UNIVERSIDADE
BEIRA INTERIOR

Low Speed Airfoil Optimization for Improved Multi-Point Performance of ACC Aircraft
(versão final após defesa)

Guilherme Filipe da Silva Pangas

Dissertação para obtenção do Grau de Mestre em

Engenharia Aeronáutica

(Mestrado integrado)

Orientador: Prof. Doutor Pedro Vieira Gamboa

janeiro de 2025

Declaração de Integridade

Eu, Guilherme Filipe da Silva Pangas, que abaixo assino, estudante com o número de inscrição a41176 do Mestrado Integrado em Engenharia Aeronáutica da Faculdade de Engenharia, declaro ter desenvolvido o presente trabalho e elaborado o presente texto em total consonância com o **Código de Integridades da Universidade da Beira Interior**.

Mais concretamente afirmo não ter incorrido em qualquer das variedades de Fraude Académica, e que aqui declaro conhecer, que em particular atendi à exigida referenciação de frases extratos, imagens e outras formas de trabalho intelectual, e assumindo assim na íntegra as responsabilidades da autoria.

Universidade da Beira Interior, Covilhã 08/01/2025

Dedication

To my family, for their incredible patience and support.
To my friends, for the moments and experiences we've shared on this journey.
And to a very special person, whose persistent support throughout the dissertation allowed me to accomplish this achievement. Without their assistance, this milestone would not have been possible.

Resumo

O avanço das capacidades computacionais permitiu uma análise e iteração mais eficiente do desenho de perfis alares. Consequentemente, tornou-se possível alargar o espaço de desenho e explorar novas geometrias e configurações. No entanto, o estado atual ainda não permite uma otimização "premir e correr". As novas capacidades simplesmente transportam a abordagem tentativa-erro, anteriormente utilizada para a geometria, para a formulação do problema de otimização.

O objetivo deste trabalho é estudar a formulação de um problema de otimização e propor uma nova metodologia que melhor traduza os requisitos da aeronave para o desempenho do perfil alar. O novo objetivo, baseado no Aircargo Challenge 2022, é implementado através da modificação de uma ferramenta existente.

Este programa tem implementado uma otimização multiponto com restrições para melhorar o desempenho do perfil alar da aeronave. A otimização é baseada na técnica de gradiente livre denominada Particle Swarm Optimization (PSO), utilizando a parametrização B-spline e um método de interação acoplado viscoso/invíscido.

A nova função objetivo, adicionada ao programa, estima o desempenho da aeronave desenvolvida para a competição, tal como o peso á descolagem, a velocidade de subida, e a velocidade máxima de cruzeiro e de volta. As estimativas proveem de um método que extrapola as características do perfil alar, analisados através de uma sequência de condições de voo, para o desempenho da aeronave. Uma penalização é depois adicionada à pontuação, caso alguma das restrições impostas não seja cumprida, e a soma usada como valor da função objetivo.

A dissertação inclui dois casos de estudo. No primeiro, o otimizador PSO é avaliado através do efeito que algumas das suas configurações têm na exploração do espaço de resultados e no perfil alar resultante. O estudo concluiu que a opção "exhaustive" obtém os melhores e mais consistentes resultados entre as configurações estudadas. Para além de fornecem uma estimativa da variação do perfil alar e da pontuação entre as diferentes otimizações. Em termos da função objetivo, este caso também revela a tendência para o aumento da carga útil transportada, a fim de obter uma pontuação de voo mais elevada na competição.

No segundo caso de estudo é analisado o comportamento da nova função objetivo com diferentes condições iniciais. Esta investigação revelou as mesmas tendências em termos de pontuação e validou/refutou algumas das decisões da equipa ACC2022.

Palavras-chave

Desenho de perfis alares, Baixo número de Reynols, Otimização Multiponto restrita, Otimização por exame de partículas, Bspline, Método de interação acoplado viscoso/invíscido.

Abstract

The advancement of computational capabilities has allowed for more efficient analysis and iteration of airfoil designs. Consequently, it has become possible to expand the design space and explore new geometries and configurations. However, the current state does not allow for a press-and-run optimization. The new capabilities have simply carried over the trial-and-error approach, previously used for the geometry, to the formulation of the optimization problem.

The goal of this work is to study the formulation of an optimization problem and propose a new methodology that better portrays the aircraft's requirements for airfoil performance. The new objective, based on the Aircargo Challenge 2022, is implemented by modifying an existing tool.

This software has implemented a constraint multi-point optimization to improve the aircraft's airfoil performance. The optimization is based on the free-gradient technique called Particle Swarm Optimization (PSO), using B-spline parametrization and a coupled viscous/inviscid interaction method.

The new objective function, added to this program, estimates the performance of the aircraft developed for the competition, such as lift-off weight, the climb speed, and the maximum cruise and turn velocity. The estimations are done using a method that extrapolates the characteristics of the airfoil, analyzed through a sequence of operating points, into the aircraft's performance. A penalty is then added to the score if any of the restrictions imposed are not met, and the sum is used as the objective function value.

The dissertation includes two case studies. First, the PSO optimizer is evaluated through the effect of some of its settings on design space exploration and the resulting airfoil. The study concluded that the exhaustive option obtains the best and most consistent results among the settings studied. These results also provide an estimation of the airfoil and the score variance across different optimizations. Furthermore, in terms of the objective function, this case also reveals the tendency to increase the payload carried in order to obtain a higher flight score in the competition.

In the second case study, the behavior of the new objective function under different initial conditions is analyzed. This investigation revealed the same trends in terms of scoring and validated/refuted some of the ACC2022 team's decisions.

Keywords

Airfoil design, Low Reynolds Number, Multi-point constrained optimization, Particle Swarm Optimization, B-spline, coupled viscous/inviscid interaction method

Contents

List of Figures

List of Tables

Nomenclature

A	Wing aspect ratio	$-$
A_1	Lift-off rotation factor	$-$
A_{wheel}	Wheel frontal area	m^2
a	Coefficient	$-$
a	Instantaneous acceleration	m/s^2
a_{av}	Average acceleration	m/s^2
a_{te}	Trailing camber line angle	deg
$B_{i,n}$	Bernstein polynomial of degree n	$-$
b_{te}	Trailing wedge angle	deg
$\mathbf{C}$	B-splines coordinates vector	$-$
C	Class function	$-$
C_D	Aircraft drag coefficient	$-$
$C_{D_{induced}}$	Induced drag coefficient	$-$
$C_{D_{misc}}$	Miscellaneous drag coefficient	$-$
$C_{D_{parasite}}$	Parasite drag coefficient	$-$
C_{D_w}	Wing drag coefficient	$-$
C_d	Airfoil drag coefficient	$-$
C_f	Flat-plate skin friction coefficient	$-$
C_L	Aircraft lift coefficient	$-$
$C_{L_{max}}$	Aircraft maximum lift coefficient	$-$
C_l	Airfoil lift coefficient	$-$
$C_{l_{max}}$	Airfoil maximum lift coefficient	$-$
C_m	Airfoil moment coefficient	$-$
c	Wing mean chord	m
c_{le}	Leading edge direction	$-$
c_1	Cognitive thrust factor	$-$
c_2	Social thrust factor	$-$
D	Drag force	N
$D/q_{\,lg}$	Landing gear drag-to-dynamic pressure ratio	m^2
d	Drag force per unit of span	N/m
$dist$	Distance traveled	m
e^n	Transition criterion	$-$

e_w	Wing Oswald efficiency factor	$-$
f_{naca}	Expanded NACA modes	$-$
$\bar{f}_{naca}$	Expanded NACA modes normalized by maximum mode	$-$
h	Height	m
$h_1,\ h_2$	Hicks-Henne functions parameters	$-$
j_c	Curvature of the camber crest	$-$
j_t	Curvature of the thickness crest	$-$
k	Skin roughness	m
k_1	Tangent function coefficient	$-$
k_2	Arctangent function coefficient	$-$
L	Lift force	N
L_{runway}	Runway length	m
l	Characteristic length	m
l	Lift force per unit of span	N/m
Obj	Objective function value	$-$
M	Mach number	$-$
m	Moment per unit of span	N
N_c	Number of constraints	$-$
N_{crit}	Log amplification factor of the most amplified frequency	$-$
N_{DV}	Number of design variables	$-$
$N_{function}$	Number of functions	$-$
$N_{i,k}$	B-splines basis functions	$-$
N_p	Number of airfoil points	$-$
N_{pop}	Population size	$-$
N_1, N_2	Class functions parameters	$-$
n	Load factor	$-$
$\mathbf{P_i}$	Coordinates of discrete control points	$-$
$\mathbf{P_i}$	Particle's best position vector	$-$
PE	Potential energy	J
$PS_{altitude}$	Altitude pre-score	$-$
$p_{d,i}$	Design variable's best position vector	$-$
$prop$	Aerodynamic propriety	$-$
Q	Interference factor	$-$
q	Dynamic pressure ratio	Pa
R	Radius	m

RC_{max}	Maximum rate of climb	m/s
Re	Reynolds number	$-$
Re_{cutoff}	Cutoff Reynolds number	$-$
r_{le}	Leading edge radius	$-$
S	Shape function	$-$
S	Wing surface area	m^2
$S_{altitude}$	Partial score Altitude during the flight	$-$
$S_{distance}$	Partial score Distance during the flight	$-$
$S_{exposed}$	Wing exposed area	m^2
$S_{payload}$	Partial score Payload during the flight	$-$
S_{tail}	Tail surface area	m^2
S_{wet}	Wetted area	m^2
s	Instantaneous position	m
s_G	Ground run distance	m
s_0	Initial position	m
T	Thrust force	N
t	Iteration	$-$
t	Knot position	$-$
t	Time	s
t_{accel}	Acceleration time	s
t/c	Airfoil maximum thickness chord ratio	$-$
t_{extra}	Extra time for acceleration	s
t_{max}	Maximum iterations	$-$
tot_{C_l}	Lift coefficient tolerance for XFOIL initialization	$-$
tot_{C_d}	Drag coefficient tolerance for XFOIL initialization	$-$
V	Airspeed	m/s
$\hat{v}$	Velocity threshold	$-$
$\mathbf{V_i}$	Particle's position rate of change vector	$-$
V_{av}	Average airspeed	m/s
v_c	Minimum velocity threshold	$-$
V_{ck}	Change of the velocity threshold controller	$-$
V_f	Final airspeed	m/s
V_i	Initial airspeed	m/s
$v_{i,d}$	Design variable's position rate of change vector	$-$
V_{LOF}	Lift-off airspeed	m/s

$V_{RC_{max}}$	Horizontal airspeed for best rate of climb	m/s
V_{stall}	Stall airspeed	m/s
V_v	Vertical airspeed	m/s
V_{inf}	Freestream velocity	m/s
W	Weight	N
W_{empty}	Empty weight	N
$W_{payload}$	Payload weight	N
$W_{payload,max}$	Maximum payload weight	N
$W_{payload,min}$	Minimum payload weight	N
$W_{struture}$	Structure weight	N
$W_{systems}$	Systems weight	N
W_{total}	Total weight	N
w	Inertial weight	–
w	Width	m
w_i	Operating point weight	–
$\bar{w}_i$	Operating point normalize weight	–
w_{end}	Final Inertial weight	–
w_{rate}	Inertial weight rate of change	–
w_{start}	Initial Inertial weight	–
$\mathbf{X_c}$	Population design variables centroid vector	–
$\mathbf{X_i}$	Particle's position vector	–
x	Airfoil chordwise direction axis	–
x_{base}	Airfoil transformed x coordinates	–
x/c_m	Airfoil maximum thickness relative position	–
$x_c,\ y_c$	Location of the camber crest	–
$\bar{x}_{c,d}$	Design variable centroid position	–
x_i	Airfoil deformed x coordinates	–
$x_{i,d}$	Design variable's position vector	–
$x_t,\ y_t$	Location of the thickness crest	–
$x_0^c,\ x_1^c,\ x_2^c,\ x_3^c$	Bezier-PARSEC 333 control points x coordinates	–
$xtrb$	Lower surface transition point	–
$xtrt$	Upper surface transition point	–
z	Airfoil vertical direction axis	–
z_{base}	Airfoil transformed z coordinates	–
z_i	Airfoil deformed z coordinates	–

z_{te}	Trailing edge vertical displacement	–
$z_0^c,\ z_1^c,\ z_2^c,\ z_3^c$	Bezier-PARSEC 333 control points z coordinates	–

Greek Letters

α	Angle of attack	deg
δ	Flap deflection	deg
δ_i	Penalty value for ith constraint	–
δ_{total}	Total penalty value	–
δ_{limit}	Limit penalty value	–
Δh	Altitude climbed	m
μ	Air viscosity	Ns/m^2
μ	Ground friction coefficient	–
ν	Kinematic viscosity	$1/(m^2 s)$
ξ_{abs}	Initial absolute perturbation	–
ξ_{rel}	Initial relative perturbation	–
ρ	Air density	kg/m^3
ρ	Scaling factor	–
σ	Constrained property value	–
τ	Lift curve slope correction factor	–
ϕ	Bank angle	deg
φ_i	Aerodynamic objective of ith operating point	–

Subscripts

$cruise$	Level flight
fus	Aircraft fuselage
g	Best overall position
i	ith particle
j	jth particle
lo	Lift-off
lg	Landing gear
ref	Reference value
run	Ground run acceleration
$seed$	Seed airfoil
$tail$	Aircraft tails
to	Take-off

$turn$	Sustained turn

List of Abbreviations

ACC	Aircargo Challenge
BPP	Bezier-PARSEC 3333
CBPE	Current Best Performance Evaluation
CFD	Computational Fluid Dynamics
DCA	Aerospace Sciences Department
DE	Differential Evolution
EA	Evolutionary Algorithm
ES	Evolution Strategy
EP	Evolutionary Programming
FLC	Fuzzy Logic Controller
FPSO	Fuzzy Particle Swarm Optimization
GA	Genetic Algorithm
GP	Genetic Programming
HH	Hicks-Henne
LE	Leading edge
MSE	Mean Square Error
PSO	Particle Swarm Optimization
PSO-IW	Inertial Weight Particle Swarm Optimization
RMSBL	Root Mean Squares of the Boundary Layer
TE	Trailing edge
UBI	Universidade da Beira Interior
UAV	Unmanned Aerial Vehicle

Chapter 1

Introduction

This chapter presents an introduction to the thesis, addressing the formulation of a low speed airfoil optimization problem based on the Aircargo Challenge Competition 2022 aircraft. In the following sections the motivation, the problem statement, the objectives and the thesis outline are presented.

1.1 Context and Motivation

Over the last few years, we have witnessed a rise in the global popularity of Unmanned Aerial Vehicles (UAV). As stated in [3], this upward trend in UAV markets is expected to persist, with projections of growth from USD 14.04 billion in 2022 to USD 77 billion by 2030. The forecast justifies the increased demand for UAVs through the reduction of size and weight, the improvement in reliability, and the integration of advanced sensors that enable a wider spectrum of missions. According to [4], UAVs are a cost-efficient solution across various industries, reducing the manual labor and minimizing the risk of human accidents. The key services provided by UAVs include aerial photography, product delivery, surveillance and inspection, as well as data acquisition and analysis.

In sum, the studies mentioned above depict a product that is integrated into numerous sectors. This fact imposes a challenge for the UAV industry as it needs to meet a fair amount of specific requirements, depending on the application.

In the development of new UAVs, the aerodynamic performance is a key factor for meeting the necessary requirements. This performance, in both fixed and rotary wing vehicles, is widely influenced by the airfoil design. However, as the unique sets of requirements increase with the rising diversity of UAV applications, advancing methods to automate the airfoil design optimization represents an economical path to reach an optimal solution. With a focus on low speed or low Reynolds numbers optimizations for fixed wings UAVs, which represent a considerable part of the market.

It is not difficult to see how the engineering process has changed with the improvement of computational simulations, such as computational fluid dynamics (CFD), to perform aerodynamics analysis. Paper [5] gives a review of how the improvement in computational sim-

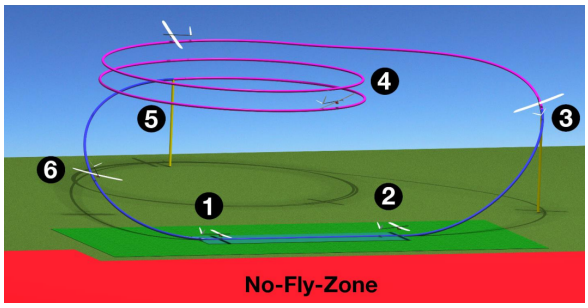
ulations can be combined with evolutionary algorithms (EAs) to create a robust solution for the automation of airfoil optimizations.

However, although the automation of airfoil optimization removes the cut-and-try approach in geometry, this approach is not eliminated from airfoil design. As mentioned in [6], it instead transfers that iterative component to the formulation of the optimization problem. This emphasizes the importance of the optimization problem formulation and the necessity to understand how the problem formulation affects the solution.

1.2 Problem statement

The Aircargo Challenge Competition (ACC) is a biannual competition for engineering students, where the students have only 10 months to design, build, and fly an unmanned aircraft. The rest of the time is devoted to the organizing team, the winner of the previous edition, to develop the set of requirements and rules that the competing teams must follow. This allows for the consideration of new objectives for each edition, hopefully challenging each team to think differently every year.

The 2022 edition of ACC introduces, for the first time, a simulation of a real-life problem as the main goal for the teams to solve. The competition aimed to simulate a medical emergency in a village, cut-off from outside contact due to an avalanche or flood, where it is necessary to transport medical supplies, specifically blood bags. Thus, in this edition, the students had to develop an aircraft that can transport as much cargo as possible, that needs minimum distance to take off, that can quickly attain a safe altitude, and can cover the greatest possible distance in 2 minutes. Also, it must require little space to transport and minimum time to assemble.



1. Take-off
2. Climb
3. After 60 seconds -> Climb assessment over, start of 120 seconds distance flight
4. Fly at will
5. End of distance flight -> Distance assessment over
6. Safe landing -> Payload assessment over

Figure 1.1: Graphic of the flight path.[1]

Outlined in the participation handbook [1], each team needs to design a fixed wing aircraft powered by one of two possible prescribed propellers and a given common electric motor. The aircraft must be remotely controlled by a pilot on the ground using radio con-

trol. In terms of size restrictions, the disassembled aircraft must fit into a transportation box with interior dimensions of 1100 x 400 x 250 mm^3. When fully assembled, it must fit into a rhombus-shaped box with an edge length of 1.5 m each and an arbitrary side angle.

Regarding the flight task, the aircraft has to take-off in a 60 m grass runway with a chosen payload. Following take-off, it must climb to an altitude as close as possible to 100 m within 60 seconds. Immediately after the 60 seconds, it starts the distance task where it needs to travel as far as possible within 120 seconds. Lastly, the aircraft must perform a safe landing.

Following the tradition, the Aerospace Sciences Department (DCA), from Universidade da Beira Interior (UBI), repeated the participation through the development of a composite aircraft with 2.56 meters of wing span and 1.139 meters of length, Figure 1.2.

Figure 1.2: UBI's aircraft from the 2022 edition.

Due to the features of this edition, there is not a single key factor for the design stage to focus on. Rather, the best aircraft will be obtained through a trade-off among different flight stages. This forces the aircraft, of this edition, to achieve an equilibrium between carrying a great amount of payload, flying as fast as possible, and still maintaining a considerable rate of climb.

To achieve this equilibrium, the aerodynamic optimization is one of the key components, as it affects the overall performance of the aircraft and its ability to meet the necessary requirements. Thus, the study of automated tools for the airfoil optimization can be highly beneficial to the team, as the airfoil design plays a significant role in the aircraft optimization. This tool would enable the change of the airfoil performance to better achieve the trade-offs between the different flight stages, thereby enhancing the overall performance in the competition.

1.3 Objectives

The primary goal of this thesis is to research the formulation of an airfoil optimization problem for a low-speed UAV. And then, try to develop a feasible method for optimizing an airfoil during the project's conceptual phase. The proposed approach aims to better translate the key performance parameters into the objective function, thereby providing a more accurate representation of trade-offs between the different flight stages. To achieve this goal, three objectives must be fulfilled:

- Research the PSO optimization method for a deeper understanding of how different settings, such as population size, trust factor, inertia factor, and iterations affect the outcome.
- Modify an existing tool for optimizing airfoils at low speeds by incorporating an objective function based on its impact on the aircraft's performance and additionally, to integrate performance constraints.
- Conduct multiple optimization runs with the implemented tool, varying initial airfoils and performance settings to understand the influence of the starting point on the final outcome.

1.4 Thesis outline

The current thesis comprises five chapters. Chapter 1 introduces the motivation, defines the problem, and outlines the objectives of this study. Chapter 2 provides an overview of the theory behind airfoil design and optimization methods. Chapter 3 details the methodology employed and the modifications made to an existing tool to implement this approach. Chapter 4 discusses the study cases and presents their results. Finally, Chapter 5 summarizes the significant findings of this research and offers suggestions for future work.

Chapter 2

Literature Review

This chapter provides an overview of related works and introduces key concepts for this thesis. The section 2.1 introduces the general concepts of the airfoil design and optimization. The Section 2.2 reviews different works of airfoil optimization and highlights the slight variations between them. Finally, Section 2.3 explores the optimization method employed.

2.1 Introduction to airfoil optimization

The methodology employed, from the beginning, in wing analysis has been to split it into two parts. Studying the cross-section of the wing and then, adapting its performance to account for the finite wing characteristics.

The cross-section of a wing is known as an airfoil, with a standard nomenclature shown in Figure 2.1 according to Raymer [7].

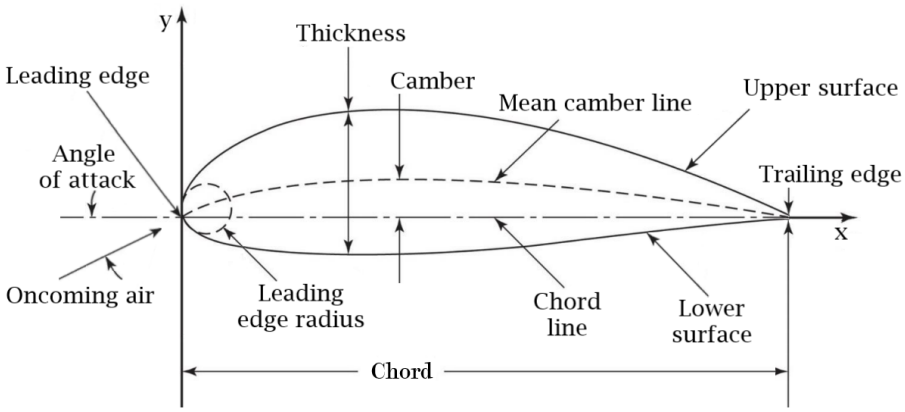


Figure 2.1: Airfoil Nomenclature.

The leading edge is the airfoil's most forward point: the origin of the coordinate system. And, the trailing edge, the rear end of the airfoil. The straight line that connects both edges is defined as chord line, and its length referred as chord. In most situations, the trailing edge is thick, hence the reference axis is located vertically in the middle of that thickness.

The upper surface defines the top curve that connects the leading and trailing edges. And, the bottom curve, the lower surface. The front of the airfoil, when the slopes of the upper and lower surfaces are equivalent to those of the circle, is refer as the leading edge radius.

The thickness distribution of an airfoil is the distance between the upper and lower surfaces measured perpendicular to the chord line. The airfoil thickness is the airfoil's maximum distance value. This value is generally expressed by the ratio t/c, which is the thickness divided by the chord. The camber indicates the characteristic curvature of the airfoil. The mean camber line is the equidistant line from both the upper and lower surfaces.

Finally, the angle of attack is defined as the angle formed between the oncoming air and the chord line.

By changing the velocity of the incoming air over and under itself, the airfoil generates a resultant aerodynamic force and moment per unit of span. The resultant aerodynamic force is divided in two: the lift force, perpendicular to the oncoming air, and the drag force, parallel to the oncoming air.

However, to evaluate and compare the airfoil performance, dimensionless coefficients are used to separate the contributions of air density, freestream velocity, and body size. These coefficients are calculated using the expressions 2.1.

$$C_l = \frac{l}{0.5\rho V_\infty^2 c} \tag{2.1a}$$

$$C_d = \frac{d}{0.5\rho V_\infty^2 c} \tag{2.1b}$$

$$C_m = \frac{m}{0.5\rho V_\infty^2 c^2} \tag{2.1c}$$

The Reynolds number, represented by Equation 2.2, is another relevant criterion to assess the airfoil properties. This dimensionless quantity describes the relative magnitude between inertial forces and viscous friction. And it is significant for the evaluation of the flow separation tendencies and the nature of the flow, as low values indicate laminar flow and high values indicate turbulent flow.

$$Re = \frac{\rho V_\infty c}{\mu} = \frac{V_\infty c}{\nu} \tag{2.2}$$

The last helpful parameter, to the airfoil design, is the Mach number, defined as the ratio of the freestream velocity, V_∞, to the speed of sound a, translated into Equation 2.3. This number quantifies the compressibility effects, which become significant at Mach 0.3.

$$M = \frac{V_\infty}{a} \tag{2.3}$$

The airfoils are designed nowadays to fit the unique requirements of aircraft missions, hence boosting the aircraft's performance. For that, there are a number of design methods, including *direct analysis*, *inverse design*, and *numerical optimization*.

Historically, the direct analysis was the first design method to be employed. In this method, empirical studies were used to determine the aerodynamic characteristics of a predefined airfoil geometry in a flow of specified Reynolds and Mach numbers.

The beginning of computational capabilities allowed the exploration of better approaches, such as the inverse airfoil design method. According to Selig [8], this method involves defining the desired velocity distribution based on boundary-layer characteristics and subsequent performance criteria, from which the airfoil geometry is then calculated. Hence, the inverse design represents a significant advancement towards the control of the desired performance.

The advancement of computational capabilities allowed engineers to analyze and iterate airfoil designs more efficiently. Consequently, it was possible to expand the design space, enabling the exploration of new geometries and configurations. The airfoil design optimization is typically structured as follows:

1. Formulate the problem by defining the objectives, constraints, and initial conditions;
2. Parameterize the initial airfoil geometry to determine the initial design variables;
3. Initial airfoil analysis and performance assessment;
4. Application of the optimization algorithm;
   (a) Generate new design variables;
   (b) Use the design variables to calculate the new airfoil geometry;
   (c) Assess the new airfoil geometry performance;
   (d) Check stop condition. If those conditions are not satisfied, return to 4a.
5. Output the optimal solution.

2.2 Optimization Algorithms

Even though numerical optimizations have a concrete structure, different algorithms undergo minor variations, adjusting to distinct goals and/or conditions. These modifica-

tions occur in four different parts of the algorithm: the objective definition, the optimization method, the flow solver, and the airfoil parametrization.

For the first variation, the objective formulation, the algorithms are divided into two groups: single- or multi-objective. The algorithms that use a single objective have the goal of identifying the single optimal solution for a specified objective function while satisfying any constraint imposed. In contrast, according to Deb [9], a multi-objective strategy finds multiple non-dominant solutions simultaneously, thus depicting the trade-offs between them. This produces a set of pareto-optimal solutions, which must be further processed to get a single solution. A clear distinction between the two techniques is presented in the study [10].

Nonetheless, most of the single-objective optimizations do not use a single operating point, as displayed in the previous study. An example of this is the work done in [11], where a simple concept of average lift/drag ratio over various operational points is employed. This integration of multiple operating points is important to achieve an optimized performance across a broader range of conditions. However, most studies use different weights for each operational point, like [12] and [13], favoring the design point over off-design points. Note that, the relative size of the weights reflects their relevance, and their assignment is one of the operator's major goals. Furthermore, to design an airfoil that is less sensitive to disturbances, a robust optimization problem is introduced in [14], where the objective function combines the mean value and the standard deviation of a property across a sequence of points.

The next modification occurs in the optimization method. The vast majority of the airfoil optimization methods, according to Skinner and Zare-Behtash [15], employ metaheuristic methods. These gradient-free approaches increase the possibility of converging to a global optimum solution since they do not require continuity and predictability over the design space. The two most popular kinds of these techniques are the genetic algorithms (GA) and particle swarm optimization (PSO), which are based on naturally occurring phenomenons. Nowadays, these methods are implemented with some alterations and are explored in more detail in Section 2.3.

A further distinction can be seen in the flow solver used to evaluate the aerodynamic characteristics of airfoils. These models are predominantly selected based on the flow characteristics during flight. For instance, the optimization [10] adopts the coupled boundary layer and potential flow method in the XFOIL solver [16] for investigating airfoils under low Reynolds number conditions. Conversely, the study [17] utilizes a 2D finite volume Navier-Stokes RANS solver [18] to evaluate aerodynamic performance in transonic flow conditions,

aiming to diminish the drag caused by shock wakes. Additionally, [12] and [19] apply the Euler equations within STRUCT2D [20] and the Navier-Stokes equations using the NES code [21], respectively.

The last difference is the airfoil parametrization, a technique used to describe the airfoil shape using a set of parameters that are easily adjustable and reduce the complexity and cost of the optimization process. Some common methods are Hicks-Henne functions, orthogonal NACA functions, B-splines, class shape functions, and Bezier-PARSEC.

The Figure 2.2 provides an overview of the numerical optimization scope. Additional examples can be extracted from the review [5].

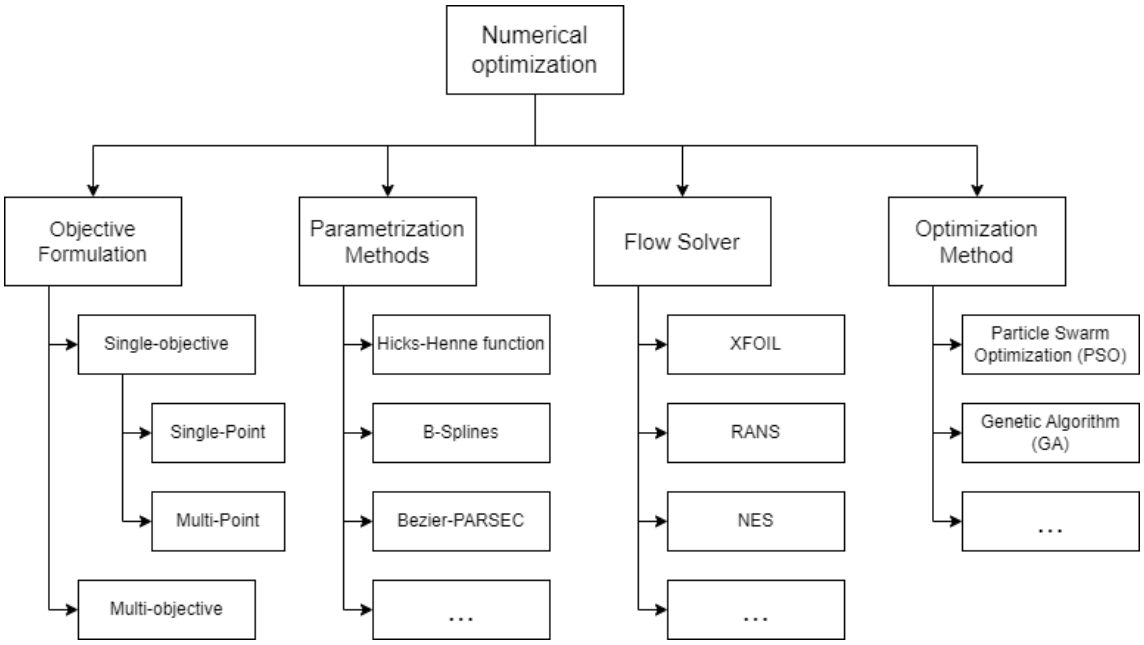


Figure 2.2: Numerical optimization.

2.3 Particle Swarm

Most of the optimization techniques in the field of soft computing research are inspired by the processes of natural evolution. This mimicking of nature has led to the creation of several evolutionary techniques, such as the genetic algorithm (GA) [22], genetic programming (GP) [23], differential evolution (DE) [24], evolution strategy (ES) [25], and evolutionary programming (EP) [26].

These algorithms share a common approach. A population of individuals, each representing a potential solution to the problem, undergoes ”genetic” operations such as mutation, crossover, and reproduction according to the rule of survival of the fittest. The best solution is discovered through the evolution across generations.

2.3.1 Standard Particle Swarm

In 1995, Eberhart and Kennedy introduced a new optimization algorithm that defies convention by simulating the social behavior of animals [27, 28, 29, 30]. This algorithm draws inspiration from the behavior of flocks of birds, hence it received the name particle swarm optimization (PSO). Within PSO, each individual is "evolved" through cooperation and competition among both the members of the swarm and themselves across generations. Each particle adjusts its flight path based on its own flying experience and that of its companions.

Each particle, in the D-dimensional space, is denoted by three vectors:

1. Position, $\mathbf{X}_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,d}, \dots, x_{i,D}\}$ - the position of the ith particle.

2. Velocity, $\mathbf{V}_i = \{v_{i,1}, v_{i,2}, \dots, v_{i,d}, \dots, v_{i,D}\}$ - the rate of position change. The direction and length of the ith particle movement.

3. Personal best, $\mathbf{P}_i = \{p_{i,1}, p_{i,2}, \dots, p_{i,d}, \dots, p_{i,D}\}$ - the best position, in terms of objective value, recorded of the ith particle.

