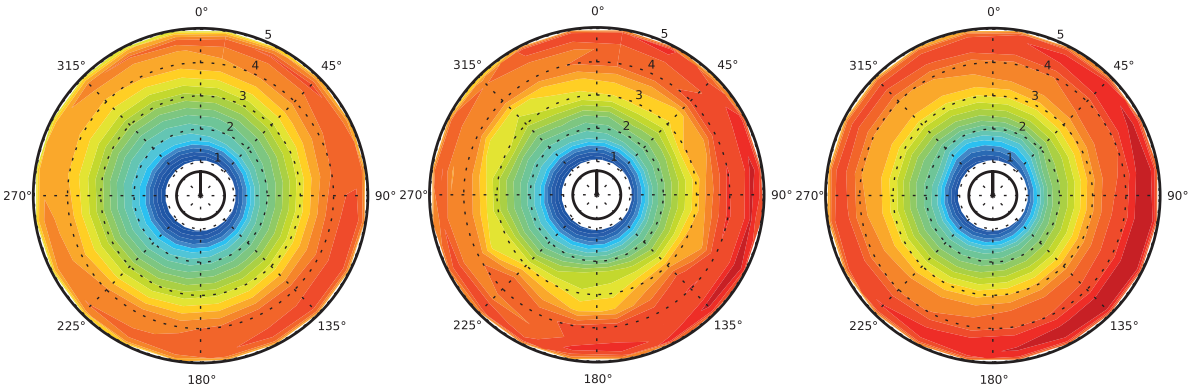
TÉCNICO LISBOA



Aeroacoustic Optimization of Wind Turbine Blades

Simão Santos Rodrigues

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Examination Committee

Chairperson:	Professor Fernando José Parracho Lau
Supervisor:	Doctor André Calado Marta
Members of the Committee:	Professor João Manuel Gonçalves de Sousa Oliveira

November 2012

"You have brains in your head. You have feet in your shoes.
You can steer yourself
any direction you choose.
You're on your own. And you know what you know.
And YOU are the one who'll decide where to go..."
Dr. Seuss, *Oh, the Places You'll Go!*

Acknowledgments

I would like to thank my thesis supervisor Dr. André Calado Marta for all his teachings and guidance throughout the development of this thesis. I would also like to leave a thank you to my family for supporting me financially and emotionally through my studies and to my friends and colleagues, with whom I shared this journey of my life.

Resumo

A produção eólica tem vindo a aumentar, com cada vez mais áreas a serem usadas como quintas eólicas e as turbinas cada vez maiores. Este desenvolvimento levou ao aumento da percepção dos efeitos das turbinas eólicas no ambiente e na saúde pública. Muita pesquisa tem sido feita no sentido de prever e reduzir o ruído por elas produzido. Neste trabalho, um modelo baseado na teoria dos elementos de pás-momento linear é utilizado para prever a performance aerodinâmica de uma turbina eólica e é acoplado a um modelo aeroacústico empírico, baseado nos trabalhos de Brooks et al (1989) e Amiet (1975), que utiliza código XFOIL na computação dos parâmetros da camada limite. O código foi validado com dados experimentais das turbinas NREL Phase II e AOC 15/50 e posteriormente utilizado no módulo de optimização pyOpt, com o algoritmo genético NSGA-II. A geometria da pá foi parametrizada utilizando curvas NURBS para os perfis 2D e curvas de Bezier para a distribuição da corda e torção ao longo da pá. Foram efectuadas várias optimizações nas duas turbinas, tanto uni- como multi-objectivo, com um numero de variáveis de design que chegou aos 62. As soluções óptimas são indicadas nas frentes de Pareto obtidas e as suas geometrias discutidas em detalhe. Estas soluções variam desde um aumento de 139.4 % da energia produzida anualmente até uma redução dos níveis de ruído em 10.7 %. Foi demonstrado que uma redução significativa no nível de ruído pode ser obtida, à custa de uma ligeira penalidade aerodinâmica.

Palavras-chave: Algoritmos genéticos, NURBS, Optimização multidisciplinar, Optimização multi-objectivo

Abstract

Power production from wind energy has been increasing for the past few decades, with more areas being used as wind farms and larger wind turbines being built. With this development, awareness of the impact of wind energy on the environment and on human health as also increased. Much research has been done to predict and reduce the noise generated by wind turbines. In this work, a blade element momentum theory model is used to predict the aerodynamic performance of a wind turbine, coupled to an empirical aeroacoustic noise model based on the works of Brooks et al (1989) and Amiet (1975), and using the XFOIL panel code for the boundary layer computations. The aeroacoustic prediction code developed was validated against measurement data of the NREL Phase II and AOC 15/50 wind turbines and used in the optimization framework pyOpt, using the genetic algorithm NSGA-II. The geometry of the blade was parameterized using NURBS curves for the cross sectional airfoil shapes and Bézier curves for the twist and chord distributions. Various optimizations were performed in blades of the two previous turbines, both single- and multi-objective, totaling up to 62 design variables. The optimal solutions are indicated in the obtained Pareto fronts and their geometries are discussed in detail. These solutions ranged from an increase in annual energy production of 139.9% to a reduction in noise levels of 10.7%. It was demonstrated that significant noise reduction could be obtained at an expense of a minor aerodynamic penalty.

Keywords: Genetic algorithms, NURBS, Multi-disciplinary design optimization, Multi-objective optimization

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α	Angle of attack, $^\circ$.
δ	Boundary-layer thickness, m.
δ^*	Boundary-layer displacement thickness, m.
ω	Wind turbine angular velocity, rad s^{-1}.
Ψ	Trailing edge solid angle, $^\circ$.
ρ	Density of air, kg m^{-3}.
σ	Local blade solidity.
θ	(i) Geometric angle of attack of the blade element, $^\circ$.
θ	(ii) Boundary-layer momentum thickness, m.

Roman symbols

$\bar{D}_l, \bar{D}_h$	Low and high frequency directivity functions.
A	spectral shape function for TBL-TE noise.
a	Axial Induction Factor.
a'	Tangential induction factor.
B	spectral shape function for separation noise.
c	Airfoil chord, m.
c	Local chord length, m.
c_0	Speed of sound, m s^{-1}.
F	Tip- and hub-loss correction factor.
f	frequency, Hz.
G_1	spectral shape function for LBL-VS noise.

G_2 R_c-dependence for LBL-VS noise peak amplitude

G_3 Angle dependence for G_2 function.

G_4 Peak level function for G_5.

G_5 Spectral shape function for TE bluntness noise.

H Boundary layer shape factor.

h TE thickness (degree of bluntness), m.

I_{turb} Turbulence intensity, %.

L Airfoil section span, m.

L_p Sound pressure level, dB.

L_W Sound power level, dB.

L_{turb} Turbulence length scale, m.

N_B Number of blades of the rotor.

P Wind turbine aerodynamic power, W.

R Rotor radius, m.

R_c Reynolds number based on chord length.

R_{hub} Rotor hub radius, m.

V_0 Mean Wind Speed, m s^{-1}.

C_d Coefficient of drag.

C_l Coefficient of lift.

C_P Wind turbine power coefficient.

C_T Thrust coefficient.

Subscripts

p Pressure side of airfoil.

s Suction side of airfoil.

Chapter 1

Introduction

For many years people have harnessed the energy of wind, starting with the propulsion of ships using sails and the windmills used to grind grain and pump water for irrigation. With the invention of the steam engine and other technologies for converting fossil fuels into useful energy, however, the role of wind in energy generation would reduce to become almost insignificant. In the beginning of the 1990s, the situation was different, and the reversal which had started in the late 1960s was becoming apparent. The wind energy industry had been increasing capacity, and it was also in that decade when the shift to megawatt-sized wind turbines happened, together with a consolidation and reduction of wind turbine manufacturers. This change in direction was triggered by the oil crisis in the mid 1970s, that led to the sudden increase in the price of oil, as well as the emerging awareness of the finiteness of the Earth's fossil fuels reserves and of the consequences of burning those fuels, enhanced by books such as *Silent Spring* by Carson (1962) or *Limits to Growth* by Meadows et al. (1974). All this stimulated the creation of a large number of Government-funded research and development programmes focused on renewable energies in general and wind energy in particular.

1.1 Wind Energy Worldwide

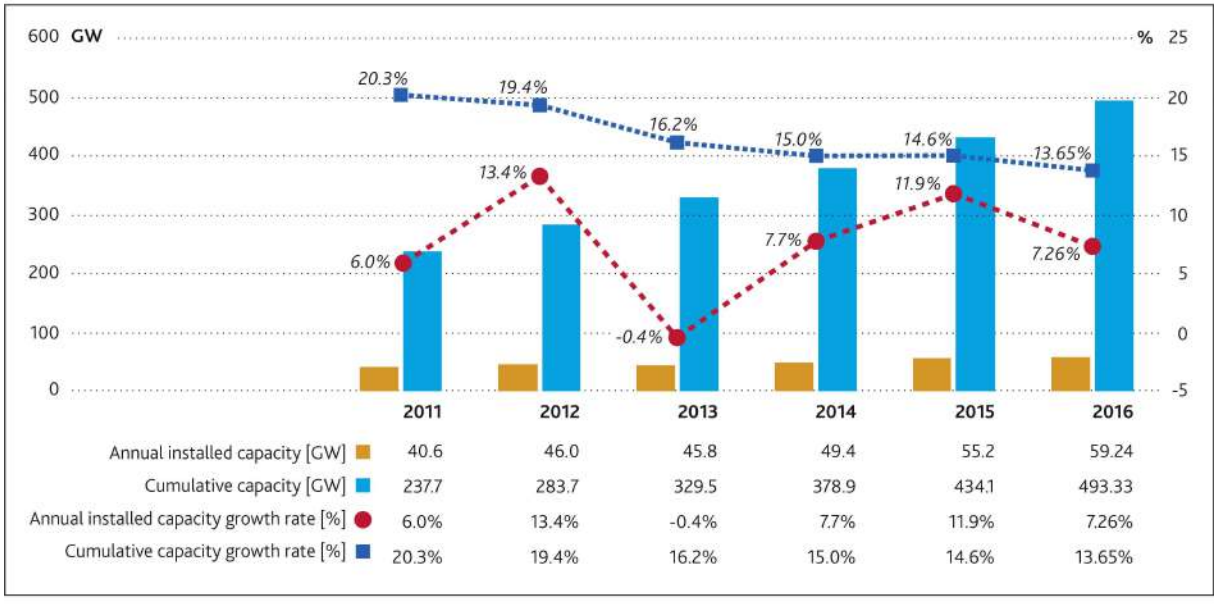
Nowadays, the use of wind power to generate electricity is widely spread across the world. In the last few years, the worldwide wind energy production has been strongly increasing (see Tab. 1.1) and by the end of June 2012, the total worldwide installed capacity reached 254 GW. China, the USA and Germany lead the energy production capacity while Portugal occupies the 10th place, with a production capacity of 4398 MW (Tab. 1.1). This capacity is expected to keep increasing according to the forecast elaborated by the Global Wind Energy Council presented on Fig. 1.1. Supporting this forecast are studies such as Marvel et al. (2012), which claims that there is enough power in Earth's winds to be a primary source of near-zero-emission electric power as the global economy continues to grow through the twenty-first century.

Position	Country	Total Capacity by June 2012 [MW]	Added Capacity first half 2012 [MW]	Total Capacity end 2011 [MW]	Added Capacity first half 2011 [MW]	Total Capacity end 2010 [MW]
1	**China**	67'774	5'410	62'364	8'000	44'733
2	**USA**	49'802	2'883	46'919	2'252	40'180
3	**Germany**	30'016	941	29'075	766	27'215
4	**Spain**	22'087	414	21'673	480	20'676
5	**India**	17'351	1'471	15'880	1'480	13'065
6	**Italy***	7'280	490	6'787	460	5'797
7	**France****	7'182	650	6'640	400	5'660
8	**United Kingdom**	6'840	822	6'018	504	5'203
9	**Canada**	5'511	246	5'265	603	4'008
10	**Portugal**	4'398	19	4'379	260	3'702
	Rest of the World	35'500	3'200	32'227	3'200	29'500
	Total	254'000	16'546	237'227	18'405	199'739

* till end of May 2012 ** till end of April 2012 © WWEA 2012

Table 1.1: Top 10 leading countries in wind energy production (WWEA, 2012).

Market Forecast 2012-2016



Source: GWEC

Figure 1.1: Wind energy market forecast of the 2012-2016 period (GWEC, 2011).

1.2 Environmental Impact of Wind Energy

All technologies have flaws and wind energy is no exception. Even being regarded in general as environmentally friendly, wind farms have impact, and with the increase of the installation of wind turbines, awareness towards them has also increased. The potential negative effects of wind energy can be divided into the following categories:

- Avian / bat interaction;
- Visual impact;
- Wind turbine noise;
- Electromagnetic interference;
- Land-use impact;
- Other.

All these categories should be addressed in certain phases of a wind project and each one has its own regulation, which varies from country to country or even from site to site.

The problem of wind turbine noise has been one of the most studied environmental impact subjects. It is controversial as, although noise levels can be measured, its impact on the environment and the public's perception of the noise is partly subjective. There have been, however, many studies on the impact of wind turbine noise on human health and wildlife (Stephens, 1982; Colby et al., 2009).

1.3 Legislation on WT Noise

As mentioned before, the legislation regarding noise varies from country to country, as a result of the inexistence of common international noise standards or regulations for sound pressure levels. However, the basics are the same: they establish maximum sound levels that can be produced for a particular location (or type of location) and time of day (day/night). Taking Portugal as an example, the limits for noise exposure according to it's law are presented in Tab. 1.2.

Area type	L_{den} (dB(A))	L_n (dB(A))
sensitive	< 55	< 43
mixed	< 65	< 55
not defined	< 63	< 53

Table 1.2: Noise limits of sound pressure level according to the Portuguese legislation (Ministério do Ambiente do Ordenamento do Território e do Desenvolvimento Regional, 2007).

There are three different zones defined by the Portuguese reference: *sensitive* areas, defined as being oriented towards residential use, schools, hospitals, etc; *mixed* areas, which are defined as having the same profile of the sensitive areas, with the exception of a clause allowing their use for other unplanned purposes; and the areas which are neither sensitive, neither mixed, referred to as *not defined*. In the table, two values are given: L_{den}, which is the averaged sound pressure level over a period of 24 hours, and L_n, which is the sound pressure level during night time.

Other countries around Europe have similar limits of noise levels, as seen in Tab. 1.3.

Country		Commercial	Mixed	Residential	Rural
Denmark				40	45
Germany	day	65	60	55	50
	night	50	45	40	35
Netherlands	day		50	45	40
	night		40	35	30

Table 1.3: Noise limits of equivalent sound pressure levels L_{eq} (dB(A)) in some European countries (Gipe, 1995).

Sometimes a penalty of typically 5 dB(A) can be added to the noise limits, due to the tonal noise, and the legislations also specify how the noise levels should be measured. In Wagner et al. (1996) and Hubbard and Shepherd (1990) reviews of noise measurement techniques are given.

1.4 State of The Art

Noise Prediction Research on the prediction of noise generated by wind turbines has been done for many years and bibliography listing technical papers on the subjects of wind turbine acoustics is available (Hubbard and Shepherd, 1988). In Lowson (1993) a prediction model is presented, based on the empirical models of Brooks et al. (1989) and Amiet (1975). Fuglsang and Madsen (1996) implemented a wind turbine aeroacoustic noise prediction model, similar to Lowson's, coupled with an aerodynamic prediction model, having validated its predictions against measurement data from the Bonus Combi 300 kW and performed optimization of the geometry of the blade. Similar implementations of aeroacoustic wind turbine were done by Zhu (2004), Leloudas (2006), Vanhaeverbeke (2007) and Vargas (2008). A thorough description of the various different prediction models available is also presented in Wagner et al. (1996).

Nowadays many codes exist that predict the noise of a wind turbine by computing all of its components, such as the SILANT (Boorsma and Schepers, 2012), the IAGNOISE (Kamruzzaman et al., 2009) or the FAST codes, to which a aeroacoustic prediction model was added by Moriarty and Migliore (2003). Tickell et al. (2004) made a comparison of various other prediction codes available. Recent developments in noise prediction include the use of Computational Fluid Dynamics (CFD) (Tadamasa and Zangeneh, 2011).

Noise Reduction The aim of every prediction codes is to be used in the reduction of wind turbine noise. The project SIROCCO (Schepers et al., 2007) was developed with the aim to reduce wind-turbine aerodynamic noise significantly while maintaining aerodynamic performance by designing new aero-acoustically optimized airfoils whose measurements were used to validate the SILANT code.

While some works aimed to reduce the noise by changing the shape parameters of the blade (Leloudas, 2006; Vesel Jr, 2009), other works performed optimization on the 2D airfoil shape to obtain

low noise airfoils (Bizzarrini et al., 2011; Coimbra, 2012; Göçmen and Özerdem, 2012). Oerlemans et al. (2009) presented acoustic measurement data showing that using acoustically optimized blades on a wind turbine reduced the overall noise up to 0.5 dB and using trailing-edge serrations on the blade reduced the noise levels up to 3.2 dB, without reducing the aerodynamic performance of the turbine.

1.5 Report Overview

This thesis consisted in the aeroacoustic optimization of wind turbine blades. For this, a wind turbine aeroacoustic prediction code was developed which can be divided into three parts: the aerodynamic prediction, the acoustic prediction and the geometry definition. The theory behind the developed code is presented in chapters 2, 3 and 4, respectively. Each chapter starts with an introduction to the basic concepts behind the theory followed by the description of the models. In chapter 5, a description of the structure of the implemented code along with the features of the code is presented. The validation of the code is also presented in chapter 5. The optimization framework developed is described in chapter 6 and in chapter 7 the results of the optimizations are presented and discussed. The thesis ends in chapter 8 with overall conclusions and remarks, as well as future work options.

Chapter 2

Wind Turbine Aerodynamics

A wind turbine extracts mechanical energy from the kinetic energy of the wind by slowing down the wind. It can either be a Horizontal-Axis Wind Turbine (HAWT) or a Vertical-Axis Wind Turbine (VAWT), depending on either it rotates around its horizontal axis or vertical axis, respectively. In the present work, only HAWTs will be treated.

If the mass of air passing through the turbine is assumed to be separated from the mass that does not pass, the separated part of the flow field remains a long stream tube lying up and downstream of the turbine. As the flow approaches the turbine, its velocity drops and, in order to compensate for this drop, the stream tube expands (Fig. 2.1).

Many methods for computing the performance of wind turbine exist. In the 1930s, Betz and Glauert derived the classical analysis method, the Blade Element Momentum (BEM) theory, which combines the Blade Element and Momentum theories.

In this chapter, this theory is revisited, together with some more recent developed corrections. The chapter ends with the introduction to some corrections to the aerodynamic data used in the theory.

2.1 Actuator Disk Concept

The actuator disk concept describes the basic aerodynamic flow in the wind turbine. According to it, the wind turbine is considered as an ideal actuator disk: frictionless, with an infinite number of blades and with no rotational velocity component in the wake. The flow is also considered homogeneous and steady, while the air is considered incompressible.

From the non-dimensionalization of the difference between the free stream velocity V_0 and u, which is the axial induced velocity, the axial induction factor is defined as

$$a = \frac{V_0 - u}{V_0}. \tag{2.1}$$

The shaft power P can be found by using the energy equation on a control volume defined by the streamtube and assuming no change in the internal energy of the flow (since it is assumed to be

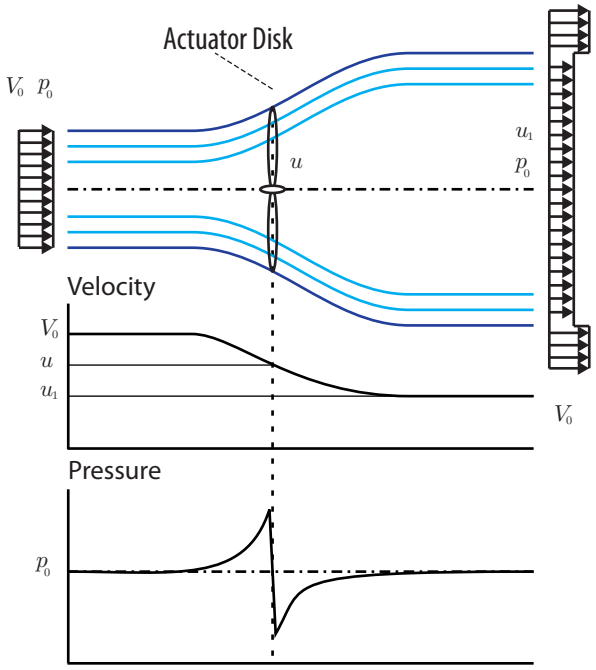


Figure 2.1: Actuator disk.

frictionless). It yields

$$P = 2\rho V_0^2 a\,(1-a)\,A_R, \tag{2.2}$$

where A_R is the area of the rotor and which is often non-dimensionalized with respect to P_{avail} as a power coefficient C_P,

$$C_P = \frac{P}{\frac{1}{2}\rho V_0^3}. \tag{2.3}$$

The power coefficient for the ideal one-dimensional wind turbine may also be written as

$$C_P = 4a(1-a)^2. \tag{2.4}$$

Differentiation C_P with respect to a yields

$$\frac{dC_P}{da} = 4\,(1-a)\,(1-3a)\,. \tag{2.5}$$

From Eq. (2.5) the maximum value of $C_P = 16/27 \approx 0.593$ is obtained for $a = 1/3$. This theoretical maximum value is known as the Betz limit and it is not possible to design a wind turbine that goes beyond it.

2.2 Blade Element Momentum Theory

Although it does not produce as accurate results as a full Computational Fluid Dynamics (CFD) simulation or other prediction methods such as vortex line theory, the Blade Element Momentum method is the most widely used theory for rotor design and analysis due to its speed and simplicity. It equates the force and

torque relations derived from momentum theory and blade element theory in order to model the axial and tangential induction factors, where the following assumptions are made: there is no aerodynamic interaction between blade elements; the radial velocity component is ignored and the forces on the blade element are functions only of the aerodynamics characteristics of the 2D airfoil of the element.

The full derivation of the BEM theory equations can be found in most wind turbine design handbooks (Hansen, 2012; Burton et al., 2001) and only a summarized derivation of the method will be presented.

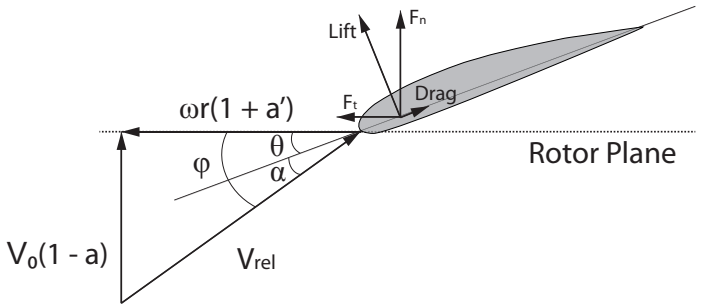


Figure 2.2: Velocities at the rotor plane.

We first start by inspection Fig. 2.2, from which it can be derived that

$$\tan\phi = \frac{(1-a)\,V_0}{(1+a')\,\omega r} \tag{2.6}$$

where V_0 is the local inflow velocity, a is the induction factor, representing the fraction of the incoming flow velocity that is removed, a' is the tangential induction factor, representing the rotational speed in the wake, ω is the rotational speed, and r is the distance to the rotor center. The angle θ indicated in the figure represents the geometrical angle of attack of the element.

From the momentum theory, the thrust extracted by each rotor annulus is given by

$$dT = 4\pi r \rho V_0^2 a\,(1-a)\,F dr, \tag{2.7}$$

while the torque extracted in the same annular section is given by

$$dM = 4\pi r^3 \rho V_0 \omega\,(1-a)\,a' F dr. \tag{2.8}$$

where F is a factor that takes the hub- and tip-losses into account. If these losses are not to be taken into account F is equal to 1. This shall be further explained in Section 2.2.1.

On the other hand, the thrust and torque distributed through an annulus of width dr can be obtained, from blade element theory, using

$$dT = \frac{1}{2}\rho N_B \frac{V_0^2(1-a)^2}{\sin^2\phi} c C_n dr \tag{2.9}$$

and

$$dM = \frac{1}{2}\rho N_B \frac{V_0\,(1-a)\,\omega r\,(1+a')}{\sin\phi\cos\phi} c C_t r dr. \tag{2.10}$$

where C_n and C_t are the coefficients of the resulting aerodynamic forces in the direction normal and tangent to the rotor plane (see Fig. 2.2), N_B is the number of blades and c is the chord of the element.

Equalizing Eq. (2.7) and Eq. (2.9) for dT, the expression for the axial induction factor a is obtained as

$$a = \frac{1}{\frac{4F\sin^2\phi}{\sigma C_n} + 1}, \tag{2.11}$$

where $\sigma = c(r)N_B/2\pi r$ is the local solidity. If Eq. (2.8) and Eq. (2.10) are equalized for dM, the expression for a' is obtained as

$$a' = \frac{1}{\frac{4F\sin\phi\cos\phi}{\sigma C_t} - 1}. \tag{2.12}$$

2.2.1 Corrections to BEM theory

Tip Loss Model

The original blade element momentum theory does not account for the influence of the vortexes shed from the blade tips into the wake on the induced velocity field. Prandtl derived a correction factor F to compensate for this deficiency in BEM theory (Glauert, 1935) and it is computed as

$$F = \frac{2}{\pi}\cos^{-1}\left(e^{-f}\right), \tag{2.13}$$

where

$$f = \frac{N_B}{2}\frac{R-r}{r\sin\phi}, \tag{2.14}$$

where R is the blade radius and r the radius at a specific location.

Hub Loss Model

As there are also vortexes being shed near the hub of the rotor, another correction factor can also be applied to correct the induced velocity. The hub-loss model is nearly identical to the tip-loss model, with the following equation replacing Eq. (2.14):

$$f = \frac{N_B}{2}\frac{r-R_{hub}}{r\sin\phi}. \tag{2.15}$$

Each element of the blade can be affected by both the tip-loss and the hub-loss factors, with the total factor being a multiplication of the tip-loss factor by the hub-loss factor.

$$F = F_{hub}F_{tip} \tag{2.16}$$

Glauert Correction

As previously noted, the BEM theory is based on various assumptions, which exclude the 3D characteristics of the flow, turbulence or separation. These assumptions work when the wind turbine is working

in conditions where 3D characteristics can be ignored. This is not the case when the induction factor is greater than about 0.4, where the 3D effects cannot be ignored and the basic BEM theory becomes invalid. When the turbine operates at hight tip speed ratios (e.g. constant speed turbine at low wind speeds), the rotor enters the *turbulent wake state* ($a > 0.5$), where, according to momentum theory, some of the flow in the far wake starts to propagate upstream - a violation of the basic assumptions of the BEM theory. What happens in reality is that more flow entrains from outside the wake and the turbulence increases, slowing down the flow behind the rotor while the thrust on the rotor increases. Glauert (1935) developed a correction to the rotor thrust coefficient to compensate for this effect, based on experimental measurements of helicopter rotors with large induced velocities. This correction has also been used to correct the local coefficient of the individual blade elements, when used with the BEM theory. Buhl (2005) derived a modification to Glauert's relation, including the tip-loss correction:

$$C_T = \frac{8}{9} + \left(4F - \frac{40}{9}\right) a + \left(\frac{50}{90} - 4F\right) a^2 \tag{2.17}$$

which, solving for the induction factor, leads to:

$$a = \frac{18F - 20 - 3\sqrt{C_T\,(50 - 36F) + 12F\,(3F - 4)}}{36F - 50} \tag{2.18}$$

The previous equation substitutes Eq. (2.11) and this correction should be applied to the BEM model when $C_T > 0.96F$ (Buhl, 2005). Figure 2.3 shows an example of the Glauert correction for a tip loss factor of 1.0. When the induction factor a is equal to 1, the BEM theory and the Glauert correction produce the same thrust coefficient of 0.96.

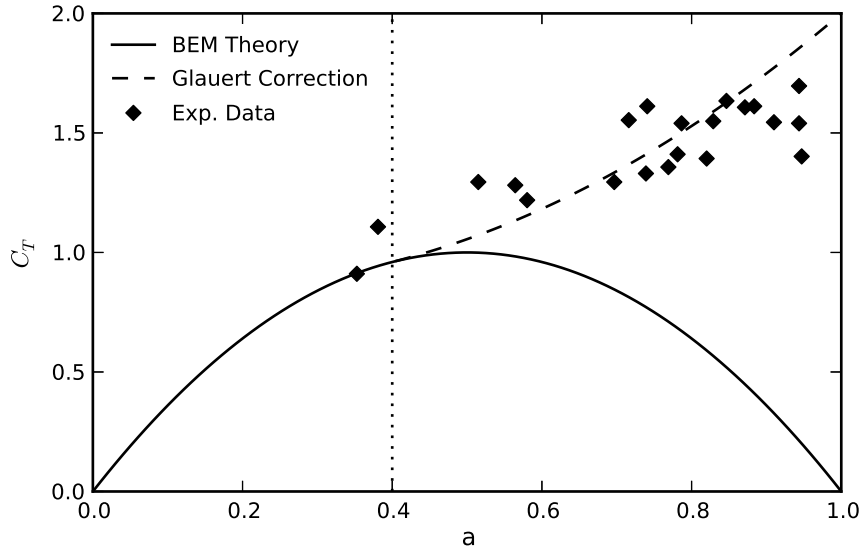


Figure 2.3: Glauert correction for tip loss factor $F = 1.0$ (Exp. data from Lock et al. (1926)).

2.2.2 Iteration Procedure

With all the necessary equations for the BEM model derived, the iteration procedure can be summarized. This procedure is repeated for each element of the blade.

Step 1 Initialize a and a' (usually $a = a' = 0$).

Step 2 Compute the flow angle ϕ using Eq. (2.6)

Step 3 Compute the local angle of attack α with

$$\alpha = \phi - \theta. \tag{2.19}$$

Step 4 Obtain C_l and C_d from an aerodynamic table.

Step 5 Compute C_n and C_t.

Step 6 Compute the tip- and hub-loss corrections using Eq. (2.14) and Eq. (2.15).

Step 7 Compute the thrust coefficient for the element using

$$C_T = \frac{\sigma(1-a)^2 \left(C_l \cos\phi + C_d \sin\phi\right)}{\sin^2\phi} \tag{2.20}$$

Step 8 Compute the axial induction a using Eq. (2.11), or, if $C_T > 0.96F$, using Eq. (2.18).

Step 9 Compute tangential induction factor using Eq. (2.12).

Step 10 If a and a' have changed more than a certain tolerance, go to Step 2 or else finish.

2.3 Annual Energy Production

With the wind turbine power curve (the shaft power as a function of the wind speed V_0) obtained, it is then possible to compute the annual energy production of the wind turbine. In order to do so, it is necessary to combine this production curve with a probability density function h for the wind. Typically, the probability density function of the wind is given by either a Rayleigh or a Weibull distribution (see Fig. 2.4). The Weibull distribution can be modeled through of a scaling factor A and a form factor k:

$$h_w\left(V_0\right) = \frac{k}{A}\left(\frac{V_0}{A}\right)^{k-1} \exp\left(-\left(\frac{V_0}{A}\right)^k\right) \tag{2.21}$$

From the Weibull distribution, the probability $f\left(V_i < V_0 < V_{i+1}\right)$ that the wind speed lies between V_i and V_{i+1} is given by

$$f\left(V_i < V_0 < V_{i+1}\right) = \exp\left(-\left(\frac{V_i}{A}\right)^k\right) - \exp\left(-\left(\frac{V_{i+1}}{A}\right)^k\right). \tag{2.22}$$

The total annual energy production (AEP) can then be estimated as the product of the probability function and the power curve. In the discrete form it yields

$$AEP = \sum_{i=1}^{N-1} \frac{1}{2} \left(P\left(V_{i+1}\right) + P\left(V_i\right) \right) \cdot f\left(V_i < V_0 < V_{i+1}\right) \cdot 8760, \tag{2.23}$$

where 8760 is the number of hours in a year.

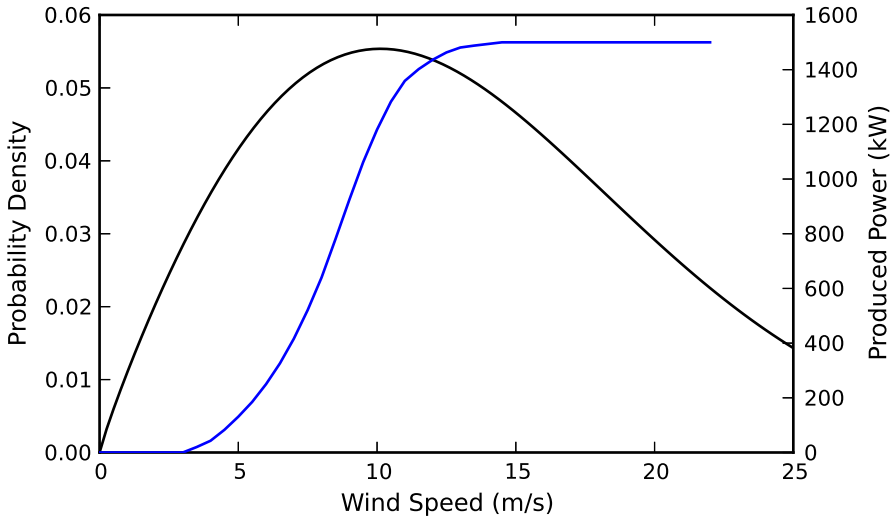


Figure 2.4: Weibull distribution curve ($A = 10$; $k = 1.9$) (black line) and power curve (blue line) for a generic wind turbine.

