Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

John Håvard Aarvåg

Yaw Error Estimation on the NREL 5-MW Offshore Baseline Wind Turbine

Master's thesis in Cybernetics and Robotics
Supervisor: Morten Dinhoff Pedersen
Co-supervisor: Finn Matras
June 2023

NTNU
Norwegian University of
Science and Technology

John Håvard Aarvåg

Yaw Error Estimation on the NREL 5-MW Offshore Baseline Wind Turbine

Master's thesis in Cybernetics and Robotics
Supervisor: Morten Dinhoff Pedersen
Co-supervisor: Finn Matras
June 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Abstract

The advancement of computationally efficient and precise inflow dynamics, coupled with the growth in computational power, paves the way for exploring the relatively unexplored approach of state estimation for deriving the yaw error estimate.

This thesis provides the groundwork needed for implementing simulations fit for state estimation to obtain an estimate of the yaw error. This necessitates the development of a nonlinear process model that encompasses structural dynamics, inflow dynamics, torque control, a wind model, and the computation of forces and torques on the blades. Alongside the specifications of the wind turbine, these elements constitute a comprehensive nonlinear process model, which is subsequently linearized for state estimation using the Kalman filter and Extended Kalman filter.

The simulations conducted in this study demonstrate the Extended Kalman filter's ability to produce accurate yaw error estimates. These estimates are subsequently utilized in yaw control, highlighting the potential benefits of possessing a precise yaw error estimate.

Sammendrag

Utviklingen av beregningseffektive og nøyaktige inflow-dynamikker, sammen med økende beregningskapasitet, muliggjør undersøkelse av den relativt uutforskede tilnærmingen til tilstandsestimering for å finne et feilestimat for yaw.

Denne avhandlingen gir grunnlaget som trengs for å implementere simuleringer egnet for tilstandsestimering av yaw-feil. Dette krever utvikling av en ikke-lineær prosessmodell som omfatter strukturell dynamikk, inflow-dynamikk, momentkontroll, en vindmodell, og beregning av krefter og moment på bladene. Sammen med vindturbinens spesifikasjoner utgjør disse elementene en omfattende ikke-lineær prosessmodell, som deretter blir linearisert for tilstandsestimering ved bruk av både Kalman-filter og Extended Kalman-filter.

Simuleringene som er utført i denne oppgaven demonstrerer Extended Kalman-filterets evne til å produsere nøyaktige estimater av yaw-feilen. Disse estimatene blir deretter brukt i yaw-kontroll, for å understreke de potensielle fordelene ved å ha et nøyaktig estimat av yaw-feilen.

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Morten D. Pedersen, for his invaluable guidance and exceptional mentorship throughout the process of completing this thesis. Morten's unique perspective and approach to problem-solving have profoundly influenced my thinking and shaped the direction of my research. His ability to identify the hidden potential in any result has been instrumental in pushing the boundaries of my work. I am truly grateful for his patience, insightful feedback, and constant encouragement.

I am also indebted to my co-supervisor, Finn Matras, for his exceptional dedication and unwavering support throughout this journey. Finn's commitment to my academic progress and his willingness to go the extra mile has been truly remarkable. Despite the late nights and countless Teams messages, Finn always found the time to explain complex concepts in a way that made them accessible and comprehensible. His patience and willingness to guide me through advanced topics have been invaluable.

I would also like to thank my friends and family for their unwavering support and understanding throughout this demanding period. Their encouragement and belief in me have been vital sources of motivation. I am truly grateful for their love, patience, and understanding.

Contents

List of Tables

List of Figures

Chapter 1

Motivation

1.1 Introduction

Renewable energy, encompassing resources like wind and solar power, has gained significant relevance due to its potential to reduce greenhouse gas emissions by providing alternatives to burning fossil fuels. While the conscious use of renewable energy has been a relatively recent development, the utilization of natural resources for energy has been around since early humans discovered fire. According to the Global Wind Energy Council's Global Wind Report 2022, the total global wind power capacity in 2021 reached 837 GW. This capacity is "helping the world avoid over 1.2 billion tonnes of CO2 annually; equivalent to the annual carbon emissions of South America" [4]. This statistic underscores the significant impact of renewable energy sources in mitigating the effects of climate change.

In 2021, the total global wind power capacity increased by 93.6 GW, and GWEC aims for further increase in the coming years with a projected total capacity illustrated below.

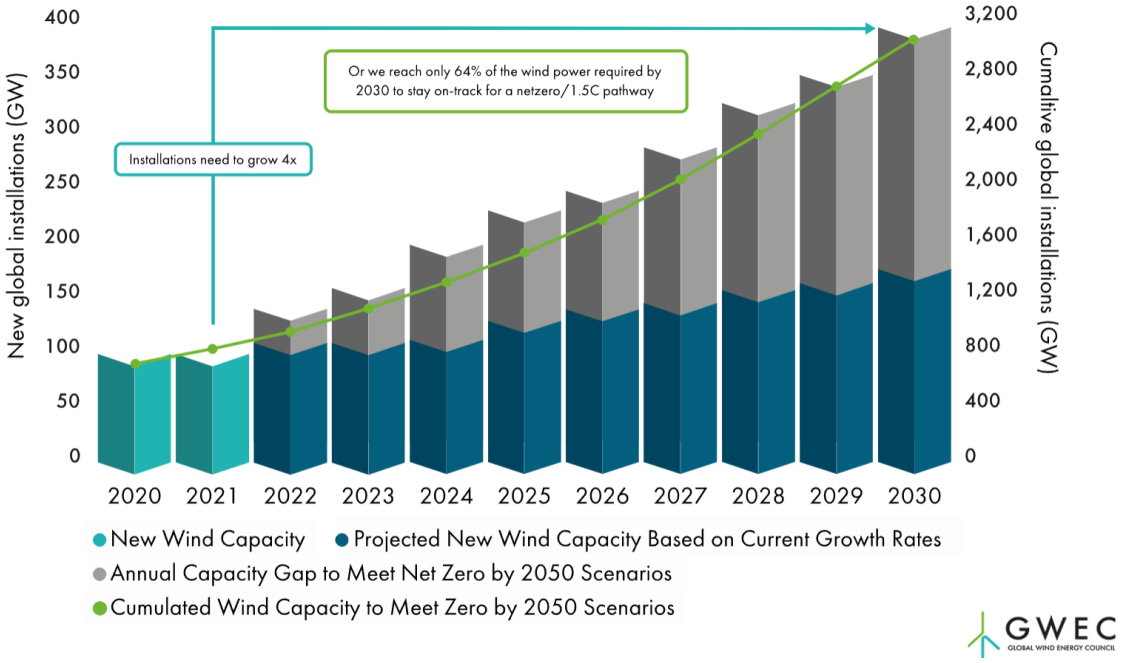


Figure 1.1: Trend wind energy. From [4]

With the increased capacity the importance of efficiency grows. The more efficiently the kinetic energy of the wind can be converted into electrical energy, the greater the benefit from the total power capacity. The kinetic energy of the wind can be expressed by the following equation:

$$E = \frac{1}{2}\rho A U_\infty^3 \tag{1.1}$$

where ρ is the air density, A is the swept area and U_∞ is the free-stream flow of the wind. 100% efficiency would imply zero final velocity of the wind after passing the turbine which is not realistic. Betz's law [5] suggests that no turbine[1] can capture more than 16/27 (59.3%) of the kinetic energy in wind. This assumes constant linear velocity consequently neglecting rotational forces such as vortex shedding, wake rotation and turbulence caused by drag which will further reduce the maximum efficiency [6].

A plethora of research in the wind energy field has been and is ongoing to increase the efficiency, the approaches range from advanced control schemes[7] to structural changes[8]. Yaw control is an essential control system for proper wind turbine operation [9]. The goal of this control system is generally to align the nacelle, which is the housing at the top of the wind turbine that contains the generator and other mechanical components, with the wind to maximize power capture.

Yaw error is the misalignment between the nacelle of the wind turbine and the wind direction. For a single wind turbine, yaw error during operation leads to sub-optimal power extraction. This becomes clear in the steady state case, where the generator shaft torque is equal to the respective aerodynamic torque such that the generator shaft power can be expressed as:

$$E(\Omega) = \frac{1}{2}\rho A C_p^* U_\infty^3 \cos(\psi) \tag{1.2}$$

where C_p^* is the optimal power coefficient, U_∞ is the free-stream flow of the wind, Ω is the angular velocity of the rotor and ψ is the yaw angle of the wind. It is evident that $\psi = 0$ leads to maximum power extraction, however controlling the angle of the wind is not feasible. The objective of the yaw control is then to yaw the wind turbine such that the angle between the yaw of the turbine θ and the yaw of the wind ψ, that is the yaw error, is zero. Note that θ is the direction the nacelle is facing, Figure 3.2 illustrates the wind turbine in a NED reference frame with angles θ and ψ defined in relation to the BODY frame. Thus, by minimizing the yaw error maximum power extraction can be achieved.

In the case of a wind farm this might not be the case as downstream flow might benefit from a non-zero yaw error for wake deflection [10, 11]. Whichever the case, the yaw error must be controlled to get optimal power extraction, either to zero or a set value.

Proper yaw control, which relies on good yaw error estimates, is also crucial for keeping the wind turbine within its operating range. Operation outside of the operating range can lead to forced shut downs and unnecessary stress on the wind turbine. Downtime due to failures in the yaw system comprised 13.3% of the total downtime of wind turbines in Swedish power plants in 2000-2004 [12], which motivates proper yaw control.

To be able to control the yaw error to a desired value, information about the yaw error must be obtained. The wind-vane is a common tool for yaw error measurements, however, as the blades of the turbine rotate it can create a wake disrupting the incoming flow to the wind-vane, which is typically mounted on the nacelle [13, 14]. As an alternative light detection and ranging(LIDAR) based yaw measurements has seen some traction in the last few years [15, 13, 11, 16, 17]. The usage of LIDAR and wind-vanes often comes with comprehensive correction schemes. As an example [18] stated that measurements using a spinner-based LIDAR along with correction for wind shear and temporal averaging produced yaw error estimates within a few degrees of the true value. However, it is mentioned as a shortcoming that the estimates only consider situations with varying magnitude of wind shear.

The challenges and limitations of yaw error estimation using wind-vane and LIDAR are further discussed in Section 1.3. From this, it is evident that there is a need for a more reliable and

[1] Betz limit is based on an open-disk actuator i.e the limit applies to the cross-section of the structure.

accurate method for obtaining information on the yaw error. This thesis aims to address this gap by exploring state estimation as a potential solution for inaccurate yaw error estimation in wind turbines. Specifically, we will investigate the usage of Extended Kalman filtering for yaw error estimation. The objectives of the thesis will be further elaborated in the next section.

1.2 Objective

The main objectives in this thesis can be segmented into the following parts:

- Explore the usage of the Extended Kalman filter for yaw error estimation.
- Explore available wind models.
- Develop a nonlinear process model of the wind turbine.
- Implement the nonlinear process model in a simulation.
- Linearize the process model.
- Make sure that the resulting linear system is observable with chosen measurments.
- Implement the Kalman filter for initial state estimation.
- Implement the Extended Kalman filter for state estimation.
- Utilize the yaw error estimate for yaw control.
- Compare the results.

These objectives will be reached by using `Julia` [19] for simulations and measurements from NREL Flatirons Campus (M2) [20].

The learning objectives related to this thesis are:

- Learn how Blade element theory(BET) can be used to approximate the aerodynamical forces and torques on a wind turbine.
- Learn how the BET method can be numerically linearized with respect to the dynamics involved in the linear and angular velocities.
- Get a first look at model identification of a stochastic process using data based optimization on a Gauss-Markov style model.
- Learn how model augmentation can be implemented to penalize statistically unlikely changes in the state estimate.

1.3 Background

This section presents a review of the literature relevant to the first two objectives of this thesis: exploring the usage of the Extended Kalman filter for yaw error estimation and investigating available wind models. The aim of this review is to provide an overview of the current state of the field and identify gaps in the existing research.

1.3.1 Yaw error estimation

This section delves into the yaw error estimation schemes found in the literature. As mentioned in the introduction, yaw error estimation is a critical aspect of wind turbine operation. Commonly, these estimations are based on measurements obtained using a wind-vane or LIDAR, often coupled with a correction scheme to account for various factors that may affect the accuracy of the measurements. The following discussion will explore some yaw error estimation schemes.

- [14] proposed a compensation scheme of the wind-vane measurements based on data-mining using 10-minute averages of data from Wind turbine supervisory control and data acquisition(WT SCADA) and removing outliers.
- [21] introduced rotor speed dependent yaw control, applying corrections to the wind-vane measurements depending on the rotor speed. The correction scheme was derived by comparing data from a meteorological(met) mast with the wind-vane measurements. The correction scheme is highly dependent on the wind turbine and wind conditions in question. [15] Suggested that LIDAR could be used if a met mast is not available, to find the rotor speed dependent correction.
- Similar to [14], [22] and [23] both utilized SCADA data to obtain correction schemes for the wind-vane measurements.
- [24] suggested estimating the yaw error by looking at the error between the rotor speed and expected rotor speed according to MPPT, see Subsection 2.5.3. This strategy relies on accurate wind measurements and it also requires rotation of the nacelle to obtain the sign of the yaw error.

These are some examples of yaw error estimation schemes found in the literature. A common theme among these approaches is the focus on reducing the yaw error, with less emphasis on quantifying it. Despite this, these studies reported a reduction in yaw error as a result of their proposed methods, underscoring the potential of these approaches to improve yaw error estimation. Interestingly, no usage of the Kalman filter was found for yaw error estimation in the literature, suggesting that this approach represents new territory in this field.

1.3.2 Wind models

This section summarizes common methods for generating a wind signal for wind turbine simulation.

- **Von Karman Turbulent Power Spectral Density:** Introduced by [25], this model is widely used in turbulence studies. It provides a mathematical description of how energy is distributed over different frequencies in a turbulent flow.
- **Spectral Factorization:** [26] shows how spectral factorization can be utilized to obtain a shaping filter, which can be used for artificial turbulence generation. This method allows for the creation of a wind signal that has a specific power spectral density.

The wind models found primarily describe turbulent flow fluctuating around a mean value. As seen in the measurements [20], real wind exhibits largely varying yaw. This highlights a gap in the current wind modelling approaches, as no model was found that appropriately describes the changes in the yaw of the wind.

1.4 Outline of thesis

The thesis is structured into the following chapters.

I - Motivation

This chapter introduces the importance of yaw error estimation in wind turbines for maximizing power extraction and ensuring efficient operation. It highlights the current methods for yaw error estimation, such as wind-vane and LIDAR measurements, and their limitations. The chapter also explores the potential of using Extended Kalman filtering for more accurate yaw error estimation, suggesting that there exists a research gap in the field. The chapter concludes by outlining the main objectives of the thesis, which include exploring the usage of the Extended Kalman filter, developing and implementing process models, and comparing the results. The chapter also sets the stage for the subsequent chapters that delve deeper into these topics.

II - Modelling

Chapter 2 introduces the necessary dynamics for modelling a wind turbine generator. This includes detailing the relevant structural and inflow dynamics, and presenting a method for calculating the forces and torques acting on the blades. The specifications of the wind turbine and the control systems in use are also considered in this chapter.

III - Wind modelling

In this chapter, two distinct methods for wind modelling are introduced. The first method, here referred to as the Cartesian wind model, incorporates turbulent flow on top of a mean component in two directions, resulting in a wind model with varying free-stream velocity and yaw. The second method presents a novel approach, employing a Gauss-Markov style identification strategy based on measurements. This model characterizes the changes in yaw and free-stream flow of the wind, hence it is referred to as the Yaw-Velocity wind model.

IV - Process model

The dynamics presented in chapter II and III are condensed into two different process models. The first model makes use of the Cartesian wind model, while the second makes use of the Yaw-Velocity model. Initially, a nonlinear process model is derived. Following this, the necessary linearization is carried out to yield a linearized process model.

V - State estimation

This chapter will introduce different levels of state estimation from using a constant scaling between two states to an Extended Kalman filter. It provides an overview of the principles behind these methods and their implementation. The chapter also discusses how these different estimation techniques can be applied to the process models developed in the previous chapter.

VI - Results

In this chapter, the simulation results are presented and thoroughly discussed. The chapter begins with the application of the Kalman filter to the Cartesian wind model, followed by the implementation of the Extended Kalman filter on both wind models. Additionally, a simple yaw control is

implemented to illustrate the practical utility of the yaw error estimate. The results are analyzed in the context of their implications for wind turbine control and performance.

VII - Conclusion

The final chapter concludes the thesis, summarizing key findings and results. The chapter concludes with a discussion of potential avenues for future research and improvements to the current work.

Chapter 2

Modelling

This chapter is dedicated to the comprehensive modelling of a wind turbine generator, a task that requires insight into various dynamic modes. The first section outlines the system structure of a wind turbine generator, which is divided into three basic modes: Structural Dynamics, Airloads, and Inflow Dynamics. Each mode represents an essential element in the operation of the wind turbine and is consequently separated for a more straightforward analysis of the dynamics. Following this overview, the dynamics of the respective modes are presented in individual sections. Subsequent to these sections, the control systems involved in a full-scale wind turbine are presented, enabling the incorporation of appropriate dynamic adjustments. The concluding section introduces the National Renewable Energy Laboratory 5MW (NREL) wind turbine, which is used for simulations in this thesis.

2.1 System structure

In [27], a well-known representation of aeroelastic dynamics is introduced, which suggests using three different modes in a feedback form similar to the one shown in Figure 2.1. This allows for separate modelling of the modes, which can then be analyzed independently.