The best overall position within the population is also stored and symbolized as g. The performance of each particle position is assessed according a fitness function, that aligns with the problem at hand. The system of Equations 2.4 expresses how the particles move through the search space, at each iteration t.

$$v_{i,d}(t+1) = v_{i,d}(t) + c_1\, rand_1()\, (p_{i,d}(t) - x_{i,d}(t)) + c_2\, rand_2()\, (p_{g,d}(t) - x_{i,d}(t)) \quad \text{(2.4a)}$$

$$x_{i,d}(t+1) = x_{i,d}(t) + v_{i,d}(t+1) \quad \text{(2.4b)}$$

In the Equation 2.4a, c_1 and c_2 are two positive constants and $rand_1()$ and $rand_2()$ are functions that generates a random number between $[0, 1]$.

Iteratively, the particles "evolved" based on Equation 2.4b. Their new position, $x_{i,d}$, is calculated by adding the new velocity, $v_{i,d}$, to their current position, $x_{i-1,d}$. The velocity, in each iteration, is derived from Equation 2.4b, which is composed of three terms. The second term, $c_1\, rand_1()\, (p_{i,d}(t) - x_{i,d}(t))$, introduces a "cognition" component to the equation, reflecting the particle's own thinking with the difference between the particle's current position and its best position, $(p_{i,d} - x_{i,d})$. The third term, $c_2\, rand_2()\, (p_{g,d}(t) - x_{i,d}(t))$, includes a "social" element encouraging collaboration within the particle swarm. This component is brought into the equation through the distance from the particle's current position to the

population's best position, $(p_{g,d} - x_{i,d})$. Lastly, the first term, $v_{i,d}$, introduces a "flying" component to the equation by passing on the previous particle's velocity to the new one, thus maintaining the tendency to explore the search space, as explain in [30].

It becomes apparent that when the last two terms are excluded, the particles will maintain a steady movement until reaching the boundary. In other words, the particles will have a tendency to expand the search space. On contrast, in the absence of the first term, the particles' behavior is solely defined by their current and global best positions. If particle i initially holds the global best position, it will remain stationary until another particle finds a better position. In this scenario, the search space contracts with each iteration.

To conclude, the first term enhances the global search capability, while the last two terms focus on the local search ability. The trade-offs between the two search abilities are crucial because different problems may require different balances between them.

2.3.2 Particle Swarm with Inertia Weight

Taking this into account, Shi and Eberhart introduced a fresh perspective by incorporating a new inertia weight, w, into Equation 2.4, as displayed in Equation 2.5 [31].

$$v_{i,d}(t+1) = w\, v_{i,d}(t) + c_1\, rand_1()\, (p_{i,d}(t) - x_{i,d}(t)) + c_2\, rand_2()\, (p_{g,d}(t) - x_{i,d}(t)) \quad \text{(2.5a)}$$

$$x_{i,d}(t+1) = x_{i,d}(t) + v_{i,d}(t+1) \quad \text{(2.5b)}$$

The new inertia weight, w, balances both global and local searches, and it might be a positive constant, as well as a positive linear or nonlinear function of time.

Figures 2.3, 2.4, and 2.5, adapted from [2], clearly illustrate the impact of control parameters c_1, c_2, and w on overall performance.

Comparing the cognitive parameter values, $c_1 = 0$ and $c_1 = 3$, it is clear that as c_1 increases, the particle's own best position gains more significance, leading to greater exploration of the design space. Conversely, the comparison of the social parameter values, $c_2 = 0$ and $c_2 = 3$, shows that as c_2 decreases the particles can not communicate, causing them to be trapped in their own best positions.

Finally, as seen in Figure 2.5, the inertia weight w affects the particles flying. A higher inertia weight decreases the dependence on the initial solution and enhances exploration capacity, while a lower inertia weight increases the exploitation capacity.

Furthermore, it is visible that the step sizes of the particles are largely reliant on the con-

trol parameters applied. The control settings must be carefully selected to avoid divergent behavior caused by high particle's velocity values. According to the theoretical study of [32] and [33], the conditions 2.6 will result in convergence behavior.

$$-1 < w < 1 \qquad and \qquad 0 < c_1 + c_2 < \frac{24(1 - w^2)}{7 - 5w} \tag{2.6}$$

Under these conditions, the standard control parameters used in literature, $c = 1.49618$ and $w = 0.7298$ [34], fit the criteria in question. However, the theoretical research conducted to accomplish the Equation 2.6 results from the assumption that the optimization is already stagnated.

2.3.3 Review of different inertia weights

In the optimization search algorithm, it is advantageous to prioritize exploration at the beginning to discover a promising seed. Then, shift towards exploitation to search the area around this seed. As stated before, the inertia weight is an important parameter to control the exploration (global search) and exploitation (local search) by balancing their capabilities.

The goal of this section is to review several kinds of inertial weights, including linear, nonlinear, fuzzy rules, random, and other strategy-based inertia weights, as exposed in [35].

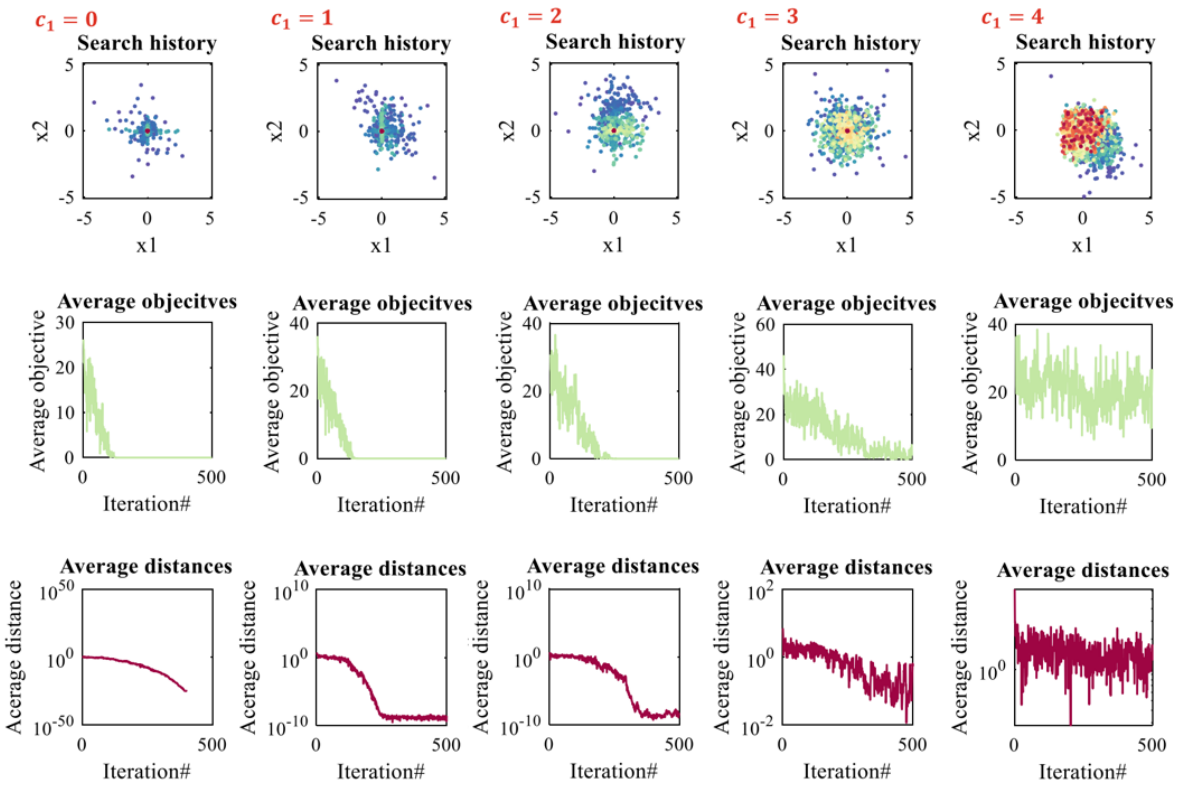


Figure 2.3: The performance of PSO when the cognitive constant is set to 0, 1, 2, 3, and 4. (Adapted from [2])

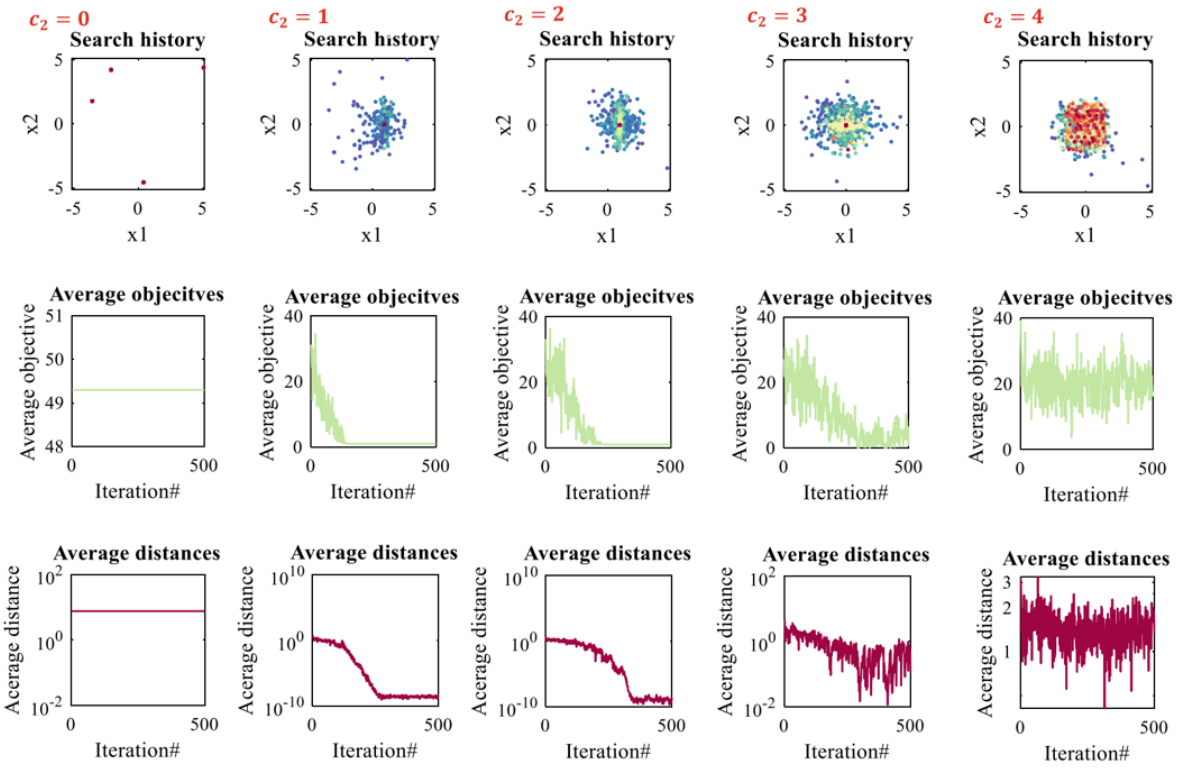


Figure 2.4: The performance of PSO when the social constant is set to 0, 1, 2, 3, and 4. (Adapted from [2])

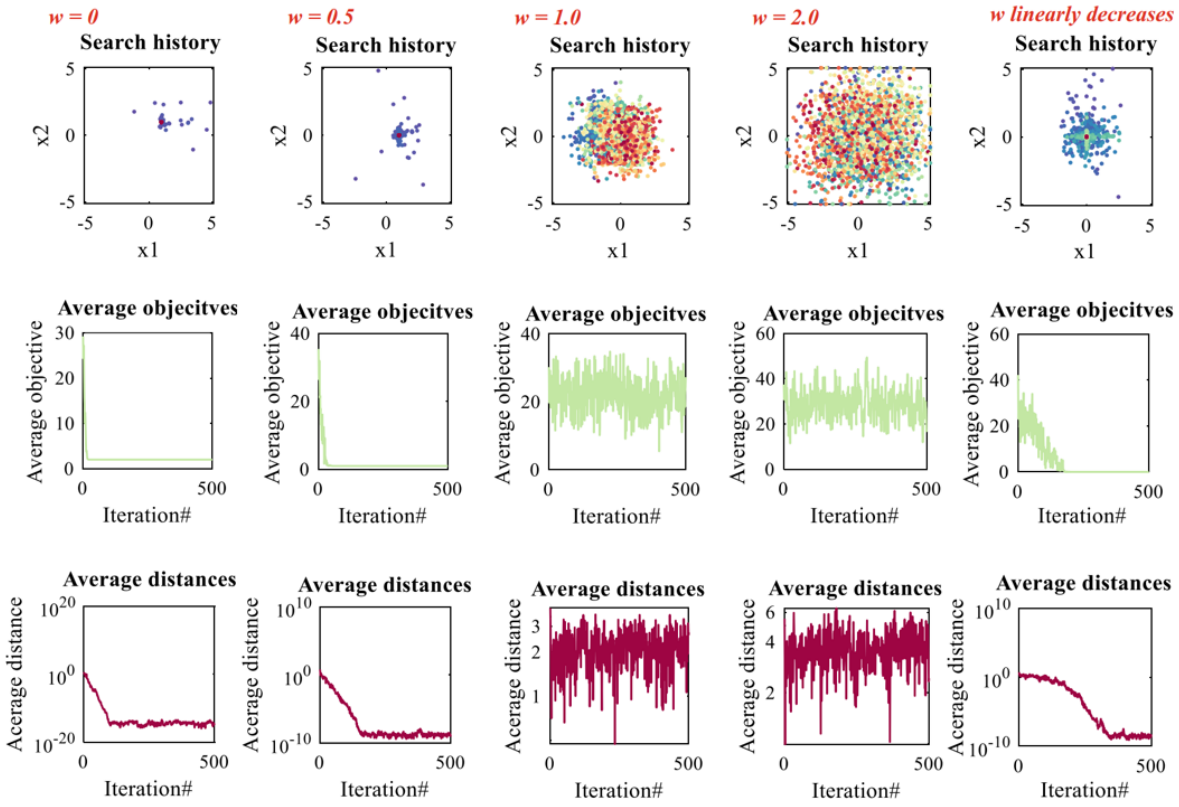


Figure 2.5: The performance of PSO when the inertial weight is set to 1, 0.5, and 0. It also adaptively decreases from 0.9 to 0.2 proportionally to the number of iterations. (Adapted from [2])

2.3.3.1 Linear strategies to adjust inertia weight

Initially, the inertia weight was studied as a constant, as documented by [31]. In this research, it is evident that a larger w contributes more to a global search, while a smaller w facilitates a more local search. To achieve the equilibrium emphasized in Section 2.3.3, the author also proposes a linearly decreasing inertia weight. Under this approach, the value of w linearly decreases from an initial value w_{start} to a final value w_{end} as the number of iterations, t, increases up to the maximum iterations, t_{max}:

$$w(t) = \frac{t_{max} - t}{t_{max}}(w_{start} - w_{end}) + w_{end} \tag{2.7}$$

This new approach reveals an overall improvement in performance, as noted in [31], and in further study [36].

2.3.3.2 Nonlinear strategies to adjust inertia weight

Influenced by the concept of decreasing inertia weight, Chen et al. in [37] studied the impact of two natural exponent inertia weights. This weight function has the capability to converge faster than a linear approach. In this study, the coefficient $a = 10$ in Equation 2.8a and $a = 4$ in Equation 2.8b.

$$w(t) = w_{end} + (w_{start} - w_{end}) \exp\left(-t/\left(\frac{t_{max}}{a}\right)\right) \tag{2.8a}$$

$$w(t) = w_{end} + (w_{start} - w_{end}) \exp\left(-\left[t/\left(\frac{t_{max}}{a}\right)\right]^2\right) \tag{2.8b}$$

Another class of three nonlinear decreasing inertia weight was tested. The class included a parabola opening upwards, a parabola opening downwards and an exponential curve [38]. The simulations indicate that while the concave-based function outperformed the linear function, the convex-based function did not surpass the performance of the linear function .

$$w(t) = -(w_{start} - w_{end}) \left(\frac{t}{t_{max}}\right)^2 + w_{start} \tag{2.9a}$$

$$w(t) = (w_{start} - w_{end}) \left(\frac{t}{t_{max}}\right)^2 + (w_{end} - w_{start}) \left(\frac{2t}{t_{max}}\right) + w_{start} \tag{2.9b}$$

$$w(t) = w_{end} \left(\frac{w_{start}}{w_{end}}\right)^{1/(1+10t/t_{max})} \tag{2.9c}$$

The paper [39] presents two nonlinear inertia weights based on the tangent and arctangent functions. The results of the paper show that the tangent function has a stronger ability to escape local optima in later periods, while the arctangent function exhibits a faster convergence rate.

$$w(t) = (w_{start} - w_{end}) \tan\left[0.875 \cdot \left(\frac{t}{t_{max}}\right)^{k_1}\right] + w_{end} \tag{2.10a}$$

$$w(t) = (w_{start} - w_{end}) \arctan\left[1.56 \cdot \left(\frac{t}{t_{max}}\right)^{k_2}\right] + w_{end} \tag{2.10b}$$

2.3.3.3 Fuzzy rules to adjust inertia weight

Innovating the inertia weight, Shi and Eberhart propose in [40] the idea of a fuzzy PSO (FPSO). The concept of FPSO is applying an adaptive fuzzy logic controller (FLC) to dynami-

cally change the inertia weight. The fuzzy logic controller has three variables: two inputs and one output. The two inputs are the normalized current best performance evaluation (CBPE) and the current inertia weight. The only output variable is the change of inertia weight (δw).

Following this thought, H. Liu and A. Abraham in [41] developed a two-input and two-output FLC. In this FLC, the inputs are the normalized CBPE and the current velocity of particles. The outputs are a scaling factor to control the domain of particles, ρ, and V_{ck}, which controls the change of the velocity threshold, v_c, according to 2.11.

$$v_c = e - [10(1 + V_{ck})] \tag{2.11}$$

This strategy aims to encourage the lazy particles, allowing them to explore better solutions. If a particle's velocity drops below a threshold v_c, a new velocity is reassigned by 2.12b.

$$v_{i,d}(t+1) = w\ \hat{v} + c_1\ rand_1()\ (p_{i,d}(t) - x_{i,d}(t)) + c_2\ rand_2()\ (p_{g,d}(t) - x_{i,d}(t)) \tag{2.12a}$$

$$\hat{v} = \begin{cases} v_{i,d} & ,\ if\ |v_{i,d}| \geq v_c \\ rand(-1,1)v_{max}/\rho & ,\ if\ |v_{i,d}| < v_c \end{cases} \tag{2.12b}$$

Here, $rand(-1,1)$ represents a random number uniformly distributed in the interval $[-1,1]$, v_c the minimum velocity threshold, and v_{max} the maximum velocity of the particles.

Nevertheless, both proposed strategies face a limitation. Normalizing the CBPE requires prior information about the objective function, which can be challenging to estimate.

In reference [42], P. Yadmellat et al. introduce a fresh perspective on a FLC with three variables. In this approach, the number of interactions, t, and the particle's average relative velocity, δV_{av}, serve as inputs, while the output is the inertial weight, w.

All the FPSO methods, despite promising performance across multiple test functions, pose difficulties in terms of implementation.

2.3.3.4 Random strategies to adjust inertia weight

Most real world applications exhibit dynamic nature. With this in mind, a random inertia weight PSO was proposed in [43] to track and optimize these dynamic systems. The random inertia weight is established by the following Equation 2.13:

$$w = 0.5 + rand()/2 \tag{2.13}$$

Here, the term $rand()$ denotes a random number uniformly distributed between $[0, 1]$. The random inertia weight drives under the dilemma of deciding whether exploration (a larger inertia weight value) or exploitation (a smaller inertia weight) will be better in a given time.

2.3.3.5 Other strategies to adjust inertia weight

The last example is an enhanced PSO with an adaptive inertia weight. In this case, the inertia weight adjusts dynamically in response to a regulation function.

The work [44] suggests an adaptive inertia weight that depends on the current swarm diversity and the congregation degree. Here, the aggregation state is evaluated by computing the mean square error (MSE) of the swarm fitness. This strategy seeks to prevent premature convergence, as indicated by the reduction in population density.

An additional adaptive inertia weight was proposed in [45]. In this case, the inertia weight is adjusted in accordance to the evolutionary speed factor and aggregation degree. Lastly, [46] introduced the method in which the inertia weight varies with the location and velocity. With these improvements, the particles have a better capability to adapt to complex and non-linear problems, granting them greater flexibility to vary the global and local search abilities, thus avoiding the convergence into a local minimum.

Chapter 3

Methodology

The study performed in this work expands on Palmeira's research [47] with the computational tool XOPTFOIL [48]. This chapter describes the relevant formulation of the tool and the modifications implemented to enable the new proposed objective. Section 3.1 outlines the algorithm used by the tool. Chapter 3.2 describes the initial airfoil transformations and the available parametrization approaches. Section 3.3 discusses the strategies used for airfoil evaluation. Finally, Section 3.4 contains the formulation of the optimization algorithm.

3.1 XOPTFOIL

XOPTFOIL [48] aims to determine the airfoil design that minimizes a specific objective function generated from a defined set of operating points, while maintaining compliance with imposed constraints. The algorithm that drives this tool is detailed in Figure 3.1 and briefly summarized as follows:

1. Read the input file, which contains the general settings, the airfoil coordinates file location, parametrization options, output options, constraint options, optimization options, and XFOIL options.

2. Check if the inputs provided are appropriate, and if not, stop the optimization.

3. Load the seed airfoil coordinates, normalize the airfoil, and calculate the design variables for the specified shape functions type.

4. Calculate the seed airfoil properties under constraints and obtain the airfoil aerodynamic properties from the XFOIL analysis for each operating point. Then, save the values of these properties to use as references.

5. Determine the upper and lower limits for the design variables and the maximum velocity from those. Then, calculate the particles' initial design variables and velocity.

6. Tune the particle swarm and start the optimization.

7. Compute the new design variables and the airfoil coordinates from them. Check the constraints and analyze the operation points with XFOIL to calculate the objective function value.

8. Update the best individual and overall particle, if appropriate, and update each particle's velocity.

9. Check if the population has converged or if the iteration number has reached its limit. If not, return to 7.

Note that the description and flowchart provided have already been customized to the application of a particular optimization approach, the PSO.

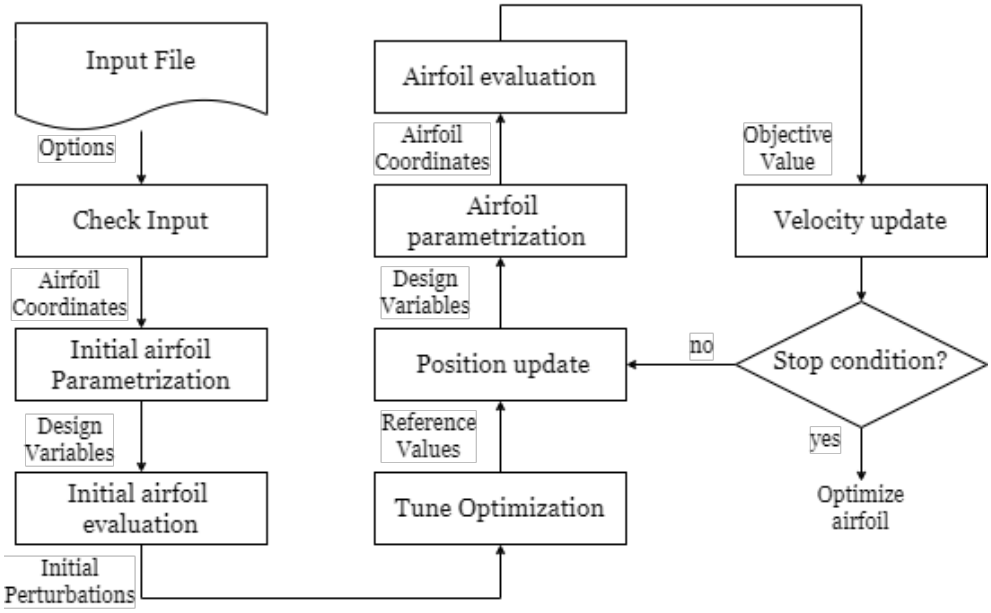


Figure 3.1: XOPTFOIL tool flowchart.

XOPTFOIL requires an input file that contains all the specifications for the intended airfoil optimization. This file is composed of Fortran namelists that contain:

- The location where the tool will obtain its seed airfoil coordinates. These coordinates must begin on the upper trailing edge, move towards the leading edge, and close on the lower trailing edge. This airfoil is used as reference during the optimization;

- The options for parametrization, including the shape function type, the function parameters, and the number of design variables per surface;

- The data options that specify which data to save throughout the optimization process, such as design coordinates, polars, pressure distribution, and boundary layer data at

the required operation points for each improved design. Also, the constraints values, the objective function values, and the optimization history are stored mandatorily in each iteration. Additionally, command the amount of control information provided in each evaluation;

- The operating conditions under which the airfoil is optimized. This includes the implementation or not of flaps, the choice of the flap connection type, the optimization or fixed positioning of the flap hinge along the chord, whether to maintain, set, or optimize the trailing edge value, as well as the determination of the total number of operating points and their respective conditions:
  - Aerodynamic conditions: the choice between the use of angle of attack or lift coefficient and their respective values (including the starting point, increment, and endpoint for the maximum lift search), the Reynolds number, the Mach number, and the transition criterion parameter necessary to XFOIL, N_{crit}.
  - The objective conditions: the optimization objective/type, the weight assigned to each objective, the utilization of polar results from other operating points, and the definition or optimization of the flap angle deflection or even the use of other operating points for the flap deflection.
- The airfoil constraints, comprising the set of geometrical, aerodynamic, and flight constraints on the airfoil;
- The particle swarm optimization options, including the parameters for the optimization process, such as population size, maximum iterations, speed limit, maximum iterations, and particle radius before triggering a stop condition;
- The XFOIL run options for the aerodynamic calculation, such as the number of points and paneling distributions, the freestream turbulence parameter, whether or not to utilize the viscous mode, if to reinitialize the boundary layer, and the strategy for non-converged cases;
- The aircraft data including the total weight, thrust function, the wing, fuselage, and tail characteristics, as well as, the landing gear drag and additional drag components.
- The flight phases parameters, specific to the ACC competition:
  - Take-off: the runway altitude and length, the rotation factor, the friction coefficient, the aircraft's empty weight, and the reference payload weight.

- Climb: the average altitude, the time and altitude to climb, the score function coefficients, the acceleration activation, and the starting velocity for acceleration without the take-off phase.
- Dash: the time to travel, the dash altitude, the reference distance traveled, the acceleration to dash speed activation, the starting velocity to acceleration, and extra time without climb phase.
- Turn: the activation of turn, the turn altitude, the load factor, and the maximum leg length.

Note that, the last two points of the input data are added to the original input for setting the new objective function, presented in Section 3.3.4. The XOPTFOIL algorithm with the new objective function implemented is available in [49]. Furthermore, the aerodynamic conditions reveal that the tool only supports the type 1 XFOIL analysis, keeping the Reynolds number and the Mach number constant. This information is significant because when using the angle of attack input, the lift coefficient can vary, modifying the optimization conditions.

3.2 Parametrization routine

After the input file has been read and validated, the initial airfoil coordinates are retrieved and stored at the provided location. With the airfoil coordinates, the tool starts the execution of the initial parametrization routine. As described in Section 3.1, this method seeks to determine the airfoil design variables for the specified shape function.

The parametrization routine comprises three distinct phases. Firstly, the airfoil is redefined with the XFOIL PANGEN subroutine to generate new coordinates based on the XFOIL paneling options.

Secondly, the airfoil is transformed to standardize its dimensions to a unit chord length and a zero angle of attack. For that, it is first determined the trailing edge midpoint and the leading edge coordinates. Then, the airfoil is translated so the leading edge becomes the origin $(0, 0)$ and scaled to achieve a unit chord length. Finally, using the coordinates of the leading and trailing edges, the original angle of attack is determined and the airfoil is rotated accordingly. Optionally, if a specified trailing edge thickness is set as an input value, additional transformations are applied to adjust the upper and lower surface coordinates to meet this requirement.

Finally, parametrization methods are employed to determine the initial design variable. This phase is also used during the optimization process in the opposite direction to construct

the airfoil geometry to be analyzed using XFOIL solver.

The XOPTFOIL had already implemented two methods, the Hicks-Henne functions and orthogonal NACA functions. Palmeira [47] added three other methods: B-splines, class shape functions, and Bezier-PARSEC 333.

3.2.1 Hicks-Henne functions

Hicks and Henne formulated the Hicks-Henne (HH) functions, consisting of linear combinations of sine functions added to the baseline airfoil. The resulting airfoil shape is defined by Equation 3.1. [50]

$$z_i = z_{base,i} + \sum_{j=1}^{N_{functions}} a_j \sin^{h_{1,j}}\left(\pi x_{base,i}^{\frac{\log 0.5}{\log h_{2,j}}}\right) \ , \ \forall i \in [1, N_p] \tag{3.1}$$

According to [51], each function is described by three variables: the coefficient a, the parameter h_1 that regulates the function width, and the parameter h_2 that specifies the position of the base function's maximum.

As the functions serve to disturb the initial airfoil, these parameters are considered design variables, and the starting design parameters are zero. However, variables h_1 and h_2 cannot be zero, so they are restricted to $h_1 \in [h_{1,min}, 10]$ and $h_2 \in]0, 1[$.

3.2.2 Orthogonal NACA functions

The orthogonal NACA functions are an orthogonal approach proposed by Chang et al. [52], which couple the modes generators of the NACA 4-digit airfoils and the extensions of NACA functions for supercritical airfoils to deform the base airfoil. The airfoil shape is generated by Equation 3.2, with the expanded modes $f_{naca,j}(x)$ given by Equation 3.3, and then normalized by the maximum value of each mode.

$$z_i = z_{base,i} + \sum_{j=1}^{N_{functions}} a_j \bar{f}_{naca,j}\left(x_{base,i}\right) , \ \forall i \in [1, N_p] \tag{3.2}$$

$$\bar{f}_{naca,j}\left(x_i\right) = \begin{cases} x_{base,i}^{\frac{1}{(j+1)/2+1}} - x_{base,i}^{\frac{1}{(j+1)/2}} & \text{if } j \text{ is odd} \\ x_{base,i}^{j/2}(1 - x_{base,i}) & \text{if } j \text{ is even} \end{cases} \tag{3.3}$$

As the approach transforms the basic airfoil, the coefficients a_j serve as design variables. Thus, when the coefficients are zero, the initial airfoil is obtained.

3.2.3 B-splines

The B-splines indicate a smooth curve technique that is obtained by the product of basis functions $N_{i,k}(t)$ and an array of spatially defined discrete control points P_i. [53] The Equation 3.4 establishes the B-spline curves for a planar curve, $P_i \in \mathbb{R}^2$ and $C(t) \in \mathbb{R}^2$, with n basis functions of order $k < n$ and scalar $t \in [T_{k-1}, T_{N_{cp}}]$.

$$C(t) = \sum_{i=0}^{n-1} N_{i,p}(t)\mathbf{P}_i \tag{3.4}$$

The B-Splines method can represent airfoils in various ways, but to meet the constraints imposed is used two B-Splines, one for each upper and lower surfaces. For each B-spline, P_0 is fixed at the leading edge $(0, 0)$, P_{n-1} is fixed at the trailing edge, and P_1 is aligned vertically with the leading edge. These applies to both upper and lower surfaces.

The tool's algorithm offers two options for the control points' x coordinates: free or fixed. For the fixed option, Equation 3.5 uses a cosine distribution for the x coordinates. With the z coordinates of control points 1 to $n-2$ the only free coordinates. Thus, resulting in $n-2$ design variables per surface.

$$\mathbf{P}_0 = (0, 0), \quad \mathbf{P}_i = \left(\frac{1}{2}\left[1 - \cos\left(\frac{\pi(i-1)}{n+1}\right)\right], a_i\right), \quad \mathbf{P}_{n-1} = (1, z_{te}) \tag{3.5}$$

In the free variation, both coordinates are free for control points 2 to $n-2$, plus z for P_1, resulting in $2(n-3)+1$ design variables for each surface.

3.2.4 Class shape functions

The class shape functions (CSF), by Kulfan and Bussoletti [54], are the product of the class function $C(x)$ and the shape function $S(x)$. The airfoil shape, considering the geometric constraints imposed, needs to be represented by two curves, for upper and lower surfaces, as described in Equations 3.6a and 3.6b. An additional term is added to account for the trailing edge thickness.

$$z_{Up}(x) = C_{N2}^{N1}(x).S_{Up}(x) + x\frac{z_{TE}}{2} \tag{3.6a}$$

$$z_{Lo}(x) = C_{N2}^{N1}(x).S_{Lo}(x) + x\frac{z_{TE}}{2} \tag{3.6b}$$

The class function parameters N_1 and N_2 control the class in use, as shown in Equation 3.7. In the tool used, the airfoil class parameters are $N_1 = 0.5$ and $N_2 = 1$.

$$C_{N2}^{N1}(x) = x^{N1}\,(1-x)^{N_2} \tag{3.7}$$

Kulfan in [55] proposed expressing $S(x)$ as a linear combination of Bernstein polynomials of degree n, $B_{i,n}(x)$, and coefficient a_i. With the addition of a further term known as leading edge modification (LEM) to enhance the fidelity at the leading edge, shown in Equation 3.8.

$$S(x) = \sum_{i=0}^{n} a_i B_{i,n}(x) + \underbrace{a_{n+1}x^{0.5}(1-x)^{n-0.5}}_{LEM} \tag{3.8}$$

3.2.5 Bezier-PARSEC 333

The Bezier-PARSEC 3333 (BPP) approach, presented by Derksen et al. [56], combines the PARSEC formulation and the Bezier curves. The Bezier-PARSEC 333 defines the airfoil through four third-degree Bezier curves, two for the camber line and two for the thickness distribution, with Equation 3.9 describing a third-degree Bezier curve.

$$x(t) = x_0^c(1-t)^3 + 3x_1^c t(1-t)^2 + 3x_2^c t^2(1-t) + x_3^c t^3 \tag{3.9a}$$

$$z(t) = z_0^c(1-t)^3 + 3z_1^c t(1-t)^2 + 3z_2^c t^2(1-t) + z_3^c t^3 \tag{3.9b}$$

where t denotes the parameter that ranges from 0 at the start to 1 at the end. The curves are generated exclusively using twelve aerodynamic parameters: the leading edge radius, r_{le}, the trailing camber line angle, a_{te}, the trailing wedge angle, b_{te}, the trailing edge vertical displacement, z_{te}, the leading edge direction, c_{le}, the location of the camber crest, (x_c, y_c), the curvature of the camber crest, j_c, the position of the thickness crest, (x_t, y_t), the curvature of the thickness crest, j_t, and the half thickness of the trailing edge, $z_{te}/2$.