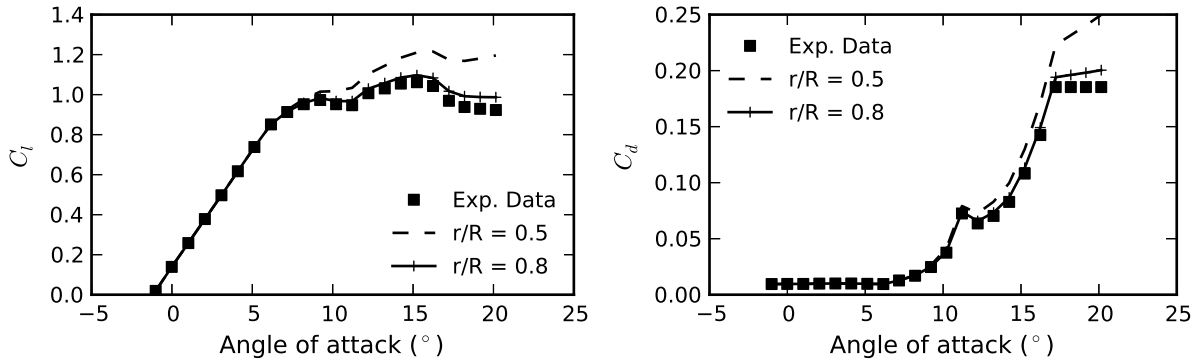
2.4 Airfoil Aerodynamic Data

The aerodynamic data for use in the blade element momentum can be obtained by a series of methods, such as experimental measurements or computational method. Due to the fact that the BEM model is as good as the data itself, there should be an effort to supply it with the most accurate data possible. With this in mind, two corrections to the aerodynamic data are introduced in this section. The first is a stall-delay 3D correction, which accounts for the 3D effects felt by the flow around the blade. The other is not so much a correction but an extrapolation of the available data to a larger range of angles of attack.

2.4.1 3D Stall-Delay Correction

As previously mentioned, one of the simplifying assumptions of BEM theory is that flow in the radial direction is neglected. In reality, the presence of spanwise flow may cause significant changes in the aerodynamic behavior of the airfoils when compared to a two-dimensional case, especially near the blade root. The airfoil data used in the BEM method usually comes from 2D wind tunnel measurements or computational methods. If that is the case, a stall-delay model can be used to apply corrections to the 2D airfoil data, which can significantly increase the accuracy of the BEM predictions. In the present work, the stall delay model used is the one from Du and Selig (1998), with the drag adjustment from Eggers

et al. (2003). Fig. 2.5 shows the lift and drag coefficients before (markers) and after the correction using the 3D stall-delay model. The correction is applied considering two different radial positions r/R of the airfoil in the blade: 0.5 (dashed line) and 0.8 (solid line). It can be observed that the difference between the original and the corrected data is larger near the root, where the 3D characteristics of the flow are stronger.



(a) Lift coefficient

(b) Drag coefficient

Figure 2.5: Lift and drag coefficients of the S809 airfoil before (markers) and after (solid and dashed lines) 3D stall delay correction. (Exp. data from Delft University (Schepers et al., 2001))

2.4.2 Polar Extrapolation

During the BEM iterations and particularly for elements in the root region, the angle of attack calculated for the element can become very large. To obtain the aerodynamic data over this range of angles of attack, the polar data can be extrapolated by assuming that the airfoil behaves like a flat plate at high angles of attack. This has the advantage of the aerodynamic coefficients being dependent only of the aspect ratio of the plate. Viterna and Janetzke (1982) developed such procedure, which was adopted here.

The procedure first starts by calculating the maximum value of the drag ($\alpha = 90°$) as a function of the aspect ratio of the blade AR as

$$C_{d_{max}} = 1.11 + 0.018AR \tag{2.24}$$

The drag at each angle of attack is then given by

$$C_d = C_{d_{max}} \sin^2\alpha + B_2 \cos\alpha, \tag{2.25}$$

with

$$B_2 = \frac{C_{d_s} - C_{d_{max}} \sin^2\alpha_s}{\cos\alpha_s}, \tag{2.26}$$

where the s subscript denotes the value at stall. The lift coefficient is similarly obtained, with the expressions

$$C_l = \frac{C_{d_{max}}}{2} \sin 2\alpha + A_2 \frac{\cos^2\alpha}{\sin\alpha}, \tag{2.27}$$

and

$$A_2 = (C_{l_s} - C_{d_{max}} \sin\alpha_s \cos\alpha_s) \frac{\sin\alpha_s}{\cos^2\alpha_s}. \tag{2.28}$$

This procedure is executed for angles of attack between the stall angle and 90° . For angles between 90° and 180° , the lift coefficient values are obtained by scaling and reflecting the values from the higher available data to 90 ° .

Figure 2.6 shows the original aerodynamic data of the S809 airfoil and the corresponding extrapolated data for two different blade aspect ratios.

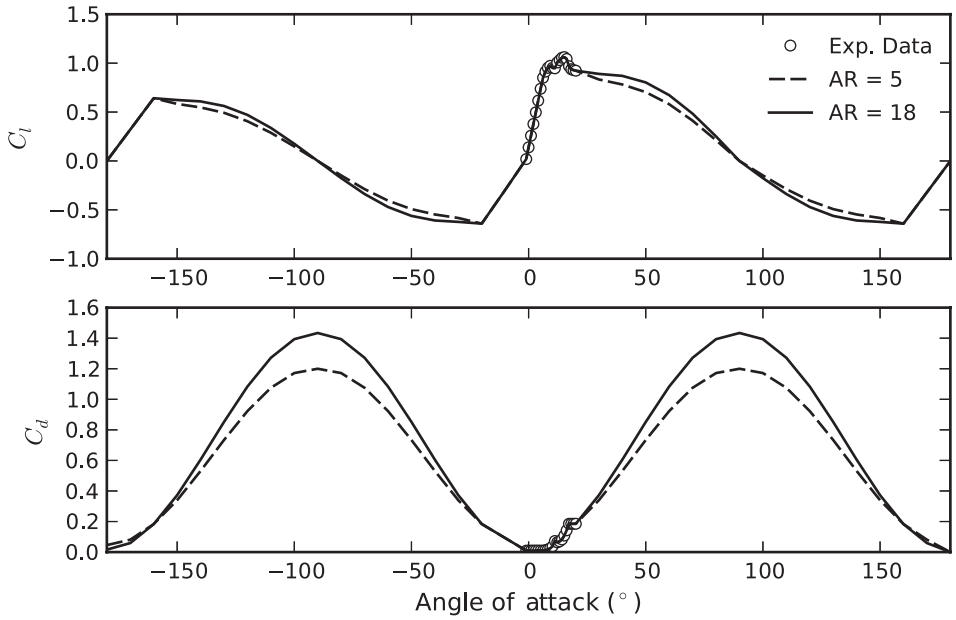


Figure 2.6: Original (Schepers et al., 2001) and extrapolated S809 airfoil data for two different blade aspect ratios.

Chapter 3

Noise from Wind Turbines

In this chapter, the theory behind the generation of noise from a wind turbine is presented. It starts with a brief introduction to some key concepts followed by a description of the wind turbine noise generation mechanisms. It ends with the description of the noise prediction model developed in this work.

3.1 Principles of Acoustics

Sound is an oscillation of pressure propagating through a medium as sound waves. It can be generated by various different mechanisms. Some well-known examples are the loudspeaker, where the sound is produced by a vibrating surface, or the siren, where the sound is created by the periodic injection of air. The sound waves are characterized in terms of their wavelength λ, frequency f, and velocity c_0 (approximately $340\,\mathrm{m/s}$ in air, at standard conditions). After being partially absorbed, reflected or attenuated, these waves may reach the human eardrum of an observer, where they produce a sensation of hearing, depending on the amplitude of the sound wave. This sensation might turn the sound into *noise*, if it is considered unwanted. The classification of sound as noise is very subjective as it depends on factors such as the sensitivity of the listener and the situation, as well as measurable quantities like level and duration.

In this section, only some basic concepts are introduced and a reader less familiar with the aeroacoustic fundamental theories of aeroacoustics should refer to available literature such as (Lau, 2011) or (Goldstein, 1976).

3.1.1 Sound Pressure and Power Levels

The response of the human ear to the amplitude of sound pressure is not linear. For example, if the amplitude of the sound pressure is doubled it produces a sensation of a louder sound, however, it seems far less than twice as loud. This leads to the definition of a logarithmic scale to characterize sound pressure amplitudes, as it approximates the actual response of the human ear. The sound pressure level

L_p is expressed in deciBel (dB) and can be defined as

$$L_p = 10\log_{10}\left(\frac{p_{rms}^2}{p_{ref}^2}\right), \tag{3.1}$$

where p_{ref} is a reference pressure (usually $20 \times 10^{-5}\,\mathrm{Pa}$), and p_{rms} is the root mean square sound pressure defined by

$$p_{rms}^2 = \lim_{T\to\infty}\left(\frac{1}{T}\int_0^T p^2(t)dt\right). \tag{3.2}$$

According to this definition, the doubling of the sound source results in a increase of sound pressure level of 3 dB. In Fig. 3.1 the relationship between the sound pressure level and the actual sound pressure is represented as well as some examples of sound sources and their typical sound levels.

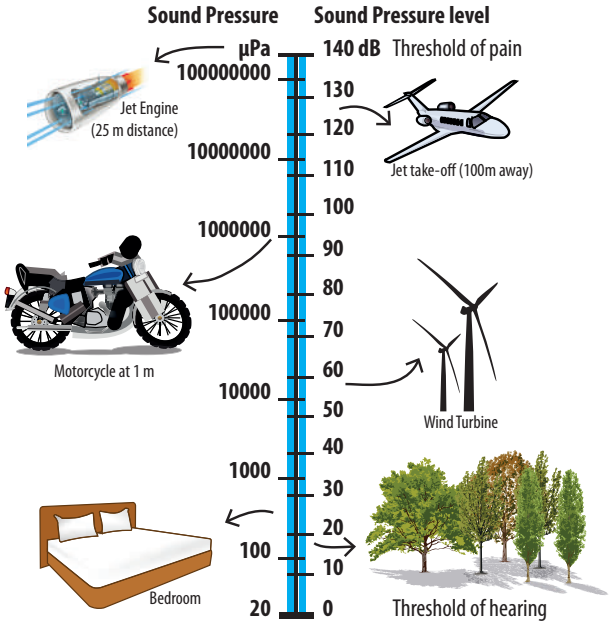


Figure 3.1: Sound pressure level examples.

Sound pressure level is a property of a field position, this is, an observer will experience different sound pressure values for different positions relative to the sound source. In order to characterize the strength of a source of sound, it is common to use the sound power emitted by that source. The sound power level L_W scales this power in the same way the sound pressure level does to the sound pressure. It relates to the sound pressure level by

$$L_W = L_p + 10\log_{10}\left(\frac{4\pi R^2}{S_0}\right), \tag{3.3}$$

with R being the distance to the source, and S_0 a reference area (typically 1 m^2).

3.1.2 Sound Frequency Spectrum

Besides the amplitude of the emitted sound, the frequency spectrum is necessary to characterize the source. The spectrum indicates the prevalent frequencies in a sound pressure signal, revealing whether

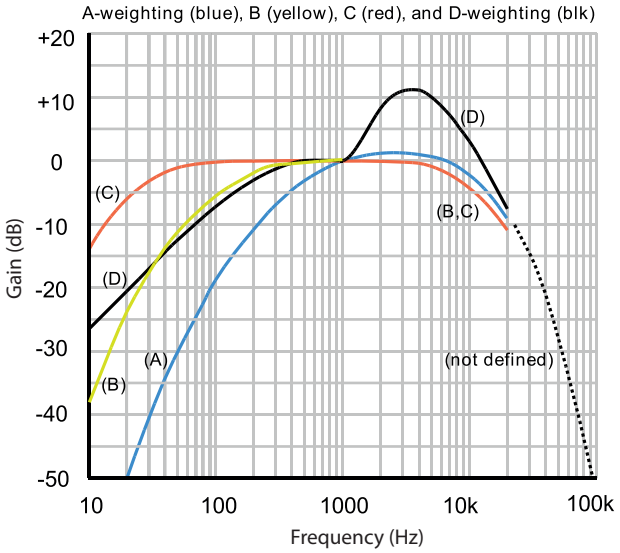


Figure 3.2: A-, B-, C- and D-weightings (Skirrow, 2005).

there are tonal components of a broadband swishing. The frequency range of the spectrum can be divided into several bands and the three most commonly used types of bands are the narrow bands, 1/3-octave bands, and 1/1-octave bands.

In the narrow-band spectrum representation each frequency band has the same width Δf and thus this representation gives the most detailed picture of a sound signal. In a 1/1-octave band, the upper bounding frequency is double the lower (which in music is called an octave). In a 1/3-octave band, the upper frequency is $\sqrt[3]{2}$ times the lower frequency. Since the frequencies obtained from these formulas are not integer numbers, the center frequencies are rounded according to (ISO 266, 1997).

The ear does not respond equally to tones of different frequencies. At the frequencies where the maximum response occurs (between 3000 and 4000 Hz), the threshold of hearing is somewhat less than 0 dB, while a 100 Hz tone must have an intensity level of at least 40 dB to be heard. A-, B-, and C-weighted sound levels were introduced in order to de-emphasize the lower frequencies in a manner similar to human hearing. Fig. 3.2 shows the gains of the various weightings as a function of frequency. A-weighting is the most commonly used and is well suited for not too high levels (Wagner et al., 1996), while B- or C- weightings are more appropriate when strong low-frequent sound levels occur. The A-weighted sound level is measured in dB(A).

3.2 Noise Mechanisms in Wind Turbines

The mechanisms by which the noise from a wind turbine is generated can be distinguished between mechanical and aerodynamic, as depicted in Fig. 3.3.

3.2.1 Mechanical Noise

Mechanical noise is generated mainly from the relative motion of mechanical components, and the dynamic response among them. The main sources of mechanical noise are the gearbox, generator,

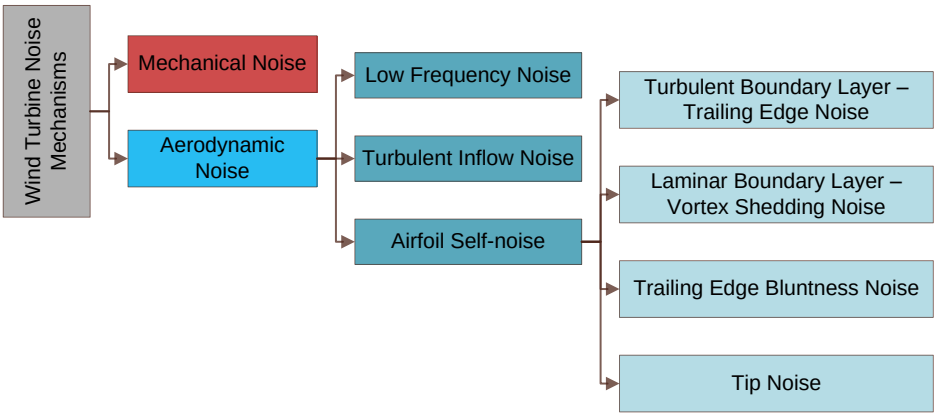


Figure 3.3: Wind turbine noise mechanisms.

cooling fans and auxiliaries such as the oil coolers or hydraulic power packs for blade pitch. Of these noise sources, the gearbox and generator are the ones that produce the highest noise emission.

The spectrum of the mechanical noise of a wind turbine shows a number of prominent tones.

Modern large wind turbines feature many noise reduction techniques, such as nacelle's acoustic isolation, vibrations and loadings damper systems, etc.

3.2.2 Aerodynamic Noise

Mainly associated with the interaction of turbulence with the blade surface, aerodynamic noise can be divided into three main types: low frequency noise, turbulent inflow noise and airfoil self-noise. Low frequency noise occurs due to the passage of the blades through the towers wake. Turbulent inflow noise is generated due to the interaction of turbulence of the incoming flow with the turbines blades and airfoil self-noise is the result of the interaction between the blade and the turbulence produced in its boundary layer and near wake. Airfoil self noise can be subdivided into five different mechanisms: Laminar Boundary Layer Vortex Shedding noise (LBL-VS), Turbulent Boundary Layer Trailing Edge noise (TBL-TE), Separation-Stall noise, Trailing Edge Bluntness Vortex Shedding noise (TEB-VS) and Tip Vortex Formation noise (TVF).

Low Frequency Noise

Low frequency noise describes the sound in the region below about 200 Hz. Generated by the change in flow encountered by the rotor blades due to the presence of the tower, this noise is specially important for downwind turbines, as the airflow will not follow the curvature of the tower and will separate, developing a turbulent wake. The presence of the tower will reduce the velocity of the flow that enters the rotor in that region, for both an upwind and downwind configuration. This reduction is, however, more pronounced downstream of the tower. As the blades encounter this lower speed region, the loading on the blade will change in dynamic pressure and local angle of attack, acting as a dipole type loading noise.

The passage of the blades is directly related to this noise mechanism, therefore, the blade passing

frequency f_B and its harmonics f_b, which are functions of the rotor frequency f_R, dominate the spectrum.

$$f_B = n_B \cdot f_R \tag{3.4}$$

$$f_b = n \cdot f_B \tag{3.5}$$

For wind turbines, f_B is typically in the order of 1-3 Hz and this low frequency noise contributes only to the low frequency part of the wind turbine noise spectrum.

Nowadays, most wind turbines have an upwind configuration and the ones with a downwind configuration incorporate design features that reduce impulsive noise, such as the positioning of the rotor further away from the tower. Moreover, due to low rotor frequencies, low frequency noise is not an important part of the spectrum of large wind turbines. It may, however, become important for small wind turbines with higher rotational speeds, where the passing frequency and its harmonics may shift to the audible part of the spectrum.

Turbulent Inflow Noise

When atmospheric turbulence encounters the blade, it causes a broadband noise radiation, called turbulent inflow noise.

There are two causes for atmospheric turbulence: aerodynamic and thermal. Aerodynamic turbulence is generated by the interaction of the flow with the ground surface, while thermal turbulence is generated by the buoyancy of the air due to local heating by the sun. The two most important characteristics of turbulence are turbulence intensity, which indicates the turbulence fluctuations and is defined as the ratio of the standard deviation and the averaged mean wind velocity, and the turbulence length scale, which indicates the size of the eddies.

The size of the eddies is related to the frequency of the generated noise by this mechanism. Eddies larger than the airfoil chord will generate low frequency noise as it changes the airfoil loading as a whole while eddies smaller than the airfoil chord will induce localized pressure fluctuations, thus producing high frequency noise.

Turbulent Boundary Layer Trailing Edge Noise

Considered one of the major contributors to airfoil self-noise, turbulent boundary layer trailing edge (TBL-TE) noise is the result of the interaction between the turbulent boundary layer that develops over the blade surface and the trailing edge.

In any airfoil subject to a flow, a boundary layer develops on its surface, starting from the stagnation point close to the leading edge. As certain angle of attack and Reynolds number conditions are met, the boundary layer transitions from laminar to turbulent at a certain chordwise position. Beneath this boundary layer, the turbulence induces a fluctuating pressure field. Turbulent boundary layer trailing edge noise is perceived as a swishing sound i.e. broadband. Its peak frequency is typically in the order of 500-1500 Hz (Wagner et al., 1996). This mechanism is schematically represented in Fig. 3.4.

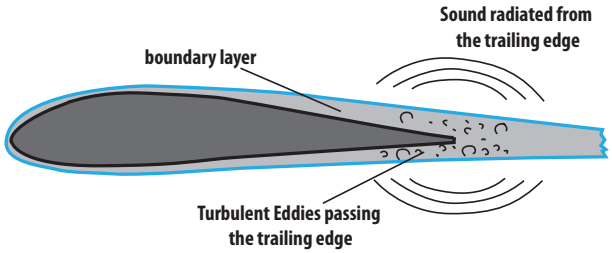


Figure 3.4: Representation of the TBL-TE noise mechanism.

Separated Stall Noise

As the angle of attack increases, separation can occur on the suction side of the airfoil, forming a zone containing highly unsteady, recirculating flow. The shedding of these vortices into the wake produces the so called separated stall noise. This noise is broadband with its peak frequency typically intermediate to low. This mechanism is schematically represented in Fig. 3.5.

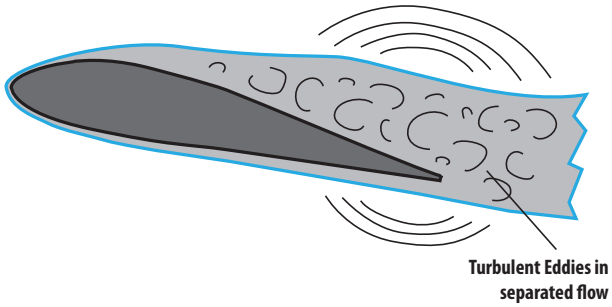


Figure 3.5: Representation of the separation-stall noise mechanism.

Trailing Edge Bluntness Vortex Shedding noise

Trailing edge bluntness vortex shedding noise (TEB-VS) is the result of the vortex shedding from blunt trailing edges, also known as the van Karman type vortex sheet. As the coherent vortexes are shed, they cause a fluctuating surface pressure differential across the trailing edge, which results in a tonal radiation of discreet frequencies at the trailing edge. This noise mechanism is strongly dependent on the detailed geometry of the trailing edge, increasing with TE thickness and angle. According to Blake (1986), it can increase the relative vibration amplitude up to 360% as well as a decrease down to approximately 1%, when compared to the squared-off blunt edge. Therefore, in order to reduce the noise generated by this mechanism, the airfoils trailing edge angle and thickness should be as small as the manufacturing processes allow. This mechanism is schematically represented in Fig. 3.6.

Laminar Boundary Layer Vortex Shedding Noise

Laminar boundary layer vortex shedding noise is generated due to an interaction between laminar boundary layer instabilities and the vortexes shed at the trailing edge. Most modern turbines operate at much higher local Reynolds numbers, i.e. $Re > 3 \times 10^6$, therefore this noise mechanism is of minor importance. This mechanism is schematically represented in Fig. 3.7.

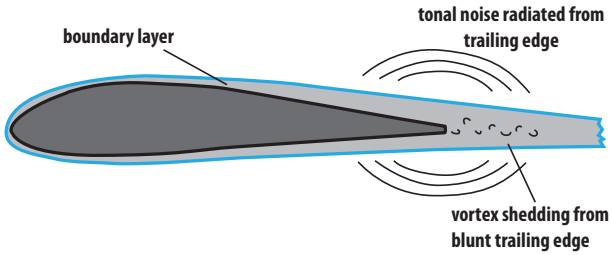


Figure 3.6: Representation of the TEB-VS noise mechanism.

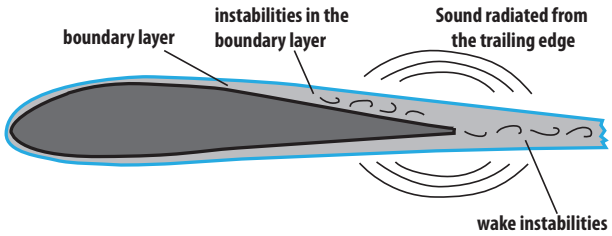


Figure 3.7: Representation of the LBL-VS noise mechanism.

Tip Vortex Formation Noise

Tip vortex formation noise is the noise generated, according to Brooks et al. (1989), due to the interaction between the vortexes shed from the tip of the blade and the tip surface, in a way analogous to the TBL-TE noise mechanism. The vortexes are shed due to the difference in pressure between the suction and pressure sides of the blade. This noise is of broadband nature, and its level is strongly dependent on the geometry of the tip. This mechanism is schematically represented in Fig. 3.8.

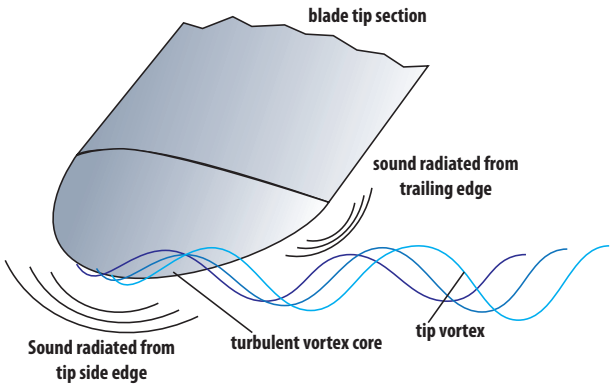


Figure 3.8: Representation of the tip vortex formation noise mechanism.

3.3 Noise Prediction Model

The wind turbine noise prediction method developed in this work consists in dividing the turbine blade into segments, much like what is done in the BEM theory for aerodynamic prediction of a wind turbine. This way, each segment is considered to have a certain airfoil shape and span length. The noise radiation process for any blade section can, according to Lowson (1993), be assumed as identical to that for an equivalent airfoil section. Therefore, the sound pressure level can be computed at each blade noise

element and the total sound power level generated by the rotor is the result of a summation of the noise from each blade element,

$$L^j_{p,total} = 10\log_{10}\left(\frac{N_B}{N_{az}}\sum_i 10^{\frac{L^j_{p,i}}{10}}\right), \tag{3.6}$$

where N_B is the number of blades, N_{az} is the number of azimuthal positions where the blade is computed and $L^j_{p,i}$ is the total sound pressure level generated by the i^{th} blade noise element at frequency band j.

The total sound levels at each frequency band can also be summed,

$$L_{p,overall} = 10\log_{10}\left(\sum_j 10^{\frac{L^j_{p,total}}{10}}\right), \tag{3.7}$$

thus leading to the Overall Sound Pressure Level (OASPL).

The model is limited to the rotor, since it is the object of interest. Therefore, noise contributions from the interaction with the tower and nacelle are not taken into account, nor are mechanical noises considered.

3.3.1 Prediction Model for Inflow Turbulence Noise

Based on the work of Amiet (1975) on experimental airfoil measurements, Lowson (1993) formulated an empirical relation for the inflow turbulence noise. It models both the low and high frequency behavior of this noise mechanism, with the total noise being a summation of the high and low frequency terms, as follows

$$L_{p,\text{inflow}} = L^H_{p,\text{inf}} + 10\log_{10}\frac{K_c}{1+K_c}. \tag{3.8}$$

The sound pressure level for high frequencies is defined as

$$L^H_{p,\text{inf}} = 10\log_{10}\left(\frac{\rho^2 c_0^2 L}{2r_e^2}M^3u^2L_{turb}I^2_{turb}\frac{\bar{k}^3}{\left(1+\bar{k}^2\right)^{7/3}}\bar{D}_h\right) + 58.4 \tag{3.9}$$

where L_{turb} is the turbulence length scale, I_{turb} is the turbulence intensity, u is the mean wind speed, L is the span of the airfoil section, r_e is the effective observer distance, $\bar{D}_h$ is a directivity function (see subsection 3.3.6), $\bar{k} = \pi f c/U$ is the local wave number, c is the local chord length and U is the local velocity over the airfoil section.

The low frequency correction factor K_c in Eq. (3.8) is defined as

$$K_c = 10S^2M\frac{\bar{k}^2}{\beta^2}, \tag{3.10}$$

where $\beta^2 = 1 - M^2$ and S is the compressible Sears function defined, as suggested by Amiet, as

$$S^2 = \left(\frac{2\pi\bar{k}}{\beta^2} + \frac{1}{1+2.4\bar{k}/\beta^2}\right)^{-1} \tag{3.11}$$

In Moriarty et al. (2004), the model of Amiet is preferred, due to the opinion of the researchers that it

is more reliable. In Amiet's model, Eq. (3.8) is replaced by

$$L_{p,\mathrm{inf}}^{H} = 10\log_{10}\left(\frac{\rho^2 c_0^4 L}{2r_e^2} M^5 L_{turb} I_{turb}^2 \frac{\hat{k}^3}{\left(1+\hat{k}^2\right)^{7/3}} \bar{D}_h\right) + 78.4 \tag{3.12}$$

where $\hat{k} = k/k_e$ is the wave number $k = 2\pi f/U$ normalized by the wave number range of energy-containing eddies $k_e = 3/(4L_{turb})$. Moriarty et al. also propose the use of a correction factor which corrects the minor effect of the angle of attack on the turbulent inflow noise. The new correction factor is given by

$$K_c = 10S^2 M \frac{\bar{k}^2}{\beta^2}\left(1-\alpha^2\right), \tag{3.13}$$

with α being the angle of attack given in radians.

Prediction of Turbulence Parameters

Prediction of the turbulence characteristics is of great importance to this model. The intensity of the turbulence and its length scale are dependent on the evaluation height (relative to the ground), the roughness of the ground and on the meteorological conditions at the evaluated site. Some examples of typical roughness lengths and their associated terrain types are presented on Tab. 3.1.

The mean wind speed also varies with height, and can be described with the power law relationship

$$V_z = V_{ref}\left(\frac{z}{H_{ref}}\right)^{\gamma}, \tag{3.14}$$

where z is the height, H_{ref} is the reference height at which the velocity was measured, and γ is the power law factor. This power law factor is a function of the surface roughness length z_0 and its estimate is given by Counihan (1975) as

$$\gamma = 0.24 + 0.096\log_{10} z_0 + 0.016(\log_{10} z_0)^2. \tag{3.15}$$

Turbulence intensity can be obtained using the relationship given by Snyder (1981), which is a function of the height z, the ground roughness z_0 and the power law factor γ,

$$I_{turb} = \gamma \frac{\ln(30/z_0)}{\ln(z/z_0)}. \tag{3.16}$$

The turbulence length scale is formulated as

$$L_{turb} = 25 z^{0.35} z_0^{-0.063}. \tag{3.17}$$

Correction to the Model

Being based on the flow over a flat plate, one of the main limitations of this prediction model is that it does not account for the geometry of the airfoil. In their work, Moriarty et al. (2004) present a very complete

Type of terrain	Roughness length (m)
Water, snow or sand surfaces	0.0001
Open, flat land, mown grass, bare soil	0.01
Farmland with some vegetation (reference)	0.05
Suburbs, towns, forests, many trees and bushes	0.3

Table 3.1: Typical roughness lengths associated with different terrain types.

method which relies on the boundary-element method created by Guidati et al. (1997) to perform the computation of the mean flow and the interaction of sound waves with the solid airfoil surface, thus being able to predict with a higher level of accuracy the differences in sound pressure level resulting from different airfoil geometries. This model, however, is very computationally demanding and, although it is considerably faster than any Computational AeroAcoustics (CAA) code, is still not fast enough to be used in a wind turbine noise design code. With this in mind, Moriarty et al. (2005) derived a simplified model of the boundary-element by reducing it to a simpler linear relation of geometric dimensions of the airfoil. This simplified model, instead of predicting the absolute sound pressure level for a certain geometry, predicts the difference to the flat plate theory. By coupling this model with the previously presented model for the flat plate, the absolute sound power level for different airfoil geometries can be predicted.

The complete equation for the difference in sound pressure level between an airfoil and a flat plate is

$$\Delta L_p = -\left(1.123\left(D_{rel,1\%} + D_{rel,10\%}\right) + 5.317\left(D_{rel,1\%} + D_{rel,10\%}\right)^2\right)\left(\frac{2\pi f c}{U} + 5\right), \tag{3.18}$$

where $D_{rel,1\%}$ and $D_{rel,10\%}$ are the relative thicknesses at 1% and 10% chord, respectively. The total inflow turbulence noise for a certain airfoil is then calculated by summing the results from the two models,

$$L_{p,\text{airfoil}} = \Delta L_p + L_{p,\text{flat plate}} + 10, \tag{3.19}$$

where 10 is a fudge factor to match with NLR data (Moriarty et al., 2004). According to Moriarty et al. (2005), the model presents good accuracy for Strouhal numbers up to 75, therefore the correction is only applied up to that value.

3.3.2 Prediction Model for Turbulent Boundary Layer - Trailing Edge Noise

The prediction model for the 1/3-octave spectrum of the turbulent boundary layer trailing edge noise of the pressure and suction sides and separation stall noise was developed by Brooks et al. (1989), based on the analysis performed by Ffowes-Williams and Hall (1970). The total TBL-TE noise spectrum is predicted from the sum of the suction side, pressure side, and separation-stall contributions,

$$L_{p,\text{TBL-TE}} = 10\log\left(10^{L_\alpha/10} + 10^{L_s/10} + 10^{L_p/10}\right), \tag{3.20}$$

where the subscripts p and s denote the pressure and suction sides of the airfoil.

The pressure side contribution to this noise is given by

$$L_{p,p} = 10\log\left(\frac{\delta_p^* M^5 L\bar{D}_h}{r_e^2}\right) + A\left(\frac{\mathrm{St}_p}{\mathrm{St}_1}\right) + (K_1 - 3) + \Delta K_1, \tag{3.21}$$

the suction side by

$$L_{p,s} = 10\log\left(\frac{\delta_s^* M^5 L\bar{D}_h}{r_e^2}\right) + A\left(\frac{\mathrm{St}_s}{\mathrm{St}_1}\right) + (K_1 - 3)\,, \tag{3.22}$$

and the separation-stall contribution by

$$L_{p,\alpha} = 10\log\left(\frac{\delta_s^* M^5 L\bar{D}_h}{r_e^2}\right) + B\left(\frac{\mathrm{St}_s}{\mathrm{St}_2}\right) + K_2, \tag{3.23}$$

where δ^* is the boundary layer displacement thickness and A and B are empirical spectral shape functions based on the Strouhal number, defined in Appendix B.

For angles of attack higher than 12.5° , which is the stall angle for the NACA 0012 airfoil, the TBL-TE noise source is mainly the separation-stall noise and equations (3.21) to (3.23) are replaced by

$$L_{p,p} = -\infty, \tag{3.24}$$

$$L_{p,s} = -\infty, \tag{3.25}$$

and

$$L_{p,\alpha} = 10\log\left(\frac{\delta_s^* M^5 L\bar{D}_\ell}{r_e^2}\right) + A'\left(\frac{\mathrm{St}_s}{\mathrm{St}_2}\right) + K_2, \tag{3.26}$$

where A' is the shape function A for a Reynolds number three times the actual Reynolds number.

The Strouhal number definitions are

$$\mathrm{St}_p = \frac{f\delta_p^*}{U}, \tag{3.27}$$

$$\mathrm{St}_s = \frac{f\delta_s^*}{U}, \tag{3.28}$$

$$\mathrm{St}_1 = 0.02M^{-0.6}, \tag{3.29}$$

and

$$\mathrm{St}_2 = \mathrm{St}_1 \times \begin{cases} 1 & (\alpha_* < 1.33) \\ 10^{0.0054(\alpha_* - 1.33)^2} & (1.33 \leqslant \alpha_* \leqslant 12.5) \\ 4.72 & (12.5 < \alpha_*) \end{cases} \tag{3.30}$$

where f is the frequency and U is the local mean velocity.

