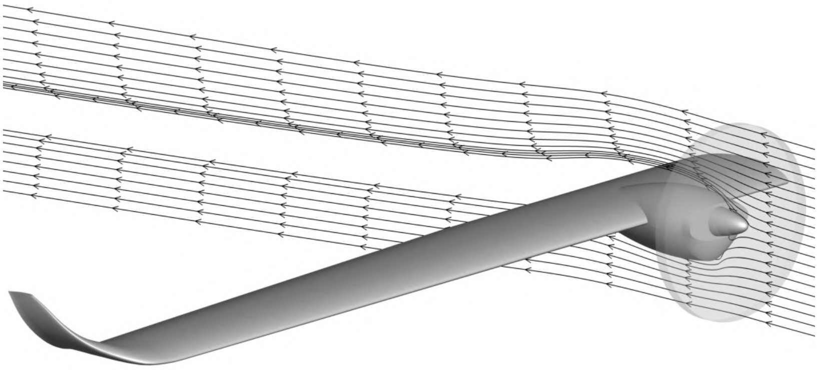


Handling of Propeller-Driven Propulsion System in High-Fidelity Aerodynamic Shape Optimization of Aircraft

Vicente Lhansol Massapina Vaz

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor: Prof. André Calado Marta

Examination Committee

Chairperson: Prof. João Manuel Melo de Sousa
Supervisor: Prof. André Calado Marta
Member of the Committee: Prof. Frederico José Prata Rente Reis Afonso

November 2025

Dedicated to my family.

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

First and foremost, I would like to express my deepest appreciation to my supervisor, Professor André Calado Marta at Instituto Superior Técnico, whose constant support, availability, guidance, and expertise were invaluable to this project.

I am also extremely grateful to the rest of the work group: Eng. Nuno Matos at TEKEVER, Eng. Pedro Marques Cardoso and Eng. Luís Dias Pinheiro at TEKEVER, for generously dedicating their time and patience to assisting me and for sharing their valuable knowledge.

I must also express my deepest gratitude to TEKEVER UAS for granting me the opportunity to pursue this subject, for their funding, and for all the necessary data essential to this work.

I would also like to thank MDO Lab at the University of Michigan, especially Professor Joaquim Martins for his insight and feedback, and Dr. Eirikur Jonsson, for allowing access to the MACH-Aero framework.

To my parents and my sister, I am extremely grateful for your unconditional support and endless words of encouragement. Thank you for making sure I always had everything I needed.

Finally, I would like to thank all the institutions and teams I have been part of throughout my academic journey, as well as my professors, colleagues, and friends, whose support and collaboration have contributed to my growth as both a student and a person.

Resumo

A interação entre a hélice e a asa em aeronaves provoca alterações significativas no desempenho da asa, e a sua compreensão permite introduzir modificações para a melhorar. Os modelos de propulsão são utilizados para estimar os efeitos da hélice em simulações numéricas, sendo o modelo do disco atuador conhecido pela sua simplicidade, custo computacional moderado e precisão nos resultados. Neste trabalho, estudamos o modelo do disco atuador e aplicamo-lo a um caso inspirado no cenário real do TEKEVER AR5. É apresentada uma revisão dos efeitos da hélice e são exploradas as limitações deste modelo. Para completar o sistema propulsivo, é introduzida uma geometria da carenagem. Uma vez que investigações anteriores, que utilizaram este modelo em mecânica dos fluidos computacional de alta fidelidade, encontraram melhorias negligenciáveis na otimização da forma aerodinâmica da asa de um veículo aéreo não-tripulado, iremos revisitar o mesmo assunto, focando-nos agora na região da carenagem. Faremos otimizações baseadas em gradientes, com Reynolds-averaged Navier-Stokes e o método adjunto para análise de sensibilidade. A forma aerodinâmica da carenagem e da asa é modificada utilizando uma caixa de deformação de forma livre, que inclui um constrangimento de volume para garantir espaço suficiente no interior da carenagem para o motor. No nosso contexto, uma abordagem em duas etapas revelou-se benéfica: primeiro, otimizar a forma da carenagem isoladamente, e depois otimizar a configuração conjunta da asa e da carenagem. No nosso caso, este método resultou numa redução de resistência de 5% em comparação com o caso otimizado sem o disco atuador.

Palavras-chave: Otimização baseada em gradientes, método adjunto, deformação de forma livre, disco atuador, configuração com asa e carenagem

Abstract

The interaction between the propeller and the wing in wing-mounted engine aircraft causes significant changes in the wing's performance, and its understanding allows for design changes that can improve performance. Propulsion models are used to estimate propeller effects in numerical simulations, with the actuator disk model being commonly adopted due to its simplicity, medium computational cost, and accurate time-averaged results. In this work, we study the actuator disk model and apply it to a case inspired by the TEKEVER AR5 drone. A review of the propeller effects is documented, and the limitations of this propulsion model are explored. To complete the propulsion system, a nacelle geometry is introduced. Since previous research using this model embedded in high-fidelity computational fluid dynamics found negligible improvements in unmanned aerial vehicle wing aerodynamic shape optimization, we revisit the same premise while focusing on the nacelle region. A gradient-based optimization is employed in conjunction with a Reynolds-averaged Navier-Stokes flow solver, utilizing an adjoint method for sensitivity analysis. The nacelle and wing shapes are modified using the free-form deformation method, which includes a volume constraint to ensure sufficient space for the engine inside the nacelle. In our context, a two-step approach proved beneficial: first, optimizing the nacelle shape alone to improve design freedom, and then optimizing the wing and nacelle configuration. In our test case, this approach yields a 5% drag reduction compared to the case optimized without the actuator disk.

Keywords: Gradient-based optimization, adjoint method, free-form deformation, actuator disk, volume constraint, wing and nacelle configuration

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α	Angle of attack
α_p	Propeller pitch angle
α'_p	Propeller angle of attack
Δs	Initial wall spacing
Δ	Difference in quantity
δ	Boundary layer thickness
γ	Adiabatic constant
μ	Molecular viscosity coefficient
μ_t	Turbulent viscosity coefficient
∇	Gradient
ρ	Density
Θ	Azimuthal angle of the propeller

Roman symbols

b	Half span
c	Chord length
C_D	Coefficient of drag
C_L	Coefficient of lift
C_p	Coefficient of pressure
C_{D_f}	Skin friction drag coefficient
C_{D_p}	Pressure drag coefficient
D	Drag

E	Total energy
F	Force
h	Cruise altitude
L	Lift
Q	Torque
q	Heat flux
R	Propeller radius
S	Projected area
T	Thrust
t	Thickness
y^+	Non-dimensional wall distance
p	Pressure
U	Velocity vector
u, v, w	Velocity Cartesian components

Subscripts

∞	Free-stream condition
i, j, k	Computational indexes
n	Normal component
ref	Reference condition
x, y, z	Cartesian components

Superscripts

$\hat{}$	Reference value

Glossary

AD	Actuator Disk
ANK	Approximate Newton-Krylov
BEM	Blade Element Momentum
BET	Blade Element Theory
BFGS	Broyden–Fletcher–Goldfarb–Shanno algorithm
CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
CGNS	CFD General Notation System
CPU	Central Processing Unit
CRM	Common Research Model
CSM	Computational Structural Mechanics
DNS	Direct Numerical Simulation
DVE	Distributed Vorticity Element
eVTOL	Electric Vertical Takeoff and Landing
FFD	Free-Form Deformation
FIM	Full-Interaction-Mode
GMRES	Generalized Minimal Residual Method
HALE	High Altitude Long Endurance
HPC	High Performance Computing
IHC	Implicit Hole Cutting
IU	Inboard Up rotation direction
KKT	Karush–Kuhn–Tucker
LES	Large Eddy Simulation
MAME	MediumAltitude Medium Endurance
MDO	Multidisciplinary Design Optimization
NACA	NationalAdvisory Committee for Aeronautics
NASA	National Aeronautics and Space Administration
NK	Newton-Krylov
NPSE	Nonlinear Parabolized Stability Equations
OU	Outboard Up rotation direction

PEGASUS	Parallel Electric-Gas Architecture with Synergistic Utilization Scheme
PPA	Propeller Pitch Angle
PTD	Propeller Tilt Down
RAM	Random Access Memory
RANS	Reynolds-Averaged Navier-Stokes
SA	Spalart-Allmaras
SIM	Single-Interaction-Mode
SLSQP	Sequential Least Squares Programming Algorithm
STOL	Short Takeoff and Landing
UAM	Urban Air Mobility
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
URANS	Unsteady Reynolds-Averaged Navier-Stokes
VLM	Vortex-Lattice Method
VTOL	Vertical Takeoff and Landing
XDSM	eXtended Design Structure Matrix

Chapter 1

Introduction

1.1 Motivation

With the recent interest in hybrid and electric aircraft in today's highly competitive and ever-growing market of unmanned aerial vehicles (UAVs), the demand for improved designs of propeller aircraft has never been so high. In fact, UAV market has spiked in recent years, reaching $US\$33.7$ billion in worldwide revenue in 2023 and projections foreseeing an increase to $US\$54.6$ billion by the end of the decade [1]. Additionally, electric-powered advanced air mobility concepts have also drawn great attention in urban air transport, which holds the potential to change city development and the way we connect [2–4].

When developing new aircraft of the aforementioned type, the positioning and characteristics of the propulsion systems are of great importance, since a good solution can present many improvements in the aerodynamic efficiency of the design. It is possible to take advantage of aerodynamic and propulsion-integration benefits by considering the interactions between propellers and the main wing [5].

Therefore, the motivation is driven by the desire of a Portuguese leading unmanned aerial system (UAS) manufacturer, TEKEVER UAS, in developing a new Medium-Altitude, Medium-Endurance (MAME) UAV, whose propulsion system will include propellers mounted on the UAV's main wing, similar to the current model TEKEVER AR5, presented in Fig. 1.1.

Figure 1.1: TEKEVER AR5 UAS [6].

The present work intends to provide optimal and practical solutions to the integration of propulsion systems in a UAV. To achieve this, optimization tools with coupled aerodynamic high-fidelity computational analysis tools will be used. This work will continue on the previous wing and wing-plus-fuselage aero-structural optimization [7], and should progress toward the inclusion of the propeller effects in the preliminary study of the new MAME UAV.

1.2 Topic Overview

Propeller-wing interactions have been a relevant subject since the beginning of aviation. One early propeller-wing interaction study was conducted by Prandtl in 1921 [8], in which it was reported that the aforementioned interactions could be distinguished in two types of influences: variations in the inflow velocity and variations in air current direction. Whilst the propeller's performance and thrust were mainly affected by the variations in the inflow velocity, the wing's drag was mainly influenced by the altered direction of the airflow, and these changes were heavily reliant on their relative positions and offset distance. Later in 1969 [9], a study reported an experiment that showed that wing-tip mounted propellers could have an effect on the wing lift and drag by changing the effective aspect ratio in flight, and that these benefits also depended on the direction of rotation of the wing-tip propeller. It was found that the best rotation direction for minimized drag was one that countered the direction of the wingtip vortices, since this would also increase the effective aspect ratio. A study performed by Kroo in 1986 [10] discovered that for a wing with an inboard propeller, the wing's lift distribution which minimized induced losses differed noticeably from the theoretical elliptic lift distribution of an unmounted wing. Another study performed in 1989 [11] determined the time-averaged performance of a propeller and wing in a conventional pull configuration, using both experimental and computational analyses, which revealed that increasing propeller power would decrease wing-induced drag, thus increasing propeller efficiency.

Understanding these propeller-wing interactions has been a topic of great importance since many benefits can be easily obtainable by optimizing the shape, position, orientation of propeller rotation, or any other characteristic, not just to maximize aircraft efficiency, but also to improve other scenarios in which this problem arises, such as wind turbines.

Early studies of wing optimization considering this topic are not abundant [12]. The scarcity of investigations in this topic in latter years can be attributed to the fact that the practice of aerodynamic shape optimization itself started in 1974 [13], with Pironneau presenting analytical calculations that used optimal control [14], achieving a shape with minimum drag and laying the mathematical groundwork for numerical solutions. In fact, it was only later in 1988 that Jameson's implementations between more advanced computational fluid dynamics (CFD) and the adjoint method were applied to aircraft wing design and ultimately to every other aircraft component that could benefit from aerodynamic optimization [13].

In 2000, Veldhuis and Heyma analyzed the aerodynamic optimization of a rectangular wing in multi-engined tractor propeller arrangements, intending to minimize drag by altering the wing twist distribution, while including a constraint for the lift coefficient [15]. They used a solver based on lifting-line theory to model the behavior of lifting surfaces, and for the drag calculations, they utilized fast calculation based on a Trefftz-plane method along with a viscous-drag estimation technique. The results show that it is beneficial to increase wing twist behind the down-going-blade region and decrease twist behind the up-going-blade region. Furthermore, their findings also suggest that a propeller-wing configuration can benefit from a relatively large negative propeller angle of attack due to the asymmetry introduced in the slipstream velocity distribution, except when viscous effects are too great, which can negate the benefits of localized leading-edge suction.

More recently, in 2018, Alba et al. [16] introduced a multidisciplinary design optimization (MDO) framework for aircraft wing design while incorporating the aerodynamic effects of tractor propellers. They also performed an optimization study of a Tecnam P2012 wing, in which the objective function was to minimize fuel weight while maintaining constraints on maximum takeoff weight and fuel tank volume. Using design variables such as wing planform, airfoil shape parameters, and twist distribution, they managed to achieve total fuel savings of 6.8% compared to the baseline wing.

However, the studies presented so far were limited by the use of low-fidelity models, such as the lifting line theory, vortex lattice methods (VLM), and semiempirical models [11]. The majority of studies that tackle propeller-wing interactions together with aerodynamic shape optimization and with high-fidelity computational models have only been made in recent years [5, 12, 17–21], which accompany the increasing availability of CFD and high-fidelity aero-structural and propulsive models, as well as the rise of modern electric aircraft designs and their need for high values of propulsive efficiency, as we will see in the next section.

Many ongoing projects work heavily on this subject, including urban air mobility designs such as the all-electric vertical takeoff and landing (eVTOL) made by Joby Aviation and NASA [2], Airbus' CityAirbus NextGen [3], and Wisk Aero's air taxi [4], as presented in Fig. 1.2.


(a) Joby Aviation and NASA [2]. (b) CityAirbus NextGen [3]. (c) Wisk Aero's Generation 6 [4].

Figure 1.2: Examples of ongoing urban air mobility projects.

These concepts present many similar benefits, such as low emissions (due to their all-electric nature), structural simplicity, and most importantly, distributed propulsion systems [22]. This aspect can reduce aircraft weight and size and facilitate the control of vertical take-off and landing (VTOL) and short take-off and landing (STOL) [22, 23]. Another advantage of distributing the thrust over many propellers is that it can reduce the generation of perceived noise and improve safety through redundancy, which is important for aircraft whose main mission is urban mobility [23]. The use of many propellers will also result in many instances of propeller-wing interactions, revealing the significance of this issue and how it could represent a large topic for efficiency improvement of these configurations.

The subject of propeller-wing interactions also plays a major role in the design of high-altitude long-endurance (HALE) UAVs. The choice of a propulsion system is arguably the most important subject in this process, since it is directly correlated to the aircraft's endurance capabilities [24]. Many different methods of propulsion are used in these configurations, some examples include: electric motors, liquid hydrogen engines, piston engines, and single turbofan engines for higher payload capacities [24], but the most popular method is propeller-based propulsion. To maximize the endurance of these configurations, propeller-wing interactions need to be considered and taken advantage of. Some examples are presented in Fig. 1.3.

(a) NASA's Centurion [25].

(b) Boeing's Phantom Eye [26].

(c) Aurora Flight Sciences' Theseus [27].

Figure 1.3: Examples of HALE UAV configurations.

The Centurion (Fig.1.3a) benefits from the propeller-tilt-down (PTD) configuration that was discussed previously and presented in [15], which could reduce the overall wing-induced drag for low values of air speed. Since the Centurion was developed for flights of low Reynolds numbers, it can definitely benefit from this configuration, since viscous effects are not prevalent. The Phantom Eye (Fig. 1.3b) possesses two turbocharged liquid hydrogen engines [24] mounted close to the root, to avoid large bending forces on a main wing with high aspect ratio. It utilizes a tractor propeller configuration, unlike the Theseus (Fig. 1.3c) which presents a push configuration, avoiding the clash of the propeller's wake on the main wing and allowing for increased authority of the vertical and horizontal stabilizers. However, it suffers from disturbed inflow in the propeller, which reduces its efficiency. It is also important to note that both aircraft have the propellers mounted upward to the main wing, which undoubtedly affects the propeller-wing interactions.

In fact, a study [17] investigated how the vertical offset of the propeller in relation to the wing affected performance, by simulating various analyses with steady Reynolds-Averaged Navier-Stokes (RANS) CFD and an actuator-disk approach. It was found that the trailing wing's lift and drag distributions are strongly dependent on the vertical offset, especially when comparing a positive offset (above the wing) with a negative offset (below the wing). The results showed that if the value of the offset was large enough (1.5 times the propeller radius), the effects of the propeller could almost disappear, meaning that, for example, the aforementioned distributions along the blade would assimilate to the distributions for an isolated wing, but the magnitude of the total values along the span would vary. For a propeller above the wing, upwash increases and therefore so does the C_l across the span of the wing. For a propeller below the wing, C_l across the span of the wing decreases due to the augmented downwash.

The position of the propeller along the span of the wing is also extremely important for the wing's performance, as well as the direction of rotation. The subject of optimum propeller location is also being explored for larger aircraft configurations, for instance, the Parallel Electric-Gas Architecture with Synergistic Utilization Scheme (PEGASUS), an aircraft developed by NASA, with projections to enter service in 2030 [28], could conceptually use multiple strategically located propulsors to obtain aerodynamic benefits [21]. This aircraft is presented in Fig. 1.4a.

(a) Concept design [21].

(b) Propulsion architecture [28].

Figure 1.4: PEGASUS.

As seen in Fig. 1.4b, it was proposed to equip the aircraft with four propellers mounted at the leading edge of the main wing, and a push propeller at the tail [21]. Each group of propellers would provide thrust at different stages of the flight envelope: the tip-mounted propellers would act as the primary cruise propulsion source, since these allow for induced and total drag reductions of the wing, as the study [19] showed; while the inboard propellers would provide additional thrust required for take-off and climb [28]. Wingtip mounted propellers also have a greater influence on the wing's drag [12], due to the increased vorticity of the wing tip vortices, if the direction of rotation is outboard up, or counteracted wing tip vortices, if the direction is inboard up. This explains why propeller inboard up rotation is found to be beneficial in many studies [12, 19, 29], since higher lift-drag ratios are typically obtained when compared to outboard up rotations, for constant power settings [29].

It is for these reasons that positioning the propellers on the wingtip can provide many advantages, especially when positioned with an inboard up direction of rotation, such as the case of the V-22 Osprey [30] and Airbus' experimental high-speed compound helicopter [31]. However, implementing the propeller on the wingtip comes with added challenges, since the wing's structure needs to withstand the increased bending moment. Additionally, this larger distance also implies larger yaw moments applied to the aircraft in case of an engine failure.

The effects of different diameters of tip propeller, the effects of disk loading, and how to optimally divide the thrust generated by the inboard and wingtip propellers have also been investigated [28], with recommendations to future work referencing that the wing shape and propeller design optimization could also be made in conjunction, for example, determining the optimum wing twist distribution in the presence of the propeller slipstream.

Other studies searched this possibility, where the effects of the propeller on the wing are examined to optimize the shape of a propeller-mounted rectangular wing of a theoretical UAV with twin propellers [5, 12]. However, the aerodynamic shape optimization results, made for operating points in climb and cruise (with an angle of attack of $4°$ and $0°$, respectively), proved to have little drag reduction improvements in relation to making the aerodynamic optimization on the baseline wing without the mounted propeller, when made with a simplified propulsion model, using momentum theory (actuator disk model). In [12], it is concluded that this occurs because when performing optimizations while accounting for the propeller slipstream, the optimizer attempts to decrease the drag behind the propeller, but it is limited by the lift requirements, which penalize the drag improvement and the benefit of including the actuator disk. This remark implied that for similar configurations and operating conditions, both optimized wings (with and without actuator disk) recovered the propeller's swirl as effectively as the other, rendering the inclusion of propeller forces unnecessary. Additionally, since the tangential and axial propeller velocities are approximately proportional to disk loading [12, 32] and correlated to other flow conditions, smaller or larger configurations operating at similar conditions with similar disk loadings should also have these minimal benefits when performing optimization with an actuator disk.

In fact, this could explain the contradiction between Veldhuis et al. results in [15] and Chauhan et al. in [5], since the first study investigated the optimization of the Fokker F50, a high velocity and high disk loading aircraft, and the second study investigated lower velocity aircraft, with lower disk loading, which was the aforementioned theoretical UAV. This could be explained by the fact that the aircraft that feature high disk loading and numerous swept blades to achieve high cruising speeds cause strong swirl velocities in the slipstream, which, when paired with the increased dynamic pressure, will create much more deformed lift and drag distributions [29]. Such is the case of the Airbus A400M [33]. It was also mentioned in [5] that many other aspects can change these results, such as recreating the aerodynamic shape optimization of the wing for different operation points, different sizes of the propeller, since these also cause large changes on the flow on the wing [22], or using a different propulsion model that takes into account the actual geometry of the rotor blades, such as full blade or actuator line models. The latter is especially relevant for analyzing pusher-propeller configurations, since, if it was coupled with a propeller-loading model, the computed flowfield could be used in the inflow for calculating the propeller loading distributions [12].

Moreover, propulsion systems can also affect the aircraft's stability and control, due to the changes in weight distribution and stall speed [18]. The handling of the propulsion systems' location could also offer other benefits, besides strictly aero-structural. The General Atomics MQ-9 Reaper [34] is an example of an aircraft with a rear-mounted propeller, allowing for an unobstructed view for the front camera, and an efficient defrosting of the propeller with the exhaust heat, thus also reducing the UAV's heat signature and increasing its stealth capability.

1.3 Objectives and Deliverables

The main objective will be to accomplish a series of aerodynamic shape optimizations of the propulsion model mounted on the wing of the AR5. Completing this objective will also entail: performing aerodynamic analysis with high-fidelity numerical tools, setting up high-performance computing (HPC), creating a propulsion model, carrying out a study of a simple actuator disk model to estimate the wake and loads for propeller-driven aircraft, and demonstrating its effects on a simple engine on-wing case. This work intends to show how these effects are tied to the propeller placement and how the propeller forces change the aerodynamic shape optimizations. Additional objectives include documenting the state and accuracy of the actuator disk model in the CFD solver and providing an improvement to that actuator disk model.

The propulsion model will be comprised of the nacelle, modeled and mounted on the wing, along with the actuator disk model of the propeller, effectively capturing the aerodynamics of the full configuration of a wing. Furthermore, a surface constraint will be implemented, acting as a custom volume constraint inside the nacelle, whose shape will resemble the piston engine.

In terms of deliverables, this study should provide a familiarization with the high-fidelity numerical tools and the development of an aeropropulsive framework design to be utilized in a case study of the TEKEVER AR5.

The baseline wing for this study was designed in a previous work, which performed aerodynamic optimization of the main wing without the presence of a propulsion system [7], as illustrated in Fig.1.5.

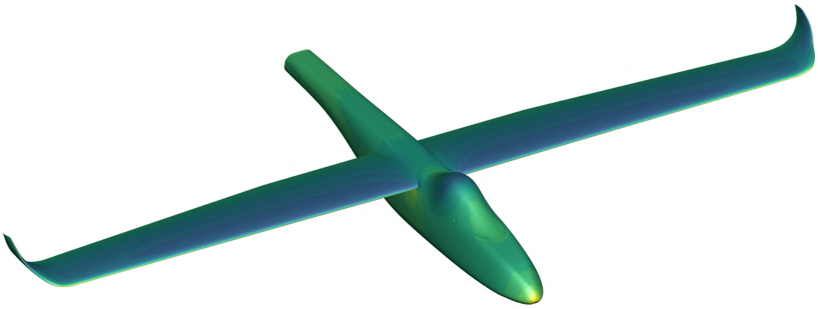


Figure 1.5: Optimized wing and fuselage of previous a work [7].

Since the MAME UAV in question is fitted with propellers, it is necessary to estimate the new aerodynamic impact on the mounted wing. The RANS-based aerodynamic shape optimization of this study will use the wing presented in Fig. 1.5 but with the inboard-mounted nacelle. To implement the actuator disk model in our simulations, modifications to the flow solver will be made, similar to prior studies [5, 12, 17]. The optimization results of the new propeller-mounted wing will then be compared to the previously obtained optimized solution in order to assess how much the propellers affected the performance of the wing. By minimizing these effects, and possibly taking advantage of them, we can further improve the preliminary design of an improved nacelle of the UAV.

1.4 Thesis Outline

The thesis is divided into six chapters. In Chapter 1, we saw a brief introduction to the history and the state of the art of the knowledge surrounding propeller-wing interactions.

Chapter 2 presents how the aerodynamic numerical analyses will be performed, along with a description of the flight conditions to be tested on the TEKEVER AR5 wing and how the volume meshes are created. Additionally, a brief discussion on the errors of flow separation predictions in our numerical model is introduced, and lastly, the cruise performance of the wing is analyzed.

Chapter 3 documents several methods to simulate propeller forces and introduces the actuator disk model, which will be used in this work. After a refinement study on both the new wing's volume mesh and the actuator disk, the impact of the propeller on the wing is studied for the current AR5 configuration. A series of studies on the effect that the position of the propeller has on the forces acting on the wing is also shown to test the limits of the actuator disk model. A small improvement in the actuator disk model is then proposed. To complete the propulsion model, the nacelle is added to the configuration.

Chapter 4 introduces the concept of gradient-based optimization using the adjoint method, and how the complex geometry of the nacelle joined with the wing is parameterized. The defined Free-Form Deformation (FFD) volume, design variables, geometric constraints, and the optimization problems are also presented, together with some of the difficulties encountered when trying to create an acceptable, but conjoined, parameterization for both the nacelle and wing.

In Chapter 5, various optimizations of the full TEKEVER AR5 configuration are performed, namely, a case with a propeller rotating with an inboard up direction, another with an outboard up rotation direction, and another without the propeller.

Finally, the conclusions from this work are presented in Chapter 6, together with suggestions for future work.

Chapter 2

Aerodynamic Analysis

Before we consider the propeller-wing interactions, we require a method of simulating the flow around a given shape. Computational fluid dynamics in aerodynamic optimization is typically performed by solving the Reynolds-Averaged Navier-Stokes (RANS) equations coupled with turbulence models [35]. Additionally, before we perform any relevant analysis, it is important to verify that the flow solver is working correctly and the framework is properly implemented. To achieve this, a simple aerodynamic analysis of the TEKEVER AR5 isolated wing will be made and compared with the given results and a previous study [7], for verification. This chapter provides the necessary context for understanding this approach and outlines the setup for flow simulation used in this work.

2.1 Numerical Model

The Navier-Stokes equations can be described by [36]

$$\nabla \cdot (\rho \mathbf{U}) = 0 \tag{2.1}$$

$$\nabla \cdot (\rho \mathbf{U}\mathbf{U}) + \nabla p - \nabla \cdot \tau = 0 \tag{2.2}$$

$$\nabla \cdot (\rho \mathbf{E}\mathbf{U}) + \nabla \cdot (p\mathbf{U}) - \nabla \cdot (\tau \cdot \mathbf{U}) + \nabla \cdot q = 0 \quad , \tag{2.3}$$

where ρ represents the density, p denotes the pressure, E refers to the total energy, and the velocity vector is given by $\mathbf{U} = [u, v, w]$, in which u, v, and w are the velocity components in a Cartesian coordinate system. The term q represents the heat flux, and τ is the viscous stress tensor, which depends on $\mathbf{U}$, the molecular viscosity μ, and the turbulent viscosity μ_t. Additionally, all body forces and internal heat sources are neglected in this work.

In laminar flow, these equations can be solved deterministically. However, as the Reynolds number surpasses a critical threshold, the flow becomes unstable due to the nonlinear convection terms. This instability leads to the occurrence of turbulence, and to address this complexity, various models of differing levels of accuracy and computational resources have been developed. Some examples include, in order of descending requirement in computer resources [37]: direct numerical simulation (DNS), large eddy simulation (LES), nonlinear parabolized stability equations (NPSE), e^N type methods based on linear stability theory, and RANS.

In the context of design optimization of aerodynamic shapes with relatively low Reynolds numbers (such as gliders and UAVs), we require the modeling of laminar-turbulent transitions. For these needs, there are typically two models to choose from [38]: DNS, which can fully capture more intricate details of the physics involved, and RANS, which provides practical but less detailed predictions and requires a transition model. The DNS method involves solving the complete three-dimensional and time-dependent solution of the Navier–Stokes equations described previously. Although it provides highly accurate results, it is impractical for most scenarios due to the complexity of the transport equations involved and it requires too many computational resources to be used within an optimization loop. The RANS approach consists of decomposing an instantaneous quantity of a flow variable into a mean time-averaged component and a fluctuating component, which allows for a simplification of the Navier-Stokes equation, and thus, less computational resources are needed to achieve reliable solutions, making it ideal for optimization problems.

For the aerodynamic analysis and posterior optimization, we will be using ADflow, which is a CFD solver for multidisciplinary analysis and optimization. ADflow consists of a core module from MACH-Aero, an open-source framework for performing gradient-based aerodynamic shape optimization, developed by the MDO Lab at the University of Michigan [39]. It calculates the Euler, laminar Navier-Stokes, and RANS equations using a second-order-accurate finite-volume approach for the spatial discretization, and computes the gradients using adjoint methods [35]. This allows for efficient aerodynamic optimization of three-dimensional shapes, as typically in these problems, the design variables outnumber the functions of interest [36]. It was originally produced to compute flows inside jet engines, but it still stands as a completely general solver [39].

Other options for open-source CFD-based optimization frameworks are quite scarce since many of the ones available are restricted to research groups in academia and government laboratories [38], with few exceptions such as SU2 [38, 40], and OpenFOAM extensions [41, 42]. Even commercial CFD solvers have implemented adjoint solvers only relatively recently and were not developed with aerodynamic shape optimization in mind, unlike ADflow and SU2.

ADflow is very advantageous due to its code efficiency [35]. The importance of this aspect is critical for MDO, as these codes are run repeatedly during the iterative optimization process, amplifying the computational costs. Aside from using the adjoint method for gradient calculation, efficient solver restarting is also a key feature in this context, allowing the flow solver to leverage the converged solution from the previous iteration as an initial guess for subsequent simulations, as well as the ability to change algorithms during the solution search process [35].

For large design changes, the solver uses robust start-up strategies to prevent failures, namely, the multi-grid algorithm, or approximate Newton-Krylov (ANK), depending on user choice. The latter algorithm was created to add robustness to a pure Newton-Krylov (NK) algorithm [43], however, it does so at the expense of efficiency [35]. This constitutes a necessary trade-off since the optimizer might try to give unfeasible intermediate designs, thus failing and interrupting the optimization process, which is a costly affair. The numerical solutions supposedly become valid as the optimum shape is approached, but intermediate designs are not accurate and mostly non-physical, rendering them useless in the case of failure [38]. Conversely, for small design changes, often occurring in the final optimization stages, the previous solution serves as a reliable initial guess, enabling rapid convergence of the NK solver. Consequently, ADflow can efficiently solve new problems involving slightly altered designs, optimizing performance in iterative MDO processes.

To solve the equations presented in the previous section, ADflow uses a variety of turbulence models designed for RANS simulations, including the Spalart-Allmaras (SA) model [44], the Wilcox k-ω model [45], and the Menter shear stress transport model [46]. By default, the SA model is used in ADflow, as it has been fully differentiated to support gradient calculations with the adjoint method [35]. The SA was determined to be the most suited for the optimization of a wing [7], due to successful validations considering other solvers and experimental data of a CRM wing, when it was coupled with RANS in ADflow [38]. All the other modules required for this study are also included in MACH-Aero's framework, such as pyGeo and pyHyp, which will be required for the analysis and shape optimization of the mounted wing, and will be presented in following sections.

In this work, ADflow will be configured to solve the RANS equations, coupled with the SA model. However, this will constitute a rather rough estimation that will be explored in section 2.3, since the SA turbulence model considers fully turbulent flow.

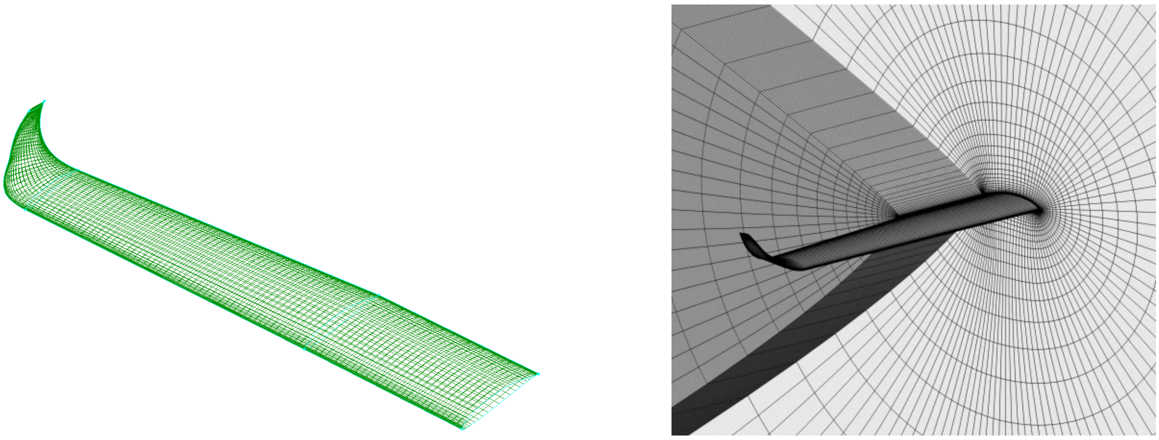
2.2 Mesh Generation

Since we are initially considering solely aerodynamic analysis, the mesh presented in this section consists of the surface mesh provided by Gameiro [7], as were most of the resources in this work. With this mesh, a baseline solution of the wing loading will be provided, and it will serve as a benchmark for other meshes created in the subsequent chapters.

2.2.1 Surface Mesh

To perform the aerodynamic analysis, we first require the surface mesh of the baseline TEKEVER AR5 wing. Despite it being provided by Gameiro [7], the steps to make it will still be described in this Section 2.2.1. The analysis results of this isolated wing will be compared subsequently with an overset mesh, which will allow the addition of an actuator disk (AD) (Section 3.5) and an added nacelle geometry (Section 3.7.3).

The first step is to define the wing's geometry. We can define this geometry with computer-aided-design (CAD) software or pyGeo, a package from the MACH-Aero Framework that allows for the generation and manipulation of said geometry. For basic wings, a simple script in Python using a pyGeo module is sufficient to create the wing's geometry. However, for complex geometries such as the TEKEVER AR5 wing, the use of CAD software is required. SolidWorks® was used for the manipulation and development of every geometry defined in this work. Secondly, the geometry needs to be transformed into a surface mesh with appropriate software. The surface mesh in Fig.2.1a was created using Pointwise® and exported to a CFD General Notation System (CGNS) file.



(a) Surface mesh created with Pointwise®.

(b) Volumetric mesh obtained with pyHyp.

Figure 2.1: TEKEVER AR5 half wing.

Care was taken in the meshing process to ensure that the mesh refinement was appropriate in regions where the gradients of the flow parameters were expected to be highest, such as the wing's leading edge, trailing edge, and wing tip. The dimensions of the TEKEVER AR5's wing were provided by TEKEVER and can be consulted in Tab.2.1, along with the number of faces present in this structured mesh.

Table 2.1: Dimensions and parameters of the surface mesh of the TEKEVER AR5 half wing.

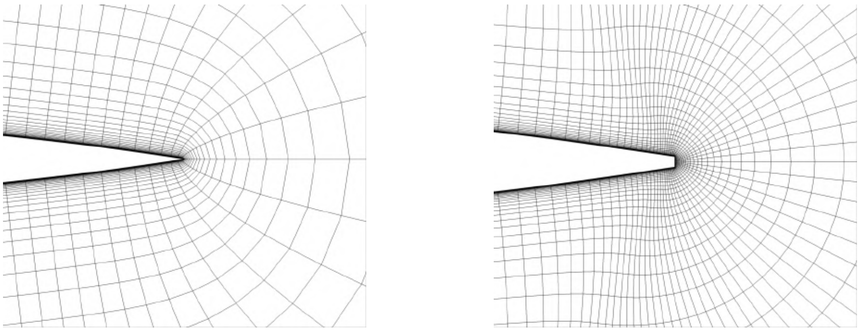
Cell count		$20\,228$	
Mean chord	c	0.6091	m
Projected area	S	2.1691	m^2
Half span	b	3.7825	m

2.2.2 Volume Mesh

Subsequently, the volume mesh was generated using the pyHyp package, a hyperbolic mesh generator that extrudes the input surface mesh (in this case, it is the previously presented mesh) in successive layers until it reaches a defined distance from the original mesh surface (defined as the farfield distance), thus meshing the surrounding space of the geometry, as presented in Fig.2.1b.

