

Parametric Modeling and Optimization of Advanced Propellers for Next-Generation Aircraft

Ing. Peter Cornelis Klein

Flight Performance & Propulsion



Parametric Modeling and Optimization of Advanced Propellers for Next-Generation Aircraft

by

Ing. Peter Cornelis Klein

To obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday August 31, 2017 at 2:00 PM.

Student number:	4250079	
Project duration:	April 1, 2016 – August 31, 2017	
Thesis registration number:	149#17#MT#FPP	
Thesis committee:	Prof. Dr.-Ing. G. Eitelberg,	TU Delft (FPP)
	Dr. ir. M. Pini,	TU Delft (FPP)
	Dr. J. Sodja,	TU Delft (AS&CM)
	Ir. T. Sinnige,	TU Delft (FPP), Daily supervisor
	Ir. S. Vitale,	TU Delft (FPP), Daily supervisor

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

Preface

I hereby present the final report of my thesis entitled "Parametric modeling and Optimization of Advanced Propellers for Next-Generation Aircraft". This report is written in the pursuit of the degree of Master of Science at the Delft University of Technology, at the faculty of Aerospace Engineering, at the department of Flight Performance & Propulsion.

I've dedicated the last year to produce a thesis that I am very proud of. It was a hard and stressful but rewarding year that marks the end of my time at the Delft University of Technology, which started because I felt I wasn't challenged enough at my previous education. Well, it is safe to say that I got what I wished for, here at the faculty of Aerospace Engineering. It has been five intense, challenging, but amazing years, in which I have had the privilege of getting to learn from inspiring teachers and other people.

I would like to thank first Tomas and Salvatore, who guided me through a difficult thesis project. They have been very patient with me and were always willing to help or offer advice at the tough moments. I would like to thank the remaining members of the committee who made themselves available so that I can graduate. And of course, my parents, Evelien, and Jelle; you are simply amazing.

*Ing. Peter Cornelis Klein
Delft, August 2017*

Summary

As traditional fossil fuels become scarcer and more attention is given to environmental impact of the combustion of fossil fuels, for next-generation aircraft, the focus of development will mainly be on reducing fuel consumption. Open rotor engines have the advantage over conventional turboprops that they are expected to perform 25% to 30% better in terms of fuel consumption. The topic of this thesis focuses on one of the current challenges related to open rotor configurations; the unwanted unsteady effects imposed on the aerodynamic characteristics of a propeller, imposed by an upstream pylon. There is a potential for reducing undesired propeller-ylon installation effects by taking these effects into account in aerodynamic design process of the propeller. A propeller design optimization routine is implemented that includes a parametric modeling tool, a low- and high-fidelity performance analysis method and an existing installation effects performance model.

The low-fidelity isolated propeller performance analysis method was used to approximate the isolated performance of a propeller in an optimization environment. Three existing tools were initially considered: Xrotor, JavaProp and a scheme by Adkins & Liebeck. A validation study was performed, based on a low-speed propeller design (N250), to assess accuracy of each method's performance prediction capabilities, and also to compare the differences between the different methods in order to make a selection for one method to be used in the remainder of this thesis. It was shown that out of the three existing methods considered, Xrotor was the most accurate and the second most reliable method. JavaProp was the least accurate method and the Adkins & Liebeck scheme was the least reliable method. It was therefore decided to use Xrotor for the remainder of the thesis.

The high-fidelity isolated propeller performance analysis method was used to verify the precision of the low-fidelity isolated propeller performance analysis method for the benefit of the optimization routine. A CFD analysis method with the open-source solver SU2 was set up, including the implementation of a mesh generator. Then, a validation study was performed, based on a high-speed propeller design (APIAN), to assess the accuracy of the CFD analysis method. Moreover, the method's sensitivity to mesh refinement was studied, in order to study the method's sensitivity to mesh refinement, in order to make a selection for one mesh refinement level to be used in the remainder of this thesis. The validation of the high-fidelity isolated propeller performance analysis method showed that the implemented CFD method is valid and able to predict isolated propeller performance with acceptable accuracy. The most notable differences include an increasing deviation between the CFD thrust coefficient results and the reference data with increasing advance ratio. Moreover, the prediction of the tip vortex intensity in the wake of the propeller and the position of the blade leading edge vortex on the blade suction side turned out to be challenging. For the levels of mesh refinement considered here, no significant improvements were observed in the CFD results. For the remainder of this thesis, one refinement level was selected based on the sensitivity results, corresponding to a grid of 4.7 million cells.

The propeller parametrization tool was used to accurately represent a complex propeller geometry with the least amount of parameters, for the optimization routine. Some highlights of the tool include the possibility to parametrize a spinner and hub geometry, the modeling of blade chord, pitch, sweep, and lean distributions, as well as methods for defining blade thickness and camber. A verification study was performed in which three different tests were conducted. In the first test, it was demonstrated that, for two different propellers (N250, APIAN) of different levels of complexity, it is possible to accurately recreate the propeller geometry with the parametrization tool. In the second test, the parametrized N250 propeller model was used to demonstrate that for the low-fidelity isolated propeller performance analysis method, it is possible to achieve almost identical accuracy as with the reference N250 propeller model. In the final test, the parametrized APIAN propeller model was used to demonstrate that for the high-fidelity isolated propeller performance analysis method, it is possible to achieve similar accuracy as with the reference APIAN propeller model.

In the setup of the propeller blade optimization for minimized installation effects, an existing installed propeller performance analysis method was implemented. For the objective function, two specific unwanted installation effects are considered: unsteady fluctuations in thrust performance and worse installed noise performance. No noise prediction methods were included in the analysis, since they are not within the scope of this thesis. Instead, worse installed noise performance was represented by focusing on the main aerodynamic drivers of installed propeller noise. These effects were formulated in different candidate objective functions, for which the sensitivity of each objective function is investigated. Furthermore, a secondary ob-

jective function is implemented by applying a multi-objective optimization strategy with the goal of maximizing mean installed efficiency performance. A single constraint is implemented to keep the mean installed thrust performance constant.

The results of the propeller blade optimization for minimized installation effects include a study of potential candidate objective functions, a study of the influence of different operating points on the optimization routine and a detailed investigation of the performance of three different optimum solutions selected from the first two. In the multi-candidate objective optimization study, it was discovered that at a high propeller thrust setting, the potential gain in mean installed efficiency is limited to about 10%. The different candidate objective functions have different design trends, meaning that if one optimizes for one, the other installation effects become worse. The candidate objective function related to installed noise performance was used for the remainder of this thesis. In the multi-operating point optimization study, two optimizations were performed at different propeller thrust settings (intermediate and low). In the detailed investigation into the performance of three different optimum solutions, selected from the previous studies, it was discovered that the optimum solutions for high thrust setting and low thrust setting have converged to more or less the same solution, which means that the overall performance is also very similar. The optimum solution at intermediate thrust setting performed significantly better throughout the entire operating range, compared to the other two. This solution was therefore presented as the ultimate optimum solution for the objective of minimizing installation effects.

The main design changes in the final optimum propeller design, compared to the baseline propeller design were an increase in chord in the blade hub region and an increased pitch distribution gradient and a higher advance ratio. For a high thrust setting, the final optimum solution was a 30% installed noise metric improvement over the baseline propeller design. At the same thrust setting, the installed propeller efficiency improved by 8%, the unsteady fluctuations in installed thrust performance were reduced by 20% and the unsteady fluctuations in installed torque performance were reduced by 26%. For a low thrust setting, the final optimum solution was a 25% installed noise metric improvement over the baseline propeller design. At the same thrust setting, there was no significant improvement in installed propeller efficiency, the unsteady fluctuations in installed thrust performance were reduced by 29% and the unsteady fluctuations in installed torque performance were reduced by 36%.

It is concluded that the original hypothesis of this thesis, which was that there is a potential for reducing unwanted propeller-pylon installation effects, by taking these effects into account during the aerodynamic design process of the propeller, is confirmed. considerable improvements were made, compared to the reference propeller. It is recommended that for future work, additional improvements can be made on the current setup. This includes extending the functionality of the low-fidelity isolated propeller performance analysis method to take sweep and lean effects into account, additional scrutineering of the differences found in the CFD analysis method validation study, and a high-fidelity verification study using CFD of the final blade design resulting from the optimization for minimized installation effects.

Contents

Preface	iii
Summary	v
Nomenclature	ix
1 Introduction	1
1.1 Background	1
1.2 Research aim and objectives	2
1.3 Thesis outline	3
2 Low-fidelity propeller performance analysis method	5
2.1 Overview of existing tools	5
2.1.1 Xrotor	5
2.1.2 JavaProp	8
2.1.3 Adkins & Liebeck scheme	9
2.2 Validation study	10
2.2.1 Reference case	10
2.2.2 Results	10
2.3 Low-fidelity performance analysis method selection	11
3 High-fidelity propeller performance analysis method	15
3.1 Setup of CFD analysis method	15
3.1.1 Mesh generator	15
3.1.2 Solver setup	18
3.2 Validation of CFD method	20
3.2.1 Mesh sensitivity study setup	20
3.2.2 Mesh Sensitivity study results	22
3.2.3 Discussion of results	29
4 Parametrization tool	31
4.1 Tool structure	31
4.1.1 Preliminary design method	33
4.1.2 Blade parametrization	34
4.1.3 Spinner parametrization	39
4.2 Parametrization tool verification	40
4.2.1 Low-fidelity analysis verification	40
4.2.2 High-fidelity analysis verification	44
4.2.3 Discussion of verification results	53
5 Propeller blade optimization for minimized installation effects - Setup	55
5.1 Installed performance model	55
5.1.1 Pylon wake model	55
5.1.2 Modeling installed effects	57
5.1.3 Example of installation effects	58
5.2 Definition of optimization problem	59
5.2.1 Reference baseline case	59
5.2.2 Formal definition of optimization problem	61
5.2.3 Optimization algorithm	69
6 Propeller blade optimization for minimized installation effects - Results	71
6.1 Accuracy of Low- and high-fidelity analysis method	71
6.2 Multi-candidate objective optimization study	72
6.2.1 Minimizing installed thrust performance	73
6.2.2 Minimizing individual aerodynamic blade noise drivers	75

6.2.3	Minimizing combined aerodynamic blade noise drivers	78
6.2.4	Selection of final objective function	81
6.3	Different operating point optimization study	83
6.3.1	Optimization at intermediate thrust setting: 1.40	83
6.3.2	Optimization at low thrust setting: 1.75	84
6.3.3	Selection of optimum designs	84
6.4	Detailed performance of selected optimum designs	86
6.4.1	Overview of optimum designs	86
6.4.2	Optimum designs in off-design conditions.	88
6.4.3	Selection of final optimum design	90
7	Conclusions & Recommendations	93
7.1	Conclusions.	93
7.2	Recommendations for future work	95
	Bibliography	97
A	CFD validation results	99
A.1	Overall performance related results.	99
A.2	Blade pressure distribution related results	101
A.3	Wake flow field related results.	105
B	Parametrization tool CFD verification results	109
B.1	Blade pressure distribution results	109
B.2	Spinner pressure distribution results	110
B.3	Wake velocity results	111
C	Additional different operating condition optimization results	113
C.0.1	Initial operating point at advance ratio equal to 1.40.	113
C.0.2	Initial operating point at advance ratio equal to 1.75.	115

Nomenclature

List of symbols

Symbol	Description	Unit
B	Propeller number of blades	-
C	Sutherlands constant	-
D	Aerodynamic drag	N
F	Prandtl tip loss factor	-
J	Advance ratio	-
J_i	Bessel function of the first kind of order i	-
K_i	Modified Bessel function of the second kind of order i	-
L	Aerodynamic lift	N
L	Characteristic Reynolds length	m
M	Mach number	-
P	Propeller power required	W
Q	Propeller torque	Nm
R	Propeller tip radius	m
	Specific gas constant	$\text{J kg}^{-1} \text{K}^{-1}$
Re	Reynolds number	-
S	Installed performance Sears function	-
T	Propeller thrust	N
	Temperature	K
V	Velocity or speed	m s^{-1}
a	Propeller induced velocity factor	-
	Speed of sound	m s^{-1}
b_W	Pylon wake semi-width	m
c	Chord length	m
h	height	m
l	length	m
p	Pressure	N m^{-2}
q	Dynamic pressure	N m^{-2}
r	Radius	m
s	Distance along the curvature from the LE of an airfoil	m
t	Airfoil thickness	m
$y+$	CFD dimensionless wall distance	-

List of greek symbols

Symbol	Description	Unit
α	Angle of attack	rad
β	Propeller blade pitch angle	rad
	Prandtl Glauert compressibility factor	-
ρ	Free-stream air density	kgm^{-3}
Λ	Propeller blade sweep angle	rad
Ω	Propeller rotational speed	rads^{-1}
η	Propeller efficiency	-
γ	Specific heat ratio	-
λ	Propeller tip velocity ratio	-
μ	Dynamic viscosity	$\text{kgm}^{-1}\text{s}^{-1}$
ω	Vorticity	s^{-1}
ϕ	Propeller blade section flow angle	rad
ϕ_w	Installed performance polar angle	rad
ψ	Propeller blade lean angle	rad
σ	Propeller blade solidity	-
	Reduced frequency	-
θ	Gradient of the curvature of an airfoil	rad
ζ	Propeller blade section velocity displacement ratio	-

List of abbreviations

Abbreviation	Description	Abbreviation	Description
APIAN	Advanced Propulsion Integration Aerodynamics and Noise	NASA	National Aeronautics and Space Administration
BEM	Blade Element Momentum	PIV	Particle Image Velocimetry
CAD	Computer Aided Design	RANS	Reynolds-Averaged Navier-Stokes
CATIA	Computer Aided Three-dimensional Interactive Application	RMS	Root-Mean-Square
CCW	Counter-Clock Wise	SA	Spalart-Almaras
CFD	Computational Fluid Dynamics	SLSQP	Sequential Least-Squares Programming
COBYLA	Constrained Optimization BY Linear Approximation	SQP	Sequential Quadratic Programming
CP	Control Point	SST	Shear-Stress-Transport
CW	Clock Wise	STEP	Standard for the Exchange of Product model data
GE	General Electric	TE	Trailing Edge
LE	Leading Edge	UDF	Unducted Fan
NACA	National Advisory Committee for Aeronautics	rpm	Revolutions per minute

Introduction

1.1. Background

Today, most transport aircraft are propelled by turbofans, which are very capable in producing thrust in order to propel hundreds of tons at transonic speeds with acceptable fuel consumption. However, as traditional fossil fuels become scarcer and more attention is given to environmental impact of the combustion of fossil fuels (now and in the future), for next-generation aircraft, the focus of development will mainly be on reducing fuel consumption. Besides efforts to reduce aircraft weight and aerodynamic drag, a lot of efforts focus on the propulsion systems themselves. The traditional turbofan is limited in achieving higher levels of propulsive efficiency (up to 65% of installed propulsive efficiency, see figure 1.1).

The main source of thrust of a turbofan is the thrust generated by the bypass. Potentially, a reduction in fuel consumption can be achieved by increasing the size of the bypass. This increases the thrust production for the same engine power, which means that engine power can be reduced to achieve the same thrust. While this is certainly possible to a certain degree, increasing the bypass size effectively means an increase in nacelle size, which adds weight and aerodynamic drag. Also, an increased bypass size results in a much larger tip speed of the fan blades, which increases the structural loads on the blade, requiring stiffer, heavier blades. To overcome these issues there are two competing ideas. The first is a geared turbofan. This concept uses a complex gear system to decouple the main fan from the first compressor stage, to allow the main fan to operate at a lower rotational velocity. This decreases the tip speed of the fan blades, thereby decreasing the structural loads and thus allows for increasing the bypass size. While this solution solves the high fan blade tip speed issue, it will still have an increased nacelle size, which adds weight and aerodynamic drag.

The second idea is to switch to an open rotor configuration, shown in figure 1.2. Open rotor engines have the advantage over conventional turbofans that they allow for significantly larger bypass ratios, very low fan pressure ratios and the elimination of the duct, which reduces the nacelle drag. Additionally, a popular variation off the open rotor concept is the implementation of a counter-rotating propulsion system. This is a dual propeller setup, where the second propeller recovers the swirl of the first propeller, which will increase the propulsive efficiency [3]. In terms of fuel consumption, open rotor systems are expected to perform 25% to

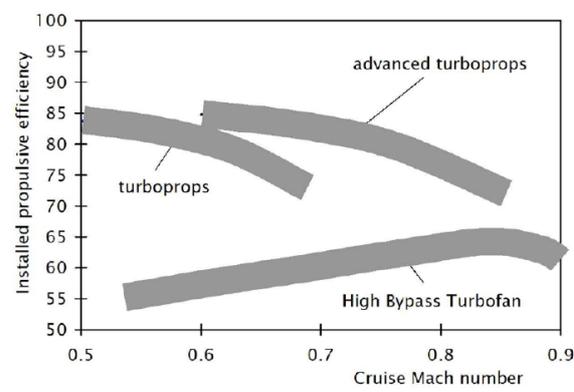


Figure 1.1: Indication of installed propulsive efficiency for different forms of aircraft propulsion [1].

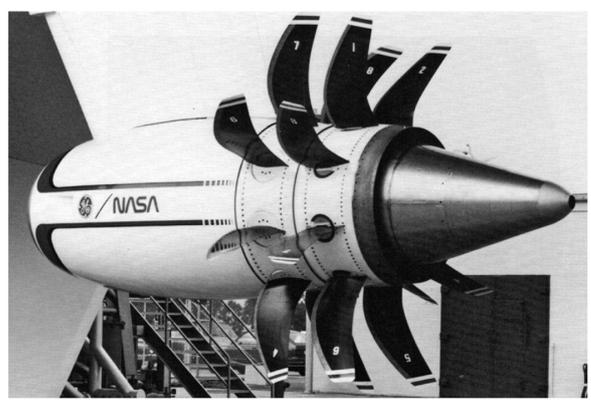


Figure 1.2: Example of open rotor concept, GE NASA UDF engine (1987) [2].

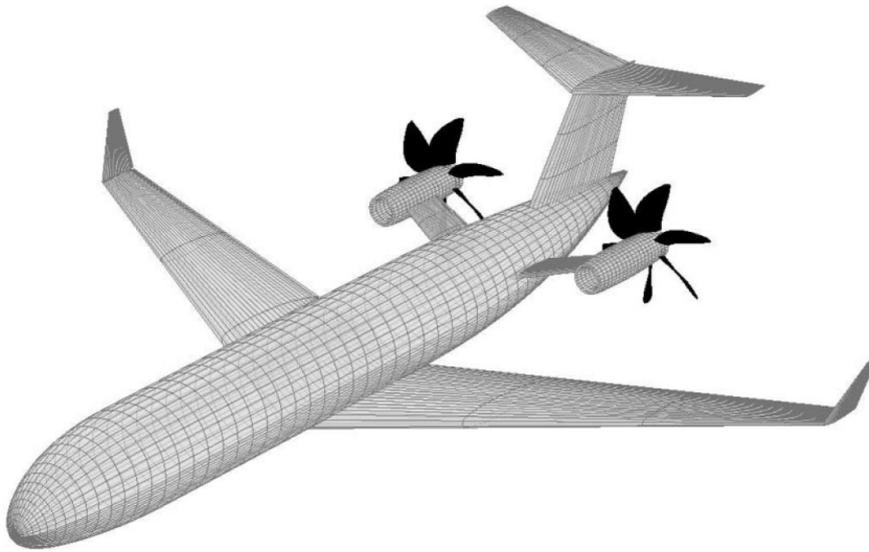


Figure 1.3: Example of a pylon-mounted open rotor configuration installed in a pusher layout [5].

30% better than traditional turbofans [4]. However, there are a number of fundamental challenges that need to be solved, before the open rotor concept can be considered as a viable alternative to traditional turbofans. These challenges are mainly focused on reducing noise emissions, and the reliability of the pitch change mechanism (especially with contra-rotating rotors), according to Hayes [4].

A popular solution to limiting the noise impact of the open rotor concept, is to opt for a rear-mounted engine configuration as shown in figure 1.3. In this way, the engine noise source is positioned at the farthest point from the cabin. This solution will solve in some degree, the noise issues associated with open rotor propulsion, by reducing the noise experience inside the cabin, but not the ground noise experience. However, it also creates new challenges such as installation issues, blade containment issues and issues related to disturbed, unsteady rotor inflow, that need to be solved before the open rotor can be considered as a viable alternative to traditional turbofans.

1.2. Research aim and objectives

The topic of this thesis focuses on one of the challenges related to open rotor, rear-mounted engine configurations, more specifically, the unwanted unsteady effects imposed on the aerodynamic characteristics of a propeller, imposed by an upstream pylon. Sinnige et al. [5] demonstrated that the propeller- pylon installation effects originate from the impingement of the pylon wake on the propeller, and the resulting non-uniform propeller inflow leads to unsteady blade loading with associated performance, and noise penalties. Moreover, it can be envisioned that unsteady blade loading also leads to structural penalties due to vibrations and unsteady stresses acting on joints and connections. The hypothesis of this thesis is that there is a potential for reducing undesired propeller- pylon installation effects by taking these effects into account during the aerodynamic design process of the propeller.

The Flight Performance and Propulsion group at the aerospace engineering faculty of Delft University of Technology is currently developing a tool which will incorporate rapid analysis and design optimization of propellers and various turbo-machinery rotor configurations. For this purpose, parametric modeling techniques combined with high-fidelity CFD software can be employed to obtain high-quality results. The current thesis assignment aims at extending the capabilities of this tool to allow the rapid analysis and design optimization of propellers. When finished, this tool could be used to improve and speed-up the conceptual design process of propeller blades. For industry, the potential lies in a reduction in development cost of propellers. For the university, the potential lies in aiding further research into propellers and help to solve the current problems associated with open rotor systems. For now, the developed tool is used as a platform that allows for investigating the hypothesis of this thesis.

The main goal of this thesis is to study the potential benefit on propeller performance of considering installation effects during the conceptual design phase. This is achieved by applying the rapid propeller analysis & design tool and a simplified model that predicts the propeller- pylon installation effects. It should be noted that the concept of rapid propeller analysis can only realistically be achieved by applying low-fidelity propeller performance analysis methods or tools. These have limited accuracy and will require additional ver-

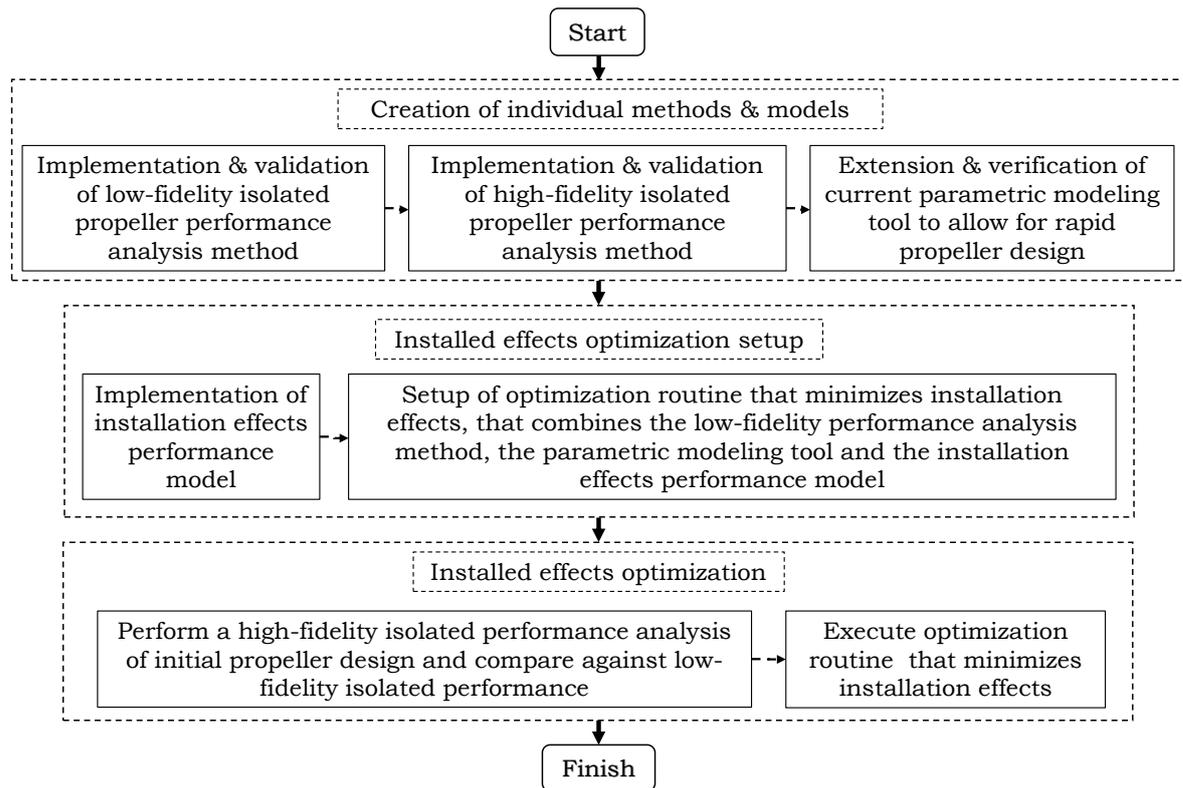


Figure 1.4: Overview of the research work performed in this thesis.

ification, by applying a high-fidelity propeller performance analysis method such as CFD. In order to achieve the main goal of this thesis, research objectives are defined and given in figure 1.4.

1.3. Thesis outline

As indicated in figure 1.3, the main body of the thesis work consists of three main phases. In the first phase, individual methods, tools and models are created that are required to set up and perform the optimization routine, later. This phase starts with the implementation and validation of a low-fidelity isolated propeller performance analysis method. Three existing methods are initially considered, which were discovered during the literature study. Then, a validation study is performed with a relevant propeller case, in order to select and validate the most appropriate method. This is the subject of chapter 2. In order to be able to verify results generated with the low-fidelity isolated propeller performance method, a high-fidelity CFD method is implemented. For this research, the CFD solver SU2 is used, which requires the implementation of a coupling with a mesh generator tool and a post-processing method. After the implementation, the CFD method is validated with existing experimental data. This is the subject of chapter 3. In the next step, the capabilities of the current blade modeling tool are extended to allow for the parametric modeling of propeller blades. It is essential to define a parametric model for propeller blades, that is both efficient in its use of design variables and versatile in its ability to generate different, relevant propeller geometries. Once correctly implemented, the parametrization tool is validated by reconstructing an existing propeller geometry and comparing the results from a low- and high-fidelity propeller performance analysis for the reference and the reconstructed propeller. This is the subject of chapter 4.

In the second phase, a mathematical model is implemented that is able to predict the influence of installation effects on propeller performance. An existing model by Sinnige et al. [5] is used. Then, the setup of the optimization routine that minimizes installation effects, is defined. The parametric modeling tool is used for rapid generation of new propeller designs and relevant input for the performance analysis modules. The low-fidelity isolated propeller performance analysis method and the installation effects performance model are used to quantify the installation effects in terms of overall performance. A summary of the Sinnige installed performance model and the setup of the installation effects optimization are given in chapter 5.

In the third and final phase, the optimization routine that minimizes installation effects, is executed. First, a high-fidelity isolated performance analysis is performed with the initial propeller design. The results of this initial analysis are compared to results from the low-fidelity isolated performance method to identify their

differences. Then, the optimization routine as defined in the second phase, is executed. This is the subject of chapter 6.

This report is concluded in chapter 7, where the whole process is reviewed. Conclusions and recommendations are made, based on the findings in the other chapters and the hypothesis of this thesis is reviewed.

2

Low-fidelity propeller performance analysis method

In order to be able to achieve the main goal of this thesis, it is necessary to implement a low-fidelity propeller analysis tool. This will be used to approximate the performance of a propeller in an optimization environment. The aim of this chapter is to present an investigation into different existing propeller analysis methods, a validation study with the tools with the most potential and finally the selection of one of the methods that will be used throughout this research.

2.1. Overview of existing tools

During the literature study, a number of low-fidelity propeller design and analysis tools were identified that showed potential in terms of applicability, capability and validity. These were Xrotor [6], JavaProp [7], Jblade [8], OpenProp [9] and PROP_DESIGN [10]. However, additional requirements associated with implementation into an automated scripting environment limit the choice to only Xrotor and JavaProp. Because of this, the number of available options is too limited in order to make a proper selection. Therefore, these existing tools are considered, with the addition of a scheme that needs to be implemented, which was found in a relevant paper by Adkins & Liebeck [11]. It presents a relatively simple scheme for analyzing and designing propellers, based on blade element momentum theory. It is also the original source material for JavaProp, [7], however initial testing of the Adkins & Liebeck scheme and JavaProp revealed that there are significant differences in the results. Therefore, both are considered here. The remainder of this section gives a detailed overview of the implementation of the three tools that are considered.

2.1.1. Xrotor

Xrotor [12] is an interactive program for the design and analysis of ducted and free-tip propellers and windmills. Xrotor was created by Mark Drela & Hal Youngren, who are also the creator of the well-known Xfoil [12]. Xrotor is a Fortran-based software; hence it requires a Fortran compiler to create a command prompt based working environment. The process for performing an analysis in Xrotor is given in figure 2.1.

Inputs

In order to perform a propeller analysis in Xrotor, a comprehensive description of the propeller is required. As shown in figure 2.1, the required inputs consists of a set of data describing the propeller geometry and the operating conditions.

The required Xrotor operating conditions include advance ratio, free-stream velocity and secondary free-stream flow conditions such as speed of sound, air density, dynamic viscosity and altitude. Furthermore, it is required to discretize a propeller blade into two-dimensional sections, of which the pitch, chord and aerodynamic properties are known. In order to determine the aerodynamic properties of each section, the overall shape (non-dimensionalized with the chord) of each section must be defined in the form of sets of coordinates. Finally, it is also required to define trivial geometric parameters, such as number of blades and hub & tip radius.

Xrotor requires detailed aerodynamic performance characteristics for each blade section [6]:

- Reference Reynolds number
- Zero-lift angle of attack

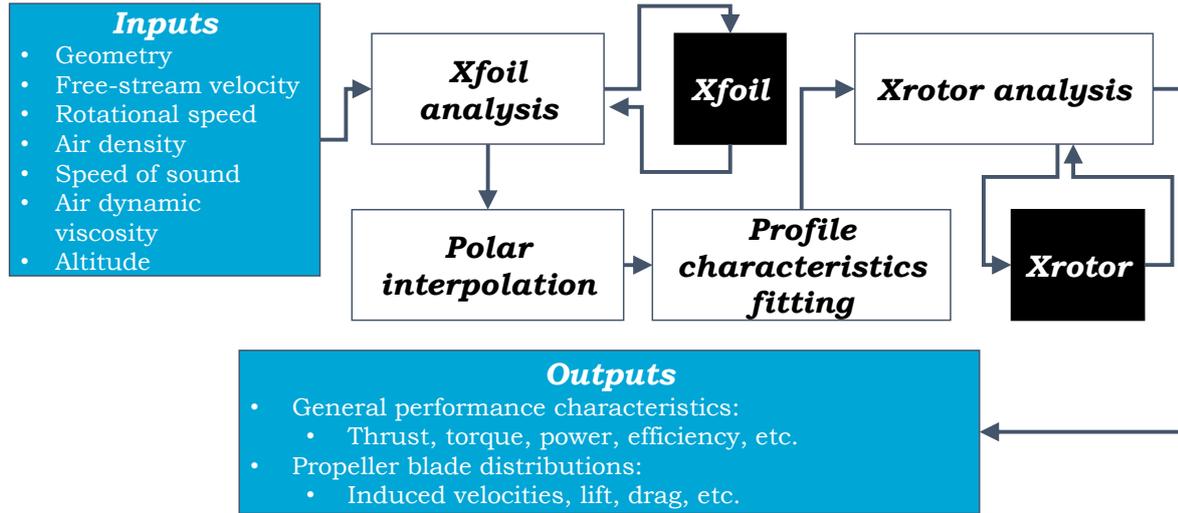


Figure 2.1: Overview of the Xrotor analysis process. Blue boxes represent start and end of process, white boxes represent processes programmed in Python, black boxes represent external computational tools.

- Lift slope
- Lift slope in stall region
- Maximum lift coefficient
- Minimum lift coefficient
- Lift increment to stall
- Minimum drag coefficient
- Lift coefficient at Minimum drag coefficient
- Induced drag slope (dc_d/dc_l^2)
- Pitching moment coefficient at aerodynamic center
- Reynolds number scaling exponent
- Critical Mach number

These so-called profile characteristics are dependent on section shape geometry and thus additional steps are required to determine them. The Reynolds number scaling exponent is used by Xrotor to adjust the profile characteristics for any change in Reynolds number due to the existence of induced velocities compared to the reference Reynolds number, and also in case the same airfoil profiles are used for analyzes at different advance ratios. Initial testing revealed that any change of this variable resulted in Xrotor's solution diverging. Therefore, the scaling factor is kept constant at zero and for every advance ratio, a separate set of profile characteristics is generated. Throughout this research, it is assumed that any change in Reynolds number due to existence of induced velocities is small enough, so that only the change in advance ratio has to be taken into account.

Xfoil analysis

As mentioned before, Xrotor requires the definition of profile characteristics at each blade section, which can only be found through additional steps. The first step in determining the profile characteristics is to determine the lift drag and moment polars, as a function of angle of attack. The preferred solution is to use a dedicated tool from the same creator as Xrotor called Xfoil [12]. With Xfoil, it is possible to determine the lift, drag and moment polars, by applying a viscous analysis without Mach correction. No Mach correction is applied, because Xrotor itself applies already a Mach correction, which makes the Xfoil Mach correction redundant.

To perform a viscous analysis in Xfoil, a local Reynolds number needs to be supplied. The local Reynolds number is defined in equation 2.1:

$$V_{\text{eff}}(J, r) = \sqrt{V_{\infty}(J)^2 + (\Omega r)^2} \quad (2.1a)$$

$$Re(J, r) = \frac{\rho \cdot V_{\text{eff}}(J, r) \cdot c(r)}{\mu} \quad (2.1b)$$

Because Xfoil is sensitive to high panel angles and large increases of angle of attack, the analysis is split into two separate analyzes, one ranging from 0 to -35 degrees and one ranging from 0 to +35 degrees, with steps

Table 2.1: Results from a preliminary sensitivity study into the influence of boundary layer development on method's accuracy. General propeller performance data is generated for N250 propeller, one advance ratio equal to 0.4 and one blade pitch angle equal to 25 degrees. Default Xrotor settings are applicable.

Case	Thrust coefficient			Power coefficient		
	Value [-]	Deviation [-]	Error [%]	Value [-]	Deviation [-]	Error [%]
Reference	0.3053	0.0	0.0	0.2675	0.0	0.0
$n = 0.01$	0.2758	-0.0295	-9.6626	0.2908	0.0233	8.7178
$n = 4.5$	0.3492	0.0439	14.3858	0.2988	0.0313	11.6935
$n = 9.0$	0.3499	0.0446	14.6151	0.2990	0.0315	11.7570

of 0.25 degrees. This range is deemed appropriate because, for the first step, the only requirement is that, all typical profile characteristics are captured within a large enough range. It is impossible to know beforehand, for every analysis performed in this thesis, realistic, general limits. However, the fact that these limits are set does not mean that Xfoil will actually analyze the full angle of attack range. In most cases, Xfoil has difficulty converging in the stall region and will exit prematurely, before the upper or lower limit.

A fully turbulent boundary layer is assumed in this thesis because the flow over the propeller is complex and accurately predicting the boundary layer of a three-dimensional propeller blade, using two-dimensional airfoil sections is almost impossible. Xfoil uses the e^n method to determine the transition point at which the boundary layer transitions from a laminar to a turbulent boundary layer. A preliminary sensitivity study is performed into the influence of the n factor on Xrotor's accuracy, of which the results are presented in table 2.1. Three n factors are considered, where $n = 0.01$ relates to a fully turbulent boundary layer, $n = 9.0$ is the default value of Xfoil and the e^n method and finally $n = 4.5$ is a convenient middle point. Overall, the results presented here are not very accurate, however, keep in mind that default Xrotor settings were used, and improvement is expected, once these are optimized. If the different cases are compared, it is clear that opting for a fully turbulent boundary layer at $n = 0.01$ results in the most accurate performance prediction, thereby supporting the original assumption.

A separate Xfoil analysis is performed to determine the critical Mach number. The critical Mach number is defined as the free stream Mach number at which the local velocities at the airfoil first exceed the speed of sound. Consider the well-known Prandtl-Glauert relation that relates the pressure distribution to the Mach number distribution over an airfoil:

$$c_p = \frac{2}{\gamma M_\infty^2} \left[\left(\frac{\gamma + 1}{2(1 + 0.5(\gamma - 1)M_\infty^2)} \right)^{\frac{\gamma}{1-\gamma}} - 1 \right] \quad (2.2)$$

The critical Mach number can be found by taking the minimum pressure coefficient on the suction side and iteratively solving equation 2.2 for the Mach number. An Xfoil analysis is performed to determine the pressure distribution for a given angle of attack. Since the angle of attack cannot be anticipated in advance, an angle of attack of zero is deemed to be appropriate.

Profile characteristics fitting

Xfoil is only capable of determining the airfoil profiles for each station. In order to determine the profile characteristics, a fitting procedure is required that finds all the Xrotor profile characteristics. Sinnige & Veldhuis [13] created a routine to produce profile characteristics, based on the Xfoil airfoil polars. Xrotor uses a mathematical model to describe the airfoils polars with the profile characteristics, such that lift, drag and moment can be computed for a given angle of attack. The routine uses this mathematical model in a numerical optimization routine, with the objective of finding the profile characteristics, by fitting the mathematical model to the original airfoil polars.

Xrotor analysis

Xrotor offers three different mathematical models that can be used to performance the propeller performance analysis with; a graded-momentum formulation, a potential formulation and a vortex formulation. According to the Xrotor documentation [6], graded-momentum formulation is the classical theory of propellers revived recently by Larrabee [14]. It relies on the Betz-Prandtl tip loss fudge factor which assumes that the rotor has a low advance ratio and/or many blades. The main advantage of this model is the very low computation time, however, the disadvantages of this model are the limited accuracy as demonstrated in table 2.1, and the small range of advance ratios for which this model is suitable (upto 0.5). According to the Xrotor documentation [6], potential formulation is a more modern approach which solves for the helical-symmetric potential

Table 2.2: Results from a preliminary sensitivity study into the influence of mathematical performance analysis model on Xrotor's accuracy. General propeller performance data is generated for N250 propeller, one advance ratio equal to 0.4 and one blade pitch angle equal to 25 degrees.

Case	Thrust coefficient			Power coefficient		
	Value [-]	Deviation [-]	Error [%]	Value [-]	Deviation [-]	Error [%]
Reference	0.3053	0.0	0.0	0.2675	0.0	0.0
Graded-momentum	0.2758	-0.0295	-9.6626	0.2908	0.0233	8.7178
Potential	0.2929	-0.0124	-4.0747	0.2690	0.0015	0.5533
Vortex	0.2997	-0.0056	-1.8244	0.2694	0.0019	0.6953

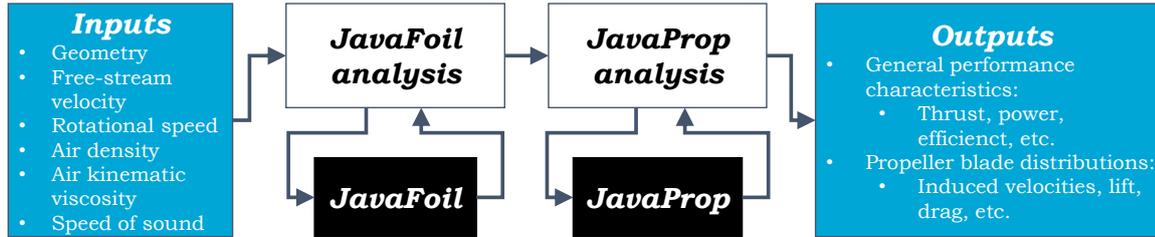


Figure 2.2: Overview of the JavaProp analysis process. Blue boxes represent start and end of process, white boxes represent processes programmed in Python, black boxes represent external computational tools.

flow about a rigid helicoidal wake and hence is valid for all blade numbers and advance ratios. It's an extension of Goldstein's 2 and 4 blade solution [15] to all blade numbers and arbitrary radial load distributions. While this model requires more computation time compared to the graded-momentum formulation, it is suitable for all advance ratios. According to the Xrotor documentation [6], vortex formulation uses a discrete vortex wake for induced velocity calculations. The velocity on the propeller lifting line is calculated from discrete line vortices on the rigid helicoidal wake. It is computationally the most expensive, however, it can treat more advanced propeller geometries.

A preliminary sensitivity study is performed into the influence of each formulation on Xrotor's accuracy, of which the results are presented in table 2.2. Although the advance ratio is selected such, that it is within the range of suitable advance ratios for the graded-momentum formulation, this mathematical model performs significantly worse compared to the rest. It under-predicts the thrust by 9.7% and over-predicts the power by 8.7%. These errors are too large for the purpose of this research. Vortex and potential formulation both predict the power coefficient with a small error of the same order, however potential formulation is slightly better. The main difference is in the thrust coefficient prediction, in which the vortex formulation performs much better with an error of 1.8% compared to a 4.1% error for the potential formulation. Finally, during this preliminary sensitivity study, the computation time was experienced and this revealed that the computation time of the vortex formulation was not significantly worse than that of the potential formulation. Therefore, the decision is made to use the vortex formulation, throughout this research.

To perform an Xrotor analysis, only an input file is required that contains the description of the propeller geometry, operating conditions and profile characteristics for each blade station with a specific format. Then, Xrotor is operated through a command-prompt based environment, which is scripted. Since the Reynolds scaling model is neglected, it is only appropriate to perform an analysis for a single operating point at a time.

2.1.2. JavaProp

JavaProp [7] is a design and analysis tool for propellers and wind turbines. It was created by Martin Hepperle, who also created the airfoil analysis tool JavaFoil [16]. As the name suggests, the program is written in Java and is usable as a Java application with a graphical user interface. However, this tool has been created with a scripting library, which means it is applicable in an automatized optimization environment. The process for performing an analysis in JavaProp is given in figure 2.2.

Inputs

In order to perform a propeller analysis with JavaProp, a comprehensive description of the propeller is required. As shown in figure 2.2, the required inputs consists of a set of data describing the propeller geometry and the operating conditions. The required JavaProp operating conditions include propeller rotational speed,

free-stream velocity and secondary free-stream flow conditions such as speed of sound, air density and kinematic viscosity.

The propeller geometry description consists of general parameters such as number of blades, spinner diameter and propeller diameter, as well as discrete distributions describing the pitch and chord distribution for a propeller blade. There is no limit to the size of these discrete distributions. Just as with Xrotor, the geometry is discretized into two-dimensional sections, where the airfoil shape is represented by the aerodynamic properties of that section. Unfortunately, JavaProp only allows for the definition of aerodynamic properties at four, equally spaced sections ($r/R = [0, 0.33, 0.66, 1.0]$). The aerodynamic properties are determined with JavaFoil [16], which requires the definition of a set of non-dimensional coordinates describing the airfoil shape.

JavaFoil analysis

The aerodynamic properties for a given section are determined with JavaFoil [16], an airfoil analysis tool of the same creator as JavaProp. The reason for using JavaFoil is that JavaProp requires that the aerodynamic properties are defined in a specific JavaFoil file format. JavaFoil has a similar interface as JavaProp and is similar to Xfoil [12], in that it uses a higher-order panel method for the potential flow analysis and employs an integral boundary layer method to calculate the viscous boundary layer contribution. According to the JavaFoil documentation [16], this integral boundary layer method does not handle laminar separation bubbles or large scale separation, and works best in a Reynolds number regime between 500,000 and 20,000,000. This method integrates simplified boundary layer equations from the stagnation point, until it finds the transition point, based on one of the transition prediction methods that JavaFoil employs. Once transition is detected, it stops the integration and an empirical drag penalty is added. One of the transition prediction methods available is the same method used by Xfoil; the e^n method. Unfortunately, while it is possible in Xfoil to specify the n factor, thereby allowing control over the boundary layer type, in JavaProp, this possibility is not available. This applies to the remaining available transition prediction methods as well. As demonstrated in the setup of Xfoil and Xrotor, the unavailability of this option will have a negative effect on the accuracy of the propeller performance prediction method. It is convenient to use the same transition prediction method throughout this research, which means that for JavaFoil, the e^n method is used.

In order to perform a relevant JavaFoil analysis for JavaProp, beside free-stream conditions, a reference Reynolds number and an appropriate angle of attack range needs to be supplied. According to the JavaProp documentation [7], an appropriate angle of attack range is from -45 to 45 degrees with a step of 2.5. The JavaFoil profile files are loaded into the JavaProp installation directory, so that they can be accessed by JavaProp.

JavaProp analysis

According to the JavaProp documentation [7], the JavaProp analysis module is a so-called "Blade element Momentum Method" (BEM method), which is based on the work of Adkins & Liebeck [11] and Larrabee [14]. It should be noted that the scheme of Adkins & Liebeck is treated is also considered individually in the next sub-section. This BEM method is comparable to the graded-momentum formulation from Xrotor, which means that it is not computationally expensive, only suitable to a limited regime of advance ratios and less accurate.

JavaProp offers two methods for performing an analysis: a multi-point or a single point analysis. However, when using the multi-point analysis, no radial distributions of relevant parameters are generated. More importantly, the polar data supplied by the JavaFoil analysis is only valid for one single operating point. Both in the JavaProp documentation [7] and in the Adkins & Liebeck scheme [11], no mention is made of any Reynolds number scaling model as is the case for Xrotor. Because of this a single-point analysis is used.

2.1.3. Adkins & Liebeck scheme

The Adkins & Liebeck scheme [11] is based on propeller lifting-line theory. A scheme is supplied for both for propeller performance analysis and propeller design.

Inputs

In order to perform an analysis, just as with JavaProp and Xrotor, a number of inputs are required. Compared to JavaProp and Xrotor, this method has significantly more flexibility in terms of defining the propeller. The propeller geometry and the operating conditions are the same as for Xrotor, however, for this scheme to work it is required to supply for each station the airfoil polar in raw data form. The method of generating raw profile data for Xrotor, can be re-applied here.

Performance analysis

The Adkins & Liebeck analysis scheme revolves around iteratively solving for the flow angle at each blade section:

1. Determine initial flow angle neglecting induced velocities, with the relation: $\phi = \tan^{-1} \left(\frac{V_{\infty}}{\Omega r} \right)$
2. Determine aerodynamic properties (lift and drag coefficient), based on angle of attack: $\alpha = \beta - \phi$
3. Determine induced velocity components, based on aerodynamic properties and Prandtl tip loss factor
4. Calculate new flow angle based on induced velocity components
5. If flow angle is converged, step 6, otherwise, step 2
6. Determine propeller performance characteristics based on flow angle

In the case of square-tip propellers, some adjustments to the scheme are required as suggested by Adkins & Liebeck [11]. The scheme uses the Prandtl tip loss factor to compensate for 3D effects in the calculation. At the tip, the Prandtl tip loss factor is equal to zero. According to the definition of the induced velocity factors in equation 2.3, and ignoring the nominator σ term for the moment, a zero tip loss factor will result in induced velocity factors of -1 and +1, which are physically impossible. For zero-chord tips, this is not an issue since the nominator σ term will make the induced velocity factors zero anyway. However, for square-tip propellers, this does not occur. To solve these issues, one solution is to fix the amplification factors for the tip as suggested by Adkins & Liebeck [11]. The suggested value for both amplification factors is 0.7.

$$a_{\text{axial}} = \frac{\sigma K_{\text{axial}}}{F - \sigma K_{\text{axial}}} \quad (2.3a)$$

$$a_{\text{tangential}} = \frac{\sigma K_{\text{tangential}}}{F + \sigma K_{\text{tangential}}} \quad (2.3b)$$

Where F is the Prandtl tip loss factor and:

$$K_{\text{axial}} = \frac{c_y}{4 \sin^2 \phi} \quad (2.4a)$$

$$K_{\text{tangential}} = \frac{c_x}{4 \cos \phi \sin \phi} \quad (2.4b)$$

$$\sigma = \frac{Bc}{2\pi r} \quad (2.4c)$$

Finally, there are convergence issues in off-design conditions. Adkins & Liebeck [11] suggest some recursive combination of the main iteration variable; the flow angle. The final solution proposed involves averaging the old and the new value of the iteration variable to slow down the divergence and improve the success rate of the scheme in off-design conditions. By monitoring the internal data flow, it is noticed that the most critical point for this scheme in off-design conditions, is the significant difference between the initial guess and the first iteration value of the flow angle. The initial guess method significantly under-predicts the flow angle, which causes the scheme to fail. This phenomenon cannot be solved with a recursive combination of the old and new iterative variables alone which only dampens the phenomenon. Therefore, in off-design conditions, it is recommended that for this scheme to converge, the initial flow angle guess must be scaled such that it is closer to the expected final value, in order to prevent divergence of the iterative solver process.

2.2. Validation study

To make a proper selection of the final tool, first each prospective tool is tested and validated using a relevant reference case.

2.2.1. Reference case

The reference case used is the N250 Propeller, which is widely used as a reference case at the department of flight performance and propulsion. Examples are the works of van Arnhem [17] and Jansen [18]. This case contains experimental data of performed wind-tunnel experiments in low speed operating conditions and a complete geometry description. The conducted experiment tested the performance of the propeller for five different blade pitch angles ranging from 25 to 35 degrees, at a constant rpm of 10,000, for different advance ratios, ranging from 0.4 to 1.8. According to van Arnhem [17], these experiments were performed at standard sea level conditions. The N250 propeller consists of 6 unswept blades with a squared tip, which have varying local blade geometry defined at 23 local stations. The propeller geometry is given in figure 2.3. The reference data were interpolated with a cubic method to achieve the resolution required for the airfoil analysis in Xfoil and JavaFoil.

2.2.2. Results

Figures 2.4 and 2.5 give an overview of the results of the validation study. In general, all methods demonstrated that they are capable of a reasonable prediction of the propeller performance. The remainder of this section discusses the performance of each method separately.

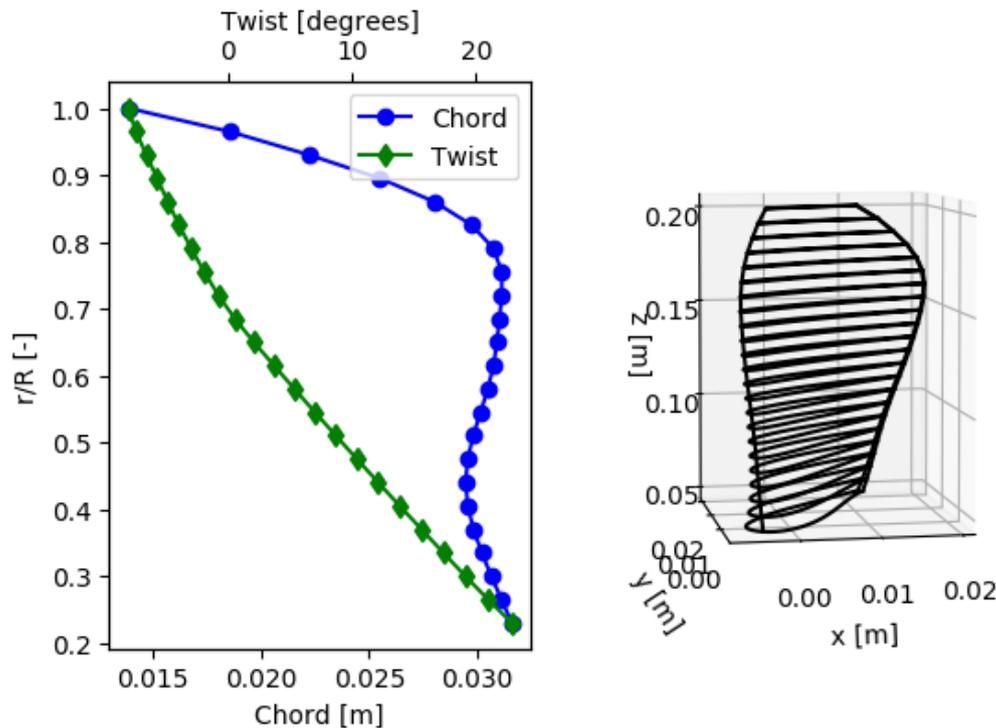


Figure 2.3: N250 Propeller geometry

Xrotor

When examining the Xrotor results in figures 2.4 and 2.5, it is concluded that Xrotor is able to predict performance to an acceptable level. In the high loading region (the first half of the operating range), Xrotor tends to over-predict both the thrust and power of the propeller. In the low loading region (the second half of the operating range), Xrotor slightly under-predicts the thrust and power, however, it performs better than the other methods considered here.

JavaProp

The implementation for JavaProp is the least complex, however this comes at a cost. Only very limited aerodynamic properties can be defined, which limits the accuracy of the results. When examining the JavaProp results in figures 2.4 and 2.5, it is concluded that, in case of thrust, JavaProp is the worst method in predicting it, except for a blade pitch of 35 degrees, in the high loading region. This is however, more of a coincidence, than it is a true reflection of the capabilities of JavaProp. At this point, the reference data is at a peak, while the trend of the JavaProp data demonstrates the same behavior as Xrotor and the Adkins & Liebeck scheme, which leads to the conclusion that JavaProp crosses over from over-prediction to under-prediction much earlier in the operating range, compared to the other two methods. The power trend is very similar to the the Adkins & Liebeck scheme and the reference data. It is the worst method for predicting both the high and low loading region.

Adkins & Liebeck scheme

When examining the Adkins & Liebeck scheme results in figures 2.4 and 2.5, it is concluded that in the high loading region, this method performs better compared to Xrotor. However while Xrotor over-predicts both the thrust and power required, this scheme under-predicts the power required. Throughout the range, the Adkins & Liebeck scheme is able to predict the power required trend very well, but consequently under-predicts it. This has consequences for the design region, in that it is worse compared to Xrotor, but still better than JavaProp. It must be noted that, for the Adkins & Liebeck scheme, the number of sections limited, while Xrotor has interpolation functionality, that allows for improving the distribution of sections and therefore producing better results.

2.3. Low-fidelity performance analysis method selection

A single low-fidelity propeller performance analysis method is selected, such that it can be implemented in an optimization routine. This selection is based on how well each method performs when considering two

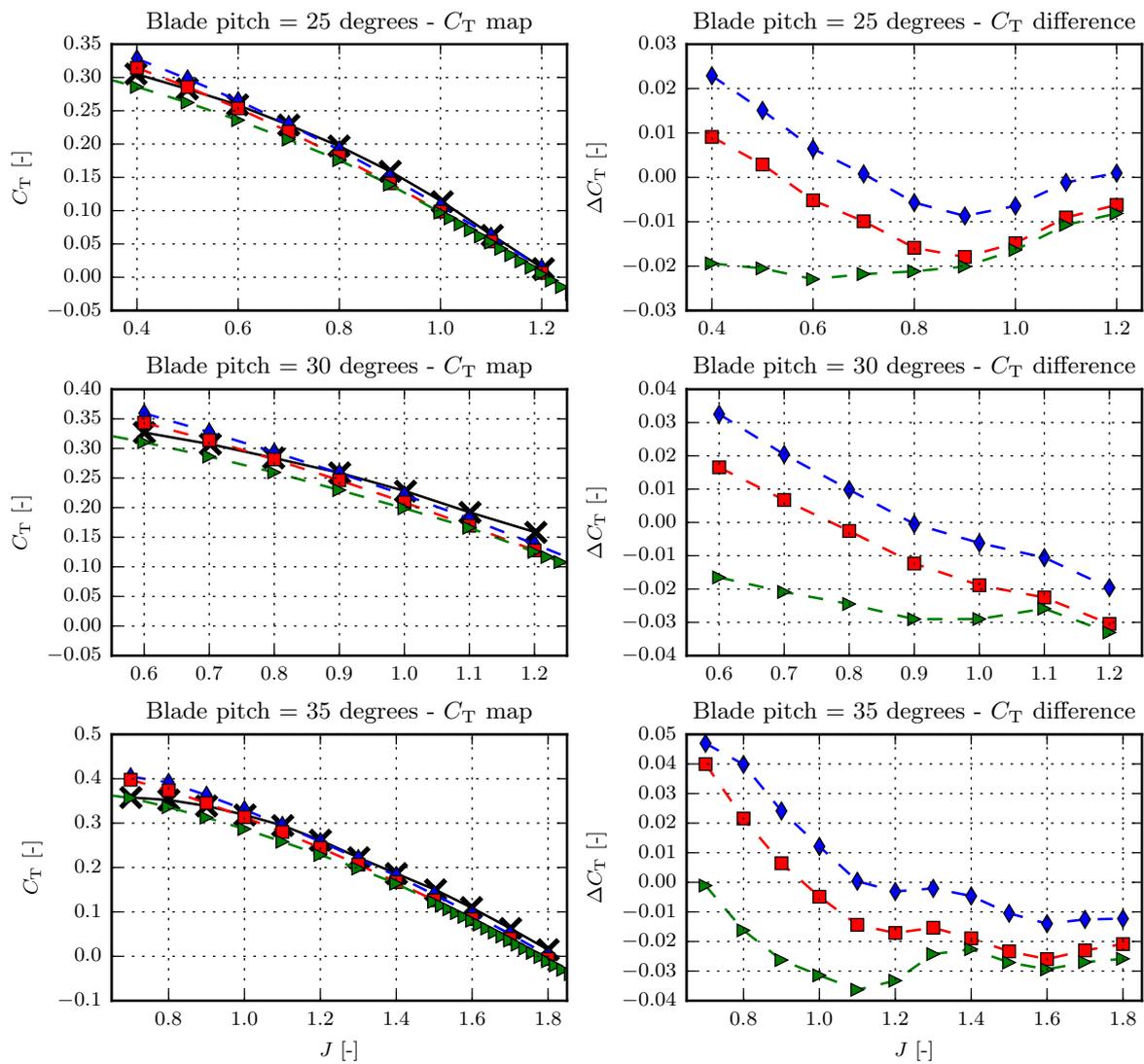


Figure 2.4: N250 Propeller thrust coefficient vs. advance ratio for different methods, at three different blade pitch angles. Left side shows the actual values and the right side shows the difference between the methods and the reference data. Black crosses =reference case, blue diamonds = Xrotor, red squares = Adkins & Liebeck scheme, green triangles = JavaProp

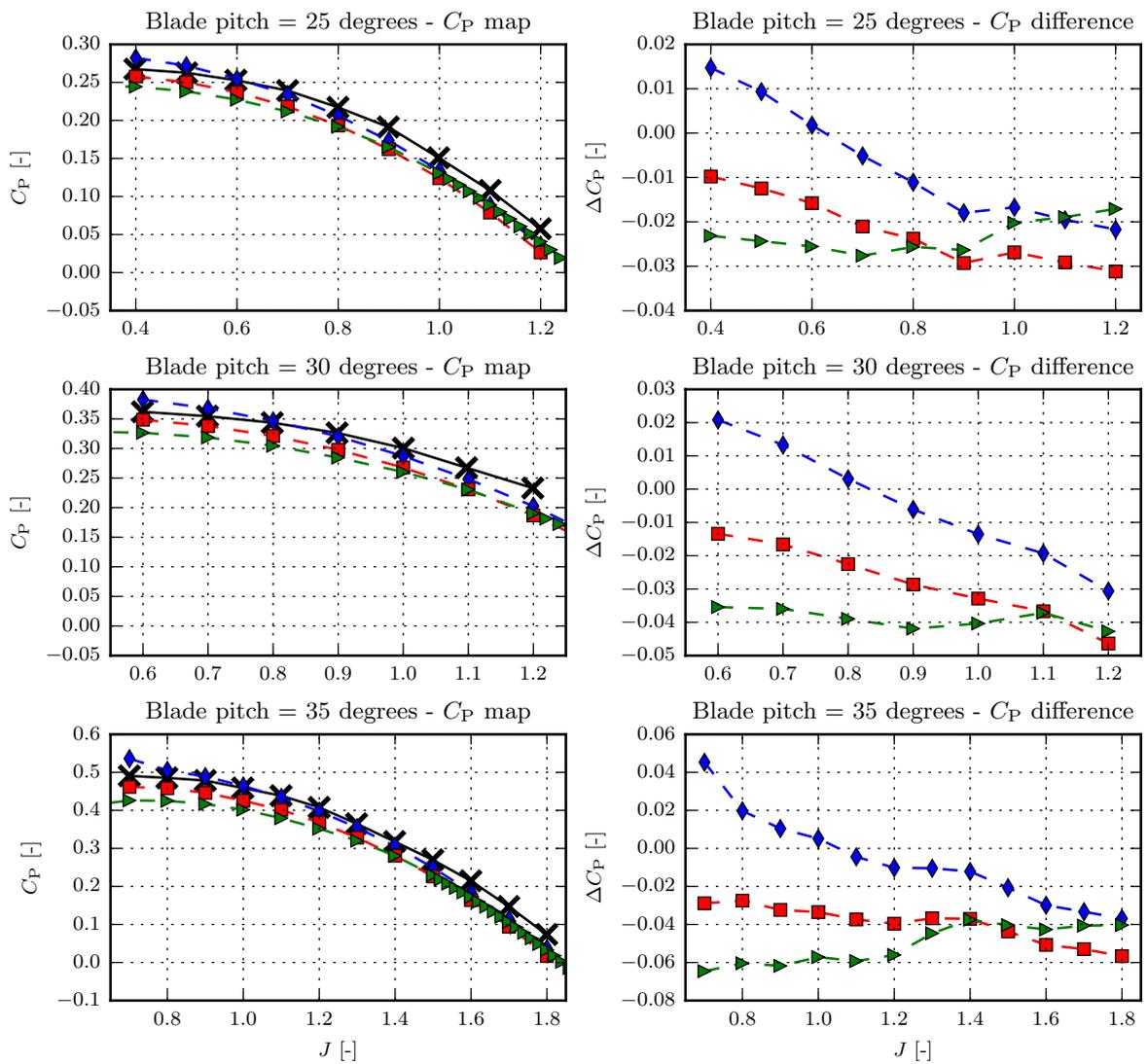


Figure 2.5: N250 Propeller power coefficient vs. advance ratio for different methods, at three different blade pitch angles. Left side shows the actual values and the right side shows the difference between the methods and the reference data. Black crosses =reference case, blue diamonds = Xrotor, red squares = Adkins & Liebeck scheme, green triangles = JavaProp

main criteria; overall performance in the validation study and stability.