3.3.3 Prediction Model for the Laminar Boundary Layer - Vortex Shedding Noise

The scaling approach taken in (Brooks et al., 1989) for the LBL-VS noise is similar to that taken for the TBL-TE noise with the difference that, instead of being a function of the boundary layer displacement thickness (among other parameters), the LBL-VS prediction model is a function of the boundary layer

thickness δ_p at the pressure side. The noise spectrum in a 1/3 octave presentation is predicted by

$$L_{\text{LBL-VS}} = 10\log\left(\frac{\delta_p M^5 L\bar{D}_h}{r_e^2}\right) + G_1\left(\frac{\mathrm{St}'}{\mathrm{St}'_{\text{peak}}}\right) + G_2\left[\frac{R_c}{(R_c)_0}\right] + G_3\left(\alpha_*\right), \tag{3.31}$$

where G_1, G_2 and G_3 are spectral shape functions, defined in Appendix B.

The Strouhal definitions are

$$\mathrm{St}' = \frac{f\delta_p}{U}, \tag{3.32}$$

and

$$\mathrm{St}'_{\text{peak}} = \mathrm{St}'_1 \times 10^{-0.04\alpha_*}, \tag{3.33}$$

where

$$\mathrm{St}'_1 = \begin{cases} 0.18 & \left(R_c \leqslant 1.3\times10^5\right) \\ 0.001756R_c^{0.3931} & \left(1.3\times10^5 \leqslant R_c \leqslant 4.0\times10^5\right) \\ 0.28 & \left(4.0\times10^5 < R_c\right) \end{cases}. \tag{3.34}$$

3.3.4 Prediction Model for Trailing Edge Bluntness Vortex Shedding Noise

Based on the same scaling method as the TBL-TE and LBL-VS noise, the prediction model for the TEB-VS noise derives from an experiment by Brooks and Hodgson (1981). Its spectrum in a 1/3-octave presentation is predicted by

$$L_{\text{BLUNT}} = 10\log\left(\frac{hM^{5.5}L\bar{D}_h}{r_e^2}\right) + G_4\left(\frac{h}{\delta^*_{avg}}, \Psi\right) + G_5\left(\frac{h}{\delta^*_{avg}}, \Psi, \frac{\mathrm{S}\,t'''}{\mathrm{S}\,t'''_{\text{peak}}}\right), \tag{3.35}$$

where h is the bluntness thickness and $\delta^*_{avg} = (\delta^*_p + \delta^*_s)/2$ is the average boundary layer displacement thickness.

The Strouhal number here is defined with the bluntness thickness instead of a boundary layer thickness parameter.

$$\mathrm{S}\,t''' = \frac{fh}{U}. \tag{3.36}$$

The peak Strouhal number is defined as a function of the thickness ratio $h/\delta*$ and the trailing edge angle Ψ,

$$\mathrm{S}\,t'''_{\text{peak}} = \begin{cases} \dfrac{0.212 - 0.0045\Psi}{1 + 0.235\left(h/\delta^*_{avg}\right)^{-1} - 0.0132\left(h/\delta^*_{avg}\right)^{-2}} & \left(0.2 \leqslant h/\delta^*_{avg}\right) \\ 0.1\left(h/\delta^*_{avg}\right) + 0.095 - 0.00243\Psi & \left(h/\delta^*_{avg} < 0.2\right) \end{cases}. \tag{3.37}$$

The function G_4 determines the peak level of the spectrum and is given by

$$G_4\left(h/\delta^*_{avg}, \Psi\right) = \begin{cases} 17.5\log\left(h/\delta^*_{avg}\right) + 157.5 - 1.114\Psi & \left(h/\delta^*_{avg} \leqslant 5\right) \\ 169.7 - 1.114\Psi & \left(5 < h/\delta^*_{avg}\right) \end{cases}, \tag{3.38}$$

while the spectral curve fitting function G_5 is predicted as an interpolation between $\Psi = 14^\circ$ and $\Psi = 0^\circ$,

$$G_5\left(\frac{h}{\delta^*_{avg}}, \Psi, \frac{\mathrm{St'''}}{\mathrm{St'''_{peak}}}\right) = (G_5)_{\Psi=0^\circ} + 0.0714\Psi\left[(G_5)_{\Psi=14^\circ} - (G_5)_{\Psi=0^\circ}\right]. \tag{3.39}$$

The definition of functions $(G_5)_{\Psi=0^\circ}$ and $(G_5)_{\Psi=14^\circ}$ is presented in Appendix B.

3.3.5 Prediction Model for Tip Vortex Formation Noise

The prediction for the tip noise is, as given by Brooks et al. (1989),

$$L_{\mathsf{TIP}} = 10\log\left(\frac{M^2 M_{\max}^3 \ell^2 \bar{D}_h}{r_e^2}\right) - 30.5\big(\log \mathrm{St''} + 0.3\big)^2 + 126, \tag{3.40}$$

where ℓ is the spanwise extent of separation due to tip vortex at the trailing edge.

The Strouhal number is

$$\mathrm{St''} = \frac{f\ell}{U_{\max}}. \tag{3.41}$$

For the case of a rounded tip, ℓ is given by

$$\ell/c \approx 0.008\alpha_{\mathsf{TIP}}, \tag{3.42}$$

where α_{TIP} is the angle of attack of the tip region to the oncoming flow. For the case of a flat tip,

$$\ell/c = \begin{cases} 0.0230 + 0.0169\alpha'_{\mathsf{TIP}} & (0^\circ \leqslant \alpha'_{\mathsf{TIP}} \leqslant 2^\circ) \\ 0.0378 + 0.0095\alpha'_{\mathsf{TIP}} & (2^\circ < \alpha'_{\mathsf{TIP}}) \end{cases}, \tag{3.43}$$

where α'_{TIP} a redefined angle of attack given by

$$\alpha'_{\mathsf{TIP}} = \left[\left(\frac{\partial L'/\partial y}{(\partial L'/\partial y)_{\mathrm{ref}}}\right)_{y\to\mathsf{TIP}}\right]\alpha_{\mathsf{TIP}}. \tag{3.44}$$

The maximum Mach number $M_{\max}$ of the flow within or around the separated flow region at the trailing edge is

$$M_{\max}/M = (1 + 0.036\alpha_{\mathsf{TIP}}), \tag{3.45}$$

where M is the Mach number of the oncoming flow to the airfoil tip region and the velocity corresponding to $M_{\max}$ is

$$U_{\max} = c_0 M_{\max}. \tag{3.46}$$

3.3.6 Sound Directivity

The directivity functions for both high and low frequencies, presented by Brooks et al. (1989), are based on the research on the work of Amiet (1975). They are presented normalized by the trailing edge noise emitted in the $\Theta_e = 90^\circ$ and $\Phi_e = 90^\circ$ direction (see Fig. 3.9), meaning that $\bar{D}_h\,(90^\circ, 90^\circ) = 1$, where

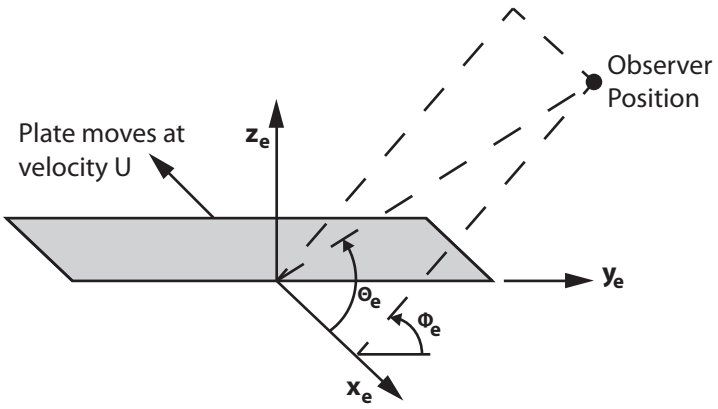


Figure 3.9: Angles for sound directivity computation.

the sound directivity reaches its maximum.

For high frequencies the directivity is given by

$$\bar{D}_h\left(\Theta_e, \Phi_e\right) \approx \frac{2\sin^2\left(\Theta_e/2\right)\sin^2\Phi_e}{\left(1+M\cos\Theta_e\right)\left[1+\left(M-M_c\right)\cos\Theta_e\right]^2}, \tag{3.47}$$

where $M_c \approx 0.8M$. For low frequencies the directivity function is as

$$\bar{D}_\ell\left(\Theta_e, \Phi_e\right) \approx \frac{\sin^2\Theta_e \sin^2\Phi_e}{\left(1+M\cos\Theta_e\right)^4}. \tag{3.48}$$

For the case of an airfoil fixed in space, relative to the observer, the calculation of the directivity functions is rather simple, as the angles and distance are easily computed. When predicting the noise on a wind turbine, however, as the blade is rotating, and the observer does not follow it, it is necessary to perform a series calculations to obtain the geometric parameters Φ_e, Θ_e and r_e.

A set of coordinate systems need to be defined, as shown in Fig. 3.10a. The first coordinate system (system 1) is placed at the base of the tower. System 2 is non-rotating and placed at the nacelle, system 3 is fixed to the rotating shaft and system 4 is aligned with one of the blades. There is also a fifth system which is system 4 rotated around its x axis. Appendix A describes the transformation matrices between the various systems.

With the coordinate systems defined, a vector $\mathbf{r}$ can be obtained by a vector summation as shown in Fig. 3.10b,

$$\mathbf{r} = \mathbf{r_1} + \mathbf{r_2} + \mathbf{r_3}. \tag{3.49}$$

The sound directivity angles Φ_e, Θ_e can be found by projecting the vector $\mathbf{r}$ in two directions, according to Fig. 3.9.

3.3.7 Boundary-Layer Calculation

The model of Brooks et al. (1989) uses boundary-layer thickness parameters obtained from a prediction method based on experimental results from the NACA 0012 airfoil. This method, while producing reasonable results for that airfoil, is not suitable for generalization to other airfoils (particularly cambered

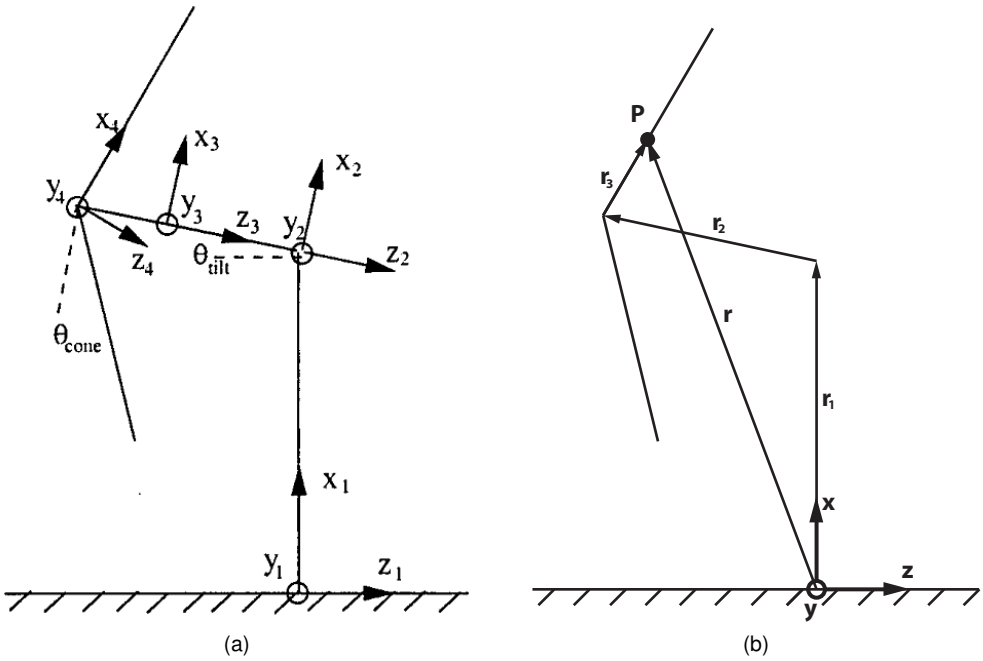


Figure 3.10: Wind turbine coordinate systems.

ones). For this reason, in the present airfoil noise prediction method the boundary-layer parameters can also be calculated using an external panel method code such as XFOIL (Drela, 1989) or RFOIL (Montgomerie et al., 1997), similarly to what is done in the TNO model (Parchen, 1998) or the SILANT code (Boorsma and Schepers, 2012).

When XFOIL is used, one extra computation needs to be performed, since the code does not compute the boundary layer thickness δ. This parameter can be obtained from the relation given by Drela and Giles (1987),

$$\delta = \theta \left(3.15 + \frac{1.72}{H_k - 1} \right) + \delta^*, \tag{3.50}$$

where θ is the boundary-layer momentum thickness and the definition of H_k is that derived by Whitfield (1978) for adiabatic flow in air,

$$H_k = \frac{H - 0.290M^2}{1 + 0.113M^2}, \tag{3.51}$$

where H is the boundary layer shape factor.

Chapter 4

Blade Geometry and Parameterization

In order to analyze a wind turbine, it is necessary to model the geometry of its blades. This model should have as few input parameters as possible, since the objective of this work is to perform optimization of the blade geometry. In this work, the geometry is defined by setting the radial distribution of three independent parameters: twist, chord and airfoil shape. With these distributions, the blade geometry is generated by the following steps:

1. Connect the airfoil shapes to form the blade basis, with unit chord;
2. Apply the chord distribution to the basis blade, scaling each spanwise station by the local chord value;
3. Apply the twist distribution, rotating each section about its aerodynamic center (usually at $1/4$ chord).

Using this method, the geometry can be easily introduced into the BEM method (where the twist, chord and airfoil aerodynamic data are separate inputs) or used in a more complex CFD computation, where, in order to generate the mesh, the full 3D geometry is required. The fact that they are independent of each other also allows for their separate optimization.

4.1 Twist and Chord distribution

Both the twist and chord distributions are defined by a set of control points which are either linearly interpolated or used to define a Bézier curve. The degree of the Bézier curve will depend on the number of control points. Figure 4.1 shows an example of a twist and chord distribution using Bézier curves.

4.2 Airfoil Shape Distribution

The unitary chord, non-twisted shape of the blade is defined by linear interpolation of control cross-sections airfoils along the blade, which can be defined by their coordinates. As the intent of this work is to perform optimizations on the blade geometry, this approach is not acceptable, leading to the need to represent the cross-sections shapes using a smaller number of parameters.

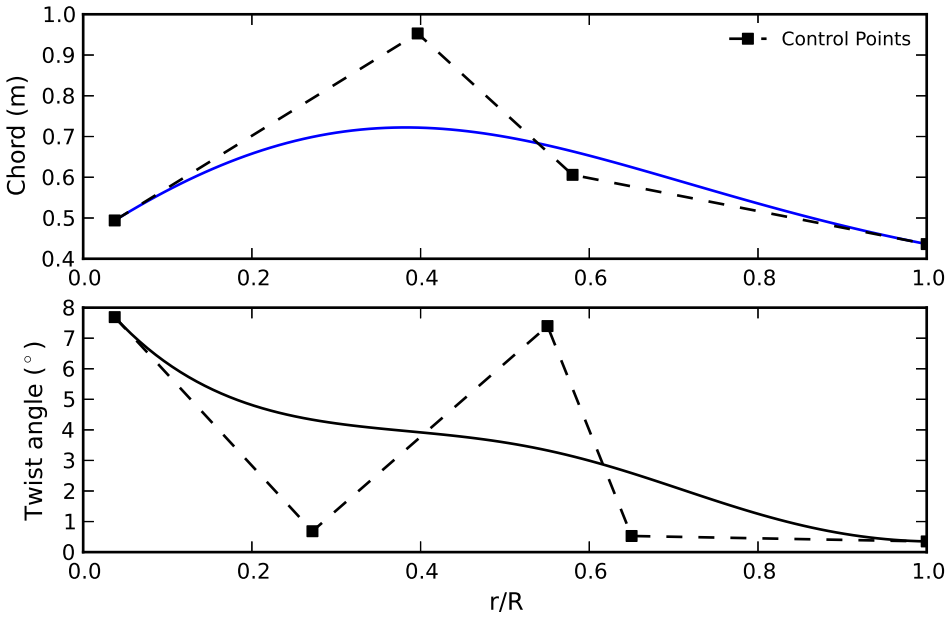


Figure 4.1: Twist and chord distribution using Bézier curves.

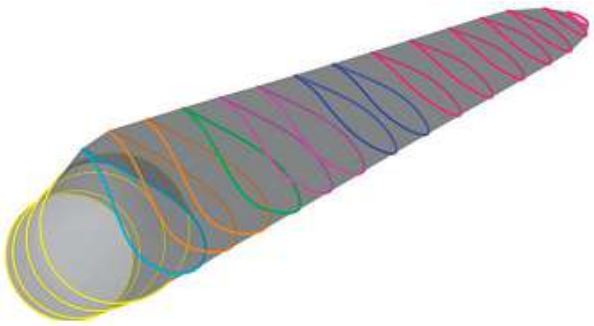


Figure 4.2: Airfoil sections superposed on the wind turbine blade (Bazilevs et al., 2011).

There are numerous methods to represent an airfoil shape. Grasso (2008) presents some of them (Hicks-Henne Functions, Legendre Function, Spline curves and Bézier curves) and discusses their advantages and disadvantages. A much more recent approach is the use of Non-Uniform Rational B-Spline (NURBS) curves and surfaces to represent airfoils and wing-like surfaces (Bentamy et al., 2002). Ivanović et al. (2009) performed an optimization of helicopter blades using NURBS for the parameterization and Bazilevs et al. (2011) also uses NURBS to define the geometry of a wind turbine blade.

A comparison between some parameterization methods is presented in Tab. 4.1, and after an evaluation of their advantages and disadvantages, the NURBS representation was chosen. A detailed explanation of the method follows.

Parameterization	Advantages	Disadvantages
Hicks-Henne	Few parameters	Harmonic expressions Not easy user usage
3rd degree splines	Polinomial expressions	Necessity of segmentation for accurate representation
3rd degree Bézier curves	Polinomial expressions; Direct connection between paremeters and geometry; Easy inflection points controlalbility; Easy user usage; Approximant formulation.	Necessity of segmentation for accurate representation
NURBS curves	All of the advantages of Bézier curves; Local approximation.	

Table 4.1: Comparison between various airfoil parameterization methods (based on Grasso (2008)).

NURBS Curves Properties

A Non-Uniform Rational B-Spline curve is a vector-valued piecewise polynomial function of the form (Piegl and Tiller, 1997)

$$\mathbf{C}(u) = \sum_{i=1}^{n} R_{i,p}(u)\,\mathbf{P}_i, \tag{4.1}$$

where $\mathbf{P}_i$ are the control points and $R_{i,p}$ are *rational basis functions* defined as

$$R_{i,p}(u) = \frac{N_{i,p}(u)\,w_i}{\sum_{i=1}^{n} N_{i,p}(u)\,w_i}, \tag{4.2}$$

with w_i being the weights, and $N_{i,p}$ the normalized B-spline basis functions functions of order p. These basis functions can be defined recursively as

$$N_{i,0}(u) = \begin{cases} 1 & u_i \leqslant u < u_{i+1} \\ 0 & \text{elsewhere} \end{cases} \tag{4.3}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u), \tag{4.4}$$

where u_i are the so called knots forming a knot vector $\mathbf{U} = \{u_0, u_1, ..., u_m\}$.

The degree p, number of knots $m+1$, and number of control points $n+1$ are related by $m = n+p+1$. If the knot vector takes the form $\mathbf{U} = \{\alpha, ..., \alpha, u_{p+1}, ..., u_{m-p-1}, \beta, ..., \beta\}$ with the end knots α and β repeated with multiplicity $p+1$, the NURBS curve is a Bézier-like curve, interpolating the endpoints. It is usually assumed α and β to be 0 and 1, respectively.

As a consequence of the analytical properties of the rational basis functions $R_{i,p}$, a NURBS curve will exhibit the following geometric characteristics (Piegl and Tiller, 1997):

- $\mathbf{C}(0) = \mathbf{P}_0$ and $\mathbf{C}(1) = \mathbf{P}_1$;
- Bézier and nonrational B-spline curves are special cases;
- Local approximation: If a control point is moved or a weight is changed, it will affect the curve only in $p + 1$ knot spans;
- Strong convex hull property: if $u \notin [u_i, u_{i+1})$, the $\mathbf{C}(u)$ lies within the convex hull of $\mathbf{P}_{i-p}, ..., \mathbf{P}_i$

The last property is particularly interesting in airfoil design problems as it guarantees that there will be no inflection points in a curve as long as the domain is simply connected. This means that if inflection points are not wanted, they can be avoided a priori, eliminating the necessity of performing checks to verify their existence.

Airfoil shape parameterization using NURBS curves

A parameterization using two NURBS curves was chosen, one for the upper and other for the lower side of the airfoil. Each curve is defined by 6 control points, a knot vector $\mathbf{U} = \left\{0, 0, 0, 0, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, 1, 1, 1, 1\right\}$ and unitary weights (see Fig. 4.3). The leading edge control points are set to $(0, 0)$ and the trailing edge control points to $(1, y_i)$. In order to have a smooth geometry, and guarantee continuity of the curve and its derivatives, the two control points next to the leading edge have their x coordinate set to $x = 0$. With this parameterization, the maximum number of variables that can be used to modify the airfoil shape is 20 (see Fig. 4.3, where the control points of the NURBS parameterization are presented, along with their respective free degrees of freedom).

It should be noted, however, that this parameterization allows the use of any number of control points and different knot vectors, with the only limitation that the knot vectors and number of control are the same at all control sections of the blade. Another important observation is that if a knot vector $\mathbf{U} = \left\{0, 0, 0, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 1, 1, 1, 1\right\}$ is chosen for each of the NURBS curve, due to the fact that Bézier curves are special cases of NURBS curves, the parameterization becomes the one defined by Grasso (2008).

In order to verify if this representation method would be able to reproduce the various airfoils that are used in wind turbines, a tool was created to fit the NURBS airfoil to a given set of coordinates. It performs a minimization of the following objective function:

$$\sum_{i=0}^{N} (u_i - u'_i)^2 + (y_i - y'_i)^2 \tag{4.5}$$

where u' and y' are the given coordinates and u and y the coordinates of the points given by the NURBS parameterization.

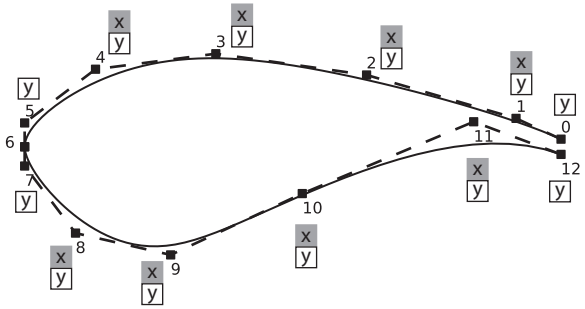
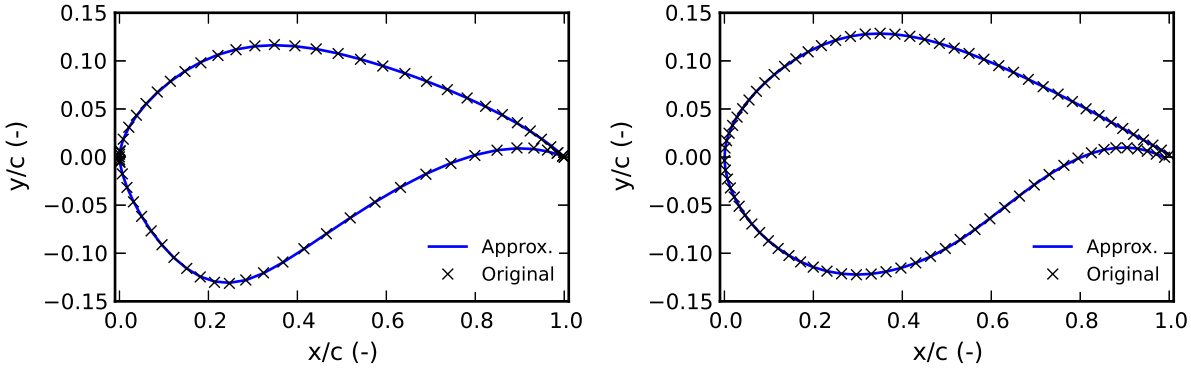


Figure 4.3: Generic airfoil representation using two NURBS curves and its control points.

In Fig. 4.4 some examples are shown for some popular WT airfoils. It can be seen that the NURBS curves can accurately represent the desired airfoil shapes.



(a) NREL S814

(b) DU 91-W2-250

(c) NACA 63220

(d) NREL S820

Figure 4.4: Approximation of some airfoil shapes using the NURBS curves parameterization.

Chapter 5

Wind Turbine Noise Prediction Code

A wind turbine performance and noise prediction code was developed using the models introduced in the previous chapters. The code was written in Python and C++, taking both advantage of the speed of the C++ language and the flexibility of the Python language. In this chapter, the code structure is described, together with an explanation of the design choices and approaches taken in the development. In the end of the chapter, the code is validated against experimental data and data obtained using another simulation tool.

5.1 Code Structure

The code was developed making use of the object-oriented programming capabilities of both the Python and C++ languages, thus allowing to develop a robust and modular code. This approach allows the possibility to include different aerodynamic or aeroacoustic prediction models without much effort. The interface between the two languages is done using SWIG (Beazley, 1996), a software development tool that connects programs written in C and C++ with a variety of high-level programming languages. By using this interface, the need of the use of files for transferring data between the codes was eliminated.

The code can be divided into three parts, the geometry definition classes (`Rotor`, `Blade` and `Section`), the rotor analysis class (`Analysis`) and the airfoil aerodynamic analysis classes `Polar` and `BLayer`. The `Analysis` class contains two functions, the aerodynamic and the aeroacoustic analysis functions, which make use of the `BEM` and the `NOISE` C++ libraries, respectively, as well of the `Polar` and `BLayer` classes. A non extensive description of some of the data and functions of each class is given in Tables 5.1 to 5.6 and the interaction between the classes of the code is schematically represented in Fig. 5.1.

5.2 Code Inputs and Output

In this section, the inputs and output of the code are described regarding the geometry of the blade, the analysis and the aerodynamic polars. Each of the previous can be defined via an input file, but it should be noted that they can also be defined with a Python script file.

SECTION	Represents a cross-sectional shape of the blade.	
Data	*params*:	Parameters that define the airfoil shape. Can be either coordinates, a filename of a set of parameters defining the NURBS representation of the airfoil shape.
	ptype:	String declaring the type of parameters which the section uses.
Functions	*pnts*:	Evaluates the section and returns the coordinates of its points. Takes as input the number of points, scaling factor, angle of rotation to be applied to the airfoil and point distribution type.
	thick:	Returns thickness at 1 and 10% chord
	trail:	Returns trailing edge thickness and angle
	plot:	Plots the section shape.

Table 5.1: Listing of some of the features of the `Section` class.

BLADE	Represents a wind turbine blade.	
Data	*maxR* / *minR*	blade tip radius and hub start radius.
	Control Sections	Control sections (`Section` class) and their radial position.
	Chord and Twist distributions	Chord and Twist control points values and their radial position.
	Polars	Polars (`Polar` class) and their radial position
Functions	*add (Twist / Chord / Section)*	Set of functions to insert Twist, Chord control points or Control Sections.
	(twist / chord / sect)At	Evaluates twist, chord or control sections at certain radial position and returns the chord or twist values or, if the case a `Section` linearly interpolated from the control sections.
	pnts3D	Returns the coordinates of the points of a cross section of the blade at a certain radial position.
	discretize	Discretizes the blade with the required number of elements and element distribution type.
	createPolars	Creates a determinate number of polars at certain radial positions.

Table 5.2: List of some of the features of the `Blade` class.

ROTOR	Represents a wind turbine rotor.	
Data	*Blade*:	A `Blade` class representing the geometry of the blades of the rotor.
	hubheight:	Height of the hub.
	nblades:	Number of blades.
	pitch:	Set pitch angle.
Functions	*load*:	loads a rotor file and creates the `Blade` class.
	save:	Saves the rotor, blade and control sections geometries to files

Table 5.3: Listing of some of the features of the `Rotor` class.

POLAR	Represents airfoil polar (or set of polars).	
Data	*Original*:	Original (2D) polar data (α, C_l and C_d).
	Corrected:	Original polar data corrected for 3D effects.
	Extrapolated:	Extrapolated Original/Corrected polar data.
Functions	*createX*:	Creates a polar (or set of polars, for various Reynolds numbers) using XFOIL.
	createR:	Same as createX but using RFOIL.
	correct3D:	Applies 3D Stall-delay correction to all 2D polars.
	extrapolate:	Extrapolates original/corrected polars using Viterna's method.

Table 5.4: Listing of some of the features of the `Polar` class.

BLAYER	Represents the boundary layer parameters of an airfoil.	
Data	*Thickness*:	Boundary layer thickness.
	Disp. Thickness:	Boundary layer displacement thickness.
	Mom. Thickness:	Boundary layer momentum thickness.
Functions	getX:	Obtains the boundary layer parameters for a set of Reynolds numbers and angles of attack using XFOIL.
	getR:	Same as getX but using RFOIL.
	data:	Interpolates and returns the boundary layer parameters at given angle of attack and Reynolds number.

Table 5.5: Listing of some of the features of the `BLayer` class.

ANALYSIS	Class responsible for the aeroacoustic analysis of the rotor.	
Data	*Options*:	A set of parameters defining the analysis (see Appendix C)
Functions	*bem*:	Analyzes the `Rotor` given as input using the BEM model.
	noise:	Analyzes the `Rotor` with the aeroacoustic model using the results obtained from the *bem* function.

Table 5.6: Listing of some of the features of the `Analysis` class.

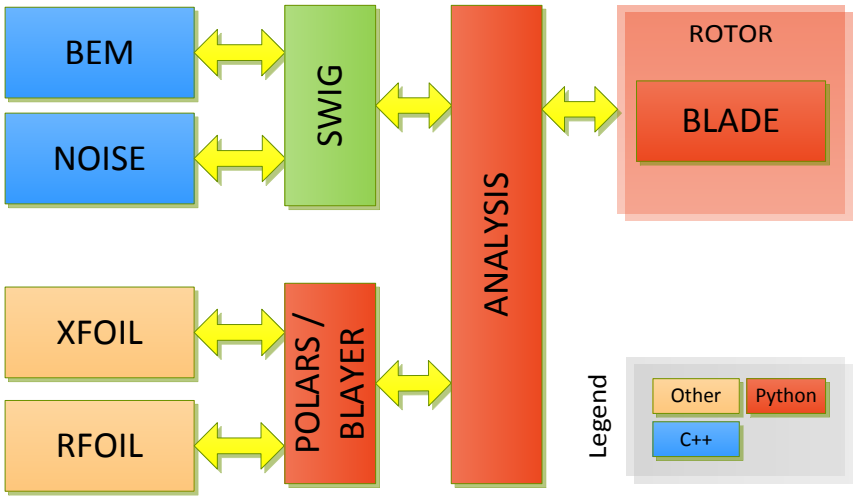


Figure 5.1: Structure of the developed code.

Sample input files of the analysis and rotor definitions are presented in Appendix C and an example of a script using the various classes in an optimization framework can be seen in Appendix D.

The code can output the various results from the BEM computation such as power curve, radial distribution of angle of attack, local Reynolds, relative velocity, local power, etc, in a Comma Separated Values (CSV) file. The noise spectrum of the computation of the rotor can also be output to a CSV file, as sound pressure levels or sound power levels, which can be A-weighted or not.

The blade geometry can be exported in various different formats besides the custom format previously mentioned, such as a Tecplot 360 compatible file or a DXF file.

5.3 Code Validation

Validation of a code is of great importance, as it confirms whether it produces an accurate representation of the real world from the perspective of the intended uses of the model or not. It allows to identify and quantify error and uncertainty through comparison of the simulation results with experimental data, or data obtained from sources *known to be accurate*.

In this section, the validation of the developed code is presented in three parts: the validation of the airfoil two-dimensional noise prediction, the validation of the wind turbine aerodynamic performance prediction and the validation of the the complete wind turbine noise prediction.

Parameter	Value	Parameter	Value
Chord Lenght (m)	0.3048	Turbulence Intensity (%)	15.7
Airfoil Span (m)	0.2286	Turbulence Length Scale (m)	69.8
Freestream Velocity ($m\,s^{-1}$)	71.3	Thickness @ 1% chord (-)	0.02
Angle of Attack ($^\circ$)	4.0	Thickness @ 10% chord (-)	0.12
Trailing Edge Thickness (m)	0.000 50	Observer distance (m)	1.22
Trailing edge solid angle ($^\circ$)	20.0	Angle relative to spanline ($^\circ$)	90.0
Speed of Sound ($m\,s^{-1}$)	340.46	Angle relative to chordline ($^\circ$)	90.0
Kinematic viscosity ($m^2\,s^{-1}$)	1.4529×10^{-5}	Air Density ($kg\,m^{-3}$)	1.225

Table 5.7: Parameters of the NACA 0012 noise prediction validation.

5.3.1 2D Airfoil Noise Prediction Validation

TBL-TE, LBL-VS and Bluntness noise

The TBL-TE, LBL-VS and Bluntness noise validation is performed on the NACA 0012 airfoil. Predictions are made using both the airfoil self-noise analysis tool NAFnoise (Moriarty, 2003) and the custom developed code. The untripped boundary layer thickness parameters were computed using the empirical method of BPM and the conditions at which the simulations were run are given in Tab. 5.7. The results of the simulations are presented in Fig. 5.2 and Fig. 5.3, where a it can be seen that the two predictions are virtually the same, and therefore validating the prediction of this noise mechanisms.

