Politecnico di Milano

SCHOOL OF INDUSTRIAL AND INFORMATION ENGINEERING
Master of Science – Aeronautical Engineering

Enhanced General Procedures for Optimal Design of Aircraft Propellers

Supervisor
Prof. Lorenzo Trainelli

Co-Supervisor
Prof. Carlo E. D. Riboldi

Candidate
Sergio Stefanelli – 900057

Academic Year 2019 – 2020

Gutta cavat lapidem non vi, sed saepe cadendo
— Tito Lucrezio Caro

Ringraziamenti

Essere arrivato fin qui vuol dire per me essere giunto all'apice di un percorso lungo, faticoso e, allo stesso tempo, gratificante. Prima di procedere con la trattazione, vorrei dedicare qualche riga a tutti coloro che hanno contribuito, per via diretta o indiretta, alla stesura di questa tesi.

Innanzitutto, sono profondamente grato ai miei due relatori, prof. Trainelli e prof. Riboldi, per avermi seguito sin dai primi giorni di lavoro e per i loro preziosi consigli.

Il ringraziamento più sincero va poi alla mia famiglia, ai miei genitori e le mie sorelle, per il loro sostegno costante e instancabile in tutti questi anni e per aver sempre creduto in me, soprattutto nei momenti in cui persino a me risultava difficile farlo. Senza di voi non ce l'avrei mai fatta.

Come non menzionare poi tutti gli amici che hanno riempito questi anni universitari di esperienze indimenticabili che porterò sempre con me.

Seba, Lollo, Macchia, Memma, Marti, Rocco, Pietro, Aldous, Virtu, Jonny, Vitto, Nicco e tutti gli altri sparsi per l'Italia: grazie per tutte le feste, cene, serate al Nepentha, i viaggi, i capodanni, le risate fino alle lacrime... Siete stati la mia seconda famiglia per un fuori sede come me.

Quisiera agradecer también a Jaime De Zubeldia por haber estado tan atento y disponible y cuyos consejos fueron fundamentales para coger el camino correcto desde el principio del estudio.

Asimismo, quisiera dedicar unas palabras a mio tío Toni, quien siempre ha estado a mi lado apoyándome infatigablemente.

Infine, una dedica va al mio amico ed ex compagno di studi Pit, che ora non c'è più ma sarà sempre presente nei cuori di tutti noi.

Sommario

In tempi recenti, gran parte della comunità scientifica ha mostrato sempre più interesse nello sviluppo di modelli e metodi per progettare correttamente, a seconda dei requisiti, uno dei principali - nonché uno dei primi - strumenti usati nella propulsione aeronautica: l'elica.

In questo senso, il presente elaborato ha come oggetto lo sviluppo di una metodologia che permette di svolgere un progetto ottimo e analisi di prestazione di eliche, prendendo le mosse da quella precedentemente sviluppata da De Zubeldia all'interno della sua tesi di laurea magistrale, nell'intento - andato in seguito a buon fine - di migliorarla ed ampliarne le possibilità. Il calcolo delle prestazioni viene svolto applicando la teoria del momento dell'elemento di pala (BEMT), la quale richiede la disponibilità dei coefficienti aerodinamici bidimensionali del profilo adottato in funzione dell'angolo di incidenza, numero di Reynolds e numero di Mach. Ciò è stato svolto utilizzando il software XFOIL e dati sperimentali.

Rispetto allo sviluppo precedente, il database aerodinamico, inizialmente contenente solo i dati relativi al profilo NACA0012, è stato esteso anche ai profili NACA16-012, RAF-6 e Clark-Y. Inoltre, grazie ad uno studio di affidabilità di XFOIL tramite il confronto con i risultati provenienti da CFD per mezzo delle librerie di OpenFOAM, è stato possibile utilizzare il software di M. Drela fino a un numero di Mach pari a 0.6, rispetto al limite usato da De Zubeldia di 0.5.

In aggiunta, la formulazione è stata modificata in modo tale da includere gli effetti rotazionali nei coefficienti aerodinamici, i quali, specie ad alti angoli di incidenza, risentono molto di tali effetti e possono variare in maniera considerevole. Per far ciò, dopo aver compiuto una scrupolosa analisi sullo stato dell'arte, è stata applicata la correzione proposta da P. Chaviaropoulos e M. Hansen.

In seguito, questa nuova formulazione è stata implementata in ambiente MATLAB® presentando due modalità principali di funzionamento: (i) calcolo delle prestazioni per una geometria data e (ii) ottimizzazione della geometria. In questo modo, è stato possibile svolgere diversi tipi di studi numerici. Innanzitutto, è stata svolta un'analisi parametrica delle costanti usate nel sopracitato modello correttivo, che si è dimostrato essere molto preciso nel rettificare gli andamenti non fisici nei grafici delle prestazioni presenti nello studio di De Zubeldia soprattutto nella zona a basso rapporto di avanzamento dell'elica (J), che corrisponde per l'appunto ad alti angoli di incidenza ed effetti rotazionali elevati. Successivamente, sono stati compiuti svariati cicli di ottimizzazione che hanno dimostrato come, a parità di *guess* iniziali e funzioni costo, il software derivante dalla nuova formulazione ottenga in generale delle corde minori lungo l'apertura rispetto alla sua versione precedente. Inoltre, è stata simulata la presenza di un meccanismo di regolazione del passo dell'elica, dimostrando che, grazie ad esso, l'efficienza dell'elica supera una certa soglia in un intervallo di J molto più ampio.

Nel complesso, il presente studio si è rivelato un successo. Tutti gli obiettivi iniziali sono stati raggiunti, rappresentando un grosso passo in avanti: la formulazione è più completa e il software che ne deriva è diventato più versatile e preciso. Potenziali miglioramenti vanno dall'espansione del database aerodinamico ad altri profili allo sviluppo di *sub-routines* strutturali e acustiche.

Abstract

Scientific community has shown, in the last years, more and more interest on developing models and methods to design, basing on the project requirements, one of the most important - as well as one of the first - aeronautical propulsion devices: the propeller.

In this sense, the objective of the present analysis is to develop a formulation that would help in the early design phase of a propeller, starting from a previous research performed by De Zubeldia in his M.Sc. thesis, in an attempt - which proved to be successful - to improve it and expand its possibilities. Propeller's performance is computed applying the Blade Element Momentum Theory (BEMT), which requires the knowledge of the two-dimensional airfoil data in function of the angle of attack and Reynolds and Mach numbers. This has been carried out through the software XFOIL and experimental data.

Compared to the old methodology, the aerodynamic database, which initially included only NACA0012 airfoil data, has been also extended to airfoil NACA16-012, RAF-6 and Clark-Y. Additionally, thanks to a reliability study of XFOIL performed through a comparison with results coming from CFD simulations within OpenFOAM libraries, it was possible to adopt M. Drela's software up to a Mach number of 0.6, whereas De Zubeldia's limit was 0.5.

In addition, the formulation has been further modified to take into account rotational effects in aerodynamic coefficients, which, particularly for high angles of attack, are greatly affected by those effects and can vary considerably. In order to achieve this, after a scrupulous literature survey, correction proposed by P. Chaviaropoulos and M. Hansen has been applied.

Thereafter, this new formulation has been implemented in MATLAB® environment presenting two principal work modalities: (i) performance computation for a given geometry and (ii) geometry optimization. In this way, several numerical studies have been performed. Firstly, a parametric analysis of the aforementioned corrective model constants has been carried out. This correction turned to be very effective in rectifying the non-physical behavior of performance curves found in De Zubeldia's analysis, particularly at low advance ratio (J) regions to which correspond, indeed, high angles of attack, separated flow and hence considerable rotational effects. Subsequently, various optimizations loops have been carried out and it has been shown how, starting with the same initial guesses and cost functions, the software derived from the new formulation gives as a result generally smaller chord values along the span compared to the previous one. Furthermore, variable pitch mechanism presence has been simulated and it has proven that efficiency is over a given threshold for a much higher J range with a Constant Speed Unit (CSU). Finally, a preliminary attempt to introduce in the program the possibility of having different airfoil sections along the span has been performed.

All in all, this work was a success. All its objectives were met, constituting a consistent improvement: the formulation is now more complete and the resulting code is more versatile, precise and leads to more physically convincing results. Potential future enhancements of this work might include, among others, the expansion of the aerodynamic database to other airfoils families or the development of a structural and acoustic sub-routines.

Contents

Ringraziamenti **v**

Sommario **vii**

Abstract **ix**

Contents **xii**

List of Figures **xiv**

List of Tables **xv**

List of Symbols **xxi**

List of Acronyms **xxiv**

1 Introduction **1**
- 1.1 Objectives 1
- 1.2 Literature survey 3
- 1.3 Structure of the thesis 5

2 Formulation **7**
- 2.1 Baseline formulation 7
  - 2.1.1 Background theory 7
  - 2.1.2 SinglePerformance 9
  - 2.1.3 Optimizer 10
  - 2.1.4 Database 10
- 2.2 Extension to multiple airfoils 11
  - 2.2.1 Implications 11
  - 2.2.2 XFOIL vs CFD 12
- 2.3 Rotational effects 25
  - 2.3.1 Du and Selig method 25
  - 2.3.2 Chaviaropoulos and Hansen method 27
  - 2.3.3 Adopted method 28

3 Numerical studies **31**
- 3.1 Parametric study 31
- 3.2 Design optimizations 38
  - 3.2.1 Linear chord - linear twist initial guesses 40

3.2.2 Constant chord - linear twist initial guesses 44
3.2.3 Optimized performance for the different airfoils 48
3.3 Variable pitch case 51
3.4 Variable airfoil propeller 55

4 Conclusions 57
4.1 Further studies 59

A OpenFOAM detailed meshing procedure 61

B Stall-delay principle theory 65
B.1 Du and Selig model 65
B.2 Chiaviaropoulos and Hansen model 68

C Comprehensive graphical results 73

Bibliography 83

List of Figures

2.1 Cruise-optimized propeller chord distribution for different initial guesses. . 12
2.2 Fractional contributions to the total stress in the near wall region. 14
2.3 aw of the wall representation. 14
2.4 Example of a single `blockMesh` block. 17
2.5 `snappyHexMesh` vs `blockMesh` utility meshing. 18
2.6 Airfoil dimensions. 18
2.7 Final meshed domain. 19
2.8 Mesh magnification. 20
2.9 C_l vs α and C_d vs α at $Ma = 0.15$. 20
2.10 C_l vs α at $Ma = 0.25$ and $Ma = 0.35$. 21
2.11 C_l vs α at $Ma = 0.45$ and $Ma = 0.55$. 21
2.12 C_l vs α at $Ma = 0.65$ and $Ma = 0.75$. 22
2.13 Airflow Mach number around the airfoil for different angles and Ma_∞. . . . 23
2.14 Pressure field around the airfoil for different angles and Ma_∞. 23
2.15 Database airfoil contours. 25
2.16 Selig-Du method outputs compared to old and new code - Clark-Y airfoil. . 29

3.1 RAF-6 airfoil: η variation for different corrective coefficients values. 32
3.2 RAF-6 airfoil: T variation for different corrective coefficients values. 33
3.3 RAF-6 airfoil: Q variation for different corrective coefficients values. 34
3.4 NACA16-012 airfoil: η variation for different corrective coefficients values. . 35
3.5 NACA16-012 airfoil: T variation for different corrective coefficients values. 36
3.6 NACA16-012 airfoil: Q variation for different corrective coefficients values. 37
3.7 NACA16012 geom. optimization results-linear chord and twist initial guess 41
3.8 NACA16012 perf. optimization results-linear chord and twist initial guess . 42
3.9 RAF-6 geom. optimization results-linear chord and twist initial guess . . . 42
3.10 RAF-6 perf. optimization results-linear chord and twist initial guess 43
3.11 NACA16012 geom. optimization results-constant chord and linear twist initial guess 45
3.12 NACA16012 perf. optimization results-constant chord and linear twist initial guess 46
3.13 RAF-6 geo. optimization results-constant chord and linear twist initial guess 46
3.14 RAF-6 perf. optimization results-constant chord and linear twist initial guess 47
3.15 Comparative optimization results-linear chord and linear twist initial guess 49
3.16 Comparative optimization results-constant chord and linear twist initial guess 50
3.17 Constant speed propeller hub. 52
3.18 Optimized propeller (RAF6) - const chord linear twist - variable pitch performance 53

3.19 Optimized propeller (Clark-Y) - const chord linear twist - variable pitch performance 54
3.20 Performance output obtained with a mixed blade - half Clark-Y and half RAF-6 airfoil. 56

A.1 Case organization. 61
A.2 Aerofoil dimensions. 62
A.3 Domain vertices. 62
A.4 Curved edges. 63
A.5 Inlet and outlet definition. 63

B.1 Coordinate system attached to rotating blade. 65

C.1 Clark-Y airfoil: η variation for different corrective coefficients values. . . . 74
C.2 Clark-Y airfoil: T variation for different corrective coefficients values. . . . 75
C.3 Clark-Y airfoil: Q variation for different corrective coefficients values. . . . 76
C.4 Clark-Y optimization results-linear chord and twist initial guess 77
C.5 Clark-Y optimization results-constant chord and linear twist initial guess . 78
C.6 Optimized propeller (RAF6) - linear chord and twist - variable pitch performance 79
C.7 Optimized propeller (Clark-Y) - linear chord and twist - variable pitch performance 80
C.8 Optimized propeller (NACA16012) - const. chord and linear twist - variable pitch performance 81
C.9 Optimized propeller (NACA16012) - linear chord and twist - variable pitch performance 82

List of Tables

1.1 Typical values of Ncrit for various situations. 3

2.1 Air properties during CFD simulations. 18
2.2 Domain dimensions. 19
2.3 Database airfoil list. 24
2.4 Fitted coefficients of Chaviaropoulos and Hansen model. 28

3.1 First optimization: initial guess. 40
3.2 First optimization-NACA16-012: D and Ω results. 41
3.3 First optimization-RAF-6: D and Ω results. 44
3.4 Second optimization: initial guess. 44
3.5 Optimized propellers performance. 51
3.6 Advance ratio interval where $\eta > 0.6$. 55

List of Symbols

a	First constant in Chaviaropoulos and Hansen correction law
a'	First Selig-Du method empirical correction factor
a_x	Axial induction factor
a_y	Circumferential induction factor
B	Number of propeller blades
B_{x0c}	First Bezier coefficient for chord distribution
B_{x0t}	First Bezier coefficient for twist distribution
B_{x3c}	Third Bezier coefficient for chord distribution
B_{x3t}	Third Bezier coefficient for twist distribution
b	Second Selig-Du method empirical correction factor
C	Generic constant in the logarithmic velocity law
c	Local section chord
C_d	Drag coefficient
$C_{d,0}$	2D drag coefficient when $\alpha = 0$
$C_{d,2D}$	2D drag coefficient
$C_{d,2D_{min}}$	Minor 2D drag coefficient
$C_{d,3D}$	3D drag coefficient
C_f	Friction coefficient

C_l	Lift coefficient
$C_{l,2D}$	2D lift coefficient
$C_{l,3D}$	3D lift coefficient
$C_{l,p}$	Lift coefficient according to potential theory
$C_{l,inv}$	Lift coefficient according to potential theory
C_P	Pressure coefficient
C_p	Power coefficient
$C_{P_{inc}}$	Incompressible pressure coefficient
C_Q	Torque coefficient
C_T	Thrust coefficient
C_x	Axial-directed force coefficient
$C_{x,2D}$	Generic 2D coefficient
$C_{x,3D}$	Generic 3D coefficient
C_y	Circumferential-directed force coefficient
C_L	Aerodynamic lift coefficient
D	Propeller diameter
d	Third Selig-Du model empirical correction factor
F	Loss factor
F_1	Blending function number 1
F_2	Blending function number 2
f_d	Drag coefficient correction function
f_l	Lift coefficient correction function
G_κ	Represents the generation of turbulence kinetic energy due to the mean velocity gradients

H	Boundary layer shape factor
h	Second constant in Chaviaropoulos and Hansen correction law
h_{wall}	Height of the first cell
J	Propeller advance ratio
k	Velocity gradient
Ma	Mach number
Ma_{∞}	Asymptotic Mach number
Ma_{lim}	Maximum database Mach number
Ma_{tip}	Mach number at propeller tip
N_{crit}	Log of the amplification factor of the most amplified frequency which triggers transition
N_{2D}	Two-dimensional Navier-Stokes scalar operator
n	Third constant in Chaviaropoulos and Hansen correction law
P_{in}	Input power
P_{iout}	Output power
Q	Aerodynamic torque
q	Work distribution parameter
R	Propeller radius
Re	Reynolds number
Re_i	Reynolds number at the i-th station
Re_{min}	Minimum database Reynolds number
r	Local section radial position
S	Strain rate magnitude

s	Separation point on airfoil surface
T	Aerodynamic thrust
T_{TO}	Take-off thrust
T_c	Cruise thrust
u	Flow velocity
u^+	Dimensionless flow velocity
u_τ	Friction flow velocity
V_∞	Asymptotic velocity
V_x, V_y	Normal and circumferential velocity component seen by the propeller
W_r, W_z, W_Θ	Velocity vector radial, axial and circumferential components
y	Absolute wall distance
y^+	Dimensionless wall distance
z_g	Z-direction grading
α	Angle of attack
α_0	Zero-lift angle of attack
β_w	Limiting stream angle
Γ	Pohlhausen shape parameter
Γ_κ, Γ_ω	Effective diffusivity for κ and ω
γ	Air heat capacity ratio
Δ	Generic quantity increment
δ	Boundary layer thickness
δ_1, δ_2	Boundary-layer displacement thickness
ϵ	Dissipation rate

η	Propeller efficiency
η'	Non-dimensional boundary layer height
η_c	Propeller efficiency in cruise condition
θ	Twist angle
θ_1, θ_2	Boundary-layer momentum thickness
K	Von Karman constant
κ	Turbulent energy
Λ	Modified tip speed ratio
λ	Tip speed ratio
μ	Dynamic viscosity
μ_τ	Turbulent viscosity
ν	Kinematic viscosity
ξ_i	Non-dimensional radial coordinate corresponding to i-th station
ξ_{hub}	Non-dimensional radial coordinate at the hub
ρ	Air density
σ	Blade solidity
σ_κ, σ_ω	Prandtl number for κ and ω
τ_w	Wall shear stress
Φ	Pressure-like term including centrifugal effect
ϕ	Inflow angle
Ω	Propeller angular velocity
ω	Specific dissipation rate

List of Acronyms

A/C	Aircraft
AoA	Angle of Attack
BEMT	Blade Element Momentum Theory
BL	Boundary Layer
CF	Cost Function
CFD	Computational Fluid Dynamics
CSU	Constant Speed Unit
NACA	National Advisory Committee for Aeronautics
NASA	National Aeronautics and Space Admnistration
OBJ	Wavefront Object format
PISO	Pressure Implicit with Splitting of Operators
RHS	Right Hand Side
SIMPLE	Semi-Implicit Method for Pressure-Linked Equations
SPL	Sound Pressure Level
SQP	Sequential Quadratic Programming
SST	Shear Stress Transport
STL	Stereolithography format
TC	Transonic Correction
UAV	Unmanned Air Vehicle

VPP	Variable Pitch Propeller

Chapter 1

Introduction

Nowadays, renewed attention is being focused on the first aeronautical propulsion device: the propeller. This is due to the increased use of unmanned air vehicles (UAVs), the growing market of general aviation, the increasing interest in ultralight categories or light sport air vehicles, and the growing importance of environmental issues that have led to the development of all-electric emission-less aircraft which make use of propeller engines [1].

In this sense, the objective of the present analysis is to the develop a formulation that would help in the early design phase of a propeller, starting from a previous research performed by De Zubeldia in his M.Sc. thesis [2], in an attempt - which proved to be successful - to improve it and expand its possibilities. Propeller's aerodynamic performance is computed applying the Blade Element Momentum Theory (BEMT), a very efficient method that requires very small computer resources as compared to detailed vortex models or computational-fluid-dynamic models. The efficiency comes at the price of having to know the two-dimensional airfoil data, namely, $C_l(\alpha,\ Ma,\ Re)$ and $C_d(\alpha,\ Ma,\ Re)$, over a wide range of angles of attack, Mach and Reynolds numbers. This has been performed, as previously pointed out, through XFOIL and experimental data for high subsonic Mach numbers.

The new formulation has been finally implemented in a software, presenting two principal working modalities: (i) performance computation for a given geometry and (ii) geometry optimizations. In this fashion, several numerical studies have been performed, showing that the new methodology leads to more precise and physically convincing results.

1.1 Objectives

As already stated, the present work intends to be the natural continuation of the aforementioned thesis research which, at the present stage, presents some limitations.

Firstly, in the previous formulation the aerodynamic database included data relative to the sole NACA0012 section, which is intrinsically not suitable for a propeller because of its relatively advanced position (with respect to the leading edge) of the maximum thickness. As a matter of fact, one of the most used geometry for propeller airfoils is that of NACA16 family, which, thanks to a minimum-pressure point unusually far aft on both surfaces and - consequently - a small continuously favorable pressure gradient from the leading edge to the position of minimum pressure, are low-drag high-critical-speed wing sections [3] (Chapter 6, page 118).

This choice of working only with NACA0012, despite of not being optimal for the purpose, was pursed by De Zubeldia due to the large amount of experimental data obtained using the latter airfoil. In particular, XFOIL was used to retrieve aerodynamic coefficients only up to 0.5 Mach while for high subsonic regimes, often encountered in the outer blade sections for high advance ratios (J), his methodology relied on the previous mentioned experimental data. The extension to other profiles was hence limited by this lack of reliable aerodynamic data in that part of the flight envelope. In the present study attention has been posed to this aspect in order to expand the possibilities of the formulation and to obtain more reasonable results in terms of geometry of the blade. In particular, the goal was to understand exactly up to which Mach number was possible to rely on XFOIL for the creation of the aerodynamic database, avoiding the need of relying on experimental data for high subsonic Ma, often non-existent for most airfoils.

The second enhancement applied to the original formulation consists in a refinement of aerodynamic coefficients computation at low advance ratios. At the present stage, the code derived from De Zubeldia's methodology presented, for particular geometries, some irregularities in the calculations at low J. An attempt to use classical methods - namely the BEMT - to calculate the performance of propellers in this part of the flight envelope exhibits large differences between the calculated and measured thrust, while the same methods exhibit excellent agreement at higher advance ratios. The differences between the calculations and measurements increase as larger portions of the blades experience high cross-sectional angles of attack, beyond the stall limit. Thus, it becomes clear that there are problems in modeling the stall characteristics of cross sections of propellers' blades. It turns out that the measured propeller's thrust is higher (sometimes much higher) than the thrust predicted by calculations where the two-dimensional stall characteristics of a non-rotating airfoil are used [4]. The effect was attributed in various studies ([5],[6],[7],[8],[9],[10]) to the presence of a Coriolis force, having the same effect as a favorable pressure gradient. In addition, the centrifugal force causes an outward displacement of fluid particles, through which the boundary layer becomes thinner compared to a non-rotating boundary layer. This phenomenon is often known as *stall-delay*.

For the sake of truth, 3D post stall effects are beneficial to propellers and helicopter rotors (more thrust and less power) and consequently have not been studied in any great detail, whereas most of the recent analyses have been focused on wind turbines. In fact, when contrasting propellers and helicopter rotors with wind turbine rotors, it is important to keep in mind that for wind turbines the lift is directed to produce torque or power; whereas, for propellers it is directed to produce thrust. Thus, a 3D lift enhancement produces more power for a wind turbine and more thrust for a propeller. This means that not considering 3D effects on a propeller would cause in the worst case an underestimation of the thrust, which is not, safety speaking, negative. On the other hand, in the case of stall regulated wind turbines, still widely used for small and medium wind turbines because of its ease in manufacturing and low operational costs, stall condition is encountered on a regular basis. Therefore, the accuracy in load prediction under the aforementioned condition is very important. Despite this aspect, the goal of the present analysis was then to overcome this limitation, through a correct description of the stall phenomenon expressing appropriate correction laws for airfoil data that would improve the reliability of the simulations as will be shown later in detail. In this fashion, the formulation results more complete and the resulting code more versatile and precise, in particular when dealing with those kinds of propeller-driven aircraft which can not rely on variable pitch (e.g. ultralight, low price general aviation aircraft or UAVs) and can experience high AoA in large portions of the blade in certain flight conditions.

1.2 Literature survey

In light of what has been exposed in the previous section, the analysis started with a scrupulous inspection of De Zubeldia's formulation [2] and derived software.

In the presentation of his software XFOIL [11], Mark Drela shows its main features and limitations. The code combines a potential flow panel method and an integral boundary layer formulation for the analysis of the flow around airfoils. It was developed to rapidly predict the airfoil performance and its convergence is achieved through the iteration between the outer and inner flow solutions on the boundary layer displacement thickness. Thus, the XFOIL code calculates the viscous pressure distribution and captures the influence of limited trailing edge separation and laminar separation bubbles.

An approximate e^N envelope method is used to calculate transition. With this method the code tracks only the most amplified frequency at a given point on the airfoil downstream from the point of instability to obtain the amplitude of that disturbance. Transition is assumed when this integrated amplitude reaches an empirically determined value. The procedure requires the user-specified parameter N_{crit}, which is the log of the amplification factor of the most-amplified frequency which triggers transition. A suitable value of this parameter depends on the ambient disturbance level in which the airfoil operates, and mimics the effect of such disturbances on transition. In Table 1.1 are reported typical values of Ncrit for various situations.

Table 1.1: Typical values of Ncrit for various situations [12].

Situation	N_{crit}
Sailplane	12-14
Motorglider	11-13
Clean wind tunnel	10-12
Average wind tunnel	9†
Dirty wind tunnel	4-8

†*Standard "e^9 method".*

A Karman-Tsien compressibility correction is incorporated, allowing good compressible predictions all the way to sonic conditions. In particular, compressible speed q and pressure coefficient C_P can be approximately determined from their incompressible flow values q_{inc} and $C_{P_{inc}}$ by

$$C_P = \frac{C_{P_{inc}}}{\beta + \lambda(1+\beta)C_{P_{inc}}/2} \tag{1.1a}$$

$$q = \frac{q_{inc}(1-\lambda)}{1-\lambda(q/q_\infty)^2_{inc}} \tag{1.1b}$$

where $\beta = \sqrt{1 - Ma^2_\infty}$ and $\lambda = Ma^2_\infty/(1+\beta^2)$.

In the same article [11], the author reports a visual comparison between a viscous calculation of a RAE 2822 airfoil with experimental data at Mach 0.676. Reasonable agreement is obtained, despite the fact that, as noticeable from equations 1.1, the Karman-Tsien correction formally breaks down in supersonic flow.

These evidences legitimately questioned the constraint used in De Zubeldia's methodology [2] when creating the aerodynamic database. In particular, the limit of $0.5Ma$ seemed too stringent.

J. Morgado et al. [13] completed a performance prediction comparison between XFOIL and CFD for high lift, low Reynolds number airfoils. A similar study has been conducted by O Günel et al. [14]. Both analyses gave compatible results until stall angle, utilizing OpenFOAM with κ-ω SST (Shear Stress Transport) turbulence model.

Hence the decision of performing an analogous investigation in order to assess up to which Mach number is XFOIL reliable in terms of aerodynamic coefficient computation. This procedure passed from an initial validation of the two-dimensional steady CFD case, which will be discussed thoroughly in the next chapters.

As already pointed out, despite being quite simple, the BEMT method yields surprisingly accurate predictions of the aerodynamic loads provided that "proper" lift and drag-incidence curves are used for the airfoils mounted on the rotor blade. The evident question is how these proper data are obtained, especially for stall conditions (namely low J).

Himmelskamp [5], who was the first to address the three-dimensional effects of rotation, found lift coefficients as high as 3 near the hub of a fan blade.

Introducing for the first time a quasi-3D approach, based on the viscous-inviscid interaction method, Snel et al. [6] proposed a semi-empirical law for the correction of the 2D lift curve, identifying the local chord to radii (c/r) ratio of the blade section as the main parameter of influence. Chaviaropoulos and Hansen [7] followed the same path and once again three-dimensional and rotational viscous effects are investigated by means of a quasi-3D Navier-Stokes model for stall controlled wind turbines, whose physics are comparable to those of propellers. The governing equations of the model are derived from the 3D primitive variable incompressible Navier-Stokes equations written in cylindrical coordinates in the rotating frame of reference. The resulting quasi-3D model suggests that three-dimensional and rotational effects are strongly related to the local chord by radii ratio and the twist angle, which was not taken into consideration by Snel et al. [6]. Hence, the 3D stall delay model consisted of two key parameters (the ratio of local chord to local radius c/r and the twist angle) and three empirical correction factors. Also, both laminar and turbulent flow simulations are performed using the κ-ω model with wall functions, assuming fully turbulent flow conditions. The former is used for identifying the physical mechanism associated with the 3D and rotational effects, while the latter for establishing semi-empirical correction laws for the load coefficients, based on 2D airfoil data. Comparing calculated and measured power curves of a stall controlled wind turbine, it is shown that the suggested correction laws may improve significantly the accuracy of the predictions.

Similar observations have been made by Gur and Rosen [4], who proposed a simple correction model to better depict the propeller's aerodynamic performance at low advance ratios. The model includes unknown constants that are determined after applying a least-squares procedure of matching between calculated results and test data. According to the authors, this correction model is probably a function of the specific airfoil, Mach number of the cross-sectional flow, and speed of rotation. Yet it was shown that a correction model that was obtained for a certain propeller also gave good results for another propeller that uses the same airfoils.

An analogous path was followed by Du Z. and Selig M. S. [10], who started with a description and analysis of the fundamental flow phenomena that characterize the boundary layer on rotating blades, investigating the 3D integral boundary-layer equations for a reference system rotating with the blade and describing the effects of rotor rotation on the separation point and its causes. Consequently, basing on the developed theory, they continued determining a simple correction formula to obtain rotating rotor lift coefficient $C_l(\alpha)$ and drag coefficient $C_d(\alpha)$ data from 2D airfoil data. As for [7], the preliminary 3D stall delay model consisted of two key parameters and three empirical correction factors. This time twist was not taken into account and, apart from the local chord to local radius ratio (c/r) which is in common between the two theories, the ratio of rotation speed to free stream velocity was considered.

In this study, a procedure inspired first to Du and Selig [10] and second to Chaviaropoulos and Hansen [7] - whose works showed the most promising results - has been performed as will be later shown in depth.