Xrotor performs best in terms of overall performance in the validation study. The results of the validation study show a clear trend that for the majority of the operating range, and for multiple blade pitch angles, the error for Xrotor is smallest. The other methods perform better in specific areas, such as the power curve trend, but this not enough to disqualify Xrotor. The Adkins & Liebeck scheme is a close second. The biggest difference between Xrotor and this method is in the bigger error in the low loading region. JavaProp systematically under-predicts the performance and is only at the same level in the low loading region with the other two methods, but is not significantly better.

JavaProp performs best in terms of stability. The implementation of JavaProp is less complex compared to the other two methods. JavaFoil uses less sophisticated boundary layer models than Xfoil, which make the results less accurate, but it is a more stable analysis. Xrotor is the second most stable method. While its setup is the most complex of the three, if the implementation is done correctly and it has been proven to work, it is a reliable method. The Adkins & Liebeck scheme is the least reliable. The convergence issues in off-design conditions, means that its iterative process must be altered for every design point.

Based on the arguments made here, it is concluded that Xrotor is the best low-fidelity propeller performance analysis method for the purpose of this research. It has performed best in the validation study, and it is assumed that the implementation demonstrated here is done carefully and correctly, such that it is reliable enough.

3

High-fidelity propeller performance analysis method

The low-fidelity propeller performance analysis method is implemented so that it is possible to have rapid propeller performance analysis within an optimization routine, with limited available computational resources. However, the downside of using a low-fidelity method is that it will give less accurate results. It is therefore useful to implement a high-fidelity propeller performance analysis method, so that it is possible to verify results which are generated with the low-fidelity method. For this purpose, a CFD analysis method is implemented using an open-source CFD solver SU2 [19]. Once implemented, the CFD analysis method itself must be validated for it can be used to verify other parts of this thesis. The aim of this chapter is to present the setup and the validation of the CFD analysis method.

3.1. Setup of CFD analysis method

This section describes the setup of the propeller CFD method. The main focus of this section is on the pre-processing and processing stage, specifically the setup of the mesh generator and the flow solver.

3.1.1. Mesh generator

In order to assess the fluid-propeller interaction through a CFD analysis, it is required to discretize a given volume that surrounds a propeller. Since the geometry is axisymmetric and the wake is cyclic with the number of blades [20], it is convenient to use a periodic mesh and analyze a single blade only. The overall shape of a periodic mesh for this case, is typically characterized as a wedge, and an example is given in figure 3.1. The dimensions of the wedge are characterized by the angle of the wedge, the radius of the wedge and the distance at which the inlet and outlet are positioned. The angle of the wedge is inversely proportional to the number of blades, while the other characteristics must be selected with care. It is necessary to position the far-field boundaries at a sufficient distance from the blade, to ensure that it is safe to assume that at the far-field boundaries, free-stream conditions are applicable and that they do not influence the flow in the region near the blade.

The setup of the mesh generator is outlined in this section. First the motivations for selecting a specific mesh generator are discussed. When the motivations are clear, the procedure for generating a mesh is given.

Motivation for selected mesh generator

As there are different mesh generators available, it is required to select a mesh generator that is most suited for this particular research. First, the mesh generator must be reliable. Based on the capabilities of the parametrization tool and the scope of this research, it is to be expected that the mesh generator must be able to handle complex propeller geometries. If a prospective mesh geometry could not generate a mesh for a specific case, then it would not be possible to complete this research. Second, the mesh generator must be able to produce a mesh in the correct format, which is dictated by the solver SU2. While this requirement is vital for the setup of the CFD method, it is not a hard requirement for the mesh generator, providing there is the possibility of creating a mesh in the correct format through intermediary steps. Finally, it is required that the mesh generator must have advanced refinement capabilities, in order to allow for targeted refinement of important areas. This minimizes the number of cells, which is beneficiary for computational cost.

Based on these requirements and initial testing of a number of mesh generators, it is concluded that Numeca Autogrid5 [21] is most applicable in the context of this research. This is a dedicated meshing tool for

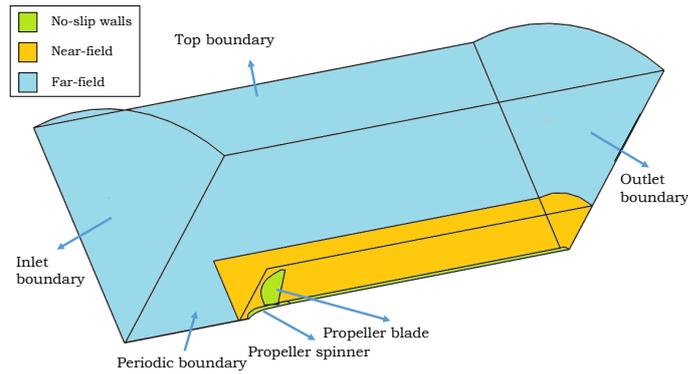


Figure 3.1: Typical shape for a periodic propeller mesh.

turbo-machinery applications and produces structured, periodic meshes. While most of its functionality is focused on turbo-machinery applications, it is also possible to produce a mesh for propellers. The main advantage of using Autogrid5, is that the meshing process is an automated process. This means that in order to produce a mesh, it is only required to specify propeller geometry, overall domain size and mesh refinement, after which the tool will generate an appropriate periodic structured mesh. The only issue with using Autogrid5 is that it generates a multi-block mesh, intended for a specific Numeca solver. In order for it to be used with the selected solver in this research, it is required to reduce the multi-block mesh to a single block mesh, identify boundary faces and convert to the correct format. Which is achieved with a secondary tool.

Mesh definition

Autogrid5 is used to define and create an initial mesh. This requires a manual procedure, by operating the graphical user interface of Autogrid5. Each step is discussed in the remainder of this subsection. It is important to note that a portion of the settings mentioned here will be kept constant throughout this research, while the remainder is variable for now, but will be determined by performing a mesh dependency study later.

The example domain in figure 3.1 is a simple straight wedge, but this is not always the case and depends on the type of grid. A simple straight wedge is considered to be appropriate for an unstructured grid, however, this is not the case for a structured grid. In order to increase the possibility of successful solver convergence, structured grids are generally aligned with the expected flow direction. An example of a structured grid that is aligned with the flow is given in figure 3.3.

Input geometry In the first step it is required to define the input geometry, which in this case consists of a hub geometry and a blade geometry. While Autogrid5 allows for different ways of defining the input geometry, the preferred method in this case is by creating and supplying a “geomTurbo” file. This is a dedicated file format in which the geometry is defined as a set of curves; each curve is then described by a discrete set of coordinates.

The blade is modeled by defining a variable number of airfoil sections, and each section is made up out of two curves describing the suction and pressure side. The concept of this method is straightforward, however in order to model the geometry with a certain degree of accuracy, it is necessary to choose a large number of airfoil sections and a dense set of coordinates.

The hub is assumed to be axisymmetric and modeled by a single curve describing the outline. There is one critical requirement for the hub curve and it is defined by the solver. In the case of a wedge shaped periodic mesh, the two periodic mesh faces cannot share an edge, which would be the case in the region upstream of the hub. This means that instead of an edge, an additional bottom wedge boundary is required, as shown in figure 3.2. Therefore, the hub leading point radius must always be larger than zero and thus must be selected with care. The choice for hub leading point radius is based on finding a balance between two considerations. First, selecting a value closest to zero, for geometric accuracy. Second, in order to have a fine distribution of cells in a structured grid at the hub and blade, the radial grid spacing upstream of the hub will be very dense, as shown in figure 3.2. Based on these considerations, the hub leading point radius is fixed throughout this research at 10% of the spinner radius.

Row wizard Once the input geometry is defined, the meshing process is started with the “Row Wizard” of Autogrid5. The “Row Wizard” is a useful feature that allows for rapid generation of an initial mesh, which can

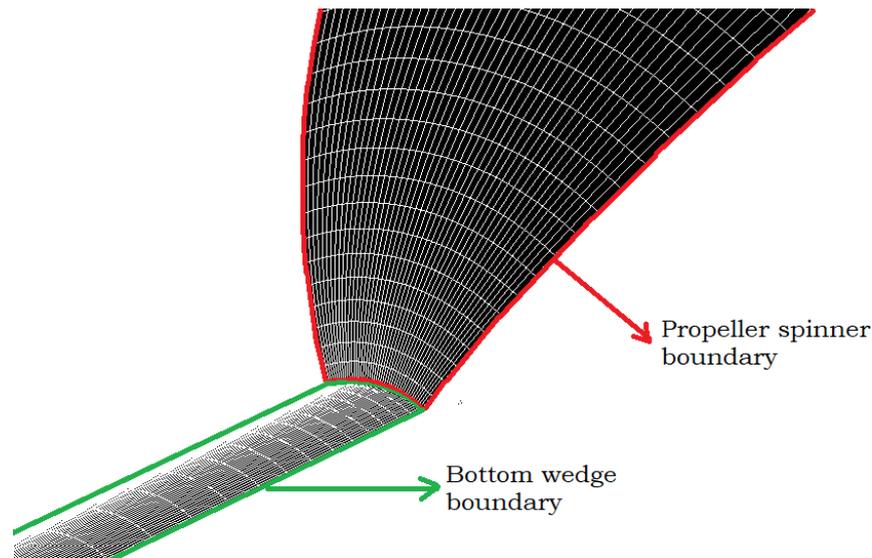


Figure 3.2: Example of radial grid spacing at connection of the upstream bottom wedge boundary (green) and the spinner wall boundary (red). With a structured grid, the fine mesh grid at the spinner must transition to the bottom wedge boundary, thereby squeezing the radial grid spacing.

be refined at a later stage with more advanced options. Below follows a list of “Row Wizard” settings and how they are treated throughout this research:

- *Row type*: Autogrid5 has a number of pre-configured blade types, mainly focusing on turbo-machinery applications. For meshing a propeller blade, three row types are applicable; the “Wind turbine”, the “Marine propeller” and the “Contra rotating fan”. Initial testing revealed that the “Contra rotating fan” type was the most reliable and generated better distributed grids. Based on this, the “Contra rotating fan” type will be used throughout this research.
- *Periodicity and rotation speed*: This is equal to the number of blades, which is dependent on the propeller geometry. While the exact rotation speed at this stage is not important, it is required to at least reflect the rotation direction in the rotation speed. If the rotation direction is not correct, the Autogrid5 will not align the structured grid with the flow. For counter-clockwise rotation (according to the same definition as used for the parametrization tool) the rotation speed must be negative and for clockwise rotation positive.
- *Blade channel control*: At this stage it is required to define the basic dimensions of the blade channel. This includes the position of the inlet and outlet with respect to the blade origin. Stokkermans [20] recommends that the inlet is positioned at 5 times the propeller radius from the blade origin and the outlet is positioned at 10 times the propeller radius from the blade origin. In a turbo-machinery case the blade channel is defined from the hub up to the shroud. However, in the propeller case, the shroud is considered to be artificial, because at a later stage a far-field channel will be added. However, because of this turbo-machinery background, Autogrid5 requires that shroud radius is smaller than the blade radius. This means that the shroud radius must be chosen as close to the blade radius as Autogrid5 allows. Initial testing revealed that a non-dimensional shroud radius of 0.99995 is the optimum setting.
- *Blade channel grid points*: Autogrid5 allows for defining the first cell height, the number of grid points and portion of the spanwise cells that are constant in height. For example, if you specify 50 cells in the blade channel and the portion of constant cells is 40%, then the first 15 cells grow proportionally from the first cell height at the spinner, the next 20 have the same height and the final 15 cells shrink proportionally to the prescribed cell height at the top of the blade channel. The height of the first cell is dependent on the selection of a desired y^+ value, and is applied to the hub, the artificial shroud, blade sides and blade tip. The height of the spanwise cells grows proportionally, however it is not clear how each of these variables influence the growth rate. In order to have an appropriate distribution of cells, while varying the number of grid points, it is decided to keep the portion of constant cells constant at 66%.
- *Far-field channel control*: The three applicable row types are unique in Autogrid5 because they allow for defining a far-field channel on top of the blade channel. Stokkermans [20] recommends that the radius of the far-field upper boundary should be five times the propeller radius, which translates into Autogrid5 as a non-dimensional far-field thickness of four. Furthermore, the same blade channel grid

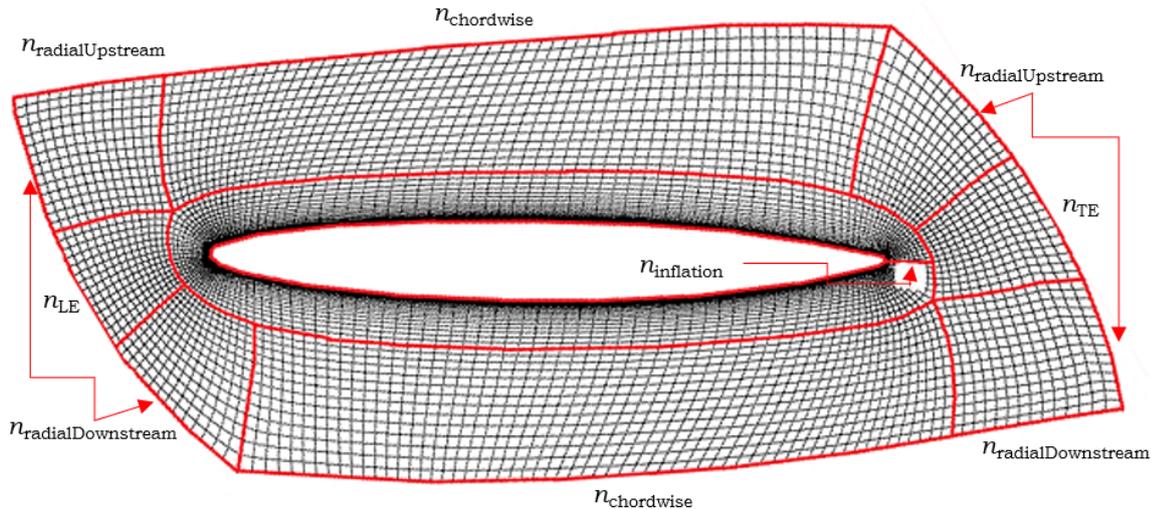


Figure 3.3: Blade-to-blade grid points definition in the Autogrid5 meshing tool.

point settings apply here as well. This means that it is possible to vary the number of spanwise grid points in the far-field, as well as the portion of constant cell that are constant in height. While the volume of the far-field is considerably larger than the blade channel, it is not an area that requires mesh much refinement, except near the artificial shroud. Based on this, the portion of constant cells is kept constant throughout this research, at 33%.

Blade-to-blade definition Once the row wizard is finished, it has generated the definition of an initial mesh, which requires additional refinement in certain areas. One of these areas is what Autogrid5 calls the “blade-to-blade” definition. Here it is possible to define the radial, streamwise and the inflation grid in the near-blade region. Figure 3.3 shows how the blade-to-blade grid points are defined in Autogrid5.

Streamwise grid definition Another area which requires additional refinement is the streamwise grid distribution. Autogrid5 divides the grid into a five of streamwise sections. The streamwise distribution of these sections is dependent on the selected row type. In the propeller case, the default streamwise distribution is corrected such that each streamwise hub discontinuity is accurately modeled in the mesh. Furthermore, in order to refine wake region behind the propeller, the final wake streamwise section is split up into two sections. Autogrid5 streamwise grid refinement is done by defining the number of streamwise grid points for every section, excluding the blade section which has already been defined in the blade-to-blade definition. The streamwise cell width distribution is automatically determined by the mesh generator. It starts at the blade section and increases the cell width by an unspecified growth rate which is influenced by the number of specified grid points of each section.

3.1.2. Solver setup

The preferred CFD solver is SU2, which is an open-source suite for multi-physics simulation and design [19]. While development continues, the majority of its functionality has been validated and its capabilities are continuously extended. SU2 does not have a graphical interface, but is operated through the command line and by supplying a mesh file and a configuration file. The mesh file is generated by the mesh generator described in section 3.1.1, while the configuration file is a dedicated format describing the solver physics setup, boundary conditions and convergence criteria. The original reason for selecting SU2 was that, due to simplicity of the operating process, implementation into an optimization routine would be relatively straightforward, which was part of the original research plan. However, in order to achieve the current research objective this specific characteristic is no longer required, since the high-fidelity analysis is no longer to be performed inside the optimization loop.

SU2 employs Reynolds-Averaged Navier-Stokes (RANS) equations for viscous, compressible simulations [19]. In order to reduce computational cost, only a steady-state simulation is performed. The only reason for opting for an unsteady simulation would be that unsteady behaviour is expected due to external influences such as non-uniform, non-axisymmetric inflow (i.e. gusts, turbulent inflow, free-stream flow at an angle of attack, etc) or multiple objects that have a have different dynamics (i.e. a rotating propeller blade with a

stationary stator vane downstream of the propeller). These are all not present for an isolated propeller case, so any unsteady effect will be very limited.

Turbulence model

In order to reduce computational cost further, in the propeller case it has been accepted practice to perform analysis with a fully turbulent boundary layer in mind, using an appropriate turbulence model [20]. SU2 offers two turbulence models, the Spalart-Almaras (SA) model and the Shear-Stress-Transport (SST) model. While both are appropriate in the for propellers, they behave differently and also influence the setup differently. The one-equation SA model works by resolving the entire boundary layer, including the viscous sub-layer, requiring a very refined grid near walls [20]. The Menter SST turbulence model is a two-equation model for the turbulent kinetic energy and specific dissipation [19]. Throughout this research, only the Spalart-Almaras model is used. As mentioned before, this influences the grid refinement near the walls. A y^+ value between 1 and 5 is deemed appropriate in this case, according to van Arnhem [17]. In order to solve the turbulence model, SU2 employs a scalar upwind numerical method. The spatial order of integration of this method is first order, based on default SU2 settings.

Boundary and free-stream conditions

In the propeller case, the meshing tool generates a single block, wedge-shaped periodic domain. In order to perform an analysis with this domain, it is required to define free-stream and boundary conditions. In the case of a compressible simulation, SU2 dictates that the following flow characteristics need to be known:

- Free-stream Mach number [-]
- Angle of attack and side-slip angle of the flow [degrees]
- Free-stream temperature [Kelvin]
- Free-stream Reynolds number [-]
- Characteristic Reynolds length [m]

SU2 is a generic CFD solver and does not take specific applications into account when defining parameters. In this case, a free-stream Reynolds number would be difficult to quantify for a propeller. However, these generic flow characteristics are only used to determine other free-stream condition characteristics (i.e. density, static pressure, etc). With the characteristic Reynolds length parameter, it is possible to have control over the remaining free-stream condition parameters. So any arbitrary value is correct, as long as it is consistently used throughout the CFD analysis. Therefore, a characteristic Reynolds length of one meter is assumed. Throughout this research, only uniform flow is used. The angle of attack and side-slip angle of the flow are then only based on the definitions of the reference frame used in both the mesh generator and the solver. The remaining flow conditions are used internally to determine other flow characteristics, such as free-stream static pressure, air density, velocity and more. It is assumed that the available information about the free-stream conditions is limited to free-stream velocity, air density and static pressure. These must be used to determine the SU2-specific free-stream conditions. First, the free-stream temperature is found by applying the well-known ideal gas law of equation 3.1.

$$T_{\infty} = \frac{p_{\infty}}{\rho_{\infty} R} \quad (3.1)$$

Where R is the specific gas constant. In order to determine the free-stream Mach number, the speed of sound is calculated using the Newton-Laplace definition of equation 3.2.

$$M_{\infty} = \frac{V_{\infty}}{a} = \frac{V_{\infty}}{\sqrt{\gamma \cdot \frac{p_{\infty}}{\rho_{\infty}}}} \quad (3.2)$$

Where γ is the specific heat ratio. In order to determine the free-stream Reynolds number, it is first required to determine the dynamic viscosity. This can be acquired by applying Sutherland's equation. The definition of the free-stream Reynolds number follows in equation 3.3

$$Re_{\infty} = \frac{V_{\infty} \rho_{\infty} L}{\mu} = \frac{V_{\infty} \rho_{\infty} L}{\mu_0 \cdot \frac{T_0 + C}{T + C} \cdot \left(\frac{T}{T_0}\right)^{1.5}} \quad (3.3)$$

Where C , μ_0 and T_0 are Sutherlands Constant and standard sea-level conditions for dynamic viscosity and temperature, respectively. SU2 uses the same procedure in reverse to determine the remaining flow characteristics. Table 3.1 gives an overview of the default SU2 values.

The solver simulates propeller rotation by the rotating frame principle. According to [22], in this case the rotation of the coordinate system is included in the equations of motion describing the flow. In the real

Table 3.1: Overview of default SU2 flow determining characteristics that are used to determine SU2-specific free-stream conditions.

Variable	Unit	Value
Specific gas constant	$\text{J kg}^{-1} \text{K}^{-1}$	287.058
Specific heat ratio	-	1.4
Sutherland's constant	K	110.4
Dynamic viscosity at standard sea-level conditions	$\text{kg m}^{-1} \text{s}^{-1}$	1.716×10^{-5}
Temperature at standard sea-level conditions	K	273.15

Table 3.2: Classification of boundary types.

SU2 boundary type	Mesh boundary faces
Wall - constant heatflux	Spinner, blade, hub
Periodic	Periodic 1 & 2
Symmetry	Wedge
Far-field	Inlet, Top, Outlet

world, the propeller is rotating in a certain direction while the surrounding volume it acts on is stationary. In the rotating frame case, the domain is rotating itself, while the propeller is stationary w.r.t. the domain. To include rotation in SU2, it is required to specify the rotation speed of the domain, in radians per second.

Finally, it is required to define the boundary conditions. The definition for the mathematical model is crucial, in order to get converged and, more importantly, meaningful results. Just like other more popular solvers such as Ansys Fluent [22], SU2 comes with a pre-defined set of different types of boundaries, and all that is required is to match each boundary face of the mesh with the appropriate SU2 boundary type. In the mesh definition in section 3.1.1, the boundary faces of the mesh are identified. Table 3.2 shows how each mesh boundary face is classified as an SU2 boundary type. In a viscous compressible case, there are two no-slip wall boundary types available. One where the heat flux at the wall cells is constant at a specified value and the temperature is non-uniform. The other achieves the opposite behavior; a specified constant temperature across all wall cells while the heat flux is allowed to be varied. In this case, SU2 recommends a constant heat flux wall type, where the constant heat flux is equal to zero [19]. The boundary type of the hub and the wedge depends on the expected flow direction in the wake and upstream of the propeller, respectively. Since they are not far enough away from the blade they cannot be classified as far-field boundaries, which means that the only other option is to classify them as a slip (symmetry) or a no-slip (wall, constant heat flux). In order to classify a face as a slip wall, it must fulfill a requirement that states that the expected flow on the surface has no normal component. The argument made is that the wedge face meets this requirement, since it is upstream of the propeller. However the hub face does not meet this requirement and thus is classified as a constant heat flux wall, because the flow is in the wake of the blade.

3.2. Validation of CFD method

In the previous section, the setup of the CFD method is given. Before the method is applied to a particular case, it must be validated first. This approach will ensure that any interpretation of results, produced by the method, is relevant. First an appropriate validation case is selected, of which the propeller geometry and detailed experimental performance data is available. Then, a mesh sensitivity is performed using the given propeller geometry. By performing a sensitivity study, it is possible to determine the influence of mesh refinement on the results. This is then used to make an educated decision on a specific mesh refinement, that is based on a compromise between accuracy and computational cost.

3.2.1. Mesh sensitivity study setup

Ideally, it is favorable to have the finest mesh for better accuracy, while having the least amount of cells for lower computation time. Since these are contradicting requirements, mesh refinement must be carefully selected. Defining the mesh fineness is not an exact science and depends on the application, the meshing tool capabilities and the used solver. Because of this, it is considered best practice to perform a mesh refinement sensitivity study, in order to assess the influence of mesh refinement on the accuracy of the result.

The APIAN propeller has been selected as a validation case. According to Stokkermans [20], the APIAN propeller was developed in an joined European project [23], which investigated acoustic and aerodynamic installation effects of the propeller. This is considered to be a relevant case because, the complexity of the blade geometry is in line with the capabilities of the parametrization tool. Another reason for selecting this

Table 3.3: Overview of used operating conditions throughout validation study.

J [-]	1.05	1.40	1.75
V_∞ [m s ⁻¹]	60.0667	60.0	60.0333
P_∞ [Pa]	101964.3	101946.7	101919.3
ρ_∞ [kg m ⁻³]	1.1803	1.1817	1.1810
Ω [rad s ⁻¹]	-707.8846	-530.0390	-425.2041
M_∞ [-]	0.1727	0.1726	0.1727
T_∞ [K]	300.936	300.545	300.633
Re [-]	3.83×10^6	3.846×10^6	3.83×10^6

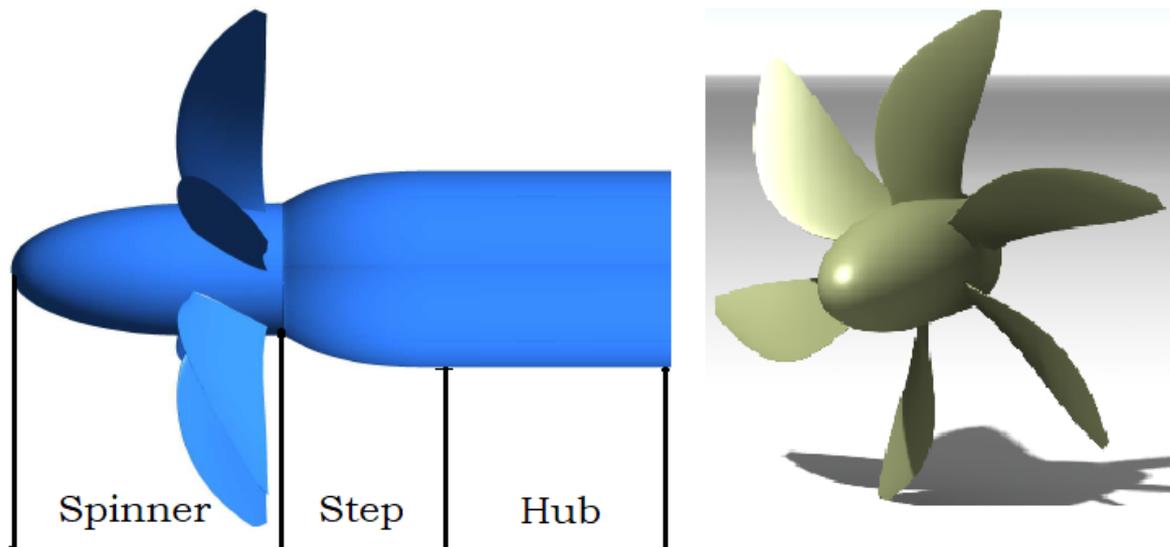


Figure 3.4: CAD model of the APIAN propeller.

validation case is that there is a diverse range of reference data at a wide range of operating conditions available. This allows for assessing the validity based on multiple flow characteristics, as well as the CFD method's sensitivity due to varying operating conditions. Three operating points will be used that describe the operating range. Table 3.3 gives an overview of the used free-stream conditions in SU2, which are imposed by wind tunnel conditions.

The APIAN propeller is a six-bladed propeller with a diameter of 0.508 meter and a collective blade pitch angle of 40.4 degrees. It also has both sweep and lean geometry characteristics, see figure 3.4. The spinner geometry consists of a forward typical spinner geometry and a curved step section in the wake. This layout has been chosen to closely match the wind tunnel lay-out used to determine its performance. In order to generate a mesh based on this geometry additional steps are required. By using a dedicated CAD tool CATIA [24], it is possible to create slices from the original blade geometry and extract sets of coordinates that describe each slice. In order to study the sensitivity of the CFD method only, a mesh geometry input file is created based on the extracted set of coordinates.

In order to study the sensitivity of mesh refinement, it is necessary to use a wide range of refinement and multiple meshes. In this case, five different levels of mesh refinement are used, which are classified as ranging from extra coarse to extra fine. These meshes are created using the mesh generator of subsection 3.1.1 and require input, that need to be varied for varying mesh refinement. One crucial mesh variable is the width of the first cell at the wall, which is dependent on free-stream conditions and the desired y^+ value. Since free-stream conditions do not vary significantly, the first cell width is constant throughout the validation case at 5×10^{-6} meters. The remaining variables are varied and given in table 3.4. As previously mentioned in subsection 3.1.1, the consequence of creating a structured mesh is that refinement of one particular area will influence the number of cells in another less important area as well. A careful refinement strategy is required that does not allow the total number of cells to grow exponentially. For this reason, a strategy is adopted where an initial coarsest mesh is created that has limited refinement of only the most important areas. Then, based on this initial mesh, for each subsequent finer mesh, the refinement of one specific area is improved.

Table 3.4: Overview of used mesh defining variables and resulting mesh size for mesh sensitivity study.

Mesh refinement	Extra coarse	Coarse	Moderate	Fine	Extra fine
<i>Spanwise variables:</i>					
• Blade channel grid points	75	100	100	110	120
• Far-field channel grid points	50	50	50	50	50
<i>Streamwise variables:</i>					
• Upstream grid points	25	25	50	50	50
• Spinner grid points	45	45	75	75	75
• Step grid points	25	25	40	40	60
• Hub grid points	125	125	200	200	230
<i>Blade-to-blade variables:</i>					
• LE and TE grid points	15	15	15	15	15
• Chordwise grid points	50	75	75	100	125
• Radial up- and downstream grid points	13	15	15	17	19
• Inflation grid points	35	35	35	35	35
Total number of cells	1,842,422	2,773,140	3,674,460	4,650,356	6,170,780

3.2.2. Mesh Sensitivity study results

This section gives the results of the sensitivity study and discusses whether this validates the CFD method. This is done based on flow characteristics that are known from the reference case. This includes top level propeller characteristics, pressure distributions at three radial sections, and two-dimensional wake flow fields. In general, all meshes used in the mesh sensitivity study converged well. The main residual related to air density, on average, reduced by 4-5 orders, although this required a considerable number of iterations. The maximum required number of iterations was 40,000 for the finer meshes, while the minimum was 27,500 iterations for the coarsest mesh. The results shown here are compared with the available performance data of the original experiment, which was performed in a wind tunnel [23]. These include overall propeller performance characteristics, blade pressure measurements and wake particle image velocimetry (PIV) measurements.

Overall performance results

Figure 3.5 shows how the top level propeller characteristics vary over the operating range. Additionally in figure 3.6, the mesh sensitivity of the thrust and torque coefficient are plotted. These results represent the total performance of the propeller, which includes both the contribution of the propeller blades and the spinner, which is in accordance with the setup of the original experiment.

The torque coefficient in figure 3.5 is well predicted over the entire operating range, for every mesh refinement. The deviation is minimal, while the trend for the entire operating range is that the deviation decreases minimally with increasing advance ratio. The thrust coefficient is less well predicted compared to the torque coefficient, independent of mesh refinement. This applies especially on the trend for the entire operating range. There is an increase in deviation for increasing advance ratio, which is not expected. The cause for the increase of deviation in thrust coefficient is traced back to the spinner thrust contribution. In the reference experiment, both the spinner and propeller blade thrust/drag contributions are taken into account. These individual contributions are also available in the CFD analysis and can be found in appendix A. When these are analyzed, it is concluded that, when isolated, the overall propeller blades thrust coefficient performance is improved and the gradient error is reduced to 3% from 5%. The torque is minimally affected.

The sensitivity of mesh refinement on propeller torque is clearly visible in figure 3.6. For $J = 1.05$ a clear converging trend is visible, while for increasing advance ratio, the trend is decreasingly convergent. However, when considering the value range for all three operating points, it is concluded that the torque converges quickly, with minimal change from 0.0001 at $J = 1.75$ and 0.001 at $J = 1.05$. The mesh sensitivity of the thrust coefficient in figure 3.6, compared to the mesh sensitivity of the torque coefficient is less clear. This is largely attributed to two of the used meshes; the coarse and fine meshes. It is clearly visible that, for increasing advance ratio these two meshes are increasingly off the convergence trend, compared to the other meshes. The value range for all three operating points is slightly bigger compared to the torque coefficient, but still acceptably small. The same plots are given for the individual spinner and propeller blades contributions in appendix A. The isolated spinner contribution shows that the source of the oscillating trend in the total thrust coefficient convergence trend in figure 3.6 is the result of the same behavior at the spinner. Another interesting feature is that for $J = 1.05$, the spinner is generating thrust, while for increasing advance ratio, the spinner is increasingly generating drag. The total thrust decreases with increasing advance ratio as expected, which means that the increasing drag for increasing advance ratio has a more significant effect at higher advance ra-

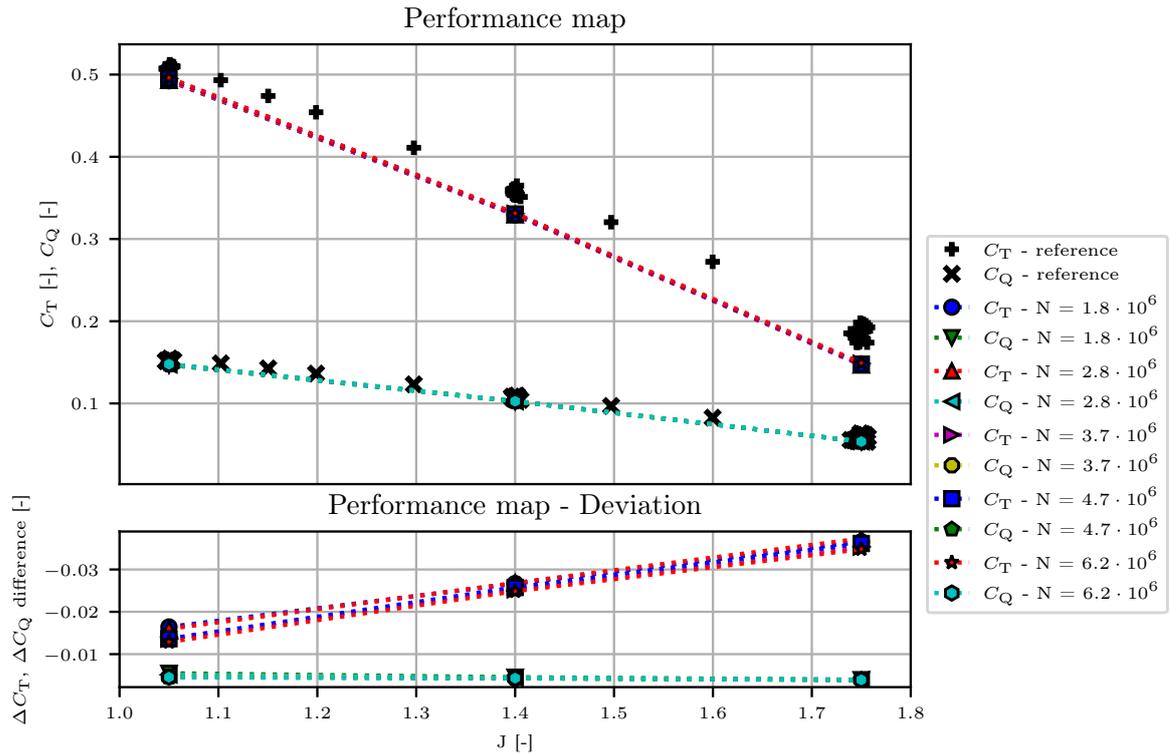


Figure 3.5: Propeller performance map for mesh sensitivity study. Upper plot shows the total thrust and torque coefficient for both the reference case and the different mesh refinements. Lower plot shows the difference between the reference and the different mesh refinements.

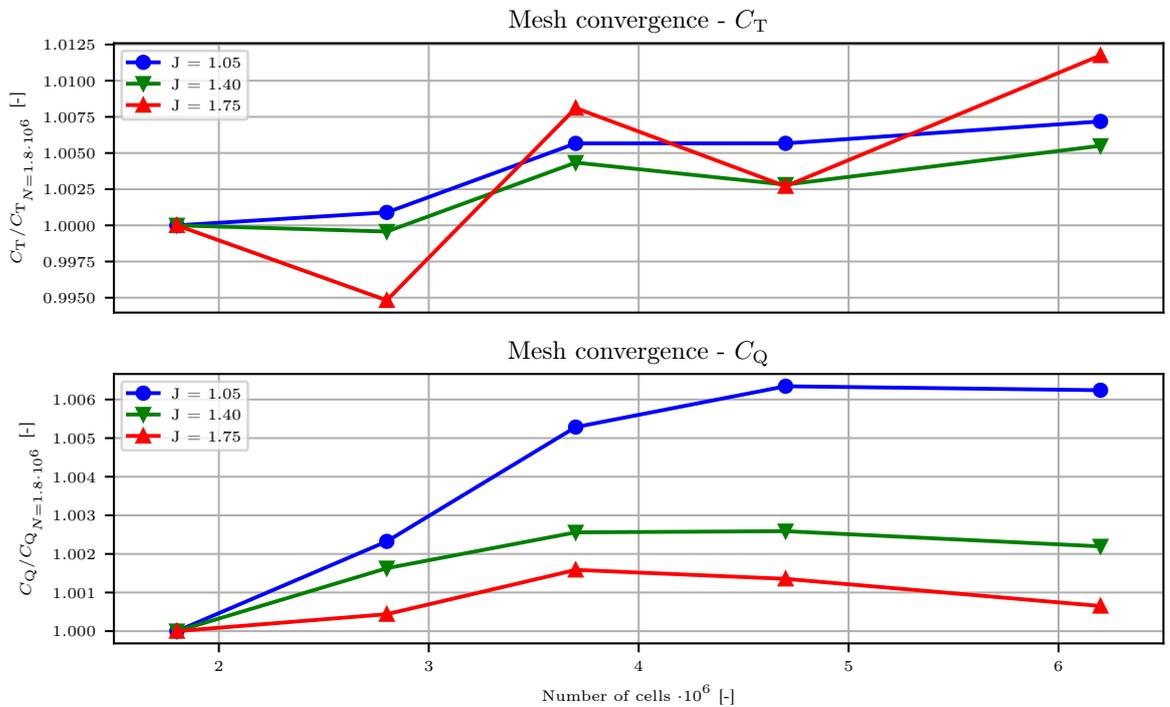


Figure 3.6: Propeller performance mesh convergence for sensitivity study. It shows in detail how the main propeller performance characteristics vary with mesh refinement.

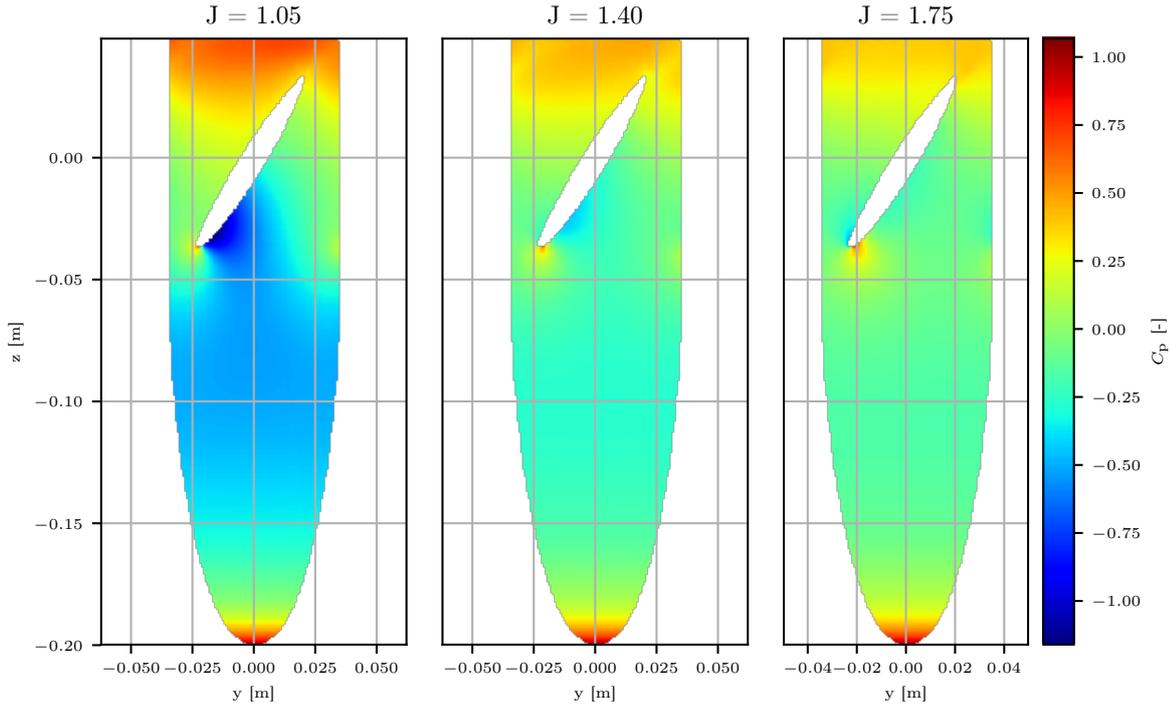


Figure 3.7: Spinner pressure distribution for varying advance ratio.

tios. This is the reason why the witnessed spinner thrust effect is damped at $J = 1.05$, but increasingly present at the remaining operating points.

It has been demonstrated that the cause of the significant increase of thrust coefficient deviation for increasing advance ratio, as well as the thrust coefficient oscillation in mesh sensitivity, can be traced back to the spinner thrust contribution. The reason why the spinner behaves as it does is cause for speculation and no definitive reason has been found. Figure 3.7 shows the pressure field for a sixty degree wedge section of the spinner, for the moderate mesh and the three operating points. It shows that the blade has a major impact on the pressure distribution on the spinner, both upstream and downstream. For $J = 1.05$ there is a significant negative pressure area, where the geometry is at a significant curvature, thus generating thrust. At $J = 1.40$, there is still a negative pressure distribution in the same area, but it is significantly weaker than at the high loading condition. The relatively high positive pressure areas on the nose of the spinner and the area around and aft of the blade are no longer compensated with a large negative pressure area, which means that the spinner starts to generate drag. Finally, for $J = 1.75$, the negative pressure area ahead of the blade has dissipated, which means that most of the spinner area is now generating drag. These effects are caused by the propeller, which leads to a contraction of the stream-tube into the propeller. This leads to a lower static pressure in this area.

One reason for the differences is that a fundamental difference exists between the wind tunnel blade-spinner junction geometry and the CFD blade-spinner juncture geometry, which are shown in figure 3.8. The wind tunnel blade has a so-called “shank” blade geometry where the blade base is not flushed with the spinner but “floats” on top of it and is only connected to the spinner at the propeller axis. This is useful for wind tunnel applications because a wide range of pitch angles can be studied, without having to make changes to the spinner. However for CFD applications, a shank blade geometry is very complex and flow structures in the blade-spinner juncture region cannot be accurately modeled. This is the reason for choosing a flush or “ideal” blade-spinner junction. The difference between the geometry blade root-spinner junction might have been the reason for the significant increase of local deviation for increasing advance ratio. Based on actuator disk theory, one could argue that a low pressure area ahead of the blade is expected. However, the counter-argument is made that the low pressure in the streamtube would only influence the spinner pressure distribution if the blade-root is in the spinner’s sphere of influence. It is generally known that one of the characteristics of a boundary layer is that it acts as a boundary between two flow structures. If the blade-root is not inside the spinner boundary layer, the argument is made that that the spinner pressure distribution is not affected quite so significantly as shown in figure 3.7. Unfortunately almost no research in literature has been found that investigated propeller blade-spinner juncture flow and no data from the original experiment

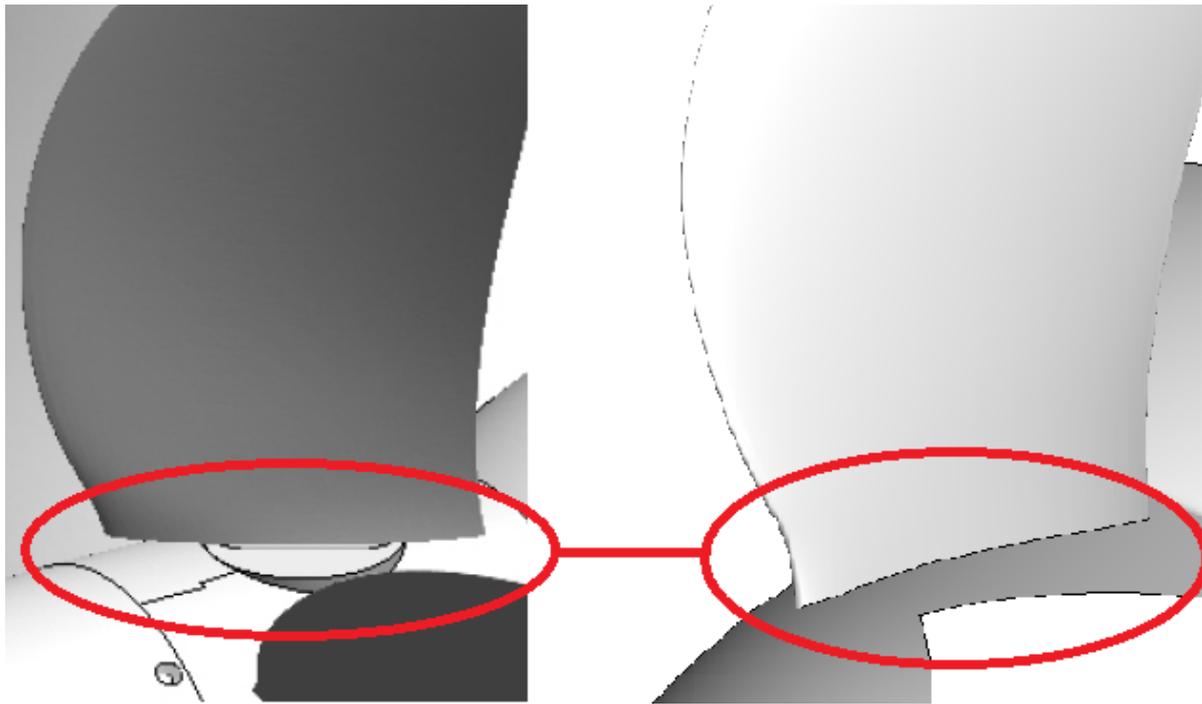


Figure 3.8: Left: Original wind tunnel spinner-blade juncture geometry [20]. Right: CFD spinner blade juncture geometry.

about the spinner pressure distribution is available. Only one study into this subject was found by Keith et al. [25], where it was concluded that the pressure distribution over the spinner is influenced by the blade-root, as long as the base of the blade is in the boundary layer of the spinner.

Pressure distribution

In the reference experiment, pressure measurements were performed, which allows for more detailed validation of how accurate the CFD method is able to predict the flow at the blade surface. The reference data was acquired using kulites integrated into the blade surface [5]. Pressure distribution data is available for all operating points and at three blade stations, at $r/R = 0.65, 0.75$ and at 0.85 of the propeller radius. Figure 3.9 shows how the CFD method compares with the reference case, as well as the sensitivity for $J = 1.05$ at $r/R = 0.65$. The pressure side is predicted very well and the difference between the different mesh refinements is negligible. The suction side is more interesting. What stands out is that there are two pressure peaks, one at the leading edge and one just aft of the leading edge. The first peak is caused by the geometric properties of the blade station, while the second is caused by the leading edge vortex. The first three reference data points are predicted very well by the CFD method, while the remaining aft data points are slightly off. When mesh refinement is considered on the suction side, it is clear that the most noticeable differences are in the region of the leading edge vortex peak. The extra coarse mesh is very peaky at the top of the pressure peak, which is due to a too coarse mesh. After the peak and the adverse pressure gradient region, there is a dip in the pressure. The reference data point in this region shows that for increasing refinement, the solution becomes more accurate. Similar plots and detailed findings for the remaining combinations of operating points and blade stations are presented in appendix A. In general, the findings for the pressure distribution at $J = 1.05$ and $r/R = 0.65$, are applicable to the remaining combinations.

Pressure distribution convergence In order to fully validate the pressure results, it is useful to check for mesh convergence. In order to achieve this, a method of representing each pressure distribution into a single value is required. This is achieved by determining the normal force coefficient, by integrating the difference between the suction and pressure side. According to Anderson [26]:

$$c_n = \frac{1}{c} \int_0^c (c_{p_{\text{suction}}} - c_{p_{\text{pressure}}}) dx \quad (3.4)$$

In order to limit the effect of the different levels of mesh refinement on the numerical integration, the 2nd order Simpson method is used. The mesh convergence for integrated normal coefficient is given in figure 3.10. A general trend is noticed where the normal coefficient range increases for increasing blade station,

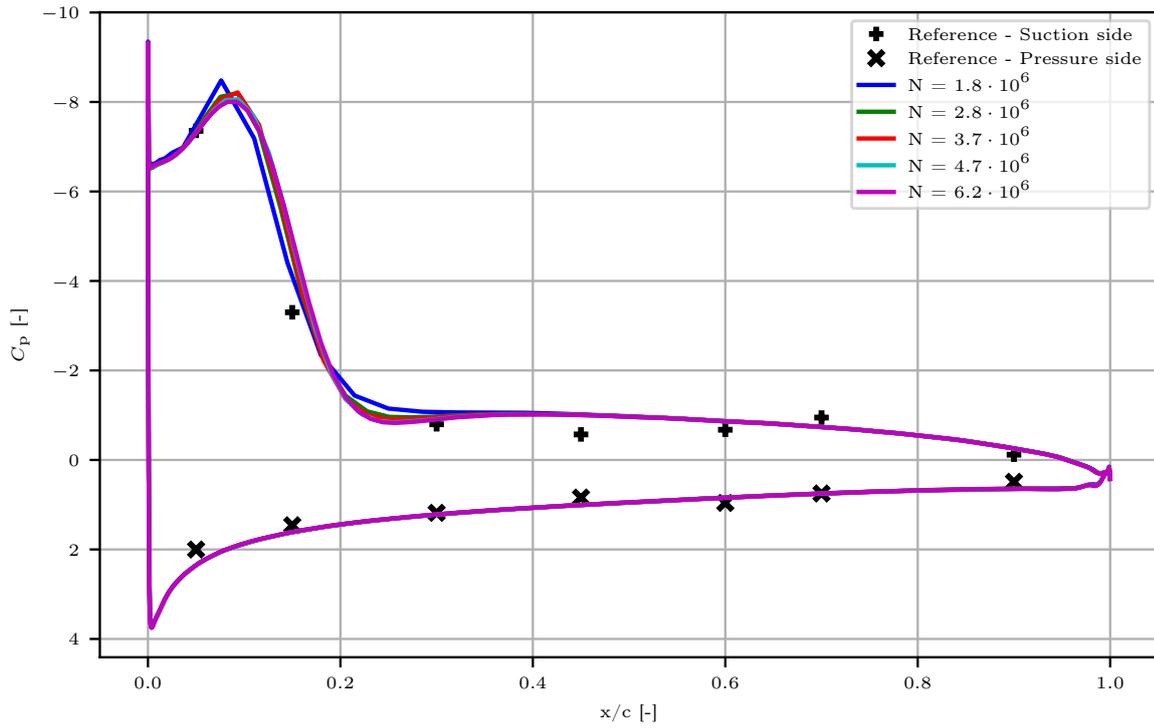


Figure 3.9: Blade pressure distribution for $J = 1.05$ at blade station $r/R = 0.65$.

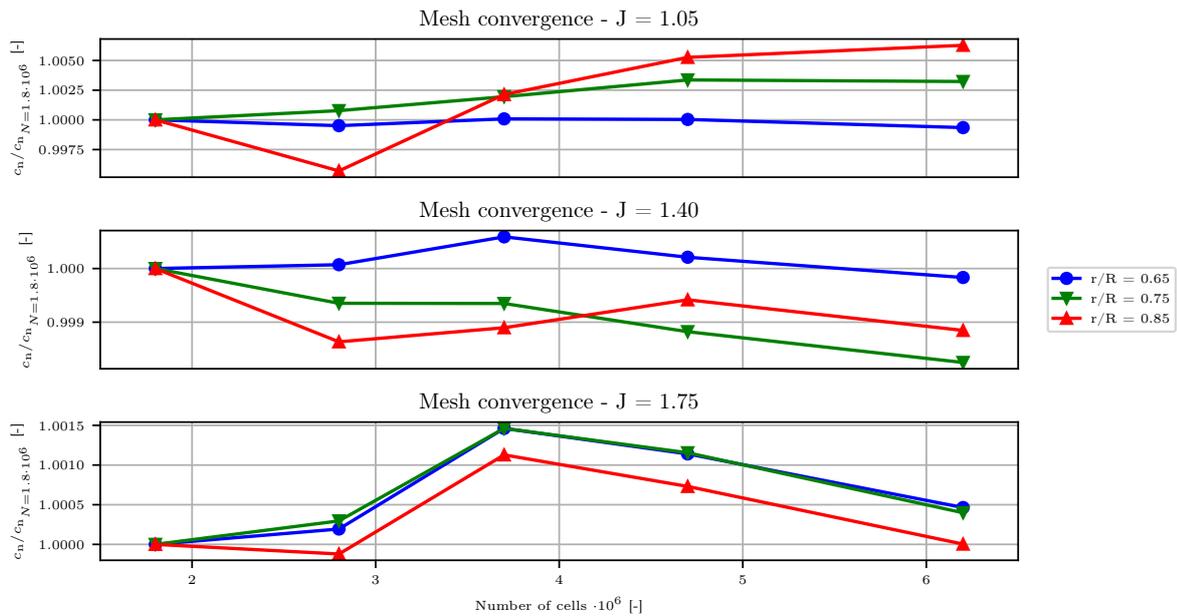


Figure 3.10: Integrated normal force coefficient from pressure distributions for varying mesh refinement.

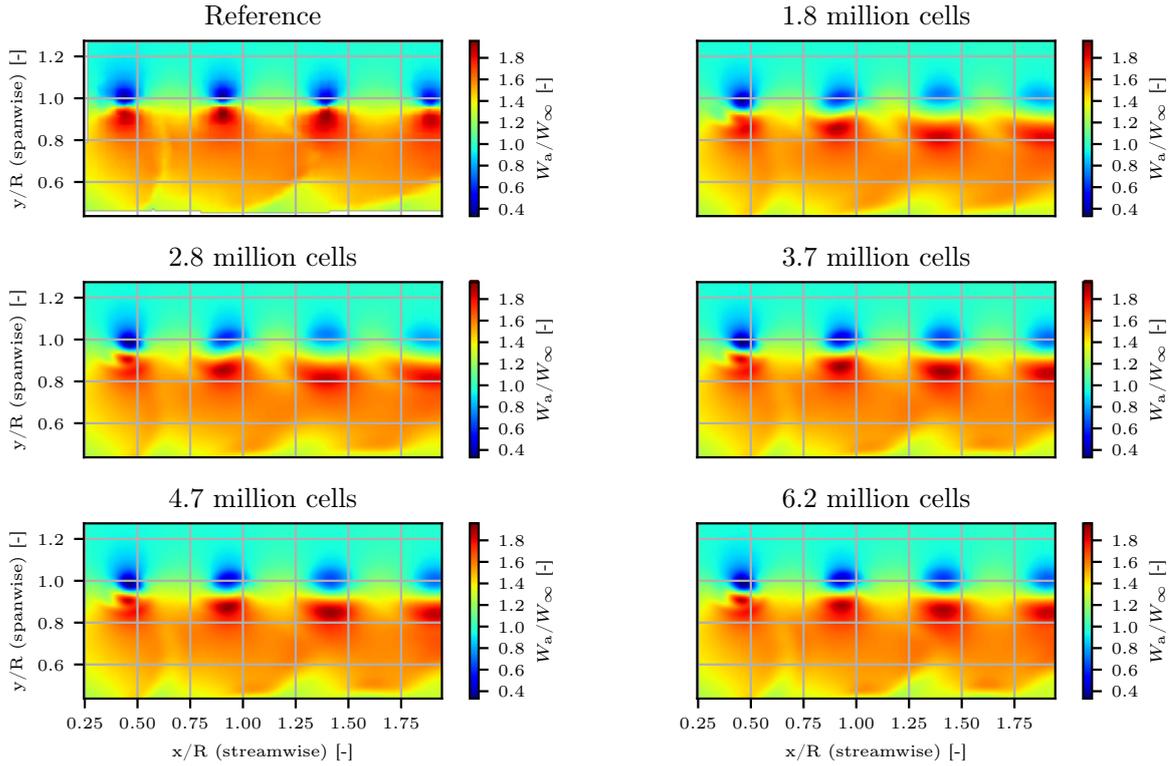


Figure 3.11: 2-Dimensional wake flow field - Non-dimensional axial velocity for $J = 1.05$.

while it decreases for increasing advance ratio. This is logical since the same behavior was noticed when examining the pressure distributions and has mainly to do with the presence, size and position of the vortex pressure peak and the reduced ability of the CFD method to be able to accurately predict it.

Two-dimensional velocity fields in the propeller slipstream

Sinnige et al. [5] investigated the velocity field in the wake of the APIAN propeller by taking particle image velocimetry (PIV) measurements. This allows for validation of the CFD method's capability of modeling the wake behind the propeller. A two-dimensional flow field in streamwise and spanwise direction is considered here. Three velocity field characteristics are discussed: axial velocity, tangential velocity and vorticity in the wake. The used definitions for these properties are given in equation 3.5.

$$V_{\text{axial}} = V_z \quad (3.5a)$$

$$V_{\text{tangential}} = \sqrt{V_x^2 + V_y^2} \quad (3.5b)$$

$$\omega_y = \frac{\partial V_x}{\partial z} - \frac{\partial V_z}{\partial x} \quad (3.5c)$$

The wake flow field properties for $J = 1.05$ are given in figures 3.11, 3.12 and 3.13, which include the reference data and the mesh refinement data. The same plots for the remaining advance ratios are given in appendix A.

In general, it is concluded that all of the important propeller wake flow characteristics are at least partially captured from the lowest order of refinement onward. For increasing refinement, the CFD method is able to predict the wake better, by increasingly improving the intensity and level of detail. Most obvious, are the propeller tip vortices in the wake of the propeller. They re-occur with equal spacing, which is dependent on the rotation speed of the propeller. The rotation speed decreases for increasing advance ratio, which translates in the wake of the propeller to an increasing spacing between two vortex structures. The intensity of the first vortex structure is captured to an acceptable degree, for all cases. However, compared to the reference case, it shows that, for the subsequent vortex structures, the intensity dissipates and a very fine level of refinement is required to partially capture the intensity of the vorticity inside the flow field. This is especially true at the two highest operating points, because the reference case shows that, compared to $J = 1.05$, the tip vortex has a significantly smaller diameter than in the results obtained from the simulations.

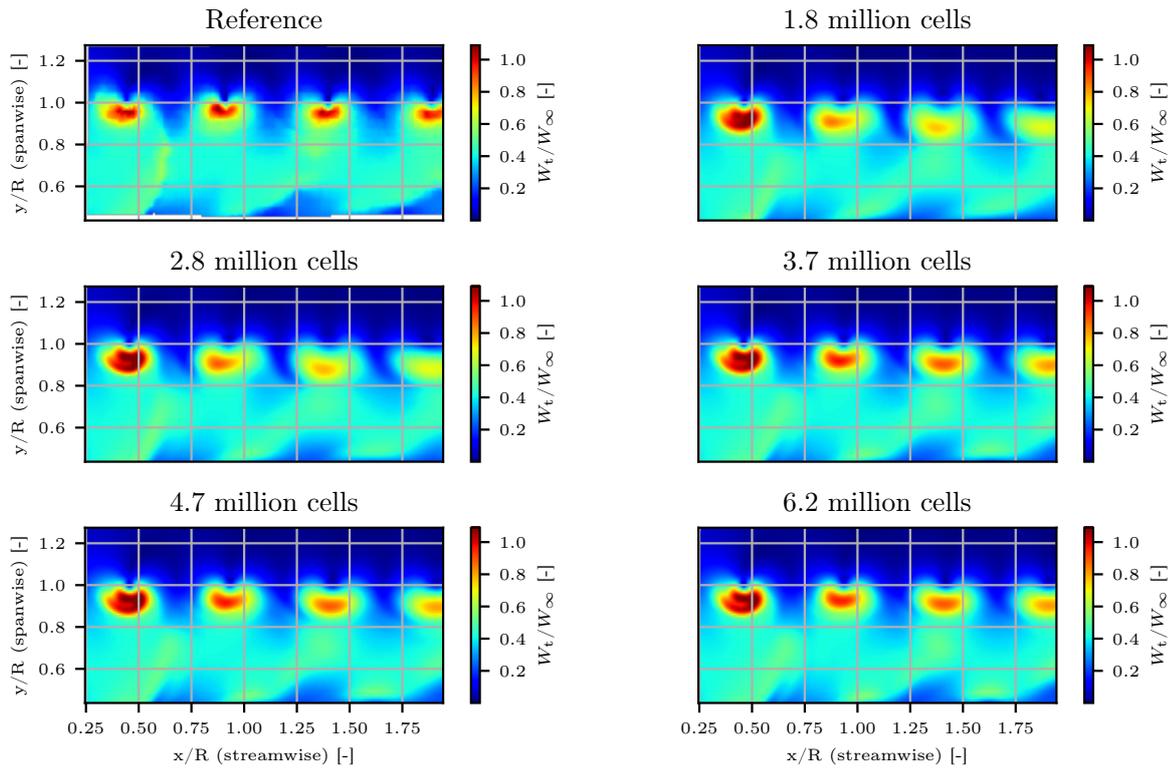


Figure 3.12: 2-Dimensional wake flow field - Non-dimensional tangential velocity for $J = 1.05$.