The first mode, *Airloads*, uses Blade Element Theory (BET) to calculate the generalized force-torque vector $\boldsymbol{\tau}$ in $\mathbb{R}^6$, hereinafter termed as *airloads*, driven by the relative flow $\boldsymbol{\nu}$.

The second mode, *Inflow dynamics*, is based on the works presented in [28]. This model uses a novel approach to approximate the induction factor of inflow while guaranteeing correct steady-state behavior.

Lastly, the *Structural dynamics* mode describes how the airloads affect the physical system. This includes the dynamic responses of the wind turbine's components to the airloads. However, in this mode, only the inertial properties of the rotor are considered.

Together, these three modes provide a comprehensive representation of the aeroelastic dynamics of a wind turbine, allowing for a detailed analysis of its operation and performance.

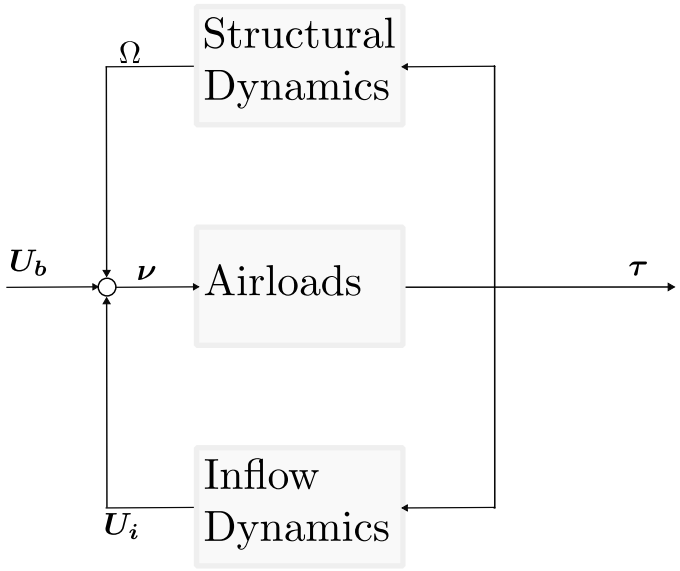


Figure 2.1: In this figure, $\boldsymbol{\tau}$ represents the airloads, while $\boldsymbol{U_b}$ denotes the linear velocities of the wind in the wind turbine's BODY frame. The induced inflow is represented by $\boldsymbol{U_i}$, and Ω is the angular velocity of the rotor. Lastly, $\boldsymbol{\nu}$ symbolizes the relative velocities. The terms used in this figure will be explained in more detail in the following sections.

2.2 Structural dynamics

The structural dynamics of a wind turbine involve the dynamic responses of its components to various loading conditions. These conditions arise from factors such as wind, gravity, and operational processes, which generate dynamic loads capable of causing vibrations, deflections, and fatigue in the components. Such effects can lead to component wear and, consequently, reduce the overall lifetime of the wind turbine [29]. Generally, wind turbine systems comprise five main physical components: the rotor, transmission, generator, support structure, and control system [30]. These components exhibit significant structural dynamics, which have to be accounted for in the design process. In Subsection 2.5.3 torque control and its effect on structural dynamics is briefly discussed. In this section, we examine the rotor dynamics and what information it can provide us to be useful in the process model.

2.2.1 Rotor dynamics

The dynamics of the rotor play a crucial role in the overall stability and performance of a wind turbine. These dynamics are significantly influenced by the aerodynamic forces and moments acting on the rotor. For a detailed explanation of these forces and moments, refer to Section 2.4. Furthermore, the inertial properties of the rotor are fundamental to its dynamic behavior [27].

One of the key principles governing rotor dynamics is Newton's second law for rotational motion. It relates the aerodynamic torque experienced by the rotor to its moment of inertia and angular acceleration as shown:

$$-M_z = J\dot{\Omega} \tag{2.1}$$

where M_z[1] represents the aerodynamic torque acting on the rotor in z, J is the rotor's moment of inertia and $\dot{\Omega}$ is the angular acceleration of the rotor. This equation illustrates how changes in the aerodynamic torque can lead to changes in the rotor's angular velocity. However, as the purpose of a wind turbine is to convert aerodynamical power into electrical power this will affect the net torque as shown in Subsection 2.5.3.

2.3 Inflow dynamics

The performance and efficiency of a wind turbine are linked to the characteristics of the wind it encounters. The relative flow, commonly referred to as airflow, is the relative velocity of the linear and angular velocities of the wind as perceived by the rotor system.

The incoming wind as perceived by the rotor is different from the free-stream wind as the wind turbine itself will cause distortions of the incoming wind. These distortions, commonly described as induced inflow, change or develop over time, a process captured by inflow dynamics.

The linear induced inflow is subtracted from the linear velocities of the incoming wind and the angular induced inflow is subtracted from the angular velocity of the rotor, as defined in Definition 2.3, to yield the relative flow.

To formalize these concepts, we introduce the following definition:

Definition 2.1. *Let the vector of linear, rotational and generalized velocities in* $\mathbb{R}^6$ *be denoted by:*

$$\boldsymbol{v} \triangleq \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}, \quad \boldsymbol{\omega} \triangleq \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad \boldsymbol{\nu} \triangleq \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{bmatrix} \tag{2.2}$$

In the context of wind turbine dynamics, this vector $\boldsymbol{\nu}$ will be referred to as the relative velocities.

The literature on inflow dynamics modelling is extensive, with several models proposed over the years. One of the early models for calculating induced inflow is the Rankine-Froude momentum theory [31]. More recent models include the Actuator Disk Vortex Theory (ADVT), which uses wake vorticity to compute the inflow at the rotor disk [32], and Computational Fluid Dynamics (CFD) models [33]. Other notable models are cited as well [34, 35, 36, 37, 38], please refer to [39] for a comprehensive summary of nonuniform inflow model developments in the time period 1926-1988.

The work by Matras and Pedersen, currently under review for publication [28], introduces a novel approach for dynamic inflow modelling in $\mathbb{R}^6$. This dynamic inflow model, which forms one of the fundamental components of our analysis in this thesis, is further detailed in Subsection 2.3.1.

2.3.1 Inflow as feedback

The following provides a concise overview of the dynamic inflow model utilized in this thesis, as detailed in [28].

The research objective of this paper is "to present a method for modelling the averaged induced inflow on one or multiple coplanar rotors in a manner that is feasible for real-time Model Predictive Control (MPC) and simulation."[28]. The resulting model is concluded to have a simple state-space form that has excellent computational efficiency while also begin validated against well-established theories in the applicable regions [35, 38].

Based on actuator plane vortex theory(APVT), presented in [37], the findings suggest that the inflow dynamics can be condensed into a fluid memory prefilter by assuming quasi-static behavior.

[1] The negative sign comes from how the torque is defined in Section 2.4.

This approach offers the advantage of a computationally light model that guarantees correct steady-state behavior while providing a reasonable approximation of the inflow dynamics. The implication of this is that this novel approach has led to a more versatile inflow model suitable for simulation.

Fluid memory prefilter: Single rotor system

Going into detail on the derivation of the dynamic inflow model is beyond the scope of this thesis. Instead the results can be condensed into the following system of equations:

Key Result 1.

$$\begin{aligned} Q^{-1}P(\chi)\dot{\boldsymbol{\sigma}} &= \hat{\boldsymbol{\tau}} - \boldsymbol{\sigma}, \\ \boldsymbol{\eta} &= L(\chi)\boldsymbol{\sigma} + D(\chi)\bar{\boldsymbol{\tau}}, \end{aligned} \tag{2.3}$$

In these equations, $\boldsymbol{\sigma} \in \mathbb{R}^5$ is an internal variable that represents the force corresponding to a given inflow at steady state, due to the quasi-static assumption. The matrices Q, P, L, and D define the system structure. Among them, P, L and D are dependent on the yaw error (χ), while Q remains independent of it. The aerodynamic force-torque vector, or airloads, is represented by $\boldsymbol{\tau}$, which is discussed in more detail in Section 2.4.

The normalized form of $\boldsymbol{\tau}$ is given by

$$\bar{\boldsymbol{\tau}} = \frac{\boldsymbol{\tau}}{\frac{1}{2}\rho\pi R^2 U_\infty^2} \in \mathbb{R}^6, \tag{2.4}$$

where ρ represents the air density, R is the rotor radius and U_∞ is the free-stream flow of the wind.

The reduced form of $\bar{\boldsymbol{\tau}}$, represented by $\hat{\boldsymbol{\tau}}$, comprises the first five elements of $\bar{\boldsymbol{\tau}}$:

$$\hat{\boldsymbol{\tau}} = [\bar{\boldsymbol{\tau}}_1, \bar{\boldsymbol{\tau}}_2, \bar{\boldsymbol{\tau}}_3, \bar{\boldsymbol{\tau}}_4, \bar{\boldsymbol{\tau}}_5]^\intercal. \tag{2.5}$$

The matrices Q, P, L, and D are defined as:

$$P = \begin{bmatrix} -\frac{8\cos(\chi)}{3\pi^2R^2(\cos(\chi)+1)} & 0 & \frac{8\tan(\frac{\chi}{2})}{3\pi^2R^2} & 0 & -\frac{\cos(\chi)}{\pi R^3(\cos(\chi)+1)} & 0 \\ 0 & -\frac{8}{3\pi^2R^2(\cos(\chi)+1)} & 0 & \frac{1}{\pi R^3(\cos(\chi)+1)} & 0 & 0 \\ \frac{8\tan(\frac{\chi}{2})}{3\pi^2R^2} & 0 & \frac{8}{3\pi^2R^2} & 0 & \frac{\tan(\frac{\chi}{2})}{\pi R^3} & 0 \\ 0 & -\frac{1}{\pi R^3(\cos(\chi)+1)} & 0 & \frac{64\sec^2(\frac{\chi}{2})}{15\pi^2R^4} & 0 & 0 \\ \frac{1}{\pi R^3(\sec(\chi)+1)} & 0 & -\frac{\tan(\frac{\chi}{2})}{\pi R^3} & 0 & \frac{64\cos(\chi)\sec^2(\frac{\chi}{2})}{15\pi^2R^4} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$Q = \begin{bmatrix} -\frac{1}{2\pi R^2} & 0 & 0 & 0 & -\frac{8}{3\pi^2R^3} & 0 \\ 0 & -\frac{1}{2\pi R^2} & 0 & \frac{8}{3\pi^2R^3} & 0 & 0 \\ 0 & 0 & \frac{1}{\pi R^2} & 0 & 0 & 0 \\ 0 & -\frac{8}{3\pi^2R^3} & 0 & \frac{4}{\pi R^4} & 0 & 0 \\ \frac{8}{3\pi^2R^3} & 0 & 0 & 0 & \frac{4}{\pi R^4} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$L' = \begin{bmatrix} \frac{1}{-4\sec(\chi)-4} & 0 & \frac{1}{4}\tan(\frac{\chi}{2}) & 0 & -\frac{4\cos(\chi)}{3\pi R\cos(\chi)+3\pi R} & 0 \\ 0 & \frac{1}{-4\cos(\chi)-4} & 0 & \frac{4}{3\pi R\cos(\chi)+3\pi R} & 0 & 0 \\ \frac{1}{4}\tan(\frac{\chi}{2}) & 0 & \frac{1}{4} & 0 & \frac{4\tan(\frac{\chi}{2})}{3\pi R} & 0 \\ 0 & -\frac{4}{3\pi R\cos(\chi)+3\pi R} & 0 & \frac{\sec^2(\frac{\chi}{2})}{R^2} & 0 & 0 \\ \frac{4\cos(\chi)}{3\pi R\cos(\chi)+3\pi R} & 0 & -\frac{4\tan(\frac{\chi}{2})}{3\pi R} & 0 & \frac{\cos(\chi)\sec^2(\frac{\chi}{2})}{R^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} \frac{\sec(\chi)}{4} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\sec(\chi)}{4} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\sec(\chi)}{2R^2} \end{bmatrix}$$

The resulting output $\boldsymbol{\eta}$ is the induction factor of inflow which is used to find the induced inflow:

Definition 2.2 (Induced inflow)**.**

$$\boldsymbol{U_i} \triangleq \boldsymbol{\eta} U_\infty \tag{2.6}$$

The definition of induced inflow along with Key Result 1 concludes the summary of the dynamic inflow model. What remains is to define the relative flow.

Definition 2.3 (Relative flow)**.**

$$\boldsymbol{\nu} = \begin{bmatrix} \boldsymbol{U_b} \\ 0 \\ 0 \\ \Omega \end{bmatrix} - \boldsymbol{U_i}, \quad \boldsymbol{U_b} = \begin{bmatrix} U_s \\ 0 \\ U_n \end{bmatrix} \tag{2.7}$$

In this definition, $\boldsymbol{U_b}$ represents the linear wind velocity in the BODY reference frame of the wind turbine. The BODY reference frame is aligned such that U_s corresponds to the side wind along the x-axis and U_n corresponds to the normal wind along the z-axis. The specific orientation and details of these reference frames will be further discussed in Section 3.1. The term $\boldsymbol{\nu}$ represents the relative flow as described in the intro of Section 2.3.

2.4 Blade Element Theory:Airloads

The aerodynamic forces and torques over the blades of a wind turbine can be approximated using Blade element theory(BET) [40, 41]. This method breaks down the complex analysis of calculating the forces and torques acting on the individual blades into an integral of the forces and torques in each *blade element*. The blade elements, hence the name BET, are segments of the blade as depicted in Figure 2.2. Each blade element is treated as a two-dimensional airfoil, hence airfoil theory is applicable. The lift and drag of each blade element are then calculated using the relative velocities local to the blade element. The forces and torques on each blade element are then mapped to a stationary frame[2] defined in Figure 3.1 before being integrated to obtain the total

[2]Note this frame is in BODY as defined in Section 3.1.

force-torque vector $\boldsymbol{\tau}$. As the parameters are introduced a key realization is that the BET method can be described as a function of the linear and angular relative velocities, i.e. the relative flow. Lastly, the relation between BET and BEMT is briefly discussed.

2.4.1 Notation

Initially, let's establish the necessary notation to facilitate a complex derivation of the Blade element theory method.

Definition 2.4. *Let the vector of forces, torques and airloads be denoted by:*

$$\boldsymbol{\mathcal{F}} \triangleq \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}, \quad \boldsymbol{\mathcal{M}} \triangleq \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix}, \quad \boldsymbol{\tau} \triangleq \begin{bmatrix} \boldsymbol{\mathcal{F}} \\ \boldsymbol{\mathcal{M}} \end{bmatrix} \tag{2.8}$$

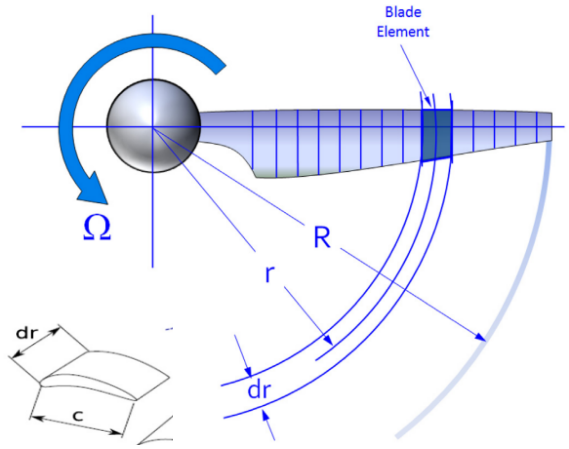


Figure 2.2: Blade element theory terms. Ω and R is the angular velocity and rotor radius, respectively. The blade elements chord is denoted by c and width dr. r is the radial station, i.e the distance from a blade element to the hub. This figure is a modification of two figures found in [40] and [41].

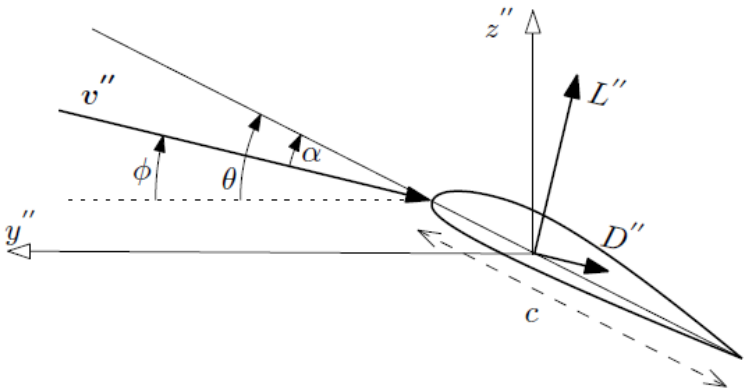


Figure 2.3: Notations used in derivation of BET, described below. From [2]

Here the notation is:

- $''$ denotes the blade element frame $'' \triangleq \begin{bmatrix} x'' = y'' \times z'' \\ y'' \\ z'' \end{bmatrix}$.
- $\boldsymbol{v}'' \triangleq \begin{bmatrix} v_x'' \\ v_y'' \\ v_z'' \end{bmatrix}$ is the relative velocity in $''$.
- L'', D'' is the lift and drag, respectively.
- $\phi \triangleq \tan^{-1}(\frac{v_z''}{v_y''})$ is the angle of the incoming relative velocity.
- θ is the sum of blade twist Θ and collective pitch deflection β
- $\alpha \triangleq \theta - \beta$ is the angle of attack and c is still the chord.