3.3 Airfoil evaluation

The airfoil evaluation is a crucial component of the tool, responsible for computing the objective function's value used to analyze the designs developed during the optimization. Within the framework of XOPTFOIL, which is a constrained multi-point single-objective approach, the airfoil's performance is assessed by a hybrid function. This function incorporates the contributions from the selected operating points as well as the penalties for non-compliance with the constraints. The calculation of the objective value follows the scheme:

1. Calculate all geometric properties under constraint and assess the penalty value resulting from the non-compliance. If the penalties value exceeds the limit, the penalty value multiplied by the factor 10^6 is returned as the objective function. Otherwise, the evaluation proceeds.
2. Assess the operating point aerodynamic characteristics with XFOIL solver.
3. Check the XFOIL analysis consistency and if necessary, repeat the analysis with slight perturbed Reynolds number. Compare the two analysis and keep the more conservative value.
4. Calculate the penalty values for the aerodynamic constraints based on the XFOIL results. Then, proceed in the same manner as in step 1.
5. Calculate the flight performance of an input aircraft based on the XFOIL results of the airfoil in study.
6. Calculate the penalty value for the flight performance constraints, and follow the step 1 approach.
7. Calculate the objective function value based on the specific aerodynamic coefficients or the flight performance score plus the total penalty value.

Note that, this scheme already incorporates the new objective function, which is the focus of this work and will be explored in greater detail in Section 3.3.4.

3.3.1 Penalties

As previously discussed, the XOPTFOIL allows the incorporation of both geometrical and aerodynamic constraints in the airfoil design. Additionally, it was implemented flight constraints to ensure a minimum performance during each flight phase. The compliance of the constraints is ensured by a penalty system, as shown in Equation 3.10.

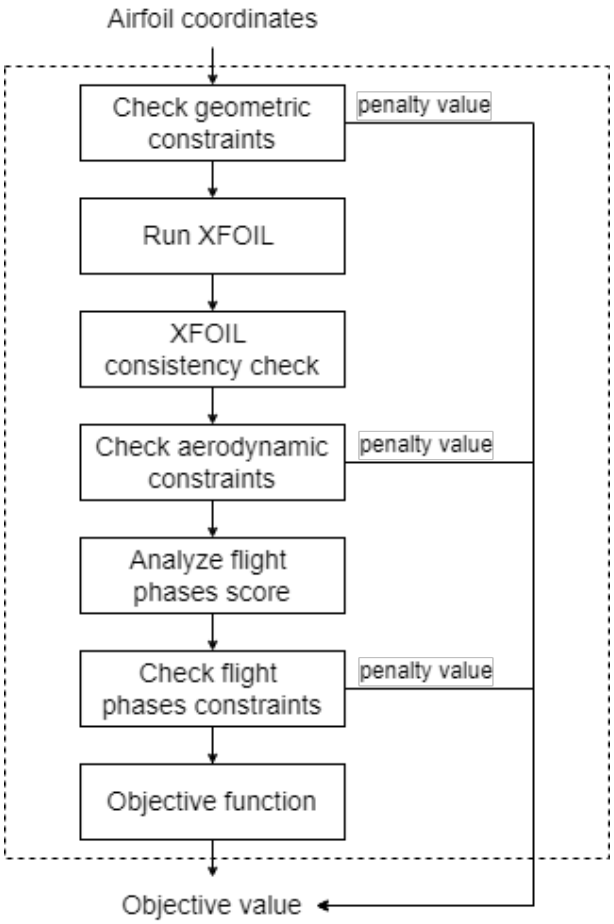


Figure 3.2: Airfoil evaluation flowchart.

$$\delta_i = \begin{cases} \frac{\sigma_i - \sigma_{ref,i}}{|\sigma_{ref,i}|} & if\ \sigma_{ref,i}\ is\ a\ upper\ limit \\ \frac{\sigma_{ref,i} - \sigma_i}{|\sigma_{ref,i}|} & if\ \sigma_{ref,i}\ is\ a\ upper\ limit \end{cases} \tag{3.10}$$

where the penalties δ_i, for each i^{th} constraint, represent the relative difference between the actual property value, σ_i and the specified upper or lower reference value, $\sigma_{ref,i}$. Negative penalty values indicate that the property is within limits, so they are disregarded.

As mentioned in the Section 3.3, the penalty value is assessed at each step of the airfoil evaluation to verify if it is within the predefined limits. If the value is within limits, the evaluation proceeds; otherwise, the evaluation ends, and the objective value receives the penalty value multiplied by a factor of 10^6, as outlined in Equation 3.11.

$$Obj = 10^6 \cdot \delta_i,\ if\ \exists \delta_i > \delta_{limit} \tag{3.11}$$

The penalty limit is defined by a linearly decreasing function, with initial and final values specified as inputs. This approach gives the algorithm greater freedom to explore initially and then gradually forces the final designs to meet all constraints. The total penalty value δ_{total} is the sum of the penalties for all constraints, N_c.

$$\delta_{total} = \sum_{i=1}^{N_c} \delta_i, \; if \; 0 < \delta_i < \delta_{limit} \tag{3.12}$$

Detailing the geometric properties under constraint, several thickness constraints are imposed, including maximum and minimum thickness, as well as the thickness at a specific chord position. For each of these three thickness constraints, an acceptable range of values is defined. In contrast, the camber line is evaluated solely based on its maximum value.

The leading and trailing edges are also properties under control. The LE is bound by the angles of the upper and lower surfaces to determine if the LE is not too blunt or too sharp. Additionally, the difference between these angles is evaluated to ensure that it is not wildly distinct.

These angles are calculated from the coordinates of the first two points of each surface.

For the trailing edge (TE), two constraints are considered: the trailing edge thickness and wedge angle. The TE wedge angle constraint is included to prevent the optimized airfoil from converging to an excessively thin cusped aft section. For that, the user must specify the chord position at which the wedge angle begins to be evaluated.

Regarding surface's constraints, it is included the number of curvature reversals on each surface. This constraint aims to prevent the airfoil surfaces from having bumps, with the user setting the curvature radius threshold for this constraint. Additionally, the angle between adjacent surface panels, calculated using the XFOIL routine CANG, and the growth rate of the panels are also evaluated.

The flap properties are another set of constraints, where the algorithm verifies that various deflections and the hinge position are within predefined ranges.

Finally, the software computes the overall geometric penalties by adding all the individual penalties and verifying whether they remain within the penalty limits.

If the operator does not want to use a specific constraint, it can be set a high value for maximum constraints and zero or a high negative value for the minimum constraints.

In terms of aerodynamic constraints, it is penalized the unconvergence of the aerodynamic simulations. For this constraint, the XFOIL convergence criterion RMSBL must be below 10^{-4} to consider the solver converged.

Additionally, the user has the option to apply constraints on the minimum moment and lift coefficients, as well as the maximum drag coefficient, for specified operating points. In these constraints, the user can define the limits or use the seed airfoil as reference values.

As previously discussed, in addition to the geometrical and aerodynamic constraints, flight constraints were incorporated. These constraints establish the minimum performance requirements during the take-off, climb, cruise, and turn phases. Specifically, the constraints specify the minimum acceptable values for the aircraft's total weight, climb speed, cruise speed, and turn velocity.

3.3.2 XFOIL analysis

XFOIL, used in XOPTFOIL for the flow analysis, is a collection of menu-driven routines that perform multiple functions, such as viscous or inviscid analysis and inverse or direct airfoil design. XFOIL, developed by Drela [16], intends to easily estimate the airfoil performance at low Reynolds numbers by combining high-order panel methods with the fully-coupled viscous/inviscid interaction method, according to documentation in [57].

In this method, the viscous solution (boundary layer, BL, and wake) is described with a two-equation lagged dissipation integral BL and an envelope e^n transition criterion, taken from the ISES code developed by Drela and Giles. This viscous solution is iterated with the incompressible potential flow, via the surface transpiration, to properly calculate the limited separation regions. The panel solution with the Karman-Tsien compressibility correction obtains the total velocity at each point on the airfoil surface, and the wake is incorporated into the viscous equations, creating a non-linear elliptic system that is solved by a full-Newton method.

In the e^n method, the transition is triggered by the log of the amplification factor of the most-amplified frequency specified by the user, N_{crit}.

XFOIL is integrated into XOPTFOIL as a set of external modules, containing the necessary subroutines directly retrieved from the source code in [57]. The interaction between XOPTFOIL and XFOIL is enabled by the presence of two modules.

The first, xfoil_inc, declares and saves the XFOIL variables, while the second, xfoil_driver, runs the subroutines in the XFOIL modules to perform the flow analysis based on the operating points and XFOIL settings.

To perform XFOIL flow analysis, the following parameters must be loaded into the routines: the airfoil coordinates, XFOIL paneling and run options, and the aerodynamic conditions and flap variables for each operating point. Specifically, for each operating point, the

user must set the Reynolds number, the Mach number, and the N_{crit}, along with the type and mode of operation point. This last setup determines whether the analysis will be in a single operating point or in a sequence of operating points and whether each point will be run at a constant angle of attack or lift coefficient. Furthermore, the flap angle deflection and hinge position must be supplied to the routine, whether specified by the user or as design variables.

The algorithm, with this information, calculates the angle of attack, the moment, lift, and drag coefficients, as well as XFOIL's convergence variable RMSBL (root mean squares of the boundary layer). If asked, the routine will also supply the turbulent transition location for the lower and upper surfaces. The subroutine run_xfoil, contained in the module xfoil_driver, will be the one used to calculate these results, following the methodology below:

1. Invoke the xfoil_inic module to assign the XFOIL options values, which control the execution of XFOIL calculations. Subsequently, use the BLPINI command to set the boundary layer calibration parameters, and finally, assign the XFOIL paneling options to the module variables.

2. Check if any operating point wants to assume the previous values. And if true, save the previous characteristics at this operating point and proceed to step 5. If not, continue with the analysis.

3. Process the airfoil geometry by assigning the airfoil coordinates. Next, use function PANGEN to conform the airfoil coordinates to the paneling options and apply the flap deflection into the airfoil coordinates if requested.

4. Assign the Reynolds number, Mach number, and N_{crit} for each operating point. Next, use the function COMSET to set the compressible Karma-Tsien parameter from the freestream Mach number. Finally, check if the reinitialization option is enable and if so, set the appropriate variables.

5. Check the type of XFOIL analysis (single or sequence):

   (a) For the single-point analysis, the user selects one of three initialization options ('none', 'uncorvergence' and '

      i. For the 'none' and 'unconverge' options, the analysis is performed without initialization. So, depending on whether the angle of attack or the lift coefficient is provided, the functions SPECAL and SPECCL, respectively, are called to perform the inviscid calculations. The viscous flow is then calculated using the function VISCAL. Finally, RMSBL is used to evaluate the convergence of

the XFOIL solver. If the analysis does not converge and the 'unconverge' option has been selected, proceed to the next step. Otherwise, proceed to step 6.

ii. For the analysis or reanalysis with initialization is created a sequence of angles of attack or lift coefficients, and then each operating point is analyzed as described for a single point.

(b) To analyze a sequence of angles of attack, the XFOIL uses the process described in point 5(a)ii and then saves the best value of the aerodynamic property from the point's sequence.

6. Verify if all operating points have been analyzed to conclude the XFOIL routine. If not, return to step 2.

Note that only a type 1 analysis is possible in the current implementation of XFOIL, as it requires specifying the Reynolds number, the Mach number, and the N_{crit}. Also, in the current implementation, the sequence analysis is only triggered when the goal is to search the maximum lift coefficient.

After the XFOIL run, the results undergo a consistency check. In this check, the results of each operating point are compared with the minimum drag or the maximum lift coefficients values achieved so far by each operating point, multiplied by a tolerance factor introduced by the user. The reanalysis of any operating point, $i \in [1, N_{op}]$, is triggered if the previous conditions, summarized in Equation 3.13, are met.

$$\begin{cases} C_{l,i}(t) = (1 - tol_{C_l}) C_{l_{max,i}}(t) \\ C_{d,i}(t) = (1 - tol_{C_d}) C_{d_{min,i}}(t) \end{cases} , i \in [1, N_{op}] \tag{3.13}$$

The consistency check involves reanalyzing the XFOIL results with a slightly perturbed Reynolds number, set to 99.7% of the original value. The results obtained are then compared, and the more conservative of the two runs is selected. This procedure is necessary to prevent the optimization process from exploring singularities within the XFOIL solution that might give artificially improved results.

3.3.3 Objective function

The airfoil performance evaluation is performed using an objective function, Obj, whose goal is to be minimized during the optimization. This objective function is the sum of the

relative objective values of each operating point multiplied by their respective normalized weights plus the total penalty value, Equation 3.14. The use of relative aerodynamic objectives, through the division by the reference value $\varphi_{i,ref}$, retrieves the influence of the magnitude of each aerodynamic objective, φ_i, from the function Obj.

$$Obj = \sum_{i=1}^{N_{opt}} \left(\bar{w}_i \, \frac{\varphi_i}{\varphi_{i,ref}} \right) + \delta \tag{3.14}$$

The algorithm takes the aerodynamic objectives and their corresponding weights from the input file. The input weights are normalized using Equation 3.15.

$$\bar{w}_i = \frac{w_i}{\sum_{i=1}^{N_{opt}} w_i} \tag{3.15}$$

The algorithm includes different sorts of objectives that assign which property to improve. Equation 3.16 shows the types of operation points and their corresponding aerodynamics objectives (φ_i). The equation also displays the target property where a certain property is aimed at a specific value.

$$\begin{cases} \varphi_i = \frac{1}{C_l}, \; if \; maximize \; C_l \\ \varphi_i = C_d, \; if \; minimize \; C_d \\ \varphi_i = \frac{C_d}{C_l}, \; if \; maximize \; \frac{C_l}{C_d} \\ \varphi_i = \frac{C_d}{C_l^{\frac{3}{2}}}, \; if \; maximize \; \frac{C_l^{\frac{3}{2}}}{C_d} \\ \varphi_i = \frac{1}{0.5(xtrt+xtrb)}, \; if \; maximize \; xtr \\ \varphi_i = \frac{1}{\frac{dC_l}{d\alpha}}, \; if \; maximize \; \frac{dC_l}{d\alpha} \\ \varphi_i = \left(\frac{prop_{target} - prop}{prop} \right)^2, \; if \; target \; property \end{cases} \quad , \forall i \in [1, N_{op}] \tag{3.16}$$

Using the seed airfoil, Equation 3.16 also produces the reference aerodynamic objective, $\varphi_{i,ref}$. Note that, if the goal is to maximize a parameter, the operation point objective is inverted, since the aim of the objective function is to be minimized.

For this thesis, a new objective based on how the airfoil performs in a specific aircraft is added. This new objective presents a method to weight the relevant conditions of the case study based on the competition's overall score. The idea, with slight modifications, is to use

the score as φ_i and the total points without bonuses as $\varphi_{i,ref}$.

3.3.4 Performance determination

The tool's newly introduced objective is presented in this section. The proposed objective, in contrast to the current ones that focus on optimizing a predetermined equilibrium of aerodynamic properties, aims to discover the optimum equilibrium and then optimize it. For that purpose, a direct method to evaluate the impact of the airfoil characteristics on the aircraft's overall performance is applied.

The case study, as stated in Section 1.2, is divided into three flight tasks, each task receiving a maximum of 1000 points without bonus. These tasks include the take-off phase scored by the payload, the climb phase scored by the altitude, and the cruise phase, scored by the distance traveled. Equation 3.17 represents the proposed objective function based on this flight path.

$$Obj = 1000 \left(\frac{\bar{w}_{take-off} + \bar{w}_{climb} + \bar{w}_{cruise}}{\bar{w}_{take-off} S_{payload} + \bar{w}_{climb} S_{altitude} + \bar{w}_{cruise} S_{distance}} \right) + \delta \tag{3.17}$$

Taking advantage of the input weights, the operator is able to adjust the weight of each flight phase, providing the function additional flexibility to adapt to new situations. For the present scenario, all normalized weights are set to $\frac{1}{3}$.

The first goal is to maximize the payload, keeping the take-off distance within the given limits. For that purpose, an equation relating the payload to the take-off distance is required. According to Gudmundsson's book [58], the kinematic relationship between acceleration, speed, and distance is utilized to compute the take-off distance. Assuming that the ground run begins at rest without relative wind, we obtain,

$$S - S_0 = \frac{V_f^2 - V_i^2}{2a_{av}} \Leftrightarrow s_G = \frac{V_{LOF}^2}{2a_{av}} \tag{3.18}$$

In this case, s_G symbolizes the ground run distance, V_{LOF} the lift-off velocity, and a_{av} the average acceleration. To compute the acceleration during the ground run on a horizontal flat runway, the equation of motion is applied.

$$\frac{dV}{dt} = g \left[\frac{T}{W} - \frac{D}{W} - \mu \left(1 - \frac{L}{W} \right) \right] \tag{3.19}$$

From this equation, the acceleration during the ground run, $\frac{dV}{dt}$, depends on the aircraft's total weight, W, thrust, T, drag, D, lift, L, and ground friction coefficient, μ. However, it is also dependent on velocity, as thrust, drag, and lift all change with it. The average acceleration is then calculated using the average velocity, V_{av}, given by the Equation 3.20.

$$V_{av} = \sqrt{\frac{V_f^2 + V_i^2}{2}} = \frac{V_f}{\sqrt{2}} = \frac{V_{LOF}}{\sqrt{2}} \tag{3.20}$$

The final velocity is the lift-off velocity, V_{LOF}, given by the stall velocity V_s times a factor, A_1. This factor takes into consideration the distance covered during the rotation for lift-off.

$$V_{LOF} = A_1 V_s = A_1 \sqrt{\frac{2W}{\rho S C_{L,max}}} \tag{3.21}$$

By utilizing Equations 3.21 and 3.20 to compute the average acceleration via Equation 3.19, and subsequently inserting that value into Equation 3.18, along with the employment of adimensional coefficients for drag and lift, we derive the relation between take-off distance and the aircraft's total weight represented by Equation 3.22.

$$s_G = \frac{A_1^2 \left(\frac{W}{S}\right)}{\rho g \left[\left(\frac{T_{V=Vav}}{W} - \mu\right) C_{L_{max}} + \left(\frac{A_1^2}{2}\right) \left(\mu C_{L,to} - C_{D,to}\right)\right]} \tag{3.22}$$

Note that the drag and lift coefficients employed in Equation 3.22 refer to the aircraft, not the airfoil. In Section 3.3.5, the transformation from the airfoil coefficients to the aircraft coefficients will be explained. The payload appears in Equation 3.22, as the aircraft's total weight is the sum of the payload and the empty weight, which is the weight of the aircraft's structure and systems. Therefore, the payload is calculated using Equation 3.23.

$$W_{payload} = W - W_{empty} = W - (W_{systems} + W_{structure}) \tag{3.23}$$

The XFOIL analysis of the airfoil provides the non-dimensional characteristics of the airfoil, so the purpose is to compute the payload using the drag and lift coefficients. However, Equation 3.22 does not allow for the direct calculation of payload based on drag and lift coefficients at the required operating points and a specified take-off distance. The payload is

then computed via an iterative approach, as seen by the flowchart in Figure 3.3.

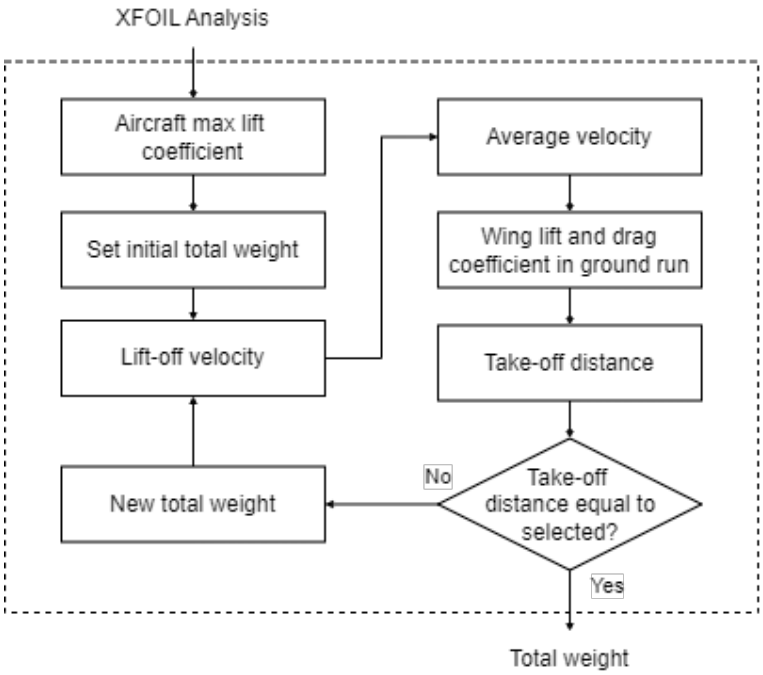


Figure 3.3: Total weight flowchart.

The payload score is then obtain from Equation 3.24. Because of the partial scoring method used in this competition, the aircraft payload is divided by a reference payload, corresponding to the higher payload carried by any team, given in Table 4.2.

$$S_{payload} = 1000 \, \frac{W_{payload}}{W_{payload,ref}} \tag{3.24}$$

In contrast, to the objective implied in Chapter 1.2, the second goal will not be achieved by maximizing the altitude. According to German legislation, the aircraft cannot exceed 120 meters in height. Therefore, the organizing team developed Equation 3.25, and the purpose of the climb phase shift to maximize the value of this equation. Equation 3.25 yields a maximum pre-score of 1203 points at 100 meters of height.

$$PS_{altitude} = -3.92e^{-5} \, h_{60s}^4 + 1.08e^{-2} \, h_{60s}^3 - 1.156 \, h_{60s}^2 + 64.2 \, h_{60s} - 537 \tag{3.25}$$

The final score is then calculated by dividing the pre-score by the reference altitude pre-score, as shown in Equation 3.26. Due to the ability to calculate the highest possible altitude pre-score, this value is used as the reference.

$$S_{altitude} = 1000 \, \frac{PS_{altitude}}{PS_{altitude,ref}} \tag{3.26}$$

According to Gudmundsson's book, a climb is defined as a change in potential energy per unit of time [58]. The rate of change can be seen in Equation 3.27.

$$\frac{\Delta \left(PE \right)}{\Delta t} = W \frac{\Delta h}{\Delta t} => \frac{d \left(PE \right)}{dt} = W V_V \tag{3.27}$$

The equations of motion for an aircraft in the climb phase reveal that the product WV_V represents the difference between available power (P_{AV}) and the required power (P_{REQ}) at level flight. The vertical airspeed is subsequently determined using Equation 3.28.

$$V_V = \frac{P_{AV} - P_{REQ}}{W} = \frac{TV_\infty - DV_\infty}{W} \tag{3.28}$$

Take note that thrust, T, and drag, D, vary with air density, ρ, and air density depends on altitude. Thus, thrust and drag change with altitude. The average altitude is then used as an approximation to compute an average vertical velocity, which allows us to calculate the climbed height, h, using Equation 3.29.

$$h = V_V \, \Delta t \tag{3.29}$$

Equation 3.29 states that, for a limited time period, maximizing the altitude climbed requires maximizing the vertical speed. This creates the need to calculate the aircraft's maximum vertical speed. The approach used to determine this value involves conducting a sequence of XFOIL analyses at various conditions, compute the aircraft's lift and drag coefficients at each condition, and use them to calculate the corresponding vertical velocities. These velocities are then compared to select the maximum value and its corresponding operating condition.

Knowing the highest vertical velocity, we can calculate the maximum altitude reached by the aircraft in 60 seconds and verify if it reaches 100 meters. However, the simple application of Equation 3.29 may limit the vertical velocity to the exact minimum necessary to achieve the desired altitude. Forgetting that, if the airplane reaches the required height before 60

seconds, it can use that time to accelerate to cruise speed, referred to as extra time in Equation 3.30.

$$t_{extra} = 60s - \frac{100m}{V_{V,max}} \tag{3.30}$$

Furthermore, the highest vertical velocity occurs at a certain horizontal speed, so we must consider the time it takes the aircraft to accelerate from the lift-off velocity to the horizontal speed that leads to the maximum vertical velocity. For that, the XFOIL analysis, of a set of operating points, is converted into the aircraft's velocity, thrust, and drag. Then, Equation 3.31 may be used to compute the acceleration at each position using the equations of motion for a level flight.

$$a_i = \frac{g}{W}\left(T_i - D_i\right) \tag{3.31}$$

Applying a linear approximation for the acceleration, Equation 3.32, provides the time to accelerate from one operating point to another. At the starting point, the velocity V_{i-1} is the lift-off velocity, and the acceleration, a_{i-1}, is assumed to be the same as the first point, $a_{i-1} = a_i$.

$$t_{accel} = \left(V_i - V_{i-1}\right)\frac{2}{\left(a_i + a_{i-1}\right)} \tag{3.32}$$

Aside from attempting to achieve a more accurate portrayal of the problem, this formulation assigns weight to operational points besides the maximum vertical velocity, thus trying to prevent a singular solution according to [6]. However, this trait is optional, so the operator can enable or disable this feature from the input file.

Equation 3.33 calculates the time necessary for the aircraft to reach 100 m of height, with the acceleration time included.

$$t_{climb} = \frac{100m}{V_{v,max}} + t_{accel} \tag{3.33}$$

If the time is less than 60 seconds, the maximum score is awarded and the extra time is computed by 3.30. In contrast, if the time exceeds 60 seconds, the altitude is calculated

using Equation 3.34 while the score is derived through Equations 3.25 and 3.26, with the extra time being disregarded.

$$h = V_v(60 - t_{accel}) \tag{3.34}$$

A climb score flowchart, illustrated in the Figure 3.4, provides a comprehensive explanation of the entire process of determining the climb performance.

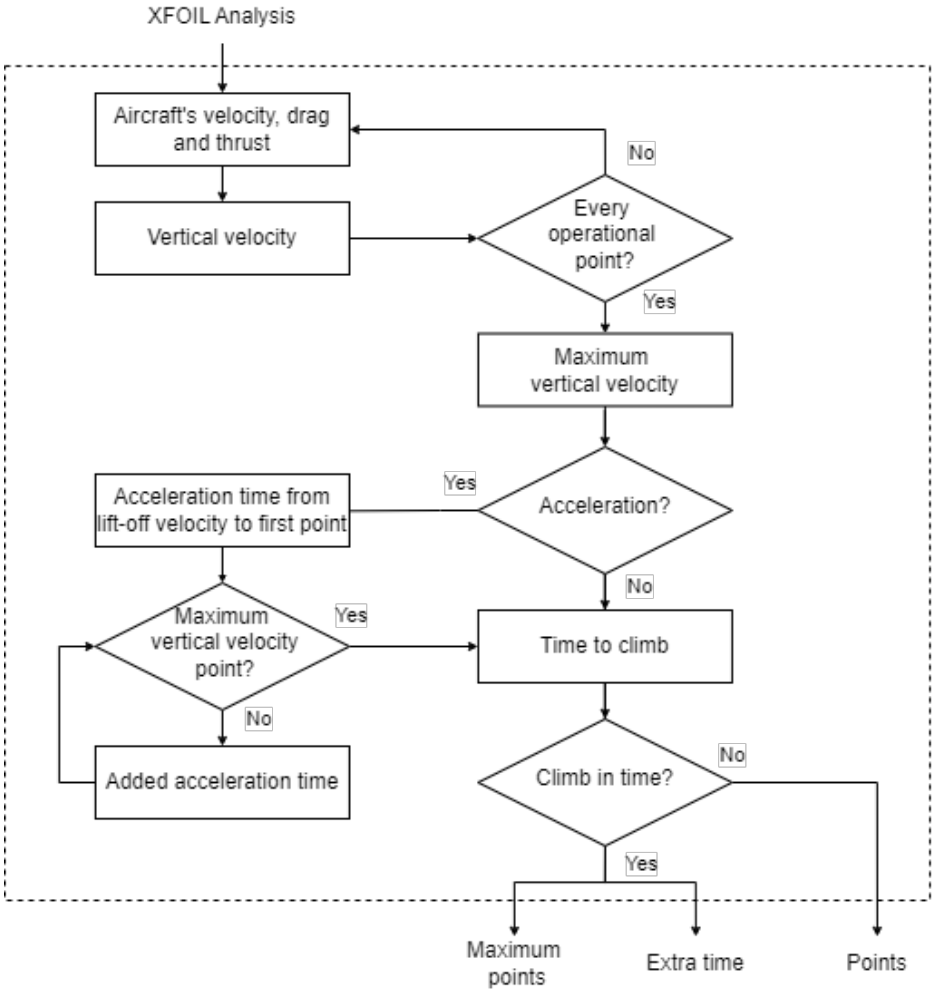


Figure 3.4: Climb score flowchart.

Finally, the last goal is to cover as much distance as possible in 120s. According to Gudmundsson's book [58], the Equation 3.35 uses the kinematic relationship between speed and distance to calculate the distance traveled in cruise.

$$s = \int_0^t V dt \tag{3.35}$$

Equation 3.35 states that to maximize the distance traveled, the velocity must be maximized throughout all the free flight phase. However, the cruise trajectory needs to be split into two stages, straight and turn trajectories, because of the peculiarities of the airfield described in the handbook [1].

In the planar equations of motion for steady level flight a key baseline can be established to compute the maximum straight velocity, as depicted in Equations 3.36.

$$D = T \tag{3.36a}$$

$$L = W \tag{3.36b}$$

In relation to level flight, Equations 3.36 illustrate the balance between the aircraft's lift and drag, which counteracts its weight and thrust forces, respectively.

To ascertain the maximum straight-line velocity from this equilibrium, the aircraft's lift and drag coefficients are determined through the XFOIL analysis. Taking advantage of these coefficients alongside the weight obtained during the take-off phase, the velocity is computed based on the lift coefficient principle. Subsequently, the aircraft's drag and thrust forces are assessed relative to the acquired velocity, with the maximum velocity occurring when the drag force equals the thrust force.

The primary goal is then to identify the conditions at which the aircraft's drag force matches its thrust force. To accomplish this, a series of operating points are systematically evaluated, seeking the two nearest points to $D = T$. A linear approximation method is then employed to derive the maximum velocity based on these two identified points.

Similarly, the process for determining the maximum turn velocity follows a comparable methodology. However, it is essential to note that the equilibrium conditions during a level turn differ. Equations 3.37 articulate the new equations of motion for a flight in the horizontal plane.

$$D = T \tag{3.37a}$$

$$L\cos\phi = W \tag{3.37b}$$

$$L\sin\phi = \frac{W}{g}\frac{V_{turn}^2}{R_{turn}} \tag{3.37c}$$

In this scenario, the lift generated by the aircraft must not only counteract its weight but also balance the centrifugal force induced by the aircraft's inertia during the turn. Within

the context of Equations 3.37, the variable ϕ denotes the bank angle, R_{turn} signifies the turn radius, and V_{turn} represents the horizontal velocity during the turn.

In the pursuit of determining the maximum turn velocity, it becomes essential to introduce a new parameter that depicts the turn dynamics. This selected parameter, denoted as the load factor n, combines the aircraft's structural limitations into the definition of turn dynamics. Its formulation is given by Equation 3.38.

$$n = \frac{L}{W} = \frac{1}{\cos\phi} \tag{3.38}$$

Upon the setup of the load factor by the operator, Equations 3.37a and 3.37b enable the computation of the maximum turn velocity, employing a methodology similar to that used to calculate the level flight velocity.

Now, the challenge shifts to quantifying the distance traveled during both turn and straight-line segments. By subtracting the turn radius from the maximum length of the airspace, the distance traveled in a straight line may be calculated, allowing the definition of the complete lap. Equation 3.39 enables the determination of the turn radius using the load factor and turn velocity following Equation 3.37c.

$$R_{turn} = \frac{V_{turn}^2}{g\sqrt{n^2 - 1}} \tag{3.39}$$

As previously mentioned, the calculation of the straight segment length from the maximum length of the airspace can be derived from the turn radius using Equation 3.40.

$$l_{cruise} = l_{max} - 2\ R_{turn} \tag{3.40}$$

In order to determine the distance covered within the 120 s of the cruise, an average velocity weighted by the distance traveled in each phase during a single lap is employed by Equation 3.41. This methodology eliminates the need to track the aircraft's position throughout a specified time period.

$$V_{av} = \frac{l_{cruise}\ V_{cruise,max} + 2\pi R_{turn,max}\ V_{turn}}{l_{cruise} + 2R_{turn}} \tag{3.41}$$

Using the same approach as in the climb phase, the time that takes to accelerate from the horizontal climb speed to the maximum velocity is estimated.

This methodology is equal to the one used for the climb phase by using a set of operating points in XFOIL analysis to calculate the acceleration and velocity at each one, and then with Equations 3.31 and 3.32, calculate the acceleration time.