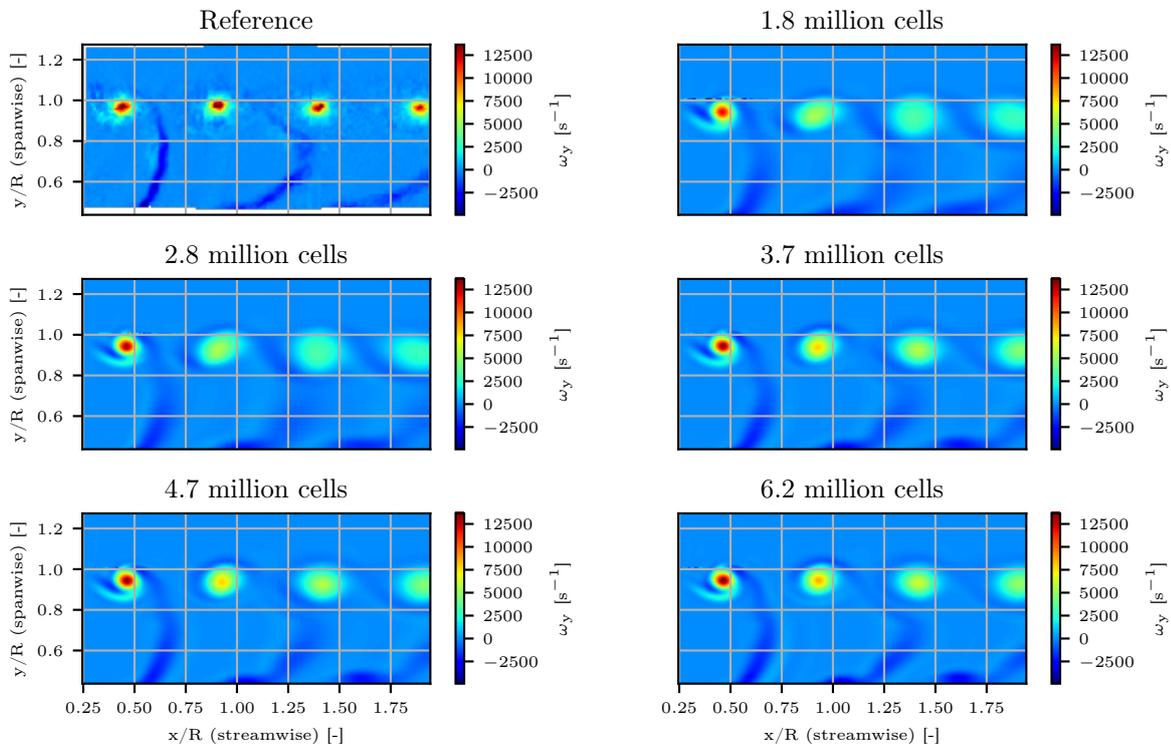


Figure 3.13: 2-Dimensional wake flow field - Vorticity for $J = 1.05$.

This means that for higher advance ratio the refinement in this particular area is not fine enough to capture the flow in this region with enough detail. Thus, in order to capture the intensity of the wake flow, the CFD method is capable, as long as a level of refinement is used that is appropriate for that particular operating point.

3.2.3. Discussion of results

In order to be able to use the CFD method, it needed to be validated and an appropriate level of refinement for the remainder of this research needs to be selected. The most crucial requirement for predicting propeller performance, is being able to accurately predict overall propeller performance characteristics. It is proven to an acceptable degree that this is achieved. For the chosen validation case, there does exist a deviation in slope over the used operating range, between the reference data and the CFD results, however it has been demonstrated that this is caused by a difference in either the validation geometry and the CFD geometry, or how the spinner contribution is taken into account. It is assumed that this is isolated to this particular validation case. The CFD method will be used to verify the performance of an optimized propeller, which is created using low-fidelity tools. These tools don't take the spinner geometry into account anyway when determining the propeller performance. Based on this, the decision is made to only use the isolated propeller blades contribution as the total propeller performance, for the remainder of this research. The sensitivity of the overall performance, due to mesh refinement is limited.

Analyzing the pressure distributions revealed that, independent of mesh refinement, it is difficult to predict the size and location of the leading edge vortex with high accuracy. This is based on the reference data shown here, although no information is available on the uncertainties of the used measurement techniques in the reference case. For the purpose of this research, it is concluded that the method is valid and the results are acceptable. The analysis of the pressure on the blade surfaces, revealed an interesting insight into how to refine the mesh even further without significantly increasing the total number of cells of the mesh. It has been demonstrated that mesh sensitivity is significant in areas that see advanced flow structures, which in this case is the suction side of the propeller blades, while the pressure side is insensitive to any refinement. Throughout the mesh sensitivity study, the number of chordwise grid points is equal on both sides, however the sensitivity study has revealed that the pressure side does not have to have the same refinement as the suction side, in order to predict the flow with the same level of accuracy. Unfortunately, due to nature of a structured grid and the periodicity requirement, the refinement level on the pressure side, must be equal to the refinement on the suction side.

The two-dimensional velocity fields in the propeller slipstream results show that all important propeller wake flow characteristics are modeled with the CFD method. However, for lower refinement, the tip vortex intensity dissipates quickly when moving away from the propeller. Only for the finer refinement levels, are the results acceptable. It has also confirmed a trend that is noticed in the overall performance and pressure distribution results. Mesh refinement should be based on not only the complexity of geometry, but also on expected flow conditions. For higher advance ratios, the different flow structures that are present in the wake are considerably smaller compared to the high loading operating point. This means that, in order to capture the flow phenomena at a higher advance ratio, more mesh refinement is required. This needs to be taken into account when selecting the appropriate level of refinement for the remainder of this research.

Based on these findings, it is concluded that the fine refinement level (4.7 million cells) is most suitable to be used throughout this research. This is mainly based on the fact that the velocity fields in the propeller slipstream are more accurately predicted compared to coarser results and since the CFD analysis is not used inside the optimization loop, a more computationally expensive refinement level can be afforded.

4

Parametrization tool

The proposed approach for optimizing propeller performance requires the geometry a prospective propeller to be modeled, by applying parametrization. The main purpose of the parametrization tool is to accurately represent a complex geometry with the least amount of parameters, for optimization purposes. Additionally, the tool must be able to construct a propeller geometry model that is usable in the CFD analysis. Secondary requirements state that the parametrization tool must be usable in future projects. This means that the inner workings of the tool must be well structured, easy to understand and the propeller design approach must be flexible to account for different demands. The tool must be able to generate a propeller geometry based on a full set of data describing the propeller geometry, as well as be able to generate a propeller geometry using a preliminary design method. This chapter gives an overview of the created parametrization tool.

4.1. Tool structure

The parametrization tool is created within the OptiBlade environment, which is a Python-scripted environment for modeling different turbo-machinery applications, developed at the Delft University of Technology faculty of Aerospace Engineering [27]. It offers a wide range of geometric modeling functions, which can be used for the propeller parametrization tool as well. Most importantly, it has integrated CAD scripting capabilities, using the open-source CAD tool FreeCAD [28]. The parametrization tool will be created with an object-oriented programming approach. This will ensure a well structured code that is stable and easy to understand. Figure 4.1 shows the activity diagram of the main class used to generate all the necessary parts.

The parametrization tool is initiated with an “initiator” text file, which contains all the required relevant design parameters and settings. Some obvious, general parameters are identified and given below:

- *Case type* - [*developed-case/preliminary-case*]: A decision parameter that defines whether a preliminary or a developed case is defined, which is necessary for operational purposes. When the case type is a “developed-case”, the tool expects an initiator file that fully describes the propeller geometry. In the “preliminary-case”, the tool will only require a minimal description of the propeller and performs a minimum induced loss design routine to find the remaining parameters.
- *Assembly type* - [*blade-spinner/propeller*]: Another decision parameter that defines what type of final assembly will be generated. Either a single blade or all blades with the spinner, respectively. This option has been introduced, because it offers a computationally low-cost option for inspecting the detailed blade geometry.
- *Number of blades* - [-]: An obvious top level propeller design parameter that is mainly used in the propeller assembly class in order to generate a CAD model of the complete propeller. In case of preliminary design case also important factor for determining the required force thrust production for a single blade.
- *Rotation direction* - [*CCW/CW*]: The rotation direction of the propeller is an important design parameter, because it influences how the blades are positioned, with respect to the oncoming flow. Throughout the parametrization tool, a standard aircraft reference frame is used. The rotation direction is thus defined as observing the propeller from behind.

The definition of the parametrization tool is divided into three main parts. The first part focuses on generating a preliminary design, using an established method from chapter 2. Subsection 4.1.1 gives an overview of different methods found in literature and a full description of the final selected method. The most important part of the parametrization tool is the definition and creation of the blade geometry. It is clear that the tool must be able to generate a blade geometry with advanced blade shapes. A full description is given in

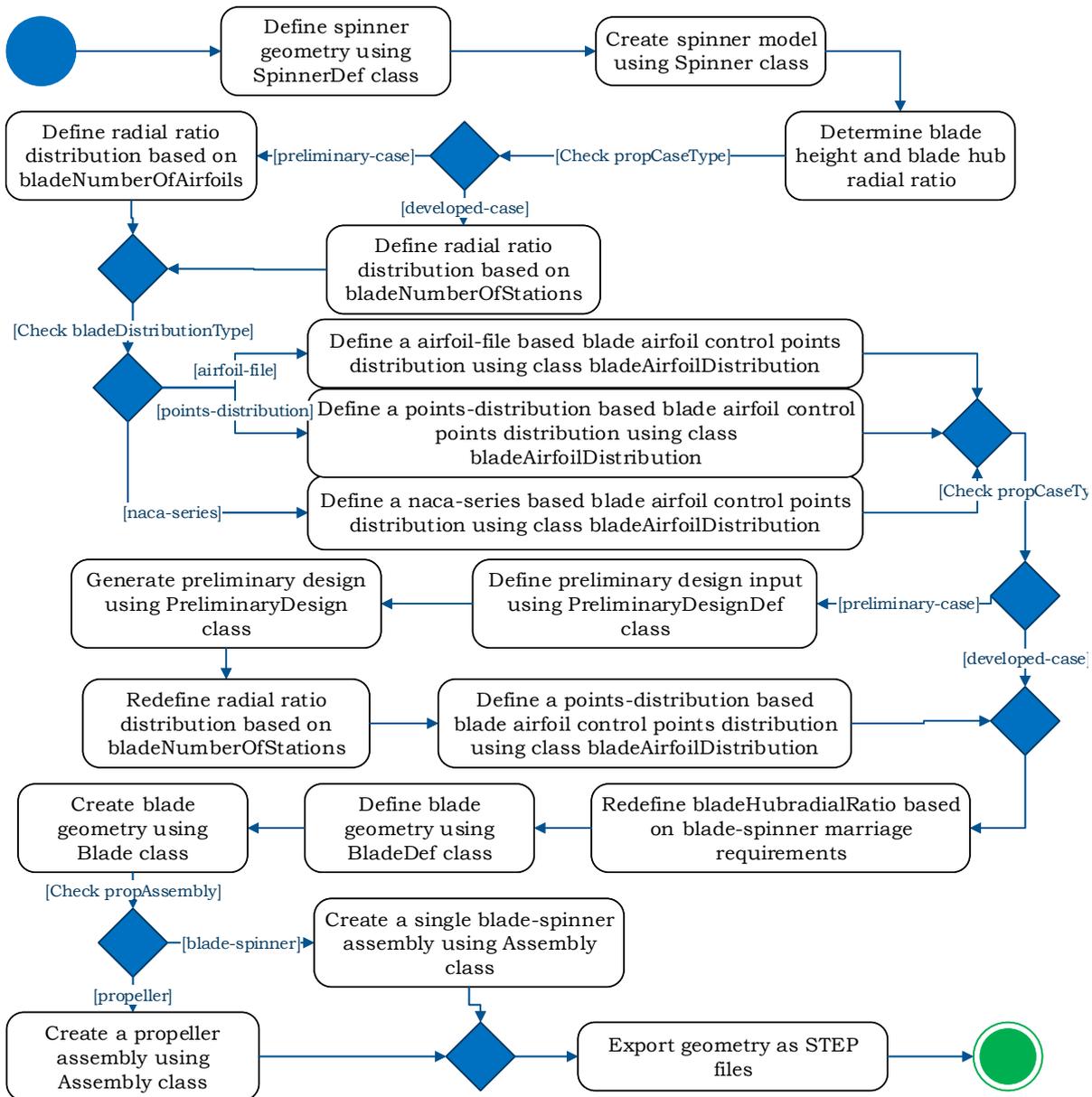


Figure 4.1: UML activity diagram of main propeller class.

subsection 4.1.2. It is also required to define the spinner geometry. A full description is given in subsection 4.1.3.

4.1.1. Preliminary design method

In chapter 2, three different methods were considered to be used as a low-fidelity propeller performance analysis method: Xrotor [6], JavaProp [7] and a scheme by Adkins & Liebeck [11]. Also, the observation was made that JavaProp is based on the Adkins & Liebeck scheme. For performance analysis, each method is very sensitive to the required inputs, while for a design method, this is less the case. Based on this, JavaProp is assumed to be equal to the Adkins & Liebeck scheme, which means that only Xrotor and the Adkins & Liebeck scheme are considered here. This section gives the motivation for selecting one method over the other and how it is implemented in the parameterization tool.

Motivation for method selection

The design routine of Xrotor is based on the minimum induced loss principle [6], originally developed by Betz [29], Goldstein [15] and Theodorsen [30]. It takes basic design parameters such as the number of blades, hub and tip radius, airspeed and propeller speed, desired thrust or power output and design lift coefficient, and determines the remaining blade geometry. Xrotor actively involves the user in the design process, by giving the option to modify design parameters and giving immediate feedback on how the modification affects the performance of the propeller design [6].

The Adkins & Liebeck scheme [11] describes a relatively simple scheme for analysis and design of propellers. It is possible to recreate the Adkins & Liebeck scheme design procedure in an automated scripting environment. In order to generate a propeller design with this method, similar basic design parameters, compared to Xrotor, are required.

Both methods are limited to creating conventional, straight propeller blade designs and will only be used to generate a preliminary design that is compatible with the parametrization tool. Based on these requirements, the Adkins & Liebeck scheme is the preferred method. The major drawback of Xrotor is that it is required to create an interaction between Xrotor and the parametrization tool, which is inefficient. The Xrotor design procedure itself is straightforward, but the efforts required to generate input files and output files, in combination with the parametrization tool, makes Xrotor unattractive for the purpose of generating an initial, preliminary design. The Adkins & Liebeck scheme can be integrated directly into the parametrization tool, which is significantly more efficient.

Design method implementation

The main objective of the preliminary design method is to determine the chord and twist distribution and the blade pitch angle for the propeller blades. All other design aspects need to be defined. This subsection describes the propeller design scheme by Adkins & Liebeck [11]. The required parameters for the preliminary design method are:

- Propeller rotational speed [rpm]
- Free-stream velocity [m/s]
- Design lift coefficient [-]
- Design target variable [thrust/power]
- Design target variable value [N/W]
- Free-stream air density [kg m^{-3}]
- Free-stream dynamic viscosity [$\text{kg m}^{-1} \text{s}^{-1}$]

The design target variable and its value represent a constraint for the design process. Either the thrust or power required needs to be defined.

This propeller design method combines propeller lifting-line theory with the principle of minimum induced loss. During the literature study, it was concluded that in order to determine the flow and performance characteristics of a propeller, using lifting-line theory, it is required to determine the blade inflow angle distribution $\phi(r)$ and the velocity displacement ratio $\zeta(r)$. Adkins & Liebeck conclude that in order for the Betz condition to be relevant, the velocity displacement ratio must also be independent of radius. However the blade inflow angle distribution $\phi(r)$ is dependent on ζ , which means it will need to be solved iteratively. For this optimization and design approach, it is required to specify either a desired thrust or the power, delivered to the propeller. Adkins and Liebeck define these in non-dimensional form (equation 4.1):

$$T_c = \frac{2T}{\rho V_0^2 \pi R^2} \quad (4.1a)$$

$$P_c = \frac{2Q\Omega}{\rho V_0^3 \pi R^2} \quad (4.1b)$$

When equation 4.1 is combined with lifting-line theory, an alternative form of the non-dimensional thrust and power derivatives is found and defined in equation 4.2:

$$T'_c = I'_1 \zeta - I'_2 \zeta^2 \quad (4.2a)$$

$$P'_c = J'_1 \zeta + J'_2 \zeta^2 \quad (4.2b)$$

Where:

$$I'_1 = 4 \frac{r}{R} G \left(1 - \frac{c_l}{c_d} \tan \phi \right) \quad (4.3a)$$

$$I'_2 = \lambda \frac{I'_1 R}{2r} \left(1 + \frac{c_l}{c_d \tan \phi} \right) \sin \phi \cos \phi \quad (4.3b)$$

$$J'_1 = 4 \frac{r}{R} G \left(1 + \frac{c_l}{c_d \tan \phi} \right) \quad (4.3c)$$

$$J'_2 = \frac{J'_1}{2} \left(1 - \frac{c_l}{c_d} \tan \phi \right) \cos^2 \phi \quad (4.3d)$$

Only one equation is required to determine ζ , which means that when desired thrust is specified, the sub-equation in equation 4.2 related to the thrust is used, while the other sub-equation is simply used to determine the required engine power, for example. The constraint equation for the case of specified desired thrust, becomes:

$$\zeta = \frac{I_1}{2I_2} - \sqrt{\left(\frac{I_1}{2I_2} \right)^2 - \frac{T_c}{I_2}} \quad (4.4)$$

And when engine power is specified:

$$\zeta = \sqrt{\left(\frac{J_1}{2J_2} \right)^2 + \frac{P_c}{J_2}} - \frac{J_1}{2J_2} \quad (4.5)$$

In order to determine the propeller geometry, an iterative procedure is required to reach an optimum minimum induced loss design. According to Adkins & Liebeck, the design process proceeds in the following steps:

1. Select an initial estimate for ζ ($\zeta = 0$ will work)
2. Determine for each blade section:
 - (a) Prandtl tip correction factor: $F = \frac{2}{\pi} \cos^{-1}(e^{-f})$
 - (b) The blade inflow angle
 - (c) The aerodynamic characteristics such as local Reynolds number, angle of attack, lift and drag characteristics, which are determined by applying the external tool Xfoil [12]
 - (d) Determine the blade geometry, the chord follows from the relation $Wc = 4\pi\lambda GVR\zeta/(C_l B)$, the blade twist angle follows from the relation $\beta = \phi + \alpha$
 - (e) Determine the four derivatives from equation 4.3
3. Numerically integrate the derivatives from the previous step along the blade span
4. Determine the velocity displacement ratio ζ from either equation 4.4 or equation 4.5, depending on the specified constraint
5. If the new ζ is not sufficiently close (Adkins and Liebeck suggest 0.1% [11]) return to step 2 with the new ζ , until it converges
6. Convert the found pitch and chord distribution into usable formats for the parametrization tool.

4.1.2. Blade parametrization

The generation of a blade geometry is the main purpose of the propeller parametrization tool. A good parametric model for optimization, offers good manipulation capabilities, while limiting the number of variables controlling the geometry. These characteristics are contradictory, which means that a good compromise is required. A common method for modeling blade geometry is to radially discretize a blade into two-dimensional airfoil sections and loft through each section to generate a solid blade. This is appropriate for the purpose of this research, because it is achievable with the current scripting library, and it allows for compartmentalization of the code structure such that it is well structured and easy to understand. This means that, for each two-dimensional section, its overall shape, size, orientation and position need to be defined. A major drawback of this method is that there is a potential for a huge amount of design parameters controlling the shape of the blade, if each aspect is not parametrized carefully. In order to limit the number of parameters considerably, Bezier curves are used. By applying Bezier curves, there is no need for the definition of a discrete radial distribution describing one design aspect of a blade. Instead, a limited number of Bezier control points are defined that will describe the discrete radial distribution using a Bezier curve.

Table 4.1: Initial overview of propeller blade design parameters, as proposed by Gur [31].

Blade specific variables	Section specific variables
Pitch or twist angle distribution	Thickness ratio
Chord distribution	Design lift coefficient
Sweep angle distribution	
Dihedral or lean angle distribution	
Hub and tip Radius	

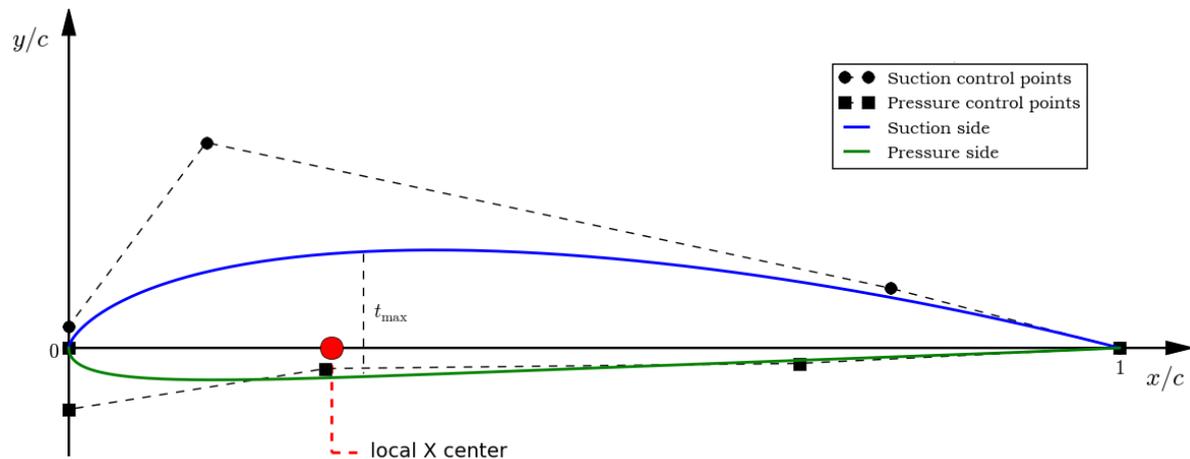


Figure 4.2: Graphic representation of the airfoil distribution definition.

During the literature study, an initial overview was given of possible design variables for a potential blade parametric model, which is based on the work of Gur [31], and is given in table 4.1. The pitch or twist angle of a section will describe its orientation, the chord length describes its size, the sweep and lean angle combined with a radial position distribution describes its global position. The shape of a section is section specific.

Section position definition

The section position definition consists of two components. First, it is convenient to define a local reference point on a section that can be used to position each section with respect to another. The local reference point is also useful as it can be used as a reference point for the orientation of the section. At each section, this point is defined as the local center axis, which is characterized as a point on the chordline at a variable distance from the leading edge. Figure 4.2 gives its definition graphically. The second part of the section position definition describes how the local reference points of all sections are positioned with respect to each other. This is defined as the propeller center axis, which has a radial, sweep and lean component.

Blade center axis radial definition This distribution starts at the hub radius and ends at the tip radius. The most obvious method for defining this distribution is to equally distribute the sections from hub to tip. This method is inadequate because a more dense distribution is desired near the hub and tip, since it is expected that the variation in section shape, size and orientation is more significant in this region. A definition is required that offers a dense distribution near the hub and tip and less dense in the center of the blade. This is achieved by applying a third degree Bezier curve as shown in figure 4.3. The required design parameters are the hub and tip radius, the number of sections and a parameter that controls the radial position of the two center Bezier control points.

A requirement from the CFD for this parametrization tool states that the spinner-blade junction must be flush. This can be achieved in three ways. First, the entire blade geometry is sunk into the spinner, by translating the geometry such that there is no longer a gap between the spinner and blade hub. This is a major design change and will seriously impact the expected performance. Second, the hub section is extended downwards into the spinner, such that there is no longer a gap between the spinner and blade hub. There are two ways to define the extension. The first method consists of defining a new “sink” section that is identical to the hub section, except the radial position. The second method consists of defining a new “sink” section that has the same shape and size of the hub section, but follows the radial trend of the position and orientation. Both are preferred over the translation method, however, in the case of a small spinner radius a

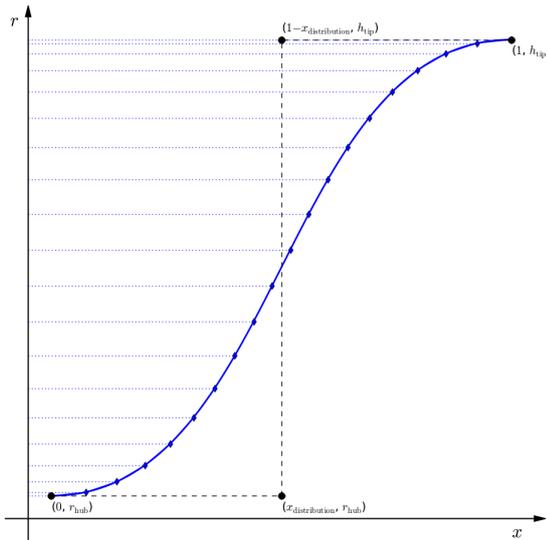


Figure 4.3: Graphic representation of the radial position distribution definition.

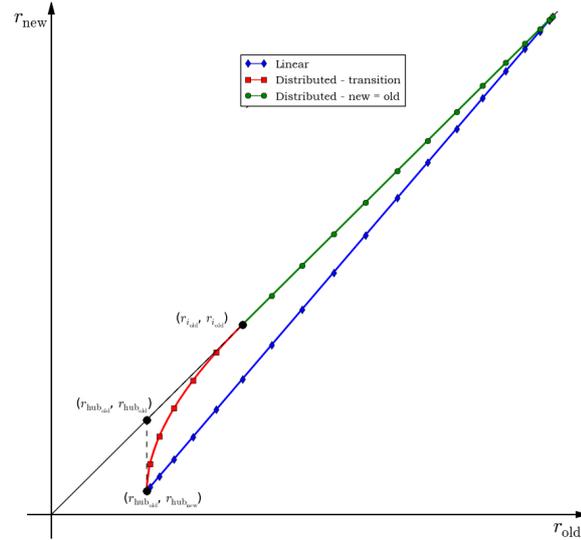


Figure 4.4: Graphic representation of the stretch methods required to generate a flush spinner-blade junction.

large extension is required, which means that there is either a significant discontinuity in the blade geometry, or the junction is at a large offset, which are not realistic junctions. The third and only realistic method for creating a flush spinner-blade junction is by stretching the radial position distribution, such that there is no longer a gap between the spinner and blade hub. There are two ways of stretching: linearly or distributed. Both are demonstrated in figure 4.4. The main advantage of the linear stretch method over the distributed stretch method is that it does not generate an additional design parameter. The main disadvantage of the linear stretch method over the distributed stretch method is that all sections (with the exception of the tip section) are out of position, compared to the intended design. In the distributed method, only the region near the hub is out of position, which is less important compared to the rest of the blade in terms of performance. For the purpose of this research, the distributed stretch method is preferred and implemented. This creates an additional design parameter, which is the section number that is the point where the stretched region transfers into the normal region.

Blade center axis sweep and lean definition In order to be able to investigate advanced blade shapes, it is preferred to include blade sweep and blade lean characteristics. Blade sweep can be characterized as an offset between the different sections in rotational direction, while blade lean can be characterized as an offset in longitudinal direction. Sweep and lean can therefore be interpreted as definitions for the global position of each section. It is important to be able to generate realistic sweep and lean distributions, while limiting the number of design parameters. This is achieved with low degree Bezier curve. The sweep and lean distributions are defined by second-degree Bezier curves, that ultimately define the lateral and longitudinal position of each section, respectively. The exact definitions are given in figures 4.5 and 4.6.

The sweep curve is controlled by three parameters: the hub sweep angle, the normalized radial position of the interior control point and the tip sweep angle. The lean curve is also controlled by three parameters: the hub lean angle, the normalized radial position of the interior control point and the tip lean angle. Blade sweep has an impact on the used method of defining blade geometry with a distribution of two-dimensional shapes. While the blade radius remains constant, the blade height reduces, which results in a mismatch between the different distributions, which means that it will not produce the intended blade geometry. To solve this, all radial definitions will start at the hub radius and end at the tip height, not the tip radius.

Section size and orientation definition

Section size and orientation are defined as the blade chord length and pitch angle. The chord length is used to scale the section shape, while the pitch angle is used to rotate the section shape about the local reference axis. It is convenient for propellers to split the pitch angle of each section into a twist angle contribution and a blade pitch angle contribution. The blade pitch angle is a collective pitch angle that is added to the twist distribution to get the blade pitch distribution. The twist angle is, like the chord length, specific for each section. The radial chord and twist distributions must be parametrized, in order to reduce the number of design parameters.

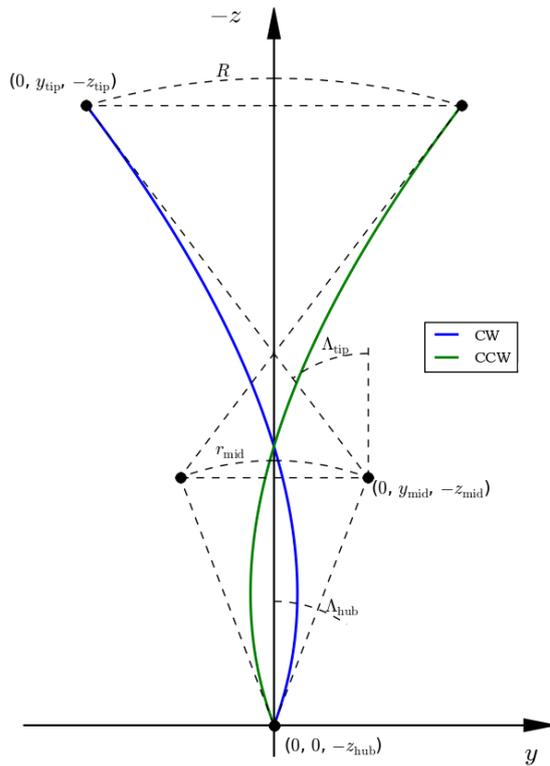


Figure 4.5: Graphic representation of the sweep distribution definition.

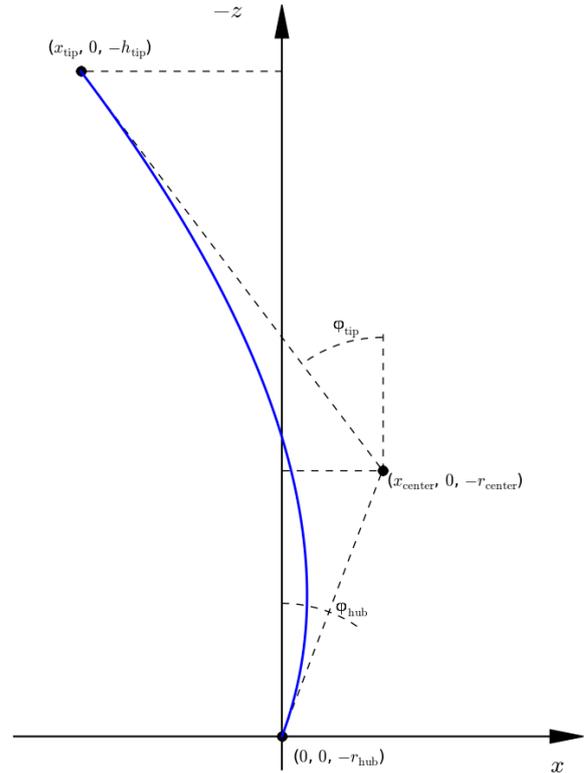


Figure 4.6: Graphic representation of the lean distribution definition.

The chord and twist both have significant impact on the performance of a propeller blade. It is impossible to define a fixed twist and chord definition that is able to anticipate different distributions. Furthermore, in the case of a preliminary design, a discrete distribution for twist and chord is generated, which must be transformed into the same fixed twist and chord distribution definitions. It is therefore deemed appropriate to parametrize these distributions using high degree Bezier curves, with a variable number of Bezier control points that are not bound to any specific definition. In the case of a developed case, it is required to define lists of two-dimensional Bezier control points that describe radial twist and chord Bezier curves.

Section shape definition

It is a common practice to model propeller blades with two-dimensional airfoil shapes. The most common method for describing an airfoil shape is the use of coordinates. In this case, it is not the preferred method since it will generate too many design parameters. Instead of coordinates, it is possible to model each airfoil shape with Bezier curves, one describing the suction side and one describing the pressure side, as shown in figure 4.2.

By applying this method to all sections, the number of design parameters reduces considerably, but not enough. A second level of parametrization is required that is able to describe these Bezier curves in a limited number of parameters. It is also important to take into account different desired approaches. This parametrization tool makes it possible to define an airfoil distribution in three different ways. This will ensure that when setting up the optimization process or for future use, enough options are available to model the airfoil distribution.

Airfoil coordinates file method This method requires a single user-defined airfoil coordinates file, which describes a single airfoil that will be applied to all sections. These coordinates are imported and must be converted to Bezier control points, which is achieved by applying a fitting procedure. The number of control points for both the suction and pressure side is a design parameter and does not include the leading and trailing edge, which are trivial. This means that the minimum degree of the Bezier curves is two. It is up to the user to find a balance between accuracy and computation time to determine the appropriate number of control points.

The Bezier curves that define the airfoil shape are applied to all stations. One airfoil for all sections might not give enough control of the blade shape. Because of this, the method is extended to include a maximum

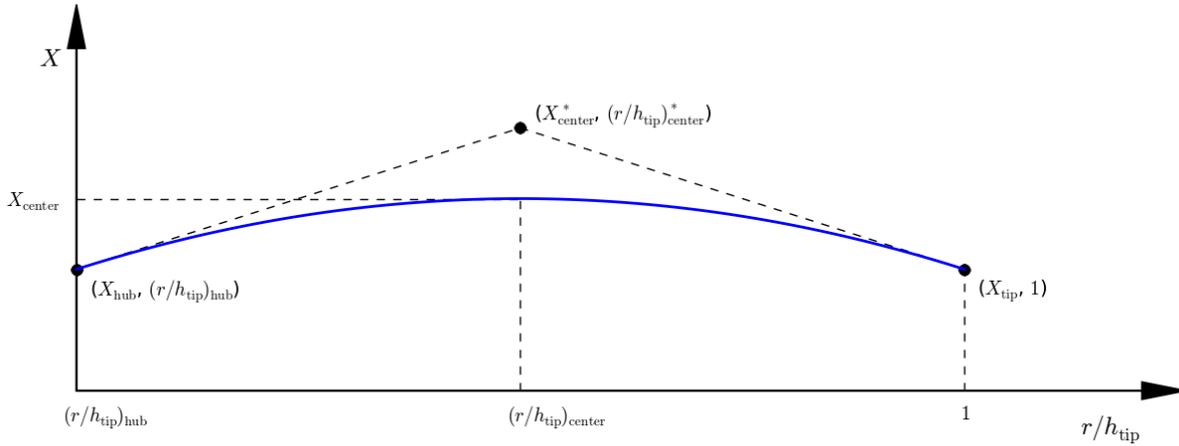


Figure 4.7: Graphic representation of the generalized distribution definition, which applies to the maximum thickness distribution, maximum camber distribution, maximum camber position distribution and a design lift coefficient distribution.

thickness distribution. This is a radial distribution that dictates the maximum thickness of each section. Figure 4.7 shows how the maximum thickness distribution is defined as a second degree Bezier curve, with X_{hub} , X_{center} , X_{tip} and $(r/h_{\text{tip}})_{\text{center}}$ as the design parameters, controlling the Bezier control points. When no change in maximum thickness is required, all parameters must be set to zero, with the exception of $(r/h_{\text{tip}})_{\text{center}}$.

NACA four/five series method This method allows an airfoil distribution to be modeled, based on NACA four/five digit series airfoils. The shape of these airfoils are described by a set of equations that are dependent on a number of airfoil characteristics:

- NACA four series
 - Maximum thickness [%]
 - Maximum camber position [%]
 - Maximum camber [%]
- NACA five series
 - Maximum thickness [%]
 - Maximum camber position [%]
 - Design lift coefficient [-]
 - Standard/reflex camber line

A NACA airfoil is used at every section, which means that in order to minimize the number of parameters, Bezier curves are applied to create radial distributions of each NACA airfoil characteristic. The decision parameter “type of camber line” (in case of NACA five series) applies to all sections. The same type of definition is used for all NACA radial distributions. Figure 4.7 shows how every distribution is defined as a second degree Bezier curve.

With all required input parameters defined, coordinates are generated using well known NACA four/five equations, which then need to be converted into Bezier control points. For this method, the number of control points is fixed at six, for both pressure and suction side. This is done because, unlike in the airfoil file method, the expected shape is known in advance, which means that the balance between accuracy and computation time can be determined, and the number of parameters reduced. In literature, methods exist that directly determine the Bezier control points, based on NACA airfoil coordinates (i.e. Fazil & Jayakumar method of reverse engineering a Bezier curve [32]). However, no method was found that generated reliable and accurate control points. The only other option is to apply a similar fitting procedure as used in the airfoil file method.

Bezier control points method The final method allows the user to define the Bezier control points that make up a non-dimensional airfoil, for a limited number of sections. This method is implemented for cases where more freedom is required in using distinctly different shapes at different sections. The required input for this method is given below:

- Number of defined airfoil sections
- Discrete non-dimensional radial position distribution
- Radial distribution of Bezier control points describing each airfoil’s suction side
- Radial distribution of Bezier control points describing each airfoil’s pressure side

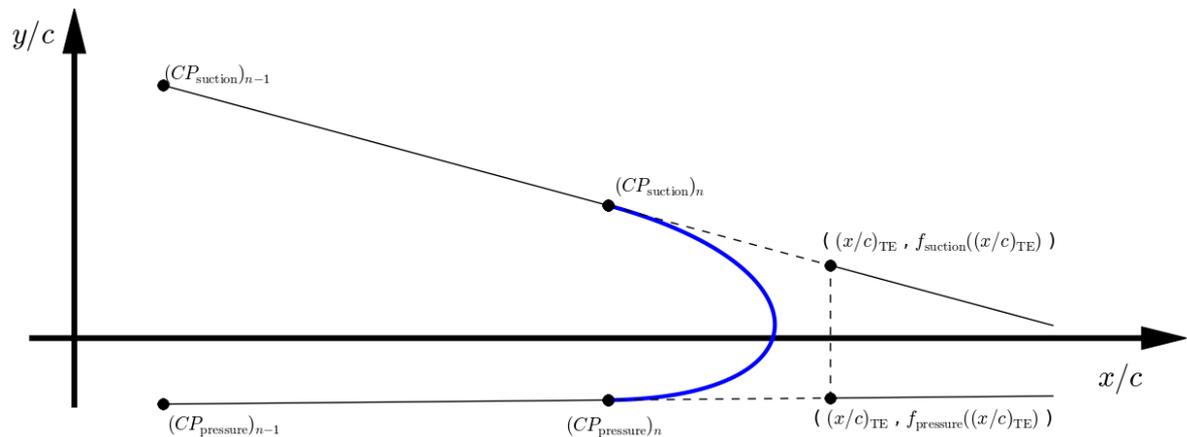


Figure 4.8: Definition of the trailing edge gap curve. A third degree Bezier curve that is tangent with the suction and pressure side.

The Bezier control points need to be defined for at least one airfoil. In the case of multiple defined airfoils, the number of Bezier control points must be equal for each section. For this method, no maximum thickness scaling is available, because it would be redundant since the maximum thickness can be controlled through the individual control points.

Since the parameters that control the airfoil shapes are already in the correct Bezier format, the implementation is more straightforward. The only problem that needs to be solved is that the limited Bezier control points distribution needs to be applied to all the sections that make up the blade. This is achieved by applying an cubic interpolation procedure. Because the number of control points must be equal for a specific side, the radial distribution of airfoil sections, each described by Bezier control points can be re-arranged to create radial distributions for the first, second, etc, Bezier control point of each airfoil section. Then, cubic interpolation is applied to create radial distributions of Bezier control points for all geometric sections, which are then re-arranged and the end product is a radial distribution of all geometric sections.

Trailing-edge gap curve The three different methods for defining an airfoil distribution are only focused on generating the suction and pressure side of the blade. In order to generate a valid blade geometry, it is required to close the trailing-edge gap with some curve. The most obvious solution is a simple straight section, however for the purpose of this research, a smooth curved section is required. The only requirement is that the curve must be tangent at the points where the trailing-edge curve connects with the pressure and suction curves. Multiple configurations were considered, but in the end, a third degree Bezier curve proved to be the solution that would work with all possible tangency conditions. The configuration is given in figure 4.8.

The curve is dependent on the last two Bezier control points of the suction and pressure side curves and the design parameter that controls the streamwise position of the two middle Bezier control points. In the case of the “Airfoil coordinates file” method and the “NACA four/five series” method, the design parameter that controls the streamwise position is defined with the same radial distribution definition as the maximum thickness distribution, given in figure 4.7. In the case of the “Bezier control points” method, a discrete radial distribution must be supplied that is related to the supplied discrete distribution of airfoil sections.

4.1.3. Spinner parametrization

The definition of the spinner is less complex as the definition of the blades, and should at least represent a realistic shape that is defined with a minimal number of design parameters. In order to simplify the spinner design, only axisymmetric shapes are considered, which will consist of a curved front section (dome) and a straight aft section (hub). In the parametrization tool, the outline of the axis-symmetric spinner is defined, after which it is revolved to create a solid CAD object.

The dome is modeled by defining its outline with a third-degree Bezier curve. The hub outline is modeled as a simple straight section with the same radius as the height of the dome outline. The exact definition of the outline of the spinner is given in figure 4.9, the resulting spinner design parameters are:

- l_{dome} : Dome length [m]
- $F_{\text{domeLength}}$: Dome control point length ratio [-]
- $F_{\text{domeHeight}}$: Dome control point height ratio [-]
- r_{hub} : Hub radius [m]

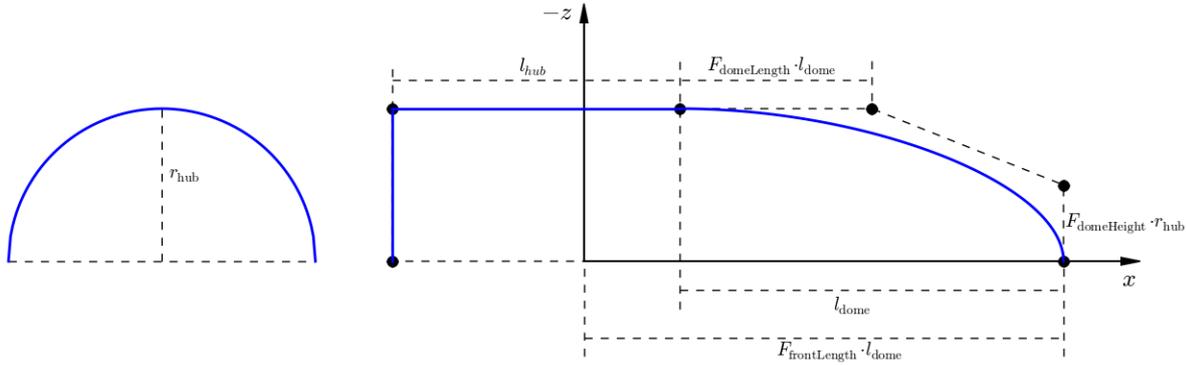


Figure 4.9: Graphic representation of the spinner geometry definition.

Table 4.2: Summary of all defined general blade input parameters for the N250 propeller.

Variable	Unit	Value
Number of blades	-	6
Rotation direction	-	CW
Tip radius	m	0.2032
Hub radius	m	0.04674
Local center axis	-	0.5
Hub sweep angle	degrees	0.0
Hub lean angle	degrees	0.0
Mid sweep radial ratio	-	0.6
Mid lean radial ratio	-	0.6
Tip sweep angle	degrees	0.0
Tip lean angle	degrees	0.0

- l_{hub} : Hub length [m]
- $F_{\text{frontLength}}$: Front length factor [-]

4.2. Parametrization tool verification

Before the parametrization tool can be applied, it must first be verified that it is capable of generating propeller geometries that are realistic, and that any simulation or analysis using the output of the parametrization tool is relevant. Because of this, a verification study is performed with both the low- and high-fidelity performance analysis methods.

4.2.1. Low-fidelity analysis verification

Since the parametrization tool and the low-fidelity performance analysis method of chapter 2 interact, it is necessary to verify parametrization tool in combination with the low-fidelity performance analysis method, to ensure that, any conclusions that are made based on results from the low-fidelity performance analysis method are reliable. The setup of this part of the verification is very similar to the setup of the validation of the low-fidelity performance analysis method in chapter 2. The N250 propeller is used as a reference case. For the verification, the low-fidelity results of the validation study are compared against low-fidelity results which are generated using a parametrized model of the N250 propeller. The remainder of this subsection describes the definition of the parametrized version of the N250 propeller, and presents the results of the verification study.

Parametrization of the N250 propeller

The N250 propeller geometry has already been introduced in chapter 2. It is a six-bladed propeller with no lean and sweep. The majority of the required parameters required to describe the propeller are simply known and are given in table 4.2.

Blade twist and chord distribution The blade twist and chord distribution are known in the form of discrete distributions. All that is required is to describe each distribution with a Bezier curve, with a variable number of control points. The number of used control points must be based on the desire for a small number of control

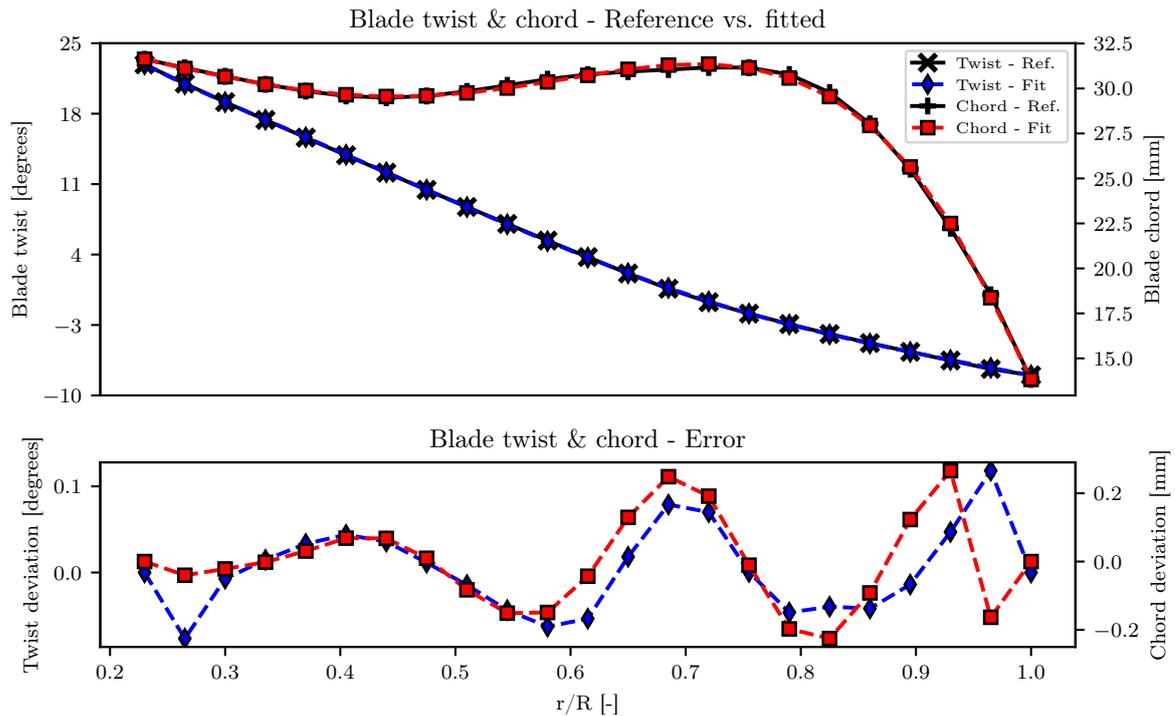


Figure 4.10: Top: N250 propeller blade twist & chord distribution versus fitted third-degree Bezier curve. Bottom: Twist & chord error, defined as the local difference between the reference data and the fitted distribution

points, while achieving a small fitting error at the same time. The number of used Bezier control points is found by assessing the complexity of each distribution and the error sensitivity. Ultimately, for the chord and twist distribution, six and three control points are used, respectively. In order to determine the actual control points, a fitting procedure is applied that finds the position of each control point by minimizing the total error, which is defined as the sum of the absolute local differences between the reference and the fitted curve, normalized with a relevant parameter. For the chord distribution, this is the mean chord, while for the twist distribution, this is the blade pitch angle. The blade pitch angle is not explicitly defined and is variable. Instead, in chapter 2 experimental data is used for varying blade pitch angles, ranging from 25 to 35 degrees. In the case of the chord distribution, the fitting procedure yielded an average error of 0.35% and a maximum local error of 0.93%. For the twist distribution, the fitting procedure yielded an average error of 0.13% and a maximum local error of 0.39%. A detailed overview is available in figure 4.10.

Blade section shape definition For the parametrization tool, it is required to define a section shape distribution. The parametrization tool offers three different methods for defining section shapes. Since the section shapes are only available in the form of coordinates, the only applicable method is by defining a Bezier control points distribution. In short, it is required to define a limited number of section shapes by Bezier curves. From the available twenty-one sections (labeled from “CC” at the hub to “YY” at the tip), twelve sections equally spaced are used to create the Bezier control points distribution.

In the parametrization tool, each section consists of a suction side, a pressure side and a trailing edge side. Initial assessment of the available data revealed that all sections are defined with a trailing edge gap, which means that there are no trailing edge sides to model. The trailing edge side is modeled by a third-degree Bezier curve and controlled by a single parameter: the horizontal position of the middle control points. For the N250 propeller, the decision is made to assume a constant value of 1.001 across all the sections. The suction and pressure side of each section are defined by Bezier curves with a variable number of control points. The choice for the number of control points is very important. It is a compromise between the desire for a small fitting error and a small number of control points. The N250 propeller has significantly changing camber and thickness characteristics and is therefore difficult to describe accurately with only a low number of control points. Therefore, eight control points are applied for each side. The set of Bezier control points is found by applying a fitting procedure. A complete overview of the errors found at every section is given in table 4.3 and more detailed results for the first section is given in figure 4.11.

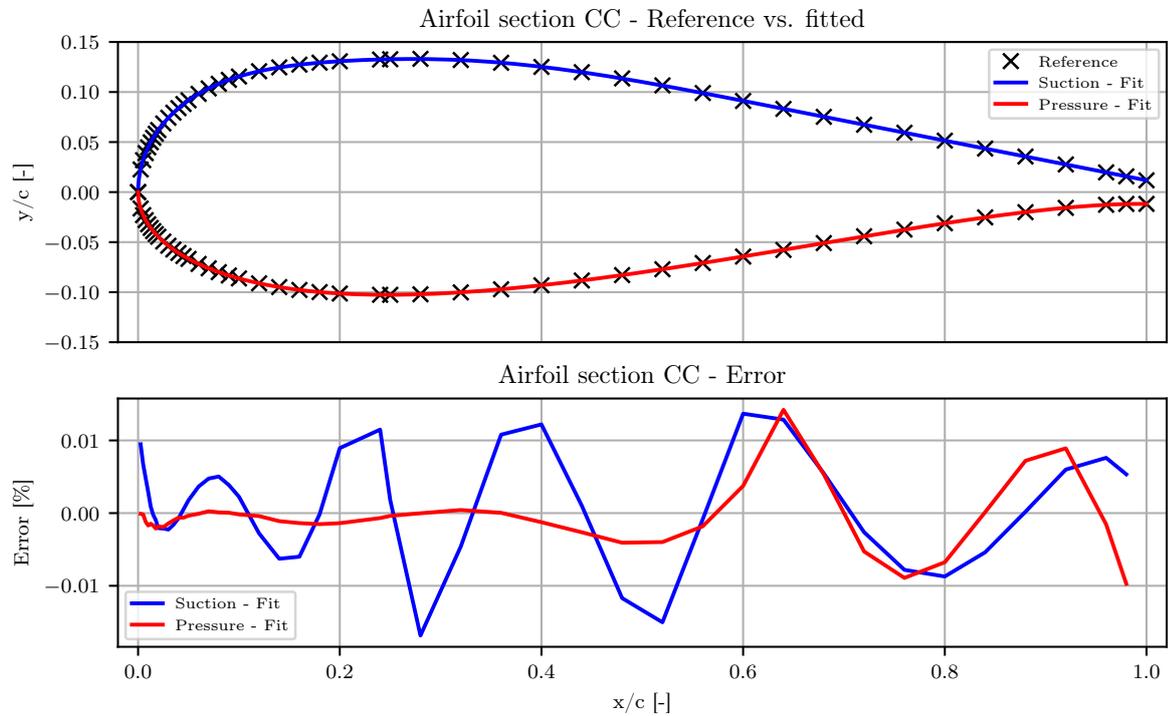


Figure 4.11: Top: N250 propeller section CC airfoil versus fitted seventh-degree Bezier curve, both on suction and pressure side. Bottom: fitted error, defined as the local difference between the reference data and the fitted distribution, normalized with the mean blade chord.

Table 4.3: Overview of fitting errors of airfoils sections.

Section	Maximum error [%]	Average error [%]	Section	Maximum error [%]	Average error [%]
CC	0.0168	0.0039	OO	0.0289	0.0058
EE	0.0304	0.0041	QQ	0.0463	0.0067
GG	0.0220	0.0052	SS	0.0388	0.0063
II	0.0099	0.0026	UU	0.2777	0.0324
KK	0.0146	0.0034	WW	0.0302	0.0058
MM	0.0227	0.0056	YY	0.0323	0.0060

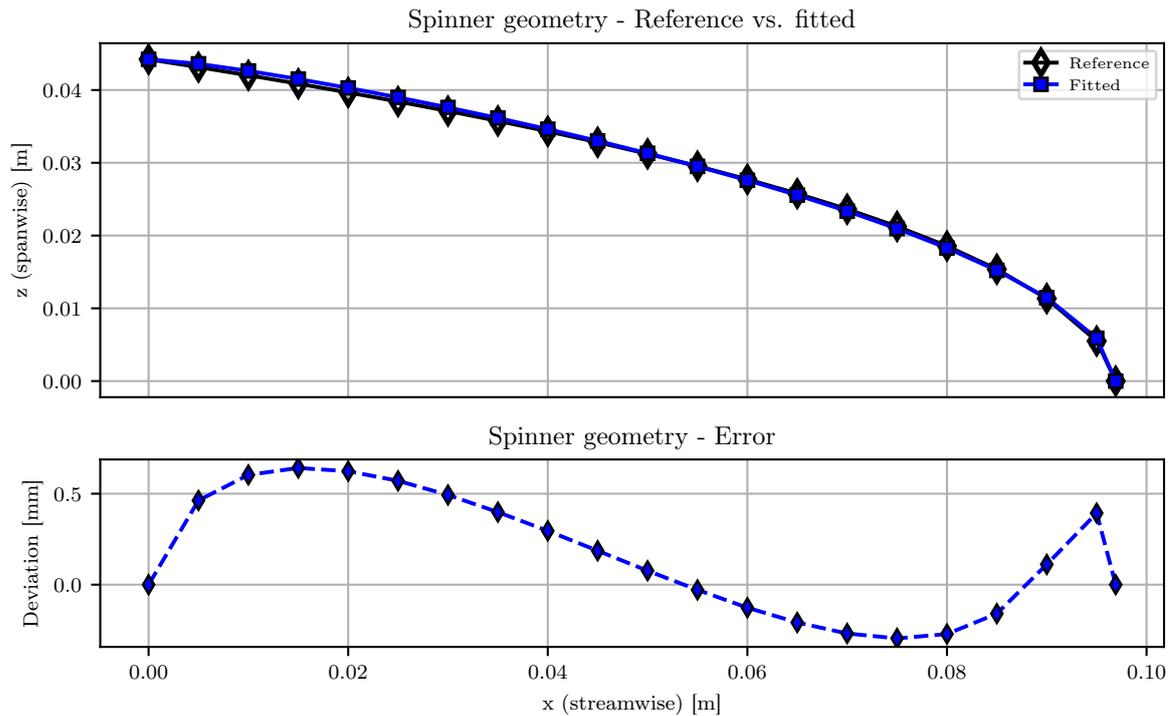


Figure 4.12: Top: N250 Spinner dome outline versus fitted third-degree Bezier curve. Bottom: Spinner dome distribution fitting error, defined as the local difference between the reference data and the fitted distribution

Table 4.4: Summary of all defined spinner input parameters for the N250 propeller.

Variable	Unit	Value
Dome length	m	0.096906
Front length factor	-	0.0816544
Hub radius	m	0.044222
Hub length	m	0.02
Dome length control point ratio	-	0.1
Dome height control point ratio	-	0.51

Spinner geometry The spinner geometry of the N250 propeller is known, in the form of a discrete distribution describing the outline of the spinner. In the parametrization tool, the spinner consists of two distinct sections; the dome and the cylindrical hub. However, initial assessment of the available N250 data revealed that no clear cylindrical section is present in the data. Therefore, in order to model the spinner in the parametrization tool, the dome will be created based on the entire data set, and the hub is of a certain length, such that the blade hub is not sticking out.

In the parametrization tool, the dome is defined by five parameters, of which three can be directly derived from the available data and are given in table 4.4. The remaining two parameters control the position of the Bezier control points that define the dome shape. This requires a more detailed analysis of the shape, which is achieved by applying a fitting procedure. Since there are only two unknowns, a simple procedure is implemented that creates a two-dimensional data array of total errors, for varying dome height and length control point ratios, after which the minimum is selected. The total error is defined as the sum of the local differences between the reference data and the fitted data, normalized with the spinner hub radius. This results in an average error of 0.7% and a maximum error = 1.5%. A more detailed overview is given in figure 4.12.

Results

Figure 4.13 shows the overall propeller performance characteristics, for varying operating conditions and for three different blade pitch angles. Generally, the difference between the validation and the parametrized results is negligible.

There is one outlier at $J = 0.7$, with a blade pitch angle of 35 degrees, which itself shows a contradictory

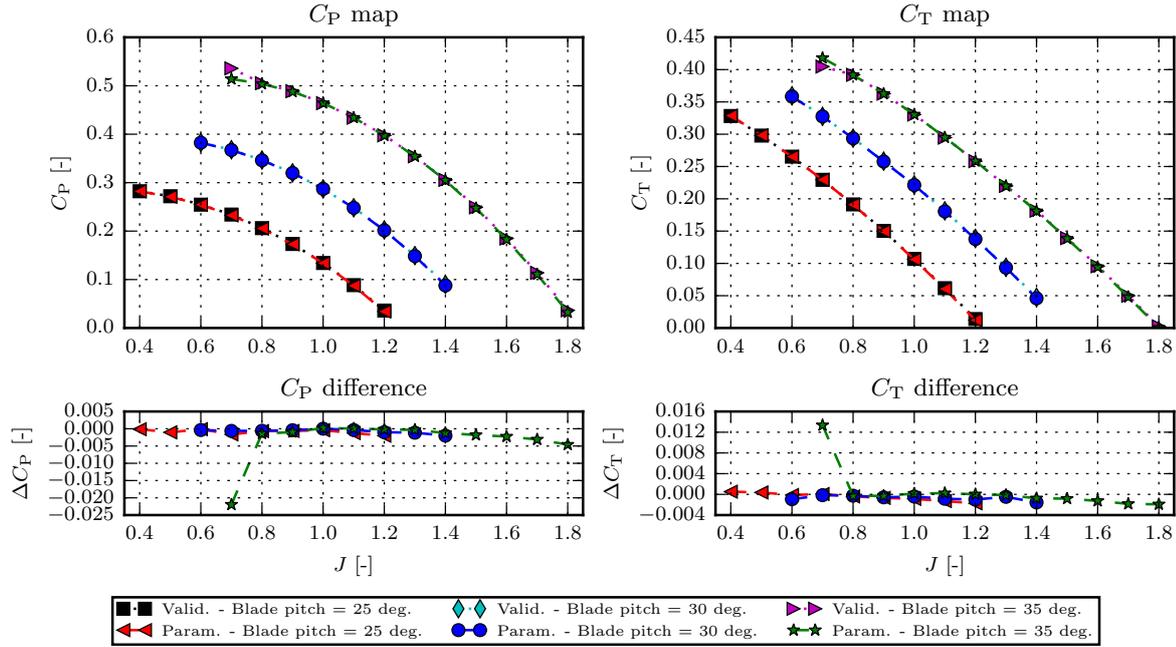


Figure 4.13: N250 performance map comparison between low-fidelity validation results and low-fidelity parametrized results.

trend. This is most likely caused by the fact that this point is in the stall region. In the validation results in chapter 2, it was shown that in the high-loading region, both the thrust and power is over-predicted. However, in the parametrized results, power is under-predicted compared to the validation results, which means it is closer to the true value. The thrust is over-predicted compared to the validation results, so the error becomes larger compared to the true value. It is unclear which of the two low-fidelity results is causing this behavior. In general, it can be stated that, while there might be a very small difference in geometry, it only concerns one data point. Furthermore, as stated in chapter 2, the low-fidelity performance method is sensitive to the distribution of coordinates describing each two-dimensional section. Compared to the validation model, the parametrization tool allows for better distribution of coordinates for the parametrized model, which should lead to better results.

Figure 4.14 shows the blade lift and drag distribution for $J = 1.0$ and a blade pitch angle of 30 degrees. Generally, the parametrized model matches the validation results very well and all the important trends match. The parametrized data starts at a slightly smaller radial coordinate, which is due to the blade-spinner sinking functionality described in sub-sub-section 4.1.2, however this has a negligible effect on the overall performance.

4.2.2. High-fidelity analysis verification

In order to successfully compare the initial and final design using CFD, the parametrization tool must be verified in combination with the CFD method, to ensure that, any conclusions that are made based on results from the CFD method are reliable. The setup of this part of the verification is very similar to the setup of the validation of the high-fidelity performance analysis method in chapter 3. The APIAN propeller is used as a reference case. Then, the CFD results of the validation study in chapter 3 are compared against CFD results which are generated using a parametrized model of the APIAN propeller. The remainder of this section is devoted to presenting the parametrization of the APIAN propeller and the verification results.