To generate a mesh of good quality, several aspects need to be considered, such as maintaining orthogonality whenever possible and minimizing the skewness angle between cells, which translates to generating cells that are as close as possible to hexahedral cubes throughout the entire volume. For the mesh to be valid, it requires no negative volume cells in it, and no or a very small number of negative quality cells. The cell quality is computed by calculating the Jacobian of each cell using a finite volume scheme employed by common flow solvers [47]. During the mesh generation process, pyHyp exhibits the cell with the minimum quality and minimum volume at each generated layer, facilitating this process.

While the airfoil geometry in the trailing edge makes it harder to enforce these qualities, some alterations can be made to the wing geometry to improve the cell quality in this region, such as slicing the wing close to the trailing edge, to amend the high skewness angle. We can see this procedure in Fig.2.2.



(a) Cells with high skewness angles due to the sharp geometry of the trailing edge.

(b) Corrected geometry allows for cells with lower skewness angles on the trailing edge.

Figure 2.2: Comparison between volume meshes.

In fact, in Fig. 2.2a, minimum cell quality achieved negative values, whereas in Fig. 2.2b only positive values were attained. For a volume mesh to be defined, pyHyp requires inputs that specify the size of the mesh, the level of refinement, along with the aforementioned file CGNS wing surface mesh, and it also allows for changes in the extrusion algorithm. In this study, the default algorithm was used.

An important parameter to evaluate to ensure sufficient mesh refinement is the non-dimensional wall distance y^+ for the near-wall cells. Since our solver will use the Spallart-Allmaras turbulence model, we will require this value to be at most $y^+ \approx 1$ to solve the turbulent boundary layer with sufficient accuracy [7, 48]. Using flat-plate boundary layer theory, an initial value for the wall spacing can be estimated as

$$\Delta s = \frac{y^+ \mu}{U_{\text{fric}}\ \rho} \quad , \tag{2.4}$$

where $U_{\text{fric}} = \sqrt{\frac{\tau_{\text{wall}}}{\rho}}$, with $\tau_{\text{wall}} = \frac{C_f \rho U_\infty^2}{2}$, and $C_f = \frac{0.027}{\text{Re}_x^{1/7}}$. From this analysis, an initial wall spacing of approximately $\Delta s \approx 1.3 \times 10^{-5}$ was determined, and consequently, the volume meshes were generated with an initial wall spacing of $\Delta s = 1 \times 10^{-5}$, as determined in the previous study [7]. With this value, the maximum y^+ value achieved in this mesh was 2.3322, which is higher than the recommended value. However, this represents only a negligible portion of the wing, with most having values between 0 and 1, as shown in [7].

As stated in [7], it was reported that an acceptable domain distance was between 20 and 25 chords, and so it was chosen for all meshes in this work to have 20 chords ($0.6091 \times 20 \approx 12$ m).

The remaining input is the number of volume layers from the surface mesh until the farfield boundary. The extrusion was performed with 97 layers, which results in a volume mesh with $7,544,832$ cells.

2.3 Flow Simulation

For both the baseline isolated wing aerodynamic analysis and the mounted-wing aerodynamic analysis in this study, the same flow conditions were established, which are the TEKEVER AR5's cruise conditions, found later in this section. Table 2.2 consists of relevant performance parameters of the AR5 that were provided by TEKEVER for a previous study [7].

Table 2.2: Performance parameters of the TEKEVER AR5.

Cruise speed	U_∞	100	km/h
Cruise altitude	h	305	m
Wing lift coefficient	$C_{L_{\text{wing}}}$	0.8932	

In this study, all the aerodynamic analyses will be performed with ADflow, and more specifically, using the *AeroProblem* Python function. Some input values are required, such as altitude, reference area, reference chord, angle of attack, and Mach number. A few parameters are already given in both Tables 2.2 and 2.1, but the Mach number, M, needs to be obtained using the given cruise speed from

$$M = \frac{U_\infty}{\sqrt{\gamma R T}} \quad , \tag{2.5}$$

in which γ represents the adiabatic constant, R, the gas constant and $T = 286.169$ K [7], resulting in $M = 0.08164$. Additionally, the Reynolds number is given by

$$Re = \frac{\rho U_\infty c}{\mu} \quad , \tag{2.6}$$

where c is the mean rectangular wing chord (Tab.2.1), μ is the dynamic viscosity and ρ is the air density at the known altitude in Tab.2.2, according to the 1976 U.S. Standard atmosphere [49] (similarly to [7]). Therefore, the values used were $\rho = 1.1895$ kg/m^3 and $\mu = 1.7798 \times 10^{-5}$ N $\cdot$ s /m^2 which resulted in $Re \approx 1.1 \times 10^6$.

As stated in [7], the trimmed condition for cruise of the TEKEVER AR5's isolated main wing resulted in an angle of attack of $1.9°$, since this achieved the provided $C_{L_{\text{wing}}}$ in Tab. 2.2. Since all the analyses of Chapter 2 and Chapter 3 considered cruise conditions, the value of $\alpha_{\text{trim}} = 1.9°$ was used in the simulations, unless otherwise specified. As mentioned throughout this chapter, this work's CFD simulations will solve the steady RANS equations, while coupled with the Spalart-Allmaras turbulence model.

Some solver values provided in [7] were altered: `L2Convergence` was considered to be 10^{-8}, and `anksecondordswitchtol` to be 10^{-4}. The convergence study is presented in Appendix A.

Regarding discretization, central difference with JST scalar dissipation will be used, and other parameters were kept at the default values as determined by Gameiro [7].

For the NK solver, a GMRES subspace of dimension 60 was employed to solve the linear systems. The Eisenstat-Walker algorithm was utilized to control linear convergence at each iteration. An additive Schwarz method served as the global preconditioner for the NK, ANK, and adjoint solvers. The Jacobian matrix used in the preconditioner is lagged by 20 iterations in the NK solver and by 10 iterations in the

ANK solver, by default [7, 50].

All the simulations in this study were performed with HPC, in a machine with 128 GB of memory and an AMD® Ryzen 9 7950x 16-core processor.

2.3.1 Laminar-Turbulent Transition Considerations

With the inputs defined, it is also important to recall that ADflow will solve the RANS equations together with the SA turbulence model. This will result in a simulation that considers fully turbulent flow, which is not ideal due to the low Reynolds numbers present in this scenario. Typically, a UAV such as the AR5 is designed for these low Reynolds conditions in which the extent of laminar flow is considerable, and so one should take into account the laminar-turbulent transition [38]. However, to model a laminar-turbulent transition while using RANS, a transition prediction method is required. Increasing computational complexity is especially unwanted in any optimization study, but it could be argued that this constitutes an oversimplification. Therefore, a verification was conducted with XFOIL [51] to determine if the error was too significant.

XFOIL is a tool designed to quickly generate airfoil performance data. It is used for designing subsonic airfoils operating in both viscous and inviscid flows at low Reynolds numbers, making it well-suited for this application.

The airfoil used in the following XFOIL simulations is the TEKEVER AR5's constant airfoil of its main wing, a custom Eppler-based airfoil. The default options, applied by the command *pane* in XFOIL, were used for the paneling of the airfoil in the 2-D simulations.

XFOIL predicts flow using a simplified e^N transition method, which is valid for situations where the increasing 2-D Tollmien-Schlichting waves are the dominant cause for initiating transition, as is the case for many airfoil applications [52]. The parameter N_{crit} is user-specified and consists of the logarithm of the amplification factor of the most amplified frequency that triggers transition. The standard value is typically $N_{crit} = 9$, with higher values decreasing the sensitivity to transition, and lower values increasing it. For sailplanes, the values that better adjust to experimental data yield $N_{crit} = 12$ to 14, and for a dirty wind-tunnel, they yield $N_{crit} = 4$ to 8 [52]. Considering that the TEKEVER AR5 has a smooth wing that minimizes drag, but it does not have the typical high aspect ratio of a sailplane wing, the N_{crit} value that is best applied for would be between 9 and 12. However, this disregards the eventual dirtiness that occurs after some time in flight, such as insects colliding with the wing, and thus hinders the smoothness of the wing, especially on the leading edge. To consider all of these scenarios, the resulting XFOIL plots were generated considering viscous flow for $Re = 1.1 \times 10^6$ and a Mach number of 0.08164, which were previously determined in Section 2.3. To observe the dependency of flow separation on the N_{crit} parameter, Fig.2.3 was created.

Figure 2.3 compares the angle of attack with the normalized chord location of the predicted start of flow transition, with $x/c = 0$ being the leading edge and $x/c = 1$ the trailing edge. It can be concluded that for any of the presented values of turbulence sensitivity, laminar flow will constitute a predominant portion of the airfoil for the considered flow scenario. More importantly, this will occur for both the upper

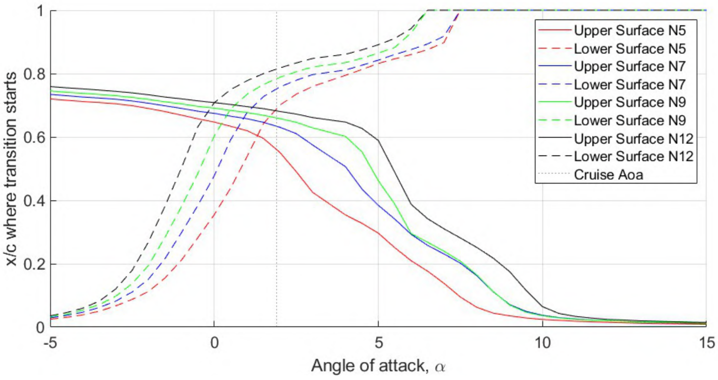
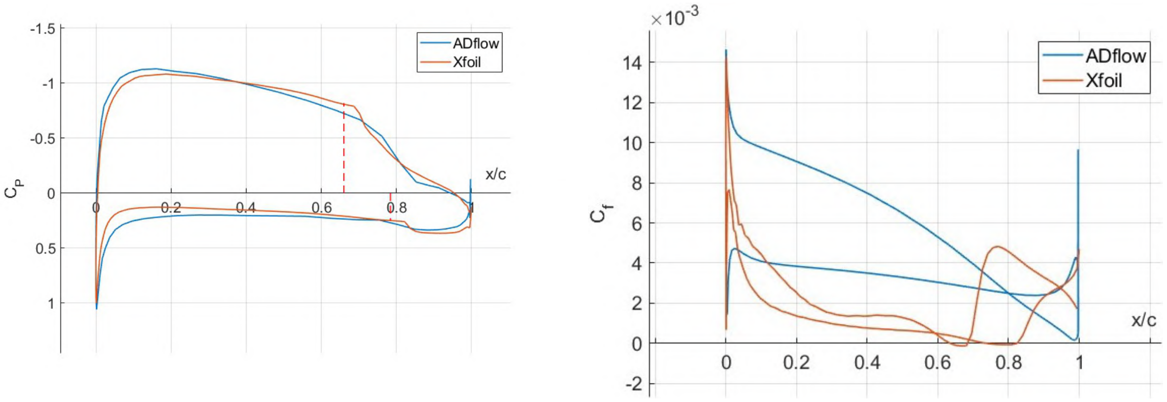


Figure 2.3: Location of transition occurrence versus angle of attack of the AR5's airfoil.

surface and lower surface within the range of angles of attack near the cruise conditions. The predicted transition in cruise of the upper surface for $N_{crit} = 5$ occurs slightly after half of the chord, and for increasing values of N_{crit}, the transition is postponed, as expected.

Therefore, there should be a considerable discrepancy between the fully turbulent ADflow simulations and the supposed true flow around the wing. To understand how the simulation results will differ from a flow considering laminar-turbulent transition, Fig. 2.4 was created to compare the ADflow results with the XFOIL data. Both simulations considered the same cruise conditions, with $\alpha = 1.9°$, and for the XFOIL case, the value of $N_{crit} = 9$ was chosen. To obtain a close distribution to the one found on an infinite wing, such as the 2-D scenario computed in XFOIL, the extracted slice from the ADflow solutions was a wing section located at $0.1\ m$ from the wing root. The distributions of the coefficient of pressure, C_p, and coefficient of friction, C_{D_f}, were plotted in Fig.2.4a and Fig.2.4b, respectively.



(a) C_p distribution.

(b) C_{D_f} distribution.

Figure 2.4: Simulation results comparison for ADflow simulation (airfoil sliced at $0.1\ m$ from the wing root) and XFOIL.

These results confirm what was expected: laminar-turbulent flow is very important in this UAV setting, as there will be noticeable differences in the simulation results, specifically in the C_{D_f} values, which are reduced in laminar flow. In Fig.2.4a, the vertical red dashed lines indicate the start of transition, whose x_{tr}/c position was taken from Fig.2.3. This position should also be coincident with where the value of C_{D_f} reaches zero first, but eventual small inconsistencies can be mitigated by increasing the panel resolution of the airfoil in the transition region, as advised in [52].

We can see that despite the distributions in Fig.2.4a appearing similar, the pressure ”bubbles” are present only in the XFOIL data. For the upper surface of the airfoil, the bubble occurs within $x_{tr} = 0.65c$ and $x_{tr} = 0.70c$, signifying flow separation and eventual transition to turbulent flow, unlike the distribution provided by ADflow with SA, which assumes fully turbulent flow and therefore no ”bubble” is present. This is also more easily assessed by noticing that the C_{D_f} distribution of ADflow in Fig.2.4b only has positive values, unlike the distribution provided by XFOIL. This will result in an overestimation of friction drag, as observed in Fig.2.4b, especially in the lower surface of the airfoil, where the area difference between the two results is notably larger.

Table 2.3: Total bi-dimensional integration values from the distributions in Fig.2.4.

		ADflow with SA model	XFOIL with e^N method
Lift coefficient	C_L	0.96026	0.93570
Friction drag coefficient	C_{D_f}	0.00883	0.00394
Pressure drag coefficient	C_{D_p}	0.027107	0.00217

As we can see from Tab.2.3, if this ratio continued for all the wing sections (and 3-D wing effects were disregarded), our ADflow simulation would overestimate the friction drag by a factor of 2.2, and lift by a less jeopardizing factor of 1.03, if we consider the XFOIL data as reference. The largest difference is in the pressure drag coefficient, which will be overestimated by a factor of 12.5. This is due to the fact that the C_{D_p} depends upon the displacement thickness, $\delta*$, imposed by the boundary layers of the upper and lower surfaces of the airfoil. The ADflow case presents a fully turbulent boundary layer since the leading edge, and this allows for increased growth of $\delta*$, when compared to the XFOIL case, where the laminar regime prevails. Consequently, the simulations in ADflow will provide C_{D_p} values much higher than those one should expect to see in reality.

Additionally, to ensure generality, another case was considered, for a rectangular wing with the NACA 0012 profile and dimensions equal to the ones presented in Tab.2.1. The process was repeated, and all of the previous parameters required for the simulation were unchanged.

Table 2.4: Resulting integration values for a rectangular wing.

		ADflow with SA model	XFOIL with e^N method
Lift coefficient	C_L	0.1863	0.1949
Friction drag coefficient	C_{D_f}	0.0090	0.0057

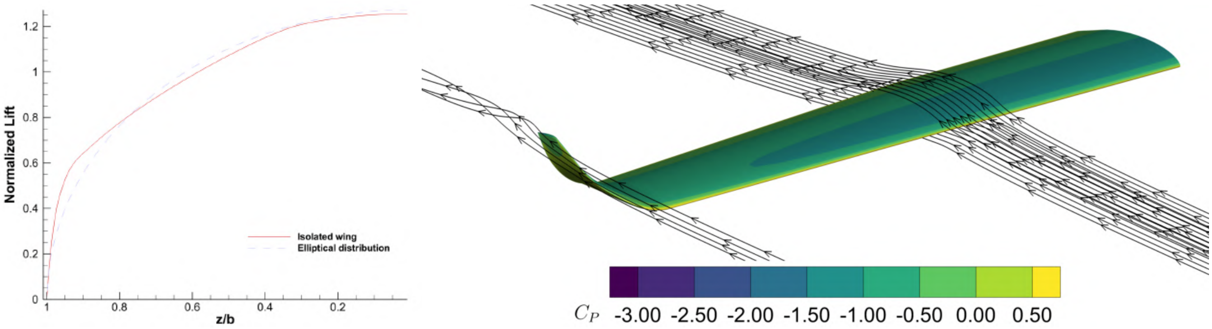
From Tab.2.4, we can deduce that the ADflow simulation would overestimate the friction drag by a factor of 1.5. Since the airfoil is different, with larger thickness, and in this case, the transition occurs sooner, the factor of overestimation will decrease significantly.

Unfortunately, this will constitute a solution error that we will subsequently ignore until the conclusions, since for the purposes of this study, there is no access to other numerical tools, and implementing a transition model in ADflow was out-of-scope.

2.3.2 Isolated Wing Aerodynamic Analysis

This subsection presents an aerodynamic analysis of the TEKEVER AR5 wing, obtained for the settings previously described, the objective being to explicitly state its aerodynamic characteristics without an acting propeller, and to serve as a baseline solution to which other meshes will be compared.

The solution for the isolated wing using the method was obtained with a processing time of 1 hour and 37 minutes with 16 allocated cores. Regarding the post-processing aspect of this study, the software used to visualize simulation data calculated with ADflow was Tecplot®. Using this software, the figures showing the resulting normalized lift distribution of the TEKEVER AR5 wing, delineated in red (Fig.2.5a) and the occurrence of a wing tip vortex (Fig.2.5b) were created.



(a) Normalized lift distribution of the rectangular wing.

(b) Wing tip vortices on the rectangular wing.

Figure 2.5: Visualizing the ADflow simulation results.

Figure 2.5a also exhibits a small discrepancy between the elliptical distribution and the lift distribution of the isolated wing, which is to be expected from a modern-day aircraft. In Fig.2.5b we can observe the 3D wing effect through the generated streamlines. The end of the wing creates a meeting region between a high-pressure zone and a lower-pressure zone, resulting in the wing tip vortex. The streamlines on the inner wing region show an ordinary and undisturbed flow behavior as expected.

Additionally, these results can be verified through the values in Tab. 2.2, since no other experimental data is available. For a simulation which recreates the cruise conditions, the resulting value of total C_D and C_L over the simulated wing is 0.0331 and 0.8966 respectively, the latter corresponding to a relative error of 0.38% when compared with the prescribed value of $C_{L_{\text{wing}}}$ for the isolated wing (0.8932). This indicates, albeit insufficiently, that the results are acceptable and that the numerical analysis framework used so far is working adequately.

Chapter 3

Propulsion Model

In order to make assessments about propeller-wing interaction, we first need to model the aerodynamic forces exerted by the propeller itself. When implementing a propeller model on a wing, we are choosing not only its geometry but also a way to represent the forces that it generates and how these will affect the wing. In this section, an overview of propeller models will be done, and one will be implemented on the rectangular wing, previously analyzed in Chapter 2.

3.1 Isolated Propeller Models

There are many propeller models with various degrees of fidelity and computational cost. Blade element theory (BET) was suggested by Drzwiecki in 1892 as a low-cost method of analyzing propellers, by assuming its blade sections as two-dimensional [53]. By neglecting the three-dimensional interactions of the propeller blades, it is possible to calculate the aerodynamic forces considering only 2D aerodynamics and integrating over the blade length [53].

Blade element momentum (BEM) methods combine BET with momentum principles to estimate tangential velocities caused by a propeller. These methods require detailed information about the propeller blade geometry, such as airfoil shapes, twist, and chord distributions. Additionally, these rely on airfoil lift and drag data and involve solving the problem iteratively to account for aerodynamic interactions. Some studies that followed this approach used CCBlade [54, 55], an open-source BEM method for analyzing wind turbine aerodynamic performance, offering reliable convergence, fast convergence rate, and a continuously differentiable formulation for efficiency and precision [56]. CCBlade can compute axial and tangential induced flow distributions using 2D airfoil data, including the hub and tip loss corrections, to properly model induced wake velocities [55].

Another well-known open source alternative is XROTOR [57], which is also a BET-based method that requires 2D airfoil data to calculate axisymmetric and spatially non-uniform distributions of slipstream induced velocities and estimate thrust and power relationships [16]. It can only provide the induced velocity at given planes, for instance, the propeller and farfield plane (as shown in Fig.3.1), and it does not account for slipstream contraction or radial flow, modeling only inflows parallel to the propeller axis, limiting its ability to capture more complex flow interactions. To take into account other phenomena, such as slipstream contraction, radial flow, intermediate velocity profiles along the flow, and different angles of attack, additional methods need to be considered. In [16], the complementary methods used were Conway's vortex–theory–based method to determine the radial flow and slipstream contraction and De Young's adjustments to address the remaining critical aspects.

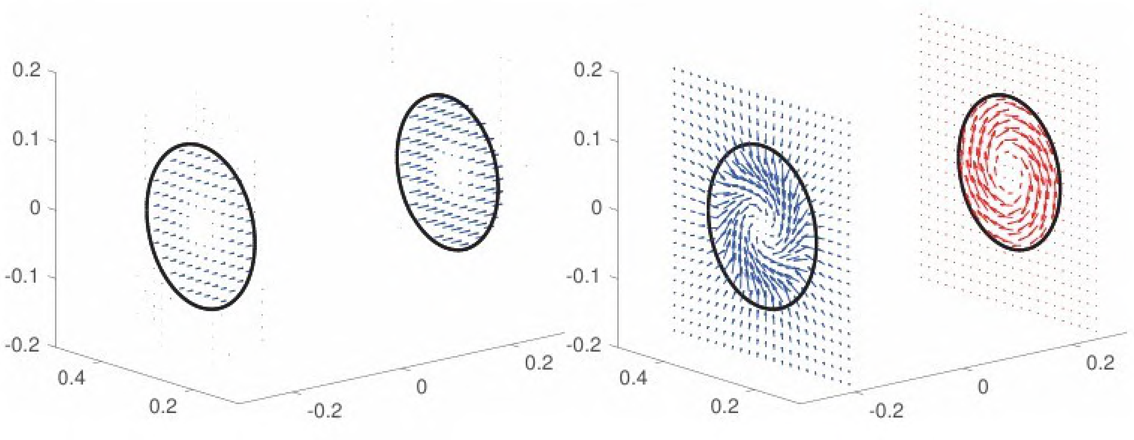


Figure 3.1: "Slipstream induced velocities: axial (left) and in plane (right)" [16].

Figure 3.1 shows Conway's vortex–theory–based method being applied to the data provided by XROTOR, allowing the assessment of induced velocity values anywhere in the flow field [16].

Besides these, there are still many other available BET methods that are often used, such as JavaProp, JBlade, RAALF, and PropCODE [58]. Despite its low fidelity, BEM methods are still very useful, not only for preliminary studies, due to their low computation cost, but also for rapid propeller loading predictions, which allowed a study [59] to use XROTOR to discover an optimal blade pitch setting of the propeller, to be used in a posterior aerodynamic analysis with a coupled wing. For even faster calculations, Chauhan also developed a method of estimating propeller inducted radial velocities by hand calculations [32].

Modeling the interaction between propellers and wings with these lower fidelity propeller models requires a way for calculating the aerodynamic forces acting on the lifting surfaces, which is typically a vortex-lattice method (VLM) model, a discretized lifting line theory that calculates the inviscid lift and lift-induced drag for non-uniform freestream and geometry [60].

For medium fidelity methods, there are the vortex-based approaches, which model the propeller's effects by representing it as a series of vortices, and depending on the specific method, they may involve solving systems of linear or nonlinear equations. Similar to BEM methods, vortex models may also require blade geometry and airfoil performance data. More advanced vortex techniques include: free-wake methods, which allow the wake to evolve naturally based on aerodynamic forces, offering higher accuracy, and vortex-particle methods, which simulate the wake using discrete particles to represent vorticity, providing detailed flow predictions at the cost of increased computational demand [32]. Such an example is the distributed vorticity element (DVE) method [61], which is a higher-order vortex-lattice method used to determine the aerodynamic parameters of non-planar geometries, by extending the traditional lifting-line and vortex-lattice approaches and modeling lifting surfaces and their wakes with DVEs. These are finite trapezoidal planar elements, which are composed of a leading-edge and a trailing-edge vortex filament with opposite circulations (to satisfy Helmholtz's theorem) and a vortex sheet between these filaments with linear vorticity distribution, creating a grid of vortex sheets which represent all aerodynamic surfaces, both the propeller and the wing, as shown in Fig.3.2.

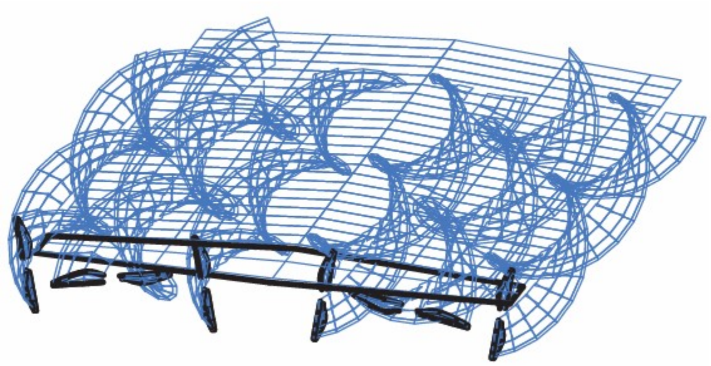


Figure 3.2: Vortex-based method for propeller modeling [61].

Higher fidelity methods require RANS based simulations to achieve higher accuracy. Methods that model an authentic full-blade three-dimensional rotating propeller geometry and simulate fully unsteady flow provide the best results, but this implies very high computational costs. This constitutes a hindrance in aerodynamic shape optimization scenarios, but for high-fidelity aerodynamic analysis, it can be used to investigate the performance of propellers and the blades themselves [59, 62].

To simplify this, an actuator line model [63] can be used, which replaces the physical propeller blades with distributions of momentum sources and energy sources along lines. One variation of this method was used in [64]. Figure 3.3 shows the reference frame, which is located in the propeller plane, employed, where n is the rotational speed, T the thrust, and Q the torque.

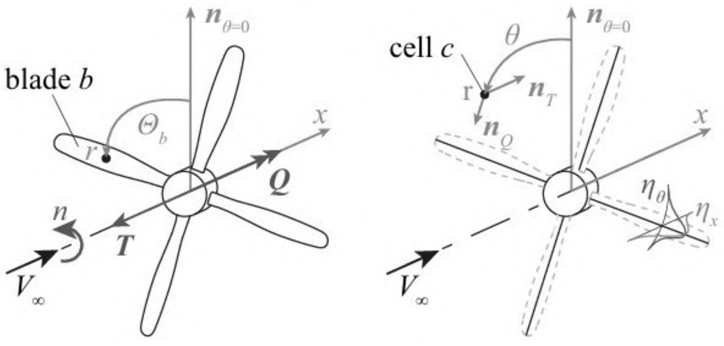


Figure 3.3: Full blade model (left) to actuator line model (right) [19].

Radial thrust and torque distributions per unit length are defined as $T'(r, \Theta_b)$ and $Q'(r, \Theta_b)$, where Θ_b is the azimuthal location of blade b. For a cell c in the propeller plane, the momentum source term vector $F(x, r, \theta)$ and the energy source term $S(x, r, \theta)$ per unit of time and volume are given as

$$F(x, r, \theta) = \eta_x \eta_\theta \left(T' \mathbf{n}_T + \frac{Q'}{r} \mathbf{n}_Q \right) \quad , \tag{3.1}$$

$$S(x, r, \theta) = \eta_x \eta_\theta F \cdot \mathbf{V} \quad , \tag{3.2}$$

where $\mathbf{n}_T$ is a unit vector opposing the thrust direction, $\mathbf{n}_Q$ a unit vector opposing the local torque direction, with η_x and η_θ being regularization functions that smooth the source terms. These functions use Gaussian kernels to smooth the source terms over the axial and azimuthal directions, defined as

$$\eta_x(x) = \frac{1}{\epsilon\sqrt{\pi}} e^{-\left(\frac{|x|}{\epsilon}\right)^2} \quad , \tag{3.3}$$

$$\eta_\theta(r, \theta) = \sum_{b=1}^{B} \frac{1}{\epsilon\sqrt{\pi}} e^{-\left(\frac{|\theta - \theta_b|}{\epsilon}\right)^2} \quad , \tag{3.4}$$

where ϵ is a constant to adjust the strength of the regularization, Θ_b is the azimuthal location of blade b, and B is the total number of blades. In the radial direction, no regularization is needed as thrust and torque are distributed per unit of radial length. The model then calculates the x, r, θ locations for each cell and updates the azimuthal position Θ_b over time, and the actuator-line movement is produced [19].

However, Veldhuis [29] states that while the propeller inherently generates an unsteady flow field, numerous studies have demonstrated that, for most practical design calculations, it is reasonable to approximate the flow as steady.

To further reduce the computational effort and expedite the optimization loop, an actuator disk model [5, 12, 17] can be used (as previously mentioned in Section 1) to simplify the propeller geometry by distributing its time-averaged forces over a defined disk, and thus also allowing for steady RANS-based simulations. Nevertheless, the actuator disk model can also be paired with unsteady RANS-based CFD [5]. In this work, we will be using this method together with steady RANS-based simulations.

There is also a distinction between models that have only implemented the propeller effects on the wing, which are denoted as single-interaction-mode (SIM), and more complex modes that also allow for the wing to impact the propeller performance, known as full-interaction-mode (FIM) [29]. In our case, the used method is SIM.

3.2 Actuator Disk Model

The actuator disk model represents the propeller mounted on this work's wing. The model simplifies the propeller's geometry to a distribution of time-averaged forces over a disk using source terms or boundary conditions [5]. The numerical formulas and their theory will be presented in this section. After coupling the aforementioned geometry with the wing explicit before, we get a configuration represented in Fig.3.4

As it has been mentioned previously, propeller forces are comprised of two types, axial and tangential forces. A typical simple method to model these distributions of axial and tangential loading on the propeller was developed by Hoekstra [65]. The description of this method is presented next.

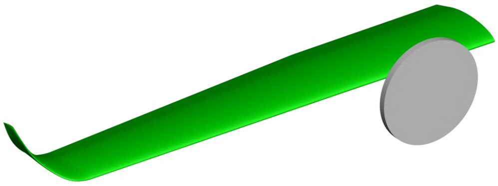


Figure 3.4: Wing surface mesh and the disk surface used to specify the actuator zone.

The radial distribution for the axial force is given by

$$f_x(r) = \hat{F}\hat{r}^m \left(\frac{a-\hat{r}}{a}\right)^n \quad , \tag{3.5}$$

where the parameters a, m, and n determine the shape of the force distribution, f_x is the axial force per unit radius, $\hat{F}$ is a reference value adjusted to obtain the required total force, and $\hat{r}$ is defined as

$$\hat{r} = \frac{r - r_{\text{in}}}{R - r_{\text{in}}} \quad , \tag{3.6}$$

where r is the radial distance from the axis of rotation, r_{in} is the spinner radius of the propeller and R is the outer radius of the propeller. The radial distribution for the tangential forces can be given by

$$f_\phi(r) = f_x \left(\frac{P/D}{\pi(r/R)}\right) \quad , \tag{3.7}$$

where f_ϕ is the tangential force per unit radius and P/D is the propeller pitch-to-diameter ratio. The calculation of this tangential force is very important for computing the swirl induced velocities which will also affect the lift and induced drag distributions on the wing [10, 11, 15], and it will also dictate the rotation direction of the rotor, which we have seen in Chapter 1 to have negligible influence on the wing for this study's purpose. Contrasting with Eq.(3.1), we can notice the simplification of blades to a disk, and how the time variable was averaged. By considering that the propeller is made of an infinite number of blades, both forces will depend only on the radial position. As such, the rotational speed, azimuthal position, and number of blades are no longer required. For simulations with different values of rotational speed using the actuator disk model, one would simply have to recalibrate the total thrust and torque to obtain congruent results.

The force per unit volume applied to the disk cells is calculated as

$$\frac{f_x}{2\pi r t} \quad , \tag{3.8}$$

where t is the total thickness of the disk of volume cells where the forces are applied. Chauhan et al. use this method and expressions in many of their studies [5, 12, 17], and in our work, they will be adopted with a slight modification to the radial force, f_ϕ. In this version, $f_{\phi_{\text{new}}}$ will be modeled independently of f_x as

$$f_{\phi_{\text{new}}}(r) = \hat{F}'\hat{r}^m \left(\frac{a-\hat{r}}{a}\right)^n \quad . \tag{3.9}$$

This means that $f_{\phi_{\text{new}}}$ will be defined in an equal manner to f_x, except $\hat{F}'$ now stands for a reference force adjusted to obtain the total torque. This modification was provided by Matos (private communication, Dec. 2024).

Propeller Inputs

This propeller model is defined to emulate the forces generated by the TEKEVER AR5's propeller, by using Eq.(3.5) and Eq.(3.9) and creating the resulting distribution plotted in Fig.3.5, which is then rotated asymmetrically across the given axis of thrust direction.

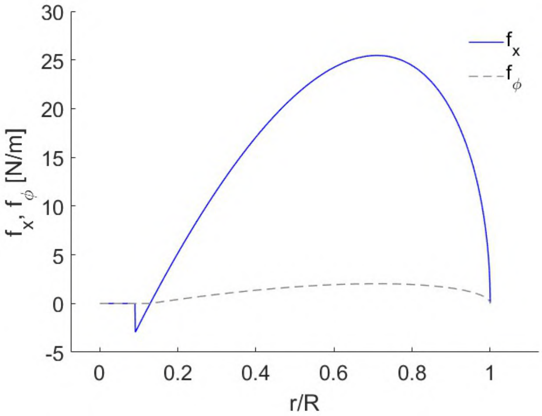


Figure 3.5: Resulting axial and tangential forces distribution.

Similarly to the real AR5 propeller, the generated disk has a radius of 0.3874 m, and a spinner radius of 0.05 m. In this study, the distribution parameters for Eqs.(3.5) and (3.9) were $a = 1$, $m = 1$, since this is an open duct propeller, as suggested by Hoekstra [65], and $n = 0.5$. The pitch-to-diameter ratio is not considered because it is not required to model the used tangential force distribution in Eq.(3.9), and the total torque generated is a known value. Additionally, due to the chosen orientation of the mesh of the wing, this work assumes that a positive f_ϕ will result in a rotation direction of inboard up. To obtain results with an outboard up rotation direction, we simply compute the opposite value of f_ϕ. The inputs for total propeller thrust and torque are 109.26 N and 8.70 N · m, and to obtain both distributions, the values of $\hat{F}$ and $\hat{F}'$ are computed internally in the code [5].

As seen in Fig.3.5, this distribution generates negative thrust between $r/R \approx 0.090$ and $r/R \approx 0.129$, which is deliberate, and mimics what occurs near the root of the propeller blade due to flow separation. This reversed thrust region was also seen in [19]: by using a full blade model and plotting the average radial distribution over the propeller blade, it was found that the blade's circular root section would cause flow separation and time-varying loading. We also consider that no force due to torque is generated in this interval, similarly to other studies [5, 12, 17].

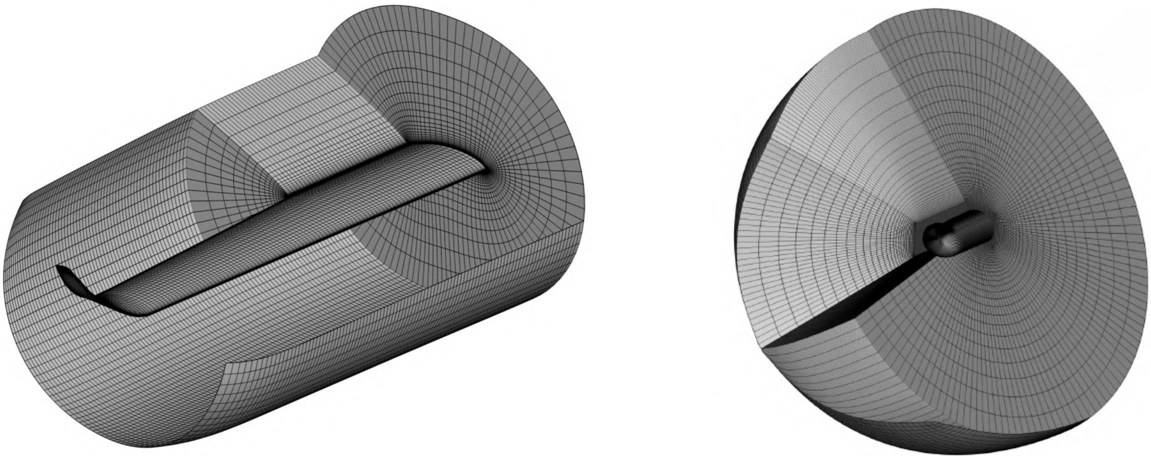
3.3 Overset Mesh Refinement Study

In this study, every mesh was created while taking into account the node equation,

$$N = 2^n m + 1 \quad , \tag{3.10}$$

as expressed in [66], which dictates how many nodes an edge of a surface mesh must have to perform a grid study correctly.

Regarding the volume mesh in this work, a similar approach to [5, 12, 17] was taken. A near field volume mesh for the wing was created, as shown in Fig.3.6a, and over-setted on the background mesh. This meshing strategy is required not only to be able to adequately define the actuator disk region, but also to support the subsequent integration of a nacelle overset mesh.



(a) Near field volume mesh of the wing.

(b) Background volume mesh combined with the near field volumetric mesh.

Figure 3.6: Near field mesh and combined mesh (medium refinement mesh).