The only variation is the application of a linear approximation for the velocity to calculate the distance traveled during acceleration, outlined in Equation 3.42.

$$dist_{accel} = \frac{V_i + V_{i-1}}{2}\, t_{accel} \tag{3.42}$$

In Equation 3.42, when i equals one, velocity V_{i-1} corresponds to the climb speed. Additionally, when estimating the time using Equation 3.32 for $i = 1$, it is assumed that $V_i = V_{i-1}$.

The climb phase supplies the extra time available for accelerating to maximum speed, which is unused during that phase. As a result, the acceleration time and distance are only considered when the acceleration time exceeds this extra time. So, if the acceleration time is less than the extra time, the acceleration stage is disregarded. In reverse, if the acceleration time exceeds the extra time, the extra time is subtracted from the acceleration time, and the distance traveled during this period is not considered. The system of Equations 3.43 calculates the total distance traveled.

$$\begin{cases} dist = (120 - (t_{accel} - t_{extra}))\ V_{av} + dist_{accel} & ,\ if\ t_{accel} > t_{extra} \\ dist = 120\ V_{av} & ,\ if\ t_{accel} \leq t_{extra} \end{cases} \tag{3.43}$$

The traveled distance is evaluated to determine its corresponding score, as described by Equation 3.44. The distance scoring is relative to the highest distance traveled by any team, which serves as the reference distance and it is given in Table 4.2.

$$S_{distance} = 1000\, \frac{dist}{dist_{ref}} \tag{3.44}$$

A distance traveled flowchart, illustrated in the Figure 3.5, provides a comprehensive explanation of the entire process of determining free flight performance.

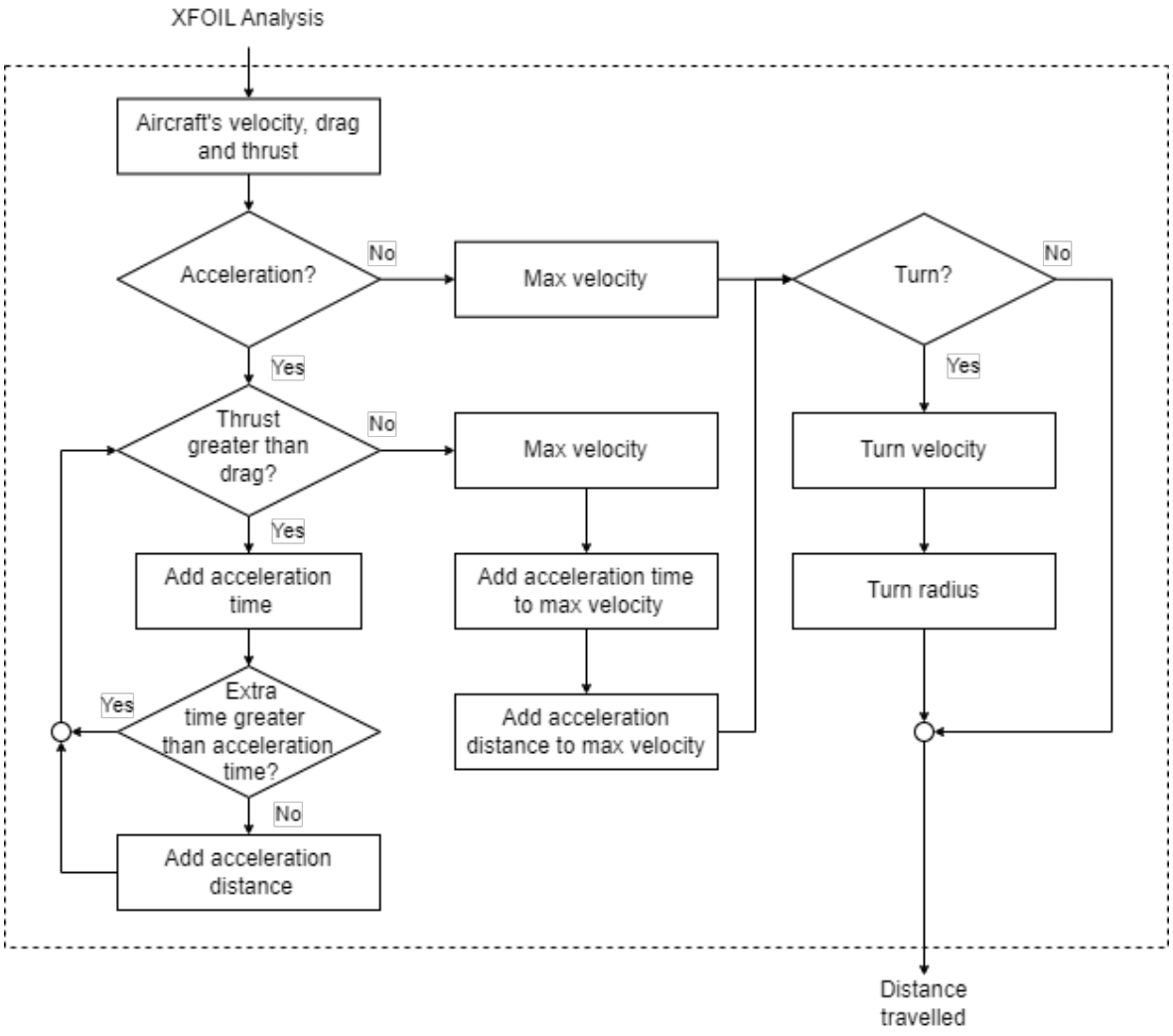


Figure 3.5: Distance traveled flowchart.

3.3.5 Aircraft Aerodynamic Coefficients

In addition to the scores calculated, it is now described the methodology to estimate the aircraft's characteristics via the XFOIL analysis applied to the airfoil under assessment. A simplified model outlined by Raymer in [7] is used to minimize the computing power and processing speed required.

The first goal is to convert the lift coefficient of the airfoil into that of the aircraft. Knowing the wing's predominate role in generating lift, the contributions from other components such as tails and/or fuselage are disregarded as an approximation.

The wing lift coefficient, C_L, is then calculated from the airfoil lift coefficient, C_l, using the leading edge suction method and the induced angle of attack, α_i, in Equations 3.45.

$$\begin{cases} \alpha_i = \frac{C_L}{\pi A}(1+\tau) \\ C_L = C_l \cos\alpha_i \left(\frac{S_{expose}}{S}\right) \end{cases} \tag{3.45}$$

It is worth noting that the C_L equation has been adjusted to take into account the fraction of the wing covered by the fuselage. Additionally, as an approximation, $1/(1+\tau)$ will be considered equal to the wing's Oswald efficiency, e_w.

A different approach is used to compute the maximum lift coefficient. For high aspect ratio wings with moderate sweep and large airfoil leading edge radius, the maximum wing lift coefficient is typically 90% of the airfoil's maximum lift coefficient. Accounting for the coverage of the wing by the fuselage, Equation 3.46 gives the aircraft's maximum lift coefficient.

$$C_L = 0.9 C_l \left(\frac{S_{expose}}{S}\right) \tag{3.46}$$

The second step is to compute the aircraft's drag coefficients using the Equation 3.47, where the total drag is the sum of the contributions of each component of the aircraft plus a miscellaneous drag, which allows the user to account for extra parts.

$$C_D = C_{D_w} + C_{D_{fus}} + C_{D_{tail}} + C_{D_{lg}} + C_{D_{misc}} \tag{3.47}$$

In the case of low Reynolds, wing drag is divided into two components, parasite and induced drag. The parasite drag is assessed using the airfoil's drag coefficient derived from XFOIL analysis. Meanwhile, induced drag is determined utilizing the Oswald span efficiency method. The wing drag is then represented by Equation 3.48.

$$C_{D_{wing}} = C_{D_{parasite}} + C_{D_{induced}} = C_d + \frac{C_L^2}{\pi A e} \tag{3.48}$$

As the tail and fuselage are not considered to produce lift, they only cause parasite drag. This parasite drag is computed by the component buildup method, where the flat-plate skin friction coefficient, C_f, and the component "form factor", FF, which estimates the pressure drag due to viscous separation, are calculated. The interference effects on the drag are cal-

culated as a factor Q, thus the parasite drag is estimated by Equation 3.49.

$$C_D = \frac{C_f \; FF \; Q \; S_{wet}}{S} \tag{3.49}$$

The measurement of the skin-friction drag requires the consideration of the flow conditions across the aircraft's surfaces. In a composite aircraft with a carefully design, the laminar flow covers around 50% of the wings and tails, as well as 25% of the fuselage. Equation 3.50 describes the skin friction coefficient for laminar flow regions, whereas Equation 3.51 is used for turbulent flow regions.

$$Laminar: \quad C_f = 1.328/\sqrt{Re} \tag{3.50}$$

$$Turbulent: \quad C_f = \frac{0.455}{(\log_{10} Re)^{2.58}} \tag{3.51}$$

The formulations in Equations 3.50 and 3.51 are naturally dependent on the Reynolds number, Re, which acts as an indicator for flow type. Equation 3.52 determines the Reynolds number by dividing the product of characteristic length, l, and velocity, V, by the ratio of the air's density and viscosity, ν. The characteristic length, l, assumes the value of the chord, c, for wings and tails, and the fuselage length, l_{fus}, for the fuselage.

$$R_e = \frac{Vl}{\nu} \tag{3.52}$$

In instances where the skin surface is particularly rough, the flow tends to be more turbulent. This correction is done by introducing a hypothetical "cutoff Reynolds number," which is calculated by Equation 3.53 using the characteristic length l and a skin-roughness parameter k. For a smooth molded composite, k is assumed to be 0.052×10^{-5} m. If the estimated cutoff Reynolds number is less than the actual Reynolds number, the roughness amplifies the drag. Hence, the cutoff Reynolds number is used in Equations 3.51 and 3.50.

$$R_{cutoff} = 38.21 \, (l/k)^{1.053} \tag{3.53}$$

To account for the pressure drag induced by the flow separation, the estimated flat-plate skin-friction coefficient must be increased. This modification is done by an empirical "form

factor," which is developed from a combination of theoretical and empirical findings. Equations 3.54 and 3.55 define the form factor for estimating drag in subsonic flow.

$$Wing\ and\ tail: \quad FF = \left[1 + \frac{0.6}{(x/c)_m}\left(\frac{t}{c}\right) + 100\left(\frac{t}{c}\right)^4\right] \tag{3.54}$$

$$Fuselage: \quad FF = \left(0.9 + \frac{5}{f^{1.5}} + \frac{f}{400}\right) \tag{3.55}$$

where,

$$f = \frac{l}{d} = \frac{l}{\sqrt{(4/\pi)A_{max}}} \tag{3.56}$$

The interaction among the components also increases the aircraft's drag. This interference drag is denoted by the symbol Q in Equation 3.49. The interference effects on tail surfaces range from approximately 3% for a clean V-tail to $4-5\%$ for a standard tail configuration. Conversely, negligible interference is observed in high, mid, or well-faired low wings, resulting in a Q factor close to 1.0. Similarly, the interference factor of the fuselage remains low in most scenarios, hence yielding $Q = 1.0$.

The drag method described above is suitable for streamlined objects like wings, tails, and fuselages. However, for objects like the landing gear, experimental data and empirical approaches are used, where drag is expressed as the drag-to-dynamic pressure ratio (D/q), also known as "drag area".

Reference [59] provides the drag coefficient, $C_{D_{wheel}}$, based on the wheel frontal area for a retracted wheel in the fuselage. The drag-to-dynamic pressure ratio is calculated by multiplying the wheel area by the drag coefficient. The landing gear parasite drag coefficient is calculated by dividing (D/q) by the wing reference area, as shown in Equation 3.57.

$$C_{D_{ld}} = \frac{C_{D_{wheel}}\, S_{wheel}}{S} \tag{3.57}$$

The final step to solve the motion equations is to estimate the thrust, T, produced by the propulsion system. For an electric propelled aircraft, the output thrust depends mainly on the air density, free stream velocity, and the thrust setting, $T = T(\rho, V, \delta)$. Thus, thrust studies are conducted through wind tunnel tests.

These tests evaluate the thrust values at different free stream velocities while maintaining

constant air density and thrust settings. If assumed that the effect of air density variations on thrust during the competition course is negligible. Then, if the atmospheric conditions during the tests are similar to those encountered during the course, thrust data from wind tunnel tests can be applied directly.

Also, within the competition course, each flying phase requires maximum thrust from the propulsion system. Thus, the thrust setting is always set at full throttle. Simplifying, the propulsion system equation to be solely dependent on the free stream velocity, $T = T(V)$.

The work of Zombori [60] provides the thrust data for the propulsion system used in case studies. Figure 3.6 depicts this data.

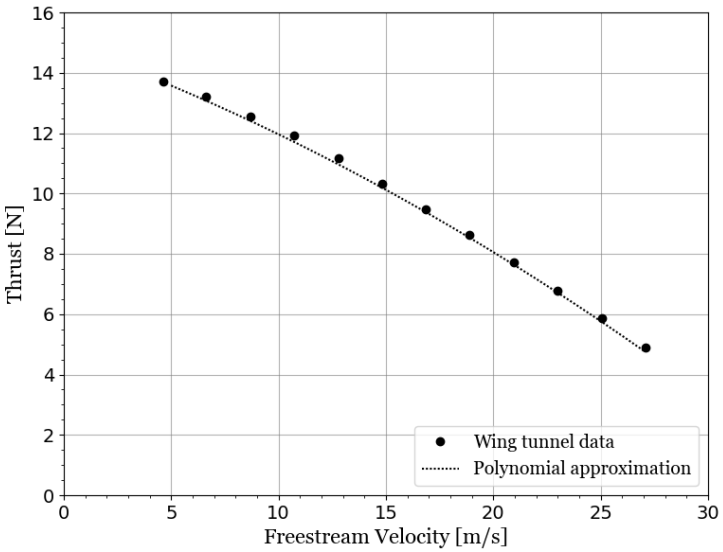


Figure 3.6: Wind tunnel full throttle thrust as functions of the free-stream velocity.

A polynomial approximation approach is used to estimate the thrust at different free-stream velocities. A satisfactory approximation is achieved using a second-degree polynomial, presented in Equation 3.58.

$$T(V) = -0.0045V^2 - 0.2554V + 14.964 \tag{3.58}$$

3.4 Optimization Set Up

Particle Swarm Optimization with inertia weight (PSO-IW) is the optimization approach used in this work and described in Section 2.3. This optimization strategy was incorporated into the XOPTFOIL algorithm [48], with Palmeira's slight adjustments outlined in [47]. This section describes the implementation of the PSO in the selected tool.

In airfoil optimization, each ith particle within the population N_{pop} defines an airfoil through its position vector $\mathbf{X}_i$, expressed by an array of design variables. These particles, during the optimization, iteratively evolve by the process described in Section 2.3.2 and summarized by equations 2.5.

During the optimization, three distinct types of inertial weight are implemented. The first type, designated as "standard", employs a linearly decreasing inertial weight strategy, defined by the recursive Equation 3.59.

$$w(t+1) = w(t) + w_{rate} \tag{3.59}$$

In addition, the other two types, labeled "exhaustive" and "quick", utilize a natural exponent inertial weight, as depicted by Equation 2.8a and described in Section 2.3.3. The implementation of these variants also follows a recursive formula outlined by Equation 3.60.

$$w(t+1) = w(t) - w_{rate}(w(t+1) - w_{end}) \tag{3.60}$$

In addition to varying the inertial distributions, the distinct settings also employ different inertial weight ranges, $[w_{start}, w_{end}]$, as well as different cognitive c_1 and social c_2 parameters within PSO optimization equations, see Equation 2.5. The classification of each type is summarized in Table 3.1.

Table 3.1: Parameters of different types of optimization.

Parameter	c_1	c_2	w_{start}	w_{end}	w_{rate}
Standard	1.49	1.49	0.9	0.4	$\frac{w_{end}-w_{start}}{t_{max}-1}$
Exhaustive	1.4	1.0	1.8	0.8	0.02
Quick	1.2	1.2	1.4	0.6	0.05

Upon tuning the PSO optimization, it proceeds to compute the particle's initial positions and velocities. This process begins by setting the upper $x_{d,max}$ and lower $x_{d,min}$ bounds of each design variable. These limits are defined by Equation 3.61, which incorporates both relative perturbation (ξ_{rel}) and absolute perturbation (ξ_{abs}) terms set by the user. This procedure is performed for the total number of design variables, N_{DV}.

$$x_{max,d} = x_{seed,d} + x_{seed,d}\,\xi_{rel} + \xi_{abs}\ ,\ \forall d \in [1, N_{DV}] \tag{3.61}$$

$$x_{min,d} = x_{seed,d} - x_{seed,d}\,\xi_{rel} - \xi_{abs}\ ,\ \forall d \in [1, N_{DV}] \tag{3.62}$$

The initial population is then generated randomly within the limits defined for the design variables, as outlined in Equation 3.63. Note that $rand_{(0,1)}$ acts as an uniformly random function from the interval $[0, 1]$.

$$x_{i,d}(0) = (x_{max,d} - x_{min,d})\,rand_{(0,1)} + x_{min,d}\ ,\ \forall d \in [1, N_{DV}]\ ,\ \forall i \in [2, N_{pop}] \tag{3.63}$$

However, there is an exception to ensure that at least one particle fulfills all the constraints imposed. So the initial population's first particle is set equal to the seed airfoil. This process is essential to guarantee that the optimization establishes the global best and forces convergence to positions near it.

$$x_{1,d}(0) = x_{seed,d}\ ,\ \forall d \in [1, N_{DV}] \tag{3.64}$$

The process of generating the particle's initial velocity adheres to a similar principle utilized for the initial positions. Each particle's velocity is determined by the product of a uniformly random function sampled from $[-1, 1]$, denoted as $rand_{(-1,1)}$, and the maximum velocity input, v_{max}, as described in Equation 3.65.

$$v_{i,d}(0) = v_{max}\ rand_{(-1,1)}\ ,\ \forall d \in [1, N_{DV}]\ ,\ \forall i \in [1, N_{pop}] \tag{3.65}$$

Equations 2.5, combined with the Equations 3.63 and 3.65, might generate particles beyond the established boundaries. Thus, to handle such constraints, the particle's position and velocity are updated to bring them into the design space, as described by Juárez-Castilho [61]. For the position is used a technique known as projection, where the design variables are changed to the violated boundary value. For the velocity the direction is reverted, forcing the particle to go back, and the magnitude is multiplied by an uniformly random function from [0,1], $rand_{(0,1)}$, a method known as random back. This constraint handling for each particle's design variable is summarized in Equations 3.66 and 3.67.

$$x_{i,d}(t) = \begin{cases} x_{max,d}\ ,\ if\ x_{i,d}(t) > x_{max,d} \\ x_{min,d}\ ,\ if\ x_{i,d}(t) < x_{min,d} \end{cases} ,\ \forall d \in [1, N_{DV}]\ ,\ \forall i \in [1, N_{pop}] \tag{3.66}$$

$$v_{i,d}(t) = \begin{cases} -v_{i,d}(t) rand_{(0,1)}\ ,\ if\ x_{i,d}(t) > x_{max,d} \\ -v_{i,d}(t) rand_{(0,1)}\ ,\ if\ x_{i,d}(t) < x_{min,d} \end{cases} ,\ \forall d \in [1, N_{DV}]\ ,\ \forall i \in [1, N_{pop}] \tag{3.67}$$

Another constraint that remains to be addressed is the particles' maximum velocity. When a particle's velocity exceeds the maximum value, the modulus of its velocity vector, $\mathbf{V}_i$, is changed to comply with the imposed velocity limit, Equation 3.68. This bound prevents the particle from moving rapidly away from the reference.

$$\mathbf{V}_i(t) = \frac{V_{max}}{||\mathbf{V}_i(t)||}\mathbf{V}_i(t)\ ,\ \forall i \in [1, N_{pop}] \tag{3.68}$$

The optimization process stops when one of the stop conditions is triggered. The algorithm contains two separate stop conditions, each of them being sufficient to end the optimization:

- Reaching the maximum number of iterations, t_{max} .
- Achieving population convergence. The population diversity is assessed using the design radius r, and once r reaches the predefined limit r_{limit}, the population is considered converged.

To compute the design radius, it is necessary to scale every design variable between [-1,1] based on the design limits, as outlined in Equation 3.69.

$$\bar{x}_{i,d} = -1 + 2(x_{i,d} - x_{min,d})/(x_{max,d} - x_{min,d})\ ,\ \forall d \in [1, N_{DV}]\ ,\ \forall i \in [1, N_{pop}] \tag{3.69}$$

Subsequently, the centroid of each design variable, $\bar{x}_{c,d}$, is calculated using Equation 3.70.

$$\bar{x}_{c,d} = \frac{\sum_{i=0}^{N_{pop}} \bar{x}_{i,d}}{N_{pop}}\ ,\ \forall d \in [1, N_{DV}] \tag{3.70}$$

Lastly, the design radius is expressed as the population average distance between the position vector $\bar{\mathbf{X}}_i$ and the centroid vector $\bar{\mathbf{X}}_c$ divided by the number of design variables for each particle, as defined in Equation 3.71.

$$r = \frac{\sum_{i=0}^{N_{pop}} \frac{||\bar{\mathbf{X}}_i - \bar{\mathbf{X}}_c||}{N_{DV}}}{N_{pop}} \tag{3.71}$$

Chapter 4

Case studies analysis

This chapter reveals and discusses the results of the applied approach across three separate sections. The Section 4.1 presents the general features of the aircraft and the score obtained in the competition, as well as the competition characteristics. Section 4.2 evaluates the PSO-IW algorithm alongside the newly implemented objective function. This includes assessing the impact of some PSO-IW parameters on design space exploration and also examining the effect of a new objective function on the optimized airfoil. Finally, the Section 4.3 examines the effect of different initial conditions on the optimization outcome.

4.1 Aircraft and competition features

The ACC handbook [1] provides details on the competition area, including flight limits, and runway type and dimensions, which are summarized in Table 4.1.

Table 4.1: Airfield characteristics.

Flight area length	Flight area width	Runway altitude	Runway pavement	Max runway length	Max climb altitude
[m]	[m]	[m]	–	[m]	[m]
375	425	474	Grass	60	120

Also available from the organizing team, the official results [62] provide the best performance of the UBI team aircraft across the five rounds, along with the highest values achieved by any aircraft, used as reference points. These values are compiled in Table 4.2.

Table 4.2: Competition performance and reference values.

$W_{payload}$	Δh	$dist$	$W_{payload,ref}$	Δh_{ref}	$dist_{ref}$
[N]	[m]	[m]	[N]	[m]	[m]
31.47	101	3255	31.47	100	3255

The characteristics of the composite aircraft developed to participate in the ACC competition and shown in Figure 1.2 are presented in Tables 4.3, 4.4, and 4.5. Table 4.3 presents an overview of the aircraft's general features and its wing. It lists the aircraft's empty weight W_{empty}, as well as the wing area S, aspect ratio A, Oswald efficiency factor e_w, exposed

area S_{expose}, and the airfoil used. Additionally, it provides the maximum payload weight $W_{payload,max}$, based on the aircraft's compartment capacity, and the minimum payload weight $W_{payload,min}$, required to achieve longitudinal stability.

Table 4.3: Aircraft general characteristics.

W_{total}	W_{empty}	$W_{payload,min}$	$W_{payload,max}$	S	A	e_w	S_{expose}	Wing airfoil
[N]	[N]	[N]	[N]	[m²]	–	–	[m²]	–
41.59	23.92	11.77	32.36	0.426	11.3	0.98	0.402	S9000

Table 4.3 focuses on the fuselage dimensions, including height h_{fus}, width w_{fus}, length l_{fus}, and exposed surface area $S_{wetted,fus}$, along with fuselage characteristics such as skin roughness k_{fus} and interference factor Q_{fus}. Furthermore, it covers the landing gear dimensions and characteristics, including the wheel frontal area A_{wheel} and the drag-to-dynamic pressure ratio D/q_{lg}.

Table 4.4: Fuselage and landing gear characteristics.

h_{fus}	w_{fus}	l_{fus}	k_{fus}	$S_{wetted,fus}$	Q_{fus}	A_{wheel}	$(D/q)_{lg}$
[m]	[m]	[m]	[m]	[m²]	–	[m²]	–
0.175	0.124	0.710	6.34E-6	0.274	1.0	0.0024	0.09

Finally, Table 4.4 details the tail configuration and dimensions, such as the mean chord c_{tail}, and surface area S_{tail}. In addition, the tail airfoil and its characteristics are presented, including the maximum thickness chord ratio $(t/c)_{tail}$, and the maximum-to-thickness relative position $(x/c)_{m,tail}$.

Table 4.5: Tail characteristics.

Configuration	Airfoil	c_{tail}	S_{tail}	$(t/c)_{tail}$	$(x/c)_{m,tail}$	k_{tail}
–	–	[m]	[m²]	–	–	[m]
V-tail	NACA0009	0.125	0.027	0.09	0.3	6.34E-6

Based on the aircraft performance data from the competition, presented in Table 4.2, the flight conditions in each phase can be approximated. Table 4.6 outlines the take-off condition, including the runway altitude h_{runway} and length used for take-off s_{to}, along with the loaded payload $W_{payload}$. It also details the take-off properties, such as the aircraft's maximum lift coefficient $C_{L_{max}}$, which determines the stall speed V_S, and the rotation factor A_1, which sets the lift-off speed V_{lo} and the average run speed V_{av}. Additionally, it shows the aircraft's lift coefficient ($C_{L_{run}}$), drag coefficient ($C_{D_{run}}$), and thrust force (T_{run}) under the ground run condition. Finally, Table 4.6 includes the airfoil take-off conditions, such as the

maximum lift coefficient $C_{L_{max}}$, the ground run airfoil lift $C_{l_{run}}$, and drag $C_{d_{run}}$ coefficients, in addition to their respective angle of attack α_{lo} and α_{run}, and flap deflection δ_{lo} and δ_{run}.

It is important to note that the normal friction coefficient of a grass runway is $\mu = 0.07$, as indicated by [58]. However, during the competition, the aircraft experienced difficulties due to tufts of grass, which interfered with the acceleration on the runway. Therefore, the friction coefficient was adjusted to $\mu = 0.2626$ to better portray the actual performance of the aircraft during the competition.

In addition, an extra drag coefficient $C_{d_{extra}}$ is introduced to account for aerodynamic factors not considered by the implemented model. These factors include the rear slope and the pressure drag associated with the aft section of the fuselage, along with the discontinuities between components and manufacturing imperfections. The inclusion of $C_{d_{extra}}$ is also necessary to match the aircraft's estimated performance with the competition score, outlined in Table 4.2.

Table 4.6: Take-off condition.

h_{runway}	s_G	$W_{payload}$	δ_{lo}	$C_{l_{max}}$	$C_{L_{max}}$	V_S	A_1	V_{LOF}
[m]	[m]	[N]	[deg]	–	–	[m/s]	–	[m/s]
474.0	60.00	17.67	25.00	1.713	1.455	10.71	1.100	11.78
V_{av}	δ_{run}	α_{run}	$C_{l_{run}}$	$C_{d_{run}}$	$C_{L_{run}}$	$C_{D_{run}}$	T_{run}	μ
[m/s]	[deg]	[deg]	–	–	–	–	[N]	-
8.330	0.000	7.000	1.033	0.02022	0.9743	0.06498	12.52	0.263

For the climb phase, Table 4.7 exposes the condition where the aircraft has its maximum rate of climb RC_{max}, considering an average altitude h_{climb}. In this condition, the horizontal speed where the best rate of climb is achieved $V_{RC_{max}}$, the aircraft lift $C_{L_{climb}}$, and drag $C_{D_{climb}}$, coefficients, and the drag D_{climb}, and thrust T_{climb}, forces are presented. Also, the airfoil's lift $C_{l_{climb}}$ and drag $C_{l_{climb}}$ coefficients are presented with the corresponding attack angle α_{climb}, flaps deflection δ_{climb}, and Reynolds number Re_{climb}, in which they are achieved.

Table 4.7: Climb condition.

h_{climb}	RC_{max}	$V_{RC_{max}}$	δ_{climb}	α_{climb}	Re_{climb}
[m]	[m/s]	[m/s]	[deg]	[deg]	–
524.0	2.838	15.53	4.000	1.801	1.924E5
$C_{l_{climb}}$	$C_{d_{climb}}$	$C_{L_{climb}}$	$C_{D_{climb}}$	D_{climb}	T_{climb}
–	–	–	–	[N]	[N]
0.7400	0.009400	0.6981	0.03880	2.312	9.910

Regarding the free flight phase, it is divided into two conditions. Table 4.8 refers to the level flight component that includes the aircraft lift $C_{L_{cruise}}$ and drag $C_{D_{cruise}}$ coefficients, and the drag D_{cruise} and thrust T_{cruise} forces for the maximum speed V_{max}. Furthermore, the airfoil's lift $C_{l_{cruise}}$ and drag $C_{l_{cruise}}$ coefficients are presented, as well as the angle of attack α_{cruise}, flaps deflection δ_{cruise}, and Reynolds number Re_{cruise}.

Table 4.8: Level flight condition.

h_{cruise}	V_{cruise}	δ_{cruise}	α_{cruise}	Re_{cruise}	$C_{l_{cruise}}$	$C_{d_{cruise}}$	$C_{L_{cruise}}$	$C_{D_{cruise}}$	T_{cruise}
[m]	[m/s]	[deg]	[deg]	–	–	–	–	–	[N]
574.0	27.91	0.000	-0.9559	3.451E5	0.2293	0.006647	0.2164	0.02254	4.332

Similarly, Table 4.9 refers to the other free flight phase, the sustained turn, and includes the aircraft lift $C_{L_{turn}}$ and drag $C_{D_{turn}}$ coefficients, and the drag D_{turn} and thrust T_{turn} forces for the maximum turn speed V_{turn}. The airfoil's lift $C_{l_{turn}}$ and drag $C_{l_{turn}}$ coefficients are also provided, along with the angle of attack α_{turn}, flaps deflection δ_{turn}, and Reynolds number Re_{turn}.

Table 4.9: Sustained turn condition.

h_{turn}	V_{turn}	n_{turn}	δ_{turn}	α_{turn}	Re_{turn}	$C_{l_{turn}}$	$C_{d_{turn}}$	$C_{L_{turn}}$	$C_{D_{turn}}$	T_{turn}
[m]	[m/s]	–	[deg]	[deg]	–	–	–	–	–	[N]
574.0	26.71	2.000	0.000	1.567	3.303E5	0.5008	0.007006	0.4725	0.02803	4.934

4.2 Case study 1 - Particle Swarm Optimization

4.2.1 Objective and problem definition

This chapter serves two purposes. First, it intends to evaluate the algorithm's performance across multiple PSO-IW setups through a series of experimental tests. These multiple optimizations try to establish a foundation for understanding the disparity between runs, which is critical for predicting possible result dispersion in future optimization situations.

Second, this case study examines the proposed method using the outcomes of several executions. This study will not only provide insight into the current strengths and weaknesses of the current optimization problem but will also serve as a guide for future enhancements.

To properly configure an inertial weight particle swarm optimization (PSO-IW) algorithm, it is important to adjust key variables, such as the number of iterations, population size, inertia weight distribution, and thrust factors that govern particle behavior. Moreover,

setting the particle's maximum velocity and ensuring sufficient design constraints are important to establishing the design space. All of these variables must be balanced with the available computational resources to achieve an accurate and efficient optimization.

Piotrowski investigates the influence of a population size on several single-objective PSO algorithms in [63]. The study reveals that swarms with less than 40 individuals tend to perform poorly when solving real-world problems using PSO-IW. On the other hand, larger populations, between 70 and 500 individuals, generate better optimized results. However, to employ such large swarms, a lot of computational power is required to ensure an optimization within a feasible time frame. For the presented analysis, a population size of 40 particles was selected, taking into consideration the computational resources available, which is consistent with Palmeira's options in [47].

The computational power and the available time for the optimization also impose limits on the number of iterations that the population undergoes. To achieve an algorithm that balances accuracy and efficiency, a total of 500 iterations was selected, also consistent with the work of Palmeira [47].

In this case study, two different exponential inertial weight curves will be examined and compared to the standard linear distribution, one of the most commonly used approaches in various applications and originally proposed by Shi and Eberhart in their pioneering work on PSO-IW [31]. The exponential distribution is initially used to disperse the particles through the design space for a good exploration. Then, as the inertial weight values rapidly decrease, to gradually converge the particles into the optimal design. In contrast, a linear curve provides a smoother transition between exploration and exploitation stages [37]. The three inertial weight distributions are already integrated into the program, as detailed in Section 3.4.

Table 3.1 compares the exponential distribution parameters with the standard distribution. Both exponential distributions have a lower sum of the thrust factors, leading to a quicker convergence towards the particles' reference values. Yet, this is compensated by higher initial and final inertial values compared to the standard distribution. These conclusions are based on data detailed in Section 2.3.3.

Between the two exponential distributions, the "exhaustive" distribution has both a higher initial and final inertial weight distribution value than the "quick" distribution. Thus, as implied by its name, the exhaustive distribution will disperse the particles more and converge them more slowly than the quick distribution. Furthermore, the exhaustive distribution emphasizes more the particles' own thinking by increasing the relative weight assigned to the cognitive thrust factor compared with the social factor. This lowers the convergence rate in

the exhaustive distribution even more.

In conclusion, the case study is interesting to check if lower thrust factors of the exponential distribution are balanced by the greater inertia weights, as well as to analyze who produces the best score and consistency.

4.2.2 Optimization set up

The initial stage in this study is to carefully choose the operating point conditions for optimization. This stage is critical since the definition of these conditions has a major impact on the optimization outcome. Consequently, the operating point conditions are established based on the actual flight conditions experienced by the aircraft during competition. Table 4.10 summarizes the operating points defined for optimization.