The validation results of chapter 3 are used as the reference data. Therefore, in order to minimize the possibility of uncertainties influencing the result, a mesh for the parametrized APIAN propeller is constructed with the same mesh parameters that were used in the validation study, which produced results for five different mesh fineness levels. This verification study is performed with just one mesh fineness level out of the five, more specifically, the “Fine” mesh of table 3.4. This mesh has a total number of cells of approximately 4.7 million cells and is selected because it produced accurate results in all aspects of the validation study, and there are no significant improvements when using the “Extra fine” mesh.

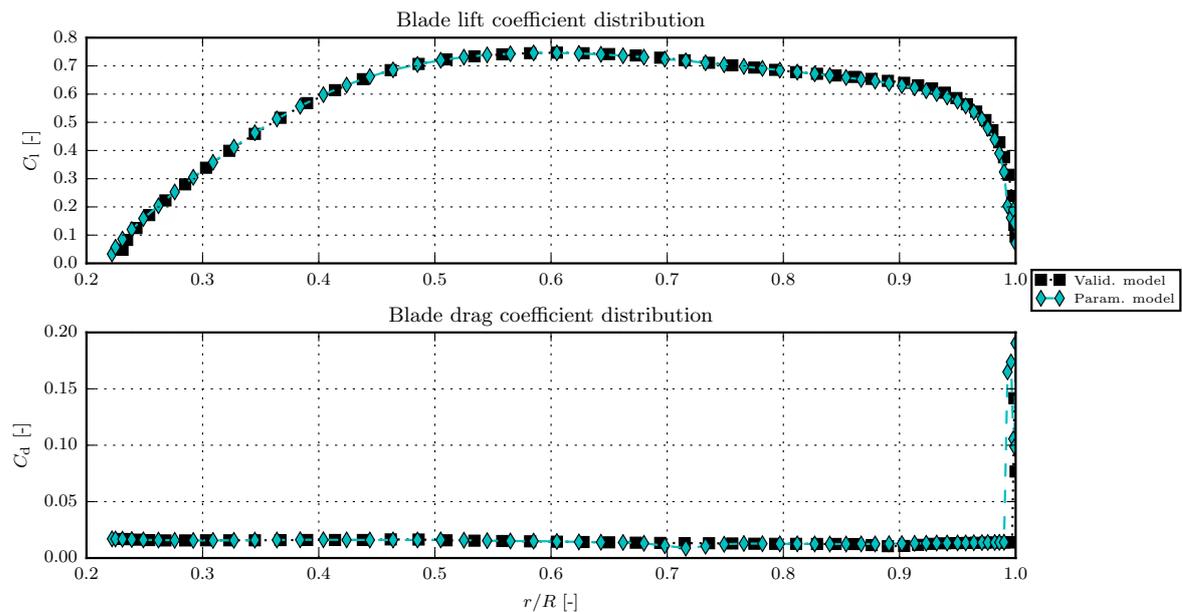


Figure 4.14: N250 lift & drag coefficient comparison between low-fidelity validation results and low-fidelity parametrized results, for $J = 1.0$ and the blade pitch angle is 30 degrees.

Table 4.5: Overview of APIAN general propeller parameters as required input for the parametrization tool.

Variable	Unit	Value
Number of blades	-	6
Rotation direction	-	CW
Blade hub radius	m	0.06095
Blade tip radius	m	0.25462

Parametrization of the APIAN propeller

The set of general propeller parameters can be determined by observing the propeller, without performing an extensive analysis. An overview of the general propeller parameters is given in table 4.5. The APIAN propeller geometry is given in the form of sectional data used to create the validation geometry model in chapter 3. They consist of discrete coordinate distributions, which need to be processed in order to derive all the missing parameters (i.e. blade twist, chord). The remainder of this section is devoted to deriving the remaining parameters. One limitation of the parametrization tool with respect to the APIAN propeller is that in the parametrization tool, all sections are parallel, meaning that the orientation of each section is only about one axis. This is not a problem for the majority of the APIAN propeller, however, the final tip section has an orientation that is about three axes, so the tip section is not parallel with the other sections. There is no solution for this, other than to neglect the final tip section. As it is not expected that the very top of the blade contributes much to the overall performance, neglecting the final tip section is deemed appropriate, but this needs to be confirmed with CFD verification.

Blade twist and chord distribution The pitch for a given section is defined in equation 4.6:

$$\phi_i = \tan^{-1} \left(\frac{x_{TE_i} - x_{LE_i}}{y_{LE_i} - y_{TE_i}} \right) - \beta_{0.75R}, \quad i = 1, \dots, n \quad (4.6)$$

To calculate the blade pitch angle, a smooth interpolation is performed, while setting the blade pitch angle to zero. Next, the assumption is made that at 75% of the radius, there is zero twist, which means that the preliminary twist at 75% of the radius is equal to the blade pitch angle, and thus equal to 40.4 degrees. Once known, the blade pitch angle is subtracted to get the blade twist distribution. For the parametrization tool, the discrete twist distribution is converted to a Bezier curve with a variable number of control points. The number of used control points is four, which is based on the desire for a small number of control points, while achieving a small error at the same time. In order to determine the actual control points, an optimization procedure is applied that finds the position of each control point by minimizing the total error, which is

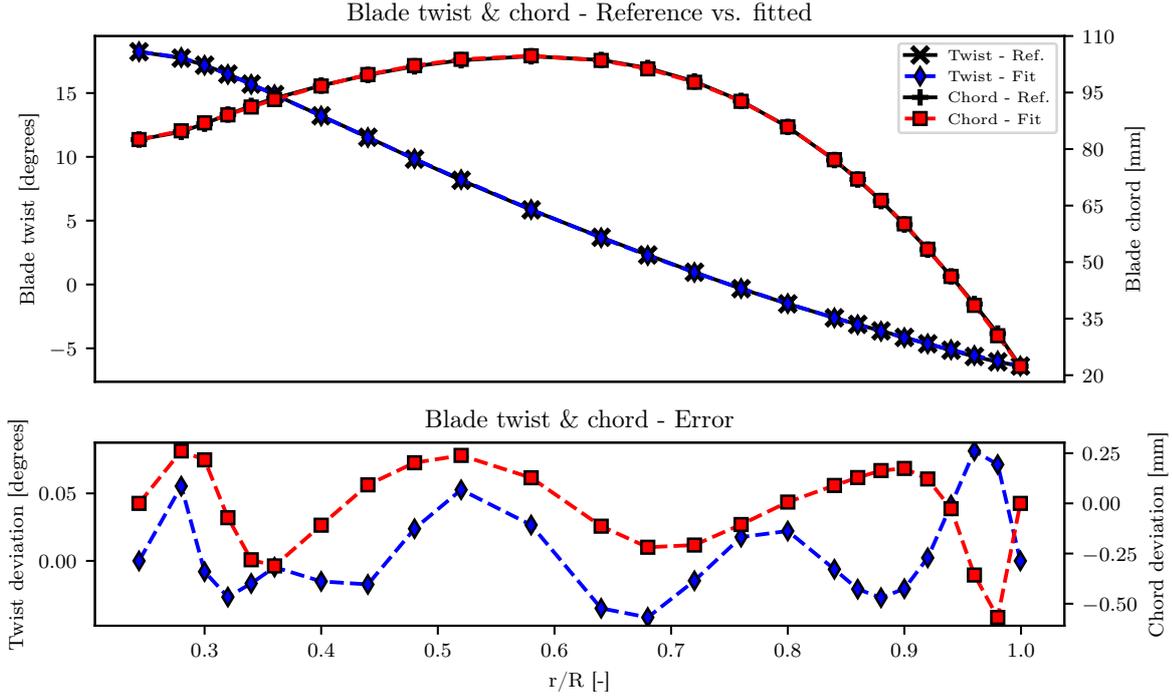


Figure 4.15: Top: APIAN propeller blade twist & chord distribution versus fitted third-degree Bezier curve. Bottom: Twist & chord error, defined as the local difference between the reference CAD model and the fitted distribution

defined as the sum of the absolute local differences between the reference and the fitted curve, normalized with the blade pitch angle. This procedure yielded an average error of 0.072% and a maximum local error of 0.22%, with a detailed overview available in figure 4.15.

A similar procedure is used to define the chord distribution as well. The chord at a given section is defined as:

$$c_i = \sqrt{(x_{TE_i} - x_{LE_i})^2 + (y_{TE_i} - y_{LE_i})^2}, \quad i = 1, \dots, n \quad (4.7)$$

For the parametrization tool, the discrete chord distribution is converted into a Bezier curve with a variable number of control points, using the same procedure as was used when defining the number of variable twist control points. For the chord, the appropriate number is six. The fitting procedure minimizes the total error, which is defined as the sum of the absolute local differences between the reference and the fitted curve, normalized with the blade mean chord. This procedure yielded an average error of 0.19% and a maximum local error of 0.65%, with a detailed overview available in figure 4.15.

Blade sweep and lean distribution The sweep and lean distribution are defined as the global lateral and longitudinal position of each section, respectively. The parametrization tool assumes the global center to be at the origin of the used axis system. Furthermore, it assumes that the local center of each section is at a point along the chord line at variable distance with respect to the leading edge, or local center point. Finally, the parametrization tool assumes that the first section at the hub has zero sweep and lean. This means that the local center point must coincide with the global radial axis. Initial assessment has made clear that the APIAN propeller does not uphold these assumptions. The chord line of the first section at the hub does not, at any point, coincide with the global radial axis, which means a compromise is needed. The proposed compromise assumes the propeller center axis to be at 50% of the chord line. This introduces small offsets in both directions which will be defined at the first section at the hub. For the sweep position distribution, an offset of 1.2468 mm is recorded. And for the lean position distribution, an offset of 1.3515 mm is recorded. When compensated for, the APIAN propeller does uphold the assumptions of the parametrization tool. When the section local center axis is defined at 50% of the chord line, the sweep and position distribution are defined in equation 4.8:

$$dy_i = 0.5c_i \cos(\phi_i + \beta_{0.75R}) + y_{TE_i}, \quad i = 1, \dots, n \quad (4.8a)$$

$$dx_i = 0.5c_i \sin(\phi_i + \beta_{0.75R}) - x_{TE_i}, \quad i = 1, \dots, n \quad (4.8b)$$

According to the parametrization tool sweep and lean definitions, they are parametrized as second-degree Bezier curves, each with three controlling parameters; the hub angles, the mid point radial height ratios and

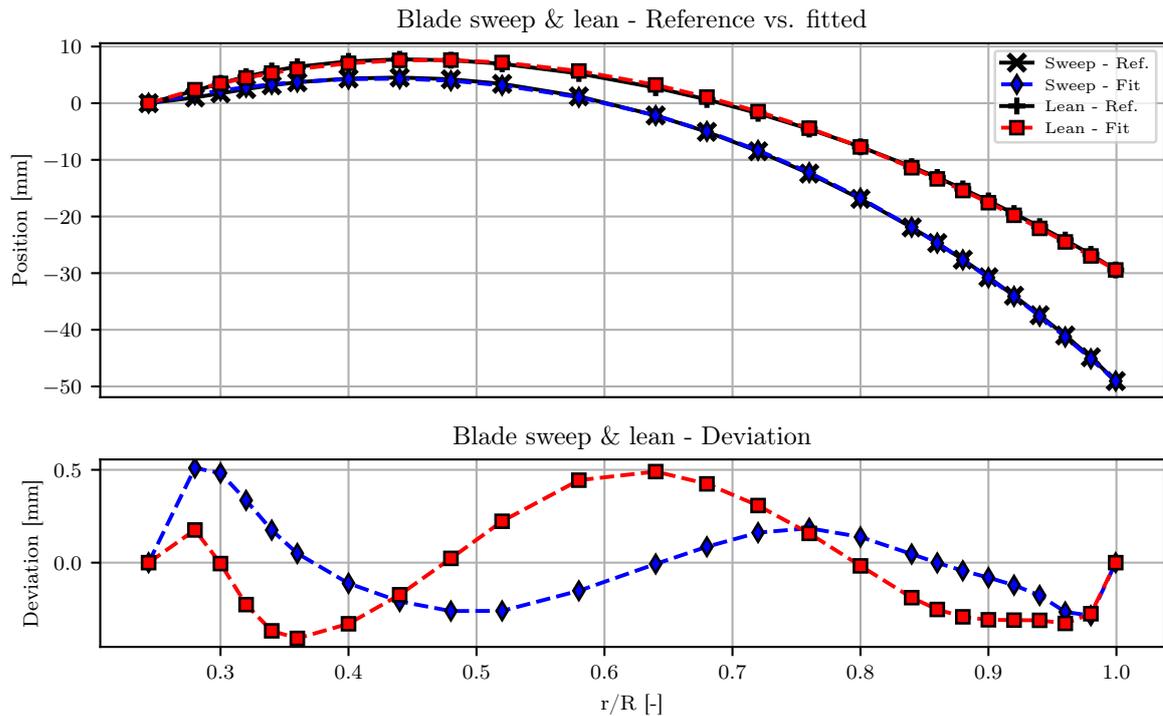


Figure 4.16: Top: APIAN propeller blade sweep and lean position distribution versus fitted second-degree Bezier curve, in lateral and longitudinal direction respectively. Bottom: Sweep and lean error, defined as the local difference between the reference CAD model and the fitted distribution.

Table 4.6: Overview of fitting errors of airfoils sections.

Section	Maximum error [%]	Average error [%]	Section	Maximum error [%]	Average error [%]
AA	0.0491	0.0110	OO	0.0564	0.0055
BB	0.0488	0.0089	SS	0.0297	0.0040
DD	0.0778	0.0059	VV	0.0357	0.0109
GG	0.0570	0.0056	XX	0.0301	0.0082
KK	0.0414	0.0071	YY	0.0648	0.0157

the tip angles. A simple procedure is applied, that finds the minimum total error, which is defined as the sum of the absolute local differences between the reference and the fitted curve. For the sweep, this results in an average error of 0.19% and a maximum local error of 0.59%. For the lean, this yields an average error of 0.28% and a maximum local error of 0.56%. A more detailed overview is given in figure 4.16. It demonstrates that, in order to minimize the total error, some characteristics are modeled less accurately.

Blade section shape definition The parametrization tool offers three distinct methods for defining section shapes. Since the section shapes are only available in the form of coordinates, the only applicable method is by defining a Bezier control point distribution. The decision is made to use ten out of the available twenty-five airfoils (labeled from “AA” at the hub to “YY” at the tip), which are distributed such, that more are positioned near the hub and the tip and less in the middle of the blade. This is necessary because it reduces the number of used parameters.

The coordinate distribution is split into separate distributions, describing the suction side, pressure side and trailing edge. The set of Bezier control points are found by applying a fitting procedure. Therefore, six control points are applied for the suction and pressure side. Once the suction and pressure side have been determined, a third optimization routine is implemented to determine the trailing edge curve, which is defined by a single parameter. Figure 4.17 gives the detailed results for one section. Table 4.6 gives an overview of the fitting errors for all sections.

Spinner geometry According to the parametrization tool, there are 6 parameters that control the outer shape of the spinner. Compared to the blade, it is a relatively simple part to model. The parametrization

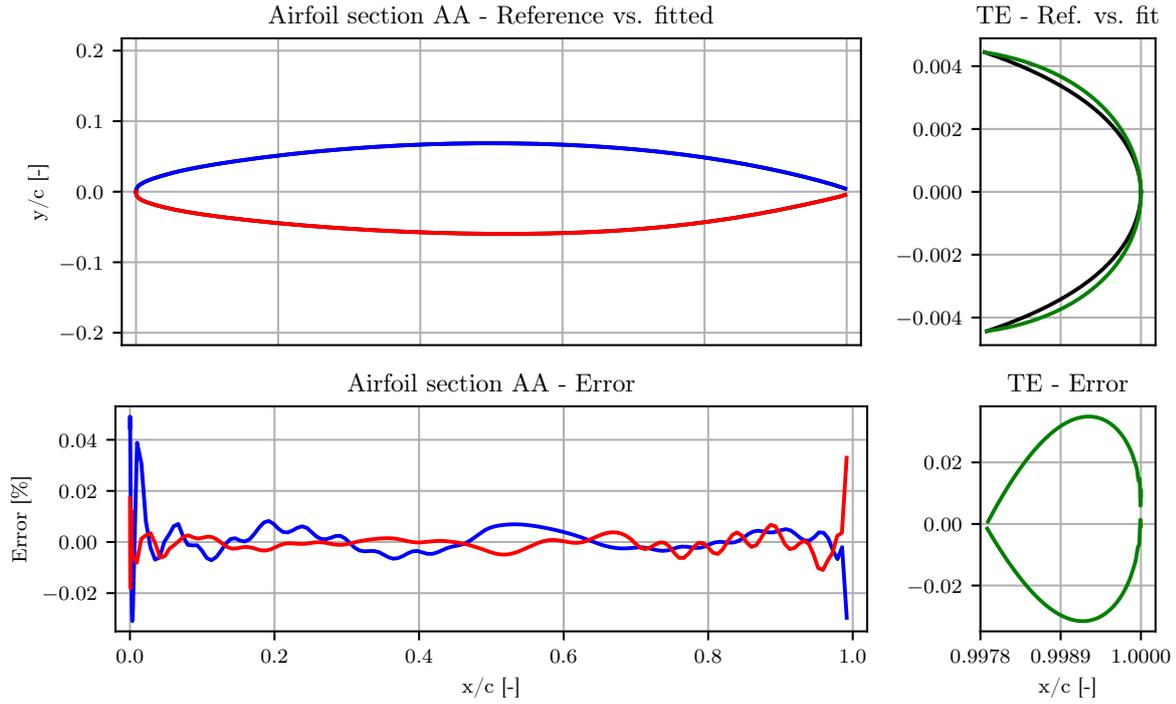


Figure 4.17: Top: APIAN propeller section AA airfoil versus fitted fifth-degree Bezier curve, both on suction and pressure side. Bottom: fitted error, defined as the local difference between the reference CAD model and the fitted distribution, normalized with the mean blade chord.

Table 4.7: Summary of all defined spinner input parameters for the APIAN propeller.

Variable	Unit	Value
Dome length	m	0.2000
Dome length control point ratio	-	0.8788
Dome height control point ratio	-	0.3512
Front length factor	-	1.000
Hub radius	m	0.06075
Hub length	m	0.049

tool defines the spinner as two objects; the dome and the hub. The hub parameters are easily deduced from the CAD model and are given in table 4.7. The dome is defined with a certain length, Bezier control points that control the outer shape and a certain position. This requires a more detailed analysis of the shape, which is achieved by applying another fitting procedure. Since there are only two unknowns, a simple procedure is implemented that creates a two-dimensional data array of total errors, for varying dome height and length control point ratios, after which the minimum is selected. The total error is defined as the sum of the local differences between the reference data and the fitted data, normalized with the spinner hub radius. This results in an average error of 1.02% and a maximum error = 2.44%. A more detailed overview is given in figure 4.18. The found dome parameters are given in table 4.7.

Blade thrust distribution definition

The blade thrust distribution was obtained by extracting sectional data for a discrete number of sections. These sectional data contain point distributions describing the section shape position, normal pressure data and shear stress data given in specific directions. These are used to determine the thrust at each section, using a standard integration method of pressure and shear stress distributions over an airfoil. The formal definition is given in equation 4.9 and the used conventions are defined in figure 4.19.

$$T_i = \int_{LE}^{TE} (F_{suc} p_{suc} \sin \theta + \tau_{suc_z}) ds + \int_{LE}^{TE} (F_{pres} p_{pres} \sin \theta + \tau_{pres_z}) ds \quad (4.9)$$

Since the shear stress in the streamwise direction is already known, the simplest approach is to determine the streamwise component of the pressure distribution ($p \cdot \sin \theta$), without undoing the global position and the

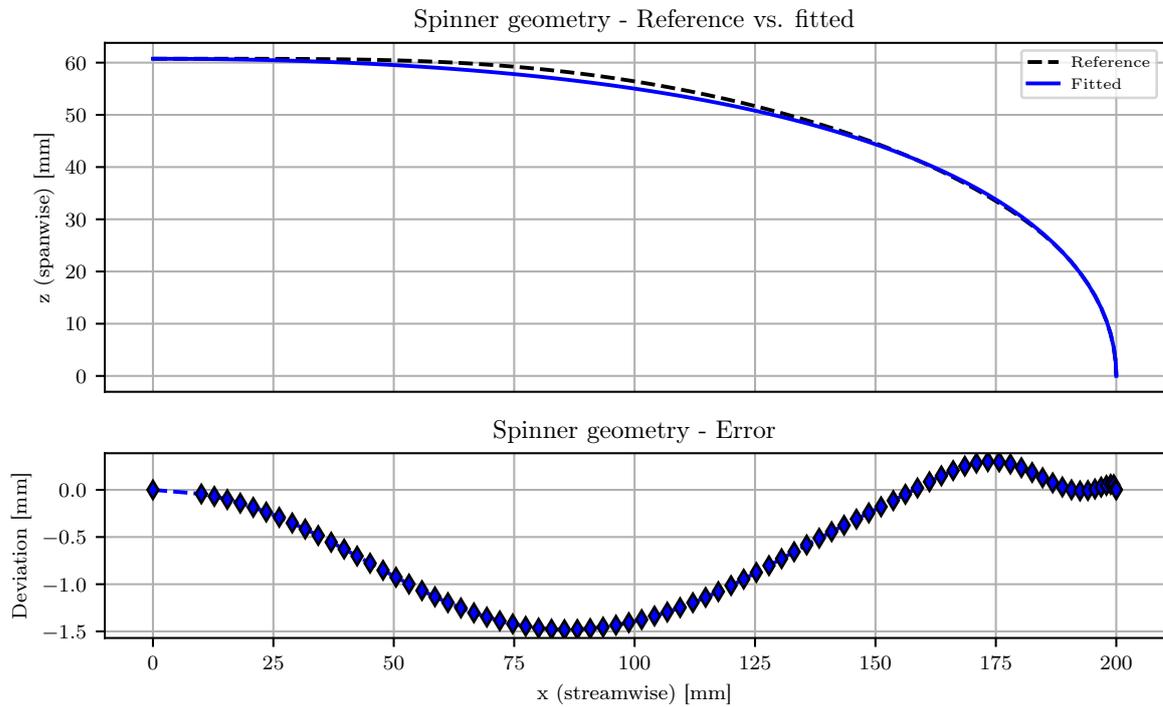


Figure 4.18: Top: APIAN spinner dome outline versus fitted third-degree Bezier curve. Bottom: Spinner error, defined as the local difference between the reference CAD model and the fitted distribution, normalized with the spinner hub radius.

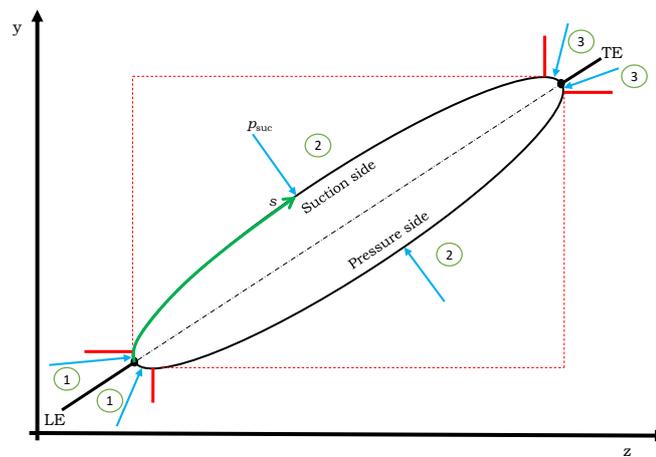


Figure 4.19: Definition of used conventions in determining the streamwise pressure component for the blade thrust distribution.

Table 4.8: Overview of applied factors to equation 4.9 based on the zones given in figure 4.19.

Zone	Suction side			Pressure side		
	pressure sign	θ angle sign	F	pressure sign	θ angle sign	F
1	negative	negative	-1	negative	negative	-1
2	negative	positive	+1	positive	negative	-1
3	positive	positive	+1	positive	positive	+1

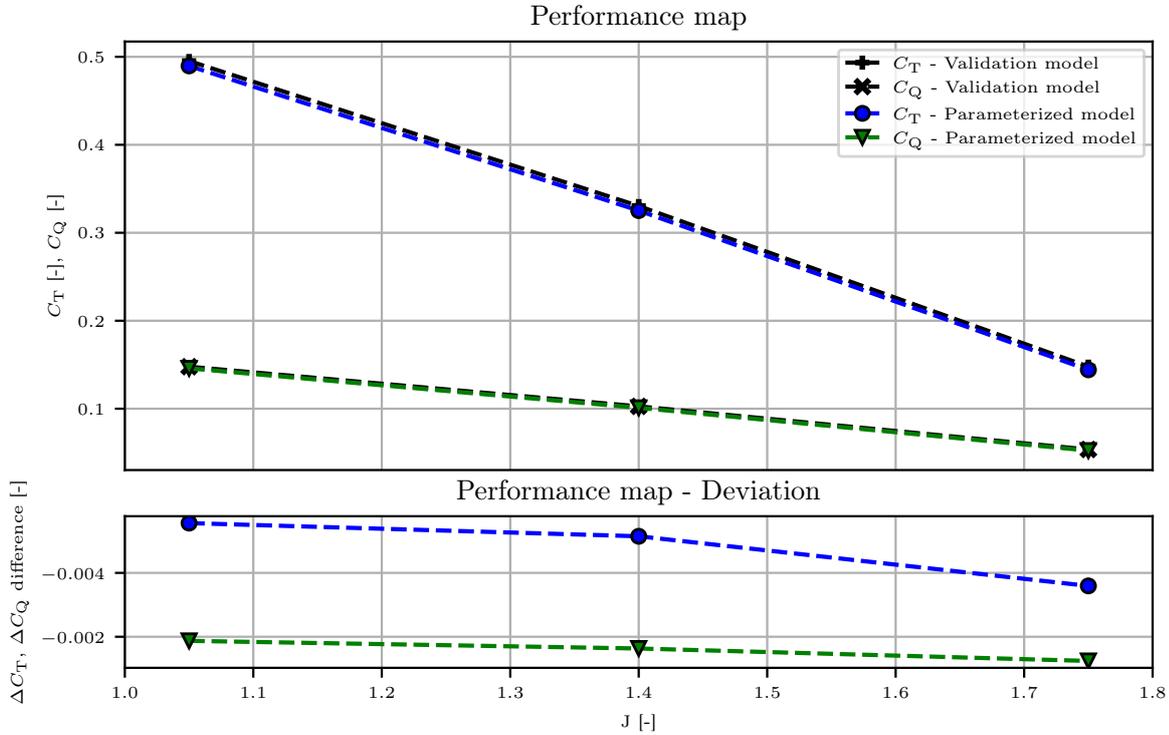


Figure 4.20: APIAN performance map comparison between CFD validation results and CFD parametrized results.

orientation of each section, as is common with this approach [26]. However, the method is not as straightforward compared to Anderson [26], due to the rounded leading and trailing edges of the APIAN propeller. That is why the standard approach is amended with the two factors F_{suc} and F_{pres} , so that the sign of each contribution switches, such that, dependent on the zone, the local pressure always correctly contributes to the total thrust of a blade section. The correct values of the two factors for each zone is given in table 4.8.

Results

Figure 4.20 shows the difference in overall performance characteristics, between the validation results and the parametrized model results, for three different operating conditions. Both the thrust and torque are slightly under-predicted throughout the operating range, compared to the validation CFD results. As in the validation study in chapter 3, the torque is more predicted with more precision than the thrust. In both the thrust and torque, the error decreases for increasing advance ratio, especially in the thrust from J is equal to 1.40 to 1.75. This trend is believed to be logical, because, as was demonstrated in the validation study, for increasing advance ratio the size and intensity of important complex flow structures, such as the leading edge vortex, decreases. This means that at the lowest advance ratio, achieving a converged solution is more difficult leading to bigger overall performance errors. This is backed up by the fact that the simulations for the two higher advance ratios converged significantly earlier.

Figure 4.21 gives the blade pressure distribution comparison, between the validation results and the parametrized model results, at three different sections, for $J = 1.05$, while in appendix B.1, blade pressure distributions are given for the remaining advance ratios. In general, all the parametrized model pressure distributions match the validation results very well. For $J = 1.05$ and $J = 1.40$, the most noticeable difference occurs at $r/R = 0.85$. Here, the value of the leading edge vortex pressure peak value is slightly under-predicted, while its position is slightly over-predicted. One possible explanation is that the curvature of the suction blade

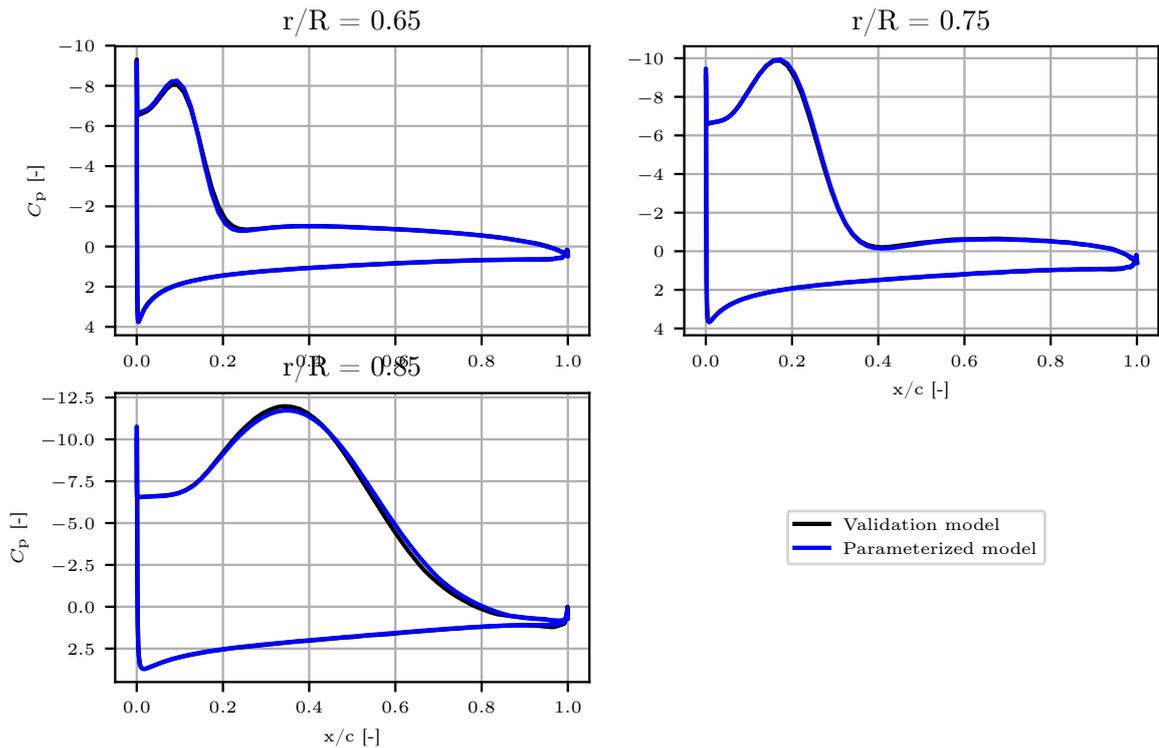


Figure 4.21: Blade pressure distribution comparison between CFD validation results and CFD parametrized results for $J = 1.05$, at three different blade sections.

face is more flattened in the parametrized case. Finally, there are no significant differences in the results for $J = 1.75$, due to lack of significant complex flow structures that could effect the results.

Figure 4.22 shows the blade thrust distribution comparison, between the validation results and the parametrized model results. The tip geometry of the parametrized model differs from the validation model, which means that the top section used here is at a maximum spanwise position where, in both cases, a complete airfoil section can be extracted. The results can be verified by integrating the blade thrust distribution and determining the propeller thrust coefficient. For the validation results, at $J = 1.05$, the integrated propeller thrust coefficient deviates 0.00566 from the propeller thrust coefficient determined by SU2. The deviation for the remaining operating points and the parametrized model results are of the same order. In figure 4.22, for $J = 1.05$ and 1.40, from the blade hub upwards, a steady increase is noticed, after which the thrust drops with a significant gradient. At $J = 1.75$, there is a maximum, but the change in slope is more gradual. The thrust at the maximum available spanwise position is not yet zero, however an interesting trend is visible regarding the maximum spanwise position. For increasing advance ratio, the ratio between the thrust at the maximum spanwise position and the maximum thrust increases. This is most likely caused by the fact that for low advance ratios the leading edge vortex is considerable in size and spills over the tip of the blade. This causes a bigger loss in thrust compared to high advance ratios, where the leading edge vortex is significantly smaller in size. In general, the difference between the validation results and the parametrized model results are small. The differences are most noticeable in the tip region, although a difference in tip geometry also affects the result.

Figure 4.23 gives comparison of the spinner pressure distribution, between the validation results and the parametrized model results, for $J = 1.05$. while in appendix B.2 the same results for the remaining operating conditions are given. There are two significant differences noticeable throughout the operating range, between the validation results and the parametrized model results. First, there is a slight reduction in pressure in the area ahead of the blade. The difference is most likely caused, by the difference in spinner curvature between the validation model and the parametrized model. The slope of the parametrized model is more gentle in this area, resulting in a reduction in pressure. The second noticeable difference is the position of the periodic flow features near the blade, especially for $J = 1.40$ and 1.75. A shift in positive y -direction is noticeable and is most likely caused by the difference in blade hub section position. as discussed in sub-sub-section 4.2.2, in order to match the APIAN propeller geometry with the parametrization tool definitions, an offset in blade position with respect to the propeller plane was neglected.

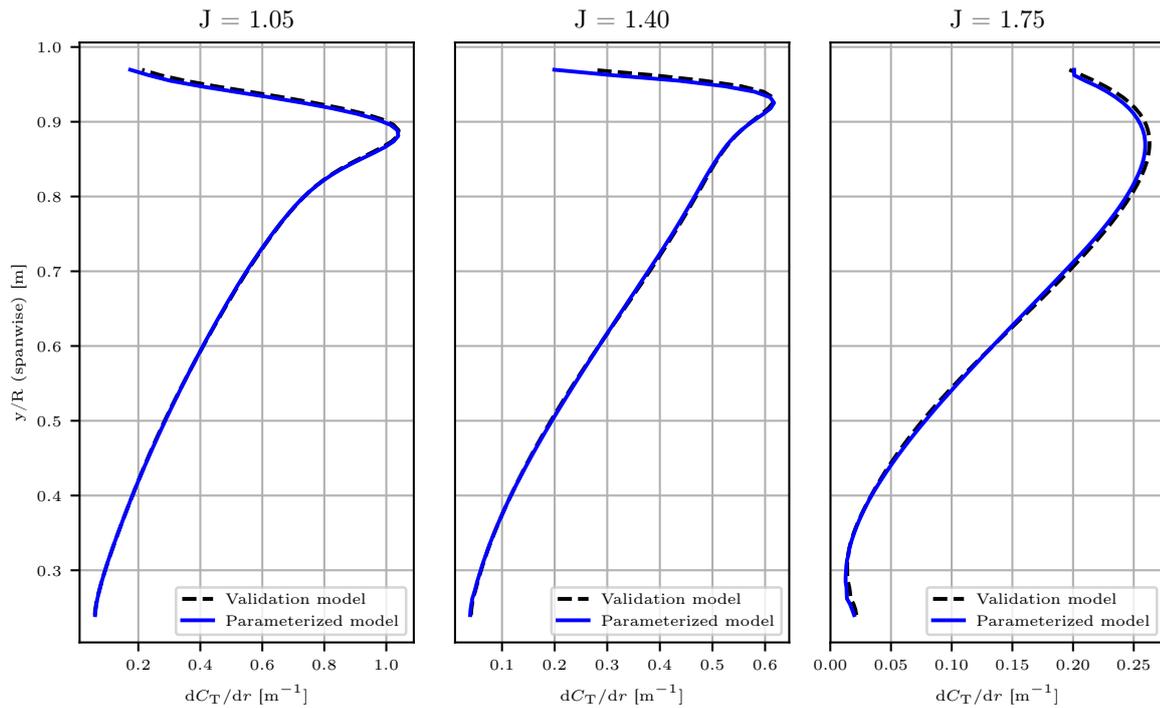


Figure 4.22: APIAN blade thrust distribution comparison between CFD validation results and CFD parameterized results. The black line represents the validation results and the blue line represents the parameterized results.

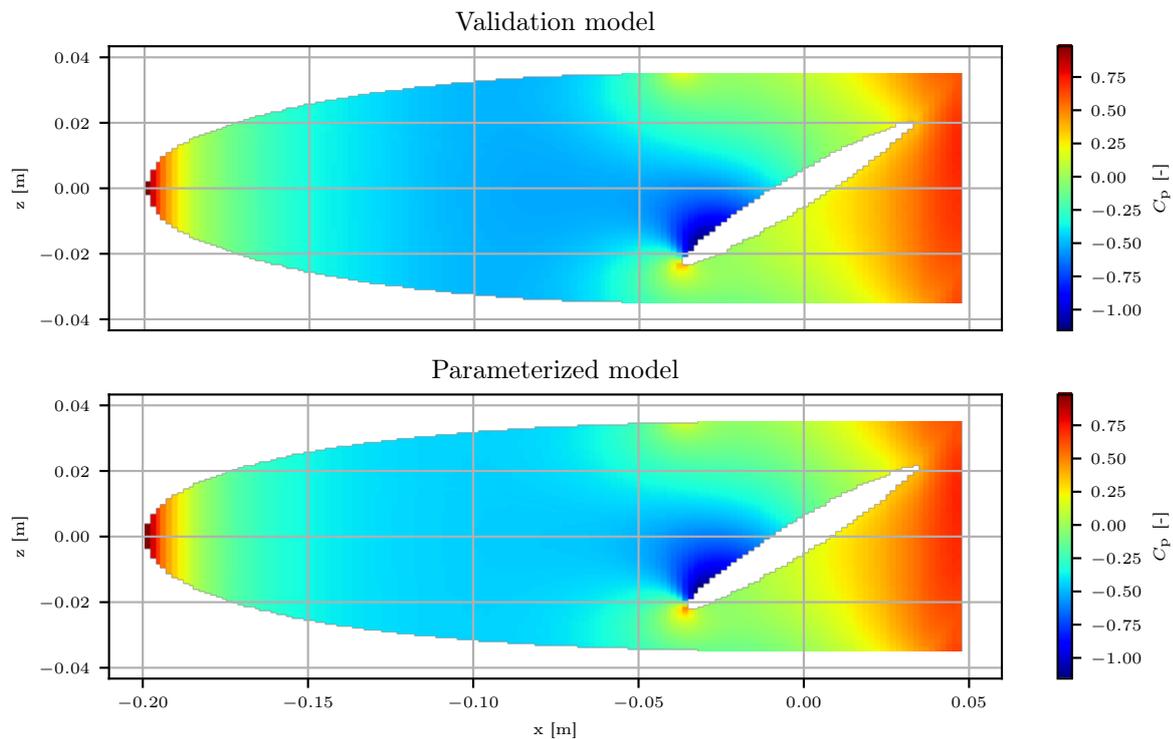


Figure 4.23: Spinner pressure distribution comparison between CFD validation result and CFD parameterized result for $J = 1.05$.

Figure 4.24 and appendix B.3 show the propeller wake velocity comparison, between the validation results and the parameterized model results. In general, the validation results and the parameterized model results are in good agreement. The positions of all propeller wake flow features align perfectly and there is only a very small difference in intensity. The parameterized model results only slightly under-predict the wake velocity intensity, but this is really insignificant when considering the overall performance.

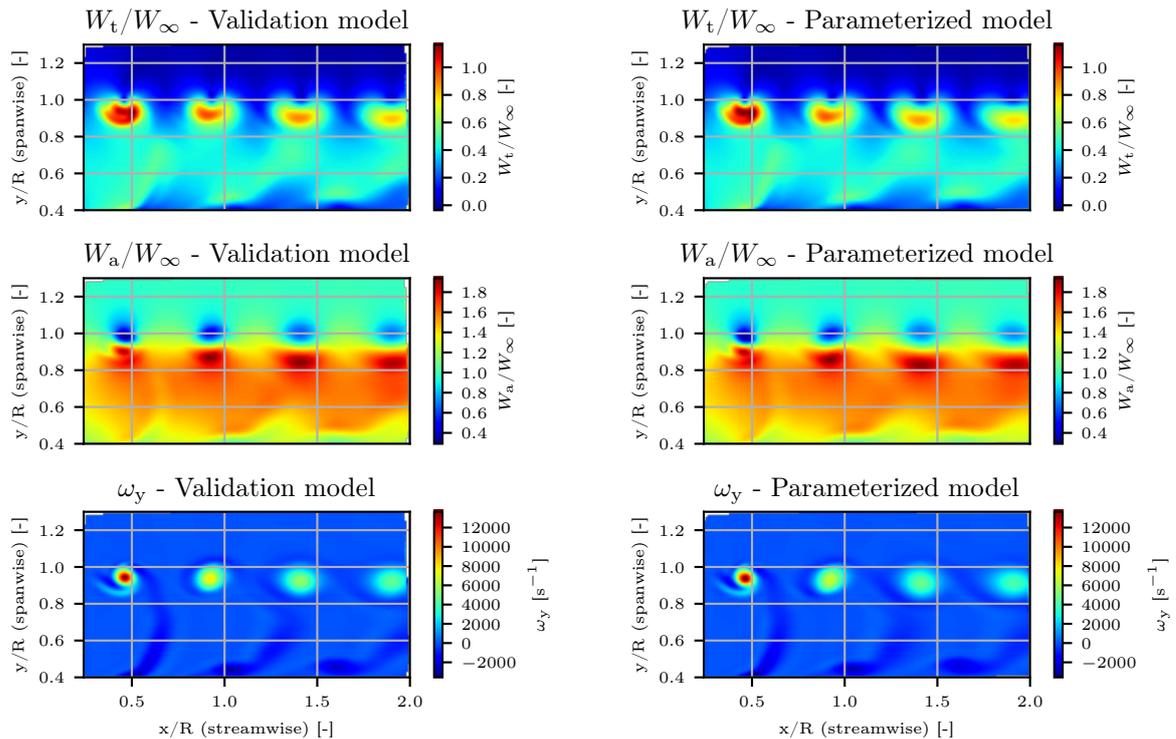


Figure 4.24: APIAN propeller wake velocity comparison between CFD validation result and CFD parametrized result for $J = 1.05$.

4.2.3. Discussion of verification results

It has been demonstrated that it is possible to perform reliable low- and high-fidelity performance analyzes in combination with the parametrization tool. It is possible to re-create complex propeller geometries using the parametrization tool.

In the case of the low-fidelity analysis verification, the structure of the parametrization tool is very much suited to the low-fidelity performance analysis method process, which makes it very easy to use in the context of an optimization routine. For the selected verification case, no significant compromises were required in order to be able to model the propeller geometry, which is backed-up by the fact that the performance deficit is insignificant.

The selected verification case in the high-fidelity analysis verification was a real test case for the capabilities of the parametrization tool. The APIAN propeller has a very complex geometry that, for the most part, could be modeled with the parametrization tool. The limits of the tool were reached in two specific areas. First, the original blade geometry has a position offset with respect to the spinner, which cannot be accounted for in the tool. Second, the blade tip section of the original blade geometry has a three-dimensional orientation, whereas in the tool, only a one-dimensional rotation is applied. For this verification both were neglected, but still only a minimal performance deficit is achieved.

5

Propeller blade optimization for minimized installation effects - Setup

In the introduction in chapter 1, the main goal of this thesis is outlined. This is to study the potential benefit on propeller performance of considering installation effects during the conceptual design phase. In the previous chapters, low- and high-fidelity isolated propeller performance analysis methods are applied, as well as a propeller parametric modeling tool. These will be used to achieve the main goal of this thesis, by combining them and the installed performance model by implemented by Sinnige & Veldhuis [13], in an optimization routine with the objective to minimize unwanted unsteady installation effects.

Unfortunately, it is not possible to perform the optimization routine with the high-fidelity isolated performance analysis method. This is due to time constraints and the fact that with the current manual setup of the high-fidelity method, it is not possible to implement an automated routine. Therefore, the optimization routine will consist of the parametrization tool (chapter 4), the low-fidelity isolated performance method (chapter 2) and the installed performance model. The high-fidelity isolated performance method is used to perform an isolated analysis of the baseline propeller, for the purpose of verifying the precision of the low-fidelity isolated performance method. The use of the low-fidelity isolated performance method introduces an additional uncertainty about its accuracy, because it is only applicable to unswept, no-lean propeller geometries. This is the case because it does not account for associated effects.

The setup of the installation effects optimization is split into two parts. The first part consists of defining the implementation of the installed performance model implemented by Sinnige & Veldhuis [13], of which a summary is given in section 5.1. The second part consists of defining an optimization problem and strategy, which is discussed in section 5.2.

5.1. Installed performance model

Throughout this research, one particular installed configuration is used, a propeller behind a pylon, as shown in figure 1.3. Based on the scope of this thesis and the availability of existing models in literature, it has been decided to use the model of Sinnige & Veldhuis [13]. They investigated the influence of the installed configuration on propeller performance and the effects of pylon blowing. The original model of Sinnige consists of three modules, one to determine isolated propeller performance, one to determine installed propeller performance and one to determine the propeller noise emissions. The isolated performance is determined using the low-fidelity performance analysis method of chapter 2. Noise emissions are not within the scope of this thesis, therefore this part of Sinnige's model will be ignored. This only leaves the installed performance module, which is described in the remainder of this section.

The isolated performance of the propeller is modeled with an undisturbed flow. Introducing a pylon upstream affects the propeller inflow and thus the installed performance. In order to assess the installed effects it is required to model the pylon wake to determine the profile of the disturbed propeller inflow. The pylon wake model is then used as input for determining the installation effects experienced by the propeller.

5.1.1. Pylon wake model

For simplicity, the pylon is modeled as a simple straight wing section. In order to model the pylon wake, a two-dimensional Schlichting wake profile is applied, in order to determine the velocity deficit in the pylon

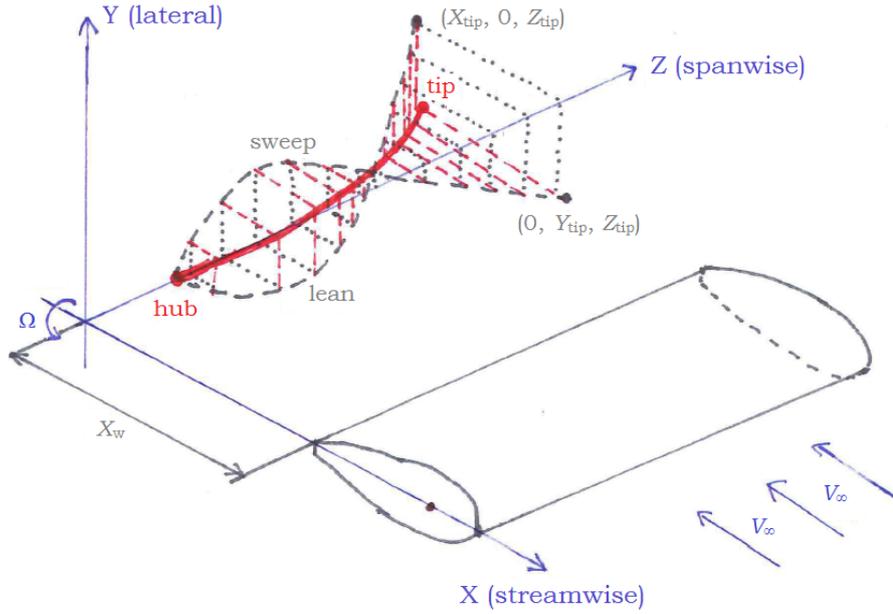


Figure 5.1: Graphic representation of installed axis system. Red represents the three-dimensional blade center axis, while the gray dashed lines are two-dimensional projections of the three-dimensional blade center axis on the Y-Z plane (sweep) and the X-Z plane (lean).

wake [13] [33], which is given in equation 5.1:

$$\frac{\Delta u}{U_\infty}(X_W, Y_W) = \frac{\sqrt{10}}{18\beta} \sqrt{\frac{c_d c}{X_W}} \left[1 - \left| \frac{Y_W}{b_W} \right|^{\frac{3}{2}} \right]^2 \quad (5.1a)$$

$$b_W(X_W) = \beta \sqrt{10 c_d c X_W} \quad (5.1b)$$

Where β is an empirical constant fixed at 0.18 and b_W is the wake semi-width. In his original thesis [13], Sinnige assumes that the pylon wake is not affected by the presence of the propeller and that the two-dimensional wake profile is applicable to the finite pylon. Since then, Sinnige [5] has demonstrated that this assumption is incorrect in the real world. While this is the case, it is not within the scope of this thesis to amend the installed performance model, so the assumption is upheld. This means that the pylon wake profile can be determined when enough pylon characteristics are supplied, namely c_d the pylon two-dimensional drag coefficient, c , the pylon chord length, X_W the pylon-propeller spacing, for varying Y_W , the vertical position in the wake. The two-dimensional wake profile is repeated from zero radius to the propeller radius, and then converted to a polar coordinate system which ranges from 0 to 360 degrees. The pylon two-dimensional drag coefficient is determined with a straightforward Xfoil analysis [12].

The model Sinnige proposed was originally created with straight, unswept propeller blades in mind. However throughout this thesis, complex blade shapes are used that allow for blade sweep and blade lean. Blade sweep affects the timing of when a particular part of the blade passes through the pylon wake, while blade lean affects the pylon-propeller spacing, in that, it is no longer constant but varies as a function of the blade lean. Figure 5.1 gives an overview of the used axis definition.

Blade lean effects are introduced by amending the definition of the Schlichting wake profile definition in equation 5.1, by expanding the definition of the pylon-propeller spacing:

$$X_{W_i}(X_{blade_i}) = X_{W_0} - X_{blade_i} \quad (5.2)$$

Based on this equation and the Schlichting wake profile in equation 5.1, it is expected that forward lean (positive X_{blade_i}) will decrease the wake semi-width b_W and increase the maximum wake velocity deficit. Backward lean (negative X_{blade_i}) does the exact opposite.

In order to include blade sweep effects, it is required to determine the polar angle shift of when each blade section passes through the pylon wake. Since blade sweep depends on the propeller rotation direction, it has implications for the polar angle shift as well. For counter-clockwise rotation (as defined in the parametrization tool):

$$\phi_{W_i}(Y_{blade_i}, Z_{blade_i}) = \phi_{W_0} + \arctan\left(\frac{Y_{blade_i}}{-Z_{blade_i}}\right) \quad (5.3)$$

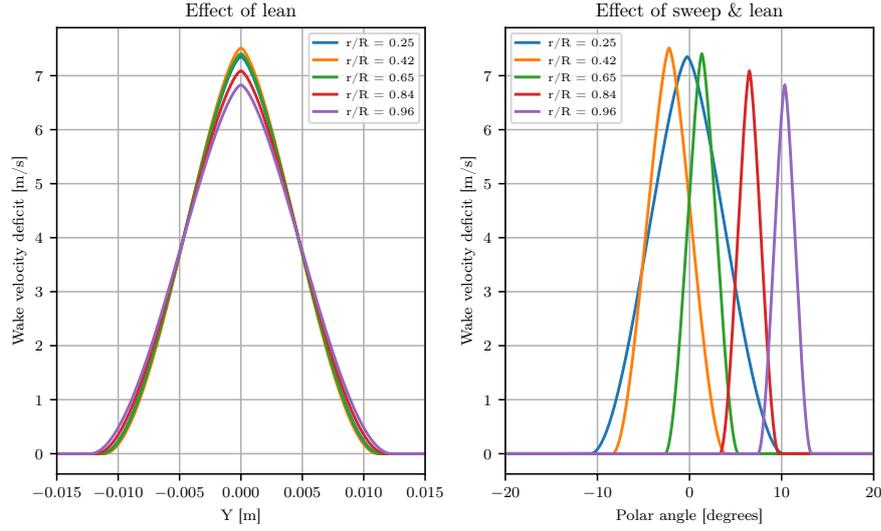


Figure 5.2: Effect of lean (left) and sweep (right) implementation on wake velocity deficit at different sections. The APIAN propeller geometry is used, which has both forward sweep and lean in the hub region, and backward sweep and lean in the tip region.

For clockwise-rotation:

$$\phi_{W_i}(Y_{\text{blade}_i}, Z_{\text{blade}_i}) = \phi_{W_0} - \arctan\left(\frac{Y_{\text{blade}_i}}{-Z_{\text{blade}_i}}\right) \quad (5.4)$$

The right side of figure 5.2 shows the effect of sweep on the phase of the velocity deficit experienced by the propeller blade.

5.1.2. Modeling installed effects

The approach of Sinnige to model installation effects is similar to the approach used in a blade element method. In a blade element method, thrust and torque of a particular blade section are based on the lift and drag properties of the same section:

$$T_i = \frac{1}{2} \rho_\infty W_i^2 (c_{l_i} \cos \phi_i - c_{d_i} \sin \phi_i) c_i \Delta r_i \quad (5.5a)$$

$$Q_i / r_i = \frac{1}{2} \rho_\infty W_i^2 (c_{l_i} \sin \phi_i + c_{d_i} \cos \phi_i) c_i \Delta r_i \quad (5.5b)$$

Installation effects are characterized as changes in the blade loads due to the presence of a non-uniform flow field, which occurs due to the presence of the pylon and its wake. Sinnige assumes that the final installed blade loading can be computed by following the principle of superposition [13], and identifies two separate contributions besides the isolated performance. First, is the change in blade loads due to the change in dynamic pressure in the pylon wake region, assuming zero change in angle of attack relative to the isolated case. Second, is the change in blade loads due to the change in angle of attack in the pylon wake region, assuming zero change in the dynamic pressure relative to the isolated case. The installed effects are then collected in the drag and lift terms of equation 5.5:

$$c_{l_i} = c_{l_i}^{\text{isolated}} + (c_{l_i}^{\Delta Q} + c_{l_i}^{\Delta \alpha})^{\text{installed}} \quad (5.6a)$$

$$c_{d_i} = c_{d_i}^{\text{isolated}} + (c_{d_i}^{\Delta Q} + c_{d_i}^{\Delta \alpha})^{\text{installed}} \quad (5.6b)$$

Blade loads due to change in dynamic pressure

Sinnige characterizes a change in dynamic pressure as a change in the local effective velocity [13]. In the isolated case, the local effective velocity is a vector based on the free-stream, rotational and induced velocities. By introducing a non-uniform inflow, the velocity deficit in the wake must be accounted for:

$$W_{\text{installed}} = \sqrt{(U_\infty + u_{\text{inda}} + \Delta u)^2 + (\Omega r - u_{\text{ind}_t})^2} \quad (5.7)$$

As mentioned earlier, due to the application of superposition, performance characteristics are treated separately. Therefore, Sinnige argues that it is safe to assume that isolated performance characteristics are not

influenced by a change in dynamic pressure [13]. Therefore the lift and drag contribution due to a change in dynamic pressure can be determined with:

$$c_l^{\Delta q} = c_l^{\text{isolated}} \left(\frac{W_{\text{installed}}^2}{W_{\text{isolated}}^2} - 1 \right) \quad (5.8a)$$

$$c_d^{\Delta q} = c_d^{\text{isolated}} \left(\frac{W_{\text{installed}}^2}{W_{\text{isolated}}^2} - 1 \right) \quad (5.8b)$$

Blade loads due to change in angle of attack

When imagining a typical velocity diagram for a given 2-dimensional blade section, it is clear that due to introducing a non-uniform inflow, the section experiences an angle of attack based on the orientation of the velocity vector. And a change in angle of attack affects the local lift and drag production. In order to model this change, Sinnige proposes to compute this variation using Sears' method [13]. Furthermore, Sinnige has chosen to neglect the drag contribution.

First, Sears' method works by transforming the calculated velocity deficit profile into a periodic gust normal to the propeller blades. It is assumed that the gust velocity is in the direction normal to the blade chord. Next, Sears' method moves to the frequency domain. In order to accomplish this the periodic normal gust is redefined as a complex Fourier series, such that Fourier coefficients can be found through a fast Fourier transform [13]:

$$v_{g_{\text{normal}k}} = \frac{1}{2\pi} \int_0^{2\pi} V_{g_{\text{normal}}} e^{-ik\phi} d\phi \quad (5.9)$$

where k is the harmonic order. Because the periodic normal gust is now transformed to the frequency domain, it is possible to model the change in blade loads due to change in angle of attack.

$$c_{lk}^{\Delta\alpha} = 2\pi \frac{v_{g_{\text{normal}k}}}{W_{\text{isolated}}} S \quad (5.10)$$

Sinnige argues that for S , which is an aerodynamic transfer function, the theory of Sears is particular useful and can be expressed in terms of the Sears function, which is given in equation 5.11:

$$S(\sigma, M_{\text{iso}}) = \frac{S_{M=0}(\sigma \beta_{\text{iso}}^2)}{\beta_{\text{iso}}} \left[J_0 \left(\frac{M_{\text{iso}}^2 \sigma}{\beta_{\text{iso}}^2} \right) + i \cdot J_1 \left(\frac{M_{\text{iso}}^2 \sigma}{\beta_{\text{iso}}^2} \right) \right] \cdot e^{-\frac{i\sigma f(M_{\text{iso}})}{\beta_{\text{iso}}^2}} \quad (5.11)$$

Where σ is the reduced frequency, M_{iso} is the isolated helical Mach number, β_{iso} the Prandtl-Glauert compressibility factor based on the isolated helical Mach number, J_0 and J_1 the zeroth- and first-order Bessel functions of the first kind, and:

$$\sigma = \frac{k\Omega c}{2W_{\text{iso}}} \quad (5.12a)$$

$$S_{M=0}(\sigma) = \frac{J_0(\sigma)K_1(i\sigma) + iJ_1(\sigma)K_0(i\sigma)}{K_1(i\sigma) + K_0(i\sigma)} \quad (5.12b)$$

$$f(M_{\text{iso}}) = (1 - \beta_{\text{iso}})\ln(M_{\text{iso}}) + \beta_{\text{iso}}\ln(1 + \beta_{\text{iso}}) - \ln(2) \quad (5.12c)$$

With K_0 and K_1 the modified zeroth- and first-order Bessel functions of the second kind. A Sears function vector diagram in the complex plane for ranging reduced frequencies, is presented in figure 5.3. It demonstrates the behavior of the Sears function. The final step involves transforming the blade loads due to change in angle of attack back to the time domain, by applying an inverse Fourier transform:

$$c_l^{\Delta\alpha} = \sum_{k=-\infty}^{\infty} c_{lk}^{\Delta\alpha} e^{ik\phi} \quad (5.13)$$

5.1.3. Example of installation effects

Before the optimization problem is defined, it is useful to display the typical installation effects computed using the model of Sinnige & Veldhuis [13] given in the previous sub-section. This gives a more complete overview on the size of the problem that needs to be solved, as well as a better understanding of the behavior of these installation effects. The results in this sub-section are generated with the APIAN propeller at an advance ratio of 1.05 and the pylon characteristics of table 5.1. In figure 5.4, the two-dimensional installed lift and drag coefficients are plotted for one single blade, at different radii. The results are split into individual

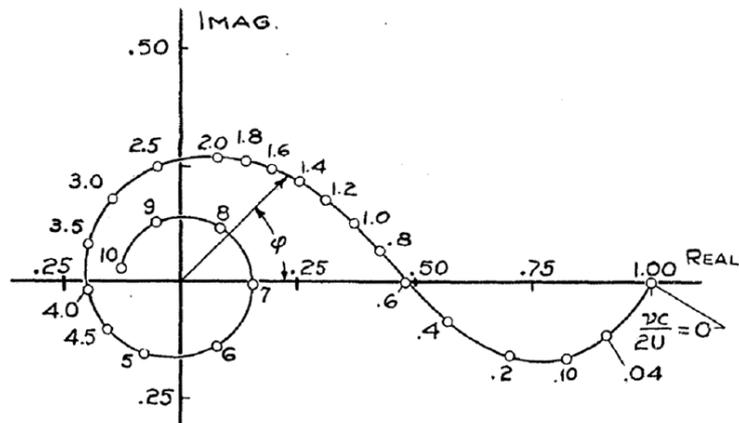


Figure 5.3: Sears function vector diagram in the complex plane for reduced frequencies ranging from zero up to and including ten, from Sinnige & Veldhuis [13]

contributions due to change in dynamic pressure and change in angle of attack. the lift and drag contributions related to a change in dynamic pressure have the same phase shift for each blade station. This phase shift is caused by the sweep of the propeller. The same phase shift also occurs in the lift contribution related to a change in angle of attack, however, the Sears function also influences the phase shift, which makes it hard to distinguish each effect individually. It will be the same phase shift as for the contributions related to a change in dynamic pressure. According to Sinnige & Veldhuis [13], The phase shift behavior due to the Sears function is dependent on the sign and value of the imaginary term demonstrated in figure 5.3. A negative imaginary component results in a negative phase shift, while a positive imaginary component results in a positive phase shift. Near the hub, the reduced frequency is higher than the rest of the blade due to the low effective isolated velocity.

In figure 5.5 the installed propeller thrust and torque performance are shown. Here, all the lift and drag contributions are re-written into thrust and drag contribution, which are then integrated for all the blades, to get the final installed propeller thrust and torque installation effects. This particular propeller has six blades, of which you see evidence of, in the fact that for each blade passing the pylon wake, a response is visible. As expected, the contributions due to a change in dynamic pressure reduce the overall installed performance, while a change in angle of attack enhances it. The amplitude of the installation effects is for this case small. This is a high loading case for which installation effects are expected to have a smaller impact. The value of the velocity deficit peak is only dependent on the pylon geometry, however, the installed lift and drag contributions are dependent on the ratio between the installed and isolated effective velocity. The influence of the velocity deficit on a large isolated effective velocity is small, compared to a low loading case. Then, the isolated effective velocity is small so the influence of the same velocity deficit is large.

5.2. Definition of optimization problem

The ultimate goal of the optimization is to minimize undesired propeller installation effects, by only changing the propeller blade geometry. In order to achieve this, the optimization setup and strategy is defined in this section. The optimization setup consists of selecting a reference baseline case, formally defining the optimization problem and implementation of the optimization.

