeronautical University

Daytona Beach, Florida

OPTIMIZATION OF A PLATE BEAM SYSTEM FOR ENERGY HARVESTING

USING A PIEZOELECTRIC MATERIAL

By

Jose Manuel Almendros Espantaleon

This Thesis was prepared under the direction of the candidate's Thesis Committee Chair, Dr. Mandar Kulkarni, Department of Aerospace Engineering, and has been approved by the members of the Thesis Committee. It was submitted to the Office of the Senior Vice President for Academic Affairs and Provost, and was accepted in the partial fulfillment of the requirements for the Degree of Master of Science in Aerospace Engineering.

THESIS COMMITTEE

Mandar D. Kulkarni Digitally signed by Mandar D. Kulkarni Date: 2024.08.07 10:14:44 -04'00'

Chair, Dr. Mandar Kulkarni

William A. Engblom Digitally signed by William A. Engblom Date: 2024.08.06 22:54:56 -04'00'

Member, Dr. William Engblom

Vladimir Golubev Digitally signed by Vladimir Golubev DN: cn=Vladimir Golubev, o=ERAU, ou=Aerospace Engineering, email=golubd1b@erau.edu, c=US Date: 2024.08.07 07:08:44 -04'00'

Member, Dr. Vladimir Golubev

Hever Moncayo Digitally signed by Hever Moncayo Date: 2024.08.07 13:58:25 -04'00'

Graduate Program Coordinator,
Dr. Hever Moncayo

Date

Dean of the College of Engineering,
Dr. James W. Gregory

Date

Associate Provost of Academic Support,
Dr. Kelly Austin

Date

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my parents Jose Manuel and Maria Luisa for their support, love, and encouragement throughout my entire life. Their belief in my potential has been a constant source of motivation. My sister Oliva has also been a pillar of support, always ready to lend a listening ear and offer valuable advice. Special thanks to Alejandra for her patience, understanding, and encouragement, which have been invaluable during my academic journey. I am also grateful to my friends from Spain, whose fellowship has been a source of joy and balance. Additionally, I thank my new friends in the USA, who have made this experience truly memorable.

Special thanks to my thesis advisor Dr. Mandar Kulkarni, whose constant support and insightful advice have been instrumental in the successful completion of this work. I would also like to thank the committee members Dr. William A. Engblom and Dr. Vladimir V. Golubev for their valuable feedback and guidance throughout this research.

I extend my sincere thanks to Embry-Riddle Aeronautical University for its warm hospitality during my stay in Dayona Beach, which provided a conducive environment for my research. I am equally thankful to Universidad Europea of Madrid for the dual degree program and for offering me the opportunity to study abroad, which has enriched my academic and personal experiences.

My achievements would not be the same without all of you.

ABSTRACT

With a continuously growing demand for power, driven by the need to reduce our environmental footprint, this research provides an examination of the potential of energy harvesting with smart materials technology and its practical applications. The energy harvesting system considered here works on generating energy through vibrations of a piezoelectric material beam which will undergo sustained vibrations due to flow of air over its surface. It is assumed that sustained limit cycle oscillations of this system will occur at the flutter velocity. This research creates an optimization framework to obtain the best values of parameters that will result in the minimum flutter velocity for the system. Minimization of flutter velocity may lead to the use of the energy harvesting system at lower air speeds, thus increasing its applicability in multiple low-velocity vehicles/scenarios. The study begins with an in depth explanation of piezoelectricity, its fundamental concepts, operational mechanisms, and various applications. Next, the phenomenon of flutter is explained in detail as it is essential for identifying conditions where vibrations can be harnessed for energy generation. Two codes are developed to determine the flutter speed for both steady and unsteady flows, which are also verified against previous studies, ensuring their accuracy and reliability. Further, several codes are created to optimize the minimum flutter speed, initially focusing on a single parameter, then expanding to two parameters and finally optimizing all four parameters simultaneously. With the last case, the flutter velocity is reduced an 80% from its starting value. So far, to the author's knowledge, there are not too many works that follow a detailed optimization process of the parameters involved in piezoelectric power generation. This optimization process is particularly significant as it lays a foundation for future studies, enabling a more comprehensive and efficient optimization of energy harvesting systems.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

a	Dimensionless distance from leading edge to elastic axis
AI	Artificial intelligence
b	Half chord length
c	Chord length
$C(k)$	Theodorsen's function
C_L	Lift coefficient
$CFRC$	Carbon fiber-reinforced concrete
D_h	Plunging dissipative structural damping
D_θ	Pitching dissipative structural damping
e	Dimensionless distance from leading edge to center of gravity
g_h	Plunging damping coefficients
g_θ	Pitching damping coefficients
GA	Genetic algorithm
h	Plunge
$H_n^{(2)}(k)$	Second order Hankel functions
I_C	Moment of inertia around the center of gravity
I_P	Moment of inertia around the elastic axis
ILP	Integer linear optimization
k	Reduced frequency
K	Kinetic energy
k_h	Plunging spring constant
k_θ	Pitching spring constant
L	Lift

L_g	Lagrangian
LCO	Limit cycle oscillation
LP	Linear optimization
M	Moment around elastic axis
m	Mass
M_∞	Mach number
M_Q	Moment around quarter chord point
MFC	Microfiber composite
MILP	Mixed-integer linear optimization
NLP	Nonlinear optimization
p	Reduced eigenvalue (p method)
P	Potential energy
r	Radius of gyration
SMA	Shape memory alloy
SQP	Sequential quadratic programming
$T_{lh}, T_{l\theta}$	Theodorsen's dimensionless aerodynamic lift
$T_{mh}, T_{m\theta}$	Theodorsen's dimensionless aerodynamic moment
U	Flow velocity
V	Reduced velocity
V_F	Reduced velocity of flutter
v_C	Velocity of the center of gravity
v_P	Velocity of the elastic axis
v_Q	Velocity of the quarter-chord point
x_θ	Dimensionless distance from center of gravity to elastic axis

Z	Reduced eigenvalue (p-k method)
Γ	Modal damping
θ	Pitch
μ	Ratio of mass
ρ_∞	Density of airflow
σ	Ratio of uncoupled frequencies
Ω	Modal frequency
ω	Frequency of motion
ω_h	Plunging uncoupled natural frequency
ω_θ	Pitching uncoupled natural frequency

1. INTRODUCTION

In recent decades, we have witnessed a dramatic increase in global energy consumption. From 1965 to 2022, this consumption has experienced an impressive 400% increase [1], as shown in Figure 1.1. This phenomenon has been driven by several factors, including population growth, industrial and technological development, as well as rising living standards in many parts of the world.

Even though the use of renewable energy sources has experienced exponential growth, reaching an outstanding 2400% increase, renewable energy is less than 10% of the total energy consumption. We still face significant challenges in the transition to a more sustainable energy system. Continued investment and research are needed to overcome technological and economic barriers and achieve a truly global energy transformation. With a continued dedication to innovation, international collaboration, and collective action, we can move towards a cleaner, more secure, and equitable energy system for future generations.

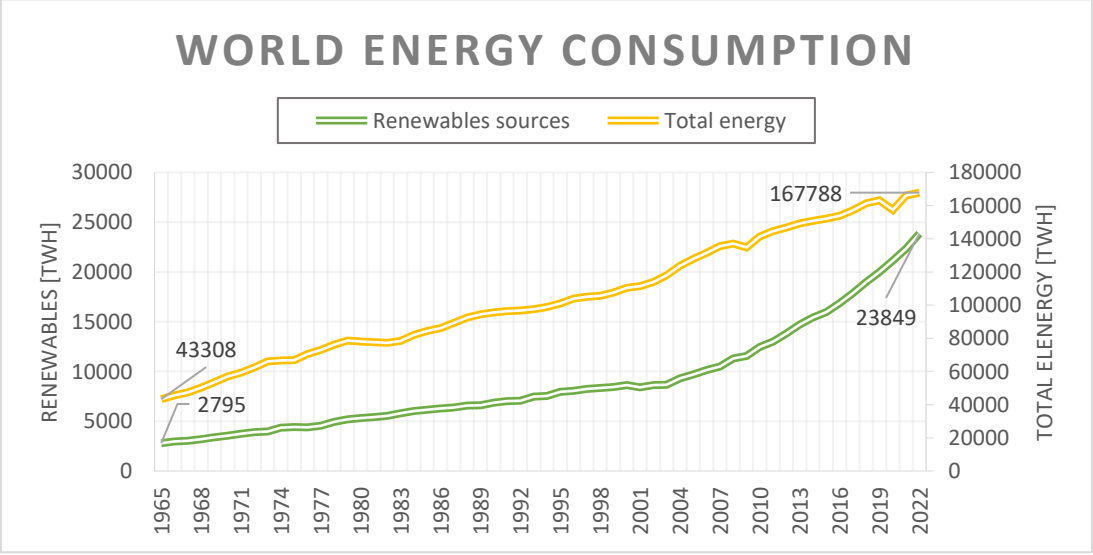

Figure 1.1 Energy consumption 1965-2022 [1]

My commitment to the transition towards a more sustainable energy future has been strengthened. That is why I have decided to dedicate my Master's Thesis to the study of energy harvesting in the aerospace sector, to contribute to the knowledge and development of innovative solutions in this field. My motivation lies in the recognition of the positive impact that renewable energy research can have in the fight against climate change and the construction of a more just and cleaner world for future generations.

For these reasons, I decided to investigate new forms of generating renewable energy. Concretely, I wanted to study smart materials and their possible impact on more sustainable aviation. The principal objective of my thesis is to create a framework to harvest the maximum amount of energy with a piezoelectric patch that works with the vibrations of a beam. First, a model will be created to obtain flutter velocity of an airfoil-cross section beam that is being vibrated due to air flowing over it. In other words, flutter velocity is the velocity of air flow which triggers limit cycle oscillations of the vibrating beam. The energy harvesting system that is considered here works on converting the energy of such sustained oscillations into electrical energy through smart materials such as a piezoelectric material. Further, a framework will be created to optimize various parameters of this model to minimize the flutter velocity. This is aimed at creating an energy harvesting system that could generate energy at the lowest air speeds possible. I hope this study will help to develop this system in future aviation, as well as other energy harvesting concepts.

In this section, the principal components of this work will be discussed. Some topics are the need for energy harvesting applications, smart materials used in energy harvesting, applications, flutter velocity, limit cycle oscillation mechanisms, and optimization models.

1.1 Energy harvesting

It is necessary to investigate new directions in energy harvesting to find ways to utilize unexplored resources while reducing the impact on the environment. In this subsection, the need for energy harvesting applications will be explained, focusing on their significant role in the development of sustainable energy solutions.

Energy harvesting, or power scavenging, is known as the process of extracting energy from the ambient environment and converting it into consumable electrical energy [2]. Since several years ago, the most common forms of extracting energy from the ambient environment are with photovoltaic cells, wind turbines (onshore or offshore). The difference between energy harvesting and the other common sources of renewable energy such as wind, geothermal, hydropower, ocean, etc. is that in the former one, the amount of energy harvested is much smaller than from the conventional renewable sources, as it is usually used for small and low powered autonomous devices and applications.

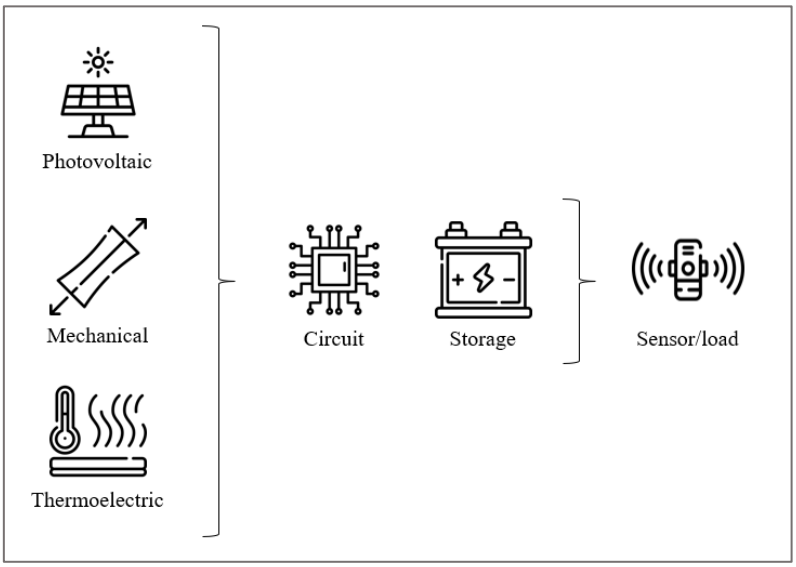

Figure 1.2 Energy harvesting system

Every energy harvesting method is usually formed by three basic part components: the harvester or transducer, which transforms the ambient power from the environment into useable electrical energy, the circuit and storage, which collects, analyses, and processes that energy, and the load which puts the energy into use, as shown in Figure 1.2.

There are a lot of types and forms of energy harvesting. They are classified based on the form of energy they use to extract power. Some of the most common energy harvesting sources are solar energy, mechanical energy from strain and vibration, thermal energy from furnaces or combustion engines, human energy from different bodily functions and metabolisms, sound energy, etc. In the following sections, we are going to deal in more depth with the mechanical energy coming from vibrations.

1.2 Smart materials used in energy harvesting

As has been said before, the world is facing a critical moment with the highest energy consumption, and most of it being generated by limited non-renewables resources. The need for alternative energy sources is pressing due to the increasing reliance on technology and population increase, which is placing a strain on existing energy sources. This is where smart materials promise to be useful. Because of their adaptable qualities, smart materials react quickly to outside inputs such as electromagnetic waves, temperature fluctuations, and mechanical stress. Through the utilization of their distinct properties, such as thermoelectricity, shape memory alloys, or piezoelectricity, smart materials provide opportunities for capturing and transforming ambient energy sources into functional power. Those materials play a crucial role in facilitating the research for

sustainable energy solutions due to their adaptability to a variety of environmental situations that were previously untapped.

One of the most important smart materials is piezoelectric materials. These materials are used because when they are strained, they generate electricity, as shown in Figure 1.3. They are tiny crystals or ceramics that convert mechanical movements or vibrations into electrical energy. That is the key to the present work: harvesting energy from the vibrations of a cantilevered beam using a piezoelectric patch. Piezoelectricity will be explained widely in the next section: 1.3 Piezoelectricity.

Piezoelectric materials are used in devices like sensors, actuators, and importantly, in energy harvesters. By incorporating them into structures like beams or panels, we can capture otherwise wasted energy from vibrations, footsteps, or even the wind, turning it into usable electricity. A good example of this in the aeronautical industry is about structural health monitoring work, which is proving highly significant in avoiding the premature collapse of structures of aircraft [3].

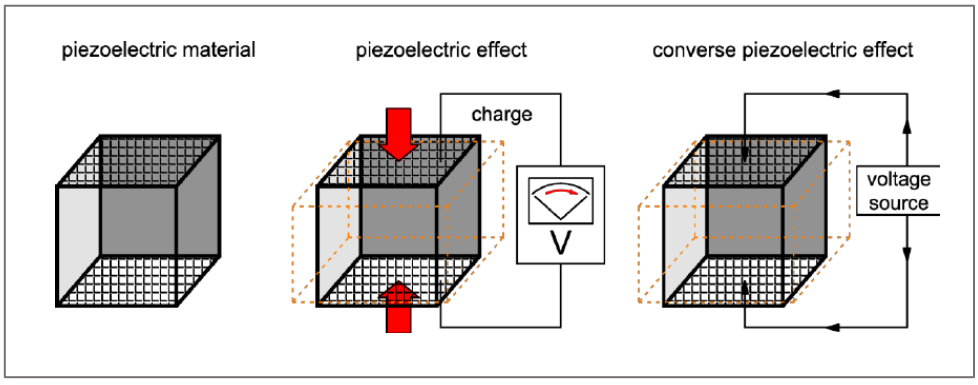

Figure 1.3 Piezoelectricity basic scheme [4]

Another set of smart materials is thermoelectric materials. They can produce electricity when there is a temperature differential across them. Those materials are utilized in equipment that produces electricity from residual heat, such as that produced by industrial operations or automobile engines. Thermoelectric materials assist in reducing environmental impact and promote more effective energy utilization by using this lost heat energy.

There has been some research where thermoelectric smart materials have been applied. One example are thermoelectric modules which were woven directly into the fiber textiles of some clothes to harvest energy from body heat, with temperature differences of around 44 K [5]. Another study of thermoelectric materials was conducted through carbon fiber-reinforced concrete (CFRC). Others demonstrated that the energy harvested increased with the content of short carbon fiber in it [6]. That smart material could be used in critical civil infrastructure systems, such as bridges and dams, to prevent them from cracking due to cement hydration and sun radiation.

An additional group of smart materials includes shape memory alloys (SMAs). These materials have a remarkable ability to return to a predetermined shape when subjected to certain temperatures. This unique behavior finds applications in devices like actuators, valves, and even in biomedical implants. In energy harvesting, SMAs are being explored to capture heat energy and convert it into mechanical movement, which can then be transformed into electricity.

SMAs have been used for plenty of different applications, also in the aerospace industry. Concretely, for many years they have been utilized in spaceships as low-shock release mechanisms as well as in smaller spacecraft or microsatellites [7]. Also, in airplanes, new methods with SMA materials are being investigated. For example, the

patent of Knowles & Bird [8] was used to develop an innovative telescopic wing system using those kinds of smart materials.

1.3 Piezoelectricity and applications using flutter and LCO mechanisms

The word “piezo” derives from the Ancient Greek word “πιέζω,” which means press, squeeze, or compress. In 1880, the French physicists Paul-Jacques Curie and his brother Pierre Curie discovered that in certain crystals, when a mechanical force was applied, they became electrically polarized and that this polarization was proportional to the applied voltage [9]. They named this effect piezoelectricity, and they used crystals such as tourmaline, topaz, quartz, and Rochelle salt, among others.

The Curie brothers started their research with the study of pyroelectricity, generating electricity by applying thermal loads to certain materials, which was discovered in the 18th century. They got a conclusive relation between electric charge and mechanical stress.