1.3 Structure of the thesis

This thesis consists in four chapters, considering the introductory present one. In Chapter 2, after a brief recapitulation of De Zubeldia's formulation and derived software working principles (for a more detailed explanation refer to his work [2]), the improvements presented in 1.1 are analyzed in depth. Firstly, the process of aerodynamic database extension is shown, along with the CFD simulations needed to obtain it and its implications. Particular attention is paid to the simulation part, where XFOIL predictions are compared to those of OpenFoam. Thereafter, how rotational effects are taken into account in the present formulation when computing aerodynamic coefficients is exposed. In particular, two different models are considered, that of Chaviaropoulos and Hansen and of Selig and Du. Both of them are analyzed and the one that worked better for the present analysis is chosen. Thereafter, the new formulation has been implemented in MATLAB® environment and several numerical studies have been performed. Corresponding results are shown in Chapter 3. Initially, a parametric study of the a, h and n coefficients effect on the correction law (eq. 2.31) have been carried out for all the database airfoils. In particular, given a certain twist and chord distribution, performance outputs were computed varying the aforementioned coefficients and reported in the same plot, where is also shown the curve relative to the old methodology. Thereafter, optimization loops with the software derived from the enhanced formulation (both in terms of database and rotational effects) have been performed, focusing on the differences with respect to the old one starting with the same initial guesses. In addition to this, the presence of a variable pitch mechanism has been simulated changing by a (positive or negative) constant value the optimized twist distribution and computing the performance output for each resulting case. Furthermore, in Section 3.4 a preliminary attempt of introducing in the software the possibility of having different airfoil sections along the span has been performed. This constituted only an initial study and had more a demonstrative purpose but gave an idea of what would actually be to use more than one airfoil for a single blade. Finally, Chapter 4 contains conclusions suggested by the present work and its possible future developments. This thesis includes also three Appendices. In Appendix A details about meshing procedure for the 2D case are given. In Appendix B theoretical aspects behind the corrective models that take into account three-dimensional effects on a rotating propeller are extensively reported. Lastly, in Appendix C a comprehensive list of all non-reported plots (for brevity reasons) in the main corpus is given.

Chapter 2

Formulation

The present formulation is an extension of the already existent one presented by De Zubeldia in [2], which, as already said, suffers from some limitations. After a brief summary of the aforementioned thesis work - which represents the starting point of this analysis - given in section 2.1, nature of the improvements is given in sections 2.2 and 2.3.

2.1 Baseline formulation

2.1.1 Background theory

The objective of De Zubeldia's project [2] was to develop a methodology which allows the user to do the calculations necessary in the preliminary design of aircraft propellers. Afterwards, the author implemented it in a software. The chosen environment was MATLAB®, due to its ease of use. BEMT has been used to obtain propeller characteristics in the software. The aforementioned model is the union of two simpler models: momentum theory, which uses a control-volume analysis, and blade element theory, which is based on known airfoil data at each section. In particular, momentum theory applies conservation laws on a control volume extending far upstream and downstream of the propeller, modeled as a disc. Assuming known induced velocities, momentum theory predicts [15]

$$dT = C_T \frac{1}{2} \rho V_x^2 dA \tag{2.1a}$$

$$dQ = C_Q \frac{1}{2} \rho V_x^2 dA r \tag{2.1b}$$

With

$$C_T = 4F a_x (1 + a_x) \tag{2.2a}$$

$$C_Q = 4F a_y (1 + a_x) \frac{V_y}{V_x} \tag{2.2b}$$

Where dT and dQ are the differential thrust and torque, C_T and C_Q are the thrust and torque coefficients, V_x and V_y are the normal and circumferential velocity components seen by the propeller, dA is the area for a radial slice of the propeller, ρ is air density, F is the loss factor, and r is the radial position.

Blade element theory uses specified airfoil polars to compute the forces on a blade-by-blade basis given their local angles of attack at radial sections. The differential thrust and torque predicted at a radial section are [15]

$$dT = BC_x \frac{1}{2} \rho V^2 c dr \tag{2.3a}$$

$$dQ = BC_y \frac{1}{2} \rho V^2 c r dr \tag{2.3b}$$

Where B is the number of blades, c is the chord, dr is the differential radial width, and V is the magnitude of the velocity seen by the section, which depends on the axial and circumferential induction factors (a_x and a_y, respectively). C_x and C_y are found projecting aerodynamic coefficients C_l and C_d in the axial and circumferential directions

$$\begin{bmatrix} C_x \\ C_y \end{bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} C_l \\ C_d \end{bmatrix} \tag{2.4}$$

With the local inflow angle ϕ computed from $\phi = \theta$ - α, where θ is the local twist angle.

Now, equating torque and thrust predicted by momentum theory and blade element theory at each radial section it is possible to compute the axial and tangential induced velocities, a_x and a_y

$$a_x = \frac{C_x \sigma}{4F \sin^2\phi - C_x \sigma} \tag{2.5a}$$

$$a_y = \frac{C_y \sigma}{2F \sin 2\phi + C_y \sigma} \tag{2.5b}$$

Where σ is the blade solidity, given by $\sigma = \frac{Bc}{2\pi r}$. Since the right-hand sides of the above equations depend on ϕ and V, which in turn depend on a_x and a_y, this is a coupled system of equations. However, V appears only in the Reynolds number, and is only weakly dependent on the induction factors, so this system can be approximated as a single equation in ϕ only, which allows to use a bracketed search method that has been proven to be convergent by A. Ning [16]. The residual function, rearranged to avoid singularities, is given by

$$V_y \left(4F \sin^2(\phi) - C_x \sigma\right) - V_x \left(2F \sin 2\phi + C_y \sigma\right) = 0 \tag{2.6}$$

Valid for $V_x > 0$ and $V_y > 0$.

Other quantities useful to assess performance or make a comparison of propellers of different size are of advance ratio (J), thrust coefficient (C_T), torque coefficient (C_Q)[1] and power coefficient (C_p). The definition of the aforementioned coefficients is given in equations 2.7.

---

[1] Differently from the case of momentum theory shown above, C_T and C_Q expressions given hereby come from a dimensional analysis [17] and are more useful to assess a generic propeller performance.

$$J = \frac{V}{n_s D} \tag{2.7a}$$

$$C_T = \frac{T}{\rho n_s^2 D^4} \tag{2.7b}$$

$$C_Q = \frac{Q}{\rho n_s^2 D^5} \tag{2.7c}$$

$$C_p = \frac{P_{in}}{\rho n_s^3 D^5} \tag{2.7d}$$

For what concerns the practical part, De Zubeldia's software works in the following manner. As already mentioned in the initial part of Chapter 1, the program is divided into two main "routines": the single computation one (function `SinglePerformance`) and the optimization one (function `Optimizer`).

2.1.2 SinglePerformance

For this function, required input arguments are:

- number of blades;
- diameter of the propeller;
- chord and twist distribution along the span;
- airfoil of the section;
- rotational speed;
- free stream speed;

Once all this data is known, the blade is divided into several stations following a cosine law:

$$\xi_i = \cos\left((1 - \frac{i-1}{n-1}) \sin^{-1} \xi_{hub}\right) \tag{2.8}$$

Where ξ_i represents the non-dimensional ratio r/R corresponding to the i-th station, equal to ξ_{hub} when computed at $r = r_{hub}$.

This distribution has been proven to be remarkably more effective than a uniform distribution. In particular, De Zubeldia's [2] showed how the same percentage error[2] was achieved with less than one third of stations if they were spaced according to cosine law distribution instead of the uniform one.

For each station, the residual function (eq. 2.6) is solved, giving ϕ, which, substituted in eq. 2.4, gives C_x and C_y needed to obtain the induction factors through eqs. 2.5. Having done that, the program proceeds to compute forces per unit span through eqs. 2.3. Afterwards, a spatial integration is performed, resulting in the total forces (thrust and torque) and corresponding coefficients, through which input power P_{in} and output power

---

[2] The error was computed as the difference between a certain parameter obtained with a number n of stations and the same parameter obtained with 150 stations, in percentage.

P_{out} and, consequently, propeller efficiency η are computed. Repeating this procedure for different free stream velocities (which correspond, for fixed Ω, to different J) as input, one can obtain the aforementioned quantities for all the envelope.

2.1.3 Optimizer

To this function the user is required to give as input an initial guess of the blade geometry, that is chord and twist distribution. In addition to this, also initial indication of the propeller diameter and rotational speed must be provided. Thereafter, one can choose the objective function and the constraints. In [2] the author defines the objective function as $-\eta$ in cruise condition and nine constraints involving thrust at take-off and cruise conditions, Mach and Reynolds numbers along the span and Bezier coefficients[3]. With all these parameters defined, `Optimizer` calls the built-in MATLAB function `fmincon`, which evaluates the objective function varying the input parameters at each iteration until the best solution, satisfying all the constraints, is found. It is obvious that, at every loop, the objective function evaluation involves a call to `SinglePerformance`, which, as shown in section 2.1.2, can return the efficiency value for a given propeller geometry. For this reason `SinglePerformance` can be considered a sub-routine of `Optimizer`.

2.1.4 Database

As deducible from the previous subsections, both functions have to go through the computation of C_x and C_y, needed to obtain the induction factors. As noticeable from eq. 2.4, knowledge of C_d and C_l of each section's airfoil is required. This represented one of the major challenges in De Zubeldia's work and the main reason why airfoil geometry parametrization in optimization routine was not included. In fact, as reported by both Tarraran [18] and De Zubeldia [2], this would imply a call to an external aerodynamic software to compute C_d and C_l basing on AoA, Mach and Reynolds numbers at every BEMT iteration. Even utilizing a light panel method software, such as XFOIL, would cause an exponential increase in CPU time, since the call to an external program forces MATLAB to wait for it to be launched, run and terminated, before going on with the optimization. Hence, even if the program itself is very fast, these dead times make airfoil geometry parametrization impractical (if not unfeasible). Therefore, the author followed another path, consisting in a pre-computed aerodynamic database. This database comprises a series of text files in which the aerodynamic coefficients of a fixed airfoil are stored. These text files were then transformed into `.mat` files, that, once loaded, are directly accessible from the current folder.

As already introduced, the database was created for the sole NACA 0012 airfoil exploiting XFOIL [11]. Angles of attack ranged from $-7°$ to around $17°$. Outside of this range of AoA, in order to obtain full-range polars, the data is extended post-stall using the Viterna method, which uses an inviscid flat plate assumption. Details about this procedure can be found in [2], Chapter 2, Section 1. For negative angles (less than $-10°$) the following approximation was applied:

---

[3] Bezier curves were used for the definition of chord and twist distributions.

$$C_l = 0 \tag{2.9a}$$
$$C_d = 0.13 \tag{2.9b}$$

Text files that form the polar database consist in columns of data containing lift and drag coefficients in function of the angle of attack. Such text files are obtained for different couples of Reynolds and Mach numbers, which take discrete values from $5 \cdot 10^4$ to $1 \cdot 10^6$ in steps of $5 \cdot 10^4$ and from 0 to 0.5 in steps of 0.05, respectively. For higher Mach numbers, since it was not known exactly when XFOIL became inaccurate, De Zubeldia relied on other sources. Data coming from “NASA memorandum 81927” [19], where NACA 0012 was tested in wind tunnels up to high subsonic Mach numbers, was added to the database to cover also that part of the envelope. During each BEMT iteration, the program computes Reynolds, Mach number and α at every station. It then proceeds to scroll the aerodynamic database looking for the two closest Re, Ma and α available. At this point, an interpolation is performed and the corresponding aerodynamic coefficients obtained.

2.2 Extension to multiple airfoils

The structure of the software as designed by De Zubeldia is pretty robust and the rate of failure is very low[4]. For this reason, the decision in the present study was to maintain its main framework unaltered, when implementing the new formulation. Nonetheless, some limitations were present, which have been already introduced in section 1.1.

In this section, it will be shown how the first modification was carried out, that is the aerodynamic database extension to other airfoils.

2.2.1 Implications

Having at disposal only one airfoil in the aerodynamic database, limits intrinsically the methodology and, consequently, the derived software in its calculations. In particular, during the optimization routine it can only operate with that particular airfoil which, has already mentioned, is far from being the optimal choice for an aeronautical propeller. As a matter of fact, looking at the optimizations results in [2], Chapter 4, section 4, to obtain feasible chord values while satisfying thrust constraints during take-off and cruise, an extra blade had to be added compared to similar aircraft’s engines, such as that of Beech 1900, which was taken as reference by De Zubeldia in his study. Despite adopting this solution, chord dimensions reached values as high as 0.72 m after both optimizations were performed[5], as noticeable in Figure 2.1:

Taking into account the considerations on impracticality of including airfoil geometry as a variable parameter during propeller optimization made by both Tarraran [18] and De Zubeldia [2], the choice was to expand the database, pre-computing aerodynamic polars for different airfoils. Once created and loaded as a local variables in `.mat` format, the user has to choose between the airfoils present in the database, which will be then fed to the program and kept as a constant during both `Optimizer` and `SinglePermormance`

---

[4] If the database contains some `NaN` due to XFOIL crashing, resolution of eq. 2.6 is not possible through MATLAB built in function `fsolve`.

[5] Linear chord and twist distribution and constant chord and linear twist distribution as initial guess.

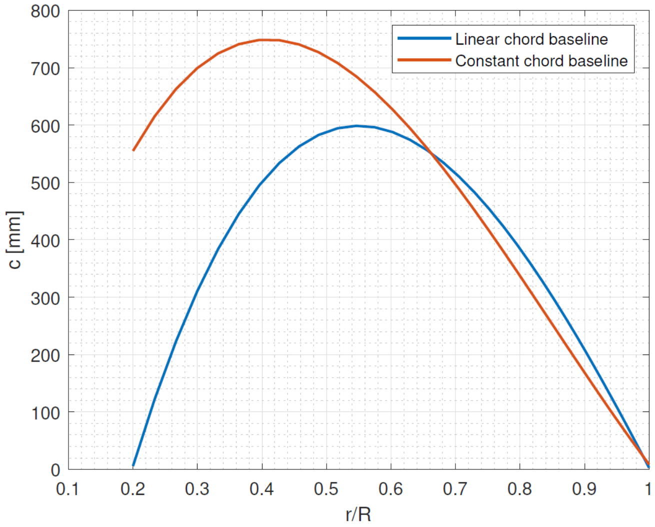


Figure 2.1: Cruise-optimized propeller chord distribution for different initial guesses [2].

routines. In this fashion, the advantages for the derived software are twofold. If the user is trying to compute the performance of an existent geometrically defined propeller, having more than one airfoil in the database increases the possibility that one of them matches the one used in the propeller under study. On the other hand, it is possible to perform the same optimization in terms of goal function and constraints but repeating it with different airfoils (among the available ones) and compare the different outcomes in order to choose the one that better suits the desired characteristics.

2.2.2 XFOIL vs CFD

The tool used to expand the aerodynamic database was once again XFOIL due to its capacity of dealing with different geometry airfoils, as long as their coordinates are correctly passed to the program through the `LOAD` command. `LOAD` recognizes four airfoil file formats: Plain, Labeled, ISES, MSES. In the present work the chosen one was the Labeled, since it was deemed the best format to use as suggested by M. Drela in [12]. Hence, each file contained only the X,Y coordinates, which run from the trailing edge, round the leading edge, back to the trailing edge in either direction plus the airfoil name string on the first line.

As already introduced in section 1.2, basing on existing evidences [11], the limit of $Ma = 0.5$ used by De Zubeldia seemed legitimately too stringent. An investigation, analogous to those of J. Morgado et al. [13] and O. Günel et al. [14], was conducted in order to assess up to which Mach number is XFOIL reliable in terms of aerodynamic coefficient computation. This procedure was performed comparing XFOIL outcomes with CFD results, obtained through OpenFOAM libraries.

OpenFOAM Pre-Processing

Before proceeding to the comparison, it is necessary to validate the CFD two-dimensional case in order to consider it as a reliable source at higher Mach numbers, where XFOIL has to be tested. In order to do this, consolidated data for low Mach numbers coming from XFOIL, I. H. Abbot book [3] and NASA technical memorandum 4073 [20] were taken as reference to perform a fine tune of the CFD case.

Turbulent flows are an omnipresent phenomenon in CFD and are significantly affected by the presence of walls, where the viscosity-affected regions have large gradients in the solution variables. An accurate representation of the near wall region determines a successful prediction of wall bounded turbulent flows. In order to deal with the near wall region, two ways are usually proposed [21].

1. One way is to integrate the turbulence to the wall. Turbulence models are modified to enable the viscosity-affected region to be resolved with all the mesh down to the wall, including the viscous sub-layer. When using a modified low Reynolds turbulence model to solve the near-wall region, the first cell center must be placed in the viscous sub-layer (preferably $y^+ = 1$) leading to the requirement of abundant mesh cells. Thus, substantial computational resources are required.

2. Another way is to use the so-called wall functions, which can model the near wall region. Wall functions are equations empirically derived and used to satisfy the physics in the near wall region. The first cell center needs to be placed in the log-law region ($30 < y^+ < 300$) to ensure the accuracy of the results. Wall functions are used to bridge the inner region between the wall and the turbulence fully developed region. When using the wall functions approach, there is no need to resolve the boundary layer causing a significant reduction of the mesh size and the computational domain.

The path followed in this study was that of point 2.

One of the most prominent parameters when judging the applicability of wall functions is the so-called dimensionless wall distance y^+ denoted by:

$$y^+ = \frac{y u_\tau}{\nu} \tag{2.10}$$

Where u_τ is the so called friction velocity, y is the absolute distance from the wall and ν is the kinematic viscosity.

It is possible interpret y^+ as a local Reynolds number, which means that its magnitude can be expected to determine the relative importance of viscous and turbulent processes. Figure 2.2 shows the fractional contributions to the total stress from the viscous and Reynolds stresses in the near wall region of channel flow.

One can easily see that if are in the viscous wall region with $y^+ < 50$, there is a direct effect of the viscosity on the shear stress. Conversely, in the outer layer with $y^+ > 50$, the effect of viscosity is negligible.

For the sake of completeness, friction velocity u_τ is briefly introduced. It should be evident that the viscosity ν and wall shear stress τ_w are important parameters. From these quantities and ρ it is possible to define it:

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \tag{2.11}$$

With:

$$\tau_w = \rho\nu \left(\frac{dU}{dy}\right)_{y=0} \tag{2.12}$$

At this point, the dimensionless velocity is given by:

$$u^+ = \frac{u}{u_\tau} \tag{2.13}$$

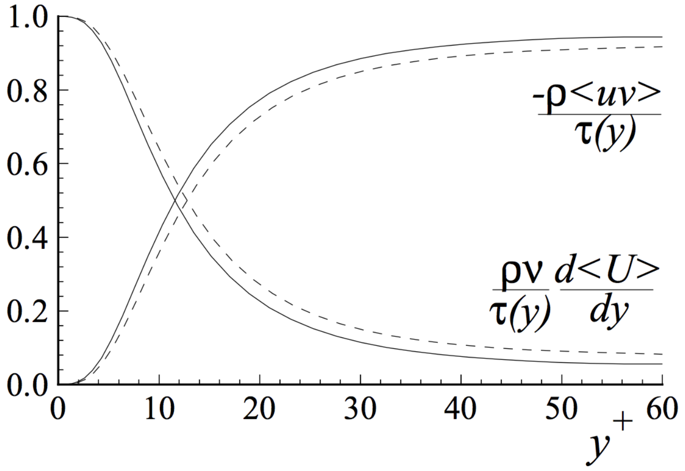


Figure 2.2: Fractional contributions to the total stress from the viscous and Reynolds stresses in the near wall region [22].

Wall functions rely on the universal law of the wall, which basically states that the velocity distribution very near to a wall is similar for almost all turbulent flows. Its graphical representation is given in Figure 2.3.

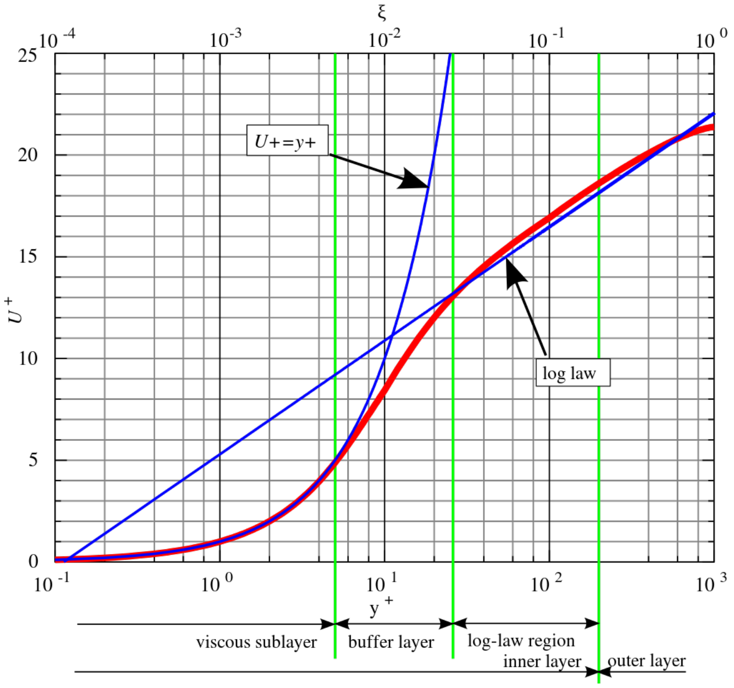


Figure 2.3: Law of the wall representation [22].

As noticeable from Figure 2.3, three zones are distinguishable:

- **The viscous sub-layer** ($y^+ < 5$): In the viscous layer, the fluid is dominated by the viscous effect, so it can be assumed that the Reynolds shear stress is negligible. The "linear velocity law" is given by:

$$u^+ = y^+ \tag{2.14}$$

- **The logarithmic area** ($y^+ > 30$): In the logarithmic layer, turbulence stress dominate the flow and velocity profile varies very slowly with a logarithmic function along the distance y. That is:

  $$u^+ = \frac{1}{K} \ln y^+ + C \tag{2.15}$$

  Where K represents the Von Karman constant.

- **The buffer layer** ($5 < y^+ < 30$): The buffer layer is the transition region between the viscosity-dominated region and turbulence-dominated part of the flow. Viscous and turbulent stresses are of similar magnitude and since it is complex, the velocity profile is not well defined and the original wall functions avoid the first cell center located in this region.

In the pre-processing stage of the simulation it is important to make sure that y^+ is in the desired range. Following path 2 requires, as already said, $30 < y^+ < 300$. This is a rather general guideline. In fact, for very high Re, y^+ can be higher if still in logarithmic layer and for very low (but still turbulent) Re, the logarithmic layer may not extend far enough away from the wall for the use of wall functions to be valid.

That being said, it necessary to know a suitable size for the first layer of grid cells (inflation layer) so that y^+ falls in that range. Unfortunately, the actual flow-field will not be known until the solution has been computed (and indeed it is sometimes unavoidable to have to go back and re-mesh the model on account of the computed y^+ values), hence it is not possible to invert eq. 2.10 to find the height of the first cell. In fact, this would require the knowledge of u_τ and, consequently, τ_w, which can not be computed in advance since the velocity profile at the wall (see eq. 2.12) is undetermined before starting the simulation.

To overcome this problem, the following well-known procedure has been performed.

- After having rearranged eq. 2.10:

  $$y^+ = \frac{y u_\tau}{\nu} \Rightarrow y = \frac{y^+ \nu}{u_\tau} \tag{2.16}$$

- u_τ is computed through eq. 2.11.

- The wall shear stress τ_w is now calculated from the skin frcition coefficient C_f:

  $$\tau_w = \frac{1}{2} C_f \rho U_\infty^2 \tag{2.17}$$

- Skin friction coefficient is computed through the flat plate case:

  $$C_f = 0.0576 Re^{-\frac{1}{5}} \tag{2.18}$$

Since Re is known, it is possible to compute eqs. 2.18 and 2.17 and insert everything in 2.10, obtaining the looked first cell height.

The chosen turbulent model was the $\kappa - \omega$ SST (Shear Stress Transport), a two-equation eddy-viscosity closure. This formulation, first developed by Menter [23], effectively blends the robust and accurate formulation of the standard $\kappa - \omega$ turbulence model in the near wall region with the $\kappa - \epsilon$ model behavior as the model switches to the latter away from the wall. In fact, the former performs much better than the latter for boundary layer flows, but is overly sensitive to the freestream value of ω, which is not the case for $\kappa - \epsilon$ model. This makes $\kappa - \omega \; SST$ suitable for a wide range of Reynolds number flows.

The $\kappa - \omega \; SST$ model has similar formulation to the standard $\kappa - \omega$ model developed by Wilcox [24]. The transport equations are defined as presented in eqs. 2.19:

$$\frac{\partial}{\partial t}(\rho\kappa) + \frac{\partial}{\partial x_i}(\rho\kappa u_i) = \frac{\partial}{\partial x_j}\left(\Gamma_\kappa \frac{\partial \kappa}{\partial x_j}\right) + G_\kappa - Y_\kappa + S_\kappa \tag{2.19a}$$

$$\frac{\partial}{\partial t}(\rho\omega) + \frac{\partial}{\partial x_i}(\rho\omega u_i) = \frac{\partial}{\partial x_j}\left(\Gamma_\omega \frac{\partial \omega}{\partial x_j}\right) + G_\omega - Y_\omega + D_\omega + S_\omega \tag{2.19b}$$

G_κ represents the generation of turbulence kinetic energy due to the mean velocity gradients, calculated according eq. 2.20.

$$G_\kappa = -\rho\overline{u_i' u_j'}\frac{\partial u_j}{\partial x_i} \tag{2.20}$$

The effective diffusivities for the $\kappa - \omega$ model are given by eqs. 2.21.

$$\Gamma_\kappa = \mu + \frac{\mu_t}{\sigma_\kappa} \tag{2.21a}$$

$$\Gamma_\omega = \mu + \frac{\mu_t}{\sigma_\omega} \tag{2.21b}$$

Where σ_κ and σ_ω are the turbulent Prandtl numbers for κ and ω respectively and they are calculated as:

$$\sigma_\kappa = \frac{1}{\frac{F_1}{\sigma_{\kappa,1}} + \frac{(1-F_1)}{\sigma_{\kappa,2}}} \tag{2.22a}$$

$$\sigma_\omega = \frac{1}{\frac{F_1}{\sigma_{\omega,1}} + \frac{(1-F_1)}{\sigma_{\omega,2}}} \tag{2.22b}$$

The turbulent viscosity μ_t in eq. 2.21 is computed as in eq. 2.23.

$$\mu_t = \frac{\rho\kappa}{\omega}\frac{1}{max\left(\frac{1}{\alpha^*}, \frac{SF_2}{a_1\omega}\right)} \tag{2.23}$$

Where S represents the strain rate magnitude .

For more details about the model refer to [23].

OpenFOAM Meshing

Keeping in mind y^+ considerations made above, meshing procedure was performed.

In OpenFOAM this is done acting in the meshdict file inside the case system directory. Two different meshing strategies have been tried in this study:

1. **snappyHexMesh:** The *snappyHexMesh* utility generates 3-dimensional meshes containing hexahedra (hex) and split-hexahedra (split-hex) automatically from triangulated surface geometries, or tri-surfaces, in Stereolithography (STL) or Wavefront Object (OBJ) format. The mesh approximately conforms to the surface by iteratively refining a starting mesh and morphing the resulting split-hex mesh to the surface. An optional phase will shrink back the resulting mesh and insert cell layers. The specification of mesh refinement level is very flexible and the surface handling is robust with a pre-specified final mesh quality. It runs in parallel with a load balancing step every iteration. It is rapid, simple to use, able to depict complicate geometries with ease.
2. **blockMesh:** The `blockMesh` utility creates parametric meshes with grading and curved edges. The principle behind *blockMesh* is to decompose the domain geometry into a set of one or more three dimensional, hexahedral blocks. Edges of the blocks can be straight lines, arcs or splines. The mesh is ostensibly specified as a number of cells in each direction of the block, sufficient information for *blockMesh* to generate the mesh data. Each block of the geometry is defined by 8 vertices, one at each corner of a hexahedron. The vertices are written in a list so that each vertex can be accessed using its label, remembering that OpenFOAM always uses the C++ convention that the first element of the list has label ‘0’. An example block is shown in Figure 2.4.

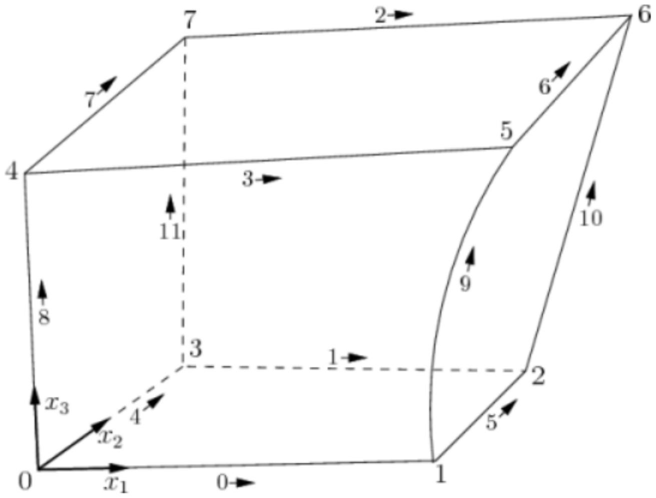
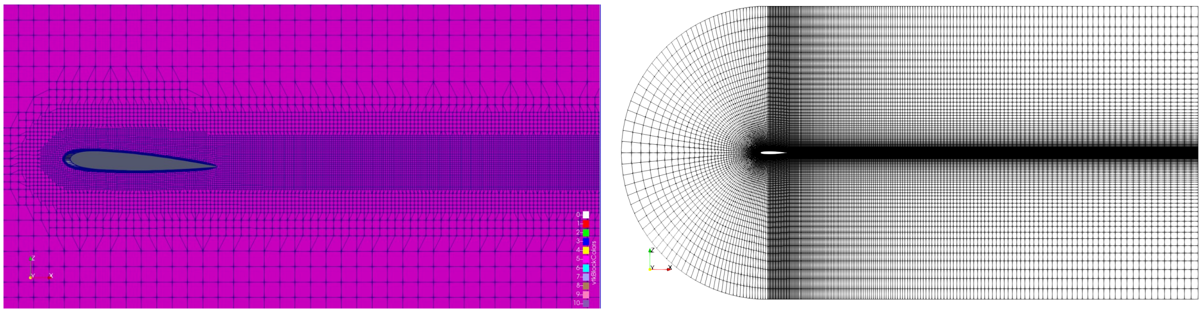


Figure 2.4: Example of a single *blockMesh* block [25].

Each block has a local coordinate system (x_1, x_2, x_3) that must be right-handed. A right-handed set of axes is defined such that to an observer looking down the O_z axis, with O nearest them, the arc from a point on the O_x axis to a point on the O_y axis is in a clockwise sense.

Results of the two meshings are shown in Figure 2.5.

After various attempts, despite being trickier to implement, *blockMesh* was chosen in favor of *snappyHexMesh* utility. In fact, for simple geometries such as an airfoil, the former is much more ordered and neater and, for a given number of cells, works better than the latter (i.e. smaller y^+ are obtained).



(a) *snappyHexMesh.* (b) *blockMesh.*

Figure 2.5: `snappyHexMesh` vs `blockMesh` utility meshing.

Chord was set equal to 1 m (see Figure 2.6), while air properties used in the simulations are listed in Table 2.1.