Table 4.10: Initial operating conditions for Case study 1.

Operating points	Re	α	C_l	M	hinge x	δ
1	151000	5^o to 9^o , step 1^o	–	0.03	0.80	25.00
2	107000	7^o	–	0.02	0.80	0.00
3	151000	–	1.20	0.03	0.80	4.00
4	162000	–	1.05	0.04	0.80	4.00
5	174000	–	0.90	0.04	0.80	4.00
6	191000	–	0.75	0.04	0.80	4.00
7	214000	–	0.60	0.05	0.80	4.00
8	302000	–	0.60	0.07	0.80	0.00
9	315000	–	055	0.07	0.80	0.00
10	331000	–	0.50	0.08	0.80	0.00
11	348000	–	0.45	0.08	0.80	0.00
12	234000	–	0.50	0.05	0.80	0.00
13	302000	–	0.30	0.07	0.80	0.00
14	331000	–	0.25	0.08	0.80	0.00
15	370000	–	0.20	0.09	0.80	0.00

The purpose of setting the first two operating points is to maximize the payload capacity by optimizing the take-off flight stage. The first point specifically focuses on finding and improving the maximum lift coefficient. The second point is to improve the average aerodynamic coefficients in the ground run configuration with a 7 degree angle of attack.

The third through seventh points are used to estimate and enhance the best climb speed and the time required to accelerate to the corresponding horizontal speed. Since the reference best climb speed is reached at C_l of 0.74, as specified in Section 4.1, one point closer, below and above this value, are necessary. Furthermore, to estimate and improve the acceleration phase, two additional higher points are added. Given the wide range of C_l values

involved in the advancement of the climb flight phase, a step of 0.15 is used to properly distribute the operating points.

The estimation and optimization of the traveled distance are divided into two components. The eighth to eleventh points are responsible for optimizing the sustained turn speed. The reference turn condition has a C_l of 0.5008, so points closer, lower, and higher than the reference are chosen with a step of 0.05 C_l. Additionally, since it is more difficult to increase speed than to decrease it, a fourth point with a C_l of 0.6 is added.

Finally, the twelfth through fifteenth points are used to optimize the cruise speed. For level flight, in the reference condition, the lift coefficient is $C_l = 0.2293$. To maintain consistency, one point slightly below and above this value are chosen, rounding them to the nearest multiple of 0.05. Furthermore, following the same logic as in the sustained turn optimization, an additional higher point is added to account for large declines in the cruise speed. Additionally, a third higher point is included to estimate the acceleration time from the climb speed to the cruise speed. This point is positioned approximately between the last defined point and the closest point to the climb speed reference.

The next stage involves the airfoil parameterization and design variables. A B-spline approach, described in Section 3.2.3, is used based on Palmeira's results, which have shown to be the more effective parameterization method. Each airfoil surface is characterized by a third degree B-spline with 8 control points. The control points' x-coordinates are fixed and spread according to a cosine distribution. Since both the leading and trailing edges are fixed, as stated in Section 3.2.3, this generates 6 design variables per surface, which are the z-coordinates of the control points between the leading and trailing edges.

Furthermore, the position of the flap hinge and the flap deflection in three distinct flight phases are also optimized, adding 4 additional design variables to the 12 required to define the airfoil surface. Table 4.11 provides the initial design variables, enumerated sequentially from the leading edge to the trailing edge.

Table 4.11: Initial design variables.

Top - 1	Top - 2	Top - 3	Top - 4	Top - 5	Top - 6
0.024245	0.058612	0.101504	0.102389	0.070469	0.034597
Bot - 1	Bot - 2	Bot - 3	Bot - 4	Bot - 5	Bot - 6
-0.009835	-0.024519	-0.012103	0.012543	0.026109	0.026518
$hinge_x$	$\delta_{take-off}$	δ_{climb}	δ_{cruise}		
0.8000	25.00	4.000	0.000		

The design space for the optimization is established by the types of constraints imposed.

The first, mentioned in Section 3.4, Equation 3.69 limits the control points' z coordinates. On top of that, to avoid the optimization from converging into unfeasible designs, the following geometric constraints are defined:

- The maximum allowable thickness ratio is constrained between 0.09 and 0.25;
- Two additional thickness constraints are imposed at the x/c positions of 0.9 and 0.8, with minimum thicknesses of 0.0152 and 0.0328, respectively;
- The camber is restricted to a range between -0.1 and 0.8;
- The minimum, maximum and difference allowable values of the leading edge angle are 60, 89.99 and 20 degrees, respectively;
- The minimum trailing edge wedge angle is set at 5 degrees;
- The number of curvature reversals on the top and bottom surfaces is limited to 2 and 3, respectively;
- The bottom surface of the airfoil's flap hinge location is limited to a range between 0.7 and 0.9 along its x/c position;
- Lastly, the flap deflection angle has a range of -10 to 25 degrees.

4.2.3 Results and discussion

The resulting airfoil shapes from the multiple runs are presented in Figure 4.1, alongside the initial airfoil. Table 4.12, which compares the performance score and airfoil shape consistency across several PSO-IW configurations, shows that the standard and exhaustive settings provide better and more consistent results compared with the quick option. The data in the Table 4.12 depicts the clear superiority of the exhaustive and standard options, which achieve better average performance metrics and lower discrepancies between scores of the multiple runs. The better consistency of results is also confirmed by a smaller design radius from the average airfoil. Furthermore, Table 4.12 also attributes a slightly better result to the exhaustive configuration compared with the standard option in all parameters, except in the higher score that was achieved by the standard setting.

These differences can be explained by the design radius history displayed in Figure 4.2. In this figure, the exhaustive option maintains a higher design radius for a greater number of iterations. Thus, keeping the optimization in an exploration state for an extended period.

This prolonged exploration state increases the probability of finding the global maximum region, thus obtaining more consistent results. Conversely, the quick setting converges rapidly to a solution, resulting in an unreliable design.

Table 4.12: PSO configurations results for Case 1.

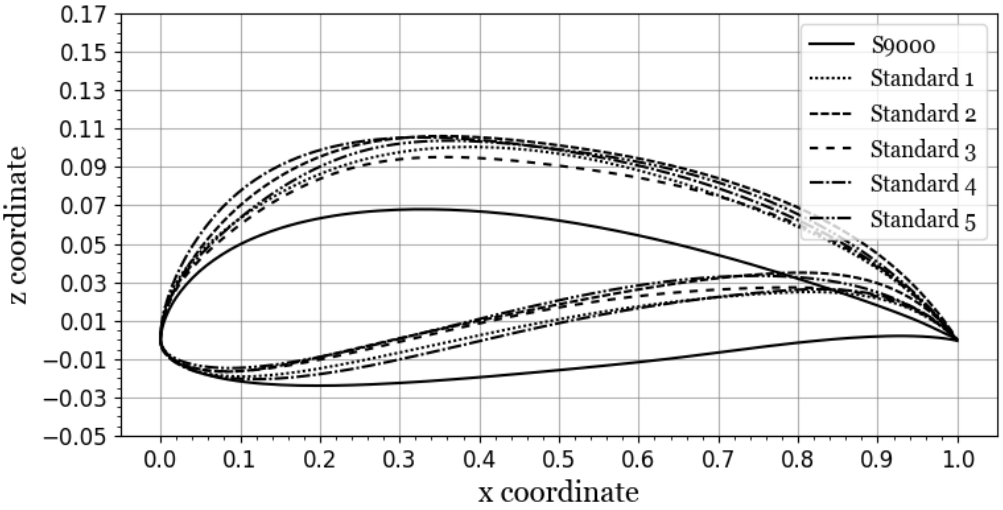
Option	Lower score	Average score	Upper score	Absolute score different	Relative score different	Design radius
Standard	886.7	890.1	895.0	4.863	0.5463	3.632e-3
Quick	872.8	880.5	889.5	9.057	1.029	7.193e-3
Exhaustive	889.8	892.5	893.8	2.687	0.3011	3.335e-3

The standard option acts as a middle term, with the particles' dispersion lower than the exhaustive setting and a slower convergence than the quick setting. This gives the standard option a good probability to find the global maximum zone and also time to converge into it. This balance can probably justify the standard option of obtaining the best score among all runs. Although the exhaustive can find the global maximum zone more times as it converges slowly, it needs more iterations than the standard option to convert into the best score.

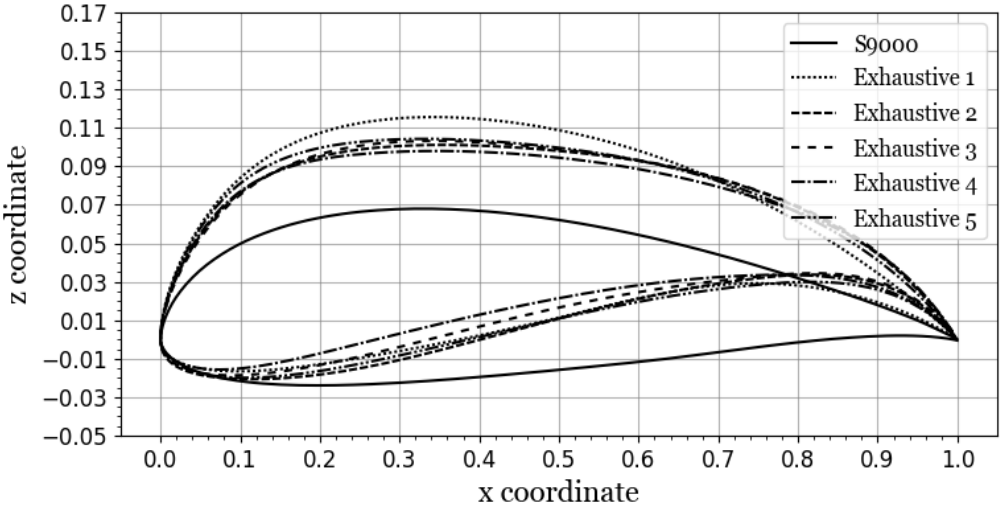
The importance of the initial particle dispersion in identifying the global maximum is confirmed by Figure 4.3. This figure illustrates the evolution of the objective value throughout the multiple iterations, where it becomes evident that significant improvements occur during the early stages of the optimization process.

In summary, it is clear that, while the exhaustive optimization option needs additional iterations to converge, it yields the most consistent results. However, if optimization time is a crucial factor, the standard technique is a feasible option since it produces good results with fewer iterations than the exhaustive method.

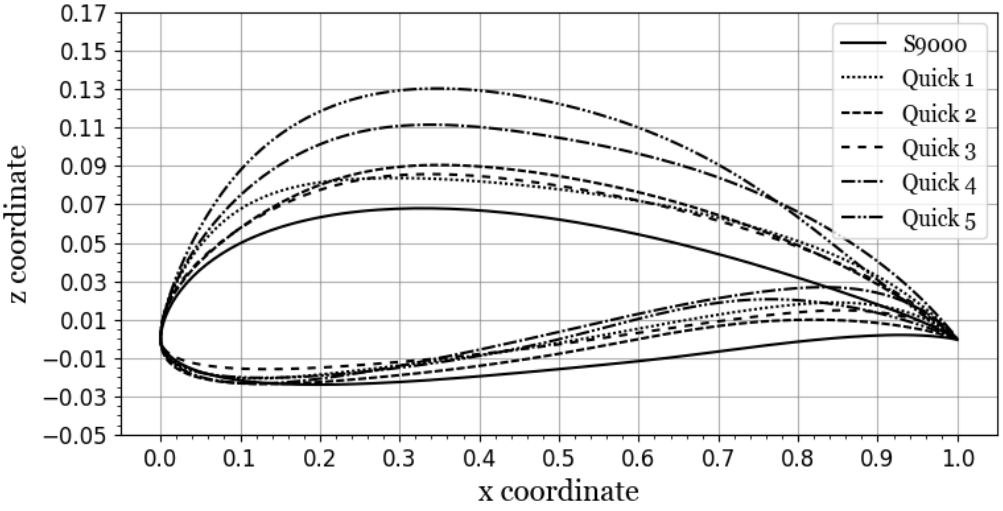
Figure 4.1 also illustrates a pattern in the optimization, where both the airfoil's maximum thickness and camber increase. In accordance to Tables 4.14, 4.15, and 4.16, this tendency aims to raise the payload score by increasing the maximum lift coefficient. The increase allows the aircraft to carry a heavier cargo while preserving the required performance for lift-off in the specified runway length. However, this improvement comes at the expense of other features. The first trade-off is the decrease in the rate of climb. This consequence is not very significant because the reminding rate can still reach an altitude of 100 m, thereby achieving the maximum score in this phase. Secondly and more notably, the improvement in the payload score comes at the cost of the distance traveled during the 120 seconds flight. The increased airfoil thickness and camber reduce the airfoil's aerodynamic efficiency in free flight phase, thereby decreasing the maximum speed in this phase.



(a) Standard

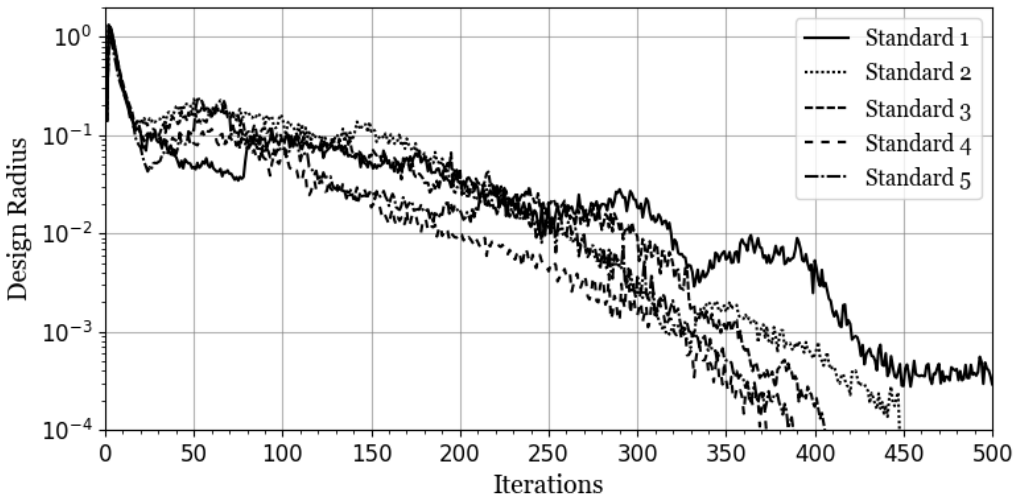


(b) Exhaustive

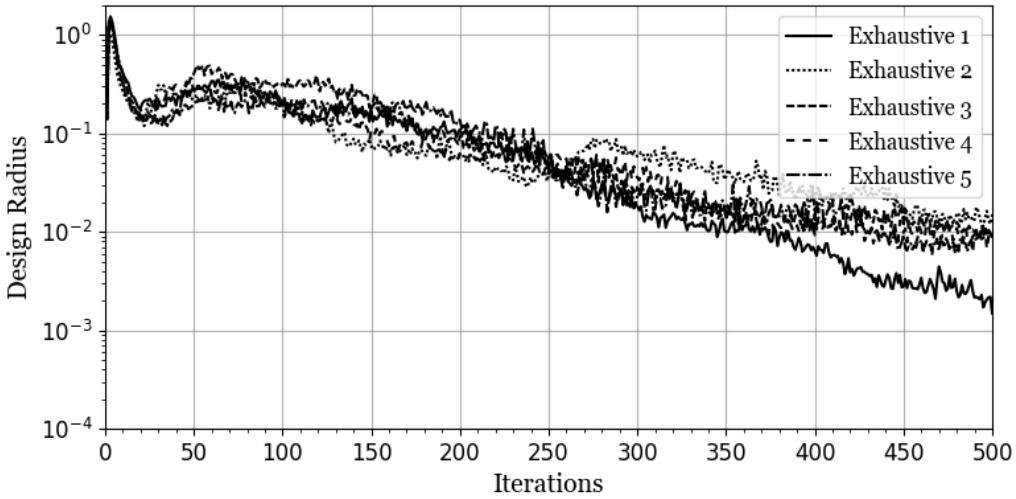


(c) Quick

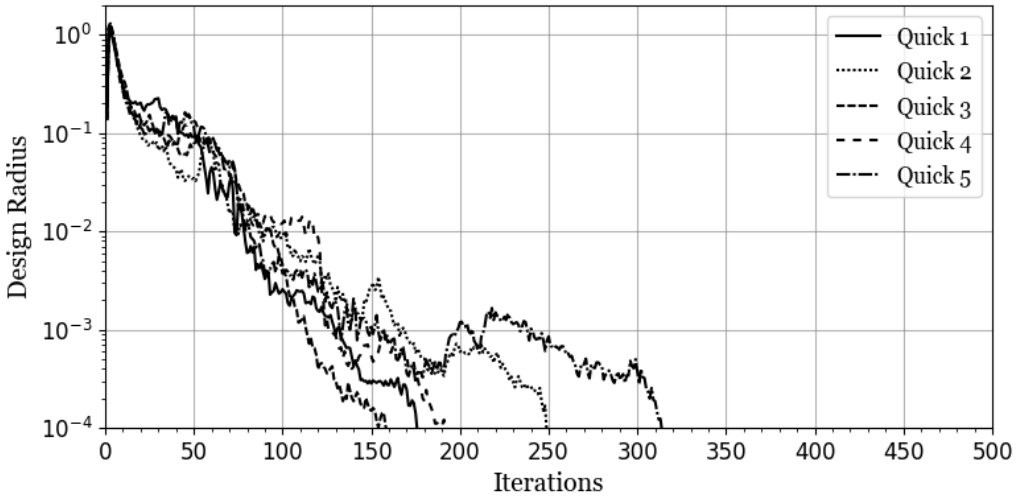
Figure 4.1: Optimized airfoils and the original airfoil (S9000) comparison for all case studies.



(a) Standard

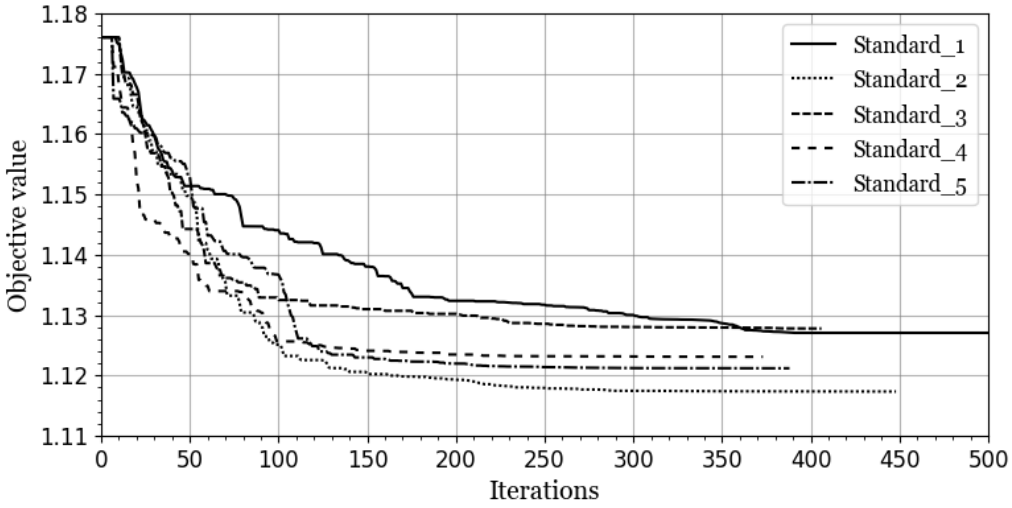


(b) Exhaustive

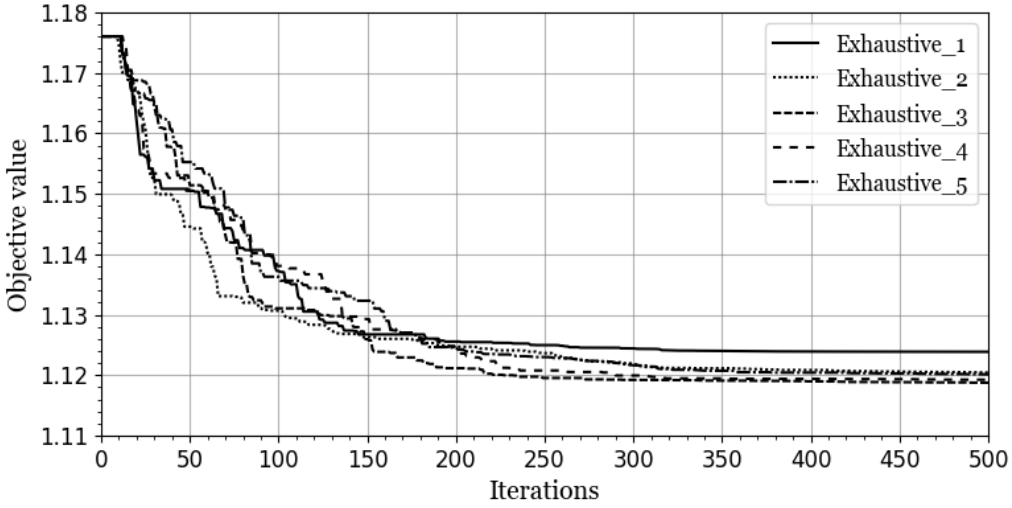


(c) Quick

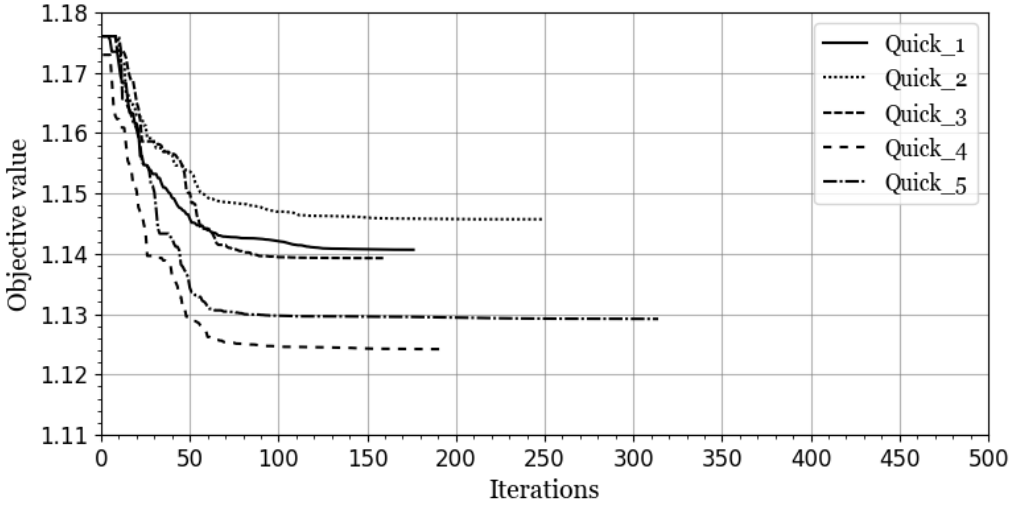
Figure 4.2: Design radius history for all study cases.



(a) Standard



(b) Exhaustive



(c) Quick

Figure 4.3: Objective value history for all study cases.

Another finding is revealed in Table 4.16. Because of the raise in the aircraft's lift-off weight from the payload improvement, the lift coefficient in both level and sustained turn conditions naturally increases. However, in the case of the sustained turn, the increase in lift coefficient is so much that it reaches the upper limit established for the turn optimization. In order to maintain a reliable estimate of the sustained turn conditions, the algorithm limits the lift coefficient to the specified range. As a result, the tendency to increase the airfoil maximum thickness and camber could go even further if this range were expanded. This demonstrates the significance of selecting the starting airfoil and emphasizes the need for a preliminary analysis of the airfoil that is optimized.

Table 4.13: Flap settings results for Case 1.

Case study	$hinge_x$	$\delta_{take-off}$	δ_{climb}	δ_{cruise}
S9000	0.800	25.00	4.000	0.000
Standard 1	0.700	24.92	-2.81	-3.73
Standard 2	0.821	21.92	0.65	-10.0
Standard 3	0.796	24.87	7.74	-7.12
Standard 4	0.736	25.00	-3.82	-5.51
Standard 5	0.737	24.85	-9.10	-6.85
Quick 1	0.772	18.11	-2.33	-5.79
Quick 2	0.751	24.81	0.46	-2.44
Quick 3	0.786	24.85	4.13	-3.97
Quick 4	0.814	21.06	3.53	-9.35
Quick 5	0.774	21.79	7.75	-9.12
Exhaustive 1	0.768	23.91	3.58	-9.99
Exhaustive 2	0.776	22.97	2.46	-9.40
Exhaustive 3	0.802	24.69	-2.42	-9.92
Exhaustive 4	0.714	24.96	3.24	-4.69
Exhaustive 5	0.772	23.58	3.55	-7.38

The Table 4.13 compares the optimized flap settings with the original configuration. The comparison reveals a tendency toward maintaining or slightly decreasing the flap hinge position. Also during lift-off, the flap deflection tends to approach the maximum allowable value, and while in cruising, the flap exhibits a negative deflection. During the climb, the flap deflection is on average close to zero, although it fluctuates significantly throughout the multiple runs. This fluctuation can be explained by the large step size of 0.15 C_l between the operating points. While the big step size is required to cover a wide range of possible climb conditions, it makes it difficult to determine the real best rate of climb.

Table 4.14: Take-off conditions results for Case 1.

Case studies	$S_{payload}$ –	W_{total} [N]	$W_{payload}$ [N]	α_{to} [deg]	$C_{l_{max}}$ –	V_S [m/s]	V_{LOF} [m/s]	$C_{l_{run}}$ –	$C_{d_{run}}$ –	V_{run} [m/s]
Reference	561.5	41.59	17.67	7.000	1.713	10.71	11.78	1.033	0.02022	8.330
Standard 1	689.2	45.61	21.69	9.000	2.026	10.31	11.34	1.472	0.02149	8.022
Standard 2	712.5	46.34	22.42	8.000	1.898	10.74	11.81	1.662	0.02785	8.354
Standard 3	682.8	45.41	21.49	9.000	1.911	10.60	11.65	1.491	0.02293	8.241
Standard 4	698.4	45.90	21.98	9.000	2.077	10.22	11.24	1.535	0.02771	7.947
Standard 5	701.2	45.99	22.07	9.000	1.964	10.56	11.62	1.575	0.02432	8.216
Quick 1	646.2	44.26	20.34	9.000	1.922	10.43	11.47	1.357	0.02966	8.111
Quick 2	629.3	43.73	19.80	9.000	1.847	10.58	11.63	1.261	0.02036	8.226
Quick 3	642.4	44.14	20.21	9.000	1.879	10.53	11.59	1.307	0.02068	8.193
Quick 4	699.2	45.92	22.00	8.000	1.980	10.47	11.51	1.572	0.02777	8.141
Quick 5	689.5	45.62	21.70	8.000	2.058	10.23	11.27	1.490	0.02634	7.959
Exhaustive 1	701.3	45.99	22.07	9.000	2.113	10.14	11.15	1.528	0.02637	7.887
Exhaustive 2	702.3	46.02	22.10	9.000	2.069	10.25	11.28	1.577	0.03116	7.973
Exhaustive 3	708.8	46.20	22.28	9.000	2.019	10.40	11.44	1.613	0.02983	8.087
Exhaustive 4	707.8	46.19	22.27	9.000	2.129	10.13	11.14	1.599	0.03362	7.876
Exhaustive 5	705.2	46.11	22.28	9.000	2.181	9.994	10.99	1.558	0.03191	7.773

Table 4.15: Climb conditions results for Case 1.

Case studies	S_{climb} –	alt [m]	RC_{max} [m/s]	V_{climb} [m/s]	$C_{l_{climb}}$ –	$C_{d_{climb}}$ –	t_{accel} [s]
Reference	1000	100.0	2.821	15.39	0.750	0.00935	1.880
Standard 1	1000	100.0	2.480	16.12	0.750	0.01082	2.850
Standard 2	1000	100.0	2.321	13.73	1.050	0.01477	1.127
Standard 3	1000	100.0	2.243	12.72	1.200	0.02078	0.615
Standard 4	1000	100.0	2.444	16.17	0.750	0.01145	3.002
Standard 5	1000	100.0	2.423	16.19	0.7500	0.01213	2.081
Quick 1	1000	100.0	2.534	15.88	0.750	0.01285	2.571
Quick 2	1000	100.0	2.639	15.78	0.750	0.01000	2.320
Quick 3	1000	100.0	2.547	14.48	0.900	0.01221	1.588
Quick 4	1000	100.0	2.223	13.67	1.050	0.02369	1.284
Quick 5	1000	100.0	2.225	13.63	1.050	0.02492	1.404
Exhaustive 1	1000	100.0	2.304	13.68	1.050	0.01757	1.467
Exhaustive 2	1000	100.0	2.277	13.69	1.050	0.01936	1.436
Exhaustive 3	1000	100.0	2.404	14.81	0.900	0.01258	2.016
Exhaustive 4	1000	100.0	2.307	13.71	1.050	0.01644	1.538
Exhaustive 5	1000	100.0	2.297	13.70	1.050	0.01754	1.615

Table 4.16: Level flight and sustained turn conditions results for Case 1.

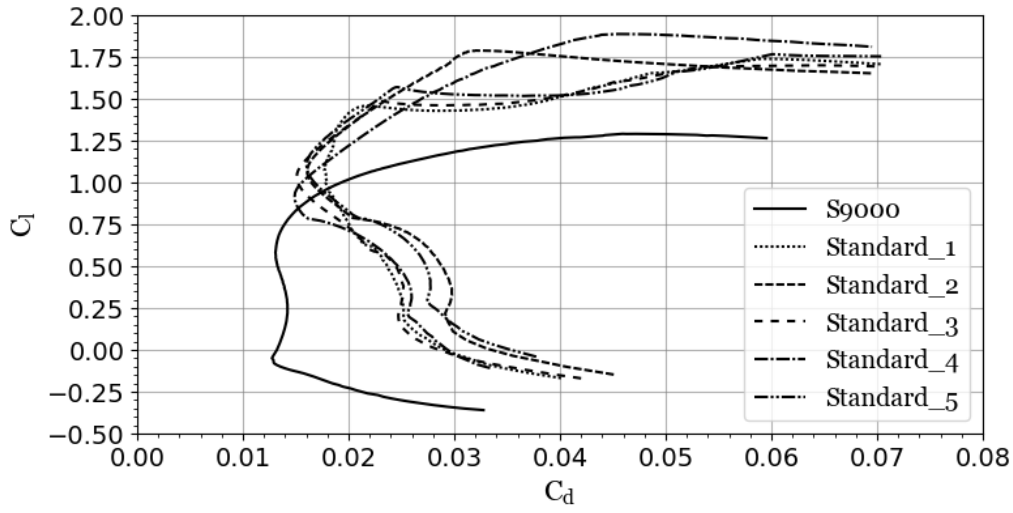
Case studies	$S_{distance}$ –	R_{turn} [m]	V_{turn} [m/s]	$C_{l_{turn}}$ –	$C_{d_{turn}}$ –	V_{cruise} [m/s]	$C_{l_{cruise}}$ –	$C_{d_{cruise}}$ –	t_{accel} [s]	$dist_{accel}$ [m]	$dist$ [m]	S_{total} –
Reference	1000.0	41.33	26.50	0.5087	0.00635	27.70	0.2327	0.00661	25.35	72.86	3255	853.8
Standard 1	972.5	38.77	25.66	0.5948	0.00903	27.01	0.2684	0.00928	23.78	182.8	3166	887.2
Standard 2	972.4	39.05	25.75	0.6000	0.00838	26.92	0.2744	0.00959	27.61	311.2	3165	895.0
Standard 3	977.3	39.44	25.88	0.5819	0.00818	27.10	0.2654	0.00890	26.42	305.8	3181	886.7
Standard 4	972.8	38.67	25.63	0.6000	0.00906	27.04	0.2694	0.00910	24.19	213.3	3166	890.4
Standard 5	972.7	38.801	25.6751	0.5990	0.00884	27.00	0.2708	0.00928	23.95	221.5	3166	891.3
Quick 1	983.9	38.33	25.52	0.5837	0.01017	27.57	0.2500	0.00709	33.85	431.6	3202	876.7
Quick 2	989.2	40.12	26.10	0.5510	0.00794	27.40	0.2500	0.00783	32.25	339.4	3120	872.8
Quick 3	990.8	39.98	26.06	0.5582	0.00796	27.53	0.2500	0.00725	33.21	382.1	3225	877.7
Quick 4	969.4	38.69	25.64	0.6000	0.00902	26.93	0.2719	0.00962	26.51	332.1	3155	889.5
Quick 5	967.2	38.44	25.55	0.6000	0.00950	26.88	0.2711	0.00986	26.47	332.7	3148	885.6
Exhaustive 1	968.1	38.75	25.66	0.6000	0.00892	26.78	0.2684	0.01026	26.00	285.3	3151	889.8
Exhaustive 2	975.2	38.80	25.67	0.5996	0.00883	27.14	0.2744	0.00869	26.66	317.3	3174	892.5
Exhaustive 3	973.5	38.93	25.71	0.6000	0.00859	27.01	0.2654	0.00922	25.78	247.7	3169	893.8
Exhaustive 4	972.6	38.986	25.73	0.5990	0.00852	26.95	0.2730	0.00948	26.89	310.0	3166	972.6
Exhaustive 5	973.0	38.85	25.69	0.6000	0.00873	27.01	0.2713	0.00923	26.68	312.0	3167	892.7

Hence, instead of optimizing the flap deflection at the real best climb condition, the algorithm optimizes the maximum rate of climb at one of the predetermined operating points. Such consideration can also be made in determining the flap deflection that maximizes the lift coefficient of the airfoil. However, the use of smaller steps in the take-off optimization mitigates this issue.