2.4.2 Blade element forces

Starting off, the lift and drag can be calculated using airfoil theory:

Definition 2.5 (Lift and drag)**.** *The lift and drag in the blade element coordinate system is given as:*

$$\begin{aligned} L'' &= \frac{1}{2}\rho|\boldsymbol{v}''|cC_l(\alpha)dr\,\boldsymbol{v}'' \times x'' \\ D'' &= \frac{1}{2}\rho|\boldsymbol{v}''|cC_d(\alpha)dr\,x'' \times \boldsymbol{v}'' \times x'' \end{aligned} \tag{2.9}$$

where C_l and C_d are the lift and drag coefficients.

The total generated force in the blade element frame then becomes:

$$dF'' = L'' + D'' \tag{2.10}$$

2.4.3 Acquiring the fluid flow on the blade cross section.

The forces on a single blade element has been established, however it depends on the fluid flow on the given blade cross section. A relation between the relative flow $\boldsymbol{\nu}$ and the fluid flow on a specific blade cross-section $\boldsymbol{v}''$ has to be established. To find this relationship three coordinate systems has to be established. A stationary frame at the root of the blade, a rotating frame with an origin coinciding with the stationary frame $'$, and a frame that is translated by the radial station from the rotating frame $''$[3]. This allows us to obtain the fluid flow on a specific blade cross section at any point of the blade rotation and at any given radial station. This is further explained in [2], however it can be summarized as:

Definition 2.6 (Fluid flow on the blade cross section)**.** *The $\boldsymbol{v}''$ can be found as:*

$$\boldsymbol{v}'' = \boldsymbol{T_v}\,\boldsymbol{\nu} \tag{2.11}$$

where $\boldsymbol{\nu}$ is the relative flow and $\boldsymbol{T_v}$ is the mapping from the stationary frame to the rotated and translated frame $''$.

$$\boldsymbol{T_v} = \begin{bmatrix} \mathcal{R}_z(-\Phi) & S(-r')\mathcal{R}_z(-\Phi) \end{bmatrix} \tag{2.12}$$

$\mathcal{R}_z(-\Phi)$ is the rotation matrix around z, Φ is the angle between stationary frame and $'$. $S(r')$ is a skew-symmetric matrix representation of the cross product: $r'\times$.

[3]Note that this is the frame of the specific blade element.

2.4.4 Total forces and torques

The forces are described in the blade element frame, and must therefore also be mapped to a stationary frame. Force in the blade element is applied at the radial station r, this results in a torque in $'$:

$$d\boldsymbol{\mathcal{M}}' = r' \times dF'' \tag{2.13}$$

and the force-torque vector for the blade element:

$$d\boldsymbol{\tau}' = \begin{bmatrix} dF'' \\ r' \times dF'' \end{bmatrix} \tag{2.14}$$

What remains then is rotating the force-torque vector into a stationary frame:

$$d\boldsymbol{\tau} = \begin{bmatrix} \mathcal{R}_z(\Phi) \\ \mathcal{R}_z(\Phi)S(r') \end{bmatrix} F'' \tag{2.15}$$

where $S(r')$ is the skew symmetric representation of the cross product, that is $S(r')dF'' = r' \times dF''$.

The forces and torques of the blade elements can now be integrated over the rotor radius R and over one full rotation to obtain the total forces and torques in the stationary frame, viz.

Definition 2.7 (Airloads)**.**

$$\boldsymbol{\tau} = \int_0^{2\pi} \int_0^R d\tau dr d\Phi \tag{2.16}$$

2.4.5 Parameters

There are a number of parameters involved in the calculation of aerodynamic forces and torques. The NREL 5MW wind turbines blade aerodynamical properties can be found in [1]. This includes tabular information about the chord $c(r)$, twist $\Theta(r)$ and, the lift and drag coefficients of the different airfoil types with respect to the angle of attack α. Since pitch control is not explored in this thesis, the collective pitch deflection is assumed to be zero.

What remains is then the relative velocities $\boldsymbol{\nu}$ consisting of the linear and angular velocities denoted as $\boldsymbol{v}$ and $\boldsymbol{\omega}$, respectively. Since BET is a method rather than an analytical expression, it is convenient to denote it as a function of linear and angular velocities, viz.

Key Result 2 (Airloads function)**.**

$$\boldsymbol{\tau} = a(\boldsymbol{v}, \boldsymbol{\omega}) \tag{2.17}$$

2.4.6 Blade Element Momentum Theory

When calculating inflow in the context of the Blade Element Theory (BET), one approach involves using the Rankine-Froude momentum theory [31]. In this approach, the inflow model solely considers linear momentum.

Conversely, Blade Element Momentum Theory (BEMT) offers a more comprehensive modelling approach, incorporating both linear and angular momentum components within its inflow model [42].

Essentially, one can view BEMT as an extension of BET that incorporates an inflow model which takes into account both the linear and angular momentum in each blade element. BEMT essentially treats each blade element as an independent entity, calculating the 2-DOF inflow at each element.

This results in a model with $2\times$ the number of integration points-DOF. However, the assumption of independence between blade elements is a simplification. In reality relative flow of one blade element can be influenced by another. Various corrections, such as tip-loss correction, are often applied when using BEMT to improve its accuracy and make it practically feasible [43].

It can therefore be argued that using an averaged inflow model in conjunction with BET is not necessarily a downgrade from BEMT. This is because the averaged inflow model, as presented in Section 2.3, incorporates both linear and angular momentum in an averaged sense across the entire rotor, providing a 6-DOF representation of the inflow.

2.5 Wind turbine control systems

A wind turbine is a sophisticated system characterized by numerous moving parts and highly nonlinear dynamics. Balancing power extraction and the lifespan of the system necessitates the implementation of several control systems. In this section, the control systems needed in a full-scale wind turbine are presented.

2.5.1 Definition of required control systems.

[9] stated that as a minimum, the normal operating range of a wind turbine should be characterized by the following set of properties and requirements:

- a maximum 10-minute mean wind speed at hub height V_{max} i.e. the stop wind speed below which the wind turbine may be in operation
- a maximum long-term mean nominal power P_{nom}, interpreted as the highest power on the power curve of the wind turbine in the wind speed interval $[V_{min}; V_{max}]$, where V_{min} denotes the start wind speed for the turbine
- a maximum nominal power Pmax, which on average over 10 minutes may not be exceeded for a wind speed at a hub height of $V_{10min,hub} < V_{max}$
- a maximum operating frequency of rotation $n_{r,max}$ for the wind turbine
- a maximum transient frequency of rotation n_{max} for the wind turbine
- a wind speed below which the wind turbine may be stopped

To keep the wind turbine within this operating range the following control systems are incorporated:

Yaw-Control aligns the wind turbine with the wind.

Pitch-Control manages power capture. In high wind speeds, the pitch can be adjusted to reduce power capture instead of shutting down. This is achieved through either constant torque or constant power strategies, with the former offering better stability [34].

Brake control system slows down or stops the rotor for maintenance or when the turbine is outside of its operating range.

Power Control System manages the electrical output of the turbine to ensure compatibility with the power grid. This includes controlling voltage, frequency, and phase.

Additionally, the following systems are incorporated:

Torque control system manages the torque applied to the generator to control the rotor's angular velocity.

Protection Control system monitors the turbine's status, detects faults, and ensures that the system is shut down and staff is alerted when problems occur.

The control systems directly relevant to the dynamics of the wind turbine are the Yaw and Torque control systems, which are further elaborated upon in 2.5.2 and 2.5.3, respectively.

2.5.2 Role and impact of yaw control

Yaw control refers to the system that adjusts the orientation of the wind turbine's nacelle. The main purpose of yaw control is to ensure that the turbine is facing into the wind to capture the maximum amount of energy. However, both [10] and [11] introduce a deliberate yaw error to alter the wake trajectory to generate potentially more favourable inflow conditions for downstream

turbines increasing the power extraction. The yaw error χ, is the angle difference between the wind direction ψ and the wind turbine yaw θ, which is the direction the nacelle is facing, as defined in Figure 3.2. This relationship is expressed as:

Definition 2.8 (Yaw error)**.**

$$\chi \triangleq \psi - \theta$$

Yaw control systems can be categorized into two types: active and passive. Active systems use sensors and motors to align the turbine with the wind, while passive systems use the force of the wind itself to align the turbine. The wind turbine used for simulations presented in Section 2.6 is an *upwind turbine* which exclusively uses active yaw control.

The main challenges associated with yaw control include accurately determining the wind direction ψ and balancing the mechanical stress imposed on the turbine from frequent reorientation against the need for optimal power capture.

2.5.3 Maximum power point tracking - Torque control

The process of converting the aerodynamical power into electrical power is not as simple as letting the wind spin the blades. Generator shaft torque M_E is the torque exerted on the wind turbine shaft by the generator. This torque plays a crucial role in power extraction as it influences the rotor's angular velocity. If the torque is too high it can cause the wind turbine to slow down or stop. Conversely, too low torque means that the wind source is under-utilized. These scenarios lead to suboptimal power extraction, therefore a control law governing the relation between rotational speed and torque is introduced.

Maximum power point tracking (MPPT) facilitates "optimal" power extraction by adjusting the rotor's angular velocity such that it maintains optimal tip-speed ratio(TSR) $\lambda^* \triangleq \frac{\Omega R}{U}$. The power coefficient has a unique optimum $C_P^*(\lambda^*)$ at the optimal tip-speed ratio, hence by following the optimal tip-speed ratio we get optimal power extraction.

Under steady state assumption we can express the generator shaft power as:

$$E = \frac{1}{2}\rho\pi R^2 C_P^* U^3 \tag{2.18}$$

inserting the rearranged tip-speed ratio $U = \frac{\Omega R}{\lambda^*}$ into (2.18) we arrive at the following:

Definition 2.9 (MPPT control law)**.**

$$E(\Omega) = b|\Omega|\Omega^2, \quad b \triangleq \tfrac{1}{2}\rho\pi R^2 C_P^* \left(\tfrac{R}{\lambda^*}\right)^3$$

where ρ is the air density, R is the rotor radius, C_P^* and λ^* is the optimal power coefficient and tip-speed ratio respectively. This control law exhibits local asymptotic stability around the optimal TSR [44]. The numerical values of these are given in Table 2.2.

Incorporating MPPT into Rotor Dynamics:

The rotational motion of the wind turbine is now dependent on both aerodynamic torque and generator shaft torque. The generator shaft torque M_E can be found from Definition 2.9 using the relationship between torque and power and is defined as:

Definition 2.10.

$$M_E \triangleq b|\Omega|\Omega \tag{2.19}$$

with this (2.1) now becomes:

Key Result 3.

$$J\dot{\boldsymbol{\Omega}} = -M_z - M_E \tag{2.20}$$

This equation covers the relevant rotor dynamics taking into account both aerodynamic and generator shaft torque by describing how the rotor's angular velocity changes with the net torque.

2.6 NREL offshore 5-MW baseline wind turbine

> "To support concept studies aimed at assessing offshore wind technology, we developed the specifications of a representative utility-scale multimegawatt turbine now known as the "NREL offshore 5-MW baseline wind turbine."" [1]

In 2009 The National Renewable Energy Laboratory (NREL) published the paper [1] describing a virtual wind turbine known as the *NREL offshore 5-MW baseline wind turbine*. While this wind turbine does not exist physically, it is widely used as a standard for performance evaluation of wind technology such as different control strategies. In this, the design and technical specifications of the aforementioned wind turbine is presented along with examples of it being used as a benchmark for research and development.

2.6.1 Design and specifications

The NREL offshore 5-MW baseline wind turbine is an upwind, three-bladed turbine with a rotor diameter of 126 meters and a hub height of 90 meters. It utilizes variable speed, collective pitch control(VSCP), hence it is not relying on passive stall as in fixed speed fixed pitch(FSFP) to avoid that the 5MW maximum power rating is not exceeded. Collective pitch, often referred to as variable pitch, allows for a less conservatively dimensioned[4] wind turbine as the system does not solely rely on passive stall regulation. Variable speed allows for control of the angular velocity, the benefit of this was explored in Subsection 2.5.3.

[4]Conservatively dimensioned e.g lower Cut-Out Wind speed.

Rating	5 MW
Rotor Orientation, Configuration	Upwind, 3 Blades
Control	Variable Speed, Collective Pitch
Drivetrain	High Speed, Multiple-Stage Gearbox
Rotor, Hub Diameter	126 m, 3 m
Hub Height	90 m
Cut-In, Rated, Cut-Out Wind Speed	3 m/s, 11.4 m/s, 25 m/s
Cut-In, Rated Rotor Speed	6.9 rpm, 12.1 rpm
Rated Tip Speed	80 m/s
Overhang, Shaft Tilt, Precone	5 m, 5^{o}, 2.5^{o}
Rotor Mass	110,000 kg
Nacelle Mass	240,000 kg
Tower Mass	347,460 kg
Coordinate Location of Overall CM	(-0.2 m, 0.0 m, 64.0 m)
Generator efficiency	94.4%
Maximum Yaw rate	0.3 deg/s

Table 2.1: Specifications of the NREL 5-MW Baseline Wind Turbine [1].

Variables such as the tip-speed ratio λ, peak power coefficient C_p and rotor moment of inertia J along with air density ρ[5] and rotor radius R are utilized in the process model Chapter 4.

Notation	**Value**
J	11776047
R	63
ρ	1.225
C_p^*	0.482
λ^*	7.55

Table 2.2: Numerical values used for simulation [1].

The remaining variables such as chord $c(r)$, twist $\Theta(r)$ and the lift and drag coefficients utilized in Subsection 2.4.5 can be found in [1] in comprehensive data sets.

Applications

As the quote in the start of this section suggests, this wind turbine is a representative utility-scale 5MW wind turbine suitable for assessing offshore wind technology. The over 5800 citations of the paper [1] show that they have succeeded. For instance [10] used two NREL 5MW wind turbines to explore optimal wind layout with and without active yaw control. The yaw control introduced intentional yaw misalignment for optimal wake redirection. [45] used NREL 5MW wind turbines in a wind farm to compute the power production and rotor loads for a given yaw, enabling the conclusion that intentional yaw misalignment's benefits are highly dependent on the topology and wind directions. [8] performed a stress and vibration analysis under steady state of the NREL 5MW blade using different materials for the blade. It was then found that Carbon fiber reinforced plastic performed better than E-glass fiber and Kevlar in terms of stress on the blades under similar load conditions.

In conclusion, the NREL offshore 5MW baseline wind turbine is a widely used tool in the wind energy research field. The comprehensive specifications encapsulate the key characteristics of a representative utility-scale wind turbine, making it an ideal benchmark. A benefit of using these widely adopted specifications is that comparison of data across different studies becomes easier. Given these factors, the NREL offshore 5MW baseline wind turbine becomes a clear choice as the set of specifications to base our turbine model on in this thesis.

[5] ρ is not unique for NREL, however it is included in the table for easier reference.

Chapter 3

Wind modelling for simulation

Accurate modelling of wind behavior is crucial for generating realistic and comparable results. There exists several different methods for generating a wind signal [46]. This chapter introduces two reference frames: the Geographic and Body-fixed reference frames. These frames are defined to accurately establish relative quantities. Subsequently, two approaches for wind modelling are presented. The first approach employs a traditional method for generating a wind signal for wind turbine simulations, using *Volume forces* to impose an artificially generated turbulence inflow. The second approach introduces a new model based on data from [20], treating the wind's yaw ψ and velocity U_∞ as a *Gauss-Markov Process*. This method involves the approximating the autocorrelation function using an exponential form. The resulting approximation is then used to identify a shaping filter, which is crucial for the generation of the model.

3.1 Reference frame

To accurately describe the dynamical properties of a wind turbine, two reference frames must be established. This section is dedicated to defining the reference frames for the wind turbine under consideration. The specific configuration of the BODY reference frame is illustrated in the following figure.

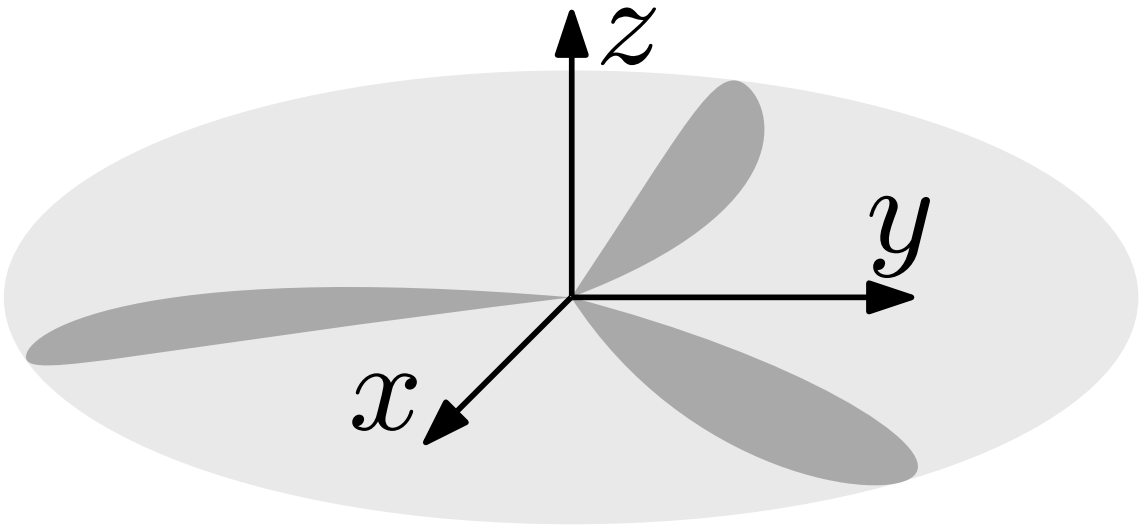


Figure 3.1: BODY fixed reference frame of the wind turbine. This figure is a slight modification of the original presented in [47].