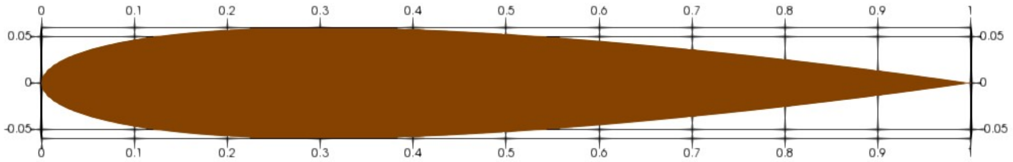


Figure 2.6: Airfoil dimensions.

Table 2.1: Air properties during CFD simulations.

Property	**Value**
γ	1.4
R	$287.05\,\mathrm{J\,kg^{-1}\,K^{-1}}$
T	$288.15\,\mathrm{K}$
ρ	$1.255\,\mathrm{kg\,m^{-3}}$
μ	$1.82 \times 10^{-5}\,\mathrm{Pa\,s}$

The domain had to be extended enough to capture wake effects, as noticeable in Figure 2.7, which is shown again for the sake of clarity. Its dimensions are reported in Table 2.2.

To give an idea of the refinement needed close to the object, an example of the `zGrading`[6] needed to have y^+ in the desired range for $Ma = 0.75$ (which translates into a $Re \approx 16.5 \cdot 10^6$) is hereby reported:

$$z_g = 559 \Rightarrow h_{wall} = 0.000373m \Rightarrow y^+ = 100 \tag{2.24}$$

This means that a conspicuous number of cells is needed to correctly run the simulation (in the case presented approximately 66000), which entailed some hours of computing time

---

[6] `zGrading` represents how many times the most external cell has to halved to obtain the height of the first wall cell.

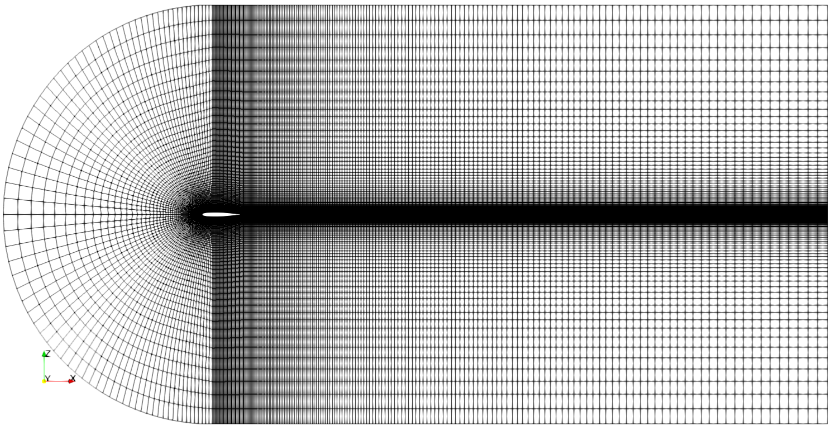


Figure 2.7: Final meshed domain.

Table 2.2: Domain dimensions.

Coordinate	**Value**
x_{min}	$-5\,\mathrm{m}$
x_{max}	$15\,\mathrm{m}$
z_{min}	$-5\,\mathrm{m}$
z_{max}	$5\,\mathrm{m}$
y_{min}	$-0.1\,\mathrm{m}$
y_{max}	$0.1\,\mathrm{m}$

in the Portable Computer used to perform the study. A magnification of the mesh is given in Figure 2.8.

More details about meshing and the 2D case in general can be found in Appendix A.

Comparative plots

After numerous refinements, CFD case has been validated at a low Mach number (0.15), comparing it to consolidated data. The outcome is depicted in Figure 2.9.

As noticeable from Figure 2.9(b), C_l trend confirms a very fine tune of the CFD case. On the other hand, the same can not be said about C_d in Figure 2.9(a) where consistent difference exists between OpenFOAM calculations and Abbot and XFOIL data. Nevertheless, this is not unexpected, since in the CFD the BL (Boundary Layer) is considered turbulent along all the chord, while XFOIL does not force transition. In fact, it is enough to look at the plot corresponding to Ladson experimental data with the roughest carborundum strips at 5% of the chord - which translates into a forced transition from that point on - to notice that the results are more similar. The still existent differences between the cyan and black plots are considered acceptable and caused by the inevitable use of wall functions, (instead of solving the BL with $y^+ < 1$, as suggested in [21] to better depict drag, almost impossible to achieve with a normal Portable Computer for such Mach numbers) and because of the applied turbulence model not being transitional but, as already said, turbulent along all the chord. In addition to that, the comparison has been performed at higher Mach numbers only between lift coefficients, for which the CFD has been proven

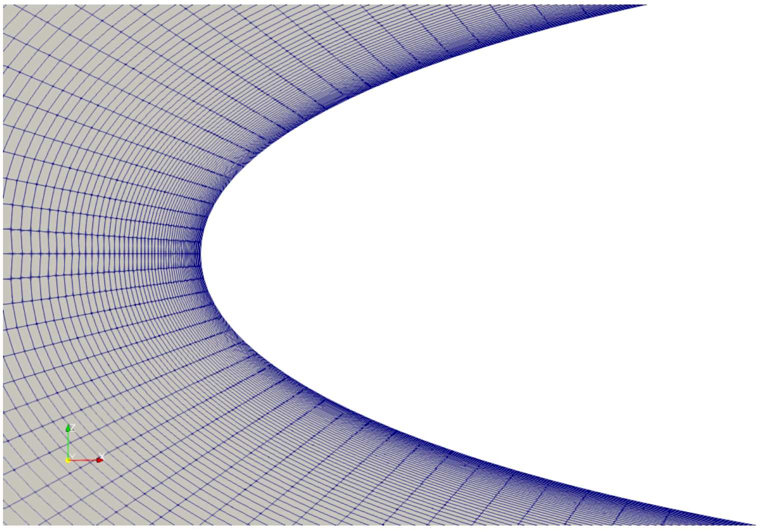
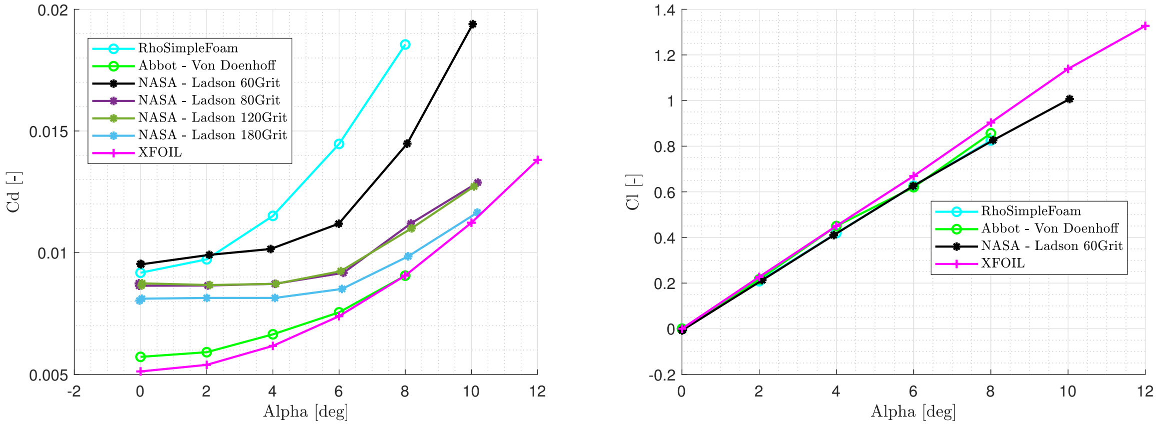


Figure 2.8: Mesh magnification.



(a) C_d *vs* α *(*$Ma = 0.15$*,* $Re = 3.5 \cdot 10^6$*).*

(b) C_l *vs* α *(*$Ma = 0.15$*,* $Re = 3.5 \cdot 10^6$*).*

Figure 2.9: C_l vs α and C_d vs α at $Ma = 0.15$.

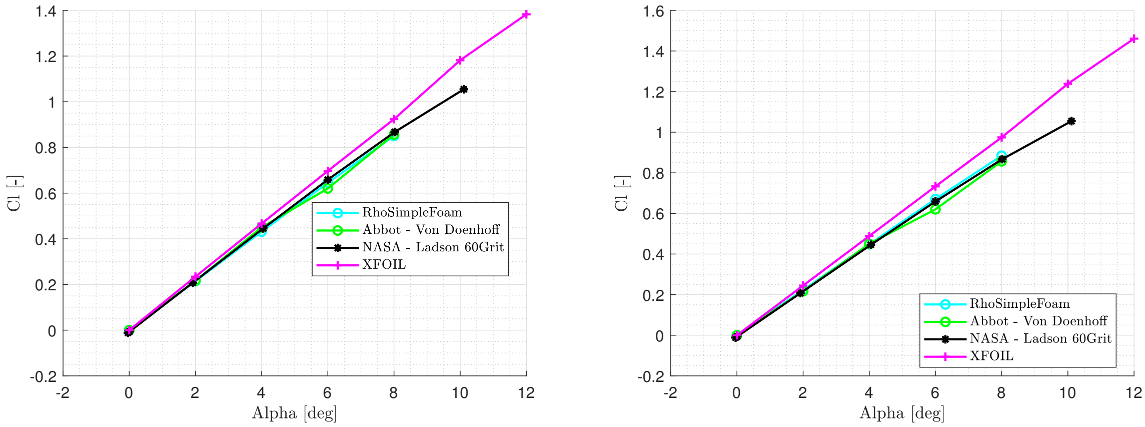
to work greatly.

`RhoSimpleFoam` solver has been used without Transonic Correction (TC) until $Ma = 0.55$, which is a steady-state solver for turbulent flows of compressible fluids. From that point on, `RhoSimpleFoam` solver with TC on plus `RhoPimpleFoam` solver (with TC applied as well) results were used as reference for the comparison with XFOIL. The latter is a transient solver for turbulent flow of compressible fluids based on OpenFOAM PIMPLE algorithm, which is a blending between PISO (Pressure Implicit with Splitting of Operators) and SIMPLE (Semi-Implicit Method for Pressure-Linked Equations).

When higher than 0.4 Mach numbers are reached, Abbot and Ladson data are no longer shown in the plots, since their experiments were performed under incompressible conditions.

In Figure 2.10 comparison at $Ma = 0.25$ and $Ma = 0.35$ is shown. As expected, all methods give similar outputs. It is worth to mention how close to experimental sources ([3], [20]) is the CFD solver `RhoSimpleFoam`.

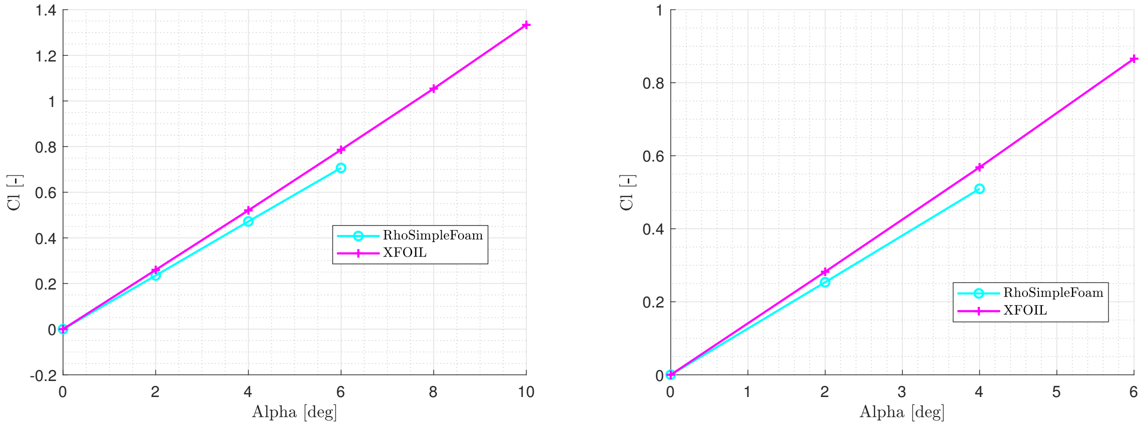
In Figure 2.11 comparison at $Ma = 0.45$ and $Ma = 0.55$ is reported. As already



(a) C_d *vs* α *(*$Ma = 0.25$*,* $Re = 5.5 \cdot 10^6$*).*

(b) C_l *vs* α *(*$Ma = 0.35$*,* $Re = 7.5 \cdot 10^6$*).*

Figure 2.10: C_l vs α at $Ma = 0.25$ and $Ma = 0.35$.



(a) C_l *vs* α *(*$Ma = 0.45$*,* $Re = 10 \cdot 10^6$*).*

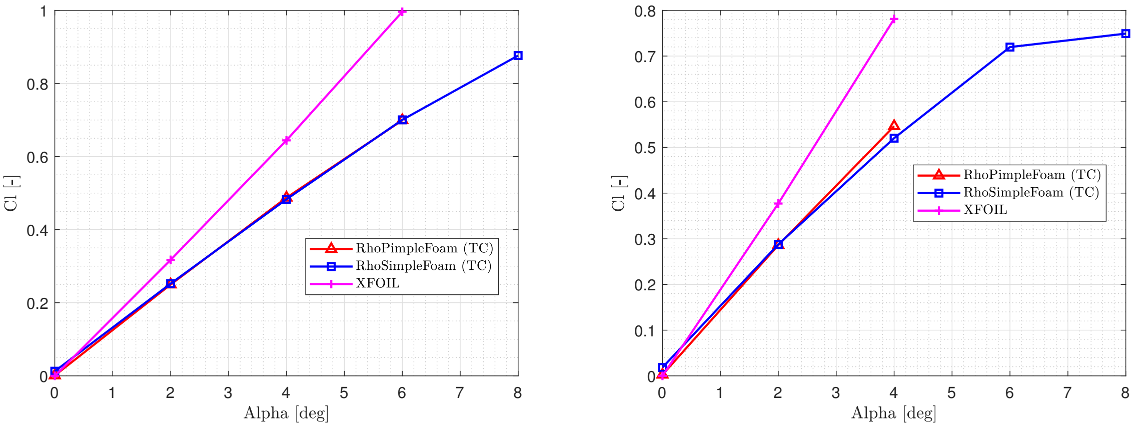
(b) C_l *vs* α *(*$Ma = 0.55$*,* $Re = 12.5 \cdot 10^6$*).*

Figure 2.11: C_l vs α at $Ma = 0.45$ and $Ma = 0.55$.

mentioned, plots corresponding to experimental data are no longer shown. It is noticeable how XFOIL is still capable of obtaining valid results even over the $Ma = 0.5$ threshold used in [2].

The same can not be said for Figure 2.12, where comparison at $Ma = 0.65$ and $Ma = 0.75$ is displayed. It is clear that XFOIL starts to depart from the calculations of the two solvers *`RhoSimpleFoam`* and *`RhoPimpleFoam`* with TC on. In particular, it overestimates the aerodynamic coefficient values. This is due probably to the Karman-Tsien compressibility correction applied by the software [11], reported in eqs. 1.1. Inspecting its expression, it is possible to notice that it formally breaks down in supersonic conditions. This is even more clear when looking at much simpler correction models, such as the Prandtl-Glauert one, according to which compressible flow field coefficients can be found by scaling the incompressible flow field coefficients by the factor reported in eq. 2.25.

$$\frac{1}{\beta} = \frac{1}{\sqrt{1 - Ma_\infty^2}} \tag{2.25}$$



(a) C_l *vs* α *(*$Ma = 0.65$*,* $Re = 14.5 \cdot 10^6$*).*

(b) C_l *vs* α *(*$Ma = 0.75$*,* $Re = 16.5 \cdot 10^6$*).*

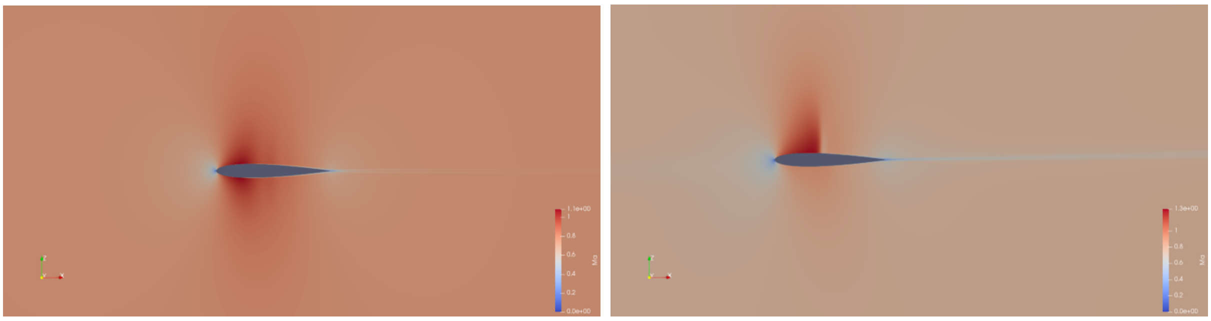
Figure 2.12: C_l vs α at $Ma = 0.65$ and $Ma = 0.75$.

These compressible flow correction theories work reasonably well up to the point where supersonic flow starts to appear near the surface of the airfoil section. This will happen before the free-stream becomes supersonic due to the acceleration of the air in the vicinity of the airfoil.

The free-stream Mach number for which supersonic flow first occurs on the wing or section is called the low critical Mach number. Not only is this an important limit for theory but also marks the start of transonic flow and the likelihood of a significant drag rise for the section. This phenomenon is reported in Figure 2.13.

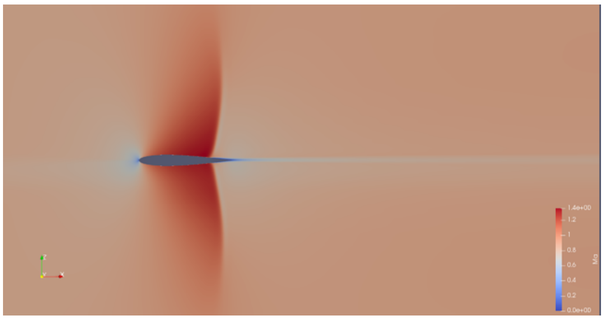
It is clearly visible in Figure 2.13(a) how in certain points of the airfoil surface, Mach values as high as 1.1 are reached. This is even more evident for higher AoA, like in Figure 2.13(b), or higher Mach numbers, like in Figure 2.13(c). From these images it is possible to deduce that, at $\alpha = 0°$, critical Mach number must be around 0.7.

For the sake of completeness, corresponding pressure field is reported in Figure 2.14, from which similar considerations can be made.



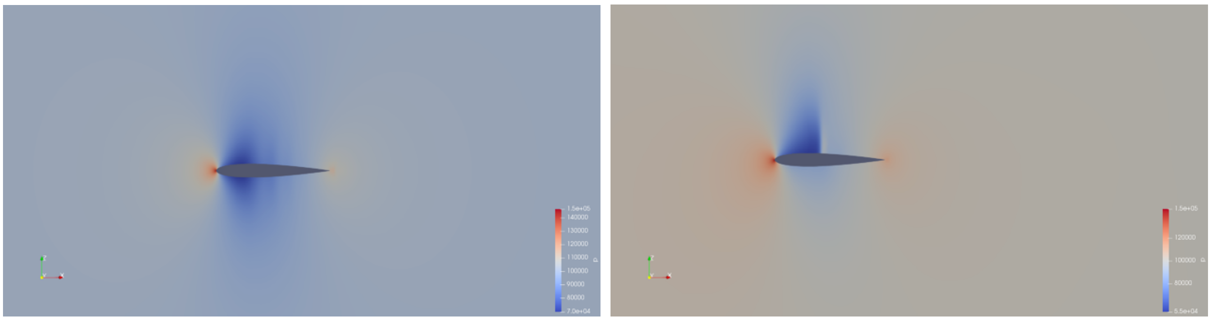
(a) *Airflow Mach number for* $Ma_\infty = 0.75$ *and* $\alpha = 0°$.

(b) *Airflow Mach number for* $Ma_\infty = 0.75$ *and* $\alpha = 2°$.



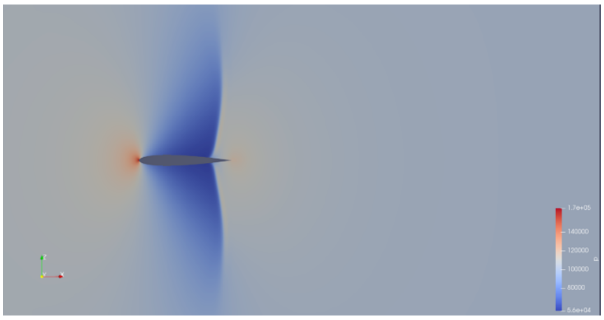
(c) *Airflow Mach number for* $Ma_\infty = 0.85$ *and* $\alpha = 0°$.

Figure 2.13: Airflow Mach number around the airfoil for different angles and Ma_∞.



(a) *Pressure field for* $Ma_\infty = 0.75$ *and* $\alpha = 0°$.

(b) *Pressure field for* $Ma_\infty = 0.75$ *and* $\alpha = 2°$.



(c) *Pressure field for* $Ma_\infty = 0.85$ *and* $\alpha = 0°$.

Figure 2.14: Pressure field around the airfoil for different angles and Ma_∞.

In conclusion, it can be stated that, after numerous initial refinements, CFD yielded satisfactory results compared to experimental data and XFOIL at low Mach, so it could be used as reference for higher Mach numbers. At $Ma \approx 0.6$ XFOIL results started to depart from those of the CFD, particularly at high AoA, where the flux began to reach sound speed. For this reason, aerodynamic database by De Zubeldia has been enhanced using as limit $Ma = 0.6$ and extended to other airfoils since the XFOIL "limit of reliability" had been found. From that point on, experimental data gathered by Harris [19] for the NACA0012 section have been used, as it was in the old version. In first approximation, this procedure has been considered satisfactory, since only in small (external) portions of the blade and for few values of V_∞ Mach number was over 0.6. Re an Ma discretization adopted was the one used by De Zubeldia described in Section 2.1.4.

Currently, NACA16, Clark, RAF and ARA-D series are the most common airfoil used in propellers [26]. Database extension was intended to make the program more versatile, hence it started including the aforementioned airfoils. Thanks to the online database Airfoil Tools [27] it was possible to retrieve `.dat` files containing the X,Y coordinates, as mentioned at the start of Section 2.2.2.

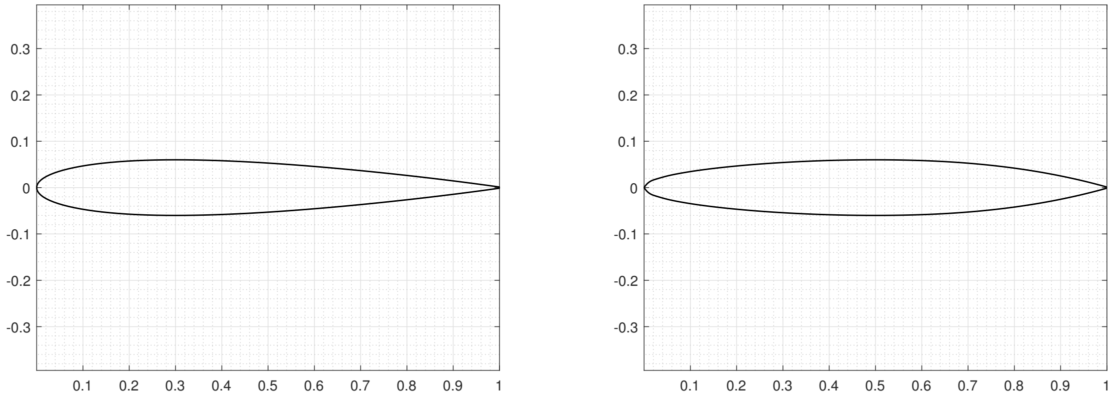
In Table 2.3 are reported the exact airfoils chosen to enhance the database, along with their main characteristics.

Table 2.3: Database airfoil list.

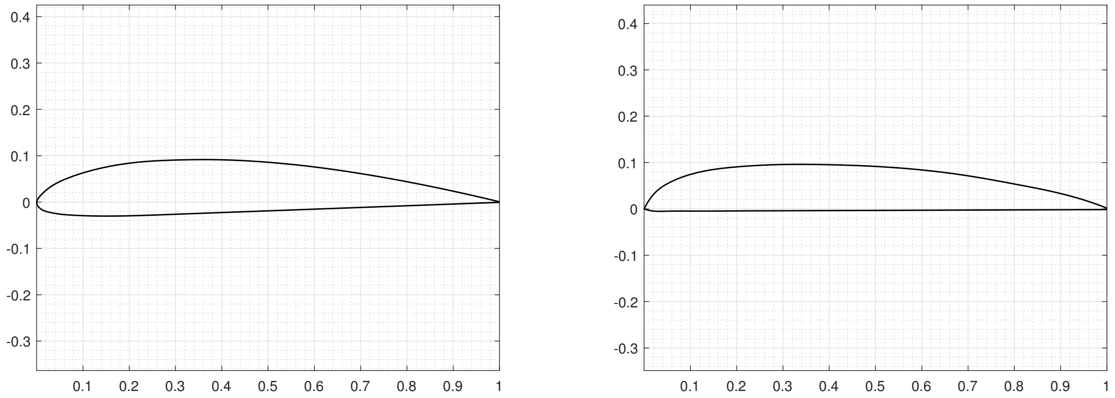
Airfoil	**Symmetric**	**Max Chamber**	**Max Thickness**
NACA 0012[†]	Yes	0%	12%
NACA16-012	Yes	0%	12%
Clark-Y	No	3.4%	11.7%
RAF-6	No	4.6%	10%

[†] *Already present in De Zubeldia's database.*

In Figure 2.15 are shown the contours of their sections.



(a) *NACA0012 airfoil contour.* (b) *NACA16-012 airfoil contour.*



(c) *Clark-Y airfoil contour.* (d) *RAF-6 airfoil contour.*

Figure 2.15: Database airfoil contours.

2.3 Rotational effects

As introduced in Sections 1.1 and 1.2, it is a known fact that, due to rotation of the blade, lift coefficients on inboard sections may exceed the 2D maximum lift coefficient. Therefore, there is a need for correction of the airfoil characteristics in the new formulation to include the 3D flow effects. In this Section, two different approaches are reported, that of Du and Selig [10] and Chaviaropoulos and Hansen [7]. Both of them have been implemented in the code and had different outcomes, as will be soon shown.

2.3.1 Du and Selig method

In their analysis [10], the authors aimed to describe and analyze the fundamental flow phenomena that characterize the boundary layer on rotating blades, and to develop a preliminary stall delay model that modifies the 2D airfoil data so as to simulate the 3D stall-delay effects. The following steps were taken in the development of the model:

1. Analysis of the 3D integral boundary layer equations for a reference system rotating with the blade.

2. Description of the effects of rotor rotation on the separation point and its causes.
3. Determination of a simple correction formula - based on theory developed in points 1, 2 - to obtain rotating rotor lift coefficient $C_l(\alpha)$ and drag coefficient $C_l(\alpha)$ from 2D airfoil data.

Passages of points 1, 2 are rather lengthy and can be found reported in detail in Appendix B.1. It is clear from this theory that boundary layer separation is postponed because of rotation, which implies that the stall angle of attack with rotation effects is larger than that in the 2D flow. Therefore, the stall delay model should account for these changes in the 2D C_l and C_d airfoil data. Moreover, to be readily implemented in BEMT methods, such as that used in the presented code, a simple correction to 2D airfoil data for 3D rotation effects is the most advantageous. It is assumed that the 3D airfoil data are approximately equal to that in 2D conditions, obtained through XFOIL, plus an increment in C_l and a decrement in C_d so as to simulate rotation effects on each radial station of the wind turbine blade. If the ΔC_l and ΔC_d denote the increments and the decrements, the 3D airfoil data can be expressed as reported in eq. 2.26.

$$C_{l,3D} = C_{l,2D} + \Delta C_l \tag{2.26a}$$

$$C_{d,3D} = C_{d,2D} - \Delta C_d \tag{2.26b}$$

Based on the stall-delay principle described in Appendix B.1, the correction formula takes the form reported in eqs. 2.27.

$$\Delta C_l = f_l \left(C_{l,p} - C_{l,2D}\right) \tag{2.27a}$$

$$\Delta C_d = f_d \left(C_{d,2D} - C_{d,0}\right) \tag{2.27b}$$

Where $C_{l,p} = 2\pi\left(\alpha - \alpha_0\right)$ is the potential theory lift coefficient and $C_{d,0} = C_{d,2D}$ for $\alpha = 0$.

The functions f_l and f_d are modeled after the separation factor showed in eq. B.11. Their expression is reported in eqs. 2.28.

$$f_l = \frac{1}{2\pi}\left[\frac{1.6\left(\frac{c}{r}\right)}{0.1267}\frac{a' - \left(\frac{c}{r}\right)^{\frac{d}{\Lambda}\frac{R}{r}}}{b + \left(\frac{c}{r}\right)^{\frac{d}{\Lambda}\frac{R}{r}}} - 1\right] \tag{2.28a}$$

$$f_d = \frac{1}{2\pi}\left[\frac{1.6\left(\frac{c}{r}\right)}{0.1267}\frac{a' - \left(\frac{c}{r}\right)^{\frac{d}{2\Lambda}\frac{R}{r}}}{b + \left(\frac{c}{r}\right)^{\frac{d}{2\Lambda}\frac{R}{r}}} - 1\right] \tag{2.28b}$$

where a', b, and d are empirical correction factors. In this study, a', b, and d were initially set to unity, as suggested by Du and Selig. Also, the parameter λ has been modified and replaced by a modified tip speed ratio Λ, whose expression is reported in eq. 2.29.

$$\Lambda = \frac{\Omega R}{\sqrt{V_\infty^2 + \left(\Omega R\right)^2}} \tag{2.29}$$

Furthermore, in f_d, a factor of 2 was introduced multiplying Λ, not present in f_l. According to [10], these changes fitted best the experimental data used in the study.

To summarize, the 3-D stall-delay model final relations are expressed in eqs. 2.30.

$$C_l = C_{l,2D} + f_l \left(C_{l,p} - C_{l,2D}\right) \tag{2.30a}$$
$$C_d = C_{d,2D} - f_d \left(C_{d,2D} - C_{d,0}\right) \tag{2.30b}$$

The model was tested by the authors applying it to existing data sets and proved to be in good agreement with measurements.

2.3.2 Chaviaropoulos and Hansen method

The path pursued by the authors in [7] is pretty similar to what described in the previous Section 2.3.1. In particular, the following steps were taken:

1. Three-dimensional and rotational viscous effects on wind turbine blades were investigated by means of a quasi-3D Navier-Stokes model, whose governing equations of the model are derived from the 3D primitive variable Navier-Stokes equations written in cylindrical coordinates in the rotating frame of reference.

2. Validity of the assumptions made in the construction of the model were cross-checked through fully 3D Navier-Stokes calculations.

