Data-driven surrogates for wind farm control

Marc Clausen

DTU Wind and Energy Systems Bachelor B-0036

December 2022

Authors:
Marc Clausen

Title:
Data-driven surrogates for wind farms control

DTU Wind and Energy Systems-B-0036
December 2022

Project period:
September - December 2022

ECTS: 15

Education: Bachelor of Science

Supervisors:
Søren Juhl Andersen
Tuhfe Göcmen
DTU Wind & Energy Systems

Remarks:
This report is submitted as partial fulfillment of the requirements for graduation in the above education at the Technical University of Denmark.

Technical University of Denmark
Department of Wind & Energy Systems
Frederiksborgvej 399
DK-4000 Roskilde
www.vindenergi.dtu.dk

Abstract

With the increasing importance of wind energy, optimisation of such production facilities is likewise becoming more relevant in recent times. A common problem one might encounter in large wind farms is the consequence of turbines standing in each other's wake. This paper aims to investigate the behaviour of downstream turbines with a combination of CFD and aeroelastic calculations. Furthermore, the possibility of strategically reducing the power of specific turbines to reduce such phenomenon will be explored. Since the flow simulations are costly, a surrogate will be trained to predict behaviour, which can then be applied for later optimisations. Lastly, it is investigated how yawing and pitching might be used to maximise the total power output without increasing the loads unnecessarily. This showed great promise, with a maximum increase of approximately seven to eight percent depending on the spacing between the turbines. Unfortunately, no certain conclusions could be made due to the high uncertainties, but the analysis gave great insight into how these could be improved. Ultimately it was concluded that additional data was needed to get accurate results and that the thesis primarily should be viewed as a basis for further research.

Preface

Marc Clausen

Marc Cl

Signature

05/12/2022

Date

Contents

Abbreviations and Concepts

WFFC	Wind Farm Flow Control
CFD	Computational Fluid Dynamics
LES	Large Eddy Simulations
DEL	Damage Equivalent Loads
BEM	Blade Element Method
TSR	Tip Speed Ratio
CDF	Cumulative Distribution Function
ψ	Yawing angle
Ω	Rotational speed
θ	Pitching angle
U	Flow velocity

1 Introduction

In the last 100 years, power consumption has increased rapidly, and power production is likewise becoming more important [8]. Such an increase introduces a great challenge since green energy production has difficulty keeping up with the rising demand. In the current state, we are involuntarily using an equally increasing amount of fossil fuels, which solves the energy problem but introduces several environmental consequences. With more than 70 countries committing to "net zero" energy production in the near future, the importance should not be overlooked. According to the International Energy Agency [14], this goal is reachable in 2050 on the condition that power production is drastically alternated. As seen in figure 1.1, the required energy source distribution needs to change substantially, with especially solar and wind energy becoming more prevalent.

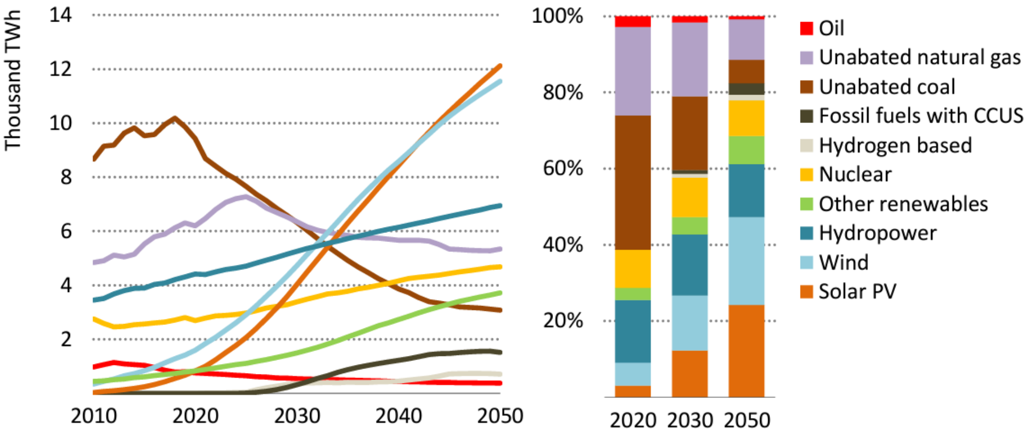

Figure 1.1: Distribution of power production. [14]

Because of this need to increase energy production by turbines, it is now more relevant than ever to optimise existing as well as new farms for them to reach their maximum efficiency. One of the many challenges when planning a wind farm is the unpredictable nature of turbulent flow between the turbines. This can be especially problematic under certain environmental conditions since the convection of the environment plays a large part in breaking down the vortexes created by the blades. Although previous studies have shown the possibility of controlling this breakdown to some extent [11], it is interesting to explore other options as well. One approach may be to change the angle of the turbine relative to the flow to redirect the wake. This specific alteration is generally referred to as yawing and is illustrated in figure 1.2.

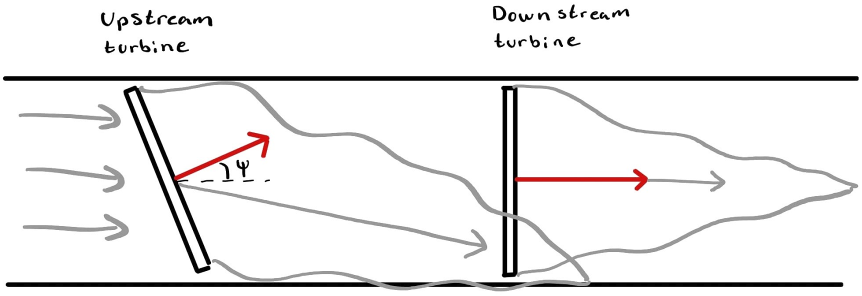

Figure 1.2: Illustration of yawing on upstream turbine

For the case of yawing, we can see how the flow creates a thrust perpendicular to the turbine, which in return affects the flow with an equal force in the opposite direction, ultimately resulting in the wake being steered away from the yawing angle. Such measures are generally referred to as Wind farm flow control (WFFC) and will be the primary point of interest throughout this thesis. More specifically, the consequences of turbulent wakes will be examined with a statistical approach through a combination of CFD and aeroelastic calculations. Based on these calculations, the effect of wake steering through yawing will be evaluated, through probabilistic surrogates, to investigate how these results may be used to optimise the total farm on both total power output and relevant loads. The goal is to produce more power without causing unnecessary fatigue and shortening the lifespan.

2 Methodology

To create a mathematical foundation for the calculations, it makes sense to explain the theory behind the scrips and programs used for the simulations. A detailed explanation would be extensive, and the explanations will therefore be kept relatively short. The goal is that a reader with sufficient knowledge within the field can refresh their theory and create an intuition on how it is applied.

2.1 CFD

To compute power output and loads on the turbines, it is essential to know how the air behaves around the turbine. Derived from Newtons 2. law, the Navier-Stokes equations can describe movements within the fluid

$$\rho \frac{Du}{Dt} = -\nabla p + \nu \nabla^2 u + F, \tag{2.1}$$

$$\nabla \cdot u = 0, \tag{2.2}$$

where u is velocity, p is pressure, μ is viscosity and S is external body forces. These equations are complicated to calculate analytically due to the nonlinearity, and solving these numerically for different input parameters is necessary. The simulations are performed in EllipSys3D [6], a program utilising high-fidelity Large Eddy Simulations (LES). The program allows us to input different turbine configurations and outputs the behaviour of the wake in a time interval.

2.1.1 Large Eddy Simulation

In broad terms, LES [10] is a method within CFD that allows us to filter out small energy eddies, drastically reducing computational time. This is achieved by introducing a filter defined by

$$\boldsymbol{u}(\boldsymbol{x},t) = \overline{\boldsymbol{u}}(\boldsymbol{x},t) + \boldsymbol{u}'(\boldsymbol{x},t), \quad \overline{\boldsymbol{u}}(\boldsymbol{x},t) = \int G(\boldsymbol{\epsilon},\boldsymbol{\Delta})\boldsymbol{u}(\boldsymbol{x}-\boldsymbol{\epsilon},t)\boldsymbol{d\epsilon}, \tag{2.3}$$

where $\bar{u}$ is the spatial mean and u' is the fluctuations. Applying this to the Navier Stokes equation 2.1 yields

$$\frac{\partial \bar{u}_j}{\partial t} + \frac{\partial \overline{u_i u_j}}{\partial x_i} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_j} + \frac{\partial}{\partial x_i}\left(\nu\left(\frac{\partial \bar{u}_j}{\partial x_i} + \frac{\partial \bar{u}_i}{\partial x_j}\right)\right) + \bar{F}_j \tag{2.4}$$

Note that $\overline{u_i u_j}$ is the beforehand mentioned nonlinearity and introduces computational problems. To alleviate this problem, we'll introduce a residual tensor defined as

$$\tau_{ij} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j, \tag{2.5}$$

which substituted into 2.4 gives

$$\frac{\partial \bar{u}_j}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_i} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_j} + \frac{\partial}{\partial x_i}\left(\nu\left(\frac{\partial \bar{u}_j}{\partial x_i} + \frac{\partial \bar{u}_i}{\partial x_j}\right)\right) + \bar{F}_j \tag{2.6}$$

We can now numerically solve the equation, except for the residual tensor τ_{ij}, which should be considered through a mathematical model instead. EllipSys3D simulations use a slightly rewritten version of this equation, where the residual is modelled by an "eddy viscosity" ν_t. Substituting $\tau_{ij} = -2\nu_t S_{ij}$ yields

$$\frac{\partial \bar{u}_j}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_i} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_j} + \frac{\partial}{\partial x_i}\left((\nu + \nu_t)\left(\frac{\partial \bar{u}_j}{\partial x_i} + \frac{\partial \bar{u}_i}{\partial x_j}\right)\right) + \bar{F}_j \tag{2.7}$$

In practice, the method applies that eddies smaller than a certain threshold are excluded from the simulations, which drastically decreases the computational time. Instead, these are described with a less accurate model, which is a forced limitation of today's technology.

2.2 Flex5

Flex5 is an aero-elastic code which utilizes blade element momentum theory (BEM) [4] to compute load distributions, from which parameters like power can be derived. Furthermore, modal analysis [19] is applied within the script, determining mode shapes that model deflections of both blades and tower. The script combines these methods to produce times series for power, loads, and deflection, either by a direct coupling on the actuator lines or through downstream inflow data, both from the CFD calculations. To further our understanding of this process, we might explain the two methods as such.

2.2.1 Blade momentum theory

Blade momentum theory [4] is a method for determining forces acting on the turbine blades. The theory works by summing up forces acting on discretized blade slices, approximating the collective work and thrust created by the wind. One of these slices is illustrated in figure 2.1

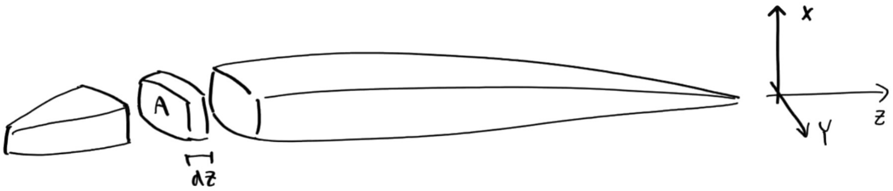

Figure 2.1: Illustration of turbine blade cross section

In normal operation, we'll have 2 wind components, one for the incoming flow U_∞ and one for the rotation U_r. The rotational wind component will naturally be in the tangential direction, while the incoming flow is angled by the yawing on the turbine. Such wind condition is illustrated on a blade slice in figure 2.2.