As can be seen, the z-axis is defined along the nacelle, the x-axis is defined horizontally with respect to the ground and the y-axis is defined along the wind turbine tower. This coordinate system is situated within a reference frame. The *North-East-Down*(NED) Geographic reference frame seems appropriate as the wind turbine is stationary, as done in [48, 49]. BODY reference frame, as defined here, is by default aligned with NED such that the z-axis coincides with the North axis and the x-axis coincides with the East axis whilst the y-axis coincides with the Down axis. The BODY reference frame inserted into the NED reference frame is depicted below.

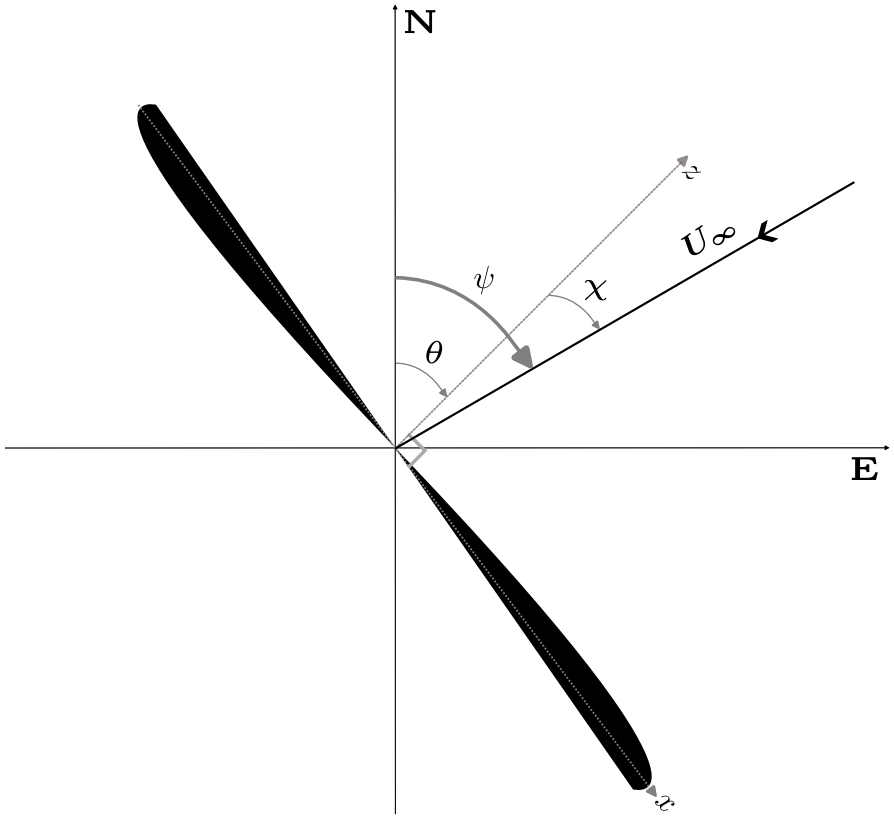


Figure 3.2: BODY inserted into NED with defined angles.

The figure illustrates the orientation of the North and East axes in relation to the wind turbine's coordinate system and the incoming free-stream flow, which is denoted as U_∞. As defined in Definition 2.8, θ represents the yaw angle of the wind turbine, ψ denotes the wind direction and χ is the yaw error.

3.1.1 Free-stream flow in BODY frame

As explained in Section 2.4, the forces and torques on the rotor can be calculated using the relative velocities in the BODY frame. Consequently, this necessitates decomposing the free-stream flow into the BODY reference frame. To facilitate this, the free-stream flow is initially decomposed into NED coordinates.

The free-stream flow can be decomposed into NED coordinates as follows:

$$\begin{bmatrix} U_E \\ U_N \end{bmatrix} = \begin{bmatrix} \sin(\psi) \\ \cos(\psi) \end{bmatrix} U_\infty \tag{3.1}$$

Subsequently, the wind is rotated into the BODY frame using the standard rotation matrix:

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{3.2}$$

Finally, the free-stream flow, now defined in the BODY frame, can be obtained as follows:

$$\begin{bmatrix} U_s \\ U_n \end{bmatrix} = \mathbf{R}(\theta) \begin{bmatrix} U_E \\ U_N \end{bmatrix} \tag{3.3}$$

where U_s represents the side wind and U_n denotes the wind normal to the wind turbine, i.e., the free-stream flow as experienced by the rotor.

3.1.2 White noise generation

White noise is defined as a stationary random process, characterized by its constant Power Spectral Density (PSD) function. Theoretically, this requires infinite variance for the PSD to remain constant across all frequencies [50]. However, generating a signal with infinite variance is not feasible in practice.

Therefore, to generate a white noise signal, it has to be limited to a constant PSD over a range of frequencies, generating a signal that has constant PSD in these frequencies. Since all generated white noise signals are bandlimited in some sense it is often referred to as white noise.

In applications like Kalman filtering and wind signal generation, unity white noise is often employed. Unity white noise is a zero-mean white noise signal with a Gaussian distribution and a variance of one [51]. It can be characterized by the following PSD:

$$\mathcal{S}_w(j\omega) = 1 \tag{3.4}$$

In this equation, $\mathcal{S}_w(j\omega)$ denotes the PSD of the unity white noise, which remains constant at 1 for all frequencies within the selected band. In `Julia` this can be done using the `randn` function.

3.2 Cartesian wind model using Von karman

A common approach for modelling a generic wind signal assumes that the wind signal can be described as:

$$U = \bar{U} + \xi \tag{3.5}$$

where $\bar{U}$ is the slowly varying mean component of the wind signal and ξ is the turbulent component of the wind signal. This approach, known as the *Volume forces method* [46], is used for imposing inflow turbulence to obtain a turbulent wind signal.

3.2.1 Artificial turbulence generation

The Von Karman spectrum is a commonly used turbulence Power Spectral Density(PSD) function [3]. This spectrum describes the frequency content of the wind-speed variations, which can be used to generate an artificial turbulent flow. An overview of artificial turbulence generation can be found in [52]. Through spectral factorization, the Von Karman spectrum can be expressed as:

$$\mathcal{S}_{\mathbf{K}} = H(j\omega)H(-j\omega)\mathcal{S}_w(j\omega), \quad H(j\omega) = \frac{\sigma_{\mathbf{K}}\sqrt{\frac{19L}{40\bar{U}_\infty}}}{\left(1 + \frac{j\omega L}{\bar{U}_\infty}\right)^{\frac{5}{6}}} \tag{3.6}$$

In this equation, $\sigma_{\mathbf{K}}$ is the turbulence intensity, L is the length scale and $\bar{U}_\infty$ is the mean free-stream flow. The shaping filter, $H(j\omega)$, is used to shape white noise into colored noise that represents the turbulent flow.

Given that $H(j\omega)$ is a fractional filter, it can be beneficial to approximate it to a first-order shaping filter using a *Padè approximant*. This approximation can enhance performance and simplify implementation.

Definition 3.1. *Let $H(s)$ be the fractional transfer function in (3.6) of order 5/6. The $[0,1]$[a] order Pade approximant of $H(s)$ is approximately given by:*

$$H_P(s) = \frac{\mathcal{K}}{1+Ts} \tag{3.7}$$

where $\mathcal{K}$ is $\sigma_{\boldsymbol{K}}\sqrt{\frac{19L}{40\bar{U}_\infty}}$ and T is $\frac{5L}{6\bar{U}_\infty}$

[a]This means that the approximant will have a constant in the numerator and a first order term in the denominator.

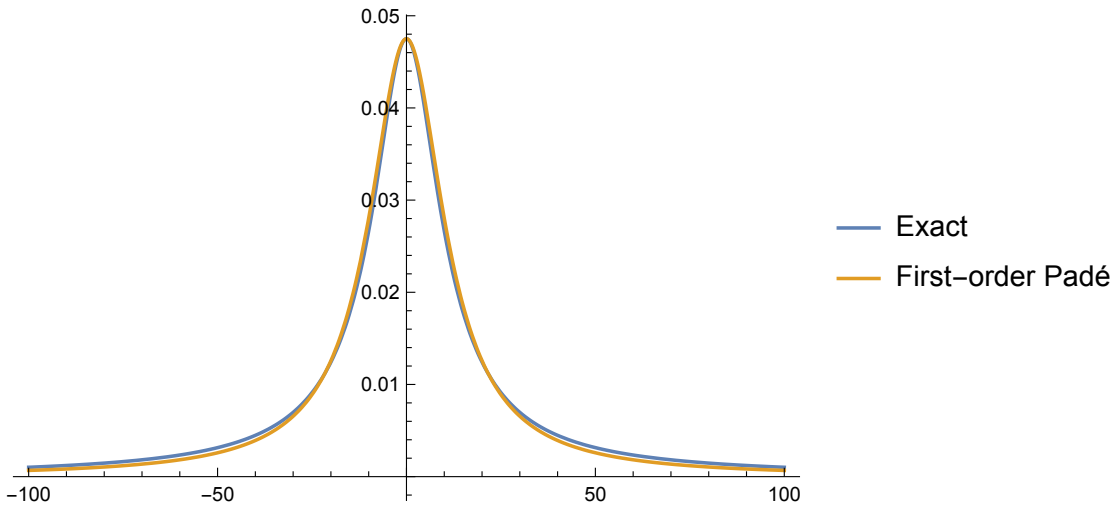


Figure 3.3: The $[0,1]$ order Padé approximant is an adequate approximation of the fractional filter.

$H_{\mathrm{P}}(s)$ describes the dynamic behaviour of the turbulent component ξ when subjected to a bandlimited white noise input w. The turbulent component's dynamics can then simply be expressed as:

Definition 3.2 (Artificial turbulence)**.**

$$T\dot{\xi} = -\xi + \mathcal{K}w$$

This describes the turbulent component in one direction. To create a two-dimensional wind model, we simply use two different white noise signals to shape the turbulent components in two directions, thus using a "Cartesian" approach as shown:

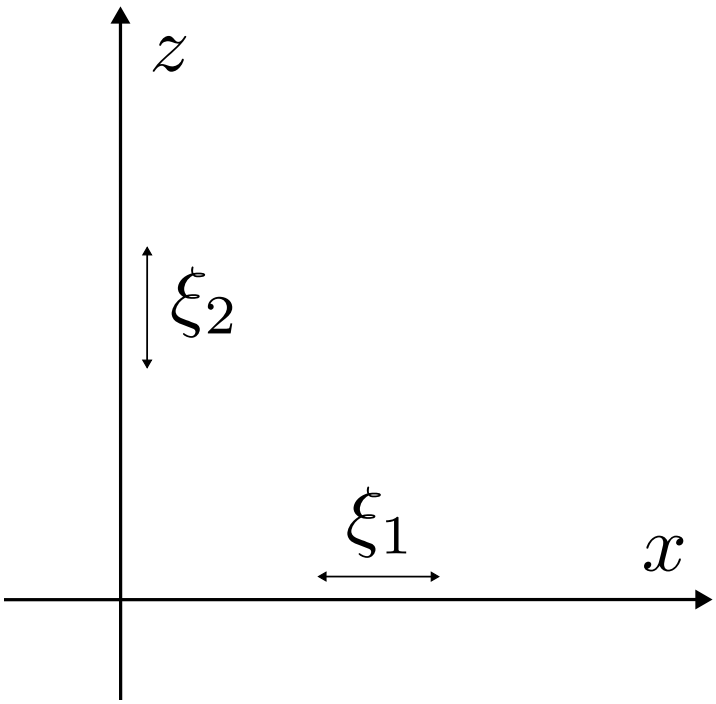


Figure 3.4: Turbulent flow ξ_1 and ξ_2 as experienced by the wind turbine. Note that the turbulent flow in the generated wind signal is imposed on the North and East axis instead of z and x axis, respectively.

This leads to to the Cartesian wind model:

Key Result 4 (Cartesian wind)**.**

$$\begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix} \begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} = - \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} + \begin{bmatrix} \mathcal{K}_1 & 0 \\ 0 & \mathcal{K}_2 \end{bmatrix} w \tag{3.8}$$

3.2.2 Length scaling and turbulence intensity

In order to utilize the above results, it is necessary to appropriately set the turbulence intensity, denoted as $\sigma_{\mathbf{K}}$, and the length scaling, represented by L. According to both [3] and [53], *the Danish standard* (DS472,1992) and *the IEC standard*(1999) are commonly referred to guidelines for determining these parameters.

The relationship between the Length scaling of the Kaimal spectrum and the Length scaling of the Von Karman spectrum is described by the following equation [3]:

$$L_{Kaimal} = 2.329L \tag{3.9}$$

This equation is particularly relevant when considering the Danish standard, which specifies the length scaling of the Kaimal spectrum for tower height above 30m. According to this standard:

Assumption 1 (Length scaling)**.**

$$L_{Kaimal} = 150 \implies L = 64.4$$

This assumption implies that for a tower height above 30m, the length scale of the Von Karman spectrum is approximately 64.4, as derived from the relationship between the Kaimal and Von Karman length scales.

The turbulence intensities for different mean wind speeds according to various standards are depicted in the following figure.

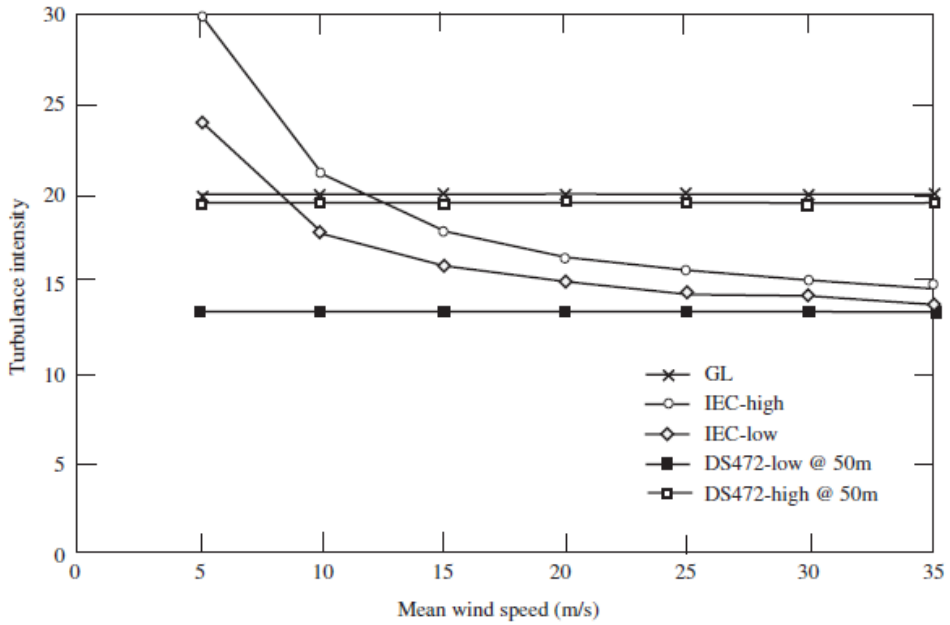


Figure 3.5: Turbulence intensities [3]

Using DS472-high it can be seen that:

Assumption 2 (Turbulence intensity)**.**

$$\sigma_{\boldsymbol{K}} \approx 19$$

This indicates that at high wind speeds, the turbulence intensity is relatively high, which can have significant implications for the operation and performance of the wind turbine.

With the length scaling and turbulence intensity set, $\mathcal{K}$ and T can be found for any nonzero free-stream flow.

3.2.3 Wind generated using the Cartesian model.

When employing the Cartesian wind model, it's important to generate a wind signal set in the NED frame. This is crucial because if the wind signal was set in the BODY frame, it would follow the wind turbine as it yaws. The turbulent components, ξ_1 and ξ_2, are therefore added to the East and North directions respectively.

In a later section it can be seen that the Kalman filter is linearized around steady-state conditions with $\chi = 0$ and $U_\infty = 4$m/s. To examine the performance of the Kalman filter around this linearization the mean component is chosen as:

$$\bar{\boldsymbol{U}} = \begin{bmatrix} \bar{U}_E \\ \bar{U}_N \end{bmatrix} = \begin{bmatrix} 0 \\ 4 \end{bmatrix}, \quad |\bar{\boldsymbol{U}}| = \bar{U}_\infty = 4 \tag{3.10}$$

This implies that the mean component is purely from the North direction, which gives zero yaw error when the yaw angle of the wind turbine is zero, i.e, $\chi = 0$ when $\theta = 0$.

The turbulent components can then be found from:

$$\mathcal{K} = \sigma_{\mathbf{K}} \sqrt{\frac{19L}{40\bar{U}_\infty}} = 52.54$$
$$T = \frac{5L}{6\bar{U}_\infty} = 13.42 \tag{3.11}$$
$$\begin{bmatrix} T & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} = - \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} + \begin{bmatrix} \mathcal{K} & 0 \\ 0 & \mathcal{K} \end{bmatrix} \begin{bmatrix} \mathrm{w}_1 \\ \mathrm{w}_2 \end{bmatrix}$$

The mean component and different white noise signals are the only differences between the North and East wind components. The resulting North wind, East wind, free-stream flow, and yaw are depicted in the figures below.