3. Equations of the model were numerically integrated by means of a pressure correction algorithm for identifying the physical mechanism associated with the 3D and rotational effects and for establishing semi empirical correction laws for the load coefficients, based on 2D airfoil data.

Again, for the sake of brevity, details about points 1, 2 can be found in B.2.

The resulting quasi-3D model suggested that the two most important parameters that trigger three-dimensional and rotational effects are the local chord by radii ratio and the twist angle. Looking for a semi empirical correction law for the load coefficients, the idea of Snel et al. [6] of expressing the 3D correction of the lift coefficient as a fraction of the difference ΔC_l between the inviscid value $C_{l,inv}$ and the corresponding 2D value $C_{l,2D}$ was adopted by the authors. This is similar to what done by Selig and Du, as shown in previous Section 2.3.1 and $C_{l,inv}$ is nothing but $C_{l,p}$ of eq. 2.27a.

The obtained results for Cd from the quasi-3D model induced the authors to believe that a similar correction law could be used for the 3D effect on the drag coefficient as well. In addition, the influence of the twist angle θ, not present in the analysis of Selig and Du seen in the previous Section 2.3.1, could be introduced through the ΔC_l multiplier. In light of the analysis conducted, Chaviaropoulos and Hansen proposed the correction law reported in eq. 2.31.

$$C_{x,3D} = C_{x,2D} + a \left(\frac{c}{r}\right)^h cos^n\theta \Delta C_x; \qquad x = l, d \tag{2.31}$$

Where

$$\Delta C_l = C_{l,inv} - C_{l,2D} \tag{2.32a}$$
$$\Delta C_d = C_{d,2D} - C_{d,2D_{min}} \tag{2.32b}$$

The three constants of the model a, h, and n were calibrated by the authors using a best-fit procedure based on the turbulent flow simulations run with the quasi-3D model described in Appendix B.2 for NACA 63-2-15 airfoil. The results of this fitting procedure are reported in Table 2.4.

Table 2.4: Fitted coefficients of Chaviaropoulos and Hansen model [7].

Coefficient	**Value**
a	2.2
h	1
n	4

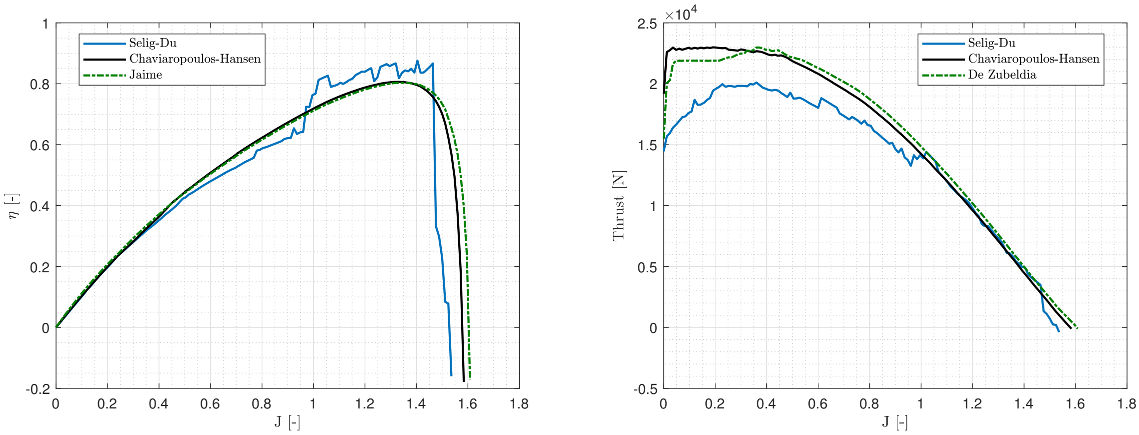
Correction formulas 2.31 and 2.32 were tested through a comparison between calculated and measured power curves on a Bonus 300 Combi stall regulated wind turbine, whose blade is based on NACA63-4XX and NACA63-2XX airfoils. It was shown that the suggested 3D corrections may improve considerably the accuracy of the predictions, in particular when the flow is separated.

2.3.3 Adopted method

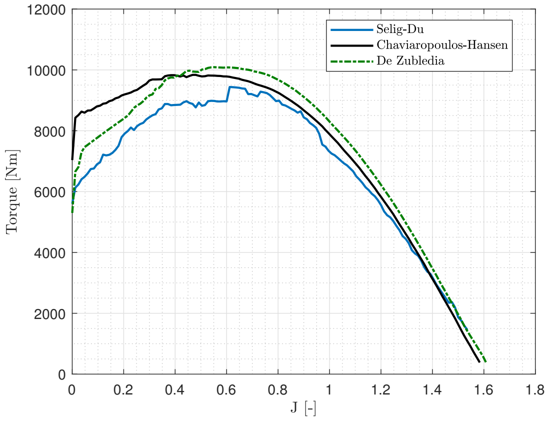
In both studies, the proposed methods showed promising results in the correction of the two-dimensional aerodynamic data. For this reason, the two of them have been applied to the present methodology and consequently implemented in the code, where their applicability was assessed.

Despite being less modern and not considering the local twist angle β as an additional parameter, Selig and Du method was the first one to be undertaken. From a software point of view, it was rather simple to apply but on the other hand it slowed down considerably the subroutine `SinglePerformance` already described in Section 2.1.2. In fact, the general strategy - for both cases - was, before performing an optimization loop (which entails a considerable amount of time even without modifications to the original code), to execute the single computation subroutine with a given geometry, radius and angular velocity and then compare the results given by the modified program and the original one.

This procedure showed on one hand that the code with Selig and Du model took almost one order of magnitude more in time compared to the unmodified one. This aspect alone would be already enough to question its applicability since normally one optimization run lasts around 2÷3 hours which would become 20÷30. On the other hand, the decisive factor that triggered its discard was the veracity of the outcomes. In particular, thrust and torque curves were irregular in a non-physical way. In addition to this, it happened in more than one simulation that thrust became zero for a higher advance ratio than the torque, consequently giving negative values of efficiency. This is obviously not possible in real conditions. In fact, decomposing the components of the aerodynamic force acting on the blades it is possible to notice that, keeping Ω constant, increasing the flight speed and, hence, the advance ratio, the force vector moves downwards until it aligns itself to the z-axis. In this condition no thrust is generated but a positive torque is still absorbed by the propeller. To reach the zero-torque condition, flight speed (and consequently the advance ratio) should be increased until the force vector is aligned to the x-axis and directed towards the tail of the aircraft. These are among the basic principles of propeller theory and can be found even in the most basic texts (e.g [28]). In Figure 2.16 it is shown a comparison between De Zubeldia's, Chaviaropoulos and Selig-Du methods in terms of performance outputs for a blade composed by Clark-Y airfoils. The geometry was the same in all cases, in particular chord distribution was linear - 0.85 m at the hub and 0 m at the tip - and twist distribution as well - 60° at the hub and 20° at the tip. Radius and angular velocity were kept constant too, measuring 2.5 m and 2000 rpm, respectively.



(a) *Efficiency.* (b) *Thrust.*



(c) *Torque.*

Figure 2.16: Selig-Du method outputs compared to old and new code - Clark-Y airfoil.

Non-physical oscillations in Selig-Du method are clearly visible in all the sub-figures. In addition to this, its thrust and torque curves numerical values are considerably different compared to the other two methods, which in turn are in accordance (expect in the low J region, where the new model has to supposedly improve the old code). These evident aspects make Selig-Du model non suitable for the subsequent analysis.

Chaviaropoulos and Hansen method was just as simple to implement from a software point of view yet it did not entailed a considerable slowdown in the code. As a matter of fact, given the geometry, radius and angular velocity of the propeller, the modified code computes its performance along the whole J envelope in almost the same time of the original one (the difference is of the order of seconds). In addition to this, the curves were smooth and the differences with respect to the ones computed through the unmodified code respected the theoretical considerations of the model (in particular point B.2 in Appendix B.2). In fact, for high J, where the AoA is low along all the blade and the flow is expected to be attached, the curves almost overlapped; on the other hand, in lower J regions, differences were more consistent as the flow is more likely to be detached and hence the 3D effects are bigger. This behavior is clearly visible in Figure 2.16 (black line) and is shown extensively during the parametric study of Section 3.1. Lastly, the blade twist angle influence is taken into account in the model, which is, as a consequence, intrinsically more complete.

In light of what just stated, the chosen model for the present formulation was that of Chaviaropoulos and Hansen. As noticeable from eq. 2.31, the correction depends on three constants. Since, for obvious reasons, it was not possible to perform a proper physical experiment and obtain reliable data to be used for constants tuning, it was decided to carry out a parametric study to investigate their effect on the correction law mentioned above. Results of this procedure are shown in Section 3.1.

Chapter 3

Numerical studies

After having defined the theoretical aspects of the model in Chapter 2, the new formulation has been implemented in MATLAB® environment. In this Chapter the outcomes of the numerical studies carried out with the software are presented. In the first part a parametric study of the a, h and n coefficients effect on the correction law (eq. 2.31) is carried out for all the database airfoils. Secondly, optimization loops with the enhanced code are performed and compared to the old one for the same initial guess in terms of blade geometry (twist and chord distributions) and diameter (D) and angular velocity (Ω). In addition to this, the presence of a variable pitch mechanism has been simulated changing by a (positive or negative) constant value the optimized twist distribution and computing the performance output for each resulting case. Lastly, a preliminary attempt to introduce in the program the possibility of having more than one airfoil section along the span has been performed.

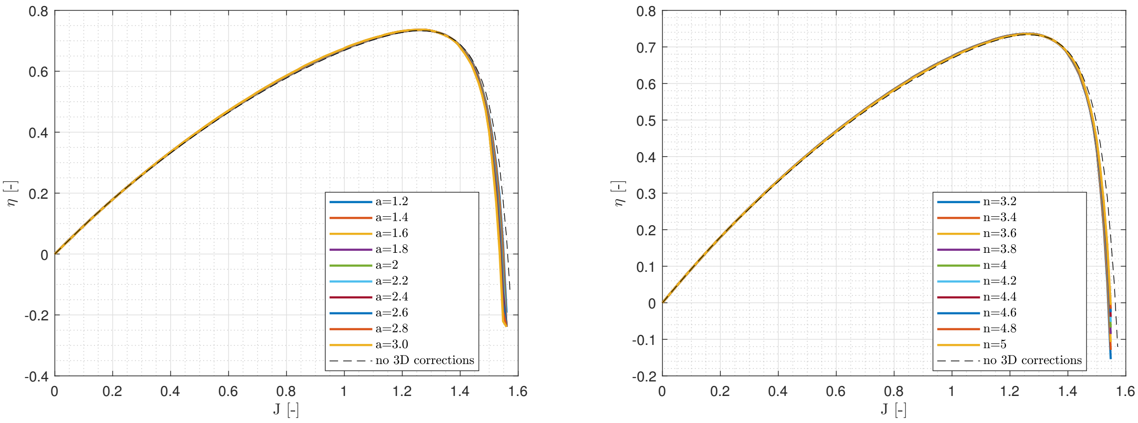
3.1 Parametric study

Keeping in mind the suggested values of Chaviaropoulos and Hansen (Table 2.4), a, h and n have been varied one at a time around their recommended value to see the effect of their change in terms of performance outcomes. During this procedure, when one coefficient was changed the other two remained constant and equal to the suggested values of Table 2.4. Geometrical parameters, diameter and angular velocity remained constant throughout all the tests. In particular, $D = 2.5\,\mathrm{m}$, $\Omega = 2000\,\mathrm{rpm}$ and $c = 0.5\,\mathrm{m}$. Twist distribution on the other hand was linear, ranging from 60 to 20 degrees at the tip.

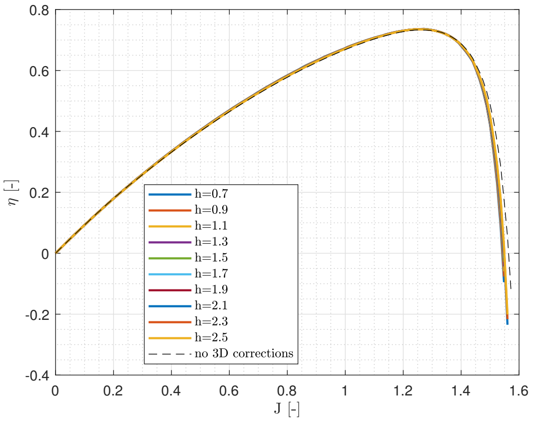
In Figures 3.1, 3.2 and 3.3 is presented the case of RAF-6 airfoil.

In general, it is possible to affirm that coefficients variation has little effect on the outcome. On the other hand, it is worth to notice that with respect to the unmodified case (black dashed line) sensible differences can be found for $J < 0.25$, while for higher J those differences become negligible. In particular, it is evident how the unmodified code underestimates both thrust (Figure 3.2) and torque (Figure 3.3) for low values of J, while afterwards results are pretty similar. This is in full accordance to “3D effects” theory, which states that rotational effects (hence differences between the two models) are bigger when the flow is more likely to be detached (i.e. low J regions to which correspond higher AoA).

For what concerns η, both models behave practically in the same manner. The reason is quite simple and is easily understandable looking at the mathematical expression of the



(a) *η variation for different a values.* (b) *η variation for different n values.*



(c) *η variation for different h values.*

Figure 3.1: RAF-6 airfoil: η variation for different corrective coefficients values.

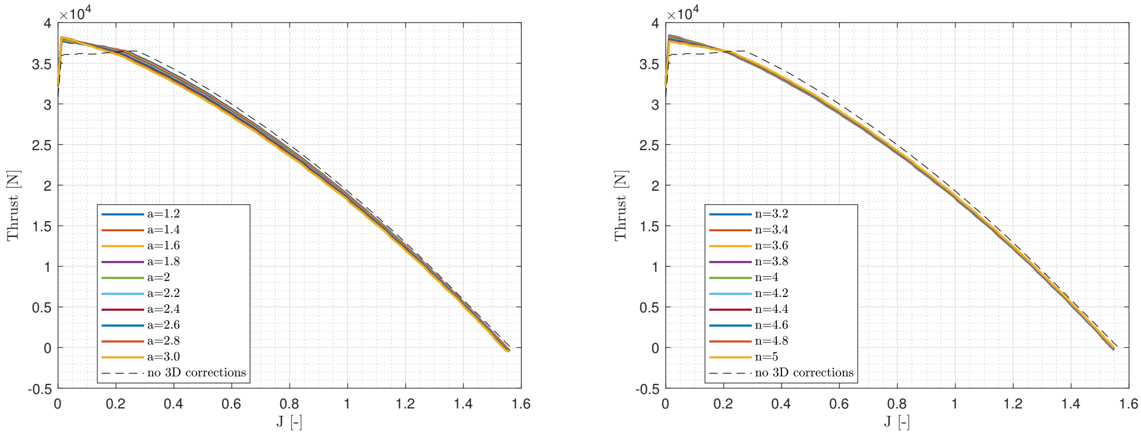
efficiency:

$$\eta = \frac{1}{2\pi} \frac{C_T}{C_Q} J \tag{3.1}$$

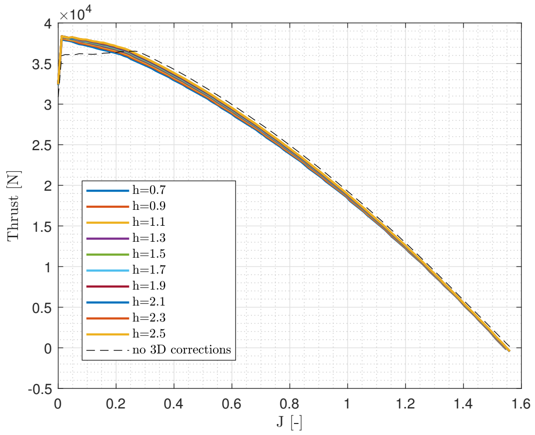
Where C_T and C_Q are the thrust and torque coefficients, respectively.

This means that if both Q and T are overestimated (or underestimated) at a given J, their ratio cancels out this difference and hence the plots are practically identical between the two models.

In the case of NACA16-012 airfoil (Figures 3.4, 3.5, 3.6), the considerations that can be made are roughly the same. This time, however, differences in terms of performance between the two models are more considerable, particularly for low J values.

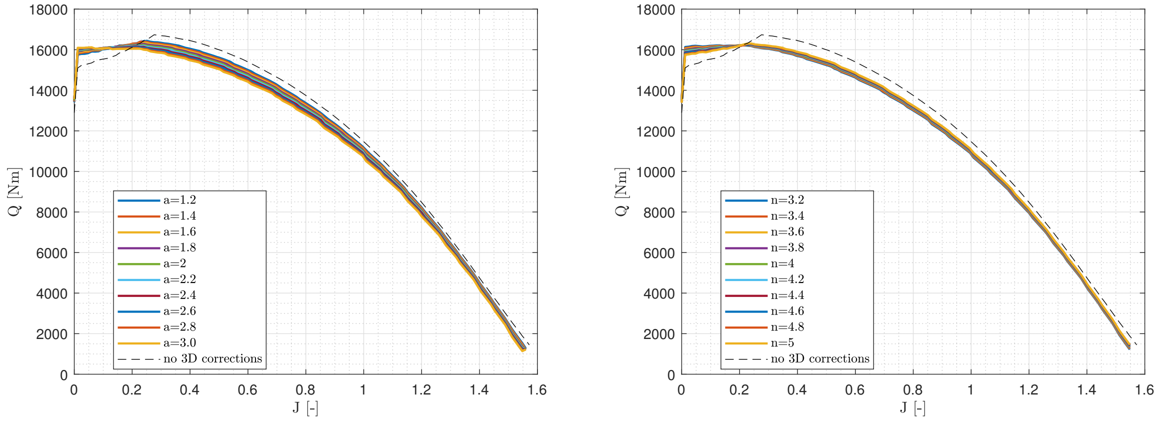


(a) *Thrust variation for different a values.* (b) *Thrust variation for different n values.*



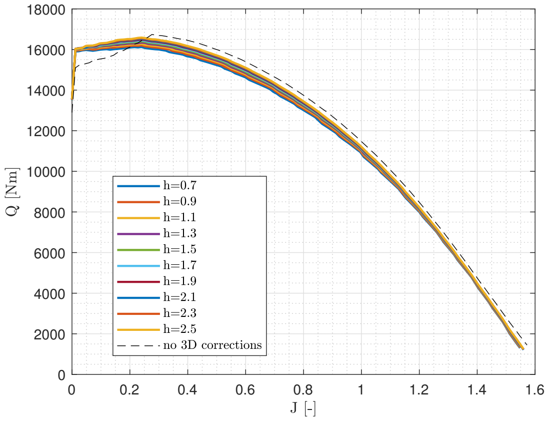
(c) *Thrust variation for different h values.*

Figure 3.2: RAF-6 airfoil: T variation for different corrective coefficients values.



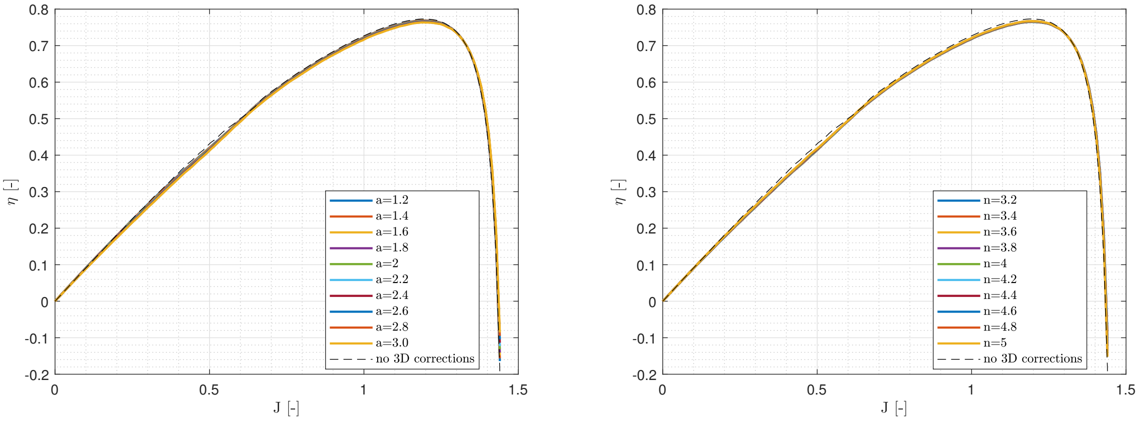
(a) *Torque variation for different a values.*

(b) *Torque variation for different n values.*



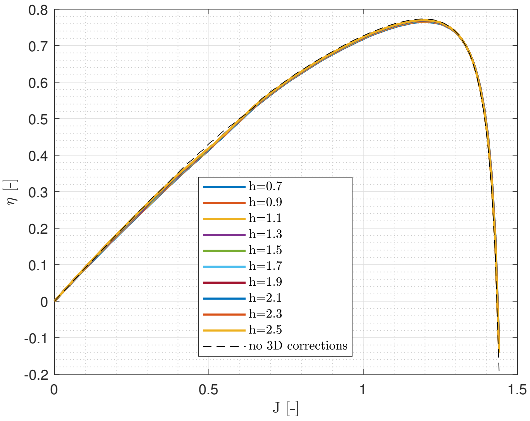
(c) *Torque variation for different h values.*

Figure 3.3: RAF-6 airfoil: Q variation for different corrective coefficients values.



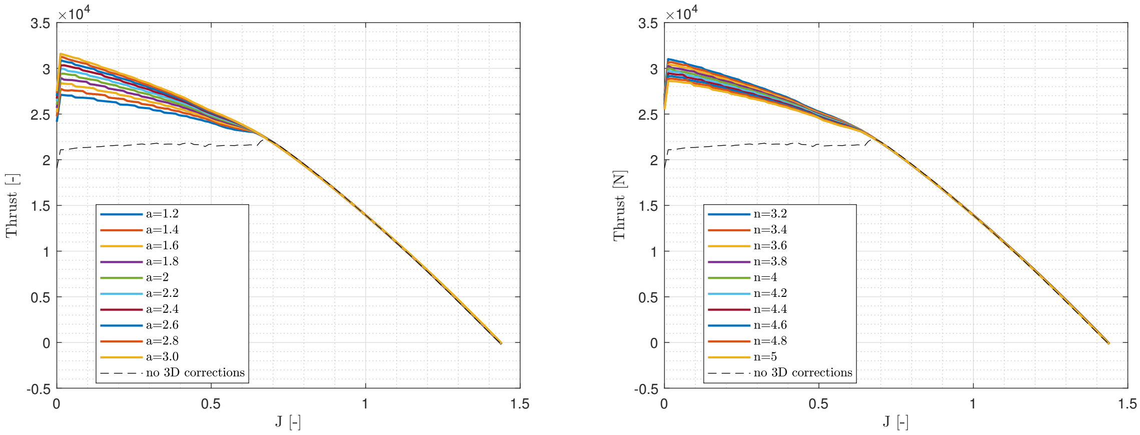
(a) *η variation for different a values.*

(b) *η variation for different n values.*



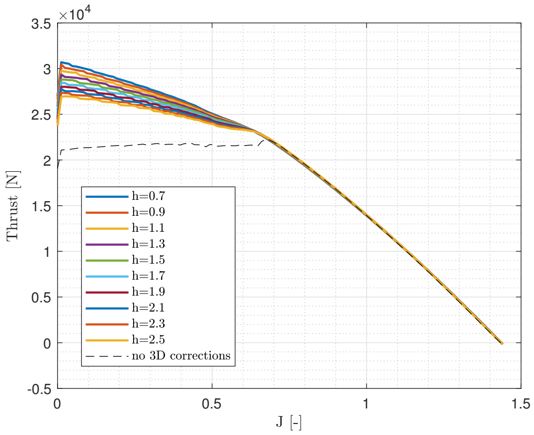
(c) *η variation for different h values.*

Figure 3.4: NACA16-012 airfoil: η variation for different corrective coefficients values.



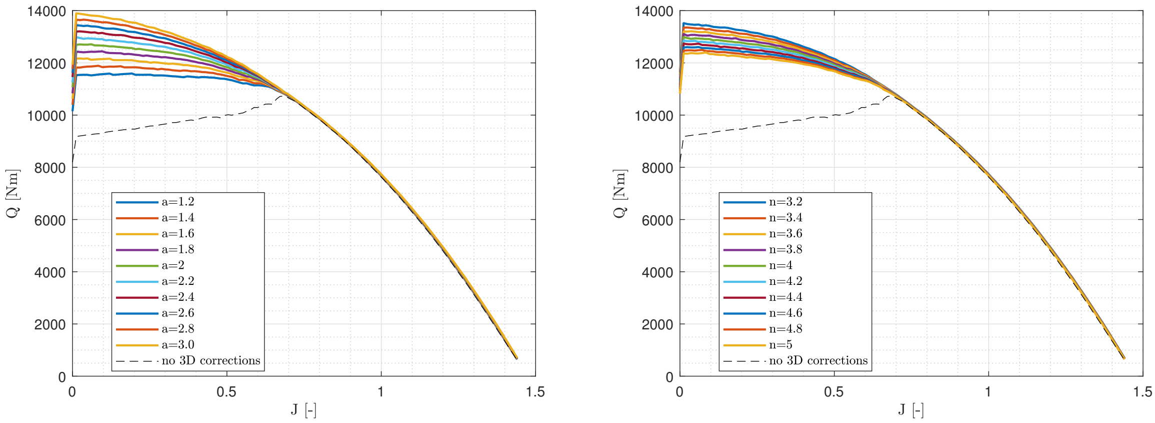
(a) *Thrust variation for different a values.*

(b) *Thrust variation for different n values.*

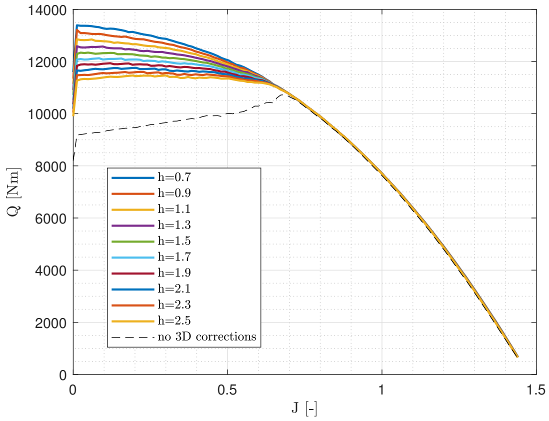


(c) *Thrust variation for different h values.*

Figure 3.5: NACA16-012 airfoil: T variation for different corrective coefficients values.



(a) *Torque variation for different a values.* (b) *Torque variation for different n values.*



(c) *Torque variation for different h values.*

Figure 3.6: NACA16-012 airfoil: Q variation for different corrective coefficients values.

It is possible to notice how, this time, the two models start to depart from each other at a much higher J (roughly 0.7). Inspecting the AoA along the blade for each J, it is noticeable that until an advance ratio of 0.7 their values exceed by far the 2D stall condition which entails an approximately constant power output. On the other hand, thanks to the rotational effects, it is known that stall occurs at a much higher value when the blade is rotating, causing the big gap in the plots in Figures 3.6 and 3.5. When, for higher J, AoA decreases under 2D stall condition, the adherence between the two codes is regained. This is again a confirmation of what stated by the theory.

For what concerns the efficiency, what stated before in the RAF-6 case still stands: the ratio between thrust and torque cancels out the difference between the two models and the result is pretty much equal (see Figure 3.4).

Again it is possible to affirm that coefficients variation has little effect on the outcome, but in this case difference at low J are more sensible, probably due to higher AoA reached with respect to RAF-6 case. This happens in particular for a and h coefficients (see Figures 3.5(a), 3.5(c), 3.6(a), 3.6(c)). In fact, looking to eq. 2.31 - which represents the correction introduced by the model - it is evident that their contribution is much higher than that of n since the latter is the exponent of a cosine, which can in turn assume values between [-1,1]. Raising to a power of something a number between [-1,1] gives again a number comprised in the same interval[1]. From here the reason of a minor influence of n coefficient. Despite this small discrepancies, the corrective model seems to work well.

For what concerns the Clark-Y airfoil case, the same considerations given for the RAF-6 apply. Graphical results are reported in Appendix C for the sake of brevity.

To summarize, the parametric study leads to the following considerations:

- The new code depicts thrust and torque behavior for low J values much better than the old one.
- For high J regions, where the AoA along the blade are lower and the flow is more likely to be attached, the two codes behave similarly, in accordance to theory.
- a, h and n coefficients variation does not change considerably the output given by the new code.
- The enhanced code can be used for the subsequent optimization studies and values of the corrective coefficients will be those of Table 2.4.

3.2 Design optimizations

Now that the modified program has been proven to be efficiently working in terms of performance estimation for a given geometry, diameter and angular velocity (`SinglePerformance` subroutine), it is worth analyzing what the differences with respect to the old code would be when performing optimization loops starting from identical initial guesses.

It is reminded that during the optimization the generated propeller must satisfy the following thrust constraints:

1. $T_{TO} > 27000N$ at $31\,\mathrm{m/s}$

---

[1] Except in the case of an even raising power. In that case the result assume values between [0,1].

2. $T_c > 6600N$ at 115 m/s
3. $Ma_{tip} < Ma_{lim}$
4. $Re_{min} < Re_i$
5. $B_{x0c} < \xi_{hub}$
6. $B_{x0t} < \xi_{hub}$
7. $1 < B_{x3c}$
8. $1 < B_{x3t}$
9. D <3 m

Limitations 1 and 2 represent thrust constraints during take-off and cruise conditions[2]. Constraint number 3 limits the maximum value of the Mach number at the blade tip to that of the maximum Mach number available in the database of polars. This constraint is directly limiting the diameter D of the propeller or, alternatively, the rotational speed Ω, as higher speeds are encountered in the tip due to high distance from the center of rotation or rotational speed. For what concerns point 4, it actually represents a group of n constraints, being n the number of stations. Reynolds number at each station is forced to be higher than the minimum Reynolds number present in the database. This constraint is particularly important as the curve that defines chord distribution may take lower than zero values along the iterations of the optimization routine, leading to negative chords. Since Reynolds number is a function of the chord, negative chords would result in negative Reynolds number at the section and the routine would utilize the polars (according to the procedure described in [2], Section 2.1.3) corresponding to the lowest Reynolds number available to compute aerodynamic coefficients, and therefore, the results obtained would not be realistic as they come from unfeasible chords. Constraints from 5 to 8 are related to the abscissa coordinate of the Bezier control points for both twist and chord distributions. They require that the first control point must be to the left of the first station (represented by the dimensionless hub coordinate ξ_{hub}) and the last one to the right of the blade end, in order to have a single value of twist or chord at each station. Constraint 9 is rather intuitive and has been posed considering the dimensions of an aircraft similar to the Beechcraft 1900.