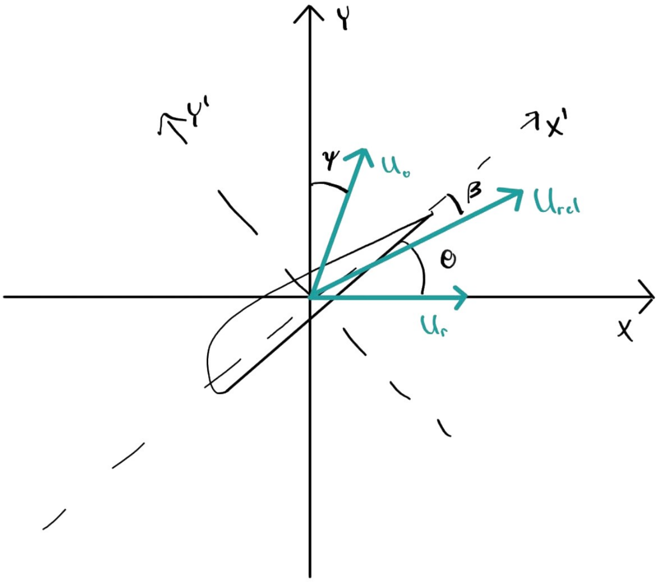

Figure 2.2: Cross section with velocity components

where ψ is the yawing angle, θ is the pitching angle and β is the angle between the relative velocity U_{rel} and the local axis of the airfoil. The velocity acting on the blade will undoubtedly be lower than the free stream velocity due to the pressure drop after the energy is extracted. Likewise does the velocity, due to rotation, depend on which part of the blade we look at. To account for this, we'll introduce axial factors α and α', and describe the velocities as

$$U_0 = U_\infty(1-\alpha), \tag{2.8}$$

$$U_r = \Omega R(1-\alpha'). \tag{2.9}$$

Additionally, it should be mentioned that α' could be expressed as $\alpha' = \big(1 - \frac{z}{R}\big)$, since the velocity increases linearly across the length of the blade. With simple geometry, the magnitude of the relative velocity then becomes

$$U_{rel} = \sqrt{\big(U_\infty(1-\alpha)\sin(\varphi) + \Omega R\,(1-\alpha')\big)^2 + U_\infty(1-\alpha)\cos(\varphi)^2}. \tag{2.10}$$

The pressure acting on the blade in each direction may be calculated as

$$P_x = \frac{1}{2}\rho C_x(l,\beta,\phi) \cdot U_{rel}^2, \tag{2.11}$$

$$P_y = \frac{1}{2}\rho C_y(l,\beta,\phi) \cdot U_{rel}^2, \tag{2.12}$$

where C_x and C_y are form factors, varying as a function of the angle of attack β, pitch ϕ, and position on the blade l. It should also be mentioned that β could be expressed by the velocity magnitudes, pitch, and yaw but is kept this way to keep expressions short. The total torque can then be summed up across the blade, which yields

$$T_x = \int_0^L P_x(z)dA = P_x(z) \cdot h(z)dz, \tag{2.13}$$

$$T_y = \int_0^L P_y(z)dA = P_y(z) \cdot b(z)dz. \tag{2.14}$$

For rotational bodies, power is generally defined as torque times rotational speed:

$$P = \Omega \cdot T_x. \tag{2.15}$$

Though Flex5 calculates other parameters from BEM, the derivation all share the same general idea. Since power and blade loads are most relevant for our analysis, the theory will be kept as is.

2.2.2 Modal analysis

BEM is useful for specific quantities but does not give insight into blade deflections. These can, however, be very important if the load fluctuates approach the natural frequency of the blades. In theory, this might cause immense deflections and stresses within the structure. To account for such situations, Modal analysis [19] can be applied to determine mode shapes and natural frequencies.

Consider an elastic structure and its belonging displacement equation. Taking possible nonlinearities into account, this might be described in a general sense as a partial differential equation. If one chose to separate linear and nonlinear terms, the relation could be written as

$$L[w] = g(w, \dot{\omega}, t), \tag{2.16}$$

where L is a differential operator, and g is a function describing dampening and periodic loads. To solve such an equation, we will start by looking at the free undampened system, where $g = 0$ and $w(x,t) = \varphi(x)\sin(\omega t)$. The differential eigenvalue problem then becomes:

$$L[w] = 0. \tag{2.17}$$

First we find the free natural frequencies $\omega = \omega_j$ and the connected oscillation modes $\varphi(x) = \varphi_j(x), j = 1, 2, ...$ Like a Fourier series, w then be described as an infinite sum of these

$$w(x,t) = \sum_{j=1}^{N} y_j(t)\varphi_j(x). \tag{2.18}$$

Note that this approximation becomes exact for $N \to \infty$. Inserting this expression in 2.16 and integrating over the length yields

$$\int_0^l \varphi_i(x) L\left[\sum_{j=1}^{N} y_j \varphi_j(x)\right] dx = \int_0^l \varphi_i(x) g\left(\sum_{j=1}^{N} y_j \varphi_j(x), \sum_{j=1}^{N} \dot{y}_j \varphi_j(x), t\right) dx. \tag{2.19}$$

By simplification, this leads to N ordinary differential equations, usually referred to as the modal equations. If the eigenvectors turn out to be orthogonal (which is often the case), this will lead to

$$\ddot{y}_j + \omega_j^2 y_j = p_j(\mathbf{y}, \dot{\mathbf{y}}, t), \quad j = 1, 2, .., N, \quad \mathbf{y} = \{y_1, y_2, .., y_N\} \tag{2.20}$$

With y_j being the modal factors. These factors can then be summed up by 2.18, giving us the deflection.

On a side note of the application of this theory for turbine blades, it should be mentioned that for transverse oscillations in beams, 2.16 becomes

$$\frac{\partial^2 \omega}{\partial t^2} + \frac{E(x)I(x)}{\rho A(x)} \frac{\partial^4 \omega}{\partial x^4} = f(x,t) - c\dot{\omega} \tag{2.21}$$

Solving this for the blades beforehand allows us to assume specific mode shapes and makes the actual calculations very fast.

2.3 Surrogates

Introductory statistics bring many methodologies for describing and predicting various data distributions. However, these mathematical models become increasingly complex when introducing additional variables and dependencies, and even more advanced multi-variable models sometimes struggle to describe these relations. Therefore one might imagine that the case of 2 turbines, where the power distribution can be affected by a single DOF, might be challenging to describe through common statistics. A way to tackle this is to go for a more empirical approach, where we will consider a random model, which we will tune to match our simulated data cases. This general line of thought is more commonly referred to as machine learning [2] and is an incredibly useful tool in such situations. More precisely, this thesis will utilize an elementary branch of machine learning called ”Feedforward neural network”, which is assumed to be adequate for the problem at hand.

2.3.1 Feed forward neural network

Definition

Consider a simple case of x inputs and y outputs. In between these apparent quantities, we will introduce a network of l layers and n_i neurons per layer, as illustrated in figure 2.3. A neuron is then defined as a scalar often restrained within a set range and determined as a function of the previous layers neurons.

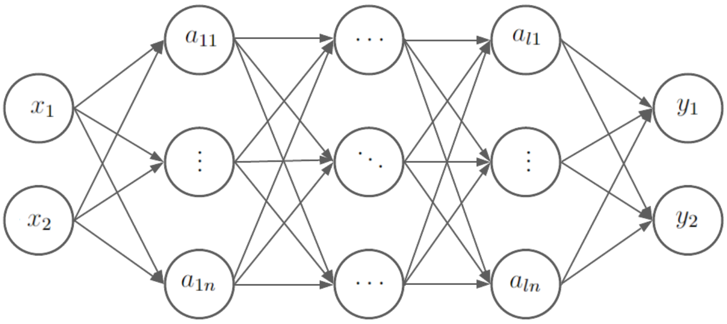

Figure 2.3: Illustration of neural network

More precisely, we will introduce an activation function to represent said scalar as a function of the previously determined scalars, all the way back to the input variables. There are many ways this might be done, but for the sake of simplicity, we will define the value of a single neuron linearly. The expression then becomes

$$a_{l,n} = \sigma\left(w_{l,1}a_{(l-1),1} + w_{l,2}a_{(l-1),2} + \ldots + w_{l,N}a_{(l-1),N} + b_{l,n}\right), \tag{2.22}$$

where l and n are indexes representing layer and neuron respectively. $w_{l,n}$, also called the weight, is a value describing how much influence each previous neuron should have on the value, $b_{l,n}$ being an additional bias and σ a function which squeezes the result to a value within the defined range. Such scaling could be expressed in multiple ways, the most common being the Sigmoid and ReLU functions [18]. It should also be mentioned that the range is not random, but defined by the applied activation function. For Sigmoid this would be between 0 and 1. Additionally, the expression could be shortened by writing in matrix form:

$$\mathbf{a}_l = \sigma\left(\mathbf{W}_{l-1}\mathbf{a}_{l-1} + \mathbf{b}_l\right). \tag{2.23}$$

Training

In layman's terms, the "training" of a network could be described as the simple act of determining the best values for the beforehand mentioned weights and biases. The process, however, is not as elementary as one might expect. To describe how well the network is performing for a given set of weights, we define a cost function as the sum of squares between the predicted values and the right output.

$$C_L = \sum_{j=1}^{N_L}\left(a_{j,L} - y_j\right)^2. \tag{2.24}$$

Since the weights and biases are the only parameters we can change, it is obvious to explore how these could be altered to minimize this function. This expression might be a little complicated due to all the compositions of different functions. We will, therefore, introduce a function beforehand to simplify the notation.

$$z_{j,L} = w_{j1,L} \cdot a_{1,(L-1)} + w_{j2,L} \cdot a_{2,(L-1)} + \ldots + w_{jN,L} \cdot a_{N,(L-1)} + b_{j,L}, \tag{2.25}$$

$$a_{j,L} = \sigma\left(z_{j,L}\right), \tag{2.26}$$

with j being a variable running over the number of neurons in each layer. With this simplification, the partial derivatives become

$$\frac{\partial C_L}{\partial w_{jk,L}} = \frac{\partial z_{j,L}}{\partial w_{jk,L}} \frac{\partial a_{j,L}}{\partial z_{j,L}} \frac{\partial C_L}{\partial a_{j,L}} = a_{k,(l-1)} \sigma'(z_{j,l}) \frac{\partial C_L}{\partial a_{j,L}}, \tag{2.27}$$

$$\frac{\partial C_L}{\partial b_{j,L}} = \frac{\partial z_{j,L}}{\partial b_{j,L}} \frac{\partial a_{j,L}}{\partial z_{j,L}} \frac{\partial C_L}{\partial a_{j,L}} = \sigma'(z_{j,l}) \frac{\partial C_L}{\partial a_{j,L}}. \tag{2.28}$$

Note that the cost function partially derived with respect to the current neuron depends on how much that neuron contributes to the next layer, hence:

$$\frac{\partial C_L}{\partial a_{j,L}} = \sum_{h=1}^{N_{l+1}} w_{hk,(l+1)} \sigma'\left(z_{h,(l+1)}\right) \frac{\partial C}{\partial a_{h,(l+1)}}. \tag{2.29}$$

We can then adjust each variable according to its importance on the cost function. This method is known as gradient ascend and allows us to slowly approach a local minimum of the cost function. Though a local minimum is not necessarily the best possible configuration, finding the global minimum is way more complex and will not be discussed nor applied in this thesis. Obviously, this requires an enormous amount of calculations, which can take much computational time. A way to alleviate this is to only test a sample of the training data before adjusting the weights. Such sampling is referred to as a batch and works by approximating the gradient for the entire data set. Though it is technically less accurate, the run time is heavily decreased, resulting in less computational time.