5.2.1. Reference baseline case

For the optimization setup, first a relevant baseline case is required. This includes the pylon geometry, propeller geometry and operating conditions. It is convenient to select a relevant case, of which performance characteristics are known. Moreover, due to low-speed limitations of the low-fidelity isolated performance method, the performance characteristics of the isolated propeller must be at low-speed operating conditions.

Based on this, the APIAN propeller is deemed appropriate because of the fact that performance characteristics are available both from external experiments and the CFD simulation from chapter 3. Because of this, there is no need for an additional high-fidelity analysis of the baseline propeller, which saves valuable time. The pylon geometry is taken from Sinnige et al. [5], who performed wind-tunnel experiments investigating installation effects with the APIAN propeller. An overview of the pylon geometry is given in table 5.1. The operating conditions are dictated by the available performance characteristics from the propeller. In this

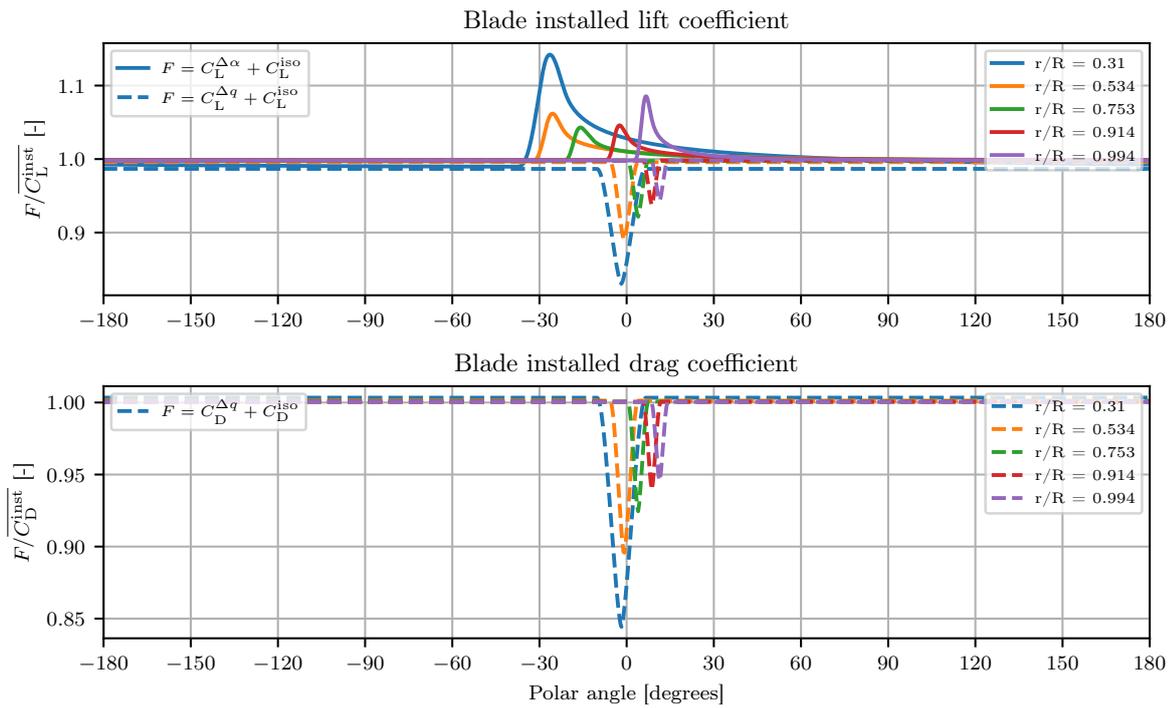


Figure 5.4: Two-dimensional blade installed lift and drag coefficient contributions at different radii, with the APIAN propeller, at $J = 1.05$.

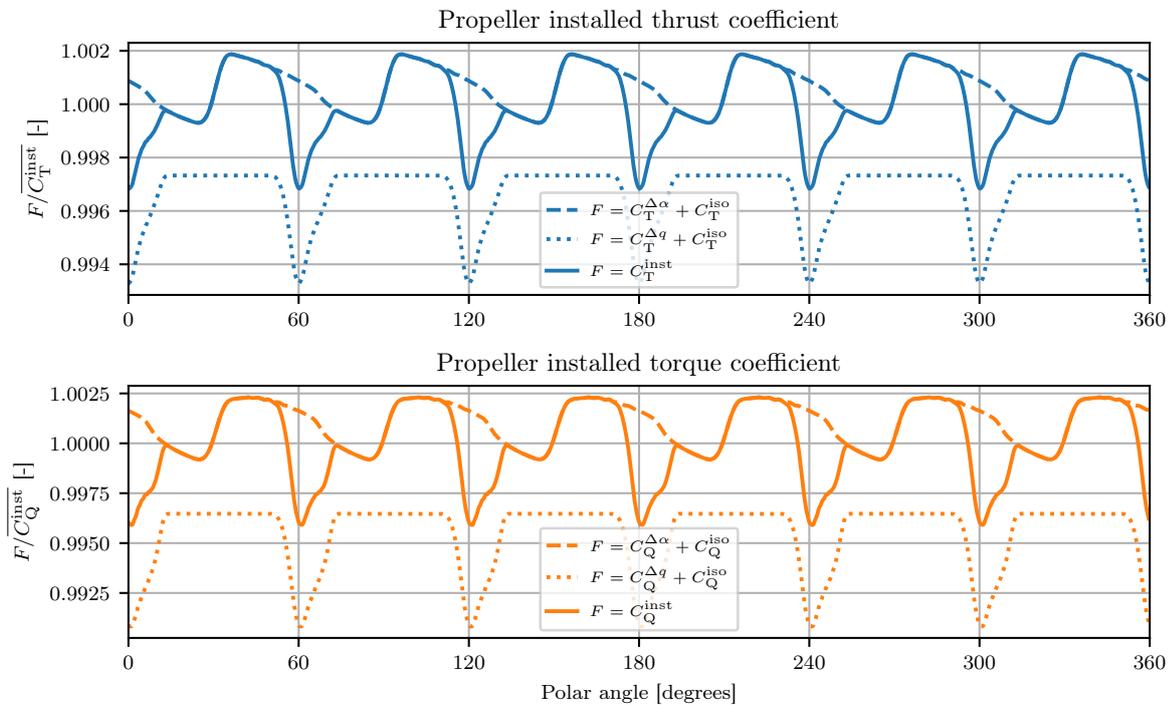


Figure 5.5: Propeller installed thrust and torque coefficient for varying polar angle, with the APIAN propeller, at $J = 1.05$.

Table 5.1: Overview of pylon geometry used throughout this research.

Parameter	Unit	Value
Airfoil	-	NACA 0012
Chord length	m	0.481
Pylon-prop spacing	m	0.160
Pylon drag coefficient	-	0.00523

case three operating points are available at $J = 1.05, 1.40$ and 1.75 . More details are given in table 3.3.

5.2.2. Formal definition of optimization problem

The formal definition of the optimization problem consists of defining the objective function, possible constraints, the design vector and bounds for the design vector.

Objective function

The goal of minimizing unwanted propeller installation effects is defined in the objective function. The definition of the objective requires specifying which aspects of adverse propeller installation effects need to be the focus of this optimization problem. In the previous section, these adverse installation effects are quantified as unsteady fluctuations in the blade lift and drag distributions, and as a consequence, unsteady fluctuations in propeller thrust and torque performance. Additionally, Sinnige [13] demonstrates with an installed noise model that installation effects also have a negative impact on propeller noise performance.

In an ideal scenario, all these aspects of adverse installation effects (noise, unsteady thrust performance) are optimized with a single objective. However, due to the expected size and complexity of this optimization problem, it is unclear at this point whether each of these separate goals will have the same optimization trend. Therefore, three candidate objective functions are defined that are aimed at optimizing for each separate goal. Then, a study is performed for a single operating point to define the effectiveness of each candidate objective function. The most effective objective function is selected and used to perform an optimization for the remaining operating points.

The first candidate objective function aims at minimizing unsteady fluctuations in installed propeller thrust. It can be envisioned that unsteady blade loading also leads to structural penalties due to vibrations and unsteady stresses acting on joints and connections. The formal definition of the objective function is given in equation 5.14:

$$\min T_{\text{installed}}^{\text{RMS}} \quad (5.14a)$$

$$T_{\text{installed}}^{\text{RMS}} = \sqrt{\frac{\sum_{j=1}^{n_{\text{polar}}} (T_{\text{installed}_j} - \overline{T_{\text{installed}}})^2}{n_{\text{polar}}}} \quad (5.14b)$$

Where $T_{\text{installed}}^{\text{RMS}}$ is the root-mean-square (RMS) of the installed propeller thrust and represents the unsteady fluctuations, and n_{polar} the size of the unsteady polar angle distributions.

The second and third candidate objective functions aim at minimizing the drivers of the noise emissions. Since the addition of isolated or installed noise prediction methods is not within the scope of this thesis, an alternative representation of noise is required. When the installed noise prediction model of Hanson [34] is analyzed, it is deduced that the installed noise contribution is driven by a single relation at local blade section level, which is given in equation 5.15:

$$f_{\text{noise}_{i,j}} = c_{i_{\text{installed}_{i,j}}} \cdot M_{\text{helical}_i}^2 \quad i, j = 1 \dots n_{\text{sections}}, n_{\text{polar}} \quad (5.15)$$

Where M_{helical_i} is the isolated helical Mach number related to the same specific blade section, In general, the tip region of a propeller blade produces a majority of the total thrust. Unfortunately, due to the large rotational speed experienced locally at the blade tip, the performance of the blade tip region is the more sensitive in terms of unsteady loading and noise. By adding the Mach number term, the optimizer will take these considerations into account when minimizing unsteady installation effects.

In order to represent the fluctuation in the alternative noise relation, the RMS is used. Compared to the installed propeller thrust, the dimension of the relation in equation 5.15 is two, which means that there are two possible ways in which to calculate the total RMS value. In the first method, the RMS is determined for

each individual section first, and then the RMS values of all sections are summed to get the total RMS value. This candidate objective function is given in equation 5.16.

$$\min \sum_{i=1}^{n_{\text{sections}}} f_{\text{noise}_i}^{\text{RMS}} \quad (5.16a)$$

$$f_{\text{noise}_i}^{\text{RMS}} = \sqrt{\frac{\sum_{j=1}^{n_{\text{polar}}} (f_{\text{noise}_{i,j}} - \overline{f_{\text{noise}_i}})^2}{n_{\text{polar}}}} \quad (5.16b)$$

In the second method, the alternative noise representation, contributions from all sections are first summed, and then the total RMS value is calculated afterward. This candidate objective function is given in equation 5.17.

$$\min = f_{\text{noise}_j}^{\text{RMS}} = \left(\sum_{i=1}^{n_{\text{sections}}} f_{\text{noise}_{i,j}} \right)^{\text{RMS}} \quad (5.17a)$$

$$f_{\text{noise}_j}^{\text{RMS}} = \sqrt{\frac{\sum_{j=1}^{n_{\text{polar}}} (f_{\text{noise}_j} - \overline{f_{\text{noise}_j}})^2}{n_{\text{polar}}}} \quad (5.17b)$$

The difference between these two noise related candidate objective functions is that for the first objective function (equation 5.16), the optimizer will try to optimize the unsteady lift distribution at each section, while the second objective function (equation 5.17) combines all sections and take the phase shifts into account. Finally, while it is not the ultimate objective of this optimization problem, there is a desire for maintaining or improving mean installed efficiency. There are two methods for including this requirement into the optimization problem. The obvious method is to introduce a constraint that limits the optimization problem. However, this method does not encourage the optimizer to search for solutions with increased efficiency. Therefore a second method is applied, which includes the efficiency requirement in the objective. This transforms the original single objective optimization problem into a multi-objective optimization problem. In this case, a weighted-summation approach is deemed appropriate and is given in equation 5.18.

$$F(\overline{X}) = \lambda \cdot \frac{f_1(\overline{X})}{f_1^{X_0}(\overline{X})} + (1 - \lambda) \cdot \left(\frac{f_2(\overline{X})}{f_2^{X_0}(\overline{X})} \right)^{-1} \quad (5.18)$$

Where λ is a weight factor ranging from zero to one, f_1 is one of the candidate objective functions in equations 5.14, 5.16 or 5.17, and f_2 is the mean installed efficiency. Both parts of the multi-objective are normalized with the baseline, but the mean installed efficiency fraction is inverted in order to achieve the goal of maximizing mean installed efficiency, compared to minimizing the candidate objective part. It is not valid to assign any value to λ beforehand, which means that a study must be performed that investigates the influence of λ on the objective, by creating a Pareto front for varying λ .

Constraints

The objective is subject to one constraint, which is given in equation 5.19:

$$0.99 \leq \left(\frac{\overline{T}}{T_{\text{baseline}}}_{\text{installed}} \right) \leq 1.01 \quad (5.19)$$

The constraint is dictated by a desire for maintaining thrust performance throughout the operating range, with respect to the baseline case. The main function of the propeller is delivering a specific amount of thrust, so it is of utmost importance to ensure that the baseline thrust performance is maintained. However, if this requirement is taken as a hard constraint, there is a significant chance that this will choke the optimizer and no significant improvement will be made. Based on this, instead of a hard equality constraint, two inequality constraints are implemented that allow for a $\pm 1\%$ deviation in thrust performance.

Design vector & bounds

The design vector and the corresponding bounds define the optimizer design space. The goal of the optimization is to reduce installation effects by changing the propeller geometry, which means that the design vector must consist of variables that influence blade size (chord distribution), orientation (pitch distribution) and

Table 5.2: Overview of design vector and design space.

Variable [unit]	X_{LB}	X_0	X_{UB}
Hub lean angle [degrees]	-30.0	-16.1278	0.0
Mid lean radial ratio [-]	0.30	0.5831	0.85
Tip lean angle [degrees]	20.0	27.3785	50.0
Hub sweep angle [degrees]	-25.0	-10.4758	0.0
Mid sweep radial ratio [-]	0.30	0.6600	0.85
Tip sweep angle [degrees]	25.0	39.6143	50.0
Advance ratio shift [-]	-0.1	0.0	0.1
Blade pitch angle [degrees]	30.0	39.6143	50.0
Chord CP 1 [m]	0.02562	0.08252	0.1218
Chord CP 2 [m]	0.07000	0.1242	0.1680
Chord CP 3 [m]	0.06282	0.1183	0.1670
Chord CP 4 [m]	0.04124	0.08403	0.09337
Chord CP 5 [m]	0.01025	0.02237	0.04162
Twist CP 1 [degrees]	11.1787	18.6787	26.1787
Twist CP 2 [degrees]	12.1406	19.6406	27.1406
Twist CP 3 [degrees]	-4.7378	2.7621	10.2621
Twist CP 4 [degrees]	-13.4471	-5.9471	1.5528

position (sweep & lean distribution). Furthermore, since the thrust performance must be maintained, one additional variable will be added to aid the optimizer in achieving the thrust performance constraints. The geometry is generated with the parametrization tool of chapter 4, which means that the design variables are picked from the APIAN parametrization model. An overview of the specific variables, reference design vector and bounds are given in table 5.2, while figure 5.6 provides an overview of the specific chord and twist design space. The sweep and lean bounds are based on what is considered realistic values considering structural requirements. The twist is also defined with a simple shift of ± 7.5 degrees, as shown in figure 5.6. Finally, the chord bounds are more complex compared to the other. Chord influences the Reynolds number, which impacts the low-fidelity isolated performance module. As discussed in chapter 2, in order to perform an isolated performance analysis, aerodynamic properties of each section need to be determined using Xfoil [12]. Unfortunately, a combination of a relative high Reynolds number and airfoil shapes with a small leading edge radius in the upper region of the blade, cause convergence issues when applying Xfoil. If the chord is not limited properly, the optimization loop becomes unstable and it would not be possible to perform an optimization. Based on this, special care is given to finding the limits of Xfoil with respect to the chord, which are given in figure 5.6 as “Aero data boundaries”. Then based on these, the actual chord control point bounds are tuned so that they are close to the “Aero data” bounds, but do not exceed them.

Both the objective and the constraints are normalized, which means that it is convenient to normalize the bounds as well. This eliminates the differences in order of size between the different design variables, so that the optimizer treats the individual design variables the same. The method of normalization is shown in figure 5.7. The method of normalization consists of defining a linear relation between the real design space and the normalized design space, where the normalized lower and upper bounds are fixed at zero and one, respectively. The initial baseline is then found with the relation in equation 5.20.

$$\overline{X_{0_{\text{norm}}}} = \frac{\overline{X_0} - \overline{X_{LB}}}{\overline{X_{UB}} - \overline{X_{LB}}} \quad (5.20)$$

Sensitivity study

Before the formal definition of the optimization is implemented, it is useful to study the sensitivity of individual components of the design vector on the candidate objective functions. This will provide a better understanding of the influence of the design space, and the effectiveness of each objective function in achieving the ultimate goal: reducing unwanted installation effects. This sensitivity study consists of analyzing objective performance for two variables of the design vector at a time.

The design vector can be split into five categories: lean, sweep, pitch, chord and propeller rotation speed. For the remaining variables, it is important to combine variables that do not affect the behavior of the other variables. For example, each of the twist and chord control point variables have a significant influence on the entire twist and chord distributions, which makes it impossible to individually study the effect of each variable. In total there are three combinations studied here; hub & tip lean angle, hub & tip sweep angle

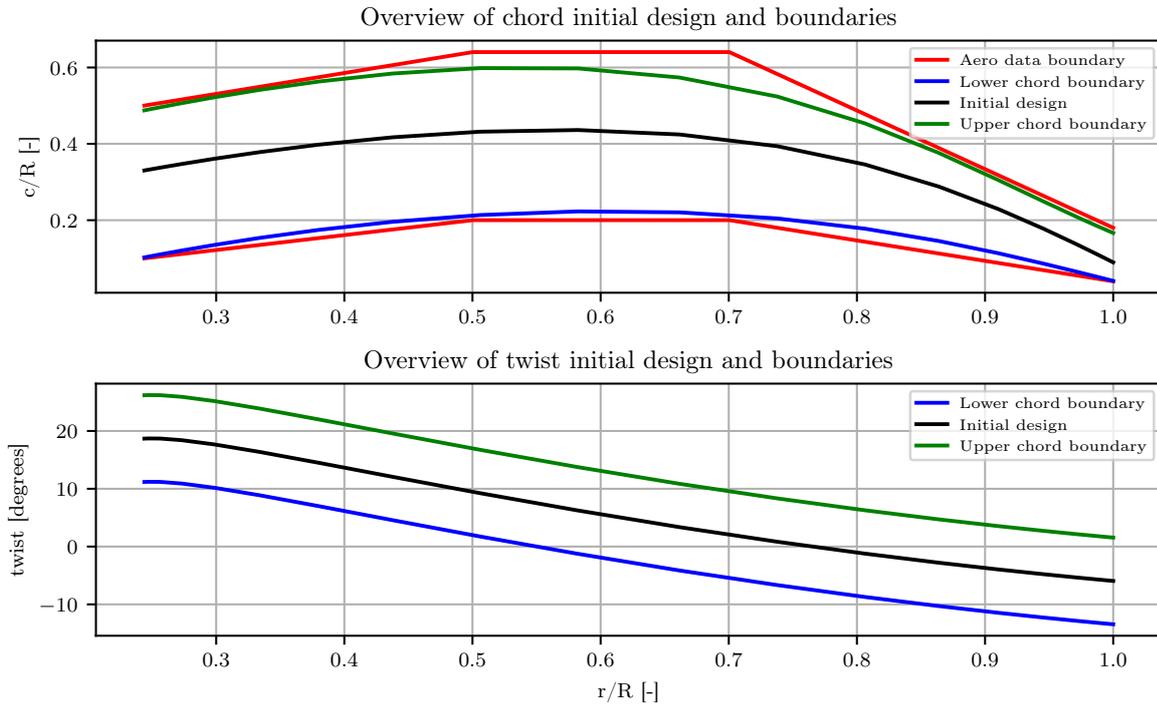


Figure 5.6: Physical implication of chord and twist control point design space.

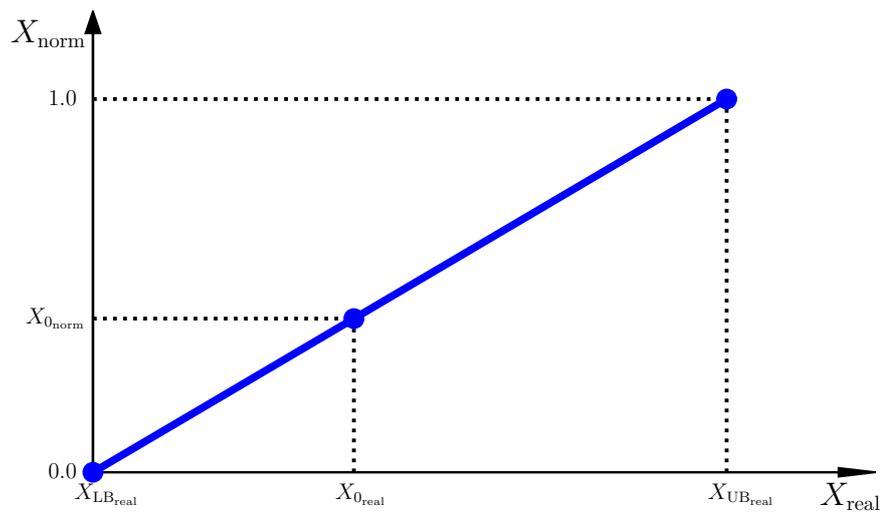


Figure 5.7: Graphic overview of method of normalization.

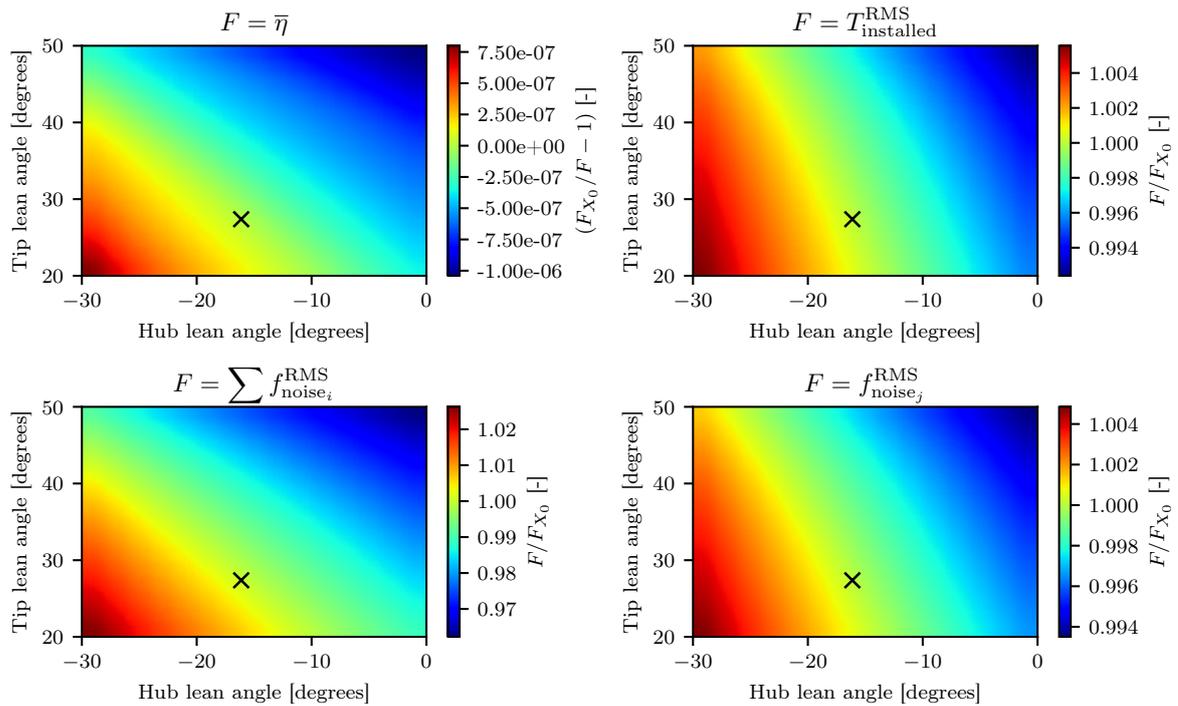


Figure 5.8: Sensitivity of hub & tip lean angle on different candidate objective functions for $J = 1.05$. The cross mark represents the initial propeller design.

and finally collective blade pitch angle & chord shift. Collective chord shift implies that the original chord distribution is collectively increased or decreased. For this study only one operating point is considered, at an advance ratio of 1.05.

Effect of hub & tip lean angle Figure 5.8 shows the sensitivity of hub and tip lean angle on the different candidate objective functions. In general, the trend for all candidate objectives is very similar. There is a clear maximum and a minimum at the lower and upper bounds, respectively. This behavior makes sense when considering that lean only influences the propeller-pylon spacing. At the upper bounds, there is no forward lean at the hub and maximum backward lean at the tip, thereby maximizing the propeller-pylon spacing. However, the baseline propeller-pylon spacing is significantly larger than the extra spacing induced by the lean angle, which leads to no significant improvement in the candidate objective values. In terms of mean installed efficiency, there is effectively no improvement because lean is not taken into account in the isolated performance, while in the other functions, the improvement ranges from approximately 0.6% to 3.5%. In the case of the $\sum f_{noise_i}^{RMS}$ plot (bottom-left), the improvement is better compared to the others, because in this case, the improved propeller-pylon spacing has a bigger impact on the performance of individual sections. In the other objectives, the contributions of all sections are summed first. Then, the improved propeller-pylon spacing has only limited impact on the performance.

Based on the findings, it is discovered that when the hub & tip lean angles are included in the design vector, the optimizer will always converge to the upper bounds of these variables. Moreover, the actual improvement on objective performance is not significant. Therefore, it is decided to not include the variables related to the lean in the design vector. When removed, it is expected that the complexity of the optimization problem and computation time is reduced.

Effect of hub & tip sweep angle Figure 5.9 shows the sensitivity of the hub and tip sweep angle on different candidate objective functions. In general, there is a difference in trend between the candidate objective functions. In the mean installed efficiency plot (top-left), there is effectively no improvement in performance, with a maximum and minimum at the upper and lower bounds, respectively. Mean installed performance is primarily determined by the isolated performance. Since sweep is not taken into account during the isolated performance analysis, the effect on mean installed efficiency will be negligible, which is seen here. In the case of the $\sum f_{noise_i}^{RMS}$ plot (bottom-left), there is again no significant improvement in performance and there is also no clear trend, with a large number of local minima and maxima. This behavior is expected since

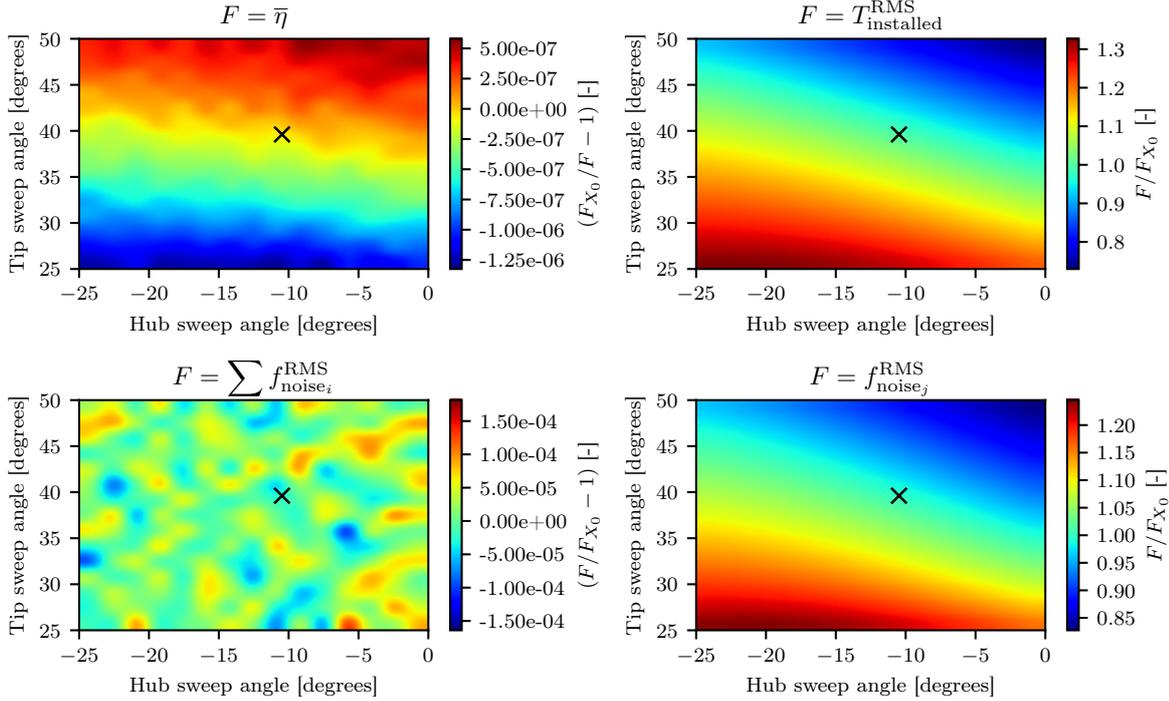


Figure 5.9: Sensitivity of hub & tip sweep angle on different candidate objective functions for $J = 1.05$. The cross mark represents the initial propeller design.

the root-mean-square is taken first, at individual sections. This approach does not take sweep into account, which explains both the trend and the insignificant performance improvement. In the two plots related to $f_{noise_j}^{RMS}$ and $T_{installed}^{RMS}$ (right plots), the root-mean-square is taken after combining the contributions of all the individual sections. This means that in these two cases sweep is taken into account and a clear trend and performance improvement is visible. There is a clear maximum and minimum at the lower and upper bounds, respectively. Sweep influences the timing of each section, passing the pylon wake. Figure 5.10 shows the effect of sweep on the unsteady lift coefficient due to the pylon wake, where the positive peak is the result of a change in angle of attack and the negative peak is the result of a change in dynamic pressure. Sweep does not affect the size of these peaks or the distance between the positive and negative peaks, it only influences the phase shift to the response of each section. The specific order of the positive and negative peak of each section is a clue that explains the behavior in the two plots related to $f_{noise_j}^{RMS}$ and $T_{installed}^{RMS}$ (right plots). By applying sweep, it is possible to position each section such that the positive and a negative peak of different sections coincide, such that their behavior cancels out. Since the phase shift of the hub section is always fixed, the ideal sweep distribution positions each section with a positive phase shift with respect to the previous section. In this case, it results in no forward sweep at the hub and maximum backward sweep at the tip.

Based on the findings, it is discovered that when the hub and tip sweep angles are included in the design vector, the optimizer will always converge to the upper bounds of these variables. While the actual improvement on objective performance is significant, it is concluded that it does not make sense to keep the all the variables related to sweep in the design vector.

Effect of collective blade pitch angle & chord shift Figure 5.11 shows the sensitivity of collective blade pitch angle and chord shift on different candidate objective functions. Reducing twist and chord to single variables allows for a limited general analysis of their effects, but it does not accurately describe the full twist and chord design space. Furthermore, collective blade pitch and chord shift will have a more significant impact on mean installed performance than sweep and lean, in this analysis. Constraints are not taken into account at this point, which means that it is possible that some of the minima displayed here will lie beyond the feasible region, dictated by the constraints. In general, clear trends for all candidate objective functions are visible, but they are very different from one another, with significant potential for improvement.

In the case of installed mean efficiency in the top-left plot of figure 5.11, the collective pitch is dominant over the chord shift, with a clear minimum at the lower bound of the collective pitch angle. Propeller effi-

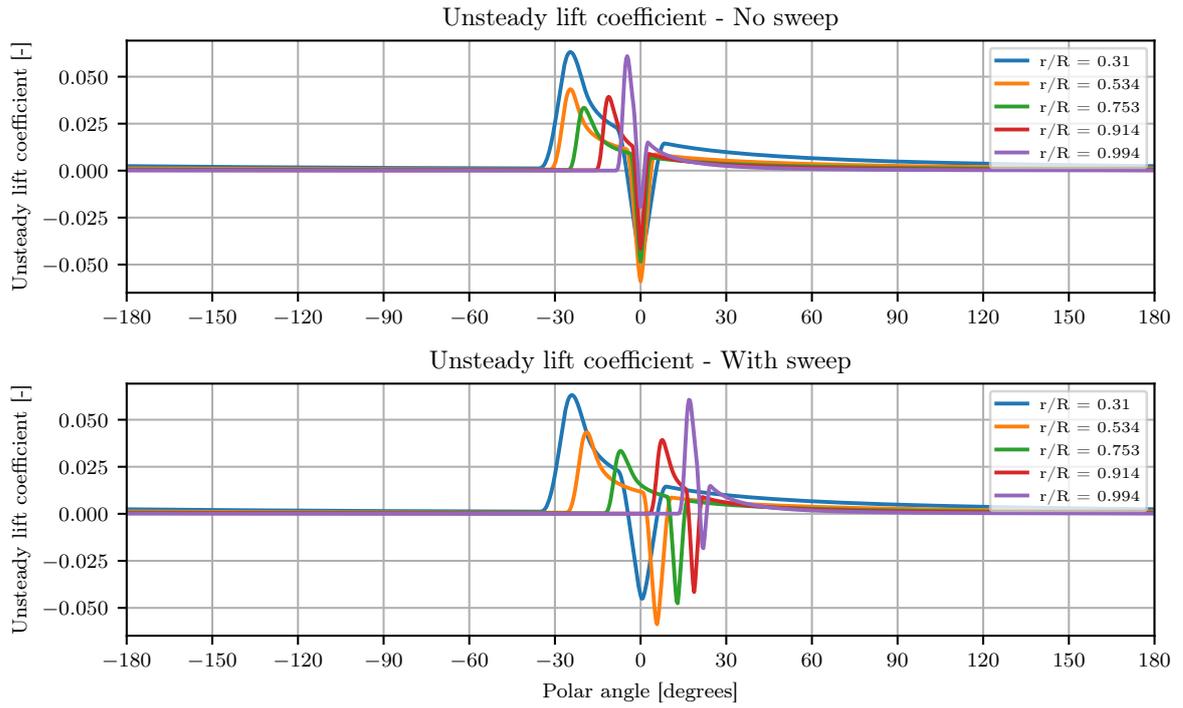


Figure 5.10: Influence of sweep on unsteady lift coefficient for $J = 1.05$, at a limited number of sections.

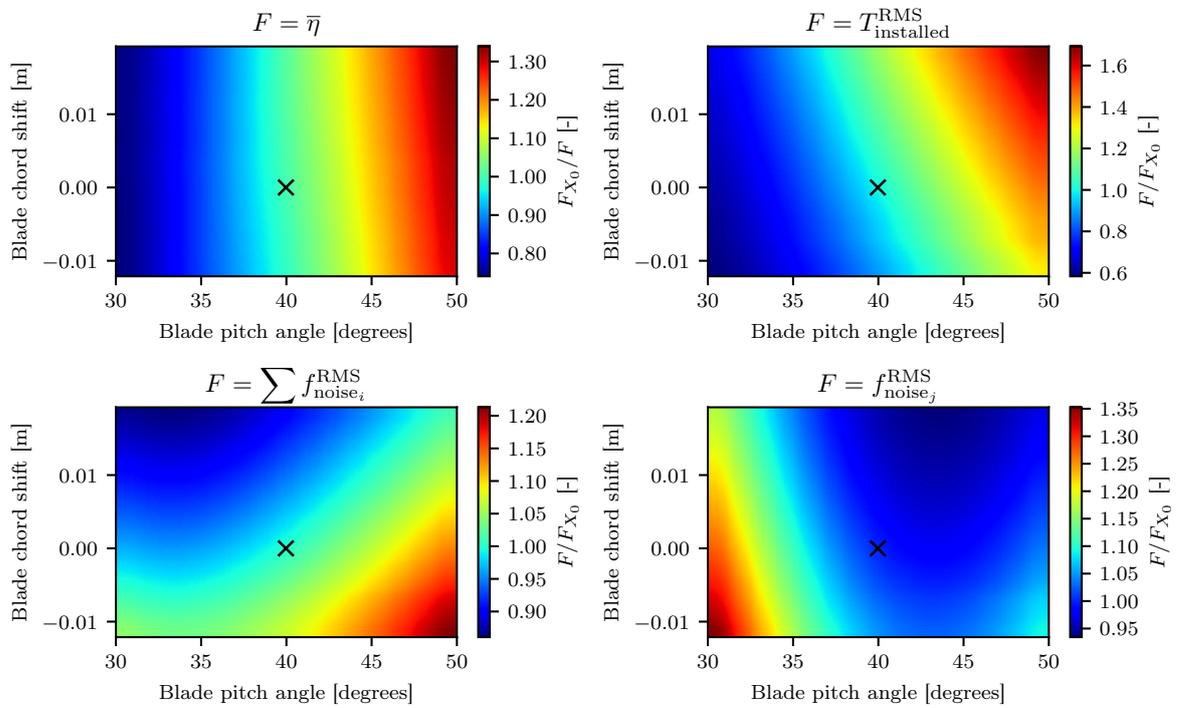


Figure 5.11: Sensitivity of collective blade pitch angle & chord shift on different candidate objective functions for $J = 1.05$. The cross mark represents the initial propeller design.

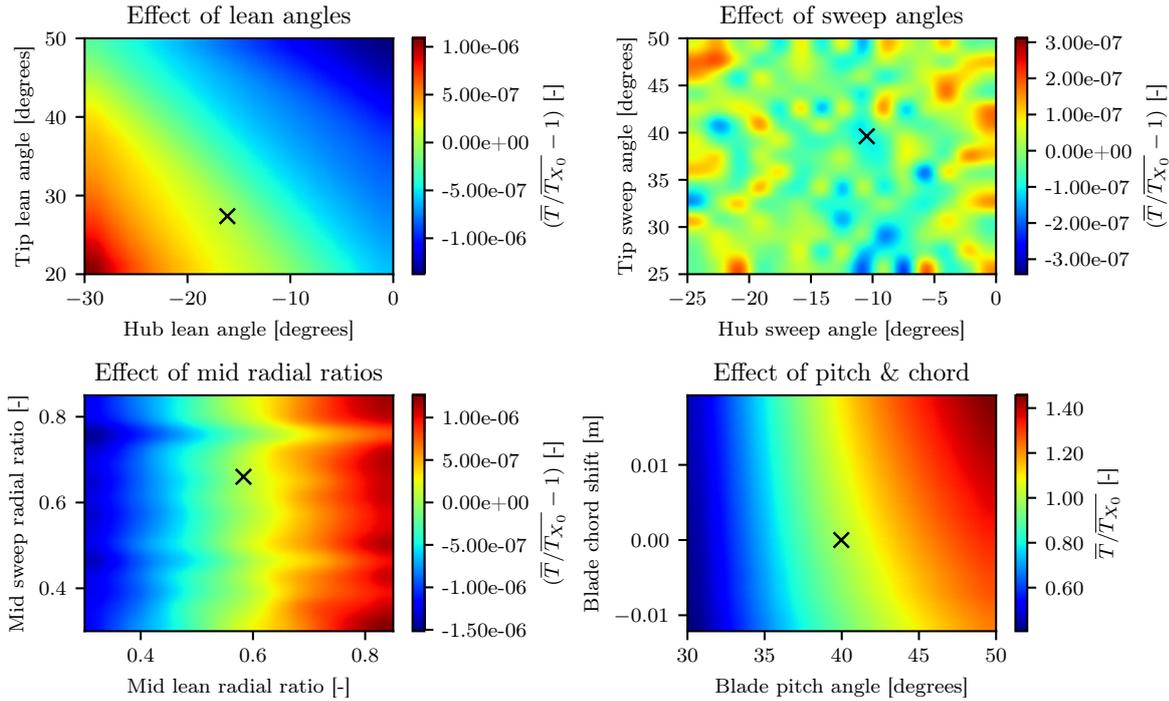


Figure 5.12: Sensitivity of all variables on thrust constraint for $J = 1.05$. The cross mark represents the initial propeller design.

ciency is defined as the ratio between the thrust and power coefficient, multiplied with the advance ratio. Figures 2.4 and 2.5 show the isolated thrust and power coefficient performance for different blade pitch angles. From those figures, it is concluded that for decreasing blade pitch angle, the power coefficient decreases more than the thrust coefficient, resulting in increasing efficiency. Therefore, the trend shown in the top-left plot of figure 5.11 is most likely the result of a change in isolated performance, rather than in a change in installed performance.

The results in the plots related to the three candidate objective functions, in figure 5.11 are influenced by pitch and chord in multiple ways. It is therefore difficult to link a specific trend to a specific point in the isolated and installed methods. What is interesting to see is that, while all candidate objective functions are defined with the same ultimate goal in mind, namely to reduce installation effects, all three functions will converge to different points in the design space. A minimum in one objective does not automatically mean a minimum in all other areas. The results in the $T_{\text{installed}}^{\text{RMS}}$ plot (top-right) can be seen as a reduction in propeller installation effects, while the $f_{\text{noise}_j}^{\text{RMS}}$ plot (bottom-right) can be considered as a reduction in blade installation effects, and finally, the $\sum f_{\text{noise}_j}^{\text{RMS}}$ plot (bottom-left) can be seen as a reduction in individual section installation effects. Based on the findings here, it is concluded that when the optimization is implemented, one of the three candidate objectives must be selected on the basis of what system (propeller, blade, individual sections) ultimately qualifies as reduced installation effects. This study has demonstrated that they are mutually exclusive.

Effect of all variables on thrust constraint Figure 5.12 shows the sensitivity of all previous design variables on the thrust constraint. What is immediately clear is that all lean and sweep variables have no significant effect on the mean installed thrust. This makes sense since the main contribution to mean installed thrust is isolated thrust, and in the current isolated propeller performance model, no sweep and lean effects are included. Pitch and chord do influence isolated thrust performance and that is visible in the lower-right plot. Blade pitch has a larger effect than chord shift in this plot, however, this is partly because only a limited collective chord shift is applied in this study due to chord tip limitations, so this will likely change when the full chord range is available in the optimization. From this plot, it is clear that the thrust constraint will have a major impact on the range of feasible collective blade pitch and twist. The trend is exactly as expected based on how these parameters affect isolated thrust performance. More blade pitch will increase the angle of attack of the blade, thereby generating more thrust. If the chord is decreased there is less blade area to generate thrust, thereby reducing thrust.

5.2.3. Optimization algorithm

For this optimization problem, it is convenient to select an optimization algorithm from the Python “Scipy optimization” library [35]. For bounded, constrained optimization, only two algorithms are available, namely, the “Sequential Least Squares Programming” (SLSQP) algorithm and the “Constrained Optimization By Linear Approximation” (COBYLA) algorithm. Both are derivative-based algorithms, which in the case of a non-linear optimization problem, potentially have difficulties reaching the global minimum. Therefore, the derivative-free “Nelder-Mead” algorithm is also considered. The default Scipy implementation of the “Nelder-Mead” algorithm does not work for bounded, constrained optimization problems. However, it is relatively easy to adjust this algorithm so that it does work for such bounded, constrained optimization problems. Each method is discussed here, after which one is selected to be used for the remainder of this thesis.

COBYLA algorithm

The COBYLA algorithm is a gradient based numerical optimization algorithm where at each iteration, a linear approximation of the gradient of the objective and constraint design space is used to determine a new search direction for the next iteration step. In the Python implementation, an initial design vector step size needs to be supplied for the linear approximation of the gradient, after which for every next iteration, the new step is half the size of the previous step. This process is independent of the rest of the algorithm and so the convergence and this algorithm’s ability to find a global minimum is mainly influenced by the initial step size.

If a small initial step size is selected, the consequent iterations have smaller and smaller steps, so if the initial guess is not on a very sharp gradient, the search area within the design space will be very limited. A larger initial step size will increase the search area within the design space. However, the linear approximation of the gradient will become less accurate. This could potentially impede the algorithm on finding a good search direction and thereby exiting prematurely. The COBYLA algorithm works well if your optimization problem is concave, meaning that the design space contains one single obvious global minimum. In other cases, this algorithm will only converge to a local minimum that is close to the initial guess, but there is no certainty that this will be the global minimum. The initial guess and the initial step size must always give solutions that are on the slope towards the global minimum [35]. Since it is not clear beforehand where the global minimum is, with respect to the initial reference point, it is concluded that this algorithm is not suitable for the optimization problem solved in this thesis.

SLSQP algorithm

The SLSQP algorithm is an adaption on “Sequential Quadratic Programming” (SQP) whereby the original objective function and constraints are modeled as a quadratic function. The algorithm then tries to solve a sequence of optimization sub-problems, that improve and then minimize the quadratic function model. The implemented adaptation uses a “Least-Squares” optimization algorithm to solve the sequence of optimization sub-problems.

The SLSQP algorithm is a derivative-based algorithm that either uses a supplied Jacobian or an approximated Jacobian using a forward-difference scheme to determine the search direction. For the optimization problem solved in this thesis, the Jacobian is unknown so only the approximated option is applicable. The forward-difference scheme requires the definition of a design vector step size to approximate the derivative of the objective and each design variable, which needs to be defined. Here, the same problems rise as with the initial step size problems with the COBYLA algorithm. A too small or large step size gives a bad approximation of the gradient, which impedes the algorithm on finding a good search direction. Furthermore, the Python implementation of this algorithm makes the step size even more important because of how it determines the initial search direction and where the algorithm starts to optimize in the design space. For the initial search direction, the algorithm approximates the Jacobian and then, depending on the size of the gradient, it takes a step in the design space that is the starting point of the optimization. It is not clear what specific criteria are used to determine the size of this particular step, but if a particular gradient is of considerable size, the subsequent step in that direction is also very large. Initial testing of this algorithm has revealed that for the optimization problem solved in this thesis, the algorithm goes very often immediately to the bounds and from that point tries to minimize further with much smaller steps. The problem of this is that if the objective design space is convex near a bound, it assumes that it has found a feasible minimum and terminates the optimization routine. If an optimizer has a bias towards the bounds which is only based on that initial search direction, it will not consider the entire design space. This also means that the the algorithm’s performance is heavily influenced by the initial guess and step size. For the optimization problem solved in this thesis, it is concluded that this algorithm is not suitable.

Nelder-Mead algorithm

The Nelder-Mead algorithm is a derivative-free algorithm that employs a simplex of dimension $n_{\text{desVars}} + 1$ where the objective function is evaluated at each test point. The potential of each test point is assessed and

this is used to replace the worst test points with a new value based on the position of the simplex. By replacing the worst test point, the simplex either moves through the design space or shrinks, depending on whether the simplex is away or near a minimum, respectively.

The Python implementation of this algorithm is for unconstrained, unbounded optimization, which does not meet the requirements of the optimization problem. To include bounds, an alternative version of the Nelder-Mead algorithm by Luersen et al. [36] is implemented. In this version, the way the bounds are included is at the point where in the algorithm a new design point is determined based on the position of the simplex. If the new design point falls out of bounds, the new design point is reflected on the bounds. With this method, the algorithm does not lose any information about the design space, it only corrects the new design point to fall within the bounds. Unfortunately, the original implementation of this algorithm does not have a method for including constraints as well. The design process of determining new points in this algorithm cannot be influenced with information about constraint violation. Instead, a common method is implemented where the constraints are included in the objective function by applying a penalty function, see equation 5.21.

$$\min \Pi(\bar{X}) = F(\bar{X}) + \sum_{i=1}^{n_{\text{constraints}}} \alpha_i \cdot \max\left([0.0, g_i(\bar{X})]\right) \quad (5.21)$$

Where α_i are weight factors for each constraint and g_i are the inequality constraints:

$$g_i(\bar{X}) \leq 0.0 \quad (5.22)$$

By implementing the penalty function in this way, the objective is penalized for violating a constraint, but not rewarded for moving far away from the constraint. For the optimization problem solved in this thesis, the weight factors are defined at 5.0. The thrust constraint is sufficiently strict that even small constraint violations need to be penalized, thereby requiring a relatively large weight factor.

The Nelder-Mead algorithm is less dependent on initial guess and initial simplex size compared to the SLSQP algorithm, but initial testing revealed that both still have influence on the algorithm's performance. Based on these findings and the performance of the other two algorithms, it is decided that the Nelder-Mead algorithm is suitable for the optimization problem solved in this thesis, but an additional strategy is required to reduce algorithm performance uncertainty due to the initial guess. This strategy is given in the next paragraph.

Multi start point strategy

It is expected that due to the complexity, the optimization problem will be severely non-linear. Even with the implementation of the most suitable algorithm, uncertainties about the influence of the initial guess on the algorithm's performance remain. Because of this, a multi start point strategy is implemented that will reduce algorithm performance uncertainty due to the initial guess. In the multi start point strategy, a considerable number of random initial guesses are defined, which will be used as starting points for multiple optimizations. This will create a pool of feasible optimized solutions, from which the best is selected and regarded as the most likely global optimum.

Initially, other strategies were considered as well, such as a multi start point strategy with a grid of initial guesses, however, any strategy (including the implemented strategy) will significantly increase the amount of computation time and work load. The chosen strategy is considered the best trade-off between computation time and covering the entire design space. From now on, one optimization is considered to be performing the multi start point strategy for one condition. Initial testing revealed that thirty random starting points is enough to ensure that the same ultimate solution is always found.

The setup of the optimization problem solved in this thesis, consists of performing one multi-candidate objective optimization study for one operating point at $J = 1.05$, and performing one different operating point optimization study for two additional operating points at $J = 1.40$ and 1.75 , and one selected objective function. In order to properly investigate the differences between the different conditions, it is decided to limit the effect of the initial starting points, by defining one set of random starting points which is used for all optimizations.

6

Propeller blade optimization for minimized installation effects - Results

The purpose of this chapter is to present the results of the propeller blade optimization for minimized installation effects. Because of the complexity of the optimization problem, the approach applied here is very methodical. First, separate optimizations for three candidate objective functions are performed at one operating point. This will give insight into the effect of each objective function on the ultimate goal of minimizing installation effects. The results for this study are given in section 6.2. Based on these results, one objective function is selected and additional optimizations are performed for two different operating points. According to Sinnige et al. [5], installation effects (both in terms of noise and overall performance) are different for varying operating conditions. This means that, in order to make any valid conclusions on what is considered to be an optimum design, multiple optimizations must be performed at different operating points. The results of this study are given in section 6.3. At this point, there will be three optimum designs, from three performed optimizations (one from the initial multi-candidate objective study in section 6.2 and two from the other operating points optimization study in section 6.3). The next step involves performing a detailed performance study for the three optimum designs. In this study, the installed performance of each optimum design is analyzed over the entire operating range of the reference propeller. The results of this study must prove the effectiveness on the ultimate goal of minimizing installation effects, throughout the operating range. This also allows for comparing the effectiveness of each optimum design, with respect to each other and finally select one optimum design that performs the best. This is treated in section 6.4.

6.1. Accuracy of Low- and high-fidelity analysis method

In the current setup of the optimization routine, the low-fidelity propeller performance analysis method is used to predict the isolated propeller performance contribution. This is significantly faster than using the high-fidelity propeller performance analysis method. However, a low-fidelity method in general less accurate than a high-fidelity analysis. With the high-fidelity method, it is possible to take three-dimensional flow phenomena into account, such as the leading edge vortex. It is important to identify the differences in results between the two performance analysis methods. This sub-section gives the results of this study. Both propeller performance analysis methods are tested for the APIAN propeller at three advance ratios equal to 1.05, 1.40 and 1.75.

In figure 6.1, the isolated thrust and torque coefficients are plotted. It was found that the low-fidelity results were more accurate than the high-fidelity results, which is contradictory. However, in chapter 2 it was found that the low-fidelity method tends to over-predict thrust performance in high-loading conditions and under-predict thrust performance in low-loading conditions. It is therefore not fair to make a comparison between the two performance analysis methods in terms of accuracy related to the reference data. If the thrust of the two performance analysis methods are compared, it is found that the low-fidelity method over-predicts the high-fidelity results, which is more in line with what is expected from the low-fidelity performance analysis method. In terms of torque at $J = 1.05$ and 1.40 , the low-fidelity method over-predicts the torque, while the high-fidelity method under-predicts the torque. At $J = 1.75$, the torque results are very similar. This again is a consequence of the low-fidelity method's tendency to over-predict in high-loading conditions and to over-predict in low-loading conditions. Based on these results, it is concluded that the low-fidelity is able to predict general propeller performance characteristics, with the same level of accuracy.

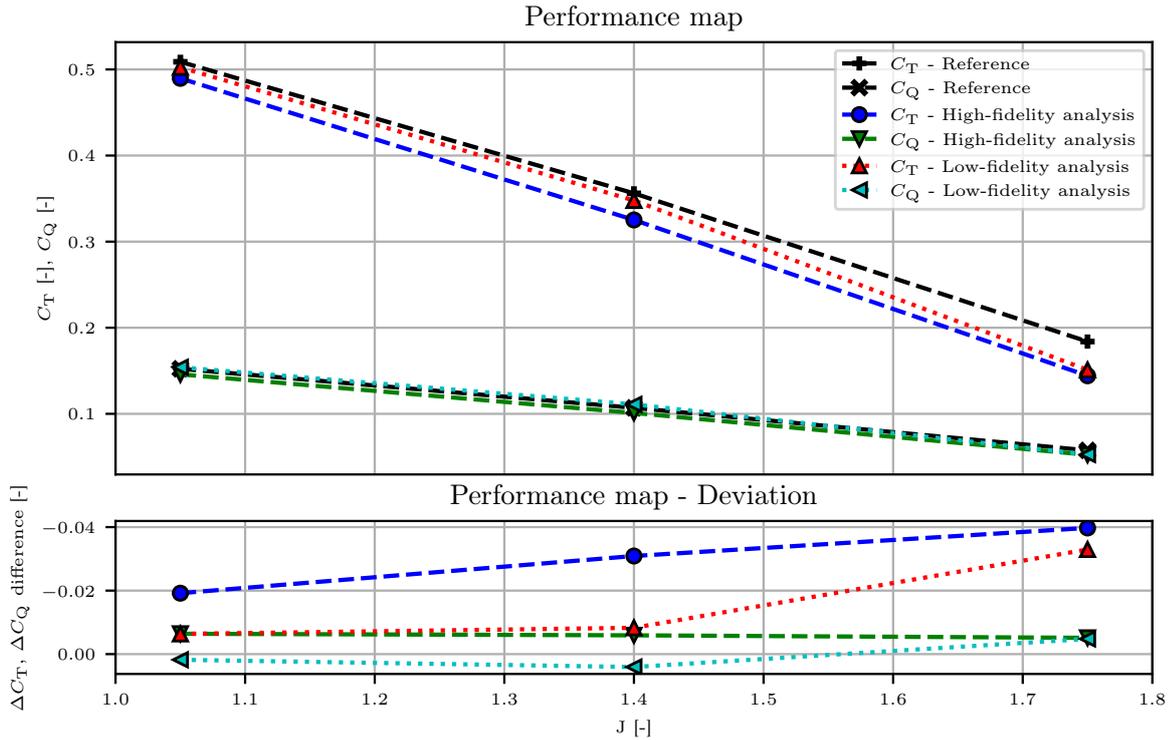


Figure 6.1: APIAN performance map comparison between high-fidelity analysis results and low-fidelity analysis results.

In figure 6.2, the blade thrust distribution is plotted for three different advance ratios. The differences between the two performance analysis methods are more significant compared to the overall performance characteristics in figure 6.1. The peak values are similar, but the distribution of thrust over the blade is very different. At $J = 1.05$, the low-fidelity method's thrust is more distributed over the blade with the peak located at $r/R = 0.69$, while the high-fidelity method's thrust is more concentrated around the peak located at $r/R = 0.88$. For increasing advance ratio it is noticed that the location of the low-fidelity method's thrust peak is increasingly positioned more outboard. For $J = 1.75$, the differences between the two methods are much smaller compared to the difference at the lower advance ratios. This also explains in part why the differences in overall performance characteristics in figure 6.1 are much smaller at $J = 1.75$. The reason for this difference in thrust distribution is related to the presence of a leading edge vortex moves the maximum loading outboard. This leading edge vortex was captured in the high-fidelity analysis, but not by the low-fidelity because in the current setup lean and sweep effects are not taken into account, and it does not have any functionality that can predict these complex three-dimensional flow phenomena. In the validation of the high-fidelity performance analysis method in chapter 3, it was found that for increasing advance ratio, the leading edge vortex starts to develop increasingly more outboard. This means that its effect on the pressure distribution on the blade suction surface becomes smaller. For $J = 1.75$, there was no leading edge vortex detected in the pressure distribution, which is the reason why in figure 6.1 the differences between the two thrust distributions is much smaller at $J = 1.75$.

6.2. Multi-candidate objective optimization study

This section gives the results for the multi-candidate objective optimization study, at a high thrust condition. While studying the effectiveness of the three candidate objective functions, at the same time, the influence of the weight factor λ in equation 5.18 is also studied. The weight factor λ assigns importance to the goal of simultaneously minimizing one of the candidate installation effects objectives and maximizing mean installed efficiency. So for each candidate objective function, a limited number of optimizations is performed, for a range of λ weight factors. This section addresses general optimization results and the design and performance implications for different candidate objective functions. The three candidate objective functions were defined in chapter 5:

- Minimizing installed thrust performance:

$$T_{\text{installed}}^{\text{RMS}}$$

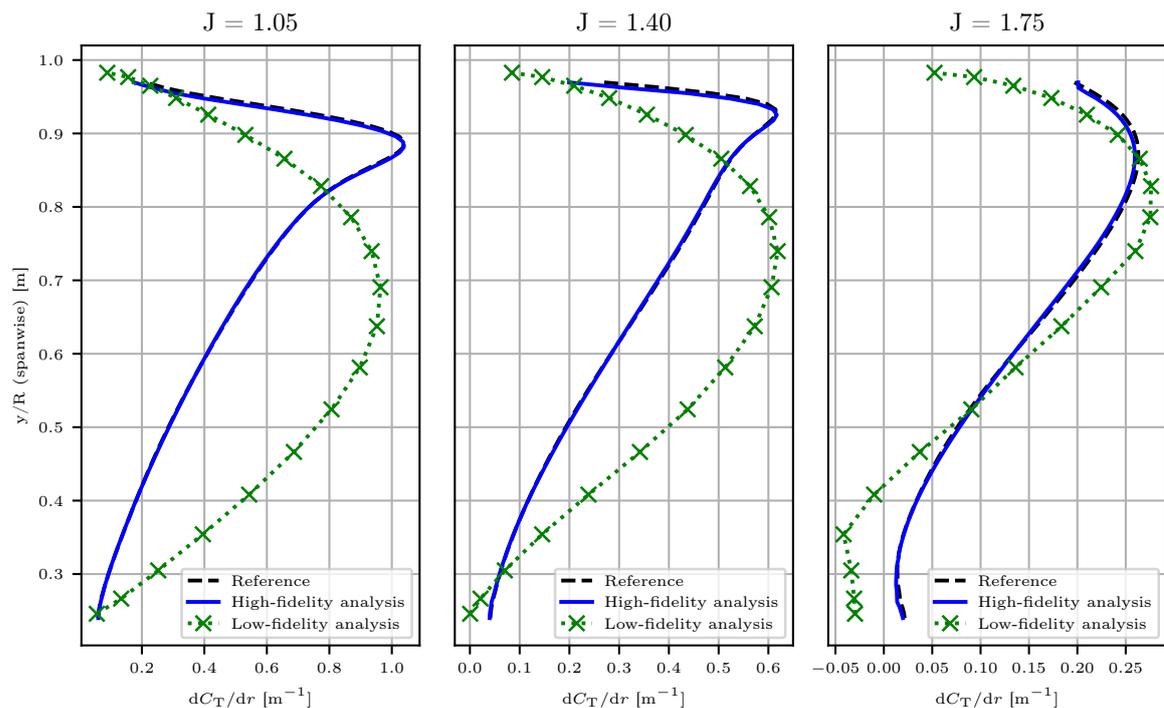


Figure 6.2: APIAN blade thrust distribution comparison between high-fidelity analysis results and low-fidelity analysis results.

- Minimizing individual aerodynamic blade noise drivers:

$$\sum_{i=1}^{n_{\text{sections}}} f_{\text{noise}_i}^{\text{RMS}}$$

- Minimizing combined aerodynamic blade aerodynamic drivers:

$$f_{\text{noise}_j}^{\text{RMS}}$$

6.2.1. Minimizing installed thrust performance

In figure 6.3, the main characteristics of the installed thrust performance optimization are plotted. Multiple optimizations are performed from thirty random initial guess. The results given in figure 6.3 represent the most potent results, out of the optimized pool of thirty, for different λ . For zero λ , the optimizer only optimizes for maximum mean installed efficiency. Figure 6.3 shows that, compared to the reference propeller, for the same thrust and at $J = 1.05$, a 10% increase in mean installed efficiency is possible. While an improvement in efficiency is achieved, at the same time installed thrust performance ($T_{\text{installed}}^{\text{RMS}}$) becomes worse with an increase of about 10%. The implication of this is that improving efficiency and installed thrust performance must have contradicting design trends. If λ is increased, the objective of improving installed thrust performance becomes more and more important. However, for small λ , efficiency is still dominant which results in the fact that the first two points have converged to almost the same solution. After the first two points, the optimizer succeeds in gradually improving installed thrust performance. This trend ultimately results in an installed thrust performance improvement of over 40%, for the case when λ is equal to one. For increasing λ , the mean installed efficiency improvement decreases, and from $\lambda = 0.5$ onwards, the efficiency becomes worse compared to the reference propeller. There are three points ($\lambda = 0.625, 0.75$ and 0.875) that show some fluctuation in the results, compared to the overall trend. It is not clear what the cause is. One explanation is that it is still possible that in certain specific conditions, the optimizer is not able to find a feasible point that follows the trend better. The bottom line is that the optimizer might have been stuck in a local minimum, despite all efforts to eliminate uncertainty due to optimizer performance.