Three meshes with different refinement levels were created: a fine, medium, and coarse refinement volume mesh. However, for simplicity, they will often be referred to as L1, L2, and L3 meshes, although they are not in mathematical terms. The mesh inputs of the fine overset mesh were chosen to generate a volume mesh similar to the mesh created in Section 2.2.1. For the near field mesh, there are 71 layers, with a farfield boundary distance of 1 m.

The background mesh was generated using the function *simpleOcart*, which creates a uniform quadrilateral mesh that encompasses the near field mesh (perceived better in Fig.3.6b) with a target edge length of 0.04 m for the quadrilateral cells. It was considered a number of 49 layers from the end of the near field mesh until the farfield boundary, and the latter at the previously established distance of 20 mean chords.

The combined mesh holds a total of $7\ 144\ 421$ compute cells. Using the overset mesh to simulate the same flow conditions in Section 2.3.2 outputs a value of 0.8929 for the total C_L of the wing, with a computation time of $27\ 217$ seconds. This corresponds to a relative error of 0.03% when compared to the prescribed value. However, it should be noted that for a real application, the mesh validation process would require a more extensive comparison to other experimental data, as illustrated in [5]. Additionally, conducting a mesh refinement study to analyze the discretization error is another important step, which is shown in Fig.3.7. The lower refinement grid levels were obtained by coarsening the surface mesh with a grid ratio of 2, and re-extruding the volume mesh with fewer layers and the same pyHyp parameters. Therefore, the L2 mesh has 49 near-field layers, holds $1\ 620\ 610$ computation cells, and the solution was obtained with 1872 seconds of computation time. The L3 mesh has 33 near-field layers, holds $493\ 975$ computation cells, and computation time was much faster, with 158 seconds. While this method of mesh refinement studies is not considered the mathematically correct method [66], it is still extensively used in this context and considered a valid option when no other options are available (Matos, private communication, April 2025).

To correctly form a Richardson extrapolation, we would need a L0 mesh and the meshes to have been coarsened uniformly [66]. As mentioned earlier, the coarsening method used in this study was not uniform. Additionally, the L0 mesh would be far too refined, resulting in a computationally too expensive task. Nevertheless, we can still apprehend the convergence behavior for this meshing strategy by considering the L1 C_D result as the correct one, since we are not able to determine the mathematically correct grid-independent estimate value of the drag coefficient [66, 67]. When comparing to other grid refinement studies, the results of Fig.3.7 show a typical behavior; the drag coefficient, C_D, tends to be overestimated with coarser meshes, and so does C_L, and the C_D curve resembles the behavior seen in [67]. The L2 mesh presents a C_L with a relative difference of 0.18% from the L1 mesh, and a C_D value with a relative difference of 4.41%. The relative difference of C_D being below the typically considered 5% threshold, and the large improvement in computation time, indicate that the L2 mesh might be the best option for qualitative analysis and optimization. The L3 mesh has a C_D relative difference of 25.19%, which is too high and therefore indicating that it is not adequate for our study.

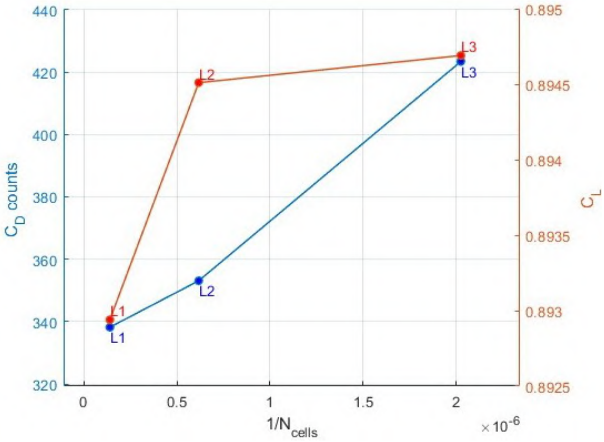
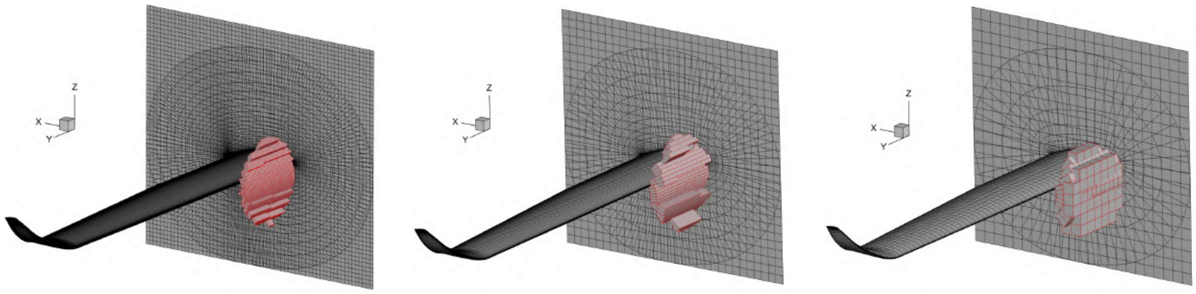


Figure 3.7: Convergence study of the overset meshes.

3.4 Actuator Disk Refinement Study

With the overset mesh correctly defined, we proceed to define the actuator disk on the mesh. In this work, instead of defining another volume mesh for the propeller and over-setting on the background mesh with the wing volume mesh, the cylindrical volume mesh for the propeller present in [5, 12, 17] was substituted by a propeller region, which is visible in Fig.3.8. Matos developed and provided this simplification, which is used for various analyses in TEKEVER (private communication, Dec. 2024).

The process to define the propeller region goes as follows: we generate the near field volumetric mesh of the wing in Fig.3.6a, combine it with the background mesh in Fig.3.6b, and afterwards, the actuator disk region is defined using the ADflow function *addActuatorRegion*, by selecting volume cells from the overset mesh, according to the parameters defined in Section 3.2. This results in an actuator region, as shown explicitly in Fig.3.8.



(a) Actuator disk region inside the L1 mesh.
(b) Actuator disk region inside the L2 mesh.
(c) Actuator disk region inside the L3 mesh.

Figure 3.8: Actuator region, delineated in red, inside the overset volumetric meshes.

One important aspect to notice in Fig.3.8 is the fact that each disk has a different thickness, t, and therefore, different force per unit volume, as seen in Eq.(3.8). Since we are using a different method for the propeller disk than the one used in [5], the propeller's refinement is coupled with the mesh refinement, which will affect the accuracy of the modeled propeller forces. This is because each refinement level of the overset mesh will affect the available cells that can be selected in *addActuatorRegion*; coarser levels will have larger cells in the propeller region, and thus coarser actuator disks (Fig.3.8c), and conversely, finer meshes will allow for more refined actuator disks (Fig.3.8a). Consequently, the disk thickness is limited in each mesh by the cell size, as setting the thickness to a value lower than the cell width in the propeller region would result in a poorly defined actuator disk.

However, to investigate how different actuator disk refinements affect the results, we recreate the scenario presented in [68] in ADflow and compare the experimental results (exactly as done in [5]) since this configuration is of similar size to the AR5. The propeller type was changed to the `simpleProp`, developed by Shamsheer in [5] (which uses Eq.3.7 for the radial distribution of the tangential forces instead of Eq.3.9) and all the ADflow inputs of the propeller were taken from the same article. We also recreated a surface mesh of the rectangular wing configuration of the NACA 64_2-A015 wing, ensuring a similar number of cells to the portion of our surface mesh located behind the propeller slipstream. To maintain meshes similar to the AR5 case, the same number of extrusion layers was initially used for each mesh level in Section 3.3. However, since this propeller is further away from the wing, we needed to increase the number of layers of the overset wing mesh. The resulting meshes are shown in Fig.3.10, and provide the loading distributions shown in Fig.3.9.

The L1 case has $5\,246\,484$ cells, the L2 has $1\,482\,406$, the L3 has $536\,196$. The results show that for

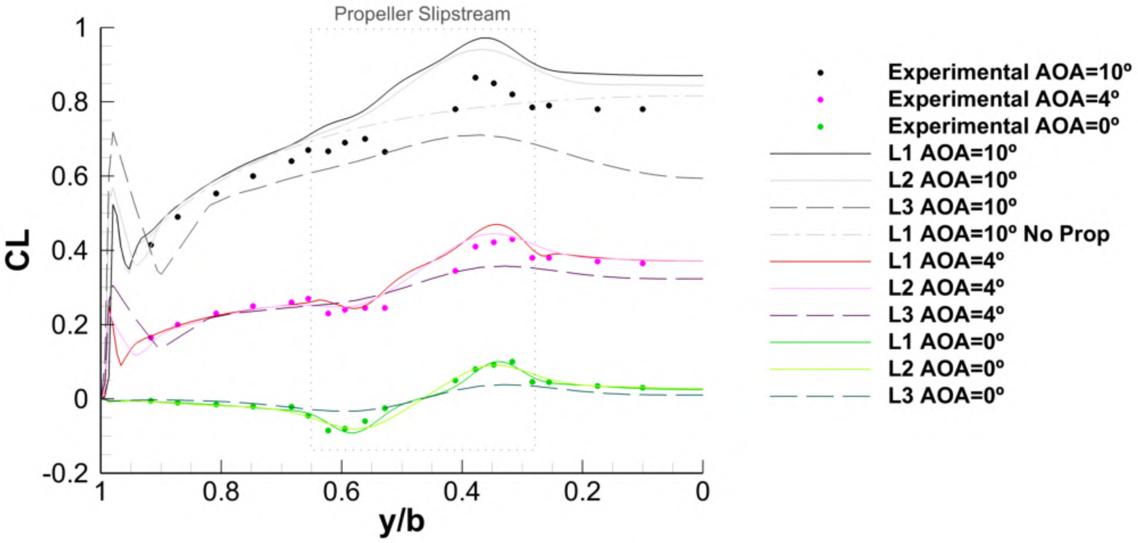
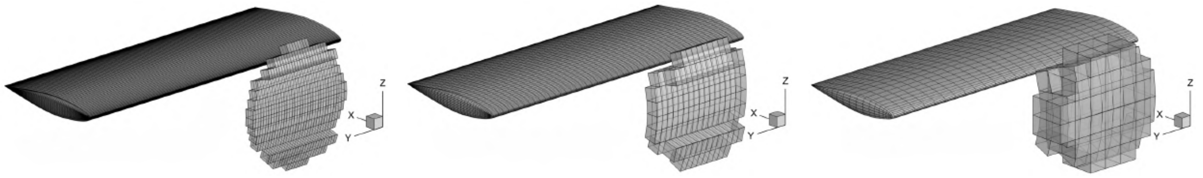


Figure 3.9: Actuator disk refinement levels.



(a) Wing and actuator disk with 372 cells and $t = 0.02$.
(b) Wing and actuator disk with 142 cells and $t = 0.04$.
(c) Wing and actuator disk with 112 cells and $t = 0.08$.

Figure 3.10: Different levels of actuator disk refinement for the NACA 64_2-A015 wing.

a zero angle of attack, the L1 and L2 meshes provide relatively good estimates for the forces acting on the wing, since the wing loading distribution is close to the experimental values obtained in [68]. The L3 actuator disk is too coarse to define the forces in Fig.3.5, and underestimates the lift acting on the wing (also seen in Section 3.3), therefore it is not adequate for our purposes. From Fig.3.9, we also conclude that coarser meshes will undermine our modeled propeller forces acting on the simulated wing. This occurrence is more notable for higher angles of attack (as seen when $\alpha = 10^\circ$ in Fig.3.9) since an overestimation of lift in the propeller region is expected, due to our actuator disk approach, as affirmed in [5].

The actuator disk is an axisymmetric propeller model that fails to account for the asymmetric loading that arises at non-zero angles of attack. In reality, the angle of attack introduces variations in the flow conditions experienced by each blade; the up-going blade encounters a lower effective angle of attack and reduced relative velocity (since a small blade velocity component is aligned with the freestream velocity). In contrast, the down-going blade faces higher values of both. This asymmetry leads to the up-going blade generating less axial and tangential force, and the down-going blade generating more at positive angles of attack [5]. Figure 3.11, shown in [29], aids the comprehension of this phenomenon.

Additionally, while in [5] it was determined through comparison with experimental values that an actuator disk with a zero angle of attack provided correct results for the wing loading, [29] also notes that the wing-induced vertical upwash can also effectively increase the propeller's angle of attack. However,

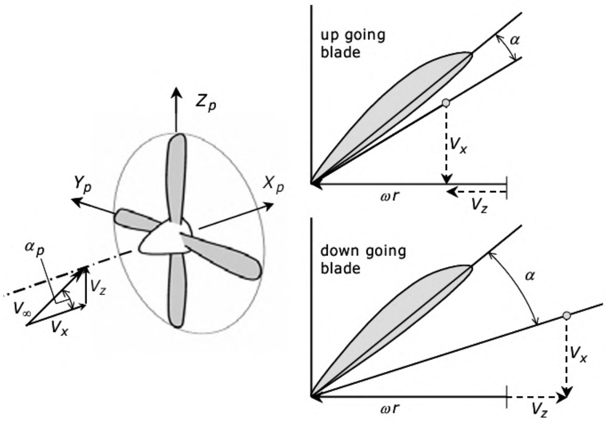


Figure 3.11: Different azimuthal propeller blade angles of attack for a configuration with positive angles of attack [29].

this effect is diminished if the tractor propeller is not closely coupled to the wing, as in the case of Fig.3.9, and [5] also considers a SIM propulsion model, so the wing-induced upwash acting on the actuator disk was ignored. Another source of errors in the lift distribution engulfed in the propeller slipstream is the fact that the experimental results were obtained with the presence of a streamline nacelle, while our modeled geometry possesses only the wing, as explained in [69].

From these results, we conclude that our method for the actuator disk is a viable option, as the results are similar to the experimental ones, and the actuator disk shows the same behavior and documented flaws as the approach used in [5]. We also note that to achieve results with acceptable accuracy, the angle of attack of the configuration should be close to $0°$, our level of refinement should be above the L2 level, and the actuator disk mesh should have more than 142 cells and minimal thickness.

With the previous statement in mind, we move on to the TEKEVER AR5 case. We use the same flow parameters of Section 2.3 and overset meshes studied in Section 3.3. For these same parameters, the new processing times of the L1, L2, and L3 meshes were $25\ 169$, $1\ 790$, and 163 seconds, respectively. The decrease in the solution time in the L1 and L2 cases is a counterintuitive result. Supposedly, the increased complexity of the flow around the wing (due to the induced swirl of the propeller) should increase the computational effort, and therefore the solution time. This matter is explored further in Appendix A, where it is shown that for lower tolerance values of convergence, simulations including the actuator disk do in fact require longer periods of time to compute.

Figure 3.12 represents the lift distribution of each refinement level of the overset mesh with an acting actuator disk, with the torque applied in an inboard up direction.

The incorrect propeller axisymmetry is hard to notice in Fig.3.12 simply by analyzing the distributions, because there is no experimental data for this particular wing and the cruise angle of attack is only $1.9°$. However, for the propeller pitch angle study in Section 3.5.3, this will hamper the accuracy of the results greatly for larger propeller angles. An explanation of how this will affect our optimization results is also expressed in Section 5.1.

The new C_L and C_D values for the L1 mesh were 0.9424 and 0.03845, respectively. Both values increased when compared to the isolated wing, and as we will see afterwards, this is expected. The L2

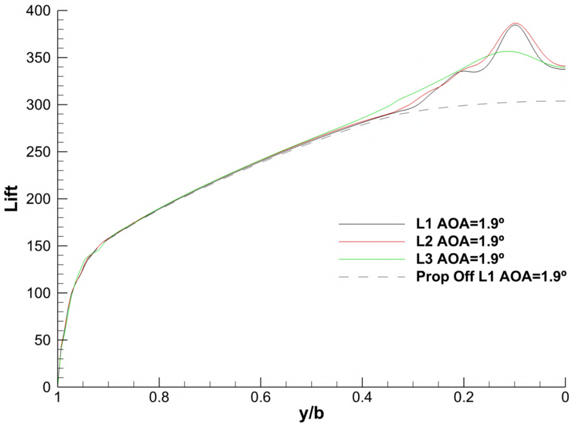


Figure 3.12: Lift distribution comparison between different mesh refinement levels.

mesh presents larger errors in this case, with a relative error of $+0.81\%$ for C_L and $+6.63\%$ for C_D. The L3 mesh exhibits a distinct lift distribution, underestimating lift in the upboard region and overestimating it in the downboard region. Similarly to the study in [5], the L3 can not correctly represent the spanwise variations created by the propeller's force. Once again, it also presents a high C_D relative error of $+23.51\%$, thus confirming that it is inadequate for the purposes of this study.

While the medium mesh is much more efficient in computational time than the fine mesh, the C_D error is slightly above the stipulated 5% margin for good accuracy. Presumably, we could improve this error by creating a new surface mesh that is more refined in the wing region directly behind the propeller slipstream. This strategy was explored, but the improvements were minimal, and the C_D error remained above 6%. Therefore, the mesh used for the analysis of the qualitative effects of the propeller on the AR5 wing was the L2 mesh, as it held the best ratio of computational time to result accuracy.

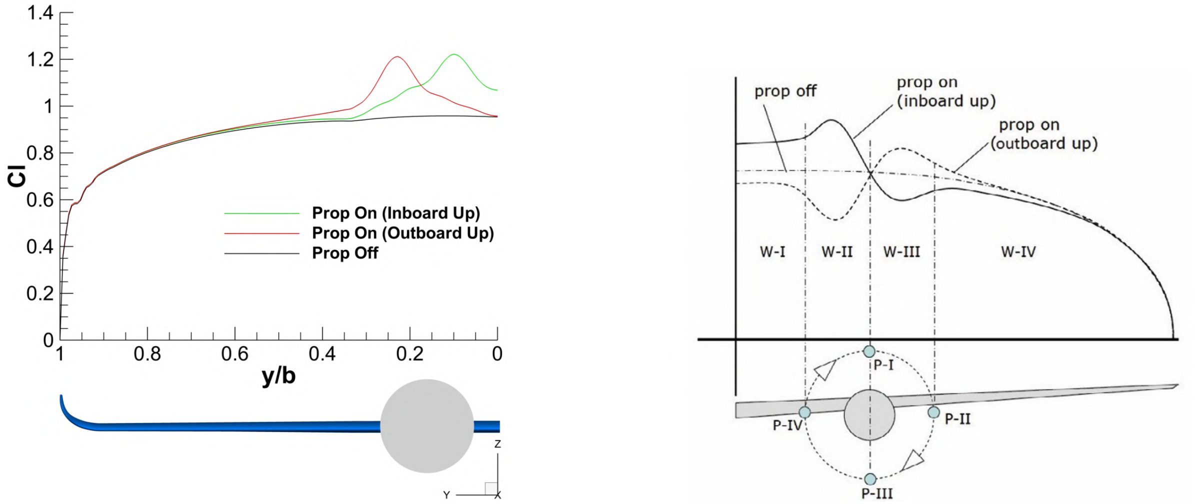
3.5 Impact of Propeller on the Wing

Now that the propeller has been correctly defined, we can interpret the resulting aerodynamic analysis. As stated previously, we will be using our L2 mesh for all the following analyses.

The most recognizable wing-propeller interaction effects are evidenced in wing loading distributions [16], which is why the validation for these methods also relied on them. Therefore, we will be analyzing several ones in the following sections. Figure 3.13a presents the lift distribution from the ADflow simulation coupled with the actuator disk, and the theoretical wing loading distribution [29] is presented in Fig.3.13b for comparison and qualitative verification.

Note that Fig.3.13a presents a reversed x axis direction because the simulated AR5 wing's side is different, and all wing loading graphs in this study have the leading edge facing the reader. This will also occur in the subsequent sections. In Fig.3.13, we see that the lift distributions for both propellers' direction of rotation obtained are congruent. We also concur visually that the direction of rotation is indeed a very important factor in the wing's lift distribution, as discussed in [9], and therefore, the induced velocities by the propeller cannot be neglected.

The regions defined by Veldhuis [29] are divided into different regions of influence of propeller slip-



(a) Comparison of the simulated lift distribution (L2 mesh) results.

(b) The different influence areas due to propeller wing interaction on the loading distributions [29].

Figure 3.13: Propeller effects on wing loading distributions.

stream, which cause a very significant change in the wing's performance, especially in regions W-II and W-III, since these are directly affected by the propeller wake. Focusing in the inboard up distribution of Fig.3.13b, the region W-II of the wing receives the propeller upwash and, as a result, it has increased dynamic pressure and the upward velocity increases the local angle of attack, which enhances the lift on that section of the wing [5, 11, 29]. In the region W-III, the wing receives the slipstream of the propeller downwash, causing the opposite, and as such, the lift on the wing is locally decreased. The remaining regions in Fig.3.13b, W-I and W-IV, are also changed due to the propeller altering the wing inflow conditions and disturbing the vorticity sheet leaving the wing [29].

We can see that the TEKEVER AR5 wing, coupled with the actuator disk in Fig.3.13a, has a slightly dissimilar loading distribution to the theoretical wing and that of similar studies [5]. Namely, the downwash region in both rotation directions does not have a lower lift when compared to the isolated wing, and for the inboard up rotation direction, the region W-IV of the AR5 wing is practically unaltered. The downwash effect fabricated by the actuator disk torque is much less accentuated than the upwash effect, and the supposedly neutral spinner zone, where the values of lift of all configurations should be the same, does not match. Subsequently, we present some possible reasons for this to occur.

The AR5 wing airfoil exhibits greater camber than other wings with a loading distribution closer to the one in Fig.3.13b, such as the wing used in [5] for the validation cases, which uses the previously stated NACA 64_2-A015, which is a symmetric airfoil. The greater curvature of the mean camber line might have the same effect as inclining the propeller angle of attack downwards, which raises the section lift in the spinner zone, as seen also in [15]. This notion will be studied further in Section 3.5.3.

Additionally, the AR5 propeller is smaller than other examples and has lower disk loading, as it is designed for a UAV. As discussed in Section 1.2, stronger propellers with higher disk solidity will also cause more noticeable distortions to the inflow of the wing. The low torque value used to simulate the actual AR5 propeller causes a weaker downwash, which is further countered by the higher local lift caused by the propeller-induced axial velocity. The simulations in Fig.3.14 were made to confirm this idea.

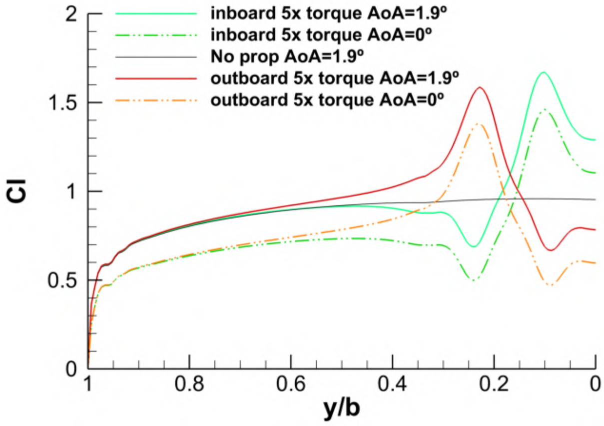


Figure 3.14: Simulated wing lift distribution with an actuator disk with increased torque.

As we can see, increasing the value of actuator disk torque, despite creating meaningless results, provides a distribution closer to the theoretical case. However, there is still an increment in lift in the propeller spinner zone, unlike in the Fig.3.13b. This offset can also be found in Fig.3.21c from [15], indicating that it occurs due to the propeller pitch angle, explored in Section 3.5.3. Since no propeller pitch angle was used in this section, the occurrence of this offset ties into the first presented reason that the camber of the airfoil could be causing this effect. Besides this, the increase in the lift in this region explains the increase in the value of the total wing C_L, and due to the aforementioned fact that the effective angle of attack increases, so will the pressure drag in this region, which explains the increase in total wing C_D. This can be visualized in the following section, in Fig.3.18a.

Additional remarks can be made. As mentioned in Section 1, Kroo [10] stated that the optimal mounted wing's lift distribution differs from the theoretical elliptic lift distribution of an isolated wing. While our mounted wing is not optimal, it is possible to understand qualitatively this reasoning by comparing the normalized lift distributions with a mounted propeller and the elliptical distribution in Fig.2.5a. The optimal solution of the lift loading for a wing with a mounted propeller will also contain "higher harmonics" [10] due to the propeller-induced velocities.

The ADflow solutions also allow for determining the variation in the pressure distribution coefficient, C_p, across the wing with a mounted propeller with an inboard up rotation direction, shown in Fig.3.15.

In Figure 3.15, both of the wings' leading edges are pointing outwards from each other, and the root of the wing is to the right. The difference was computed with the expression $\Delta C_p =$ wing with propeller $C_p -$ isolated wing C_p. Therefore, we can concur that the values of higher lift in the green line (inboard up) of Fig.2.5a will also correspond with the values of increased suction peaks shown in the Fig.3.15, with lower values of ΔC_p (in blue), and decreased suction peaks with higher values of ΔC_p (in red).

Additionally, to further understand how the actuator disk can affect the performance of the TEKEVER AR5 wing, various parameters regarding the positioning of the propeller were tested in the following sections. As previously mentioned, recent studies [12, 17] have shown that the propeller position in relation to the wing is of extreme importance when considering its effects on lift distribution over the wing. It is also important to note that our studies will be limited by using a SIM actuator disk model,

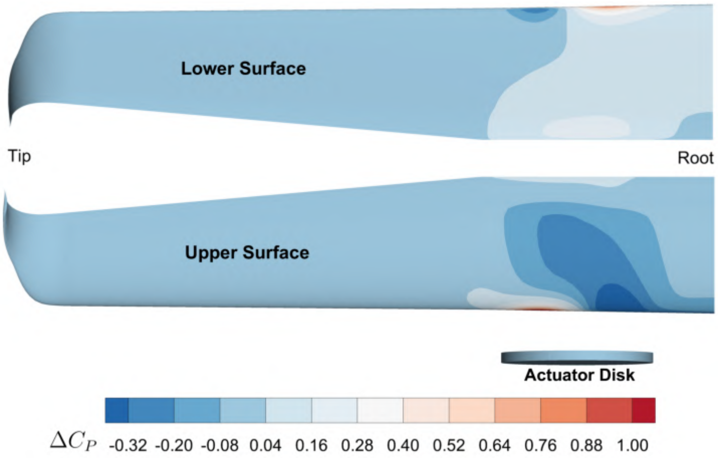
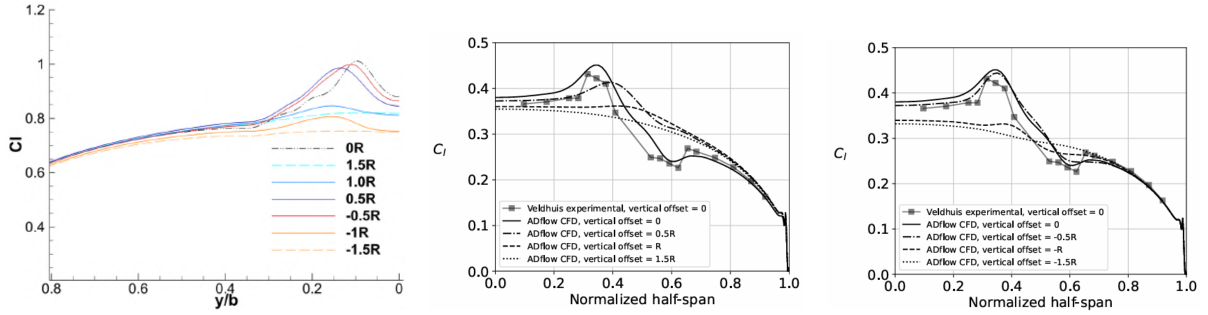


Figure 3.15: Variation in the pressure coefficient on the wing due to the propeller.

since the wing-induced effects on the propeller will change for each different propeller position [29].

3.5.1 Effect of Propeller Offset Distance

We can verify this effect by simulating various offset positions, meaning we elevate and lower the height of the AR5 propeller in relation to the wing. The results are shown in the Fig.3.16a, for an angle of attack of $0°$.



(a) TEKEVER AR5 C_l distribution for different propeller offsets.

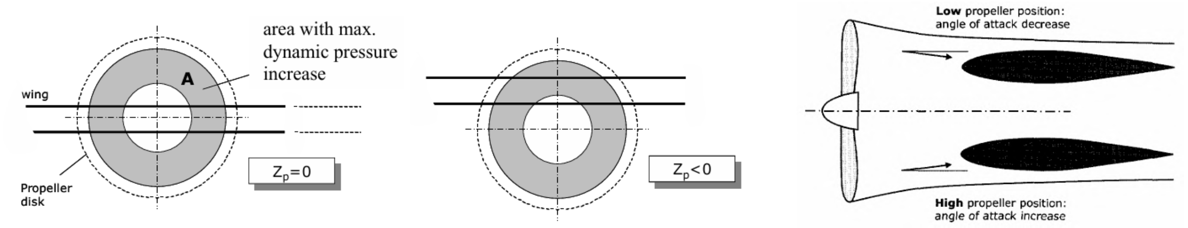
(b) Different positive offsets in the study [17].

(c) Different negative offsets in the study [17].

Figure 3.16: Effects of different propeller vertical offset positions on wing loading distributions.

The behavior of increasing or decreasing offsets in our case is, in fact, similar to that observed in other studies. Keeping in mind that in the case of Fig.3.16b and Fig.3.16c the simulations were made for the NACA 64_2-A015 rectangular wing with an angle of attack of $4°$ (and thus making the lift distribution changes more noticeable [17]) the comparison to our case can still be made to verify the results. As the offset increases, regardless of whether it is positive or negative, the distribution shrinks closer to the lift distribution without a propeller, as one would expect. However, increasing positive offset will shrink the distribution while still providing increased lift when compared to the isolated wing, whereas a negative offset will decrease it. Analyzing Fig.3.16, we see that this behavior occurs in both studies.

In [17], three factors are identified to explain this effect. The first factor is due to the varying extent of the wing immersed in the propeller slipstream, which changes the magnitude of the upwash and downwash components of the propeller swirl velocities. The latter factors were documented through experimental methods by Veldhuis in [68] and Prandtl [8]; the second factor is related to the propeller's slipstream contraction increasing or decreasing the effective angle of attack of the immersed region of the wing, depending upon the vertical offset being positive or negative, respectively, as shown in Fig.3.17b, and factor 3 is related to the varying extent of the wing immersed in the propeller slipstream area with maximum dynamic pressure, as shown in Fig.3.17a.



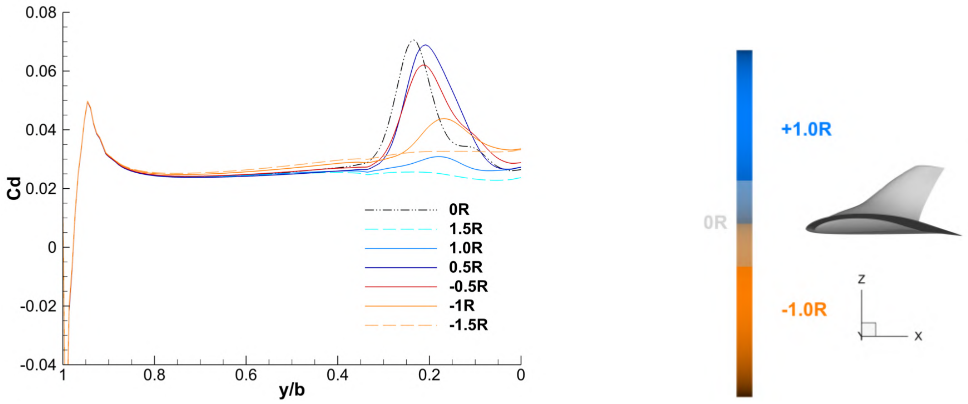
(a) Change in area of maximum dynamic pressure directly in front of the wing due to offset position of the propeller [29].

(b) Propeller slipstream contraction affecting the wing's effective angle of attack [68].

Figure 3.17: Other studies' images of the second and third factors related to propeller offset effects.

With Fig.3.17, it is also easier to understand why in [17] it is mentioned that depending upon the vertical position, some factors can dominate more than the others, which is evidenced by the differences in C_L, C_D, and pressure distributions along the wing.

For example, in our cases with a vertical offset of $1.0R$ and $-1.0R$ or more, only a reduced part of the wing is engulfed by the propeller slipstream (Fig.3.18b). Therefore, factor 2 will dominate, and as a result, the new distributions show only one lift peak, or also a lift depression for the negative cases. However, there is one discrepancy between our case and the results in Fig.3.16c. While in our Fig.3.16a we notice that the $-1.0R$ case presents only a lift peak, the result in Fig.3.16c shows a clear upwash and downwash zone in the lift distribution, meaning all three factors are at work. Once again, the results of Fig.3.16c used a higher angle of attack of 4°, and the matter relating to the camber of the airfoil remains.



(a) TEKEVER AR5 C_d distribution for different propeller offsets.

(b) Visualization of AR5 propeller $1.0R$ and $-1.0R$ offset positions.

Figure 3.18: Drag distribution and different propeller positions of the AR5 case.

Table 3.1: Wing aerodynamic performance for different vertical offsets.

Vertical Offset	$-1.5R$	$-1.0R$	$-0.5R$	$0R$ (original)	$0.5R$	$1.0R$	$1.5R$
C_L	0.7025	0.7185	0.7708	0.7672	0.7737	0.7435	0.7392
C_D	0.0297	0.0307	0.0323	0.0321	0.0324	0.0268	0.0253
L/D	23.65	23.40	23.86	23.90	23.88	27.74	29.21

While the results in Tab.3.1 indicate that the best performance is achieved when the propeller is placed at higher offsets, namely for $1.0R$ and $1.5R$, in [29], it is noted that a nacelle with this height in relation to the wing might bring structural difficulties to realize, and the new nacelle-wing aerodynamic interference might be detrimental to the propeller positioning gain. The effects of propeller vertical offset bring noticeable changes to the forces acting on the wing, but taking advantage of them requires consideration of other structural and aerodynamic factors.

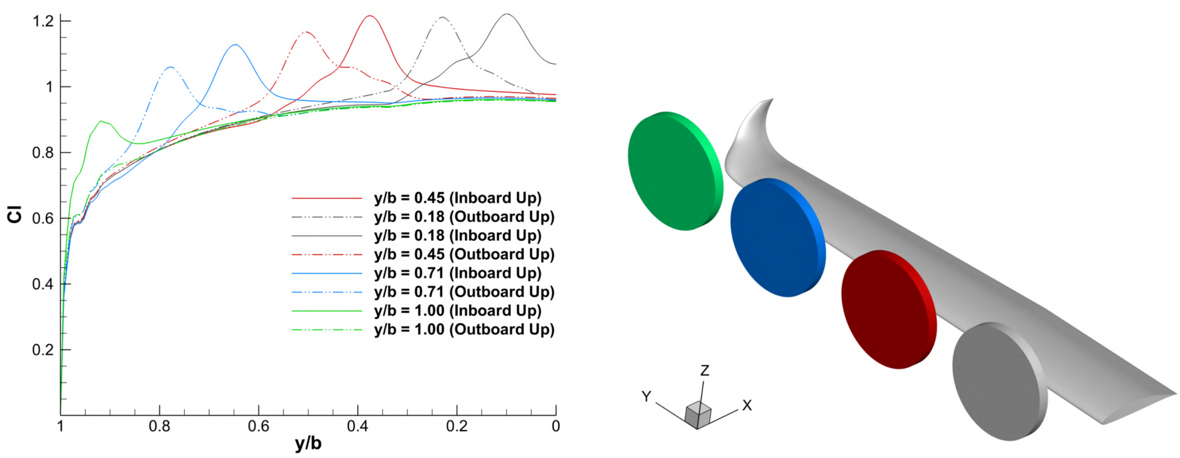
3.5.2 Effect of Propeller Spanwise Position

Continuing the positioning study of the propeller, we see that the spanwise position is also of great importance for the wing loading distribution. As addressed before in Section 1.2, earlier work [9] showed that wing-tip mounted propellers can significantly manipulate wing-tip vortices depending upon the rotation direction. With an inboard-up rotation direction, their induced swirl opposes these wing-tip vortices, weakening and shifting the wing-tip vortex core outboard and downward, thus reducing the downwash behind the wing-tip. Conversely, with an outboard-up rotation direction, the effects are the opposite; the vortex core moves inboard, and the induced drag is increased.

The tests in [9] showed that the effects from the inboard-up rotation on the propeller reduced induced drag and simultaneously increased the L/D ratio when compared with the wing without the propeller and with a propeller with an outboard-up rotation, proving it to be the best configuration for aerodynamic performance. This is the generally accepted theory. However, in more recent work [70], this idea is challenged by adopting a methodology that uses the optimum propeller power required for level-flight (achieved using QMIL [71] and an iterative trim process) in every considered propeller location, and only then comparing the results. Therefore, since this method maximizes aerodynamic and power efficiency of each case, it also allows for more appropriate metrics for assessing actual improvements for each location and rotation direction of the propeller. The conclusions showed that when applying this method to a test case of an X-57 configuration for cruise conditions, the most efficient solution is located approximately at 30% of the span of the wing, with an outboard up rotation direction, contradicting the established theory, and proving it to be case conditional.