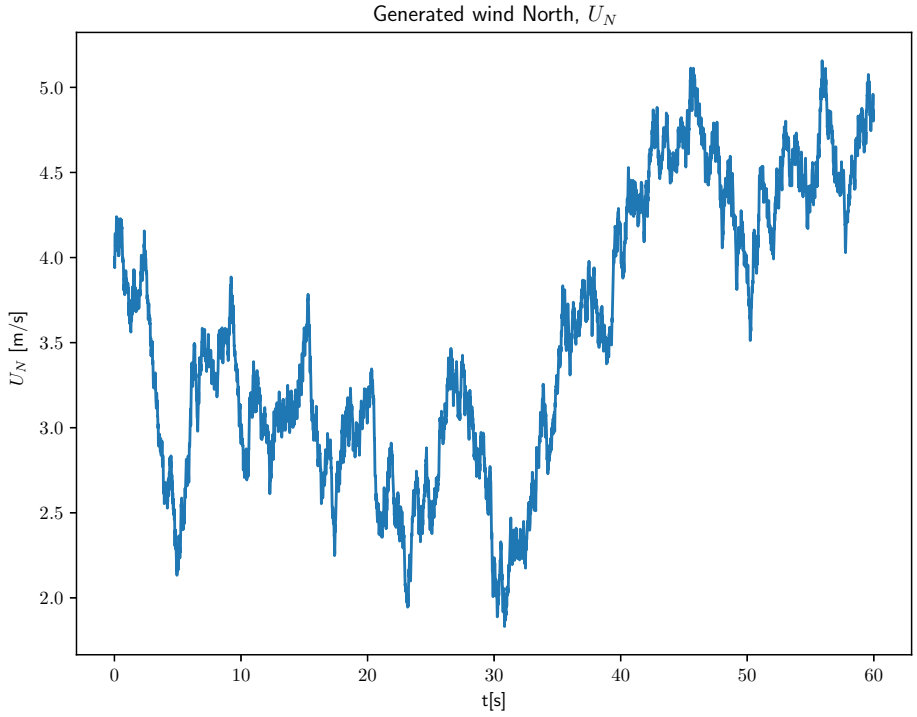


Figure 3.6: Generated wind North $U_N = U_0 + \xi_2$.

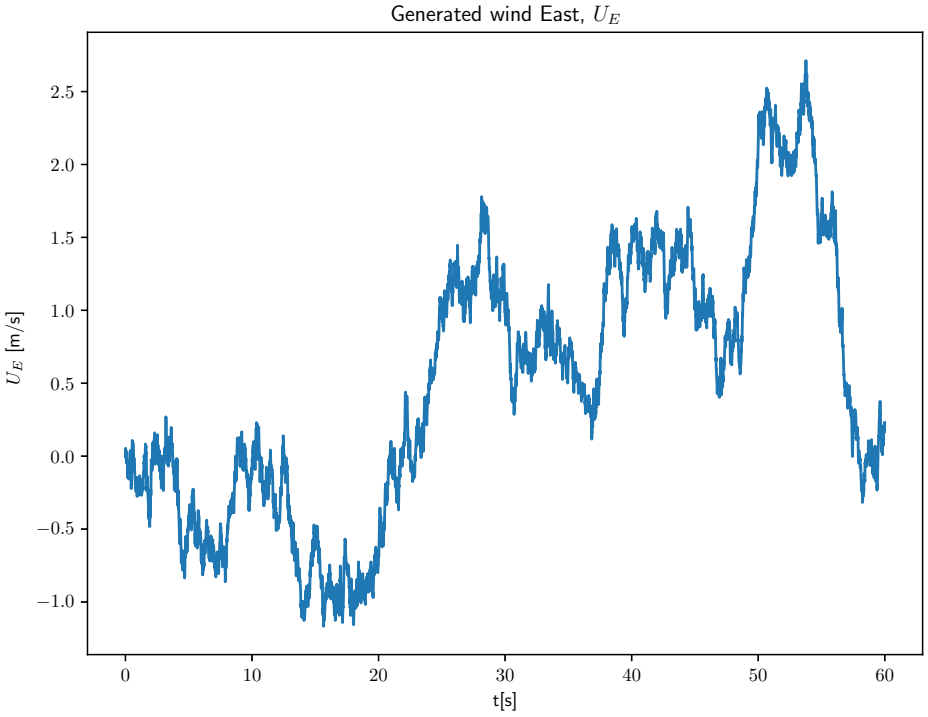


Figure 3.7: Generated wind East $U_E = \xi_1$.

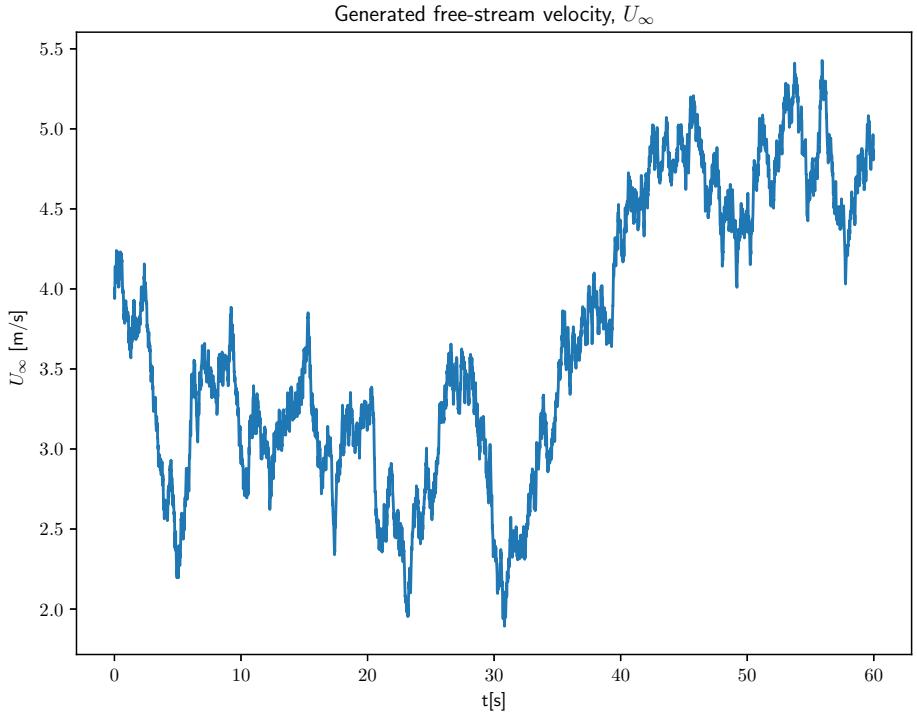


Figure 3.8: Resulting free-stream flow velocity.

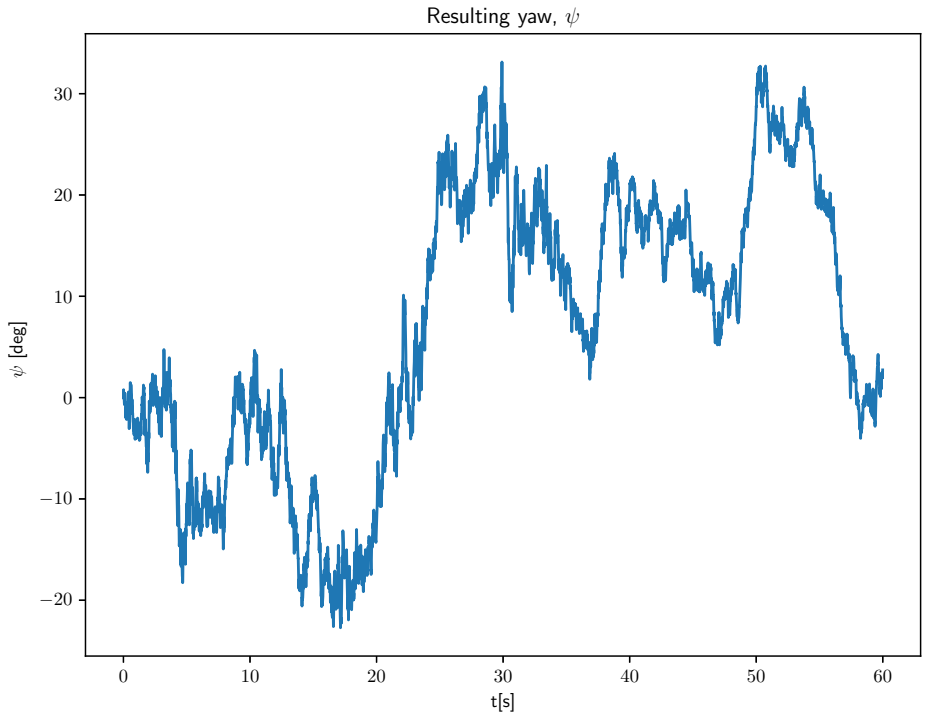


Figure 3.9: Resulting yaw of the wind.

3.3 Yaw-Velocity wind modelling based on data.

The Cartesian model assumes a constant mean wind, with artificial turbulence leading to perturbations in the mean directions, as illustrated in Section 3.2. However, in reality, wind conditions can change significantly over relatively short periods of time. Therefore, it is desirable to have a model that allows for realistic variations in the wind speed and yaw. This would allow for a more accurate representation of the dynamic nature of wind, capturing its fluctuations and variations over time.

To better understand how the wind varies in direction and speed, wind roses are often used. A wind rose is used to visualize how wind speed and direction is distributed over a time period. Figure 3.10 presents a wind rose for the NREL Flatirons Campus (M2) wind turbine at an 80m height, based on data from December 16, 2022 [20]. The wind rose provides a clear visual representation of the wind conditions, strengthening the argument of needing a new wind model.

In the following sections, we will explore a wind modelling approach that takes into account the dynamic nature of wind, as depicted by the wind rose, to provide a more realistic representation of wind conditions for wind turbine simulations.

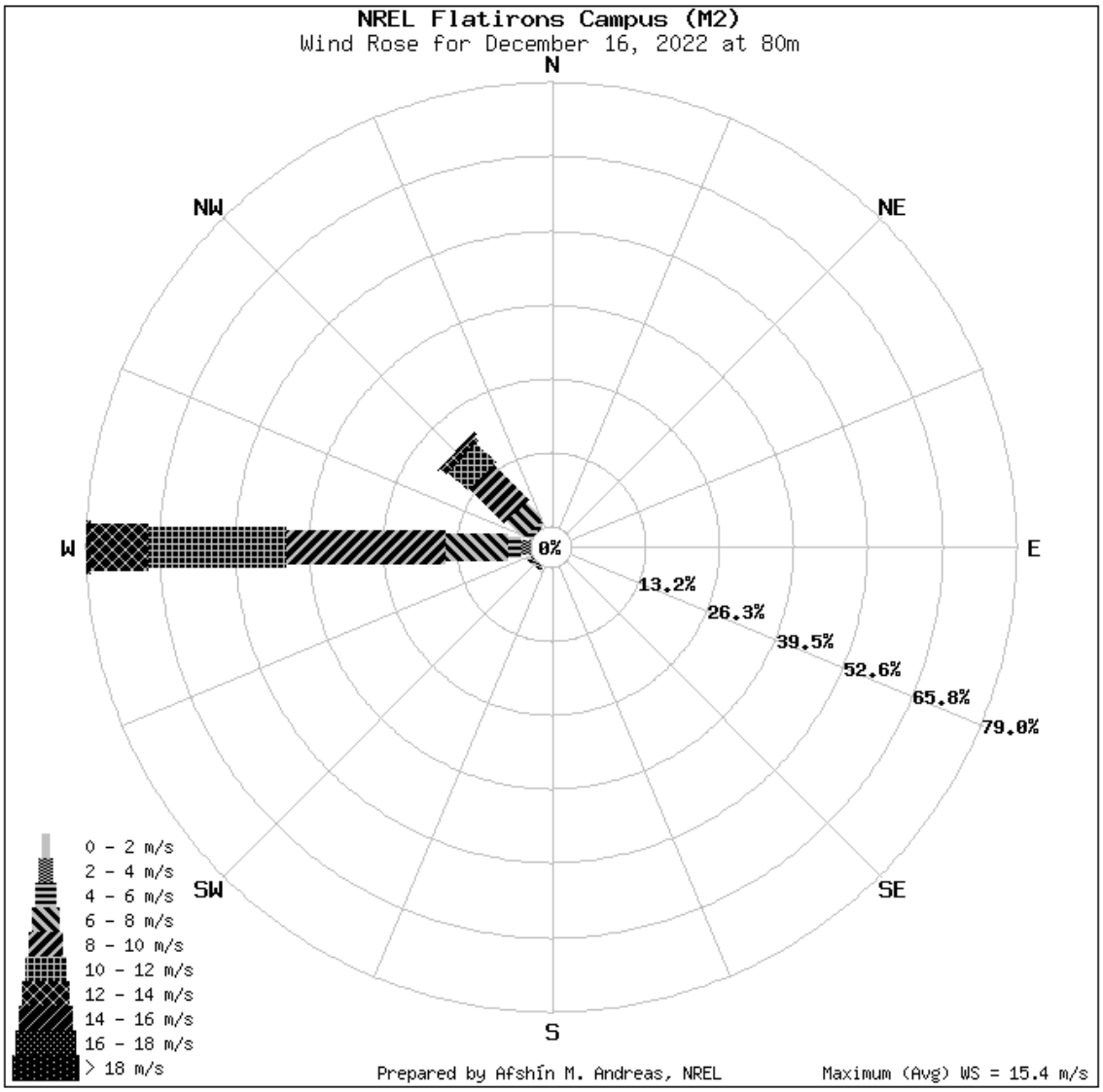


Figure 3.10: Wind rose for December 16,2022 at 80m. NREL Flatirons Campus(M2) wind turbine [20].

3.3.1 Gauss-Markov process

The wind field experienced by a wind turbine can be modelled as a stochastic process $X(t)$. Data from [20] provides wind speed and yaw angle measurements at 60-second intervals. By treating the obtained data as a stationary Gaussian process, which implies *wide-sense stationarity* (WSS), the autocorrelation can be modeled using an exponential function of the form:

$$R(\tau) = E[X(t)X(t+\tau)] = \sigma^2 e^{-\beta|\tau|} \tag{3.12}$$

This assumption of a stationary Gaussian process with an exponential autocorrelation function, as shown in (3.12), is the basis for defining a Gauss-Markov process [50].

Autocorrelation and autocovariance are terms often interchangeably used. Therefore to remove confusion:

Definition 3.3 (Autocorrelation)**.** *The mean from the data [20] is removed such that:*

$$E[X(t)X(t+\tau)] = E\{[X(t)-\mu][X(t+\tau)-\mu]\} \tag{3.13}$$

this is valid if and only if $\mu = 0$

The first step towards finding a wind model involves calculating the data's autocorrelation. Subsequently, the model parameters are optimized to fit the exponential autocorrelation function (3.12).

Upon obtaining an approximate autocorrelation function, the Wiener-Khinchine relation can be used to derive the Power Spectral Density function:

Theorem 3.3.1 (Wiener-Khinchine relation)**.**

$$\mathcal{S}(j\omega) = \mathcal{F}[R(\tau)] = \int_{-\infty}^{\infty} R(\tau)e^{-j2\pi\omega\tau}d\tau \tag{3.14}$$

This theorem is applicable to stationary processes, such as the stationary Gaussian process considered here. Essentially, it states that the PSD function can be obtained by performing the Fourier transform $\mathcal{F}$ of the autocorrelation function. Applied to the autocorrelation in (3.12) the following power spectral density function can be found [50]:

$$\mathcal{S}_x(j\omega) = \mathcal{F}[R(\tau)] = \frac{2\sigma^2\beta}{\omega^2+\beta^2} \tag{3.15}$$

(3.15) is then subjected to spectral factorization to derive a shaping filter.

$$\mathcal{S}_x(s) = \frac{2\sigma^2\beta}{-s^2+\beta^2}\mathcal{S}_w(s) = \frac{K}{Ts+1}\frac{K}{-Ts+1}\cdot 1 \tag{3.16}$$

where $S_{wn} = 1$ is the white noise spectral amplitude for a zero mean bandlimited white noise signal with a standard deviation of 1.

Solving (3.16) for σ and β, we get:

$$\beta = \frac{1}{T}, \quad \sigma^2 = \frac{K^2}{2T} \tag{3.17}$$

Inserting (3.17) into (3.12) we get the following autocorrelation function:

$$R(\tau) = \frac{K^2}{2T} e^{-\frac{|\tau|}{T}} \tag{3.18}$$

The parameters K and T are found numerically by minimizing the squared error between the autocorrelation of the measured data, $\mathcal{A}_i$, and the autocorrelation function in (3.18), viz.

$$\min_{K,T} \sum_{i=0}^{n} \left(\mathcal{A}_i - \frac{K^2}{2T} e^{-\frac{|\tau_i|}{T}} \right)^2 \tag{3.19}$$

In this equation, $i \in [0, 1 \cdots, 50]$ is the specified lags where the autocorrelation is fitted. This implies that the autocorrelation is modelled up to 50 minutes of the data.

The optimization is performed in `Julia` [19] using the `JuMP` package [54] with the `Ipopt` solver [55]. Values of K and T that minimize the sum of the squared differences are then found.

This model identification approach is used for both yaw and speed modelling, resulting in the shaping filters:

Key Result 5 (Yaw-Velocity wind model)**.**

$$\frac{\psi}{w} = \frac{K_\psi}{T_\psi s + 1} \implies T_\psi \dot{\psi} = -\psi + K_\psi w \tag{3.20}$$

$$\frac{U'}{w} = \frac{K_U}{T_U s + 1} \implies T_U \dot{U}' = -U' + K_U w \tag{3.21}$$

$$U = U' + \bar{U} \tag{3.22}$$

where K and T are the filter gain and time constant, respectively. $\bar{U}$ is the mean of measured data.