In figure 6.4 a detailed overview is given of the installed thrust and torque performance. For zero λ , the fluctuations in thrust performance are indeed worse, but more importantly, a significant drop in mean torque performance is noticeable which leads to a higher efficiency. For increasing λ the thrust fluctuations decrease and the mean torque increases as well, thereby increasing mean installed efficiency. What is interesting is that for λ is 0.5 and 1.0, the installed thrust performance trend is already similar. However, when examin-

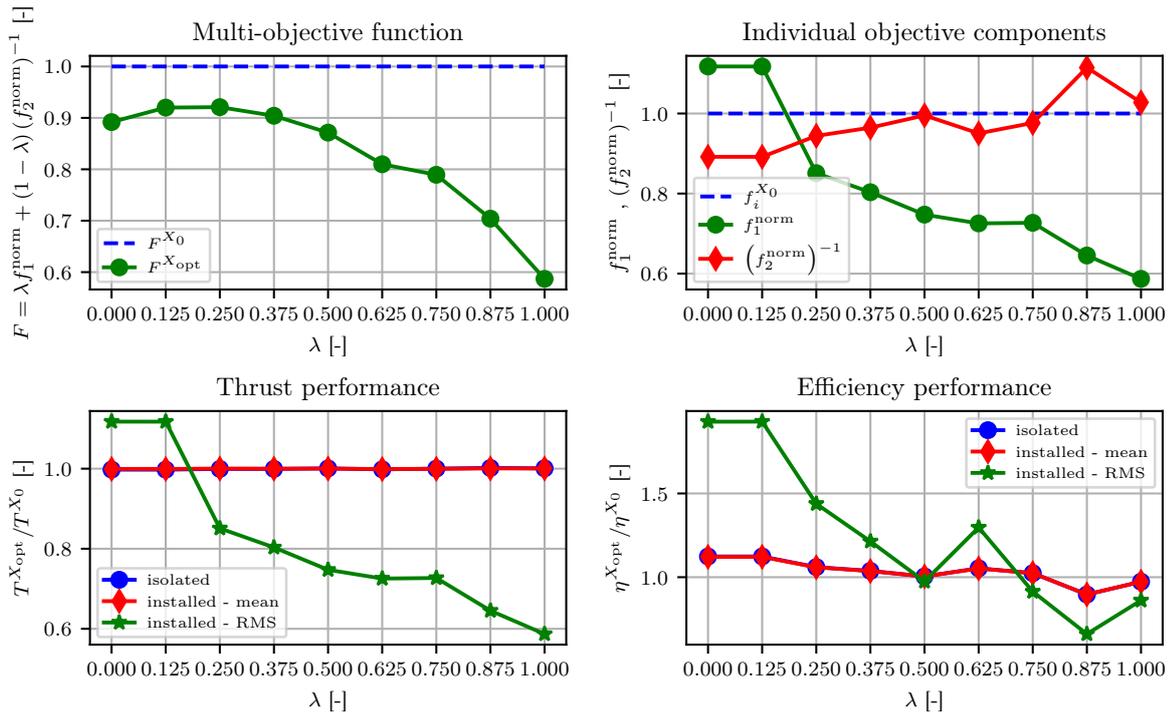


Figure 6.3: Overview of important optimization results for minimizing installed thrust performance, at $J = 1.05$.

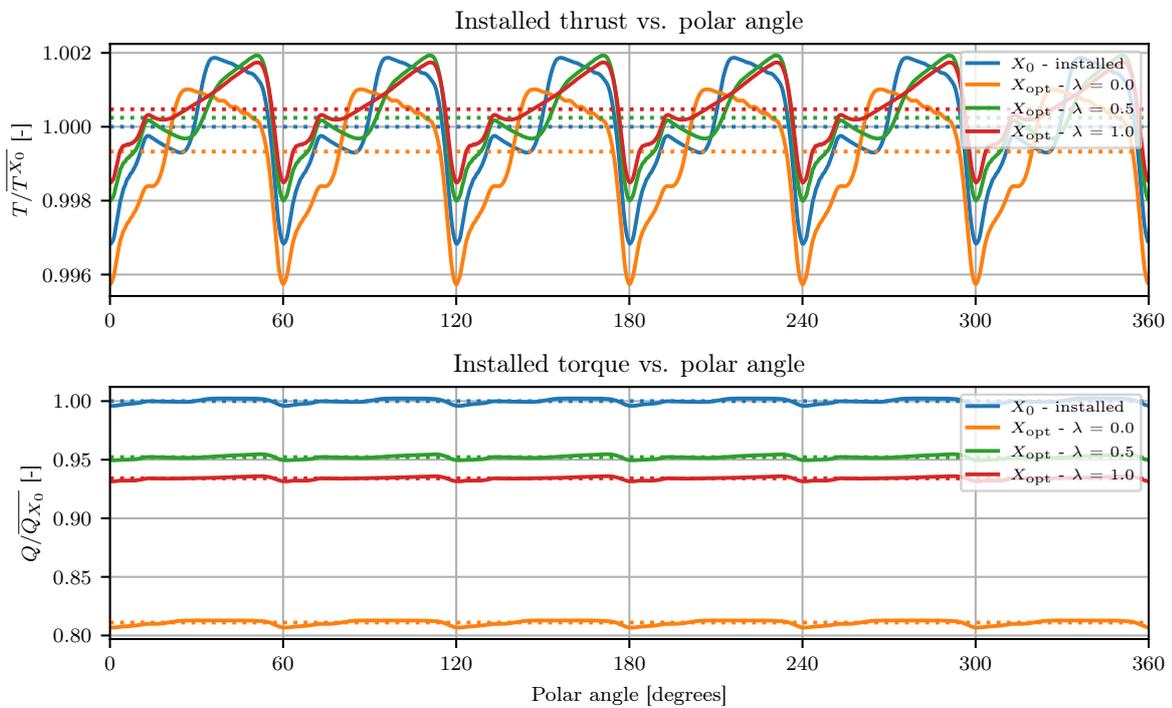


Figure 6.4: Installed thrust & torque performance for minimizing installed section noise performance, at $J = 1.05$.

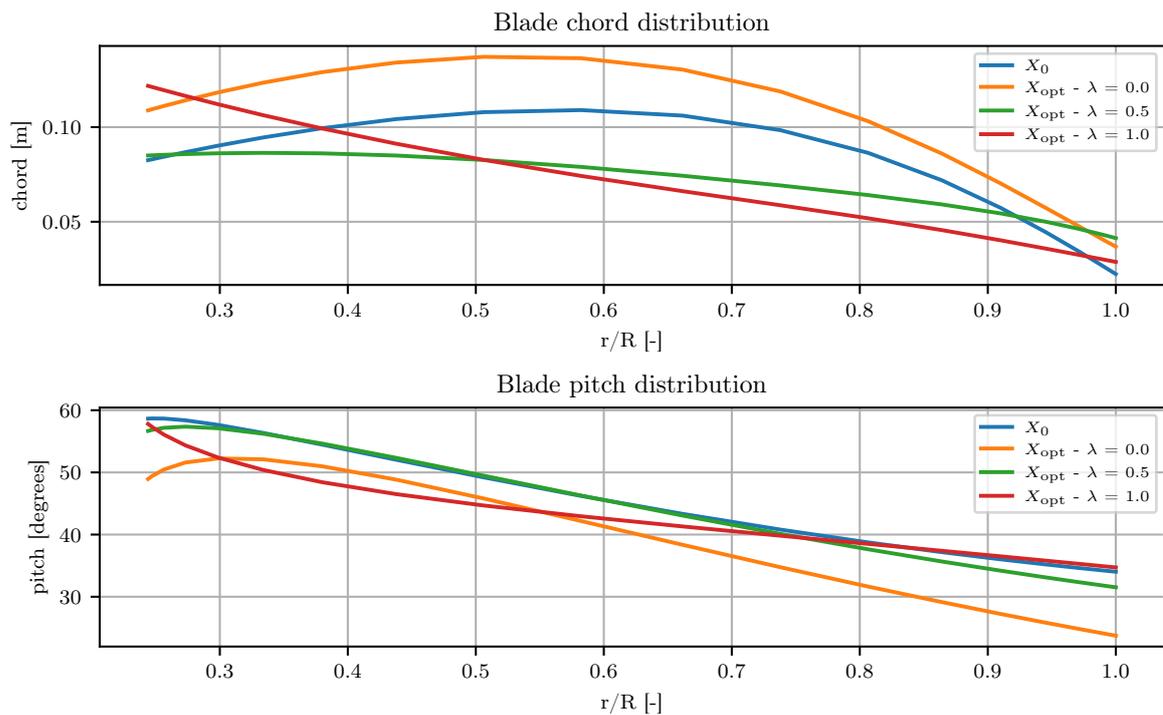


Figure 6.5: Overview of optimized geometry for minimizing installed thrust performance, at $J = 1.05$.

ing the optimized blade pitch and chord distributions in figure 6.5 and in appendix 6.6, it can be seen that the designs are dissimilar. In all cases, the optimizer converges to designs with more chord at the hub and tip. Chord influences the phase shift of the unsteady lift contribution due to a change in angle of attack, because the reduced frequency (equation 5.12) in the Sears function (equation 5.11) is dependent on chord. By changing the phase shift of individual blade sections, the overall unsteady lift it is possible to distribute the load. When only focused on optimizing for efficiency, the entire chord distribution is increased, while the blade pitch reduces. By reducing the blade pitch, the blade angle of attack decreases, which in turn reduces three-dimensional effects and thus improves efficiency to a certain degree. To compensate for the loss in lift and thus thrust performance, the optimizer must therefore increase the chord over the entire range. For increasing λ the optimizer tends to re-distribute the chord distribution, with significantly more area near the hub.

In figure 6.7, a Pareto front diagram is plotted for the installed thrust performance optimization. This plot gives for every λ , the entire optimized pool of feasible points. The Pareto front is made up of all the points where at least one of the objective components is minimum. For increasing λ , the spread of feasible optimums becomes larger. This makes sense because previous results have shown that if there is a larger potential for improvement in installed thrust performance than mean installed efficiency, then there will be a greater variety of feasible points. This behavior is also the cause of the steep slope of the Pareto front.

6.2.2. Minimizing individual aerodynamic blade noise drivers

In figure 6.8, the main characteristics of the optimization are plotted. First of all, due to the setup of the optimization problem it is expected that, for zero λ , the optimization results should be exactly the same as for the Minimizing installed thrust performance of the previous sub-section and also for the minimizing installed blade noise performance of the next sub-section. This is the case. The next four points converge to almost the same solution. For this region applies that the noise and efficiency objectives do not have contradictory design trends, since both are improvements. This is confirmed with the corresponding Pareto diagram in figure 6.9, where optimized feasible points for different λ are clustered, and most likely are the result from the same initial starting point. After this, the first objective becomes dominant, which results in additional improvement of about 17.5% at λ equal to one. For this candidate objective function, the installed thrust performance is worse for minimized noise, thereby confirming that they both have contradictory design trends.

Figure 6.10 shows the different contributions of the individual aerodynamic blade noise drivers objective function. As previously mentioned, the first five solutions in figure 6.8 are almost the same. This means that

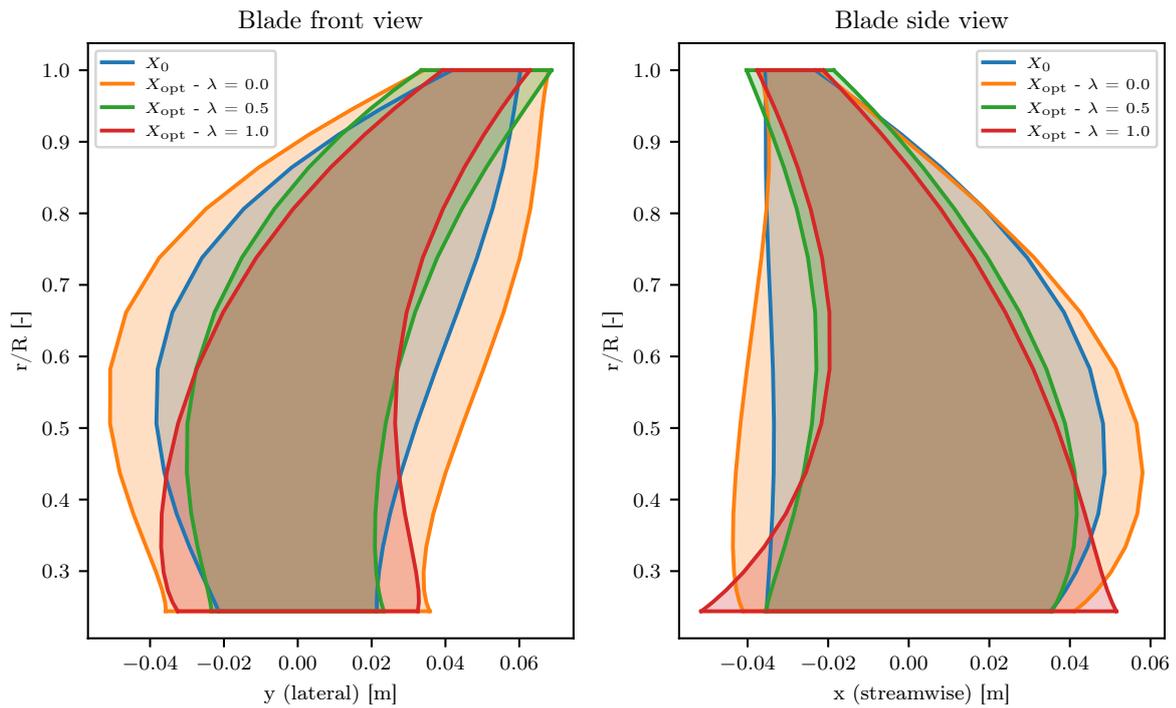


Figure 6.6: Front and side view of optimized geometry for minimizing installed thrust performance, at $J = 1.05$.

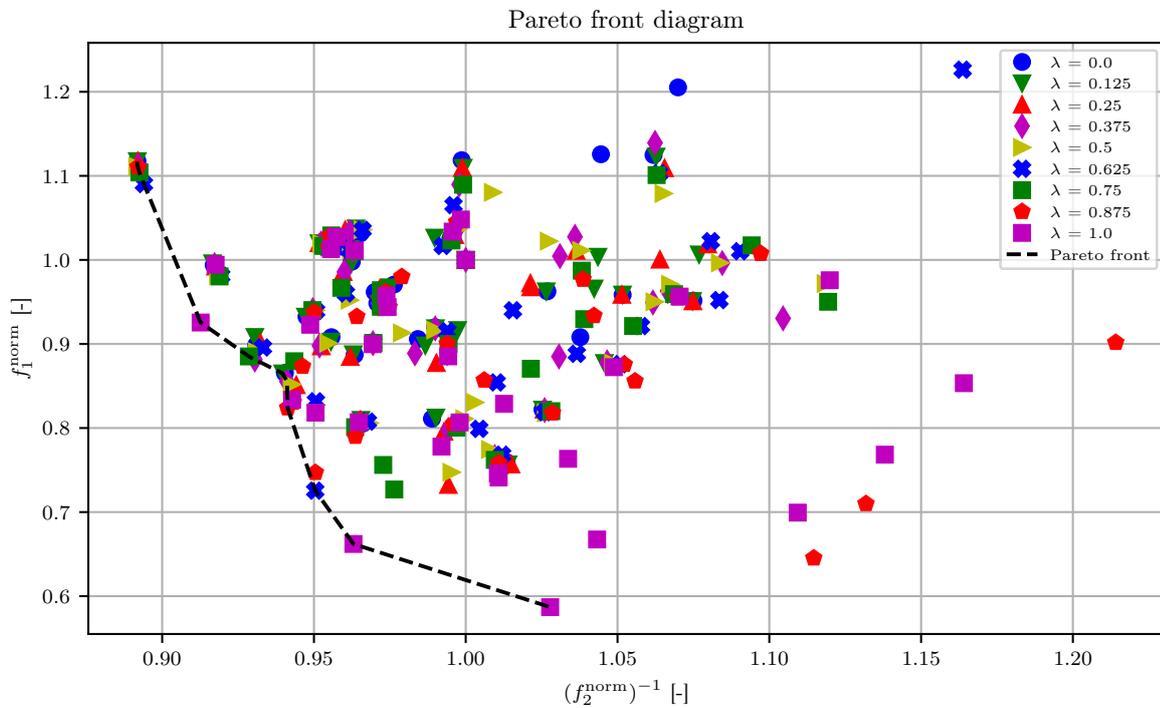


Figure 6.7: Pareto front diagram for minimizing installed thrust performance, at $J = 1.05$.

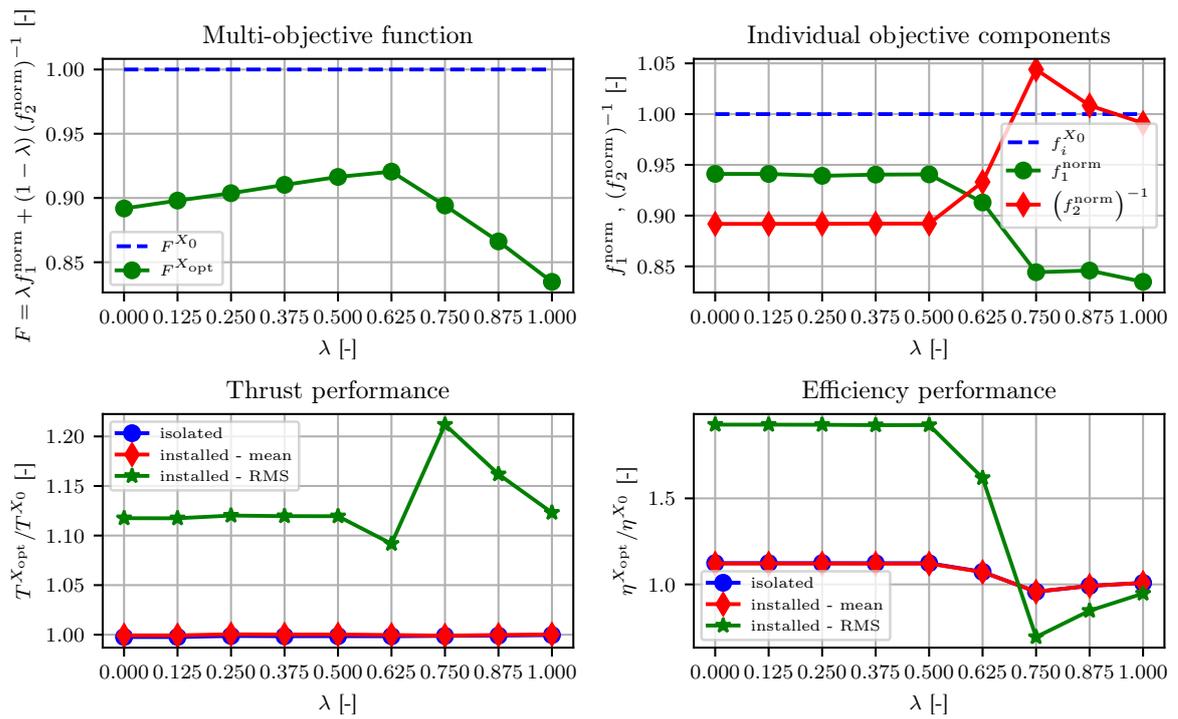


Figure 6.8: Overview of important optimization results for minimizing individual aerodynamic blade noise drivers, at $J = 1.05$.

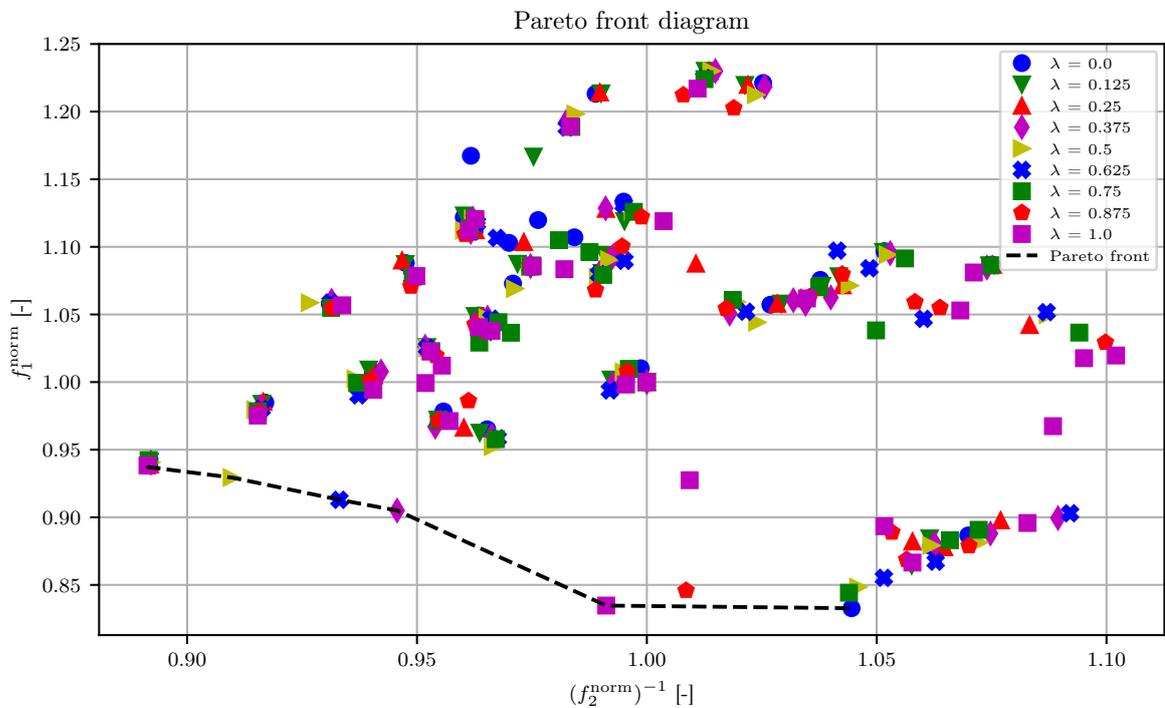


Figure 6.9: Pareto front diagram for minimizing individual aerodynamic blade noise drivers, at $J = 1.05$.

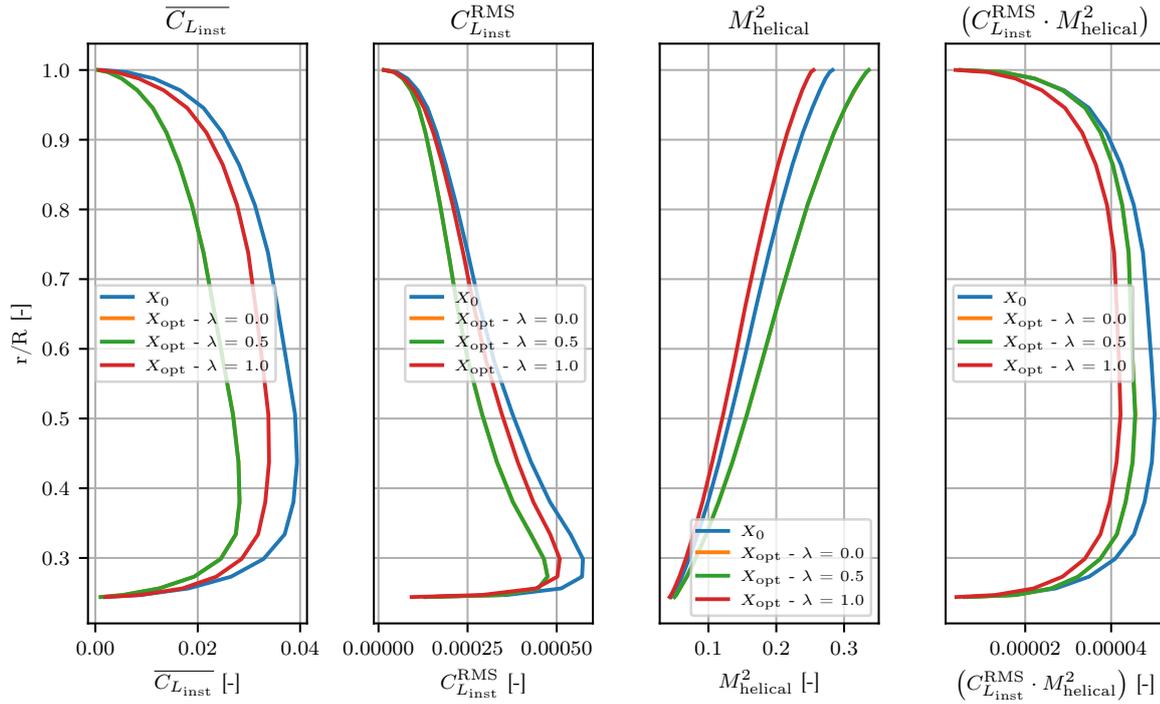


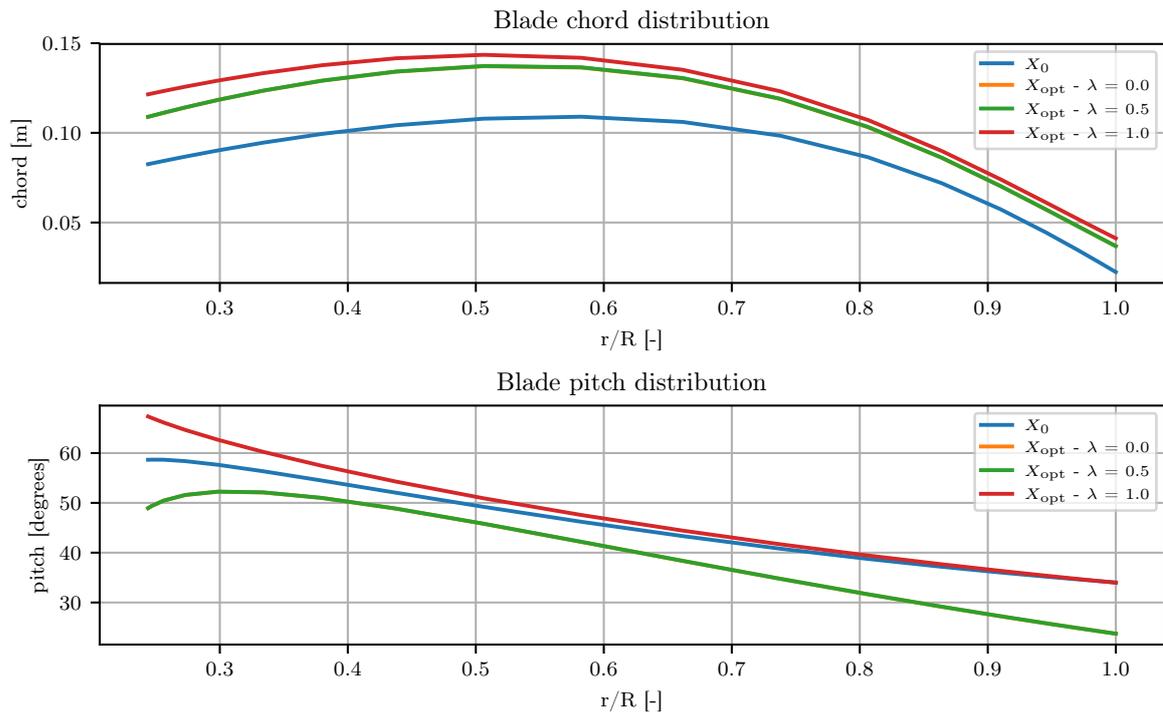
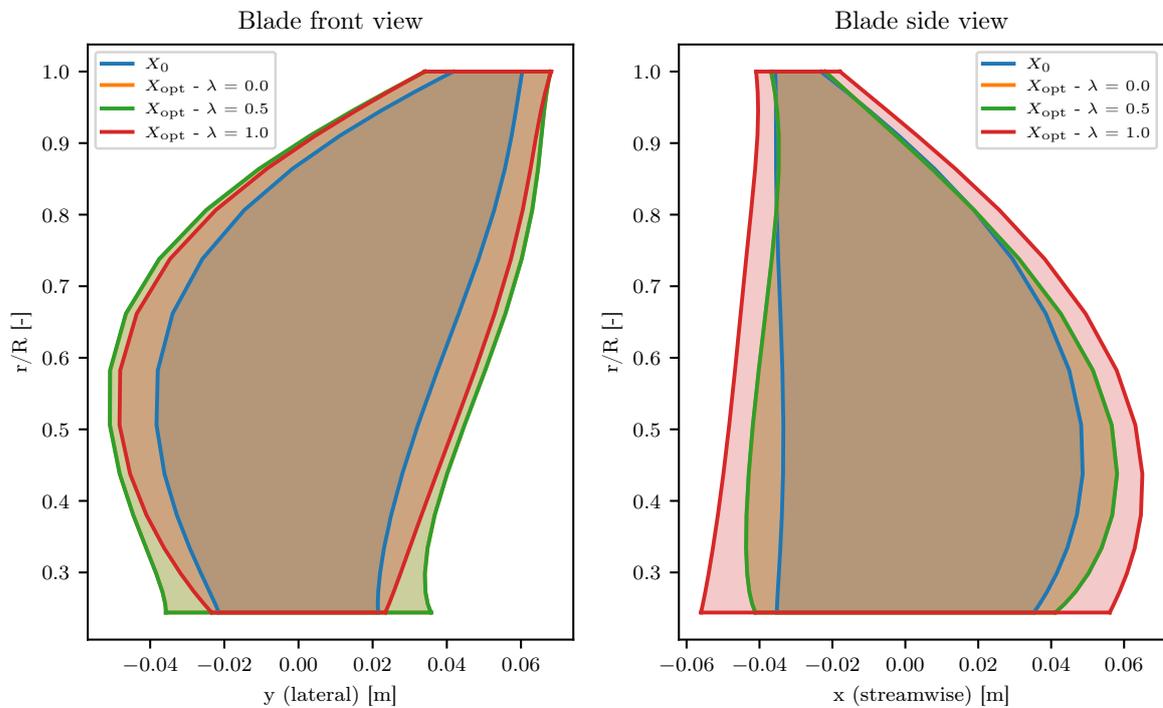
Figure 6.10: Installed thrust & torque performance for minimizing individual aerodynamic blade noise drivers, at $J = 1.05$.

the orange line belonging to λ equal to zero lies under the green line related to λ equal to 0.5. For the region where the efficiency objective term is dominant, the optimizer converges to a solution where it maximizes the propeller rotational velocity, which directly influences the helical Mach term. When increasing the rotational velocity while maintaining the same thrust, it is obvious that the lift coefficient distribution can be reduced. This explains why for λ equal to zero and 0.5, the mean lift coefficient distribution is the smallest. When the installed section noise objective term is dominant, the rotational speed is reduced, thereby requiring an increase in lift coefficient. What is interesting here is that the RMS section lift coefficient term distribution is only improved in the hub region. It is suspected that this is a consequence of a combination of the installed performance model and the formulation of this particular objective function. Figure 5.4 demonstrates that both the width, size and spacing of the unsteady lift coefficient contribution reduces in the blade tip region. The reduced frequency of the perturbation increases with increasing radial coordinate, since the local helicoidal velocity increases. As a result, the unsteady lift response will decrease. Moreover, a constant-width wake will span a decreasing circumferential range with increasing radial coordinate, making the peak 'sharper' (lower width). This results in significantly less unsteady behavior in the tip region, compared to the hub region.

The optimized geometry is given in figure 6.11 and in figure 6.12. In terms of chord, there is an increase compared to the reference propeller. In the blade pitch distribution it is clearly visible what the optimizer does in order to achieve the desirable lift distributions in figure 6.10. To achieve a lower lift coefficient distribution, the blade pitch distribution decreases, which in turn decreases the angle of attack. Compared to the geometry results for minimizing installed thrust performance in figure 6.5, the geometry for this candidate objective function is much more comparable to the reference geometry. It is suspected that this is mostly the result of a few dominant design variables (blade collective pitch, propeller rotational speed).

6.2.3. Minimizing combined aerodynamic blade noise drivers

In figure 6.13, the main characteristics of the optimization are plotted. The first two λ points converge to the same solution as was the case in the previous sub-section. Apparently, when the mean installed efficiency objective is dominant, there is either a global optimum which is found, or, it is possible that the gradient of the design space is somewhat flat and the optimizer converges early. After this, there is a significant jump over the next two points in the first objective component, as the efficiency term starts to increase. At the same time, a significant dip in installed thrust performance is visible, in the favorable direction. In the region where the second objective term is dominant, the optimizer converges once again to a single solution, which leads to the conclusion that this must be the maximum attainable improvement in installed section noise

Figure 6.11: Overview of optimized geometry for minimizing individual aerodynamic blade noise drivers, at $J = 1.05$.Figure 6.12: Front and side view of optimized geometry for minimizing individual aerodynamic blade noise drivers, at $J = 1.05$.

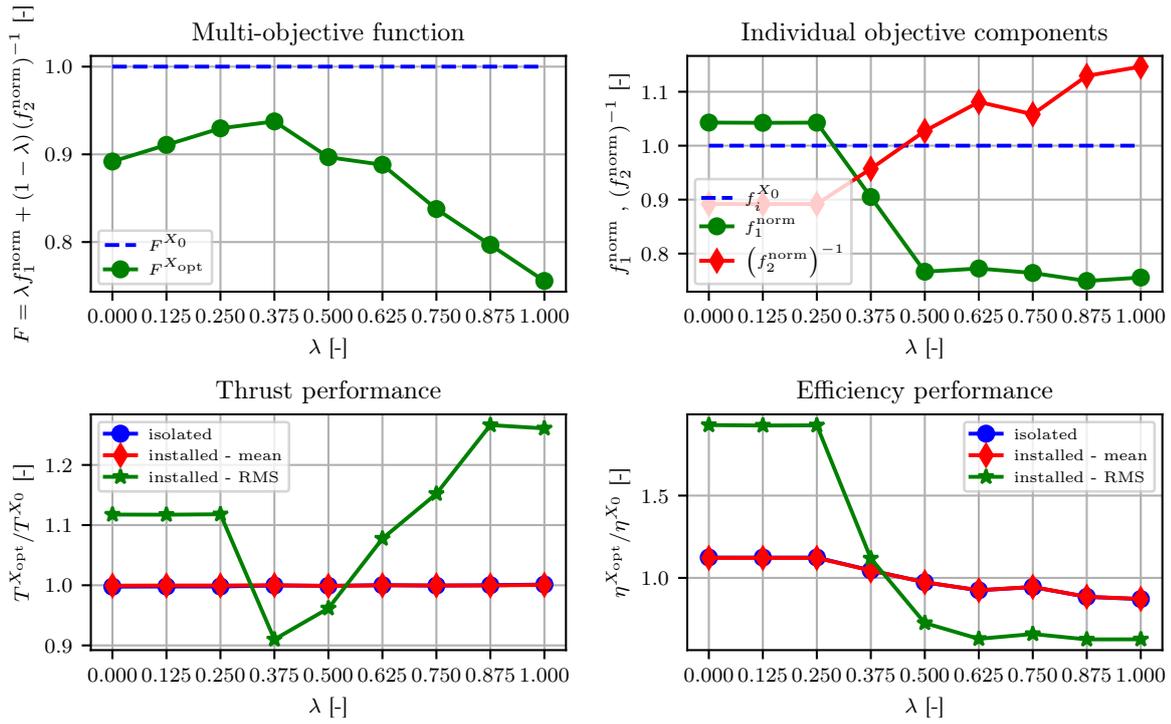


Figure 6.13: Overview of important optimization results for minimizing combined aerodynamic blade noise drivers, at $J = 1.05$.

performance, which is around 25%.

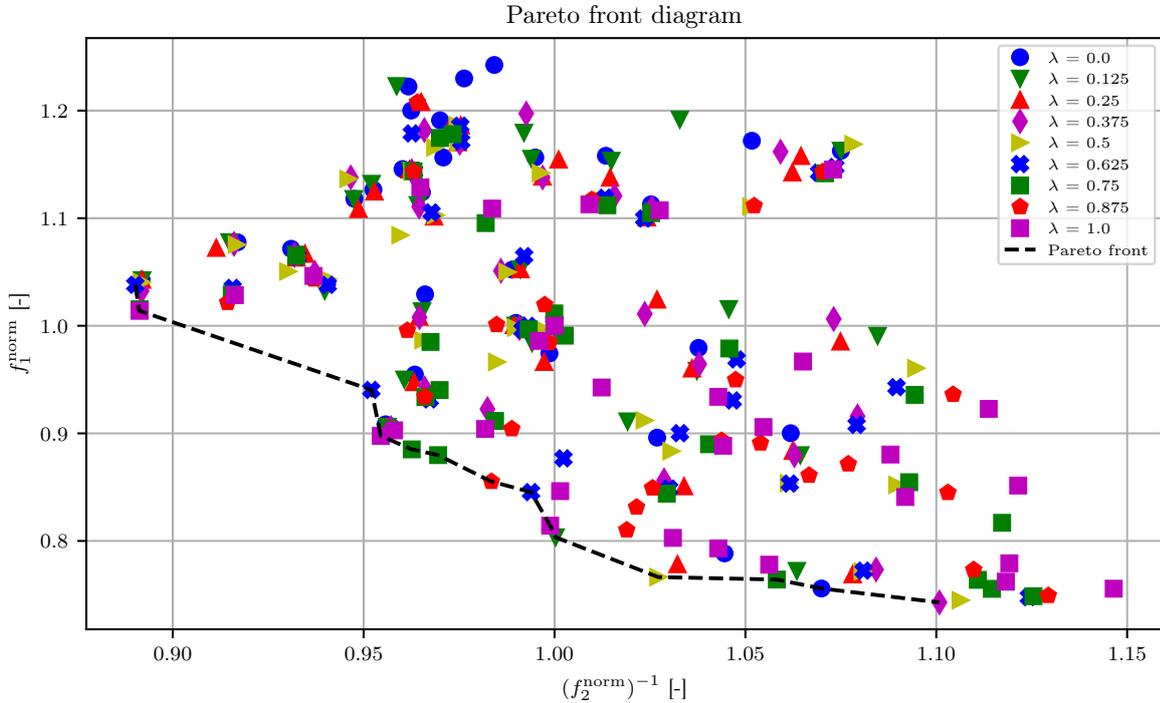


Figure 6.14: Pareto front diagram for minimizing combined aerodynamic blade noise drivers, at $J = 1.05$.

The expected benefit of the combined aerodynamic blade noise drivers objective function over the individual aerodynamic blade noise drivers objective function from the previous section is that this formulation is more realistic since it takes all design aspects into account, including the combined effect of all sections. However, beforehand it was not clear whether this would actually result in different design trends. Figure

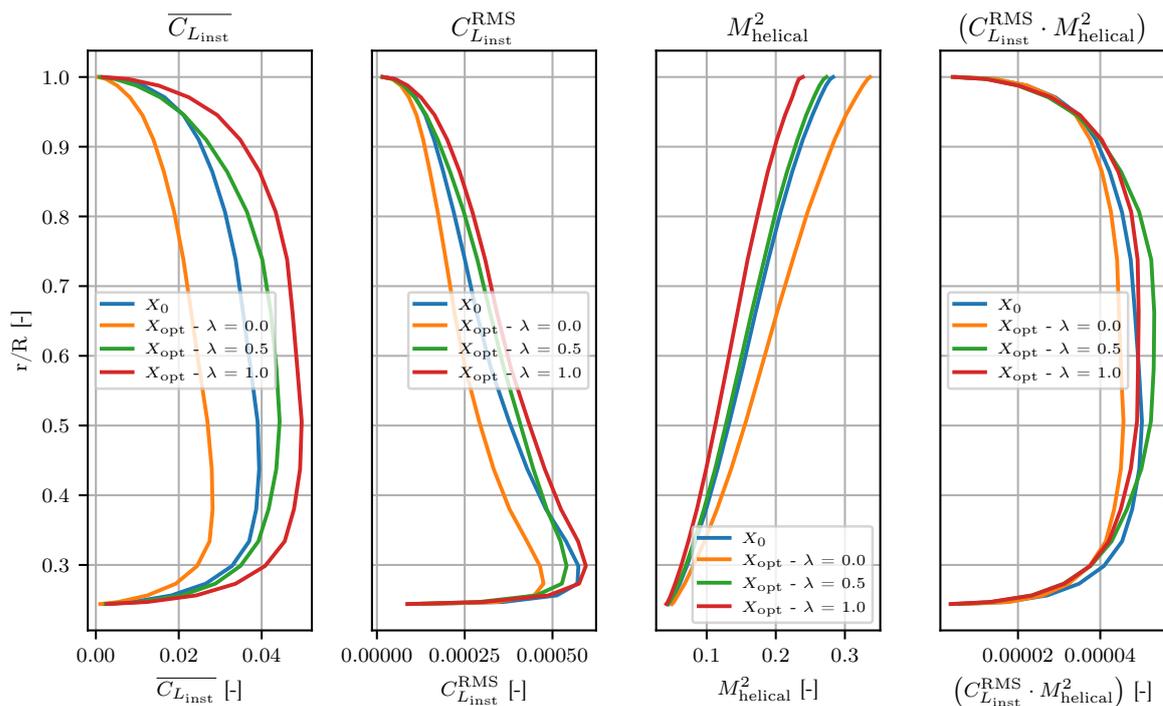


Figure 6.15: Installed thrust & torque performance for minimizing combined aerodynamic blade noise drivers, at $J = 1.05$.

6.15 shows that the design trends are in fact similar. Specifically, the mean lift coefficient distribution sees a bigger increase compared to the results in figure 6.10. The helical Mach number term behavior is the same; maximum propeller rotational speed for maximum efficiency and minimum propeller rotational speed for minimum installed blade noise performance. When the terms specific to the previous candidate objective function are analyzed, it becomes clear that optimized results for this objective function do not translate in optimum results for the other objective function related to noise.

When the geometries in figure 6.16 and in figure 6.17 are analyzed, it can be concluded that the application of the objective function results in geometries that are a mix between the other two applied objective functions. A similar trend is seen when compared to minimizing installed thrust performance geometries, where more chord is added in the hub region while the tip region is decreased. However, the chord distribution is not completely changed, instead the differences are more subtle as is the case in the geometries resulting from the minimization of the individual aerodynamic blade noise drivers. The increased lift distribution is here clearly the result of the increase in collective blade pitch. In the installed performance model of Sinnige et al. [5], blade pitch is directly related to the size of the normal gust velocity experienced at each blade section. The size of the normal gust velocity decreases for increasing blade pitch, thereby reducing the unsteady lift contribution due to the angle of attack perturbation

6.2.4. Selection of final objective function

Based on the findings presented above, the candidate objective selected for the remainder of this thesis is optimizing installed blade noise performance ($f_{noise_j}^{RMS}$). All three candidate objective functions showed that improvement in each respective objective is possible when applying the optimization routine. However, the argument is made here that optimizing installed blade noise performance ultimately will satisfy the ultimate goal of reducing the worst aspects of installation effects. Optimizing installed thrust performance resulted in the biggest improvement for that particular metric. However, this also resulted in the most radical change in propeller geometry, which, given the low-fidelity nature of the installed performance setup, needs to be approached with caution. Furthermore, this study has exposed that the installed thrust amplitude is about 0.4%, which is not very significant. Since noise is a far more critical limiting factor for advanced propellers for next-generation aircraft, the better option is to optimize for installed noise performance.

Between the two objective functions related to the unsteady-loading noise performance of the propeller formulated and studied here, optimizing installed blade noise performance is selected over section noise, because in the end it is a more realistic metric. Optimizing for installed section noise performance revealed that there are two dominant design variables (collective blade pitch and propeller rotational speed) that force

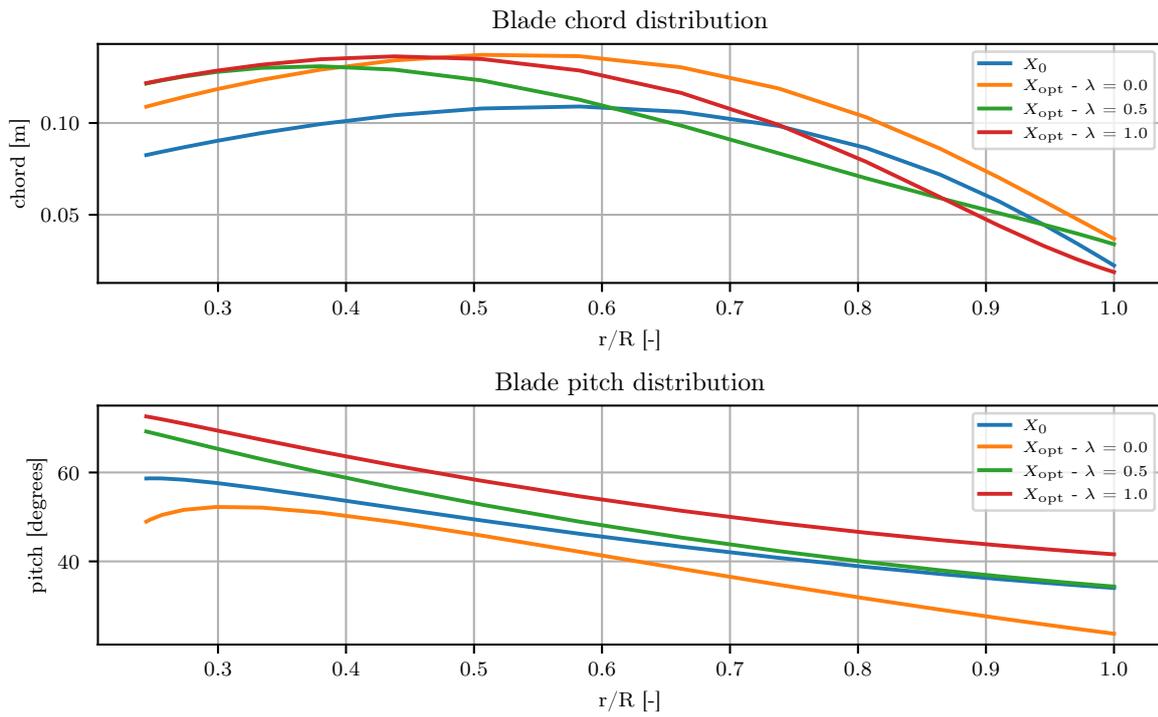


Figure 6.16: Overview of optimized geometry for minimizing combined aerodynamic blade noise drivers, at $J = 1.05$.

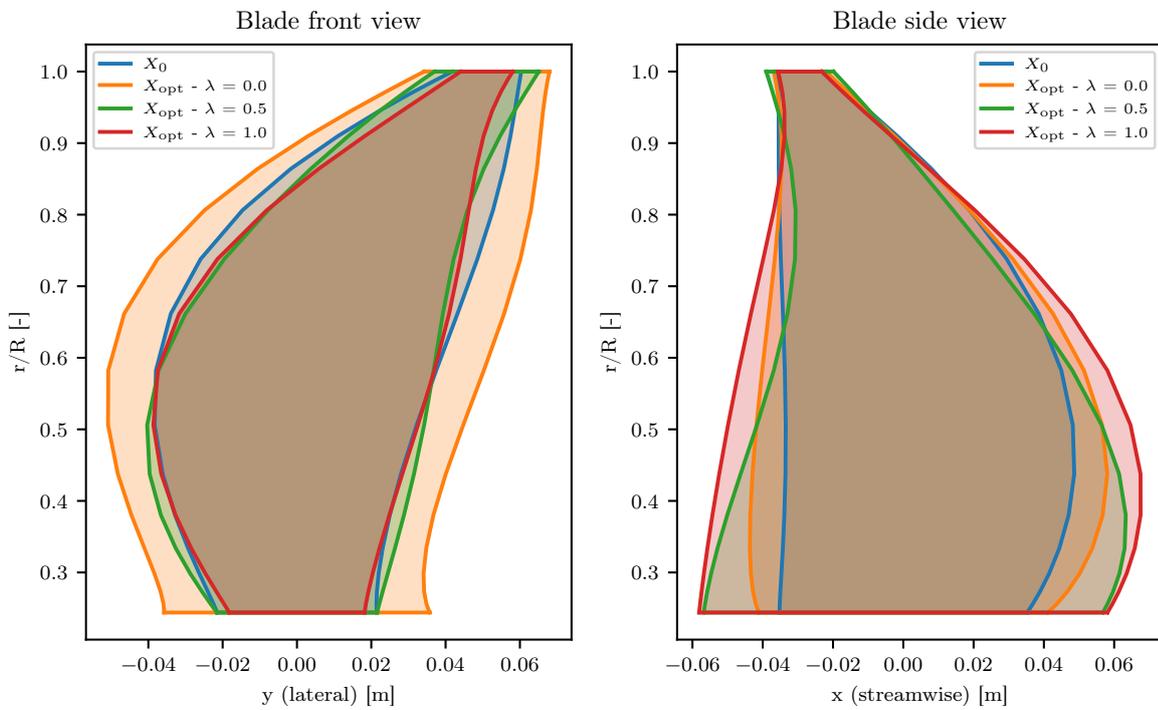


Figure 6.17: Front and side view of optimized geometry for minimizing combined aerodynamic blade noise drivers, at $J = 1.05$.

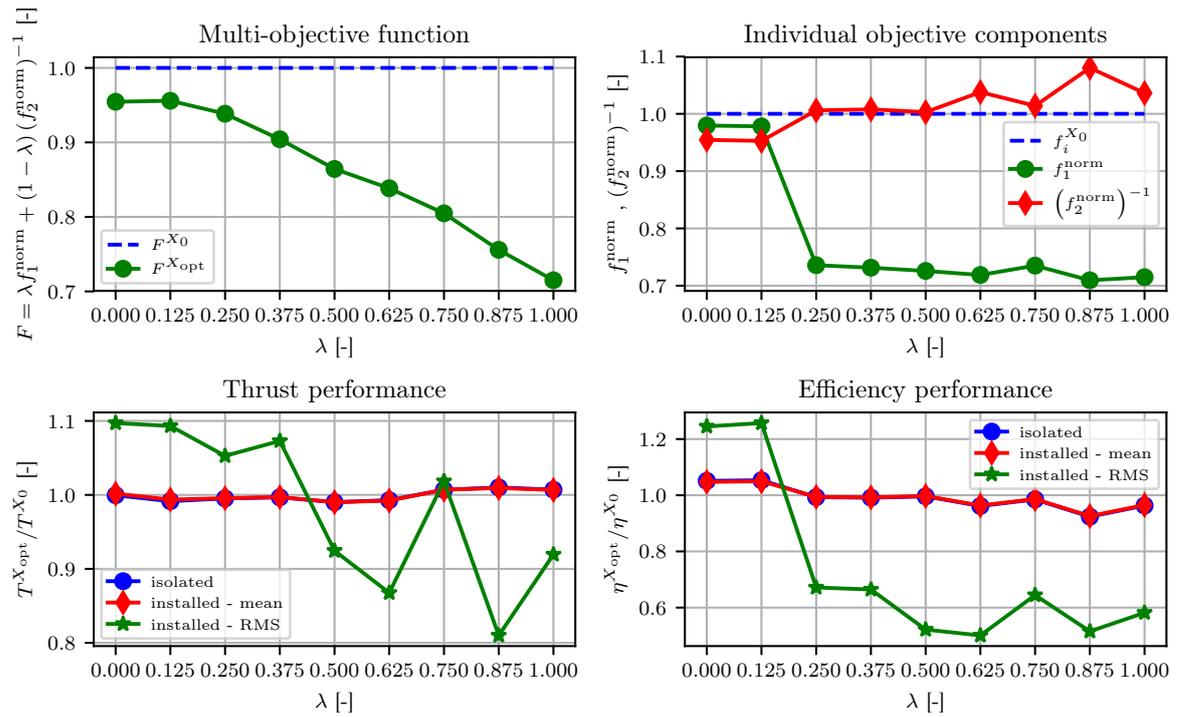


Figure 6.18: Overview of important optimization results for minimizing combined aerodynamic blade noise drivers, at $J = 1.40$.

the optimizer to converge at similar design points. This also results in a very clustered Pareto diagram, which introduces uncertainties about gaps in the design space and the validity of selected optimum design vectors. Therefore, optimizing the combined aerodynamic blade noise drivers is the best option.

6.3. Different operating point optimization study

Two additional operating points are selected to study the influence of operating conditions on potential mitigation of the adverse installation effects. In the previous section, one operating point at a high thrust setting corresponding to an initial advance ratio of 1.05 was used to study potential objective functions. The combined aerodynamic blade noise drivers objective function is applied to the remaining available operating points at lower thrust settings corresponding to advance ratios of 1.40 and 1.75.

6.3.1. Optimization at intermediate thrust setting: 1.40

In figure 6.18, the main characteristics of the optimization at intermediate thrust setting are plotted. From an initial standpoint, the overall objective trend is very similar to the one that was found for high thrust setting in figure 6.15. The next two λ points converge to the same solution as the mean efficiency objective is dominant. Also both objective goals show a contradictory trend as mean installed efficiency becomes worse for increasing λ . However, one significant difference is that the initial efficiency improvement is lower compared to the results at $J = 1.05$. This is most likely caused by the fact that at higher advance ratios, efficiency will be higher as well. In this operating region, the isolated efficiency is not yet close to the maximum attainable value which means that the performance is not yet affected by a decreasing slope of the efficiency curve. What is significant is that at $J = 1.05$, optimizing both the installed noise and thrust performance resulted in contradictory design trends, whereas for the intermediate thrust setting, both are optimized at the same time. Just as before, when the first objective term becomes dominant the optimizer converges to solutions that give more-or-less the same improvement in installed noise performance.

Figure 6.19 shows the Pareto diagram that corresponds to $J = 1.40$. The spread of the feasible points is good, its not too spread out but also not clustered. The pattern is similar to the results of $J = 1.05$. There exists a gap between the Pareto front and the feasible points around the coordinate (1,0, 0.8), it is unclear why the optimization did not find any feasible solutions here. It is also important to notice the trend that for increasing λ , the points per λ pool spread more and more. The implication of this that for the maximum efficiency objective, the feasible solutions converge to similar designs.

In appendix C, additional results are given for $J = 1.40$. Specifically, different optimum geometries for dif-

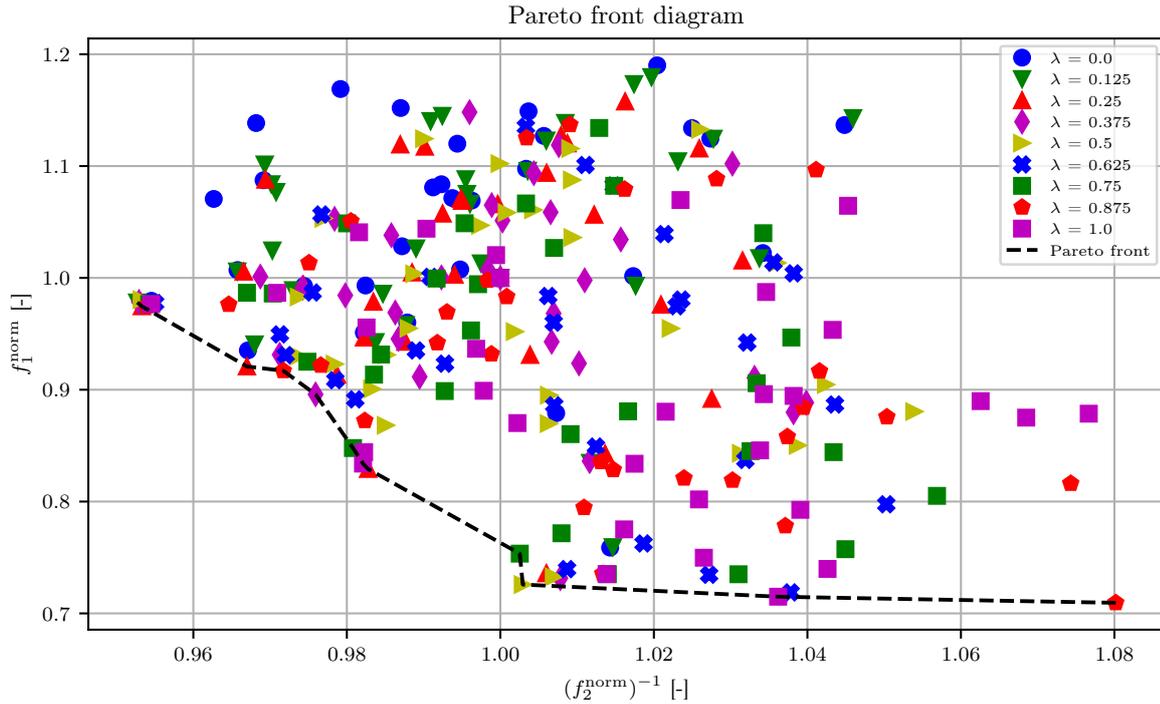


Figure 6.19: Pareto front diagram for minimizing combined aerodynamic blade noise drivers, at $J = 1.40$.

ferent weight factors as well as a detailed overview of installed thrust and torque performance. This detailed overview explains why the optimized combined aerodynamic blade noise drivers in figure 6.18 are different compared to the results for $J = 1.05$. The unsteady time history is completely different and the amplitude is also bigger. While installed thrust performance is no longer considered as an objective, it can still act as a criterion for when a final design is selected.

6.3.2. Optimization at low thrust setting: 1.75

In figure 6.20, the main characteristics of the optimization at low thrust setting are plotted. It shows a significant departure from the similar trends at $J = 1.05$ and 1.40 . The multi-objective function plot shows a linear trend. However, this is most likely caused by the variation of the weight factor. In the individual objective components plot (upper-right) it is clearly shown that throughout the entire optimization process, the optimizer has found approximately two feasible solutions that are minima, at λ equal to zero and one for all the remaining points. The feasible points shown here are better than the reference in all metrics considered. At a low thrust setting, the propeller operates in the region of maximum attainable isolated efficiency, which will affect the optimized result. Also, the equal thrust constraint is related to the baseline thrust performance. If the baseline thrust is low, the thrust constraint becomes more strict. This means that the optimizer first has to put a lot of effort navigating through the design space to reach at minimum a feasible design point. It is more likely that, when it has difficulty locating such a point, once it does then finally reach a feasible point, it will stop the optimization prematurely, before finding a global minimum.

More evidence of this can be found in the Pareto diagram in figure 6.21. The difference in scale between the f_1 and f_2 axis gives a skewed overview of the Pareto diagram. The feasible mean efficiency range is very limited. The Pareto front consists of optimized solutions, which were obtained by starting from some random points. The increase in spread of optimum solutions for increasing λ is also clearly visible.

In appendix C, additional results are given for $J = 1.75$. Specifically, different optimum geometries for different weight factors as well as a detailed overview of installed thrust and torque performance. What is interesting is that for every operating point considered here, the overall geometric shapes that are created are all very similar. Based on this, it is expected that when the optimum design points are analyzed at different operating points than originally considered, there will be no significant performance differences.

6.3.3. Selection of optimum designs

In order to make a selection of the final optimum solution, first, a pre-selection is made from the large pool of feasible designs from the three optimization studies in sub-sections 6.2.3, 6.3.1 and 6.3.2. Two requirements

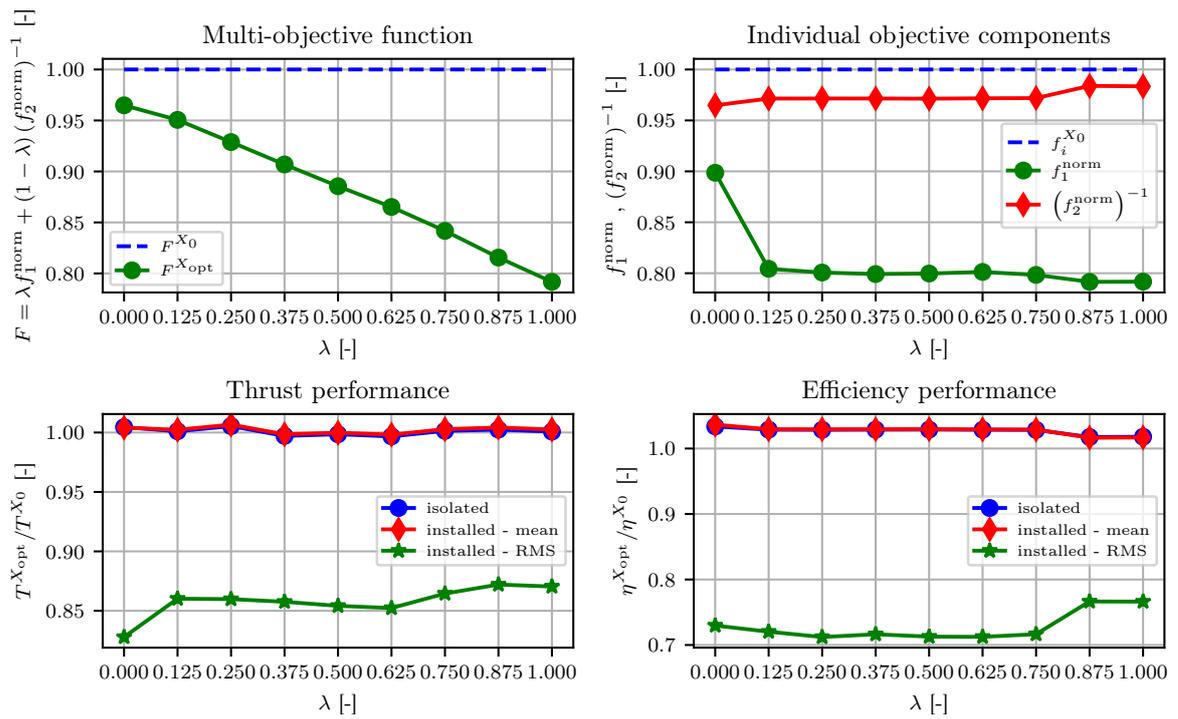


Figure 6.20: Overview of important optimization results for minimizing combined aerodynamic blade noise drivers, at $J = 1.75$.