The optimization variables are eight Bezier coefficients for each twist and chord distributions, plus propeller diameter and rotational speed. The value of the cost function is computed as follows:

- Bezier control points are transformed into curves (for more details refer to [2], Section 4.1).
- Performance is calculated in cruise conditions.
- Cost function value is computed through Equation 3.2.

---

[2] Numerical values come from [2], where the author states that the propeller is intended for a commuter aircraft similar to Beechcraft 1900.

$$CF = -\eta_c \tag{3.2}$$

With the minus in front since `fmincon` searches the minimum of a given function.

For what concerns the optimization algorithm, it was decided to use a gradient-descent sequential quadratic programming (SQP), which has been proven to be effective for this kind of optimization problems by De Zubeldia [2] and Sartori [29].

Note that - for all airfoils and different initial guesses - efficiency, thrust and torque outputs coming from the resulting optimized geometry were obtained using the subroutine `SinglePerformance` of the new code, since, as already shown, it is more precise and in this fashion it is possible to compare the results in the same plot. `SinglePerfomance` of the old code was called during the optimization routine performed through De Zubeldia's program to see how that would affect the geometrical distributions outputs. Once obtained the aforementioned outputs, there was no reason to evaluate their corresponding performance (η, T and Q) with the old subroutine, which has been proven to be less effective.

Considering the performance restrictions of points 1 and 2, it is expected that optimizations performed through the old code will result in a higher chord distribution since, neglecting rotational effects, it has been shown how it underestimates thrust output particularly at low J regions (which is a very likely condition during a take-off). In particular for NACA16-012 airfoil (see Figure 3.5). In addition to this, given the parametric analysis performed in Section 3.1, smaller differences in the optimization loops outputs between the old and new code are awaited for RAF-6 and Clark-Y airfoils.

3.2.1 Linear chord - linear twist initial guesses

The first optimization loop was performed starting with initial guess values reported in Table 3.1.

Table 3.1: First optimization: initial guess.

Variable	**Value**
D	2.5 m
Ω	2000 rpm
Chord distribution	Linear
Twist distribution	Linear

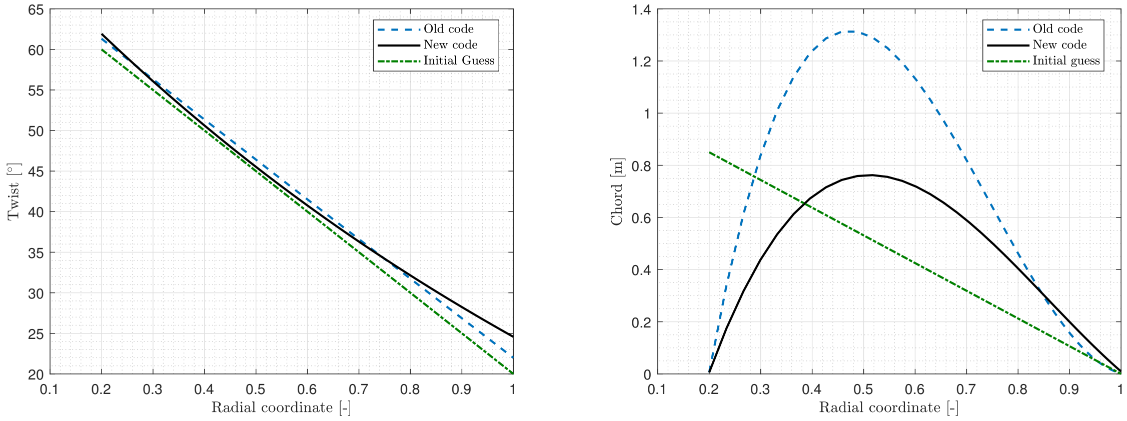
Exact values of geometrical distributions are displayed along with the results of the optimization in the following (Figures 3.7(b) and 3.7(a)).

In Figure 3.7 the results for NACA16-012 propeller[3] are displayed. Performance outputs are shown in Figure 3.8.

As expected, the optimization performed with the old code led to a much higher chord distribution (Figure 3.7(b)). In particular, at around half of the span the chord reaches values as high as 1.3 m, pretty unrealistic - or at least of questionable feasibility - for an aircraft of the size of Beech1900. Conversely, with the enhanced code the maximum chord section results almost halved (around 0.7 m). The thrust and torque outputs (Figures

---

[3] By "*airfoil type* propeller" it is meant a propeller made of sections corresponding to *airfoil type* airfoils.



(a) *Twist distribution.* (b) *Chord distribution.*

Figure 3.7: NACA16-012 airfoil geometrical optimization results - linear chord initial guess: old code vs new code.

3.8(b) and 3.8(c)) resulting from that geometry are slightly higher than those obtained from the solution obtained with the new code, which, nonetheless, verifies thrust constraints and presents a better efficiency throughout all J envelope (Figure 3.8(a)). Twist distributions assumed rather similar values for both codes and did not diverge particularly from the initial guess.

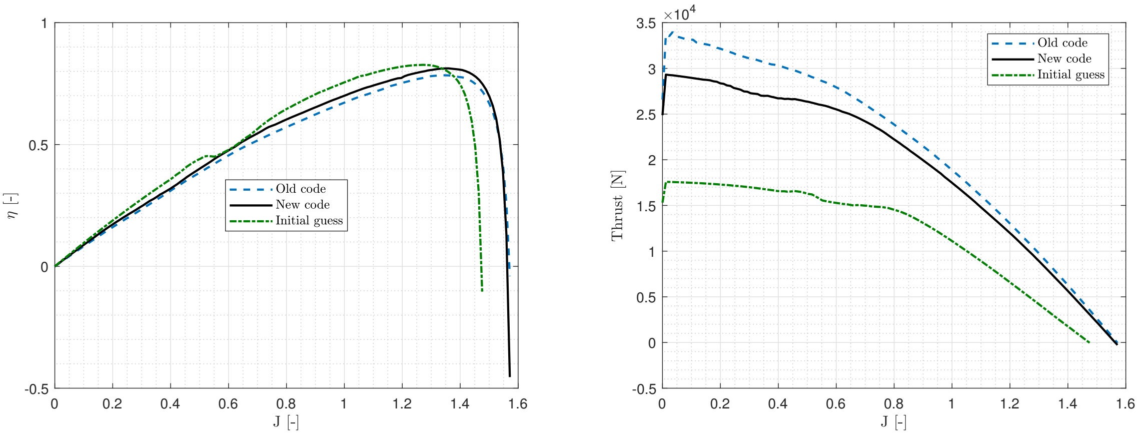
It is interesting to notice how the initial guess presents a good efficiency for almost every J but its thrust output is far from being sufficient to satisfy the constraints (in particular the take-off 1). In addition to this it is worth mentioning how its maximum J reached is slightly inferior to those coming from the optimizations through both codes, in accordance to the fact that higher twist distributions are more suitable for higher J regimes. In fact, looking at Figure 3.7(a), both the optimization outputs present a fairly increased twist along all the span, hence null thrust and efficiency values are reached at a higher J.

Finally, for what concerns the resulting D and Ω they are, likewise to twist distribution case, pretty similar for both codes and close to the initial guess. The results are listed in Table 3.2.

Table 3.2: First optimization-NACA16-012: D and Ω results.

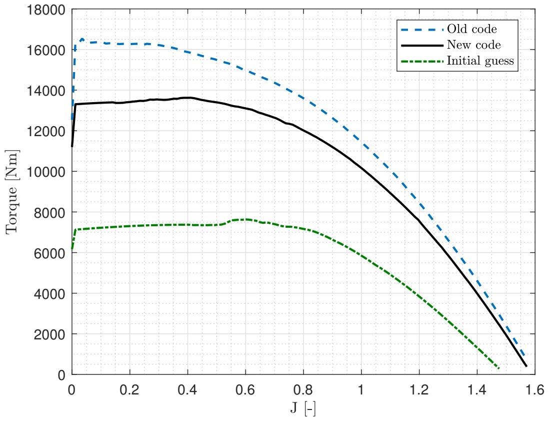
Version	D	Ω
Old	2.5584 m	2000 rpm
Enhanced	2.5586 m	1999 rpm

In Figure 3.9 the results for RAF-6 propeller are displayed. Performance outputs are shown in Figure 3.10.



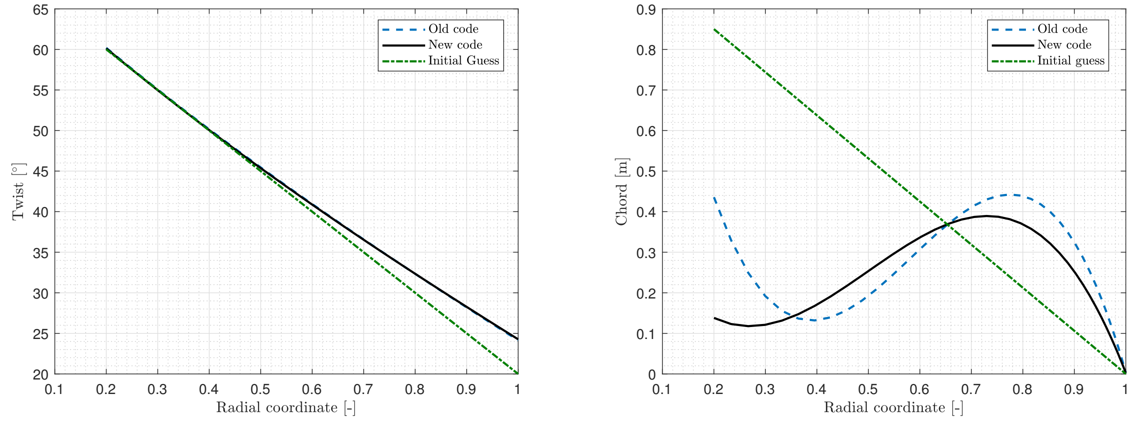
(a) *Efficiency.*

(b) *Thrust.*



(c) *Torque.*

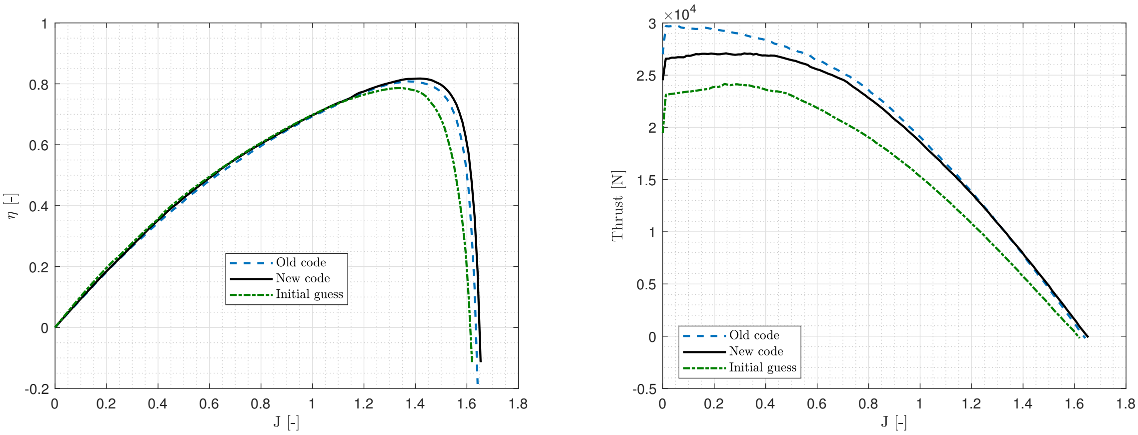
Figure 3.8: NACA16-012 airfoil performance optimization results - linear chord initial guess: old code vs new code.



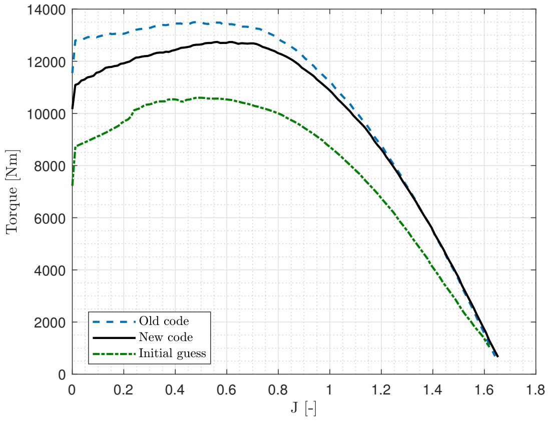
(a) *Twist distribution.*

(b) *Chord distribution.*

Figure 3.9: RAF-6 airfoil geometrical optimization results - linear chord initial guess: old code vs new code.



(a) *Efficiency.* (b) *Thrust.*



(c) *Torque.*

Figure 3.10: RAF-6 airfoil performance optimization results - linear chord initial guess: old code vs new code.

As anticipated at the start of the Section, this time differences between the two codes are less pronounced. In particular the two twist distributions (Figure 3.9(a)) are pretty much alike and they differ very little from the initial guess, with the exception of the external part of the blade. Chord distribution on the other hand changes a lot from the initial guess for both programs and their trend is similar excluding the inner sections of the blade (i.e. close to the hub). Here the enhanced code gives a much more feasible and regular distribution, as noticeable from Figure 3.9(b). Again the smaller overall value of the chord is due to the aerodynamic coefficients underestimation by the old code, as discussed in Section 3.1. Regarding the efficiency, it assumes again very similar values in both cases (although according to the new one, the blade performs slightly better). This time the initial guess η is smaller in the majority of J envelope.

In addition to this, it is worth noticing how, despite a much bigger blade area (Figure 3.9(b)), the un-optimized geometry provides less thrust and torque and, indeed, it would not be able to satisfy none of the thrust constraints (Figure 3.10(b)). This aspect highlights how important a proper chord distribution design is for an aeronautical propeller.

Finally, for what concerns the resulting D and Ω, they are exactly the same for both codes and close to the initial guess. The results are listed in Table 3.3.

Table 3.3: First optimization-RAF-6: D and Ω results.

Version	D	Ω
Old	2.5584 m	1999 rpm
Enhanced	2.5584 m	1999 rpm

The same procedure, starting from the same initial guess of Table 3.1, has been carried out for Clark-Y propeller. Similar considerations to the RAF-6 case apply. Results are displayed in Appendix C, Figure C.4.

3.2.2 Constant chord - linear twist initial guesses

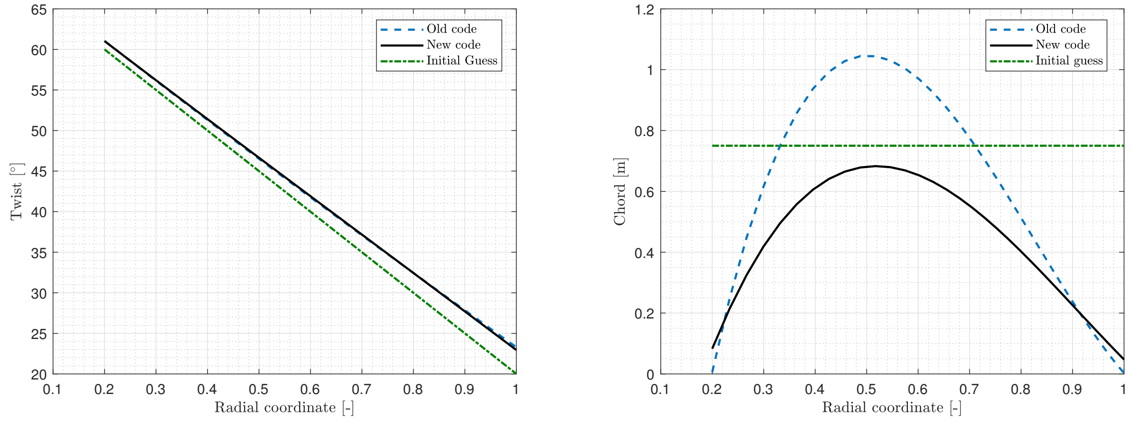
Another group of optimization loops was performed starting with initial guess values reported in Table 3.4.

Table 3.4: Second optimization: initial guess.

Variable	**Value**
D	2.5 m
Ω	2000 rpm
Chord distribution	Constant-0.75 m
Twist distribution	Linear

Exact values of geometrical distributions are displayed along with the results of the optimization in the following (Figures 3.11(a) and 3.11(b)).

In Figure 3.11 the results for NACA16-012 propeller are displayed. Performance outputs are shown in Figure 3.12.



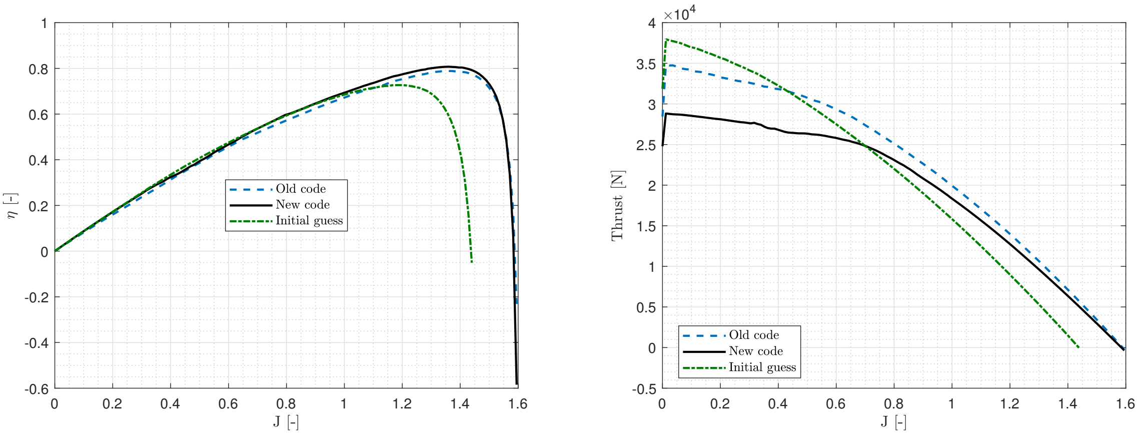
(a) *Twist distribution.* (b) *Chord distribution.*

Figure 3.11: NACA16-012 airfoil geometrical optimization results - constant chord initial guess: old code vs new code.

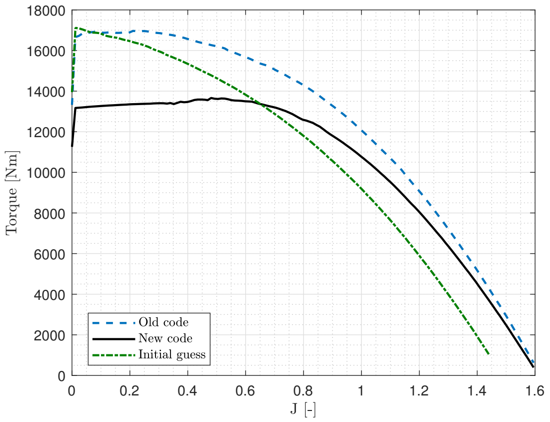
As in the case of linear chord initial guess (Section 3.2.1), chord distribution is particularly reduced when the optimization is performed through the new code (3.11(b)). The probable reason has already been discussed and relies in the underestimation of aerodynamic coefficients from the old version of the code, particularly at low J (see Figure 3.5).

For what concerns the other performance parameters coming from the optimized geometries, similar consideration to what stated in the previous Section apply. The major difference is given by the initial guess geometry which presents this time similar thrust and torque values. Nonetheless, the maximum J reached by the un-optimized propeller is still minor compared to the optimized cases. The same can be said for the apex in the efficiency curve. This means that the initial guess propeller is a "slower" one compared to the optimized geometries, which have been obtained with the intention of maximizing η in cruise conditions (hence, sustained speed).

In Figure 3.13 the geometrical results for RAF-6 propeller are displayed. Performance outputs are shown in Figure 3.14.

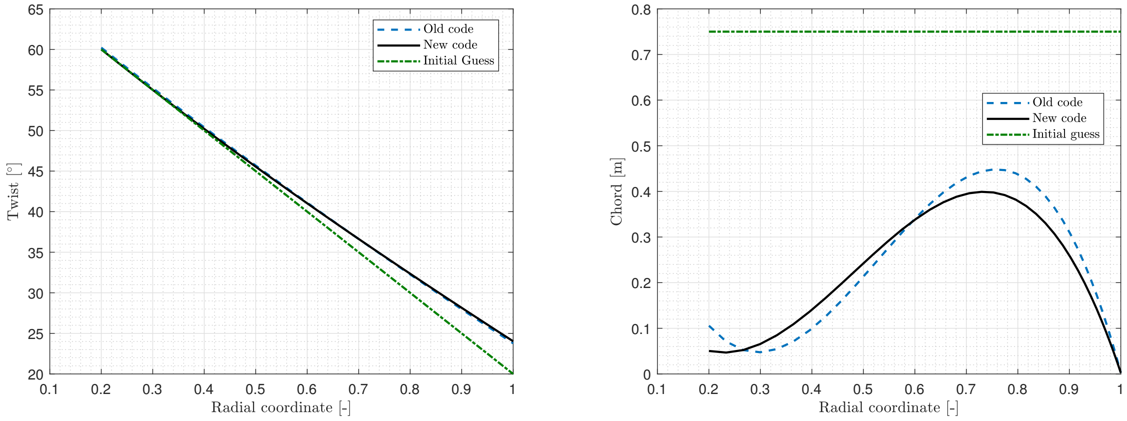


(a) *Efficiency.* (b) *Thrust.*



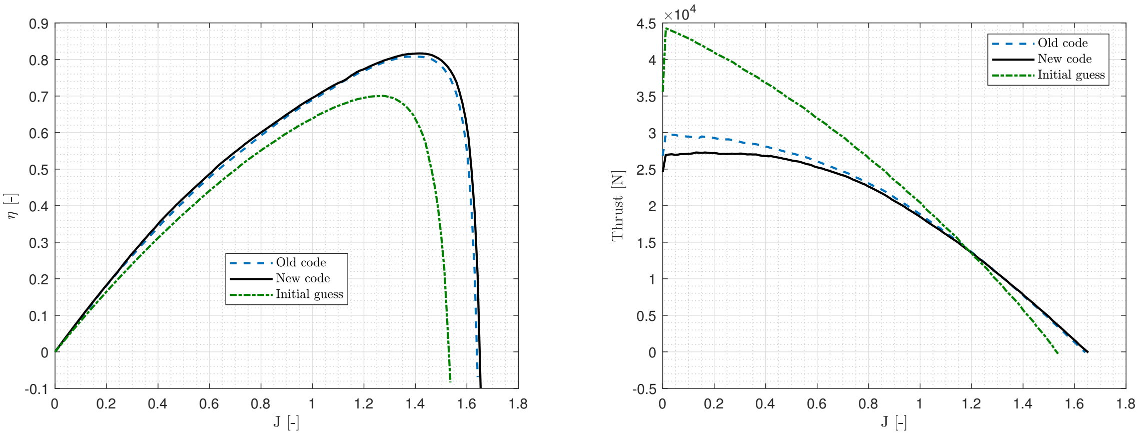
(c) *Torque.*

Figure 3.12: NACA16-012 airfoil performance optimization results - constant chord initial guess: old code vs new code.



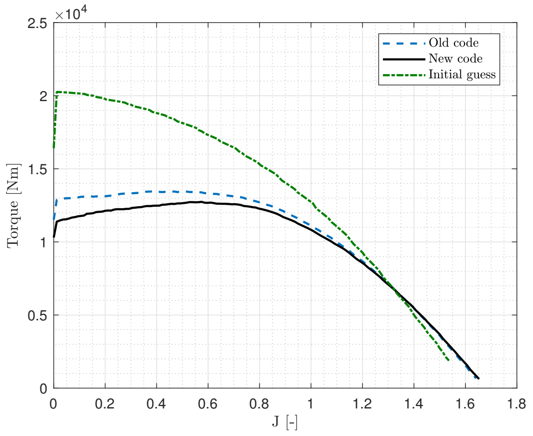
(a) *Twist distribution.* (b) *Chord distribution.*

Figure 3.13: RAF-6 airfoil geometrical optimization results - constant chord initial guess: old code vs new code.



(a) *Efficiency.*

(b) *Thrust.*



(c) *Torque.*

Figure 3.14: RAF-6 airfoil performance optimization results - constant chord initial guess: old code vs new code.

Again, optimized results are very similar to what obtained in the previous Section 3.2.1 with different initial guess hence what stated before still stands. Nonetheless, it is worth mentioning the chord distribution (Figure 3.13(b)), which is almost identical between the two codes along all the span, including the internal part. Conversely, in the previous Section 3.2.1 it was shown how in this part of the blade the two versions of the program led to different chord values (see Figure 3.9(b)).

Same procedure has been performed for Clark-Y propeller, whose results are, again for the sake of brevity, reported in Appendix C.

It is interesting to notice that, even starting with quite different initial guesses, the geometrical outputs (particularly those coming from the enhanced code) are pretty similar, meaning that the condition found is actually the optimal one to maximize efficiency in cruise.

3.2.3 Optimized performance for the different airfoils

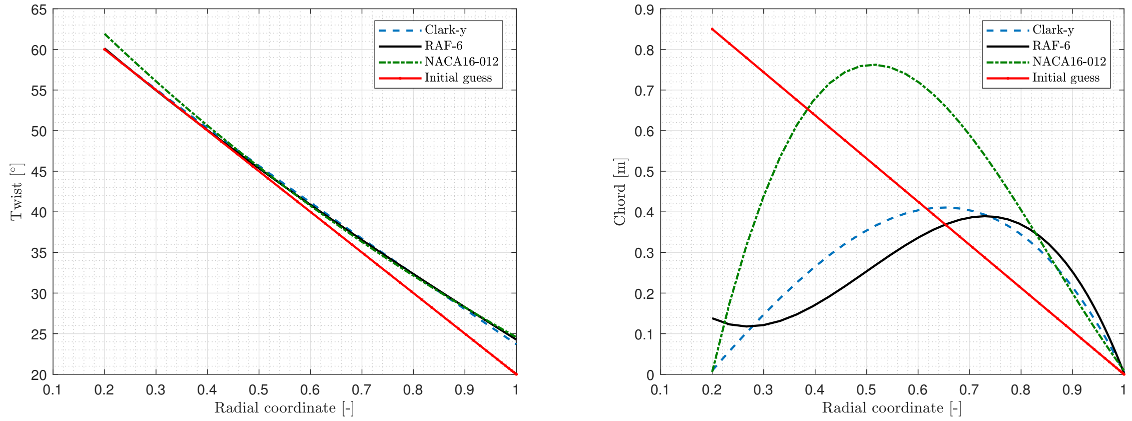
After having performed the optimization loops for every airfoil in the database, it is worth comparing the optimized performance given by the different propellers (each one obtained adopting one airfoil or another) to investigate what are the differences.

In Figure 3.15 the performance and geometrical outputs of the - linear-twist and linear-chord initial guess - optimization relative to NACA16-012, RAF-6 and Clark-Y propellers are displayed.

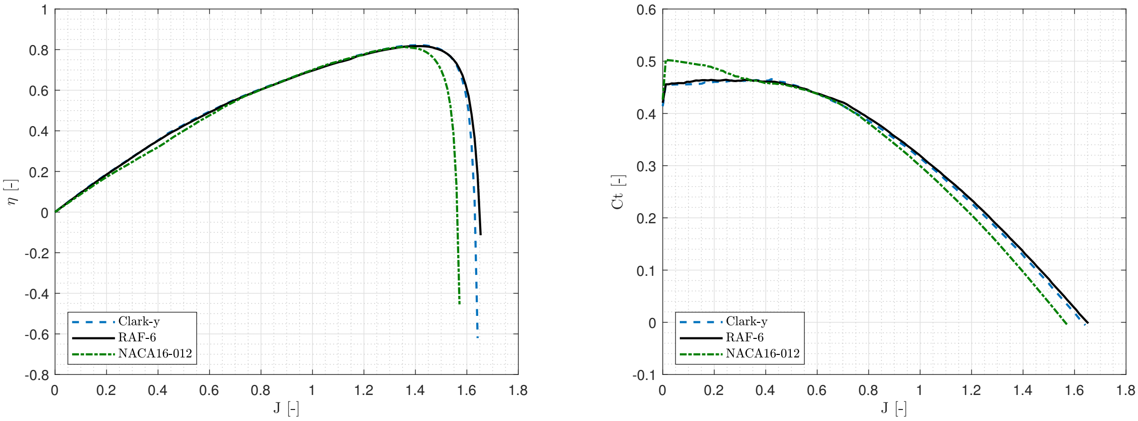
As noticeable, Clark-Y and RAF-6 propellers are almost identical in terms of performance and very similar in terms of chord and twist distributions. NACA16-012 propeller on the other hand, needs a considerably higher chord to meet all the requirements of thrust and also performance speaking behaves slightly differently from the other two airfoils. In particular both power and thrust coefficients are sensibly higher for small J regions and smaller for higher J regions. This means that mounting a propeller that adopts NACA16-012 sections would result in a minor maximum horizontal speed reachable by the aircraft. As a matter of fact null C_t is reached at around $J = 1.55$ while for a propeller mounting either RAF-6 or Clark-Y airfoils the same happens at $J = 1.65$.

For what concerns the efficiency, with the exception of the central part of the J envelope, RAF-6 and Clark-Y propellers show a better behavior and reach a maximum value that is respectively 0.5% and 1% higher than that of NACA16-012. At take-off advance ratio this difference arrives at almost 3%.

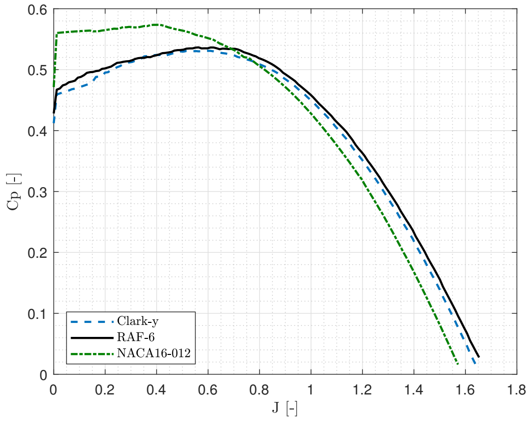
In Figure 3.16 the performance and geometrical outputs of the - linear-twist and constant-chord initial guess - optimization relative to NACA16-012, RAF-6 and Clark-Y propellers are displayed.