Application

In the case of 2 turbines, we want to predict a Weibull fit for the CDF power/load distribution depending on the configuration of the turbines. Therefore, in principle, we will have seven inputs, the three control parameters for each turbine and the spacing in between. We will, however, keep the analysis simple and let the control of the second turbine automatically determine pitch and rotational speed as it would normally. The spacing is likewise simplified by only looking at two distances. We will, as a result, only have four inputs and two outputs, with the inputs being the three control parameters for the first turbine and the outputs being the Weibull constants.

2.4 Fatigue calculations

The output of flex5 results in a large array of load data, which due to the turbulent air, wildly fluctuates around a mean value, as illustrated in figure 2.4.

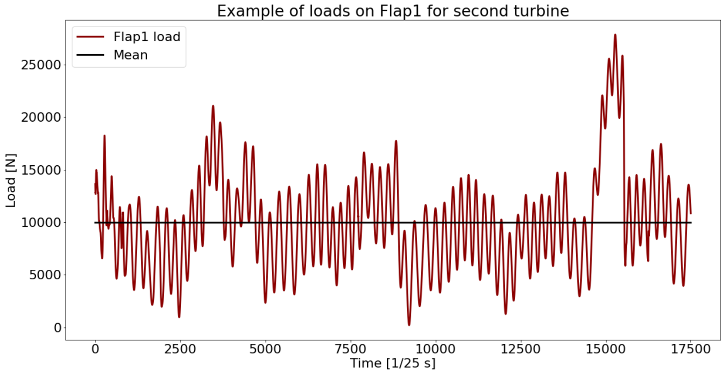

Figure 2.4: Example of load on flap for $\psi_1 = 20°$, $\theta_1 = -0.5°$, $\Omega_1 = 0.75\frac{m}{s}$, $\psi_2 = 5°$

Generally, fatigue is evaluated from a periodic load with n_i cycles of amplitude S_i, and each material has an empirical found value for cycles until failure N_i at a given amplitude. By this logic, we can describe a damage factor as the percentage of the total fatigue the material has endured.

$$d_i = \frac{n_i}{N_i} \tag{2.30}$$

According to the Palmgren-Miner linear damage hypothesis, the relationship between amplitude and cycles to failure is linear in a log-log plot with a slope of $1/m$ [7]. The damage can then be described as

$$d_i = \frac{n_i S_i^m}{S_0^m}, \tag{2.31}$$

where S_0 is the amplitude at $N_i = 0$ and m is the Wöhler exponent. This exponent is, as previously mentioned, found by empirical experiments but can, for our use case, be roughly approximated as 5 for the tower and 10 for the blades. In the case of cycles at multiple amplitudes, the damage is summed up as

$$D = \sum d_i = \frac{1}{S_0^m} \sum n_i S_i^m \tag{2.32}$$

Since we now have an expression for the total load over a time period, it might be useful to calculate the equivalent amplitude for a given number of cycles. Here we denote n_{eq} as a chosen value for testing, in practice represented as a frequency over time, and S_{eq} as the matching amplitude, which results in equal fatigue:

$$D = \frac{n_{eq} S_{eq}^m}{S_0^m}. \tag{2.33}$$

Solving for the amplitude gives us

$$S_{eq} = \left(\frac{\sum n_i S_i^m}{n_{eq}} \right)^{\frac{1}{m}}. \tag{2.34}$$

DEL is an instrumental figure since it can be used both as a practical way to test essential components and as an indicator of total damage. For our analysis, this will be used as the latter, where the damage equivalent load (DEL) will be compared between the different configurations.

2.5 Rainflow counting

DEL will be an essential measure of loads throughout the analysis, but it might be noted from equation 2.34 that the data needs to be separated into periodic loads of equal amplitude. To simplify the raw data, we use the rain flow counting algorithm [16], which is the standard for such applications. The algorithm simplifies the data and then counts cycles of each amplitude. The process can, in broad terms, be described as

1. Round data points to a fitting accuracy.
2. Remove data points outside peaks and valleys.
3. Count half and full cycles between remaining data points.

This is, of course, a very vague description but gives a general idea of the process. The resulting cycle count can then be applied to 2.34 to calculate the equivalent load.

3 Normal Operation

To further our understanding of wind turbine operation and prepare a more detailed analysis, we need a fundamental intuition of how turbines behave under normal operating conditions. A single turbine will have three control parameters; yaw, pitch, and rotational speed. However, since yawing would never benefit a single turbine under normal wind conditions, this chapter will primarily investigate the relationship between pitch and rotational speed. We will also explore how dimensionless numbers compare with their associated quantities and how they are relevant in further analysis.

3.1 Setup

Consider a turbine standing on a plane field with no significant obstacles in the near vicinity. As there is nothing obvious to disturb the flow, we might assume that the air behaves relatively linearly within our theoretical test volume, and we may therefore approximate this with a simple uniform flow. Note that this might not give a perfect representation of the loads in a more practical situation, but it should be sufficient to explain how the turbine wants to behave for a given flow velocity. Since calculations for such a simplified example are very cheap computationally, we will look into a wide range of velocities. More specifically, the simulation starts with a speed of 5 m/s and steadily increases up to 25 m/s and back again, in steps of 1 m/s. These sudden increases between each step will create small fluctuations in the data, which we will have to compensate for. A simple way to do this is to calculate each quantity for all data points with a velocity within a range of the current step. We can then take the average of these values, giving us a good estimation, assuming that the fluctuations are equally distributed around each step. To illustrate the given range and steps, an exaggerated velocity curve is shown in figure 3.1.

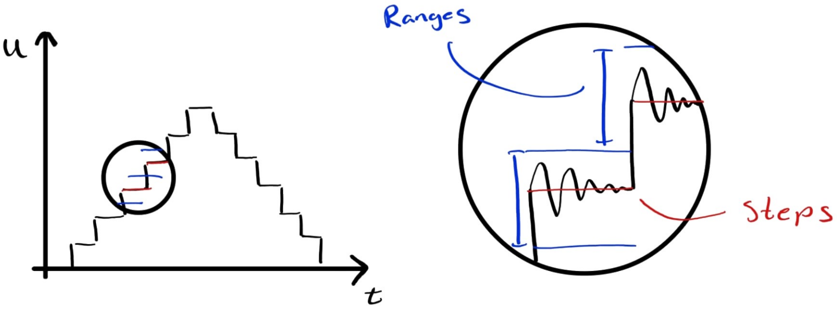

Figure 3.1: Illustration of exaggerated fluctuations on velocity curve for uniform inflow

3.2 Pitch and rotational speed

As a starting point, we would like the wind turbine to produce as much electricity as possible. However, when increasing the velocity above a certain threshold, we might encounter physical limitations regarding how fast the turbine can rotate and what force it can withstand without collapsing. In practice, this means that we have a maximum rated power output and that we have to purposely take less energy from the air by pitching if proceeding this limit. Since power and rotational speed are almost proportional, we will compare rotational speed Ω and pitch θ as a function of flow velocity U, which is illustrated in figure 3.2.

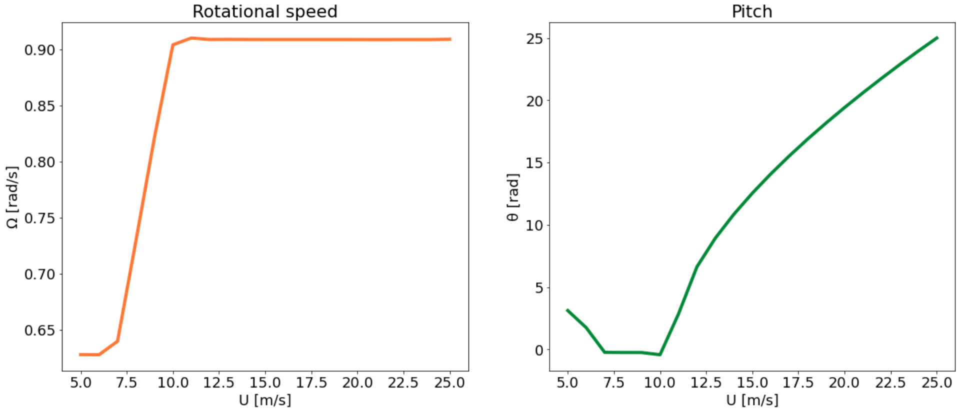

Figure 3.2: Rotational speed and Pitch as a function of velocity

As we might expect, it is seen that the turbine extracts as much energy as possible from the air for low wind speeds. In this region, the rated power/rotational speed has yet to be reached, and it does not make sense to pitch the blades. However, at approximately 10 m/s, the rotational speed reaches its maximum and stays constant for the remaining velocities. For this range, the pitching is determined automatically to match the rating and turns out to have a slightly curved relation between angle and incoming velocity. Knowing how the power and thrust behaves is also quite interesting if we want an understanding of normal operation. The plot of these is seen in figure 3.4.

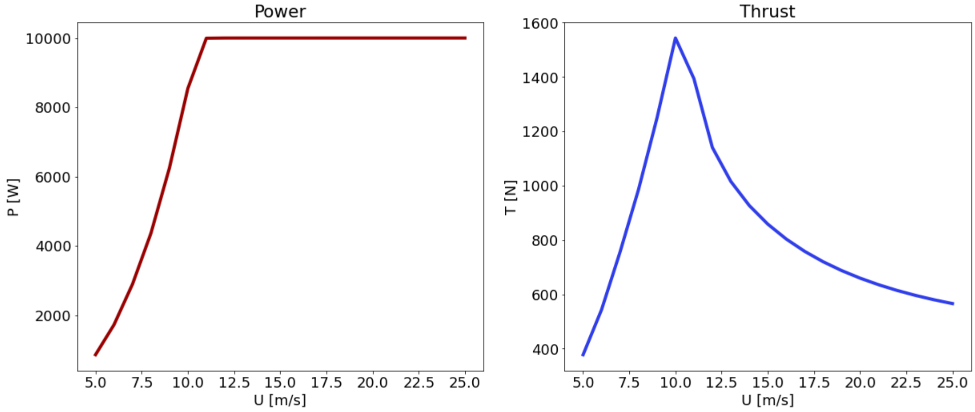

Figure 3.3: Power and Thrust as a function of velocity

The graphs are very predictable, considering the previously mentioned arguments regarding the rated power limitations. Here we see that similar to the rotational speed the power increases almost linearly up to the rated value and will afterwards stay constant at 10 MW. The trust has a very similar profile until that 10$\frac{m}{s}$ mark where the pitching starts. In actuality, power and thrust can be proven to follow a third and second-order polynomial, respectively. However, regarding them as linear is adequate for a basic understanding of the operation [15]. After that, it will steadily decrease as the blades turn into the wind and encounter less resistance.