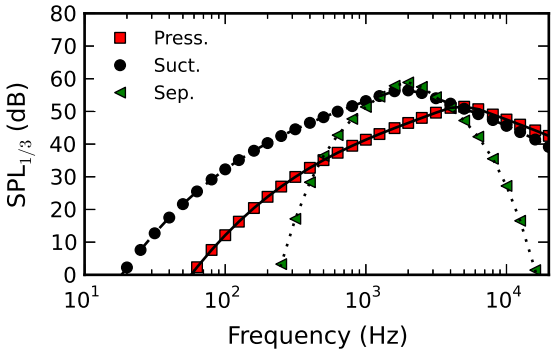


Figure 5.2: Predicted TBL-TE noise spectra for the NACA 0012 airfoil (markers: NAFnoise; lines: custom code).

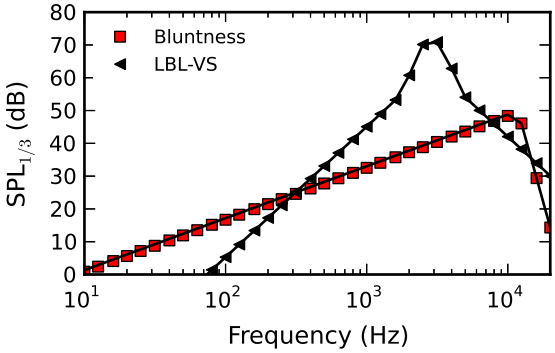


Figure 5.3: Predicted LBL-VS and Bluntness noise spectra for the NACA 0012 airfoil (markers: NAFnoise; lines: custom code).

Turbulent Inflow Noise

In order to validate the computation of the boundary layer parameters using XFOIL, together with the prediction models for the turbulent inflow noise, the NREL S822 airfoil noise prediction is compared with the experimental measurements data performed by Oerlemans and Migliore (2004). The turbulent inflow predictions and measured values at various Mach numbers, for an angle of attack of $\alpha = 4.4^\circ$ are presented in Fig. 5.4. Amiet's model is used together with the simplified version of Guidati's model. The predictions agree rather well with the experimental data.

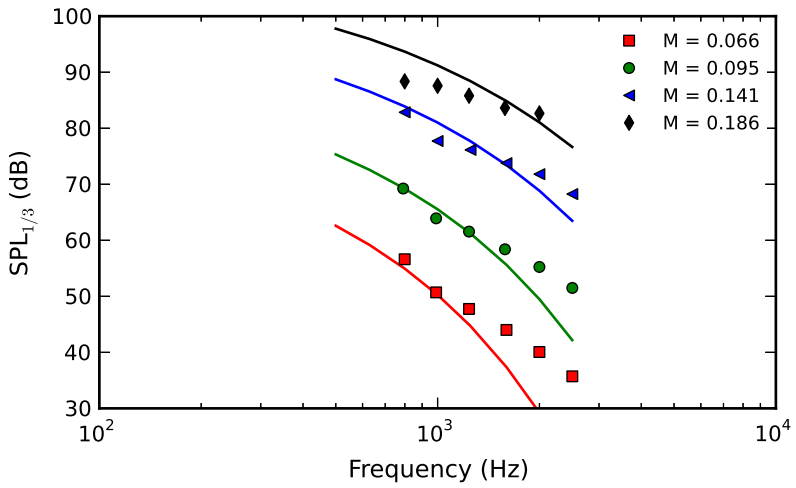


Figure 5.4: Turbulent inflow SPL prediction using simplified Guidati's model and measured data (Oerlemans and Migliore, 2004) for S822 at $\alpha = 4.4°$. (Prediction: solid curves; Exp. Data: Markers)

5.3.2 BEM Code Validation

In order to validate the BEM code, a comparison of its predictions is made against experimental data and the predictions of the Wt_Perf code (Buhl, 2012), for the NREL Phase II wind turbine.

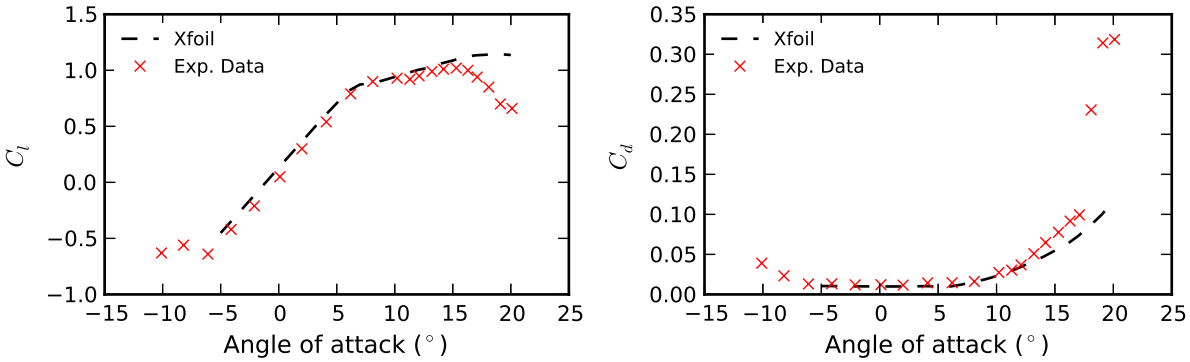
The Phase II is an untwisted and untapered wind turbine developed in United States of America's National Renewable Energy Laboratory (NREL) for testing purposes. The general characteristics of the turbine are explained in Tab. 5.8. It has a constant cross-section shape across the whole blade - the S809 airfoil. Experimental aerodynamic data of this airfoil is available from various sources and Fig. 5.5 shows the lift and drag coefficients from the wind tunnel experiments done at OSU and found in Schepers et al. (2001). The experimental data is compared with the the data obtained from XFOIL, using as input the coordinates from the NURBS representation of the airfoil.

Parameter	**Value**	**Parameter**	**Value**
Number of Blades (-)	3	Root extension (m)	0.723
Turbine Diameter (m)	10.06	Blade set angle (°)	12
Rotational Speed (rpm)	71.3	Twist (°)	None
Cut-in wind speed ($m\,s^{-1}$)	6	Chord (m)	0.4572
Rated power (kW)	19.8	Airfoil	S809

Table 5.8: NREL Phase II wind turbine characteristics.

In this validation case two simulations were performed. One with experimental data as input and another with the aerodynamic data obtained from XFOIL. The second case was compared with the results given by WT_Perf by the same aerodynamic data. All the simulations were run assuming an air density equal to 0.9793 kg/m^3 as suggested by Schepers et al. (2001).

Figure 5.6 shows the measured (Schepers et al., 2001) and predicted power curves for both the cases described. The custom code and Wt_Perf predicted rather well the aerodynamic power generated by the turbine. The power curve predicted using aerodynamic data from wind tunnel measurements, although



(a) Lift coefficient (b) Drag coefficient

Figure 5.5: Experimental and predicted aerodynamic data of the S809 airfoil (Re = 750,000).

being closer to the measured values, starts to drop at a wind speed of 17 $\mathrm{m\,s^{-1}}$. This emphasizes the sensitivity of the BEM code to the aerodynamic data that is used.

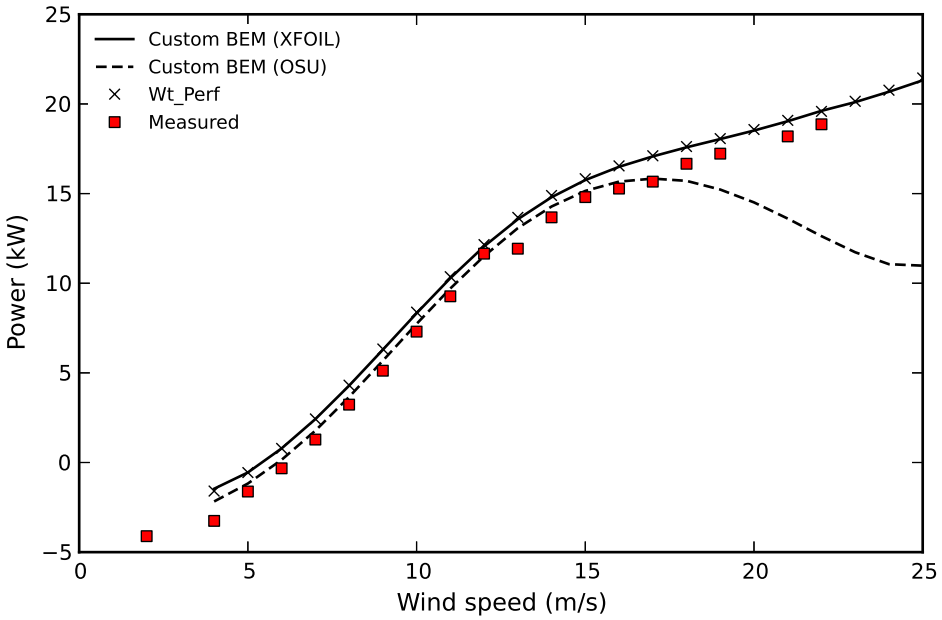
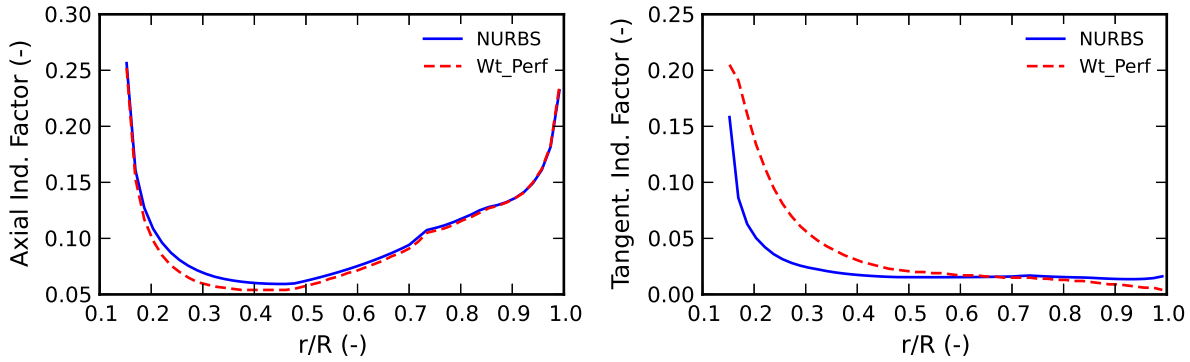


Figure 5.6: NREL Phase II measured and predicted power curve.

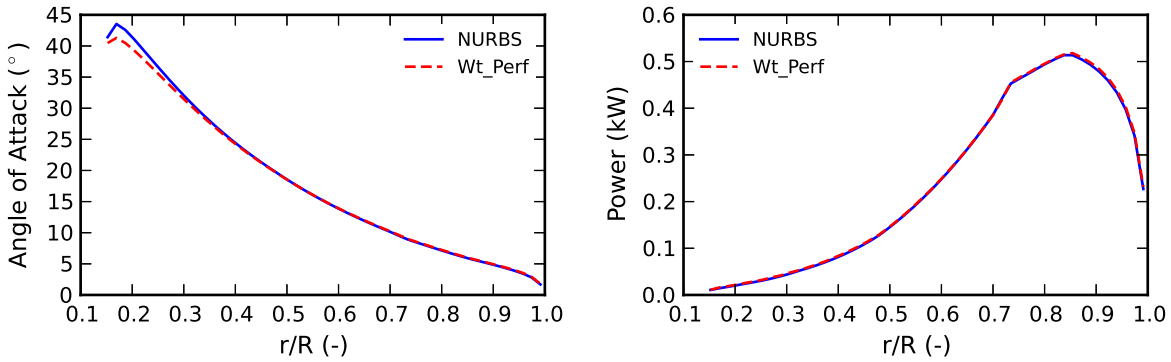
In Fig. 5.7 and Fig. 5.8, the radial distributions of some proprieties, computed both with the custom BEM code and the Wt_Perf code, are presented. Both results are in close agreement, with the exception of the tangential induction factor, which Wt_Perf predicts to be larger on the root and smaller at the tip region, when compared to the custom BEM code predictions.



(a) Axial Induction

(b) Tangential Induction

Figure 5.7: Radial distribution of axial and tangential induction factors on the NREL Phase II blade (U_0 = 12 m s^{-1}).



(a) Angle of Attack

(b) Local Power

Figure 5.8: Radial distribution of angle of attack and local power on the NREL Phase II blade ($U_0 = 12$ m/s).

5.3.3 WT Noise Code Validation

The validation of the full wind turbine noise prediction code is performed on the Atlantic Orient Corporation (AOC) 15/50 wind turbine (Seaforth Energy, 2010). It is a three-bladed wind turbine with a rated power of 50 kW. This wind turbine was chosen due to the availability of both performance (Jacobson et al., 2003) and noise (Huskey et al., 1999) data.

The blade uses the NREL S821, S819 and S820 profiles, defined at 40, 75 and 95 % of chord, respectively. The trailing edge thickness of the blade is considered to be 1% of the chord length, with an angle of 6 °. The general characteristics of the wind turbine are described in Tab. 5.9 and Fig. 5.9 shows the plot and twist radial distribution of the blade.

The predicted and measured power curves for this wind turbine are presented in Fig. 5.10. It shows a slightly constant overprediction of the power up to a wind speed of 14 m s^{-1}, after which the maximum power is slightly underpredicted. The loss of power due to stall is not predicted.

Parameter	Value	Parameter	Value
Number of Blades (-)	3	Root extension (m)	0.280
Turbine Diameter (m)	15.0	Blade set angle (°)	1.54
Rotational Speed (rpm)	64.4	Hub Height (m)	24.4
Cut-in wind speed ($m\,s^{-1}$)	6		
Rated power (kW)	50		

Table 5.9: AOC 15/50 wind turbine characteristics.

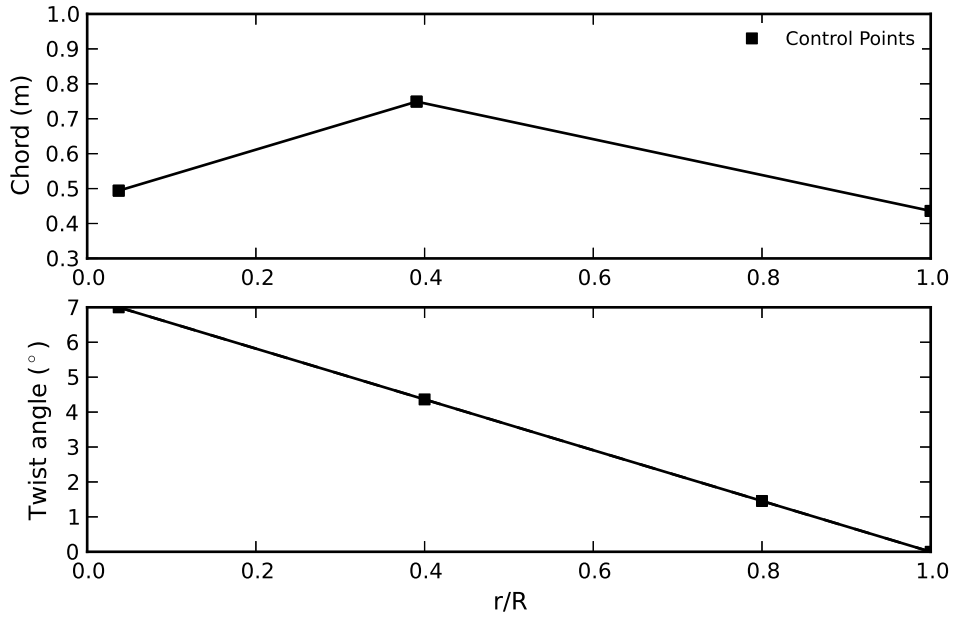


Figure 5.9: Chord and twist distributions of AOC 15/50 wind turbine blade.

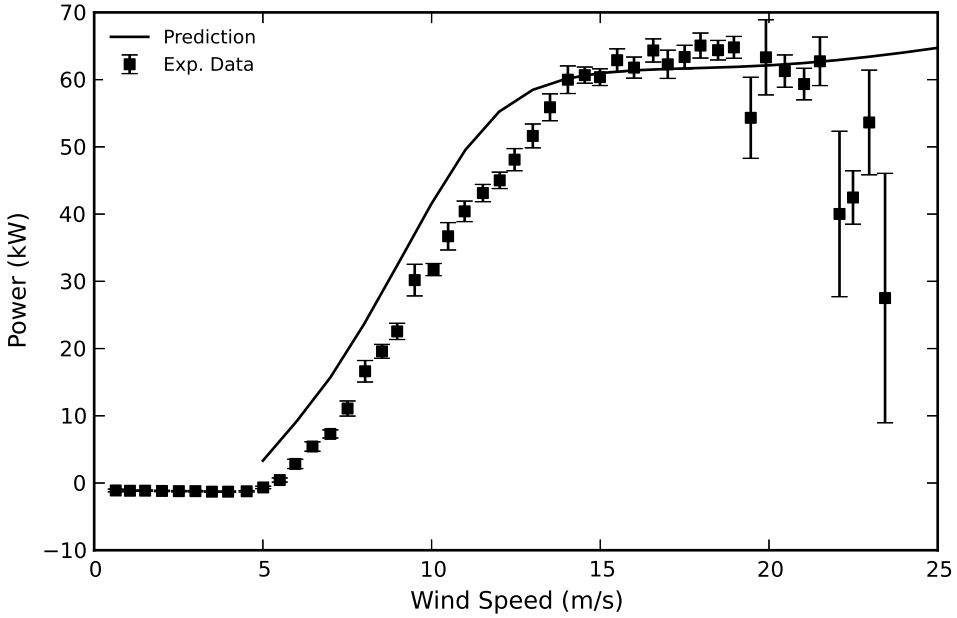


Figure 5.10: Predicted and measured power curves of the AOC 15/50 wind turbine.

Figure 5.11 shows the noise prediction and measured data of the noise generated by the wind turbine at a wind speed of 8 m s^{-1}, for an observer at ground level and a distance of 32.5 m downwind of the turbine ($(x_1, y_1, z_1) = (0, 0, -32.5)$). The turbulent inflow noise dominates the total spectrum, for the majority of the frequency spectrum. Around 2 kHz, the tonal component from trailing edge bluntness contributes significantly. The model agrees with the measured data in the range between 1 and 2 kHz and overestimates the noise for frequencies below 500 Hz. Above 2 kHz the model underpredicts the total noise levels. Also not predicted is the peak at 800 Hz.

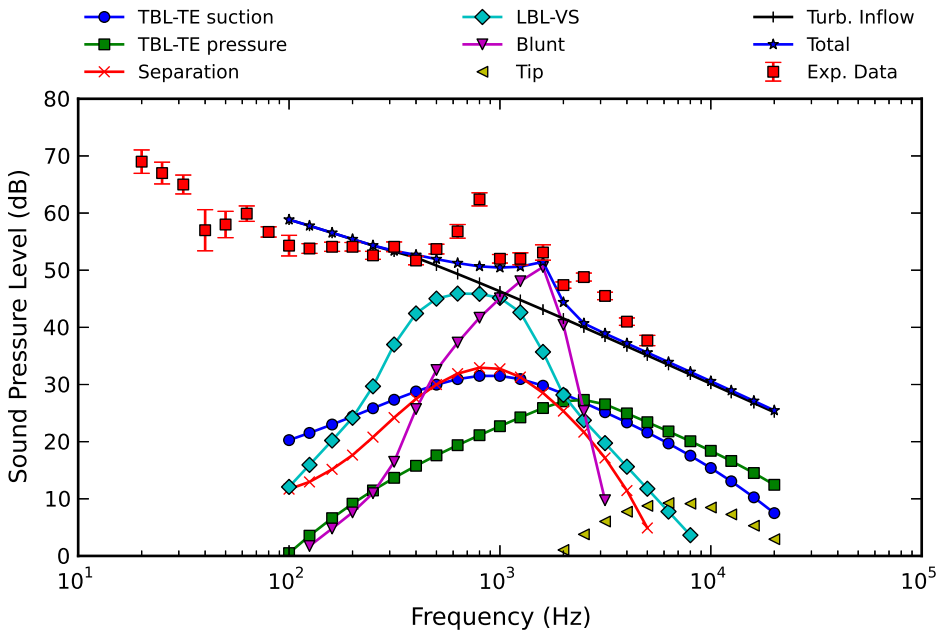


Figure 5.11: Components of noise generated by the AOC 15/50 wind turbine in 8 m s^{-1} winds.

All the simplifications made in the aerodynamic and aeroacoustic prediction models, along with the empirical nature of the latter, result in the previously described discrepancies between the predictions and the experimental data. The geometric model of the blade is also a source of error, as it is based on multiple assumptions, due to the lack of available detailed information on the geometry of the blade. With this in mind, the results predicted by the developed code can be considered acceptable and should provide enough detail and accuracy to use in optimization problems.

Chapter 6

Optimization Framework

Every optimization problem aims to optimize an *objective function* (either by minimizing or maximizing its value) which is a function of a set of *design variables* and is subject to a series of *constraints*. Mathematically, the problem takes the form

$$\begin{aligned} &\text{minimize} && f(\mathbf{x}) \\ &\text{subject to} && g_i(\mathbf{x}) \leqslant 0, && i = 1, \ldots, m \\ &&& h_j(\mathbf{x}) = 0, && j = 1, \ldots, n \\ &&& x_k^L < x_k < x_k^U, && k = 1, \ldots, p \end{aligned} \tag{6.1}$$

where $\mathbf{x}$ is a vector containing the design variables $x_i, ..., x_p$, $f(\mathbf{x})$ is the objective function, and $g_i(\mathbf{x})$ and $h_j(\mathbf{x})$ are the *inequality* and *equality constraints*, respectively. Each variable is bounded by its lower and upper limits x_k^L and x_k^U, referred to as *side constraints*. The design space region in where the variables satisfy all equality, inequality and side constrains is called the feasible search region. For a more detailed definition of the optimization problem and insight on the subject of numerical optimization, the reader should refer to the available literature (Bonnans et al., 2006; Pedregal, 2003).

In this chapter, the optimization framework developed is presented. First the numerical optimization algorithms used in this work are introduced, followed by the description of the framework structure. The chapter ends with convergence analysis of the selected optimization algorithm as a function of the optimization parameters.

6.1 Numerical Optimization Methods

The optimization field has been growing rapidly for the past few decades, with many new theoretical, algorithmic and computational contributions to solve various problems. Optimization methods can be divided in two main different approaches: deterministic and heuristic (see Fig. 6.1). Deterministic methods take advantage of the analytical properties of the problem to generate a sequence of points that converge to a local optimal solution. Examples of deterministic methods are Linear Programming, Nonlinear

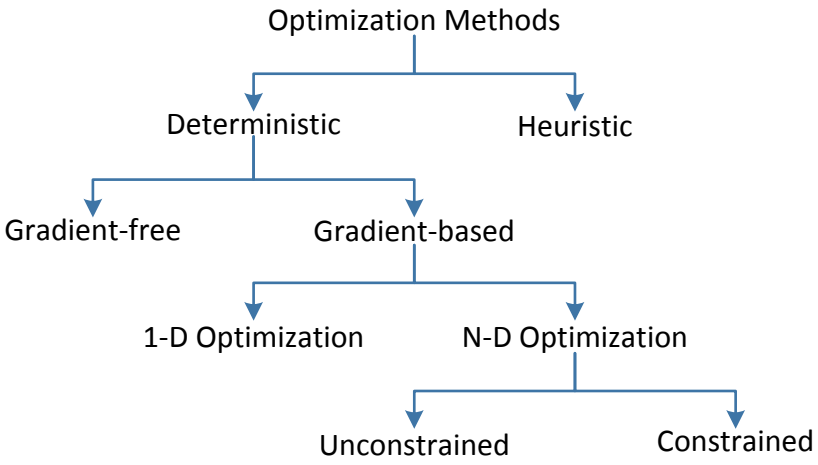


Figure 6.1: Different categories of numerical optimization methods.

Programming or Sequential Quadratic Programming. On the other hand, heuristic methods make use of concepts found in nature to find the global optimal solution. This category includes methods such as Genetic Algorithms, Ant Colonies, Differential Evolution, Particle Swarm, etc.

Heuristic methods are generally more flexible than deterministic methods. However, the quality of the solution cannot be guaranteed, the computational cost increases and the probability of finding the global optimum decreases with increasing problem size. This is the reason why two methods are sometimes used together, with a deterministic method being used to refine the solution obtained by a heuristic method.

Gradient-based and and genetic algorithms are two of the most popular optimization methods because of their efficiency and robustness, respectively, and a brief description of these methods follows.

6.1.1 Gradient-Based Algorithms

Gradient-based (GB) algorithms are search methods that use the gradient of the objective function (also known as derivative or sensitivity) to find an optimal solution. They are some of the oldest optimization algorithms and some of the most widely used. They work by adjusting each decision variable, as the solution goes towards a lower objective function value, in a way proportional to the variation of the objective function. The dependence on the local values of the objective function means that the optimal function might converge to a local minimum. This type of algorithms works best on well-behaved systems where there is one clear optimum, and will work well in high-dimensional spaces, provided these do not have local minima. Usually, the increase of the number of parameters of the search space makes it harder to guarantee the non-existence of local minima, thus increasing the complexity of the optimization techniques. Some examples of GB algorithms are the Modified Feasible Direction (MDF), Sequential Linear Programming (SLP) or Sequential Quadratic Programming (SQP) (Nocedal, 1999). If the objective

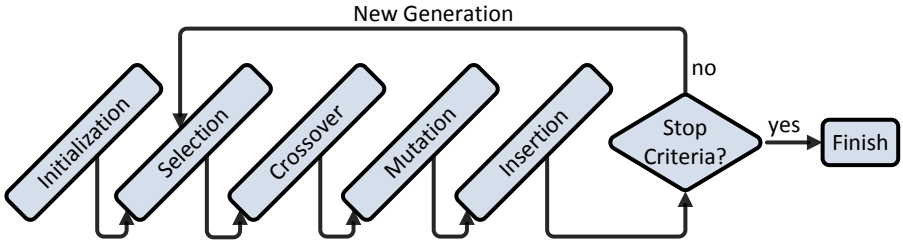


Figure 6.2: Genetic algorithm procedure.

and constraints functions are smooth and their gradients can be evaluated cheaply, the GB algorithms exhibit the best performance compared to all other methods.

6.1.2 Genetic Algorithms

Genetic algorithms (GA) (Goldberg, 1989; Mitchell, 1998) are heuristic search methods, which mimic Darwin's theory of natural evolution. It is a robust strategy for aerodynamic design, with the ability to find the global optimum even when other algorithms fail to do so. The terminology of these algorithms is much like the one used in biology. A *gene* is a design variable, an *individual* is a candidate solution obtained from a set of design variables and a *population* is a group of individuals. The successive populations are referred as *generations*. The GA procedure is shown in Fig. 6.2. It starts by creating the initial population which will then be subjected to a series of operations, in order to generate the following generations. The operations consist in the following:

Selection: Selection of the best individuals (based on their objective function) to be subjected to crossover.

Crossover: Combination of genes of the selected individuals to generate children.

Mutation: Random change of the value of random genes of the children individuals, within the allowed range. It increases the variability of the population

Insertion: Children are inserted into the new population.

The optimization procedure will then terminate whenever the stopping criterion is met. It can either be a specified number of generations or any other specific criterion.

6.2 Framework Description

The optimization framework was developed using pyOpt, a Python-based package for formulating and solving nonlinear constrained optimization problems (Perez et al., 2012). It acts as an interface between the Python environment and many optimization algorithms. With pyOpt, solving an optimization problem is *as simple as* defining a Python function (the objective function) which receives as argument the design variables and outputs the objective function(s) and constrain(s) and passing it to the optimizer.

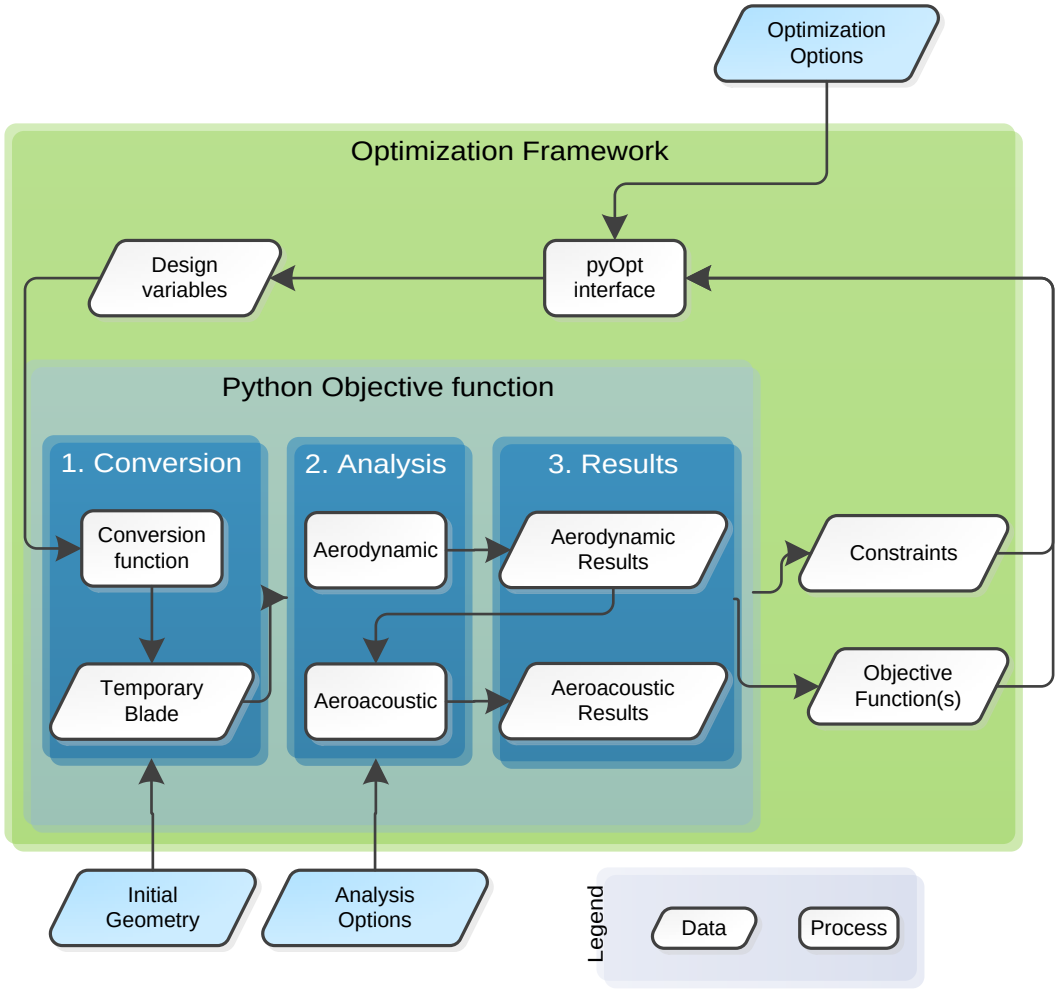


Figure 6.3: Optimization framework flowchart.

The diagram in Fig. 6.3 shows how the optimization framework is constructed and an example of an optimization script using the pyOpt module is presented in Appendix C.

The principal algorithm chosen for the optimizations was the NSGA-II (Deb et al., 2002), which is included in the pyOpt framework. It is a Non-dominated Sorting Genetic Algorithm (NSGA) that solves non-convex and non-smooth single and multi-objective optimization problems. This algorithm has recently been improved and used in the optimization of wind turbine blades by Wang et al. (2011).

A refinement using the Sequential Least-Square Quadratic Programming (SLSQP) algorithm (Kraft, 1988), for the cases where there is a single objective function was also implemented. By using as a starting point the best solution obtained by the previous optimization using GA, this algorithm should, in principle, reach a more optimal solution.

Convergence of the Optimization The stopping criteria of pyOpt's implementation of the NSGA-II algorithm is the maximum number of generations. Increasing this value will increase the probability that

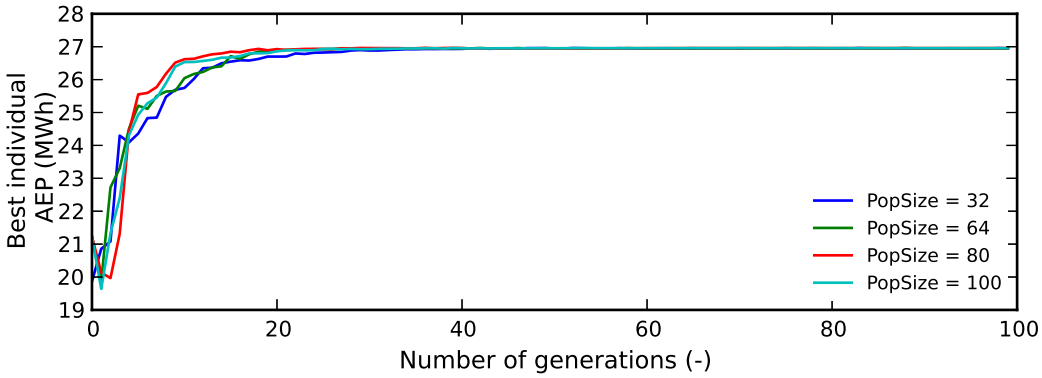


Figure 6.4: Evolution of the fitness of each generation of an optimization case with 6 design variables (twist).

the final solution set has converged, but will also increase the time spend on the optimization. On the other hand, increasing the size of the population might reduce the number of generations required to achieve convergence. Investigation on the optimal size of the population has been performed by Alander (1992) and Gotshall and Rylander (2008), and a general conclusion is that each problem is different and should be analyzed individually. With this in mind, a study of convergence was performed on some of the optimization cases (see chapter 7) in order to gain insight on the requirements of both the number of generations and the size of the population for the optimization problem in question.

Analyzing Fig. 6.4, it is clear that increasing the size of the population slightly decreases the number of generations necessary to achieve a converged population. When looking at the number of function evaluations however, the conclusion is the opposite. In Fig. 6.5, the AEP is presented vs the number of calls of the objective function. The figure shows that a converged solution is achieved using less calls of the function for the smaller population, and although the optimization with the larger population results in a higher final AEP value, the difference between them is less than 1%. The same conclusion can also be taken from Fig. 6.6, where the annual energy production versus number of function calls is presented for a case with 8 design variables. As the number of function calls is directly proportional to the required CPU time, the least function calls the better, therefore, it can be concluded that a population size between n and $2n$ (with n being the number of design variables) guarantees the convergence of the solution while not being too demanding regarding computational time.