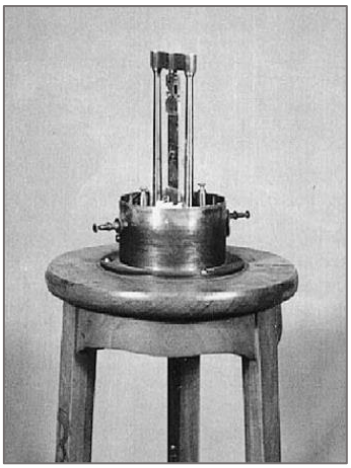

Figure 1.4 Curie compensator [10]

To demonstrate piezoelectricity, the brothers created the Curie compensator, as shown in Figure 1.4, where a quartz or a Rochelle salt crystal lamella was compressed, producing small electric currents, around a few tenths of picoamperes. One year later, in 1881, Gabriel Lippmann deduced the converse effect, where applying a small electric field to some materials produced deformations [11]. That discovery helped the Curie brothers to keep obtaining proof of the reversibility in piezoelectric materials.

Since the discovery of piezoelectricity in 1880, it had not been used until the second decade of the 1900s with the invention of the famous SONAR (Sound Navigation and Ranging), a technique that uses sound propagation underwater primarily to navigate, communicate, or detect submerged objects [12].

Due to the loss of the Titanic in 1912, and other such accidents, there was a need to detect submerged elements under the ocean. Also, with the emerging First World War (1914-1918), armies knew the enormous advantage of using submarines to detect enemies' ships. Those two events coupled, made researchers around the world investigate a solution. Paul Langevin designed the first successful SONAR in 1917. His invention consisted of a medium of acoustic localization, where the acoustic signal was generated by piezoelectricity.

In the post-war period, many of the piezoelectric applications that we use today (microphones, accelerometers, signal filters, etc.) began to be developed, although the materials available (mainly quartz) greatly limited the performance capabilities of the devices. Then, during the Second World War, investigations in the United States, Japan, and the Soviet Union, found that some ceramic materials had piezoelectric constants up to 100 times higher than ordinary crystals previously used.

Physically, piezoelectricity is defined as the property of some crystals to cause their dipoles to polarize electrically when a mechanical or electrical voltage is applied to them, producing electrical energy from the deformation or compression of the material. Thus, the degree of polarization is directly related to the applied voltage.

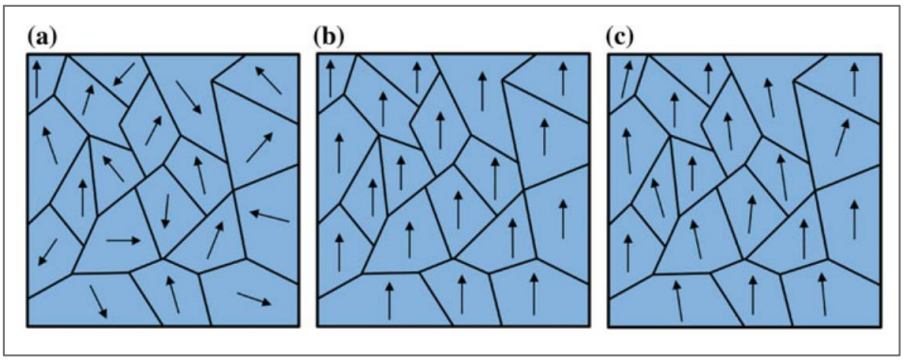

Figure 1.5 Polarization of ceramic [13]

Weiss domains are sets of dipoles having the same alignment. In Figure 1.5 (a), we can see the random distribution of Weiss domains with varying alignment inside the ceramics. When an electric field is applied, it causes the dipoles present in the crystal to polarize, Figure 1.5 (b). This process is known as polarization, where the material exhibits its piezoelectric properties causing a deformation in it. With time, that piezoelectric effect relaxes and ages, Figure 1.5 (c), so most of the dipoles remain not identical but slightly misaligned to those in Figure 1.5 (b), reducing its piezoelectric effect.

We can see plenty of different applications where the piezoelectric effect was used to harvest energy. One example is an experiment conducted in Greece, having human body as a source of generation [14]. They built a glove with PVDF material, which produces energy from the force exerted by the fingers on a flat surface such as a tablet or mouse, demonstrating that it can produce 0.3236 W, see Figure 1.6.

Figure 1.6 Piezoelectric glove [14]

As has been explained, harvesting energy with a piezoelectric patch is based on the tensile and compressive stresses, produced by the vibrations of a beam-like structure. Further, it is needed to make an effort to find a kind of vibrations that could be maintained more or less constant over time. That is why it is needed to study flutter and limit cycle oscillations.

Flutter could be defined as a self-excited and oscillatory aeroelastic instability in which the inherent modes of vibration of a flexible body combine with its aerodynamic forces to create oscillations that amplify themselves to trigger structural collapse. This can occur in aircraft, buildings and even bridges, and is the result of the interaction of inertial and stiffness forces on the structure. The idea of an energy harvester based on this concept is to set a flexible piezoelectric material vibrate close to its flutter boundary. The piezoelectric materials would experience mechanical strain as a result of this vibration, which results in electrical charges. Prediction of flutter velocity will be explained in section 4.2 Flutter.

On the other hand, limit cycle oscillations, also known as LCO, are stable and repetitive oscillations that arise when a system is subjected to periodic forces or vibrations. These oscillations maintain a relatively constant amplitude, which is crucial for energy harvesting because it allows for a consistent generation of electrical power. LCO provides a predictable and stable motion that can be exploited to continually produce electrical energy. In the remainder of this thesis, the word "flutter" is used in the context of limit cycle oscillations, and thus the mention of a beam undergoing flutter actually implies that it is set under in LCO, rather than it undergoing a destructive vibration.

These mechanisms are incredibly important for energy harvesting, especially in this project. The vibrations would be induced by flutter and would become repetitive limit cycle oscillations that would ensure a steady and predictable generation of electrical charges. This predictability is key in maximizing energy harvesting because it allows for the consistent capture and conversion of mechanical energy into electrical power.

There has been some previous research on the importance of flutter and LCO for energy harvesting. It was used an electromagnetic energy harvester that included an airfoil vibrating under LCO response and the accompanying power generation, for cases under stall conditions [15]. Another example was an experimental proposal that took advantage of the oscillations (fluttering) of a flag with piezoelectric material (PVDF) to collect wind energy with a power generation of 1-5mW/cm^3 for wind speeds ranging from 5 to 9 m/s [16]. This experiment opens up new opportunities to use wind energy to power small electronic devices, thereby powering small electronic devices, thus expanding the range of possible applications in which energy harvesting applications can be used.

1.4 Optimization models

Once the part of piezoelectricity has been explained, it is needed to introduce the optimization models. One of the key parts in this thesis will be optimizing the parameters of the beam so as to generate energy at the lowest possible flutter speeds.

Optimization is the process when there is a search through a wide range of potential solutions to find the optimal one. The final approach is called optimal, as it either maximizes or minimizes the desired outcome. Without realizing it, we solve optimization problems almost every day. For example, while determining the most efficient path to get to work, when conducting our weekly grocery shopping, or when determining how to spend our free time. Even though it is easy to solve these daily problems with just our minds, it is needed to employ other methods when attempting to solve other types of problems that are more complicated.

An optimization model is a real-world problem that has been mathematically represented using formulas that we have identified from reality. After the problem is described, the best solutions can be quickly identified by using some algorithms.

Optimization models usually have four primary components: parameters, variables, objective function, and constraints, as seen in Figure 1.7. Parameters are the facts which values are determined by the problem. Variables are those properties that are studied, and which values will vary until the optimal solution is gotten. As implied by its name, the objective function represents our desired outcome, basically, what is wanted to be optimized. Lastly, the restrictions are those that govern the problem, external constraints, which keep the solution from becoming unachievable.

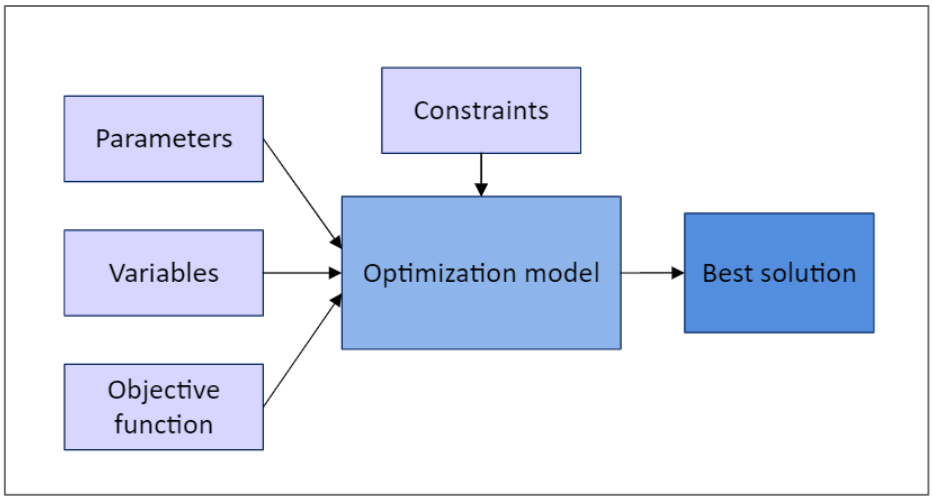

Figure 1.7 Optimization process scheme

Optimization models can be classified into several types, each suitable for different types of problems. The following is an introduction to the main optimization models, their differences, advantages, and disadvantages.

Linear optimization (LP): It is used to maximize or minimize a linear objective function with linear constraints. This kind of optimization is widely applied because of its simplicity and its effective algorithms like the interior point technique and the simplex approach. The main disadvantage is its linearity, it cannot be used in complex situations. Some variations of the linear optimization are the integer linear optimization (ILP) and the mixed-integer linear optimization (MILP), where some or all the variables are integer. These last are able to model discrete decisions and get exact solutions but require specialized algorithms and could be computationally intense.

Nonlinear optimization (NLP): It is used to maximize or minimize a nonlinear function with nonlinear constraints. It has flexibility to model a wide range of real-world problems because it allows dealing with complex relationships between variables. The disadvantage is that these algorithms require a higher computational cost. Inside NLP, the

most used algorithms are interior point, sqp, and active set. These last ones will be further explained in chapter 5.1.2 Algorithms.

Stochastic optimization: This kind addresses problems in which some of the data are uncertain and are modeled as random variables. Objective functions and constraints usually include expectations, variances, and other statistical measures. Even though data variations, it allows robust decision making but requires advanced techniques such as simulation and dynamic programming. This approach is essential in areas such as risk management, finance, and planning under uncertainty.

Genetic algorithms (GA). This is one of the newest methods and is inspired by the evolutionary theory of natural selection proposed by Charles Darwin. It is especially useful for large search spaces, nonlinear optimization problems, and is also used with artificial intelligence (AI). This methodology differs from those mentioned above as it works with a population of potential solutions that evolve over generations.

In conclusion, each type of optimization model has its own set of characteristics. The choice of the appropriate model depends on the problem, the characteristics of the objective function and constraints, as well as the specific needs and priorities of the application. For this thesis, the optimization model chosen is the nonlinear optimization (NLP). The choice is justified by its ability to solve complex and realistic problems with nonlinear relationships. Also, NLP offers flexibility and accuracy, and there are plenty of advanced tools available for its implementation. NLP will allow optimizing a minimum flutter speed for the variables of the size of the beam determined. The optimization model will be explained in the last chapter.

2. OBJECTIVES

The focus of the present work is to create an optimization framework to obtain the best values of parameters that will result in the minimum flutter velocity for the system. The objectives of this research are multi-faceted, addressing fundamental concepts, aerodynamic phenomena, and the development of computational tools for optimization.

Objective 1: Understanding the piezoelectricity effect. This involves studying how certain materials can generate electrical energy when subjected to mechanical stress. That is the first part to be explained, by exploring its principles, operational mechanisms, and applications.

Objective 2: Flutter, Critical Velocity, and Limit Cycle Oscillations. LCOs are those oscillations which are found when the flow velocity is increased beyond the linear flutter boundary, at its critical velocity. This objective involves studying the conditions under which these phenomena occur and how they can be used for energy generation using piezoelectric materials.

Objective 3: Flutter analysis. Developing computational tools to determine flutter velocity. This includes creating two distinct codes that can accurately predict the flutter velocity for both steady and unsteady flows and verifying them against existing studies.

Objective 4: Optimization process. The final objective is to create several codes for optimizing the minimum flutter velocity with respect to one, two, and four parameters simultaneously: mass ratio, radius of gyration, and distances to the elastic center and center of gravity. This multi-parameter optimization serves as a starting point for future research, enabling a more complete and efficient optimization for energy harvesting.

3. LITERATURE REVIEW

As have been said before this study focuses on the optimization of flutter velocity and its application with piezoelectric materials in aviation for power harvesting. Through this literature review, studies ranging from the fundamental principles of piezoelectric materials to the most recent applications in aviation will be reviewed. Those studies help not only to understand better the concepts of piezoelectricity, but to understand what research has already been done, and what aspects will be developed in this study.

The literature review is structured as follows: diverse applications of piezoelectricity, theoretical foundations of critical flow velocity, empirical studies with real experiments; and finally, the importance of some parameters in energy harvesting. That will be the most important part of the literature review, as this thesis presents an emphasis on the optimization of these parameters.

Through this literature review, it will be seen that previous studies have shown that it is possible to use piezoelectric materials to transform mechanical energy from vibrations into electrical energy, providing creative solutions for airplanes. Nevertheless, to the author's knowledge, there are not so many studies where parameter optimization has been developed.

3.1 Applications of piezoelectricity

Since piezoelectricity was invented, there have been plenty of applications where piezoelectric effect was used to harvest energy. Some examples are explained below.

The first case was the SONAR, a radar invented in 1915 by Paul Langévin [17]. After the loss of the Titanic in 1909 and the emerging First World War, a lot of scientists

got inspired and began to investigate underwater radars. SONAR used piezoelectric materials and it could detect submerged objects under the ocean.

Another example, more recently, consisted of a glove that had the body as a source of energy generation [14]. They obtained energy from the force produced by the fingers on a flat surface such as a tablet or mouse.

On the other hand, last year, some laboratory colleagues developed the following shoes prototype [18]. In this study, piezoelectric shoe soles were introduced as a way to use walking as an alternative energy generation source. The sole was connected to the charging circuit so that every step helps in the charging of the battery. They built it in the laboratory and got experimental results, see Figure 3.1.

*Figure 3.1 Pseudo-p*iezoelectric foam shoe [18]

3.2 Critical flow velocity

One of the fundamental parts of this work is to find the critical velocity: the air flow velocity beyond which any small perturbation will produce instability, making the plate flutter. There has been a lot of research on critical velocity, but here is a summary of the most interesting.

Amandolese [19] discussed fluid-structure instability in aerodynamic bodies. Particularly, they studied a dimensionless two degrees of freedom flat plate inside a wind tunnel with low-speed airflow. In the article, they presented experimental results to enhance understanding of sustained vibrations with significant amplitudes. The model configuration is a flat plate that has only two degrees of freedom: plunge and pitch. Lift and moment coefficients were measured with different angles of attack and were compared to previous work.

With the experiments, it is observed that with velocities under the critical velocity, plunging and pitching responses are damped to a small vibration regime. At the critical velocity, any small perturbation will produce pitch and plunge frequencies to increase until the oscillation amplitudes saturate, producing an LCO with amplitude varying slowly in time (the growth rate is almost null). However, if the flow velocity is increased the dynamical response changes dramatically: after some instabilities, the system becomes in a large harmonic oscillations LCO with more than double effect in plunge and pitch than with the critical flow velocity.

In conclusion, under critical velocity, the system remains stable even if small perturbations are applied, and its response is damped. When the critical velocity is reached, any small perturbation would induce the system into a low amplitude plunge and

pitch LCO regime. However, if the flow velocity is increased, the system is stable, but it becomes a high-amplitude LCO regime.

Other authors have investigated the opposite effect: preventing and eliminating aeroelastic instabilities with smart materials. Concretely, one of them studied it in wind turbine blades [20]. In this work it is pretended to understand the LCO frequencies and how to use them in a wind turbine blade with smart materials, to delay flutter appearance. Two wind turbine blades were considered: one standard blade and the other one with a piezoelectric patch on it.

After the analysis, it was concluded that for both the standard and the piezoelectric wind turbine blades, the torsion displacement was not a problem since its eigenvalues are negative (which indicates a stable mode of vibration). Additionally, the possibility of reaching flutter with torsion motion was less probable in the blade with the piezoelectric patch than in the other one. Studying the flap motion, it has been observed that for both wind turbine blades, there was a moment when its eigenvalues became positive (which indicates instability). However, it was demonstrated that for the wind turbine blade with the piezoelectric patch, the critical velocity was 5.4% higher than the regular blade, so the instability would be reached later in the one with the piezoelectric patch.

3.3 Experimental approach and parameter optimization

Other studies where people made experiments to prove energy harvesting on beams will be discussed in this part. These studies are very interesting because while researchers work on experiments, I will be developing the theoretical model, so they could help to verify it. Also, one of the key factors in this work is the optimization of the different parameters of beam to minimize the velocity of flutter. There have been a lot of

people who studied how the piezoelectric patch characteristics affect the energy harvesting model, but not many of them made it with an appropriate optimization model.

Zacaria et al. [21] studied energy harvesting in a beam which was excited by the aerodynamics loads in a wind tunnel. The beam was subjected to air flow at particular angles of attack, and there were noticeable static deflections that resulted in limit cycle oscillations. Most of the energy harvested was achieved at certain angles of attack, which vary with flow velocity. In this work, it was also studied the effect of the resistive loads on the energy harvested. However, an optimization process was not conducted.

Another remarkable project was the study of energy harvesting on a beam connected to a flap, mounted in a wind tunnel [22]. They also stated that there must be a minimum cut-in wind speed below which the flutter energy harvesting cannot function. They concluded that the design of the energy harvester would depend on the characteristics of the flow conditions for the intended application. Also, they studied the effects that electromechanical coupling produces on flutter speed. Even though they found out the optimal resistive load for the highest cut-in wind speed, they did not investigate the influence of the rest of parameters.

Similarly, the following work studies another geometry of cantilevered beam [23]. The beam was made of a single piezoelectric layer with a couple of asymmetric tip masses. Power was extracted from the bending and torsion movements of the beam with a piezoelectric patch. Apart from modeling the structural analysis, they studied the effect of some parameters on power extraction. They studied six different resistive loads and five different distances to the masses. Although they got the highest value of voltage output between their values, they did not go through an optimization process.