3.3 Dimensionless analysis

One might also be interested in how the power production and thrust compare to the wind condition in terms of potential energy and momentum. We can investigate such by plotting the ratio between Power/thrust and their reference value as a dimensionless number. The ratios can be expressed as

$$C_P = \frac{P}{\frac{1}{2}\rho A U^3}, \quad C_T = \frac{T}{\frac{1}{2}\rho A U^2} \tag{3.1}$$

Plotting as a function of U yields the graphs displayed in figure 3.4

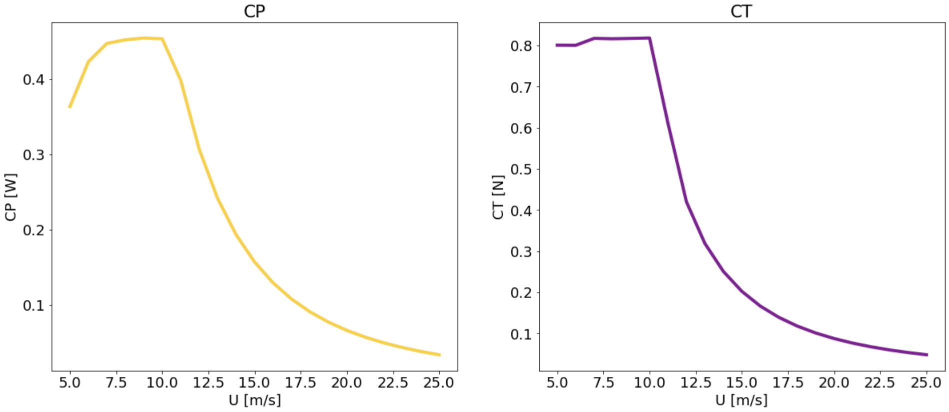

Figure 3.4: C_P and C_T as a function of velocity

Starting with the power, we can see that for very low wind speeds, the turbine struggles to extract as much energy as it wants. After that, there is a small range from about 6 m/s to 10 m/s, where the value is constant at roughly $C_P = 0.46$. Interestingly, this is not the period of highest energy production but rather a practical limit of how much of the energy potential we can extract. The value is about what we expect for a modern turbine and is a substantial percentage of the theoretical max at just over 59% [3]. Lastly, our ratio decreases rapidly as the faster velocities contain a lot of energy that we cannot extract with the given turbine. For the thrust, we can primarily use the same argument. However, what is notable is that the value for low speeds is much higher than C_P. This means, in practice, that the turbine experience much greater loads relative to the power produced than it would for higher wind speeds. This is very much in conformity with the thrust graph and means that though we will not produce more energy at higher velocities, it is still beneficial due to the lower loads resulting in a longer lifespan.

4 Changing control parameters

With our new understanding of the normal operation and the meaning of dimensionless numbers, it makes sense to investigate how these behave when tweaking the turbine parameters. Plotting C_P and C_T as a function of the yaw and pitch-angle should hopefully give an intuition of how they affect power and loads in further analysis. Power and load are, of course, also dependent on rotational speed, so two contour plots are created to visualise this 4-dimensional space.

4.1 Dimensionless power

Starting by exploring how yaw and pitch affect CP, the appropriate plot is illustrated in figure 4.1.

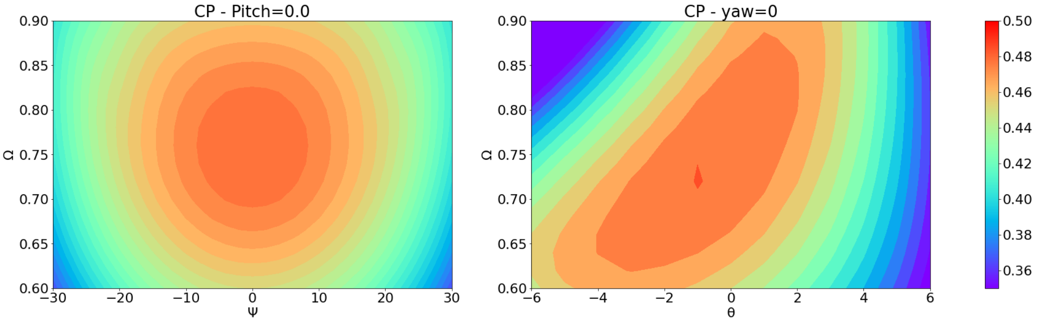

Figure 4.1: C_P as a function of Ω, ψ, θ for $U = 8m/s$

What is immediately apparent is that the direction of the yawing does not seem to have any significant effect on the turbine within a reasonable range. Though this might be expected, it is not necessarily given, and it is pleasant to see the simulations backing up our intuition. Additionally, the turbine is most effective, operating at approximately $0.75rad/s$, which matches the normal operation seen in figure 3.2. For pitching, the optimal angle seems dependent on the rotational speed and appears negative for low rotational speeds and positive for high rotational speeds. However, similar to the yawing, this dependency is not very sensitive. We will therefore conclude that it is possible to both yaw and pitch to a certain degree without sacrificing any notable power. However, when reaching high angles, both seem to suffer significantly, which we should keep in mind for further analysis.

4.2 Dimensionless thrust

To do a similar analysis for CT, a matching plot is illustrated in figure 4.2

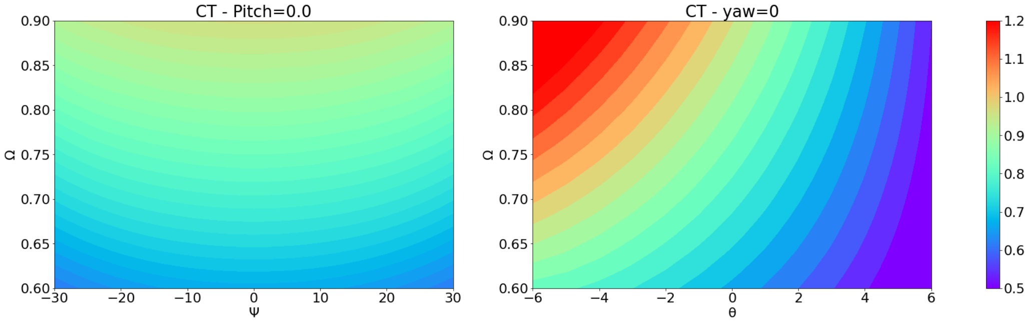

Figure 4.2: C_T as a function of Ω, ψ, θ for $U = 8m/s$

Looking at C_T as a function of yaw and rotational speed, we again see that the yawing direction does not seem to have any effect, just like we concluded for CP. What is, however, notable is that CT slightly decreases with larger angles. Though this is technically an advantage, we should remember from figure 4.1 that it also reduces power. For pitching, we see a similar trend for positive angles, as this also follows CP. However, negative angles induce much higher thrust while simultaneously producing less power. As one might expect, this is very unfavourable, and large negative pitching angles are, therefore, less interesting for further analysis.

5 Data

To better understand the optimisation and network limitations, it makes sense to explain how the data is produced, how it is organised, and how it can be applied. The data generation can largely be separated into CFD and Aeroelastic simulations.

5.1 CDF

The CDF calculations are created through EllipSys3D, which computes wind data through Large Eddy Simulations. The necessary theory is briefly explained in 2.1.1 and will not be elaborated further. The turbine used is an IEA10MW, a theoretical turbine used for research applications [9]. For the setup, we use 12 cases of the upstream turbine, modelled with an actuator disk approach [1]. We will consider only negative yawing angles as configurations go, as we know these to behave symmetrically from section 4.1. For pitching, we only consider positive or very slight negative angles since large negative angles were not beneficial, as determined in section 4.2. The 12 different combinations of control parameters are displayed in table 5.1.

Case nr.	0	1	2	3	4	5	6	7	8	9	10	11
Omega	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.87	0.76	0.84	0.68	0.81
Pitch	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	2.06	1.38	1.30	3.44	2.88
Yaw	0	-5	-10	-15	-20	-25	-30	-15	-20	-25	-15	-20

Table 5.1: Configurations used in CDF simulations

For each of these cases, the flow is captured at 28 different downstream distances one radius apart, starting at $0R$ and going up to $27R$. Additionally, the simulation is separated into 21 smaller periods to get a statistical groundwork for the somewhat unpredictable nature of the flow, with each separation containing 17501 data points, over a $700s$ period, for each of the three velocity components.

5.2 Aeroelastic

Regarding the Aeroelastic simulations, we are only concerned about the flow entering the turbine, which implies that the second turbine's wake is irrelevant. We, therefore, utilise a "ghost turbine" approach, in which the down steam turbine does not affect the flow and does not need to be included in the CFD simulations. The turbine's reaction to wind conditions is determined through Blade element theory, explained in 2.2.1. What is especially important to realise is the computational gain through this method. Since we do not have to perform heavy CDF calculations for each two-turbine configuration, we can test a wide variety of yawing angles. This is important when training the surrogates, where the extra data is very beneficial.

It is easy to imagine that there will be little to no benefit in yawing for very large spacings. It is also logical that alterations will not be efficient for very close distances since the turbines perform poorly under any circumstance. For this reason, it's interesting to investigate which distances yield the greatest benefit, if any, for this kind of optimisation. However, due to the project's time constraint, we can not evaluate this properly, and will therefore, temporarily assume that the turbines perform similarly to smaller turbines analysed in previous studies. A very recent paper [5] showed that the highest relative increase is seen at approximately $6R$, which will be the basis for the analysis. We will also look into a more standardised distance of $14R$ to see how this compares. For the second turbine yawing, we will investigate the following angles.

$$s \in \{6R, 14R\}, \quad \psi_2 \in \{-30°, -28°..28°, 30°\}$$

Note that these calculations must be performed for all 21 periods at each configuration, which results in massive data that needs to be modelled to lower the computational cost. This is done by collecting the average value of relevant parameters for each period, yielding 21 similar averages. Plotting these as a cumulative distribution function gives a familiar s-curve well-known for normally distributed quantities. To represent this as simply as possible, we will fit a Weibull curve to the cumulative distribution through maximum likelihood estimation [13]. A randomly selected example of such fit is illustrated in figure 5.1

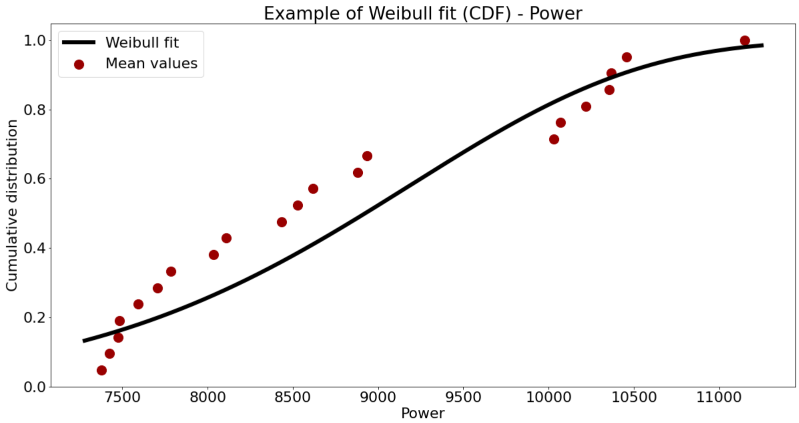

Figure 5.1: Weibull fit on power CDF for $\psi_1 = 25°$, $\theta_1 = -0.5°$, $\Omega_1 = 0.75\ m/s$, $\psi_2 = 15°$