Since our study focuses solely on the effects of different spanwise propeller positions on wing loading, lift, and drag, the optimum propeller methodology used in the previous study is not used. This means that the modeled propeller is the same for all positions, similarly to Section 3.5.1. The considered positions of the propeller are shown in Fig.3.19b. These were tested for both possible rotation directions.

Figure 3.19 shows that changing the rotation direction of the propeller and its position causes a



(a) Lift distributions for different propeller spanwise positions.

(b) The different propeller spanwise positions.

Figure 3.19: Lift distributions and the respective propeller spanwise positions.

shift of the lift peak of the Tekever AR5 wing's loading, according to the up-going blade's change of position (or more correctly, in our case, the change of position of the upwash region). The wing loading is considerably changed, but to analyze the total effect on the aerodynamic performance of the wing, we compare the results in Tab.3.2.

Table 3.2: Wing aerodynamic performance for different propeller spanwise locations.

	Rotation Direction	$y/b = 0.18$ (Original)	$y/b = 0.45$	$y/b = 0.71$	$y/b = 1.0$
C_L	Inboard Up	0.9501	0.9487	0.9330	0.9195
C_L	Outboard Up	0.9456	0.9482	0.9278	0.9003
C_D	Inboard Up	0.0410	0.0398	0.0378	0.0350
C_D	Outboard Up	0.0407	0.0411	0.0391	0.0370
L/D ratio	Inboard Up	23.17	23.83	24.68	26.27
L/D ratio	Outboard Up	23.23	23.07	23.72	24.33

Throughout all the positions, C_D changed mainly due to the pressure drag, C_{D_p}, since the viscous drag component remained approximately the same value of 0.01. As established before, the outboard up direction is noticeably less favorable than the inboard up direction, regardless of the spanwise position, most notably when positioned at the wingtip, as seen in [12]. These results are in accordance with the ones seen in theory [9], since the value of C_{D_p} of the isolated wing has the lowest C_D of all configurations and the best L/D ratio, showing that the inboard up propeller at the wing tip improved aerodynamic performance, and that the outboard up rotating propeller aggravated the wing tip vortex, thus increasing pressure drag and worsening the L/D ratio.

As Kroo stated in [10], the spanwise position of the propeller ”plays a small role in wing-propeller interaction” unless it is positioned outboard near the wingtip. Indeed, for the first three positions in Tab.3.2 we see negligible changes in the L/D ratio of the wing, concluding that within these spanwise positions, the interactions between the propeller and the wing remain largely unaffected, as also seen in Fig.3.19a. It is only in position $y/b = 1.0$, at the wingtip, that noticeable changes occur to the wing's performance. As described in another study [72], the inboard-up rotating propeller provides the largest reductions in total induced drag because the wing is immersed only in the beneficial up-going side, which is also evidenced in Fig.3.19, which only shows the lift loading increase on the wingtip, as the down-going region does not interact directly with the wing. The reason why the up-going region decreases induced drag on the wing will be explained in Section 3.5.3. It is important to note that the wingtip case is purely theoretical, since it would be more structurally complicated to have the propeller mounted on the winglet, and these considerations are out-of-scope for the purposes of this work.

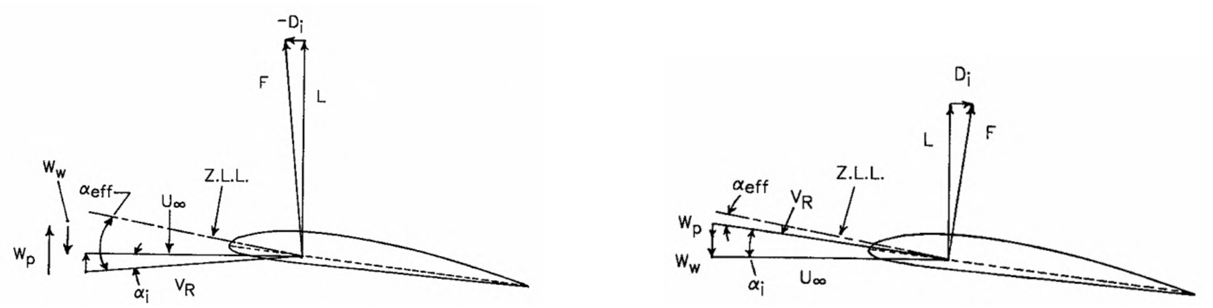
However, we must take into consideration the findings in [70] and recall that the spanwise positions of $y/b = 0.45$ and $y/b = 0.71$ could provide larger changes to the wing's aerodynamic performance if the propeller were optimized for each position, and that if the power required for cruise in these locations were lower, the induced drag from the propeller could also be lower and provide better L/D ratios.

3.5.3 Effect of Propeller Pitch Angle

Another parameter discussed in Section 1.2 is the propeller pitch angle, or, as described in [68], propeller angle of attack, α_p. The propeller pitch angle can also alter the performance of the mounted wing, and it has been established that a PTD configuration with a relatively large angle of attack can be beneficial, since it reduces the overall induced drag more effectively than the direct forces acting on it reduce lift and thrust [15, 29]. For this Section 3.5.3, only an inboard up rotation direction was considered since it is also stated in [15] that a PTD configuration achieves better performance in this scenario. Another important thing to consider for the simulation is the definition of propeller pitch angle, α_p, and propeller geometric angle, $\alpha'_p = \alpha_p + \alpha_c$ (with α_c being the configuration angle of attack, or the freestream velocity direction, given as an input to ADflow), as defined in [15]. In this case, since cruise conditions were considered, the propeller geometric angles of the AR5 in this section are defined by $\alpha'_p = \alpha_p + 1.9°$.

Similarly to the angle of attack of the propeller discussed in Section 3.5, when the propeller has a positive pitch, $\alpha_p > 0$ (as in Fig.3.21a), the down-going blade region experiences the same increased local angle of attack, generating higher local thurst and torque, while the up-going blade sees a reduction in its local angle of attack, producing less thrust and torque [15, 29]. For a negative pitch, $\alpha_p < 0$, the opposite occurs, meaning the up-going blade will generate more thrust and torque, and a down-going blade less of the same. The reason why a propeller-pitch-down configuration could be beneficial can be understood by analyzing Fig.3.20.

As we have seen in Section 3.5, the propeller's upwash increases section lift, but simultaneously, it also rotates the resulting force vector forward, which reduces the induced drag component, thus generating a "localized wing thrust" [11] in this region. Note that this is only possible when the propeller upwash exceeds the wing's downwash (created due to the finite wing effects), otherwise, the resulting force would remain tilted backward. However, while on the upgoing blade side in Fig.3.20a, the section lift will increase and the propeller upwash will tilt the force vector forward, in the downgoing blade side in



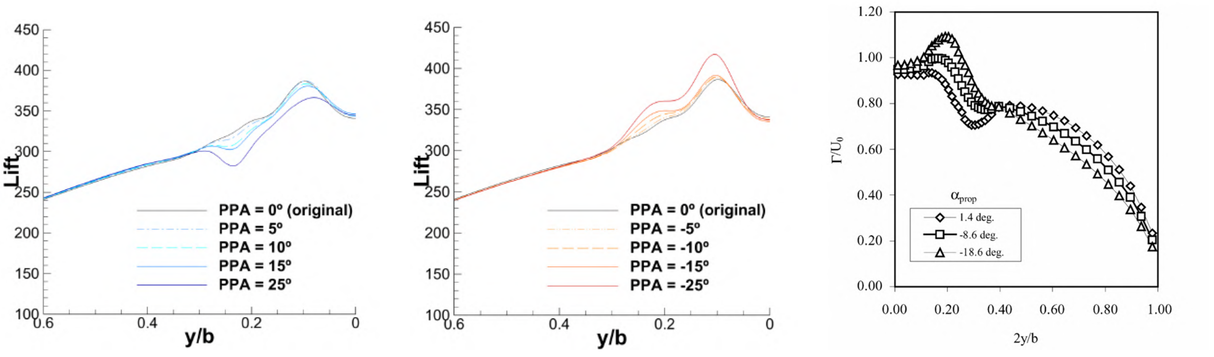
(a) Wing section in propeller upwash (b) Wing section in propeller downwash

Figure 3.20: Theoretical forces on wing sections directly behind the propeller upwash [11].

Fig.3.20b we see the opposite happening, lift will decrease and the force direction will be tilted backward, since the propeller downwash will act together wing the wing's downwash, thus increasing the induced drag in this section. Nevertheless, the net result tends to be favorable for most operating conditions, including a propeller with no pitch [11], as is the case shown in Fig.3.20.

However, decreasing the pitch angle ($\alpha_p < 0$) will amplify this effect, as found in [15], since, by inclining the propeller downwards, we are further increasing the propeller's upwash on the upgoing blade region, decreasing the induced drag component of the wing directly behind this region even more (Fig.3.20a). Coincidentally, the downgoing region will also reduce its downwash due to the reduction of the blade angle of attack, and the net reduction of induced drag will increase. Once again, this imbalance creates a non-axisymmetric propeller slipstream, which becomes more accentuated with increasing absolute values of the propeller angle of attack.

As we have seen, the actuator disk fails to capture this, since the thrust and torque distribution added to the flow are always axisymmetric, regardless of the α'_p. However, we can still draw conclusions from the various simulated cases in Fig.3.21a and Fig.3.21b, despite being computed with an actuator disk. Figure 3.21 shows the impact on the wing loading of many different propeller angle of attack configurations.



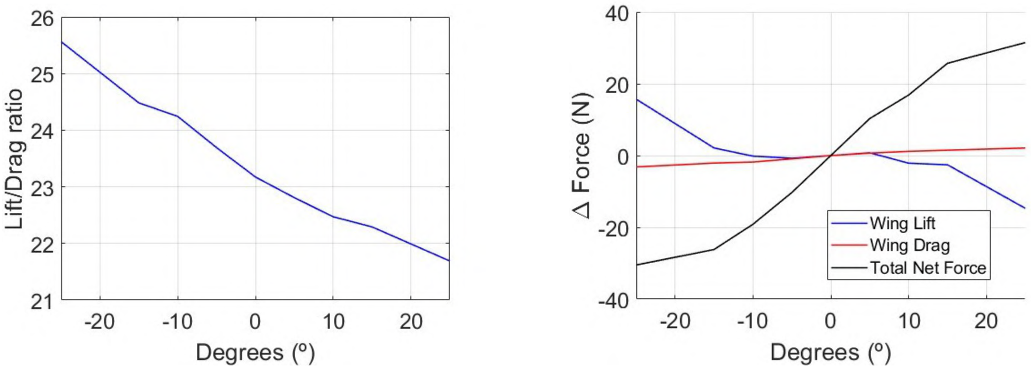
(a) AR5 wing loading for positive (pointing upwards) propeller angles of attack, α_p. (b) AR5 wing loading for negative (pointing downwards) propeller angles of attack, α_p. (c) Fokker F50 wing loading for different propeller angles of attack [15].

Figure 3.21: Propeller angle of attack effects on wing loading distributions.

The effect of the PTD configuration will also result in an increase of the effective angle of attack of the wing directly behind the propeller's slipstream, due to increasing the leading edge suction in this region. Since this is due to the propeller slipstream being placed at an angle, this effect can also be captured with our actuator disk, even if the propeller forces are inaccurate. This explains why the lift peak increases in all propeller regions Fig.3.21a, and decreases in Fig.3.21b, in a similar manner to Fig.3.21c.

There are also some notable similarities between the resulting wing loadings of the AR5 wing and the Fokker F50 wing from Fig.3.21c [15]. With lower pitch angles of the propeller in Fig.3.21c, we see the downgoing blade region getting higher lift, starting to resemble the lift distribution we have for the AR5, where only a lift increment in the propeller region is noted. On the other hand, analyzing Fig.3.21b and the positive value in Fig.3.21c, we see that increasing the propeller pitch will lower the lift behind the propeller region, since we are lowering the effective angle of attack of the wing. For our case in Fig.3.21b, this will result in a lift distribution akin to the theoretical one in Fig.3.13b. This ties into the aforementioned idea that the highly cambered airfoil of the AR5 might be the reason why the lift distribution behind the propeller region has only lift increase, and the typical downwash region is not visible. We can also note a difference between the region of the wing out of the propeller's direct wake, region W-IV, in Fig.3.13b. In Fig.3.21c, the entire span of the wing is affected by the stronger propeller, with the negative propeller pitch angles causing a noticeable decrease of lift in region W-IV, and positive angles causing an increase. In the case of the AR5 in Fig.3.21a and Fig.3.21b, we notice once again that this region is negligibly affected, so much so that only the close-up on the region directly behind the propeller shown in Fig.3.21 was affected.

Finally, we can also compare the lift and drag improvements of the baseline $\alpha_p = 0°$ case with the other cases presented in Fig.3.22.



(a) Lift drag ratio change.

(b) Lift and drag improvements.

Figure 3.22: Propeller pitch angle effects on the wing forces.

The numerical results concerning the forces acting on the wing, ΔLift and ΔDrag, were determined by subtracting the value of a configuration with propeller pitch and a configuration without pitch, $\alpha_p = 0$. To determine the overall net force result in the lift direction, we add ΔLift and the corresponding component of the propeller thrust (given in Section 3.2), $T_l = T\ \sin(\alpha_p)$. This result is presented as total net force in Fig.3.21.

Since the propeller thrust component is pointing down in a PTD configuration, we can obtain a total force that results in a loss of lift. However, this goes against the results in [15] and [29]. In fact, as stated in [15], improvements are only noticeable for relatively large propeller angles of attack, supposedly lower than $-10°$. In [29], it is concluded that generally, it is beneficial until $\alpha_p < -15°$. However, in our case, the total net force result seems more detrimental than the lift and drag ratio increase in Fig.3.22a, which is small even within these values.

However, once again, the main reason for these results could be that the inaccuracy of the actuator disk forces increases for larger propeller pitch angles. The only way to ascertain this would be to change the actuator disk's force distribution to the asymmetric force distribution occurring with respect to the inflow angle of attack. Such a method was utilized in [29], in which a higher fidelity version of the actuator disk was used, along with higher fidelity RANS turbulence models, both the Realizable $k-\epsilon$ and the Reynolds Stress Model with non-equilibrium wall functions, as explained in [73]. Their results obtained good compliance with experimental results, including a case with an angle of attack of $4°$.

Concluding, we have strong evidence to affirm that the effect of the PTD configuration is hindered by the limitations of our current actuator disk model, as aforementioned in Section 3.5. Supposedly, the effects stated in this section on the propeller blades would amplify the drag reductions and reduce the downwards thrust component, possibly improving the results found in Fig.3.21.

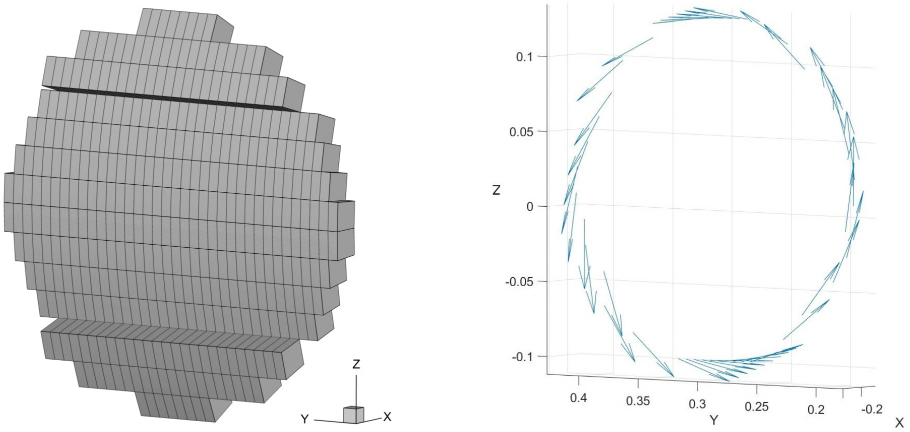
3.6 Modified Actuator Disk Force Distribution

The asymmetry of propeller loading has been implemented in the previous study [29], also using RANS simulations and an actuator disk (albeit with different turbulence solvers), with good accordance to experimental results. In [69], it has been suggested that a coupled actuator-disk model be developed, which computes its force distributions using the inflow conditions from the CFD simulation, as this would be useful for optimizing UAM in large angle-of-attack conditions. A similar solution was applied in [74], where it was shown that an axisymmetric actuator disk can be improved by imposing variations in the propeller blade sectional lift and drag coefficients in relation to its local effective flow angle. This was denoted as the ”enhanced actuator disk”, which addresses the inaccuracies described previously and provides a steady-state solution in good agreement with a full-blade URANS solution (in a specific orientation). In this section, we attempt to experiment with this conclusion within the context of ADflow.

While it may be imprecise, we can modify the way the actuator disk force distribution is defined in the `actuatorregion.F90` file by implementing an azimuthal change and a reduction in the magnitude of thrust and torque on the cells that define the actuator disk, with respect to the propeller angle of attack. Such a change would not couple the actuator disk model to the inflow conditions in the sense described in [74], since it would only depend upon the global propeller angle of attack, instead of the local angle of attack, as done in the aforementioned study. This means the actuator disk remains a SIM model, because the effect of a wing or a nacelle (that changes the inflow conditions of the propeller) would still not affect the disks' performance. Regardless, it would allow, for example, for more accurate results of a propeller pitch angle study, as done in Section 3.5.3, or an analysis of the propeller and wing in climb flight condition.

To modify the actuator disk's force distribution correctly, we first need to understand how the forces should change according to the azimuthal angle when the propeller angle is non-zero. Recalling from [5], the forces change because the relative velocity of the blades in the flow changes when the propeller is tilted. We can obtain the direction of the velocity vector of the blades by printing the tangential velocity of the selected cells of the actuator disk (since these are calculated internally in `actuatorregion.f90`).

Consider only the propeller of the experimental scenario used for validation in Section 3.4 with $\alpha_p = 4°$. For ease of vector calculations, we set the angle of attack to $0°$ and impose the α_p geometrically by rotating the propeller's geometry on the mesh. If we then compute the L1 actuator disk with an inboard up rotation direction and store the forces acting on the selected cells of the mesh, we obtain Fig.3.23.



(a) Actuator disk of the L1 mesh with $\alpha_p = 4$.

(b) Tangential forces acting on the outermost cells of the actuator disk to simulate the propellers' vorticity.

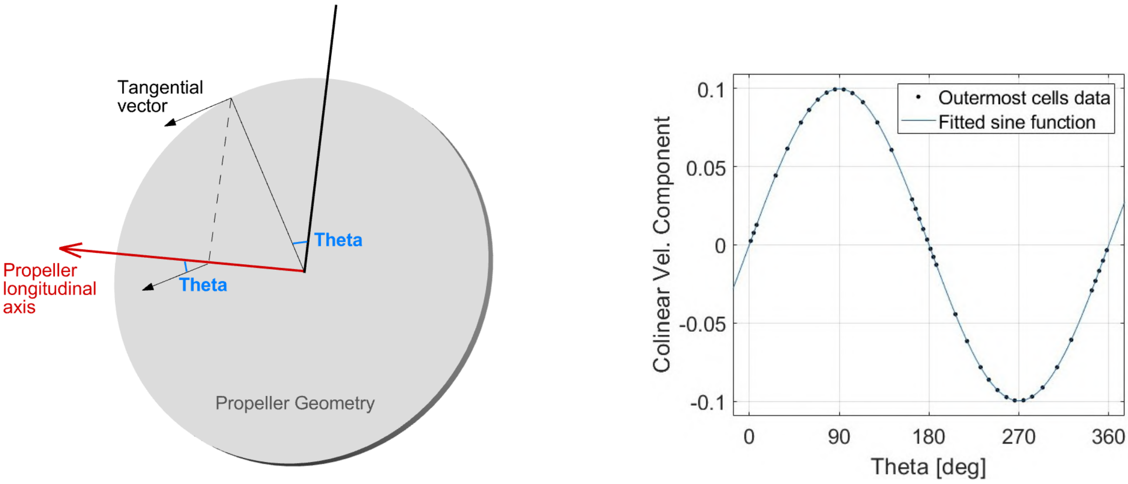
Figure 3.23: Actuator disk cells and tangential velocity direction for an inboard up rotating case.

The tangential forces shown in Fig.3.23b are only for the outermost cells in the radial direction. Now we can compute the azimuthal position of each cell. This is achieved by calculating the angle between two known vectors of the actuator disk: the vector of the tangential velocity direction on the current cell, and the vector of the longitudinal axis of the propeller, which is provided by the user. For example, the vector of the longitudinal axis of the propeller in Fig.3.23a is $[0, 1, 0]$ or $[0, -1, 0]$. Having two vectors, the angle between them is calculated with the equation,

$$\Theta = \arccos(\texttt{Longitudinal\_Vector} \times \texttt{Tangential\_Vector}) \quad , \tag{3.11}$$

providing the azimuthal angle for each cell in `actuatorregion.F90`, as shown in the beginning in Appendix B. The angle between these two vectors will correspond to the azimuthal angle of the cell's center location, as shown in Fig.3.24a.

Given both the vector component and the azimuthal angle of each cell, we can plot them and observe how the tangential vector (which is also the blade's velocity vector) component, which is collinear with the freestream velocity, behaves along the azimuthal angle on the disk. Recalling the fact that the propeller's geometry is tilted and the freestream flow velocity is parallel to the x axis, by analyzing Fig.3.23b, we see that the collinear component of the tangential vector is solely the vector component on the x axis. The result is shown in the following Fig.3.24b, which might indicate that the real time-averaged force distribution of the propeller will change sinusoidally along the azimuthal angle.



(a) Visualization of how the azimuthal angle (Theta, Θ) calculation was determined.

(b) Variation of the magnitude of the component of the blade vector that is colinear with the freestream flow direction.

Figure 3.24: Behavior of the tangential vector along the azimuthal angle.

Considering that it is now known how the relative velocity of the blades varies when the propeller is tilted in relation to the flow, the distribution of the force can be altered accordingly. Therefore, in the cells' force calculation line of code, we multiply the thrust variable by $(1+\texttt{angledpropCorr}\times\texttt{SineProp})$ if they are on the downgoing side, and $(1-\texttt{angledpropCorr}\times\texttt{SineProp})$ on the upgoing side of the actuator disk, for $\alpha'_p > 0$, and vice-versa for $\alpha'_p < 0$. For the torque variable, we also multiply `angledpropCorr` by a `torquefactor` because thrust and torque do not have the same relation to the inflow angle of attack (according to blade element momentum theory [75]). In our validation, we found that our results were closer to the experimental data when `torquefactor` was within the interval $(0, 1)$. The variable `SineProp` is the sine of the azimuthal angle, Θ, of the current cell, which means that the forces will vary exactly as depicted in Fig.3.24b, and the `angledpropCorr` is a variable whose magnitude depends upon the angle of attack, and can be fine-tuned by the user. Its expression is given in Appendix B. When $\alpha'_p = 0°$, the `angledpropCorr` is zero, and therefore no changes are employed to the distribution. Similarly, the cells in positions P-I and P-III (in Fig.3.13b) will have `SineProp` equal to zero because the blades in this position are always perpendicular to the flow direction, thus remaining unaffected. Applying this modification to the `simpleProp` propeller type in the `actuatorregion.F90` file will provide the following new results, shown in Fig.3.25.

Despite some imperfections, the new distributions show better accordance with the experimental results with non-zero angles of attack than the previous ones in Section 3.4. The overestimation of lift on the upgoing side of the disk has been diminished, with the $\alpha = 4°$ being the most accurate. Some discrepancies remain, which are suspected to be due to the absence of a mounted nacelle on the wing. As we will see in the subsequent Section 3.7.3, the presence of the nacelle will cause lift losses on the propeller region (shown in Fig.3.34), and for larger angles of attack it will greatly diminish lift due to the occurrence of easier flow separation, which explains why our lift distribution is superior across the all the span of the wing for the $\alpha = 10°$ case. Additionally, we reiterate that our actuator disk is not as

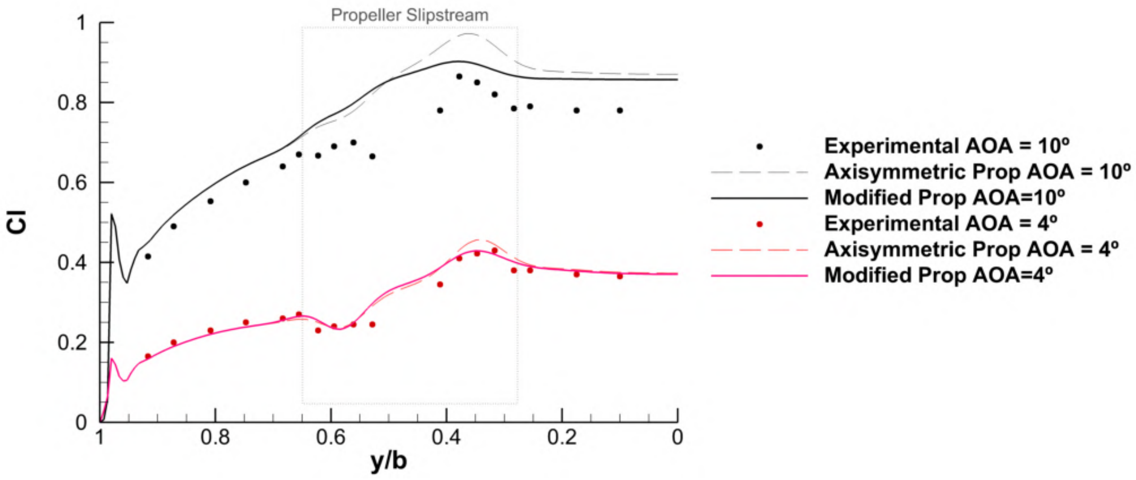
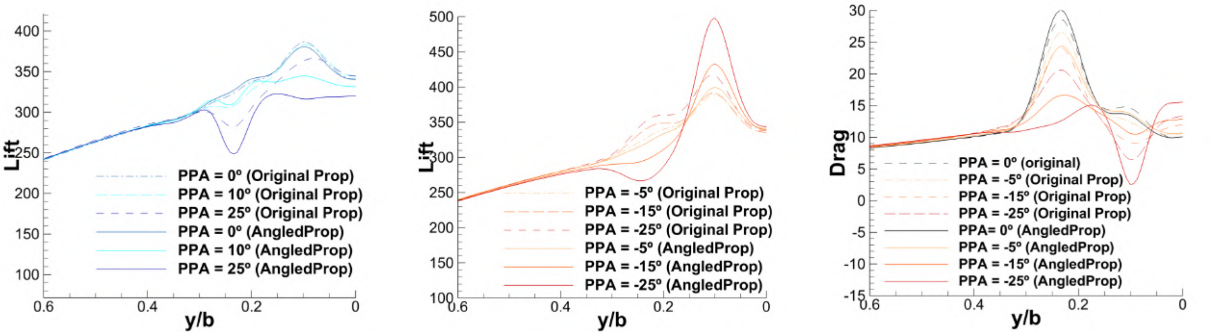


Figure 3.25: Modified actuator disk C_L distribution comparision.

refined as those in other studies [5], which will also impact the accuracy of the results. However, a more thorough study would require additional validations, such as quantitative wake surveys, as done in [29], which would provide needed insight into the validity of this modification.

3.6.1 Effect of Propeller Pitch Angle with Modified Actuator Disk

Now that we have modified the actuator disk and verified it, we can apply it to the AR5 case again to analyze a more correct representation of the effect of the propeller pitch angle. To achieve this, we also modify the code of the propeller type used for the AR5 (which utilizes Eq.3.9). Since we lack experimental data for the AR5's propeller, we use the same values of `angledpropCorr` and `torquefactor` that were determined iteratively to obtain results in Section 3.6. The new results are presented in Fig.3.26.



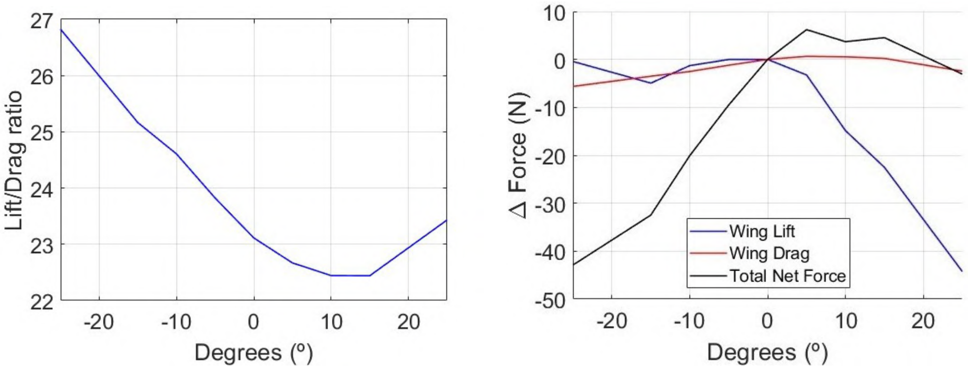
(a) AR5 lift distribution for positive propeller angles of attack.

(b) AR5 lift distribution for negative propeller angles of attack.

(c) AR5 drag distribution for negative propeller angles of attack.

Figure 3.26: Comparison of propeller angle of attack effects on the wing loading in relation to the actuator disk version.

Obviously, once again, the discrepancy between the distributions of each propeller type is more pronounced for the cases with higher α_p, as expected. We also notice in Fig.3.26a that for our original case (PPA $= 0^\circ$, thus $\alpha'_p = 1.9^\circ$), the lift and drag distributions are very similar, confirming the fact that for our baseline cruise conditions, the asymmetry of our original actuator disk does not cause very considerable accuracy errors, because α_p' is small. The lift distributions obtained with `AngledProp` in Fig.3.26a and Fig.3.26b present a much more similar behavior to those in Fig.3.21c, because our original propeller only increased the local angle of attack, whereas the modified actuator disk also introduced the force distribution asymmetry that benefits the performance for a PTD configuration. For positive propeller pitch angles, these will reduce the beneficial upwash on the upgoing side, because the relative velocity of the blades to the inflow is reduced, and on the downgoing side, the opposite will occur, the forces there become stronger, and the downwash is increased. Conversely, the negative propeller pitch angle should provide clear performance improvements. Despite it not improving the overall lift, the drag distributions in Fig.3.26c show the induced drag reductions described in [15] and previously in Section 3.5.3, since, analyzing Fig.3.26b, the increase in lift on the upgoing side will reduce the drag in the same wing sections in Fig.3.26c. Additionally, we calculate once again the lift and drag improvements, as done in Section 3.5.3, but for the values obtained with the modified actuator disk.



(a) Lift Drag ratio change.

(b) Lift and drag improvements.

Figure 3.27: Propeller pitch angle effects on the wing forces with the modified actuator disk.

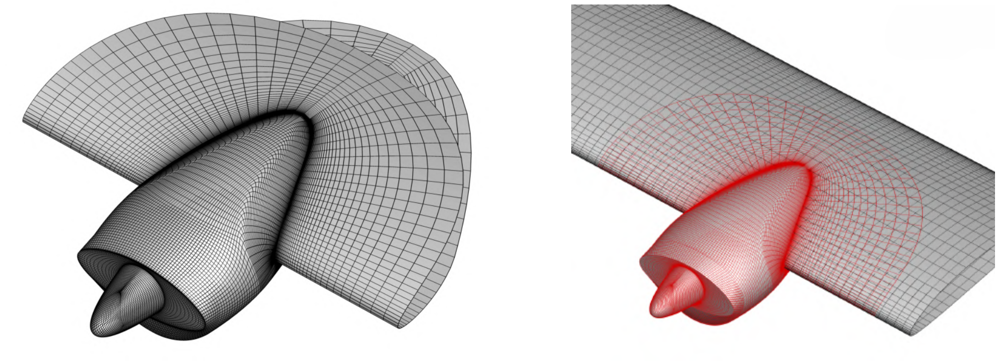
From Fig.3.27, we conclude that even with more accurate depictions of the propeller's force distributions, the results show that the PTD configuration might not be beneficial for the AR5 case. Despite the improvement in the L/D ratios, the negative propeller pitch angle does not increase the lift on the wing (contrary to [29]), and the thrust component pointing down might outweigh the reductions in induced drag. On the other hand, the positive pitch angle range within $[5^\circ, 15^\circ]$ might provide a small increase in lifting force, at the cost of a small reduction in the L/D ratio. Overall, these results do not appear to differ significantly from those obtained with the original actuator disk (excluding $\alpha_p = 25^\circ$ and $\alpha_p = -25^\circ$), indicating that, for the AR5 case, the most important interaction in this context is the change in local angle of attack on the wing resulting from the change in the angle of axial force of the propeller.

3.7 Nacelle Model

Since this study focuses on the propulsion systems of a real case, such as the TEKEVER AR5, it is also relevant to analyze the aerodynamic effects that the nacelle geometry has on the wing. We will see that the nacelle of the AR5's engine, the Evolution MK2.4, has a very significant impact on the wing's performance, and thus, it is an interesting addition to the optimization process.

3.7.1 Nacelle Surface Mesh and Volume Mesh

To include the nacelle in the simulation, new surface meshes had to be generated. TEKEVER provided the geometry of the Evolution MK2.4, and it was simplified to the geometry shown in Fig.3.28a. To ensure the nacelle's compatibility with the AR5 wing, slight adjustments had to be made to the wing's surface shape. Typically, in CFD studies involving complex geometries connected to the wing, such as fuselages, both geometries are separately defined and connected with a collar mesh that defines the region of intersection with greater detail [7, 76], in this case, a half-collar method [76] was adopted when defining the nacelle mesh, meaning that the surface mesh of the nacelle includes a half-collar surface that is coincident with the AR5 wing, as shown in Fig.3.28a. This method also facilitates the Implicit Hole Cutting (IHC) process immensely by reducing the number of overset blocks [76].



(a) Nacelle and half-collar representing only one surface mesh.

(b) Nacelle mesh placed coincidentally with the AR5 wing surface mesh.

Figure 3.28: Nacelle overset surface (L2 mesh).

According to [76], the collar mesh should surround at least 5 cells of the overlapped mesh, and the sizes of the collar cells should transition to similar sizes of the overlapped mesh (as seen in Fig.3.28b). Additionally, the curves of intersection between the nacelle and the wing should be well refined, since this region is geometrically complex and it will have a complicated flow around it. The higher refinement in this zone will also facilitate the IHC process since lower volume cells are always primarily chosen to be the computed cells [77]. It is also for this reason that a smaller initial wall spacing is typically chosen for collar meshes [76, 77]. For the nacelle and half-collar mesh, a value of $\Delta s = 0.5 \times 10^{-5}$ was considered.

Slight modifications have to be made to the pyHyp extrusion parameters, since the ones defined previously in Section 3.3 originated orphan cells. The new values are explicit on the Tab.3.3. It is important to note that the pyHyp extrusion parameters of the nacelle in Tab.3.3 are only for the L2 mesh refinement level, as the other levels required slight adjustments to improve mesh quality.

Table 3.3: New pyHyp extrusion values to ensure a correct IHC process with the nacelle.

	`epsE`	`epsI`	`theta`	`volCoef`	`volBlend`	`volSmoothIter`	`slExp`	`cMax`
Wing	1.0	2.0	3.0	0.2	0.0005	25	0.15	0.1
Nacelle	4.0	8.0	6.0	0.95	0.0005	20	0.15	1.0

3.7.2 Nacelle Mesh Refinement Study

A convergence grid study was made to examine the nacelle grid and see if it provided accurate solutions with the most favorable computing time. We use very similar wing surface meshes generated in Section 3.3, with the same number of cells in the chordwise and spanwise direction. Therefore, three mesh levels were considered once again: fine, medium, and coarse, as presented in Tab.3.4. However, in this case, the number of extrusion layers had to be reduced; otherwise, the fine mesh would contain more than 10 million cells, since now it included a nacelle surface mesh with the same level of refinement. This would cause a computation failure since the RAM available in our machine translates to a maximum number of elements of 7-8 million [7]. Considering this value, the maximum number of layers for the fine mesh was 55 for the near field wing volume and 45 for the nacelle. The nacelle and wing layers are not equal in order to maintain both grid ratios close [76], since, as expressed before, the initial wall spacing in the nacelle and collar mesh was half of the wing's for the IHC algorithm to prioritize those cells during the flooding process [76, 77]. This meant that the extrusion layers could not have a refinement factor of 2 like the surface mesh, otherwise, the L2 and L3 meshes would have too few layers, and the IHC process would fail. Therefore, the number of wing and nacelle layers of the remaining mesh levels were obtained through experimentation, and the results were: 45 wing layers and 40 nacelle layers for the L2 mesh and 35 wing layers and 30 nacelle layers for the L3 mesh. The results of the mesh are shown in the Tab.3.4. It is presented the resulting C_L and C_D of the L1 mesh and the remaining mesh levels have their error in relation to the finest mesh.