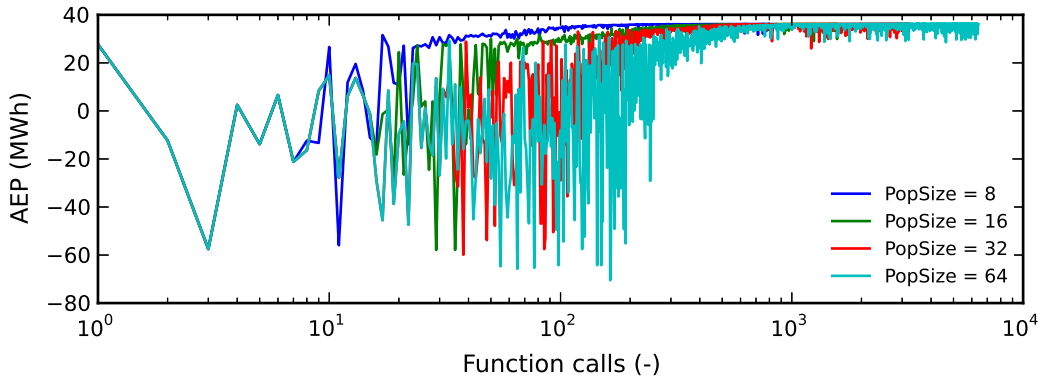


Figure 6.5: Evolution of the fitness of the population of an optimization case with 6 design variables (twist).

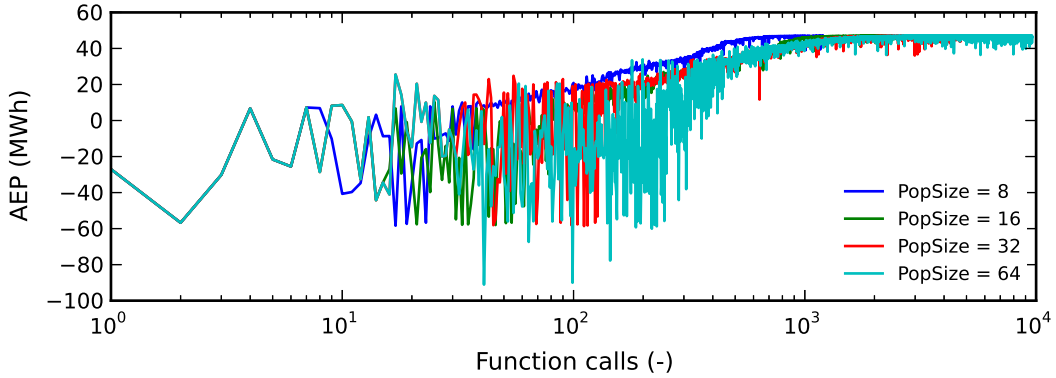


Figure 6.6: Evolution of the fitness of the population of an optimization case with 8 design variables (twist + chord).

Chapter 7

Optimization Results

After introducing all the theory behind the wind turbine aeroacoustic prediction model and the optimization framework, the most relevant results of the various optimization cases that were run are presented. The approach for this work was to start with simple optimization problems, such as single objective optimization on just a few design variables, and gradually increase the number of variables, as well as increasing the number of objective functions to two. The final optimization problems would be multi-objective optimizations of both the airfoil shape and chord and twist distribution.

All the optimizations were performed assuming a wind distribution represented by a Weibull curve with the parameters A and k of 6.48 m s^{-1} and 1.99, respectively, corresponding to an average wind speed of 6.11 m s^{-1}. These parameters represent the wind distribution in the Portuguese municipality of Vila do Bispo, in the southwest of Portugal (Costa, 2004).

The performance of the turbine was predicted for a wind speed range from 4 to 25 m s^{-1}, in 1 m s^{-1} intervals. Due to this, the noise was predicted for a wind speed of 6 m s^{-1}, as it is the closest to the average wind speed of the selected site. A ground roughness value of 0.08 m was chosen and the properties assumed for the air are presented in Tab. 7.1 (assuming standard atmosphere conditions at sea level (McCormick, 1994)). The observer was considered to be located at ground level, at a distance of 32.5 m downwind of the turbine ($(x_1, y_1, z_1) = (0, 0, -32.5)$).

Density	Kinetic viscosity	Speed of sound
1.225 kg m^{-3}	1.46e-5 m^2 s^{-1}	340.4 m s^{-1}

Table 7.1: Properties of the air used in the optimizations.

The boundary layer parameters were computed using XFOIL. It as chosen over RFOIL due to the impossibility of turning off the graphical output of RFOIL, which increased significantly the duration of each simulation and therefore the optimization.

7.1 Optimization of NREL Phase II Turbine Blade

The first set of optimizations was performed on the NREL Phase II turbine blade introduced in chapter 5. The summary of the optimization cases is shown in Tab. 7.2, where the number of design variables and objective functions is presented.

Case Nr.	Design variables			Objective functions	
	Chord	Twist	Sections	AEP	OASPL
1	2	6	0	Yes	No
2	2	6	0	Yes	Yes
3	0	0	2x20	Yes	No
4	0	0	2x20	Yes	Yes
5	2	6	2x20	Yes	Yes

Table 7.2: NREL Phase II optimization cases summary.

The noise was predicted considering all source mechanisms (see section 3.3) and using Guidati's correction. The bluntness noise was computed assuming a TE angle of 6 ° and a TE thickness of 1% of the chord. The complete set of settings used in the simulations is presented in Appendix C and the AEP and OASPL computed using the initial non-optimized blade are presented as reference in Tab. 7.3. Also as reference, the radial distribution of generated noise by the blade is presented in Fig. 7.1.

AEP	OASPL
23.6 MW h	52.18 dB(A)

Table 7.3: Initial OASPL and AEP values of the NREL Phase II wind turbine.

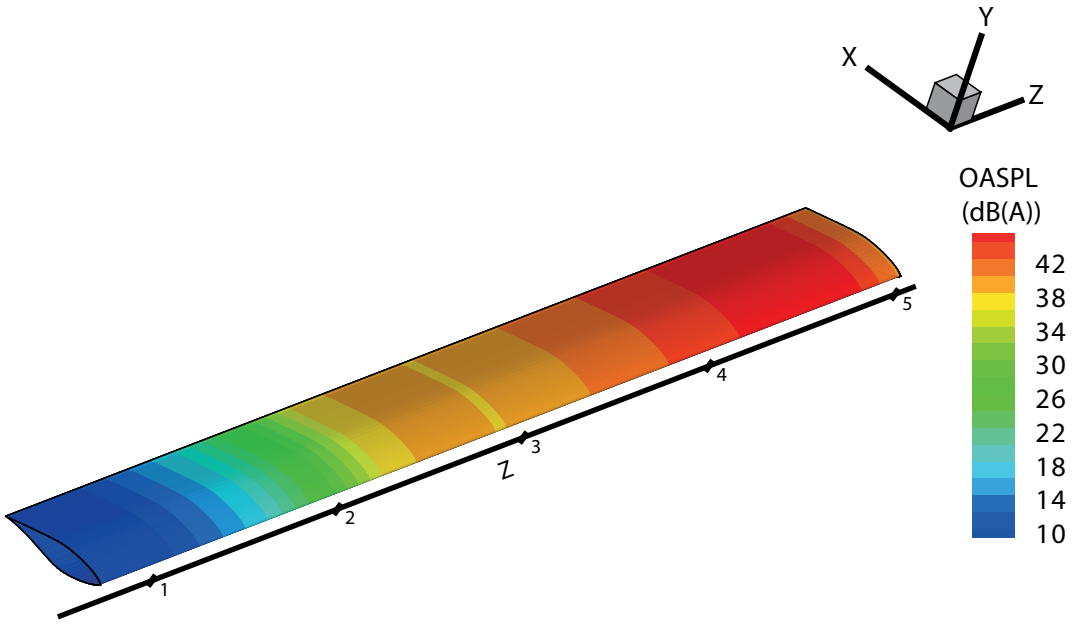


Figure 7.1: Radial distribution of generated noise on the NREL Phase II blade.

7.1.1 Chord and Twist Optimization

The first optimization case presented (**Case 1**) is the optimization of both the chord and the twist of the blade. The chord was defined linearly along the blade, with two control points and the twist with a 5th order Bézier curve, leading to a total of 8 design variables (see Tab. 7.2 and Fig. 7.2).

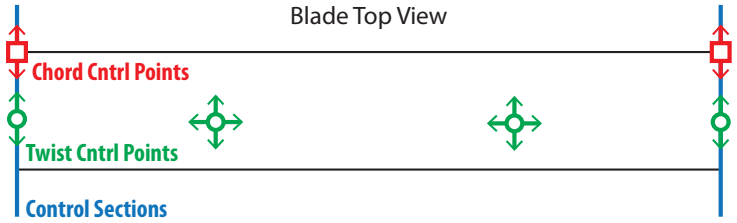


Figure 7.2: Chord and twist control points used in optimization cases 1 and 2.

The AEP was used as the objective function and the following constraints were applied:

Chord and Twist Both chord and twist were constrained so that there was a reduction of their values towards the tip of the blade:

$$\begin{aligned} x_i^{cp} &\leq x_{i+1}^{cp} \\ y_i^{cp} &\geq y_{i+1}^{cp} \end{aligned} \tag{7.1}$$

The optimization can be considered to have achieved convergence, as seen from Fig. 7.3, and resulted in a increase of the AEP up to 46.98 MW h. The optimized chord and twist distributions are presented in Fig. 7.4 and Fig. 7.5, respectively.

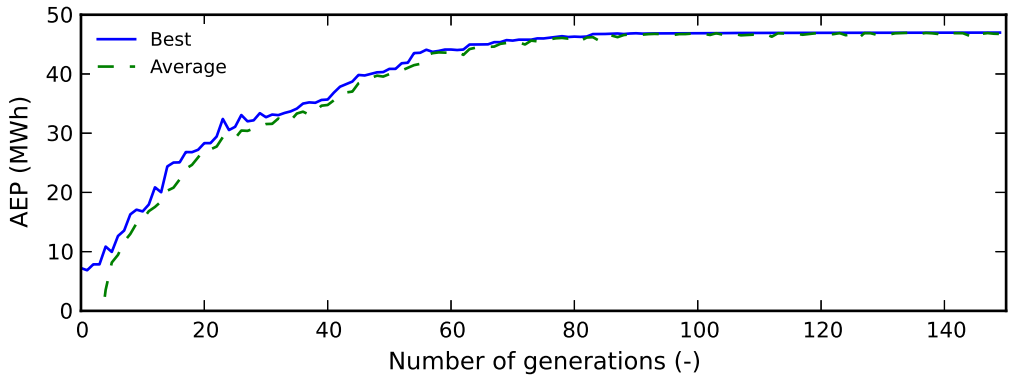


Figure 7.3: Variation of population average and best AEP with number of generations (case 1).

The chord in the optimized blade was maximized up to the defined upper bound of 0.754 m. This is in accordance with what was expected, as the local power is a function of the chord. The optimized twist distribution is similar to the optimum distributions described in the literature (Manwell et al., 2010), increasing towards the root.

Although this optimization did not account for the noise, the OASPL of the turbine with the aerodynamically optimized blade was predicted to gain insight of the relation between increasing power generation and generated noise. The optimization resulted in a noise increase of 2.1 %, with the optimized turbine generating an OASPL of 53.28 dB(A).

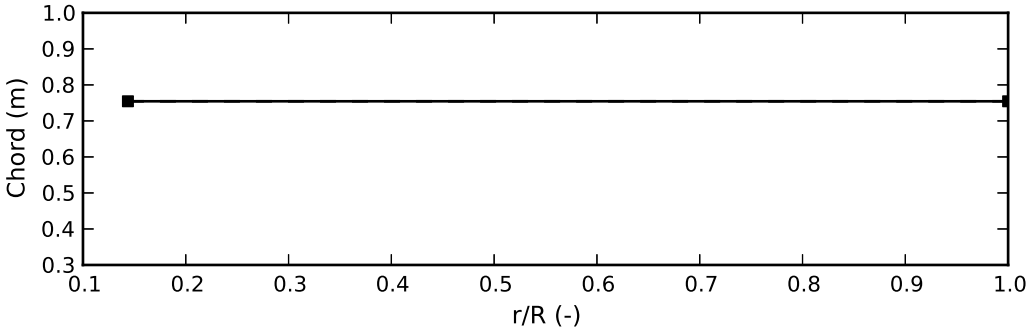


Figure 7.4: Optimal chord distribution in case 1.

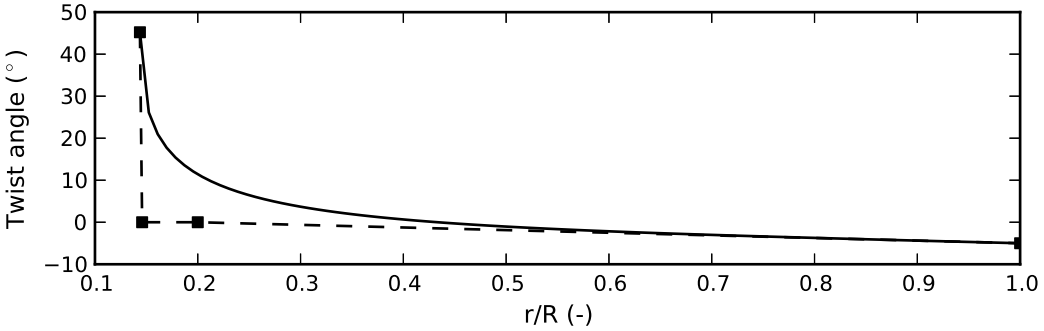


Figure 7.5: Optimal twist distribution in case 1.

The solution obtained with the GA algorithm was refined with the SLSQP algorithm, as mentioned in the previous chapter. The results of the refinement are also presented in Tab. 7.4. It can be seen that the refinement did not improve the solution, indicating that the solution achieved by the GA is very close to the global optimum. For this reason, the refinement was not performed in the single objective optimization cases here after.

	Initial	**Final (GA)**	**Difference**	**Refined (SLSQP)**	**Difference**
AEP [MW h**]**	23.6	46.98	+ 99.1%	46.98	+ 99.1%
OASPL [dB(A)**]**	52.18	53.28	+ 2.1%	53.28	+ 2.1%

Table 7.4: Summary of AEP and OASPL values in case 1.

These results also indicate a trade-off between the energy production and noise levels, which justifies the use of multi-objective optimization as is the case of the next presented optimization case (**Case 2**).

In this optimization case, the OASPL was also used as an objective function, together with the AEP. A new constraint was also added, requiring the AEP to be positive,

$$AEP \geq 0. \tag{7.2}$$

with the intent of forcing the solutions to converge faster to that region.

The optimizer was able to obtain a set of optimal solutions starting from a dispersed set, as seen in Fig. 7.6 where the initial and final populations are presented, and the Pareto front resulting from this optimization case is shown in more detail in Fig. 7.7. It is visible from that figure that a reduction of almost

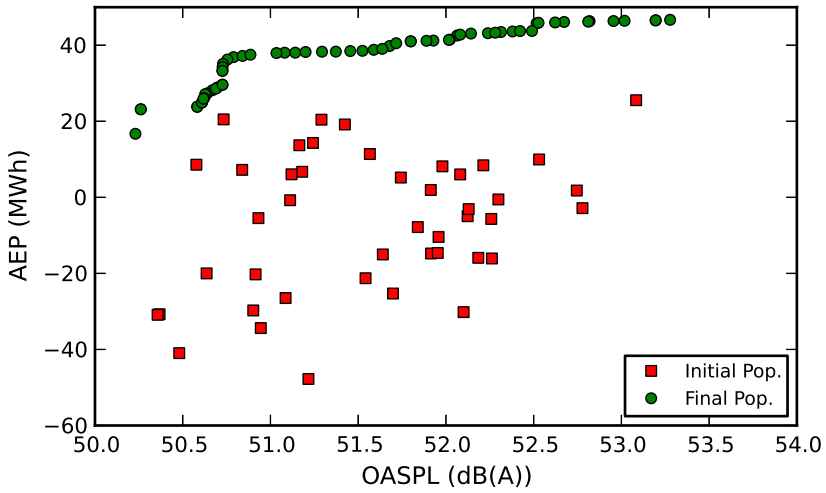


Figure 7.6: Initial and final populations in optimization case 2.

2 dB(A) is possible, with no reduction in performance. On the other hand, maintaining the same noise level, an increase in AEP of 20 MW h can be achieved. It is also visible from Fig. 7.7 that the reduction of noise levels can be achieved between certain ranges of OASPL while maintaining almost the same energy production.

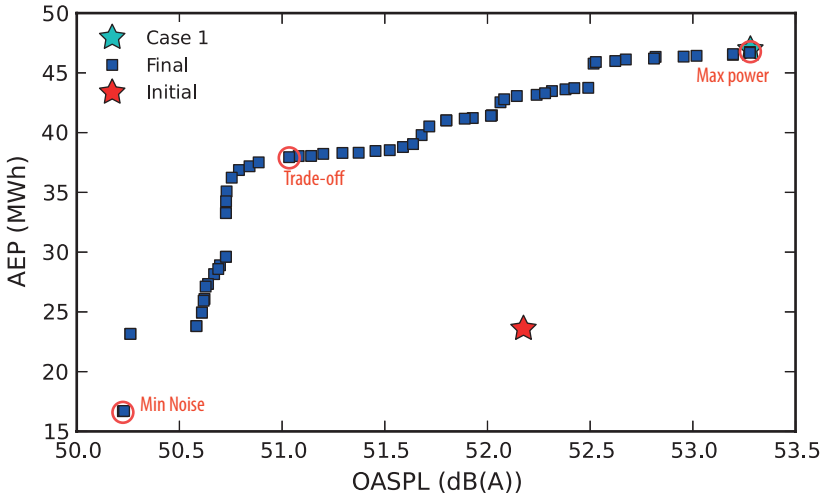


Figure 7.7: Pareto front in optimization case 2.

	AEP [MW h]	Difference [%]	OASPL [dB(A)]	Difference [%]
Initial	23.6	0	52.18	0
Min. Noise	16.7	−29.2	50.22	−3.8
Trade-off	38.04	61.2	51.08	−2.1
Max. Power	46.74	98.1	53.28	0.2

Table 7.5: Summary of AEP and OASPL values in case 2.

Figures 7.8 and 7.9 show the chord and twist distribution, respectively, of three optimized blades: one

that maximizes energy production, one that minimizes the noise levels and another which is a *trade-off* between the others. The three solutions are indicated in Fig. 7.7 and their respective AEP and OASPL values presented in Tab. 7.5.

Regarding the chord distribution, the optimized solution for maximum energy production maximizes the chord throughout the blade, which was expected as the torque produced by each element is proportional to the element chord (see Eq. (2.10)). On the other hand, the optimal solution for minimum noise minimizes the chord down to the lower bound. The noise produced by each element is a function of the boundary thickness parameters, which are proportional to the chord of the element, thus explaining the lower chord values. The trade-off solution varies from the upper bound at the root to the lower bound at the tip, thus reducing the chord at the region where most noise is produced (see Fig. 7.1).

The twist, similarly to what happens in case 1, varies almost linearly through most of the blade, increasing almost exponentially near the root. It can be seen that the minimum noise solution has higher twist angles than the other two solutions, which reduces the effective angle of attack along the blade to lower values. This solution also presents a lower twist change rate near the root when compared to the other two.

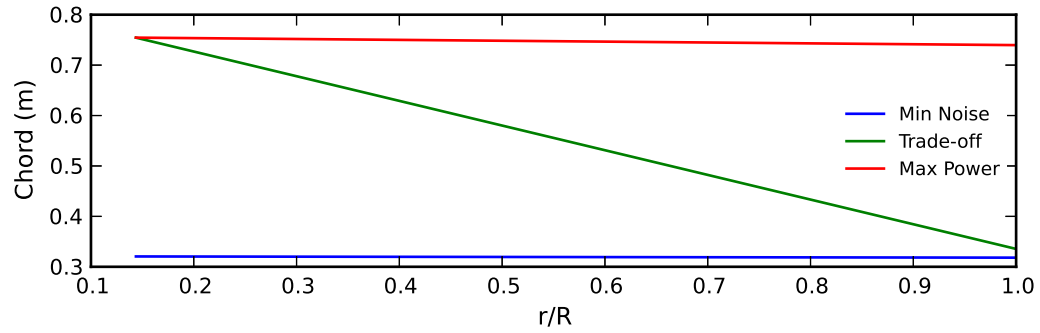


Figure 7.8: Chord distributions of optimized blades in case 2.

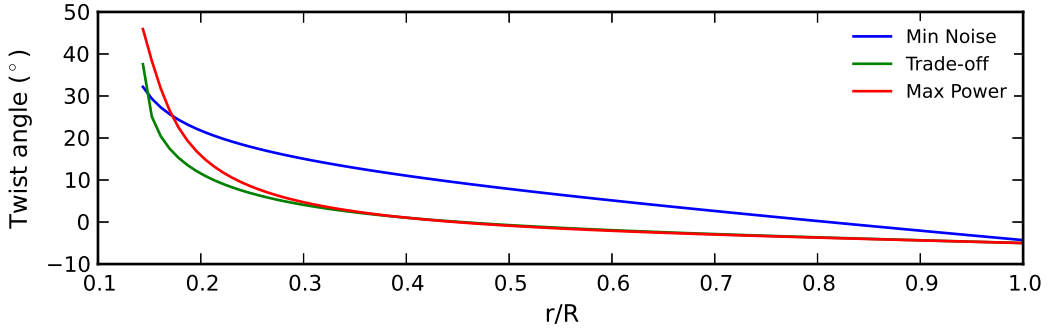


Figure 7.9: Twist distributions of optimized blades in case 2.

7.1.2 Airfoil Shape Optimization

In **Case 3**, the only design variables were the points controlling the airfoil shape in two sections of the blade, at the root hub and tip positions. The only objective function was, like in case 1, the AEP.

The search space of the design variables was defined around the initial airfoil shape variables with a range of $\pm 10\%$ the initial value. The number of design variables at each control section is 20, as schematically represented in Fig. 7.10, leading to a total of 40 design variables (see Tab. 7.2).

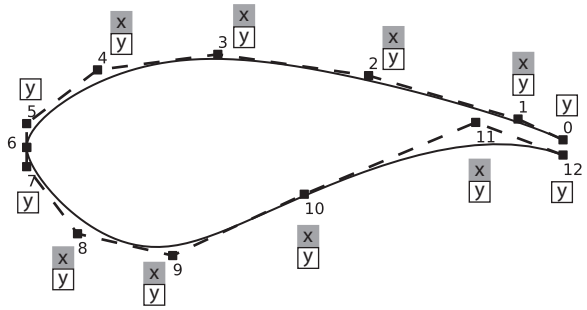


Figure 7.10: Control sections variables used in cases 3, 4 and 5.

The following constraints were used in this optimization case, in order to guarantee the representation of realistic shapes by the NURBS parameterization.

Control Sections

$$\begin{aligned} x_i^{cp} &\geq x_{i+1}^{cp}, \quad \text{upper curve} \\ x_i^{cp} &\leq x_{i+1}^{cp}, \quad \text{lower curve} \end{aligned} \tag{7.3}$$

The optimization resulted in an AEP value of 39.22 MW h, as summarized in Tab. 7.6 and the solution can be considered to have converged, as seen from Fig. 7.11.

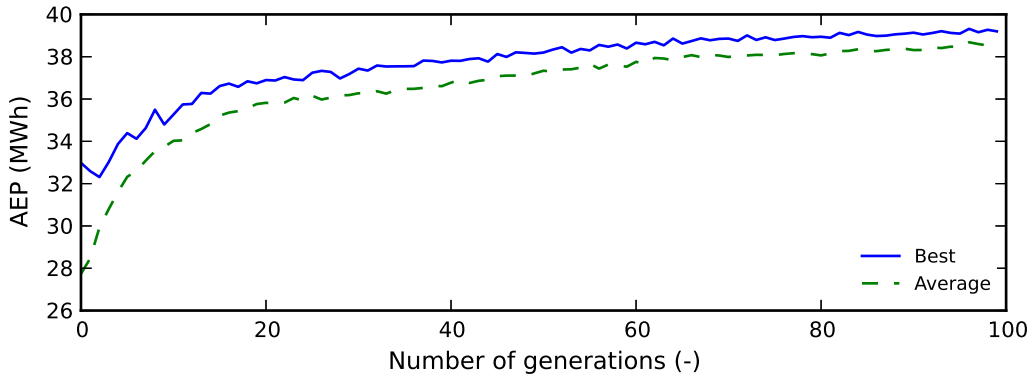


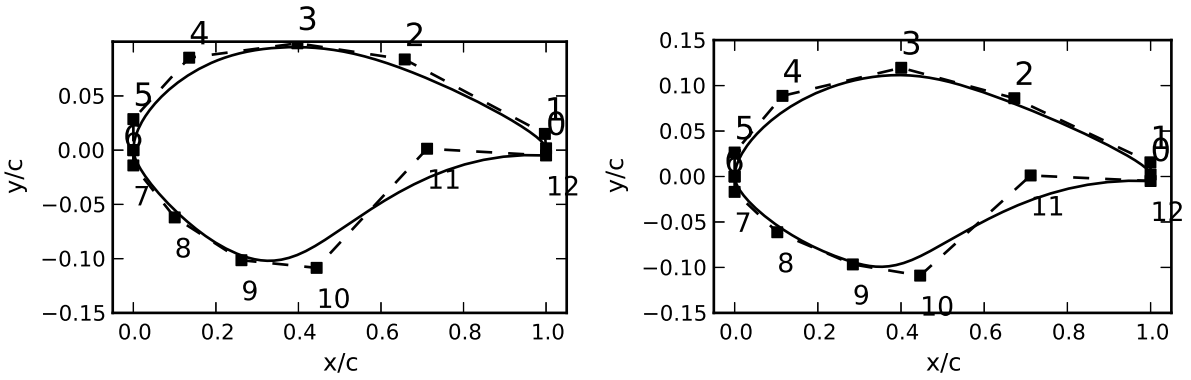
Figure 7.11: Variation of population average and best AEP with the number of generations (case 3).

	Initial	**Final**	**Difference**
AEP [MW h**]**	23.6	39.22	+ 66.2 %
OASPL [dB(A)**]**	52.18	52.11	+ 0%

Table 7.6: Summary of AEP and OASPL values in case 3.

The optimal control points for the sections are presented in Fig. 7.12 and the corresponding shapes are compared to the initial airfoil shape in Fig. 7.13. A more accentuated S-tail is noticeable, which results

in an increase in aft-loading, as shown in Fig. 7.14, where the pressure coefficient C_p around optimized and initial airfoils is presented, for an angle of attack of 5 ° .



(a) Root (b) Tip

Figure 7.12: Optimized airfoil shapes and control points in optimization case 3.

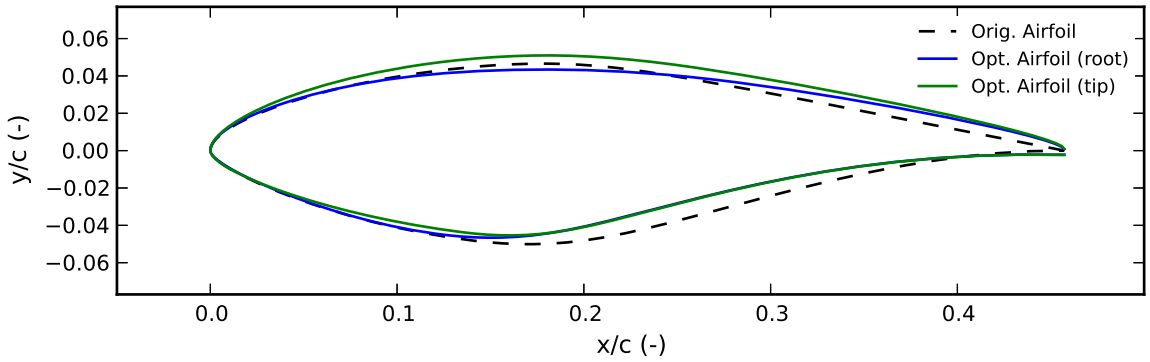
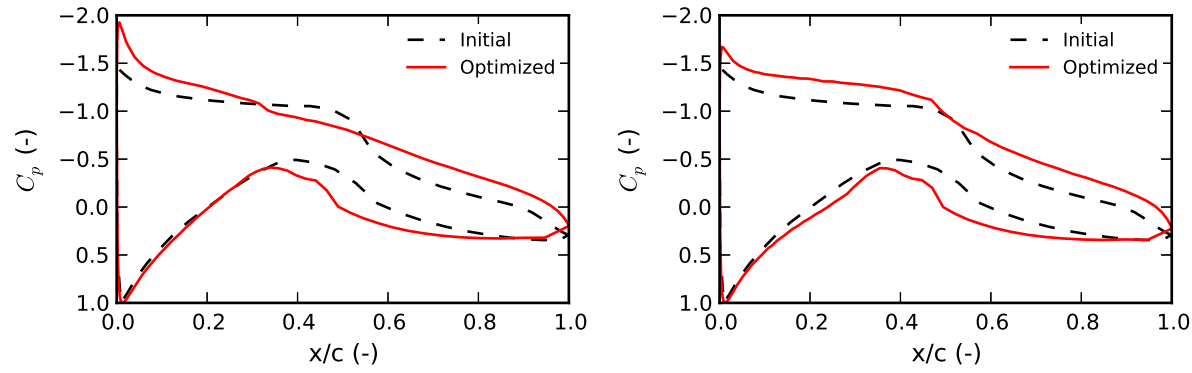


Figure 7.13: Initial and optimized airfoil shapes in optimization case 3.



(a) Root (b) Tip

Figure 7.14: Pressure coefficient around initial and optimized airfoils in optimization case 3, for an angle of attack of 5 ° and Re = 1.5e6.

It is also visible from Fig. 7.13 that the airfoil at the tip has a larger relative thickness t/c than the inboard airfoil. This is not in accordance with the design goals of airfoils for wind turbines, due to structural

reasons, amongst others. The lack of constraints regarding thickness and the limited variable search space are possible reasons for this result.

The multi-objective optimization of the blade, regarding the airfoil shape (**Case 4**) resulted in the Pareto front presented in Fig. 7.15. Like in the Pareto front obtained in case 2, this front shows a range of possible solutions where the noise levels can be reduced with only a small decrease in energy production. Table 7.7 presents the AEP and OASPL values of the three different solutions indicated in Fig. 7.15.

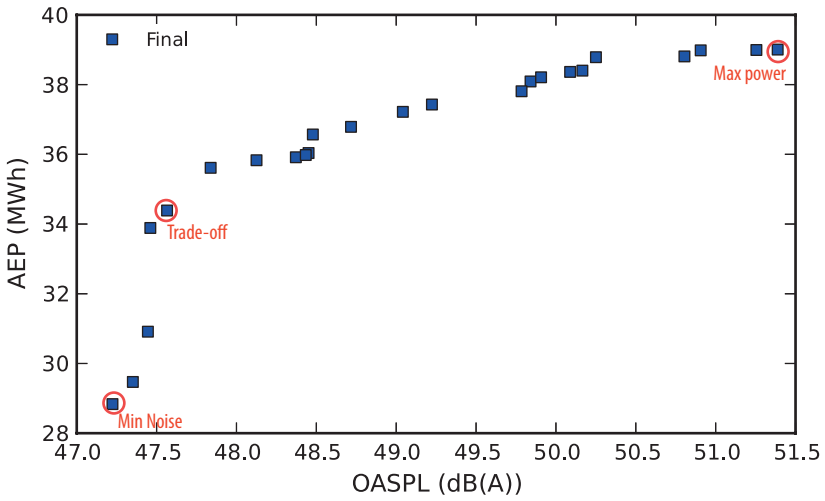


Figure 7.15: Pareto front in optimization case 4.

	AEP [MW h]	Difference [%]	OASPL [dB(A)]	Difference [%]
Initial	23.6	0	52.18	0
Min. Noise	28.83	22.2	47.22	−9.5
Trade-off	35.61	50.9	47.84	−8.3
Max. Power	39.01	65.3	51.38	−1.5

Table 7.7: Summary of AEP and OASPL values in case 4.

Figures 7.16 and 7.17 show the airfoil shapes of three different solutions, like in the previous section, which are indicated in the Pareto front of Fig. 7.15. The minimum noise and compromise shapes are more similar, particularly at the tip, which can be explained by the noise being mainly generated on the outer part of the blade (see Fig. 7.1).

In Fig. 7.18, the three blade geometries are presented with the contour plot of the radial distribution of generated noise levels. There is an evident reduction in the noise levels in the outer region of the blade.

By analyzing the results from optimization case 4 and comparing them to case 2, it can be concluded that while changing the shape of the airfoil can reduce the predicted turbine noise levels to an higher extent than by changing the twist and chord distributions, the later results in higher AEP values.

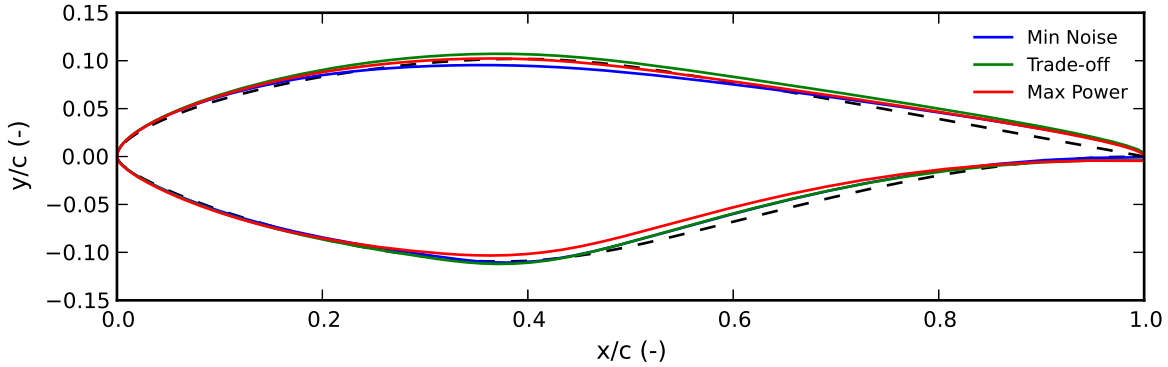


Figure 7.16: Initial and optimized airfoil shapes at the root in optimization case 4.