(a) *Twist distribution.* (b) *Chord distribution.*

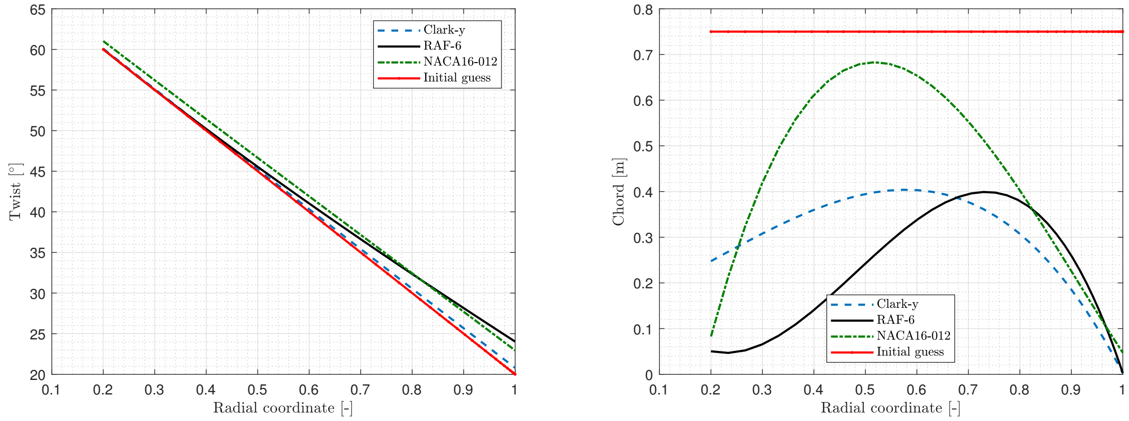


(c) *Efficiency.* (d) *Thrust coefficient.*

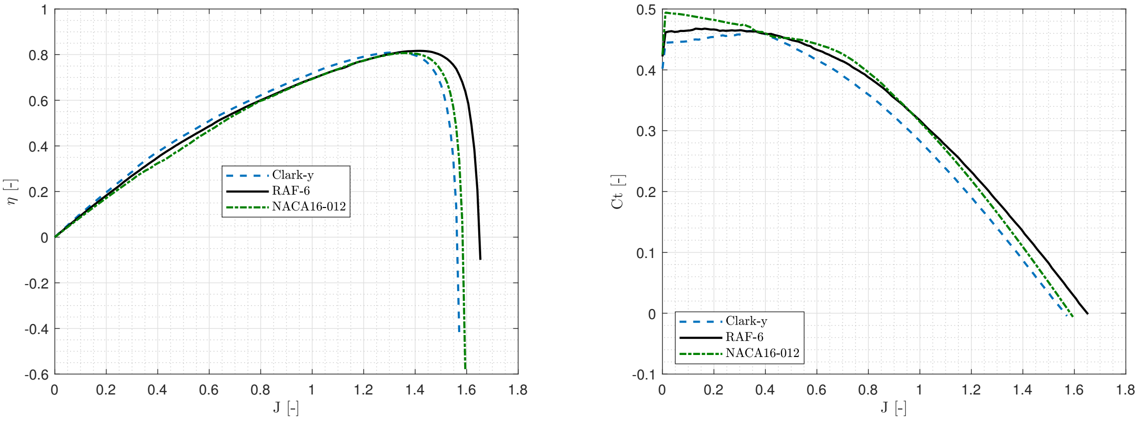


(e) *Power coefficient.*

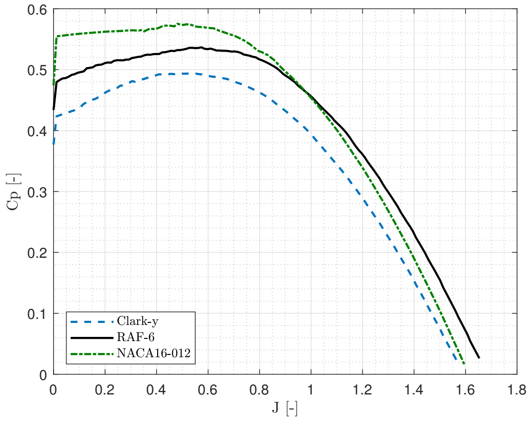
Figure 3.15: Optimization results - linear chord and linear twist initial guess: different airfoil comparison.



(a) *Twist distribution.* (b) *Chord distribution.*



(c) *Efficiency.* (d) *Thrust coefficient.*



(e) *Power coefficient.*

Figure 3.16: Optimization results - constant chord and linear twist initial guess: different airfoil comparison.

This time optimized propellers coming from the adoption of the various airfoils differentiate more in terms of chord distribution with respect to the previous case. Again, the biggest blade area is obtained adopting NACA16-012 airfoil and after that Clark-Y and RAF-6, in descending order. Thrust coefficient is similar among all the three propellers, with the one adopting the RAF-6 airfoil reaching a null value at a J of 1.65 versus 1.55÷1.6 of the Clark-Y and NACA16-012, respectively. This means that this time the fastest propeller is still that of RAF-6, but that of Clark-Y is the slowest. As before, at low J values NACA16-012 propeller presents the highest power coefficient followed by RAF-6 one. From $J > 1$ the situation turns around. In turn, Clark-Y propeller presents a lower C_p throughout all the envelope

For what concerns the efficiency, until $J = 1.3$ Clark-Y propeller has a slight edge over the other two (around 0.5÷1%) but from that point on RAF-6 behaves better and reaches the highest maximum value (81.6%).

In Table 3.5 find the summary of the most interesting performance indicators regarding all the optimized propellers that have been discussed above.

Table 3.5: Optimized propellers performance.

Optimized propeller	**Initial guess geometry distribution**	**Optimized max η**	**Optimized max C_t**	**Optimized max C_p**
RAF-6	Constant chord & linear twist	0.816	0.468	0.537
	Linear chord & linear twist	0.817	0.464	0.536
Clark-Y	Constant chord & linear twist	0.810	0.464	0.494
	Linear chord & linear twist	0.822	0.466	0.531
NACA16-012	Constant chord & linear twist	0.807	0.494	0.575
	Linear chord & linear twist	0.812	0.503	0.574

3.3 Variable pitch case

Most engines produce their maximum power in a narrow rotational speed band. This is especially true for a turboprop engine, such as the Pratt & Whitney Canada PT6A-67D mounted on the Beech 1900.

Supposing that the optimal rounds per minute (RPM) of this engine correspond to the value obtained after the optimization performed in the previous Section ($\approx$ 2000 rpm) and that the adopted airfoil is the RAF-6, for the optimized twist and chord distributions (Figures 3.13(a), 3.13(b)) the efficiency outcome (Figure 3.14(a)) shows that satisfactory (i.e. higher than 0.6) values of η are reached only for a small interval of J ($0.85 \div 1.55$).

To overcome such problem a Variable Pitch Propeller (VPP) can be introduced. This kind of propeller can be efficient for the full range of rotational speeds and load conditions, since its pitch will be varied to absorb the maximum power that the engine is capable of producing. In the case of a constant speed propeller, the blade pitch is designed to automatically change to allow to maintain a constant RPM, irrespective of the amount of engine torque being produced or the airspeed or altitude at which the aircraft is flying. This is accomplished by means of a Constant Speed Unit (CSU), or governor, integrated into the propeller design.

Most constant speed units work on the principle of centrifugal force and incorporate a speeder spring and a set of fly weights. The speeder spring is tensioned to balance the fly weights at a specific propeller RPM and, in some installations, is pilot adjustable allowing more than one target RPM to be selected. Should the propeller exceed the pre-selected RPM, the fly weights will be forced outward whereas a propeller under-speed would cause the fly weights to swing inward. In both cases, this changes the tension on the speeder spring. In early constant speed propellers, the movement of the weights would drive a mechanism to mechanically change the pitch of the propeller - increasing it in response to an over-speed and decreasing it in response to an under-speed. In newer propeller models, the blade pitch change is accomplished by porting oil, under pressure, through a pilot valve in response to an under or over speed condition. The oil, which might either be from the engine or integral to the propeller itself, causes the propeller blade angle to change as required to maintain the selected RPM. An example of a centrifugal constant speed propeller hub is given in Figure 3.17.

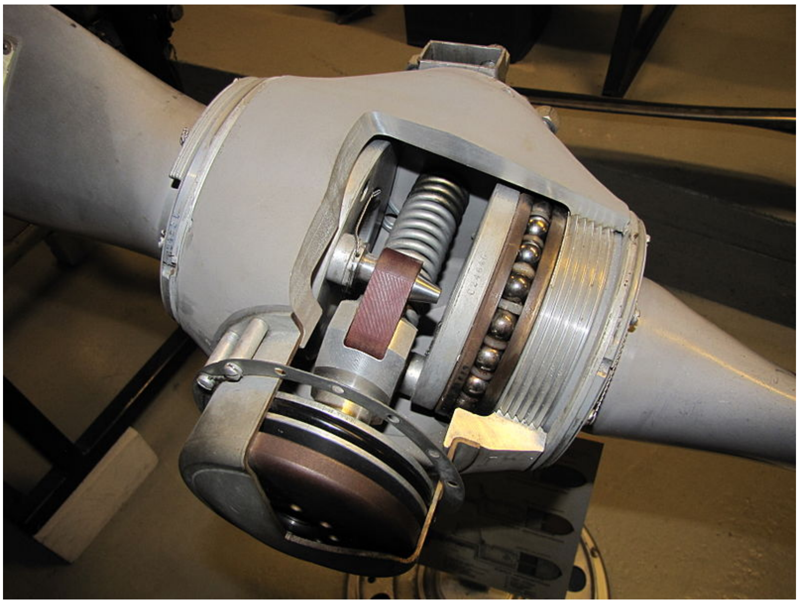
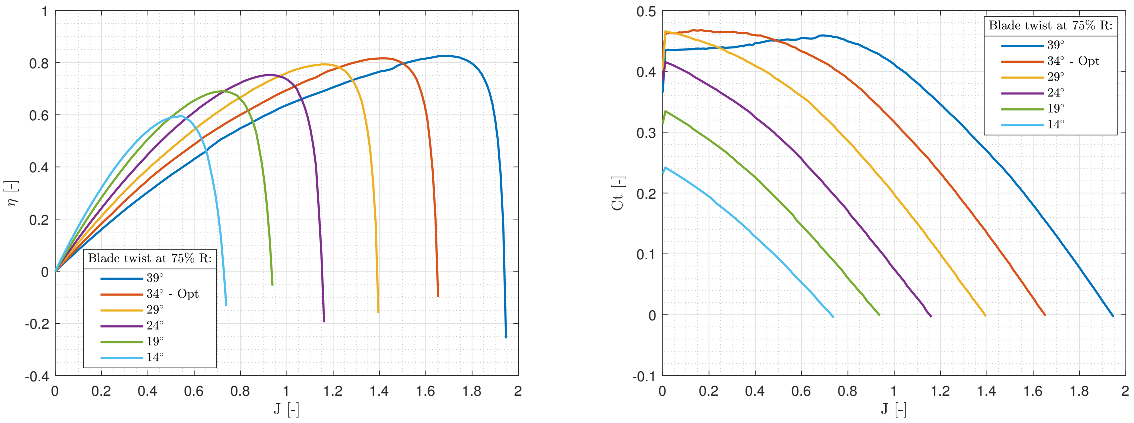


Figure 3.17: Constant speed propeller hub [30].

In this fashion, it is possible to operate the propeller and engine at the most efficient RPM and torque for the phase of flight. RPM is controlled automatically by varying the pitch of the propeller blades – that is, the angle of the blades with relation to the plane of rotation. For any given power setting, as the blade angle is reduced, the torque required to spin the propeller is reduced and RPM of the engine will tend to increase. Conversely, if the blade angle is increased, the torque requirement to maintain a constant RPM increases. If the power is not changed, then the engine and the propeller will tend to slow down. In essence, this relationship allows both the propeller and the engine to be set to maintain their respective optimum RPM. Adding fuel (by moving the power lever forward) increases the power output of the engine and would tend to increase the engine RPM - however, to maintain the selected propeller RPM, the blade angle increases to absorb the additional torque that the engine is now producing allowing the engine RPM to remain at its original value. The converse is true for a reduction in power lever position where, as less torque is available, the propeller RPM decreases allowing the engine RPM to remain more or less constant.

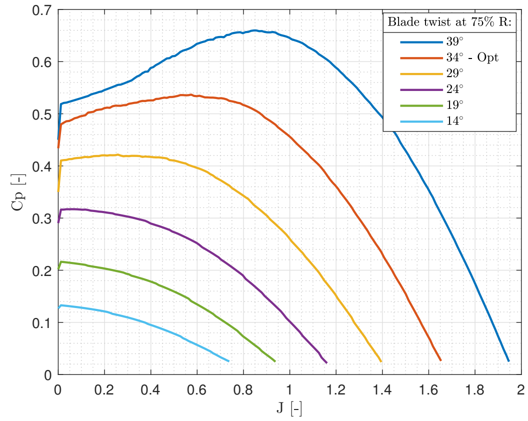
Such pitch control has been simulated adding or subtracting a constant value to the twist distribution - just as the CSU would do - and computing the performance for each resulting distribution. Now, the optimization outcome gave a non-zero thrust until $J \approx 1.6$ for both initial guesses and airfoils, which corresponds, for $\Omega = 2000\,\text{rpm}$, to roughly 140 m/s. Maximum Beech 1900 airspeed is around 135 m/s, so the twist has been varied adding a constant value ranging from 5° to $-25°$ with a step of 5°. In fact, for what stated above, to higher twist distributions correspond higher reachable maximum speeds so there was no need to investigate performance for much increased values of twist.

In Figure 3.18 it is shown what would be like to adopt the optimized RAF-6 propeller coming from the optimization case of Section 3.2.2, that is constant chord and linear twist initial guess.



(a) *Efficiency.*

(b) *Thrust coefficient.*



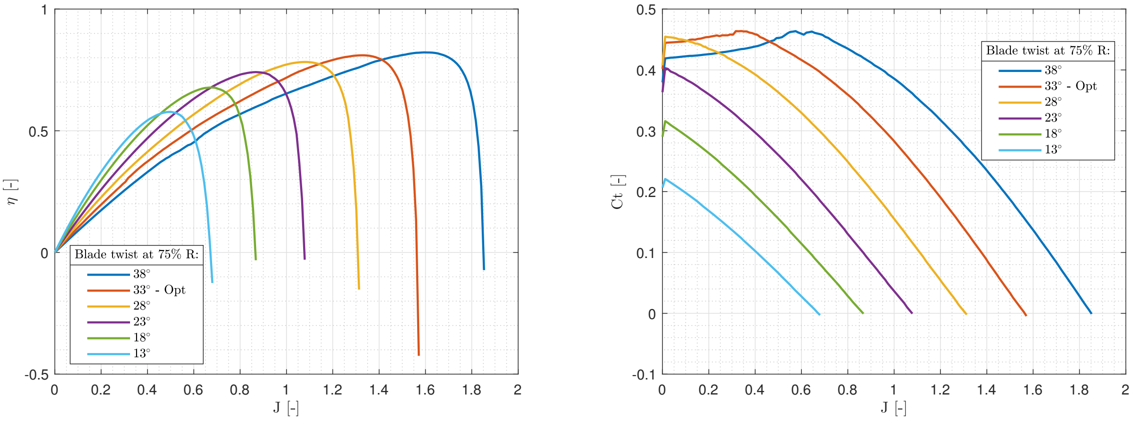
(c) *Power coefficient.*

Figure 3.18: Optimized propeller - constant chord and linear twist initial guess - variable pitch performance. RAF-6 airfoil.

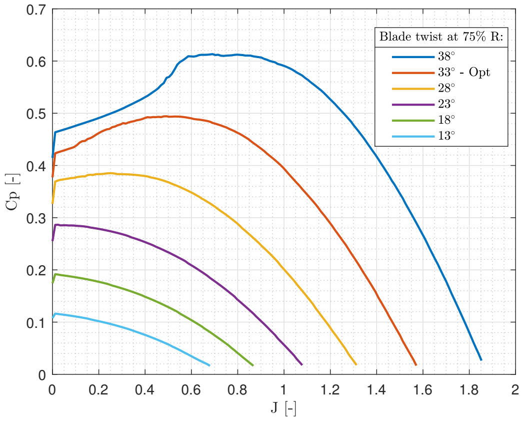
As noticeable, decreasing propeller's pitch allows to obtain a fair value of η even at low J regions. Compared to the unmodified optimized case (orange curve in Figure 3.18(a)), for which the efficiency is higher than 0.6 only for $0.85 < J < 1.55$, CSU expands this interval considerably (roughly $0.5 < J < 1.9$). This means, in practice, that if the optimized propellers were to be used in a commuter aircraft similar to Beech 1900 equipped with a variable pitch mechanism it would perform well also far from cruise condition (which was the point where the optimization was performed).

Similar considerations can be made for a propeller mounting the same airfoil with a geometry distribution coming from initial guess of Section 3.2.1 (linear chord and linear twist). Results are shown in Appendix C.

In Figure 3.19 it is shown would be like to adopt the optimized Clark-Y propeller coming from the optimization case of Section 3.2.2, that is constant chord and linear twist initial guess.



(a) *Efficiency.* (b) *Thrust coefficient.*



(c) *Power coefficient.*

Figure 3.19: Optimized propeller - constant chord and linear twist initial guess - variable pitch performance. Clark-Y airfoil.

Again, decreasing propeller's pitch allows to obtain a fair value of η even at low J regions. Compared to the unmodified optimized case (orange curve in Figure 3.19(a)), for which the efficiency is higher than 0.6 only for $0.75 < J < 1.55$, CSU expands this interval considerably (roughly $0.45 < J < 1.8$). With respect to RAF-6 blade, in this case the optimized-for-cruise twist distribution works better, that is the interval for which $\eta > 0.6$ is wider. Nonetheless, it is evident how the situation improves with the adoption of a CSU.

Results for Clark-Y propeller with a geometry distribution coming from initial guess of Section 3.2.1 (linear chord and linear twist) and the for NACA16-012 propeller coming from both optimization cases of Sections 3.2.1 and 3.2.2, for which almost identical considerations apply, are reported in Appendix C.

In Table 3.6 is shown the improvement introduced by the CSU in terms of J interval where efficiency exceeds 60%.

Table 3.6: Advance ratio interval where $\eta > 0.6$.

Airfoil	**Initial guess geometry distribution**	**J interval with CSU**	**J interval no CSU**
RAF-6	Constant chord & linear twist	$0.5 \div 1.9$	$0.85 \div 1.55$
	Linear chord & linear twist	$0.5 \div 1.9$	$0.8 \div 1.6$
Clark-Y	Constant chord & linear twist	$0.5 \div 1.8$	$0.75 \div 1.5$
	Linear chord & linear twist	$0.45 \div 1.9$	$0.8 \div 1.6$
NACA16-012	Constant chord & linear twist	$0.5 \div 1.85$	$0.8 \div 1.55$
	Linear chord & linear twist	$0.5 \div 1.85$	$0.8 \div 1.55$

In each case the advance ratio interval almost doubles, showing the great effectiveness of having a variable pitch propeller.

3.4 Variable airfoil propeller

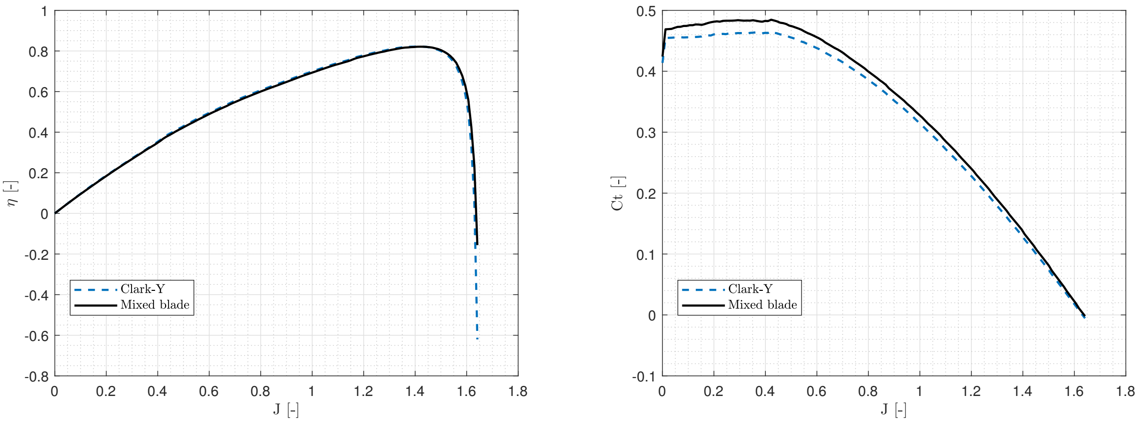
A preliminary attempt of introducing in the program the possibility of having different airfoil sections along the span has been made. In this fashion, if the designer wants to adopt more than one airfoil for his propeller, the tool is able to give an estimation of its effect, as long as the airfoils used are among the ones of the database.

In practical terms, the subroutine `SinglePerformance` has been changed in order to accept two different databases (each one corresponding to the two airfoils to be used along the span) and to draw on the correct one in accordance to the division of the blade sections decided by the user. Indeed, the latter has to choose from the main at which percentage of the span wants to change airfoil and the program behaves consequently, treating the first part of the blade as composed by one airfoil and the last one as composed by the other airfoil.

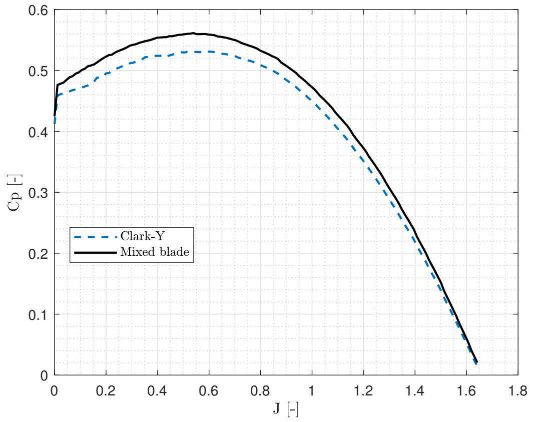
RAF-6 and Clark-Y propellers resulting from the optimizations loops performed in the previous subsections have been used as “geometrical base” on which the two airfoils were mixed in various ways. In Figure 3.20 is represented the performance output of a propeller with geometry of Figures C.4(a) and C.4(b) with the first half (without considering the hub) of the blade composed by Clark-Y sections, the remaining part composed by RAF-6 sections.

Regarding thrust and power coefficients combining the airfoils exhibits some effects for low advance ratios, where the mixed blade provides higher values of both C_p and C_t. The efficiency in turn, is pretty much unchanged.

As already said, this modification of the program was only a preliminary study and, as such, suffers from some limitations. Firstly, the user has to manually change the percentage of the span at which the airfoil has to change. Secondly, only two different airfoils are admitted along the span. In addition to this, taking into account the actual realization of the propeller, having an abrupt change of airfoil would introduce a discontinuity in the



(a) *Efficiency.* (b) *Thrust coefficient.*



(c) *Power coefficient.*

Figure 3.20: Performance output obtained with a mixed blade - half Clark-Y and half RAF-6 airfoil. Geometry of the blade is that of the optimized Clark-Y propeller.

blade shape at the point where the section changes. Of course, this solution could not be accepted in practice, where some sort of blending between the two airfoil in order to guarantee geometrical compatibility would be necessary. Nonetheless, it is still worth to have shown the results to have an idea of what would actually be to use more than one airfoil for a single blade. In addition to this, it opens the way to further investigations in this sense in eventual future developments of the program.

Chapter 4

Conclusions

As a natural continuation of De Zubeldia's work [2], this study aimed to develop a formulation that would help in the early design phases of a propeller. With respect to the previous, the present formulation has been successfully improved and its possibilities expanded. Description of this process is given thoroughly in Chapter 2. Firstly, the aerodynamic database, which initially included only NACA0012 airfoil data, has been extended to three of the most used airfoil in propeller design, that is NACA16-012, RAF-6 and Clark-Y. This has been possible thanks to the successful XFOIL reliability study, where its predictions were compared to those of OpenFoam. In the CFD simulations the turbulence model adopted was the κ-ω SST developed by Menter [23] with the use of wall functions. To validate the two-dimensional CFD case, low Mach number simulations have been performed first and confronted with XFOIL and consolidated results coming from Abbot [3] and Ladson experiments [20]. From here, Mach number has been increased to assess at which point the outcomes started to diverge. Results in terms of C_l have been used in this phase. In Section 2.2.2, it is shown that major differences appeared from a Mach number of 0.65. In light of this, the limit used in the expansion of the aerodynamic database through the use of XFOIL was 0.6. From that point on, experimental data gathered by Harris [19] for the NACA0012 section have been used, as it was in the old version. In first approximation, this procedure has been considered satisfactory, since only in small (external) portions of the blade and for few values of V_∞ Mach number was over 0.6. In Section 2.3 the second enhancement applied is discussed, that is the refinement of aerodynamic coefficients computation at low advance ratios, where the code derived from De Zubeldia's methodology presented, for particular geometries, irregularities and non physical behavior in the curves. It is a known fact that, due to rotation of the blade, lift coefficients on inboard sections may exceed the 2D maximum lift coefficient. Therefore, there was a need for correction of the airfoil characteristics to include the 3D flow effects. The differences between the calculations applying the classical[1] BEMT and actual measurements increase as larger portions of the blades experience high cross-sectional angles of attack, beyond the stall limit. In particular, it turns out that the measured propeller's thrust is higher (sometimes much higher) than the thrust predicted by calculations where the two-dimensional stall characteristics of a non-rotating airfoil are used [4]. The effect was attributed in various studies to the presence of a Coriolis force, having the same effect as a favorable pressure gradient. In addition, the centrifugal force causes an outward displacement of fluid particles, through which the boundary layer becomes thinner compared to a non-rotating boundary layer. Two different approaches were inspected in this analysis, that of Du and Selig [10] and Chaviaropoulos and Hansen [7]. Both of them have been implemented in a software to assess their applicability. The theory and the rationale behind each model is reported in Sections 2.3.1, 2.3.2 and Appendix B. In Section 2.3.3 it is shown how Chaviaropoulos and Hansen method suited better the case of study in terms of CPU efficiency and results veracity, which is why it was finally chosen for the present formulation.

---

[1] By classical is intended the one where the 2D aerodynamic coefficients are used.

After having defined the theoretical aspects of the model in Chapter 2, the new formulation has been implemented in MATLAB® environment, presenting two principal working modalities: performance computation for a given geometry and geometry optimization starting from a generic initial guess and cost function to minimize. In Chapter 3 the outcomes of the numerical studies carried out with the software are presented. In the first part a parametric study of the a, h and n coefficients effect on the correction law (eq. 2.31) is carried out for all the database airfoils. Secondly, optimization loops with the enhanced software are performed and compared to the old one for the same initial guess in terms of blade geometry (twist and chord distributions), diameter (D) and angular velocity (Ω). In addition to this, the presence of a variable pitch mechanism has been simulated changing by a (positive or negative) constant value the optimized twist distribution and computing the performance output for each resulting case. Finally, a preliminary attempt to introduce in the program the possibility of having more than one airfoil section along the span has been performed. In Section 3.1 the parametric study reveals that, firstly, the new code depicts thrust and torque behavior for low J values much better than the old one, which underestimated them considerably. On the other hand, for high J regions, where the AoA along the blade are lower and the flow is more likely to be attached, the two codes behave similarly, in accordance to theory. Secondly, it is evident that a, h and n coefficients variation does not change considerably the output given by the new code. In view of this aspect, corrective coefficients adopted in new version of the code will be that of Table 2.4. Thereafter, in Section 3.2 it is analyzed what are the differences with respect to the old code when performing optimization loops starting from identical initial guesses. The first optimization loop was performed starting with initial guess values reported in Table 3.1; the second with initial guess values reported in Table 3.4. This procedure has been carried out for all the database airfoil. It turned out that the old code, given its limitations in performance computation at low J where it underestimates forces acting on the blade, obtained bigger optimized chord values along the span in order to satisfy thrust constraints. In other words, not considering three-dimensional effects leads, as shown in Section 3.1, to an underestimation of aerodynamic forces acting on the blade which, inserted in a optimization loop, causes a over-sizing of the chord along the span in order to satisfy the constraints, particularly that of take-off thrust. In this fashion, it has been demonstrated that adopting the new code a much more feasible and regular chord distribution is obtained as well as a more precise output, without resulting in useless over-sizing of the blade. Comparing the optimized propellers, the RAF-6 presented a more “manufacturable” chord distribution and slightly better efficiency. The comparison is treated in depth in Section 3.2.3. Furthermore, in Section 3.3 it is displayed the variable pitch mechanism simulation. It is a known fact that most engines produce their maximum power in a narrow rotational speed band, especially turboprop ones, such as the Pratt & Whitney Canada PT6A-67D mounted on the Beech 1900. To overcome such limitation a Variable Pitch Propeller (VPP) is usually introduced to make propeller pitch vary to absorb the maximum power that the engine is capable of producing. In order to depict this mechanism, the resulting optimized twist from Sections 3.2.1 and 3.2.2 has been varied adding a constant value ranging from $+5°$ to $-25°$ with a step of 5°. Compared to the unmodified optimized case, for which the efficiency is higher than 60% only for a rather narrow J interval, Constant Speed Unit (CSU) expands it considerably. Table 3.6 shows that the J interval where efficiency exceeds 60% is practically doubled for all the optimized propellers adopting a CSU. This means, in practice, that if the optimized propellers were to be used in a commuter aircraft similar to Beech 1900 equipped with a variable pitch mechanism it would perform well also far from cruise condition (which was the point where the optimization was performed).

Finally, a preliminary attempt of introducing in the program the possibility of having different airfoil sections along the span has been made. In Figure 3.20 is shown that combining two airfoils can achieve higher values of both C_p and C_t at low J regions. Again, this constituted only an initial study and had more a demonstrative purpose. As a matter of fact, it suffered from some limitations that can be found in Section 3.4 but gave an idea of what would actually be to use more than one airfoil for a single blade.

In light of the analysis conducted in Chapter 3, RAF-6 propeller is considered the best in both “manufacturability” and performance, including when CSU was simulated. Nonetheless, also propellers adopting Clark-Y and NACA16-012 airfoils performed well and could constitute a more than satisfactory design. As a matter of fact, since the differences are not that pronounced (between RAF-6 and Clark-Y in particular), the final choice could rely also on other aspects than performance indicators.

4.1 Further studies

On the whole, this analysis was a success. All the objectives that had been posed at the start of the study were met and constituted a consistent improvement with respect to De Zubeldia’s analysis. In particular, the formulation is now more complete and the resulting code is more versatile, precise and, consequently, leads to more physically convincing results.

Nonetheless, there are some aspects that could be analyzed in order to further develop the present work and make the formulation one more step closer to reality.

- Aerodynamic database could be further expanded to other airfoils in order to cover a bigger family of sections. From there, it could be useful to perform the reliability study for each airfoil, since the Mach number limit up to which XFOIL is precise might vary from one to another.

- In Section 2.2.2 it has been pointed out as, for $Ma > 0.6$, data coming from experiments on NACA0012 airfoil have been used. An additional improvement could be adopting aerodynamic coefficients of the airfoil that is actually being used also in this Ma range, after having obtained them via experiments or CFD.

- In Section 3.1, it has been said that the corrective constants a, h and n used in the 3D correction (eq. 2.31) are that of Table 2.4, namely those used by Chaviaropoulos and Hansen in [7], obtained through a comparison between calculated and measured power curves on a Bonus 300 Combi stall regulated wind turbine, whose blade is based on NACA63-4XX and NACA63-2XX airfoils. It could be interesting to perform a similar study for each database airfoil and fine tune the coefficients consequently. Nonetheless, in the same Section it has been shown that results change slightly even for considerable variations of the corrective constants, hence an effort in this sense might be time consuming compared to the effective change in the program output.