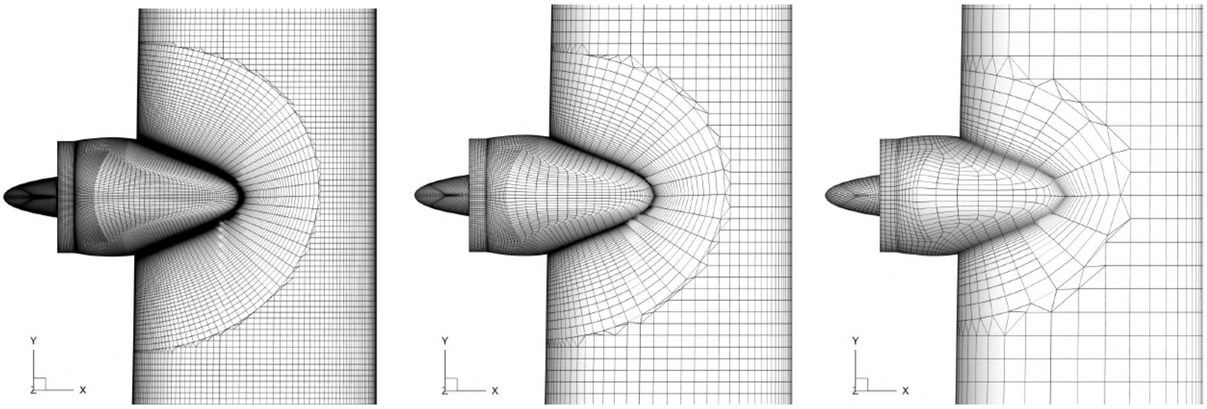
Table 3.4: Grid study values for meshes including the AR5 nacelle.

Surface mesh	Mesh cells (millions)	Computation time	Maximum y^+	C_L	C_D
L1	7.75	9 h. 30 min.	2.3	0.8760	0.0399
L2	2.13	1 h. 3 min.	2.91	$+0.38\%$	$+7.02\%$
L3	0.69	4 min.	2.10	$+0.03\%$	$+36.60\%$

We recall that for the SA turbulence method, the ideal value for y^+ is approximately 1 [7, 48], as discussed in Section 2.2.2, meaning that all meshes seemingly do not comply with this stipulation, however, the inspection of this value on the surface meshes reveals that only a very small portion of the wingtip has the excessive values of y^+, and the vast majority of the wing complies with the rule.

The surface meshes can be visualized in the Fig.3.29. We can see the disparity between the mesh quality and the computation time required to solve an analysis.

The present results guide us to a similar conclusion in the previous refinement study; despite the improvement in simulation time, the coarse quality mesh is not suitable for our study due to the large overestimation of drag, and we find that the L2 mesh is once more the best candidate for our optimization problem, since the relative errors obtained for C_L are very small and the computation time is greatly improved. The C_D error was $\approx 7\%$, which is large. However, this mesh already imposes an intense analysis that requires 1 hour to complete. Attempts were made to improve the result by making a second mesh study for the L2 mesh, but the solutions found that lowering the C_D error would be too



(a) L1 surface connected mesh, with a total of $141\ 193$ faces.

(b) L2 surface connected mesh, with a total of $35\ 406$ faces.

(c) L3 surface connected mesh, with a total of $8\ 895$ faces.

Figure 3.29: Nacelle on wing surface meshes with three refinement levels.

computationally expensive, and so, the L2 mesh was selected for the subsequent studies once again.

Regarding the actuator disk region, we notice that the propeller zone becomes much more refined as a result of including the nacelle in the configuration, thus so will the actuator disk. Figure 3.30 illustrates a front view of the selected cells of the actuator disk in the nacelle and wing configuration. The actuator disk region is much more refined in this configuration, improving especially along the azimuthal direction.

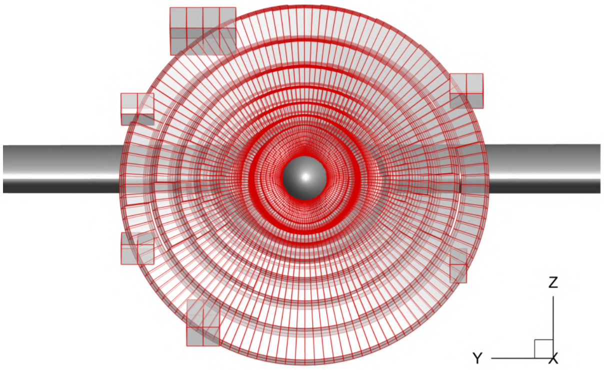


Figure 3.30: Selected cells for the actuator disk of the L2 wing and nacelle mesh.

The region inside the spinner is empty, meaning the actuator disk has no cells selected for this region, which is acceptable, since, recalling Fig.3.5, this zone is not supposed to add any energy to the flow.

3.7.3 Impact of the Nacelle on the Wing

Before discussing the impact of the nacelle on the wing's performance, it is essential to acknowledge the limitations of our simulation models. In [74], it has been shown that a wing and nacelle configuration with an axisymmetric actuator disk will bear accuracy errors for the time-averaged solution, since the propeller model does not account for airframe interference or the aforementioned blade angle of attack variations for non-zero angles of attack. As mentioned previously in Section 3.6, the ”enhanced disk” solution provided good results because these effects would change the actuator disk distribution, albeit with a slight overestimation of total wing lift and a large underestimation of drag. In our previous sections, ignoring the effects of the wing on the propeller was acceptable, since it consisted of a tractor configuration and the wing was sufficiently far away from the propeller [29], but the same cannot be assumed when a nacelle is present right behind the propeller. Its proximity will have a considerable effect on the inflow of the propeller, thus changing the real thrust and power distribution generated by it.

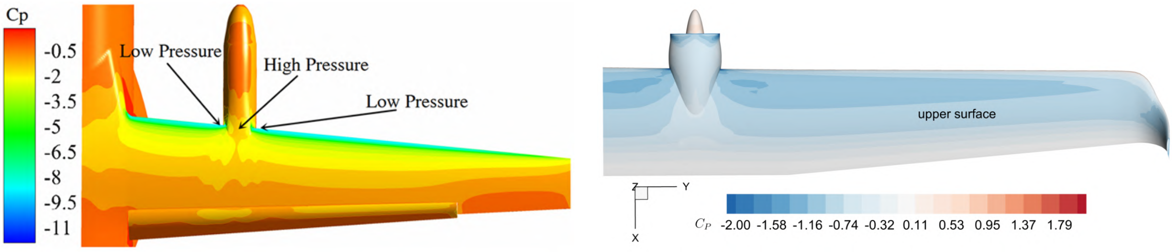
It is seen in [74] that despite the errors of the axisymmetric disk, the total estimated lift and drag of the configuration will be similar to that of the ”enhanced disk”, for $\alpha = -2°$. In fact, the distribution of both types of actuator disks was found to be similar. The bulk of the inaccuracies is in the direct forces of the actuator disk interacting with the wing and nacelle. Still, a passable solution can be obtained with the ”original” actuator disk. Considering this, and since no better alternatives were found, the inclusion of the axisymmetric actuator disk on the nacelle was also explored for the TEKEVER AR5 case study.

We now move on to the discussion of the nacelle’s impact. Other studies have dealt with the aerodynamic effects of a nacelle on a propeller-wing regarding a larger turboprop transport airplane [78], and a smaller configuration closer to the AR5 size [79]. These studies show that the interactions between the nacelle and the rest of the configuration significantly influence the flow in that region. We will verify them for our specific case by performing various simulations.

In [79], a configuration that was very similar but smaller than the one used for the actuator disk validation in Section 3.4 was tested. The study was performed under flow conditions similar to those in the AR5 case. However, as is the case for all the cited studies in this work, the nacelle used in the experiment had a much more streamlined volume. The cylindrical-shaped nacelle in this study caused negligible differences in lift and drag in the pre-stall region compared to the clean wing configuration. It was also shown that, when the propeller was powered, the resulting swirling flow over the wing surface quickly transitions the boundary layer to turbulent. This physical occurrence is favorable for our ADflow simulations, which include the full configuration, as it will decrease the accuracy errors discussed in Section 2.3.1 due to the SA model’s consideration of fully turbulent flow. It also led to the elimination of laminar separation bubbles and the formation of complex three-dimensional flow patterns, which will be visualized in this section. Another work [29] also states that the presence of a nacelle similar to this one causes a small axial velocity increase in some regions of the propeller.

In [78], the study discloses four unfavorable effects caused by the engine nacelle on the aerodynamic performance of the wing for high-lift configuration (high angle of attack and flap deflection), which, when combined, exacerbate the adverse pressure gradients between the wing’s leading edge and the high-pressure flow expand border on the upper surface of the main wing behind the nacelle, thus accelerating flow separation in this region. This will result in worse aerodynamic performance compared with the clean wing configuration, and the occurrence of stall will be expedited. Figure 3.31a accompanies this observation, which we can also recreate for our AR5 case, shown in Fig.3.31b.

To mitigate this issue, the authors in [78] suggested decreasing the height of the nacelle’s connection to the wing or modifying the leading edge junction of the nacelle with the wing. Decreasing the installation height would intensify the vortices emerging from the nacelle, which would improve the continuity of the low pressure zone found on the leading edge junction with the nacelle in Fig.3.31a, thus reducing the lift loss of the landing configuration, at the cost of increased drag for cruise conditions. The solution of



(a) Dash8-Q400 high lift configuration [78].

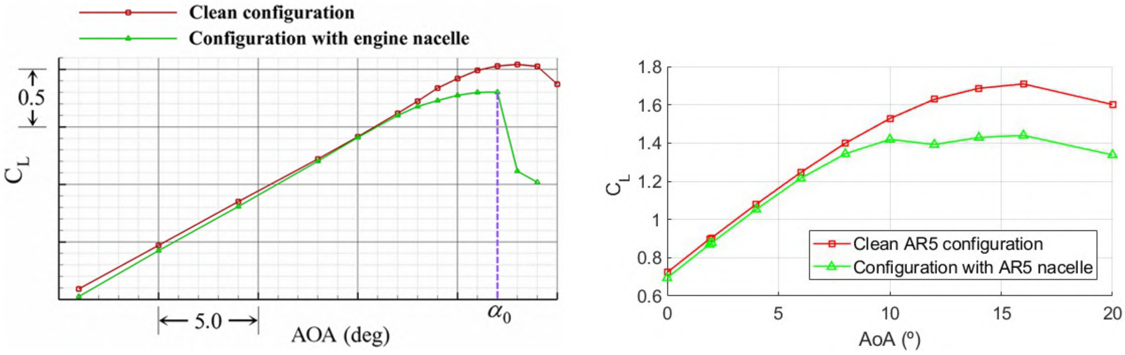
(b) TEKEVER AR5 cruise configuration.

Figure 3.31: Upper surface view of C_P distributions.

modifying the geometry of the junction zone also managed to provide the same strong vortices, which solved the problem of increased drag while delivering good performance during high-lift configurations.

It is essential to note that the turbulence model employed in the study [78] is the $k-\omega$ shear stress transport model, and the flow conditions differ significantly from those simulated in the AR5. Additionally, no flap deflection was introduced in the AR5 case. However, by comparing both pressure distributions in Fig.3.31, we can still notice a similarity between them regarding the pressure gradients in the nacelle region, and thus we might be able to conclude that the AR5 also experiences the higher-pressure flow expansion effect on the upper surface of the main wing, albeit with much weaker adverse pressure gradients in cruise conditions. This effect will cause the leading edge of the nacelle and its surrounding region to be more prone to flow separation. Considering also the blunt geometry of the nacelle's front, we can expect that this zone will have separated flow for the majority of flight conditions.

Additionally, Fig.3.32 confirms that, for high lift configurations of the AR5 (which in this case will only be larger angles of attack), the strength in these adverse pressure gradients will also increase, causing sooner high lift losses when compared with the clean AR5 wing, as seen in [78].

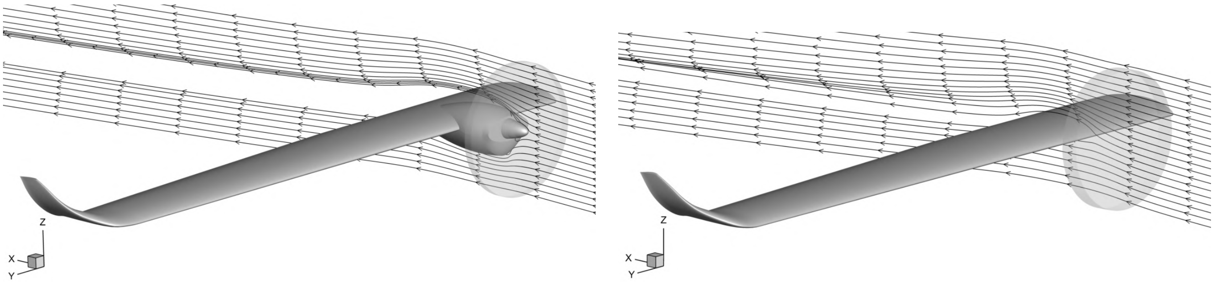


(a) Dash8-Q400 high lift configuration [78].

(b) TEKEVER AR5.

Figure 3.32: C_L curve with and without the nacelle and no actuator disk.

As aforementioned, it is important to note that the Fig.3.32a was obtained with activated flaps on the simulations, but the curve behavior is the same, since the effects of the nacelle are similar in both cases. For lower angles of attack, the nacelle diminishes the lift coefficient by an average of 3%, and for higher angles, the increased circulation intensifies spanwise pressure gradients [79], causing stall to occur earlier in the suction side of the nacelle region than it would for a clean wing. To see how these effects intertwine with the propeller, we can also add the actuator disk to the simulations which include the nacelle. The biggest impact will be changes to the vorticity inserted by the actuator disk, which now interacts with both the wing and the nacelle. As stated in [80], the most effective methods for visualizing vorticity include seeding streamlines near vortex cores and creating slice planes with contours of the velocity vectors. Following this advice, Fig.3.33 is presented, with streamlines passing through the actuator disk, on the nacelle region, with an inboard-up rotation direction at cruise conditions. The pressure distribution on the surface of the wing and nacelle was also plotted.



(a) Streamlines deflected by the full configuration.

(b) Streamlines deflected by the wing and actuator disk.

Figure 3.33: Impact of nacelle on pressure distribution and velocity streamlines.

We can see that right after the actuator disk, all streamlines in Fig.3.33 are immediately affected by it. The streamlines passing near the spinner region of the actuator disk (vortex core) become contorted downstream of the wing. The streamlines below the spinner show a shift toward the root of the wing, and those above shift toward the tip, as they are rotating in accordance with the disk's inboard up rotation. Additionally, the flow in Fig.3.33a does not attach to most of the suction side of the wing, signifying possible separation in this region, contrary to what is shown in Fig.3.33b. The early separation in the nacelle region will create a larger wake, and thus an increase in pressure drag, C_{D_p}, is expected [81].

Table 3.5: Impact of the nacelle on aerodynamic performance.

Mesh Cells (Millions)	Nacelle	Actuator Disk	C_L	C_D	C_{D_p}	C_{D_f}
2.13	Yes	Yes	0.9179	0.0471	0.0369	0.0102
2.13	Yes	No	0.8759	0.0427	0.0330	0.0097
1.62	No	Yes	0.9501	0.0410	0.0304	0.0106
1.62	No	No	0.8945	0.0353	0.0255	0.0098

We confirm this comprehensively from the results obtained for all possible configurations in Tab.3.5. The presence of the nacelle causes a lift loss, as established before, and the increase in C_{D_p} is verified (of 38% for the case including the actuator disk, and 35% without it). Furthermore, an initially contradictory result emerges from the fact that C_{D_f} is lower or very similar when comparing the cases with and without the nacelle. Intuitively, the wing with the nacelle has a larger wetted area at the same flow conditions. Since the CFD data is nondimensionalized using the same reference area, they should exhibit higher C_{D_f}. However, as shown in Fig.3.33, the nacelle region is more susceptible to separation and, in this case, the balance between the extra surface area with separated flow and turbulent flow causes the total C_{D_f} to remain approximately the same when compared to the clean wing configuration.

To visualize how the nacelle impacts the wing loading, all the configurations are shown in Fig.3.34. Once again, the inboard-up rotation direction was considered. We can observe that despite the actuator disk, the presence of the nacelle causes a very noticeable impact on the lift distribution. The nacelle only occupies a small portion of the wing, but its presence disrupts one-third of the span of the flow around the wing, resulting in a $C_L = 0.8793$, which corresponds only to a loss of lift of -1.7% when compared with the isolated wing in Section 2.3.2, but a large increase in pressure drag of 22.73%. The effect of the disruption is a lift crater similar to that caused by the fuselage in [7], showing again another possible effect of flow separation in this region. However, the lift losses caused by the AR5's nacelle are not representative of the actual wing performance, since, as seen before, the actuator disk is responsible for increasing lift. In fact, the configuration including both the nacelle and the actuator disk lift distribution is the closest to the isolated wing distribution, since it counteracts the lift loss effect of the nacelle, providing a C_L of 0.9179, near to the manufacturer-prescribed value $C_{L_{wing}} = 0.8932$, required for cruise.

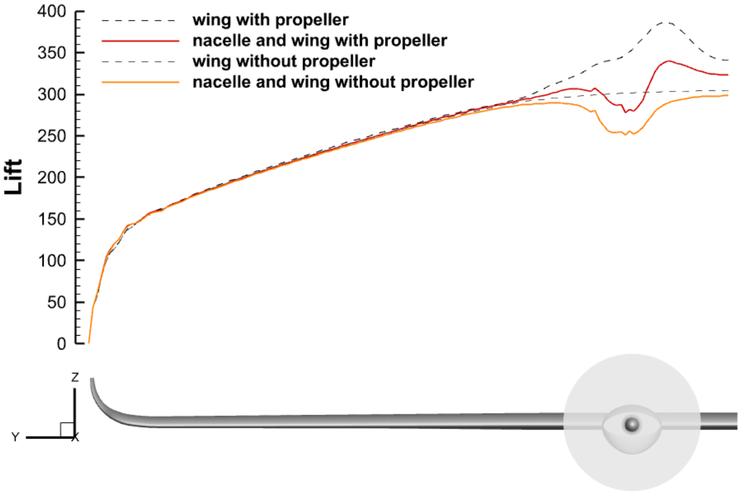
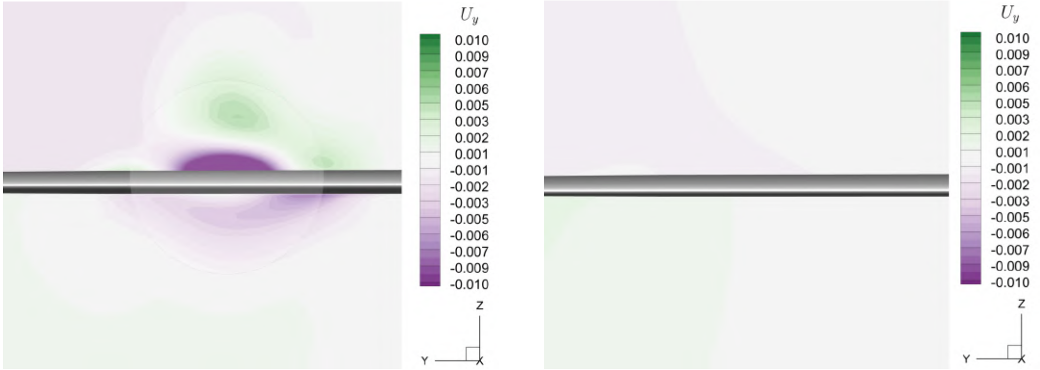


Figure 3.34: Effect of the nacelle on the wing lift distribution.

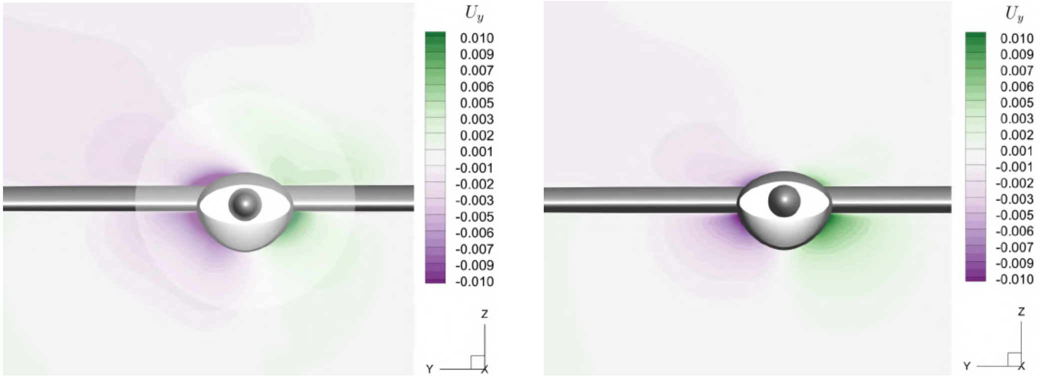
This result is not in accordance with [79], where it is determined experimentally that the nacelle has negligible effects on the lift and drag (for the nested flap case, which is also the case of every AR5 configuration simulated in this study). This could be because the nacelle in [79] has a more streamlined geometry, unlike the AR5's nacelle, which has a more perturbing, box-shaped volume, causing a more significant impact on the wing's performance.

Following the advice in [80], it is shown in Fig.3.35, the slices placed at the wing midchord and their contour that was filled with the spanwise flow velocity, (component on the y axis) U_y, showing how with an actuator disk with an inboard up rotation direction affects the U_y velocity distribution of the isolated wing in Fig.3.35a, due to the tangential velocity addition to simulate the vorticity of the propeller. We also see the Coanda effect in Fig.3.35d on the surface of the nacelle, by the fact that the U_y velocity accompanies the geometry of the surface. When the propeller is present, we obtain the full configuration Fig.3.35c, clearly adding negative U_y velocity on the lower surface (contrary to the y axis) and positive velocity on the upper surface, which follows the inboard up rotation. However, in this case, the nacelle's interference will reduce the induced vorticity of the propeller, completely changing it from what is seen in Fig.3.35a.



(a) AR5 wing with propeller.

(b) Isolated AR5 wing.



(c) AR5 wing and nacelle with propeller.

(d) Isolated AR5 wing and nacelle.

Figure 3.35: Effect of nacelle on spanwise flow velocity (leading edge view of contour slices).

It can be argued, then, that the nacelle will improve the swirl recovery factor of the configuration, which, according to [82], will lead to an increase in induced drag, possibly contributing towards the results in Tab.3.5.

In conclusion, we find that the presence of the nacelle disrupts the typical flow characteristics of the isolated AR5 wing, reducing lift and increasing pressure drag due to flow separation and its aggravation. From [78], we also see that the slight changes to the positioning or the geometry of the nacelle in relation to the wing can deliver great improvements for its aerodynamic performance. Additionally, as observed in [79], cases with electric motors feature smaller, cylindrical nacelles with a negligible impact on performance at low angles of attack, suggesting that nacelle geometry may not be as detrimental in this aspect for configurations operating under flight conditions similar to those of the AR5. For these reasons, one can expect that it might be beneficial to include the nacelle while optimizing the wing. Despite the accuracy errors of the actuator disk force distribution, in this work, we will perform optimization shape of the nacelle while accounting for the propeller forces.

Chapter 4

Shape Optimization Framework

In this chapter, the subject of optimization will finally be tackled. The optimization algorithm employed in this work will be presented, along with the sensitivity method. A brief explanation of the FFD boxes will be presented, along with the actual strategy to model the AR5 wing with a mounted nacelle.

4.1 Optimization Algorithm

There are various optimization algorithms available, which can be broadly categorized based on their use of randomness: heuristic or deterministic. Heuristic methods are inherently stochastic and do not rely on gradients, whereas deterministic methods can be gradient-based or gradient-free. These methods are typically chosen whenever gradient-based methods are inadequate, for example, when the functions are multimodal, discontinuous or noisy. Additionally, the number of function evaluations required for these methods is much higher, and as such, they have considerably higher computational cost [83]. The gradient-based alternative is shown to be more favorable in aerodynamic shape optimization cases [35, 84], since it solves these problems much more efficiently [85].

The Sequential Least Squares Quadratic Programming (SLSQP) algorithm was chosen [86]. SLSQP has been applied in aerodynamic shape optimization and is well-suited for solving constrained nonlinear optimization problems [85], which aligns with the nature of the problem addressed in this study [7]. The algorithm employs the Han-Powell quasi-Newton method with a BFGS update of the B-matrix, along with an L1-test function in the step length (line search) procedure [85].

4.2 Sensitivity Method

Gradient-based algorithms require the flow solver to compute sensitivities, which are the derivatives of the objective and constraint functions with respect to the design variables [35], in order to determine the search direction. The calculation of this gradient is typically the most computationally expensive task during the optimization process, therefore it is crucial to use methods that calculate sensitivities both accurately and efficiently [7, 83].

Several methods are available, including symbolic differentiation, finite differences, complex-step, algorithmic differentiation, and analytic methods [84]. The latter are considered to be the most accurate and efficient options available, but also the hardest to implement since they require a good understanding of the governing equations and the algorithm that solves them [83, 87]. Analytic methods are also divided into two approaches, depending on the process taken to resolve the equation of the total derivative,

$$\frac{df}{dx_n} = \frac{\partial f}{\partial x_n} - \underbrace{\frac{\partial f}{\partial y_i}\rlap{$\overbrace{\phantom{\left[\frac{\partial \mathcal{R}_k}{\partial y_i}\right]^{-1}\frac{\partial \mathcal{R}_k}{\partial x_n}}}^{-\mathrm{d}y_i/\mathrm{d}x_n}$}\left[\frac{\partial \mathcal{R}_k}{\partial y_i}\right]^{-1}}_{-\Psi_k}\frac{\partial \mathcal{R}_k}{\partial x_n} \quad , \tag{4.1}$$

where y_i represents the state variables vector, x_n represents the vector of design variables, f are the functions of interest, and $\mathcal{R}_k$ are the residuals of the flow governing equations (Section 2.1).

If we start by calculating $\mathrm{d}y_i/\mathrm{d}x_n$, we are using the direct method, which requires the solution of the matrix for each design variable x_n, meaning that the computational cost of this approach will be proportional to the number of design variables.

The alternative approach for computing the total sensitivity in Eq.(4.1) is by using the adjoint vector, Ψ_k, that can be obtained by solving the adjoint equations,

$$\frac{\partial \mathcal{R}_k}{\partial y_i}\Psi_k = \frac{\partial f}{\partial y_i}. \tag{4.2}$$

The inverse of the Jacobian $\frac{\partial \mathcal{R}_k}{\partial y_i}$ does not have to be explicitly calculated, especially in large iterative problems, such as ours. The presented vectors hold information regarding all mesh cells [36], resulting in a great size for this matrix that prohibits explicit calculations [83]. This vector Ψ_k is then substituted into Eq.4.1 to compute the total sensitivity. Contrary to the direct method, the adjoint vector does not depend on the design variables, x_n, but it will have to be recomputed for each function. However, in aerodynamic shape optimization problems, the number of design variables is often much larger than the number of functions of interest, making the adjoint method very computationally efficient in this scenario. It was first developed to solve optimal control problems, and Jameson's advancements allowed it to be used for aerodynamic shape optimization [88], which are considered the most important development in this subject [38].

It is for these reasons that the adjoint method has been implemented in ADflow [36]. The approach that ADflow uses to compute the terms necessary to form the discrete adjoint equations is automatic differentiation, which results in accurate derivatives [35].

4.3 Shape Deformation Technique

As previously mentioned, this work will perform aerodynamic shape optimization on the wing and mounted nacelle configuration, two distinct aircraft components, joined using overset meshes. Therefore, this problem will address component intersections, which other studies have explored in various cases, such as wing and fuselage intersections in [89, 90] and strut-braced wings [91].

The aerodynamic design of these component intersections poses a recurring challenge that necessitates special handling of deformations to maintain mesh quality within the aerodynamic shape optimization process [90]. The most straightforward method using Free-Form Deformation (FFD) is to embed the entire configuration in a single FFD box, which avoids much of the complexity of the problem. However, this approach does not allow for independent changes between each component, potentially providing worse results [90].

There are some strategies with FFD boxes that one can adapt to solve this problem. In one study, Yildirim et al. [89] were able to perform aerodynamic shape optimization of an intersection between the wing and the fuselage of the DLR-F6 configuration with smooth deformations to the CFD surface mesh and no topological changes to the overset mesh. This was achieved by defining one FFD volume for each component, specifically, one for the wing, and another for the region of the fuselage where the intersection occurred, allowing for separate parametrization of each component and thereby independent deformation. The process described in this study, which builds upon the work by Secco et al. [91], provides for separate parametrization of the geometry of each component, and obtains a 2.2% reduction in C_D when comparing the optimized design with the baseline geometry.

Another strategy recently employed in another study regarding a conceptual tilt-wing VTOL vehicle [90] proved that there were substantial benefits to creating an additional component for a fillet that represents the intersection geometry in a continuous CFD surface mesh. Hajdik et al. used two separate FFD boxes for the wing and the fuselage, and a fillet that had no specifically attributed FFD box. Instead, the fillet component was deformed through changes in its adjacent components, using inverse distance warping. Similarly to other studies, this allowed for the isolation of the design changes in each component, thus achieving much better results than with aerodynamic shape optimization with alternative parametrization methods. In this study [90], it was found that performing optimization with this strategy for the intersection fillet, the C_D of the configuration was reduced by 2% more in relation to the baseline when compared to performing the optimization process with the intersection locked.

Although all the studies in this section utilized the MACH-Aero framework, their implementations are complex, and the intersection cases differ from the intersection between the nacelle and wing of the AR5 configuration. As such, a simple approach for the FFD box strategy was considered, embedding the entire configuration in a singular volume, as shown in Fig.4.1.

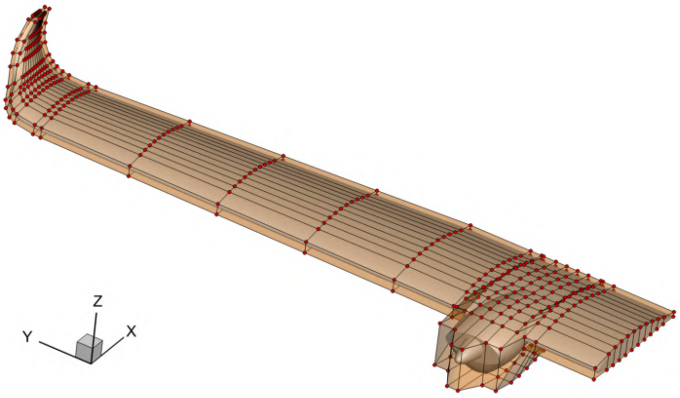


Figure 4.1: Example of an FFD box embedding the nacelle and wing (not used).

The FFD box was created using the pyGeo function `createFittedWingFFD`, which generates FFD volumes closely resembling the geometry of the wing it embeds, following the airfoil geometry in each spanwise section. For the case of the AR5 wing, which features a winglet, the function had to be modified to correctly define the volume in this region. These modifications were developed in [7] and were also used in this work. However, the presence of the nacelle required more modifications to the function `createFittedWingFFD`. Since this function was capable of finding the airfoil geometry by projecting user-given points to the geometry surface in the normal direction, it can also obtain a 2D chordwise slice of the nacelle geometry by specifying points inside the nacelle. Placing these points strategically to obtain the nacelle profile with the highest width, and then copying this profile across the nacelle region, provides us with the FFD volume shown in Fig.4.1. This simplified approach sacrifices precise control near component interfaces but remains sufficient for the scope of this study, while enabling faster iteration and lower computational costs [90].

4.4 Design Variables

In order to manipulate the FFD volume defined in Section 4.3, the MACH-aero user needs to specify the design variables by using functions from the DVGeometry class, built in pyGeo [92].

One of the most basic design variables in DVGeometry is the local shape variable, which allows for small shape modifications in a mesh. These are defined by specifying the axis of deformation, on which the FFD control point will fluctuate to deform the corresponding mesh, and the upper and lower bounds, which state how much fluctuation is allowed. While these are not the typical geometry variables used in wings for aerodynamic shape optimization (for example, twist and dihedral), they are utilized for airfoil optimization [7] and fuselage optimization [67], therefore being the most appropriate for the nacelle region. However, not all aspects of the nacelle's geometry are intended to change during optimization. For instance, the actuator disk is supposed to remain in the same location during this process, and therefore, so should the location of the spinner. This translates to fixing the location of the FFD points near the spinner, which can be achieved by using the `PointSelect` option of the local shape variable.

Each point of the FFD control volume possesses an index, which can be selected and passed to the design variable function, ensuring that only the selected points move accordingly, and the others remain fixed. The indices of the FFD volume in Fig.4.1 are presented in Fig.4.2.

For example, in this FFD volume, the points that control the spinner's geometry are $[1, 5, 2]$ to $[1, 8, 2]$, and $[1, 5, 1]$ to $[1, 8, 1]$. Therefore, they will not be selected to fix the spinner. To define deformations to the entirety of the wing with the aforementioned geometry variables, it is necessary to create additional implementations, such as a reference axis, as explained next.

To see large-scale changes to the geometric design of a wing, DVGeometry has a function that provides a method to define a reference axis of a wing from where the geometry will be altered with respect to the defined global design variables [93]. The most practical way of defining a reference axis is by using the functionality `xFraction`, within the function `addRefAxis` in the `pyGeo` module. This will create a reference axis located at a given distance from the leading edge of the wing's FFD box.

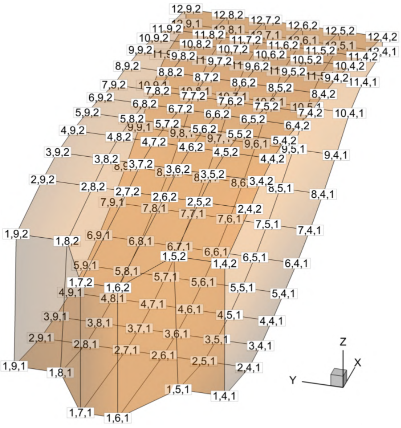


Figure 4.2: Indexes of the FFD points controlling the AR5 nacelle geometry.

However, as we can see in Fig.4.1, the nacelle region is also included in the FFD box, and so, we require a way to explicitly define the reference axis of the AR5 wing and employ it in the nacelle and wing configuration. Since we can also easily create an FFD box of the AR5 isolated wing, we can then save that axis as a data file, turn it into a `pySpline` curve, and reapply it in the case that includes the nacelle. Explicitly defining the `pySpline` curve is the recommended approach for non-conventional FFD boxes [93], such as the one in Fig.4.1, resulting in the axis shown in Fig.4.3.

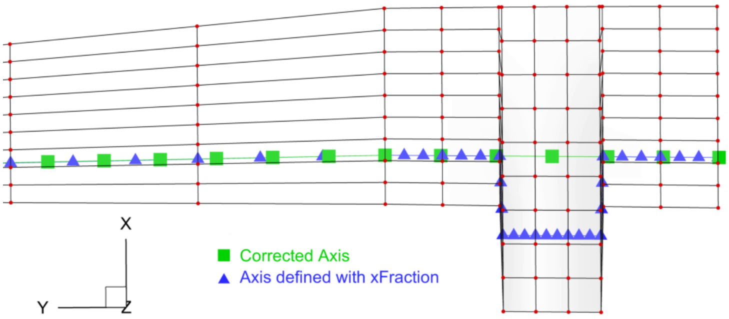


Figure 4.3: Correction of the reference axis.

Regarding the definition of the reference axis at the wingtip, Gameiro's work [7] corrected an issue arising from the initial definition of global design variables and twist using pyGeo. Since `rot_y` used the y-axis as a global reference frame to apply twist, for wings that possess a winglet, such as the AR5's, it would result in unintended deformations in this region. The rotation was kept along the reference axis by calculating each local vector using the FFD box's points along each wing section. Similarly, other corrections were made for the dihedral and sweep angles, where the FFD sections are rotated to maintain their orthogonality in relation to the reference axis, and the scale factor adjusted accordingly.

The use of the reference axis allows the parametrization of the geometry with geometric design variables, which are more intuitive, like twist, taper, or dihedral of the wing. However, in this work, the focus of the optimization was mainly on the nacelle's region on the wing, so these parametrizations were not used.

4.5 Geometric Constraints

To ensure valid physical solutions, with enough space for structural components, `pyGeo` also allows the user to define geometric constraints for wings [94]. These can be categorized into various types, of which we will be using: thickness constraints, volume and triangulated surface constraints, and leading-edge and trailing-edge constraints.

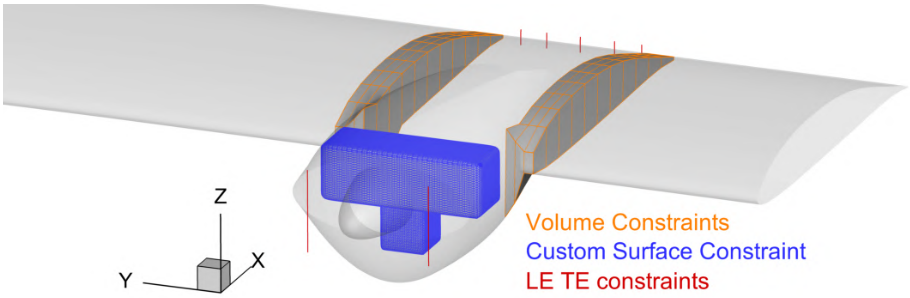


Figure 4.4: Geometric constraints applied to the AR5 wing and nacelle configuration.

Thickness constraints ensure that the final result is achieved with sufficient space for the structural components of the aircraft's wing, namely the wing ribs. Volume constraints are designed to maintain a volume inside the aerodynamic hull during the optimization process. Both geometric constraints can be defined automatically using the existing wing geometry through point projection, but to guarantee sturdiness in the intersection region, only the latter constraint was used, as represented in orange in Fig.4.4. It is also possible to manually define the volume of a given component and place it inside the configuration's hull to serve as a geometric constraint, by generating its enclosed surface with the `addTriangulatedSurfaceConstraint` function. This is a more complicated process since it involves using other software or other libraries, such as PyVista [95], to create the triangulated surface of the component, so that during the optimization process, the intersections between the component and the shape being optimized can be computed. We will use this method to represent the geometry of the engine of the AR5 (Fig.4.5) in a simplistic but efficient manner. The engine is the 210/250B2-JCH 2021 developed by ZDZ engines [96].