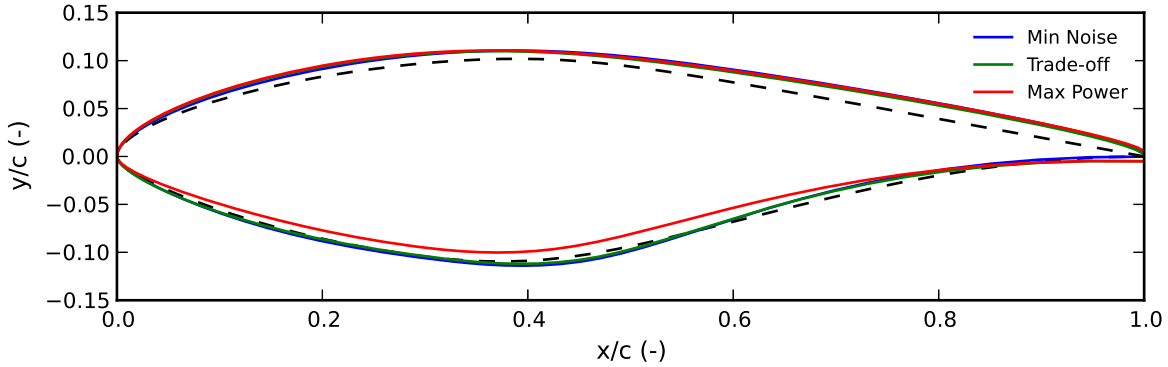


Figure 7.17: Initial and optimized airfoil shapes at the tip in optimization case 4.

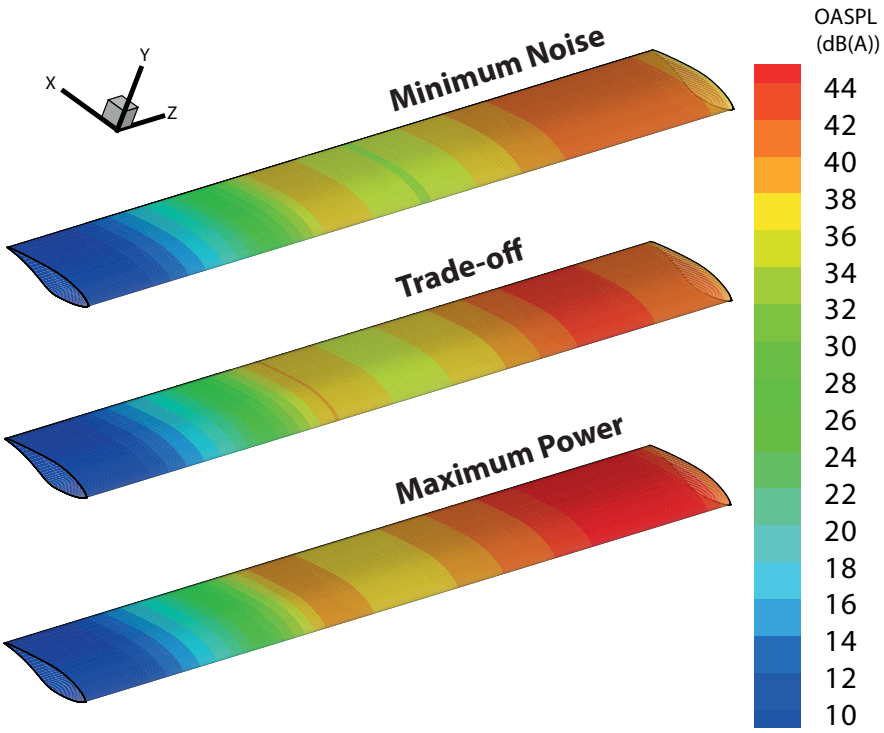


Figure 7.18: Comparison of the radial distribution of generated noise on between various optimized blades in case 4.

7.1.3 Chord, Twist and Airfoil Shape Optimization

The last optimization case performed on the NREL Phase II wind turbine blade (**Case 5**) optimized both chord, twist and airfoil shape distributions with both AEP and OASPL as objective functions. The constraints were the same as the previous optimization cases:

Chord and Twist Both chord and twist were constrained so that there was a reduction of their values towards the tip of the blade:

$$\begin{aligned} x_i^{cp} &\leq x_{i+1}^{cp} \\ y_i^{cp} &\geq y_{i+1}^{cp} \end{aligned} \tag{7.4}$$

Control Sections

$$\begin{aligned} x_i^{cp} &\geq x_{i+1}^{cp}, \quad \text{upper curve} \\ x_i^{cp} &\leq x_{i+1}^{cp}, \quad \text{lower curve} \end{aligned} \tag{7.5}$$

Objective Functions

$$AEP \geq 0 \tag{7.6}$$

The initial and final populations are presented in Fig. 7.19 showing that, like in case 2, the optimizer was able to obtain a set of optimal solutions starting from a dispersed set.

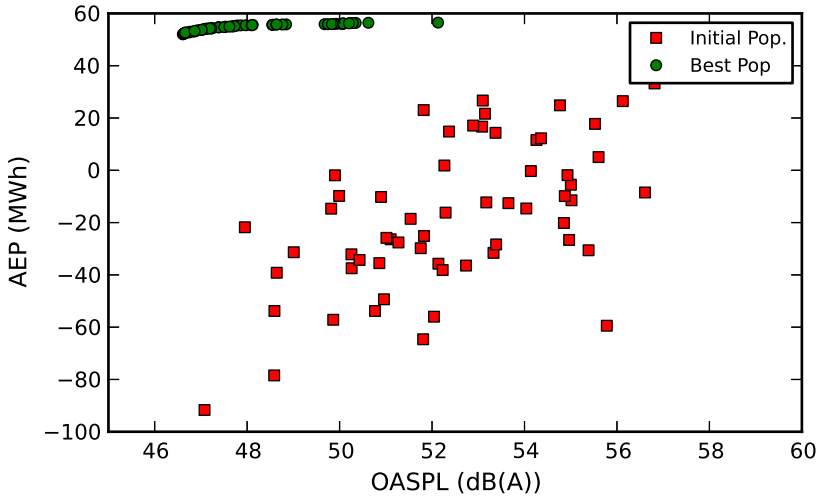


Figure 7.19: Initial and Final populations in optimization case 5.

The Pareto front of Fig. 7.19 is presented in detail in Fig. 7.20, where it can be seen that all optimized blade geometries result in a reduction of the predicted noise level. It is also visible that a reduction from 50 to 48 dB(A) can be achieved with almost no variation in energy production. The summary of the AEP and OASPL values of the solutions indicated in the Pareto front are presented in Tab. 7.8.

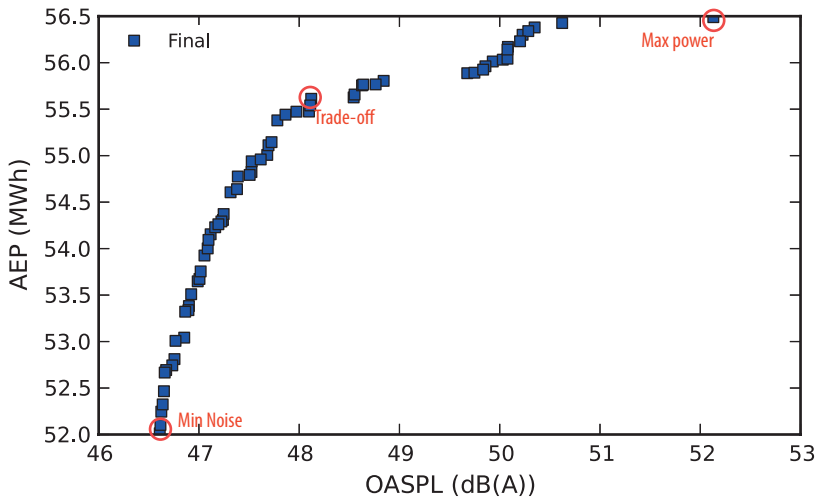


Figure 7.20: Pareto front in optimization case 5.

	AEP [MW h]	Difference [%]	OASPL [dB(A)]	Difference [%]
Initial	23.6	0	52.18	0
Min. Noise	52.01	120.4	46.61	−10.7
Trade-off	55.54	135.3	48.11	−7.8
Max. Power	56.49	139.4	52.13	−0.1

Table 7.8: Summary of AEP and OASPL values in case 5.

The chord and twist distributions of three different optimized blades are presented in Fig. 7.21 and Fig. 7.22, respectively. The three solutions (see Fig. 7.20) were chosen similarly to the ones in the previous cases (the one generating the minimum noise, the one maximizing energy production and a solution in between the previous two.

Regarding the chord distribution, the three solutions are very similar, with a chord practically constant and equal to the upper limit of 0.754 m along the blade. Although the maximum power blade presents a slightly higher reduction of chord towards the tip, it is only of about 4 cm.

The twist distribution is similar to the ones obtained in the previous cases and the three solutions are very similar between them.

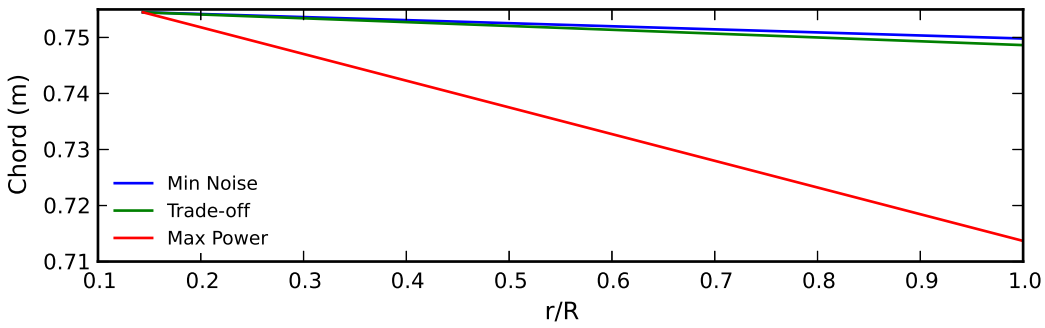


Figure 7.21: Chord distributions of optimized blades in case 5.

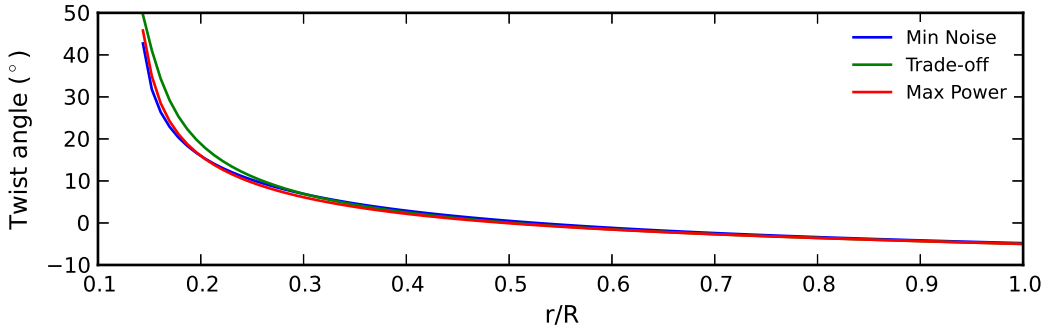
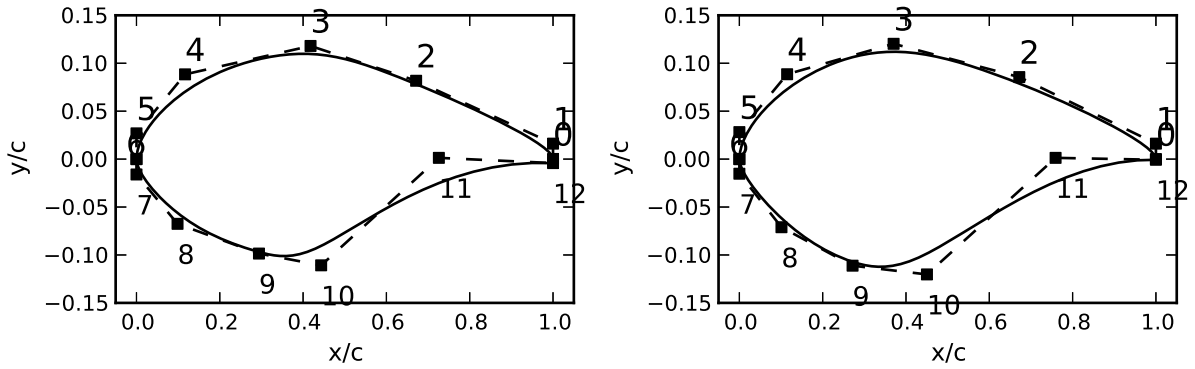


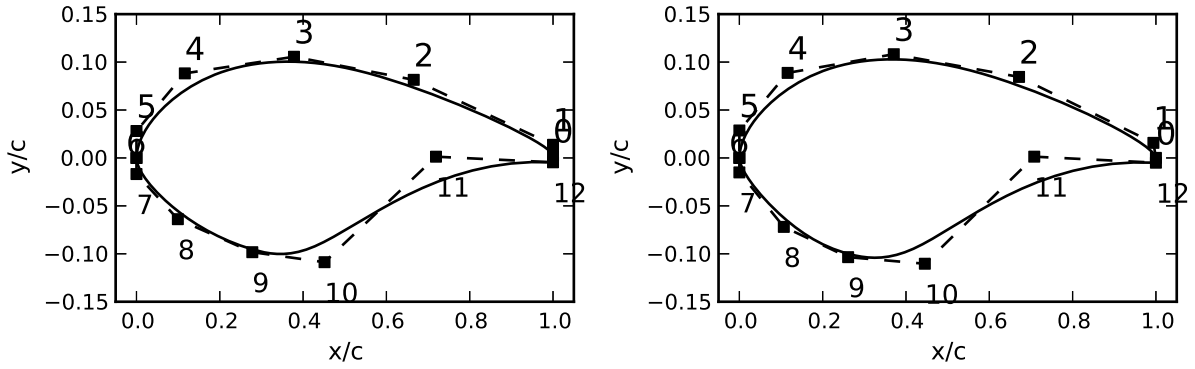
Figure 7.22: Twist distributions of optimized blades in case 5.

The control points positions of the airfoil shapes of the minimum noise and maximum power solutions are presented in Fig. 7.23 and Fig. 7.24, respectively. The initial and optimized airfoil shapes at the root and tip positions are shown in Fig. 7.25 and Fig. 7.26, respectively. Similarly to the shapes obtained in the previous cases, there is a slightly more accentuated s-tail in the optimized airfoils. In the root region, the three airfoils are very similar to each other, with the exception of the upper side of the minimum noise airfoil. At the tip, the differences between the airfoils are more visible, with the trade-off airfoil having a lower curve similar to the maximum power airfoil and an upper curve similar to the minimum noise one.



(a) Root (b) Tip

Figure 7.23: Minimum noise optimized airfoil shapes and control points in case 5.



(a) Root (b) Tip

Figure 7.24: Maximum power optimized airfoil shapes and control points in case 5.

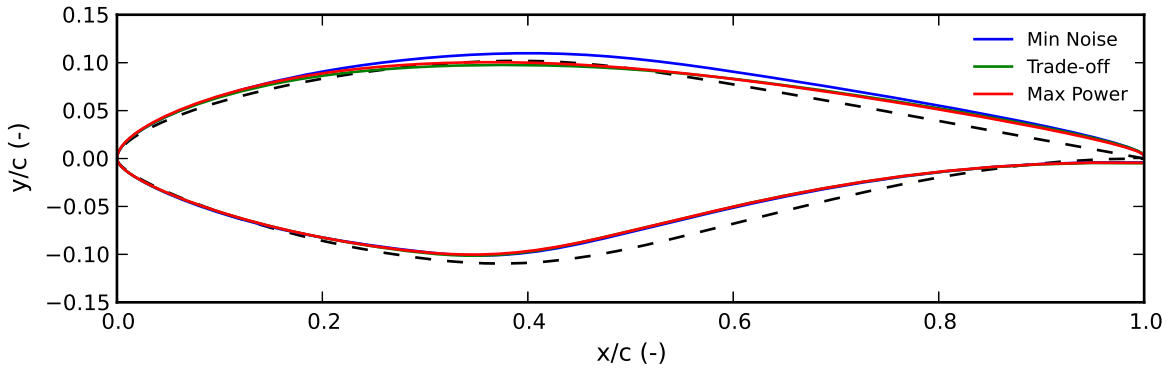


Figure 7.25: Initial and optimized airfoil shapes at the root in optimization case 5.

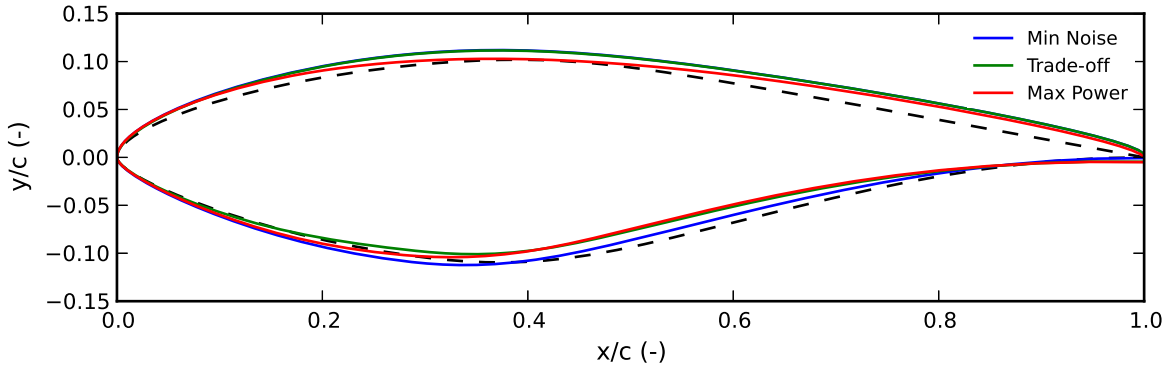
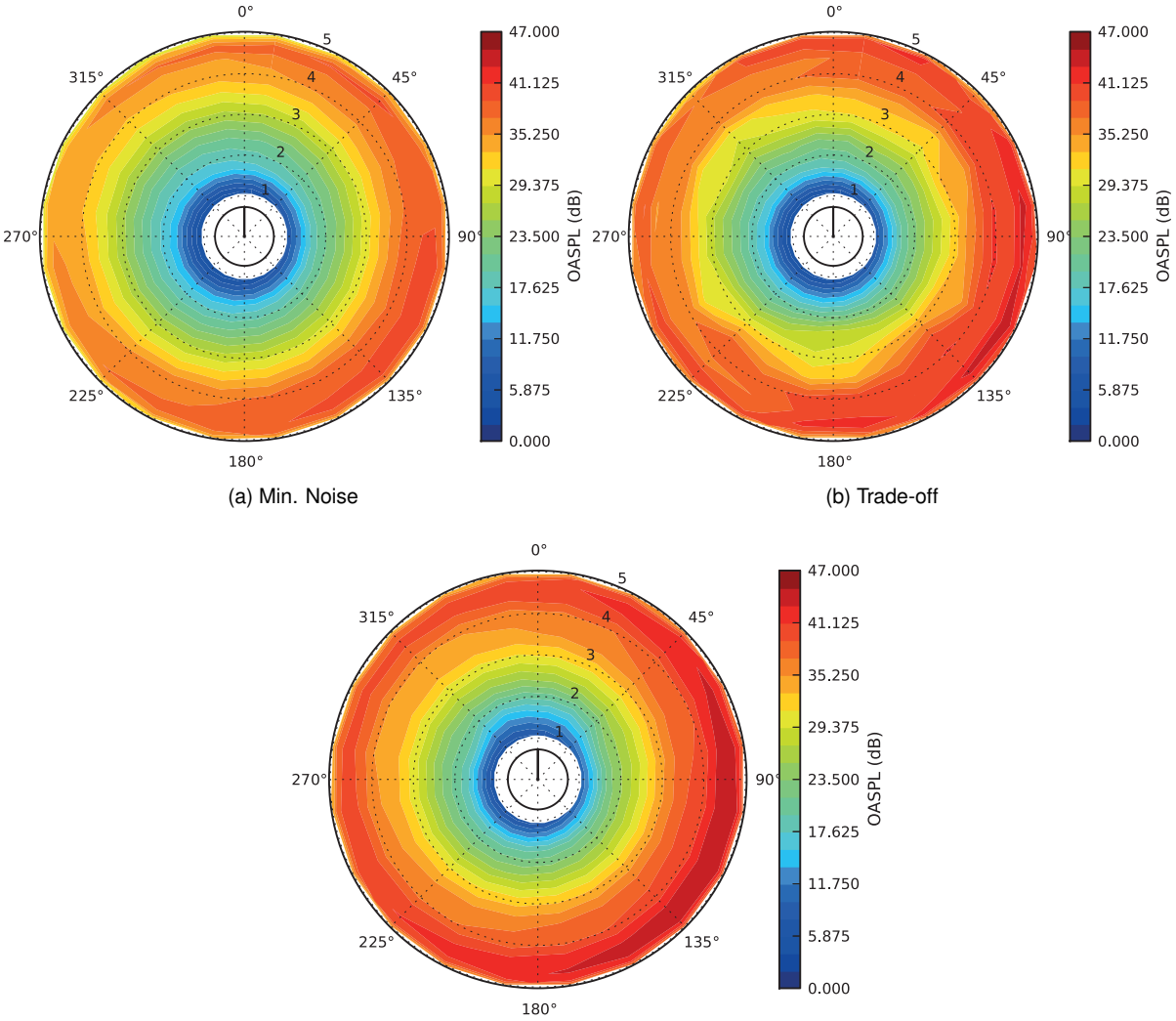


Figure 7.26: Initial and Optimized airfoil shapes at the tip, obtained in optimization case 5.

Figure 7.27 presents the noise field generated by each of the three optimized blades, where it can be seen that higher noise levels are generated when the blade is descending. This is a result of Doppler amplification, which amplifies the sound pressure level for sources moving towards the observer and decreases it for those moving away. By comparing the three solutions, it can be seen that the maximum power solution presents much higher noise values than the other two and that the reduction in noise from one solution to another happens mostly in the outer part of the blade.



(a) Min. Noise

(b) Trade-off

(c) Max. Power

Figure 7.27: Overall sound pressure level across the rotor for various optimized blades from case 5.

7.1.4 Summary of the Results

The summary of the AEP and OASPL values obtained in all the previous described optimization cases, along with the CPU time that each of the optimizations took are presented in Tab. 7.9, where it can be seen that with increasing number of variables used in the problem, the optimal solutions presents lower noise levels and higher energy production levels.

The big difference in CPU time between the single objective and multi-objective optimization cases is due to the noise computations performed in the latter. The differences between case 1 and 3 are due to the polars in the first case being computed before the optimization, as the airfoil profile of the blade remained constant.

All optimizations were performed in a Windows 7 operating system installed in a laptop with an Intel® CORE™ i7 @ 2.30 GHz processor and 6 GB of RAM.

Case Nr.		AEP [MW h]	OASPL [dB(A)]	CPU Time [hours]	Pop. Size [-]	Max Gen. [-]
	Initial	23.6	52.18	-	-	-
1		46.98	53.28	1.44	64	150
2	**Min. Noise**	16.7	50.22	15.61	64	150
	Trade-off	38.04	51.08			
	Max. Power	46.74	53.28			
3		39.22	52.11	4.24	60	100
4	**Min. Noise**	28.83	47.22	36.83	64	200
	Trade-off	35.61	47.84			
	Max. Power	39.01	51.38			
5	**Min. Noise**	52.01	46.61	40.97	64	200
	Trade-off	55.54	48.11			
	Max. Power	56.49	52.13			

Table 7.9: Summary of the optimization cases performed on the NREL Phase II wind turbine blade.

7.2 Optimization on the AOC 15/50 Turbine Blade

Due to the simplicity of the NREL Phase II blade, high aeroacoustic improvements were expected to be achieved. The results showed that reduction in noise levels is possible without much, if any, reduction in aerodynamic performance. The previous results also showed that if not properly constrained, the solution might converge in geometries that might have unwanted structural characteristics. With this in mind, an optimization was performed on the AOC 15/50 turbine blade and is presented in this section.

Design variables As previously mentioned in section 5.3, the AOC 15/50 airfoil shape is defined at 4 stations, being the first at the hub considered to be circular, and the other three at 40%, 75% and 95% of the blade. From each control section, the coordinates of 10 control points were used as design variables, as shown in Fig. 7.28 (note that in this optimization case the trailing edge of the airfoil was not altered). The twist was defined using a 5^{th} order Bézier curve, resulting in 6 design variables. The chord defined by linear interpolation of 3 control points, resulting in 2 design variables (see Fig. 7.29). The search space of the control sections variables was defined as $\pm 10\%$ in the x-direction and $\pm 20\%$ in the y-direction, relative to the initial control points. The chord variables search space was defined as $\pm 20\%$ of the initial chord values and the twist as $\pm 30\%$ in the x-direction and and $\pm 10\%$ in the y-direction. This resulted in a total of 62 design variables.

Constraints The constraints used in this optimization case were the following:

Chord and Twist Both chord and twist were constrained so that there was a reduction of their values

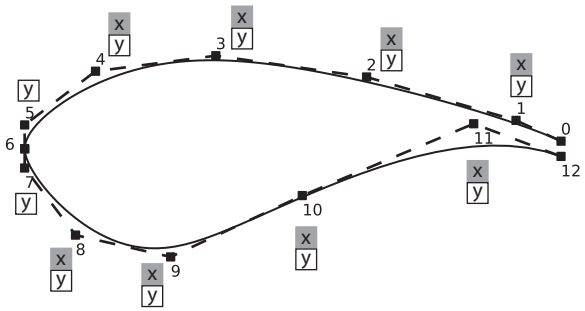


Figure 7.28: Control sections variables used in the AOC 15/50 optimization case.

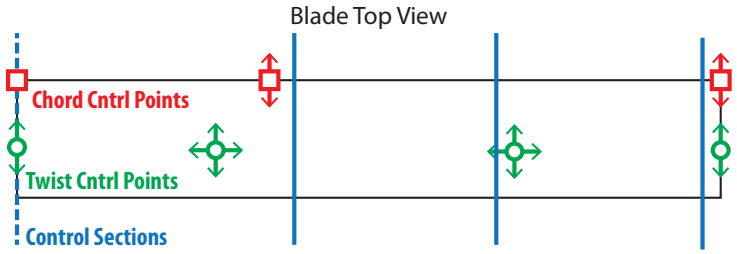


Figure 7.29: Chord and twist control points and control sections used in the AOC 15/50 optimization case.

towards the tip of the blade (not counting with the chord at the root):

$$\begin{aligned} x_i^{cp} &\leq x_{i+1}^{cp} \\ y_i^{cp} &\geq y_{i+1}^{cp} \end{aligned} \tag{7.7}$$

Control Sections

$$\begin{aligned} x_i^{cp} &\geq x_{i+1}^{cp}, \quad \text{upper curve} \\ x_i^{cp} &\leq x_{i+1}^{cp}, \quad \text{lower curve} \\ y_2^{cp} &\geq y_3^{cp} \\ y_{10}^{cp} &\geq y_9^{cp} \end{aligned} \tag{7.8}$$

Objective Functions

$$AEP \geq 0 \tag{7.9}$$

Run Conditions The TE thickness was assumed to be 1% of the chord, with a constant angle of 6 %. Due to the non aerodynamic shape of the sections up to 40 % of the blade, the noise was only computed in the 60% outer part.

Results The initial AEP and OASPL values are presented, as reference, in Tab. 7.10 and the radial distribution of generated noise by the blade is presented in Fig. 7.30.

AEP	OASPL
114.76 MW h	60.61 dB(A)

Table 7.10: Initial OASPL and AEP values of the AOC 15/50 wind turbine.

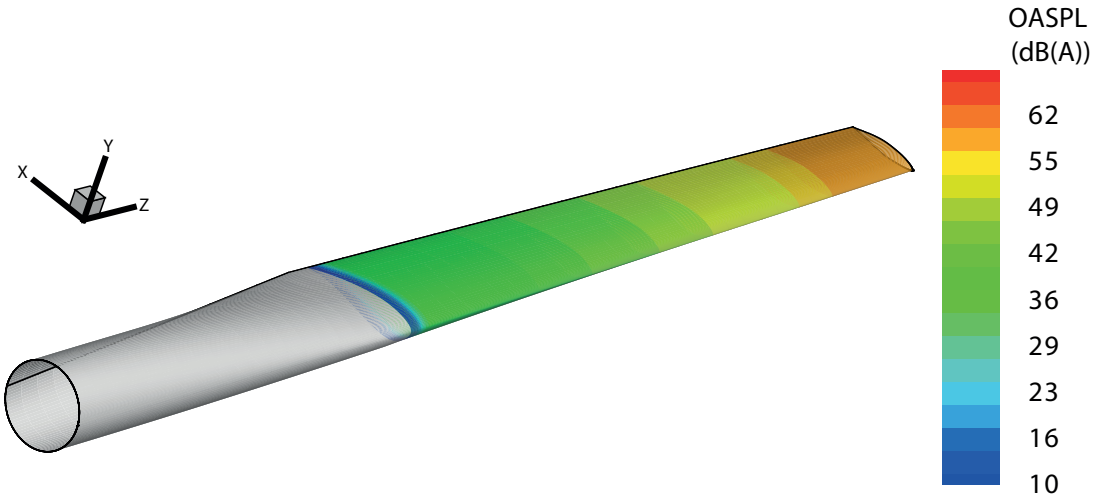


Figure 7.30: Radial distribution of generated noise on the AOC 15/50 blade.

The variation of minimum and best AEP and OASPL values of each generation are presented in Fig. 7.31.

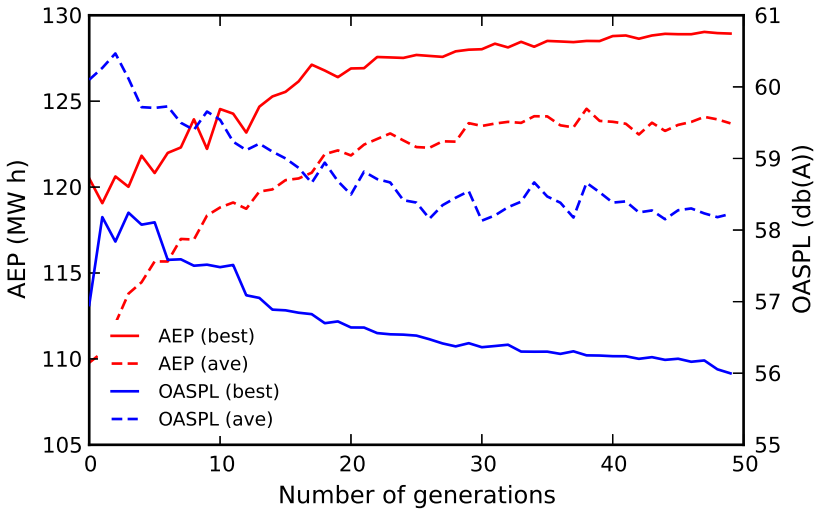


Figure 7.31: Variation of population average and best objective functions with number of generations in the AOC 15/50 optimization case.

The resulting Pareto front is presented in Fig. 7.32 where three solutions are identified, similarly the previous optimization cases. A reduction in OASPL of more than 4 dB(A) is possible, relative to the initial blade, without any loss in energy production. On the other hand, while maintaining the noise levels, an increase of 14 MW h can be achieved. The Pareto front ranges between an OASPL of 56 dB(A) to about 63 dB(A) and from about 57.5 dB(A) to 63 dB(A) the AEP varies very little. The summary of the AEP and OASPL values obtained in the optimization for the three solutions indicated in Fig. 7.32 are presented in Tab. 7.11.

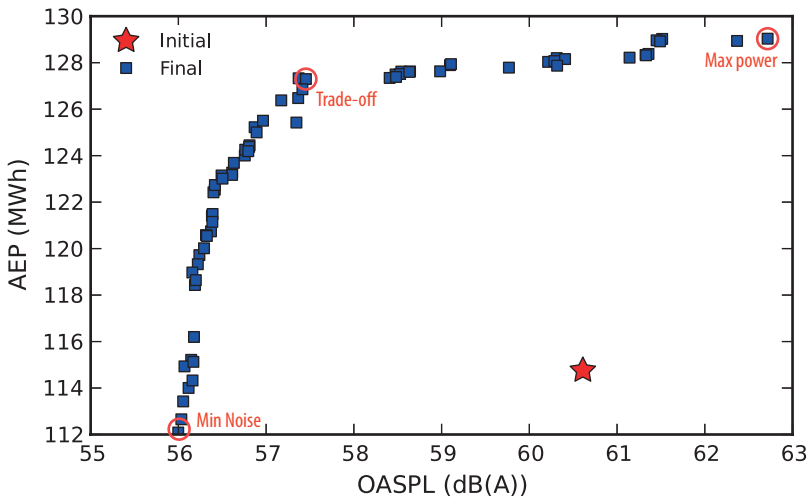


Figure 7.32: Pareto front in AOC 15/50 blade optimization case.

	AEP [MW h]	Difference [%]	OASPL [dB(A)]	Difference [%]
Initial	114.76	0	60.61	0
Min. Noise	112.09	−2.3	56	−7.6
Trade-off	127.33	11.0	57.37	−5.3
Max. Power	129.03	12.4	62.72	3.5

Table 7.11: Summary of AEP and OASPL values in AOC 15/50 optimization case.

Figures 7.33 and 7.34 present the chord twist distributions, respectively, of the three solutions indicated in Fig. 7.32. The chord was maximized to the upper bounds at 0.4 and 1.0 r/R in the three solutions, with the exception of the minimum noise solution, where the chord is slightly smaller than the other solutions at 0.4 r/R. Regarding the twist distributions, the three solutions present higher twist angles than the initial values allover the blade, with the exception of the tip region, where the twist change rate increases and the twist angles are lower than the initial values.