5.3 Weibull

A Weibull distribution is a continuous probability distribution widely used in probability theory. It can be expressed as both a probability density function (PDF) and a cumulative density function (CDF) and is defined by only two parameters, referred to as shape and scale. The CDF, being most relevant to the problem, is expressed as

$$Weibull_{CDF}(x) = \begin{cases} 1 - e^{-(x/\lambda)^k}, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{5.1}$$

With λ being the scale and k being the shape. In theory, this heavily simplifies the data both in size and complexity, but we need to investigate if this is an accurate representation before applying it in the analysis. A good measure of such accuracy is the coefficient of determination, often referred to as the R^2 value. Although there are different definitions of R^2, the value is generally expressed as

$$R^2 = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \bar{y})^2}, \tag{5.2}$$

where f_i is the modelled probability and y_i is the true value for the data. Since the fit accuracy will vary between runs, we should consider how the R^2-values are distributed. Plotting so yields the graph presented in figure 5.2

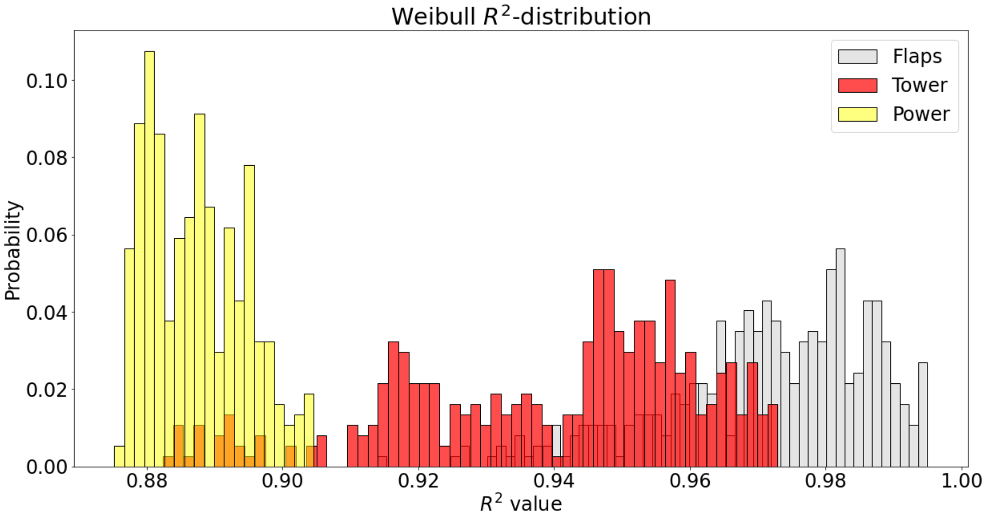

Figure 5.2: R^2-value density distribution

Firstly we will note that even for the worst fit, which seems to deviate heavily from the mean, the model can still explain over 85% of the variance within the data. An interesting trend is that power generally performs worse than loads. However, this is a small margin, and both would be considered highly correlated with the proposed fit. For further analysis, we will consider the Weibull constants an adequate representation of the data sets. When used for optimisation, it should be noted that none of these constants is directly applicable, and we should instead use both to compute another more useful measure. There are two obvious ways to represent this: the mean and median. Although it is difficult to answer which is better, we will apply the median with the argument that it is less sensitive to large individual fluctuations in the data. The median for Weibull distributions can be expressed as

$$\mu = \lambda \ln(2)^{\frac{1}{k}} \tag{5.3}$$

Additionally, we can express the uncertainty on the median as the standard deviation, which is expressed as

$$\sigma = \lambda \sqrt{\Gamma\left(1+\frac{2}{k}\right) - \Gamma\left(1+\frac{1}{k}\right)^2} \tag{5.4}$$

where $\Gamma()$ is the gamma function. By this logic, we can predict a turbine's expected power or load as $\mu \pm z \cdot \sigma$, where z denotes the number of standard deviations. In the case of the final evaluation, we use $z = 1$, which is the custom when applying standard deviation as uncertainties.

6 Surrogates

Although the simulations are quite extensive, it is difficult to apply them directly since they have trouble explaining small adjustments in the control parameters. Especially yawing for the first turbine is limited for 5-degree increments due to the high computational cost of full CDF simulations. In this chapter, we will attempt to train a network which is able to predict behaviour between these known values. Assuming that the network reaches sufficient accuracy, it can then be used to interpolate values for finer adjustments. Like the Weibull model, it is, however, necessary to test the precision of such network before applying it.

6.1 Training data

For the training of the network, we will separate the data as 60% training-, 20% validation- and 20% test data. To avoid systematic uncertainties, the data is shuffled before training, so the training, validation, and test are independent between runs. This might, however, lead to an uneven distribution of the configurations in the different sections. Imagine, for example, that every configuration where the pitching is 0 was placed in the test data. This means that we might test on a configuration in which the network is not trained, and the predictions might therefore be very poor. Although it is difficult to ensure this is not the case, when the data is relatively limited, we can check for obvious deficiencies by plotting the distributions when testing.

6.2 Network

Since the data is limited by the number of simulations, which is relatively small, it makes sense to create a network of similar proportions. To support the claim that such a network is adequate, we will investigate whether additional neurons pose any advantage for lowering the loss function. Since we expect to see the most unpredictable behaviour when the turbulence is highest, we will investigate the closest configuration. Two layers with the same amount of neurons in each allow us to plot the converged loss as a function of neurons per layer. This is done 15 times, and the average loss is calculated for each neuron amount. Additionally, we might assume the loss values to approximately follow a normal distribution, so the standard deviation is computed as a measure for the uncertainty of the average. Plotting the average as a function of the neuron amount, with the standard deviations, yields the graph displayed in figure 6.1

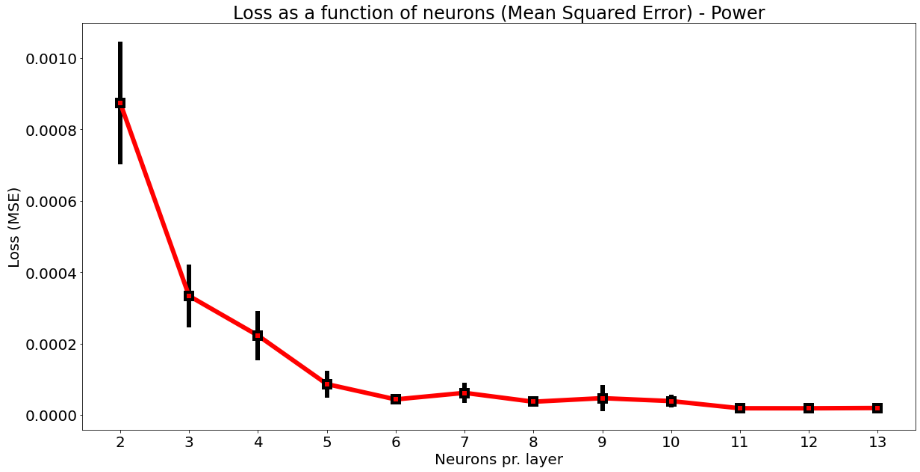

Figure 6.1: Loss as a function of neuron pr layer

As we might expect, very low neuron amounts cannot predict the turbine behaviour since it simply does not have enough parameters to describe the correlation. The question then becomes, what is the threshold where more neurons do not pose a clear benefit? What is interesting from figure 6.1 is that the graph already seems to flatten out at approximately six neurons. We can also see that the range of uncertainties is large for low numbers of neurons but drastically shrinks as the function converges. The DELs have different dependencies and might therefore require a network of different complexity. For this thesis, however, we will assume this to be sufficient for all cases due to the time constraints of the project. For further analysis, we will use six neurons per layer, as there seems to be no statistical benefit for adding additional complexity.

6.3 Accuracy

The network has reached a relatively low loss for both training and validation, but this number is difficult to relate to the actual accuracy of the predictions. To get a better intuitive understanding of the validity, we will look at the relative deviation from the predictions to the test data. The purpose is that this data is independent of the training and, therefore, an unbiased verification. Additionally, the loss function is based on scaled values, which is not the case with this approach. The relative deviation can be expressed as the difference between the prediction and the expected value normalized by size. Mathematically it is written as

$$\delta = \frac{x_a - x_p}{x_a} \tag{6.1}$$

with x_a being the true value and x_p being the prediction. Plotting all the networks, both power, and the different DELs, together as a boxplot yields figure 6.2

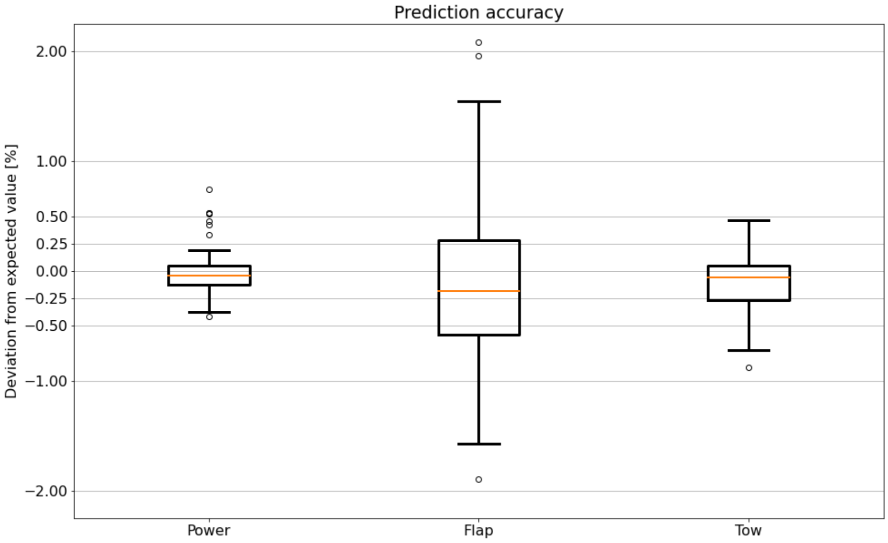

Figure 6.2: Boxplot of median prediction relative to expected value

First, we will note that all measures behave relatively well, with the first and third quantiles nearly within half a percent of the expected outcome. We should also mention that flap load does have some outliers, deviating up to two percent, which is a relatively large deviation compared to the majority of the values but still relatively accurate. We will proceed with this model, though this uncertainty should be considered before making any conclusions. To check for any deficiencies, as mentioned in 6.1, we will plot the control parameter distributions for both training, validation, and test data sets. Doing this yields the histogram displayed in figure 6.3.