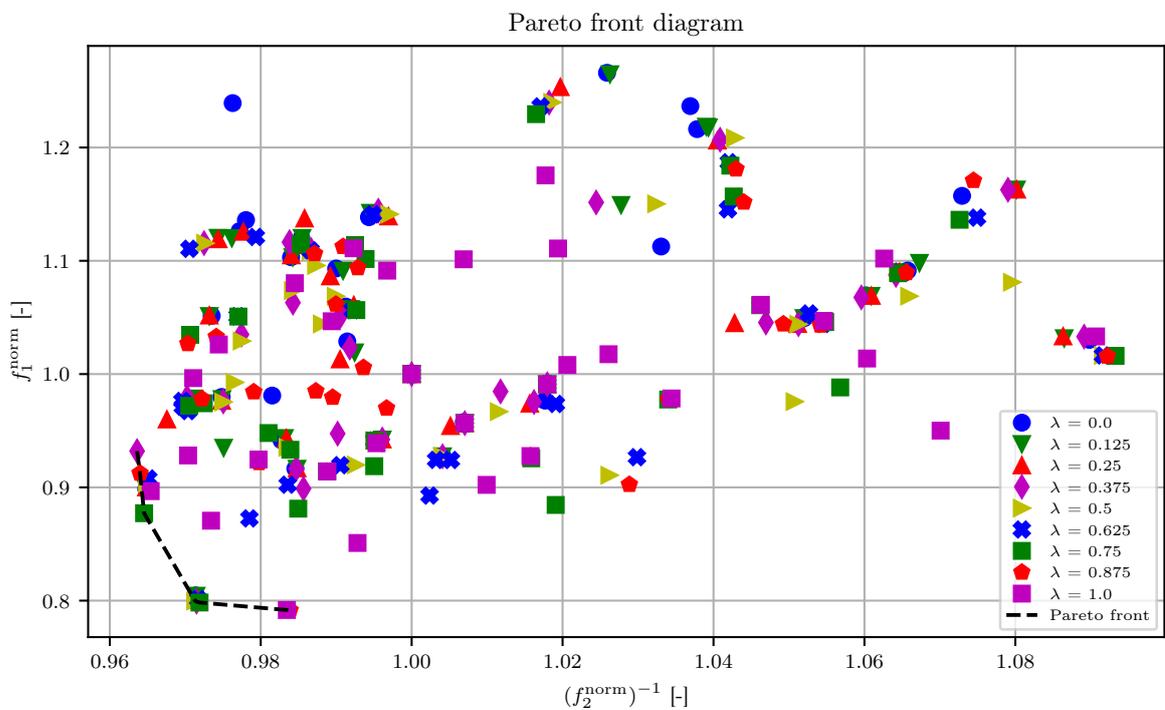


Figure 6.21: Pareto front diagram for minimizing combined aerodynamic blade noise drivers, at $J = 1.75$.

Table 6.1: Overview of selected optimized design vectors at three initial advance ratios.

Design variable	Baseline		J = 1.05		J = 1.40		J = 1.75	
	X_{norm}	X	X_{norm}	X	X_{norm}	X	X_{norm}	X
Advance ratio shift [-, -]	0.5	0.0	0.236	0.0528	0.086	0.0828	0.0063	0.0987
Blade pitch angle at 0.75 r/R [-, degrees]	0.4985	39.97	0.6111	42.22	0.5842	41.68	0.4913	39.83
Chord CP 1 [-, m]	0.5912	0.0825	0.8985	0.1121	1.0	0.1219	0.9997	0.1218
Chord CP 2 [-, m]	0.5538	0.1243	0.9979	0.1678	0.9809	0.1661	0.9982	0.1678
Chord CP 3 [-, m]	0.5330	0.1184	0.8737	0.1539	0.0045	0.0633	0.8783	0.1544
Chord CP 4 [-, m]	0.8208	0.084	0.4778	0.0662	0.8765	0.0869	0.4772	0.0661
Chord CP 5 [-, m]	0.3864	0.0224	0.8756	0.0377	1.0	0.0416	0.5452	0.0274
Twist CP 1 [-, degrees]	0.5	18.68	0.6031	20.23	0.9974	26.14	0.7329	22.17
Twist CP 2 [-, degrees]	0.5	19.64	0.7641	23.60	0.9791	26.83	0.7567	23.49
Twist CP 3 [-, degrees]	0.5	2.762	0.1413	-2.619	0.8707	8.322	0.7835	7.014
Twist CP 4 [-, degrees]	0.5	-5.947	0.4721	-6.366	0.0998	-11.95	0.4289	-7.014

were formulated to aid in the selection of the final design. First, each prospective design must lie on the Pareto front. Second, the mean installed efficiency performance must not be 1% worse than the reference propeller performance. Based on these requirements, three candidate optimum designs are:

$$\begin{aligned} \left[f_1 \left(X_{\text{opt}}^{J=1.05} \right), f_2^{-1} \left(X_{\text{opt}}^{J=1.05} \right) \right] &= [0.8034, 1.0003] \\ \left[f_1 \left(X_{\text{opt}}^{J=1.40} \right), f_2^{-1} \left(X_{\text{opt}}^{J=1.40} \right) \right] &= [0.7258, 1.0029] \\ \left[f_1 \left(X_{\text{opt}}^{J=1.75} \right), f_2^{-1} \left(X_{\text{opt}}^{J=1.75} \right) \right] &= [0.7917, 0.9835] \end{aligned}$$

6.4. Detailed performance of selected optimum designs

This section gives the results of a detailed performance study for the three candidate optimum designs that were found in the previous section. First, a detailed overview is given of the three solutions. In the second part, each design is analyzed over the entire performance range of the reference propeller. Based on these findings, one design is selected and presented as the final optimum solution.

6.4.1. Overview of optimum designs

Table 6.1 gives a complete overview of the optimum design vectors, while figures 6.22 and 6.23 give a graphic overview of the blade geometry. Two interesting observations can be made from table 6.1. First, due to the formulation of the objective function, beforehand it was expected that the optimizer would prefer a maximum increase in advance ratio. This design variable has a direct influence on the helical Mach number term, but the data have revealed that actually the advance ratio shift only progressively goes to the upper bound. It was added to aid the optimizer in meeting its thrust constraint; however, at $J = 1.05$, the optimizer cannot afford to give up more rotation speed and still meet its thrust constraint. The second interesting observation is that, while beforehand it was thought that the collective blade pitch angle would work in a similar way as the propeller rotational speed, in that it has a bigger impact than the remaining design variables. However, the data have revealed that the blade pitch angle is relatively similar compared to the reference propeller design, so apparently the twist distribution is more important here and the collective pitch was already at a nearly optimal value.

In the geometry plots in figures 6.22 and 6.23, it becomes very clear how similar the optimum designs for $J = 1.05$ and $J = 1.75$ are. Both have independently converged to the almost the same solution. Only in the tip area there are some minor differences: The optimum solution for $J = 1.05$ and 1.40 have an increased tip chord with respect to the reference, while the solution for $J = 1.75$ has a decreased tip chord. For all three solutions, the trend is more chord in the hub region, then converge to the reference chord distribution around $0.9 r/R$, after which each has a different chord distribution near the tip. A similar trend is also observed in the blade pitch distribution, where more pitch is added in the region from hub to $r/R = 0.7$, convergence to the reference pitch distribution at $0.85 r/R$ and finally a different twist distribution near the tip.

An overview of the detailed performance characteristics at optimized design conditions is given in table 6.2. Isolated and installed time-averaged thrust are very similar compared to the baseline for each operating point, since both are dominated by the equal thrust constraint. Furthermore, since each optimum design was

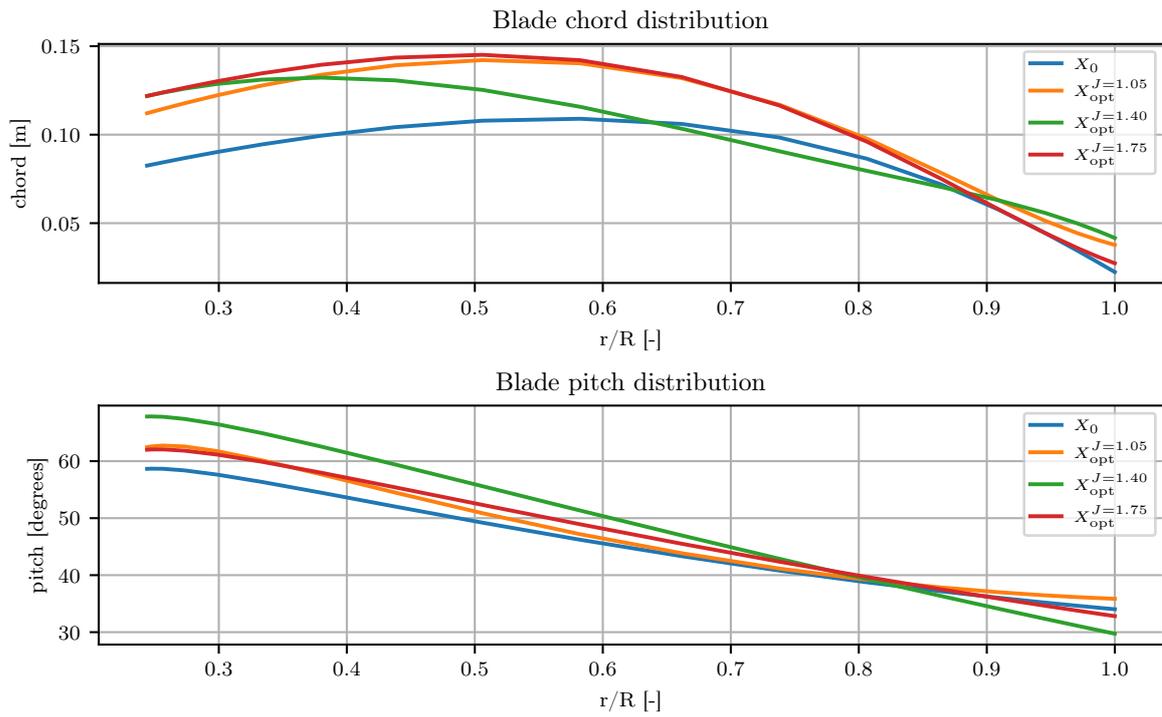


Figure 6.22: Overview of selected optimized geometry, for different operating points.

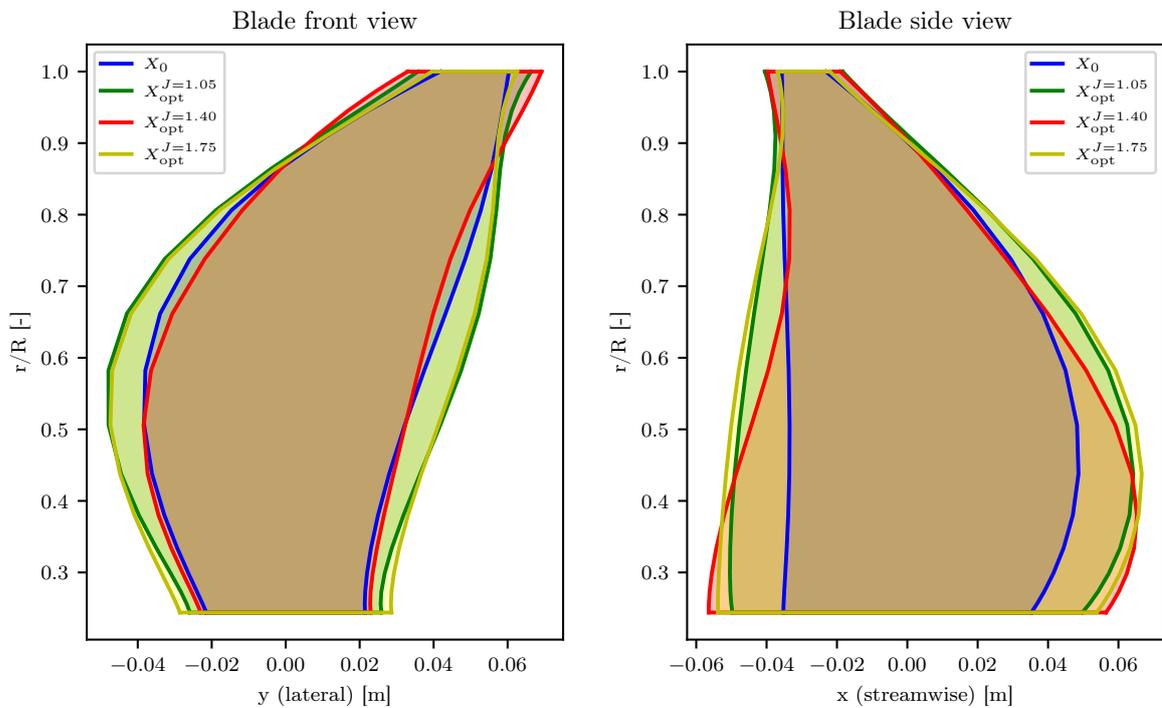


Figure 6.23: Selected front and side view of optimized geometry, for different operating points.

Table 6.2: Overview of selected optimized performance at three initial advance ratios.

Variable	J = 1.05		J = 1.40		J = 1.75	
	F/F_{X_0}	F	F/F_{X_0}	F	F/F_{X_0}	F
Thrust - Isolated[-, N]	0.999	468.9546	0.9901	180.5721	1.0008	50.966
Thrust - Installed mean [-, N]	0.9995	469.5227	0.9902	181.0203	1.0025	51.476
Thrust - Installed RMS [-, N]	1.0948	0.7346	0.9245	0.2874	0.8704	0.1591
Torque - Isolated[-, Nm]	1.0427	74.9999	1.0512	30.5751	1.0386	9.2335
Torque - Installed mean [-, Nm]	1.0432	75.0735	1.0501	30.7144	1.0416	9.4998
Torque - Installed RMS [-, Nm]	1.0395	0.146	0.7171	0.0679	0.8278	0.0735
Efficiency - Isolated[-, N]	0.9996	0.5536	0.996	0.7069	1.0179	0.8233
Efficiency - Installed mean [-, N]	0.9997	0.5532	0.9971	0.7048	1.0168	0.8074
Efficiency - Installed RMS [-, N]	0.8823	0.0003	0.5218	0.0006	0.7663	0.0038
$J_{noise_j}^{RMS}$ [-, -]	0.8034	0.0003	0.7258	0.0002	0.7917	0.0002

selected with a minimal efficiency penalty in mind, it is not surprising to see that both isolated, mean installed torque and efficiency are also very close to the reference propeller. Another trend that is confirmed here is that for increasing advance ratio, installed thrust performance improves. It appears that the different geometries for $J = 1.40$ on the one hand, and $J = 1.05$ and $J = 1.75$ on the other hand, mainly manifest in a difference in RMS of the torque and as a consequence, efficiency: for $J = 1.40$ there is a significant improvement compared to the other two. Finally, the the main objective metric is discussed: combined aerodynamic blade noise drivers. The similar optimum designs obtained for $J = 1.05$ and $J = 1.75$ result in similar improvement of approximately 20%, while the optimum design for $J = 1.40$ outperforms with a 27% improvement to the reference propeller. Based on these results, the design obtained from the optimization at $J = 1.40$ would seem the best option. However, the performance of the three blade geometries needs to be checked first in off-design conditions, which is studied in the next sub-section.

6.4.2. Optimum designs in off-design conditions

The purpose of this study is to investigate whether one or more produced solutions are able to perform on a similar level throughout the operating range. This is important because it will say something about the applicability of an improved propeller design. Until now, three initial operating points are used throughout this research. However, since an advance ratio shift was one of the design variables, the optimum designs are not valid for the baseline operating points. Instead, for the first analysis, a discrete distribution of new advance ratios is defined, at which the mean installed performance is compared with the reference propeller. The results are plotted in figure 6.24.

In all optimized solutions, both the thrust and torque are increased compared to the baseline design. The relative step in torque is greater resulting in a reduction in efficiency, however this also postpones the maximum efficiency to a higher advance ratio. When comparing the three optimized designs, it is concluded that the similarity of the solutions obtained for $J = 1.05$ and $J = 1.75$ is once more confirmed. They perform better in terms of efficiency, but worse in terms of maintaining thrust performance for increasing advance ratio. While figure 6.24 gives a good initial overview of overall propeller performance characteristics, it is not a fair way to directly compare different designs. In theory it is possible to change the collective blade pitch angle or change the advance ratio in order to modify the same performance. Therefore, a second study is performed in which these initial results are used to make a fair comparison, by using equal thrust performance as a common denominator. This is in line with what was done during the optimization with the equal thrust constraint. For this study, adverse installed performance characteristics are investigated; the results are plotted in figure 6.25.

The results shown in figure 6.25 are normalized with the corresponding reference value for the same thrust. In the top plot the components of the multi-objective function are given. The mean installed efficiency trend is gradually decreasing, however it is always an improvement over the reference propeller. The curve for $J = 1.40$ (green) performs slightly worse compared to the others, but the quantify the difference here to prove that it is not significant. However the maximum difference with the other designs is within 3%, which is considered insignificant here. This trend corresponds with the efficiency trend displayed in figure 6.24. However, in the case of the optimum blade noise metric, the differences are more significant and the optimized design at $J = 1.40$ is the better solution. This conclusion also applies to what can be considered secondary optimization goals, such as unsteady thrust and torque performance. Both the RMS of the thrust and torque at $J = 1.40$ outperform the other optimum designs. The approach used in the optimization study

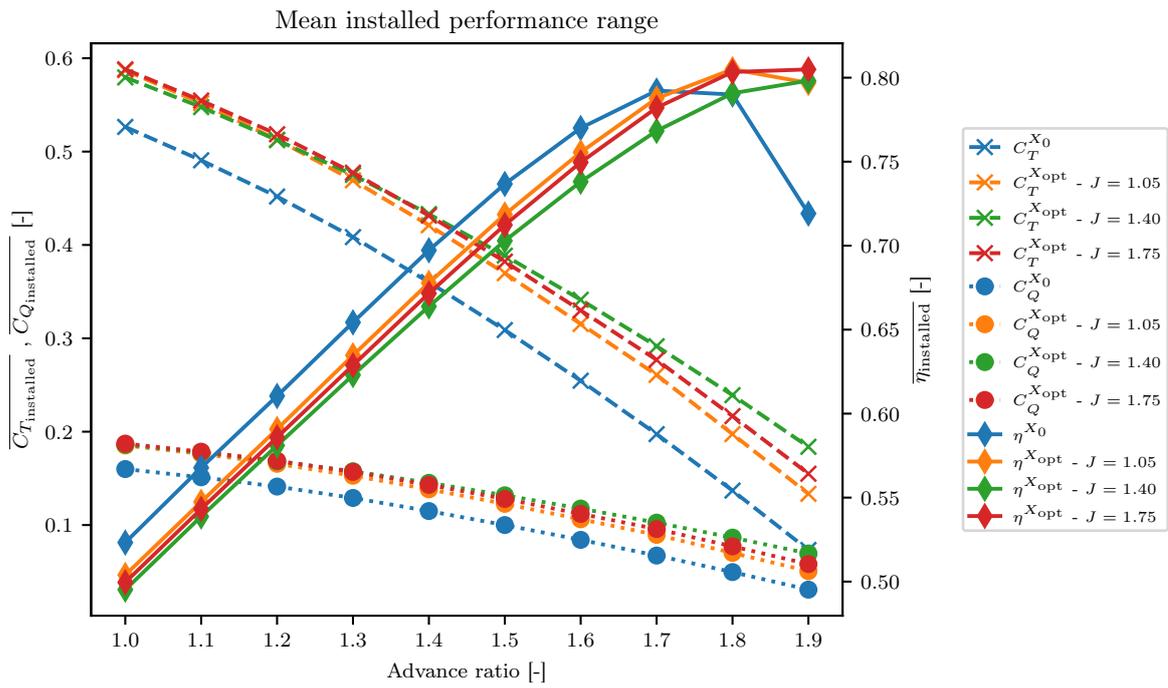


Figure 6.24: Propeller performance range, giving mean installed thrust, torque and efficiency performance for varying advance ratios.

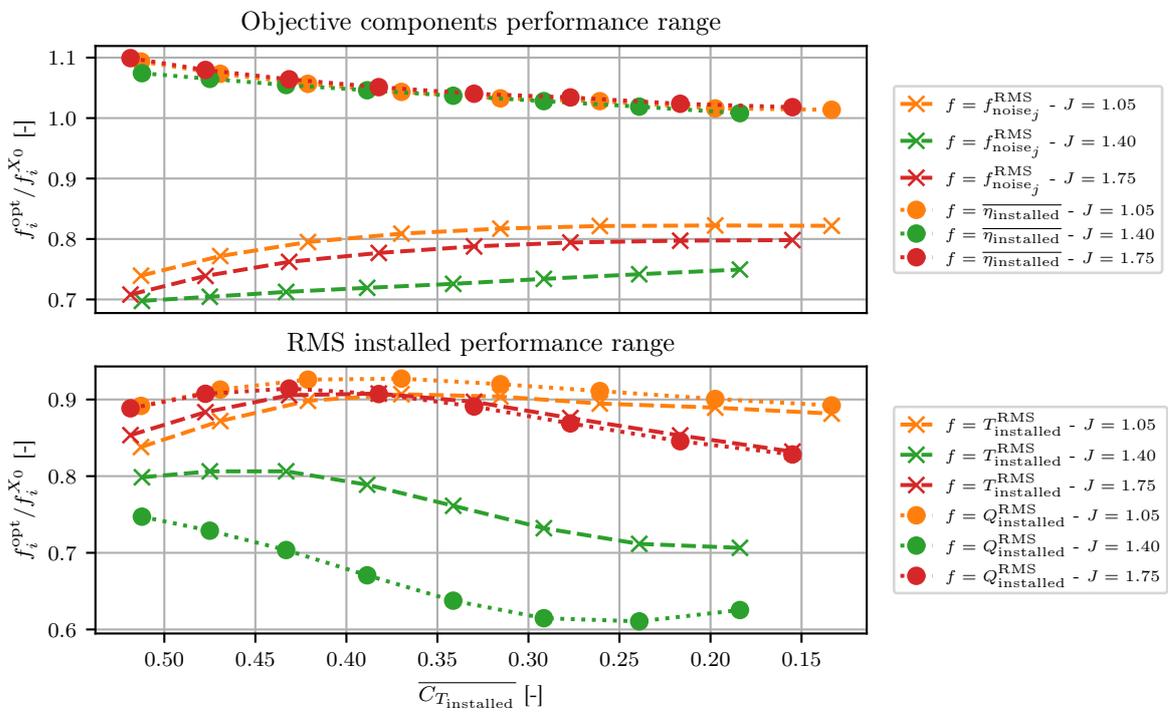
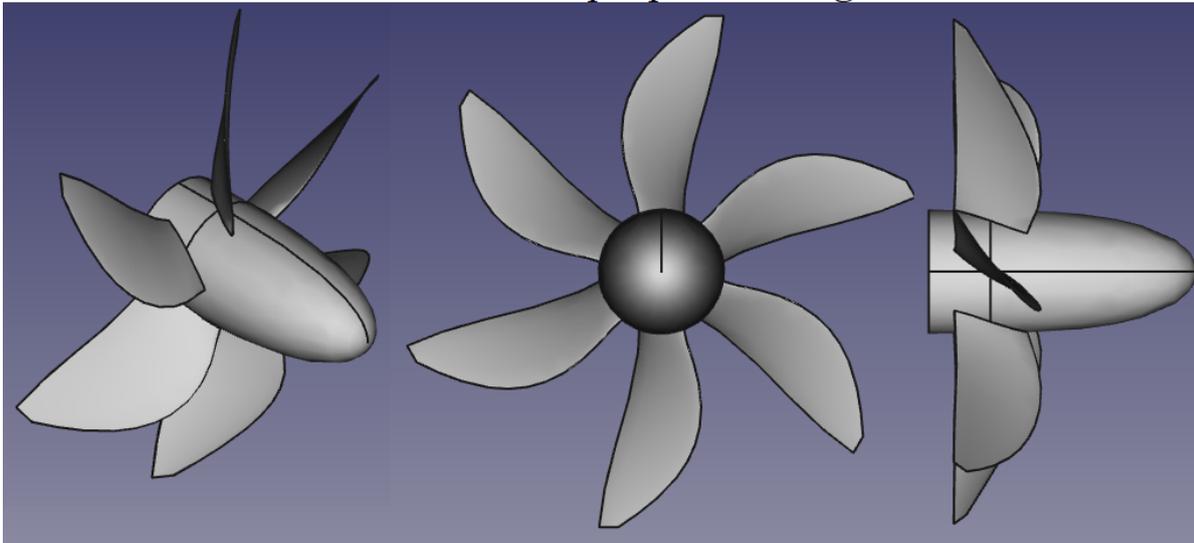


Figure 6.25: Selected front and side view of optimized geometry, for different operating points.

APIAN baseline propeller design



Final optimized propeller design

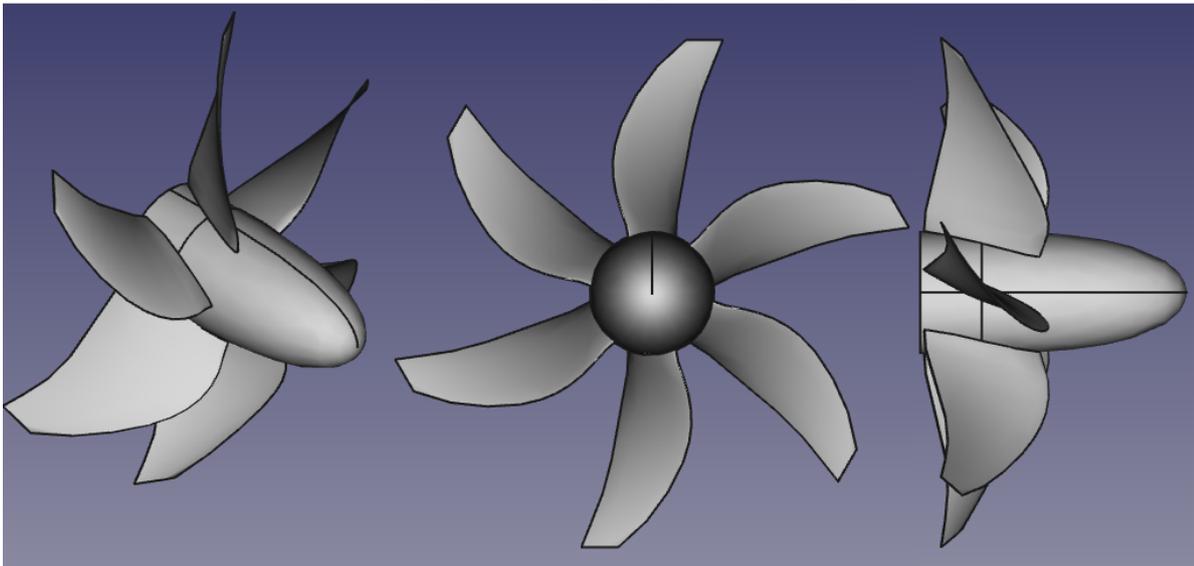


Figure 6.26: Impression of final optimum propeller design, created with the parametrization tool of chapter 4.

was exactly the same for all optimum designs. While the exact reason for the observed difference in performance of the three blade designs is unclear, it is most likely caused by the fact that the difference in slope of the reference thrust coefficient and the optimum thrust coefficient in figure 6.24 is greater for $J = 1.40$. This means that for the same thrust performance, the positive step in advance ratio is greater, which means a bigger reduction in propeller rotation speed. This in turn lowers the angle of attack, which results in a reduction in lift and drag coefficient. The important effect here is that the helicoidal Mach number is decreased by the reduction of the rotational speed.

6.4.3. Selection of final optimum design

Based on the results of this section, it is concluded with reasonable confidence that the best performing optimized design, throughout the operating range is the design found when optimization at an initial advance ratio of 1.40. Figure 6.26 gives an impression of the differences between the baseline and the final propeller geometry.

The final design only performs slightly worse in mean installed efficiency compared to the other candidate optimum designs at $J = 1.05$ and $J = 1.75$, but the differences are small enough that any penalty resulting from this is insignificant. More importantly, the potential gains in the installed blade noise metric and the

unsteady thrust and torque performance are significantly better for $J = 1.40$ than for $J = 1.05$ and $J = 1.75$. For a high thrust setting, the final optimum solution was a 30% installed noise metric improvement over the baseline propeller design. At the same thrust setting, the installed propeller efficiency improved by 8%, the unsteady fluctuations in installed thrust performance were reduced by 20% and the unsteady fluctuations in installed torque performance were reduced by 26%. For a low thrust setting, the final optimum solution was a 25% installed noise metric improvement over the baseline propeller design. At the same thrust setting, there was no significant improvement in installed propeller efficiency, the unsteady fluctuations in installed thrust performance were reduced by 29% and the unsteady fluctuations in installed torque performance were reduced by 36%.

Conclusions & Recommendations

The purpose of this chapter is to present conclusions and recommendations for future work, based on the findings presented in this thesis. The conclusions and recommendations are discussed in sections 7.1 and 7.2, respectively.

7.1. Conclusions

The topic of this thesis focused on the unwanted unsteady effects imposed on the aerodynamic characteristics of a propeller, imposed by an upstream pylon. This is relevant for rear-mounted open rotor configurations, installed in a pusher configuration to minimize cabin noise by placing the propellers as far away as possible from the passengers. The hypothesis of this thesis is that there is a potential for reducing unwanted propeller-pylon installation effects by taking these effects into account during the aerodynamic design process of the propeller. In order to realize this, a propeller design optimization routine was set up to study this potential benefit. This required the setup of different tools and methods. First, the extension of the existing turbo-machinery parametric modeling tool “OptiBlade” that models the propeller geometry. Second, the implementation of an existing low-fidelity isolated propeller performance analysis method, for rapid analysis of isolated propeller performance. Third, the implementation of a high-fidelity isolated propeller performance analysis method, in order to verify the low-fidelity isolated propeller performance analysis method. And finally, the implementation of an existing installed propeller performance analysis method, to model the unwanted installation effects.

From the validation of existing low-fidelity isolated propeller performance analysis methods, it was concluded that out of the three existing methods (Xrotor, JavaProp and Adkins & Liebeck scheme) originally considered, Xrotor performed best. In this validation study, each method’s performance prediction was compared with existing experimental data of the same propeller. It has been demonstrated that Xrotor outperformed the other two methods in terms of accuracy, by being more consistent throughout the entire operating range. In general, these low-fidelity methods have the most difficulty accurately predicting performance in off-design conditions. For Xrotor and the Adkins & Liebeck scheme, this manifested in the fact that they both over-predict isolated thrust performance in high loading conditions and under-predict isolated thrust performance in low loading conditions. Xrotor performed better in low-loading conditions, while the Adkins & Liebeck scheme performed better in high-loading conditions. JavaProp always under-predicted isolated performance and with less accuracy than the other two. The validation study was also used to evaluate the stability of each method. It is of vital importance that any tool or scheme incorporated in an optimization routine is very stable. Otherwise, the optimization routine will either act on bad data or experience errors that end any optimization prematurely. It was discovered that JavaProp was the most stable, followed by Xrotor and then the Adkins & Liebeck scheme. The Adkins & Liebeck scheme was very unstable, especially in off-design conditions. Both stability and accuracy of each method can be explained by examining the setup. For any low-fidelity propeller performance method, the setup consists of defining propeller geometry data and two-dimensional section aerodynamic data. JavaProp has a very simple setup with very limited options for defining section aerodynamic data. The Adkins & Liebeck scheme defines no limit on the amount of data and is therefore more accurate than JavaProp, but the implementation of the scheme has difficulty dealing with off-design conditions resulting in convergence problems. Xrotor has the most complex setup because there is a lot of pre-processing required, in order to generate the input data. This means that there are a lot more variables influencing the performance prediction, making it less stable, but when everything is supplied correctly, it also makes it more accurate.

From the validation of the high-fidelity isolated propeller performance analysis method, it was concluded that the implemented CFD method is valid and able to predict isolated propeller performance with acceptable accuracy for the purpose of this research. The used APIAN propeller reference case was chosen because it has a complex geometry that is a good example of the type of advanced propellers that is aspired to for this research. The validation study revealed that overall, the CFD method is able to capture all the important flow phenomena. However, there are also some discrepancies between the reference data and the CFD results. First, a comparison of the thrust coefficient at three propeller operating points revealed that the results predicted by the CFD analysis have a higher gradient compared to the reference data. It has been demonstrated that at least part of these differences can be traced back to the spinner thrust/drag contribution. However, no definitive explanation was found, leading to the formulation of recommendation for further research in the next section. A sensitivity study was also part of the validation study and was conducted in order to investigate the CFD solver's sensitivity to mesh refinement, in its ability to predict propeller performance. The mesh was refined in the most important areas, while keeping the number of cells (and therefore the required computational time) relatively low. This resulted in a mesh refinement sensitivity study where, the differences between the CFD results at different mesh refinement levels were small. In the case of the wake velocity field analysis, it was concluded that with the setup of the CFD method, it was difficult to accurately predict the intensity of the tip vortex in the wake, at the intermediate and low propeller thrust settings. It is concluded that this is because the diameter of the tip vortex decreases for increasing advance ratio. If the flow phenomenon that is to be captured becomes smaller, and the mesh grid spacing remains constant, the analysis is effectively performed with a coarser refinement level than originally intended.

From the verification of the parametrization tool, it was concluded that it is able to accurately represent a complex geometry with the least amount of parameters. A verification study was performed in which three tests were conducted. The aim of the first test was to demonstrate that, for an arbitrary complex propeller, it is possible to accurately recreate the propeller geometry with the parametrization tool. This has been verified with two different propellers: the N250 propeller, a low-speed, unswept propeller design that was used in the validation of the low-fidelity propeller performance analysis method, and the APIAN propeller, an advanced propeller with sweep and lean, which was used in the validation of the high-fidelity propeller performance analysis method. For both propellers, the weakest part of the parametrization is the spinner geometry parametrization. In the case of the N250 spinner, the reference geometry does not contain a cylindrical hub section that is tangent with the dome section, while this is assumed in the parametric model of the spinner. In the case of the APIAN spinner the dome length is quite large, compared to the overall propeller dimensions. It then becomes difficult to fit the parametric model with a small number of variable Bezier control points. Still, the differences between the reference and the parametrized spinner geometries were relatively small. The aim of the second verification test was to demonstrate that a parametrized version of the original N250 propeller will give the same low-fidelity propeller performance analysis method results as for the original propeller design. There were no significant differences found between the performance results of the parametrized and reference propeller model, which was expected because the propeller blade design was modeled accurately. Finally, the aim of the third verification test was to demonstrate that a parametrized version of the original APIAN propeller will give the same high-fidelity propeller performance analysis method results as for the original propeller design. In general, the difference between the CFD results of the parametrized and the reference propeller were very small. There was a small under-prediction in thrust coefficient where the deviation ranges from -0.0057 to -0.00035. Although no definitive cause was found, it was hypothesized that, based on the parametrization of the APIAN propeller was made. First, the APIAN propeller has a specific tip geometry where the orientation of the tip surface is at an angle with respect to the propeller center axis, whereas in the parametrization tool this is not possible and all sections are parallel. This means that the reference geometry has a small additional tip area compared to the parametrized geometry, that could generate additional thrust, enough to explain the small difference. Furthermore, it is possible that, despite best efforts in ensuring that the parametrization of the APIAN propeller was most accurate, the curvature of the blade surfaces in the tip region are slightly different compared to the reference geometry. A combination of flat, thin airfoils and small leading edge radii could potentially impair the parametrization process and result in slightly different curvatures on the blade surface. This impacts the amount of thrust generated based on that curvature.

The main goal of this thesis was to study the potential benefit on propeller performance of considering installation effects during the conceptual design of the propeller. This was achieved by setting up and performing a propeller blade optimization for minimized installation effects. During the setup, it was discovered that, in an ideal scenario, all these aspects of adverse installation effects (noise, unsteady thrust performance) are optimized with a single objective. However, this was not known in advance so three candidate objective functions were tested. The first candidate objective function aimed at minimizing unsteady fluctuations in installed propeller thrust, while the second and third candidate objective functions aimed at minimizing the

aerodynamic drivers of the noise emissions. Constraints were imposed to keep the thrust performance of the optimized propeller equal to that of the reference propeller. During the setup, a design vector sensitivity study was performed. This revealed that, the influence of the blade sweep and lean on the different objective functions was either completely insignificant or predictable. Based on these findings, it was decided not to include the blade sweep and lean in the optimization process.

From the results of the propeller blade optimization for minimized installation effects, it is concluded that the original hypothesis defined in the introduction of this thesis has been proven. First, the three candidate objective functions were applied in an optimization routine for a high-propeller loading condition. It revealed that at this thrust setting, each of the objective functions has contradictory design trends. All three showed that improvement is possible when applying the optimization routine. Since noise is a far more critical limiting factor for advanced propellers for next-generation aircraft, it was decided to optimize for installed noise performance. Between the two candidate objective functions related to noise, formulated and studied here, optimizing installed blade noise performance was selected over section noise, because it accounts for phase shifts due to blade sweep effects. Further optimizations were performed at intermediate and low thrust settings. In these cases, the relative thrust constraint had a more stricter impact than at the high thrust setting, which resulted in a smaller Pareto front. Three optimum solutions were then found at three optimizations at three different operating conditions. A sensitivity study was performed for the entire operating range, to make a final selection and verify that each new design performs better in terms of installed performance. All three performed better than the baseline APIAN propeller, however, the optimum solution found at the optimization at intermediate thrust setting proved to be the outperforming the other optimum solutions, throughout the operating range, in all installed performance metrics.

The main design changes in the final optimum propeller design, compared to the baseline propeller design were an increase in chord in the blade hub region and an increased pitch distribution gradient and a higher advance ratio. It is clear that the final optimum propeller design is created on the principle of generating more thrust with the lower half of the blade by increasing the pitch and chord, and reducing the propeller rotational velocity. The rotational velocity directly influences the installed noise metric negatively, so it makes sense that the final optimum propeller design had the lowest allowable change in rotational velocity. For a high thrust setting, the final optimum solution was a 30% installed noise metric improvement over the baseline propeller design. At the same thrust setting, the installed propeller efficiency improved by 8%, the unsteady fluctuations in installed thrust performance were reduced by 20% and the unsteady fluctuations in installed torque performance were reduced by 26%. For a low thrust setting, the final optimum solution was a 25% installed noise metric improvement over the baseline propeller design. At the same thrust setting, there was no significant improvement in installed propeller efficiency, the unsteady fluctuations in installed thrust performance were reduced by 29% and the unsteady fluctuations in installed torque performance were reduced by 36%.

7.2. Recommendations for future work

This section presents a number of recommendations for future work, based on the findings in this report and on the conclusions in the previous section.

The current implementation of the low-fidelity propeller performance analysis method with Xrotor does not take sweep and lean effects into account. For future work, it might be interesting to investigate the vortex formulation in the context of lean and sweep, and how Xrotor can be set up with lean and sweep. The comparison between the low- and high-fidelity propeller performance analysis methods, revealed that the blade thrust distribution is influenced by sweep, which has an affect on the installed performance as well.

The validation of the high-fidelity propeller performance analysis method, using CFD revealed one discrepancy between the experimental reference data and the CFD analysis, which requires additional research. This discrepancy is related to an increase in deviation in thrust coefficient between the CFD results and the APIAN results with increasing advance ratio. For future work, it might be interesting to investigate further what causes this increase in deviation. Areas of research could include a detailed review of how the original APIAN data was obtained, and whether there are differences between the CAD model used for the CFD analysis and the wind tunnel model used to collect the experimental data. Also, the same validation case could be tested with a different CFD solver, to see if this makes any difference. Additionally, during the setup and validation of the CFD analysis, the mesh refinement was continuously pushed to increasingly higher refinement levels due to an ever increasing need for more accuracy. In general, the different refinement levels presented in the validation study do not produce significantly different CFD results. For future work, it might be interesting to perform one additional CFD analysis with a significantly coarser refinement level than applied here. This could provide more insight into the current mesh convergence results, which do not show much variation between the different meshes considered.

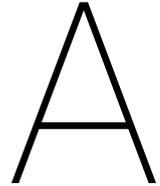
The propeller parametrization tool presented in this thesis is able to accurately represent a complex geometry with the least amount of parameters. However, for future work, there is one specific area in which the functionality of parametrization could be improved. In the current setup of the parametric tool, an open-source CAD tool, FreeCAD is used for generating all the CAD geometry related objects. However, during the setup of the parametric tool, it was found that this CAD tool has difficulty generating complex propeller geometries. Specifically in the required operation to create a loft, which is made up from the two-dimensional sections. It will perform the lofting operation, however, when the blade geometry is inspected later, it occurs often that there will be gaps between the suction and pressure blade faces. For future work, it might be interesting to investigate whether it would be beneficial to implement the parametric tool in a ParaPy environment. This environment has the same functionality as FreeCAD.

One final recommendation for future work is that a high-fidelity isolated performance analysis is performed for the found optimum solution. Unfortunately, there was no time available to perform it in this thesis. This will give a more precise indication of the results achieved with the low-fidelity isolated performance analysis method.

Bibliography

- [1] L.L.M Veldhuis. Aircraft Aerodynamics: Propeller Propulsion Aerodynamics, A (too) Short Overview, 2014.
- [2] D.R. Reddy. Seventy Years of Aero-propulsion Research at NASA Glenn Research Center. *Journal of Aerospace Engineering*, 26, Issue 2, 2013.
- [3] A. Zachariadis, C. Hall, and A.B. Parry. Contrarotating Open Rotor Operation for Improved Aerodynamics and Noise at Takeoff. *Journal of Turbomachinery*, 2013.
- [4] P. Butterworth-Hayes. Open Rotor Research Revs Up. *Aerospace America*, 2010.
- [5] T. Sinnige, D Ragni, A.M.N. Malgoezar, G. Eitelberg, and L.L.M. Veldhuis. APIAN-INF: An Aerodynamic and Aeroacoustic Investigation of Pylon-Interaction Effects for Pusher Propellers. *CEAS Aeronautical Journal*, 2017.
- [6] M. Dreila. Xrotor: An interactive program for the design and analysis of ducted and free-tip propellers and windmills. <http://web.mit.edu/dreila/Public/web/xrotor/>. Accessed December, 2015.
- [7] M. Hepperle. JavaProp: A simple tool for the design and the analysis of propellers and wind turbines. <http://www.mh-aerotoools.de/airfoils/javaprop.htm>. Accessed December, 2015.
- [8] J. Morgado. JBLADE: An open-source propeller design and analysis code written in the Qt programming language. <https://sites.google.com/site/joaomorgado23/Home>. Accessed December, 2015.
- [9] B. Epps. OpenProp: Free software for the design and analysis of marine propellers and horizontal-axis turbines. <http://engineering.dartmouth.edu/epps/openprop/>. Accessed December, 2015.
- [10] A. Falzone. PROPDESIGN: An open source, public domain, aircraft propeller design software. <http://propdesign.jimdo.com/>. Accessed December, 2015.
- [11] C.N. Adkins and R.H. Liebeck. Design of Optimum Propellers. *Journal of Propulsion and Power*, 1994.
- [12] M. Dreila. Xfoil: An interactive program for the design and analysis of subsonic isolated airfoils. <http://web.mit.edu/dreila/Public/web/xfoil/>. Accessed Februari, 2016.
- [13] T Sinnige and L.M. Veldhuis. Pylon Trailing Edge Blowing Effects on the Performance and Noise Production of a Pusher Propeller. *AIAA*, 2013.
- [14] E.E. Larrabee. Practical Design of Minimum Induced Loss Propellers. *SAE Technical Paper*, 1979.
- [15] S. Goldstein. On the Theory of Screw Propellers. *Journal of Zosen Kiokai*, 1958, 1958.
- [16] Martin Hepperle. JavaFoil: A simple tool for the design and the analysis of airfoil sections. <http://www.mh-aerotoools.de/airfoils/javafoil.htm>. Accessed December, 2015.
- [17] N. van Arnhem and Veldhuis L.L.M. Design and Analysis of an Installed Pusher Propeller with Boundary Layer Inflow. *Delft University of Technology Master Thesis*, 2015.
- [18] R.F. Jansen and Veldhuis L.L.M. The Influence of Laminar-Turbulent Transition on the Performance of a Propeller. *Delft University of Technology Master Thesis*, 2015.
- [19] T.D. Economon, F. Palacios, S.R. Copeland, T.W. Lukaczyk, and J.J. Alonso. SU2: An Open-Source Suite for Multiphysics Simulation and Design. *AIAA*, 54, Issue 3, 2016.
- [20] T.C.A. Stokkermans and Veldhuis L.L.M. Design and Analysis of Swirl Recovery Vanes for an Isolated and a Wing Mounted Tractor Propeller. *Delft University of Technology Master Thesis*, 2015.
- [21] NUMECA international. Autogrid5: A full automatic hexahedral grid generator for all types of rotating machinery. <http://www.numeca.com/product/autogrid5>. Accessed August, 2017.

- [22] Ansys. Fluent: The most powerful computational fluid dynamics (CFD) software tool available. <http://www.ansys.com/Products/Fluids/ANSYS-Fluent>. Accessed Februari, 2016.
- [23] P. Crozier. APIAN Installed Tests in the ONERA SIMA Wind Tunnel. *39th AIAA Aerospace Sciences Meeting & Exhibit*, 2001.
- [24] Dassault Systemes. CATIA: The world's engineering and design leading software for product 3D CAD design excellence. <https://academy.3ds.com/en/software/catia-v5-student-edition>. Accessed August, 2017.
- [25] A.L. Keith, G.J. Bingham, and A.J. Rubin. Effects of Propeller-Shank Geometry and Propeller-Spinner-Juncture Configuration on Characteristics of an NACA 1-Series Cowling-Spinner Combination with an Eight-Blade Dual-Rotation Propeller. *NACA Research Memorandum L51F26*, 1951.
- [26] J.D. Anderson. *Fundamentals of Aerodynamics*. Mcgraw-Hill Education - Europe, fifth edition, 2011.
- [27] R.C.W. de Koning. Development of a Parametric 3D Turbomachinery Blade Modeler. *Delft University of Technology Master Thesis*, 2015.
- [28] FreeCAD. FreeCAD: A Parametrix 3D Modeler Mad Primerily to Design Real-Lift Objects of Any Size. <https://www.freecadweb.org/>. Accessed August, 2017.
- [29] A. Betz. Screw propellers with minimum energy loss. 1919.
- [30] T. Theodorsen. *Theory of propellers*. McGraw-Hill Book Company, 1948.
- [31] O. Gur and A. Rosen. Multidisciplinary Design Optimization of a Quiet Propeller. *14th AIAA/CEAS Aeroacoustics Conference (29th AIAA Aeroacoustics Conference)*, 2008.
- [32] J. Fazil and V. Jayakumar. Investigation of Airfoil Profile Design Using Reverse Engineering Bezier Curve. *ARPN Journal of Engineering and Applied Sciences*, 6, Issue 6, 2011.
- [33] H. Schlichting. Uber das ebene Windschattenproblem. *Archive of Applied Mechanics*, 1930.
- [34] D.B. Hanson. Helicoidal Surface Theory for Harmonic Noise of Propellers in the Far-field. *AIAA Journal*, 18(10), 1980.
- [35] Scipy Community. Scipy: Optimization and Root Finding. <https://docs.scipy.org/doc/scipy/reference/optimize.html>. Accessed August, 2017.
- [36] H. Schlichting. A constrained, globalized, and bounded Nelder–Mead method for engineering optimization. *Structural and Multidisciplinary Optimization*, 27, Issue 1-2, 2004.



CFD validation results

A.1. Overall performance related results

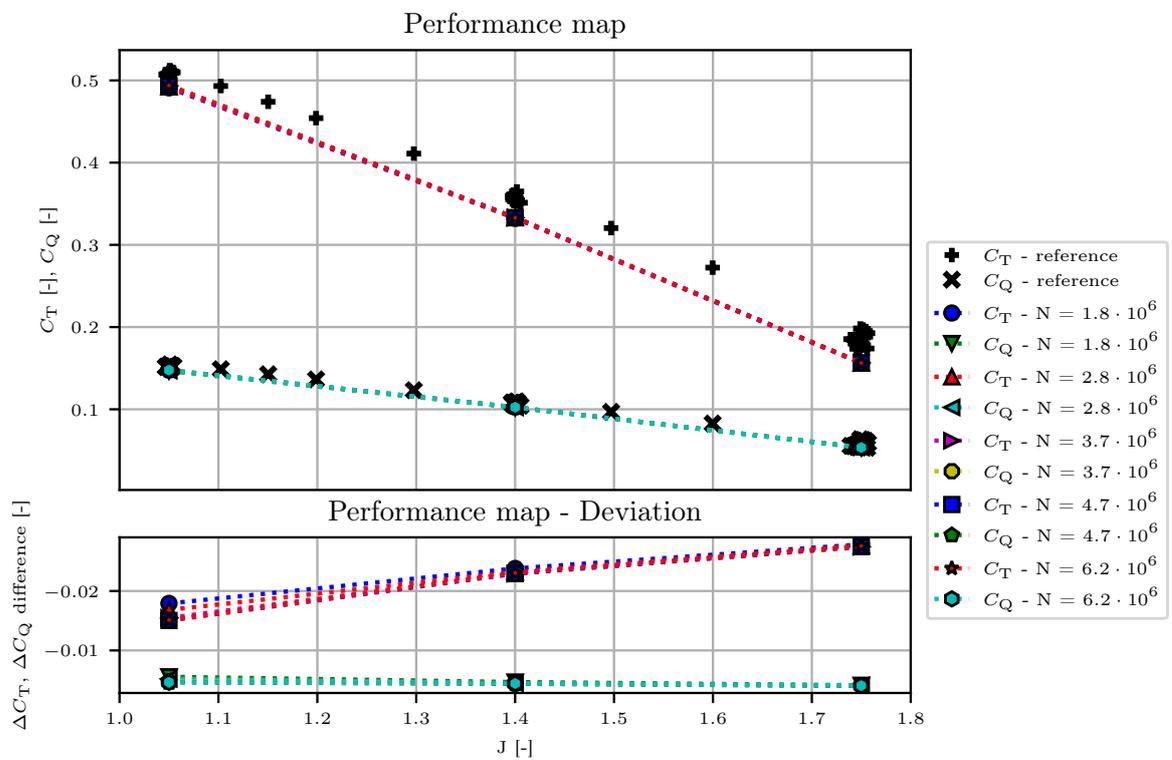


Figure A.1: Isolated propeller blades contribution performance map for sensitivity study. Upper plot shows the total thrust and torque coefficient for both the reference case and the different mesh refinements. Lower plot shows the difference between the reference and the different mesh refinements.

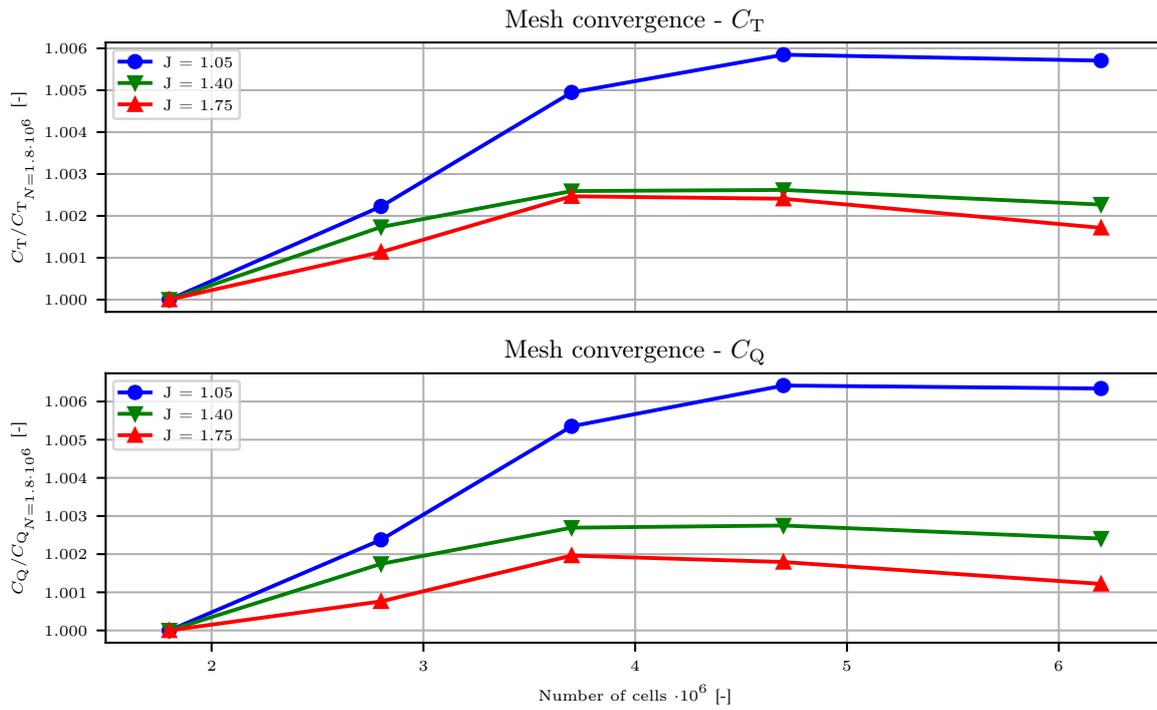


Figure A.2: Isolated propeller blades performance mesh convergence for sensitivity study. It shows in detail how the main propeller performance characteristics vary with mesh refinement.

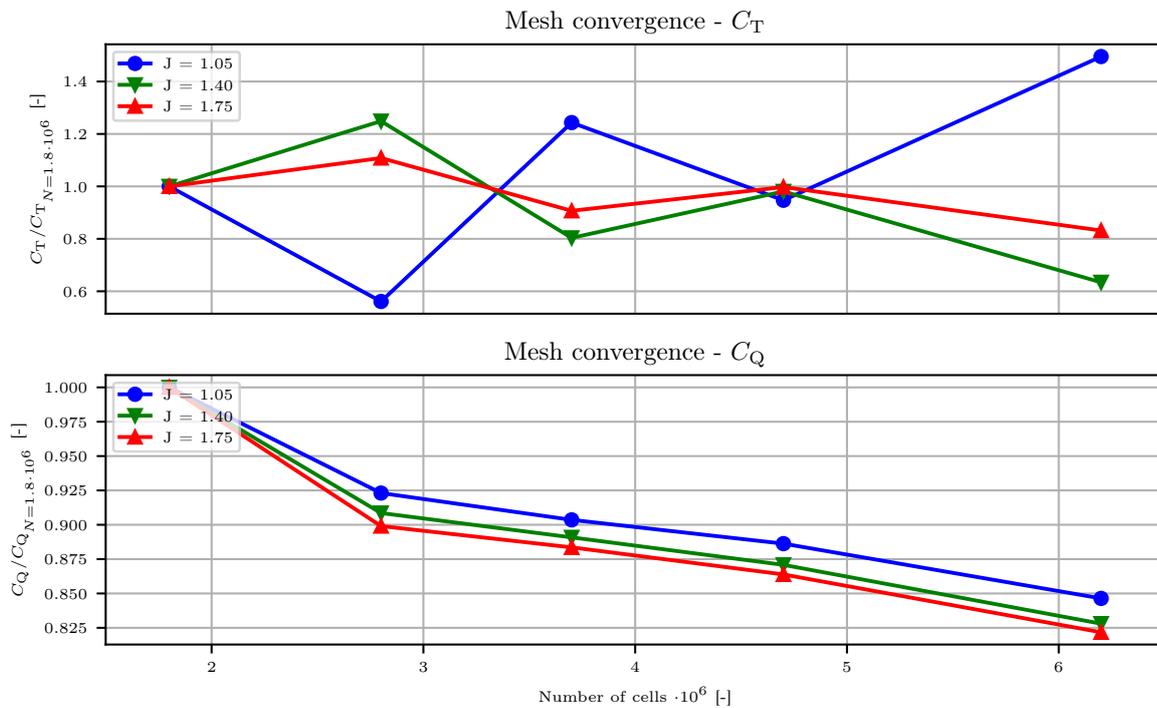


Figure A.3: Isolated spinner performance mesh convergence for sensitivity study. It shows in detail how the main propeller performance characteristics vary with mesh refinement.

A.2. Blade pressure distribution related results

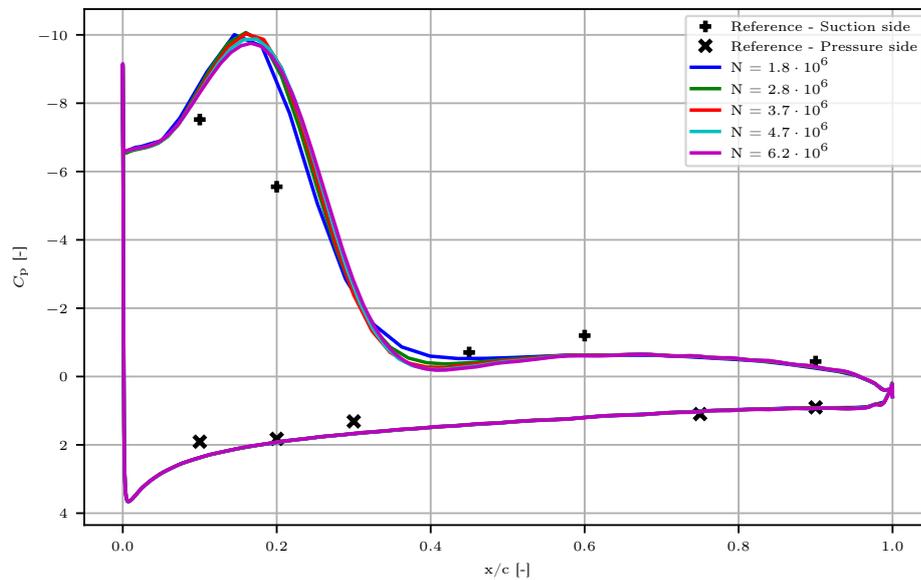


Figure A.4: $J = 1.05 - r/R = 0.75$: At $r/R = 0.75$ all flow phenomena that was witnessed at 65% are clearly visible as well. The leading edge vortex peak has moved slightly aft and the overall pressure is reduced. The pressure side is still predicted very well, however on the suction side, the width of the leading edge vortex peak region is wider for the CFD, compared to the reference. The sensitivity of mesh refinement is most noticeable in the dip behind the vortex peak, which shows similar behavior as for 65%, but the pressure in this region is in general under-predicted compared to the reference.

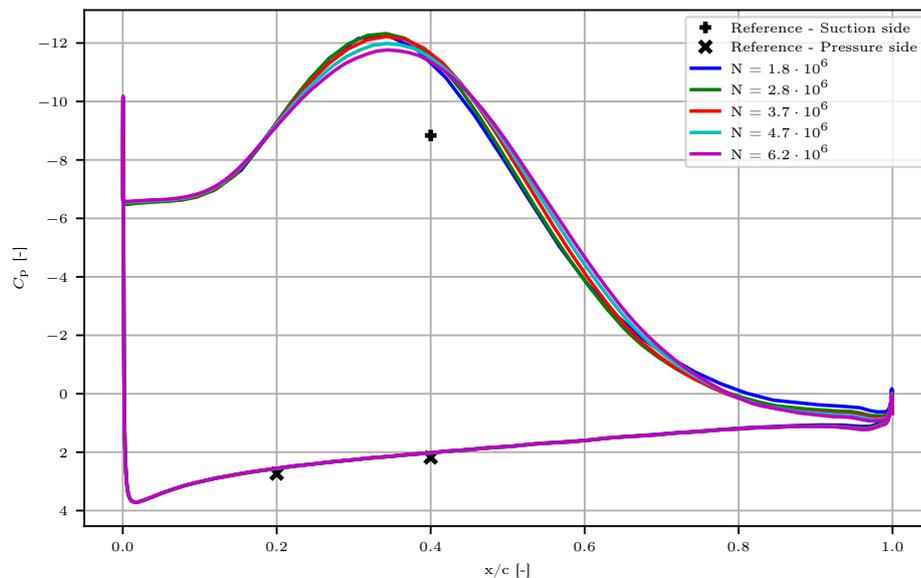


Figure A.5: $J = 1.05 - r/R = 0.85$: For $r/R = 0.85$ limited reference data points are available. For the points that are available it can be concluded that the pressure side is predicted very well and once again is not sensitivity to mesh refinement. The vortex peak has now moved even further aft. The only reference point is well of any of the CFD mesh refinements, however due to the limited reference points available, no conclusion can be made as to whether the CFD is over-predicting the height of the pressure peak, or whether it is the position or the width of the vortex peak region. The suction side is very sensitive to mesh refinement. For increasing mesh refinement, the height of the pressure peak reduces and the width increases. Finally at the trailing edge the predicted pressure decreases for increasing refinement.

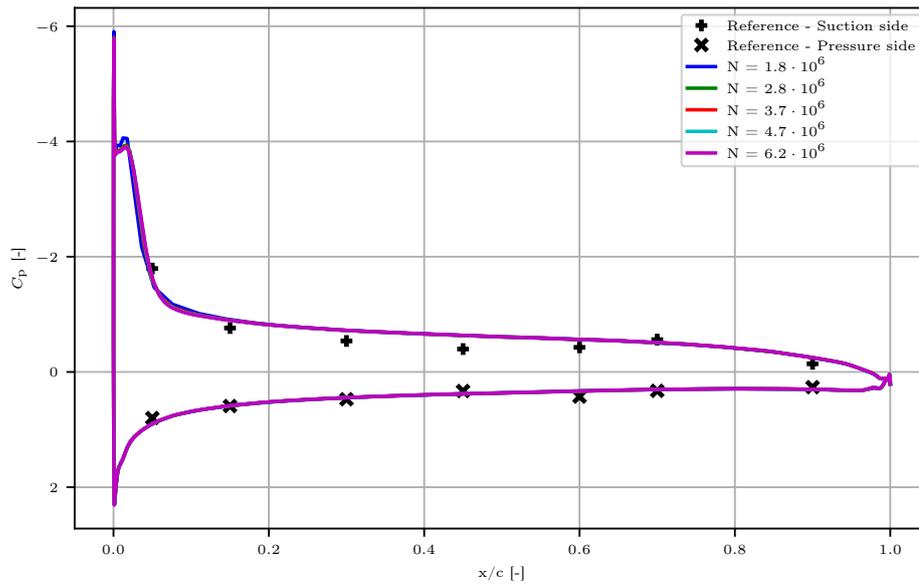


Figure A.6: $J = 1.40 - r/R = 0.65$: Compared to $J = 1.05$, the vortex peak is smaller and much closer to the leading edge. The CFD method slightly over-predicts the suction side, while it's the same story for the pressure side as before; negligible sensitivity and very well predicted. Only the extra coarse mesh shows deviating behavior at the vortex peak and subsequent just aft of the peak.