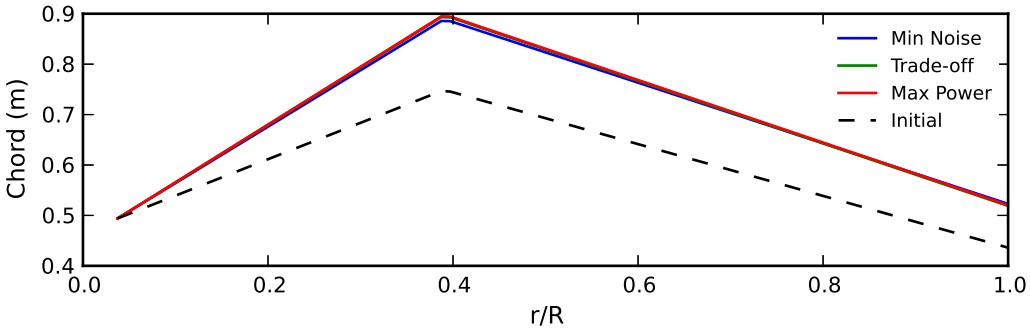


Figure 7.33: Chord distributions of optimized AOC 15/50 blades.

The optimized airfoil shapes defined at 40%, 75% and 90% of the blade are presented in Fig. 7.35, Fig. 7.36 and Fig. 7.37 , respectively. The same behavior of previous cases is observed in these results, regarding the differences between the airfoil shapes of the different solutions. While at 40% and 75% of

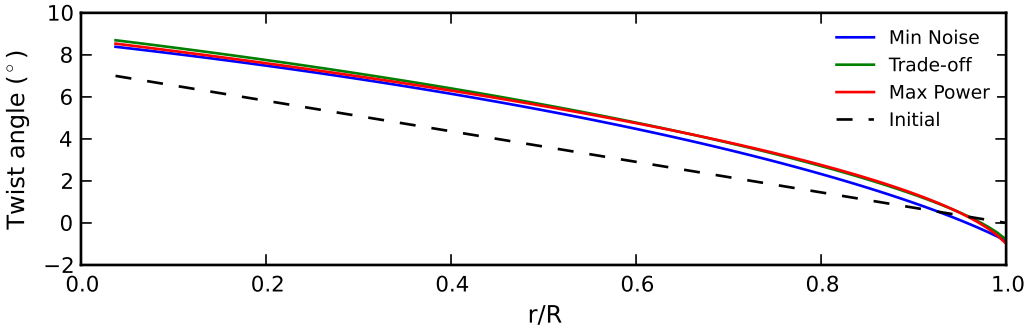


Figure 7.34: Twist distributions of optimized AOC 15/50 blades.

the blade the, Trade-off airfoil shape is a mixture of the other two, at 95% of the blade it is much closer to the Minimum Noise airfoil shape, particularlly the upper side. This comes as a result of the previously mentioned fact that the noise is mainly generated in the outer region of the blade.

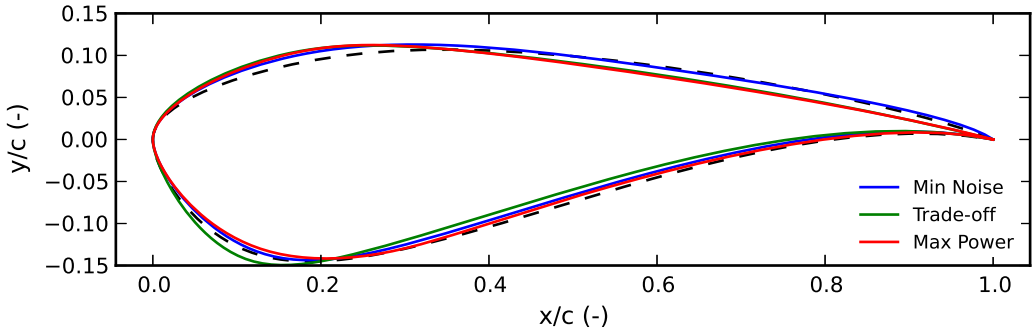


Figure 7.35: Initial and optimized airfoil shapes at 40% of the blade in the AOC 15/50 blade optimization

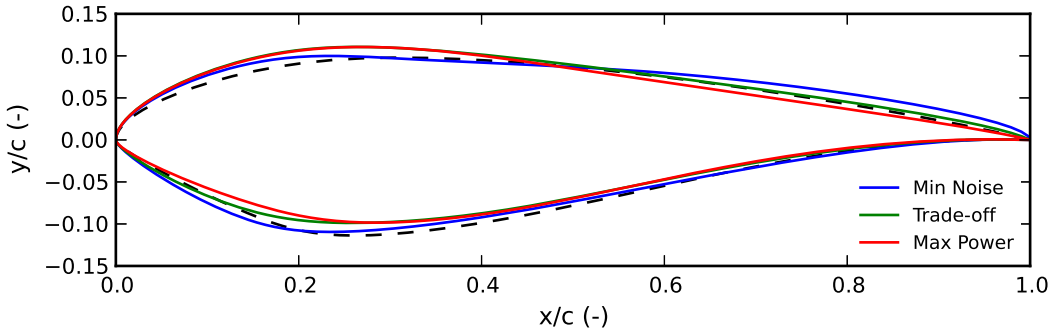


Figure 7.36: Initial and optimized airfoil shapes at 75% of the blade in the AOC 15/50 blade optimization.

Two slight unwanted inflections were obtained in the upper curve of the Minimum Noise airfoil at 75% of the blade and in the lower curve of th Minimum Noise airfoil at 95% of the blade. These were a result of the constraints defined in the optimization not being able to prevent this. To correct this, the constraint should be corrected to guarantee that control point 3 would never have an y-coordinate value smaller than the linear interpolation between control points 4 and 2, at its x-coordinate,

$$y_3^{cp} \geqslant y_4^{cp} + (y_2^{cp} - y_4^{cp}) \frac{x_3^{cp} - x_4^{cp}}{x_2^{cp} - x_4^{cp}}. \tag{7.10}$$

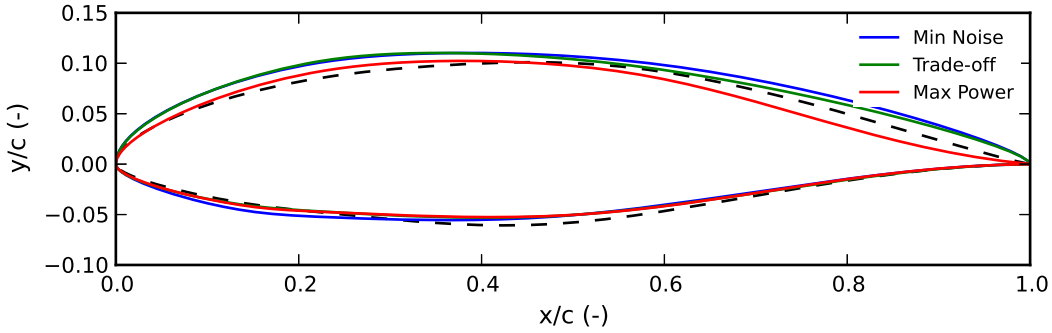
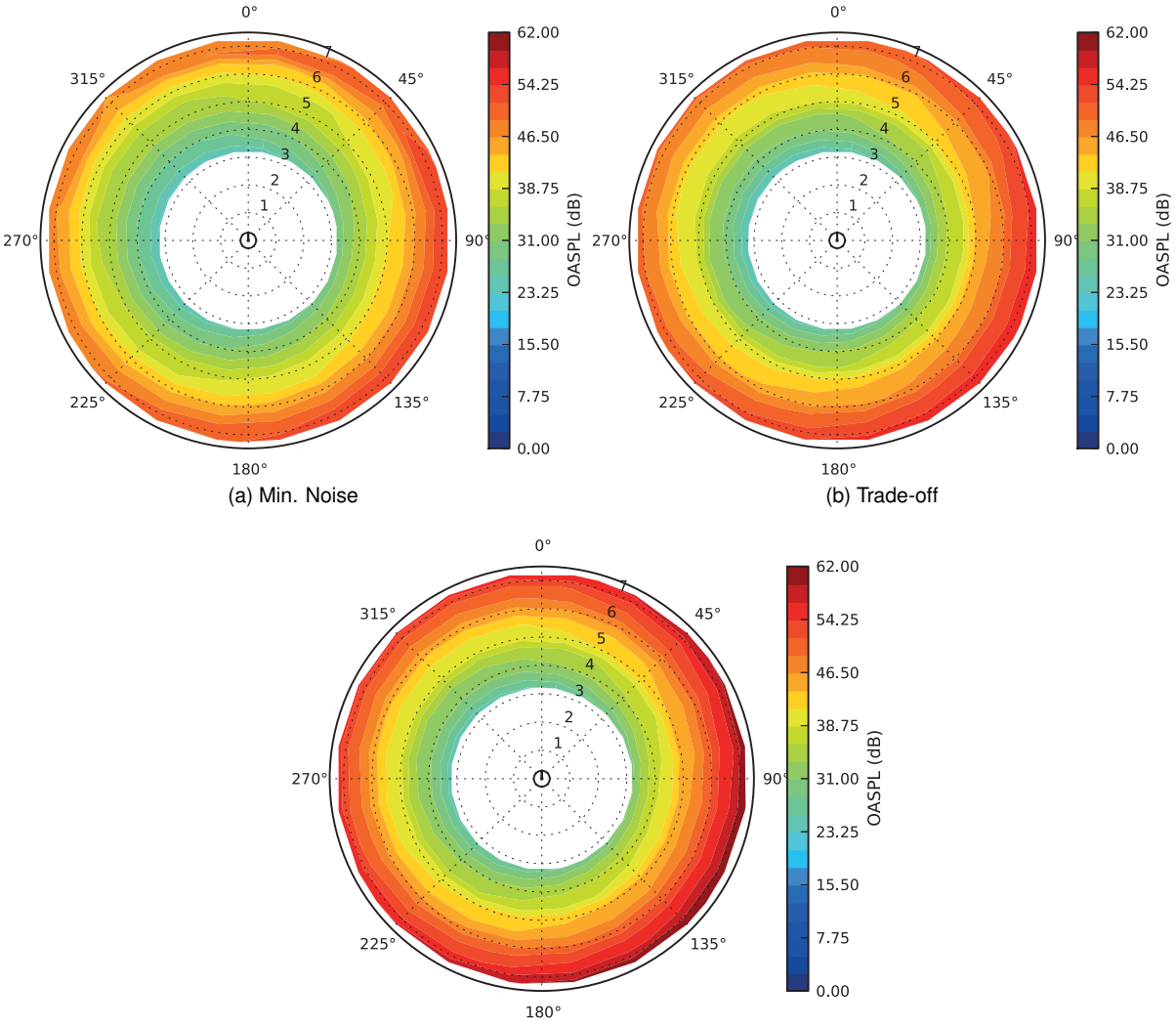


Figure 7.37: Initial and optimized airfoil shapes at 95% of the blade in the AOC 15/50 blade optimization

The same applies to control point 9, with the linear interpolation being between control points 8 and 10,

$$y_9^{cp} \leqslant y_8^{cp} + (y_{10}^{cp} - y_8^{cp}) \frac{x_9^{cp} - x_8^{cp}}{x_{10}^{cp} - x_8^{cp}}. \tag{7.11}$$

In Fig. 7.38, the the sound levels across the rotor generated by each of the optimized solutions is presented. The noise is mainly produced in the descent movement of the blade, similarly to what is shown in Fig. 7.27, and the reduction of noise from the maximum power solution to the minimum noise occurs specially in the outer part of the rotor. This is also seen in Fig. 7.39, where the radial distribution of generated noise is presented for the same three solutions.



(a) Min. Noise

(b) Trade-off

(c) Max. Power

Figure 7.38: Overall sound pressure level across the rotor for different AOC 15/50 optimized blade geometries.

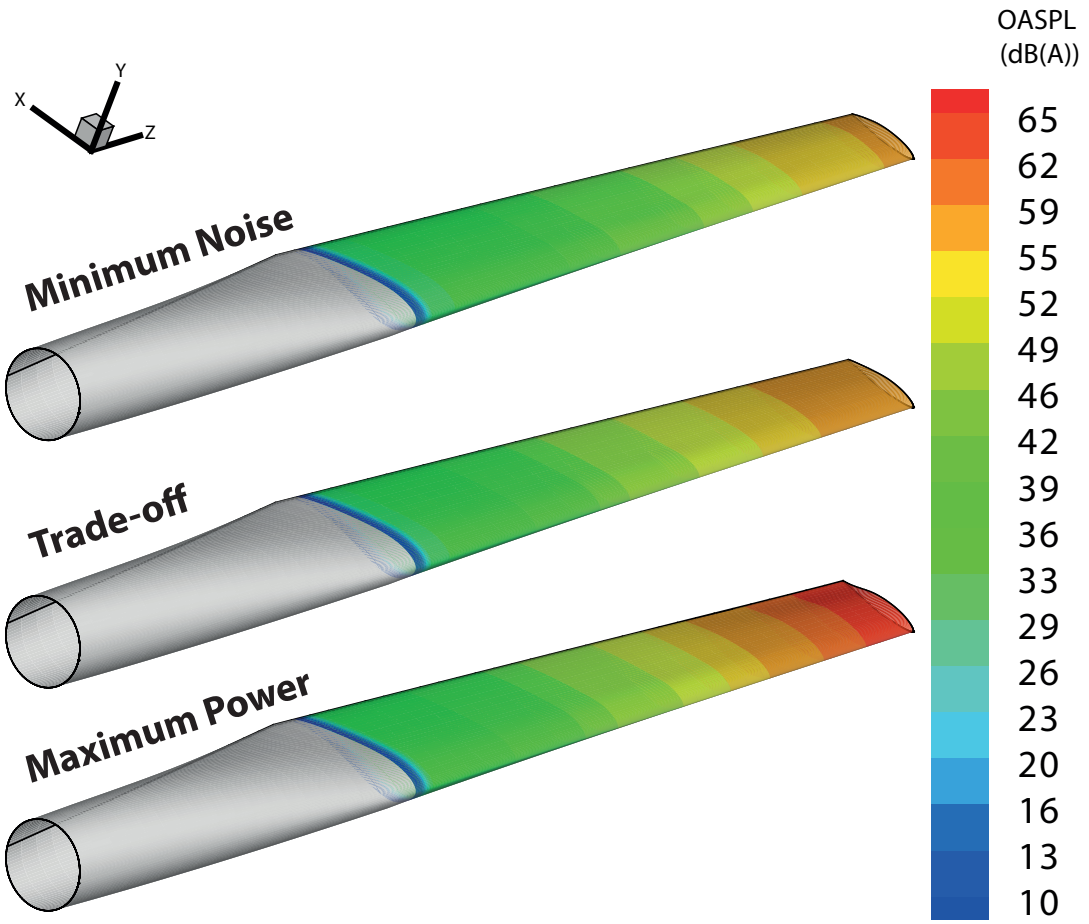


Figure 7.39: Comparison of the radial distribution of generated noise on between different optimized AOC 15/50 blade geometries.

Chapter 8

Conclusions

In this chapter, a summary of what was achieved during this work is presented, followed by some suggestions for the improvement of the aeroacoustic prediction and design framework.

8.1 Achievements

In the present work, a wind turbine aerodynamic and aeroacoustic prediction model was successfully implemented and validated against experimental data. The code developed is flexible, with various inputs and options, allowing the control of the simulations in detail. It is able to output the results of the simulations in various formats for post-processing.

A geometrical model of the wind turbine blade was also developed and implemented, using NURBS curves and Bezier curves, for the definition of the cross sectional airfoil shapes and twist and chord distributions, respectively. The NURBS parameterization of the airfoil shapes proved to be a good approach, as it was able to accurately reproduce various different airfoil shapes commonly used in wind turbines.

The code was successfully implemented into an optimization framework, being able to produce optimal solutions in the various performed optimizations. In the optimizations performed on the NREL Phase II wind turbine, a maximum increase in AEP of 139.4 % and maximum reduction of OASPL in 10.7 % was achieved. This large relative improvements showed that this particular blade was far from being optimal. In contrast, using another blade, from the commercially available AOC 15/50 wind turbine, it was possible to assess more realistically the potential of the optimization framework. In this case, the optimization resulted in a maximum improvement in AEP of 12.4 % and a maximum OASPL reduction of 7.6 %. However, these results were not achieved in the same blade geometry, and the trade-off between noise generation and energy production is visible in the results of the optimizations.

Nowadays, with wind turbine blade geometries being so optimized, tools like the framework developed are used in the design phase of any wind turbine to increase the wind turbines performance to the maximum possible extent, as it might make a big difference in terms of the commercial success of the wind turbine. The relative small computational requirement of each optimization is a key factor, as it

allows for a greater diversity of geometries and configurations to be analyzed / optimized, thus increasing the probability of obtaining a better solution.

8.2 Future Work

There were two issues that occurred during the optimizations that would be interesting to address. The first one is related to the constraints. It would be interesting to perform optimization of the blade with a new set of constraints and a larger search space, allowing for different airfoil shapes to be tested. The second was the numerical precision of the solutions saved by pyOpt's NSGA-II optimizer. The analysis of the final solutions given by the optimizer were not in total agreement with the values obtained during the optimization, thus indicating that more precision is needed in the data saved to files.

Regarding the wind turbine aerodynamic model itself, some improvements could be introduced. Moving to an unsteady BEM model or to a vortex wake method would improve the accuracy of the results and, coupled to a structural model, allow for the optimization of both aerodynamics, acoustics, and structural behavior or the wind turbine rotor.

Another improvement that would be interesting to see implemented in the code is the use of parallel processing to reduce the amount of CPU-time required by each optimization and thus allowing a higher refinement of the models or even the use of more complex models. This is rather simple to implement in a GA algorithm, as in each generation, the evaluation of the objective functions for each individual is done independently.

One last suggestion is to perform single objective optimizations with constraints on the generated noise, instead of multi-objective optimizations. The results of the optimizations provided Pareto fronts with solutions that, while producing a large increase in energy production, also increased the noise levels. As the maximum noise levels allowed are usually legislated (see chapter 1), a maximum noise level could be set as a constraint, thus forcing the optimizer to focus its search in a region where legal noise levels would be produced.

Appendix A

Coordinate Systems

In this appendix, the transformation matrices that allow a vector in one coordinate system to be expressed in another coordinate system are defined.

Starting with the transformation matrix between system 1 and system 2, $\mathbf{a_{12}}$, it is assumed that system 1 and 2 are identical with the exception of the position of the origins. System 2 is then rotated about the x-axis with the angle θ_{yaw}, giving the following transformation matrix:

$$\mathbf{a_1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{yaw} & \sin\theta_{yaw} \\ 0 & -\sin\theta_{yaw} & \cos\theta_{yaw} \end{bmatrix} \tag{A.1}$$

System 2 is rotated again, about its y-axis, with an angle θ_{yaw}, resulting in the transformation matrix

$$\mathbf{a}_2 = \begin{bmatrix} \cos\theta_{tilt} & 0 & -\sin\theta_{tilt} \\ 0 & 1 & 0 \\ \sin\theta_{tilt} & 0 & \cos\theta_{tilt} \end{bmatrix} \tag{A.2}$$

There is no rotation about the z-axis for this system, therefore, the transformation matrix for this axis is

$$\mathbf{a_3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.3}$$

The total transformation matrix between system 1 and system 2 is found with $\mathbf{a_{12}} = \mathbf{a_3} \cdot \mathbf{a_2} \cdot \mathbf{a_1}$.

The shaft is considered to be stiff, therefore the transformation between system 2 and system 3

consists only in a rotation about the z-axis of angle θ_{wing}:

$$\mathbf{a_{23}} = \begin{bmatrix} \cos\theta_{wing} & \sin\theta_{wing} & 0 \\ -\sin\theta_{wing} & \cos\theta_{wing} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.4}$$

Again, system 4 only suffers one rotation, about its y-axis with angle θ_{cone}:

$$\mathbf{a_{34}} = \begin{bmatrix} \cos\theta_{cone} & 0 & -\sin\theta_{cone} \\ 0 & 1 & 0 \\ \sin\theta_{cone} & 0 & \cos\theta_{cone} \end{bmatrix} \tag{A.5}$$

The transformation matrix between system 4 and system 5 is obtained by considering a rotation about its x-axis with a twist angle θ_{twist}:

$$\mathbf{a_{45}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{twist} & \sin\theta_{twist} \\ 0 & -\sin\theta_{twist} & \cos\theta_{twist} \end{bmatrix} \tag{A.6}$$

As an example, to transform a vector in system 5 to system 1 the following transformation is applied to the vector:

$$\mathbf{r_1}=\mathbf{a_{21}} \cdot \mathbf{a_{32}} \cdot \mathbf{a_{43}} \cdot \mathbf{a_{54}} \cdot \mathbf{r_5} \tag{A.7}$$

or

$$\mathbf{r_1} = \mathbf{a_{12}}^{\mathbf{T}} \cdot \mathbf{a_{23}}^{\mathbf{T}} \cdot \mathbf{a_{34}}^{\mathbf{T}} \cdot \mathbf{a_{45}}^{\mathbf{T}} \cdot \mathbf{r_5} \tag{A.8}$$

Appendix B

Noise Model Equations and Functions

In this appendix, the functions necessary for the computation of the noise model presented in chapter 3 are defined, as given by Brooks et al. (1989).

B.1 TBL-TE

The spectral shape function A is defined as

$$A(a) = A_{\min}(a) + A_R(a_0)\left[A_{\max}(a) - A_{\min}(a)\right]$$

where

$$A_{\min}(a) = \begin{cases} \sqrt{67.552 - 886.788a^2} - 8.219 & (a < 0.204) \\ -32.665a + 3.981 & (0.204 \leqslant a \leqslant 0.244) \\ -142.795a^3 + 103.656a^2 - 57.757a + 6.006 & (0.244 < a) \end{cases},$$

$$A_{\max}(a) = \begin{cases} \sqrt{67.552 - 886.788a^2} - 8.219 & (a < 0.13) \\ -15.901a + 1.098 & (0.13 \leqslant a \leqslant 0.321) \\ -4.669a^3 + 3.491a^2 - 16.699a + 1.149 & (0.321 < a) \end{cases}$$

and

$$A_R(a_0) = \frac{-20 - A_{\min}(a_0)}{A_{\max}(a_0) - A_{\min}(a_0)}.$$

The terms a and a_0 are defined as

$$a = \left|\log\left(\mathrm{St}/\mathrm{St}_{\mathsf{peak}}\right)\right|$$

and

$$a_0\left(R_c\right) = \begin{cases} 0.57 & \left(R_c < 9.52 \times 10^4\right) \\ \left(-9.57 \times 10^{-13}\right)\left(R_c - 8.57 \times 10^5\right)^2 + 1.13 & \left(9.52 \times 10^4 \leqslant R_c \leqslant 8.57 \times 10^5\right) \\ 1.13 & \left(8.57 \times 10^5 < R_c\right) \end{cases}.$$

Similarly to A, the spectral shape function B is defined as

$$B\left(b\right) = B_{\min}\left(b\right) + B_R\left(b_0\right)\left[B_{\max}\left(b\right) - B_{\min}\left(b\right)\right],$$

where

$$B_{\min}\left(b\right) = \begin{cases} \sqrt{16.888 - 886.788b^2} - 4.109 & \left(b < 0.13\right) \\ -83.607b + 8.138 & \left(0.13 \leqslant b \leqslant 0.145\right) \\ -817.810b^3 + 355.210b^2 - 135.024b + 10.619 & \left(0.145 < b\right) \end{cases},$$

$$B_{\max}\left(b\right) = \begin{cases} \sqrt{16.888 - 886.788b^2} - 4.109 & \left(b < 0.10\right) \\ -31.330b + 1.854 & \left(0.10 \leqslant b \leqslant 0.187\right) \\ -80.541b^3 + 44.174b^2 - 39.381b + 2.344 & \left(0.187 < b\right) \end{cases}$$

and

$$B_R\left(b_0\right) = \frac{-20 - B_{\min}\left(b_0\right)}{B_{\max}\left(b_0\right) - B_{\min}\left(b_0\right)}.$$

The terms b and b_0 are defined as

$$b = \left|\log\left(\mathrm{St}_s/\mathrm{St}_2\right)\right|$$

and

$$b_0\left(R_c\right) = \begin{cases} 0.30 & \left(R_c < 9.52 \times 10^4\right) \\ \left(-4.48 \times 10^{-13}\right)\left(R_c - 8.57 \times 10^5\right)^2 + 0.56 & \left(9.52 \times 10^4 \leqslant R_c \leqslant 8.57 \times 10^5\right) \\ 0.56 & \left(8.57 \times 10^5 < R_c\right) \end{cases}.$$

The amplitude function K_1 is defined as

$$K_1 = \begin{cases} -4.31\log\left(R_c\right) + 156.3 & \left(R_c < 2.47 \times 10^5\right) \\ -9.0\log\left(R_c\right) + 181.6 & \left(2.47 \times 10^5 \leqslant R_c \leqslant 8.0 \times 10^5\right) \\ 128.5 & \left(8.0 \times 10^5 < R_c\right) \end{cases}$$

and the adjustment for the pressure-side contribution for nonzero angles of attac ΔK_1 is given as

$$\Delta K_1 = \begin{cases} \alpha_* \left[1.43 \log\left(R_{\delta_p^*}\right)\right] - 5.29 & \left(R_{\delta_p^*} \leqslant 5000\right) \\ 0 & \left(5000 < R_{\delta_p^*}\right) \end{cases}.$$

The amplitude function K_2 is defined as

$$K_2 = K_1 + \begin{cases} -1000 & (\alpha_* < \gamma_0 - \gamma) \\ \sqrt{\beta^2 - (\beta/\gamma)^2 (\alpha_* - \gamma_0)^2} + \beta_0 & (\gamma_0 - \gamma \leqslant \alpha_* \leqslant \gamma_0 + \gamma) \\ -12 & (\gamma_0 + \gamma < \alpha_*) \end{cases},$$

where

$$\left.\begin{aligned} \gamma &= 27.094M + 3.31\gamma_0 = 23.43M + 4.651 \\ \beta &= 72.65M + 10.74\beta_0 = -34.19 - 13.82 \end{aligned}\right\}.$$

B.2 LBL-VS

The spectral shape functions G_1, G_2 and G_3 are defined as

$$G_1(e) = \begin{cases} 39.8 \log(e) - 11.12 & (e \leqslant 0.5974) \\ 98.409 \log(e) + 2.0 & (0.5974 < e \leqslant 0.8545) \\ -5.076 + \sqrt{2.484 - 506.25[\log(e)]^2} & (0.8545 < e \leqslant 1.17) \\ -98.409 \log(e) + 2.0 & (1.17 < e \leqslant 1.674) \\ -39.8 - 11.12 & (1.674 < e) \end{cases},$$

where $e = \mathrm{St}'/\mathrm{St}'_{\mathrm{peak}}$,

$$G_2(d) = \begin{cases} 77.852 \log(d) + 25.328 (d \leqslant 0.3237) \\ 65.188 \log(d) + 9.125 (0.3237 < d \leqslant 0.5689) \\ -114.052[\log(d)]^2 (0.5689 < d \leqslant 1.7579) \\ -65.188 \log(d) + 9.125 (1.7579 < d \leqslant 3.0889) \\ -77.852 + 15.328 (3.0889 < d) \end{cases},$$

where $d = R_c/(R_C)_0$,and

$$G_3(\alpha_*) = 171.04 - 3.03\alpha_*.$$

B.3 TEB-VS

The function $(G_5)_{\Psi=14^\circ}$ is defined as

$$(G_5)_{\Psi=14^\circ} = \begin{cases} m\eta + k & (\eta < \eta_0) \\ 2.5\sqrt{1-(\eta/\mu)^2} - 2.5 & (\eta_0 \leqslant \eta < 0) \\ \sqrt{1.5625 - 1194.99\eta^2} & (0 \leqslant \eta < 0.03616) \\ -155.543\eta + 4.375 & (0.03616 \leqslant \eta) \end{cases},$$

with

$$\eta = \log\left(\mathrm{S}\,t'''/\mathrm{S}\,t'''_{\mathrm{peak}},\right)$$

$$\mu = \begin{cases} 0.1221 & \left(h/\delta^*_{avg} < 0.25\right) \\ -0.2175\left(h/\delta^*_{avg}\right) + 0.1755 & \left(0.25 \leqslant h/\delta^*_{avg} < 0.62\right) \\ -0.0308\left(h/\delta^*_{avg}\right) + 0.0596 & \left(0.62 \leqslant h/\delta^*_{avg} < 1.15\right) \\ 0.0242 & \left(1.15 \leqslant h/\delta^*_{avg}\right) \end{cases},$$

$$m = \begin{cases} 0 & \left(h/\delta^*_{avg} \leqslant 0.02\right) \\ 68.724\left(h/\delta^*_{avg}\right) - 1.35 & \left(0.02 < h/\delta^*_{avg} \leqslant 0.5\right) \\ 308.475\left(h/\delta^*_{avg}\right) - 121.23 & \left(0.5 < h/\delta^*_{avg} \leqslant 0.62\right) \\ 224.811\left(h/\delta^*_{avg}\right) - 69.35 & \left(0.62 < h/\delta^*_{avg} \leqslant 1.15\right) \\ 1583.28\left(h/\delta^*_{avg}\right) - 1631.59 & \left(1.15 < h/\delta^*_{avg} \leqslant 1.2\right) \\ 268.344 & \left(1.2 < h/\delta^*_{avg}\right) \end{cases}$$

and

$$k = 2.5\sqrt{1-\left(\frac{\eta_0}{\mu}\right)^2} - 2.5 - m\eta_0,$$

where

$$\eta_0 = -\sqrt{\frac{m^2\mu^4}{6.25 + m^2\mu^2}}.$$

The function $(G_5)_{\Psi=0^\circ}$ is computed in the same way as $(G_5)_{\Psi=14^\circ}$ but replacing h/δ^*_{avg} by $\left(h/\delta^*_{avg}\right)'$, define as

$$\left(\frac{h}{\delta^*_{avg}}\right)' = 6.724\left(\frac{h}{\delta^*_{avg}}\right)^2 - 4.019\left(\frac{h}{\delta^*_{avg}}\right) + 1.107.$$

Appendix C

Sample Input Files

C.1 Rotor Definition File

```
#------------------------------ ROTOR DEFINITION ------------------------------#
 AOC 15/50 ROTOR                    # NAME
                          # Comments line 1
                          # Comments line 2
#------------------------------ Rotor CHARACTERISTICS -------------------------#
 3                          # NR.BLADES
 1.54                       # PITCH
 24.4                       # HUBHEIGHT
 0                          # CONE ANGLE (not used)
 0                          # TILT ANGLE (not used)
 0                          # YAW ANGLE  (not used)
#------------------------------ BLADE CHARACTERISTICS -------------------------#
 nurbs    # blade mode
 7.490    # Blade radius
 0.280    # Blade hub radius
#------------------------------ BLADE CONTROL SECTIOSN ------------------------#
 linear   # blade mode (fixed/linear/nurbs)
 5      # number of control sections
 0.2800    airfoils/NURBS/circle.DAT      ncircle
 2.9960    airfoils/NURBS/S821.DAT    nurbs
 5.6175    airfoils/NURBS/S819.DAT    nurbs
 7.1155    airfoils/NURBS/S820.DAT    nurbs
 7.4900    airfoils/NURBS/S820.DAT    nurbs
#------------------------------ BLADE CHORD CONTROL PNTS ----------------------#
 linear   # chord interpolation mode (linear/bezier)
 3
 0.280  0.494
 2.925  0.749
 7.49 0.436
#------------------------------ BLADE TWIST CONTROL PNTS ----------------------#
 bezier   # twist interpolation mode (linear/bezier)
 4
 0.2800000000     7.0000000000
 2.9960000000     4.3631067961
 5.9920000000     1.4543689320
 7.4900000000     0.0000000000
#------------------------------------ BLADE POLARS ----------------------------#
 blank    # left blank on purpose
 0      # number of polars
```

C.2 Rotor Analysis File

```
#------------------------------ ROTOR SIMULATION ------------------------------#
 AOC 15/50 ANANLYSIS                # SIMULATION NAME
#------------------------------ Atmospheric conditions ------------------------#
```

```
1.225          # DENSITY                                    [kg / m3]
20             # TEMPERATURE                                [deg Celcius]
1.4            # GAMMA                                      []
287            # R                                          [Jkg-1K-1]
1.46e-5        # KINEMATIC VISCOSITY                        [M2/SEC ]
340.4          # C0 (if -1 it is calculated)                [m/s]
0.08           # GROUND ROUGHNESS                           [m]
-1             # SHEAR EXP (if -1, it is calculated)        []
-1                    # TURBULENCE INTENSITY (-1: computed)          [%]
-1                    # TURBULENCE SCALE LENGHT ""                   [m]
#------------------------------ Xfoil/Rfoil settings ------------------------------#
9              # NCRIT
1.0            # XTR_P - Position of transition at pressure side
1.0            # XTR_S - Position of transition at suction side
0.005          # VACC
True           # PANE
#-------------------------------- Polar settings ----------------------------------#
0              # 0 - Xfoil / 1 - Rfoil (for BL only)
-1             # A_NEG
10             # A_POS
1              # A_STEP
1.5e6          # REYNOLDS (reynolds at which polars are computed, comma separated)
151            # NUMPER OF AIRFOIL POINTS
sin            # POINT DISTRIBUTION (unif, sin, cos)
5              # NUMBER OF POLARS
sect           # POLAR DISTRIBUTION (sect, unif, sin, cos)
True           # CORRECT_3D (True/False)
#---------------------------------- BEM settings ----------------------------------#
0              # BEM DEBUG
50             # NR_OF_ELEMENTS
sin            # ELEMENT_DISTRIBUTION
4              # AZIMUTAL DIVISIONS
1              # TIPLOSS
1              # HUBLOSS
6.48           # WEI_A
1.99           # WEI_K
#--------------------------------- Noise settings ---------------------------------#
True           # RUN_NOISE
0              # NOISE DEBUG
10             # NOISE_ELEMENTS
unif           # NOISE_ELEMENT_DISTRIBUTION
0.4            # NOISE COMPUTATION START RADIUS (0-1.0)
100            # NOISE COMPUTATION AoA CUTOFF (degree)
3              # BL_METHOD (0= BPM, no trip; 1=BPM light trip; 2= BPM eavy trip; 3 = xfoil; 4 = rfoil)
True           # I_TBLTE
True           # I_LBLVS
True           # I_BLUNT
True           # I_TIP
2              # I_TIN (0 - None; 1 - Lowsons model; 2 - Amiet's model)
1              # TIN_METHOD (0 - Off; 1 - Use simplified guidati's method)
1              # BLUNT_MODE (0 - from foil; 1-from input (%); 2-direct value)
0.75           # C2_PER (1-x_ac)
6.0            # TE_ANGLE (angle in degree)
1.0            # TE_PTHICK (percentage of chord)
True           # ROUNDED TIP - LOGICAL
1.0            # TIP LIFT CURVE SLOPE
0 0 -32.5          # OBS_POSITION (X Y Z)
runs/marcolini.freq  # FREQUENCIES FILE
#---------------------------------- Operation -------------------------------------#
4              # U_START                                    [m/s]
25             # U_END                                      [m/s]
1              # U_STEPS                                    [m/s]
64.6           # RPM                                        [rpm]
6              # NOISE_VEL                                  [m/s]
```