The inclusion of the mean of ψ is not critical for the functioning of the model. This is because the fluctuations of ψ around e.g. North or East are largely a matter of definition and do not impact the model's performance or results.

3.4 Validation of the new wind model

While utilizing the Von Karman spectra is well-established in the field, the new wind model proposed requires validation. For this purpose, we utilized wind yaw and velocity measurements [20].

Data from December 16, 2022, at an altitude of 80m was selected to illustrate the significant fluctuations in wind speed and direction that can occur over short time periods. This data provides a robust test for the new model, challenging its ability to accurately represent these rapid changes in wind conditions.

The measured wind yaw and speed from the data are presented in Figures 3.11 and 3.12, respectively. It can be seen that the wind speed fluctuates quite heavily, ranging from below 1m/s to 15m/s. Similarly, the wind yaw also exhibits large fluctuations when compared to the Cartesian wind simulation.

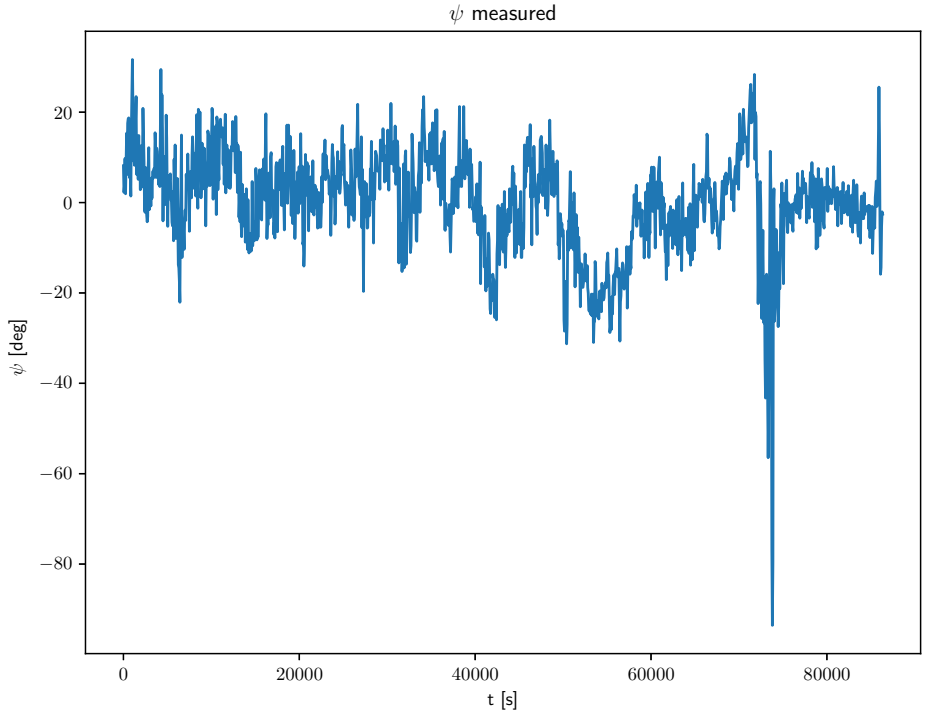


Figure 3.11: 24-hour measured Wind yaw data.

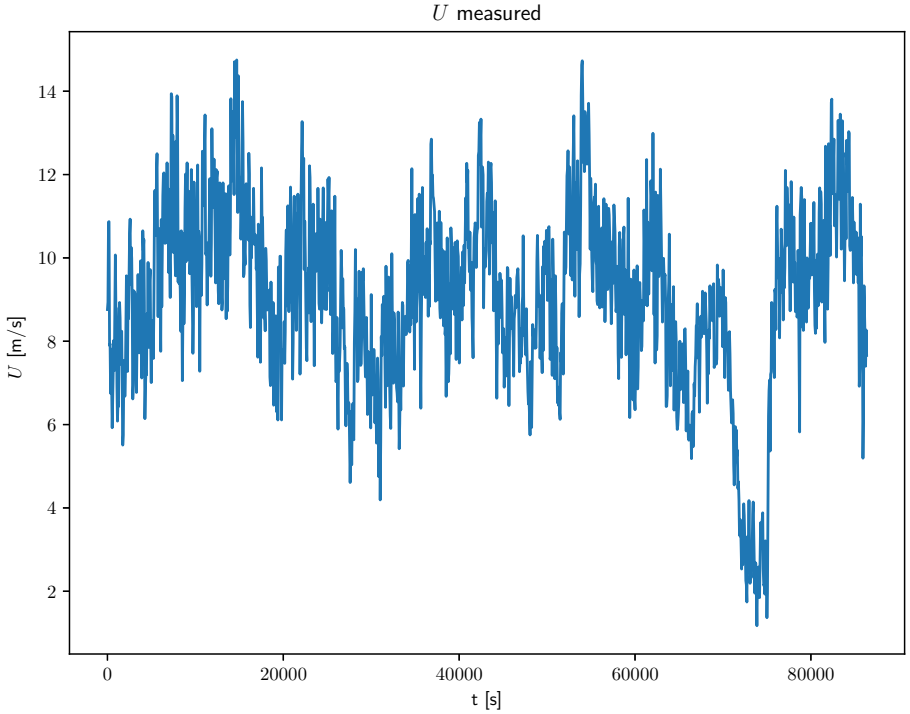


Figure 3.12: 24-hour measured Wind speed data. The mean wind speed is 9.1 m/s.

3.4.1 Model results

Based on these datasets K and T were found by solving the optimization problem (3.19). The resulting gain and time constants are shown in Table 3.1.

	K	T
ψ	9.0	1209
U	128	1901

Table 3.1: Model parameters

These parameters yield the following dynamics for yaw and speed:

$$\begin{aligned} 1209\dot{\psi} &= -\psi + 9.0\mathrm{w}_1 \\ 1901\dot{U}' &= -U' + 128\mathrm{w}_2 \\ U &= U' + 9.1 \end{aligned} \tag{3.23}$$

Here w_1 and w_2 represent two zero-mean bandlimited white noise signals with standard deviation of 1.

The wind yaw and speed generated by the new model are depicted in Figures 3.13 and 3.14, respectively.

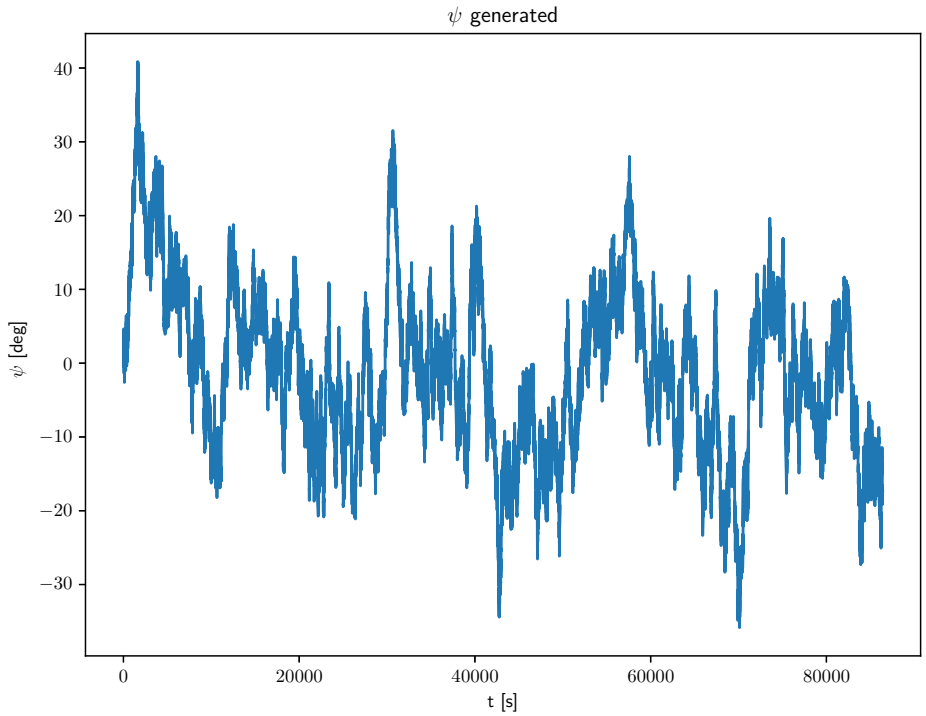


Figure 3.13: 24-hour Wind yaw simulation model.

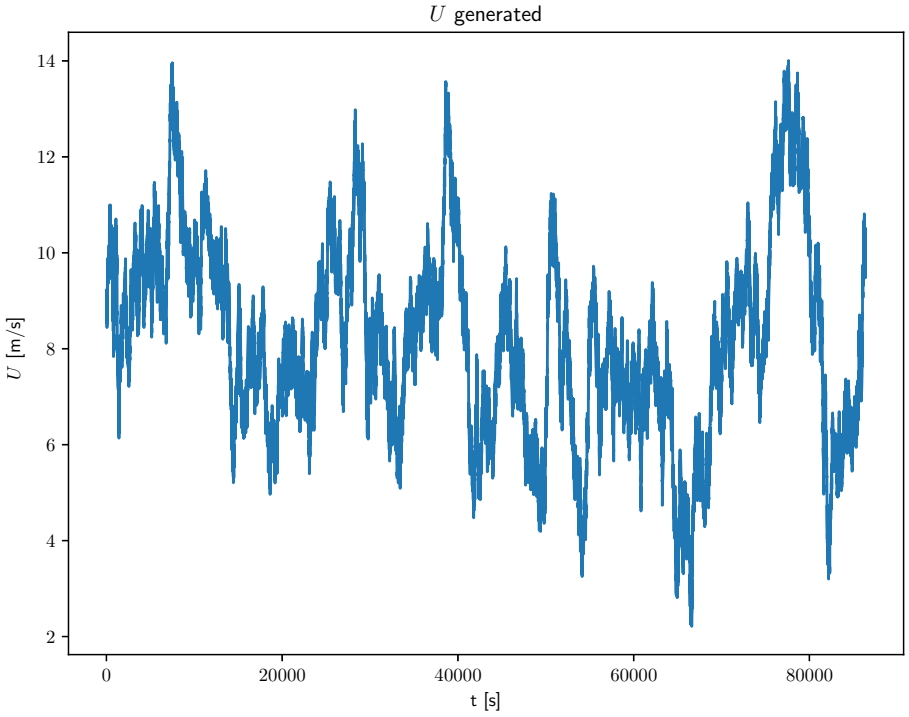


Figure 3.14: 24-hour Wind speed simulation model.

Model fit

The autocorrelation functions, derived by substituting the values from Table 3.1 into equation (3.18), are as follows:

$$\begin{aligned} R_{\psi}(\tau) &= \frac{9^2}{2418} e^{-\frac{|\tau|}{1209}} \\ R_{U}(\tau) &= \frac{128^2}{3802} e^{-\frac{|\tau|}{1901}} \end{aligned} \tag{3.24}$$

The fit of these autocorrelation functions to the autocorrelation of the data is shown in Figures 3.15 and 3.16.

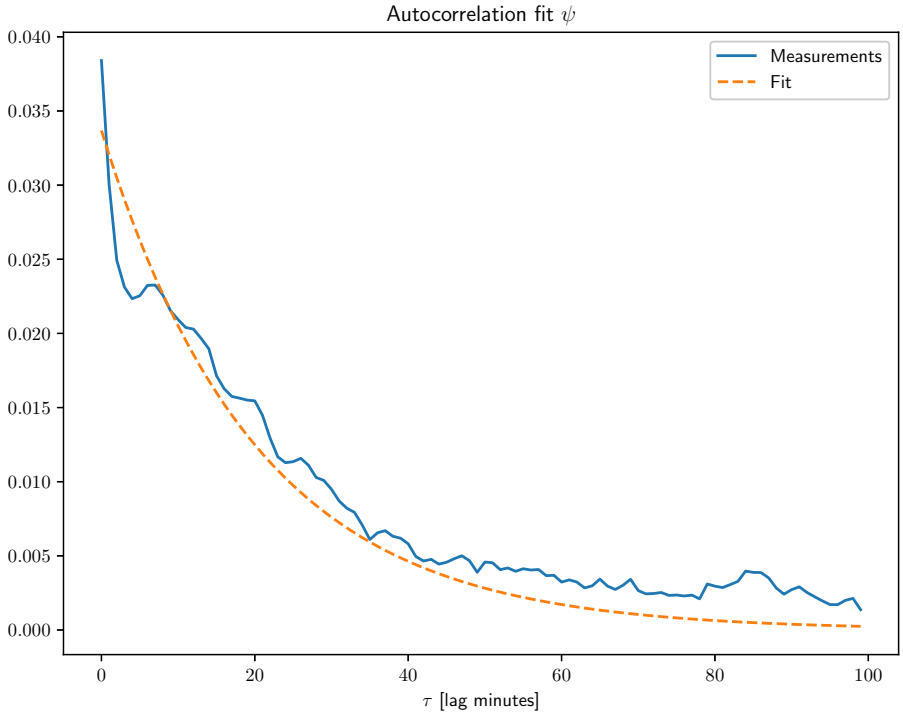


Figure 3.15: Autocorrelation fit of ψ.

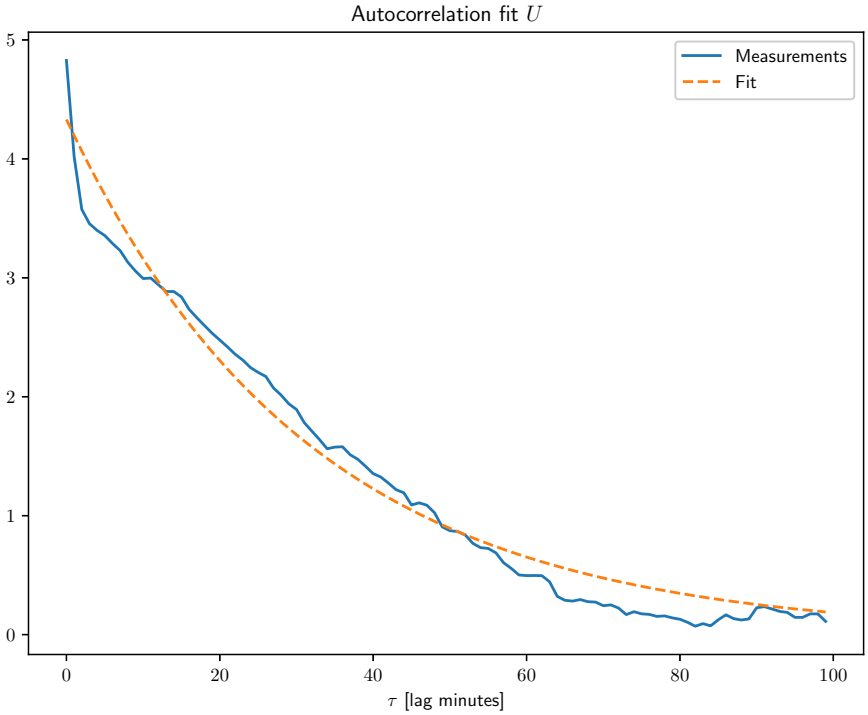


Figure 3.16: Autocorrelation fit of U.

Simulation fit

The autocorrelation of the simulated wind yaw and speed are compared to the measurements in Figures 3.17 and 3.18.

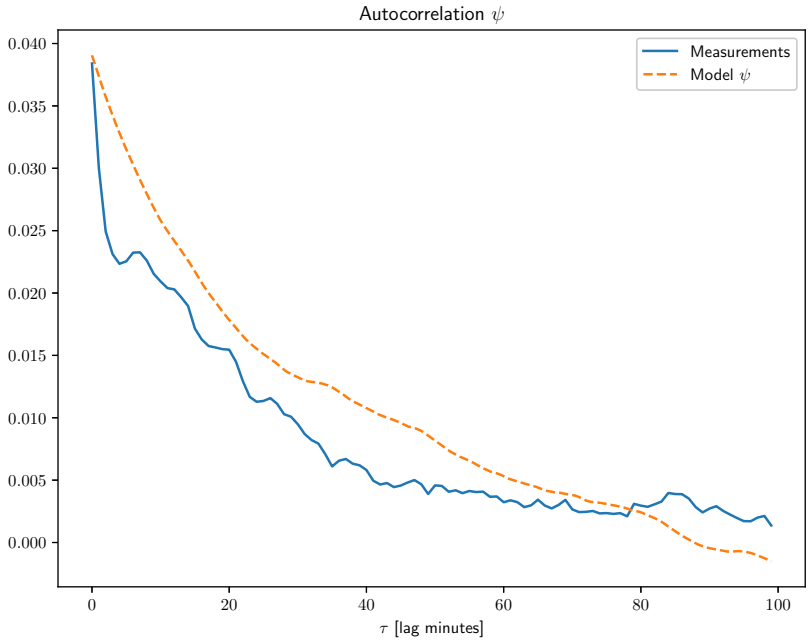


Figure 3.17: Autocorrelation of the yaw model vs measurements.