Others evaluated the performance of bimorph piezoelectric energy harvesters [24]. The key to their work is the addition of distinct levels of layers. They varied the number of layers from one to three while keeping the total thickness constant. First, they made the structural model and then they validated it with experiments. They concluded that the maximum output power was reached with double layer.

A further intriguing work was also converting the vibrations available in the environment into productive voltage through piezoelectricity [25]. A finite element model was presented to predict energy generation. At the end, they evaluated the effect of the thickness and length of the piezoelectric layers. As happened with other studies, they did not make a suitable optimization process to get the optimal values.

Another remarkable study made by Lahoti [26], a laboratory colleague, who studied of a cantilevered beam made of Microfiber Composite (MFC) for energy harvesting due to vibrations. These vibrations were created by a mechanical vibration exciter. The current thesis differs from this work in that the vibrations are created through air flowing over the beam, rather than by a vibration exciter. At first, he set up a finite element numerical model to predict the energy harvested through a piezoelectric patch on the MFC beam which was then validated with a shaker test as an experiment. Finally, he performed an optimization of three variables: length, width, and thickness. For the energy harvesting part, he obtained small discrepancies between the experiment and the model previously created. For the optimization part, he concluded that to obtain the maximum amount of energy, it was needed to increase the length of the beam and use an exponential width. This work is very interesting because it was proven again that the size of the piezoelectric patch has a significant effect on the energy harvested.

Then, Abdelrahman et al. [27] studied the effect of the piezoelectric patch size and material to suppress vibrations in a wind turbine blade. A smart material sensor and an actuator were put under and above a cantilever beam, with a cross-section of the NACA 0012 airfoil. The model of vibration reduction used in that study was previously validated [28]. It was proven that increasing width or thickness of the piezoelectric path produces a reduction of the actuation force and the maximum displacement whereas an increment in the settling time and the peak amplitude. However, this work measured the effect of these parameters mentioned above by repeating the same code varying each parameter between four different values. My contribution in this thesis will be modeling a matlab code which could optimize these parameters in a single execution, with less computational cost and time.

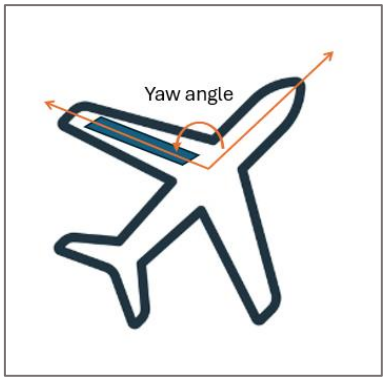

Figure 3.2 Yaw angle scheme

Other parameters can also be evaluated, such as the influence of the orientation of the piezoelectric patch. Some researchers studied which angle between the direction of the wind and the direction of the piezoelectric patch, was better for the most power extracted by energy harvesting [29]. This angle is usually known as the yaw angle, as shown in Figure 3.2. First, the effect of three different yaw angles, and four different

resistive loads is studied. It is clearly seen that as the velocity of the flow increases, the limit cycle oscillation (LCO) amplitudes at the tip increase as well. Also, the electric power extraction increases since both the velocity of the flow and the resistive load increase. For yaw angles greater than $\beta = 90^{o}$, there has not been found any limit cycle oscillation, so there has not been energy harvested. They found the best results of total energy harvested for their three different yaw angles and four resistive loads. However, in this work, it can be seen again the lack of using an optimization process to get more precise values.

Another factor that affects energy harvesting with piezoelectric materials is the transduced energy, which refers to the conversion of one form of energy into another. This term is usually used to measure the efficiency of the energy harvested from the limit cycle oscillations (LCO). In this work [29], they also studied the transduction efficiency for the different yaw angles in a cantilevered flat beam. It is deducted that the best efficiency configuration of the plate is for the yaw angle $\beta = 0^{o}$. That is explained by the greater LCO amplitude and a bending curvature velocity. That velocity quantifies how fast the curvature of the structure changes as it bends of vibrates. Even though the other yaw angles have higher LCO amplitudes, their flutter modes are based on the first bending and torsion modes, so they provide a lower transduction efficiency.

An additional standout work was the study of the influence of some parameters on the flutter velocity [20]. Changing the center of gravity, it was demonstrated that even though there was a downward trend, it was not significant. With the frequency ratio a similar trend was observed, that is, there was a decrease in the critical velocity, but it was not significant. Finally, for the mass ratio there was a gradual increase of the flutter velocity, so it would become difficult to reach flutter in small-density air. However, they

studied the effect of these parameters on the flutter velocity individually. The study was not dependent on the other parameters.

More recently, an interesting work was made by Deweese [30], a university colleague who presented an innovative aerodynamic energy harvesting. Energy was extracted from the limit cycle oscillations (LCO) in a modified Glauert profile. The inviscid unsteady thin airfoil theory was used for modeling structural analysis. The most important part of this study was the implementation of synthetic jet actuators, which were used to control air flow separation. This is a remarkably interesting paper because even though the energy harvested was doubled due to the SJAs, optimization of the parameters was left for future work.

Another similar study was applying energy harvesting with piezoelectricity to military aircraft, concretely unmanned air vehicles (UAV) [31]. The advantage of these airplanes is their flexible wings, which allow higher power generation. Finally, they studied the effect of the resistance load, but again they did not perform an optimization process beyond studying four different values independently.

In summary, despite the progress made, there are still gaps in the understanding of how to optimize specific variables, such as airfoil geometry, or structural parameters, to maximize the energy obtained. This thesis seeks to fill these gaps and provide a solid foundation for future research.

4. FLUTTER PREDICTIONS AND VERIFICATION

It has already been explained that the piezoelectric patch generates energy when they are squeezed or bent, but how could the patch be squeezed or bent? Here is where aeroelasticity comes, as the piezoelectric patch is going to be compressed and stretched due to the vibrations of the beam on which the beam is attached, produced by the incoming airflow.

4.1 Aeroelasticity

Aeroelasticity constitutes a central core that rests on three basic pillars: structural mechanics, aerodynamics, and fluid dynamics. Studying an aeroelastic phenomenon involves analyzing the interaction between the structural deformations and the aerodynamic loads dependent on it. Modern aircraft structures are not completely rigid, so aeroelastic phenomenon occurs when structural deformations induce changes in aerodynamic forces. The additional aerodynamic forces lead to an increase in structural deformations, which in turn lead to higher aerodynamic forces. These interactions may gradually become smaller until an equilibrium condition is reached, or they may diverge catastrophically.

Aeroelasticity can be divided into two fields of study: static aeroelasticity and dynamic aeroelasticity. Static aeroelasticity addresses the interaction between elastic and aerodynamic forces without considering mass properties. On the other hand, dynamic aeroelasticity studies the interaction between aerodynamic, elastic, and inertial forces.

Every static or dynamic aeroelasticity phenomenon can be classified depending on their forces interaction. That classification, also called the Collar's Diagram, can be observed in Figure 4.1. The vowels at the vertices represent the forces: aerodynamic

forces (A), elastic forces (E), and inertia forces (I). The consonants represent the instabilities: divergence (D), reversal of control (R), flutter (F), and buffeting (B). It can be clearly seen that flutter and buffeting depend on the three forces (dynamic aeroelasticity), while divergence and reversal of control do not depend on the inertia forces (static aeroelasticity).

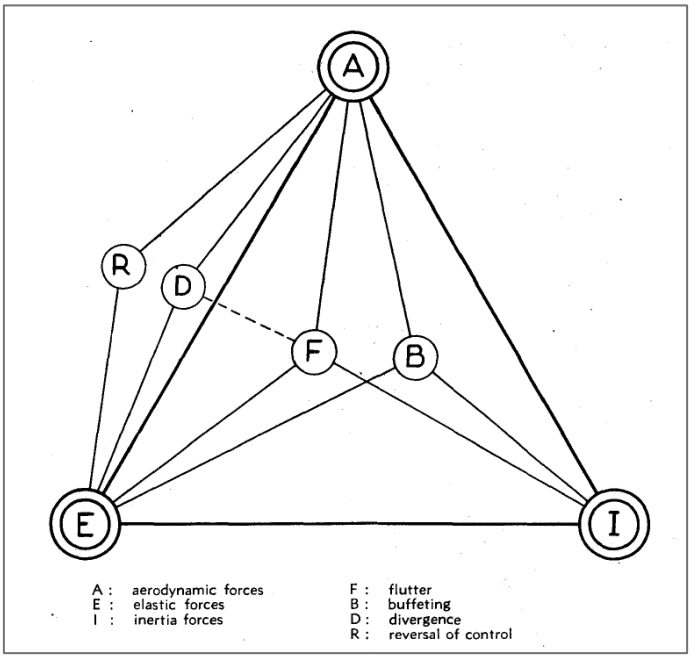

Figure 4.1 Collar's Diagram [32]

Divergence is caused by the fact that the aerodynamic forces, usually produced by an increment of the angle of attack, exceed the elastic forces of the structure and can lead to structural failure . Subjected to aerodynamic loads the aircraft structure deforms so that the aerodynamic loads increase. Upon reaching a critical aircraft speed with respect to the wind, the self-amplified deformation may result in structural failure. Divergence only appears when the aerodynamic axis is forward than the elastic axis.

Reversal of control at high speed caused by structural deformations is one of the major problems for the aircraft at high-speeds. The stability and control properties of the aircraft are highly dependent on the aerodynamic load distribution. By deflecting the aileron, lift increases, but there is also an increase in the nose-down pitch moment of the wing. This decreases the angle of attack, which also decreases lift. So, it finally causes ailerons to start to function in the opposite way that they were purposed.

Flutter occurs when the airfoil starts to oscillate in an unstable way. If the aerodynamic forces extract energy from the system, the system is damped. If they deliver energy to the system then, if the system can dissipate it, it will damp, but if the system cannot dissipate this energy, then the system may become unstable. This phenomenon will be explained in detail in the next section. The most famous flutter example was the Tacoma Narrows Bridge which collapsed in 1940 due to flutter under high-speed wind conditions [33].

Buffeting is a dynamic high-frequencies instability associated with a random forced vibration. Buffeting incidents are not so common as a result of the present development tendency towards cleaner designs. An example of buffeting could be the vortices shed in the wake of a wing which further impact the horizontal stabilizer, leading to an instability.

4.2 Flutter

In the present work, among all the aerodynamics instabilities explained before, the main focus will be set in flutter, which is usually the one that results in a catastrophic structural failure of an aircraft. As have been explained before, flutter is a dynamic instability caused by the interaction of the three different forces: aerodynamic, inertia and

elastic. Flutter could be defined as a self-excited and oscillatory aeroelastic instability in which the inherent modes of vibration of a flexible body combine with its aerodynamic forces to create oscillations that amplify themselves to trigger structural collapse. Further it will be assumed that flutter will not result in a structural failure, but it will indicate the onset of limit cycle oscillations with sustained amplitude of vibrations.

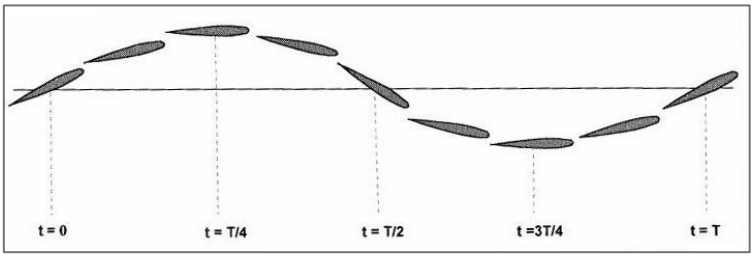

Figure 4.2 Airfoil with flutter motion [34]

In Figure 4.2, it can be observed the basic type of flutter in aircrafts. It starts with a little rotation of the airfoil, $t = 0$. Then, due to the lift the airfoil rises until the torsional stiffness of the structure makes the airfoil to zero rotation, $t = \frac{T}{4}$. After that, due to the bending stiffness the airfoil begins to descend to its original position, but now nose down, $t = \frac{T}{2}$. To reach $t = \frac{3T}{4}$, the same process as before is repeated but with the opposite direction, the airfoil plunges and the torsional stiffness of the structure makes it return to zero rotation. The same happens with the last step, due to the bending stiffness the airfoil begins to descend to its original position, but now nose up, $t = T$.

If an airfoil that is statically stable as it flies under its flutter velocity, is disturbed, the oscillatory motions will damp out in time. The problem is when an airfoil is statically unstable because is above its flutter velocity, so with a small disturbance on it, the

oscillatory motions will increase with exponentially amplitudes, until causing structural collapse. That is why in the present study is crucial to calculate the flutter boundaries, which is going to be explained in chapter 4.4 Flutter analysis.

4.3 Solving methods

When calculating flutter there are some methods that researchers usually use. While some of them only serve to obtain qualitative results, as they need to apply simple hypotheses, other methods can really get quantitative results. In this part the most standardized ones will be explained.

4.3.1 The *p* method

This is one of the oldest methods to determine the flutter boundaries from the complex roots. The letter p refers to the reduced eigenvalue which is a dimensionless parameter $p = \frac{bv}{U}$. This method is usually used to understand flutter with simple airfoil configurations. However, the *p* method is not so accurate when calculating exact values, compared to experimental results. An example of this method is developed in chapter 4.4.2 Steady flow analysis, where the steady-flow theory is applied. Nevertheless, since there is no steady-flow case of flutter in reality, it is needed to introduce other methods to this study.

4.3.2 The *k* method

The k method is one of the most used methods in the industry. This method permits us to obtain flutter results in less time due to its computational agility. This method uses similar equations to the *p* method, but it introduces structural damping. It is important to mention that this structural damping does not exist in the plate, it is artificially introduced to produce the searched motion. The letter k refers to the reduced

frequency $k = \frac{b\omega}{U}$. In the present work, the reduced frequency is used extensively in the Theodorsen's Unsteady Thin-Airfoil Theory, and its function, see Figure 4.3.

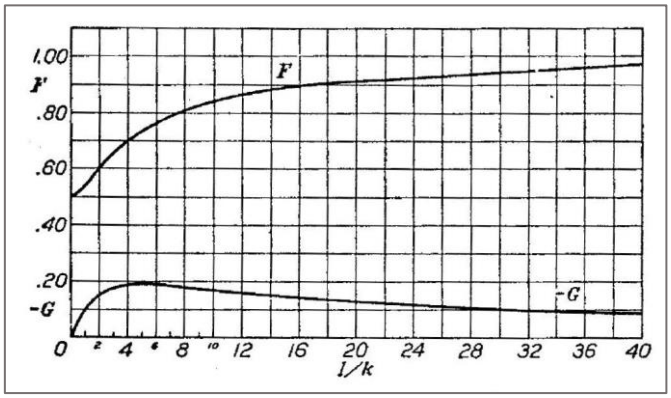

Figure 4.3 Theodorsen function [35]

Theodorsen developed the lift and pitching moment for a flat plate, with simple harmonic oscillations and incompressible flow:

$$L = 2\pi\rho_\infty UbC(k)\left[\dot{h} + U\theta + b\left(\frac{1}{2} - a\right)\dot{\theta}\right] + \pi\rho_\infty b^2\left(\ddot{h} + U\dot{\theta} - ba\ddot{\theta}\right) \qquad (4.1)$$

$$M_Q = -\pi\rho_\infty b^3\left[\frac{1}{2}\ddot{h} + U\dot{\theta} + b\left(\frac{1}{8} - \frac{a}{2}\right)\ddot{\theta}\right] \qquad (4.2)$$

where $C(k)$ is the Theodorsen function

$$C(k) = \frac{H_1^{(2)}(k)}{H_1^{(2)}(k) + iH_0^{(2)}(k)} = F(k) + iG(k) \qquad (4.3)$$

where $H_n^{(2)}(k)$ are second order Hankel functions, $F(k)$ is the real part and $G(k)$ is the imaginary part of the Theodorsen function. There will be an example of the *k* method in 4.4.3 Unsteady flow analysis where these functions of lift and moment will be applied.

4.3.3 The *p-k* method

There are other methods that calculate flutter with more precision than the two studied methods explained above. The limitations of the p method are that only the simplest models can be studied with it, as it needs some assumptions. On the other hand, the k method stands out by its speed, but is not precise enough.

The following study introduced the *p-k* method [36]. This method is the combination of mixing the p and k methods to obtain more precise values of the flutter damping and frequency. The *p-k* method introduces a new complex eigenvalue $p = \gamma k \pm ik$, where γ is the rate of decay and k is the reduced frequency.

In summary, researchers still use the three methods explained above. The most used one could be the k method due to its simplicity and speed of code. However, when precision is needed, the *p-k* method is the best out of the three. The p method is rarely used due to its viability. This study of makes a comparation between the three methods [37].

4.4 Flutter analysis. Verification.

In this part, the model used to estimate flutter velocity of an airfoil will be explained. Then, results will be presented which verified the corresponding code with a couple of previous studies. Firstly, the model geometry is shown in Figure 4.4. The shown airfoil is assumed to be the cross section of a spring restrained, two-dimensional, and rigid-wing plate, mounted on a wind tunnel section.

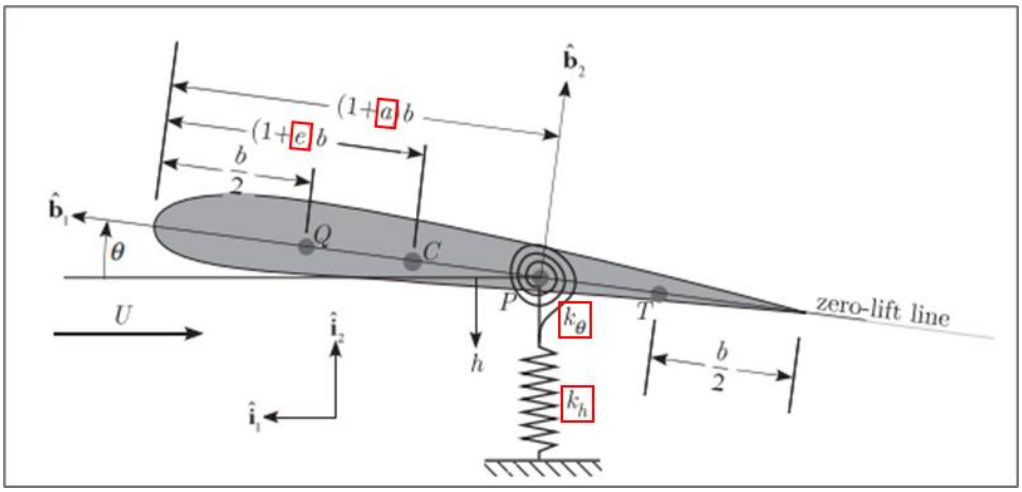

Figure 4.4 Schematic geometry [38]

4.4.1 General equations

The first steps of the section 4.4 Flutter analysis, are understanding Figure 4.4, defining its parameters and developing the general equations to solve the aeroelastic flutter problem.