Figure 4.5: ZDZ engine 210/250B2-JCH 2021 [96].

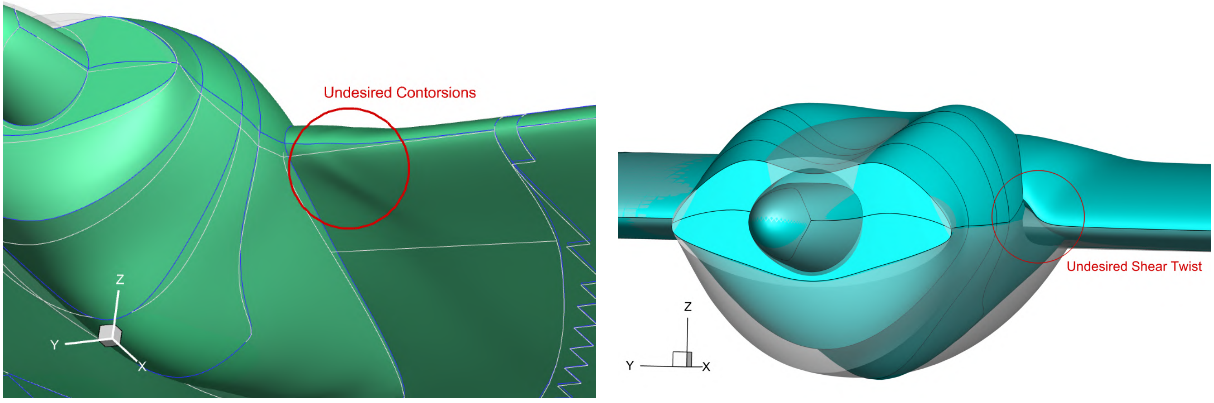
The surface consists of two intersecting quadrangular prisms, one with $100 \times 290 \times 100$ mm and another with $80 \times 80 \times 140$ mm shown in blue in Fig.4.4, and was created with PyVista. This method has been used for fuselage optimization [67], ensuring an optimized solution that deformed itself to encompass specific components. More specifically, in our nacelle and wing case, it will ensure that the engine inside the nacelle has sufficient space, and that the intersection remains with enough volume for structural strength.

The leading and trailing-edge constraints avoid shearing twist by ensuring that both top and bottom control points at those edges move in equal and opposite directions. This is necessary when the section shape variables can imitate a twist variable, such as in the case of a local section design variable in the y direction, as shown in the latter section. Without this constraint, the optimizer provides solutions with such reduced nacelle volumes that the result is unfeasible.

4.6 Limitations Regarding the FFD Box Strategy

Limitations arise from creating a fully embedded FFD box for our complex component intersection problem, whose quality of the parametrization of the nacelle and wing configuration is poor in the intersection region between the nacelle and the leading edge of the wing, due to the way the surface mesh is projected into the FFD box.

The quality of parametrization in this zone is compromised because FFD lattices are confined to a $i \times j \times k$ definition, meaning the number of nodes is constant in each axis. Therefore, the FFD region uniting the wing to the nacelle region will be highly skewed, as shown in Fig.4.6a, resulting in inadequate projections from the surface mesh to the FFD volume. This problem is amplified by applying shape design variables in the nacelle region (Fig.4.2). The nacelle shape deformations that occur during the optimization process will also cause unintended problems in the leading edge of the wing near the nacelle. If the matter at hand is not addressed, the optimization can be completed without failed meshes, but the result will improperly include shear twist if the FFD points near the intersection remain unconstrained, as seen in Fig.4.6b, and even if we apply linear constraints to the points in these regions (for example, the square defined by the points $[3.4.2]$ $[3.5.2]$ $[4.4.2]$ $[4.5.2]$ in the inboard side), local bulges will appear, contorting the leading edge geometry as seen in Fig.4.6a.

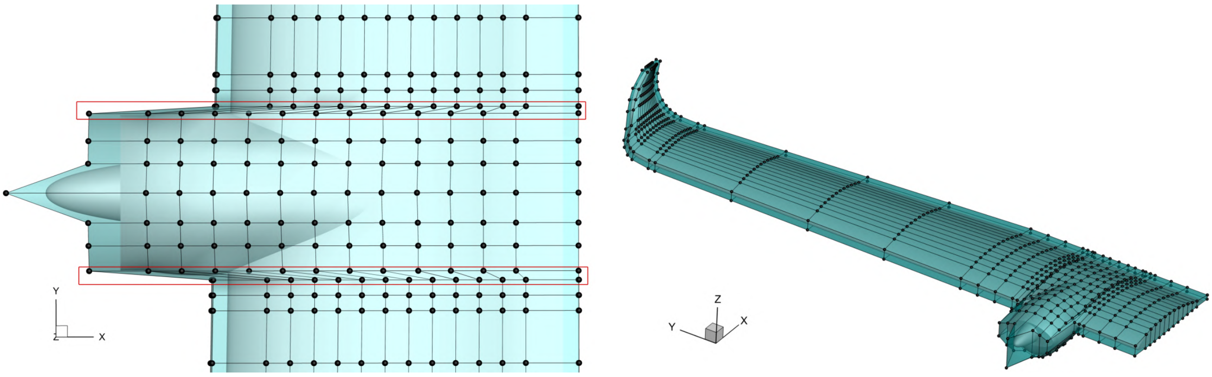


(a) Undesirable leading edge contortions in a constrained shape optimization problem.

(b) Undesirable shear twist caused by an unconstrained leading edge in a shape optimization problem.

Figure 4.6: Problems arising from the poor FFD volume.

The solution is to fix these problematic sections in a similar fashion to the points near the spinner, as explained in Section 4.4. This implies that the leading edge on the intersection zone with the nacelle, one of the most important zones that would benefit from aerodynamic shape improvement (as discussed in Section 3.7.3), will be frozen during optimization, which poses a substantial limitation for the fully embedding FFD volume. This zone is delineated in red in Fig.4.7a, which shows the chosen FFD solution for the optimizations. Figure 4.7b presets a different view of our adopted FFD strategy.



(a) Fixed sections in the adopted FFD volume.

(b) FFD volume view.

Figure 4.7: Adopted FFD volume for the following optimization problems.

In Appendix C, we also present our iterative steps of diverse FFD strategies that were considered but still did not parametrize the intersection with sufficient quality.

Chapter 5

Optimization Results

In other aerodynamic optimization problems [5, 15], it was found that the wing directly behind the propeller decreases its twist in both the up-going and down-going blade regions. However, very few have included the nacelle geometry during the optimization process. In this chapter, we notice that including the propeller will also cause asymmetry of the nacelle, due to the tangential forces of the upgoing and downgoing blade regions.

5.1 Optimization Problem Definition

Now that the design variables and constraints have been defined using pyGeo, the optimization parameters can be set up with pyOptSparse. We will be performing several single-flight-point optimization problems. In this work, similar to other studies on aerodynamic shape optimization, the objective will always be to minimize the drag coefficient. Additionally, we will determine whether the propeller effects are beneficial during an aerodynamic shape optimization process for a wing and nacelle configuration [5]. To achieve this goal, we will have one case optimized without an actuator disk, and two others with it (one for each rotation direction). After obtaining the first result (optimized without an actuator disk), we include the actuator disk to perform a leveling flight operation, which consists of an optimization with solely the angle of attack as a design variable [7], with the same $C_L = C_{L_{cruise}}$ constraint.

However, there is one important aspect regarding the propeller blade angle of attack errors that needs to be addressed. As seen previously, the higher the angle of attack, the larger the accuracy errors of the predicted lift will be. Therefore, the actuator disk should be perpendicular to the freestream flow velocity direction, which means $\alpha_p = 0^\circ$, for all cases. This poses a problem for our optimization scenarios, as one of the design variables will be the angle of attack of the entire configuration, which means altering the freestream flow direction, and thus α_p. The angle of attack of the optimized solutions can differ from one case to another, and as such, direct comparisons between them might be flawed. In other words, since we use the unmodified actuator disk (axisymmetric disk), the fact that it will not be subjected to the same freestream flow will cause discrepancies in the accuracy of the induced forces on the nacelle-wing configuration for each different case.

Previous studies [5] have resolved this issue by establishing every FFD section with twist variables, which effectively emulated the angle of attack design variable, and fixed the freestream velocity direction to have the previously stated $\alpha_p = 0°$. This method ensures that each case is subjected to the same propeller inflow, while the wing angle of attack of the baseline case can be freely adjusted to obtain the desired C_L. Nevertheless, this approach is incompatible with our optimization problems because our configuration includes a nacelle. In the latter study, the difference in relative angle between the wing and the propeller across the cases was regarded as an unaddressed design variable, and the consistency of propeller inflow was prioritized, but the presence of the nacelle requires that the spinner position and orientation remain fixed in relation to the actuator disk, disallowing this method. Our previous modification in Section 3.6 also will not work, because, as of now, the actuator disk method in ADflow works by initially defining only one geometry and force distribution that is used during the entire optimization process. This means it does not update with each iteration, and thus, if the angle of attack is intended to be a design variable in an optimization problem, this modification is invalid. With no other option available for propeller effects prediction for our cases, we must assume this error and attempt to mitigate it. To counteract this issue, we simply reduce the freedom of the angle of attack design variable and study only cruise conditions, where its value is relatively low. Then, to compare every optimization result with each other and their respective baseline, we perform a leveling flight operation on the cases optimized without the actuator disk, and on all baseline cases to achieve the desired $C_{L_{\text{cruise}}} = 0.8932$. While the value of $C_{L_{\text{cruise}}}$ was stipulated for the isolated wing [7], in this work, we consider it to be the required C_L for any configuration to maintain level flight. We call this a ”leveling flight” operation because the modeled propeller thrust force could exceed the configuration’s total drag, resulting in accelerated flight. As we will see in this chapter, all of the leveled results showed very similar values of angles of attack, and so, for this work, it was considered an acceptable accuracy error.

In this work, the convergence accuracy of the SLSQP algorithm was set to 1×10^{-3}, ensuring that the solution satisfies the Karush–Kuhn–Tucker (KKT) [97] stationarity condition with minimally acceptable numerical accuracy. This higher tolerance was used since the processing time saved was much greater than the small improvement in the obtained results. This value is two orders of tolerance above the used adjoint tolerance, which is 1×10^{-5}, since the adjoint solution is usually less accurate than the required optimality [98]. However, the adjoint tolerance value had to be tightened to 1×10^{-7} in the cases with surface constraints, due to their increased complexity. The maximum number of iterations for the optimization algorithm was 500, and, similarly to [7], we found that the convergence accuracy was satisfied much sooner than this maximum.

As an initial optimization, only the nacelle region will be optimized, as expressed in Section 4.4. This means that the local shape variable in Tab.5.1 will only apply to the five FFD sections in the nacelle region shown in Fig.4.2. These FFD points will only have design variables of vertical shape displacements. In contrast, the other FFD points across the wing will remain completely fixed during this optimization process. Lift requirements can be incorporated using custom user-given functions. Once again, to ensure that the lift coefficient reaches a prescribed value to sustain level flight, we set a lift constraint of $C_L = C_{L_{\text{cruise}}}$. Since the optimizations were performed only with shape design variables, no reference axis was defined for this problem.

The primary optimization will consist of the nacelle optimization with an engine volume constraint, which is defined by the triangulated surface constraint, shown in blue in Fig.4.4. The constraints of spatial feasibility explicit in the last two rows of Tab.5.1 relate to the engine volume constraint, which are not imposed in the initial optimization [67].

Table 5.1: Formulation of the initial nacelle optimization problem, including a triangulated surface constraint.

	Function/Variable	**Description**	**Quantity**
minimize	C_D	Drag coefficient	
w.r.t.	$0 \leq \alpha \leq 3$	Angle of attack of the entire configuration. [deg]	1
	$-0.05 \leq \Delta z \leq 0.05$	Vertical displacements of the FFD control points encompassing the nacelle sections. [m]	128
		Total design variables	**129**
subject to	$C_L = 0.8932$	Lift constraint	1
	$t/t_{\text{initial}} \geq 0.9$	Volume constraints to maintain acceptable airfoil and intersection thickness.	2
	$\Delta z_{\text{LE,upper}} = -\Delta z_{\text{LE,lower}}$	Constraints of the selected leading edge FFD control points to impede shear twist in the nacelle.	2
	$\Delta z_{\text{TE,upper}} = -\Delta z_{\text{TE,lower}}$	Same as the previous constraint, but applied on the trailing edge FFD control points in the nacelle sections.	5
	$KS_{geom} \leq 0$	Aggregated minimum distance (spatial feasibility Condition 1)	1
	$L = 0$	Non-intersection (spatial feasibility Condition 2)	1
		Total constraint functions	**12**

To achieve a solution that closely aligns with the geometry of the engine by leveraging more flexible design freedom, we require a different method of parametrization that enables the displacement of all the nacelle's points in an additional horizontal direction. The correct method would be to implement component intersections, thus separating the nacelle from the wing and allowing for independent deformation. To explore the idea of the motor being better enclosed in the nacelle, this option was tested once again; the nacelle's geometry was closed, and a triangulated mesh was generated for both it and the AR5 wing. However, within the MACH-Aero's framework, the intersection between our nacelle and wing was considered too complex for the triangle-triangle intersection detection algorithm to recognize, and many errors were found while trying to compute it. With this in mind, another approach was considered, involving optimizing the nacelle separately, explicit in the following Section 5.3.1.

5.2 Nacelle Region Optimization

For every different case of the optimization problems shown in Tab.5.1, the exit code was 0, meaning all the optimizations terminated successfully. The results in Tab.5.2 immediately show that, for the defined problem, including the propeller appears to be beneficial.

Table 5.2: Nacelle region optimization results for each case.

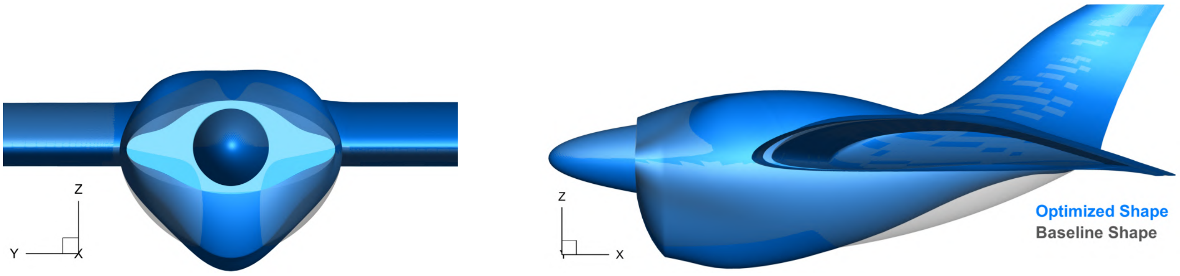
Propeller configuration	Surface constraint	Final α [deg]	ΔC_D	Func. calls	Grad. calls
No propeller	No	1.86	-6.22%	13	12
Inboard up	No	1.54	-5.47%	19	18
Outboard up	No	1.55	-3.06%	19	18

However, when including the actuator disk, the cost of the optimization increases significantly, with more gradient calls required to achieve the same convergence accuracy. The optimization case without an actuator disk took 11 hours, while the others required 17 hours. This considerable difference in time indicates that analyzing the relevance of the disk during the optimization process is important.

To understand the aerodynamic improvements identified by the optimizer, we will analyze the aerodynamic properties of each case and compare them with the unoptimized shape.

5.2.1 Case without Propeller

Figure 5.1 compares the overall shape of the original nacelle shadow and its optimized counterpart.



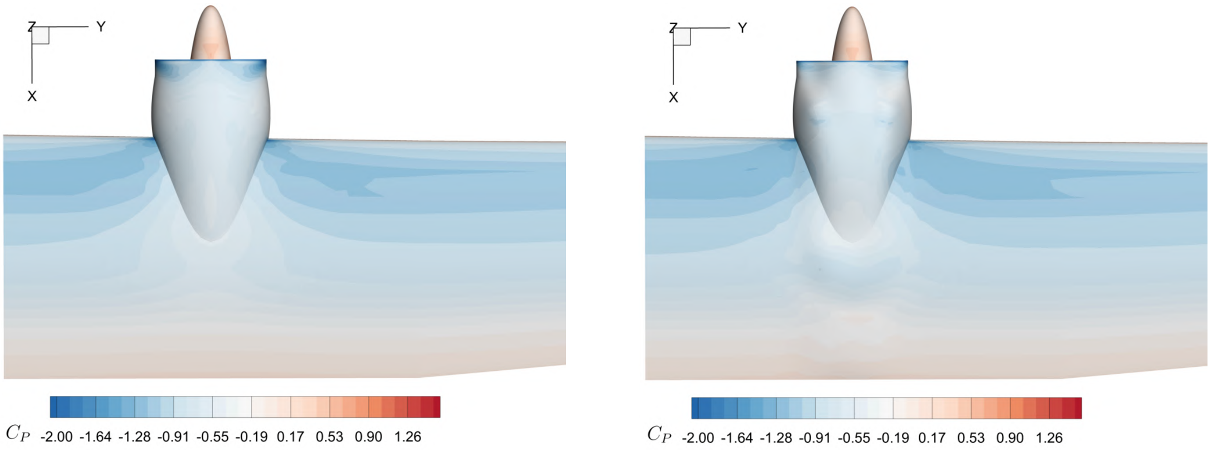
(a) Front view.

(b) Side view.

Figure 5.1: Optimized shape of a case without the propeller.

We can see that the optimized shape shrinks the nose of the nacelle, reducing the area perpendicular to the flow, making it more streamlined. This alleviates the strong adverse pressure gradient on the front of the baseline geometry in Fig.5.2a. This is complemented by the fact that the negative skin friction region in the nose in Fig.5.3 has decreased substantially from the baseline to the optimized case, which presumably means that flow separation has been reduced in this area. Further downstream on the upper surface, the shape of the nacelle grew upwards, possibly due to two reasons.

Firstly, it might be an attempt to homogenize the high pressure flow expand border discussed in Section 3.7.3, since, as seen in Fig.5.2, the pressure distribution of the upper surface of the optimized geometry has a zone with lower pressure than is baseline. Additionally, increasing the height in this section may also serve as a blockage between the very low-pressure areas in the leading edge of the wing, connected to the nacelle, and the higher-pressure zone in the nacelle's upper surface. This could effectively reduce flow separation in this region, where it was prominent for the baseline case.

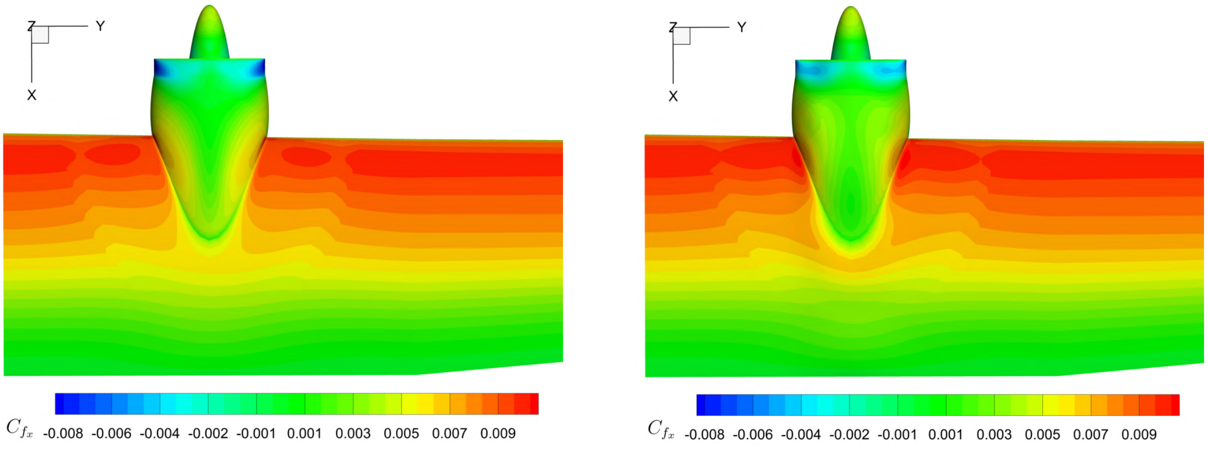


(a) Baseline shape.

(b) Optimized shape.

Figure 5.2: Top view of the pressure distribution comparison of the case without the propeller.

Secondly, the optimizer may struggle with the limited design freedom at the intersection of the nacelle and wing. Supposedly, if the connection between the nacelle and the wing was smoothed, the pressure gradients could decrease, and so would pressure drag. However, since the only method of changing the intersection's thickness involves deforming the nacelle's upper surface, it must be pulled upward.



(a) Baseline shape.

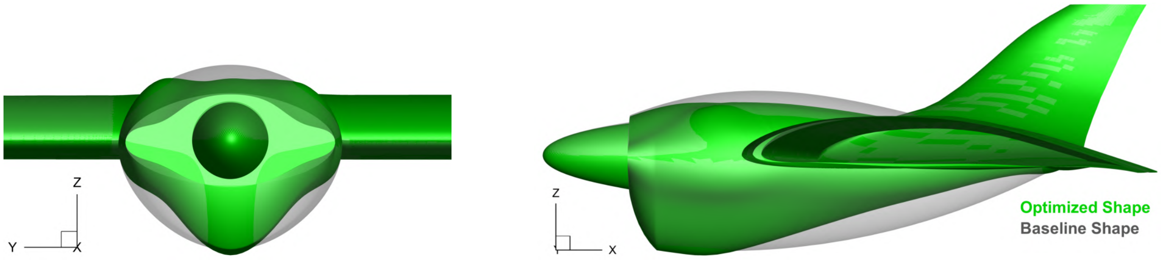
(b) Optimized shape.

Figure 5.3: Top view of the skin friction coefficient distribution comparison of the case without a propeller.

This solution obtained a reduction of 23 drag counts from the baseline configuration to the optimized version. However, we must keep in mind that this result was obtained with no actuator disk present during the optimization process.

5.2.2 Case with an Inboard Up Propeller

We now recreate the same optimization problem, but with an actuator disk rotating in the inboard up direction. There are noticeable visual differences between the optimized shapes shown so far. Most notably, the inclusion of propeller effects in the optimization loop will result in an asymmetric shape, contrary to the optimized shape in Fig.5.1. In Fig.5.4, the nacelle surface in the downwash region will protrude upward slightly more than on the upwash side. Additionally, the lower surface becomes sleeker, as seen in Fig.5.4b.

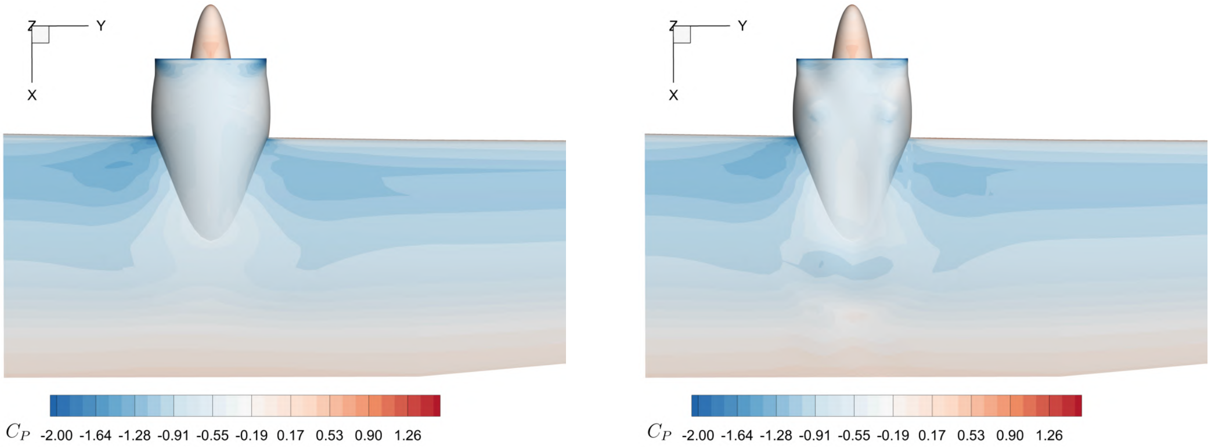


(a) Front view.

(b) Side view.

Figure 5.4: Optimized shape of a case with an inboard up rotating propeller.

It appears that since the upwash region has a lower pressure zone, the effective angle of attack increases, resulting in the nacelle's upper surface being pulled upward, which helps mitigate the stronger adverse pressure gradient on this side of the nacelle. This creates a low pressure zone on the upper surface of the nacelle near its intersection with the leading edge, which is missing in Fig.5.2a. We can also identify a similar behavior on the downwash side, which is adapted for the weaker pressure gradients on the leading edge of the wing connecting to the nacelle. This region benefits from being pulled upward slightly higher in the upper surface and from being more concave on the lower surface of the nacelle. This might help improve the swirl recovery property of the wing, similarly to how an optimized wing presented an upward twist in the downgoing blade region [15], which changes the wing lift distribution closer to the optimal distribution, thus reducing induced drag.

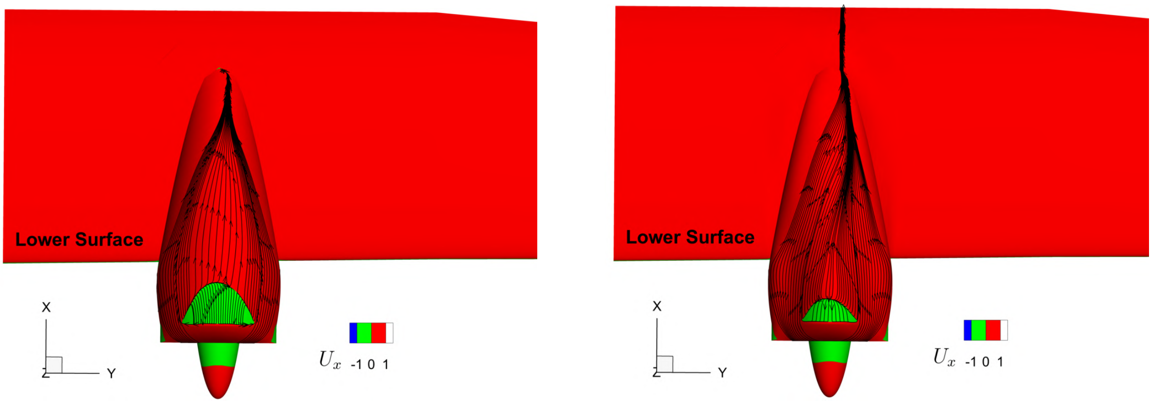


(a) Baseline shape.

(b) Optimized shape.

Figure 5.5: Top view of the pressure coefficient distribution comparison of the case with an inboard up rotating propeller.

However, this contradicts the findings in [82], and the theory in [29], which states that the vorticity of the propeller actually decreases induced drag, due to the effects explained in Section 3.5.3. Nevertheless, in our case, mitigating vorticity will result in a noticeable drag reduction, possibly because, as seen previously, the AR5 configuration benefits only slightly from this effect, as also noted in Section 3.5.3, and it also might help bringing the wing loading close to an optimal distribution (which is not necessarily elliptical, as stated in [10], due to the propeller's influence on the "clean wing" slipstream). The lower surface of the optimized shape has also been thinned while maintaining the protrusion in the middle, possibly reducing both flow separation in the region near the nose and the swirl induced by the propeller forces. To verify these ideas, the streamtraces on the lower surface can be analyzed in Fig.5.6.

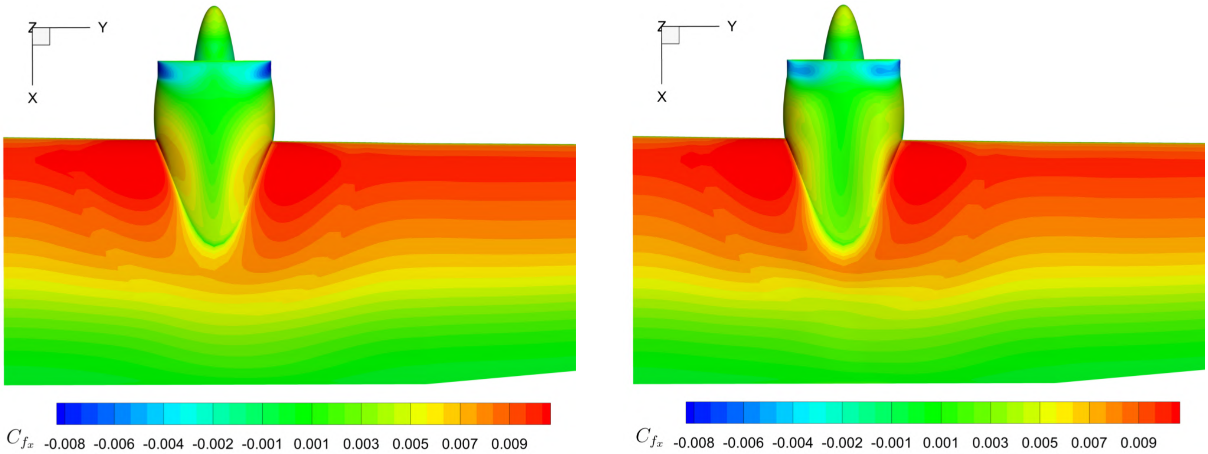


(a) Baseline shape.

(b) Optimized shape.

Figure 5.6: Bottom view of the streamtraces with a velocity contour of the optimization problem with an inboard up rotating propeller.

Flow separation should occur in the region where the streamtraces reverse direction, shown in green in Fig.5.6, and as we can see, it has clearly decreased in the optimized shape (Fig.5.6b). Moreover, the flow downstream of the optimized nacelle is much better redirected towards the trailing edge, while in Fig.5.6a there is also a small region of flow separation in the intersection of the wing and nacelle. This increased flow redirection towards the opposite tangential velocity of the propeller induced vorticity might also contribute to the swirl mitigation, since we can verify a reduction of the negative U_y region below the leading edge from Fig.3.35c to Fig.5.18b.



(a) Baseline shape.

(b) Optimized shape.

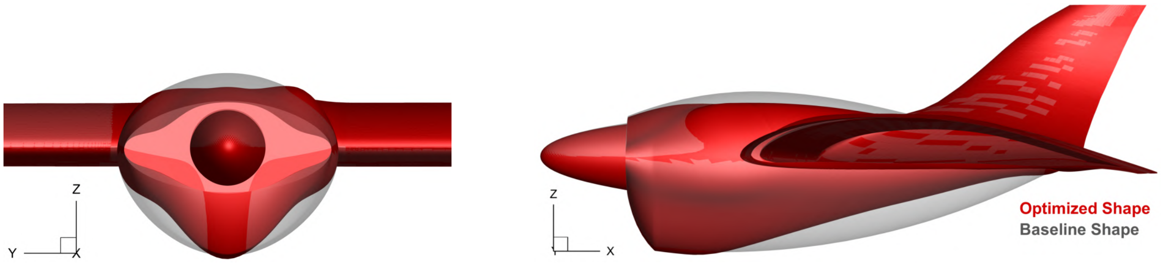
Figure 5.7: Top view of the skin friction distribution comparison of the case with an inboard up rotating propeller.

From Fig.5.5, we see a similar pressure distribution to the previous baseline case in the sense that the large pressure gradients in the front sides of the nacelle have been greatly reduced, which, when analyzed in conjunction with Fig.5.7, show that flow separation seems to have been reduced in this zone.

Throughout every optimization case in Section 5.2, it is also interesting to note that with the given design variable, the optimizer always creates a swelling aft of the nacelle's upper surface, which effectively introduces a favorable pressure gradient in the middle of the nacelle region, which might contribute to further reducing flow separation in this zone.

5.2.3 Case with an Outboard Up Propeller

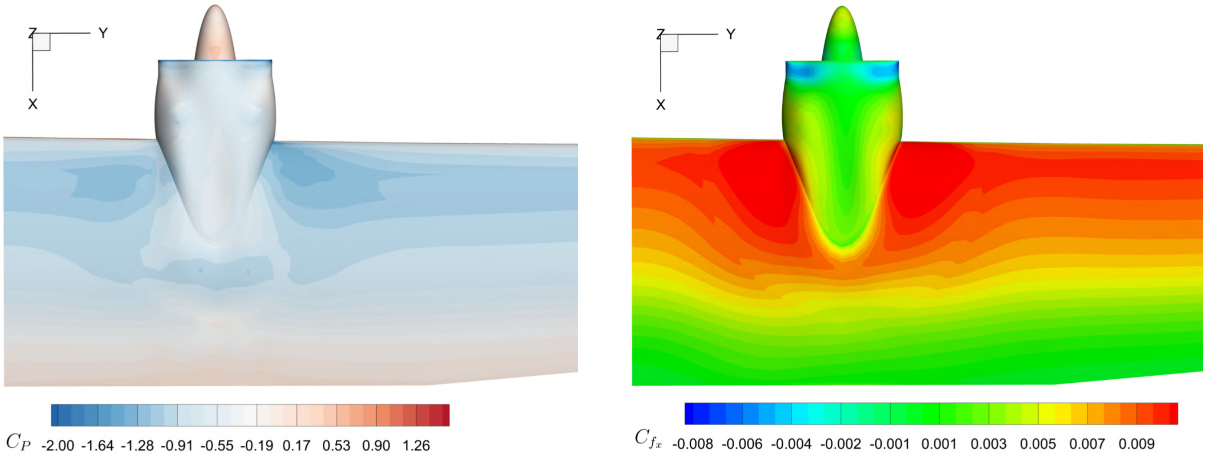
In this case, we invert the direction of the tangential velocity of the actuator disk and apply again the optimization problem described in Tab.5.1. As expected, the results show another asymmetric geometry; however, this time, it is interesting to note that the shape is mirrored in relation to the previous case (Fig.5.4).



(a) Front view of the optimized solution's geometry.

(b) Side view of the optimized solution's geometry.

Figure 5.8: Optimized shape of a case with an outboard up rotating propeller.



(a) Pressure coefficient.

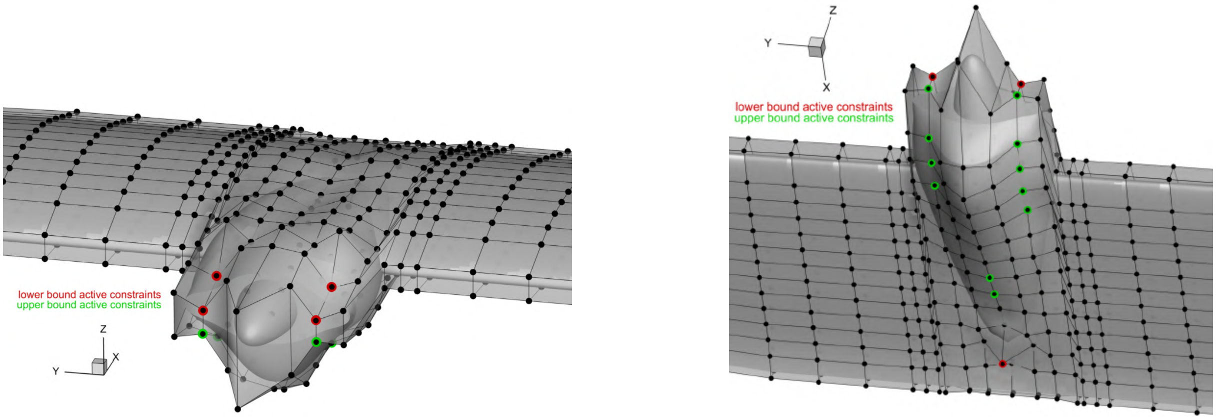
(b) Skin friction coefficient.

Figure 5.9: Top view of contour distribution comparison of the case with an outboard up rotating actuator disk.

The downwash and upwash regions have switched places, while the parametrization and design freedom remained equal. Consequently, the strategies of the previous case were adapted and applied to this situation. We can confirm this by analyzing the following Fig.5.5 and Fig.5.8 and notice that they are also practically identical to the distributions in Section 5.2.2, but mirrored along a y plane dividing the nacelle, just like the propeller forces. This solution provided a smaller improvement in drag reduction when compared to the opposite rotation, possibly because, as we have seen before, the performance of the wing configuration with this outboard rotating disk tends to be worse in many scenarios. Since the optimizer applies a similar solution to the previous one, the result is therefore worse.

5.2.4 Constraint Analysis

Many of the selected FFD points were limited by the geometric constraints and their design freedom. In Fig.5.10, the constrained points are presented for the inboard up solution.



(a) Front view.

(b) Bottom view.

Figure 5.10: FFD points with active constraints of the inboard up case.

From Fig.5.10a, we notice that the limitations are mainly active in the nacelle's front face, thus showing that in an unconstrained problem, the nacelle's front face area would have been reduced even further. This is to be expected, as most flow separation occurred in this region due to the strong adverse pressure gradients on the nacelle's upper surface. Increasing the constraint limits would increase the difficulty of the solution's convergence and provide less feasible results, as the nacelle's front could become too thin. Further downstream in the lower surface in Fig.5.10b, we also notice more active constraints, namely the upper bounded FFD points, which show that the solution could improve from further reducing the volume of the nacelle.