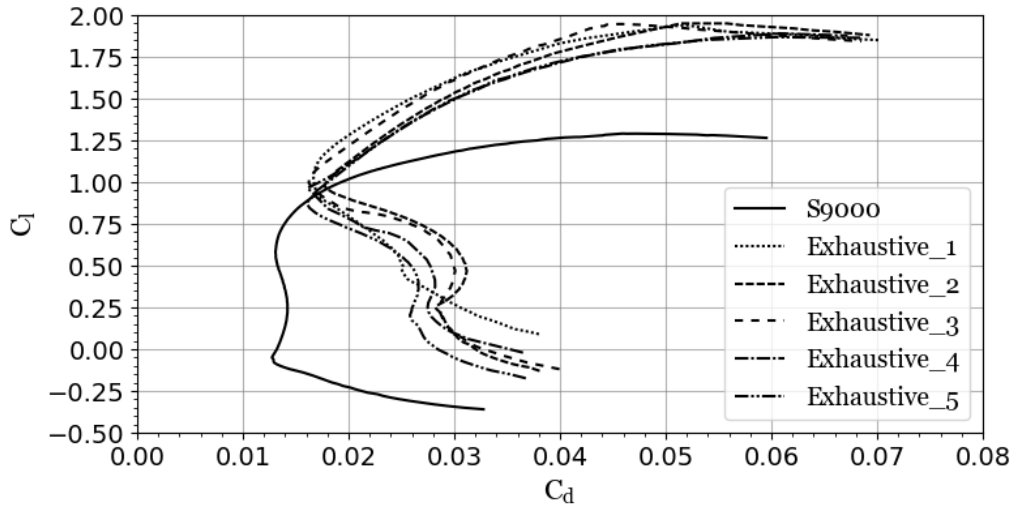
Figures 4.5 to 4.9 display the polars of the optimized airfoil compared to the initial airfoil across all the flight conditions encountered during the competition. These polars clearly demonstrate the impact of increasing airfoil thickness and camber with a rise of the maximum lift coefficient in all graphs. The images also show the consequences, with images 4.8 and 4.9 depicting clearly the increment of drag across the free flight conditions. Furthermore, the polar in Figure 4.9 highlights the impact of reaching the lift coefficient upper limit for the sustained turn. This appears to result in a localized improvement at the expense of neighboring points.

Another singular improvement is verified in the climb operating point, Figure 4.7. This localized enhancement is explained by the weight assigned to this operational point, which is responsible for integrating the acceleration time to climb speed and allowing the aircraft to achieve the target altitude. Although the aircraft effortlessly achieves the desired altitude, any remaining time after the climb can be utilized to accelerate the aircraft towards maximum speed. Thus giving the climbing operating point relevance, as it can greatly affect two scores. This, combined with an increase in the lift coefficient that provides the best rate of climb, makes this point more relevant to the acceleration time estimation with each iteration. This indicates that the acceleration method used in an effort to achieve a more realistic representation of the competition flight can be adverse to optimization if the solution is too far away from the original airfoil.

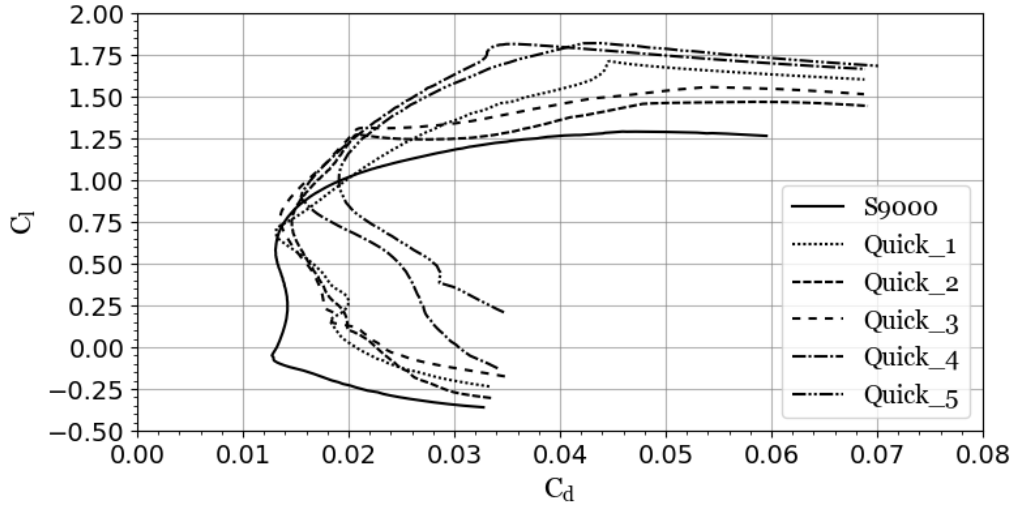
In conclusion, the effectiveness of the implemented approach is strongly dependent on the original airfoil. The aerodynamic properties of the original airfoil must be considerably similar to those of the optimized design. While this may appear to be a major limitation, any technique that sets the Reynolds and/or Mach number cannot allow the final conditions to deviate significantly from the original. Otherwise, the resulting airfoil would be developed for circumstances that will never occur in



(a) Standard

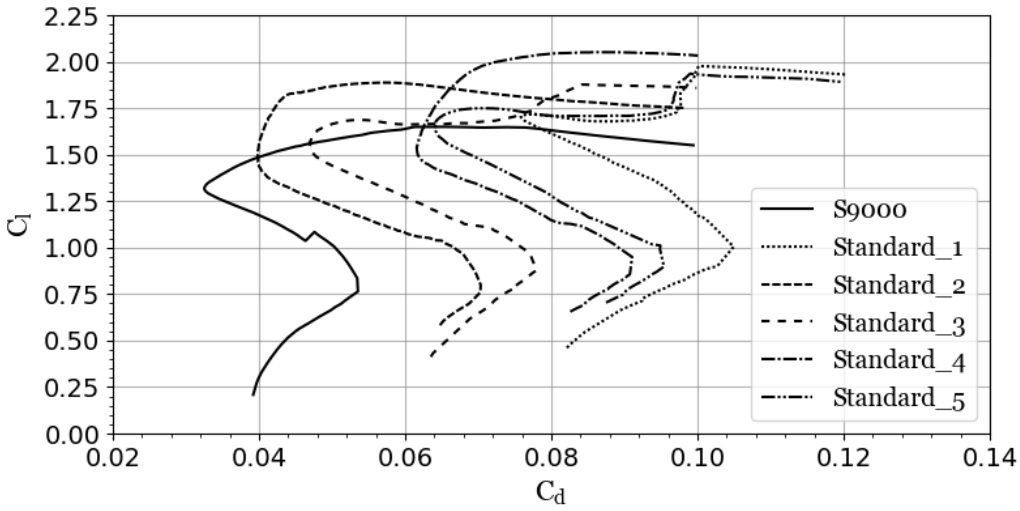


(b) Exhaustive

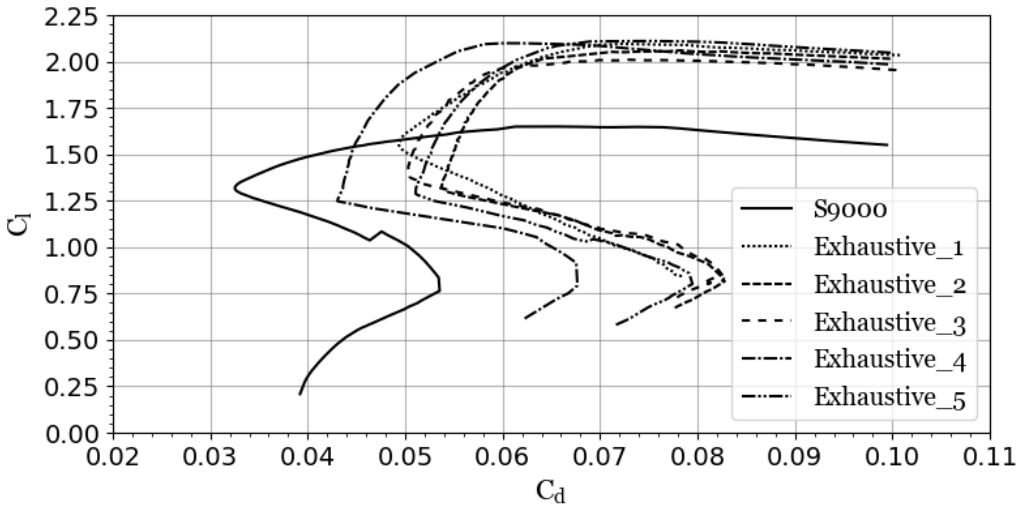


(c) Quick

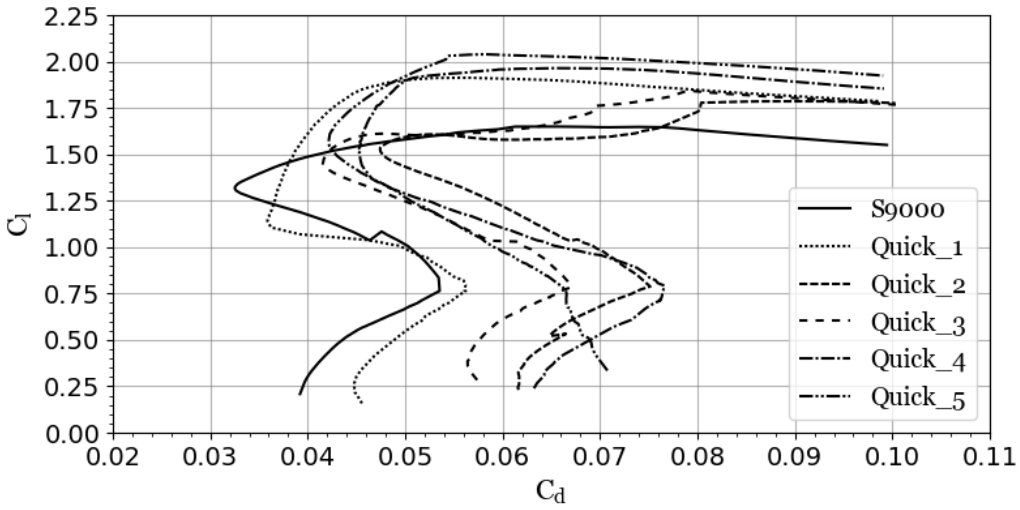
Figure 4.4: Comparison of drag polars in the ground run configuration ($Re = 1.07e5$ and $M = 0.02$) between the original and optimized airfoils for all study cases.



(a) Standard

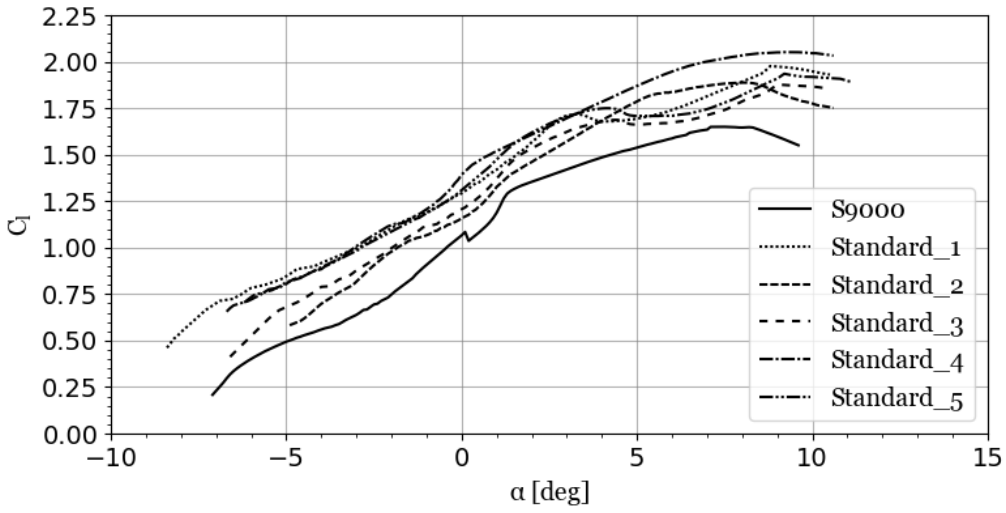


(b) Exhaustive

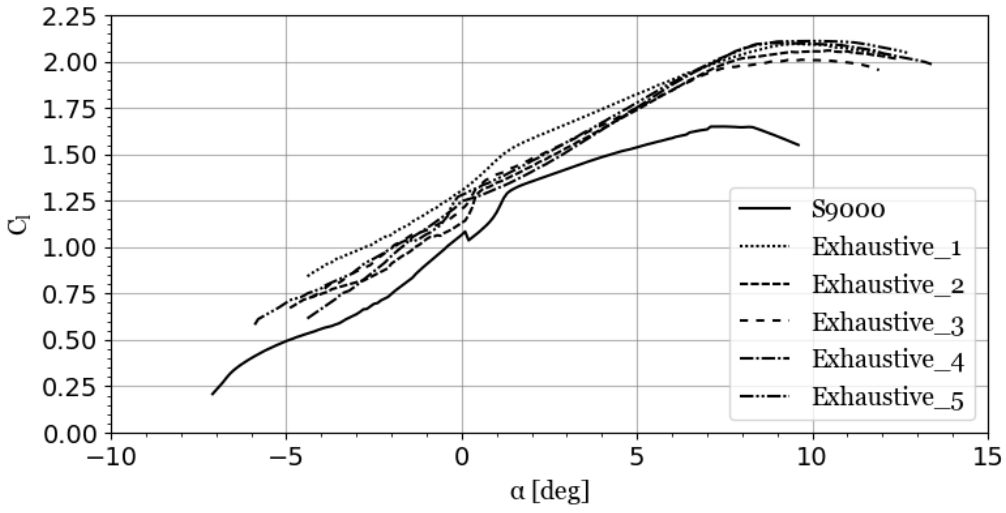


(c) Quick

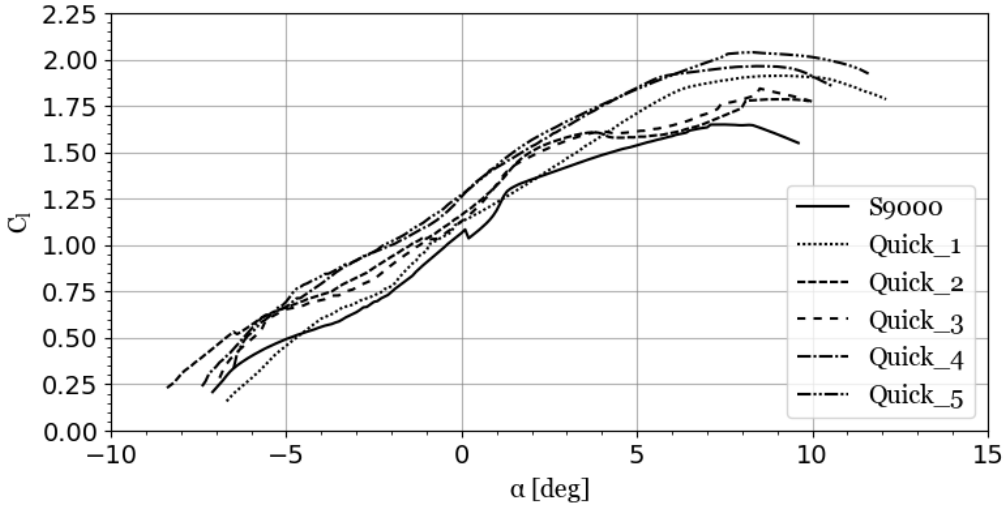
Figure 4.5: Comparison of drag polars in the take-off configuration ($Re = 1.51e5$ and $M = 0.03$) between original and optimized airfoils for all study cases.



(a) Standard

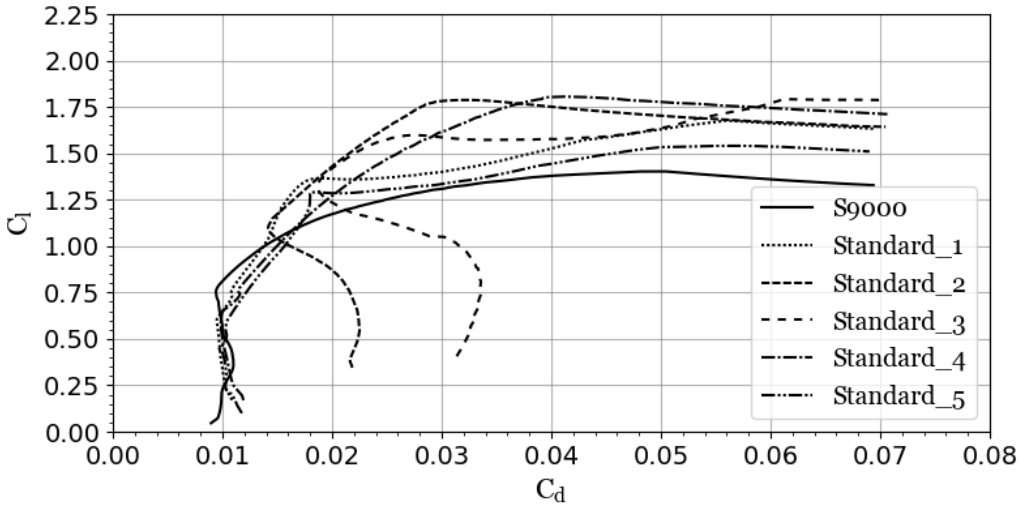


(b) Exhaustive

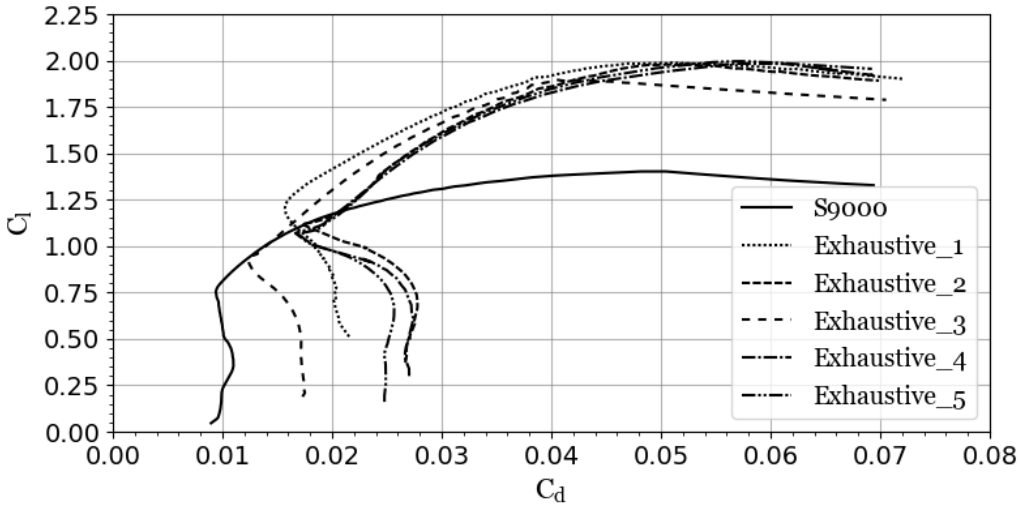


(c) Quick

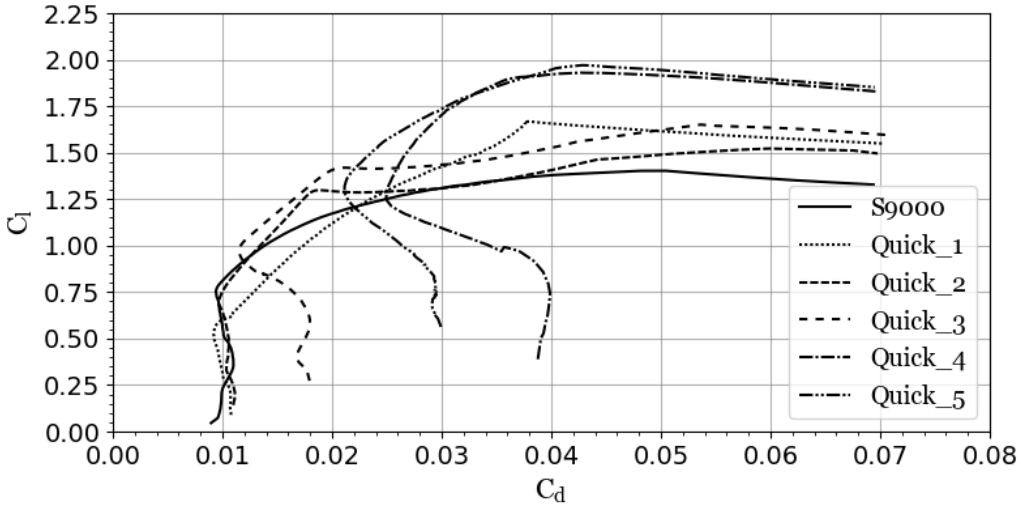
Figure 4.6: Comparison of lift coefficient vs α in the take-off configuration ($Re = 1.51e5$ and $M = 0.03$) between original and optimized airfoils for all study cases.



(a) Standard

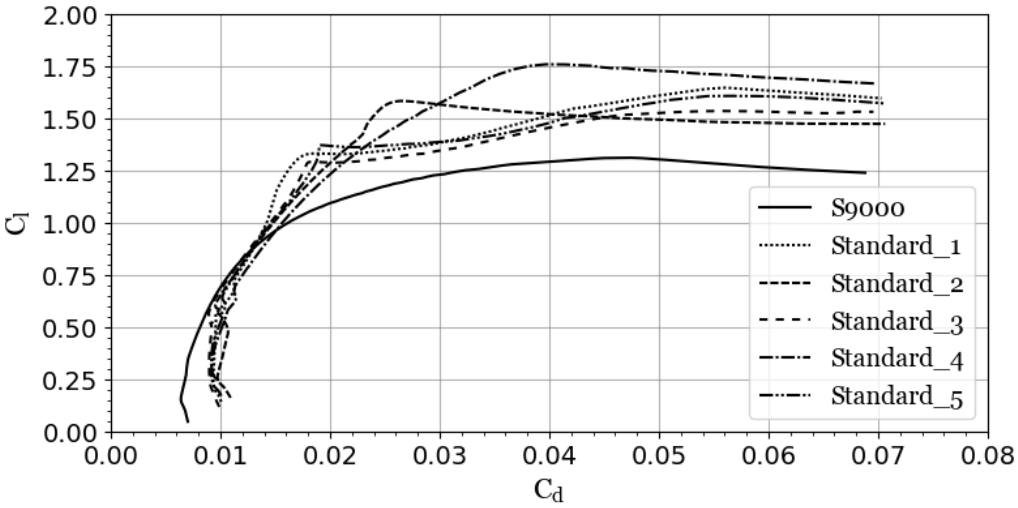


(b) Exhaustive

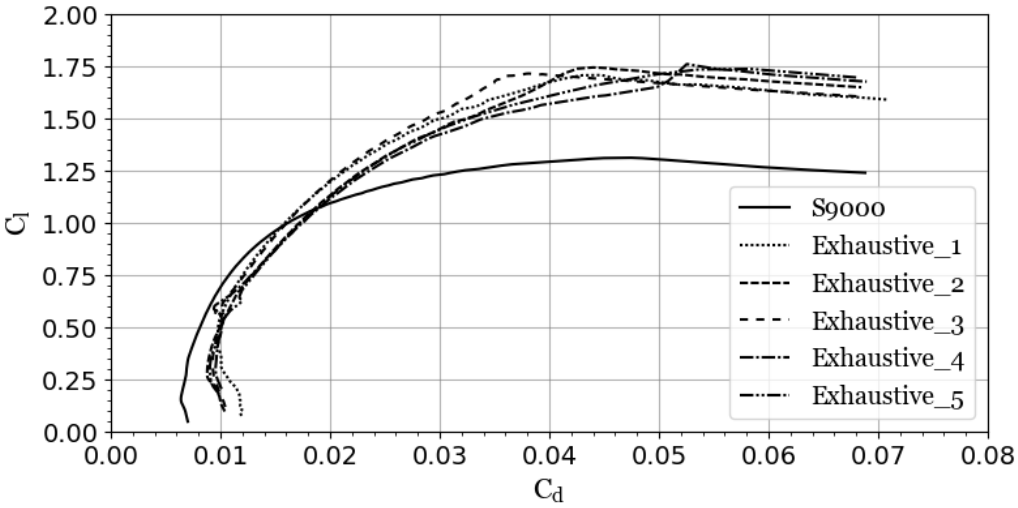


(c) Quick

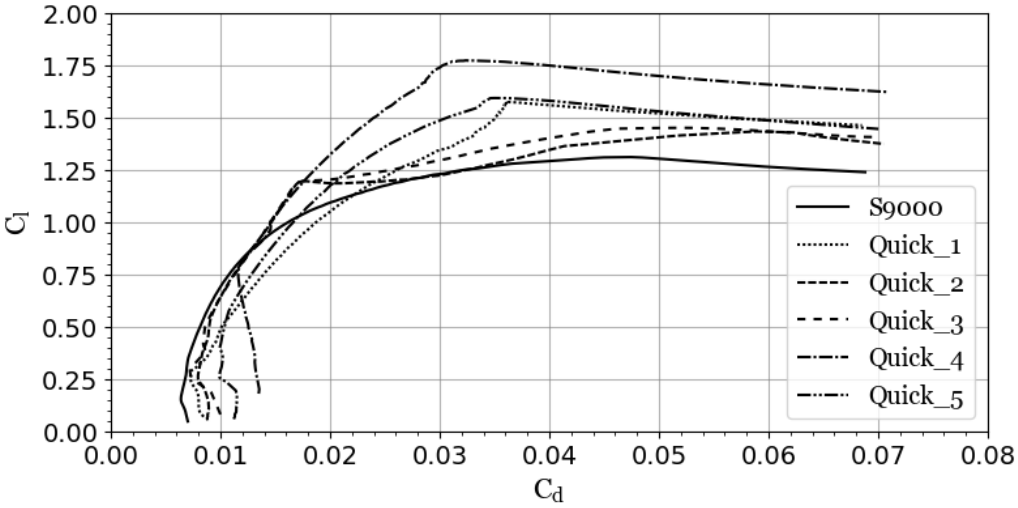
Figure 4.7: Comparison of drag polars in the climb configuration ($Re\sqrt{C_l} = 1.66e5$ and $M = 0.04$) between original and optimized airfoils for all study cases.



(a) Standard

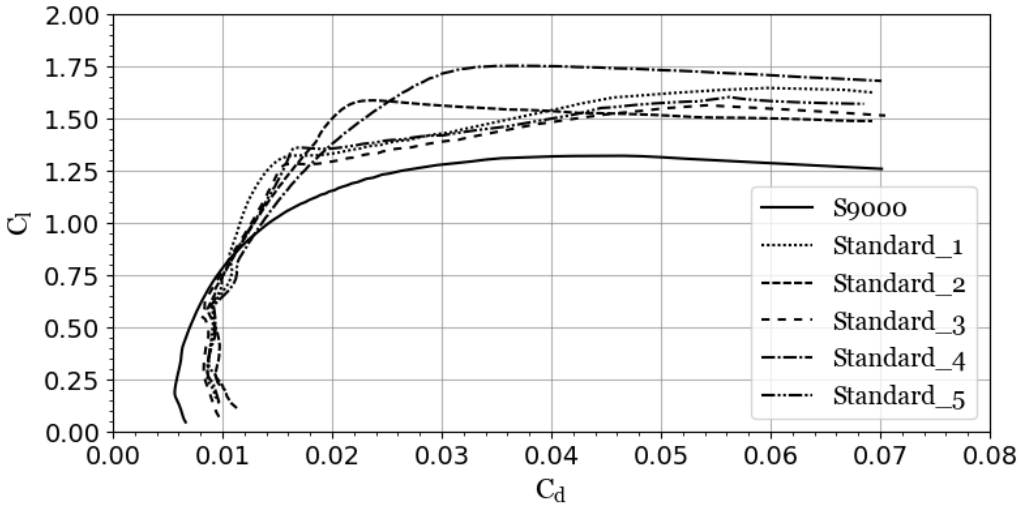


(b) Exhaustive

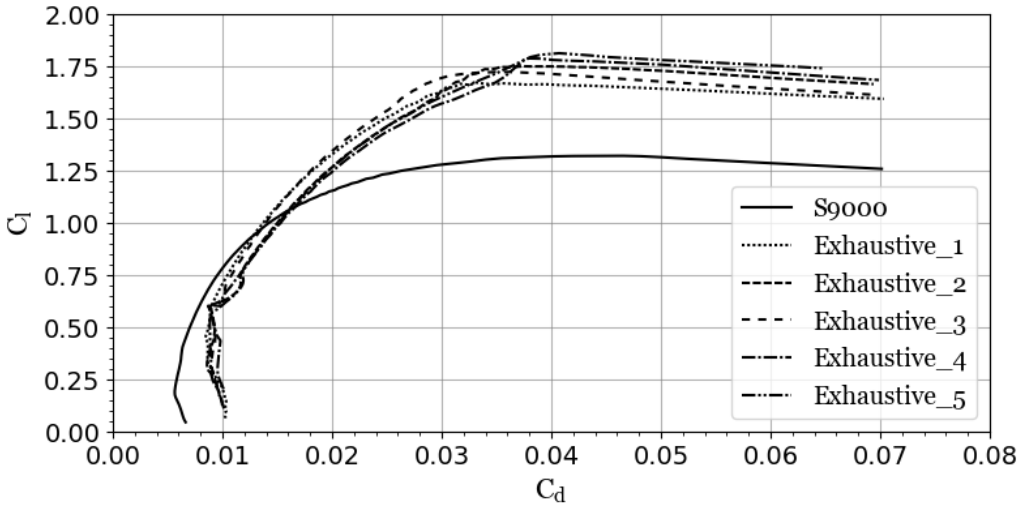


(c) Quick

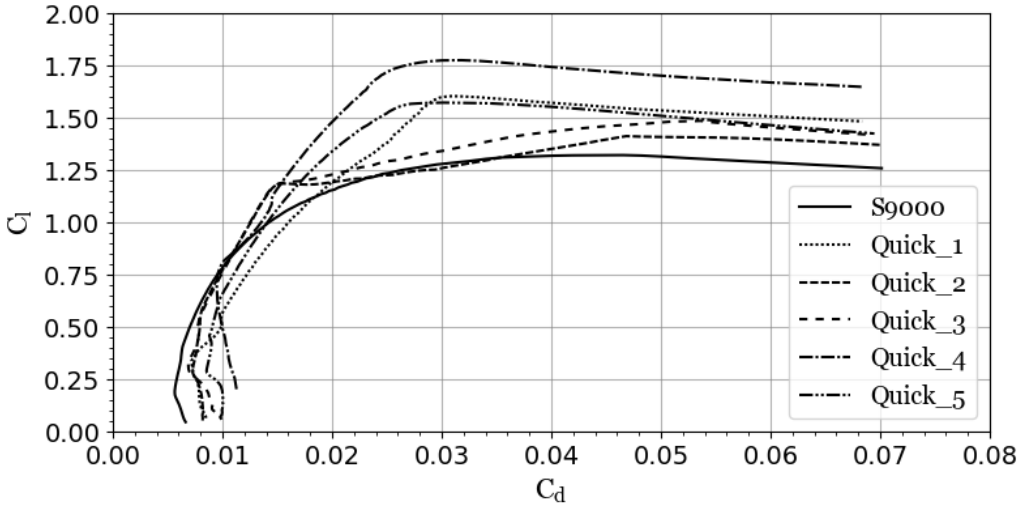
Figure 4.8: Comparison of drag polars in the cruise configuration ($Re\sqrt{C_l} = 1.65e5$ and $M = 0.08$) between original and optimized airfoils for all study cases.



(a) Standard



(b) Exhaustive



(c) Quick

Figure 4.9: Comparison of drag polars in the sustained turn configuration ($Re\sqrt{C_l} = 2.34e5$ and $M = 0.08$) between original and optimized airfoils for all study cases.

4.3 Case study 2 - AirCargo Challenge 2022

4.3.1 Objective and problem definition

The prior case study had the purpose to evaluate the effect of the PSO-IW parameters on the search for the optimal solution. Additionally, the work also examined the implemented optimization problem and the variation of the results across multiple runs. Based on the findings, the most effective and consistent PSO-IW configuration was selected, and the limitations of the implemented objective function and the algorithm's behavior, under conditions similar to those experienced in competition, were identified.

In this section, it is intended to expand the previous study and investigate the impact of specific initial conditions on the final results. First, the acceleration estimation will be disabled from the performance calculation, and the outcome airfoil characteristics compared to the optimized solutions from Case 1. This action is an attempt to understand how the addition of the acceleration estimation affects the final solution. Secondly, because one of the conclusions from Case 1 was the significant role of the initial airfoil in the final solution, it is pertinent to rerun the algorithm using another airfoil shape that has more similar features to the optimized shapes from the previous section. An airfoil that aligns with those optimized traits corresponds to the airfoil used in the prior edition of ACC. Consequently, this airfoil will serve as the starting point for a new attempt at optimizing the aircraft score.

Furthermore, it is necessary to recognize that Case Study 1 benefits from privileged knowledge, as the aircraft's performance in the competition is known. This means that, from the very beginning of this research, it was known that the aircraft had problems on the runway, which prevented it from quickly accelerate during the take-off run. This setback was extrapolated as a rise in the friction coefficient between the aircraft and the ground.

Therefore, it appears interesting to attempt an optimization without this knowledge, replicating an optimization performed in the preliminary phase of the ACC aircraft design process. With this goal in mind, it will be used the friction coefficient of 0.07, specified in [58] as the normal friction coefficient for a grass runway.