The main points of interest of this schematic airfoil are the quarter chord point (Q, that in subsonic and with the thin-airfoil theory is also the aerodynamic center), the center of gravity point (C), the elastic axis point (P, the point of reference), and the three quarters chord point (T). It can be observed that it has two degrees of freedom: plunge (h) and pitch (θ). In addition, it is needed to assume that the angles tend to be very small.

Let us start defining some parameters of the airfoil. The model has a determined mass, m, and chord length, $c = 2b$. The two dimensionless parameters are used to calculate the distance from the leading edge to the center of gravity (e), and to the elastic axis (a). Also, it is helpful to define the dimensionless distance between the center of gravity and the reference point as $x_\theta = e - a$. The two springs represent the rigid pitching and plunging with its correspondent spring constants k_θ and k_h, respectively. The

moments of inertia around the center of gravity, and around the reference point are I_C and I_P, being $I_P = I_C + mb^2 x_\theta^2$.

The first step is to calculate the kinetical and potential energy, as well as the generalized forces. The aim is that after these equations, we will be able to solve the Lagrange's equations of motion [39].

In the next equations, it can be seen how to determine the velocities of the elastic axis, the center of gravity, and the quarter-chord points, respectively.

$$v_P = -\dot{h}\hat{y}_1 \tag{4.4}$$

$$v_C = -\dot{h}\hat{y}_1 + \dot{\theta}b(a-e)\hat{y}_2 \tag{4.5}$$

$$v_Q = -\dot{h}\hat{y}_1 + \dot{\theta}b\left(a+\frac{1}{2}\right)\hat{y}_2 \tag{4.6}$$

Kinetic energy can be calculated as

$$K = \frac{1}{2}mv_C^2 + \frac{1}{2}I_C\dot{\theta}^2 = \frac{1}{2}\left(I_P\dot{\theta}^2 + m\dot{h}^2 + 2bx_\theta\dot{h}\dot{\theta}\right) \tag{4.7}$$

Deducing the potential energy is easier

$$P = \frac{1}{2}(k_\theta\theta^2 + k_h h^2) \tag{4.8}$$

Once determined kinetic and potential energy, it is needed to obtain the generalized forces. Looking to the velocity on the quarter-chord point (4.6), the virtual displacement of point Q can be deduced as

$$\partial Z_Q = -\partial h\hat{y}_1 + \partial\theta b\left(a+\frac{1}{2}\right)\hat{y}_2 \tag{4.9}$$

So, the virtual work of the aerodynamic forces, for a static equilibrium, is

$$\overline{\partial W} = \sum_{i=1}^{m} \overrightarrow{F_i}\, \partial \overrightarrow{u_i} + \sum_{j=1}^{n} \overrightarrow{M_j}\, \partial \theta_j$$
$$= \left[-\partial h + \partial\theta b\left(a + \frac{1}{2}\right)\right] L + M_Q \partial\theta \qquad (4.10)$$

and the two forces are

$$Q_h = -L \qquad (4.11)$$

$$Q_\theta = b\left(a + \frac{1}{2}\right) L + M_Q \qquad (4.12)$$

To obtain Lagrange's equation, we need first to define the Lagrangian as a function of generalized coordinates and velocities

$$L_g = L_g\left(h, \theta, \dot{h}, \dot{\theta}\right) = K - P \qquad (4.13)$$

$$Q_j = \frac{d}{dt}\left(\frac{\partial L_g}{\partial \dot{q}_j}\right) - \frac{\partial L_g}{\partial q_j} \qquad (4.14)$$

As the kinetic energy depends on velocities and potential energy depends on positions, equation (4.14) becomes

$$Q_j = \frac{d}{dt}\left(\frac{\partial K}{\partial \dot{q}_j}\right) - \frac{\partial P}{\partial q_j} \qquad (4.15)$$

In this case, j takes only the values of 1 and 2 representing h and θ. So finally, the equations of motion are:

$$m\ddot{h} + mbx_\theta \ddot{\theta} + k_h h = -L \qquad (4.16)$$

$$I_P \ddot{\theta} + mbx_\theta \ddot{h} + k_\theta \theta = M = b\left(a + \frac{1}{2}\right) L + M_Q \qquad (4.17)$$

It is important to take into account that these equations are in the time domain, and in order to solve them, it is necessary to transform them into the frequency domain with the simple harmonic motion characteristic polynomials

$$h(t) = \bar{h}e^{i\omega t}$$
$$\theta(t) = \bar{\theta}e^{i\omega t} \qquad (4.18)$$

where ω is the frequency of the motion. Their derivatives are

$$\dot{h}(t) = i\omega e^{i\omega t}\bar{h}, \quad \ddot{h}(t) = -\omega^2 e^{i\omega t}\bar{h}$$
$$\dot{\theta}(t) = i\omega e^{i\omega t}\bar{\theta}, \quad \ddot{\theta}(t) = -\omega^2 e^{i\omega t}\bar{\theta} \qquad (4.19)$$

4.4.2 Steady flow analysis

In this part, the *p* method is going to be used as explained before in 4.3 Solving methods. The considered model is going to be an airfoil similar to the one shown to Figure 4.4, but with some simplifications. It is going to be based on the steady-flow and thin-airfoil theories, where the lift coefficient is $C_L = 2\pi\theta$, and subsequently

$$L = \frac{1}{2}C_L\rho_\infty AU^2 = 2\pi\theta\rho_\infty bU^2$$
$$M_Q = 0 \qquad (4.20)$$

Here, ρ_∞ is the density of air and U is the airflow speed. Substituting these values in the previous equations of motion (4.16) and (4.17)

$$m\ddot{h} + mbx_\theta\ddot{\theta} + k_h h = -2\pi\theta\rho_\infty bU^2 \qquad (4.21)$$

$$I_P\ddot{\theta} + mbx_\theta\ddot{h} + k_\theta\theta = b\left(a + \frac{1}{2}\right)(2\pi\theta\rho_\infty bU^2) \qquad (4.22)$$

For the next steps it is needed to substitute the time domain variables to the frequency domain, with harmonic solutions as explained at the end of 4.4.1 General equations. In this case, $h = \bar{h}e^{vt}$ and $\theta = \bar{\theta}e^{vt}$, where $v = \frac{pU}{b}$ and p is the complex eigenvalue

$$v^2 m\bar{h} + v^2 mbx_\theta\bar{\theta} + k_h\bar{h} = -2\pi\rho_\infty bU^2\bar{\theta} \qquad (4.23)$$

$$v^2 I_P \bar{\theta} + v^2 m b x_\theta \bar{h} + k_\theta \bar{\theta} = b\left(a + \frac{1}{2}\right)(2\pi\rho_\infty b U^2)\bar{\theta} \qquad (4.24)$$

To simplify the equations, it is convenient to define the uncoupled natural pitch and plunge frequencies, corresponding to the two springs

$$\omega_h = \sqrt{\frac{k_h}{m}}$$
$$\omega_\theta = \sqrt{\frac{k_\theta}{I_P}} \qquad (4.25)$$

In addition, let us introduce some dimensionless parameters, so the equations will become easier

$$\sigma = \frac{\omega_h}{\omega_\theta}$$
$$\mu = \frac{m}{\pi\rho_\infty b^2}$$
$$r = \sqrt{\frac{I_P}{m b^2}} \qquad (4.26)$$
$$V = \frac{U}{b\omega_\theta}$$

Observing above, σ is the ratio of the frequencies explained above, μ is the ratio of mass, r is the radius of gyration about point P, and V is the dimensionless velocity of the airflow, also called reduced velocity. Substituting all these values above into equations (4.23) and (4.24), and taking the terms to the left hand side (LHS)

$$\left(\frac{b^2 v^2}{U^2} + \frac{\sigma^2}{V^2}\right)\frac{\bar{h}}{b} + \left(\frac{b^2 v^2 x_\theta}{U^2} + \frac{2}{\mu}\right)\bar{\theta} = 0 \qquad (4.27)$$

$$\frac{b^2 v^2 x_\theta}{U^2}\frac{\bar{h}}{b} + \left[\frac{b^2 r^2 v^2}{U^2} + \frac{r^2}{V^2} - \frac{2}{\mu}\left(a + \frac{1}{2}\right)\right]\bar{\theta} = 0 \qquad (4.28)$$

It is very intuitive that the last step needed is to substitute the dimensionless complex eigenvalue $p = \frac{bv}{U}$, as for this analysis we are using the *p* method.

$$\begin{bmatrix} p^2 + \frac{\sigma^2}{V^2} & p^2 x_\theta + \frac{2}{\mu} \\ p^2 x_\theta & r^2\left(p^2 + \frac{1}{V^2}\right) - \frac{1}{\mu}\left(a + \frac{1}{2}\right) \end{bmatrix} \begin{Bmatrix} \bar{h} \\ \bar{b} \\ \bar{\theta} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \qquad (4.29)$$

Finally, the equation (4.29) obtained above is called Flutter matrix. Two different approaches can be used to solve these equations: treating them as an eigenvalue problem, or simpler, setting the flutter determinant equal to zero. For this study, flutter determinant solving option is easier and faster, as the airfoil is a simple model.

Once solved, four complex solutions will be obtained. Furthermore, as two of these roots are the conjugates of the other two, in a practical way, there will be only two solutions with the following form

$$p_{1,2} = \frac{bv_{1,2}}{U} = \frac{b}{U}\left(\Gamma_{1,2} \pm \mathrm{i}\Omega_{1,2}\right) \qquad (4.30)$$

As it is more interesting that these eigenvalues are tied to the plunge frequency, instead of the airspeed, they can be multiplied by the reduced velocity

$$Vp_{1,2} = \frac{b}{U}\left(\Gamma_{1,2} \pm \Omega_{1,2}\right)\frac{U}{b\omega_\theta} = \frac{\Gamma_{1,2}}{\omega_\theta} \pm i\frac{\Omega_{1,2}}{\omega_\theta} \qquad (4.31)$$

The two roots can be clearly divided into two parts. The real one is the modal damping, Γ, and the imaginary part is the modal frequency, Ω. Sometimes modal damping is called g, so the graphic between the reduced velocity and modal damping is called *V*-*g* plot. The most important concept of this chapter is how to calculate flutter velocity, and it is going to be done by studying these roots. Firstly, modal damping is considered, as flutter will begin when one of the roots becomes positive. When flutter occurs, regarding

the modal frequency, it can be checked that the two frequencies of plunge and pitch will coalesce, which will result in a couple mode of oscillations. Based on that, flutter velocity can be found when the modal damping is equal to zero.

Now, setting all these equations into matlab, a new code has been created to calculate flutter velocity and plot the modal damping and frequency against the reduced velocity. To verify the code, it has been compared to an example in [38] with the inputs: $a = -0.2, e = -0.1, \mu = 20, r^2 = \frac{6}{25}, \sigma = \frac{2}{5}$, as can be seen in the following figures.

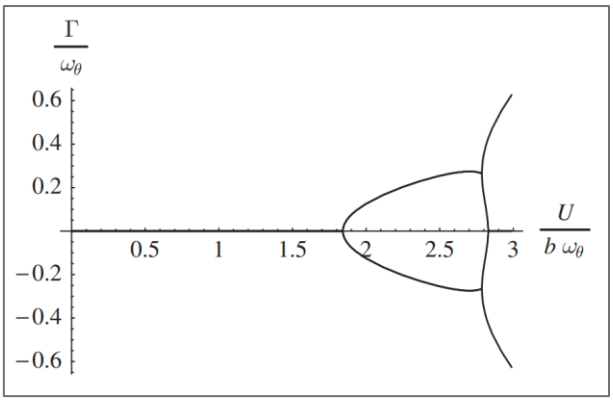

Figure 4.5 Steady modal damping vs V [38]

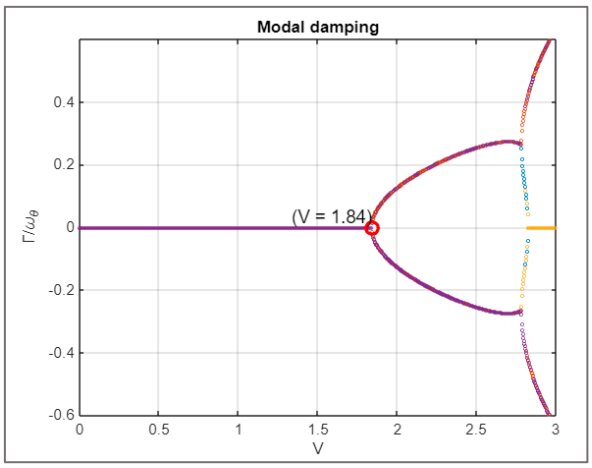

Figure 4.6 Steady modal damping vs V with the study code

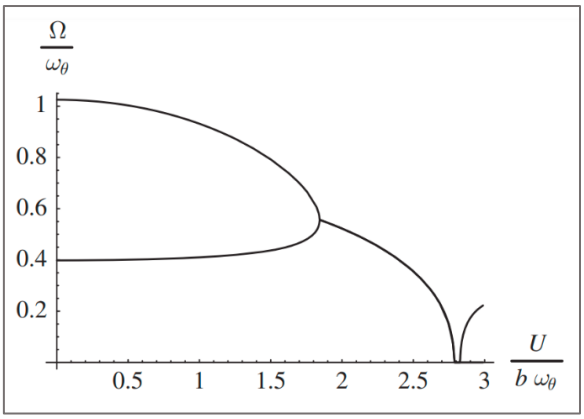

Figure 4.7 Steady modal frequency vs V [38]

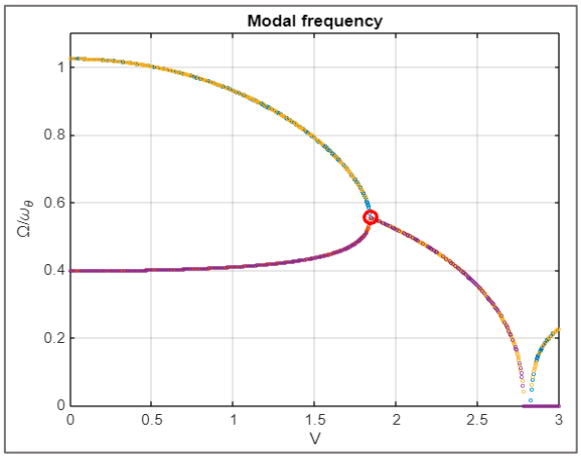

Figure 4.8 Steady modal frequency vs V with the new code

Comparing Hodges book results with those obtained with the new matlab code, it is easy to confirm that the results are very similar. Observing the modal damping in Figure 4.6, it can be figured out that the flutter reduced velocity in the new code is $V_F = 1.84$, almost the same as the Hodges book one is $V_F = 1.843$. That velocity is obtained when one of the roots of the modal damping becomes positive. At this moment, it can be also seen in Figure 4.8 that both modal frequencies from pitching and plunge coalesce. Again, equivalent results are obtained between the new code and Hodges book one, with a flutter modal frequency of $\frac{\Omega_F}{\omega_\theta} = \pm 0.55$. Also, it can also be seen that the reduced divergence

velocity would be around $V_{div} = 2.83$, obtained when the modal frequency is zero. With this configuration, it is correct that $V_F < V_{div}$.

Finally, it can be seen in these figures the deficiencies of the p method and the steady-flow theory. Comparing them to reality, it is impossible for modal damping to be zero before flutter speed. Also, this method incorrectly predicts the two modal damping values to be of the same value at the point of flutter. However, the *p* method is very useful to better understand flutter and its qualitative results. A study with more precise results will be developed in the next section 4.4.3 Unsteady flow analysis.

4.4.3 Unsteady flow analysis

As it has been explained in the last section, there are not many precise results studying flutter and considering the flow to be steady. For that reason, in this section, a two-dimension airfoil will be studied under unsteady flow. The airfoil has two degrees of freedom, pitch and plunge, and its geometry is the same as in Figure 4.4. In this case, the *k* method will be used, as well as Theodorsen's Unsteady Thin-Airfoil Theory.

Firstly, it is needed to recall equations of motion (4.16) and (4.17). As explained in chapter 4.3.2 The *k* method, it is needed to change them, introducing the structural damping

$$m\ddot{h} + mbx_\theta\ddot{\theta} + k_h h - D_h = -L \qquad (4.32)$$

$$I_P\ddot{\theta} + mbx_\theta\ddot{h} + k_\theta\theta - D_\theta = M = b\left(a + \frac{1}{2}\right)L + M_Q \qquad (4.33)$$

where D_θ and D_h are the dissipative structural damping

$$D_h = -i\omega_h^2 g_h m\bar{h}e^{i\omega t} \qquad (4.34)$$

$$D_\theta = -i\omega_\theta^2 g_\theta I_P \bar{\theta} e^{i\omega t} \tag{4.35}$$

and where g_h and g_θ are the damping coefficients. These coefficients are assumed to be very small. In this study, as these coefficients are unknown, we additionally assume that they are the same, introducing the simplification $g_h \approx g_\theta \approx g$.

At this point, while in the steady flow analysis the equations of motion were simpler, in this analysis it is easier working with their left hand sides (LHS) and right hand sides (RHS) separately.