- With respect to the preliminary study performed in 3.4, it would be worth to improve the procedure of multiple airfoil implementation along the span. For instance, automatizing the process would open the path of optimizing the disposition of the two (or more) airfoils in order to minimize some cost function. Also, a sort of blending in correspondence of the point of section change would be necessary, for the sake of geometrical compatibility without discontinuities in the blade shape.
- Another possible enhancement to be applied is the development of a structural and acoustic sub-routines. The former in order to structurally verify the final blade shape to the expected loads (e.g. centrifugal axial force, bending moment, etc.), while the latter to compute the Sound Pressure Level (SPL) emitted by a blade of a given shape. In the last case, it could be interesting to perform optimizations where the SPL is also taken into account, in order to obtain a quiet but still efficient propeller.

Appendix A

OpenFOAM detailed meshing procedure

OpenFOAM is an open source set of libraries written in C++, first developed at the Imperial College of London in the 80s. It is intended mainly to termo-fluidodynamic applications but also comprehends structural and financial solvers.

The standard case organization is structured in folders which may contain in turn other folders or files. The case must include at least three folders: 0, constant and system. Figure A.1 shows how this analysis case was organized.

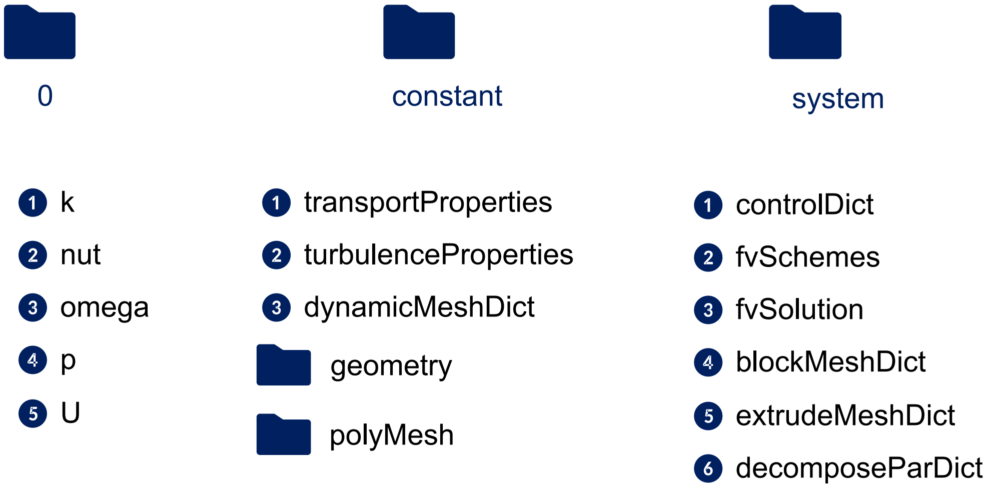

0
constant
system
① k
② nut
③ omega
④ p
⑤ U
① transportProperties
② turbulenceProperties
③ dynamicMeshDict
geometry
polyMesh
① controlDict
② fvSchemes
③ fvSolution
④ blockMeshDict
⑤ extrudeMeshDict
⑥ decomposeParDict

Figure A.1: Case organization.

The whole case folder is then launched directly from the terminal.

As discussed in Section 2.2.2, in the present analysis *`blockMesh`* utility was adopted during the meshing procedure. This is done in practice going inside the system folder and acting on *`blockMeshDict`* file. *`blockMeshDict`* has five mandatory sub-dictionaries:

- Geometry
- Vertices
- Blocks
- Edges

- Boundary

Two other sub-dictionaries were added to further simplify the set up:

- Domain
- Aerofoil

The use of *blockMesh* requires a careful design of the background mesh. First, the domain dimensions were defined; for this study dimensions were those of Table 2.2. Similarly was done for Aerofoil sub-dictionary which was defined by (x,z) coordinates of its lead, upper, trail and lower points. Graphical representation is given in Figure A.2.

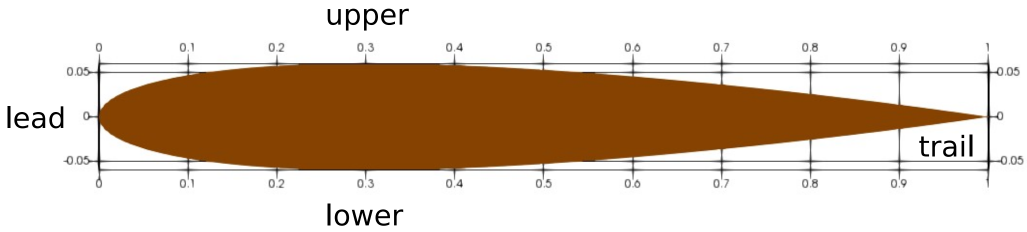


Figure A.2: Aerofoil dimensions.

Thereafter, vertices of the whole domain are defined through the function *project*. In this study, 24 vertices were defined. A graphical representation is given in Figure A.3.

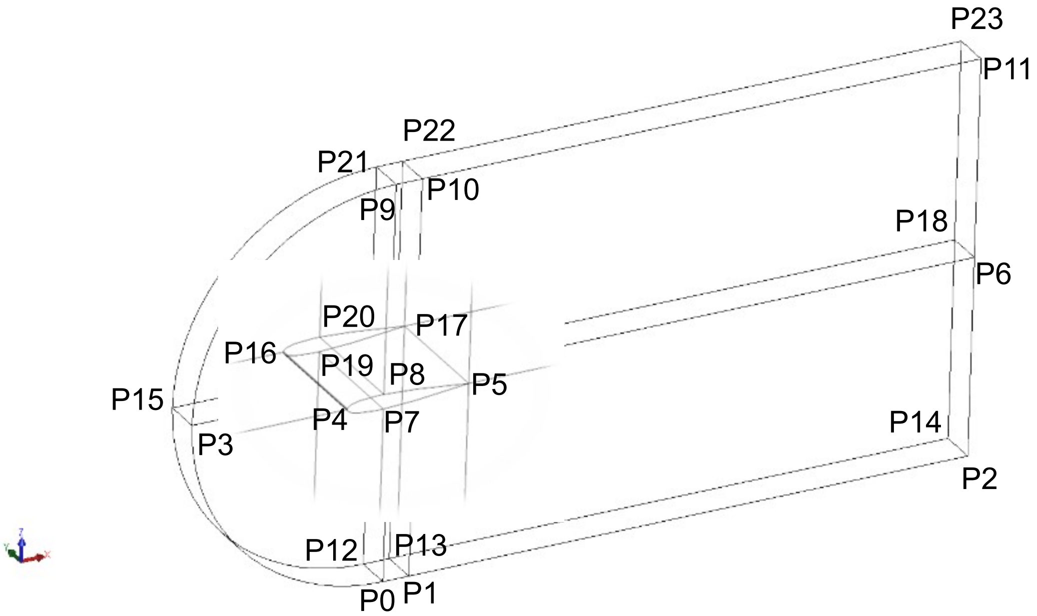
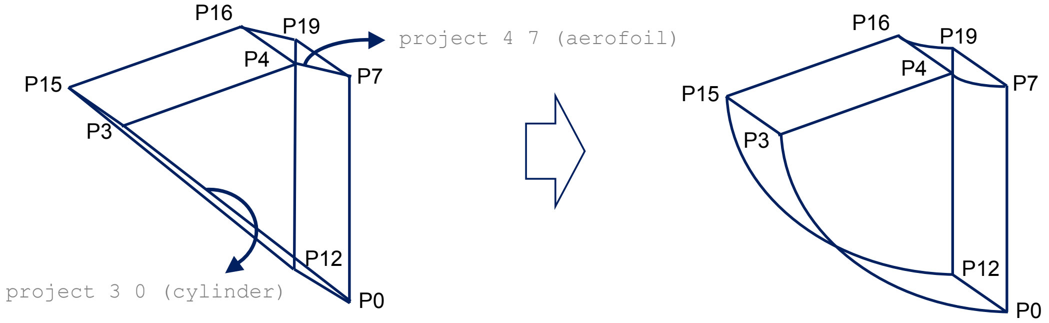


Figure A.3: Domain vertices.

The next step consisted in defining the so-called blocks from the above presented vertices. Each of them was defined, as already discussed in Section 2.2.2, in a local coordinate system (x_1, x_2, x_3) that must be right-handed. The edges of every block are straight by default, to obtain curved edges it was necessary to use again function *project* to project a generic edge defined by two vertices onto airfoil or cylinder. This procedure is better understood looking at Figure A.4.

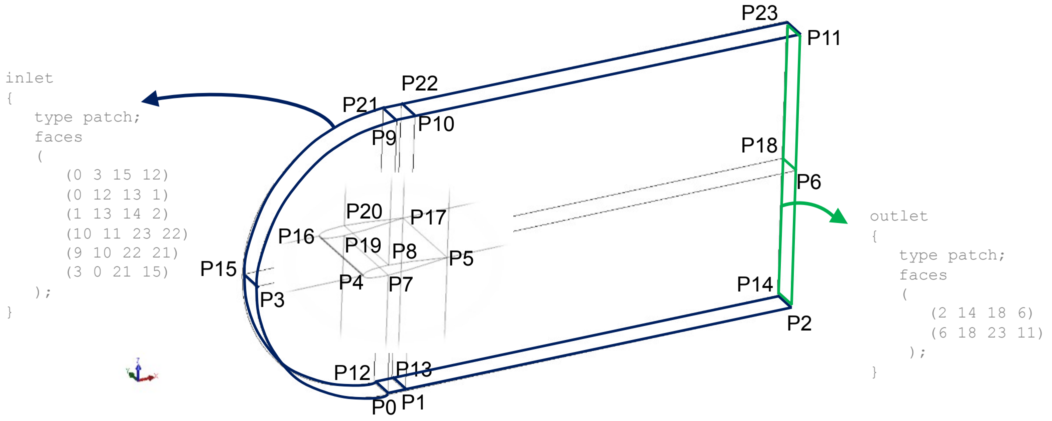

P16
P19
P4
P15
P7
P3
P12
P0
P16
P19
P4
P7
P15
P3
P12
P0

```
project 4 7 (aerofoil)
```

```
project 3 0 (cylinder)
```

Figure A.4: Curved edges.

Boundaries were then defined from vertices. Vertices defining the boundary must be numbered in a clockwise sense looking from inside the block. A graphical example on how the `inlet` and `outlet` were defined is given in Figure A.5.


P23
P11
P22
P21
P9
P10
P18
P6
P20
P17
P16
P19
P8
P5
P15
P4
P7
P3
P14
P2
P12
P13
P0
P1

```
inlet
{
   type patch;
   faces
   (
      (0 3 15 12)
      (0 12 13 1)
      (1 13 14 2)
      (10 11 23 22)
      (9 10 22 21)
      (3 0 21 15)
   );
}
```

```
outlet
{
   type patch;
   faces
   (
      (2 14 18 6)
      (6 18 23 11)
    );
}
```

Figure A.5: Inlet and outlet definition.

Another important aspect to pay attention to within *`blockMeshDicct`* is grading. In particular, when a block is defined each of its edges grading must be specified. In this fashion, it is possible to obtain different cell spacings (more coarse or finer).

Appendix B

Stall-delay principle theory

B.1 Du and Selig model

In [10], for the analysis of the flow over rotating blades, a cylindrical coordinate system attached to the blade is used, with the origin in the center of rotation as shown in Figure B.1.

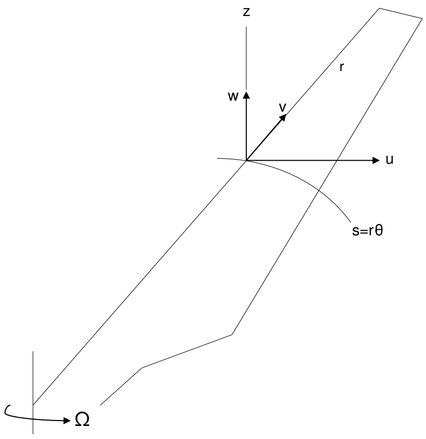


Figure B.1: Coordinate system attached to rotating blade.

The velocity components are u, v and w in directions θ, r and z, respectively. Undisturbed free stream streamlines are circular arcs with constant radius. The 3D incompressible steady boundary layer equations (continuity, r-momentum, and θ-momentum equations) in cylindrical coordinates are given by eqs. B.1.

$$\frac{1}{r}\frac{\partial u}{\partial \theta} + \frac{v}{r} + \frac{\partial v}{\partial r} + \frac{\partial w}{\partial z} = 0 \tag{B.1a}$$

$$\frac{u}{r}\frac{\partial v}{\partial \theta} + v\frac{\partial v}{\partial r} + w\frac{\partial v}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial r} + \frac{1}{\rho}\frac{\partial \tau_2}{\partial z} + \frac{(u-\Omega r)^2}{r} \tag{B.1b}$$

$$\frac{u}{r}\frac{\partial u}{\partial \theta} + v\frac{\partial u}{\partial r} + w\frac{\partial u}{\partial z} = -\frac{1}{\rho r}\frac{\partial p}{\partial \theta} + \frac{1}{\rho}\frac{\partial \tau_1}{\partial z} + 2\Omega v - \frac{uv}{r} \tag{B.1c}$$

The authors assumed that the surface on which the boundary layer develops is in the plane of rotation $z = 0$. Thus, surface curvature in chord wise direction has been neglected with respect to the boundary layer thickness. Ω is the rotational velocity (about the z-axis) with respect to an inertial system, while r is the distance to the center of rotation. τ_1 and τ_2 are the shear stresses in θ- and r- direction respectively, which can be either laminar (viscous) or turbulent (Reynolds) stresses.

The momentum equations contain a number of inertia force type terms. The θ-momentum equation contains the Coriolis-force term $2\Omega v$ and coordinate curvature term uv/r. The r-momentum equation only contains the "total" centrifugal-force term resulting from the total circumpolar velocity $(u - \Omega r)$ in the inertial frame. This term is composed however of what formally are a coordinate curvature term (u^2/r), a Coriolis-force component $(-2u\Omega r)$ and the formal centrifugal force $(\Omega^2 r)$ resulting from the rotating of coordinate system.

The integral boundary-layer equations for a three dimensional boundary layer in an incompressible flow have been developed by many researchers. The technique presented in [10] is an extension of the model developed by Snel [6] for a rotating system. It includes an order of magnitude analysis of the 3D boundary layer equations, which gives rise to a set of equations that is much simpler than the full 3D ones. In the formulation, the nonlinear convective terms were used in their 2D form, while the important 3D effects of centrifugal and Coriolis forces were retained. The simplified integral boundary-layer equations in θ- and r- directions in the rotating coordinate system are expressed as shown in eqs. B.2.

$$\frac{\partial \theta_1}{\partial s} + (2+H)\frac{\theta_1}{u_e}\frac{\partial u_e}{\partial s} = \frac{C_f}{2} + 2\Omega\frac{\delta_2}{u_e} \tag{B.2a}$$

$$\frac{\partial \theta_2}{\partial s} + 2\frac{\theta_2}{u_e}\frac{\partial u_e}{\partial s} = \frac{C_f}{2}\tan\beta_w + \frac{1}{r}\left(\theta_1 + \delta_1 - \delta - \frac{2\Omega r\delta_1}{u_e} + \frac{2\Omega r\delta}{u_e}\right) - \frac{\delta}{u_e}\frac{\partial u_e}{\partial r} \tag{B.2b}$$

Where the integral boundary layer parameters are defined as follows:

$$\delta_1 = \int_0^\delta \left(1 - \frac{u}{u_e}\right) dz \tag{B.3a}$$

$$\delta_2 = -\int_0^\delta \frac{v}{u_e} dz \tag{B.3b}$$

$$\theta_1 = \int_0^\delta \left(1 - \frac{u}{u_e}\right) \frac{u}{u_e} dz \tag{B.3c}$$

$$\theta_2 = -\int_0^\delta \frac{uv}{u_e^2} dz \tag{B.3d}$$

$$H = \frac{\delta_1}{\theta_1} \tag{B.3e}$$

$$C_f = \frac{2\tau_w}{\rho u_e^2} \tag{B.3f}$$

At this point, given the complexity of the problem, the authors assumed a laminar boundary layer distribution on the surface of the blade. The widely used Pohlhausen velocity profiles (eq. B.4) and associated closure relations were then introduced to solve the integral boundary-layer equations for θ- direction.

$$\frac{u}{u_e} = \left(2\eta' - 2\eta'^3 + \eta'^4\right) + \frac{\Gamma}{6}\left(\eta - 3\eta'^2 + 3\eta'^3 - \eta'^4\right) \tag{B.4}$$

While the cross-flow velocity profiles are given in eq. B.5.

$$\frac{v}{u} = \tan\beta_w \left(1 - \eta'\right)^2 \tag{B.5}$$

Where η is the non-dimensional boundary layer height ($\eta' = z/\delta$), Γ is the Pohlhausen shape parameter and β_w is the limiting stream angle.

From these assumed velocity profiles, the integral boundary layer parameters (eqs. B.3) can be expressed as shown in eqs. B.6.

$$\frac{\delta_1}{\delta} = \frac{3}{10} - \frac{\Gamma}{120} \tag{B.6a}$$

$$\frac{\theta_1}{\delta} = \frac{1}{63}\left(\frac{37}{5} - \frac{\Gamma}{15} - \frac{\Gamma^2}{144}\right) \tag{B.6b}$$

$$\frac{\tau_w \delta}{u_e} = 2 + \frac{\Gamma}{6} \tag{B.6c}$$

$$\frac{\delta_2}{\delta} = -\tan\beta_w \left(\frac{1}{7} + \frac{\Gamma}{252}\right) \tag{B.6d}$$

$$\frac{\theta_2}{\delta} = -\tan\beta_w \left(\frac{304}{3465} + \frac{756\Gamma}{10395} + \frac{\Gamma^2}{17820}\right) \tag{B.6e}$$

While the boundary conditions are given by eqs. B.7.

$$z = 0; \qquad u = v = 0 \tag{B.7a}$$

$$z = \infty; \qquad u_e = u_\infty \left(1 - ks\right) \tag{B.7b}$$

Where k is a constant representing the velocity gradient and s is the arc length measured from the leading edge of the blade. This means that an external flow with a linear adverse velocity distribution is assumed.

Integration of the integral boundary layer equations up to the point of laminar separation ($C_f = 0$) yields an equation of the form:

$$ks = f\left(\lambda, s/r\right) \tag{B.8}$$

Where $\lambda = \Omega r/U_\infty$ and ks is the separation factor corresponding to the point of laminar separation.

Equation B.8 represents the relation between the separation factor ks which determines the separation point on the rotating surface and the two key parameters, the ratio of local arc length to local radius s/r, and the ratio of the local rotation speed to free stream velocity λ. The authors computed the separation factor ks for different λ and s/r values, representing the outcomes in the same plot. Basing on these trends, in the development of the post-stall delay model, the ks-curves were approximated by Selig and Du with an equation of the form:

$$ks = 1.6\left(s/r\right)\frac{a' - \left(s/r\right)^{d/\lambda}}{b + \left(s/r\right)^{d/\lambda}} \tag{B.9}$$

Where a', b and d are correction factors, whose values have been retrieved tuning eq. B.9 in order to fit the numerical solution of eq. B.8. a' and b in particular have been found to vary from approximately 0.8 to 1.2, while d from 0.4 to 1.

For a 2D flow without rotation, the separation factor is $ks = 0.1267$; whereas, for a 3D blade in rotation, the separation factor becomes $ks > 0.1267$ and asymptotes to $ks = 0.1267$. Thus, for a 3D flow with rotation, the increase in ks as compared with the 2D case without rotation is given by

$$\Delta ks = \frac{1.6\left(s/r\right)}{0.1267}\frac{a' - \left(s/r\right)^{d/\lambda}}{b + \left(s/r\right)^{d/\lambda}} - 1 \tag{B.10}$$

In light of the empirical and approximate nature of the correction, the authors replaced distance parameter s/r by the blade geometry parameter c/r, that is:

$$\Delta k = \frac{1.6\left(c/r\right)}{0.1267}\frac{a' - \left(c/r\right)^{d/\lambda}}{b + \left(c/r\right)^{d/\lambda}} - 1 \tag{B.11}$$

As shown in Section 2.3.1, formula in eq. B.11 forms the basis of the key function that is used to empirically correct 2D airfoil data for rotation effects that are present in the 3D rotating flow.

B.2 Chiaviaropoulos and Hansen model

In [7] a simplified quasi-3D model has been devised in order to identify the influence of the three-dimensional and rotational effects on the blade section characteristics. Firstly, the governing equations have been derived from incompressible Navier-Stokes equations written in conservative form in the cylindrical coordinate system (Θ, z, r) which rotates with the blade with a constant rotational speed Ω. Θ denotes the peripheral, z the axial and r the radial (i.e. blade spanwise) direction. The infinitesimal length in the peripheral direction is $ds = rd\Theta$. For simplicity the - continuity, Θ-, z- and r- component, respectively - equations are presented in their laminar form:

$$\frac{\partial W_\Theta}{\partial \Theta} + \frac{\partial \left(rW_z\right)}{\partial z} + \frac{\partial \left(rW_r\right)}{\partial r} = 0 \tag{B.12a}$$

$$\frac{\partial W_\Theta}{\partial \Theta} + \frac{\partial}{\partial \Theta}\left[\frac{W_\Theta^2}{r} + \frac{\Phi}{r} - \nu\frac{\partial}{\partial \Theta}\left(\frac{W_\Theta}{r^2}\right)\right] + \frac{\partial}{\partial z}\left[W_z W_\Theta - \nu\frac{\partial W_\Theta}{\partial z}\right] + ...$$

$$...\frac{\partial}{\partial r}\left[W_r W_\Theta - \nu\left(\frac{1}{r}\frac{\partial \left(rW_\Theta\right)}{\partial r}\right)\right] = \nu\frac{2}{r}\frac{\partial}{\partial \Theta}\left(\frac{W_r}{r}\right) - 2W_r\left(\frac{W_\Theta}{r} + \Omega\right) \tag{B.12b}$$

$$\frac{\partial W_z}{\partial t} + \frac{\partial}{\partial \Theta}\left[\frac{W_\Theta}{r}W_z - \nu\frac{\partial}{\partial \Theta}\left(\frac{W_z}{r^2}\right)\right] + \frac{\partial}{\partial z}\left[W_z^2 + \Theta - \nu\frac{\partial W_z}{\partial z}\right] + ...$$

$$...\frac{\partial}{\partial r}\left[W_r W_z - \nu\left(\frac{\partial W_z}{\partial r}\right)\right] = \nu\frac{1}{r}\frac{\partial}{\partial r}\left(\frac{W_z}{W_r}\right) \tag{B.12c}$$

$$\frac{\partial W_r}{\partial t} + \frac{\partial}{\partial \Theta}\left[\frac{W_\Theta}{r}W_r - \nu\frac{\partial}{\partial \Theta}\left(\frac{W_z}{r^2}\right)\right] + \frac{\partial}{\partial z}\left[W_z W_r - \nu\frac{\partial W_r}{\partial z}\right] + ...$$

$$...\frac{\partial}{\partial r}\left[W_r^2 + \frac{p}{\rho} - \nu\frac{1}{r}\frac{\partial \left(rW_r\right)}{\partial r}\right] = -\nu\frac{2}{r^2}\frac{\partial W_\Theta}{\partial \Theta} - \frac{W_r^2}{r} + \frac{1}{r}\left(W_\Theta + \Omega r\right)^2 \tag{B.12d}$$

Where $\mathbf{W}$ stands for the velocity vector, ρ for the fluid density, p is the static pressure and ν the kinematic viscosity. Φ is a pressure-like term including the centrifugal effect:

$$\Phi = \frac{p}{\rho} - \frac{1}{2}\left(\Omega r\right)^2 \tag{B.13}$$

The authors then proceeded to integrate eqs. B.12 along the radial direction and the mean radial values were obtained. The mean value operator reads:

$$\overline{()} = \frac{1}{\Delta r}\int_r^{r+\Delta r} ()\, dr \tag{B.14}$$

Consequently:

$$\overline{\left(\frac{\partial\, ()}{\partial t}\right)} = \frac{\partial \overline{()}}{\partial t}, \quad \overline{\left(\frac{\partial\, ()}{\partial \Theta}\right)} = \frac{\partial \overline{()}}{\partial \Theta}, \quad \overline{\left(\frac{\partial\, ()}{\partial z}\right)} = \frac{\partial \overline{()}}{\partial z} \tag{B.15}$$

Thereafter, the resulting system of equations was subjected to the following assumptions:

$$\overline{\left(\frac{\partial \left(W_\Theta/r\right)}{\partial r}\right)} = \overline{\left(\frac{\partial W_z}{\partial r}\right)} = \overline{\left(\frac{\partial W_r}{\partial r}\right)} \tag{B.16a}$$

$$\overline{\left(\frac{\partial^2 \left(W_\Theta/r\right)}{\partial r^2}\right)} = \overline{\left(\frac{\partial^2 W_z}{\partial r^2}\right)} = \overline{\left(\frac{\partial^2 W_r}{\partial r^2}\right)} \tag{B.16b}$$

$$\overline{\left(\frac{\partial \left(p_t/\rho\right)}{\partial r}\right)} = q \Rightarrow \overline{\left(\frac{\partial \left(p/\rho\right)}{\partial r}\right)} = q - \frac{1}{\overline{r}}\left(\overline{W_\Theta} + \Omega\overline{r}\right)^2 \tag{B.16c}$$

$$\overline{ab} \approx \overline{a}\overline{b} \tag{B.16d}$$

That is:

- The mean value of the radial derivatives (first and second) of the contra variant relative velocity components are small and can be neglected.
- The radial derivative of the total pressure (related to the specific work) is a section-depending constant, called q.
- Mean values of products can be approximated with products of mean values.

Far from the airfoil the velocities are $W_\Theta = -\Omega r$, $W_z = const$ and $W_r = 0$, which directly satisfy the above quasi-3D assumptions.

The outcome of this analysis is then presented by the authors with eqs. B.17. They include the modified continuity equation (eq. B.17a) and the momentum balance in Θ-, z- and r- directions respectively (eqs. B.17b,B.17c,B.17d).

$$\nabla_{2D} \cdot \mathbf{W_{2D}} + \frac{\overline{W_r}}{\overline{r}} = 0 \tag{B.17a}$$

$$N_{2D}\left(\overline{W_\Theta}\right) + \frac{\partial \overline{\Phi}}{\partial s} = -2\overline{W_r}\left(\frac{\overline{W_\Theta}}{\overline{r}} + \Omega\right) + \frac{2}{Re\overline{r}}\frac{\partial \overline{W_r}}{\partial s} \tag{B.17b}$$

$$N_{2D}\left(\overline{W_z}\right) + \frac{\partial \overline{\Phi}}{\partial z} = 0 \tag{B.17c}$$

$$N_{2D}\left(\overline{W_r}\right) = -q + \frac{2}{\overline{r}}\left(\overline{W_\Theta} + \Omega r\right)^2 - \frac{1}{Re}\left(\frac{2}{\overline{r}}\frac{\partial \overline{W_\Theta}}{\partial s} + \frac{1}{\overline{r}^2}\overline{W_r}\right) \tag{B.17d}$$

Where subscript 2D denotes equivalent two-dimensional properties on the (Θ, z) plane and N stands for the Navier-Stokes scalar operator, namely:

$$\mathbf{W_{2D}} \equiv \left[\overline{W_\Theta}, \overline{W_z}\right], \quad \nabla_{2D} \equiv \left[\frac{1}{\overline{r}}\frac{\partial}{\partial \Theta}, \frac{\partial}{\partial z}\right] \tag{B.18a}$$

$$\overline{\Phi} = \frac{\overline{p}}{\rho} - \frac{1}{2}\left(\Omega \overline{r}\right)^2 \tag{B.18b}$$

$$N_{2D} = \frac{\partial}{\partial t} + \mathbf{W_{2D}} \cdot \nabla_{2D} - \frac{1}{Re}\nabla_{2D}^2 \tag{B.18c}$$

The authors then proceed to adimensionalize the set of eqs. B.17 using the local blade section c. Finally, the initial conditions needed for the solution of the quasi-3D problem are provided under the assumption that the far-field flow is uniform at the absolute frame, having a nonzero axial velocity component ($V_z = W_z$) only.

Chaviaropoulos and Hansen shared at this point some interesting considerations to better understand the consequences of their model.

- Compared to the 2D problem, the quasi-3D one includes an additional momentum equation, eq. B.17d, for the radial velocity component, while two source terms involving this radial component are added into the continuity and the peripheral momentum equations. All the extra terms are weighted with the $1/\overline{r}$ factor, or the c/r ratio in dimensional quantities. This implies that the quasi-3D effects become stronger as c/r increases (inner part of the blade), which is in full accordance with the experimental observations. The 3D and rotational effects are mainly due to the Coriolis force and not to the centrifugal force. The effect of the latter is well-hidden in the pressure-like term Φ (eq.B.13) which replaces the static pressure in the governing equations. Clearly, eqs. B.17 degenerate to the 2D Navier-Stokes equations when $c/r \to 0$ and $q = 0$ ("work distribution" parameter).

- The influence of the twist angle is implicitly taken into account in the above analysis. Actually, β along with α define the equivalent rotational speed Ω affecting the strength of the source terms appearing in the peripheral and radial momentum equations (B.17b,B.17d).

- The main production term appearing in eq. B.17b, i.e. the first RHS term, is always positive leading to the generation of a radial velocity field of positive sign (hub to tip). The production is larger as c/r increases and W_Θ decreases. W_Θ takes small values within the wall shear layer, the thickness of which depends on the Reynolds number. Thus, a Reynolds number effect should be expected. When the flow is detached, radial velocity is generated in the separation bubble. The above remarks imply that even for large c/r ratios the solutions of the 2D and the quasi-3D problem are rather identical at low angles of attack and high Reynolds numbers, where the flow remains fully attached and the boundary layer is thin, while larger differences, depending on the extent of the separation bubble, should be expected for detached flows.