Appendix D

Sample Optimization Script

```
# Optimization 10
# Optimizes Control Sections + Bezier twist + linear chord
# Objective: AEP + OASPL
# Number of variables 18*3 + 6 + 2
# Algorithm: NSGA2

#------------------------------------------#
# Import necessary libraries
from Analysis import *
from pyOpt import *
from Geometry import *

import numpy as np
import datetime
from zipfile import *
import datetime

#------------------------------------------#
# Filename / Name definitions

analyFile = 'run/AOC_OPT_10.anl'
rotorFile = 'run/AOC_NLB3.rot'

output_name = 'opt_10c'

opt_name = 'Optimization - 10 - control sections + bezier twist + linear chord - AEP + OASPL'

PopSize = 68
maxGen = 50

HOTSTART = False

#------------------------------------------#
# Load Analysis file
anl = Analysis()
anl.load(analyFile)
rotor = Rotor()
rotor.load(rotorFile)
#------------------------------------------#
# Define x vector to blade geometry converter
def x_to_blade(blade,x,g=None):

    # chord values
    blade.chord[1] = x[0]
    blade.chord[2] = x[1]

    # twist values
    blade.twist[0] = x[2]
    blade.twist[1] = x[3]
    blade.twist[2] = x[4]
    blade.twist[3] = x[5]

    # chord position
    blade.twist_radius[1] = x[6] * blade.maxR
```

```
    blade.twist_radius[2] = x[7] * blade.maxR

    t = 8

    # sections parameters
    up_cntrl = np.array([[1.0, x[t+0], x[t+1], x[t+2], x[t+3], 0.000, 0.000],
                         [0.000, x[t+8], x[t+9], x[t+10], x[t+11], x[t+12],   0.00],
                         [0.0,0.0,0.0,0.0,0.0,0.0,0.0],
                         [1.0,1.0,1.0,1.0,1.0,1.0,1.0]])
    lo_cntrl = np.array([[0.000, 0.000, x[t+4], x[t+5], x[t+6], x[t+7], 1.0],
                         [0.00,  x[t+13],   x[t+14], x[t+15], x[t+16], x[t+17], -0.000],
                         [0.0,0.0,0.0,0.0,0.0,0.0,0.0],
                         [1.0,1.0,1.0,1.0,1.0,1.0,1.0]])
    blade.params[1] = [up_cntrl,init_blade.params[1][1],lo_cntrl,init_blade.params[1][3]]
    t = t + 18

    tg = 0
    if g is not None:
        g[tg+0] = up_cntrl[0,1] - up_cntrl[0,0] # x1 < x0
        g[tg+1] = up_cntrl[0,2] - up_cntrl[0,1] # x2 < x1
        g[tg+2] = up_cntrl[0,3] - up_cntrl[0,2] # x3 < x2
        g[tg+3] = up_cntrl[0,4] - up_cntrl[0,3] # x4 < x3
        g[tg+4] = up_cntrl[0,5] - up_cntrl[0,4] # x5 < x4

        g[tg+5] = lo_cntrl[0,1] - lo_cntrl[0,2] # x7 < x8
        g[tg+6] = lo_cntrl[0,2] - lo_cntrl[0,3] # x8 < x9
        g[tg+7] = lo_cntrl[0,3] - lo_cntrl[0,4] # x9 < x10
        g[tg+8] = lo_cntrl[0,4] - lo_cntrl[0,5] # x10 < x11
        g[tg+9] = lo_cntrl[0,5] - lo_cntrl[0,6] # x11 < x12


        g[tg+10] = up_cntrl[1,2] - up_cntrl[1,3] # y2 < y3
        g[tg+11] = lo_cntrl[1,4] - lo_cntrl[1,5] # y10 < y11

        tg += 12


    up_cntrl = np.array([[1.0, x[t+0], x[t+1], x[t+2], x[t+3], 0.000, 0.000],
                         [0.000, x[t+8], x[t+9], x[t+10], x[t+11], x[t+12],   0.00],
                         [0.0,0.0,0.0,0.0,0.0,0.0,0.0],
                         [1.0,1.0,1.0,1.0,1.0,1.0,1.0]])
    lo_cntrl = np.array([[0.000, 0.000, x[t+4], x[t+5], x[t+6], x[t+7], 1.0],
                         [0.00,  x[t+13],   x[t+14], x[t+15], x[t+16], x[t+17], -0.000],
                         [0.0,0.0,0.0,0.0,0.0,0.0,0.0],
                         [1.0,1.0,1.0,1.0,1.0,1.0,1.0]])
    blade.params[2] = [up_cntrl,init_blade.params[2][1],lo_cntrl,init_blade.params[2][3]]
    t = t+18

    if g is not None:
        g[tg+0] = up_cntrl[0,1] - up_cntrl[0,0] # x1 < x0
        g[tg+1] = up_cntrl[0,2] - up_cntrl[0,1] # x2 < x1
        g[tg+2] = up_cntrl[0,3] - up_cntrl[0,2] # x3 < x2
        g[tg+3] = up_cntrl[0,4] - up_cntrl[0,3] # x4 < x3
        g[tg+4] = up_cntrl[0,5] - up_cntrl[0,4] # x5 < x4

        g[tg+5] = lo_cntrl[0,1] - lo_cntrl[0,2] # x7 < x8
        g[tg+6] = lo_cntrl[0,2] - lo_cntrl[0,3] # x8 < x9
        g[tg+7] = lo_cntrl[0,3] - lo_cntrl[0,4] # x9 < x10
        g[tg+8] = lo_cntrl[0,4] - lo_cntrl[0,5] # x10 < x11
        g[tg+9] = lo_cntrl[0,5] - lo_cntrl[0,6] # x11 < x12

        g[tg+10] = up_cntrl[1,2] - up_cntrl[1,3] # y2 < y3
        g[tg+11] = lo_cntrl[1,4] - lo_cntrl[1,5] # y10 < y11

        tg += 12

    up_cntrl = np.array([[1.0, x[t+0], x[t+1], x[t+2], x[t+3], 0.000, 0.000],
                         [0.000, x[t+8], x[t+9], x[t+10], x[t+11], x[t+12],   0.00],
                         [0.0,0.0,0.0,0.0,0.0,0.0,0.0],
                         [1.0,1.0,1.0,1.0,1.0,1.0,1.0]])
    lo_cntrl = np.array([[0.000, 0.000, x[t+4], x[t+5], x[t+6], x[t+7], 1.0],
                         [0.00,  x[t+13],   x[t+14], x[t+15], x[t+16], x[t+17], -0.000],
                         [0.0,0.0,0.0,0.0,0.0,0.0,0.0],
                         [1.0,1.0,1.0,1.0,1.0,1.0,1.0]])
    blade.params[3] = [up_cntrl,init_blade.params[3][1],lo_cntrl,init_blade.params[3][3]]

    if g is not None:
```

```
132             g[tg+0] = up_cntrl[0,1] - up_cntrl[0,0] # x1 < x0
133             g[tg+1] = up_cntrl[0,2] - up_cntrl[0,1] # x2 < x1
134             g[tg+2] = up_cntrl[0,3] - up_cntrl[0,2] # x3 < x2
135             g[tg+3] = up_cntrl[0,4] - up_cntrl[0,3] # x4 < x3
136             g[tg+4] = up_cntrl[0,5] - up_cntrl[0,4] # x5 < x4
137
138             g[tg+5] = lo_cntrl[0,1] - lo_cntrl[0,2] # x7 < x8
139             g[tg+6] = lo_cntrl[0,2] - lo_cntrl[0,3] # x8 < x9
140             g[tg+7] = lo_cntrl[0,3] - lo_cntrl[0,4] # x9 < x10
141             g[tg+8] = lo_cntrl[0,4] - lo_cntrl[0,5] # x10 < x11
142             g[tg+9] = lo_cntrl[0,5] - lo_cntrl[0,6] # x11 < x12
143
144             g[tg+10] = up_cntrl[1,2] - up_cntrl[1,3] # y2 < y3
145             g[tg+11] = lo_cntrl[1,4] - lo_cntrl[1,5] # y10 < y11
146
147             tg += 12
148
149         blade.params[4] = [up_cntrl, init_blade.params[3][1], lo_cntrl, init_blade.params[3][3]]
150
151         blade.update()
152
153         if g is not None:
154             # chord constrains
155             g[tg] = blade.chord[2] - blade.chord[1]
156             tg +=1
157
158             # twist constrains
159             for i in range(blade.twist_count-1):
160                 g[tg] = blade.twist[i+1] - blade.twist[i]
161                 g[tg+1] = blade.twist_mu[i] - blade.twist_mu[i+1]
162                 tg = tg+2
163
164         return tg
165
166
167 #-----------------------------------------#
168 # Define objective function
169 count = 0
170 def objfunc(x):
171     global count, anl
172
173
174     count += 1
175
176     # initialization
177     rotor = Rotor()
178     rotor.load(rotorFile)
179     blade = rotor.blade
180
181     t = 0
182     f = [0.0]*2
183
184     chord_cons = 1
185     twist_cons = 2 * (blade.twist_count-1)
186     sects_cons = 12 * 3
187     other_cons = 1
188
189     g = [0.0]*(chord_cons+twist_cons+sects_cons+other_cons)
190
191     tg = x_to_blade(blade,x,g)
192
193     AEP, azim_results, results, fail = anl.BEM(rotor)
194     if fail == 0:
195         ospl, oaspl,spl, spla, powa, noise_err = anl.NOISE(rotor,azim_results)
196     else:
197         oaspl = [200.00]
198
199     f[0] = -AEP / 10**6
200     f[1] = oaspl[0]
201
202     # non-negative aep constrain
203     g[tg] = f[0]
204
205     print f
206
207     return f,g,fail
208
```

```
209 #-------------------------------------------------#
210 # Define pyOpt optimization problem
211 opt_prob = Optimization ( opt_name , objfunc )
212
213 init_rotor = Rotor()
214 init_rotor.load(rotorFile)
215 init_blade = init_rotor.blade
216
217 #------------- Add Objective ------------------#
218 opt_prob.addObj('AEP')
219 opt_prob.addObj('OASPL')
220 #------------- Add Variables ------------------#
221
222 lower_chord = np.array(init_blade.chord) - 0.2 * abs(np.array(init_blade.chord))
223 upper_chord = np.array(init_blade.chord) + 0.2 * abs(np.array(init_blade.chord))
224
225 opt_prob.addVar('cyy2', 'c', lower = lower_chord[1], upper = upper_chord[1], value = init_blade.chord[1])
226 opt_prob.addVar('cyy3', 'c', lower = lower_chord[2], upper = upper_chord[2], value = init_blade.chord[2])
227
228 lower_twist = np.array(init_blade.twist) - 0.1 * abs(np.array(init_blade.twist))
229 upper_twist = np.array(init_blade.twist) + 0.1 * abs(np.array(init_blade.twist))
230
231 opt_prob.addVar('tyy1', 'c', lower = lower_twist[0], upper = upper_twist[0], value = init_blade.twist[0])
232 opt_prob.addVar('tyy2', 'c', lower = lower_twist[1], upper = upper_twist[1], value = init_blade.twist[1])
233 opt_prob.addVar('tyy3', 'c', lower = lower_twist[2], upper = upper_twist[2], value = init_blade.twist[2])
234 opt_prob.addVar('tyy4', 'c', lower = lower_twist[3], upper = upper_twist[3], value = init_blade.twist[3])
235
236 lower_twist = np.array(init_blade.twist_mu) - 0.3 * abs(np.array(init_blade.twist_mu))
237 upper_twist = np.array(init_blade.twist_mu) + 0.3 * abs(np.array(init_blade.twist_mu))
238
239 opt_prob.addVar('txx2', 'c', lower = lower_twist[1], upper = upper_twist[1], value = init_blade.twist_mu[1])
240 opt_prob.addVar('txx3', 'c', lower = lower_twist[2], upper = upper_twist[2], value = init_blade.twist_mu[2])
241
242 for j in range(init_blade.sects_count-2):
243
244     k = j+1
245
246     vari_x = 0.1
247     vari_y = 0.2
248
249     # Upper curve x
250     lower_up_x = np.maximum(init_blade.params[k][0][0,:] - abs(init_blade.params[k][0][0,:]) * vari_x,0.0)
251     upper_up_x = np.minimum(init_blade.params[k][0][0,:] + abs(init_blade.params[k][0][0,:]) * vari_x,1.0)
252     # Lower curve x
253     lower_lo_x = np.maximum(init_blade.params[k][2][0,:] - abs(init_blade.params[k][2][0,:]) * vari_x,0.0)
254     upper_lo_x = np.minimum(init_blade.params[k][2][0,:] + abs(init_blade.params[k][2][0,:]) * vari_x,1.0)
255     # Upper curve y
256     lower_up_y = init_blade.params[k][0][1,:] - abs(init_blade.params[k][0][1,:]) * vari_y
257     upper_up_y = init_blade.params[k][0][1,:] + abs(init_blade.params[k][0][1,:]) * vari_y
258     # Lower curve y
259     lower_lo_y = init_blade.params[k][2][1,:] - abs(init_blade.params[k][2][1,:]) * vari_y
260     upper_lo_y = init_blade.params[k][2][1,:] + abs(init_blade.params[k][2][1,:]) * vari_y
261
262
263     opt_prob.addVar(''.join(['S',str(k),'xx1']), 'c',  lower = lower_up_x[1], upper = upper_up_x[1], value = init_blade.params[k][0][0,1])
264     opt_prob.addVar(''.join(['S',str(k),'xx2']), 'c',  lower = lower_up_x[2], upper = upper_up_x[2], value = init_blade.params[k][0][0,2])
265     opt_prob.addVar(''.join(['S',str(k),'xx3']), 'c',  lower = lower_up_x[3], upper = upper_up_x[3], value = init_blade.params[k][0][0,3])
266     opt_prob.addVar(''.join(['S',str(k),'xx4']), 'c',  lower = lower_up_x[4], upper = upper_up_x[4], value = init_blade.params[k][0][0,4])
267
268     opt_prob.addVar(''.join(['S',str(k),'xx8']), 'c',  lower = lower_lo_x[2], upper = upper_lo_x[2], value = init_blade.params[k][2][0,2])
269     opt_prob.addVar(''.join(['S',str(k),'xx9']), 'c',  lower = lower_lo_x[3], upper = upper_lo_x[3], value = init_blade.params[k][2][0,3])
270     opt_prob.addVar(''.join(['S',str(k),'xx10']), 'c', lower = lower_lo_x[4], upper = upper_lo_x[4], value = init_blade.params[k][2][0,4])
271     opt_prob.addVar(''.join(['S',str(k),'xx11']), 'c', lower = lower_lo_x[5], upper = upper_lo_x[5], value = init_blade.params[k][2][0,5])
272
273     opt_prob.addVar(''.join(['S',str(k),'yy1']), 'c',  lower = lower_up_y[1], upper = upper_up_y[1], value = init_blade.params[k][0][1,1])
274     opt_prob.addVar(''.join(['S',str(k),'yy2']), 'c',  lower = lower_up_y[2], upper = upper_up_y[2], value = init_blade.params[k][0][1,2])
275     opt_prob.addVar(''.join(['S',str(k),'yy3']), 'c',  lower = lower_up_y[3], upper = upper_up_y[3], value = init_blade.params[k][0][1,3])
276     opt_prob.addVar(''.join(['S',str(k),'yy4']), 'c',  lower = lower_up_y[4], upper = upper_up_y[4], value = init_blade.params[k][0][1,4])
277     opt_prob.addVar(''.join(['S',str(k),'yy5']), 'c',  lower = lower_up_y[5], upper = upper_up_y[5], value = init_blade.params[k][0][1,5])
278
279     opt_prob.addVar(''.join(['S',str(k),'yy7']), 'c',  lower = lower_lo_y[1], upper = upper_lo_y[1], value = init_blade.params[k][2][1,1])
280     opt_prob.addVar(''.join(['S',str(k),'yy8']), 'c',  lower = lower_lo_y[2], upper = upper_lo_y[2], value = init_blade.params[k][2][1,2])
281     opt_prob.addVar(''.join(['S',str(k),'yy9']), 'c',  lower = lower_lo_y[3], upper = upper_lo_y[3], value = init_blade.params[k][2][1,3])
282     opt_prob.addVar(''.join(['S',str(k),'yy10']), 'c', lower = lower_lo_y[4], upper = upper_lo_y[4], value = init_blade.params[k][2][1,4])
283     opt_prob.addVar(''.join(['S',str(k),'yy11']), 'c', lower = lower_lo_y[5], upper = upper_lo_y[5], value = init_blade.params[k][2][1,5])
284
285 #------------- Add Constrains -----------------#
```

```
286 opt_prob.addConGroup('Sections',36,'i')
287
288 opt_prob.addCon('cg0', 'i')
289
290 opt_prob.addCon('tg0', 'i')
291 opt_prob.addCon('tg1', 'i')
292 opt_prob.addCon('tg2', 'i')
293 opt_prob.addCon('tg3', 'i')
294 opt_prob.addCon('tg4', 'i')
295 opt_prob.addCon('tg5', 'i')
296
297 opt_prob.addCon('fg0', 'i') # constrain for non-negative aep
298
299
300 print opt_prob
301
302 #-----------------------------------------#
303 # Run Optimization
304 nsga2 = NSGA2()
305 nsga2.setOption ( 'PopSize', PopSize) # multiple of 4
306 nsga2.setOption ( 'maxGen', maxGen)
307 [fstr1, xstr1, inform1] = nsga2(opt_prob, store_hst = output_name + '_' + str(PopSize) + '_' + str(maxGen)+ '_' + 'NSGA2', hot_start = HOTSTART)
308 print opt_prob.solution(0)
309
310
311 #-----------------------------------------#
312 # Save Optimization Files
313 f = open(output_name + '_sol.txt','w')
314 f.write(opt_prob.__str__())
315 f.write('\n')
316 f.write(opt_prob._solutions[0].__str__())
317 f.close()
318
319
320 #-----------------------------------------#
321 # Save Rotor
322 x_to_blade(init_blade,xstr1)
323 fnames1 = init_rotor.save(output_name + '_nsga.rot')
324
325 #-----------------------------------------#
326 # Save everything to zip file
327 with ZipFile(output_name + '_' + str(PopSize) + '_' + str(maxGen) + '.zip', 'w') as myzip:
328     myzip.write(output_name + '_' + str(PopSize) + '_' + str(maxGen)+ '_' + 'NSGA2.bin')
329     myzip.write(output_name + '_' + str(PopSize) + '_' + str(maxGen)+ '_' + 'NSGA2.cue')
330     myzip.write('nsga2_best_pop.out')
331     myzip.write('nsga2_final_pop.out')
332     myzip.write('nsga2_initial_pop.out')
333     myzip.write('nsga2_params.out')
334     myzip.write('nsga2_run.out')
335
336     myzip.write(output_name + '_sol.txt')
337     myzip.write(output_name + '_nsga.rot')
338
339     for name in fnames1:
340         myzip.write(name)
341
342     myzip.write(rotorFile)
343     myzip.write(analyFile)
344
345     myzip.close()
```

Bibliography

ISO 266. Acoustics - Preferred frequencies, 1997.

J. T. Alander. On optimal population size of genetic algorithms. In *CompEuro'92.'Computer Systems and Software Engineering', Proceedings.*, pages 65–70. IEEE, 1992.

R. K. Amiet. Acoustic radiation from an airfoil in a turbulent stream. *Journal of Sound and Vibration*, 41 (4):407–420, 1975.

Y. Bazilevs, M. C. Hsu, I. Akkerman, S. Wright, K. Takizawa, B. Henicke, T. Spielman, and T. E. Tezduyar. 3D simulation of wind turbine rotors at full scale. Part I: geometry modeling and aerodynamics. *International Journal for Numerical Methods in Fluids*, 65(1-3):207–235, 2011.

D. M. Beazley. SWIG: An easy to use tool for integrating scripting languages with C and C++. In *Proceedings of the 4th conference on USENIX Tcl/Tk Workshop, 1996 - Volume 4*, TCLTK'96, pages 15–15, Berkeley, CA, USA, 1996.

A. Bentamy, J. Y. Trépanier, and F. Guibault. Wing shape optimization using a constrained nurbs surface geometrical representation. In *ICAS Congress*, 2002.

N. Bizzarrini, F. Grasso, and DP Coiro. Numerical Optimization for High Efficiency, Low Noise Airfoils. 2011.

W. K. Blake. *Mechanics of Flow-induced Sound and Vibration: Complex flow-structure interactions*. Applied mathematics and mechanics. Academic Press, 1986. ISBN 9780121035020.

J. F. Bonnans, J. C. Gilbert, C. Lemarechal, and C. A. Sagastizábal. *Numerical Optimization: Theoretical and Practical Aspects*. Springer, 2006. ISBN 9783540354451.

K. Boorsma and JG Schepers. Enhanced wind turbine noise prediction tool SILANT. *Wind Energy*, 2012.

T. F. Brooks and T. H. Hodgson. Trailing edge noise prediction from measured surface pressures. *Journal of Sound and Vibration*, 78(1):69–117, 1981.

T. F. Brooks, D. S. Pope, and M. A. Marcolini. *Airfoil self-noise and prediction*. NASA reference publication 1218. National Aeronautics and Space Administration, USA, 1989.

M. L. Buhl. *A new empirical relationship between thrust coefficient and induction factor for the turbulent windmill state*. National Renewable Energy Laboratory, 2005.

M. L. Buhl. NWTC Design Codes (Wt_perf by Marshall Buhl), 2012. URL http://wind.nrel.gov/designcodes/simulators/wtperf/. accessed 09-October-2012.

T. Burton, D. Sharpe, N. Jenkins, and E. Bossanyi. *Wind Energy Handbook*. John Wiley & Sons, 2001. ISBN 9780471489979.

R. Carson. *Silent spring*. Fawcett Crest book. Fawcett, 1962.

J. F. G. Coimbra. Aero-Acoustic Optimization of Airfoils for Wind Turbines. Master's thesis, Instituto Superior Técnico, 2012.

W. D. Colby, R. Dobie, G. Leventhall, D. M. Lipscomb, R. J. McCunney, M. T. Seilo, and B. Søndergaard. *Wind turbine sound and health effects: An expert panel review*. American Wind Energy Association, 2009.

P. Costa. Atlas do potencial eólico para Portugal continental. Master's thesis, Faculdade de Ciências da Universidade de Lisboa, Lisboa, 2004.

J. Counihan. Adiabatic atmospheric boundary layers: A review and analysis of data from the period 1880-1972. *Atmospheric Environment (1967)*, 9(10):871 – 905, 1975. ISSN 0004-6981. doi: 10.1016/0004-6981(75)90088-8.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on*, 6(2):182–197, 2002.

M. Drela. Xfoil: An analysis and design system for low reynolds number airfoils. 1989.

M. Drela and M. Giles. Viscous-inviscid analysis of transonic and low reynolds number airfoils. *AIAA journal*, 25(10):1347–1355, 1987.

Z. Du and M. Selig. A 3-d stall-delay model for horizontal axis wind turbine performance prediction. In *Proceedings of the 1998 ASME Wind Energy Symposium, Reno, NV*, pages 9–19, 1998.

A. J. Eggers, K. Chaney, and R. Digumarthi. An assessment of approximate modeling of aerodynamic loads on the uae rotor. ASME, 2003.

J. E. Ffowes-Williams and L. H. Hall. Aerodynamic sound generation by turbulent flow in the vicinity of a scattering half plane. *Journal of Fluid Mechanics*, 40(04):657–670, 1970.

P. Fuglsang and H. A. Madsen. *Implementation and verification of an aeroacoustic noise prediction model for wind turbines*. 1996.

P. Gipe. *Wind energy comes of age*, volume 4. Wiley, 1995.

H. Glauert. Aerodynamic theory, vol. 4. *Berlin, Germany: Julius Springer*, pages 169–360, 1935.

T. Göçmen and B. Özerdem. Airfoil optimization for noise emission problem and aerodynamic performance criterion on small scale wind turbines. *Energy*, 2012.

D. E. Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley Longman Publishing, Boston, MA, 1989. ISBN 9780201157673.

M. E. Goldstein. *Aeroacoustics*. McGraw-Hill International Book Company, 1976. ISBN 9780070236851.

S. Gotshall and B. Rylander. Optimal population size and the genetic algorithm. *Population*, 100(400): 900, 2008.

F. Grasso. *Multi-Objective Numerical Optimization Applied to Aircraft Design*. PhD thesis, Università degli Studi di Napoli Federico II, 2008.

G. Guidati, R. Bareiss, S. Wagner, T. Dassen, and R. Parchen. Simulation and measurement of inflow-turbulence noise on airfoils. *AIAA Paper 97-1698*, 1997.

GWEC. Global statistics, 2011. URL `http://www.gwec.net/global-figures/graphs/`. accessed 8-August-2012.

M. O. L. Hansen. *Aerodynamics of Wind Turbines*. Taylor & Francis, 2012. ISBN 9781844074389.

H. H. Hubbard and K. P. Shepherd. Wind turbine acoustics research bibliography with selected annotation. 1988.

H. H. Hubbard and K. P. Shepherd. *Wind turbine acoustics*. NASA technical paper. National Aeronautics and Space Administration, 1990.

A. Huskey, H. F. Link, and C. P. Butterfield. Wind Turbine Generator System Acoustic Noise Test Report for the AOC 15/50 Wind Turbine. Technical report, AOC1550-CA-99182-1000, National Renewable Energy Laboratory, Golden, CO, 1999.

I. Ivanović, Z. Petrović, and S. Stupar. Helicopter Rotor Blade Shape Optimization Using NURBS for Airfoil Shape Parameterization. In *AIP Conference Proceedings*, volume 1168, page 131, 2009.

R. Jacobson, E. Meadors, and H. Link. Power Performance Test Report for the AOC 15/50 wind turbine, Test B in Golden, Colorado. *United States Department of Energy by National Wind Technology Centre. National Renewable Energy Laboratory*, pages 1–150, 2003.

M. Kamruzzaman, K. Meister, T. Lutz, M. Kühn, and E. Krämer. Wind Turbine Aerodynamics and Aeroacoustics at University of Stuttgart-An Overview of Research and Development. 2009.

D. Kraft. A software package for sequential quadratic programming. *Forschungsbericht- Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt*, 1988.

F. Lau. Elementos de aeroacústica. Apontamentos das aulas, Instituto Superior Técnico, 2011.

G. Leloudas. Optimization of wind turbines with respect to noise. 2006.

C. N. H. Lock, H. Bateman, and H. C. H. Townend. *An extension of the vortex theory of airscrews with applications to airscrews of small pitch, including experimental results*. HM Stationery Office, 1926.

M.V. Lowson. Assessment and prediction of wind turbine noise. 1993.

J. F. Manwell, J. G. McGowan, and A. L. Rogers. *Wind Energy Explained: Theory, Design and Application*. John Wiley & Sons, 2010. ISBN 9780470015001.

K. Marvel, B. Kravitz, and K. Caldeira. Geophysical limits to global wind power. *Nature Climate Change*, 2012.

B. W. McCormick. *Aerodynamics, Aeronautics, and Flight Mechanics*. Wiley, 1994. ISBN 9780471575061.

D. H. Meadows, D. L. Meadows, J. Randers, and W. W. Behrens III. *The Limits to growth*. Universe Books, New York, 1974. ISBN 9780876639016.

Ministério do Ambiente do Ordenamento do Território e do Desenvolvimento Regional. Decreto-lei n° 9/2007. Diário da República, 1a série N 12, Janeiro 2007.

M. Mitchell. *An introduction to genetic algorithms*. MIT Press, Cambridge, Mass, 1998. ISBN 0262631857.

B. O. G. Montgomerie, A. J. Brand, J. Bosschers, and R. P. J. O. M. van Rooij. Three-dimensional effects in stall. Technical report, Energy Research Center of the Netherlands, 1997.

P. Moriarty. NAFnoise: A program for calculating 2-d airfoil noise. *Design Code, National Wind Technology Center, USA*, 2003.

P. Moriarty and P. Migliore. Semi-empirical aeroacoustic noise prediction code for wind turbines. 2003.

P. J. Moriarty, G. Guidati, and P. G. Migliore. Recent improvement of a semi-empirical aeroacoustic prediction code for wind turbines. In *Proc., 10th AIAA/CEAS Aeroacoustics Conference, Manchester, UK*, 2004.

P. J. Moriarty, G. Guidati, and P. Migliore. Prediction of turbulent inflow and trailing-edge noise for wind turbines. *11 th AIAA/CEAS Aeroacoustics Conference*, pages 1–16, 2005.

J. Nocedal. *Numerical optimization*. Springer, New York, 1999. ISBN 0387987932.

S. Oerlemans and P. Migliore. Wind tunnel aeroacoustic tests of six airfoils for use on small wind turbines. *Report of the National Renewable Energy Laboratory NREL/SR-500-35339*, 2004.

S. Oerlemans, M. Fisher, T. Maeder, and K. Kögler. Reduction of wind turbine noise using optimized airfoils and trailing-edge serrations. *AIAA journal*, 47(6):1470–1481, 2009.

R. Parchen. Progress report draw: A prediction scheme for trailing edge noise based on detailed boundary layer characteristics. *TNO Rept. HAGRPT-980023, TNO Institute of Applied Physics, The Netherlands*, 1998.

P. Pedregal. *Introduction to optimization*. Springer, New York, 2003. ISBN 0387403981.

R. E. Perez, P. W. Jansen, and J. R. R. A. Martins. pyOpt: A Python-based object-oriented framework for nonlinear constrained optimization. *Structures and Multidisciplinary Optimization*, 45(1):101–118, 2012. doi: 10.1007/s00158-011-0666-3.

L. A. Piegl and W. Tiller. *The NURBS book*. Springer Verlag, 1997.

J. G. Schepers, A. J. Brand, H. Madsen, N. Stefanatos, D. A. Simms, M. Hand, A. Bruining, R. Rooij, Y. Shimizu, T. Maeda, et al. *IEA annex XIV/XVIII, field rotor aerodynamics*. Netherlands Energy Research Foundation, 2001.

J. G. Schepers, A. Curvers, S. Oerlemans, K. Braun, T. Lutz, A. Herrig, W. Wuerz, A. Mantesanz, L. Garcillan, M. Fischer, et al. Sirocco: Silent rotors by acoustic optimisation. In *Second International Meeting on Wind Turbine Noise*, 2007.

Seaforth Energy. AOC 15/50 Specification Sheet, 2010. URL `http://seaforthenergy.com/wp-content/uploads/2010/12/AOC1550-Specification-Sheet.pdf`.

P. J. Skirrow. Acoustic weighting curves, 2005. URL `http://en.wikipedia.org/w/index.php?title=File:Acoustic_weighting_curves_(1).svg&page=1`. accessed 14-September-2012.

W. H. Snyder. Guideline for fluid modeling of atmospheric diffusion. epa office of air quality, planning and standards, research triangle park. Technical report, EPA-600/8-81-009, 1981.

D. G. Stephens. *Guide to the evaluation of human exposure to noise from large wind turbines*. 1982.

A. Tadamasa and M. Zangeneh. Numerical prediction of wind turbine noise. *Renewable Energy*, 36(7): 1902–1912, July 2011. ISSN 09601481. doi: 10.1016/j.renene.2010.11.036.

C. E. Tickell, J. T. Ellis, and M. Bastasch. Wind turbine generator noise prediction–comparison of computer models. In *Proceedings of ACOUSTICS*, 2004.

S. Vanhaeverbeke. Simulation of aeroacoustic emission for small wind turbines. Master's thesis, Delft University of Technology, 2007.

L. F. C. Vargas. Wind Turbine Noise Prediction. Master's thesis, Instituto Superior Técnico, 2008.

R. Vesel Jr. Optimization of a wind turbine rotor with variable airfoil shape via a genetic algorithm, 2009.

L. A. Viterna and D. C. Janetzke. *Theoretical and experimental power from large horizontal-axis wind turbines*. U.S. Department of Energy, Conservation and Renewable Energy, Wind Energy Technology Division, 1982.

S. Wagner, R. Bareiss, and G. Guidati. *Wind turbine noise*. Springer, Berlin, 1996. ISBN 9783540605928.

L. Wang, T. Wang, and Y. Luo. Improved non-dominated sorting genetic algorithm (NSGA)-II in multi-objective optimization studies of wind turbine blades. *Applied Mathematics and Mechanics*, 32(6): 739–748, 2011.

D. L. Whitfield. Integral solution of compressible turbulent boundary layers using improved velocity profiles. Technical report, DTIC Document, 1978.

WWEA. Half-year Report 2012, 2012. URL `http://www.wwindea.org/webimages/Half-year_report_2012.pdf`. accessed 01-october-2012.

W. J. Zhu. Modelling Of Noise From Wind Turbines. Master's thesis, DTU, 2004.