Looking only to the RHS of the equations, the next step is assuming simple harmonic motion, and develop Theodorsen's equations, (4.1) and (4.2), with the characteristic polynomials and derivatives (4.18) and (4.19). In this step the equations will also be nondimensionalized

$$\begin{aligned}\frac{\bar{L}_{RHS}}{mb} = \frac{2\pi\rho_\infty U}{m} C(k)\left[i\omega\bar{h} + \left(U + ib\omega\left(\frac{1}{2} - a\right)\right)\bar{\theta}\right] \\ + \frac{\pi\rho_\infty b}{m}\left[-\omega^2\bar{h} + (i\omega U + \omega^2 ab)\bar{\theta}\right]\end{aligned} \tag{4.36}$$

$$\begin{aligned}\frac{\bar{M}_{RHS}}{mb^2} = \frac{\pi\rho_\infty}{m}\left[-\omega^2 ab\bar{h} + \left(\omega^2 b^2\left(\frac{1}{8} + a^2\right) - i\omega bU\left(\frac{1}{2} - a\right)\right)\bar{\theta}\right. \\ \left. + 2U\left(a + \frac{1}{2}\right)C(k)\left(i\omega\bar{h} + \left(U + i\omega b\left(\frac{1}{2} - a\right)\right)\bar{\theta}\right)\right]\end{aligned} \tag{4.37}$$

The same procedure is repeated simultaneously for the LHS

$$\frac{\bar{L}_{LHS}}{mb} = \left(\frac{\omega^2}{b} - \frac{\omega_h^2}{b} - \frac{i\omega_h^2 g}{b}\right)\bar{h} + \omega^2 x_\theta \bar{\theta} \tag{4.38}$$

$$\frac{\bar{M}_{LHS}}{mb^2} = \frac{\omega^2 x_\theta}{b}\bar{h} + \frac{I_P}{mb^2}\left(\omega^2 - \omega_\theta^2 - i\omega_\theta^2 g\right)\bar{\theta} \tag{4.39}$$

Once developed LHS and RHS of the equations of motion, both sides are equalized and simplified using the dimensionless formulas (4.26), as well as the reduced frequency, $k = \frac{b\omega}{U}$

$$\left[\mu\left(1-\sigma^2\left(\frac{\omega_\theta^2}{\omega^2}\right)(1+ig)\right)+T_{lh}\right]\frac{\bar{h}}{b}+[\mu x_\theta+T_{l\theta}]\bar{\theta}=0 \qquad (4.40)$$

$$[\mu x_\theta+T_{mh}]\frac{\bar{h}}{b}+\left[\mu r^2\left(1-\left(\frac{\omega_\theta^2}{\omega^2}\right)(1+ig)\right)+T_{m\theta}\right]\bar{\theta}=0 \qquad (4.41)$$

where T_{lh}, $T_{l\theta}$, T_{mh}, and $T_{m\theta}$ are the complex functions which represents the Theodorsen's dimensionless aerodynamic lift and moment

$$T_{lh}=\left[1-i2\frac{1}{k}C(k)\right] \qquad (4.42)$$

$$T_{l\theta}=\left[-2\frac{1}{k}C(k)\left(\frac{1}{k}+i\left(\frac{1}{2}-a\right)\right)-i\frac{1}{k}-a\right] \qquad (4.43)$$

$$T_{mh}=\left[-a+i2\frac{1}{k}\left(\frac{1}{2}+a\right)C(k)\right] \qquad (4.44)$$

$$T_{m\theta}=\left[-i\frac{1}{k}\left(\frac{1}{2}-a\right)+\frac{1}{8}+a^2+2\frac{1}{k}\left(\frac{1}{2}+a\right)C(k)\left(\frac{1}{k}+i\left(\frac{1}{2}-a\right)\right)\right] \qquad (4.45)$$

Finally, equations (4.40) and (4.41) can be also written as a matrix form, so the flutter determinant is obtained as has been done for the steady analysis

$$\begin{bmatrix}\mu(1-\sigma^2Z)+T_{lh} & \mu x_\theta+T_{l\theta}\\ \mu x_\theta+T_{mh} & \mu r^2(1-Z)+T_{m\theta}\end{bmatrix}\begin{Bmatrix}\bar{h}\\ \frac{\bar{h}}{b}\\ \bar{\theta}\end{Bmatrix}=\begin{Bmatrix}0\\0\end{Bmatrix} \qquad (4.46)$$

where $Z=\left(\frac{\omega_\theta^2}{\omega^2}\right)(1+ig)$ is the complex eigenvalue to be calculated.

Once the flutter determinant is obtained, there are two different approaches to solve it: setting the flutter determinant equal to zero, or to solve the corresponding eigenvalue problem. Due to its simplicity and speed, the first method is chosen for this analysis.

At this point, there is a problem, there are more unknowns than equations. The unknowns are the frequency ω, the ratio of mass μ, Mach number M_∞,and the speed U. The only equation is the flutter determinant equalized to zero, but as it is a complex number, it represents two equations because both its real and imaginary parts must be zero. For this reason, it is needed to follow an iteration process, as shown in the following flow diagram in Figure 4.9.

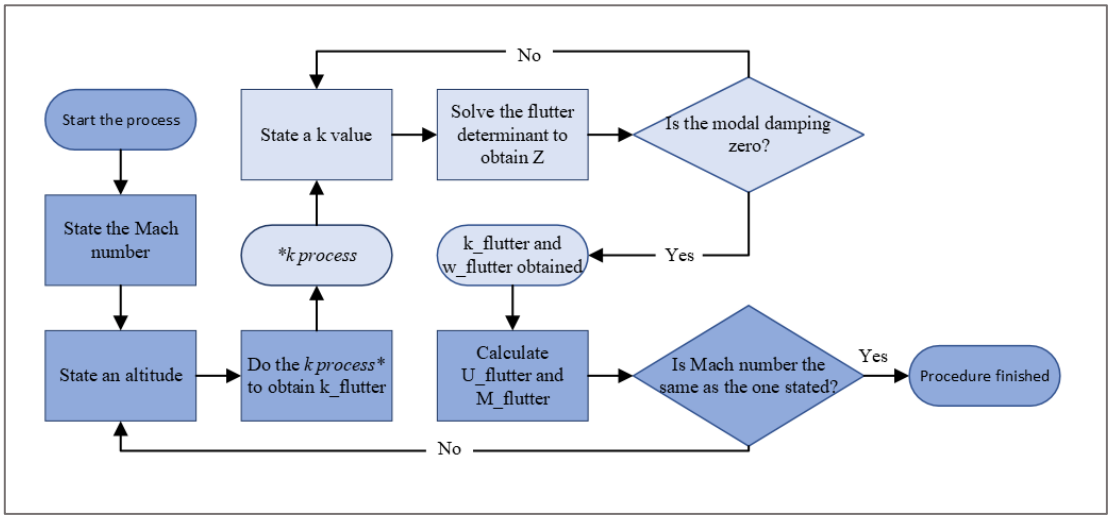

Figure 4.9 Solving procedure flow diagram

Firstly, it is needed to fix the Mach number M_∞. Also, to fix the ratio of mass μ, it is needed to state an altitude. Then, proceed the *k-iteration* to obtain k_F when flutter occurs. For this process, it is needed to iterate the flutter determinant for values of k (from

0.001 to 1 approximately), until one of the roots of the modal damping is zero. At that point, k_F and ω_F are obtained, and flutter velocity is calculated as $U_F = \frac{b\omega_F}{k_F}$. Afterwards, the Mach number at flutter is calculated as $M_{\infty F} = \frac{U_F}{a_\infty}$, and compared to the Mach number guessed in the first step. If they are not the same, it is needed to repeat the steps at another altitude. Finally, when the Mach number coincides between each other, the values of k_F, ω_F, U_F, and μ are valid for that Mach number. The same procedure is repeated for different Mach numbers.

Once solved, the solutions would be of the form where $Z = \left(\frac{\omega_\theta^2}{\omega^2}\right)(1 + ig)$, where the real part would be the modal frequency, Ω, and the imaginary part would be the modal damping, Γ. As has been already said, the key concept of this chapter is how to calculate flutter velocity, and it is going to be done by studying these roots. Again, when one of the two roots of the modal damping becomes positive, flutter will begin, with the value of the modal frequency for this velocity. In conclusion, flutter velocity is found when the modal damping is equal to zero.

Now, setting all these equations into matlab, a new code has been created to calculate flutter velocity and plot the modal damping and frequency against the reduced velocity. To verify the code, it has been compared to an example in [38] with the inputs: $a = -0.2, e = -0.1, \mu = 20, r^2 = \frac{6}{25}, \sigma = \frac{2}{5}$, as can be seen in the following figures.

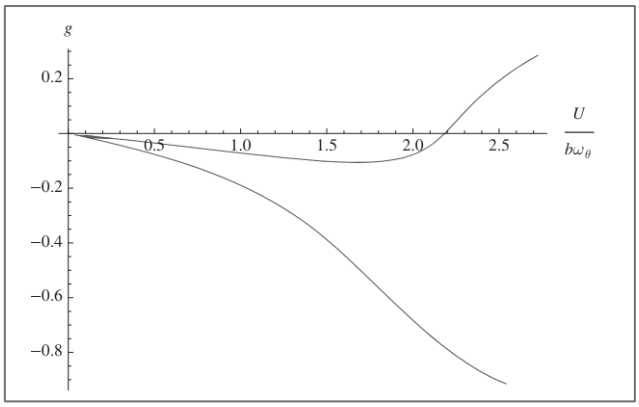

Figure 4.10 Unsteady modal damping vs V [38]

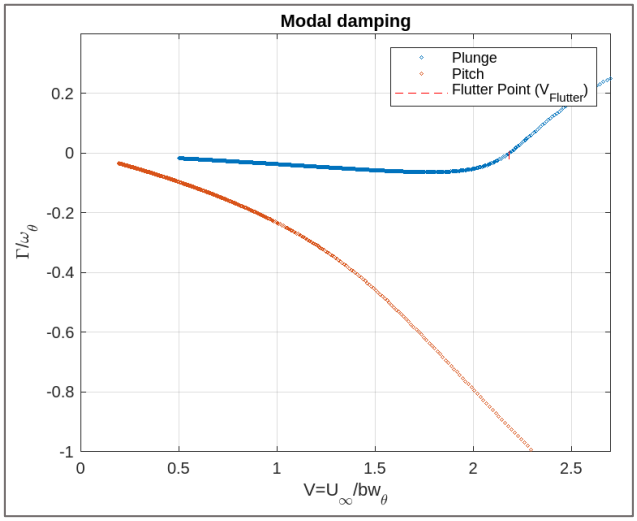

Figure 4.11 Unsteady modal damping vs V with the new code

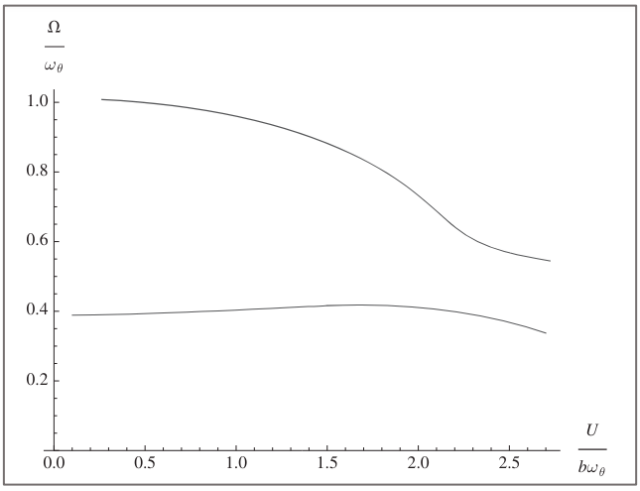

Figure 4.12 Unsteady modal frequency vs V [38]

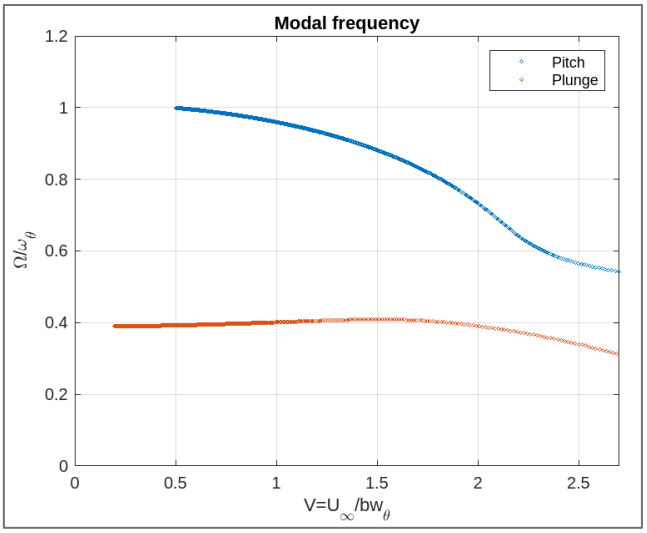

Figure 4.13 Unsteady modal frequency vs V with the new code

Looking at the results obtained with the new matlab code, and comparing with those in Hodge's book, it is easy to confirm that the results are very similar. Observing the modal damping Figure 4.10 and Figure 4.11, it can be figured out that the flutter reduced velocity is approximately $V_F = 2.18$. That velocity is obtained when one of the roots of the modal damping becomes positive. At this moment, it can be also seen in Figure 4.12 and Figure 4.13, that both modal frequencies from pitching and plunge tends to coalesce.

As a final verification of my new code, the results of my new code are compared to the ones obtained in [19]. The inputs are the following: $a = -0.159, e = 0, \mu = 1170.3, r^2 = 0.5, \sigma = 0.783$. The airfoil geometry of Amandolese work can be seen in Figure 4.14. It can be observed that the airfoil is similar to the one I have studied but differs on the distances. Verifying my codes to this work means that the codes I have created are suitable for every generic airfoil geometry problem.

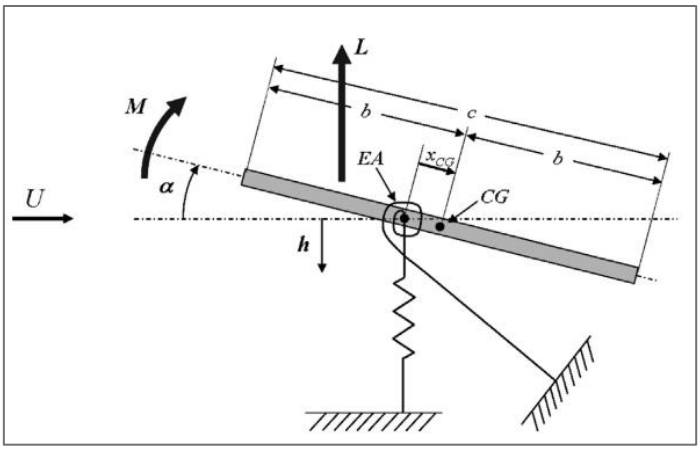

Figure 4.14 Amandolese airfoil geometry [19]

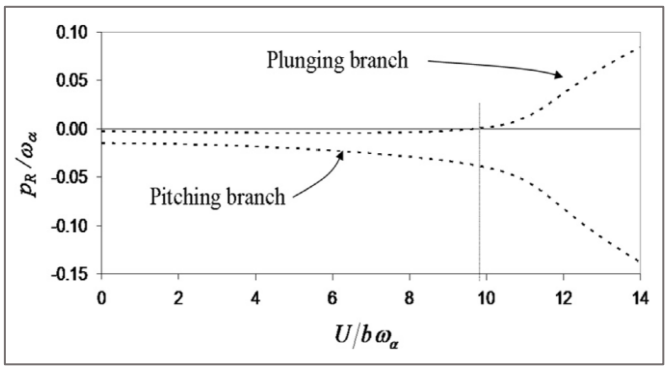

Figure 4.15 Unsteady modal damping vs V [19]

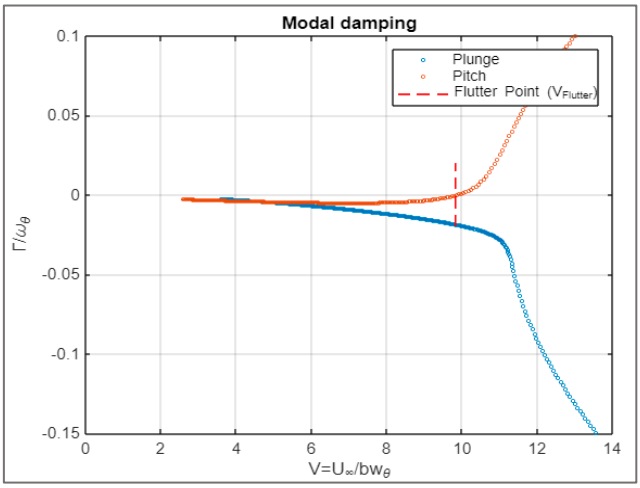

Figure 4.16 Unsteady modal damping vs V with my code

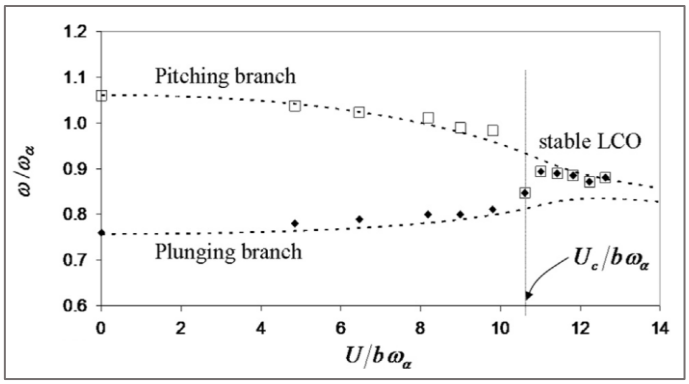

Figure 4.17 Unsteady modal frequency vs V [19]

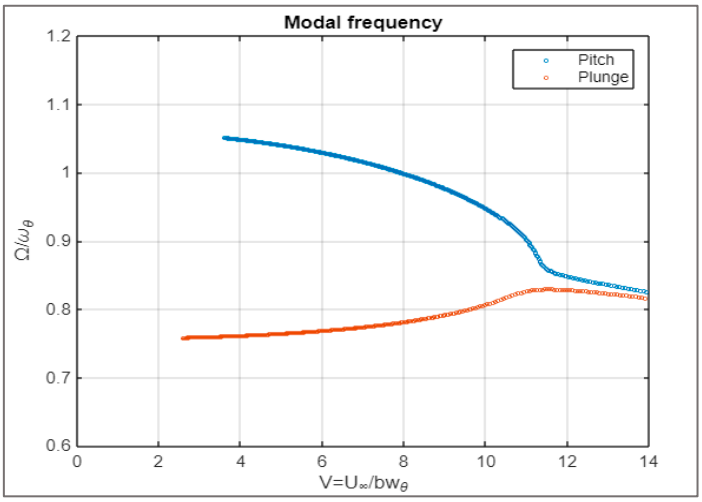

Figure 4.18 Unsteady modal frequency vs V with my code

Looking at the results obtained with my new code and comparing them with those obtained by Amandolese, it can be seen that are very similar. Observing modal damping Figure 4.15 and Figure 4.16, it can be figured out that flutter velocity is approximately $V_F = 9.8$. At this point, looking to Figure 4.17 and Figure 4.18, it can be seen again that both frequencies tend to coalesce. Flutter velocity appears when one of the two roots of the modal damping becomes positive. With those results previously exposed, it can be concluded that my code is finally verified.