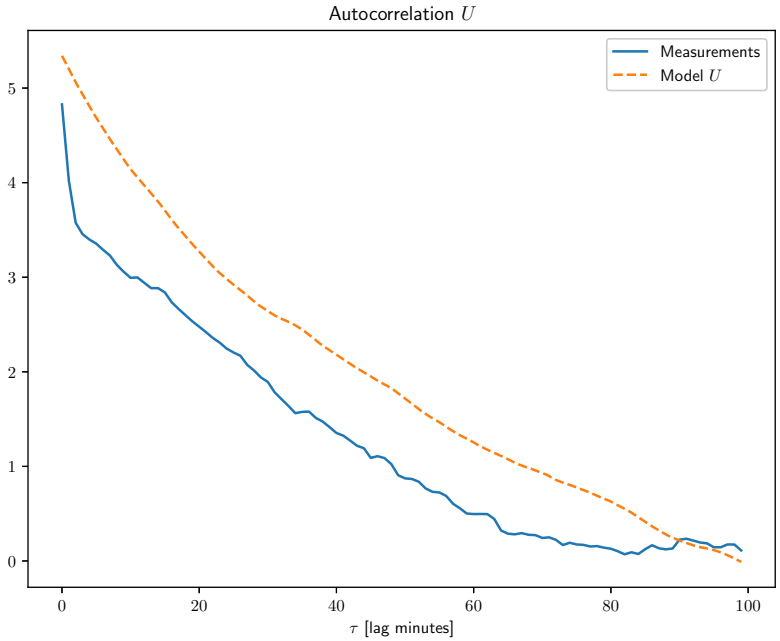


Figure 3.18: Autocorrelation of the speed model vs measurements.

From these figures it can be seen that the autocorrelations of the simulated yaw and speed fit the measured autocorrelations adequately in both cases.

Power spectral density fit

The power spectral density of the simulated wind yaw and speed are compared to the measurements in Figures 3.19 and 3.20.

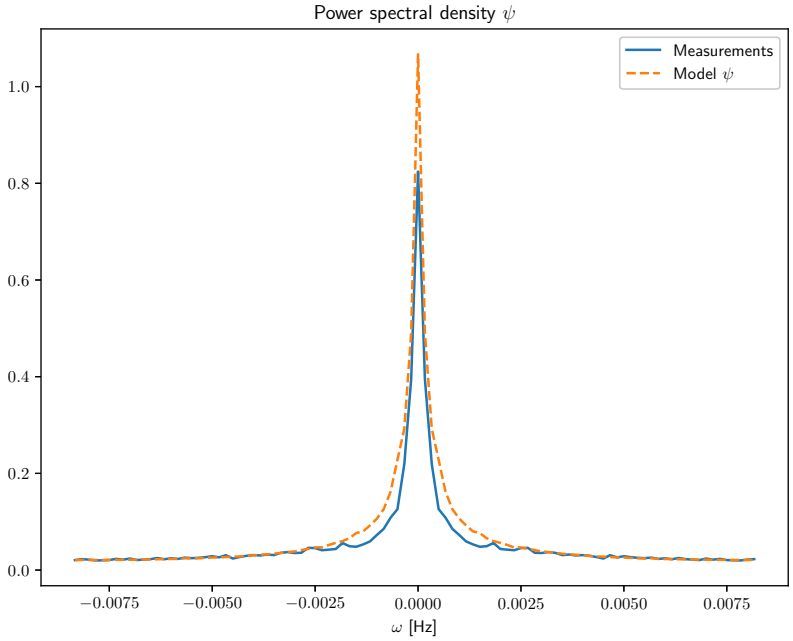


Figure 3.19: Power spectral density function of the yaw model vs measurements.

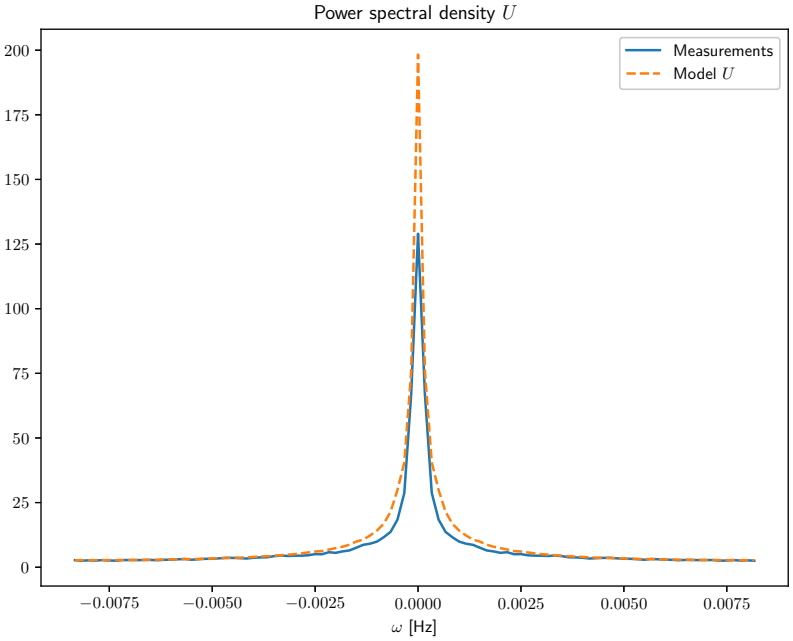


Figure 3.20: Power spectral density function of the speed model vs measurements.

These figures suggest that the power spectral density of the simulated wind yaw and speed aligns

adequately with the measured power spectral density. This alignment further validates the accuracy of the new wind model.

3.4.2 Validation summary

The results presented in this section demonstrate that the fitted models provide good estimation of the autocorrelations. Furthermore, the simulations based on these models generate autocorrelations and power spectral densities that align adequately with the observed data. This agreement validates the Yaw-Velocity model, demonstrating its capacity to appropriately reflect the dynamics of wind speed and yaw.

Although the model was fitted using the first 50 lags, its performance was evaluated over the first 100 lags. The model still provides a good fit when extended to 100 lags, indicating that it has not overfitted to the initial 50 lags. Overfitting is a common problem in statistical modelling in which a model fits the training data too closely and fails to generalize well to previously unseen data [56]. In this case, the model's ability to accurately represent data beyond the first 50 lags demonstrates its robustness. This suggests that limiting the simulations to 50 minutes is a conservative approach. However, in the context of this thesis, a conservative approach seems appropriate to assure the results' robustness and reliability.

Chapter 4

Process model

The previously introduced dynamics in Chapter 2 and Chapter 3 cover all the dynamics needed in the different modules in the system structure making up the wind turbine in a simulation. Joining these dynamics in a process model makes it more clear how the process of a wind turbine works. This insight allows for a better overview of the system along with its main use, which is providing dynamics of the system for state estimation, this is further discussed in Chapter 5.

4.1 Overview of dynamic models used

The dynamic models employed are essential for creating a functional and realistic simulation of the actual system. These same equations, or in the case of the airloads method, are also used in state estimation to depict the expected dynamics, thereby enhancing the accuracy of state estimation. An alternative approach might be to treat unmodelled dynamics as white noise. While this could be acceptable for dynamics that are not significantly noticeable or are computationally intensive and/or complex to model, the use of dynamic models provides a more accurate representation of the system's behavior. The dynamics used in this chapter are highlighted throughout the text as **Key Results**. Firstly the inflow dynamics, highlighted in Key Result 1, is reiterated here:

$$Q^{-1}P(\chi)\dot{\boldsymbol{\sigma}} = \hat{\boldsymbol{\tau}} - \boldsymbol{\sigma}, \tag{4.1}$$

Following this is Key Result 2, which provides a method for calculating the airloads $\boldsymbol{\tau}$ needed to calculate the inflow dynamics.

$$\boldsymbol{\tau} = a(\boldsymbol{v}, \boldsymbol{\omega}) \tag{4.2}$$

where $\boldsymbol{v}$ and $\boldsymbol{\omega}$ are the linear and angular relative velocities, respectively.

Then Key-Result 3 presents the rotor dynamics essential in calculating relative velocities and thereby also essential for calculating airloads.

$$J\dot{\Omega} = -M_z - M_E \tag{4.3}$$

where M_z is the aerodynamical torque in z and M_E is the Generator shaft torque as introduced in Definition 2.10.

Lastly the wind models presented in Key-Result 4 and Key-Result 5 determines the random stochastic wind, to which the wind turbine is subjected to. These two models also serve the purpose of providing information on the shaping filter rather than treating it as white noise. The models are reiterated:

Cartesian wind model:

$$\begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix} \begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} = - \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} + \begin{bmatrix} \mathcal{K}_1 & 0 \\ 0 & \mathcal{K}_2 \end{bmatrix} \mathbf{w} \tag{4.4}$$

Yaw-Velocity:

$$\frac{\psi(s)}{\mathrm{w}} = \frac{K_\psi}{T_\psi s + 1} \implies T_\psi \dot{\psi} = -\psi + K_\psi \mathrm{w} \tag{4.5}$$

$$\frac{U'(s)}{\mathrm{w}} = \frac{K_U}{T_U s + 1} \implies T_U \dot{U}' = -U' + K_U \mathrm{w} \tag{4.6}$$

$$U = U' + \bar{U} \tag{4.7}$$

These dynamics constitute all the modes in the system structure as well as the two versions of wind dynamics. This allows for realistic simulations.

4.1.1 Nonlinear process models

Gathering these dynamics results in two nonlinear process models:

Cartesian wind model:

$$\begin{bmatrix} Q^{-1}P(\chi) & 0 & 0 & 0 \\ 0 & J & 0 & 0 \\ 0 & 0 & T_1 & 0 \\ 0 & 0 & 0 & T_2 \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\sigma}} \\ \dot{\Omega} \\ \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{\tau}} - \boldsymbol{\sigma} \\ -M_z - M_E \\ -\xi_1 \\ -\xi_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \mathcal{K}_1 & 0 \\ 0 & \mathcal{K}_2 \end{bmatrix} \mathbf{w} \tag{4.8}$$

Yaw-Velocity:

$$\begin{bmatrix} Q^{-1}P(\chi) & 0 & 0 & 0 \\ 0 & J & 0 & 0 \\ 0 & 0 & T_U & 0 \\ 0 & 0 & 0 & T_\psi \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\sigma}} \\ \dot{\Omega} \\ \dot{U}' \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{\tau}} - \boldsymbol{\sigma} \\ -M_z - M_E \\ -U' \\ -\psi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ K_U & 0 \\ 0 & K_\psi \end{bmatrix} \mathbf{w} \tag{4.9}$$

These nonlinear process models delineate the dynamics of the internal variable σ, the angular velocity, wind speed and yaw.

4.2 Linearization of the process models

The use of Kalman filter or Extended Kalman filter requires a linearized process model, as these rely on a linear process model.

The nonlinearities to be addressed in the nonlinear process model are:

- M_E
- $Q^{-1}P(\chi)$
- $\boldsymbol{\tau}$

The linearization for M_E is straightforward using the first-order Taylor series expansion:

$$M_E \approx b|\Omega_0|\Omega_0 + 2b|\Omega_0|(\Omega - \Omega_0) \tag{4.10}$$

Similarly, the matrix $Q^{-1}P(\chi)$ can be linearized using a zeroth-order approximation:

$$Q^{-1}P(\chi) \approx Q^{-1}P(\chi_0) \tag{4.11}$$

Here χ_0 is the yaw error linearized about.

What is left is the linearization of $\boldsymbol{\tau}$, which is more comprehensive on its own, but it also depends on the wind model used. The following sections will explore the details of this linearization.

4.2.1 Relative velocities

The linearization of (4.2) depends on the definition of the relative linear and angular velocities. For a clearer derivation, the following definitions are introduced:

Definition 4.1 (Induction factor)**.** *The induction factor of inflow η can be expressed as:*

$$\boldsymbol{\eta} = L(\chi)\boldsymbol{\sigma} + D(\chi)\bar{\boldsymbol{\tau}} = \begin{bmatrix} \eta_x \\ \eta_y \\ \eta_z \\ \eta_\phi \\ \eta_\theta \\ \eta_\psi \end{bmatrix}, \quad \boldsymbol{\eta}_1 \triangleq \begin{bmatrix} \eta_x \\ \eta_y \\ \eta_z \end{bmatrix}, \quad \boldsymbol{\eta}_2 \triangleq \begin{bmatrix} \eta_\phi \\ \eta_\theta \\ \eta_\psi \end{bmatrix} \tag{4.12}$$

$$\boldsymbol{U_i} = \boldsymbol{\eta} U_\infty \tag{4.13}$$

Definition 4.2 (Relative velocities)**.**

$$\boldsymbol{v} = \begin{bmatrix} U_s \\ 0 \\ U_n \end{bmatrix} - \boldsymbol{\eta}_1 U_\infty, \quad \boldsymbol{\omega} = \begin{bmatrix} 0 \\ 0 \\ \Omega \end{bmatrix} - \boldsymbol{\eta}_2 U_\infty \tag{4.14}$$

$$\boldsymbol{\tau} = a(\boldsymbol{v}, \boldsymbol{\omega}) \tag{4.15}$$

U_s and U_n will depend on the wind model:::

Definition 4.3 (Rotor wind)**.**

$$\textbf{\textit{Cartesian:}} \quad \begin{bmatrix} U_s \\ U_n \end{bmatrix} = \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} + \bar{\boldsymbol{U}} \tag{4.16}$$

$$\textbf{\textit{Yaw-Velocity:}} \quad \begin{bmatrix} U_s \\ U_n \end{bmatrix} = \begin{bmatrix} \sin(\psi - \theta) \\ \cos(\psi - \theta) \end{bmatrix} U_\infty \tag{4.17}$$

where θ is the wind turbine yaw as defined in Figure 3.2.

This sets the groundwork for the linearization of $\boldsymbol{\tau}$. The process of linearizing $\boldsymbol{\tau}$ differs based on the wind model used. The two linearizations are presented in the following sections.

4.2.2 Linearization I- Cartesian

The linearization of $\boldsymbol{\tau}$ can now be performed. As $\boldsymbol{\tau}$ is not an analytical function, but rather a nonlinear mathematical methodology, numerical linearization is required. The relative linear and angular velocities as defined in 4.2, along with U_s and U_n using the Cartesian approach (4.16) fully define the relative velocities. This allows for comprehensive numerical linearization. The mean component of the wind signal is assumed to be defined as:

$$\bar{\boldsymbol{U}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} U_0 \tag{4.18}$$

where U_0 is the free-stream flow of the wind where the model is linearized about. It makes sense to define this such that the mean component is assumed to be purely in U_n, i.e. that $\chi = 0$. This also simplifies notation for the following derivation.

With the linear and angular velocities defined, they can be reiterated as the following functions:

$$\boldsymbol{v}(\xi_1, \xi_2, \boldsymbol{\sigma}, \boldsymbol{\tau}, \chi, \theta), \quad \boldsymbol{\omega}(\xi_1, \xi_2, \boldsymbol{\sigma}, \boldsymbol{\tau}, \chi, \theta, \Omega) \tag{4.19}$$

note that $U_\infty = \sqrt{U_s^2 + U_n^2} = \sqrt{\xi_1^2 + (\xi_2 + U_0)^2}$[1] and U_0 is a constant. θ is slowly varying and will not be considered.

Linearization requires the Jacobian matrix of $\boldsymbol{\tau}$, which is the partial derivatives of τ with respect to the variables involving the linear and angular velocities (4.19). The partial derivatives can be found numerically by utilizing the definition of the derivative:

$$\frac{\partial a(\boldsymbol{v}, \boldsymbol{\omega})}{\partial x} = \lim_{h \to 0} \frac{a(\boldsymbol{v}(x+h), \boldsymbol{\omega}(x+h)) - a(\boldsymbol{v}(x), \boldsymbol{\omega}(x))}{h} \tag{4.20}$$

this simplified notation's purpose is to clarify that the perturbation of a variable e.g. σ_1 will affect both linear and angular velocities and then implicitly $\boldsymbol{\tau}$.

The linearized $\boldsymbol{\tau}$ can then be put in the compact notation:

$$\Delta\boldsymbol{\tau} \approx \frac{\partial a}{\partial \boldsymbol{V}} \Delta \boldsymbol{V} \tag{4.21}$$

$$\frac{\partial a}{\partial \boldsymbol{V}} = \begin{bmatrix} \frac{\partial a}{\partial \boldsymbol{\sigma}} & \frac{\partial a}{\partial \Omega} & \frac{\partial a}{\partial \xi_1} & \frac{\partial a}{\partial \xi_2} & \frac{\partial a}{\partial \boldsymbol{\tau}} & \frac{\partial a}{\partial \chi} \end{bmatrix}, \quad \Delta \boldsymbol{V} = \begin{bmatrix} \Delta\boldsymbol{\sigma} \\ \Delta\Omega \\ \Delta\xi_1 \\ \Delta\xi_2 \\ \Delta\boldsymbol{\tau} \\ \Delta\chi \end{bmatrix} \tag{4.22}$$

where $\frac{\partial a}{\partial \boldsymbol{V}}$ is the Jacobian matrix of $\boldsymbol{\tau}$ and $\Delta\boldsymbol{V}$ are the variables involved in the linear and angular velocities.