5.2.5 Level Flight Analysis Comparison of the Nacelle Region Cases

Considering that we have optimized shapes for the nacelle that include the propeller effects with both rotation directions, as well as one without them, we can now determine if including the propeller effects is indeed a beneficial addition in the optimization process. As previously expressed, to accurately compare the performance of each configuration, we must conduct a flight leveling operation using the optimized case (without accounting for the propeller effects) from Section 5.2.1 by adding an actuator disk for both rotation directions to obtain the prescribed $C_{L_{\text{cruise}}}$. This will allow the direct comparison between the cases optimized with and without the actuator disk model, as both solutions have the same

C_L and similar flow conditions (due to similar angles of attack, the actuator disk errors are also close between both cases). The cases optimized with the actuator disk are already leveled due to the defined lift constraint. The results are shown in Tab.5.3, and the drag reductions are computed in relation to the baseline case with the corresponding propeller rotation direction.

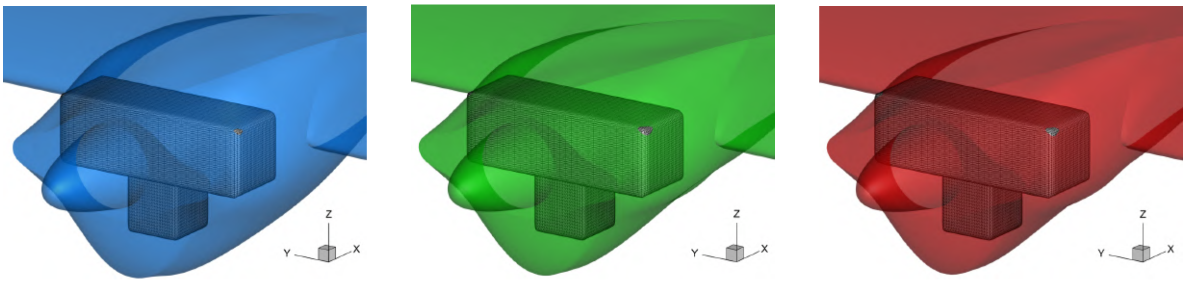
Table 5.3: Level flight results for each optimized case.

Optimized case	Added propeller configuration	α_{trim} [deg]	C_D	ΔC_D	ΔC_{M_y}
without an AD	Inboard up	1.48	0.0444	-2.84%	-1.42%
with an inboard up AD	-	1.54	0.0432	-5.47%	-0.20%
without an AD	Outboard up	1.48	0.0445	-2.84%	-1.38%
with an outboard up AD	-	1.55	0.0444	-3.06%	-0.05%

Analyzing Tab.5.3, we see that the results show minimal improvements once again for a medium-sized UAV. Including an inboard up-rotating propeller in the optimization process yields an additional improvement of 12 drag counts, while the opposite direction shows a negligible improvement of approximately 1 drag count. These results are in accordance with other studies, which quantified the benefits of the actuator disk for just the wing and a propeller [5], and also found the benefits to be minimal. In the latter study, only an improvement smaller than 1 drag count was detected when comparing two cases: one optimized while considering actuator disk forces and another without, both with design variables of twist and shape (vertical displacement of the FFD points). Our results help suggest that including propeller forces in the optimization loop yields negligible benefits, at least for the outboard up rotation direction. However, there are still some considerations to be made, namely the fact that our parametrization methods are substandard when compared to what could be implemented. Separating the nacelle's parametrization from the wing's, to be able to include twist of the wing and additional directions of shape variables for the nacelle, would increase the pertinence of the actuator disk's presence, since, as we have seen in the previous sections, the optimizer will noticeably adapt the nacelle's shape according to the rotation direction of the actuator disk. More design freedom for the nacelle could also result in better improvements for each specific case. Additionally, while stability studies of the aircraft were not considered in this work, the ΔC_{M_y} results in Tab.5.3 show small changes in relation to the leveled baseline value (especially in the cases optimized with the actuator disk), indicating that these optimized shapes will not significantly affect the initial stability condition of the AR5.

5.3 Nacelle Region with Engine Volume Constraint

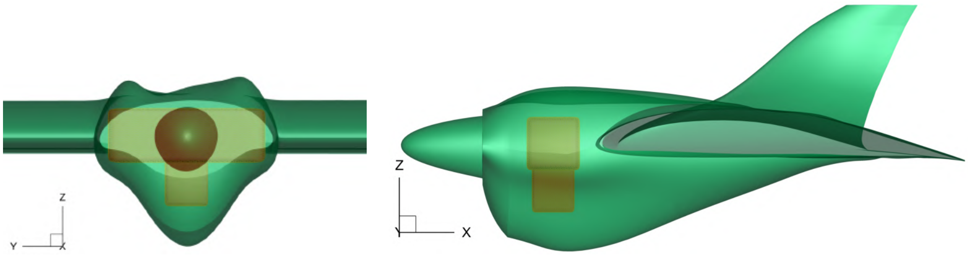
To account for room for a given geometry inside our shape in question, one method to guarantee this is by the use of triangulated surface constraints, described previously in Section 4.5.



(a) No propeller case. (b) Inboard up rotation direction case. (c) Outboard up rotation direction case.

Figure 5.11: Engine volume not fitting inside the previous optimal shape solutions.

Figure 5.11 shows that our surface geometry resembling the TEKEVER AR5 engine [96] will intersect the optimized geometry. We can perform a new optimization case including this surface constraint and analyze how the solutions will differ. Additionally, we can repeat the process of optimizing with and without the propeller and see if its presence will be more beneficial for the same problem with added constraints. However, as we can see in Fig.5.12, the resulting shapes are unusual due to the highly constrained problem. Figure 5.12 presents a solution of the optimization problem, which includes the motor surface geometry constraint and an inboard up rotating propeller.



(a) Front view. (b) Side view.

Figure 5.12: Optimized shape of the inboard up case including a surface constraint.

The problem we encounter when including this specific triangulated surface constraint is that the nacelle's sides are unnecessarily pulled up to enclose the engine surface, which does not initially fit according to the specified constraint. As seen in Section 4.4, since the intersection section of the FFD volume is fixed, the design freedom in this zone is heavily limited. The only option the optimizer has is to increase the vertical displacement of the surrounding points, to make up for the fact that the points directly above the sides of the nacelle cannot move. This will affect the results shown in Tab.5.4.

Table 5.4: Nacelle optimization results for each case with a surface constraint.

Propeller configuration	Surface constraint	Final α [deg]	ΔC_D	Func. calls	Grad. calls
No propeller	Yes	1.83	-5.76%	14	13
Inboard up	Yes	1.53	-4.81%	20	19
Outboard up	Yes	1.55	-4.59%	19	18

These results show that, despite the surface constraint, the optimizer still found good solutions that closely match the drag improvement of the previous problems. However, the restricted design freedom severely limited the solutions.

5.3.1 Separated Nacelle with Surface Constraints Problem

In this section, we explore an alternative optimization process to address the design freedom limitations that hindered the results in Sections 5.2 and 5.3. The new separated nacelle mesh is a counterpart of the existing one, which means that its geometry was obtained by closing the previous geometry (which included a half-collar and was therefore open). The mesh cells were made to be of similar size. Optimizing the nacelle separately allows for greater design freedom without compromising the wing's geometry. With this method, we can freely deform the nacelle's region nearest to the actuator disk and subsequently introduce a new, separately optimized nacelle on the wing by repeating the half-collar creation process and the optimization problems performed previously. This increase in design freedom should allow more asymmetric solutions, resulting in more custom shapes for each propeller rotation direction, thus potentially larger drag reductions. To achieve this, we generated a new FFD volume of the isolated nacelle, whose points have similar positions and quantity as the FFD region that encompassed the nacelle in the previous sections.

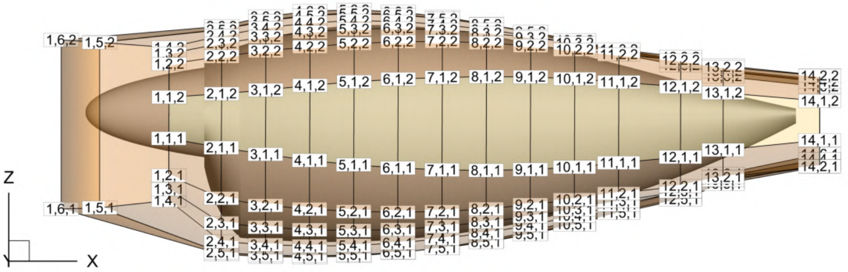


Figure 5.13: Side view of the FFD volume of the separated nacelle.

We unlock its side FFD points (for example, the $[x, 1, z]$ row in Fig.5.13, which was previously fixed on the full configuration to preserve the wing's feasible geometry) and introduce another design variable, a horizontal displacement, Δy, in our new optimization problem, shown in Tab.5.5.

Only less than half of the FFD points along the chordwise direction were selected to move, meaning only the sections of points $[1, y, z]$ until $[5, y, z]$ shown in Fig.5.13, since the latter points are supposed to be displaced subsequently in an optimization which includes the full configuration. Once again, the selected points do not include the ones that encompass the nacelle's spinner. Our chosen parameterization approach will be displacements along the y and z axis [67], since it it the most convenient for our shape, and it maintains similarity between the optimization problems.

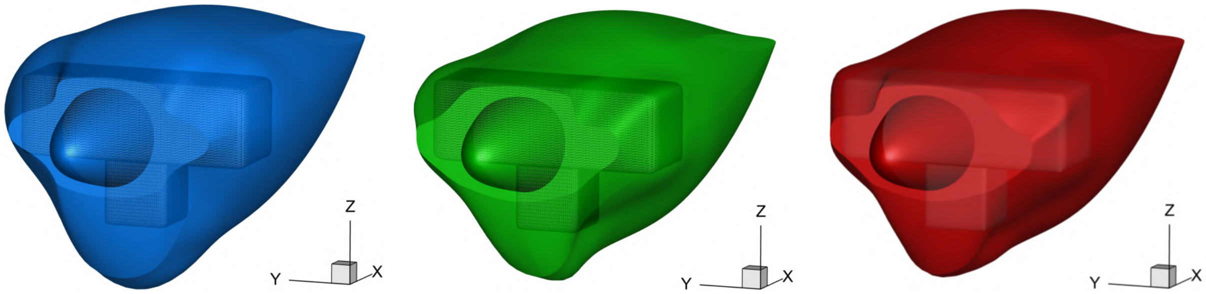
Table 5.5: Formulation of the separated nacelle optimization problem.

	Function/Variable	Description	Quantity
minimize	C_D	Drag coefficient	
w.r.t.	$-0.05 \leq \Delta y \leq 0.05$	Horizontal displacements of the FFD control points encompassing the frontal nacelle sections. [m]	90
	$-0.05 \leq \Delta z \leq 0.05$	Vertical displacements of the FFD control points encompassing the frontal nacelle sections. [m]	70
		Total design variables	**160**
subject to	$KS_{geom} \leq 0$	Aggregated minimum distance (spatial feasibility Condition 1)	1
	$L = 0$	Non-intersection (spatial feasibility Condition 2)	1
		Total constraint functions	**2**

The points in rows $[x, 1, z]$ and $[x, 11, z]$ did not include a vertical displacement to avoid infeasibility, as the nacelle tended to become pinched in its sides. Flow conditions were those of the cruise flight, with a fixed angle of attack of $1.9°$. The optimization problem represented in Tab.5.5 will be repeated for each actuator disk configuration. After deforming the nacelle in each scenario, we reinsert the nacelle's shape on the wing and recreate a half-collar mesh, essentially repeating the same steps described in the previous sections. The resulting shapes of the separate nacelles are shown in next.

5.3.2 Separated Nacelle with Surface Constraints Results

The separate nacelle optimizations provided the geometries shown in Fig.5.14, with additional views shown in Appendix D.



(a) No propeller case. (b) Inboard up rotation direction case. (c) Outboard up rotation direction case.

Figure 5.14: Optimized shapes of the separated nacelle.

The new design variable did not completely remove the frontal area, which was necessary, since it is where the airflow scoops are located to direct the airflow to the engine's radiators.

We can also deduce that including an actuator disk in a nacelle optimization problem with more satisfactory design freedom does provide a more noticeably asymmetric geometry. This is expected, since whereas in an isolated wing, the propeller is relatively far from the wing, in the full configuration, the nacelle is directly behind and very close to the propeller, meaning the vorticity has diffused and weakened less than in the isolated wing case. Comparing the results in Fig.5.14b and Fig.5.14c also shows that the inboard and outboard cases are symmetric once again. The shape of the nacelle will clearly adapt itself in relation to the tangential flow, creating smooth pressure transitions across the geometry and reducing the frontal area of the baseline geometry, thus reducing drag.

Table 5.6: Separated nacelle optimization results.

Propeller configuration	Surface constraint	ΔC_D	Func. calls	Grad. calls
No propeller	Yes	-42.30%	17	16
Inboard up	Yes	-30.89%	24	23
Outboard up	Yes	-31.03%	26	24

The optimization lasted 11 hours for the case without an actuator disk, 15 hours for the inboard up case, and 16 hours for the outboard up case. These are superior results, but the true extent of the drag reduction can only be obtained when the nacelle is simulated with the wing. Therefore, the remaining steps involve intersecting each geometry with the TEKEVER AR5 wing, generating the respective half-collar meshes, and then solving the same optimization problem defined in Tab.5.1.

5.3.3 Wing and Separated Nacelle with Surface Constraints Results

After intersecting the nacelle on the wing and recreating the half-collar, we can perform the same wing and nacelle optimization done in previous sections. These solutions will be compared to the respective optimal solutions of Section 5.2. There are similarities in the solutions, but some differences in the deformation were observed in the wing region. In the cases with the separated nacelle, the optimizer found the best solution to be shrinking the thickness of the wing near the aft end of the nacelle, as opposed to increasing and forming the lump seen in the previous solutions. This would cause the upper and lower surfaces of the wing to intersect, resulting in errors in the volume mesh that would hinder the optimization process and prevent it from converging. To prevent the mesh from intersecting itself in this optimization problem, we add a new volume constraint on the trailing edge in the problematic zone of the wing. We also set the lower bounds of this constraint to be 90% of the original thickness, and as expected, this lower bound is active in all of the final results. This lower bound was defined by [7], and it is considered to be the minimum thickness required to ensure a good balance between the structural integrity of the wing and the design freedom of the optimizer. While we use a volume constraint to guarantee no mesh errors occur, this is not strictly necessary, as the TEKEVER AR5 does not store the fuel in its wings [7]. The results of the optimization of the nacelle separately optimized and joined with the wing, are shown in Tab.5.7, but the ΔC_D gains were calculated in relation to the results of the optimized solutions in Tab.5.2.

Table 5.7: Comparison of the results of the separated nacelle on the wing with the previous ones.

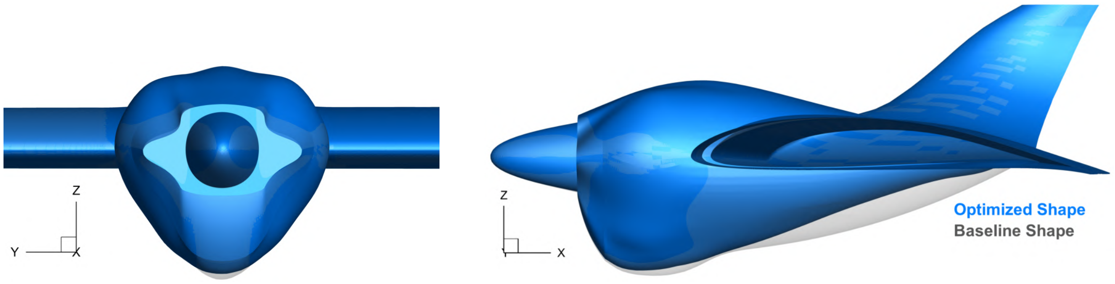
Propeller config.	Surface constraint	Final α [deg]	ΔC_D	Func. calls	Grad. calls
No propeller	Yes	1.95	-3.51%	17	17
Inboard up	Yes	1.68	-3.63%	17	16
Outboard up	Yes	1.55	-5.74%	20	19

For these optimizations, the time required for the optimization case without an actuator disk was 17 hours, with an inboard up propeller was 18 hours, and with an outboard up propeller it was 26 hours.

Not only are the obtained results in Tab.5.7 better, but they were obtained while considering the surface constraint of the engine. All of the cases provided improvements over the initial optimization results in Section 5.2, confirming that the initial design freedom was insufficient and that this method is more suitable for our optimization objective. While the cost of these optimizations seems contested, it must be considered in addition to the cost of the optimizations of the isolated nacelles, in which cases, including the propeller was more costly, requiring more gradient calls and time to converge.

5.3.4 Wing and Separated Nacelle Case without Propeller

The result of the case without propeller is presented in Fig.5.15.



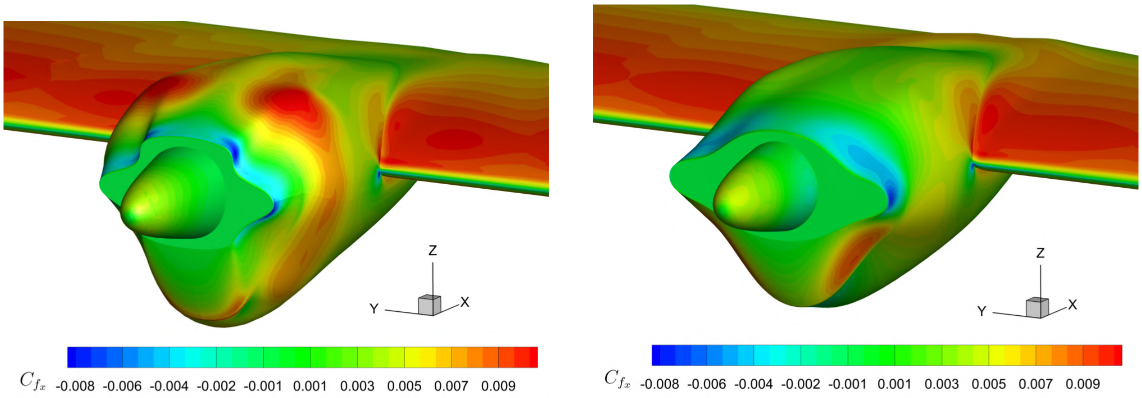
(a) Front view.

(b) Side view.

Figure 5.15: Optimized shape of the wing and separate nacelle without a propeller, with a surface constraint.

As stated before, we can find some similarities in this solution, namely, the inflation of the nacelle's midsection, the narrowing of its nose, and the reduction of volume on the lower surface, however, to a higher degree than before. In fact, we can see how much the negative skin friction region is reduced from the original case to the separated nacelle case, which shows that the previous solutions were indeed limited by the inability of the FFD points to move on the horizontal axis.

Inflating the midsection provided smoother area transitions, thus ensuring less aggravated pressure gradients, which possibly allowed for increased flow attachment. This idea is further evidenced by the values of pressure and viscous drag in each solution. The separated nacelle result presents a lower C_{D_p}, and a distribution of higher C_{D_f} around the inflated zone, as shown in Fig.5.16, showing further evidence that flow separation has been delayed.



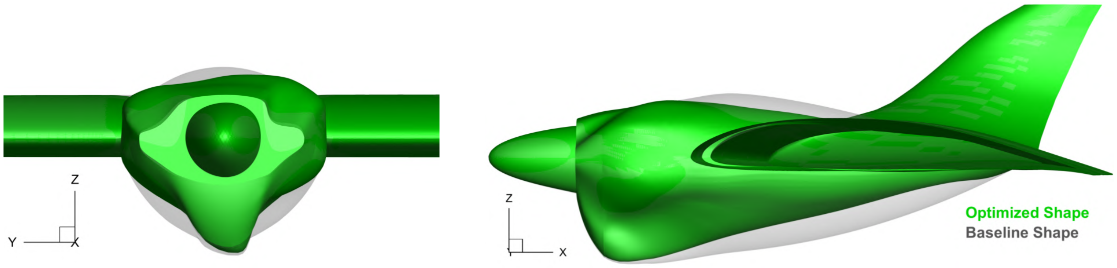
(a) Optimized solution with the separated nacelle.

(b) Optimized solution with the original nacelle.

Figure 5.16: Resulting $C_{D_{f_x}}$ distribution of the original case and the separated nacelle, without a propeller.

5.3.5 Wing and Separated Nacelle Case with Propeller

We now move on to the cases that included the propeller (actuator disk) during the optimizations. The inboard up case's result is presented in Fig.5.17.

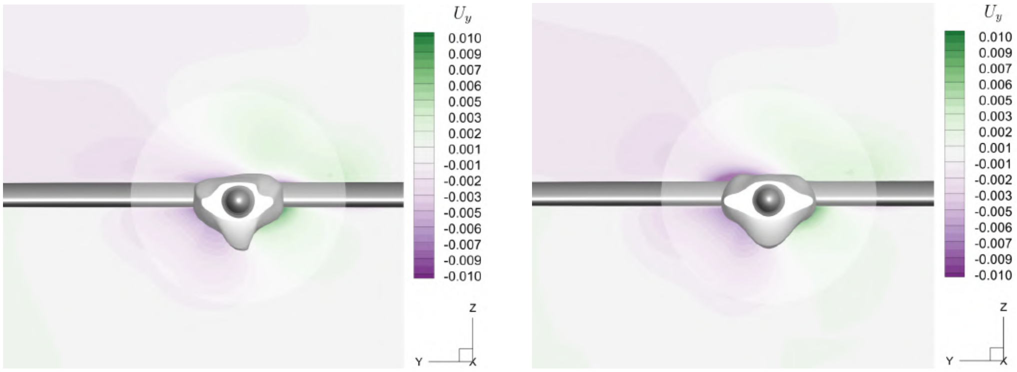


(a) Front view.

(b) Side view.

Figure 5.17: Optimized shape of the wing and separate nacelle with an inboard up rotating propeller and surface constraint.

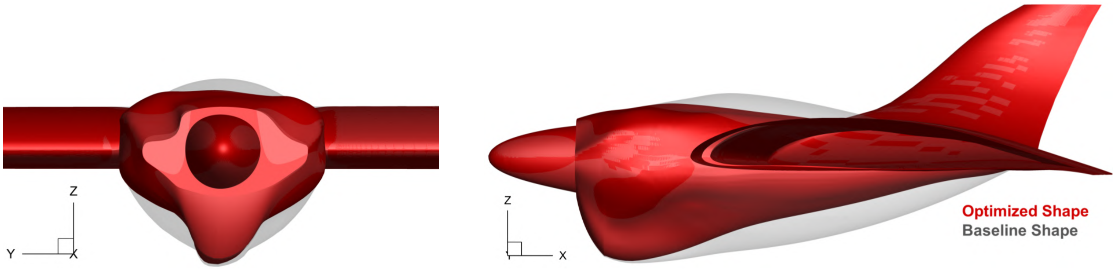
There are more differences in the cases which include the propeller. The newer solutions show reduced thickness in the region of the wing behind the nacelle, as the upper surface now has a cavity, instead of the previous slight swelling behind the raised lumps on the upper surface of the nacelle. This might occur since, with the added design freedom, the optimizer effectively has transformed the geometry of the nacelle not only to make it more streamlined and smoothed to delay flow separation, but in the presence of the propeller, it also improved the swirl recovery factor to a higher degree than before, allowing the wing region to benefit from reduced thickness to decrease induced drag by reducing the wake's width. We can visualize this once more by plotting a slice of the non-dimensional spanwise velocity component, U_y, and noticing that the newer solution in Fig.5.18a has a reduced area with lower values of U_y that accompany the inboard up rotation of the disk, when compared with the previous solution in Fig.5.18b. It is also interesting to notice that both solutions improve upon the baseline's geometry in Fig.3.35c, which is the most disturbed U_y distribution by both the nacelle and the vorticity of the disk.



(a) Optimized solution with the separated nacelle.
(b) Optimized solution with the original nacelle.

Figure 5.18: Leading edge view of contour slices filled with spanwise velocity component.

Similarly to the original nacelle optimizations, everything mentioned for the inboard up rotation case can be applied to the outboard up case, where we notice again that the optimizer's solution is a symmetric geometry to the inboard up case, in both the nacelle and the wing region behind it. The outboard up case's result is presented in Fig.5.19.



(a) Front view.
(b) Side view.

Figure 5.19: Optimized shape of the wing and separate nacelle with an outboard up rotating propeller and surface constraint.

5.3.6 Level Flight Analysis Comparison of the Wing and Separated Nacelle Cases

With all optimizations complete, we perform again a leveling operation on the solutions of the cases optimized with no actuator disk for both rotation directions. The results are presented in Tab.5.8.

Table 5.8: Level flight results for each case with the nacelle separately optimized first.

Optimized case	Added propeller configuration	α_{trim} [deg]	C_D	ΔC_D	ΔC_{M_y}
without an AD	Inboard up	1.53	0.0439	-3.94%	-0.86%
with an inboard up AD	-	1.68	0.0417	-8.75%	$+0.91\%$
without an AD	Outboard up	1.54	0.0440	-3.93%	-0.86%
with an outboard up AD	-	1.70	0.0418	-8.73%	$+0.47\%$

We find that the optimized cases with no actuator disk have improved little in relation to Tab.5.3, only 5 drag counts for both leveled rotation directions. However, for the cases optimized while accounting for the propeller forces, the improvements were more pronounced, and thus the drag reductions in relation to their isolated counterpart are more noticeable. With this parameterization method, we found that if the nacelle and wing were optimized with an actuator disk, the result would benefit from an approximately 5% total drag reduction in relation to being optimized without it, for both rotation directions. This translates to a reduction of 22 drag counts, which, if we ignore the accuracy errors described in previous sections, is a much more satisfactory result than the negligible improvements found in [5]. These results confirm our assessment: the previous optimizations were indeed limited by the inferior design freedom. Improving the parameterization of the nacelle allowed it to adapt better to the vortex of the actuator disk, thus making it a worthwhile addition to the optimization process, despite the added computational cost. Again, the ΔC_{M_y} results show small changes (less than 1%), showing that further improvements to the nacelle region will also not significantly affect the initial stability condition of the AR5.

5.4 Summary

The drag reductions of all cases presented previously are summarized in the Fig.5.20. The actuator disk seems to provide a tangible improvement for the nacelle's optimization, but the accuracy errors of the axisymmetric propeller forces might affect the true gains. The error of the angle of attack should be small due to the angle being small and similar across all solutions, but the effects of the nacelle on the inflow are not negligible. Nevertheless, it was also found that we can obtain decent drag reductions without considering the propeller forces during the optimization process, as the best case without the actuator disk shows a total reduction of 34 drag counts.

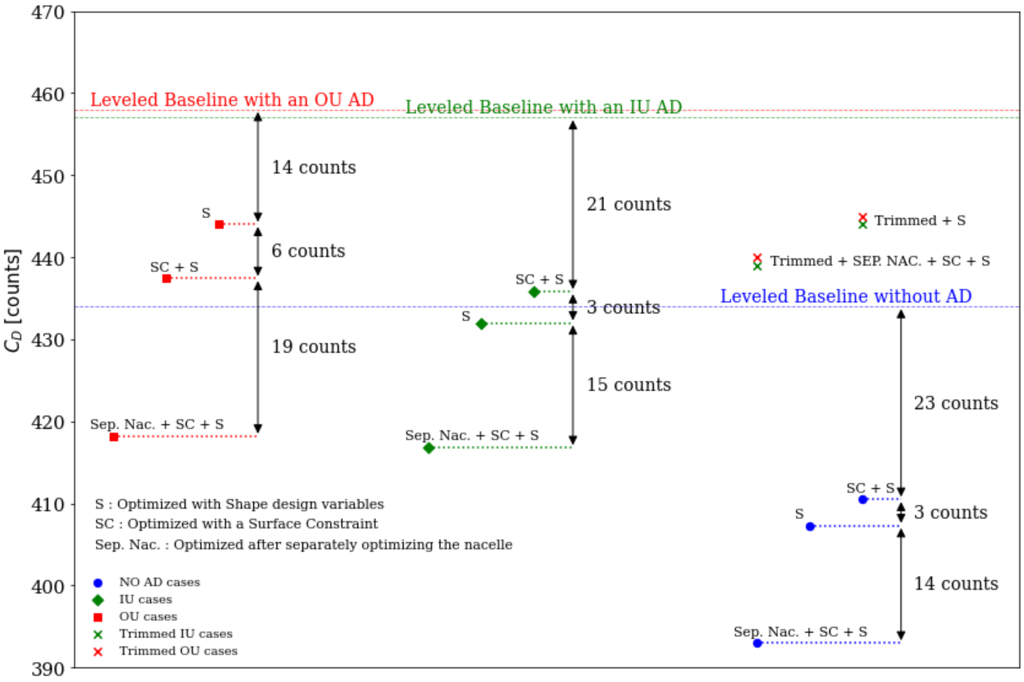


Figure 5.20: Drag improvements of all cases from the baseline values to the optimized values.

Chapter 6

Conclusions

6.1 Achievements

In this work, the MACH-Aero framework was used to perform several aerodynamic analyses and solve various aerodynamic shape optimization problems. By employing the actuator disk to model the propeller effects, we were able to provide accurate, time-averaged results using ADflow. These were carried out to determine how the forces of the propeller would change for several positions in a tractor configuration. The limitations of the current actuator disk model used in ADflow were also discussed and shown to be detrimental when performing aerodynamic analyses in which the flow velocity direction is not collinear with the propeller's thrust direction, due to the way the disk force is inserted on the flow, as other studies have pointed out. An attempt was made to improve this inaccuracy by simulating the real asymmetry of the force distributions in relation to the propeller axis, which occurs when its angle of attack is non-zero. To achieve this, the disk force distributions were altered along the azimuth of the disk in relation to its angle of attack. These alterations are adjustable, and it was found that the solution could be brought very close to experimental results. As the focus of this work is the integration of the propulsion system, a nacelle was also included on the wing. By placing the actuator disk and the nacelle on the TEKEVER AR5 wing, we were able to obtain a closer representation of the actual performance of the wing in cruise conditions.

One of the main objectives of the aerodynamic shape optimizations was to determine whether considering the propeller effects with the actuator disk would improve the optimized results. Since including the actuator disk region would imply a higher computational cost for the optimization loop, these improvements would have to be substantial. Performing tests on the nacelle was relevant because previous aerodynamic shape optimization studies had found that the actuator disk provided negligible improvements on an isolated wing. In fact, to our knowledge, this was the first instance of nacelle aerodynamic shape optimization in CFD that considered the effects of both wing and propeller. The initial parameterization of the nacelle and wing yielded results in which the actuator disk's influence on the geometry was noticeable, but the case optimized without it had similar performance. However, after implementing the surface constraint to account for the engine, the limited design freedom of the nacelle on the wing proved to be too flawed, leading to lackluster results. To address these two issues, the parameterization had to be rectified, meaning the nacelle and wing required separate parameterizations, as other studies have suggested. In this work, the parameterization while considering component intersections was also explored, but the geometry of the intersection between the nacelle and the wing was too complex for the algorithm to detect. Our alternative was to perform one optimization of the isolated nacelle separately, and then join the solution to the wing, and remake the optimization of the wing and nacelle configuration. This provided increased design freedom to the nacelle, allowing the optimization to be completed while accounting for the motor geometry, and subsequently, to be adjusted considering the wing. With this method, it was found that the optimization of the nacelle region on the TEKEVER AR5 wing would reduce drag by 22 counts for both propeller rotation directions, using only shape deformation design variables, but requiring 5 more hours of total computational time to complete the case which included the inboard up propeller, and more 14 hours for the outboard up case. Nevertheless, performing the optimization while ignoring the propeller effects still provided good results, reducing total drag by 40 counts when compared to the baseline case without the propeller forces.

6.2 Future Work

The resulting optimized shapes of the nacelle provided substantial improvements in relation to the baseline, which was a simplified version of the CAD model of the actual AR5's nacelle, facilitating the molding process for the shape optimization. Therefore, to obtain a more truthful value of the drag reductions, it would be interesting to model a mesh of the actual nacelle geometry and compare the performance of the real configuration with the optimized solutions of this study. As the optimization results of the nacelle and wing configuration that included the actuator disk showed promising improvements when compared to the ones that did not, it is pertinent to create a version of the modified actuator disk model that used the local angle of attack of the upstream cells near the actuator disk region, to be able to be more accurate when the propeller is affected by the upwash of a wing or a nacelle. This has already been made in other studies, which called it the ”enhanced actuator disk”, but not yet in ADflow (at least not from the author's recollection). The modified version in ADflow in this work helped diminish the errors caused by the propeller's angle of attack, but not those due to the upwash caused by other bodies. Additionally, this should be implemented in a way that allows the modified actuator disk to update its distribution in each new iteration during the optimization process. This further improvement would also unlock the possibility of defining the parameters of the actuator disk as design variables, for example, its position or radius. Finally, since the optimization results of our study showed good drag reductions of the isolated nacelle and wing while considering only the nacelle region of the wing, the next studies should explore the idea of performing these optimizations with more appropriate parametrization and design freedom. Even if the propeller forces are ignored during optimization, a substantial drag reduction still carries over, as seen in the level flight analysis. This means it is probably worth the effort of implementing a parametrization that takes component intersections into account, which would allow the wing to be separately optimized with global variables in conjunction with the added design freedom of the nacelle.

Bibliography

[1] E. Zanelli and H. Bödecker. Drone market report 2023-2030. *Drone Industry Insights*, pages 1–13, August 2023. URL `https://droneii.com/product/drone-market-report#download`.

[2] S. Cisneros. NASA Ames supports Joby flight tests. *NASA*, February 2023. URL `https://www.nasa.gov/aeronautics/nasa-ames-supports-joby-flight-tests/`. Accessed: 2024.

[3] Airbus. Cityairbus nextgen. *Airbus*, 2024. URL `https://www.airbus.com/en/innovation/energy-transition/hybrid-and-electric-flight/cityairbus-nextgen`. Accessed: 2024.

[4] Wisk Aero LLC. Introducing generation 6, 2025. URL `https://wisk.aero/aircraft/`. web page. Accessed: 2025.

[5] S. S. Chauhan and J. R. R. A. Martins. RANS-based aerodynamic shape optimization of a wing considering propeller-wing interaction. *Journal of Aircraft*, 58(3):497–513, May 2021. doi: 10.2514/1.C035991.

[6] TEKEVER. Tekever AR5, 2025. URL `https://www.tekever.com/models/ar5/`. Accessed: 2025.

[7] R. da Silva Gameiro. Aerodynamic design of a MAME UAV wing using high-fidelity numerical tools. Master's thesis, Instituto Superior Técnico, Portugal, November 2023.

[8] L. Prandtl. Mutual influence of wings and propeller. Technical Report 74, NACA, December 1921. An English translation of an extract from the First Report of the Göttingen Aerodynamic Laboratory.

[9] M. H. Snyder and G. W. Zumwalt. Effects of wingtip-mounted propellers on wing lift and induced drag. *Journal of Aircraft*, 6(5):392–397, 1969. doi: 10.2514/3.44076.

[10] I. Kroo. Propeller-wing integration for minimum induced loss. *Journal of Aircraft*, 23(7):561–565, 1986. doi: 10.2514/3.45344.

[11] D. P. Witkowski, A. K. H. Lee, and J. P. Sullivan. Aerodynamic interaction between propellers and wings. *Journal of Aircraft*, 26(9):829–836, 1989. doi: 10.2514/3.45848.

[12] S. S. Chauhan and J. R. R. A. Martins. Rans-based aerodynamic shape optimization of a wing with a propeller in front of the wingtip. *Aerospace*, 11(7), 2024. doi: 10.3390/aerospace11070512.

[13] J. R. R. A. Martins and A. Ning. Engineering design optimization. *Cambridge University Press*, 2021. ISBN 9781108833417.

[14] O. Pironneau. On optimum design in fluid mechanics. *Journal of Fluid Mechanics*, 64(1):97–110, 1974. doi: 10.1017/S0022112074002023.

[15] L. Veldhuis and P. Heyma. Aerodynamic optimisation of wings in multi-engined tractor propeller arrangements. *Aircraft Design*, 3(3):129–149, 2000. doi: 10.1016/S1369-8869(00)00010-0.

[16] C. Alba, A. Elham, B. J. German, and L. L. Veldhuis. A surrogate-based multi-disciplinary design optimization framework modeling wing–propeller interaction. *Aerospace Science and Technology*, 78:721–733, 2018. doi: 10.1016/j.ast.2018.05.002.

[17] S. S. Chauhan, G. P. da Silva, G. L. O. Halila, and J. R. R. A. M. João Luiz F. Azevedo. Investigating the influence of an inboard propeller's vertical offset on a trailing wing's lift and drag distributions. In *Proceedings of the 34th International Council of the Aerospace Sciences*, Florence, Italy, October 2024.

[18] M. A. S. Abdul-Kaiyoom, A. Yildirim, and J. R. Martins. *Coupled Aeropropulsive Design Optimization of an Over-Wing Nacelle Configuration*. AIAA, January 2023. doi: 10.2514/6.2023-0327. AIAA SCITECH 2023 Forum.

[19] M. Minervino, G. Andreutti, L. Russo, and R. Tognaccini. Drag reduction by wingtip-mounted propellers in distributed propulsion configurations. *Fluids*, 7:212, June 2022. doi: 10.3390/fluids7070212.