In conclusion, during this chapter, the flutter analysis has been explained. Then, general equations of motion and aerodynamic theories have been stated to calculate flutter velocity for an airfoil. After that, two different MATLAB codes have been created to evaluate the flutter velocity of an airfoil for steady and unsteady flow. Finally, both of the codes have been verified with previous studies. That verification implies that the codes I have created during this work are valid for generics problems.

5. PARAMETER VERIFICATION AND OPTIMIZATION

In the field of power generation using piezoelectric materials, there is an abundant literature that has focused on developing models for energy harvesting from structural vibrations, cited in chapter LITERATURE REVIEW. These works have demonstrated the feasibility of using piezoelectric materials to convert mechanical energy from vibrations into electrical energy, offering innovative solutions for applications in aircraft and other systems. However, despite these significant advances, there is a notable lack of studies that focus on the optimization of parameters that directly affect power generation efficiency.

Most existing work is limited to describing how piezoelectric materials can harvest energy from vibrations without considering how to optimize aerodynamic and structural parameters to maximize this energy harvesting. The goal of this work is to investigate the optimum values of the system parameters that will decrease the flutter velocity. In context of energy harvesters that work on the mechanism of generating energy due to onset of flutter or LCO, this means that the energy harvesters will be able to operate at lower air flow speeds.

For these reasons, this chapter will be one of the key parts of my thesis. The optimization code may be a game changer for the study of piezoelectric materials in aeroelastic beams. The optimization code will introduce a new way to measure the effect of the different parameters on the flutter velocity. The main advantage of using an optimization code is its low computational cost compared to other methods.

In this study, flutter velocity will be minimized by a mathematical model that captures the relationship between the parameters, the variables, the constraints, and the objective function. Flutter velocity was selected because of their considerable influence

on the system dynamics, limit cycle oscillations (LCO) and their impact on the energy harvested. Reducing flutter velocity enables getting LCOs in lower air flow conditions.

Concretely, four variables will be studied: ratio of mass μ, the radius of gyration r, and the dimensionless distances from the leading edge to the elastic axis and to the center of gravity, a and e respectively. Looking to Figure 5.1, it can be seen the design space of each variable. These graphs were obtained by varying only one parameter at a time, while keeping the other parameters constant. With that figure, it can be figured out the optimal value of each variable for minimum flutter velocity while keeping the rest of the variables constant. The objective in this last part of the thesis is to optimize all the variables at the same time to get the minimum flutter velocity.

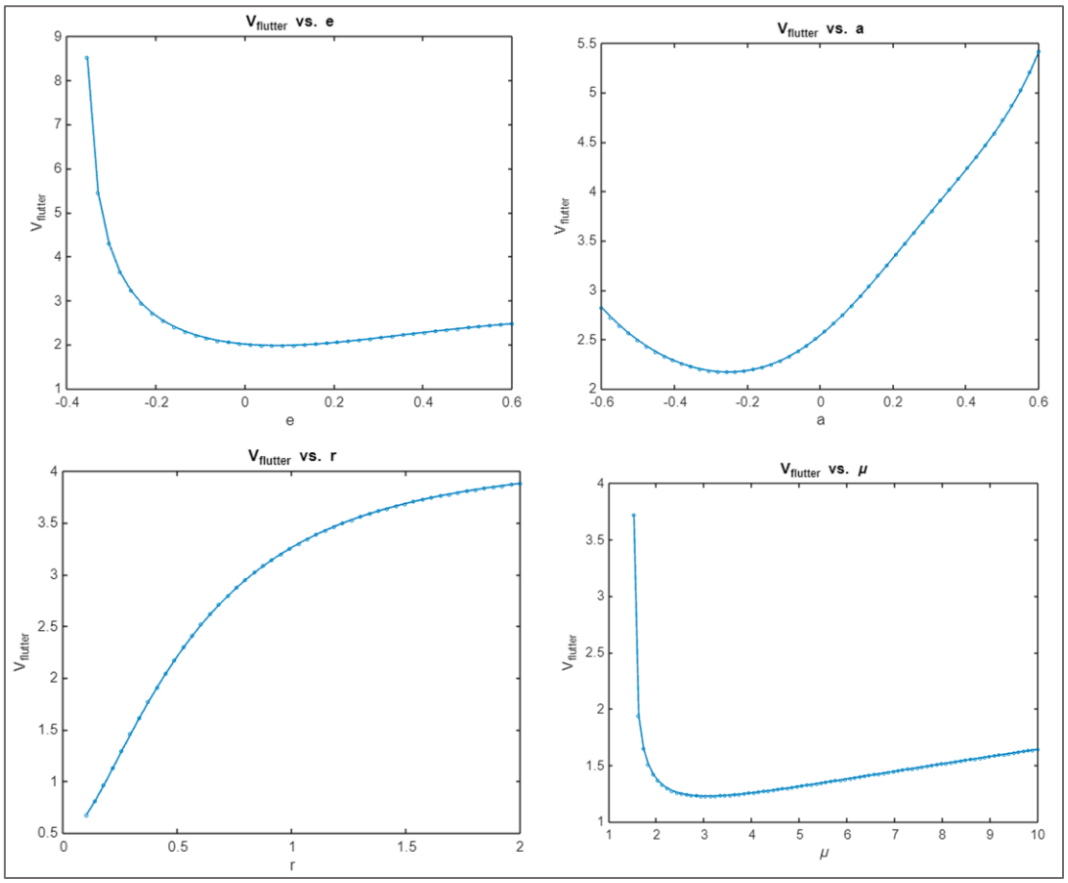

Figure 5.1 Design space of each variable against flutter velocity

5.1 Optimization code

As it has been explained in 1.4 Optimization models, there are a lot of optimization types. For this thesis, nonlinear optimization (NLP) has been chosen due its ability to solve complex and realistic problems with nonlinear relationships. Also, NLP offers flexibility and accuracy, and there are plenty of advanced tools available for its implementation. For this thesis, MATLAB's fmincon function will be used. During these chapters, the main characteristics of fmincon will be explained, and then the optimization will be performed.

5.1.1 Sensitivity analysis

One of the steps needed to start the optimization code is the sensitivity analysis. Sensitivity analysis is a technique used to evaluate how changes in the input parameters of a model affect its optimal solution. In the context of nonlinear optimization using MATLAB's fmincon function, sensitivity analysis can provide valuable information about the robustness and stability of the optimal solution to variations in model variables and parameters.

There are diverse ways to approach the sensitivity analysis: finite difference method, complex step, and discrete analytical method, to name a few. Since MATLAB's fmincon function has been used, it is much easier to implement the finite difference method, which is the default sensitivity analysis method in the fmincon function implementation.

Lagrange multipliers indicate how the objective function would change when relaxing or tightening the constraints, providing information about the sensitivity of the

solution to these constraints. During the optimization, fmincon calculates gradients and Lagrange multipliers, which are useful for sensitivity analysis.

The finite difference method is a numerical technique for estimating the sensitivity of the optimal solution with respect to small perturbations in the variables or parameters. During the optimization, fmincon uses finite differences to calculate the numerical derivatives when analytical derivatives are not provided. This is the method that has been used for sensitivity analysis in this work.

By varying parameters, analyzing Lagrange multipliers, performing perturbations, and using the finite difference method, a deeper insight into how optimal solutions respond to changes in problem conditions can be obtained. This analysis not only helps to verify optimization results, but also provides valuable information for decision making in the presence of uncertainty and variability. However, this area is beyond the context of the present work and may be something that could be investigated in future studies.

5.1.2 Algorithms

MATLAB's fmincon function provides several algorithms to address diverse types of nonlinear optimization problems. Each optimization algorithm in fmincon has its own characteristics that make it suitable for distinct types of problems [40]. The choice of the correct algorithm depends on the nature of the specific problem, including the size of the problem, the nature of the constraints and the need for accuracy in the optimal solution. A brief description of these algorithms is presented below:

Interior-point: The interior-point algorithm is a method that relies on successive approximation within the feasible set of the variables to find the optimal solution. This

algorithm is suitable for large and complex problems due to its ability to manage constraints efficiently. It manages large and nonlinear problems, and it can work with complex constraints, but it can be slower for smaller problems and requires a good initial approximation.

Trust-Region-Reflective: This algorithm sets a confidence region around the current solution and solves subproblems within this region. It is ideal for tightly constrained problems, as it requires less memory, but it may be less efficient for problems with many nonlinear constraints. Also, it could be less robust for large problems.

Active-Set: This is an iterative method that works on the subset of active constraints in each iteration. It is efficient for problems with a limited number of active constraints, as it provides accurate solutions for well-conditioned problems. Nevertheless, it may be inefficient for problems with many active constraints, so it is not suitable for very large problems.

Sequential Quadratic Programming (SQP): This algorithm solves a series of quadratic programming subproblems to approximate the optimal solution of the original nonlinear problem. It is very efficient and robust due its excellent handling of equality and inequality constraints, but it needs more memory and computation time, so it may have convergence issues in highly nonlinear problems.

Along this chapter, all of these algorithms mentioned above will be studied and compared between each other. The objective of doing that is finding the algorithm that best fits the problem stated.

5.1.3 Output function

An Output Function is a user-defined custom function that is called by fmincon at each iteration of the optimization process [41]. This function is an advanced tool that allows to monitor and control the optimization process in MATLAB. Output functions are used to obtain detailed information about the status of the optimization at each iteration, perform real-time analysis, or even interrupt the optimization under certain conditions. It receives information about the current state of the optimization, including the current values of the variables, the objective function, the gradients, and other relevant data. From this information, the output function can perform additional calculations, record data of the optimization process.

One of the pieces of information extracted from the output function is the first order optimality. First order optimality refers to the conditions that a point needs to satisfy to be considered optimal based on the first derivative or gradient of the objective function. Basically, first-order optimality is used to determine how close the current solution is to satisfy the optimality conditions. For unconstrained problems, this gradient should ideally be zero. For constrained problems, it should satisfy the KKT conditions. It is very useful for monitoring the optimization progress. During this study, the first order optimality will be used as a criterion for stopping the iterations, to make a more efficient converging.

Another piece of information that can be obtained from the output function is the norm of step. The norm of step measures the size of the step taken by the variables during an iteration of the algorithm. It is useful for monitoring how the optimization is progressing. If the step size becomes very small, this may indicate that the algorithm is approaching an optimal point. In addition, it can be used as a criterion for stopping the optimization. If the step norm is smaller than a predefined threshold, the algorithm can be considered to have converged to an optimal solution.

Lastly, with the output function, a couple of valuable graphics can be plotted. The first one is the design space, which evaluates the objective function at different values of the variables to find the optimal solution. During this study, the design space will be represented by contour plots that are especially useful for visualizing the behavior of the objective function and verifying the optimization process. On the other hand, the iteration progress graphs are visual representations that show how the objective function changes at each iteration for every single algorithm. These graphs are useful for diagnosing the behavior of the algorithm, evaluating convergence, and understanding how it is approaching an optimal solution.

5.2 One variable optimization

In this part, the MATLAB's fmincon function will be developed with one variable. The fmincon function optimizes a parameter using the objective function, subjected to some constraints while varying some variables.

The variable used in this part is the ratio of mass, μ. Looking to equation (4.26), it can be seen that the ratio of mass depends on the mass, the density of the air or altitude, and the squared half length of the beam. For this reason, the study of flutter velocity depending on the ratio of mass is very important, as we can obtain better or worse results for aircraft flying in a small or high density air.

To verify the optimization model, firstly, is needed to know the minimum value of the flutter velocity in a particular case. Then, this value will be compared to the one found with the optimization model. The example chosen to verify the code, was again from [38] with the following inputs: $a = -0.2, e = -0.1, r^2 = \frac{6}{25}, \sigma = \frac{2}{5}$.

For these values, the unsteady flow analysis has been run for different values of ratio of mass. 100 different values of the ratio of mass from 1 to 10 have been used to calculate flutter speed. That means that 100 iterations have been run to obtain that the minimum flutter speed is $V_F = 1.23$ and its correspondent ratio of mass is $\mu = 3.05$. It can be observed in Figure 5.2 a plot of the values of the flutter speed while varying the ratio of mass, while keeping other parameters constant.

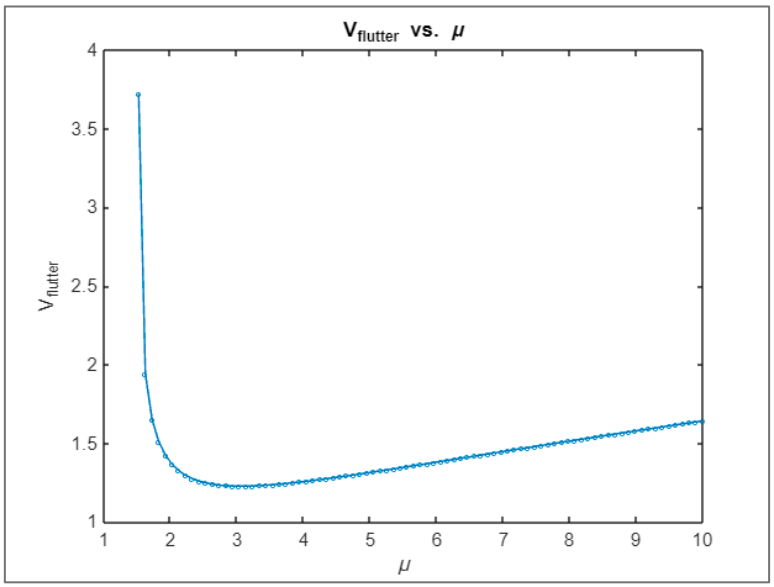

Figure 5.2 Variation of flutter speed depending on μ

Next, the optimization code was executed to obtain the optimum value of mu that gives the minimum flutter speed. It is expected that this result matches the result shown in Figure 5.2. In this case I have used the unsteady flow analysis code as the nonlinear objective function, with some variations. Also, an initial guess, upper and lower boundaries have been set up. Finally, an output function has been developed to record data of the optimization process.

After computing the optimization code for the three different algorithms, it can be clearly seen in Table 5-1that the three results of the flutter velocity and its correspondent optimal μ are very analogous. Also, they are similar to the values previously obtained without using the fmincon function, with the 100 iterations. These results can be verified comparing them to the minimum flutter velocity extracted from Figure 5.2.

Table 5-1 Optimization characteristics of the different algorithms for μ

	Interior-point	**SQP**	**Active set**	**Variation of the variable**
Optimal μ	3.06	3.06	3.06	3.05
Optimal V_flutter	1.2271	1.2271	1.2271	1.23
Number of iterations	10	9	4	100
Time (minutes)	4	4	3	28
First order optimality	9.75E-09	0	1.25E-04	N/A

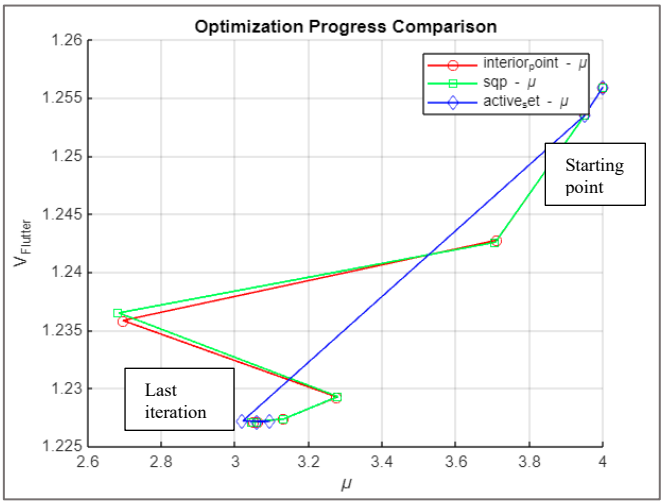

Figure 5.3 Optimization path of μ

Looking to Figure 5.3, it can be observed that the optimization path followed by the interior point and the sqp algorithms are almost the same, while the one followed by the active set algorithm is slightly different. The iterations needed by each algorithm can also be observed.

Also, we can compare these algorithms with their number of iterations and their first order optimality. Looking again to Table 5-1, it can be concluded that active set is the quickest algorithm, as it uses a smaller number of iterations. Also, it can be observed that the worst first order optimality value is obtained from the active set algorithm, as it is the highest value. On the other hand, the best value for first optimality is obtained from the sqp algorithm, as it is the lowest value, closer to zero. In addition, the first order optimality of the three algorithms is very close to zero, so it can be inferred that a correct optimization is reached for the three of them.

Table 5-2 Optimization characteristics path for interior-point algorithm

Iteration	**V_flutter**	**First order optimality**	**Norm of step**
0	1.2559	4.499E-02	N/A
1	1.2536	4.376E-02	4.735E-02
2	1.2427	3.622E-02	2.411E-01
3	1.2358	5.492E-02	1.016E+00
4	1.2292	1.818E-02	5.818E-01
5	1.2273	6.438E-03	1.457E-01
6	1.2270	1.396E-03	8.130E-02
7	1.2270	4.868E-04	1.426E-02
8	1.2270	9.818E-05	3.448E-03
9	1.2270	1.009E-06	7.031E-04
10	1.2270	9.758E-09	4.378E-06

However, using fmincon function, it is not necessary to reach that precision of first order optimality. It can be inferred from Table 5-1, that being conservative, once first order optimality reaches 10^{-3}, there are no significant changes between the following iterations. For that reason, from now on, an optimality tolerance has been implemented to 10^{-3}, so the code will be more efficient while not losing any significant precision. Moreover, it is also possible to check that the norm of step of these iterations is very small, as the flutter velocity does not significantly change. As an example, in this previous analysis, with the interior-point algorithm and being conservative, the optimization code would have stop at only 7 iterations instead of 10, as seen in Table 5-2.