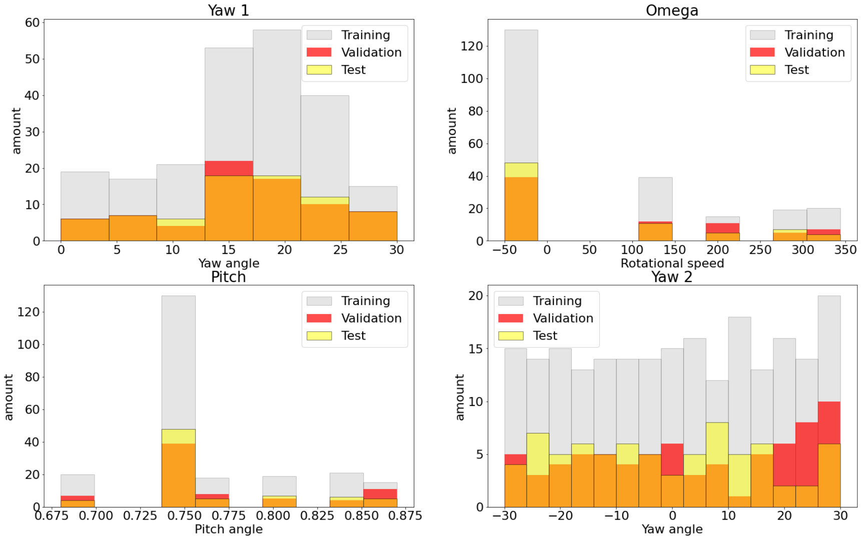

Figure 6.3: Distribution of control parameters

Although the data does not seem to be perfectly separated, there are no obvious missing values either. We must therefore conclude that the performance of the model cannot be explained by the data separation and might instead be caused by the unpredictable nature of the moments. As future work, it might make sense to investigate whether more complex networks could explain this behaviour more consistently,

To give a practical example of how the whole process works, we will plot the Weibull prediction with the data, similar to what we did in figure 5.1. Again taking the power distribution for an arbitrary configuration from the test data, gives us the graph displayed in figure 6.4

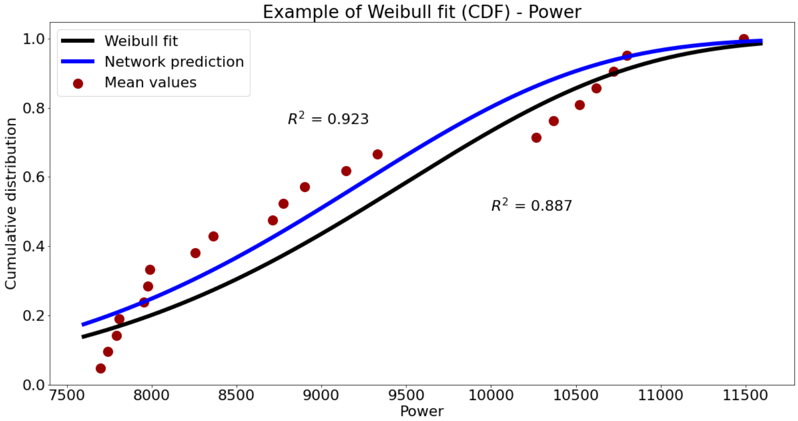

Figure 6.4: Network prediction for $\psi_1 = 25°$, $\theta_1 = -0.5°$, $\Omega_1 = 0.75 \frac{m}{s}$, $\psi_2 = 15°$

Here we see quite an interesting outcome, as the predicted curve actually explains the variance better than the original fit. Although this might seem impossible at first glance, the fit is behaving as expected. The mysterious behaviour should instead be explained as a limitation of the evaluation method, for which we chose to apply the coefficient of determination. The fit, applied through Scipy [17], is, as previously mentioned, determined by maximum likelihood estimation [13] and is therefore not meant to find the lowest possible R^2. Although it is out of the scope of this project to explain exactly how this works, it is important to note why this deviation occurs. Intuitively it should be clear that the prediction follows the data, and the R^2-value support this suspicion being over 0.92, which statistically proves a strong correlation. Ultimately this proves that the methodology, although not perfect, is good enough for the purpose, and we can proceed to the optimization.

7 Analysis

With the network now being trained, we can begin to explore how we can apply the predictions to optimize the two-turbine setup. One important thing to note for this process is that the turbine loads might be correlated with the total power output. Now imagine that we only optimize regarding the total power output and that the network has rightly predicted the optimal configuration. Though such an optimization might drastically improve power production, it does not consider the fatigue and, consequently, how this might affect the lifespan of the turbine. Ultimately it must be considered whether it is worth allowing higher loads to gain additional power production, assuming these are related. Since we are mostly interested in the effect of the yawing, we will start off by setting $\theta_1 = 0°$ and $\Omega_1 = 90\ rad/s$ to see how much the yawing can influence the power production when considering two turbines.

7.1 Yawing effects for $s = 6R$

We can investigate the relation between power gain and loads, by plotting both as a function of the yawing angles. Doing so for the distance of $6R$ yields the contours illustrated in figure 7.1

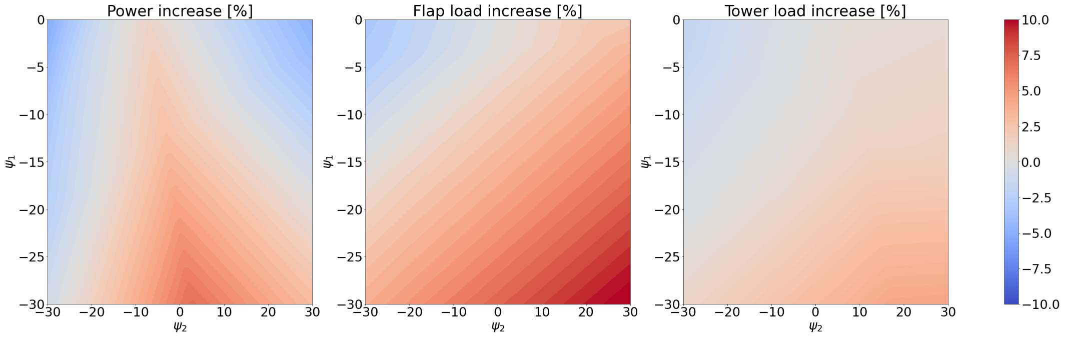

Figure 7.1: Contour plot of power/load gains as a function of yawing angles for $s = 6R$, $\Omega_1 = 90\ rad/s$ & $\theta_1 = 0°$

First, we should note that the normal operation settings of $\psi_1 = 0$ and $\psi_2 = 0$ have a gain of 0 for all measures, as this is our point of comparison. What is very interesting is that this is clearly not the best combination regarding power gain, which is immensely important since it confirms that WFFC is indeed beneficial for power production. To answer if it makes sense to apply when also considering loads is a slightly more complicated question, and the answer is not yet clear. However, before investigating this, it makes sense to comment on these contours and relate them to a more physical understanding. Firstly considering the power graph, we see a clear trend of increased power production for larger negative yawing of the first turbine. This is logical, considering that large angles redirect the wake more, which benefits the second turbine. Additionally, we reflect on figure 4.1 from section 4.1, where it was determined that the consequences of yawing were relatively small, which is compliant with the results. For the second turbine, we see quite an interesting trend, where yawing positively seems to be slightly more beneficial than negative angles, which might not immediately make sense. To understand why this is the case, we should consider the setup and illustrate how we expect the wakes to behave. Such a sketch can be seen in figure 7.2.

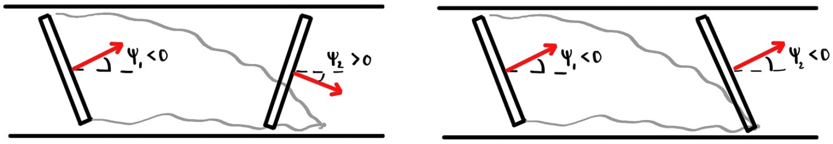

Figure 7.2: Sketch of difference between second turbine yawing in wake

From the sketch, it should be clear how we might have higher power from opposite yawing angles since the wake is more perpendicular to the turbine. Turbines are optimized for this kind of inflow, and it, therefore, makes sense that this would be better operating conditions. In return, we might also expect higher loads, which is also the case, even more so than the power. The tower load is generally not very interesting since it does not deviate significantly from normal operation. We should remember, however, that these results do not consider changes in rotational speeds and pitching of the blades for the first turbine and that these graphs might change once this is included. To see if it makes any difference, an identical plot is seen in figure 7.3, where these parameters are optimized at each point for maximum power.

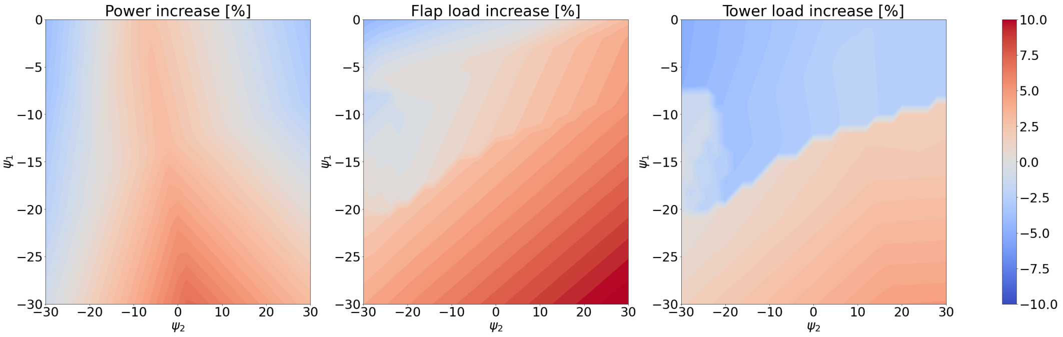

Figure 7.3: Contour plot of power/load gains as a function of yawing angles for $s = 6R$

A few interesting things are happening here. Firstly, it does not seem evident that changing the pitch and rotational speed for the first turbine has any significant impact, and we must conclude that yawing is by far the most beneficial form of WFFC out of the 3 control parameters. This does not imply, however, that pitching and changing rotational speed do not have any effect at all. If we were to closely compare figure 7.1 and figure 7.3, we would see that the power for the optimised case has a slightly wider positive contour, meaning that the yawing angles are marginally less sensitive. Also, the flap loads especially seem to be lower, although the distinction is less clear generally. What is interesting is that there seem to be ”lines” where the loads suddenly change, suggesting a steep change in either pitching or rotational speed. This could be smoothed out by evaluating more configurations at each point, but would also drastically increase the computational time. Ultimately, we can conclude that the power and load gains are somewhat correlated for this distance, and how much more power we can produce is therefore limited by how much load increase, we are willing to allow. Before looking further into this relation, however, we should compare these results with a longer and more realistic spacing.