Finally, one of the ACC team's early decisions was to design the aircraft to use a 60 m runway rather than a 40 m, which would result in a 10% bonus for lifting off on a shorter runway. The choice was based on preliminary calculations, which revealed that the points lost in the payload score from using a shorter runway were not offset by the bonus. However, for the intended purpose of this study, it is interesting to assess this statement and explore the take-off on a 40 m runway. This study will investigate both the observed friction coefficient

suffered during competition and the projected coefficient in a grass runway.

4.3.2 Optimization set up

In order to preserve the consistency, the optimizations done in Case 2 will select the operational point's conditions by applying the same framework as in Case 1. Nonetheless, the operating points must be adjusted to account for the initial condition variations in each scenario.

The payload optimization by the take-off analysis is set up using the first two points. The maximum lift coefficient search is represented by the first point, which has a range of ±2 degrees around the angle of attack that produces the highest lift. The second point, corresponding to the ground run, will remain at a 7 degree angle of attack. Even though this setup might not be ideal, as a higher number could be needed to reach the maximum lift coefficient for lift-off. The value is maintained constant to ensure coherence across different scenarios. This consistency is crucial to any research work.

In the climb phase, the third to seventh points will set the best rate of climb condition and the acceleration to the climb speed. One multiple of 0.5 closer, below, and three points above the reference will be used, maintaining a C_l variation of 1.5 between the points. Finally, the eighth to fifteenth operating points are used for the free flight optimization. The first four are dedicated to the sustained turn and the subsequent four to the level flight. In the sustained turn, the interval of 0.5 C_l between points is maintained from Case 1. Therefore, if the reference point is very close to a multiple of 0.5, that point will be used, as well as one point below and two above. The logic from the previous Section 4.2.3, where the C_l value decreases more easily than it increases, is then followed. If the reference value roughly lies between two multiples of 0.5, two values are used below and above the reference point.

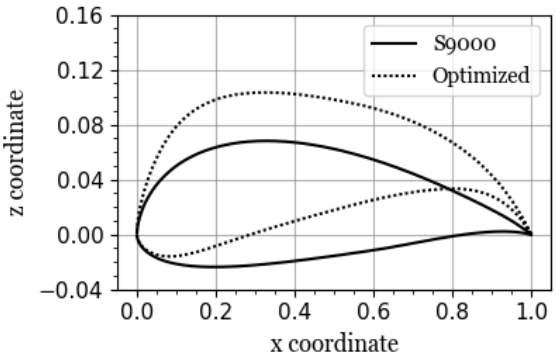
For level flight, only three points are used to determine the maximum speed. If the reference point is close to a multiple of 0.5, the closest point, as well as those above and below, will be chosen. If the reference point is between two multiples of 0.5, two points above and one below will be selected according to the identical process described previously. The operating points used in each scenario are presented in Annex B.

Regarding the airfoil parametrization and design variables, Case Study 2 will completely mimic Case 1's settings as detailed in Section 4.2.2. The B-spline method, described in Section 3.2.3, is applied using 8 control points to define each surface of the airfoil shape. This results in 12 design variables, to which 4 more are added to define the position of the flap hinge and the flap deflection in three distinct flight phases. The design space is limited by

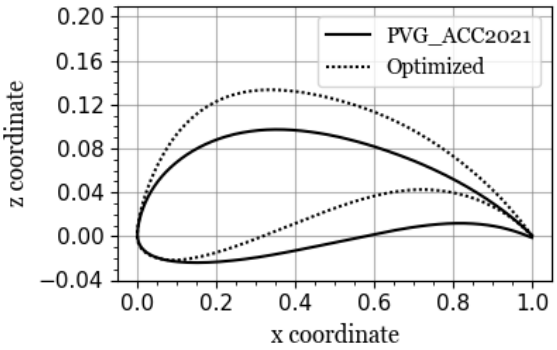
the exact same constraints as in Case 1 and detailed in Section 4.2.2.

4.3.3 Results and discussion

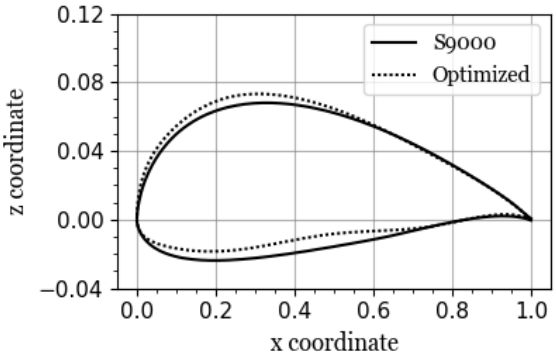
The results for each optimization are provided in Tables 4.17 through 4.20 and Figures 4.11 to 4.15. For comparison purposes, the solution from the latest exhaustive optimization run has been included in the tables. This specific run was selected because the optimized airfoil and score performance almost matches the average values from the five optimization runs.



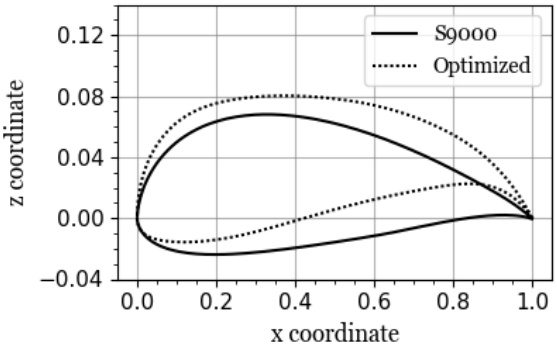
(a) No acceleration case



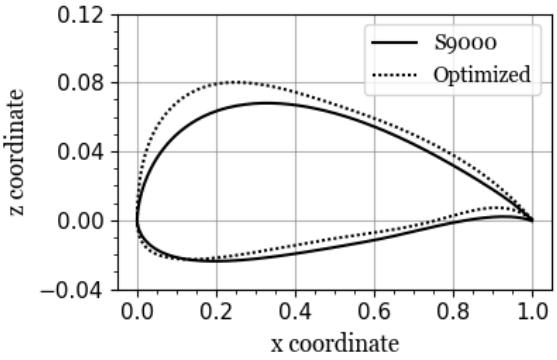
(b) Another initial airfoil case



(c) $\mu = 0.07$ case



(d) $L_{runway} = 40m$ case



(e) $\mu = 0.07$ & $L_{runway} = 40m$ case

Figure 4.10: Optimized and original airfoil comparison for all case studies.

All scenarios under optimization have shown the same tendency as the previous case

study. In general terms, the scenarios have increased their maximum thickness and camber to raise the lift coefficient and allow the aircraft to carry heavier loads, as shown by Table 4.18 and Figures 4.10. The trade-offs for this increase are the same as before, both the rate of climb and the distance traveled, as shown in Tables 4.19 and 4.20. Furthermore, almost every case has converged close to the C_l limit for the sustained turn condition.

Analyzing specifically the case in which the computation of the acceleration is disabled, besides the resultant optimized airfoil being close to the optimized airfoil from the fifth exhaustive run, significant differences emerge. The first difference between the two methods lies in the C_l condition that achieves the best rate of climb. In the first case, this value had influence on two scores. Logically, it affects the altitude score but also impacts the distance traveled. This occurs because the acceleration time and distance are reduced if the aircraft reaches the target altitude of 100 m before the 60 second mark. As a result, the optimization tends to favor a localized improvement, as seen in the previous case. In contrast, when the acceleration is disabled, the rate of climb only affects the altitude score. Considering that the aircraft easily reaches the 100 m altitude, this operating point carries less weight in the optimization. Thus it is less susceptible to a localized improvement, as seen in Figure 4.11(d). Interestingly, as seen in Table 4.19, the rate of climb in the first case is lower than in the current scenario, which can be attributed to the optimizer balancing improvements in both rate of climb and acceleration to climb speed in the former case.

Another noteworthy difference comes from the acceleration from climb to maximum speed. The inclusion of the acceleration calculation adds an extra operating point between the sustained turn and level flight condition. The weight given to this point by the aircraft acceleration and to the maximum speed in the distance traveled prevents the airfoil increase after the localized improvement of the sustained turn flight condition. Without this, the current scenario, as illustrated in Figures 4.11(f) and 4.11(e), appears to suffer more from a local improvement.

Next, it is analyzed in detail, the optimization using the PVG_ACC2021 airfoil as the initial design. This airfoil, selected due to some similarities with the average optimized airfoil features from the previous case, has continued to follow the trends outlined in the second paragraph, taking the airfoil even further in terms of maximum thickness, camber, and lift coefficient, as illustrated by Table 4.18 and Figure 4.10(b). Another notable difference is that despite converging close to the C_l limit for the sustained turn condition, it does not exhibit a localized improvement. This can be attributed to higher values within the acceptable range of C_l for the sustained turn condition. However, this statement cannot be said with com-

plete certainty, as some of the exhaustive simulations in the previous case also did not show localized enhancements at this specific point.

Table 4.17: Optimized flap design variables for Case 2.

Case study	$hinge_x$	$\delta_{take-off}$	δ_{climb}	δ_{cruise}
S9000	0.8000	25.00	4.000	0.000
PVG_ACC2021	0.8000	25.00	0.000	-6.000
Exhaustive 5	0.7720	23.58	3.550	-7.380
No acceleration	0.8100	24.23	-5.790	-8.920
Another initial airfoil	0.792	18.15	-0.740	-9.710
$\mu = 0.07$	0.7540	24.99	7.440	1.180
$L_{runway} = 40m$	0.7810	24.84	-7.750	-8.240
$L_{runway} = 40m$ & $\mu = 0.07$	0.7140	24.88	5.560	-0.7800

The only case in which, besides the payload improvement, could increase the distance traveled was the scenario where the friction coefficient assumes the typical value for a grass runway. However, the apparent enhancement was not achieved from an increase in the level or turn flight speeds. Instead, the reduction in the turn radius has enabled the aircraft to turn with a smaller radius, offering more space for the straight-line flight that has a higher speed. This suggests that, despite the slight decrease in both level and turn flight speeds, as more weight is given to level flight, the total distance traveled has increased marginally. This rebalance of importance can also be linked to the absence of a localized enhancement in the C_l value for the turn condition, besides sitting near the limit.

The most interesting insight from this case is that the optimized airfoil stays closer to the S9000 compared with the results from other optimizations, as illustrated in Figure 4.10(c). This indicates that in the preliminary design stage, where the behavior in the ground run was not known, the S9000 was not very far from the considered optimal solution by the optimization algorithm.

Another exception to the general trend arises from the scenario with a take-off run of 40 m. This optimization run was the only case where the maximum thickness was kept like the S9000, as demonstrated in Figure 4.10(d). However, the rise in the maximum camber kept the optimization to follow the pattern in the score evolution as in the other cases. Additionally, this scenario was the only one capable of compensating the time required to accelerate to maximum speed, as demonstrated in Table 4.20. This can be attributed to the lowest payload carried by any case, which allowed for a higher rate of climb. This high rate of climb minimizes the time required to reach an altitude of 100 m, thus leaving enough time for the acceleration.

Table 4.18: Initial and optimized take-off conditions for Case 2.

Case studies		$S_{payload}$ –	W_{total} [N]	$W_{payload}$ [N]	α_{to} [deg]	$C_{l_{max}}$ –	V_S [m/s]	V_{LOF} [m/s]	$C_{l_{run}}$ –	$C_{d_{run}}$ –	V_{run} [m/s]
Exhaustive 5	Init	561.5	41.59	17.67	7.000	1.713	10.71	11.78	1.033	0.02022	8.330
	Opt	705.2	46.11	22.19	9.000	2.181	9.994	10.99	1.558	0.03191	7.773
No	Init	561.5	41.59	17.67	7.000	1.713	10.71	11.78	1.033	0.02022	8.330
Acceleration	Opt	709.5	46.25	22.33	9.000	2.088	10.23	11.25	1.606	0.03115	7.956
Another	Init	633.7	43.86	19.94	11.00	1.802	10.72	11.79	1.310	0.02175	8.340
initial airfoil	Opt	743.9	47.33	23.41	11.00	2.250	9.967	10.96	1.708	0.02994	7.753
$\mu = 0.07$	Init	960.4	54.14	30.22	8.000	1.690	12.30	11.78	1.035	0.01910	9.568
	Opt	1045	56.82	32.90	8.000	1.890	11.92	11.78	1.050	0.02070	9.269
$L_{runway} = 40m$	Init	417.0	37.04	13.12	7.000	1.695	10.16	11.18	1.032	0.02075	7.902
	Opt	503.0	39.75	15.83	9.000	1.970	9.764	10.74	1.374	0.02956	7.595
$L_{runway} = 40m$	Init	709.5	46.25	22.33	7.000	1.695	10.15	12.49	1.034	0.01975	8.832
& $\mu = 0.07$	Opt	825.8	49.91	25.99	7.000	1.969	9.764	11.90	1.126	0.02431	8.414

Furthermore, the two cases that used the 40 m runway resulted in a localized improvement in the level flight condition. As observed in the scenario that uses the 60 m runway with a lower friction coefficient, this enhancement can be linked to a smaller turn radius, which emphasizes the importance of the level flight speed in the distance traveled.

Table 4.19: Initial and optimized climb conditions for Case 2.

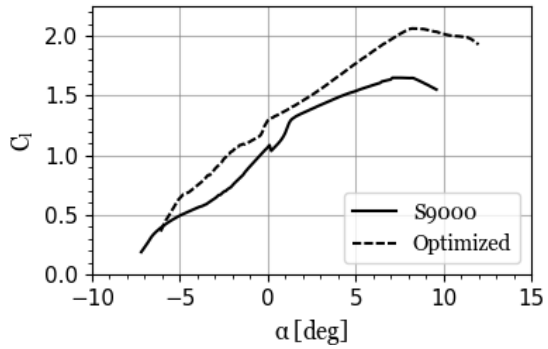
Case studies		S_{climb} –	alt [m]	RC_{max} [m/s]	V_{climb} [m/s]	Cl_{climb} –	Cd_{climb} –	t_{accel} [s]
Exhaustive 5	Init	1000	100.0	2.821	15.39	0.750	0.00935	1.880
	Opt	1000	100.0	2.297	13.70	1.050	0.01754	1.6151
No	Init	1000	100.0	2.821	15.39	0.750	0.00935	0.000
Acceleration	Opt	1000	100.0	2.434	16.23	0.750	0.01055	0.000
Another	Init	1000	100.0	2.617	15.81	0.750	0.01046	2.257
initial airfoil	Opt	1000	100.0	2.303	14.99	0.900	0.01428	2.498
$\mu = 0.07$	Init	1000	100.0	1.986	17.01	0.800	0.00940	2.835
	Opt	1000	100.0	1.856	15.99	0.950	0.01025	2.452
$L_{runway} = 40m$	Init	1000	100.0	3.223	15.04	0.700	0.00990	1.696
	Opt	1000	100.0	2.914	15.58	0.700	0.01219	2.404
$L_{runway} = 40m$	Init	1000	100.0	2.474	16.23	0.750	0.00906	2.322
& $\mu = 0.07$	Opt	1000	100.0	2.214	16.86	0.750	0.00966	3.509

From Figures 4.10(c) and 4.10(e), it is evident that the two cases with the friction coefficient of $\mu = 0.07$ reached a similar optimized airfoil. The major difference between the two airfoils is a more increased leading edge radius in the case that uses the 40 m runway. This distinction reflects the optimization's focus on increasing the lift coefficient for the 40 m runway case, which are presented by values of Cl_{max} for the 40 and 60 m runway cases in Table 4.18.

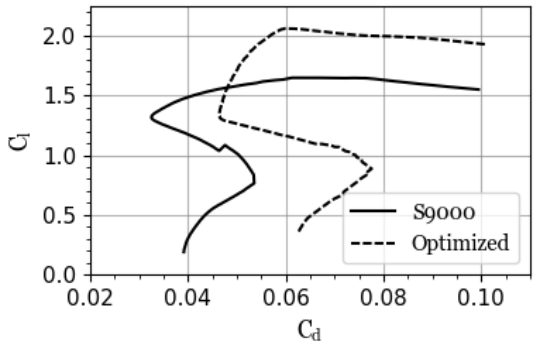
Finally, it is important to note that the conclusion made by the ACC2022 team about the better score with the 60 m runway take-off in the early stages of the design was probably wrong. The comparison of similar cases that use the different lengths of runway, made in Table 4.18 to 4.20, clearly indicates that the payload score penalty for take-off in 40 m is more than compensated by the bonus points. The total score difference is larger for the extrapolated that for the normal grass friction coefficient. This happens because the other two scores are practically unaltered by changes in payload. Therefore, to achieve the same amount of total points given by the bonus score, the increase in the payload score must rise 3 times that value.

Table 4.20: Initial and optimized level flight and sustained turn conditions for Case 1.

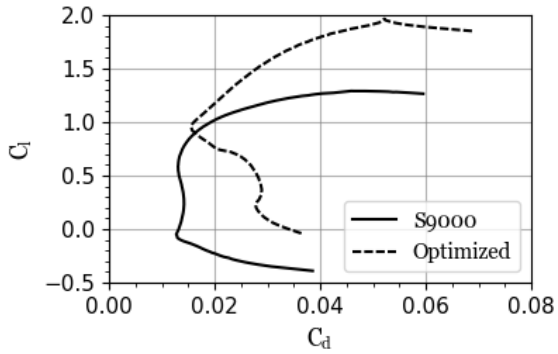
Case studies		$S_{distance}$ –	R_{turn} [m]	V_{turn} [m/s]	$C_{l_{turn}}$ –	$C_{d_{turn}}$ –	V_{cruise} [m/s]	$C_{l_{cruise}}$ –	$C_{d_{cruise}}$ –	t_{accel} [s]	$dist_{accel}$ [m]	$dist$ [m]	S_{flight} –	$Bonus$	S_{total}
Exhaustive 5	Init	1000	41.33	26.50	0.5087	0.00706	27.70	0.2327	0.00661	25.35	72.86	3255	853.8	0.000	853.8
	Opt	973.0	38.85	25.69	0.6000	0.00873	27.01	0.2713	0.00923	26.68	312.0	3167	892.7	0.000	892.7
No	Init	1000	41.32	26.49	0.5088	0.00707	27.70	0.2327	0.00662	25.35	21.91	3254	853.8	0.000	853.8
Acceleration	Opt	972.4	39.02	25.74	0.5992	0.00845	26.92	0.2739	0.00961	24.84	153.5	3165	894.0	0.000	894.0
Another	Init	981.7	39.97	26.06	0.5548	0.00838	27.16	0.2553	0.00883	21.83	59.88	3196	871.8	0.000	871.8
initial airfoil	Opt	949.5	36.82	25.01	0.6499	0.01109	26.41	0.29.14	0.01193	28.026	362.8	3091	897.8	0.000	897.8
$\mu = 0.07$	Init	960.0	35.45	24.54	0.7722	0.00953	27.51	0.3072	0.00651	29.04	566.4	3125	973.5	0.000	973.5
	Opt	964.4	33.81	23.96	0.8498	0.01018	27.44	0.3239	0.00649	34.87	838.7	3139	1003	0.000	1003
$L_{runway} = 40m$	Init	1008	42.63	26.91	0.4392	0.00953	27.77	0.2062	0.00667	18.57	0.000	3282	808.4	80.84	889.2
	Opt	995.6	41.05	26.40	0.4895	0.00800	27.56	0.2247	0.00731	20.81	0.000	3241	832.8	83.28	916.1
$L_{runway} = 40m$	Init	989.4	39.66	25.96	0.5895	0.00758	27.65	0.2598	0.00657	24.42	191.0	3220	899.6	89.96	989.6
& $\mu = 0.07$	Opt	964.8	36.14	24.77	0.6983	0.01083	27.39	0.2857	0.00732	28.85	455.5	3140	930.2	93.02	1023



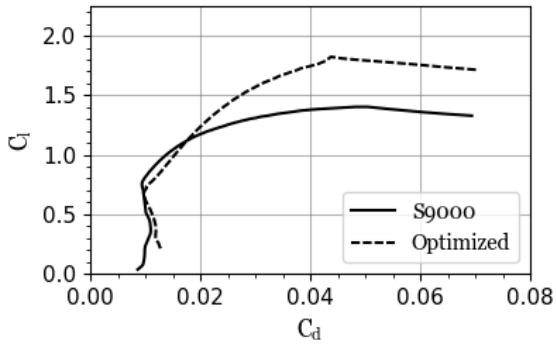
(a) Lift coefficient for take-off, ($Re = 151e5$ and $M = 0.03$).



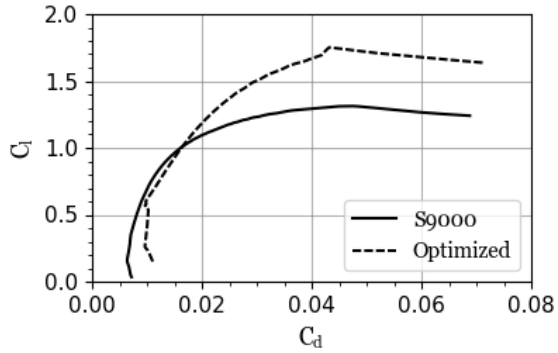
(b) Drag polar for take-off, ($Re = 151e5$ and $M = 0.03$).



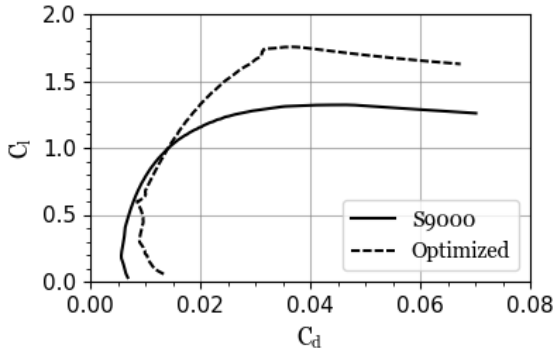
(c) Drag polar for ground run, ($Re = 107e5$ and $M = 0.02$).



(d) Drag polar for climb, ($Re\sqrt{C_l} = 166e5$ and $M = 0.04$).

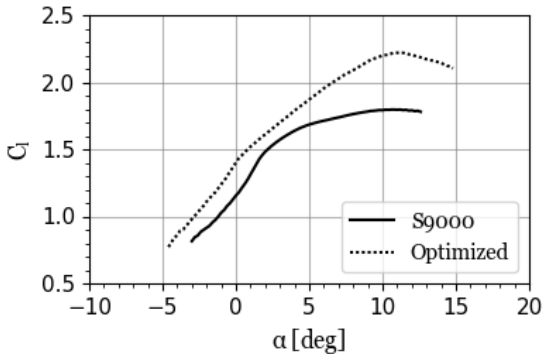


(e) Drag polar for cruise, ($Re\sqrt{C_l} = 165e5$ and $M = 0.08$).

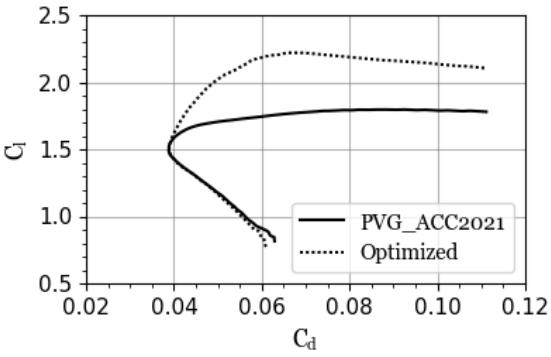


(f) Drag polar for turn, ($Re\sqrt{C_l} = 234e5$ and $M = 0.08$).

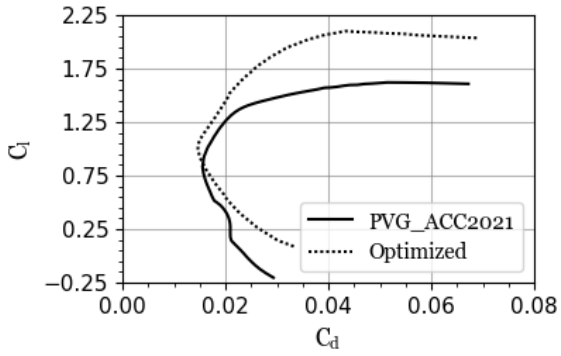
Figure 4.11: Optimized and original airfoil polars comparison for the case with no acceleration.



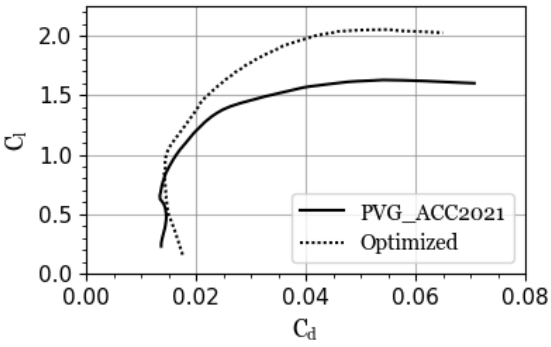
(a) Lift coefficient for take-off, ($Re = 152e5$ and $M = 0.04$).



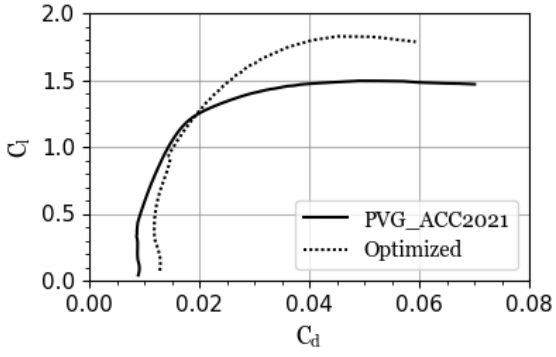
(b) Drag polar for take-off, ($Re = 152e5$ and $M = 0.04$).



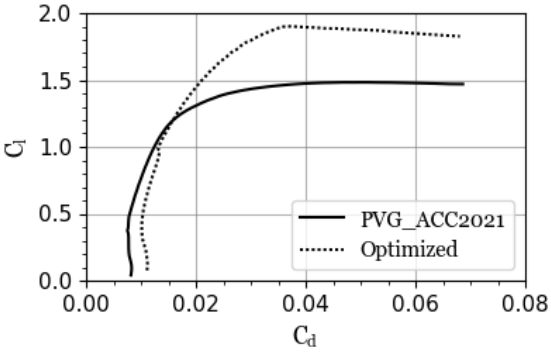
(c) Drag polar for ground run, ($Re = 108e5$ and $M = 0.02$).



(d) Drag polar for climb, ($Re\sqrt{C_l} = 170e5$ and $M = 0.05$).

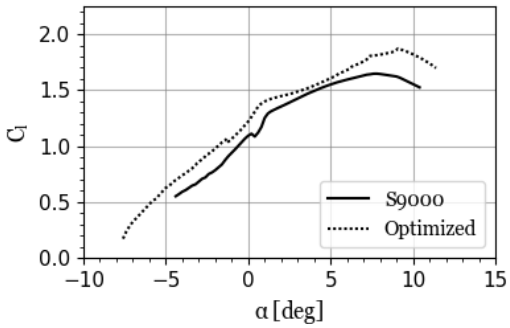


(e) Drag polar for cruise, ($Re\sqrt{C_l} = 170e5$ and $M = 0.08$).

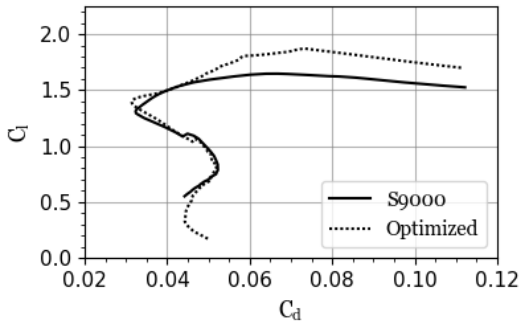


(f) Drag polar for turn, ($Re\sqrt{C_l} = 240e5$ and $M = 0.08$).

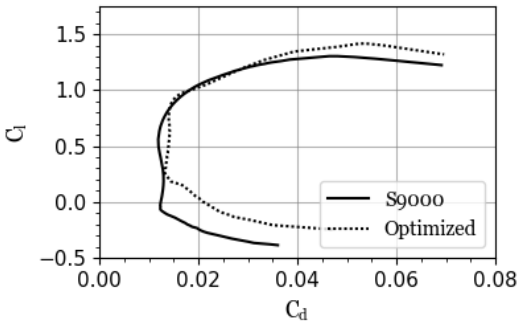
Figure 4.12: Optimized and original airfoil polars comparison for the case with the other initial airfoil.



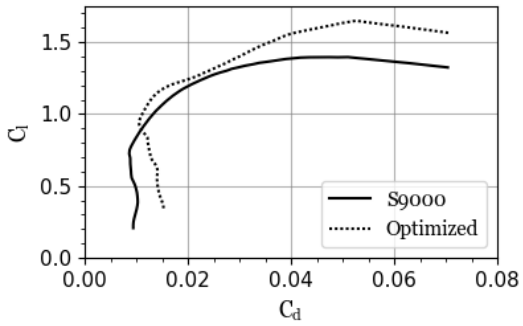
(a) Lift coefficient for take-off, ($Re = 173e5$ and $M = 0.04$).



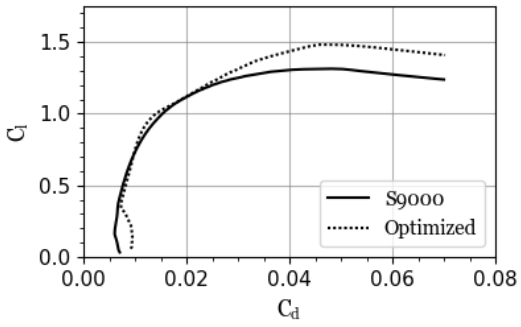
(b) Drag polar for take-off, ($Re = 173e5$ and $M = 0.04$).



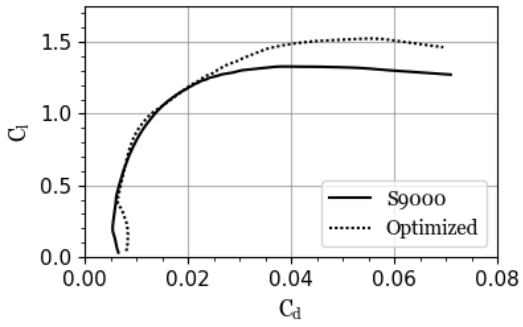
(c) Drag polar for ground run, ($Re = 122e5$ and $M = 0.03$).



(d) Drag polar for climb, ($Re\sqrt{C_l} = 189e5$ and $M = 0.06$).

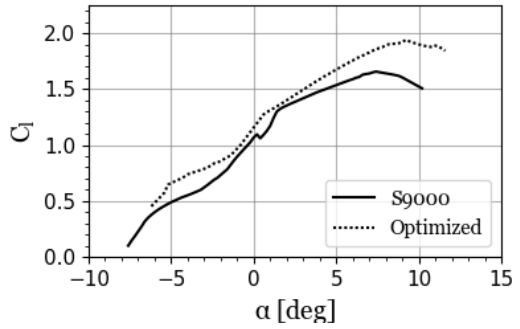


(e) Drag polar for cruise, ($Re\sqrt{C_l} = 189e5$ and $M = 0.08$).

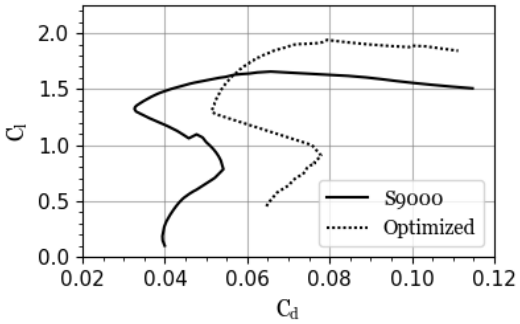


(f) Drag polar for turn, ($Re\sqrt{C_l} = 268e5$ and $M = 0.07$).

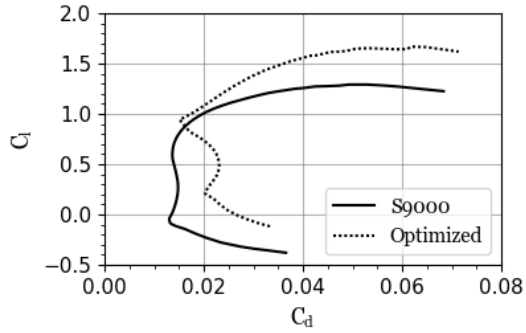
Figure 4.13: Optimized and original airfoil polars comparison for the case with $\mu = 0.07$.



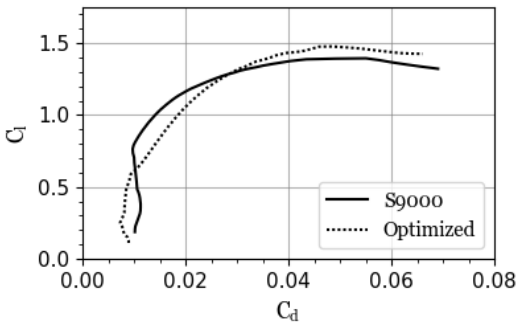
(a) Lift coefficient for take-off, ($Re = 143e5$ and $M = 0.03$).



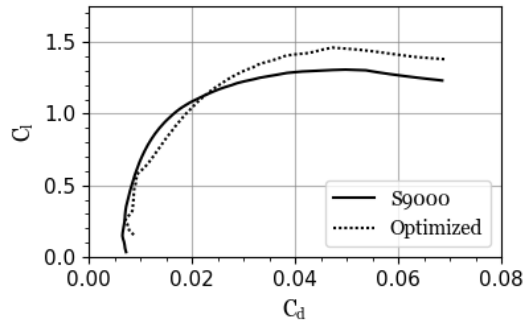
(b) Drag polar for take-off, ($Re = 143e5$ and $M = 0.03$).