With this first study, it is easy to extract the main advantage of using the optimization code against iterating the unsteady flow problem 100 times: the computing cost. While it seems to be needed around 100 iterations to obtain an approximate value of the minimum flutter speed, only less than 10 iterations are needed with the optimization code using any of the three algorithms explained before, as observed in Table 5-1. Also, the difference of computational time can be observed. While performing a parametric variation takes almost half an hour, any of the three algorithms converges in less than five minutes. That means that the new code involves an 85% reduction of computational time.

In conclusion, in this part of the thesis, the advantages of using an optimization code have been explained, as well as the optimization code using MATLAB's fmincon function has been created and verified. In addition, it has been demonstrated that the algorithms are too precise. Constraining the first order optimality, the code is more efficient without losing precision.

5.3 Two variable optimization

Once the one variable optimization code has been verified, in this part, a two variable optimization will be developed. Again, the example chosen to verify the code was [38] with the following inputs: $a = -0.2,\ e = -0.1,\ \mu = 20,\ r^2 = \frac{6}{25},\ \sigma = \frac{2}{5}$.

In this part, there will be performed several optimization cases for different combinations of the variables. The three different algorithms explained in 5.1.2 Algorithms will be compared for each case.

Similarly to the one variable optimization, for the first step to verify the code is needed to know the minimum value of the flutter velocity in a particular case, and then compare this value to the one found with the optimization model. Since in this part there are two variables at the same time, if it were wanted the same accuracy of 100 different values of each variable as done with the one variable case, 10000 iterations would be needed. For that reason, in this part only 10 different values for each variable will be calculated, so 100 iterations in total will be done.

5.3.1 Case 1: a and μ

The variables used for this case are the ratio of mass, μ, and the dimensionless distance from the leading edge to the elastic axis point, a. The design space of each variable independently can be seen in Figure 5.1.

	a=-0.6	a=-0.46667	a=-0.33333	a=-0.2	a=-0.066667	a=0.066667	a=0.2	a=0.33333	a=0.46667	a=0.6
mu=1.01	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mu=2.0089	1.4658	1.2542	1.2158	1.3741	1.9913	5.3164	11.102	5.8658	3.7868	3.4648
mu=3.0078	1.5245	1.2721	1.1812	1.2272	1.4384	1.8628	2.391	2.6768	2.7661	3.1206
mu=4.0067	1.6304	1.3499	1.2363	1.2563	1.4198	1.7434	2.1566	2.4796	2.7126	3.2055
mu=5.0056	1.7383	1.436	1.3071	1.3145	1.463	1.7615	2.1516	2.4954	2.7899	3.3558
mu=6.0044	1.8411	1.5213	1.3805	1.3803	1.5232	1.8158	2.2043	2.568	2.9035	3.5186
mu=7.0033	1.9376	1.6035	1.4531	1.4475	1.589	1.8829	2.2778	2.6609	3.0289	3.6812
mu=8.0022	2.0282	1.682	1.5235	1.5141	1.6562	1.9548	2.3597	2.7619	3.1575	3.8396
mu=9.0011	2.1133	1.7568	1.5914	1.5792	1.723	2.0283	2.4448	2.8655	3.2858	3.9931
mu=10	2.1935	1.8282	1.6569	1.6424	1.7888	2.1017	2.5308	2.9696	3.4123	4.1415

Figure 5.4 Flutter velocity for 10 different values of μ and a

Firstly, without using the fmincon function, 100 iterations for 10 different values of μ and a have been done. Looking to Figure 5.4, it can be figured out that the minimum flutter speed is reached at $V_F = 1.1812$, with $\mu = 3.0078$, and $a = -0.3333$.

Table 5-3 Optimization characteristics of the different algorithms for μ and a

	Interior-point	**SQP**	**Active set**	**Variation of the variables**
Optimal μ	2.6506	2.6474	2.645	3.0078
Optimal a	-0.32401	-0.32436	-0.32442	-0.3333
Optimal V_flutter	1.1732	1.1732	1.1732	1.1812
Number of iterations	9	9	8	100
Time (minutes)	9	8	7	26
First order optimality	1.000E-03	9.012E-04	6.710E-04	N/A

After computing the optimization code for the three different algorithms, it can be clearly seen that the three results are very similar, as seen in Table 5-3. These results, with the two variable optimization, begin to differ to the values previously obtained without using the fmincon code with the 100 iterations. **The optimal value of μ has more than a 13% error and the flutter velocity has 1% error compared with those results obtained with fmincon code**. These percentages of error can change depending on the

range of values chosen when performing the parametric variation. In this table there can be also observed the numbers of iterations and the amount of time taken by each algorithm. The reduction of time of any of the three algorithms compared to the parametric variation is significant, **with a 70% reduction of computational time**.

The number of iterations of each algorithm and their first order optimality can be also compared with the two variables optimization. It can be concluded that active set is the quickest algorithm, as it uses a smaller number of iterations. In this case, as the first order optimality has been restricted to 10^{-3}, the three algorithms stop when they get over that value. Finally, the results of the algorithms can be verified comparing them to the minimum flutter velocity extracted from Figure 5.6, the design space for the two variables.

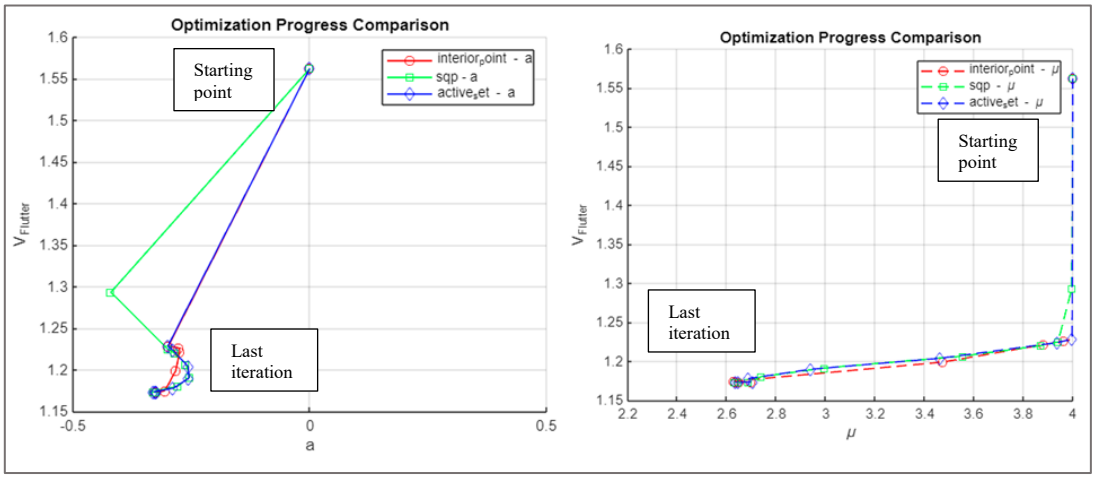

Figure 5.5 Optimization path of μ and a

Looking to Figure 5.5, it can be observed that the optimization paths of the algorithms are very similar too. **The velocity of flutter reduction can be observed, with a 25% decrease from the starting value**. Apart from this figure, it is very intuitive to

see in a graphic how flutter velocity changes while varying the two parameters: μ and a. For that reason, a design space figure has been plotted, as seen in Figure 5.6. It can be observed that the minimum flutter speed is obtained around the bottom left corner of the graphic, with the darkest blue color.

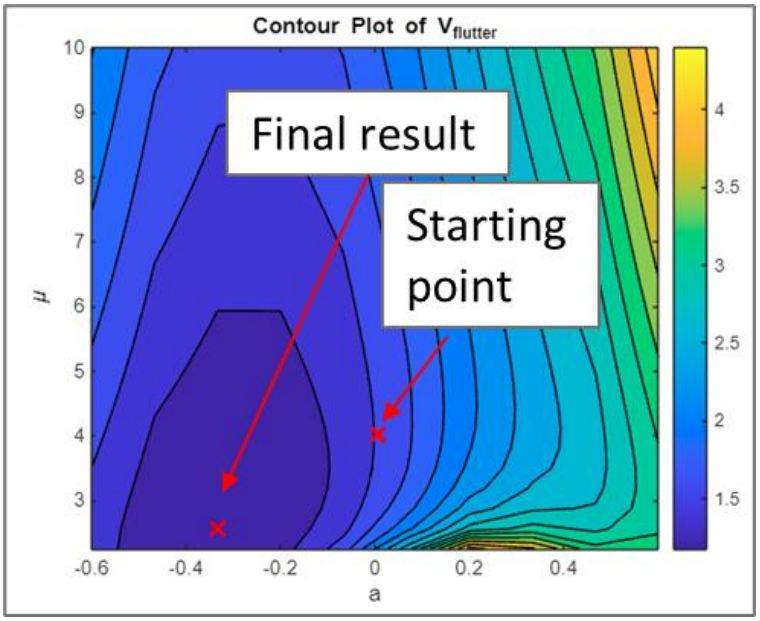

Figure 5.6 Design space for μ and a. The contour colors indicate values of V_flutter

5.3.2 Case 2: r and e

The variables used for this case are the radius of gyration of mass, r, and the dimensionless distance from the leading edge to the center of gravity, e.

Firstly, without using the fmincon function, 100 iterations for 10 different values of r and e have been done. Looking to Figure 5.7, it can be figured out that the minimum flutter speed is reached at $V_F = 1.7547$, with $r = 0.4$, and $e = -0.01111$.

	r=0.4	r=0.57778	r=0.75556	r=0.93333	r=1.1111	r=1.2889	r=1.4667	r=1.6444	r=1.8222	r=2
e=-0.5	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
e=-0.37778	NaN	13.93	9.365	8.8315	8.7544	8.7797	8.8284	8.878	8.9224	8.9605
e=-0.25556	2.898	3.4915	3.9731	4.331	4.5928	4.7857	4.9293	5.0383	5.1225	5.1885
e=-0.13333	1.9675	2.5695	3.0115	3.3296	3.5591	3.7265	3.8508	3.9449	4.0171	4.0737
e=-0.011111	1.7547	2.248	2.6171	2.8834	3.0752	3.2149	3.3183	3.3963	3.4563	3.5031
e=0.11111	1.7742	2.1714	2.4746	2.6947	2.853	2.9682	3.0533	3.1174	3.1666	3.2049
e=0.23333	1.9021	2.2342	2.4896	2.675	2.8083	2.9051	2.9764	3.03	3.0711	3.1031
e=0.35556	2.0603	2.3639	2.5959	2.7638	2.884	2.971	3.0351	2.8695	3.1199	3.1486
e=0.47778	2.2	2.5038	2.7346	2.9012	2.3057	3.1066	3.1701	3.2177	2.3412	2.9122
e=0.6	2.3037	2.6243	2.8686	3.0456	3.0451	3.265	3.0907	3.3842	3.4234	3.4539

Figure 5.7 Flutter velocity for 10 different values of r and e

Table 5-4 Optimization characteristics of the different algorithms for r and e

	Interior-point	**SQP**	**Active set**	**Variation of the variables**
Optimal *r*	0.40007	0.4	0.4	0.4
Optimal *e*	0.03198	0.03181	0.03185	-0.0111
Optimal V_flutter	1.1743	1.1743	1.1743	1.7547
Number of iterations	12	6	3	100
Time (minutes)	10	6	4	25
First order optimality	2.592E-04	2.683E-04	1.85E-01	N/A

Comparing the results with those obtained with the optimization code for the three different algorithms, it can be clearly seen that the three optimized results are very similar, as seen in Table 5-4. These results differ again to the values previously obtained without using the fmincon code with the 100 iterations. **The optimal value of the velocity of flutter has around a 50% error compared with the results obtained with fmincon code**. That means that the flutter velocity is very sensitive to the value of *e*, so the parametric variation cannot give a result as accurate as the optimization process gives. That percentage depends on the range of values chosen for the parametric variation. In addition, it can be observed that active set is the quickest algorithm by far with only 3 iterations. Also, looking to the time taken by the algorithms, it can be clearly observed

that a **significant reduction of more than 70% of the computational time**, compared to the parametric variation. The results of the algorithms can be verified comparing them to the minimum flutter velocity extracted from Figure 5.9, the design space for the two variables.

In this case, it is curious that the active set algorithm has not stopped because of the first optimality order condition imposed before. It has stopped because the objective function was non-decreasing in feasible directions. On the other hand, the slowest algorithm this time was interior point. Finally, it can be seen again that the first order optimality of the other two algorithms is similar, as it was restricted to 10^{-3}.

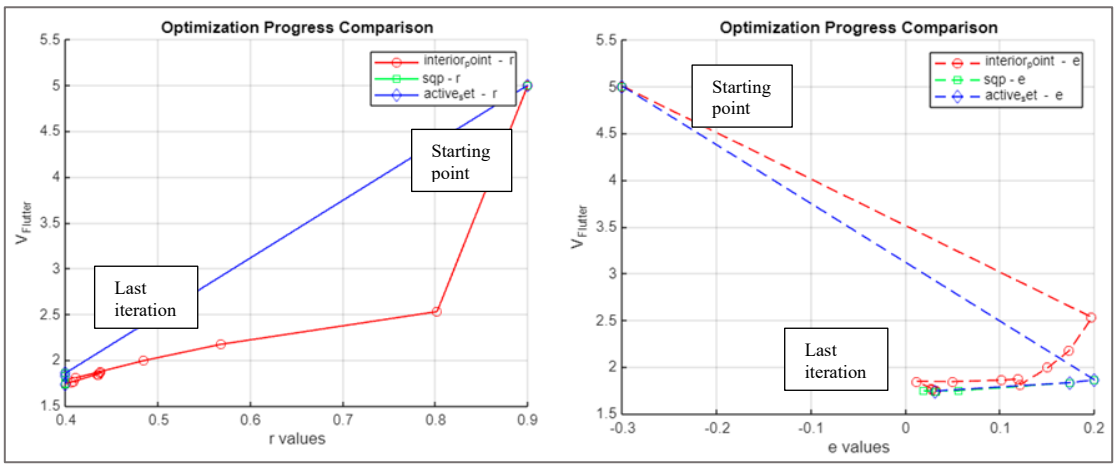

Figure 5.8 Optimization path of r and e

Looking to Figure 5.8, the optimization paths of the algorithms can be observed. This time, there are more differences between the three algorithms: sqp and active set seems to be similar, but the interior-point algorithm takes a different path. **In addition, the velocity of flutter reduction can be observed, with almost an 80% decrease from its starting value**. Additionally, every algorithm converges at the same point. Thus, it

can be said that the form optimization setup is effective for the given situation and may also be used in more complex scenarios involving more than three design variables.

Finally, the design space of the two variables can be seen in Figure 5.9. The graphic represents how flutter velocity changes while varying the two parameters: r and *e.* This graphic is very intuitive, and it can be observed that the minimum flutter speed is obtained around the middle left part of the graphic, with the darkest blue color.

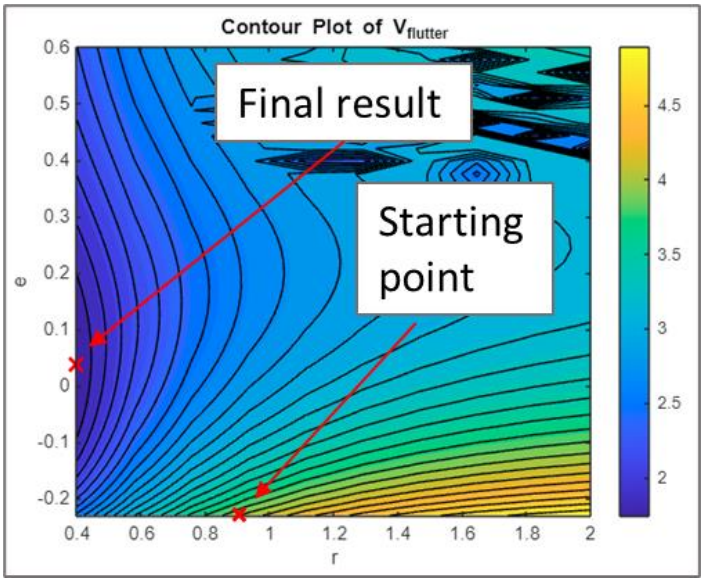

Figure 5.9 Design space for r and *e.* The contour colors indicate values of V_flutter

In summary, with these couple of cases of two variables optimization some conclusions can be extracted. Firstly, it is needed to remark the importance of doing an optimization process instead of trying to guess the optimal values with thousands of iterations. It has been demonstrated that for the three algorithms, the optimization code takes only around 10 iterations to extract the optimal value, while guessing the values for two variables takes at least 100 iterations. **In time values, the codes decrease the computational time by about a 70%**. Also, the optimization codes have demonstrated

much more precision than just iterating to guess the values, even though taking 10 times less iterations. In addition, it has been seen that the three algorithms follow a similar optimization progress path. Finally, the results have been verified with a design space plot, where it has been possible to see that the minimum flutter velocity obtained matched with the values of the two variables.

5.4 Four variable optimization

Another version of the fmincon MATLAB code has been created to optimize the four variables at the same time to get the minimum velocity of flutter. Four variables will be studied: ratio of mass μ, the radius of gyration r, and the dimensionless distances from the leading edge to the elastic axis and to the center of gravity, a and e respectively. The input this time is only the ratio of uncoupled frequencies: $\sigma = \frac{2}{5}$.

Similarly to the preceding optimizations, the objective function used is based on the flutter analysis done through previous chapters. An output function has been implemented for monitoring the information through each iteration. The upper and lower boundaries chosen for the variables can be observed in Table 5-5.

Table 5-5 Variables boundaries

$\boldsymbol{a}$	$\boldsymbol{\mu}$	$\boldsymbol{r}$	$\boldsymbol{e}$
$-0.5 \leq a \leq 0.1$	$1 \leq \mu \leq 10$	$0.4 \leq r \leq 1$	$-0.5 \leq e \leq 0$

The first step would be giving different values to the variables to get a first approximation without using the fmincon function MATLAB code. However, this time is not feasible. For the one variable optimization, 100 different values were giving to the

variable, obtaining 100 different values of the flutter velocity. For the two variable optimization, to obtain these 100 different values of the flutter velocity, only 10 different values were given to the variables, so the results were less precise. For this case, a four variables optimization, to obtain around 100 values of the flutter velocity, it is needed to only give 3 different values to the variables, so the precision would be null. On the other hand, to have better precision and give at least 10 different values to the variables, 10000 iterations would be needed, so that it would last some days with the current computational resource that is being used for this work, which is not practicable. Because of that expensive computational cost explained above, for this case, there would not be a table with the values of flutter velocity depending on the variables.