Appendix C

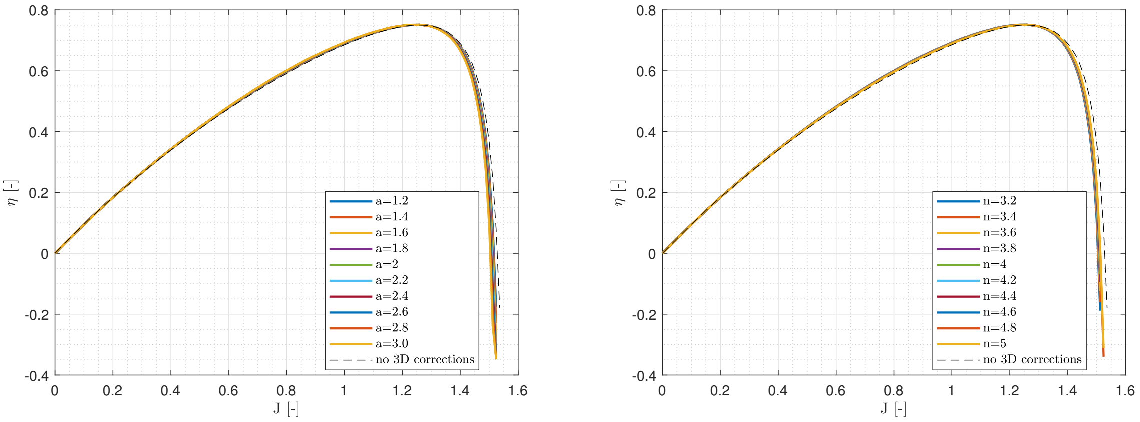
Comprehensive graphical results

In this Chapter, a comprehensive list of all non-reported plots (for the sake of brevity) in the main corpus is given.

In Figures C.1, C.2 and C.3, the parametric study of the influence of a, h and n coefficients in the 3D model for the Clark-Y airfoil is reported.

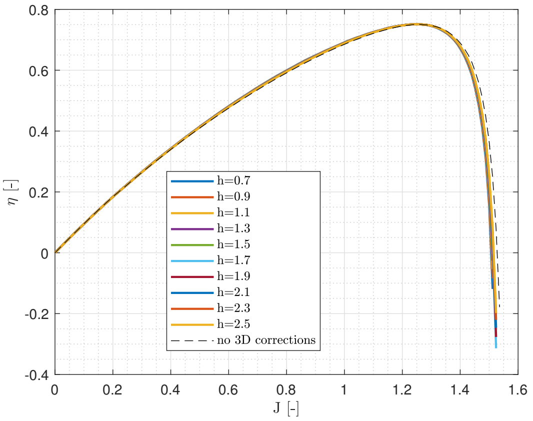
In Figures C.4 and C.5 are reported the results of the two optimizations performed adopting Clark-Y airfoil. The first starting with both chord and twist linear distributions, the second starting with constant chord and linear twist distributions.

In Figures C.6 and C.7 performance outputs obtained adopting the optimized - from linear chord and linear twist distributions as initial guess - propeller mounting RAF-6 and Clark-Y airfoil with variable pitch are shown. In Figures C.8 and C.9 are reported the performance outputs obtained adopting the optimized - from both initial guesses - propeller mounting NACA16-012 airfoil with variable pitch.



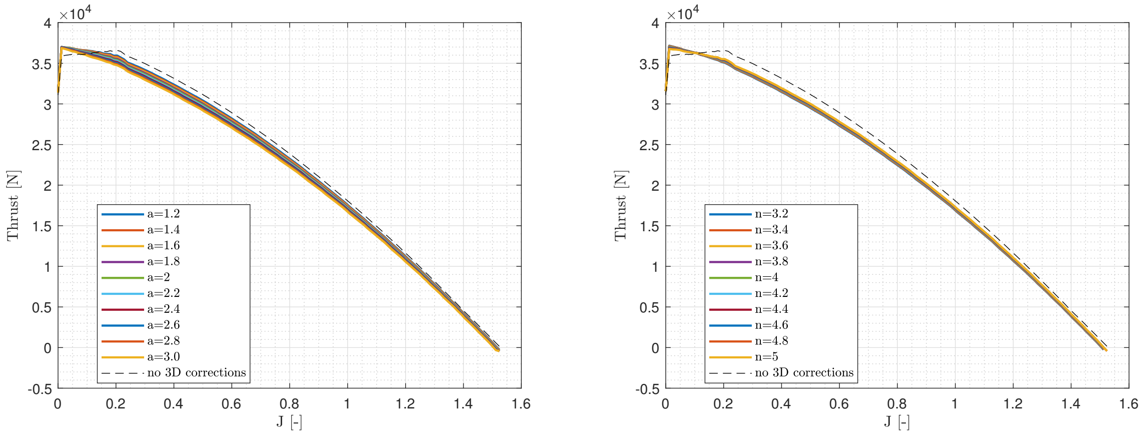
(a) η *variation for different* a *values.*

(b) η *variation for different* n *values.*



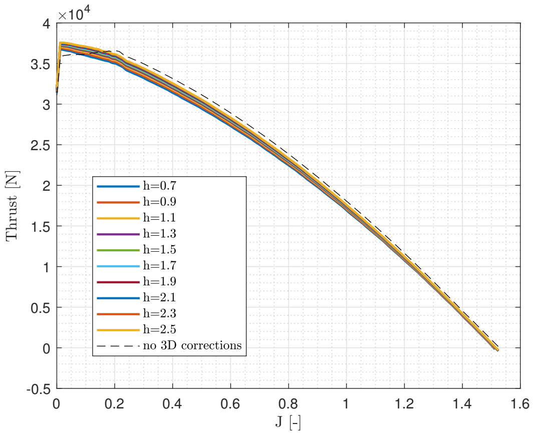
(c) η *variation for different* h *values.*

Figure C.1: Clark-Y airfoil: η variation for different corrective coefficients values.



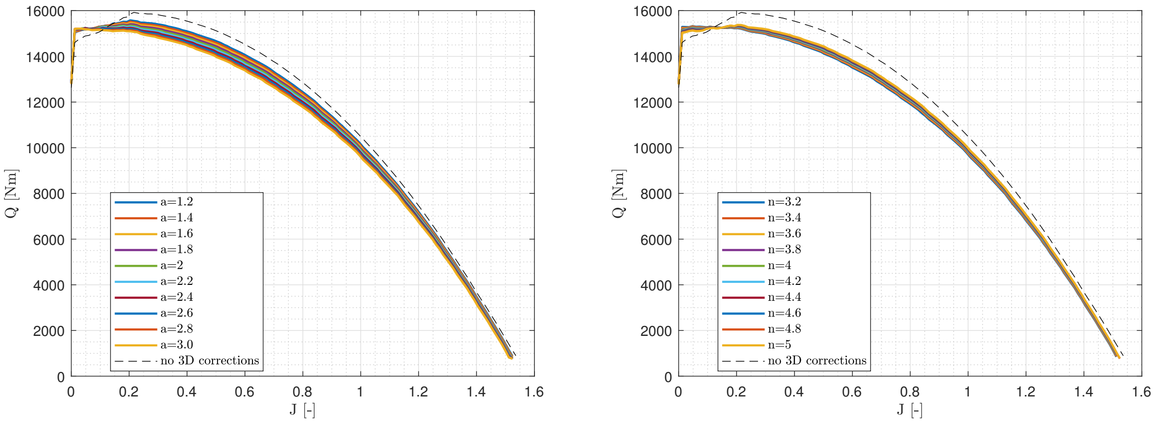
(a) *Thrust variation for different a values.*

(b) *Thrust variation for different n values.*

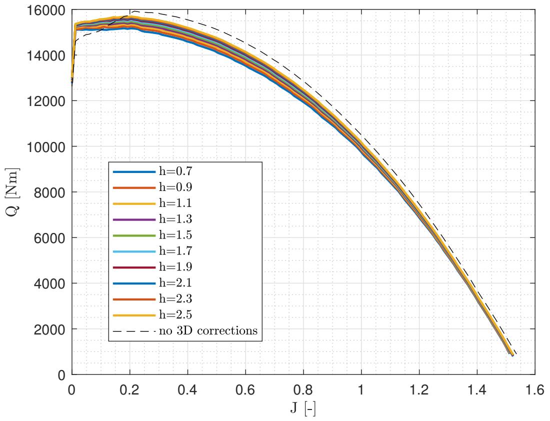


(c) *Thrust variation for different h values.*

Figure C.2: Clark-Y airfoil: T variation for different corrective coefficients values.

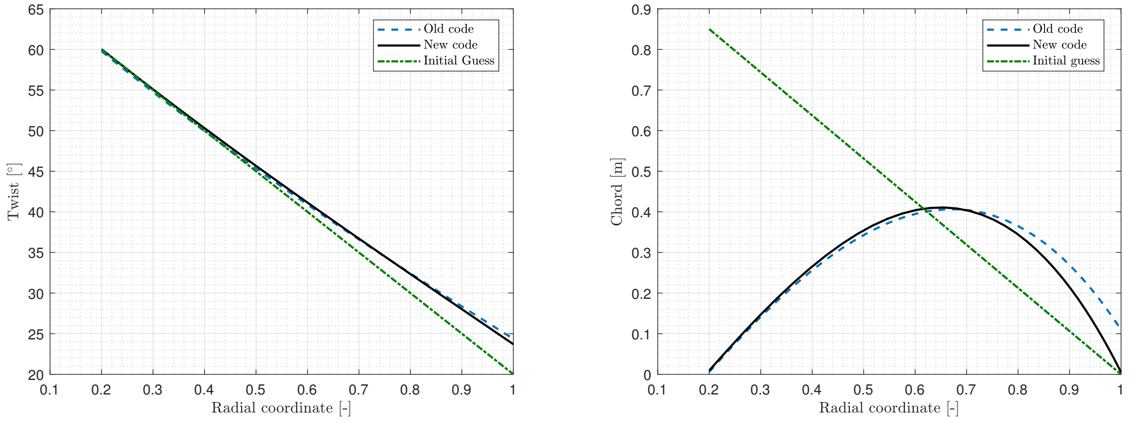


(a) *Torque variation for different a values.* (b) *Torque variation for different n values.*



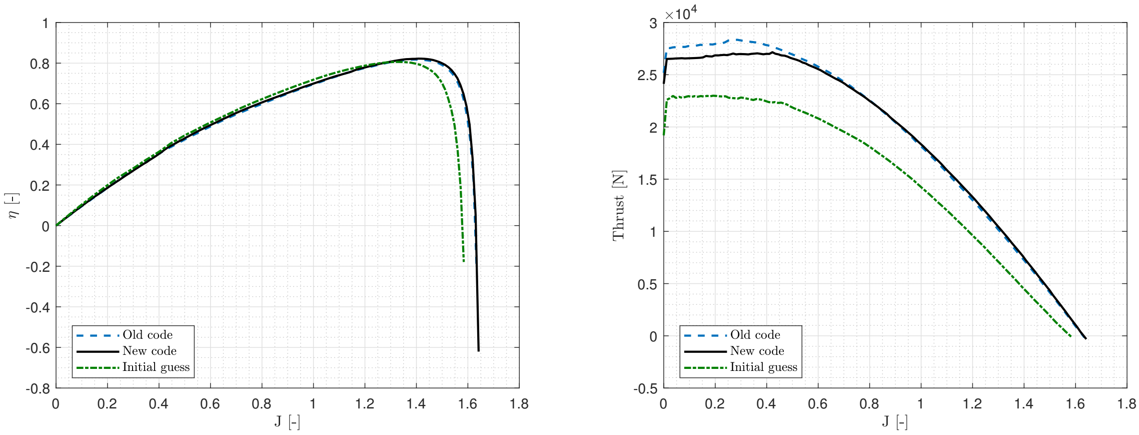
(c) *Torque variation for different h values.*

Figure C.3: Clark-Y airfoil: Q variation for different corrective coefficients values.



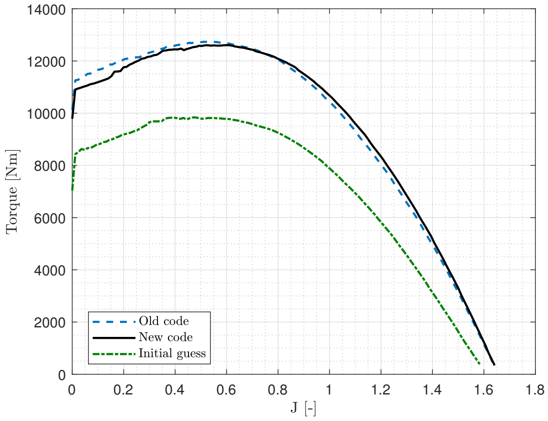
(a) *Twist distribution.*

(b) *Chord distribution.*



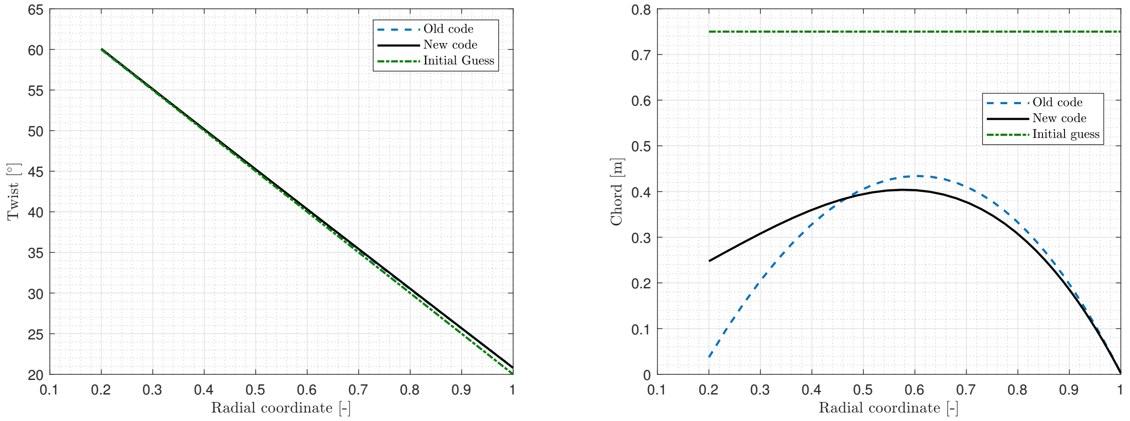
(c) *Efficiency.*

(d) *Thrust.*

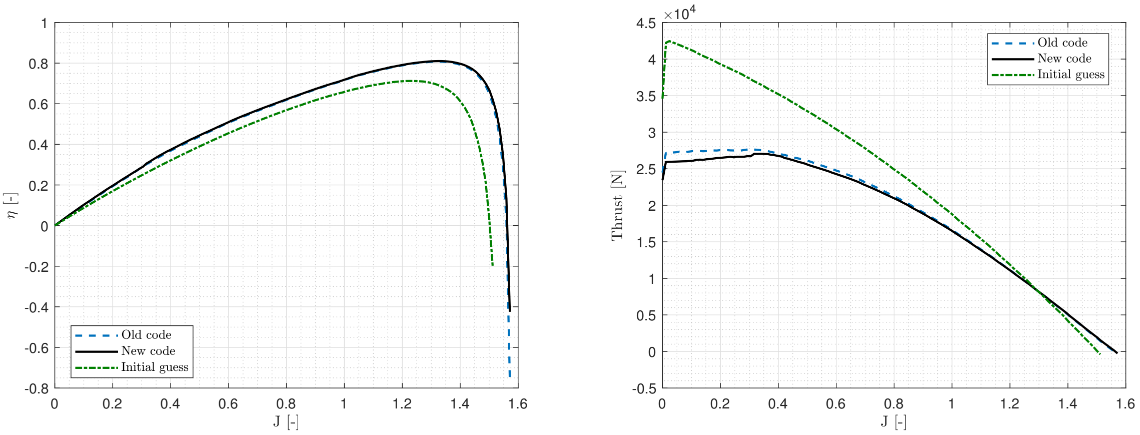


(e) *Torque.*

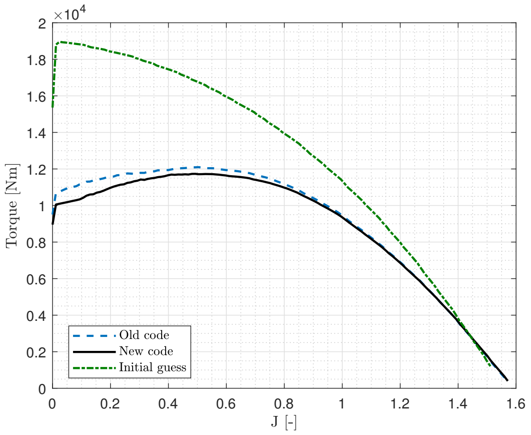
Figure C.4: Clark-Y airfoil optimization results - linear chord initial guess: old code vs new code.



(a) *Twist distribution.* (b) *Chord distribution.*

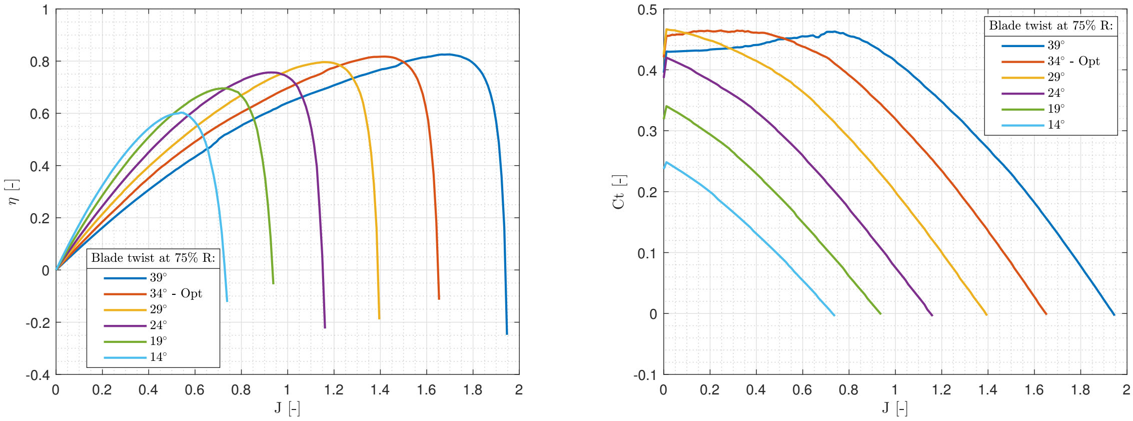


(c) *Efficiency.* (d) *Thrust.*



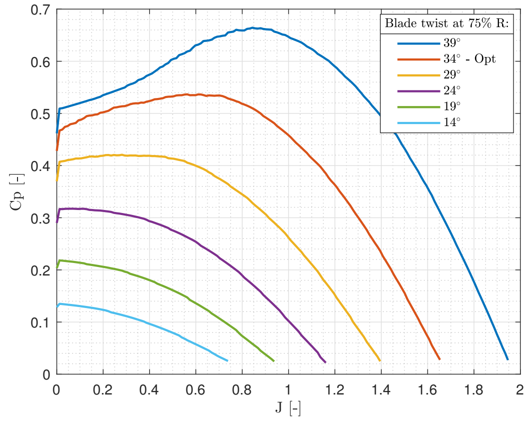
(e) *Torque.*

Figure C.5: Clark-Y airfoil optimization results - constant chord initial guess: old code vs new code.



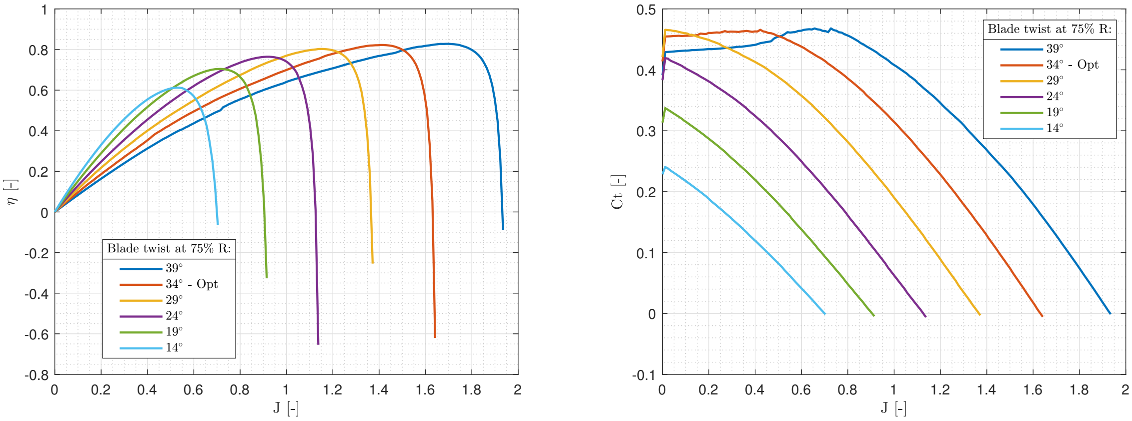
(a) *Efficiency.*

(b) *Thrust coefficient.*

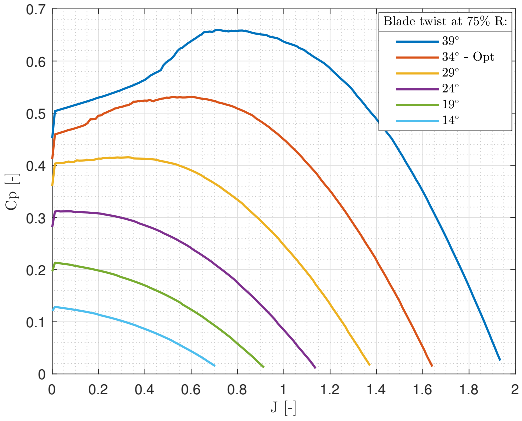


(c) *Power coefficient.*

Figure C.6: Optimized propeller - linear chord and linear twist initial guess - variable pitch performance. RAF-6 airfoil.

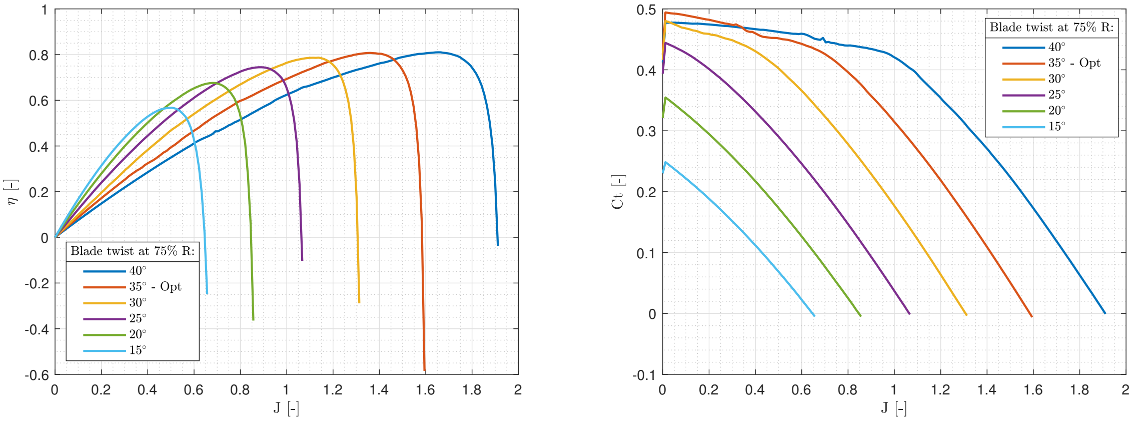


(a) *Efficiency.* (b) *Thrust coefficient.*



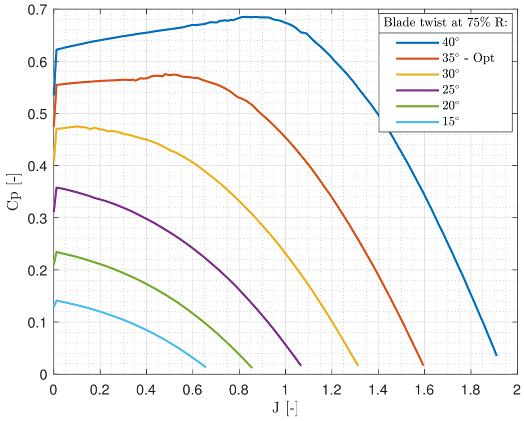
(c) *Power coefficient.*

Figure C.7: Optimized propeller - linear chord and linear twist initial guess - variable pitch performance. Clark-Y airfoil.



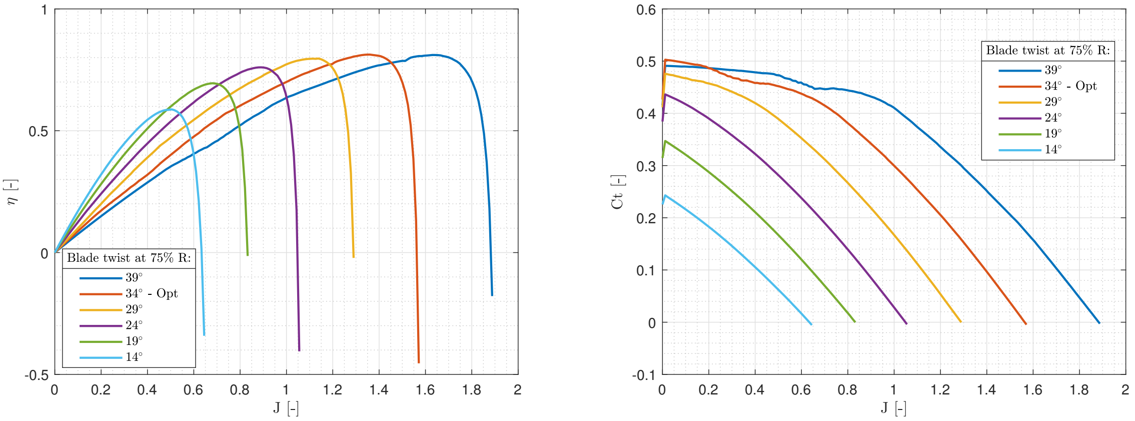
(a) *Efficiency.*

(b) *Thrust coefficient.*



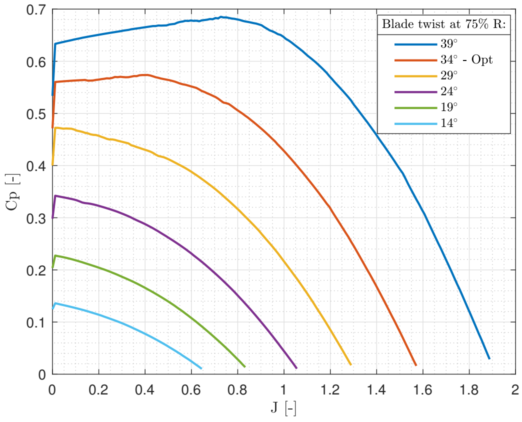
(c) *Power coefficient.*

Figure C.8: Optimized propeller - constant chord and linear twist initial guess - variable pitch performance. NACA16-012 airfoil.



(a) *Efficiency.*

(b) *Thrust coefficient.*



(c) *Power coefficient.*

Figure C.9: Optimized propeller - linear chord and linear twist initial guess - variable pitch performance. NACA16-012 airfoil.

Bibliography

[1] O. Gur. Design of a quiet propeller for an electric mini unmanned air vehicle. *Journal of Propulsion and Power*, May 2009.

[2] J. De Zubeldia Martín. Optimal design methods with choice of aerodynamic profiles for aircraft propellers. Master thesis, Politecnico di Milano, July 2020.

[3] I. H. Abbot and A. E. Von Doenhoff. *Theory of Wing Sections.* National Aeronautics and Space Administration, 1959.

[4] O. Gur. Propeller performance at low advance ratio. *Journal of Aircraft*, March 2005.

[5] H. Himmelskamp. *Profile Investigation on a Rotating Airscrew.* Ph.D. Dissertation, Gottingen, Germany, 1945.

[6] H. Snel et al. Sectional prediction of 3-d effects for separated flow on rotating blades. *National Aerospace Laboratory NLR, Amsterdam*, September 1992. The Eighteenth European Rotorcaft Forum, Avignon, France.

[7] P. K. Chaviaropoulos and M. O. L. Hansen. Investigating three-dimensional and rotational effects on wind turbine blades by means of a quasi-3d navier-stokes solver. *Journal of Fluid Engineering*, June 2000.

[8] J. Bosscher and B. Montgomerie. Influence of blade rotation on the sectional aerodynamics of rotational blades. Technical report, National Aerospace Laboratory NLR, Amsterdam, The Netherlands, September 1996.

[9] C. Thumthae and T. Chitsom. Improved 3d stall delay model for wind turbine design. *The 12th Americas Conference on Wind Engineering (12ACWE), Seattle, Washington, USA*, June 2013.

[10] Z. Du and M. S. Selig. A 3-d stall-delay model for horizontal axis wind turbine performance prediction. *AIAA, Paper 21*, 1998.

[11] M. Drela. Xfoil: An analysis and design system for low reynolds number airfoils. Technical report, The American Institute of Aeronautics and Astronics, June 1989.

[12] M. Drela and H. Youngren. Xfoil 6.9 user primer, November 2001.

[13] J. Morgado et al. Xfoil vs cfd performance predictions for high lift low reynolds number airfoils. *Aerospace Science and Technology*, March 2016.

[14] O. Günel. Comparison of cfd and xfoil airfoil analyses for low reynolds number. *International Journal of Energy Applications and Technologies*, December 2016.

[15] J.T. Hwang and A. Ning. Large-scale multidisciplinary optimization of an electric aircraft for on-demand mobility. *AIAA SciTech Forum*, January 2018. AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference.

[16] A. Ning. A simple solution method for the blade element momentum equations with guaranteed convergence. *Wind Energy*, April 2013. National Renewable Energy Laboratory , Golden, Colorado 80401, USA.

[17] Performance of propellers. `https://web.mit.edu/16.unified/www/FALL/thermodynamics/notes/node86.html`.

[18] N. Tarraran. Optimal approach to propeller design. Master thesis, Politecnico di Milano, 2018.

[19] C.D. Harris. Two dimensional characteristics of the naca 0012 airfoil in the langley 8-foot transonic pressure tunnel. Technical report, NASA technical memorandum 81927, April 1981.

[20] C. L. Ladson. Effects of independent variation of mach and reynolds numbers on the low-speed aerodynamic characteristics of the naca 0012 airfoil section. Technical report, NASA technical memorandum 4074, October 1988.

[21] ANSYS Inc. Lecture 7: Turbulence modeling - introduction to ansys fluent, February 2014.

[22] Simscale cae forum - what is y+? `https://www.simscale.com/forum/t/what-is-y-yplus/82394`, May 2018.

[23] F. R. Menter. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA J. Vol. 32*, August 1994. NASA Ames Research Center, Moffett Field, California 94035.

[24] D. C. Wilcox. Re-assessment of the scale-determining equation for advanced turbulence models. *AIAA J. vol. 26, no. 11*, 1988.

[25] Openfoam v6 user guide: 5.3 mesh generation with blockmesh. `https://cfd.direct/openfoam/user-guide/v6-blockmesh/`.

[26] X. Jian-Hua et al. An aerodynamic design method of propeller airfoils with geometric compatibility as constraints. *AIAA Aviation 2019 Forum*, June 2019.

[27] Airfoil tools. `http://www.airfoiltools.com`.

[28] ITT-Costruzioni aeronautiche. Cap.16: Elica aerea - studio dell'elica.

[29] L. Sartori. *System design of lightweight wind turbine rotors.* Doctoral Programme in Aerospace Engineering, Politecnico di Milano, January 2019.

[30] Skybrary-constant speed propeller. `https://www.skybrary.aero/index.php/Constant_Speed_Propeller`.