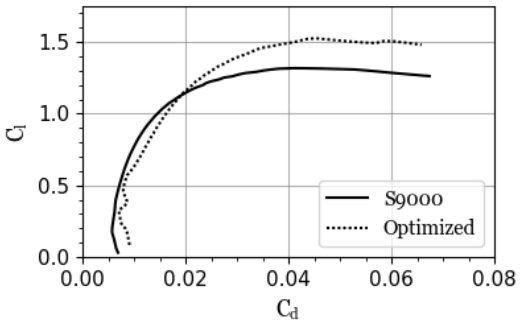
(c) Drag polar for ground run, ($Re = 101e5$ and $M = 0.02$).



(d) Drag polar for climb, ($Re\sqrt{C_l} = 157e5$ and $M = 0.05$).

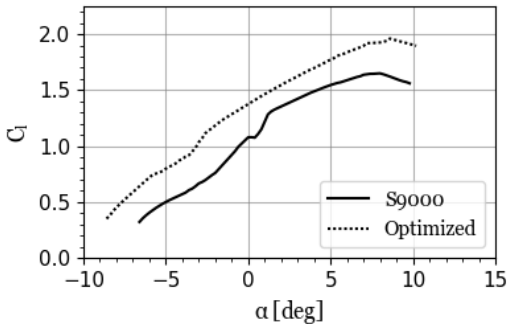


(e) Drag polar for cruise, ($Re\sqrt{C_l} = 156e5$ and $M = 0.08$).

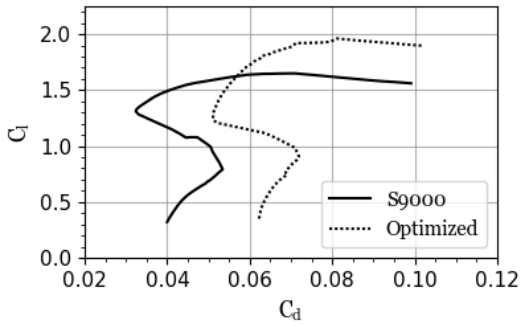


(f) Drag polar for turn, ($Re\sqrt{C_l} = 221e5$ and $M = 0.08$).

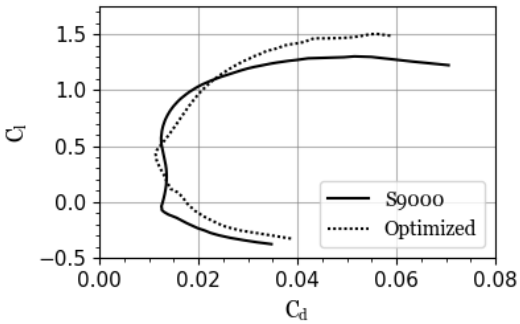
Figure 4.14: Optimized and original airfoil polars comparison for the case with $R_{runway} = 40m$.



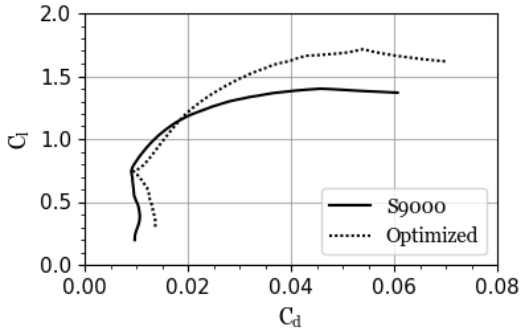
(a) Lift coefficient for take-off, ($Re = 161e5$ and $M = 0.04$).



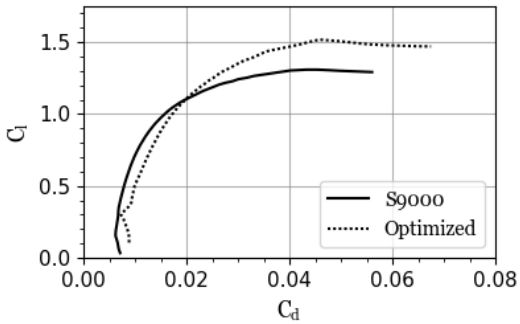
(b) Drag polar for take-off, ($Re = 161e5$ and $M = 0.04$).



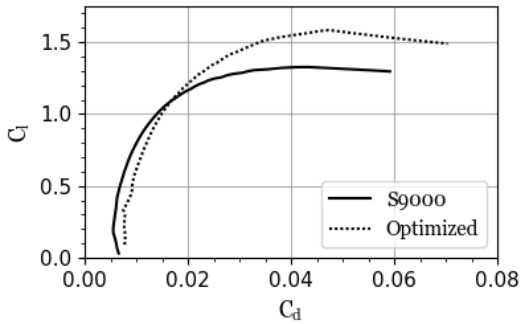
(c) Drag polar for ground run, ($Re = 114e5$ and $M = 0.03$).



(d) Drag polar for climb, ($Re\sqrt{C_l} = 176e5$ and $M = 0.06$).



(e) Drag polar for cruise, ($Re\sqrt{C_l} = 176e5$ and $M = 0.08$).



(f) Drag polar for turn, ($Re\sqrt{C_l} = 248e5$ and $M = 0.07$).

Figure 4.15: Optimized and original airfoil polars comparison for the case with $R_{runway} = 40m$ and $\mu = 0.07$.

Chapter 5

Conclusions

5.1 Summary and Conclusions

The main goal of this dissertation was to modify an existing tool, Xoptfoil, by implementing a new objective function. The new objective function does not directly evaluates the aerodynamic performance of the airfoil. Rather, it assesses the impact of the airfoil's aerodynamic properties in the aircraft's overall performance. This is accomplished through the incorporation, in the algorithm, of a method that can translate the aerodynamic properties of the airfoil into the aerodynamic characteristics of the aircraft. These translated characteristics are then used to simulate the aircraft's performance in the competition. The obtained score from the competition simulation was then used as the objective value to be optimized by the Xoptfoil algorithm.

In Case Study 1, two investigations were conducted simultaneously. First, the optimizer specific parameters were analyzed, and the results revealed that the exhaustive option offered the best and most accurate setting. As a consequence, this solution was selected for the subsequent case study. Besides that, the analysis of the data has revealed the variability among the outcomes of several optimizations. This analysis is vital for future work since it determines the margin of error for the optimization method. Secondly, the same results were utilized to investigate the behavior of the employed approach. Based on this research, the employed method has demonstrated a tendency to enhance the payload score. This resulted in an optimized airfoil with much higher maximum thickness and camber than the original shape. However, this increase was so large that the optimized profile diverged significantly from the initial design conditions. This shows that, while the developed approach has a high exploration capacity, its accuracy decreases when the operational conditions of the optimized profile deviate significantly from the original settings. This emphasizes the need of doing a preliminary analysis of the airfoil shape that will be optimized.

Case Study 2 was dedicated to analyze the impact of certain properties on the optimization results. The characteristics varied based on the possible scenarios that the aircraft might encounter during the Air Cargo Challenge 2022 competition. The results of this case have demonstrated that the optimal airfoil, under the conditions defined in the preliminary de-

sign phase, would have a slight increase in the maximum lift coefficient, although it remained relatively close to the initial shape. However, when incorporating the difficulties that the aircraft encounters because of grass tufts during the take-off run, an airfoil with substantially higher maximum thickness and camber was found to be more beneficial. Furthermore, Case 2 also revealed that the same aircraft, with a reduced payload for lift-off within 40 m, consistently outperformed the similar scenarios that were chosen the 60 m runway.

Finally, both studies have shown that employing an acceleration estimation to improve the realism of aircraft performance simulations and avoid localized improvements did not achieve the desired results. In reality, this assumption may have had a negative impact on the optimization process, potentially constraining the optimum path, especially during the climb phase.

5.2 Future Work

Future studies should focus on developing a strategy that can replace the acceleration method for smoothing the airfoil's characteristics and avoiding localized improvements. An alternative technique worth testing is the inclusion of the assessment of the relative improvements of aerodynamic characteristics at design and off-design operating points. However, this method raises the issue of assigning appropriate weights for each operation point while also balancing the competition score against relative aerodynamic improvements. This final challenge might be solved by employing a multi-objective optimization, which eliminates the need to assign specific weights and instead gives a range of possible optimal airfoils.

Bibliography

[1] AkaModell Munich, "Aircargo challenge 2022, participation handbook," 2021. [Online]. Available: https://akamodell-muenchen.de/wp-content/uploads/2022/07/regulations_acc_2022_munich_v01_15.pdf xiii, 2, 39, 51

[2] S. Mirjalili, J. Song Dong, A. Lewis, and A. S. Sadiq, *Particle Swarm Optimization: Theory, Literature Review, and Application in Airfoil Design*. Cham: Springer International Publishing, 2020, pp. 167–184. [Online]. Available: https://doi.org/10.1007/978-3-030-12127-3_10 xiii, 11, 13, 14

[3] Fortune Business Insights, "Drone-in-a-box market size, share, russia-ukraine war impact analysis and regional forecast, 2023-2030," 2023. [Online]. Available: https://www.fortunebusinessinsights.com/drone-in-a-box-market-108470 1

[4] ——, "Drone services market size, share, russia-ukraine war impact analysis, and regional forecast, 2024-2032," 2024. [Online]. Available: https://www.fortunebusinessinsights.com/drone-services-market-102682 1

[5] A. Arias-Montano, C. A. C. Coello, and E. Mezura-Montes, "Multiobjective evolutionary algorithms in aeronautical and aerospace engineering," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 5, pp. 662–694, 2012. 1, 9

[6] M. Drela, *Pros and Cons of Airfoil Optimization*. World Scientific, 1998, pp. 363–381. 2, 37

[7] D. P. Raymer, *Aircraft Design: A Conceptual Approach*, 6th ed. Reston, VA: AIAA Education Series, 2018. [Online]. Available: https://doi.org/10.2514/4.104909 5, 42

[8] M. Selig, "Low reynolds number airfoil design lecture notes," in *Lecture Series: Low Reynolds Number Aerodynamics on Aircraft including applications in Emerging UAV Technology*, 2003. [Online]. Available: https://m-selig.ae.illinois.edu/pubs/Selig-2003-VKI-LRN-Airfoil-Design-Lecture-Series.pdf 7

[9] K. Deb, "Multi-objective optimisation using evolutionary algorithms: An introduction," in *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing*. London: Springer London, 2011, pp. 3–34. [Online]. Available: https://doi.org/10.1007/978-0-85729-652-8_1 8

[10] Y. Rahmad, M. D. Robani, P. S. Palar, and L. R. Zuhal, "Single- and multi-objective optimization of a low-speed airfoil using genetic algorithm," in *7TH INTERNATIONAL*

SEMINAR ON AEROSPACE SCIENCE AND TECHNOLOGY (ISAST 2019). AIP Publishing, 2020. [Online]. Available: https://doi.org/10.1063/5.0002610 8

[11] Y. P. Ju and C. H. Zhang, "Multi-point robust design optimization of wind turbine airfoil under geometric uncertainty," *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, vol. 226, no. 2, pp. 245–261, 2012. [Online]. Available: https://doi.org/10.1177/0957650911426540 8

[12] K. Huang, Z. J. Meng, and J. Huang, "A multi-point approach to airfoil shape optimization," in *Manufacturing Engineering and Automation II*, ser. Advanced Materials Research, vol. 591. Trans Tech Publications Ltd, 2012, pp. 59–62. 8, 9

[13] N. Timnak and A. Jahangirian, "Multi-point optimization of transonic airfoils using an enhanced genetic algorithm," *PROCEEDINGS OF THE INSTITUTION OF MECHANICAL ENGINEERS PART G-JOURNAL OF AEROSPACE ENGINEERING*, vol. 232, no. 7, pp. 1347–1360, 2018. [Online]. Available: https://doi.org/10.1177/0954410017690549 8

[14] Y. Liang, X. Cheng, Z. Li, and J. Xiang, "Multi-objective robust airfoil optimization based on non-uniform rational b-spline (NURBS) representation," *SCIENCE CHINA-TECHNOLOGICAL SCIENCES*, vol. 53, no. 10, pp. 2708–2717, 2010. [Online]. Available: https://doi.org/10.1007/s11431-010-4075-4 8

[15] S. N. Skinner and H. Zare-Behtash, "State-of-the-art in aerodynamic shape optimisation methods," *Applied Soft Computing*, vol. 62, pp. 933–962, 2018. [Online]. Available: https://doi.org/10.1016/j.asoc.2017.09.030 8

[16] M. Drela, "Xfoil: An analysis and design system for low reynolds number airfoils," in *Low Reynolds Number Aerodynamics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1989, pp. 1–12. [Online]. Available: https://doi.org/10.1007/978-3-642-84010-4_1 8, 29

[17] K. Mazaheri, A. Nejati, K. C. Kiani, and R. Taheri, "The application of the gradient-based adjoint multi-point optimization of single and double shock control bumps for transonic airfoils," *Shock Waves*, vol. 26, no. 4, pp. 491–511, 2016. [Online]. Available: https://doi.org/10.1007/s00193-015-0591-2 8

[18] A. Ramezani and K. Mazaheri, "Multigrid convergence acceleration for implicit and explicit solution of euler equations on unstructured grids," *International Journal for*

Numerical Methods in Fluids, vol. 62, no. 9, pp. 994–1012, 2010. [Online]. Available: https://doi.org/10.1002/fld.2058 8

[19] S. Peigin and B. Epstein, "Constrained optimization of aerodynamic shapes via minimization of total drag," *Inverse Problems in Science and Engineering*, vol. 13, no. 3, pp. 299–327, 2005. [Online]. Available: https://doi.org/10.1080/17415970500044059 9

[20] J. Blazek, *Computational fluid dynamics: Principles and applications*, 3rd ed. Oxford, England: Butterworth-Heinemann, 2015. [Online]. Available: https://doi.org/10.1016/B978-0-08-099995-1.00012-9 9

[21] B. Epstein, T. Rubin, and S. Séror, "Accurate multiblock navier-stokes solver for complex aerodynamic configurations," *AIAA Journal*, vol. 41, no. 4, pp. 582–594, 2003. [Online]. Available: https://doi.org/10.2514/2.2012 9

[22] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press, 04 1992. [Online]. Available: https://doi.org/10.7551/mitpress/1090.001.0001 9

[23] J. R. Koza, "Genetic programming as a means for programming computers by natural-selection," *STATISTICS AND COMPUTING*, vol. 4, no. 2, pp. 87–112, JUN 1994. [Online]. Available: https://doi.org/10.7551/mitpress/1090.001.0001 9

[24] R. Storn and K. Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," *JOURNAL OF GLOBAL OPTIMIZATION*, vol. 11, no. 4, pp. 341–359, DEC 1997. [Online]. Available: https://doi.org/10.1023/A:1008202821328 9

[25] W. Vent, "Rechenberg, ingo, evolutionsstrategie — optimierung technischer systeme nach prinzipien der biologischen evolution. 170 s. mit 36 abb. frommann-holzboog-verlag. stuttgart 1973. broschiert," *Feddes Repertorium*, vol. 86, no. 5, pp. 337–337, 1975. [Online]. Available: https://doi.org/10.1002/fedr.19750860506 9

[26] D. B. Fogel, *Artificial Intelligence through Simulated Evolution*. Wiley-IEEE Press, 1998, pp. 227–296. [Online]. Available: https://doi.org/10.1109/9780470544600.ch7 9

[27] R. Eberhart, P. Simpson, and R. Dobbins, *Computational intelligence PC tools*. Oxford, England: Academic Press Professional, 1996. 10

[28] R. Eberhart and J. Kennedy, “A new optimizer using particle swarm theory,” in *MHS’95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*. IEEE, 1995, pp. 39–43. [Online]. Available: https://doi.org/10.1109/MHS.1995.494215 10

[29] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95 - International Conference on Neural Networks*, vol. 4. IEEE, 1995, pp. 1942–1948. [Online]. Available: https://doi.org/10.1109/ICNN.1995.488968 10

[30] J. Kennedy, “The particle swarm: social adaptation of knowledge,” in *Proceedings of 1997 IEEE International Conference on Evolutionary Computation (ICEC ’97)*. IEEE, 1997, pp. 303–308. [Online]. Available: https://doi.org/10.1109/ICEC.1997.592326 10, 11

[31] Y. Shi and R. Eberhart, “A modified particle swarm optimizer,” in *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*. IEEE, 1998, pp. 69–73. [Online]. Available: https://doi.org/10.1109/ICEC.1998.699146 11, 14, 55

[32] R. Poli, “Mean and variance of the sampling distribution of particle swarm optimizers during stagnation,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 712–721, 2009. [Online]. Available: https://doi.org/10.1109/TEVC.2008.2011744 12

[33] C. W. Cleghorn and A. P. Engelbrecht, “Particle swarm stability: a theoretical extension using the non- stagnate distribution assumption,” *SWARM INTELLIGENCE*, vol. 12, no. 1, pp. 1–22, 2018. [Online]. Available: https://doi.org/10.1007/s11721-017-0141-x 12

[34] R. Eberhart and Y. Shi, “Comparing inertia weights and constriction factors in particle swarm optimization,” in *Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512)*, vol. 1. ELSEVIER, 2000, pp. 84–88. [Online]. Available: https://doi.org/10.1109/CEC.2000.870279 12

[35] D. Tian and Z. Shi, “MPSO: Modified particle swarm optimization and its applications,” *SWARM AND EVOLUTIONARY COMPUTATION*, vol. 41, pp. 49–68, 2018. [Online]. Available: https://doi.org/10.1016/j.swevo.2018.01.011 12

[36] R. Eberhart and Y. Shi, “Comparing inertia weights and constriction factors in particle swarm optimization,” in *Proceedings of the 2000 Congress on Evolutionary*

Computation. CECoo (Cat. No.ooTH8512), vol. 1. IEEE, 2000, pp. 84–88. [Online]. Available: https://doi.org/10.1109/CEC.2000.870279 14

[37] G. Chen, X. Huang, J. Jia, and Z. Min, "Natural exponential inertia weight strategy in particle swarm optimization," in *2006 6th World Congress on Intelligent Control and Automation*, vol. 1. IEEE, 2006, pp. 3672–3675. [Online]. Available: https://doi.org/10.1109/WCICA.2006.1713055 15, 55

[38] G. Chen, J. J. Yuan, and H. Qi, "Study on the strategy of decreasing inertia weight in particle swarm optimization algorithm," in *Journal-Xian Jiao tong University*, vol. 40, 2006, pp. 1742–6596. 15

[39] L. Li, B. Xue, B. Niu, Y. Chai, and J. Wu, "The novel non-linear strategy of inertia weight in particle swarm optimization," in *2009 Fourth International on Conference on Bio-Inspired Computing*. IEEE, 2009, pp. 1–5. [Online]. Available: https://doi.org/10.1109/BICTA.2009.5338130 15

[40] Y. Shi and R. Eberhart, "Fuzzy adaptive particle swarm optimization," in *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, vol. 1. IEEE, 2001, pp. 101–106. [Online]. Available: https://doi.org/10.1109/CEC.2001.934377 15

[41] H. Liu and A. Abraham, "Fuzzy adaptive turbulent particle swarm optimization," in *Fifth International Conference on Hybrid Intelligent Systems (HIS'05)*. IEEE, 2005, p. 6. [Online]. Available: https://doi.org/10.1109/ICHIS.2005.49 16

[42] P. Yadmellat, S. M. A. Salehizadeh, and M. B. Menhaj, "A new fuzzy inertia weight particle swarm optimization," in *2009 International Conference on Computational Intelligence and Natural Computing*, vol. 1. IEEE, 2009, pp. 507–510. [Online]. Available: https://doi.org/10.1109/CINC.2009.180 16

[43] R. Eberhart and Y. Shi, "Tracking and optimizing dynamic systems with particle swarms," in *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, vol. 1. IEEE, 2001, pp. 94–100. [Online]. Available: https://doi.org/10.1109/CEC.2001.934376 16

[44] X. Shen, Z. Chi, J. Yang, C. Chen, and Z. Chi, "Particle swarm optimization with dynamic adaptive inertia weight," in *2010 International Conference on Challenges in Environmental Science and Computer Engineering*, vol. 1. IEEE, 2010, pp. 287–290. [Online]. Available: https://doi.org/10.1109/CESCE.2010.16 17

[45] X. Yang, J. Yuan, J. Yuan, and H. Mao, “A modified particle swarm optimizer with dynamic adaptation,” *Applied Mathematics and Computation*, vol. 189, no. 2, pp. 1205–1213, 2007. [Online]. Available: https://doi.org/10.1016/j.amc.2006.12.045 17

[46] C. Feng, S. Cong, and X. Y. Feng, “A new adaptive inertia weight strategy in particle swarm optimization,” in *2007 IEEE Congress on Evolutionary Computation*. IEEE, 2007, pp. 4186–4190. [Online]. Available: https://doi.org/10.1109/CEC.2007.4425017 17

[47] R. Palmeira, “Low speed aerofoil optimization,” MSc Dissertation, Universidade da Beira Interior, Covilhã, Portugal, 2022. [Online]. Available: http://hdl.handle.net/10400.6/13036 19, 23, 46, 55

[48] D. Prosser, “XOPTFOIL,” 2019. [Online]. Available: https://github.com/montagdude/Xoptfoil 19, 46

[49] G. Pangas, “XOPTFOIL V3,” 2024. [Online]. Available: https://github.com/Gpangas/Xoptfoil 22

[50] R. M. Hicks and P. A. Henne, “Wing design by numerical optimization,” *Journal of Aircraft*, vol. 15, no. 7, pp. 407–412, 1978. [Online]. Available: https://doi.org/10.2514/3.58379 23

[51] D. A. Masters, N. J. Taylor, T. Rendall, C. B. Allen, and D. J. Poole, “Review of aerofoil parameterisation methods for aerodynamic shape optimisation,” in *53rd AIAA Aerospace Sciences Meeting*. AIAA, 2015. [Online]. Available: https://doi.org/10.2514/6.2015-0761 23

[52] I.-C. Chang, F. J. Torres, and C. Tung, “Geometric analysis of wing sections,” Ames Research Center;U.S. Army Aviation and Troop Command, Tech. Rep., 1995. [Online]. Available: https://doi.org/10.21949/1403585 23

[53] D. A. Masters, N. J. Taylor, T. Rendall, C. B. Allen, and D. J. Poole, “Review of aerofoil parameterisation methods for aerodynamic shape optimisation,” in *53rd AIAA Aerospace Sciences Meeting*. AIAA, 2015. [Online]. Available: https://doi.org/10.2514/6.2015-0761 24

[54] B. Kulfan and J. Bussoletti, “”fundamental” parameteric geometry representations for aircraft component shapes,” in *11th AIAA/ISSMO Multidisciplinary Analysis and*

Optimization Conference. AIAA, 2006. [Online]. Available: https://doi.org/10.2514/6.2006-6948 24

[55] B. M. Kulfan. (2009) Modification of cst airfoil representation methodology. [Online]. Available: https://api.semanticscholar.org/CorpusID:198922917 25

[56] R. W. Derksen and T. Rogalsky, "Bezier-parsec: An optimized aerofoil parameterization for design," *Advances in Engineering Software*, vol. 41, no. 7, pp. 923–930, 2010. [Online]. Available: https://doi.org/10.1016/j.advengsoft.2010.05.002 25

[57] M. Drela, "Xfoil: Subsonic airfoil development system," 2013. [Online]. Available: https://web.mit.edu/drela/Public/web/xfoil/ 29

[58] S. Gudmundsson, *General Aviation Aircraft Design*. Boston: Butterworth-Heinemann, 2014. [Online]. Available: https://doi.org/10.1016/C2011-0-06824-2 33, 36, 38, 53, 73

[59] S. F. Hoerner, *Fluid-Dynamic Drag*. Bakesfield: Sighard F. Hoerner, 1965. 45

[60] V. Zombori, "Study of electronic speed control strategies for a fixed battery, motor and propeller aircraft propulsion set," MSc Dissertation, Universidade da Beira Interior, Covilhã, Portugal, 2021. [Online]. Available: http://hdl.handle.net/10400.6/11940 46

[61] E. Juárez-Castillo, H.-G. Acosta-Mesa, and E. Mezura-Montes, "Empirical study of bound constraint-handling methods in particle swarm optimization for constrained search spaces," in *2017 IEEE Congress on Evolutionary Computation (CEC)*, 2017, pp. 604–611. [Online]. Available: https://doi.org/10.1109/CEC.2017.7969366 48

[62] AkaModell Munich. (2022) Aircargo challenge 2022, final results flights. [Online]. Available: https://akamodell-muenchen.de/wp-content/uploads/2022/07/acc2022_final_results.pdf 51

[63] A. P. Piotrowski, J. J. Napiorkowski, and A. E. Piotrowska, "Population size in particle swarm optimization," *Swarm and Evolutionary Computation*, vol. 58, p. 100718, 2020. [Online]. Available: https://doi.org/10.1016/j.swevo.2020.100718 55

Appendix A

Performance Objective Input

```
1  &aircraft_data
2    weight_i = 41.59              !Aircraft total weight, [N]
3    S_w = 0.426                   !Wing area [m^2]
4    A_w = 11.3                    !Aspect ratio
5    e_w = 0.98                     !Oswald efficiency
6    S_expose = 0.402              !Expose wing area [m^2]
7
8    thrust_coeff(1) = -0.0045     !Thrust coefficient [T/V^2]
9    thrust_coeff(2) = -0.2554     !Thrust coefficient [T/V]
10   thrust_coeff(3) = 14.964      !Thrust coefficient [T]
11
12   height = 0.175                !Fuselage height [m]
13   width = 0.124                 !Fuselage width [m]
14   length = 0.710                !Fulesage length [m]
15   f_skin_roughness = 6.34E-6    !Fuselage skin roughness [m]
16   wetted_area = 0.274           !Fuselage wetted area [m^2]
17   interference_factor = 1.0     !Fuselage interference factor
18
19   tail_config = 1               !1 for V-tail, 2 for Conventional_tail
20   tail_chord(1) = 0.125         !Tail chord [m]
21   tail_surface_area(1) = 0.027  !Tail surface area [m^2]
22   t_c_ratio(1) = 0.09           !Tail thickness chord ratio
23   max_t_x(1) = 0.3              !Tail maximum airfoil thickness relative x
         position
24   t_skin_roughness(1) = 6.34E-6 !Tail skin roughness [m]
25
26   CD_ld = 0.00051               !Landing gear drag coefficient
27   CD_add = 0.0085               !Extra drag coefficient
28 /
29
30 &take_off_data
31   h_take_off = 474              !Runway altitude [m]
32   A_1 = 1.1                     !Rotation factor
```

```
33    miu = 0.2626                  !Friction coefficient
34    S_g = 60                      !Runway length [m]
35    weight_empty = 23.92          !Aircraft empty weigth [N]
36    weight_payload_ref = 31.470   !Reference payload weigth [N]
37  /
38
39  &climb_data
40    accel_to_climb = .true.       !Acceleration to climb speed activation
41    time_climb = 60               !Time to climb [s]
42    h_climb = 524                 !Average climb altitude [m]
43    dh_climb = 100                !Altitude to climb [m]
44    V_0_climb = 0                 !Starting velocity to acceleration without take-
          off phase [m/s]
45    points_coeff(1) = -3.92E-5    !Climb points coefficient [points/V^4]
46    points_coeff(2) = 1.08E-2     !Climb points coefficient [points/V^3]
47    points_coeff(3) = -1.156      !Climb points coefficient [points/V^2]
48    points_coeff(4) = 64.2        !Climb points coefficient [points/V]
49    points_coeff(5) = -537        !Climb points coefficient [points]
50  /
51
52  &cruise_data
53    accel_to_cruise = .true.      !Acceleration to dash speed activation
54    time_cruise = 120             !Time to travel [s]
55    h_cruise = 574                !Dash altitude [m]
56    dist_ref_cruise = 3255        !Reference distance traveled [m]
57    V_0_cruise = 0                !Starting velocity to acceleration without climb
          phase [m/s]
58    time_extra = 0                !Time extra without climb phase [s]
59  /
60
61  &turn_data
62    activation_turn = .true.      !Turn activation
63    h_turn = 574                  !Turn altitude [m]
64    n_turn = 2                    !Turn load factor
65    field_length = 375            !Maximum leg lenght [m]
66  /
```

Appendix B

Operating Points for Case 2

Table B.1: Initial operating conditions for the no acceleration scenario.

Flight stages	Operating points	Re	α	C_l	M	hinge x	δ
Take-off	1	151000	5^o to 9^o , step 1^o	–	0.03	0.80	25.00
	2	107000	7^o	–	0.02	0.80	0.00
Climb	3	151000	–	1.20	0.03	0.80	4.00
	4	162000	–	1.05	0.04	0.80	4.00
	5	174000	–	0.90	0.04	0.80	4.00
	6	191000	–	0.75	0.04	0.80	4.00
	7	214000	–	0.60	0.05	0.80	4.00
Turn	8	302000	–	0.60	0.07	0.80	0.00
	9	315000	–	0.55	0.07	0.80	0.00
	10	331000	–	0.50	0.08	0.80	0.00
	11	348000	–	0.45	0.08	0.80	0.00
Cruise	12	302000	–	0.30	0.07	0.80	0.00
	13	331000	–	0.25	0.08	0.80	0.00
	14	370000	–	0.20	0.09	0.80	0.00

Table B.2: Initial operating conditions for take-off in a 40 meters runway scenario.

Flight stages	Operating points	Re	α	C_l	M	hinge x	δ
Take-off	1	143000	5^o to 9^o , step 1^o	–	0.03	0.80	25.00
	2	101000	7^o	–	0.02	0.80	0.00
Climb	3	146000	–	1.15	0.04	0.80	4.00
	4	157000	–	1.00	0.04	0.80	4.00
	5	170000	–	0.85	0.05	0.80	4.00
	6	187000	–	0.70	0.05	0.80	4.00
	7	211000	–	0.55	0.06	0.80	4.00
Turn	8	313000	–	0.50	0.07	0.80	0.00
	9	329000	–	0.45	0.08	0.80	0.00
	10	349000	–	0.40	0.08	0.80	0.00
	11	374000	–	0.35	0.08	0.80	0.00
Cruise	12	233000	–	0.45	0.05	0.80	0.00
	13	313000	–	0.25	0.07	0.80	0.00
	14	349000	–	0.20	0.08	0.80	0.00
	15	403000	–	0.15	0.09	0.80	0.00

Table B.3: Initial operating conditions for the runway friction coefficient of 0.07 scenario.

Flight stages	Operating points	Re	α	C_l	M	hinge x	δ
Take-off	1	173000	6^o to 10^o , step 1^o	–	0.04	0.80	25.00
	2	122000	7^o	–	0.03	0.80	0.00
Climb	3	169000	–	1.25	0.05	0.80	4.00
	4	180000	–	1.10	0.05	0.80	4.00
	5	194000	–	0.95	0.05	0.80	4.00
	6	212000	–	0.80	0.06	0.80	4.00
	7	235000	–	0.65	0.06	0.80	4.00
Turn	8	290000	–	0.85	0.07	0.80	0.00
	9	299000	–	0.80	0.07	0.80	0.00
	10	309000	–	0.75	0.07	0.80	0.00
	11	320000	–	0.70	0.07	0.80	0.00
Cruise	12	255000	–	0.55	0.06	0.80	0.00
	13	320000	–	0.35	0.07	0.80	0.00
	14	345000	–	0.30	0.08	0.80	0.00
	15	378000	–	0.25	0.09	0.80	0.00

Table B.4: Initial operating conditions for the 40 meters runway with the friction coefficient of 0.07 scenario.

Flight stages	Operating points	Re	α	C_l	M	hinge x	δ
Take-off	1	160000	6^o to 10^o , step 1^o	–	0.04	0.80	25.00
	2	113000	7^o	–	0.03	0.80	0.00
Climb	3	160000	–	1.20	0.04	0.80	4.00
	4	171000	–	1.05	0.05	0.80	4.00
	5	185000	–	0.90	0.05	0.80	4.00
	6	202000	–	0.75	0.06	0.80	4.00
	7	226000	–	0.60	0.06	0.80	4.00
Turn	8	296000	–	0.70	0.07	0.80	0.00
	9	307000	–	0.65	0.07	0.80	0.00
	10	319000	–	0.60	0.07	0.80	0.00
	11	333000	–	0.55	0.08	0.80	0.00
Cruise	12	247000	–	0.50	0.06	0.80	0.00
	13	319000	–	0.30	0.07	0.80	0.00
	14	350000	–	0.25	0.08	0.80	0.00
	15	391000	–	0.20	0.09	0.80	0.00

Table B.5: Initial operating conditions for PVG_ACC2021 scenario.

Flight stages	Operating points	Re	α	C_l	M	hinge x	δ
Take-off	1	152000	11^0 to 15^0 , step 1^0	–	0.04	0.80	25.00
	2	108000	7^0	–	0.02	0.80	0.00
Climb	3	155000	–	1.20	0.04	0.80	0.00
	4	166000	–	1.05	0.05	0.80	0.00
	5	179000	–	0.90	0.05	0.80	0.00
	6	197000	–	0.75	0.05	0.80	0.00
	7	220000	–	0.60	0.06	0.80	0.00
Turn	8	298000	–	0.65	0.07	0.80	-6.00
	9	310000	–	0.60	0.08	0.80	-6.00
	10	324000	–	0.55	0.08	0.80	-6.00
	11	340000	–	0.50	0.08	0.80	-6.00
Cruise	12	240000	–	0.50	0.06	0.80	-6.00
	13	310000	–	0.30	0.07	0.80	-6.00
	14	340000	–	0.25	0.08	0.80	-6.00
	15	380000	–	0.20	0.09	0.80	-6.00