Table 5-6 Optimization characteristics of the different algorithms for a, μ, r, and e

	Interior-point	**SQP**	**Active set**
Optimal a	-0.19479	-0.19467	-0.19463
Optimal μ	2.4274	2.4268	2.4263
Optimal r	0.40003	0.4	0.4
Optimal e	0.000028	0	0
Optimal V_flutter	0.89369	0.89361	0.89361
Number of iterations	16	12	14
Time (minutes)	22	18	20
First order optimality	1.956E-04	8.465E-04	2.030E-04

After running the fmincon MATLAB code for the three different algorithms, it can be clearly seen that the three results are very similar, as seen in Table 5-6. This time, the minimum flutter velocity of all the previous optimization has been obtained with a value of $V_F = 0.8936$. It can also be observed that the quickest algorithm is sqp while the

slowest is the interior point with a total of 16 iterations. Finally, it can be figured out that the stopping criterion for the three algorithms is the first order optimality, constrained before to 10^{-3}.

Finally, the optimization paths of the algorithms can be observed in Figure 5.10. This time, there are more differences between the three algorithms, especially in the ratio of mass and the dimensionless distance from the leading edge to the elastic axis. SQP and active set seems to be more similar, but the interior-point algorithm takes a different path. In addition, the velocity of flutter reduction can be observed, **with an 80% decrease from the starting value**.

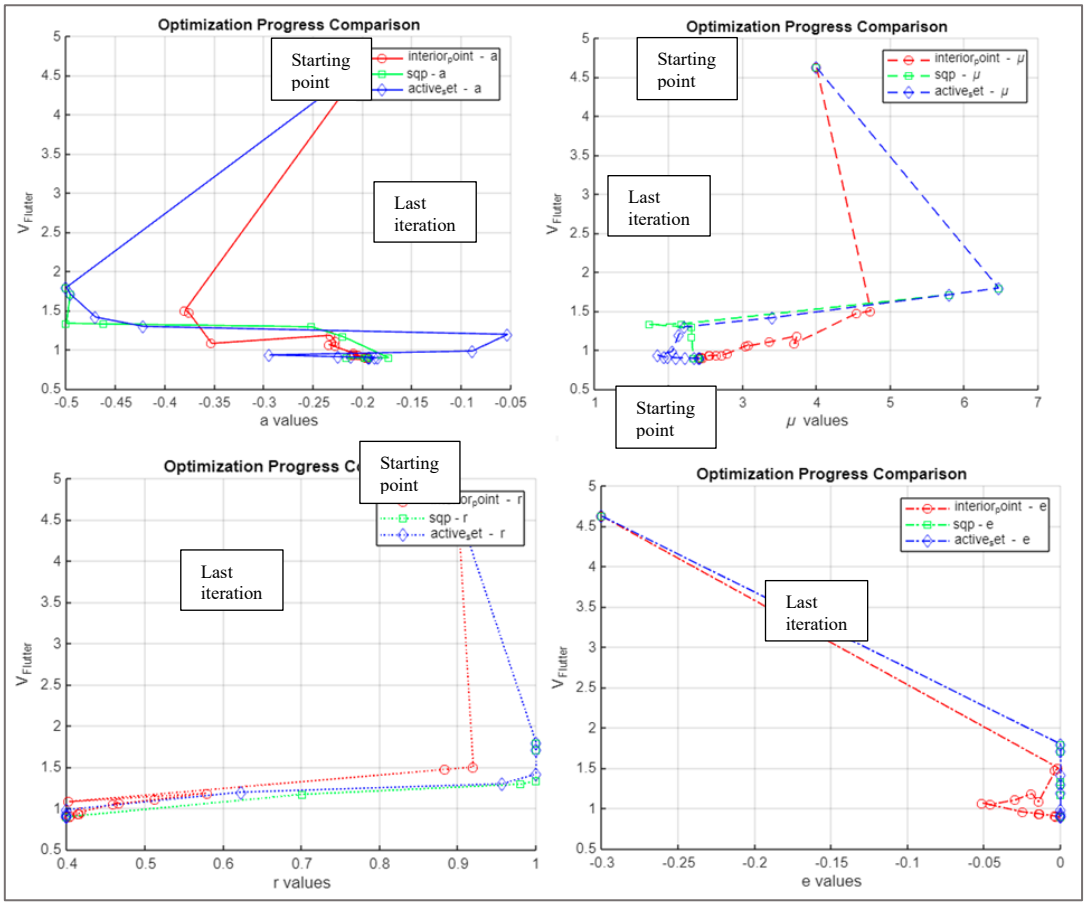

Figure 5.10 Optimization path of μ, a, r, and e

In conclusion, during this chapter, it has been proven that using optimization codes such as fmincon in MATLAB, rather than simply iterating a function many times with a parametric variation, offers numerous significant advantages, especially when managing complex problems with multiple variables and constraints. That is the main objective of my thesis, creating a flutter speed optimization model to open an innovative path for future optimizations of power generation by piezoelectric materials.

One of the main reasons of using an optimization code is its computational efficiency, as fmincon directs the search to the optimal solution efficiently and tends to converge more quickly, rather than performing an exhaustive search that can be computationally expensive. Another key factor is that it offers a variety of algorithms, such as interior-point, SQP, and active-set, which are suitable for diverse types of problems, providing a more robust and accurate solution. Finally, it cannot be overlooked that fmincon has the possibility to work with non-linear functions, as in the case of this thesis problem.

6. CONCLUSIONS AND FUTURE WORK

The main objective of my thesis was creating a flutter speed optimization model to open an innovative path for future optimizations of power generation by piezoelectric materials. The combination of theoretical knowledge in piezoelectricity, aerodynamics, and aeroelasticity with advanced optimization tools has allowed the development of innovative and efficient solutions. The study has been developed in several phases, ranging from the explanation of the fundamental concepts to the creation of advanced optimization codes.

Aircraft wing aerodynamics and the limit cycle oscillations phenomenon are crucial in identifying how and where piezoelectric materials can be effectively applied for power generation. Two codes have been developed to determine the flutter velocity under steady and unsteady flow conditions with the Theodorsen's theory. The codes were verified with a couple of studies, enabling my codes to be suitable for a generic airfoil problem. This knowledge is crucial to identify the conditions under which vibrations can be harnessed for power generation.

An optimization study of four parameters affecting minimum flutter velocity has been addressed: mass ratio, radius of gyration, and dimensionless distances from the leading edge to the elastic axis and to the center of gravity. Using optimization codes, rather than simply iterating a function many times with a parametric variation, offers numerous significant advantages, especially when managing complex problems with multiple variables. The results showed a computational time reduction of around 70% on average for the one and two variable cases, while for the four variable this percentage is expected to be much higher. In the case of the two variables r and e, the final value of flutter velocity obtained with the parametric variation has a 50% error compared to the

optimization codes one. With the last optimization, the four variable case, the flutter velocity has been reduced an 80% from its starting value.

This optimization process is especially relevant because it provides a framework for future studies including design of experiments or even conceptual design of energy harvesters. So far, to the author's knowledge, there were not too many works that follow a detailed optimization process of the parameters involved in piezoelectric power generation. Also, the codes I have created in this work are general enough to include any geometrical or aerodynamic parameters in future research. The methodologies developed here serve as a starting point for future research, enabling a more complete and efficient optimization of energy harvesting systems.

To continue to advance this area of research, several aspects can be considered for future research. It is suggested that, rather than focusing solely on flutter speed, an optimization model can be developed that directly maximizes the energy generated by piezoelectric materials. Then, to provide more realistic results, the study can be done with a more complex airfoil and three dimensional wings, as well as with higher fidelity aerodynamics theories. In addition, the optimization study could be perfected running the code from different starting locations. Finally, validating the developed models and codes through wind tunnel experiments and flight tests will ensure the accuracy and applicability of the simulations and optimizations.

In conclusion, my contribution with this work has been creating an optimization framework to obtain the best values of parameters that will result in the minimum flutter velocity for the system. The methods and results obtained provide a robust basis for future research and applications, opening up new possibilities for energy efficiency in aviation and related fields.

REFERENCES

[1] H. Ritchie, P. Rosado, and M. Roser, "Energy," *Our World in Data*, Jan. 2024, Accessed: Apr. 10, 2024. [Online]. Available: https://ourworldindata.org/energy

[2] S. Chalasani and J. M. Conrad, *A Survey of Energy Harvesting Sources for Embedded Systems - IEEE*, IEEE. 2010.

[3] A. Aabid *et al.*, "Review a review of piezoelectric material-based structural control and health monitoring techniques for engineering structures: Challenges and opportunities," *Actuators*, vol. 10, no. 5, May 2021, doi: 10.3390/act10050101.

[4] M. Górski, A. Jędrzejewska, M. Kozłowski, M. Górski, A. Knoppik-Wróbel, and M. Kozłowski, "Active monitoring with use of smart structures based on high-strength fibre composites as a method of structural elements optimisation," 2013. [Online]. Available: https://www.researchgate.net/publication/257339431

[5] T. Sun, B. Zhou, Q. Zheng, L. Wang, W. Jiang, and G. J. Snyder, "Stretchable fabric generates electric power from woven thermoelectric fibers," *Nature Communications 2020 11:1*, vol. 11, no. 1, pp. 1–10, Jan. 2020, doi: 10.1038/s41467-020-14399-6.

[6] M. Sun, Z. Li, Q. Mao, and D. Shen, "Thermoelectric percolation phenomena in carbon fiber-reinforced concrete," *Cem Concr Res*, vol. 28, no. 12, pp. 1707–1712, Dec. 1998, doi: 10.1016/S0008-8846(98)00161-6.

[7] A. D. Johnson and C. Berkeley, "Non-explosive separation device," 1992

[8] G. Knowles and R. W. Bird, "Telescopic wing system," 2004

[9] J. Curie and P. Curie, “Développement par compression de l’électricité polaire dans les cristaux hémièdres à faces inclinées,” *Bulletin de Minéralogie*, vol. 3, no. 4, pp. 90–93, 1880, doi: 10.3406/BULMI.1880.1564.

[10] P. Molinié and S. Boudia, “Mastering picocoulombs in the 1890s: The Curies’ quartz–electrometer instrumentation, and how it shaped early radioactivity history,” *J Electrostat*, vol. 67, no. 2–3, pp. 524–530, May 2009, doi: 10.1016/J.ELSTAT.2009.01.031.

[11] K. Uchino, “The Development of Piezoelectric Materials and the New Perspective,” *The Pennsylvania State University*, 2017, doi: 10.1016/B978-0-08-102135-4.00001-1.

[12] A. D’Amico and R. Pittenger, “A brief history of active sonar,” *Aquat Mamm*, vol. 35, no. 4, pp. 426–434, Dec. 2009, doi: 10.1578/AM.35.4.2009.426.

[13] T. Hehn and Y. Manoli, “CMOS Circuits for Piezoelectric Energy Harvesters Efficient Power Extraction, Interface Modeling and Loss Analysis,” 2015. doi: 10.1007/978-94-017-9288-2.

[14] S. D. Psoma, P. Tzanetis, and A. Tourlidakis, “A practical application of energy harvesting based on piezoelectric technology for charging portable electronic devices,” *Mater Today Proc*, vol. 4, no. 7, pp. 6771–6785, Jan. 2017, doi: 10.1016/J.MATPR.2017.07.004.

[15] J. Chen, A. Dhanushkodi, and C. Lee Jasper Chen, “Energy harvesting measurements from stall flutter limit cycle oscillations,” *https://doi.org/10.1117/12.2045233*, vol. 9057, pp. 139–147, Apr. 2014, doi: 10.1117/12.2045233.

[16] S. Orrego *et al.*, “Harvesting ambient wind energy with an inverted piezoelectric flag,” *Appl Energy*, vol. 194, pp. 212–222, May 2017, doi: 10.1016/J.APENERGY.2017.03.016.

[17] S. Katzir, “Who knew piezoelectricity? Rutherford and Langevin on submarine detection and the invention of sonar,” *Notes and Records of the Royal Society*, vol. 66, no. 2, pp. 141–157, Jun. 2012, doi: 10.1098/RSNR.2011.0049.

[18] S. Srihari and W. Mwangi, “An Investigation into Energy Generation using Pseudo-Piezoelectric Foam and its Applications in Smart Shoes,” *Beyond: Undergraduate Research Journal*, vol. 7, no. 1, Sep. 2023, Accessed: Jun. 04, 2024. [Online]. Available: https://commons.erau.edu/beyond/vol7/iss1/2

[19] X. Amandolese, S. Michelin, and M. Choquel, “Low speed flutter and limit cycle oscillations of a two-degree-of-freedom flat plate in a wind tunnel,” *J Fluids Struct*, vol. 43, pp. 244–255, Nov. 2013, doi: 10.1016/J.JFLUIDSTRUCTS.2013.09.002.

[20] H. Wang, J. Yi, W. Chen, and Z. Zhou, “Flutter Analysis of Piezoelectric Material Based Smart Wind Turbine Blade,” *International Journal of Acoustics and Vibrations*, vol. 26, no. 3, pp. 240–247, Sep. 2021, doi: 10.20855/ijav.2021.26.31787.

[21] M. Y. Zakaria, M. Y. Al-Haik, and M. R. Hajj, “Experimental analysis of energy harvesting from self-induced flutter of a composite beam,” *Appl Phys Lett*, vol. 107, no. 2, Jul. 2015, doi: 10.1063/1.4926876/29569.

[22] M. Bryant and E. Garcia, “Modeling and testing of a novel aeroelastic flutter energy harvester,” *J Vib Acoust*, vol. 133, no. 1, Feb. 2011, doi: 10.1115/1.4002788/469605.

[23] A. Abdelkefi, F. Najar, A. H. Nayfeh, and S. Ben Ayed, “An energy harvester using piezoelectric cantilever beams undergoing coupledbending–torsion vibrations,” *Smart Mater Struct*, vol. 20, no. 11, p. 115007, Oct. 2011, doi: 10.1088/0964-1726/20/11/115007.

[24] D. Zhu, A. Almusallam, S. Beeby, J. Tudor, and N. Harris, “A Bimorph Multi-layer Piezoelectric Vibration Energy Harvester,” *PowerMEMS*, 2010.

[25] D. J. Inman, C. De Marqui Junior, and A. Erturk, “An electromechanical finite element model for piezoelectric energy harvester plates,” *J Sound Vib*, vol. 327, no. 1–2, pp. 9–25, Oct. 2009, doi: 10.1016/J.JSV.2009.05.015.

[26] S. Lahoti, “SHAPE OPTIMIZATION OF MICROFIBER COMPOSITE ENERGY HARVESTERS,” 2020.

[27] W. G. Abdelrahman, A. Z. Al-Garni, S. I. Abdelmaksoud, and A. Abdallah, “Effect of piezoelectric patch size and material on active vibration control of wind turbine blades,” *Journal of Vibration Engineering and Technologies*, vol. 6, no. 2, pp. 155–161, Apr. 2018, doi: 10.1007/s42417-018-0020-9.

[28] S. Abd El-Maksoud and W. Abdelrahman, “Optimal Active Vibration Suppression of Smart Composite Wind Turbine Blades,” 2014.

[29] D. M. Tang and E. H. Dowell, “Aeroelastic response and energy harvesting from a cantilevered piezoelectric laminated plate,” *J Fluids Struct*, vol. 76, pp. 14–36, Jan. 2018, doi: 10.1016/J.JFLUIDSTRUCTS.2017.09.007.

[30] E. Deweese, “Numerical Study in Wind Energy Extraction from Controlled Limit-Cycle Oscillations in Modified Glauert Airfoil,” 2024. Accessed: Jun. 27, 2024. [Online]. Available: https://commons.erau.edu/edt/805

[31] C. De Marqui, W. G. R. Vieira, and D. J. Inman, “Modeling and Analysis of Piezoelectric Energy Harvesting From Aeroelastic Vibrations Using the Doublet-Lattice Method,” 2011, doi: 10.1115/1.4002785.

[32] A. R. Collar, D. Sc, and F. R. S. Ae, “The Expanding Domain of Aeroelasticity,” *The Aeronautical Journal*, vol. 50, no. 428, pp. 613–636, Aug. 1946, doi: 10.1017/S0368393100120358.

[33] A. Larsen, “Aerodynamics of the Tacoma Narrows Bridge - 60 Years Later,” *Structural Engineering International*, vol. 10, no. 4, pp. 243–248, 2000, doi: 10.2749/101686600780481356.

[34] C. Hebert, D. Cowan, and A. Peter J, “Aerodynamic Flutter,” 1971. [Online]. Available: http://www.va.afrl.af.mil/coe/comp/research/MDC/mdc_images/panel.jpg

[35] T. Theodorsen, “General Theory of Aerodynamic Instability and the Mechanism of Flutter,” Langley Field, Jan. 1949.

[36] C. A. K. Irwin and P. R. Guyett, “The subcritical response and flutter of a swept-wing model,” 1965, Accessed: Apr. 01, 2024. [Online]. Available: https://reports.aerade.cranfield.ac.uk/handle/1826.2/4073

[37] H. J. Hassig, “An approximate true damping solution of the flutter equation by determinant iteration.,” *https://doi.org/10.2514/3.44311*, vol. 8, no. 11, pp. 885–889, May 2012, doi: 10.2514/3.44311.

[38] D. H. Hodges and G. A. Pierce, “Introduction to structural dynamics and aeroelasticity, second edition,” *Introduction to Structural Dynamics and Aeroelasticity, Second Edition*, vol. 9780521195904, pp. 1–247, Jan. 2011, doi: 10.1017/CBO9780511997112.

[39] J. Kim Vandiver, “AN INTRODUCTION TO LAGRANGE EQUATIONS,” 2011. Accessed: Mar. 21, 2024. [Online]. Available: http://ocw.mit.edu

[40] “Seleccionar el algoritmo - MATLAB & Simulink - MathWorks.” Accessed: Jul. 24, 2024. [Online]. Available: https://es.mathworks.com/help/optim/ug/choosing-the-algorithm.html

[41] “Output Function for Problem-Based Optimization - MATLAB & Simulink - MathWorks.” Accessed: Jul. 24, 2024. [Online]. Available: https://es.mathworks.com/help/optim/ug/output-function-problem-based.html