[20] A. Zanotti. Experimental study of the aerodynamic interaction between side-by-side propellers in evtol airplane mode through stereoscopic particle image velocimetry. *Aerospace*, 8:239, August 2021. doi: 10.3390/aerospace8090239.

[21] K. R. Antcliff and F. M. Capristan. *Conceptual Design of the Parallel Electric-Gas Architecture with Synergistic Utilization Scheme (PEGASUS) Concept*. AIAA, 2017. doi: 10.2514/6.2017-4001. 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference.

[22] H. Koyuncuoglu and P. He. Simultaneous wing shape and actuator parameter optimization using the adjoint method. *Aerospace Science and Technology*, 130, September 2022. doi: 10.1016/j.ast.2022.107876.

[23] M. D. Patterson and B. German. *Conceptual Design of Electric Aircraft with Distributed Propellers: Multidisciplinary Analysis Needs and Aerodynamic Modeling Development*. 52nd Aerospace Sciences Meeting, January 2014. doi: 10.2514/6.2014-0534.

[24] A. F. Botwick. High altitude long endurance aircraft configurations. Department of Mechanical and Aerospace Engineering, University of Virginia, Charlottesville, VA 22902. USA.

[25] NASA. Centurion remotely piloted solar-powered airplane, February 2024. URL `https://www.nasa.gov/reference/centurion/`. photograph source.

[26] Boeing. Phantom eye. URL `https://www.aerocontact.com/en/virtual-aviation-exhibition/product/471-phantom-eye`. photograph source. Accessed: 2025.

[27] D. Taylor. Theseus remotely piloted aircraft, September 2009. URL `https://www.nasa.gov/image-article/theseus-remotely-piloted-aircraft-2/`. photograph source. Accessed: 2025.

[28] N. Blaesser. Propeller-wing integration on the parallel electric-gas architecture with synergistic utilization scheme (PEGASUS) aircraft. January 2019. doi: 10.2514/6.2019-1809. AIAA Scitech 2019 Forum.

[29] L. Veldhuis. Review of propeller-wing aerodynamic interference. ICAS 2004-6.3.1. In *24th International Congress of the Aeronautical Sciences, Yokohama, Japan*, 2004.

[30] SKYbrary. Bell-boeing v-22, 2021. URL `https://skybrary.aero/aircraft/v22`. Accessed: 2024.

[31] Airbus. Racer, 2024. URL `https://www.airbus.com/en/innovation/innovation-ecosystem/airbus-flightlab/racer`. Accessed: 2025.

[32] S. Chauhan. Estimating propeller-induced tangential velocities by hand. In *AIAA SCITECH 2025 Forum*, January 2025. doi: 10.2514/6.2025-1265. AIAA SCITECH 2025 Forum.

[33] Airbus. A400m, 2025. URL `https://www.airbus.com/en/products-services/defence/military-aircraft/a400m`. Accessed: 2025.

[34] General Atomics Aeronautical. MQ-9A ”Reaper”, January 2025. URL `https://www.ga-asi.com/remotely-piloted-aircraft/mq-9a`. Accessed: 2025.

[35] C. A. Mader, G. K. W. Kenway, A. Yildirim, and J. R. R. A. Martins. ADflow: An open-source computational fluid dynamics solver for aerodynamic and multidisciplinary optimization. *Journal of Aerospace Information Systems*, 17(9):508–527, September 2020. doi: 10.2514/1.I010796.

[36] G. Kenway, C. Mader, P. He, and J. Martins. Effective adjoint approaches for computational fluid dynamics. *Progress in Aerospace Sciences*, June 2019. doi: 10.1016/j.paerosci.2019.05.002.

[37] S. Fu and L. Wang. RANS modeling of high-speed aerodynamic flow transition with consideration of stability theory. *Progress in Aerospace Sciences*, 58:36–59, 2013. doi: 10.1016/j.paerosci.2012.08.004.

[38] J. R. Martins. Aerodynamic design optimization: Challenges and perspectives. *Computers and Fluids*, 239:105391, March 2022. doi: 10.1016/j.compfluid.2022.105391. 239:105391.

[39] MDO Lab. MACH-Aero documentation, 2022. Introduction. Accessed: 2025.

[40] T. D. Economon, F. Palacios, S. R. Copeland, T. W. Lukaczyk, and J. J. Alonso. Su2 An open-source suite for multiphysics simulation and design. *AIAAJ*, 54(3), December 2015. doi: 10.2514/1.J053813.

[41] E. M. Papoutsis-Kiachagias, K. T. Gkaragkounis, A. I. Margetis, T. Skamagkis, V. G. Asouti, and K. C. Giannakoglou. Adjointoptimisationfoam: An openfoam-based framework for adjoint-assisted optimisation. *14th International Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control*, 2021. doi: 10.7712/140121.7960.18382.

[42] V. Q. Hoang, F. Plourde, G. Vinay, J.-B. Bracq, and C. T. Dinh. An automated framework coupling dakota with openfoam for optimizing heat transfer in corrugated channels under pulsating flow conditions. *International Journal of Heat and Fluid Flow*, 116:109912, 2025. doi: https://doi.org/10.1016/j.ijheatfluidflow.2025.109912.

[43] A. Yildirim, G. K. Kenway, C. A. Mader, and J. R. Martins. A Jacobian-free approximate Newton–Krylov startup strategy for RANS simulations. *Journal of Computational Physics*, 397:108741, 2019. doi: 10.1016/j.jcp.2019.06.018.

[44] P. Spalart and S. Allmaras. A one-equation turbulence model for aerodynamic flows. *AIAA*, 439, January 1992. doi: 10.2514/6.1992-439.

[45] D. Wilcox. *Turbulence Modeling for CFD (Third Edition) (Hardcover)*. Dcw Industries, In., 3rd edition, January 2006. ISBN: 9781928729082 / 1928729088.

[46] F. R. Menter. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA Journal*, 32:1598–1605, 1994. doi: 10.2514/3.12149.

[47] W. M. Chan and J. L. Steger. Enhancements of a three-dimensional hyperbolic grid generation scheme. *Applied Mathematics and Computation*, 51(2):181–205, 1992. doi: 10.1016/0096-3003(92)90073-A.

[48] L. Eça, F. Pereira, and G. Vaz. Viscous flow simulations at high Reynolds numbers without wall functions: Is y+$\approx$1 enough for the near-wall cells? *Computers & Fluids*, 170:157–175, 2018. doi: 10.1016/j.compfluid.2018.04.035.

[49] ICAO. U.S. standard atmosphere, 1976. Technical report, National Oceanic and Atmospheric Administration, National Aeronautics and Space Administration, United States Air Force, October 1976.

[50] MDO Lab. MACH-Aero documentation, 2022. URL `https://mdolab-adflow.readthedocs-hosted.com/en/latest/solvers.html#interpreting-output`. ADflow Solvers. Accessed: 2025.

[51] M. Drela. Xfoil: An analysis and design system for low reynolds number airfoils. volume 54, June 1989. ISBN 978-3-540-51884-6. doi: 10.1007/978-3-642-84010-4_1. Accessed: 2025.

[52] H. Y. Mark Drela. *XFOIL 6.9 User Primer*. MIT, November 2001. URL `https://web.mit.edu/aeroutil_v1.0/xfoil_doc.txt`.

[53] F. S. Cunha. Helicopters - blade element theory. Technical report, Instituto Superior Técnico, 2024. Course Notes.

[54] R. Anderson, J. Willis, J. Johnson, A. Ning, and R. W. Beard. *A Comparison of Aerodynamics Models for Optimizing the Takeoff and Transition of a Bi-wing Tailsitter*. AIAA SciTech Forum, January 2021. doi: 10.2514/6.2021-1008.

[55] K. Moore and A. Ning. Takeoff and performance trade-offs of retrofit distributed electric propulsion for urban transport. *Journal of Aircraft*, 56:1–13, august 2019. doi: 10.2514/1.C035321.

[56] A. Ning. A simple solution method for the blade element momentum equations with guaranteed convergence. *Wind Energy*, 17, July 2013. doi: 10.1002/we.1636.

[57] M. Drela and H. Youngren. Xrotor, February 2022. URL `https://web.mit.edu/drela/Public/web/xrotor/`. Accessed: 2025.

[58] O. Bergmann, F. Götten, C. Braun, and F. Janser. Comparison and evaluation of blade element methods against RANS simulations and test data. *CEAS Aeronautical Journal*, 13:535–557, April 2022. doi: 10.1007/s13272-022-00579-1.

[59] T. Sinnige, T. Stokkermans, N. van Arnhem, and L. Veldhuis. Aerodynamic performance of a wingtip-mounted tractor propeller configuration in windmilling and energy-harvesting conditions. In *AIAA Aviation 2019 Forum*, June 2019. doi: 10.2514/6.2019-3033.

[60] C. E. Lan. A quasi-vortex-lattice method in thin wing theory. *Journal of Aircraft*, 11(9):518–527, 1974. doi: 10.2514/3.60381.

[61] M. Patterson and B. German. Conceptual design of electric aircraft with distributed propellers: Multidisciplinary analysis needs and aerodynamic modeling development. January 2014. ISBN 978-1-62410-256-1. doi: 10.2514/6.2014-0534.

[62] P. Aref, M. Ghoreyshi, A. Jirasek, M. J. Satchell, and K. Bergeron. Computational study of propeller–wing aerodynamic interaction. *Aerospace*, 5(3), 2018. doi: 10.3390/aerospace5030079.

[63] J. N. Sørensen and W. Z. Shen. Numerical modeling of wind turbine wakes. *Journal of Fluids Engineering, Transactions of the ASME*, 124(2):393 – 399, 2002. doi: 10.1115/1.1471361.

[64] T. C. A. Stokkermans, N. van Arnhem, T. Sinnige, and L. L. M. Veldhuis. Validation and comparison of RANS propeller modeling methods for tip-mounted applications. *AIAA Journal*, 57(2):566–580, 2019. doi: 10.2514/1.J057398.

[65] M. Hoekstra. A RANS-based analysis tool for ducted propeller systems in open water condition. *International Shipbuilding Progress*, 53:205–227, January 2006. doi: 10.3233/SHP-2006-isp011.

[66] MDO Lab. MACH-Aero documentation, 2022. Grid Refinement Study. Accessed: 2025.

[67] L. Pinheiro. Aerodynamic shape optimization of UAV fuselage and fairings with payload integration. Master's thesis, Instituto Superior Técnico, Portugal, July 2025.

[68] L. Veldhuis. *Propeller Wing Aerodynamic Interference*. PhD thesis, TU Delft, June 2005. Netherlands.

[69] S. S. Chauhan. *Optimization Studies for Aircraft Considering Propeller-Wing Interaction*. PhD thesis, University of Michigan, https://dx.doi.org/10.7302/32, 2020. USA.

[70] J. A. Cole, T. Krebs, D. Barcelos, and G. Bramesfeld. Influence of propeller location, diameter, and rotation direction on aerodynamic efficiency. *Journal of Aircraft*, 58(1):63–71, 2021. doi: 10.2514/1.C035917. URL `https://doi.org/10.2514/1.C035917`.

[71] M. Drela. *QPROP Propeller/Windmill Analysis and Design*. MIT, Boston, 2007. URL `http://web.mit.edu/drela/Public/web/qprop/`. Software Package. Accessed: 2025.

[72] R. Nederlof, R. Kooij, L. L. Veldhuis, and T. Sinnige. Contribution of swirl recovery to the induced drag of a propeller-wing system – a parametric study. In *AIAA AVIATION 2023 Forum*. AIAA, 2023. doi: 10.2514/6.2023-3543.

[73] L. Veldhuis and S. Nebiolo. A propeller-integration study comparing experimental data with numerical flow solutions based on the Navier-Stokes equations. In *International Congress of the Aeronautical Sciences*, 2000.

[74] A. Gomariz-Sancha, M. Maina, and A. J. Peace. Analysis of propeller-airframe interaction effects through a combined numerical simulation and wind-tunnel testing approach. In *53rd AIAA Aerospace Sciences Meeting*. AIAA, 2015. doi: 10.2514/6.2015-1026.

[75] M. Hansen. *Aerodynamics of Wind Turbines*. Earthscan, 3 edition, May 2015. doi: 10.4324/9781315769981.

[76] MDO Lab. MACH-Aero documentation, 2022. URL `https://mdolab-mach-aero.readthedocs-hosted.com/en/latest/machAeroTutorials/overset_theory.html`. Overset Mesh Theory. Accessed: 2025.

[77] Y. Lee and J. Baeder. Implicit hole cutting - a new approach to overset grid connectivity. In *16th AIAA Computational Fluid Dynamics Conference*. AIAA, 2003. doi: 10.2514/6.2003-4128.

[78] Y. Qiu, J. Bai, and L. Qiao. Aerodynamic effects of wing-mounted engine nacelle on high-lift configuration of turboprop airliner. *Journal of Aircraft*, 55(3):1082–1089, 2018. doi: 10.2514/1.C034529.

[79] R. Duivenvoorden, N. Suard, T. Sinnige, and L. L. Veldhuis. Experimental investigation of aerodynamic interactions of a wing with deployed fowler flap under influence of a propeller slipstream. In *AIAA AVIATION 2022 Forum*. AIAA, 2022. doi: 10.2514/6.2022-3216.

[80] M. Jiang, R. Machiraju, and D. Thompson. Detection and visualization of vortices. *The visual handbook*, December 2005. doi: 10.1016/B978-012387582-2/50016-2.

[81] J. C. F. Pereira. Aerodinâmica III. Volume 2, Instituto Superior Técnico, Portugal, August 2021. Course Notes.

[82] L. L. M. Veldhuis, T. Stokkermans, T. Sinnige, and G. Eitelberg. Analysis of swirl recovery vanes for increased propulsive efficiency in tractor propeller aircraft. In *30th Congress of the International Council of the Aeronautical Sciences*, 2016.

[83] A. C. Marta. Aircraft optimal design. Technical report, Instituto Superior Técnico, Portugal, 2023. MSc Course Notes.

[84] P. M. V. Rodrigues. Efficient aerodynamic optimization of aircraft wings. Master's thesis, Instituto Superior Técnico, Portugal, 2018.

[85] Y. Yu, P. Lyu, Z. xu, and J. Martins. On the influence of optimization algorithm and initial design on wing aerodynamic shape optimization. *Aerospace Science and Technology*, 75, February 2018. doi: 10.1016/j.ast.2018.01.016.

[86] D. Kraft. A software package for sequential quadratic programming. Technical Report DFVLR-FB 88-28, Wiss. Berichtswesen d. DFVLR, 1988.

[87] J. R. R. A. Martins. Sensitivity analysis. Technical report, Standford University, USA, 2001. AA222 - Multidisciplinary Design Optimization.

[88] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, pages 233–260, 1988. ISSN 1573-7691. doi: 10.1007/BF01061285.

[89] A. Yildirim, C. A. Mader, and J. R. Martins. A surface mesh deformation method near component intersections for high-fidelity design optimization. *Engineering with Computers*, 38:1393 – 1425, 2021. doi: 10.1007/s00366-020-01247-w.

[90] H. Hajdik, A. Yildirim, and J. Martins. Aerodynamic shape optimization of filleted intersections with surface mesh deformation. In *34th Congress of the International Council of the Aeronautical Sciences*, September 2024.

[91] N. R. Secco, J. Jasa, G. K. Kenway, and J. R. R. A. Martins. Component-based geometry manipulation for aerodynamic shape optimization with overset meshes. In *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. American Institute of Aeronautics and Astronautics, Inc., 2017. doi: 10.2514/6.2017-3327.

[92] MDO Lab. MACH-Aero documentation, 2022. URL `https://mdolab-pygeo.readthedocs-hosted.com/en/latest/DVGeometry.html`. pyGeoAPI. Accessed: 2025.

[93] MDO Lab. MACH-Aero documentation, 2022. URL `https://mdolab-pygeo.readthedocs-hosted.com/en/latest/advanced_ffd.html`. Advanced FFD Geometry. Accessed: 2025.

[94] MDO Lab. MACH-Aero documentation, 2022. URL `https://mdolab-pygeo.readthedocs-hosted.com/en/latest/DVConstraints.html`. Design Variable Constraints. Accessed: 2025.

[95] A. K. C. Sullivan. PyVista: 3D plotting and mesh analysis through a streamlined interface for the visualization toolkit (vtk). *Journal of Open Source Software*, 4(37), 2019. doi: 10.21105/joss.01450.

[96] ZDZ ENGINES. ZDZ 250B2-J CHAMPION, 2014. web page.

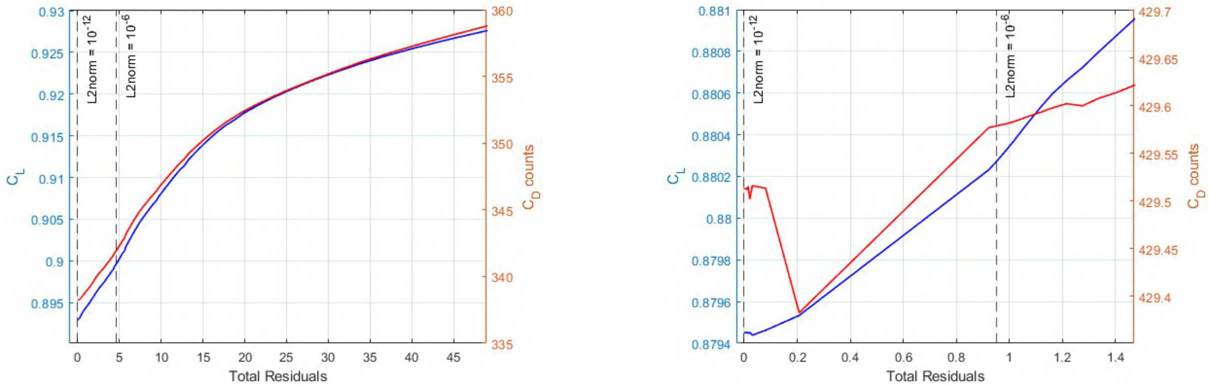
[97] H. W. Kuhn and A. W. Tucker. Non-linear programming. In I. J. Neyman, editor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, page 481–493, Berkeley, CA, January 1951. University of California Press.

[98] MDO Lab. MACH-Aero private documentation, 2022. URL `https://mdolab-adflow.readthedocs-hosted.com/en/latest/solvers.html#interpreting-output`. Pre-run checklists and troubleshooting. Accessed: 2025.

Appendix A

Solver Tolerance Study

The convergence solutions with tuned solver settings for the isolated wing and the wing with an actuator disk are presented in Fig.A.1 and Fig.A.2, respectively. It was found through experimentation that the solver parameters that provided the best performance after decreasing the `L2Convergence` value to 10^{-12} were: switching the ANK solver to a second order accurate implicit formulation at a residuals (SANK) decline of 10^{-4}, and for the last stages of convergence, switching to the NK solver when the residual decay is 10^{-8}.



(a) L2norm convergence the L1 overset mesh.

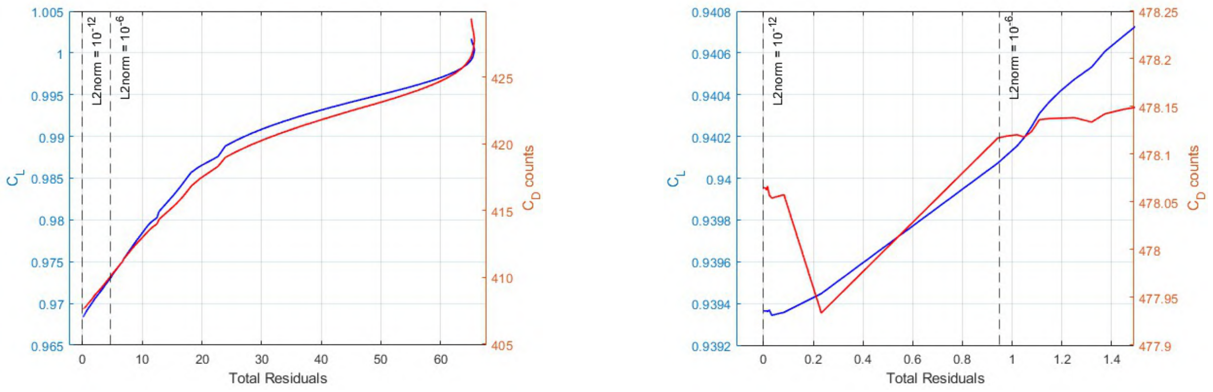
(b) L2norm convergence of the L3 overset mesh.

Figure A.1: Convergence study for the isolated wing.

Table A.1: Grid study values with L2-norm equal to 10^{-12}.

Level	Actuator Disk	Computation Time (s)	Total Linear Iter.	Final Total Residuals	C_L
L1	Off	33002	15390	3.794×10^{-6}	0.89294
L1	On	34288	15377	3.861×10^{-6}	0.96839
L3	Off	234	2020	5.219×10^{-7}	0.87945
L3	On	236	2018	5.386×10^{-7}	0.93936

As we can see from Fig.A.2, a tighter tolerance of L2-norm equal to 10^{-12} is not worth the additional



(a) L2norm convergence of the L1 overset mesh.

(b) L2norm convergence of the L3 overset mesh.

Figure A.2: Convergence study for the wing and actuator disk.

computational effort at any of the mesh refinement levels. Therefore, we set the L2-norm equal to 10^{-8}. The computation time will increase for all cases, and the accuracy gain is negligible. The results in Tab.A.1 also show that more complex flows (with an actuator disk) typically take longer to achieve accurate steady states in RANS simulations.

Appendix B

Code Implementation

In this appendix, we document how the variance of the actuator disk's force distribution was implemented to change along the azimuthal angle of the disk, with respect to the propeller angle of attack. Only the modified version of `simpleProp` is shown here, since the other is very similar.

```fortran
if (region%actType == 'AngledProp') then
! Compute cross product for tangential vector and normalize
v1 = xCen - axis2
v2 = axisVec

sss(1) = (v1(2)*v2(3) - v1(3)*v2(2))
sss(2) = (v1(3)*v2(1) - v1(1)*v2(3))
sss(3) = (v1(1)*v2(2) - v1(2)*v2(1))
sss = sss / sqrt(sss(1)**2 + sss(2)**2 + sss(3)**2)

region%cellTangentials(:, region%nCellIDs) = sss

! Cosine of the tangential vector with the propeller's longitudinal axis
! to employ azimuthal changes to the forces of the actuator disk
cosineProp = (propLongitudinalAxis(1)* sss(1) + &
              propLongitudinalAxis(2)* sss(2) + &
              propLongitudinalAxis(3)* sss(3))

! Since the force is maximized when the prop longitudinal axis
! makes 90 degs with the tangential vector, it is easier to use the sine
! of the angle between the two vectors. This also gives the absolute value.

sineProp = sqrt(one - cosineProp**2)

! Compute the dot product and subtract to get the radius
dotP = v1(1)*v2(2) + v1(2)*v2(2) + v1(3)*v2(3)
radVec = v1 - dotP * axisVec
region%cellRadii(region%nCellIDs) = &
    sqrt(radVec(1)**2 + radVec(2)**2 + radVec(3)**2)

! Compute unscaled thrust and swirl forces
```

```
if (region%cellRadii(region%nCellIDs) < spinnerRadius) then

    Ftmp = zero
    region%thrustVec(:, region%nCellIDs) = Ftmp * axisVec
    Swtmp = zero
    region%swirlVec(:, region%nCellIDs) = Swtmp &
                     * region%cellTangentials(:, region%nCellIDs)

else if (region%cellRadii(region%nCellIDs) < innerZeroThrustRadius) then

    rHat = ((region%cellRadii(region%nCellIDs) - innerZeroThrustRadius) &
          / (propRadius - innerZeroThrustRadius))
    fact = rootDragFactor / propRadius
    fact2 = rHat**mDistribParam * abs(one - rHat)**nDistribParam &
           / (two * pi * region%cellRadii(region%nCellIDs))

    if (alpha .ge. 0.0) then
    ! positive alpha ->  up-going blade will experience
    ! smaller effective angles of attack -> up side lower thrust

         if (region%cellTangentials(3, region%nCellIDs) < 0.0 ) then
         !we are in the downwash region
            Ftmp = vol(i, j, k) * fact * fact2 * (1 + angledpropCorr * sineProp)
            Swtmp = zero
         else
         !we are in the upwash region
            Ftmp = vol(i, j, k) * fact * fact2 * (1 - angledpropCorr * sineProp)
            Swtmp = zero
         end if

    else
    ! negative alpha -> up-going blade will experience
    ! larger effective angles of attack -> up side higher thrust

        if (region%cellTangentials(3, region%nCellIDs) < 0.0 ) then
        !we are in the downwash region
            Ftmp = vol(i, j, k) * fact * fact2 * (1 - angledpropCorr * sineProp)
            Swtmp = zero
        else
        !we are in the upwash region
            Ftmp = vol(i, j, k) * fact * fact2 * (1 + angledpropCorr * sineProp)
            Swtmp = zero
        end if

   end if

   thrustSum = thrustSum + Ftmp
   region%thrustVec(:, region%nCellIDs) = Ftmp * axisVec
   region%swirlVec(:, region%nCellIDs) = Swtmp &
```

```
                     * region%cellTangentials(:, region%nCellIDs)

else

   rHat = ((region%cellRadii(region%nCellIDs) - innerZeroThrustRadius) &
          / (propRadius - innerZeroThrustRadius))
   fact = one / propRadius
   fact2 = rHat**mDistribParam * abs(one - rHat)**nDistribParam &
           / (two * pi * region%cellRadii(region%nCellIDs))
   swirlFact2 = distribPDfactor / pi / region%cellRadii(region%nCellIDs) &
                * propRadius * swirlFact
   FTang = swirlFact2 * fact * fact2

    if (alpha .ge. 0.0) then
    ! positive alpha ->  up-going blade will experience
    ! smaller effective angles of attack -> up side lower thrust

        if (region%cellTangentials(3, region%nCellIDs) < 0.0 ) then
        !we are in the downwash region
            Ftmp = vol(i, j, k) * fact * fact2 * (1 + angledpropCorr * sineProp)
            Swtmp = vol(i, j, k) * FTang * (1 + angledpropCorr * torquefactor * sineProp)
        else
        !we are in the upwash region
            Ftmp = vol(i, j, k) * fact * fact2 * (1 - angledpropCorr * sineProp)
            Swtmp = vol(i, j, k) * FTang * (1 - angledpropCorr * torquefactor * sineProp)
        end if

    else
    ! negative alpha -> up-going blade will experience
    ! larger effective angles of attack -> up side higher thrust

        if (region%cellTangentials(3, region%nCellIDs) < 0.0 ) then
        !we are in the downwash region
            Ftmp = vol(i, j, k) * fact * fact2 * (1 - angledpropCorr * sineProp)
            Swtmp = vol(i, j, k) * FTang * (1 - angledpropCorr * torquefactor * sineProp)
        else
        !we are in the upwash region
            Ftmp = vol(i, j, k) * fact * fact2 * (1 + angledpropCorr * sineProp)
            Swtmp = vol(i, j, k) * FTang * (1 + angledpropCorr * torquefactor * sineProp)
        end if

    end if
    thrustSum = thrustSum + Ftmp
    region%thrustVec(:, region%nCellIDs) = Ftmp * axisVec
    swirlSum = swirlSum + Swtmp
    region%swirlVec(:, region%nCellIDs) = Swtmp &
                                * region%cellTangentials(:, region%nCellIDs)
end if
```

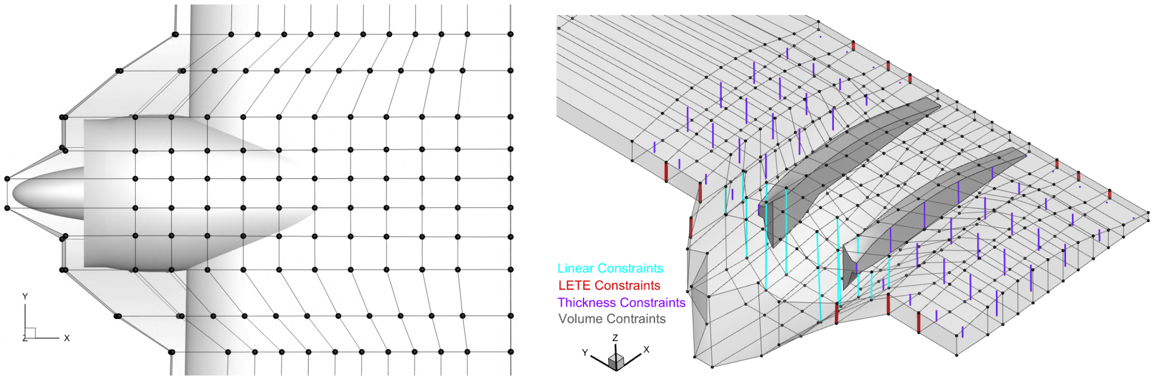
```
end if
```

Listing B.1: Code snippet of the modified actuator disk in the file actuatorregion.f90

Appendix C

Alternative Attempts of the FFD Box

The previously described shortcomings of the FFD box were initially mitigated by imposing geometric constraints near the complicated intersection region and by also modifying the FFD box itself. One initial strategy involved reducing the skew angle in the intersection region by creating an FFD box that connected the nacelle to the wing with a less sudden change in chordwise size, as shown in Fig.C.1a. This method required additional geometric constraints in the FFD points near the leading edge, as it is no longer located within the first spanwise row of the volume.



(a) FFD volume.

(b) Geometric constraints of the respective FFD volume.

Figure C.1: Initial FFD volume definition strategy.

The constraint used is `addLinearConstraintsShape`, shown in Fig.C.1b, which allows defining a linear equation constraint to the selected shape variables. The equation can be defined in such a way that recreates the `addLeTeConstraints` but for other FFD points besides the ones defining the vertical front and back faces of the FFD volume. However, we see in Fig.C.2 that applying these constraints does not fully eliminate the shear twist and only diffuses the contortion problem on the wing across a larger area. Therefore, we deduce that reducing the skewness of the FFD volume on the leading edge of the wing does improve the results, since the contortion is less pronounced in Fig.C.2 when compared with Fig.4.6a. However, this new method still does not provide the desired behavior of the geometric design variables.

An additional attempt to eliminate contortions was made. Instead of trying to dissipate the chordwise

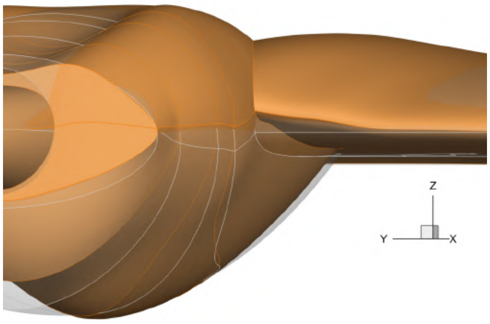
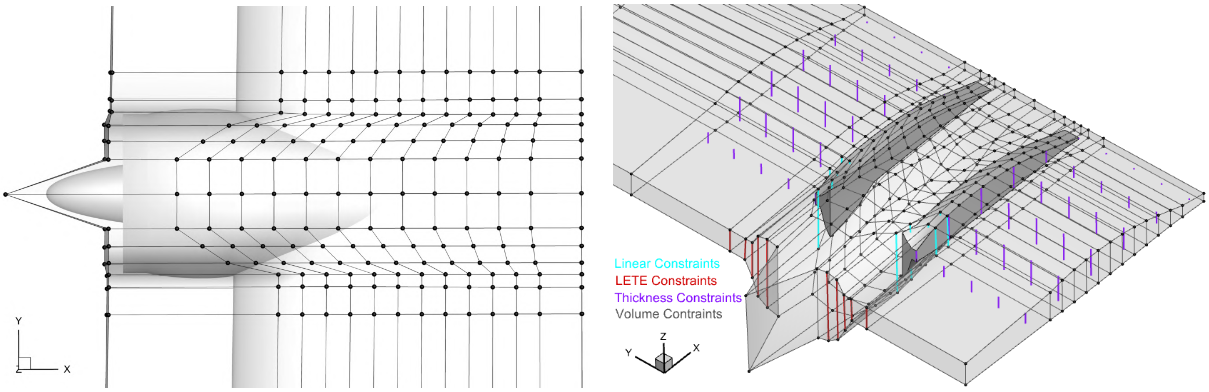


Figure C.2: FFD volume results.

length of the FFD across a larger area, the FFD skew angle can be removed in the leading edge by setting the FFD's front vertical faces all at the same length as the nacelle, as shown in Fig.C.3a. The skewed volume is contained inside the nacelle region, supposedly without affecting the intersection.



(a) FFD volume.

(b) Geometric constraints of the respective FFD volume

Figure C.3: Additional FFD volume definition attempt.

The geometric constraints used in this method were similar to the previously described ones, except the FFD indexes had to be accordingly changed to affect the intersection points. As expected, with this method, the contortion effects are no longer present in the leading edge. However, they are still noticeable where the FFD is skewed, as shown in Fig.C.4.

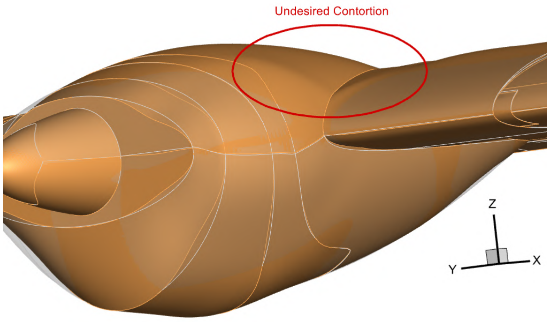


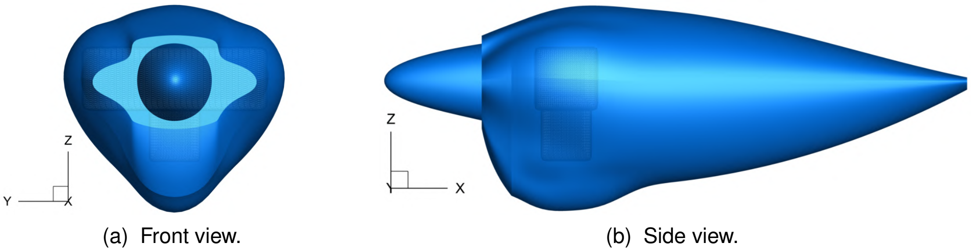
Figure C.4: Additional FFD volume attempt results.

Additionally, the parametrization will be worse, since the most important target for shape optimization, which is the nacelle's front, has fewer degrees of freedom, as shown in Fig.C.3a, thus sacrificing the ability of deformation where it would be more beneficial. Therefore, this FFD volume strategy still did not provide acceptable results. Considering all of the previous ideas, it was decided that any fully embedding FFD volume would either lead to contortions or, if unconstrained, shear twist in the leading edge connecting the wing and nacelle. For these reasons, the final adopted strategy, as shown in Fig.4.7a, locks the intersection points. Since these volume sections are fixed in place, we can also take advantage of this to position the chordwise size difference jump in this location. The FFD volume will be highly skewed in this region, but it will not affect the final result as much as the other solutions, since it can not move. This enables a more accurate parametrization of the nacelle and the wing surrounding it, while addressing the previously expressed problems.

Appendix D

Optimized Separate Nacelles Views

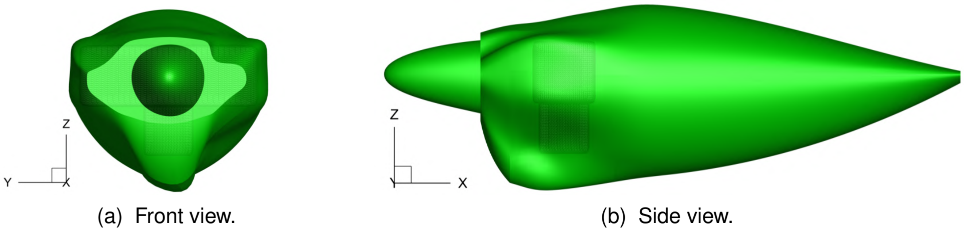
The shape of the separated nacelle optimized without the actuator disk is shown in Fig.D.1.



(a) Front view. (b) Side view.

Figure D.1: Optimized shape of the separate nacelle without a propeller, with a surface constraint.

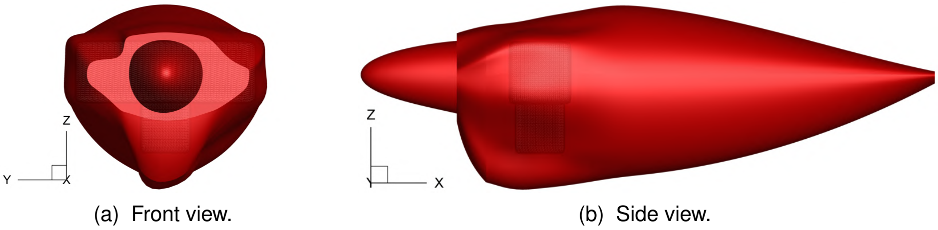
The shape of the separated nacelle optimized with an inboard up actuator disk is shown in Fig.D.3.



(a) Front view. (b) Side view.

Figure D.2: Optimized shape of the separate nacelle with inboard up rotating propeller and a surface constraint.

The shape of the separated nacelle optimized with an outboard up actuator disk is shown in Fig.D.3.



(a) Front view. (b) Side view.

Figure D.3: Optimized shape of the separate nacelle with outboard up rotating propeller and a surface constraint.