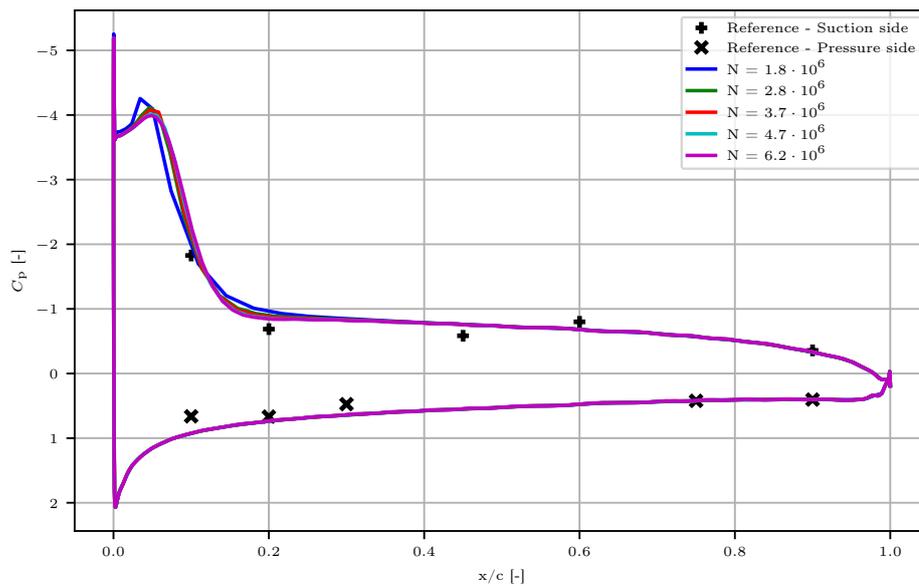


Figure A.7: $J = 1.40 - r/R = 0.75$: At 75% the vortex peak has moved slightly aft and has increased in width. The sensitivity of this region is slightly worse compared to $r/R = 0.65$, but overall prediction of the suction side is very good. The first three reference data points on the suction side suggest that in this region, the CFD method under-predicts the pressure. This is against the trend that has been shown for the previous blade stations and advance ratio.

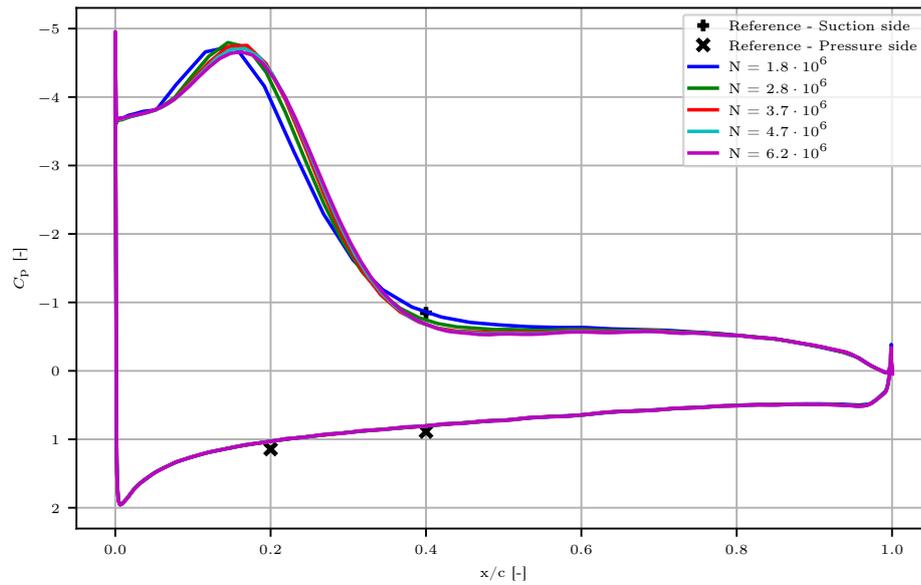


Figure A.8: $J = 1.40 - r/R = 0.85$: For 85% The vortex peak has moved slightly aft and has increased in width again. Compared to $J = 1.05$, the CFD is much closer to the single data point on the suction side. However in this case the vortex peak is in front of the data point, instead of in the same region. This leads to the general conclusion that the CFD method in general has the most difficulty with accurately predicting the location and size of this vortex pressure peak. Mesh sensitivity is comparable to 75%.

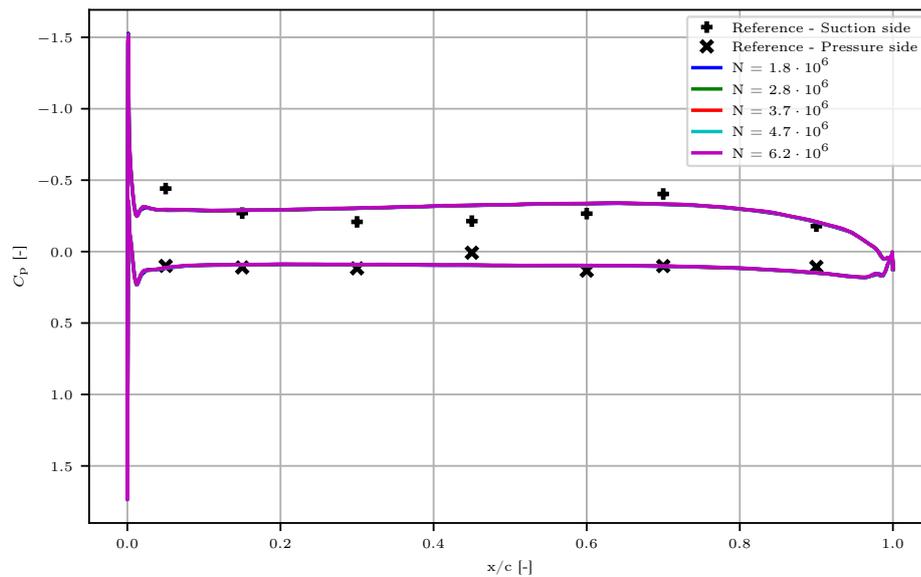


Figure A.9: $J = 1.75 - r/R = 0.65$: There is no vortex pressure peak at this blade station. The leading edge peak is followed by a small oscillation after which the pressure distribution is flat. The reference data varies more significantly, but the CFD plots appear to be the average of the varying reference data. For this case, it appears that the CFD method is not influenced by mesh refinement. It is concluded that, based on the absence of any complex flow structure, this is to be expected.

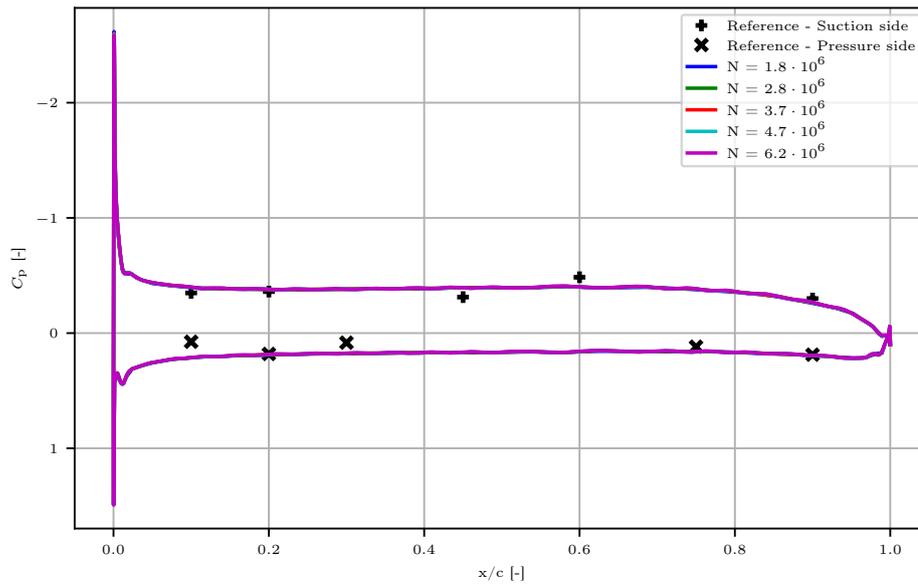


Figure A.10: $J = 1.75 - r/R = 0.75$: Very similar results as discussed at 65%. There is a faint oscillation on the suction side of the extra fine mesh. This can be caused by a not fully converged flow for this particular mesh in this particular region. Still no vortex peak visible.

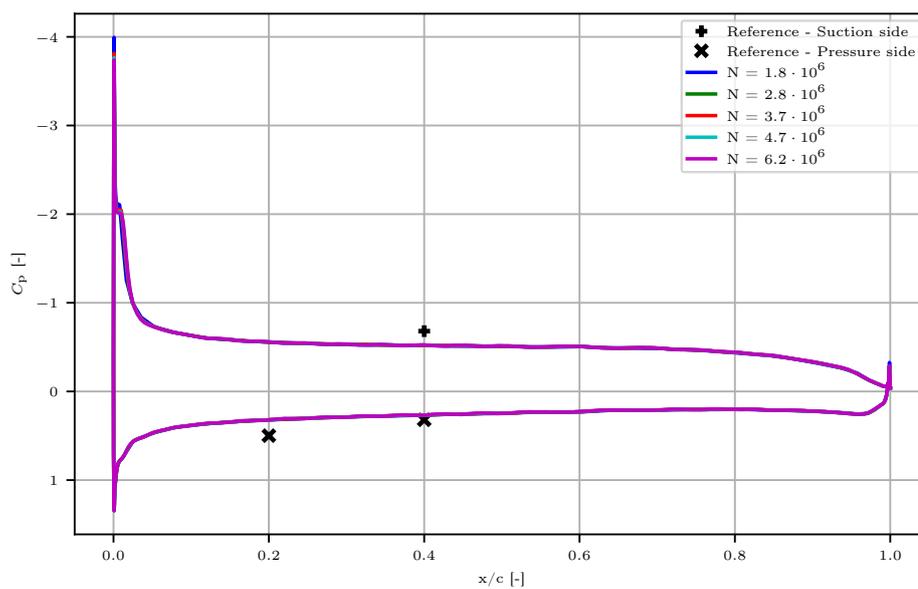


Figure A.11: $J = 1.75 - r/R = 0.85$: Finally for 85% the sensitivity is similar to other blade stations at this advance ratio. Both the pressure side and suction side are slightly under-predicted. At 85% the formation of the leading edge vortex is visible for the first time.

A.3. Wake flow field related results

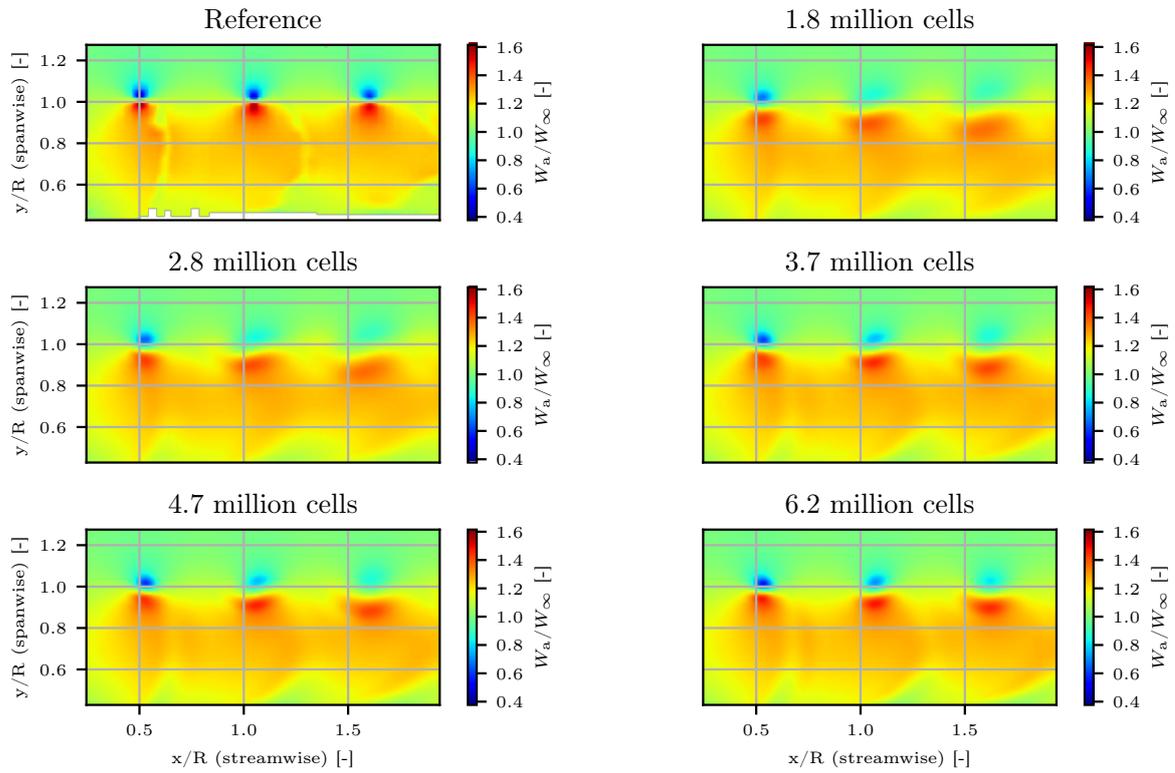


Figure A.12: 2-Dimensional wake flow field - Non-dimensional axial velocity for $J = 1.40$

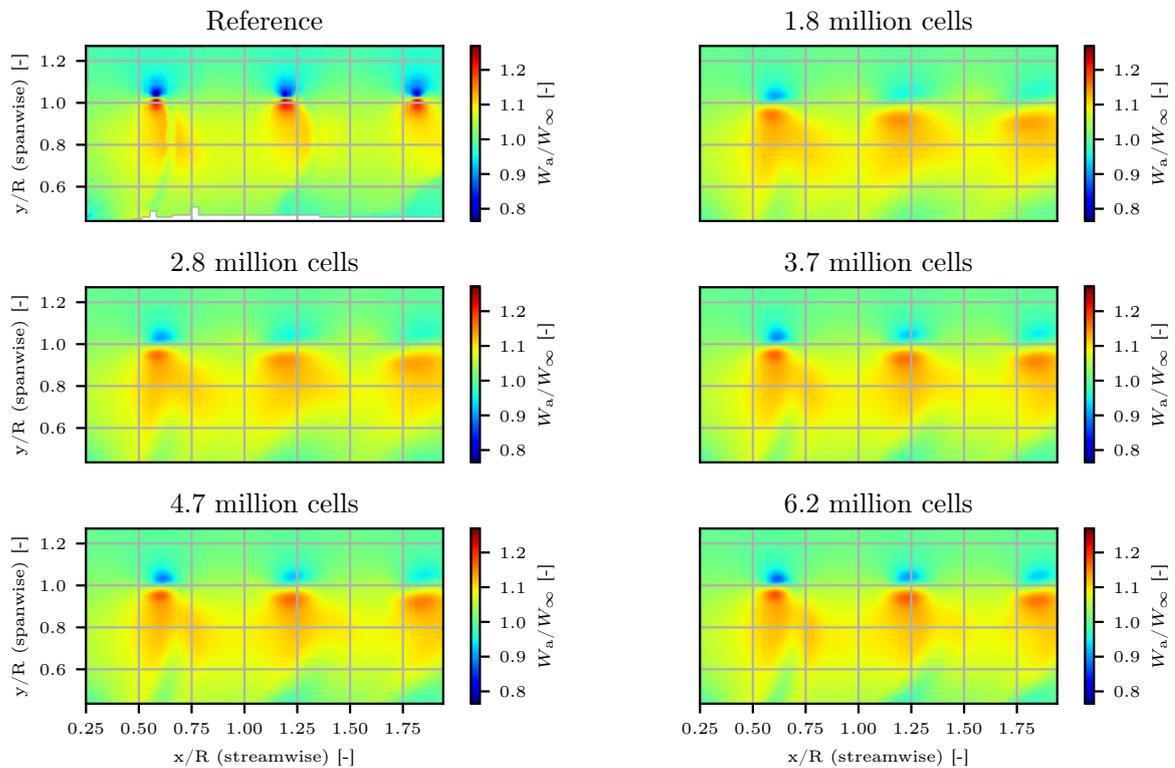


Figure A.13: 2-Dimensional wake flow field - Non-dimensional axial velocity for $J = 1.75$

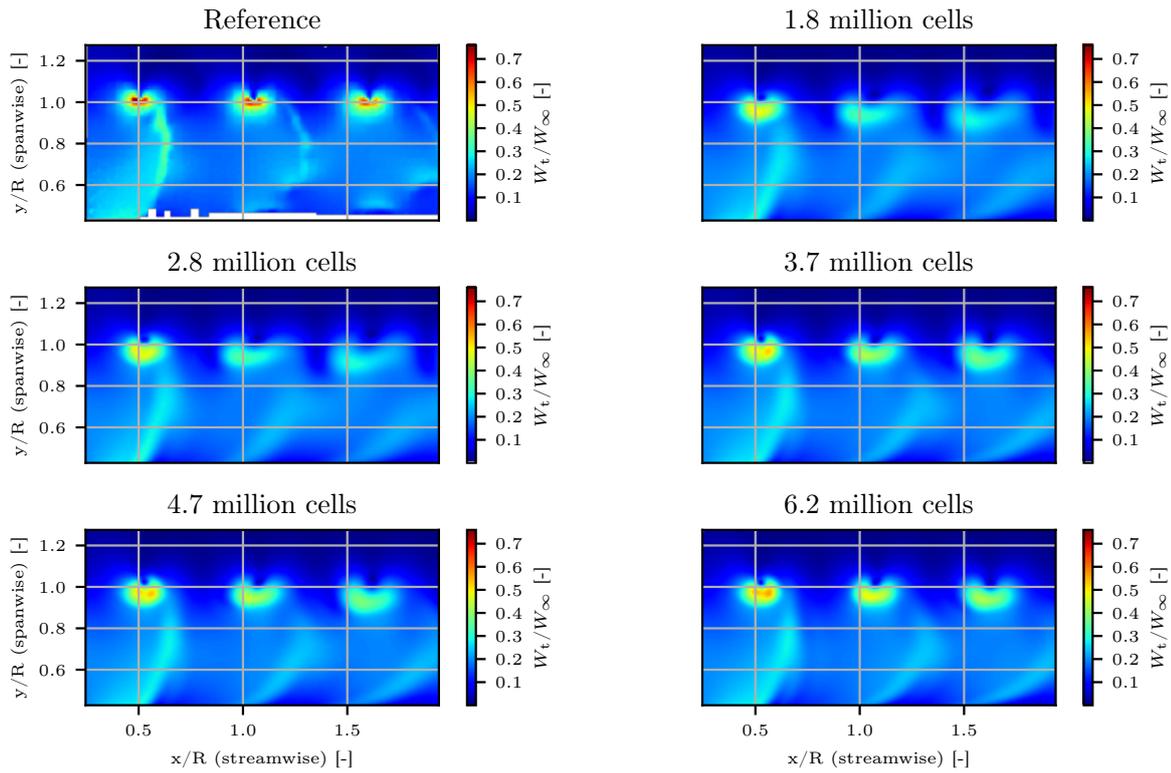


Figure A.14: 2-Dimensional wake flow field - Non-dimensional tangential velocity for $J = 1.40$

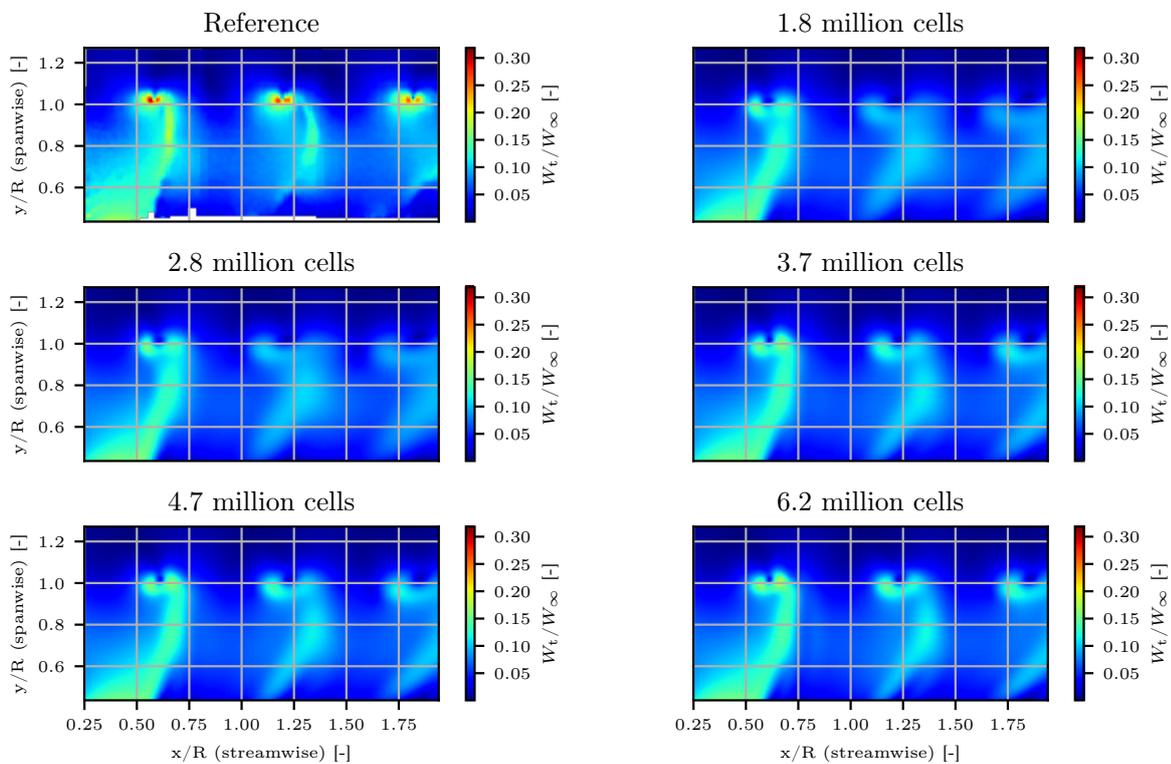


Figure A.15: 2-Dimensional wake flow field - Non-dimensional tangential velocity for $J = 1.75$

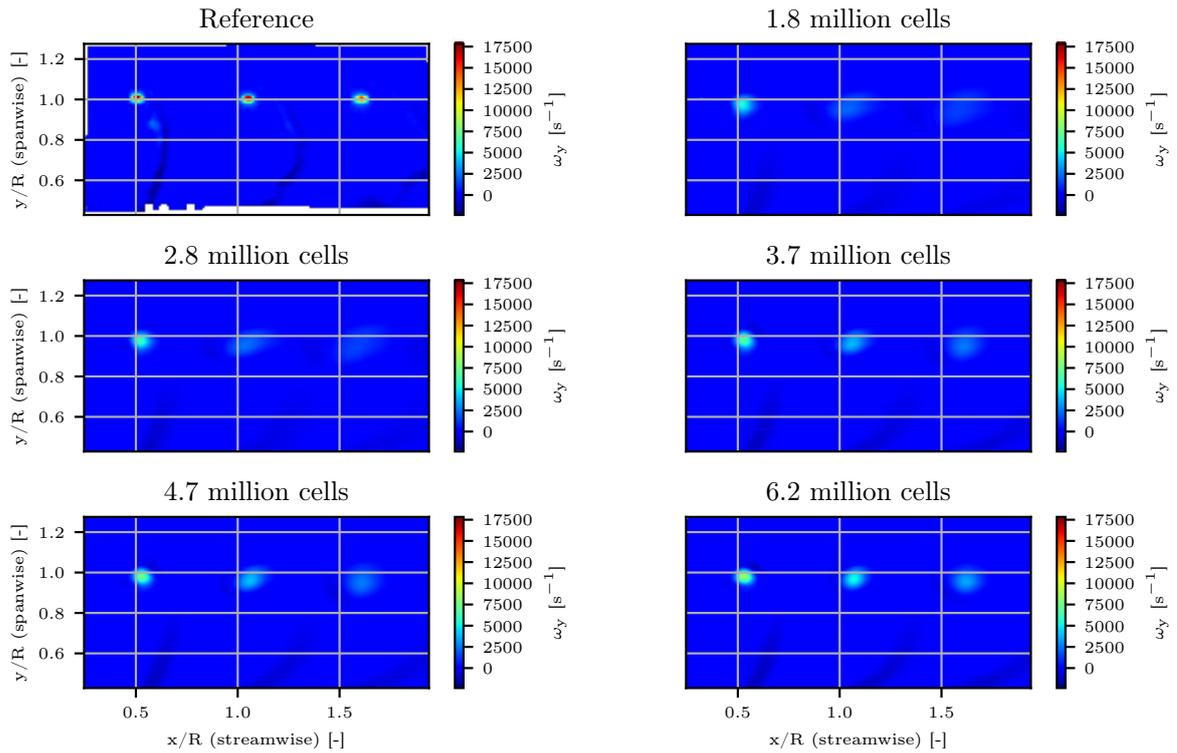


Figure A.16: 2-Dimensional wake flow field - Vorticity for $J = 1.40$

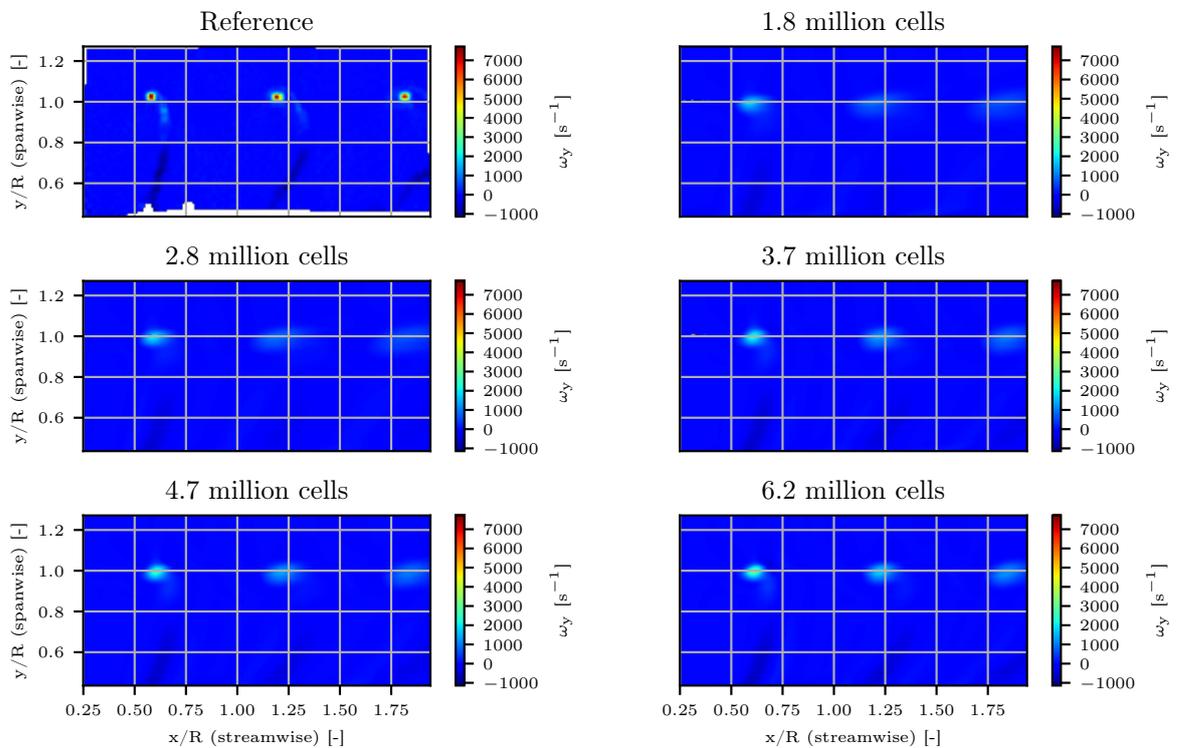


Figure A.17: 2-Dimensional wake flow field - Vorticity for $J = 1.75$

B

Parametrization tool CFD verification results

B.1. Blade pressure distribution results

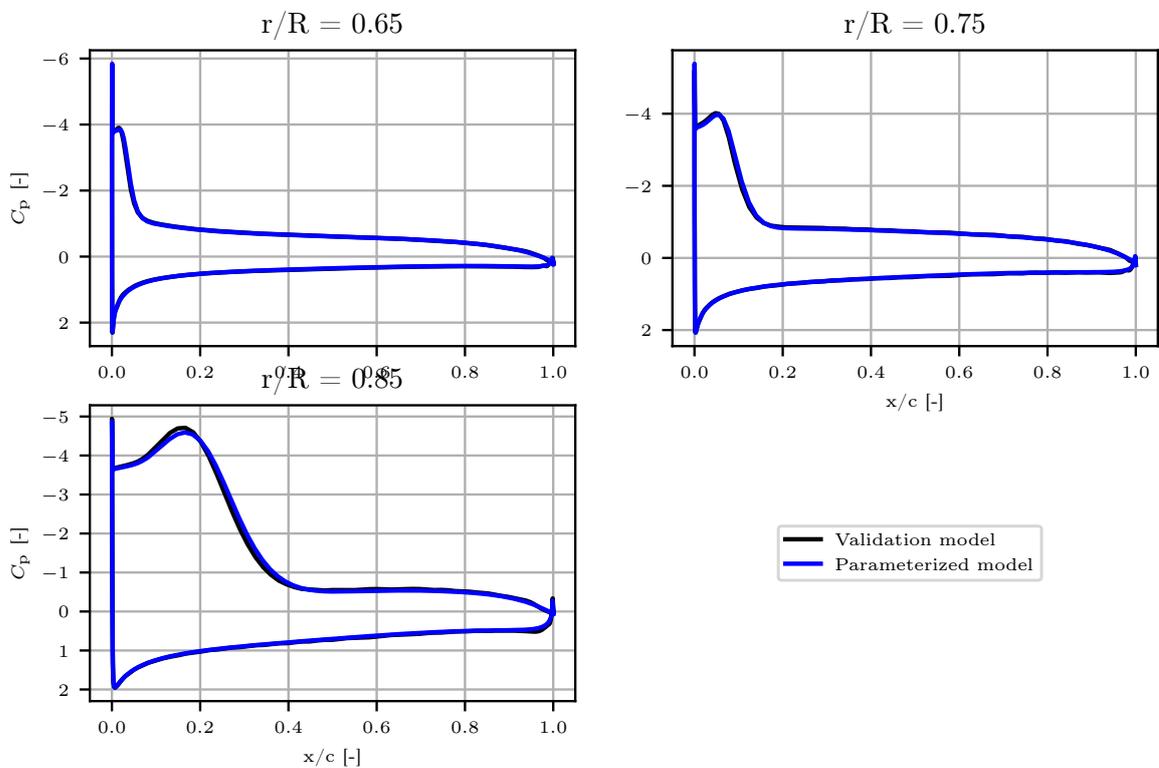


Figure B.1: Blade pressure distribution comparison between CFD validation results and CFD parametrized results for $J = 1.40$, at three different blade sections.

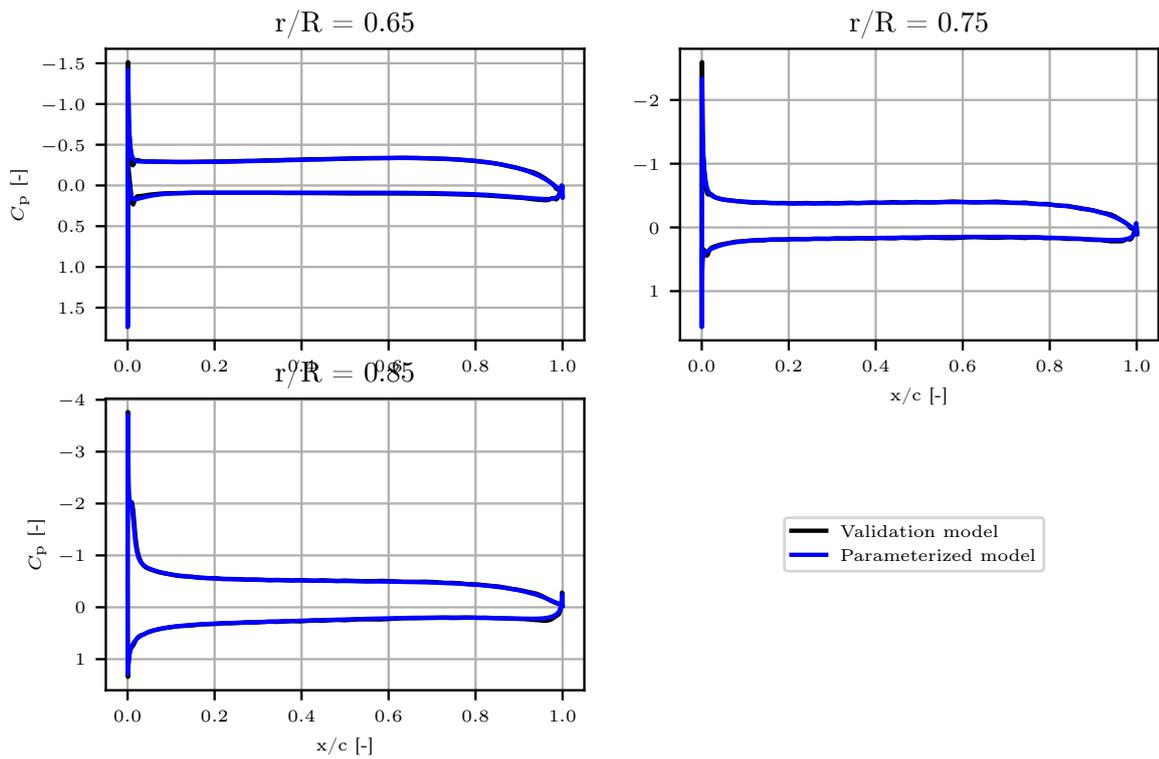


Figure B.2: Blade pressure distribution comparison between CFD validation results and CFD parametrized results for $J = 1.75$, at three different blade sections.

B.2. Spinner pressure distribution results

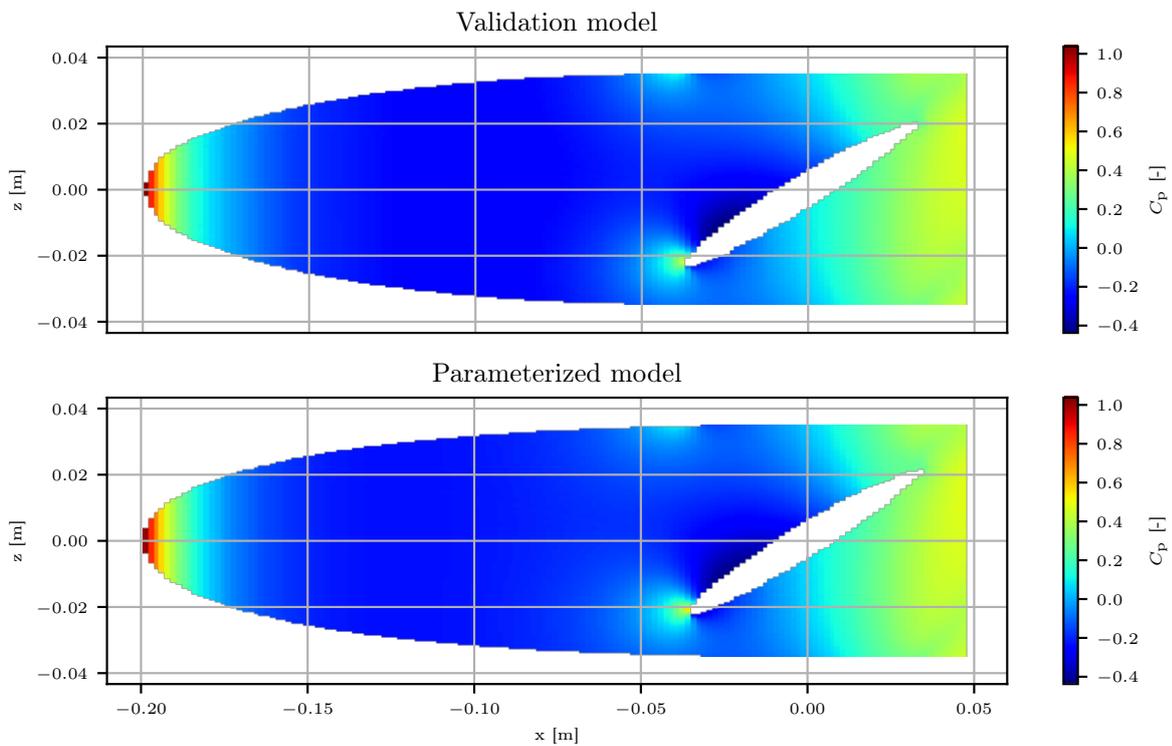


Figure B.3: Spinner pressure distribution comparison between CFD validation result and CFD parametrized result for $J = 1.40$.

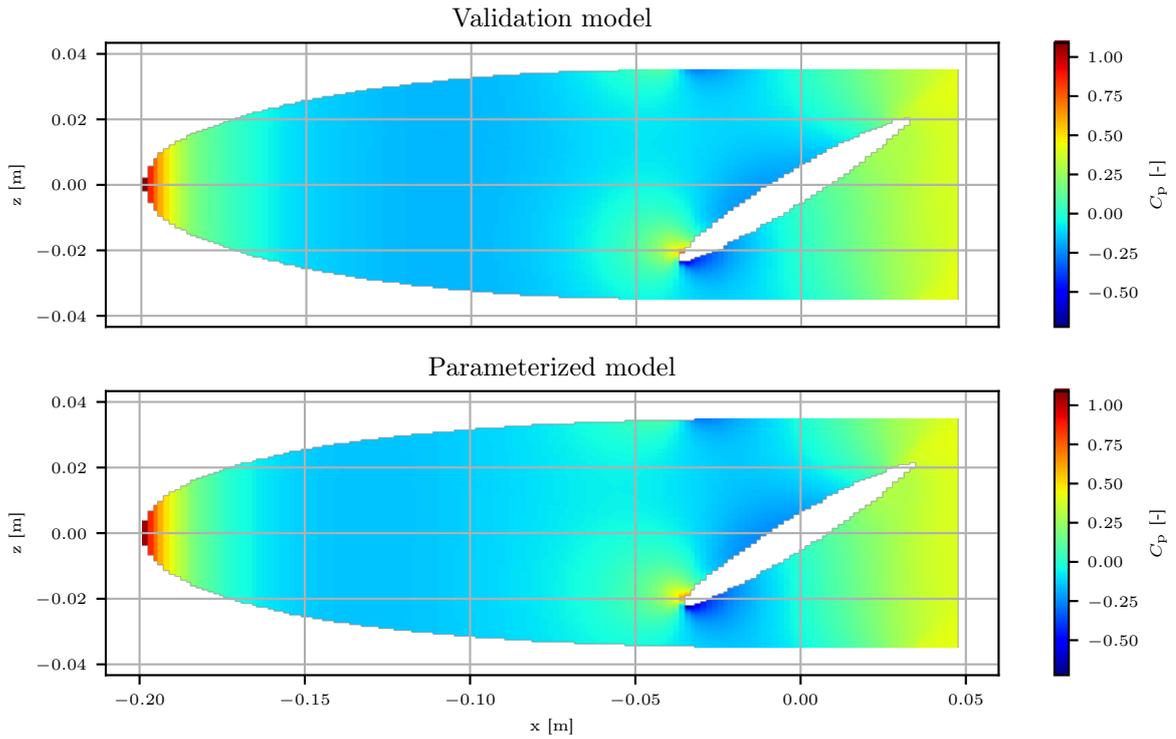


Figure B.4: Spinner pressure distribution comparison between CFD validation result and CFD parametrized result for $J = 1.75$.

B.3. Wake velocity results

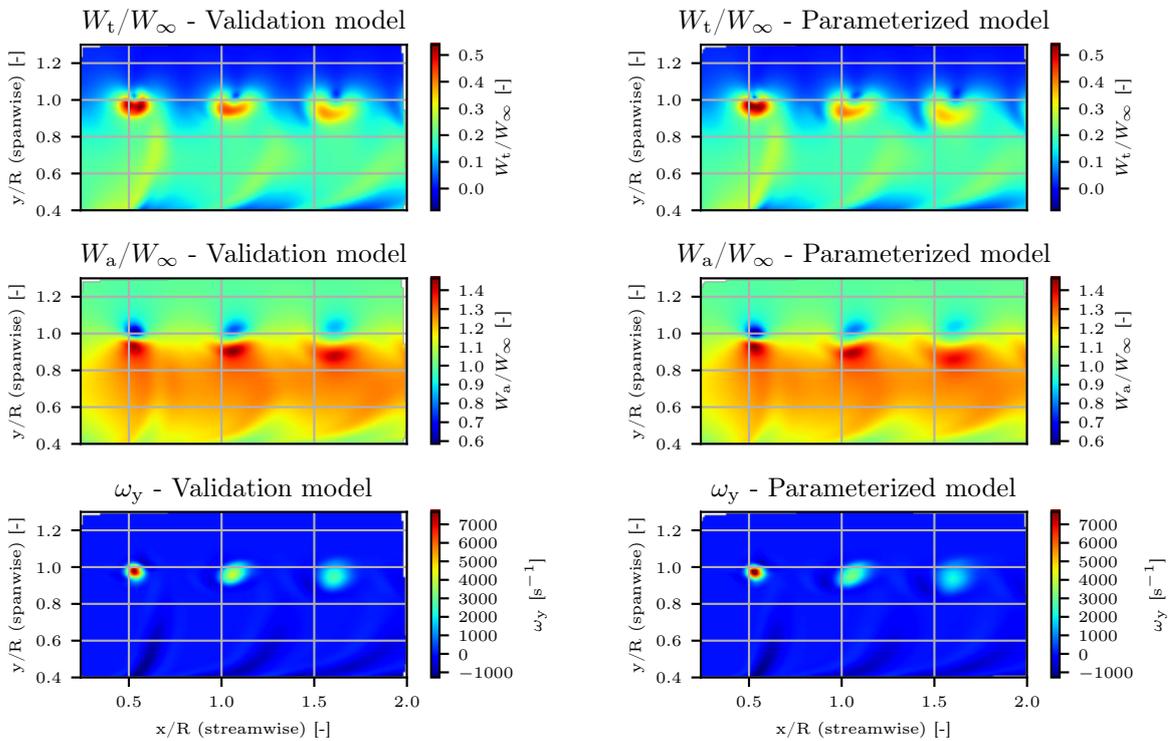


Figure B.5: Propeller wake velocity comparison between CFD validation result and CFD parametrized result for $J = 1.40$.

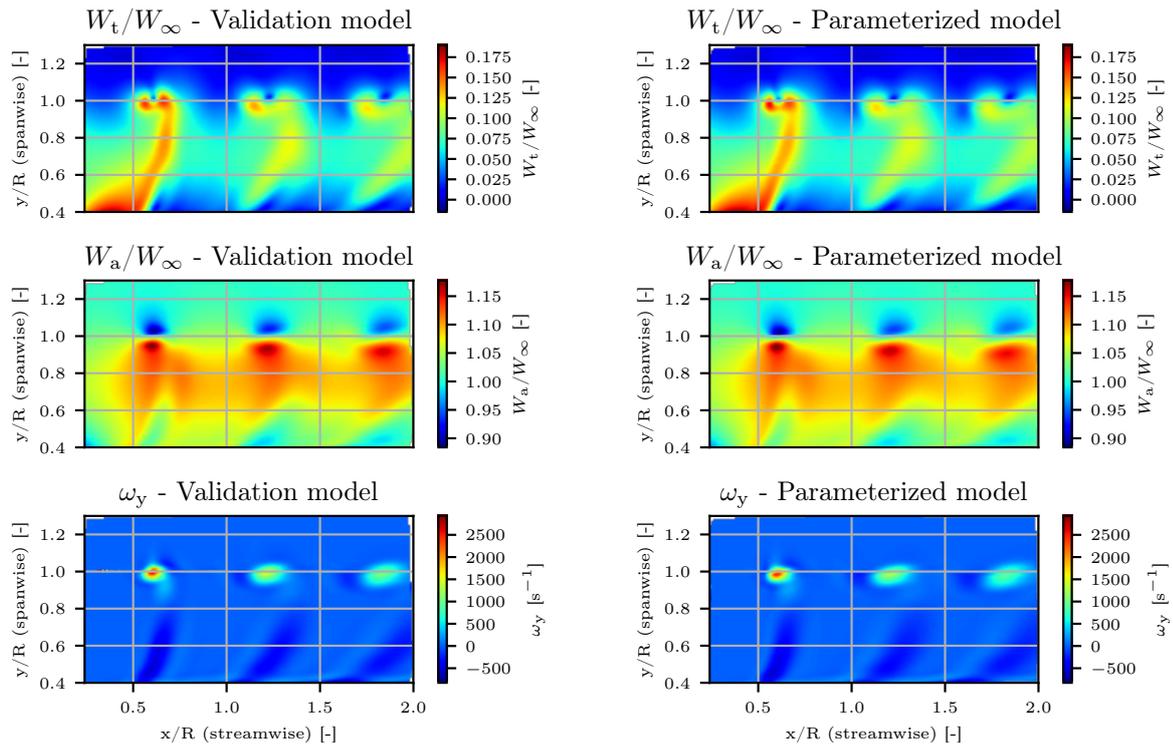


Figure B.6: Propeller wake velocity comparison between CFD validation result and CFD parameterized result for $J = 1.75$.

C

Additional different operating condition optimization results

C.0.1. Initial operating point at advance ratio equal to 1.40

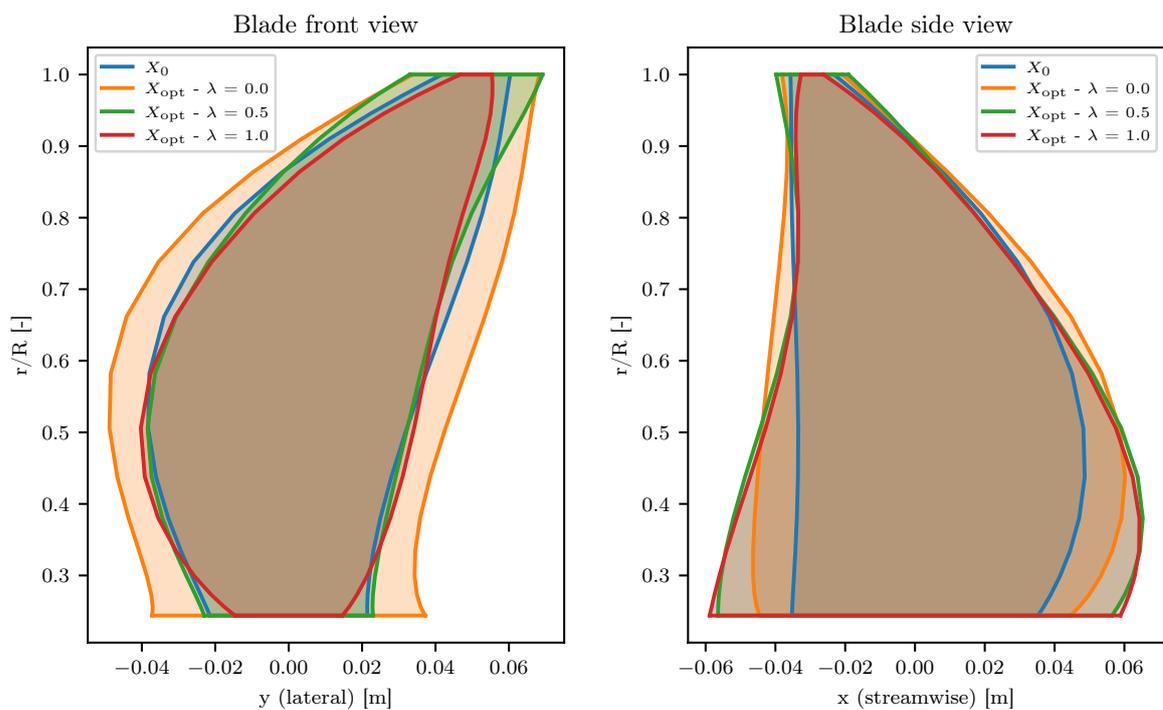


Figure C.1: Front and side view of optimized geometry for minimizing installed blade noise performance, at $J = 1.40$.

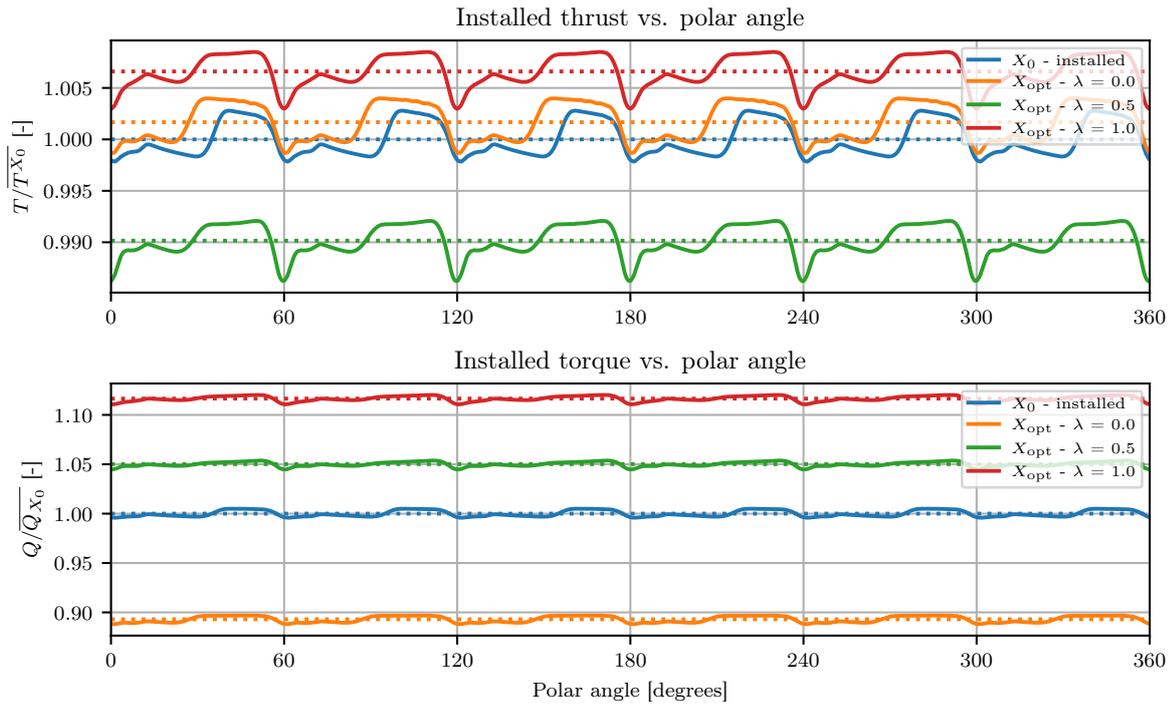


Figure C.2: Installed thrust & torque performance for minimizing blade noise performance, at $J = 1.40$.

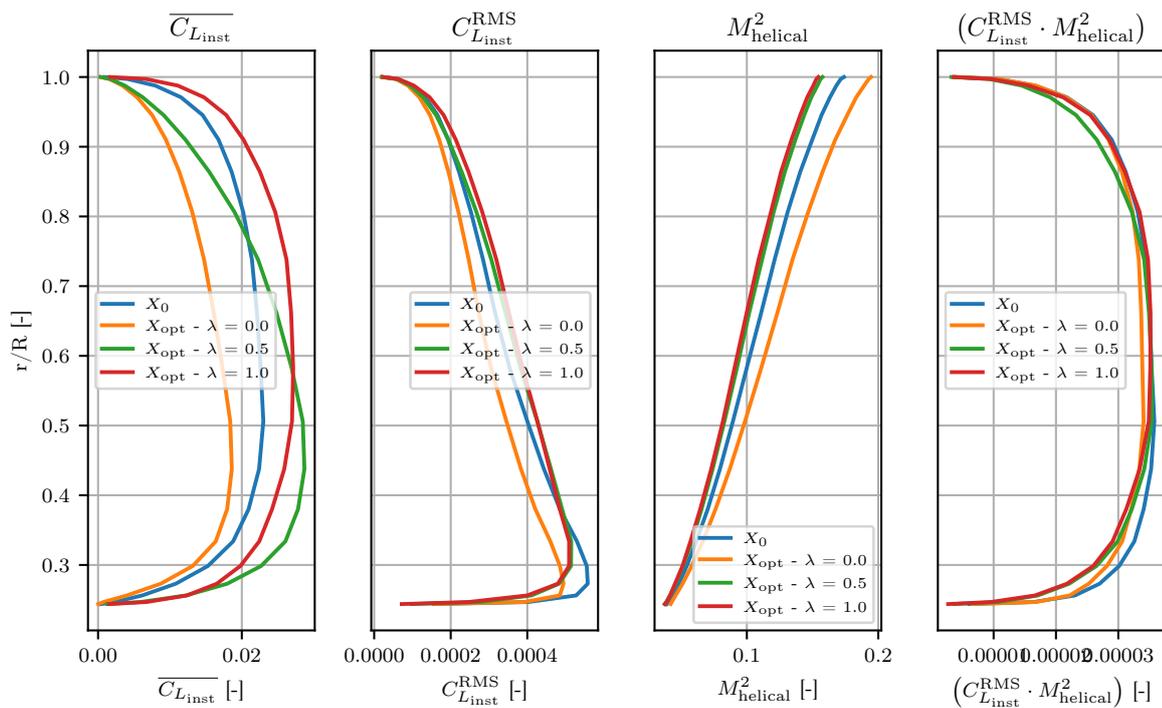


Figure C.3: Installed thrust & torque performance for minimizing installed blade noise performance, at $J = 1.40$.

C.0.2. Initial operating point at advance ratio equal to 1.75

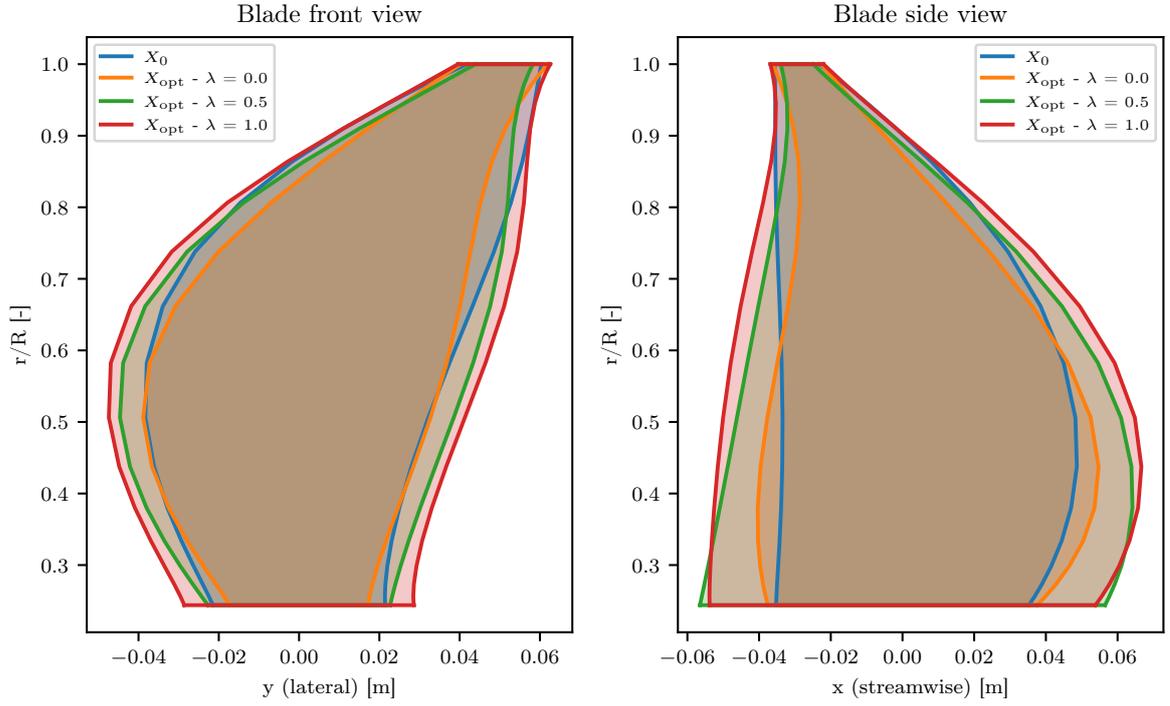


Figure C.4: Front and side view of optimized geometry for minimizing installed blade noise performance, at $J = 1.75$.

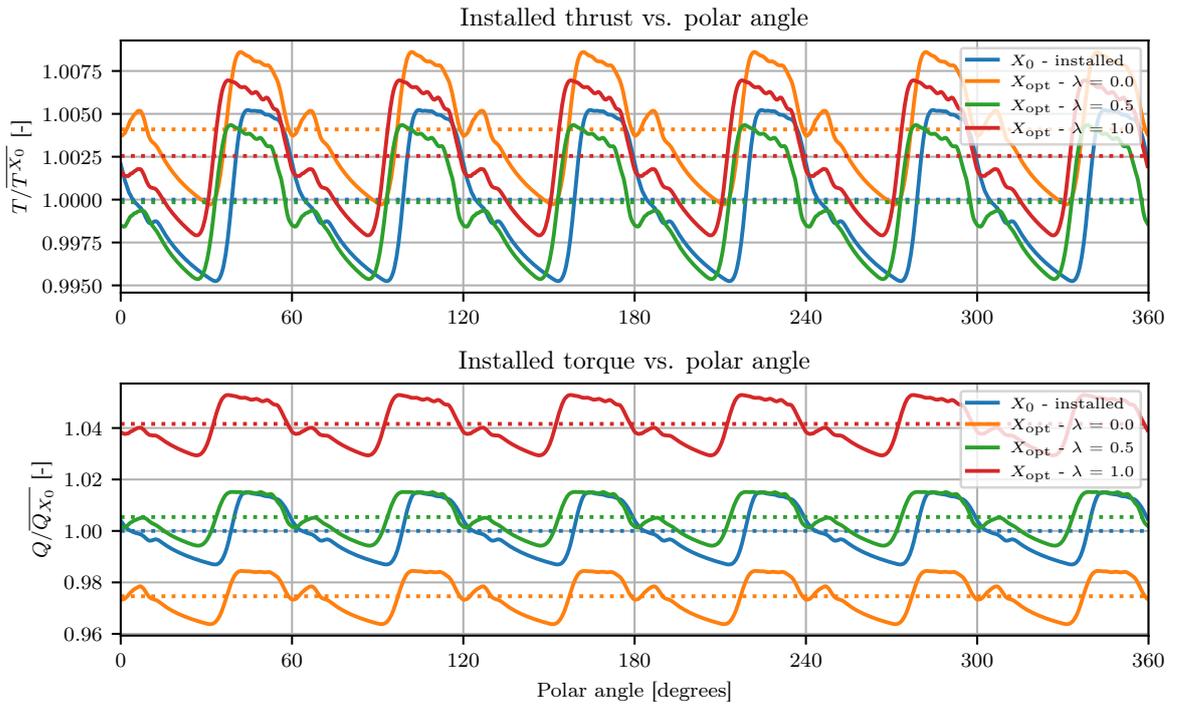


Figure C.5: Installed thrust & torque performance for minimizing blade noise performance, at $J = 1.75$.

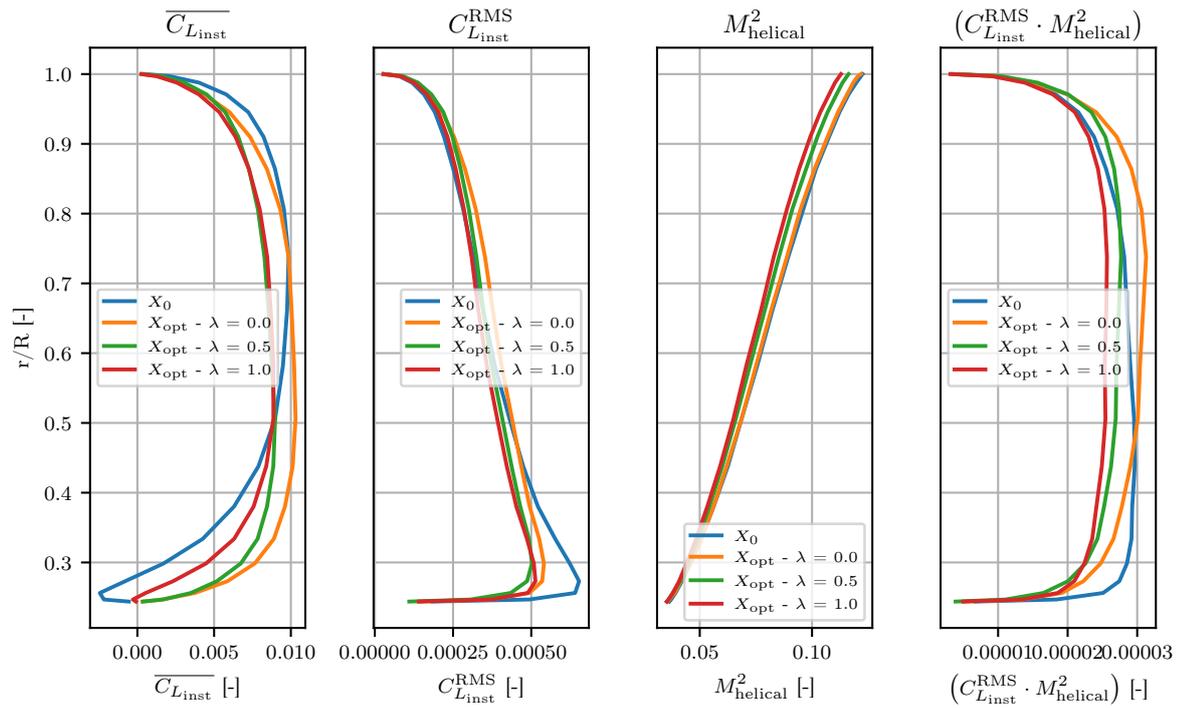


Figure C.6: Installed thrust & torque performance for minimizing installed blade noise performance, at $J = 1.75$.