7.2 Yawing effects for $s = 14R$

As a normal distance, we will look at $s = 14R$, and use this for comparison. The same plot of $\theta_1 = 0°$ and $\Omega_1 = 90\ rad/s$ for $s = 14R$ can be seen in figure 7.4

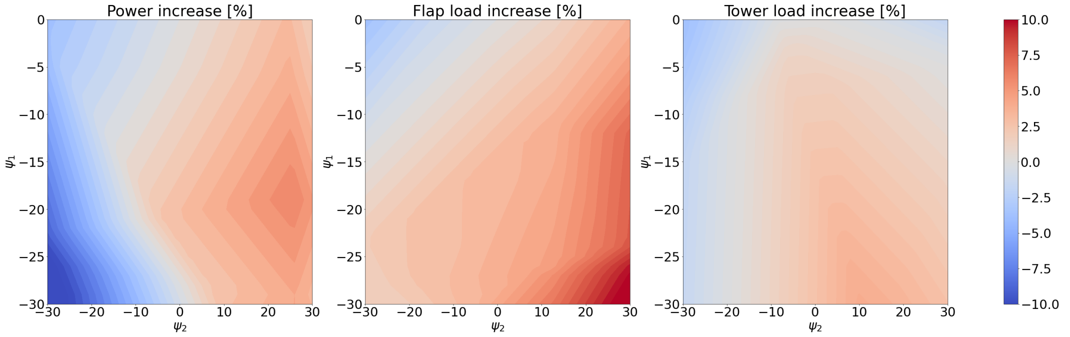

Figure 7.4: Contour plot of power/load gains as a function of yawing angles for $s = 14R$, $\Omega_1 = 90\ rad/s$ & $\theta_1 = 0°$

What is immediately clear between the graphs is that the wakes behave differently at this distance. Unlike the close distance, where the second yawing angle was very sensitive, we now have a much more tolerant range. For the yawing of the first turbine, we find an optimal angle $\psi_1 \approx 20°$, which is importantly within the range of the training data. Why this is relevant is discussed further in section 8.3.7. Another interesting trend for the second yawing is that it now prefers much higher angles than before, finding a maximum at around $\psi_2 \approx 26°$. To see if it is still the case for the optimized version, we plot this as well, as seen in figure 7.5

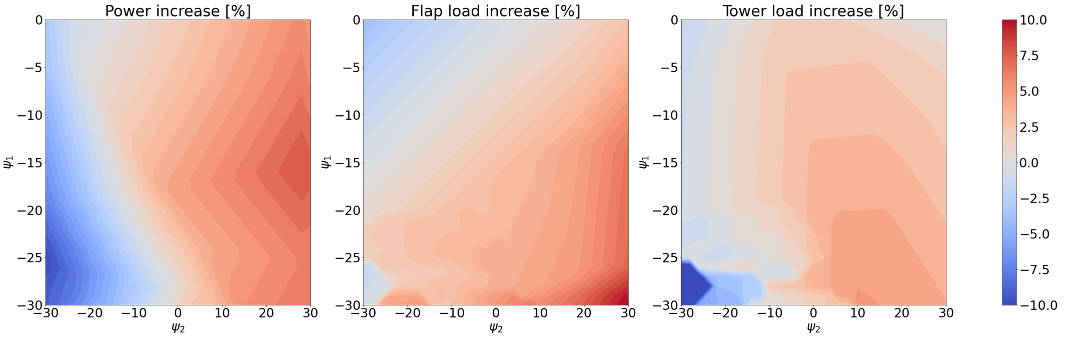

Figure 7.5: Contour plot of power/load gains as a function of yawing angles for $s = 14R$

A slight improvement is seen compared to figure 7.4, which makes sense considering the optimized control parameters. Although we still have the general shape from the locked case, we also see that the preferred yawing angle for the first turbine is slightly lower. Although it is difficult to interpret from the plots, the increase seems slightly higher than the short distance in figure 7.3, which is surprising considering that the wake is naturally breaking down between the turbines. The reason for this is unclear, and we should, therefore, be critical of the results and evaluate the uncertainties before making any conclusions.

7.3 Power as a function of load constraints

Plotting the possible power gain as a function of load constraints helps us visualising the relation between power and loads and how these compare between the distances. This plot can be seen in figure 7.6

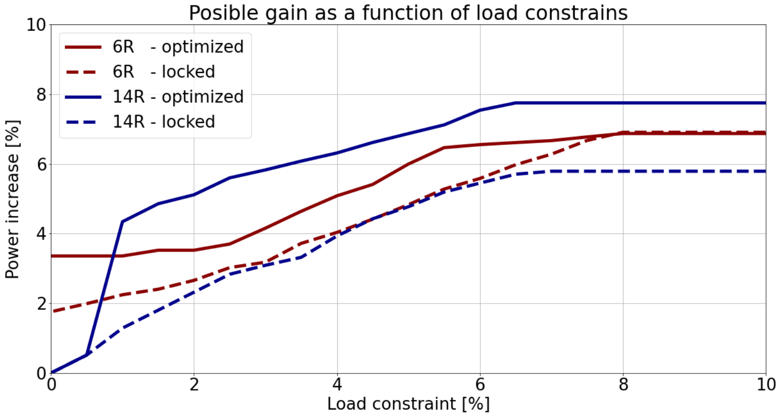

Figure 7.6: Maximum power gain as a function of load constraints

Firstly the previously mentioned suspicion that the longer distance outperforms the short distance immediately shows true. Though it is difficult to explain from a physical standpoint, it might simply be that the flow is so turbulent that it is unlikely to yield any meaningful results at this distance. If this is the case, we might expect large tails on the Weibull distributions, which will show when evaluating the results with their associated uncertainty. Besides the difference between the distances, we primarily notice that the locked cases perform marginally worse than their optimized counterpart. This deviation is what we would expect when applying the knowledge from the contour plots, and we confirm the conclusion that yawing is the most beneficial parameter to optimize. It is also intriguing how pitching and changing rotational speed seem to have no impact for the short distance when disregarding load constraints. For the long-distance, however, the impact is relatively clear, which is likewise puzzling from a physical point of view. These results generally seem a little unpredictable, which suggests that the uncertainty is not insignificant. It is also worth looking into which control parameters are applied for the different load constraints. Evaluating a few noteworthy constraints, we get the configurations displayed in table 7.1

	6R					14R				
Constraints	ΔP_{total}	ψ_1	Ω_1	θ_1	ψ_2	ΔP_{total}	ψ_1	Ω_1	θ_1	ψ_2
0%	$3.4 \pm 6.3\%$	$-6°$	$90rad/s$	$3°$	$-8°$	$0 \pm 4.5\%$	$0°$	$90rad/s$	$0°$	$0°$
2.5%	$3.7 \pm 6.3\%$	$-18°$	$90rad/s$	$2°$	$-4°$	$5.6 \pm 4.6\%$	$-2°$	$90rad/s$	$-0.5°$	$24°$
5.0%	$6.0 \pm 6.4\%$	$-28°$	$90rad/s$	$3°$	$0°$	$6.9 \pm 4.7\%$	$-16°$	$90rad/s$	$-0.5°$	$18°$
7.5%	$6.8 \pm 6.2\%$	$-30°$	$90rad/s$	$0.5°$	$0°$	$7.8 \pm 4.7\%$	$-16°$	$90rad/s$	$-0.5°$	$26°$
10%	$6.9 \pm 6.0\%$	$-30°$	$90rad/s$	$-0.5°$	$0°$	$7.8 \pm 4.7\%$	$-16°$	$90rad/s$	$-0.5°$	$26°$

Table 7.1: Table of optimal control parameters for different load constraints

First, we notice that the configurations for high increases in power all match what we expect regarding yawing angles from the contour plots. What is most interesting to comment on here is the pitch and rotational speed since these are not shown in the previous plots. Rotational speed, however, is of little interest in this analysis, since there seems to be no benefit of slowing down the first turbine. Pitching does change slightly for the short distance but is generally preferred at a value of $theta_1 = 0.5°$. What is interesting is that this half degree of pitching is the only difference between the locked and optimized case for figure 7.1, which implies that pitching this tiny angle improves power by almost 2% for the long distance. Although it is difficult to prove that it is unrealistic, we should again be critical of this result, since the physical interpretation intuitively sounds unlikely. When looking at the uncertainties, we begin to understand why these mysterious results occur. For the short distance, we have a relative uncertainty of between 85% and 185%, which applies that the simulation data fluctuates heavily at this distance. Such a large uncertainty means that, although we might suspect an increase in power generally, it cannot be proven for load constraints under 7.5%. For the distance of $14R$, the results seem slightly better, but still too uncertain to draw any certain conclusions

8 Discussion

Throughout this thesis, there have been numerous assumptions and uncertainties, some of which have been commented on and some left untouched. This chapter aims to address and comment on these in greater detail and discuss how such factors could influence the results. Although it is perplexing to comprehend the propagation of uncertainties, we wish to discuss the general validity and debate how further research could improve the results.

8.1 General validity

In chapter 7, it was determined that a power increase of nearly eight percent was possible at both distances for certain combinations of control parameters. Although this is a relatively significant improvement, it still seems intuitively plausible to achieve. It is nice to see results that do not contradict our intuition, but still a vague argumentation for the validity from a scientific standpoint. To check whether the network output and optimization make any physical sense, we will look at a direct time series at one of the better configurations and see how the turbines behave. The closest simulation we have to the optimal point for the distance of $6R$ is case 9, where we have the following control parameters.

$$\psi_1 = -25°, \quad \Omega_1 = 0.84 rad/s, \quad \theta_1 = 1.3°, \quad \psi_2 = 4°$$

Taking an arbitrary time series and plotting the power production of both turbines with the power for normal operation yields the graph displayed in figure 8.1

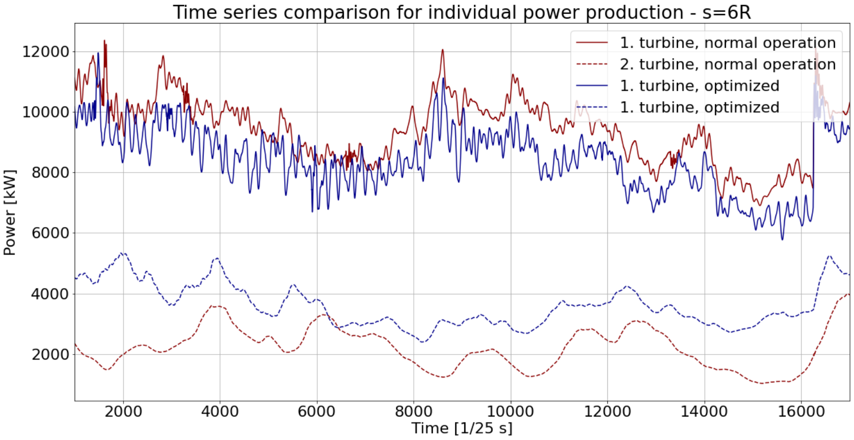

Figure 8.1: Power time series for $\psi_1 = -25°,\ \Omega_1 = 0.84 rad/s,\ \theta_1 = 1.3°,\ \psi_2 = 4°$

As seen, the power reacts exactly how we expect. For the first turbine, the power decreases as we yaw the turbine, but in return, the second turbine dramatically improves its performance. Notably, the second turbine is significantly worse than the first, and the wake effects are still highly present. It is difficult to interpret whether the improvement makes up for the exacerbation of the first turbine, so it makes sense to look at the combined power instead. This plot for the same turbines is seen in figure 8.2.

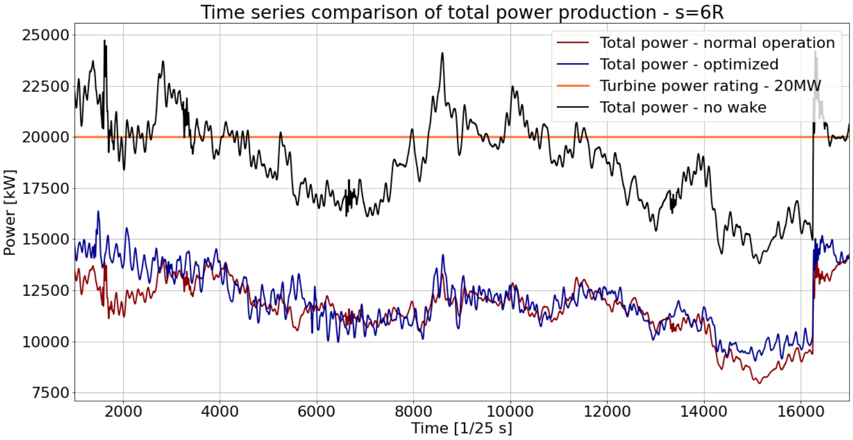

Figure 8.2: Combined power time series for $\psi_1 = -25°$, $\Omega_1 = 0.84 rad/s$, $\theta_1 = 1.3°$, $\psi_2 = 4°$

At first glance, it seems that the combined power is almost interchangeable, but this makes sense considering that we expect a maximum of eight percent difference between them. Additionally, we see that the optimized case is performing slightly better at certain points, especially at the start and end of the period. As seen in the figure, we also plotted theoretical power without wake losses, which fluctuates around the rated value of $20MW$. What is important to realize is that the loss is much greater than the improvement, and it, therefore, sounds realistic that one could gain a small increase by steering the wake. Ultimately we do not see any indication that the calculations should be incorrect, and we should instead discuss how accurate we expect them to be.

8.2 Comparison with previous studies

A recent paper by Debusscher [5] did a similar analysis for a much smaller turbine [12]. His analysis determined that the maximum power gain available was just under three percent, at a distance of $6R$. In chapter 7, we saw results of approximately eight percent, which is a substantial deviation relative to his results. What is notable in that comparison is that the optimal yawing angle of the first turbine, for maximum power, in his paper was approximately 23 degrees. This suggests that our turbine [9] is less sensitive to yawing, which would explain both the higher power gain and the high value for the optimal angle. For the loads, it seems that Debusscher [5] generally determined a higher increase, for the optimised case, of about 10% and 40% on the first and second turbine, respectively. In this thesis, we considered loads as a collective quantity, which partly explains the difference. The remaining deviation is most likely due to the different turbines, although it might make sense to investigate this further.

8.3 Uncertainties

Starting by recognizing the most significant uncertainties and creating an overview of how they affect each other, we can comment on them chronologically.

8.3.1 Use of mechanical power

From the way the data is simulated through Flex5, we need to recognize a fundamental difference between the first and second turbine. For this analysis, the Flex5 calculations were based on the flow data for both turbines, but it is also possible for the first turbine to be directly coupled to the actuator lines. Wanting to alter control parameters on one turbine creates a problem since we are forced to turn off the controller, meaning that the electrical power cannot be directly extracted. Instead, we must calculate the mechanical power from the torque and rotational speed, as seen in equation 2.15. The problem with this approach is that the turbine cannot convert all the mechanical power to electrical energy. Although it would be simple to correct for this imperfection, if we assumed the relative loss constant, there was not enough time in this project to repeat the simulations. Therefore, the actual power production will be slightly lower than portrayed whenever the first turbine is involved. However, since this is the case throughout the thesis, it should not matter significantly since it will more or less cancel out during the comparisons. It would be a definite implementation for future work, but for now, we assume that it does not have any significant impact on the results.

8.3.2 Weibull fitting

In section 5.2, it was decided to use a Weibull CDF fit to represent the data. This was necessary to reduce computational time, as the available resources could not handle the available data. In total, this assumption reduced the simulated data from approximately 1.640.000.000 to 4.500 data points by representing each time series as an average and each collection of averages as a cumulative distribution function. One thing to note is the original data fluctuations have not been considered. From figure 2.4, however, these seem relatively consistent, so we assume this to be a reasonable course of action for a long time series. It should also be commented that power has a R^2-value around 0.9 and still contains an uncertainty, which was completely disregarded for further analysis. As previously mentioned, this suggests that a Weibull is a good, but not perfect, fit. For future work, it might make sense to investigate if there are other more accurate ways to fit this distribution or find computers powerful enough to brute force the calculations.

8.3.3 Median/Mean

Going from the Weibull representation to a more practical value for the turbine outputs, we could utilize two primary methodologies: the median and mean, respectively. For a Weibull distribution, the mean is generally larger, which is amplified the more separated the data gets. For the case of wind turbines, it seems reasonable to use the median since it represents the most common occurrence, but it would be interesting to see how using the mean would alter the results and if it would have any impact on the optimal configurations.

8.3.4 Network accuracy

In chapter 6 a relatively simple network was applied. Though the accuracy of test data seen in figure 6.2 is relatively high, we still have some outliers of up to two percent. This is not insignificant when combined with the uncertainty from the distribution. An obvious way to enhance this performance would be to simulate additional data or model a more complex network.

8.3.5 Distances

In chapter 7, we looked at two distances of $6R$ and $14R$, representing a short and normal distance, respectively. The choice of distances was primarily based on the results of a previous study [5], which turned out not to be as similar as expected. It would therefore be interesting to do a similar analysis and see how this turbine changes behaviour over a broader range of distances.

8.3.6 Prediction steps

When plotting the contour plots in chapter 7, we saw some interesting behaviour of the load plots for the optimized parameters. As previously mentioned, this could be smoothed out by refining the optimization but has not been done as a limitation of the available resources. Currently, each of the six contours is based on 1.891 points of yawing combinations, each evaluated on 56 combinations of Ω_1 and θ_1. This requires approximately 106.000 network predictions per graph, which equals multiple hours of computation on a high-performance computer. The number of predictions would quadruple should we wish to double the number of steps for both Ω_1 and θ_1, which is not practical to compute. For future work, it might make sense to compute a refined version on a designated server and create a smoother plot. It might also make sense to investigate which parameter changes across these lines and relate this to a physical understanding.

8.3.7 Simulation data

A notable trend from chapter 7 is that the maximum power gain was consistently at the maximum yawing angle of the first turbine, at the distance of $6R$. This is not something we wish to see since there might be a more optimal point that we have not investigated. Unfortunately, neural networks generally perform very poorly when predicting outside the range of the training data. Therefore, we cannot expect accurate results if we explore a broader range since the simulation data is limited from -30 to 0 degrees. For future work, it will be essential to perform more CFD simulations, which will enable us to investigate this, as well as increase the network accuracy in general. It would also make sense to apply the knowledge found in this thesis to better determine the configurations for new simulations. For example, it seems unnecessary to simulate more cases with different rotational speeds when we expect this to be constant. However, yawing and pitching found values at the end of the range and should therefore be expanded.

9 Conclusion

Chapter 1 illuminated why wind energy is important in recent times based on a report from the International Energy Agency [8]. It also introduced the problem of wakes in wind farms and suggested how WFFC, as presented in this thesis, can be relevant to alleviate this problem. The methodologies of how we might investigate such optimisations were then presented mathematically to lay a foundation of understanding when applying the methods. In practice, these methods are primarily implemented through scripts, which generate the data necessary for the analysis. Building an underlying sense of how the turbines behave is useful for intuitively validating the results, and a single turbine was therefore considered under different conditions to learn about its preferred operation. The general control was analysed by applying a uniform inflow with different velocities, and dimensionless numbers were applied to evaluate how much changing control parameters affect the individual turbine.

The simulation data is the foundation of the analysis, so it is important to clarify which data is applied. The different cases for the simulations were presented in chapter 5, where it was also introduced how it would be simplified and represented through a Weibull distribution.

Based on the Weibull distribution, it was possible to train the surrogate, which generally performed well, considering the limited training data. Although it was not perfect, especially for the flap loads, this uncertainty seemed relatively low compared to the natural spread of the distribution. Generally, the analysis showed great promise of increasing power, even for lower load constraints, but was somewhat shadowed by the high uncertainties. From the discussion, we investigated a concrete time series, which behaved exactly as we would expect. Here we saw that it was indeed possible to achieve a slight improvement in total power by yawing the turbines. Additionally, the results were compared to a similar paper, which was slightly more conservative but got results of a similar proportion.

Ultimately we must conclude that there is no evidence for the analysis to be incorrect but that it is difficult to prove a certain increase due to the high uncertainties. Though this thesis may not have proven direct evidence of a beneficial optimisation, it showed a clear possibility of such and proposed multiple ways of improving the methodology and, thereby, the accuracy of the results. First, it is believed that additional CDF simulations would be necessary to determine a maximum, primarily, but not exclusively, for short-distance optimisations. Most importantly, however, should the CDF simulations be extended to prolong each period, to get a more accurate average and reduce the spread of the Weibull distributions. Such a refinement might not alter the results, but it should lower the uncertainty and strengthen the statistical argument for the increase. Generally, this thesis should not be seen as a direct application of wind farm flow control but rather as an introduction to the methodology and a basis for further research.

Bibliography

[1] *Actuator disk*. https://www.sciencedirect.com/topics/earth-and-planetary-sciences/actuator-disk. Accessed: 2022-11-09.

[2] *Backpropagation calculus*. https://www.3blue1brown.com/lessons/backpropagation-calculus. Accessed: 2022-11-20.

[3] *Betz limit*. https://energyeducation.ca/encyclopedia/Betz_limit. Accessed: 2022-11-09.

[4] *Blade element theory*. http://www.aerodynamics4students.com/propulsion/blade-element-propeller-theory.php. Accessed: 2022-11-20.

[5] Søren Juhl Andersen Charles Marie J Debusscher Tuhfe Göcmen. "Probabilistic surrogates for flow control using combined control strategies". In: (2022).

[6] *EllipSys3D*. https://the-numerical-wind-tunnel.dtu.dk/ellipsys. Accessed: 2022-09-24.

[7] *Fatigue load calculation*. https://toolbox.pages.windenergy.dtu.dk/WindEnergyToolbox/fatigue_tools/fatigue_nb.html. Accessed: 2022-09-16.

[8] *How much energy does the world consume?* https://ourworldindata.org/energy-production-consumption. Accessed: 2022-09-16.

[9] *IEA10MW*. https://www.nrel.gov/docs/fy19osti/73492.pdf. Accessed: 2022-09-25.

[10] *Large eddy simulation*. https://en.wikipedia.org/wiki/Large_eddy_simulation. Accessed: 2022-11-20.

[11] R. Steijl M. Carrión M. Woodgate and G. N. Barakos. "Understanding Wind-Turbine Wake Breakdown Using Computational Fluid Dynamics". In: (2015).

[12] Søren Markkilde Petersen. "Wind turbine test Vestas V27-225 kW". In: (1990).

[13] Norhaslinda Ali Nawal Adlina Mohd Ikbal Syafrina Abdul Halim*. "Estimating Weibull Parameters Using Maximum Likelihood Estimation and Ordinary Least Squares: Simulation Study and Application on Meteorological Data". In: (2022).

[14] *Net Zero by 2050*. https://www.iea.org/reports/net-zero-by-2050. Accessed: 2022-11-07.

[15] *Power number*. https://en.wikipedia.org/wiki/Power_number. Accessed: 2022-11-09.

[16] *Rainflow counting*. https://www.sciencedirect.com/topics/engineering/rainflow-counting. Accessed: 2022-11-09.

[17] *Scipy weibull$_m$indocumentation.* `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.weibull_min.html`. Accessed: 2022-11-20.

[18] Anidhya Athaiya Siddharth Sharma. “Activation functions in neural networks”. In: (2020).

[19] Jon Juel Thomsen. “Modalanalyse for kontinuerte systemer”. In: (2021).