Note that this involves $\frac{\partial a}{\partial \boldsymbol{\tau}}$ meaning that a is implicit. In the process of linearizing the implicit function, it is assumed that $\boldsymbol{\tau}$ has a negligible impact on the linearization. This allows for a simplification of the linearization by making it explicit. Another challenge arises from the fact that the linearization depends on the yaw error χ, i.e the quantity to be estimated. This can be resolved by approximating χ. Doing so decouples it from the linearization. The linear approximation of χ is given by:

$$\chi = \tan^{-1}\left(\frac{U_s}{U_n}\right) \approx \chi(0) + \frac{\partial \chi}{\partial \xi_1}\Delta\xi_1 + \frac{\partial \chi}{\partial \xi_2}\Delta\xi_2 = \hat{\chi} \tag{4.23}$$

[1] This is emphasized because U_∞ is used to calculate the induced inflow $\boldsymbol{U_i}$

this approximation leads to:

$$\frac{\partial a}{\partial \chi}\Delta\chi \approx \frac{\partial a}{\partial \chi}\frac{\partial \chi}{\partial \xi_1}\Delta\xi_1 + \frac{\partial a}{\partial \chi}\frac{\partial \chi}{\partial \xi_2}\Delta\xi_2 \tag{4.24}$$

Thus justifying the following modification in the partial derivatives:

$$\frac{\partial a'}{\partial \xi_1} \triangleq \frac{\partial a}{\partial \xi_1} + \frac{\partial a}{\partial \chi}\frac{\partial \chi}{\partial \xi_1}, \quad \frac{\partial a'}{\partial \xi_2} \triangleq \frac{\partial a}{\partial \xi_2} + \frac{\partial a}{\partial \chi}\frac{\partial \chi}{\partial \xi_2} \tag{4.25}$$

The linearization then becomes:

$$\Delta\boldsymbol{\tau} \approx \begin{bmatrix} \frac{\partial a}{\partial \boldsymbol{\sigma}} & \frac{\partial a}{\partial \Omega} & \frac{\partial a'}{\partial \xi_1} & \frac{\partial a'}{\partial \xi_2} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\sigma} \\ \Delta\Omega \\ \Delta\xi_1 \\ \Delta\xi_2 \end{bmatrix} \tag{4.26}$$

Normalization

It can be seen in (4.1) that $\boldsymbol{\tau}$ is normalized in the inflow dynamics, leading to nonlinearities needing to be addressed. Recall that $\boldsymbol{\tau}$ is normalized as follows:

$$\bar{\boldsymbol{\tau}} = \frac{a(\boldsymbol{v}, \boldsymbol{\omega})}{\frac{1}{2}\rho\pi R^2 U_\infty^2}, \quad U_\infty = \sqrt{\xi_1^2 + (\xi_2 + U_0)^2} \tag{4.27}$$

$\bar{\boldsymbol{\tau}}$ must therefore be linearized, which can be done in the straightforward manner:

$$\begin{aligned} \Delta\bar{\boldsymbol{\tau}} \approx \frac{1}{\frac{1}{2}\rho\pi R^2}\left(\frac{\Delta\boldsymbol{\tau}}{U_\infty(\xi_1(0), \xi_2(0))^2} - \frac{2\boldsymbol{\tau}_0}{U_\infty(\xi_1(0), \xi_2(0))^3}\Delta U_\infty(\xi_1, \xi_2)\right) \\ \approx \frac{\Delta\boldsymbol{\tau}}{\frac{1}{2}\rho\pi R^2 U_\infty(\xi_1(0), \xi_2(0))^2} \end{aligned} \tag{4.28}$$

This approximation is valid under the assumption that the second term is negligible.

The normalizing variable then becomes:

$$\text{norm}\mathbf{I} = \frac{1}{\frac{1}{2}\rho\pi R^2 U_\infty(\xi_1(0), \xi_2(0))^2} = \frac{1}{\frac{1}{2}\rho\pi R^2\big(\xi_1(0)^2 + (\xi_2(0) + U_0)^2\big)} \tag{4.29}$$

This concludes the linearization of $\boldsymbol{\tau}$ using the Cartesian wind model. The following section will introduce the linearization of $\boldsymbol{\tau}$ using the Yaw-Velocity model.

4.2.3 Linearization II- Yaw-Velocity

The linearization of $\boldsymbol{\tau}$ in the case of using the Yaw-Velocity model is quite similar to the linearization in 4.2.2, therefore only the essentials will be reiterated. $\boldsymbol{\tau}$ is still assumed to have negligible impact on the performance of the linearization as a simplification. θ is slowly varying and will not be considered in the linearization.

The variables can be identified by representing the linear and angular velocities as:

$$\boldsymbol{v}(\psi, U_\infty, \boldsymbol{\sigma}, \boldsymbol{\tau}, \chi, \theta), \quad \boldsymbol{\omega}(\psi, U_\infty, \boldsymbol{\sigma}, \boldsymbol{\tau}, \chi, \theta, \Omega) \tag{4.30}$$

The yaw error, χ, is defined as:

$$\chi = \tan^{-1}\left(\frac{U_s}{U_n}\right) = \psi - \theta \tag{4.31}$$

as previously stated. $\Delta\chi$ can then be expressed as:

$$\Delta\chi = \Delta\psi - \Delta\theta \tag{4.32}$$

This linearization will be utilized in the Extended Kalman filter, which implies that the linearization will be updated in each iteration. Given that θ is slowly varying, it is reasonable to assume that $\Delta\theta$ will be very small[2]. Therefore, the following approximation is considered to be adequate:

$$\Delta\chi \approx \Delta\psi \tag{4.33}$$

This leads to the following partial derivative:

$$\frac{\partial a'}{\partial \psi} \triangleq \frac{\partial a}{\partial \psi} + \frac{\partial a}{\partial \chi} \tag{4.34}$$

which in turn results in the following linearization:

$$\Delta\boldsymbol{\tau} \approx \begin{bmatrix} \frac{\partial a}{\partial \boldsymbol{\sigma}} & \frac{\partial a}{\partial \Omega} & \frac{\partial a}{\partial U} & \frac{\partial a'}{\partial \psi} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\sigma} \\ \Delta\Omega \\ \Delta U \\ \Delta\psi \end{bmatrix} \tag{4.35}$$

Note that $\Delta U = \Delta U'$.

Normalization

The normalization appears here aswell.

$$\bar{\boldsymbol{\tau}} = \frac{a(\boldsymbol{v}, \boldsymbol{\omega})}{\frac{1}{2}\rho\pi R^2 U_\infty^2} \tag{4.36}$$

which is linearized:

$$\Delta\bar{\boldsymbol{\tau}} \approx \frac{1}{\frac{1}{2}\rho\pi R^2}\left(\frac{\Delta\boldsymbol{\tau}}{U_\infty(0)^2} - \frac{2\boldsymbol{\tau}_0}{U_\infty(0)^3}\Delta U_\infty\right) \approx \frac{\Delta\boldsymbol{\tau}}{\frac{1}{2}\rho\pi R^2 U_\infty(0)^2} \tag{4.37}$$

The normalizing variable then becomes:

$$\text{norm}\mathbf{II} = \frac{1}{\frac{1}{2}\rho\pi R^2 U_\infty(0)^2} \tag{4.38}$$

This concludes the linearization of $\boldsymbol{\tau}$ using the Yaw-Velocity model.

[2]In the implementation of the Extended Kalman filter on the Yaw-Velocity model, the time step is 100ms. With a maximum yaw rate of 0.3 deg/s, this amounts to a maximum delta error of 0.03 degrees from this simplification.

4.3 Linear process model

With the linearization of M_E, $Q^{-1}P(\chi)$, and both versions of $\boldsymbol{\tau}$ now complete, the linear process model can be constructed. The linearization process introduces delta variables, which necessitates the transformation of the state derivatives, such as $\dot{\Omega}$, into their delta counterparts, such as $\Delta\dot{\Omega}$

From (4.10), the change in $\dot{\Omega}$ can be expressed as:

$$\dot{\Omega} - \dot{\Omega}_0 = -\Delta M_z - |\Omega|\Omega b + |\Omega_0|\Omega_0 b \approx -\Delta M_z - 2b|\Omega_0|\Delta\Omega \tag{4.39}$$

The changes in $\Delta\dot{\boldsymbol{\sigma}}$, $\Delta\dot{\xi}_1$, $\Delta\dot{\xi}_2$, $\Delta\dot{U}$ and $\Delta\dot{\psi}$ are found in a similar straightforward manner. The linearized process models for the two wind models can then be presented in the following sections.

4.3.1 Cartesian wind model

The process model utilizing Cartesian approach can now be pieced together. Firstly, to simplify notation the Jacobian is defined.

Definition 4.4 (Jacobian matrix)**.** *Let the Jacobian matrix found in* (4.26) *be denoted by:*

$$\boldsymbol{J}_1 = \begin{bmatrix} \frac{\partial a_1}{\partial \sigma_1} & \cdots & \frac{\partial a_1}{\partial \sigma_5} & \frac{\partial a_1}{\partial \Omega} & \frac{\partial a_1'}{\partial \xi_1} & \frac{\partial a_1'}{\partial \xi_2} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial a_5}{\partial \sigma_1} & \cdots & \frac{\partial a_5}{\partial \sigma_5} & \frac{\partial a_5}{\partial \Omega} & \frac{\partial a_5'}{\partial \xi_1} & \frac{\partial a_5'}{\partial \xi_2} \end{bmatrix} \frac{1}{norm\boldsymbol{I}}, \quad \boldsymbol{J}_2 = \begin{bmatrix} \frac{\partial a_6}{\partial \sigma_1} & \cdots & \frac{\partial a_6}{\partial \sigma_5} & \frac{\partial a_6}{\partial \Omega} & \frac{\partial a_6'}{\partial \xi_1} & \frac{\partial a_6'}{\partial \xi_2} \end{bmatrix} \tag{4.40}$$

$$\boldsymbol{J}_\xi = \begin{bmatrix} \boldsymbol{J}_1 \\ \boldsymbol{J}_2 \end{bmatrix} \tag{4.41}$$

Then the linearized process model takes the form: [3]

$$\begin{bmatrix} Q^{-1}P(\hat{\chi}) & 0 & 0 & 0 \\ 0 & J & 0 & 0 \\ 0 & 0 & T_1 & 0 \\ 0 & 0 & 0 & T_2 \end{bmatrix} \begin{bmatrix} \Delta\dot{\boldsymbol{\sigma}} \\ \Delta\dot{\Omega} \\ \Delta\dot{\xi}_1 \\ \Delta\dot{\xi}_2 \end{bmatrix} = \left(\begin{bmatrix} \mathbf{J}_1 \\ \mathbf{J}_2 \\ 0_{1x8} \\ 0_{1x8} \end{bmatrix} - \begin{bmatrix} I(5) & 0_{5x1} & 0_{5x1} & 0_{5x1} \\ 0_{1x5} & 2b|\Omega_0| & 0 & 0 \\ 0_{2x5} & 0 & -1 & 0 \\ 0_{2x5} & 0 & 0 & -1 \end{bmatrix} \right) \begin{bmatrix} \Delta\boldsymbol{\sigma} \\ \Delta\Omega \\ \Delta\xi_1 \\ \Delta\xi_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \mathcal{K}_1 & 0 \\ 0 & \mathcal{K}_2 \end{bmatrix} \mathbf{w} \tag{4.42}$$

This results in the compact form notation:

$$\mathcal{T}\dot{\boldsymbol{x}} = M_1\boldsymbol{x} + M_2\mathbf{w} \tag{4.43}$$

where $\boldsymbol{x}$ is the 8x1 state vector and $\mathbf{w}$ is the 2x1 white noise signal. The linearized process model on standard form is then defined.

Key Result 6 (Linearized process model utilizing the Cartesian wind model)**.**

$$A \triangleq \mathcal{T}^{-1}M_1, G \triangleq \mathcal{T}^{-1}M_2 \tag{4.44}$$

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + G\boldsymbol{w} \tag{4.45}$$

[3] Note that $Q^{-1}P(\hat{\chi})$ is the 0-th order approximation of $Q^{-1}P(\chi)$ in $\hat{\chi}$.

4.3.2 Yaw-Velocity

Definition 4.5 (Jacobian matrix)**.** *Let the Jacobian matrix found in* (4.35) *be denoted by:*

$$\boldsymbol{J}_1 = \begin{bmatrix} \frac{\partial a_1}{\partial \sigma_1} & \cdots & \frac{\partial a_1}{\partial \sigma_5} & \frac{\partial a_1}{\partial \Omega} & \frac{\partial a_1'}{\partial U} & \frac{\partial a_1'}{\partial \psi} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial a_5}{\partial \sigma_1} & \cdots & \frac{\partial a_5}{\partial \sigma_5} & \frac{\partial a_5}{\partial \Omega} & \frac{\partial a_5'}{\partial U} & \frac{\partial a_5'}{\partial \psi} \end{bmatrix} \frac{1}{norm\boldsymbol{II}}, \quad \boldsymbol{J}_2 = \begin{bmatrix} \frac{\partial a_6}{\partial \sigma_1} & \cdots & \frac{\partial a_6}{\partial \sigma_5} & \frac{\partial a_6}{\partial \Omega} & \frac{\partial a_6'}{\partial \xi_1} & \frac{\partial a_6'}{\partial \xi_2} \end{bmatrix} \tag{4.46}$$

$$\boldsymbol{J}_\psi = \begin{bmatrix} \boldsymbol{J}_1 \\ \boldsymbol{J}_2 \end{bmatrix} \tag{4.47}$$

The linearized process model takes the form [4]:

$$\begin{bmatrix} Q^{-1}P(\psi-\theta) & 0 & 0 & 0 \\ 0 & J & 0 & 0 \\ 0 & 0 & T_U & 0 \\ 0 & 0 & 0 & T_\psi \end{bmatrix} \begin{bmatrix} \Delta\dot{\boldsymbol{\sigma}} \\ \Delta\dot{\Omega} \\ \Delta\dot{U}' \\ \Delta\dot{\psi} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \mathbf{J}_1 \\ \mathbf{J}_2 \\ 0_{1x8} \\ 0_{1x8} \end{bmatrix} - \begin{bmatrix} I(5) & 0_{5x1} & 0_{5x1} & 0_{5x1} \\ 0_{1x5} & 2b|\Omega_0| & 0 & 0 \\ 0_{2x5} & 0 & -1 & 0 \\ 0_{2x5} & 0 & 0 & -1 \end{bmatrix} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\sigma} \\ \Delta\Omega \\ \Delta U' \\ \Delta\psi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ K_U & 0 \\ 0 & K_\psi \end{bmatrix} \mathbf{w} \tag{4.48}$$

This results in the compact form notation:

$$\mathcal{T}\dot{\boldsymbol{x}} = M_1\boldsymbol{x} + M_2\mathbf{w} \tag{4.49}$$

where $\boldsymbol{x}$ is the 8x1 state vector and $\mathbf{w}$ is the 2x1 white noise signal. The linearized process model on the standard form is then defined.

Key Result 7 (Linearized process model Yaw-Velocity)**.**

$$A \triangleq \mathcal{T}^{-1}M_1, G \triangleq \mathcal{T}^{-1}M_2 \tag{4.50}$$

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + G\boldsymbol{w} \tag{4.51}$$

[4] $Q^{-1}P(\psi-\theta)$ is the 0-th order approximation of $Q^{-1}P(\chi)$.

Chapter 5

State estimation

State estimation plays a crucial role in systems where direct measurements are either impractical, prohibitively expensive, or impossible. It is particularly relevant in the case of yaw error, where measurements can be significantly influenced by the system's own dynamics. For instance, a wind-vane's readings can be heavily affected by flow distortions, thereby compromising the accuracy of the measurements. Another approach to measuring yaw error involves the use of Lidar, a technology that, while effective, can be costly and prone to failure. Furthermore, Lidar measurements can still be influenced by external factors, which may lead to inaccuracies.

As mentioned in Subsection 1.3.1, the author did not find examples of this being done before. This work aims to fill this gap, demonstrating the feasibility and effectiveness of state estimation for yaw error in wind turbines.

Before getting into state estimation techniques, this chapter begins with a discussion on measurements and their role in ensuring that the system is observable. The fundamental idea behind state estimation is to use a set of measurements to provide an estimate of the states. This can be done utilizing the systems dynamics and/or properties with the measurements serving as feedback for the estimates.

Subsequently, the chapter explores several state estimation schemes, beginning with a simple constant estimate for yaw error. The principles of the Kalman filter are then discussed, followed by a detailed presentation of the algorithm. The chapter then concludes by modifying the Kalman filter algorithm to obtain the Extended Kalman filter.

5.1 Measurements

Many measurements are built-in or easily available for a wind turbine. Looking at the process model we can see that measuring all the variables in the state vector would be ideal. However, some of these variables can not be measured such as $\boldsymbol{\sigma}$ which is an internal state and not a measurable quantity. We can, however, measure the angular velocity of the rotor Ω and the airloads $\boldsymbol{\tau}$. If $\boldsymbol{\tau}$ is not directly available, it can potentially be determined using appropriate measurement techniques, such as the application of strain gauges.

In the context of a wind turbine, numerous measurements are inherently available or can be easily obtained. A review of the process model reveals that ideally, all state variables should be measured to ensure comprehensive understanding and control of the system. However, certain variables, such as $\boldsymbol{\sigma}$, represent internal states and are not directly measurable.

Despite this, there are key variables that can be effectively measured. For instance, the angular velocity of the rotor Ω and the airloads $\boldsymbol{\tau}$, can be relatively accurately captured. These measurements provide valuable insights into the operational state of the wind turbine and serve as crucial inputs for state estimation techniques.

Definition 5.1 (Measurments)**.**

$$y = \begin{bmatrix} \Omega \\ \boldsymbol{\tau} \end{bmatrix} \tag{5.1}$$

5.1.1 Observability

A prerequisite of state estimation is the observability of the system, which studies the possibility of estimating the state from the measurements[57]. To make sure that the chosen set of measurements is adequate for state estimation we can check the system's observability.

Definition 5.2 (Observability)**.** *The continuous-time LTI system:*

$$\dot{x} = Ax + Bu, \quad y = Cx + Du \tag{5.2}$$

is said to be observable if $x(0)$ can be determined uniquely by the knowledge of the input u and output y for $t \in [0, t_1]$.

To check whether a specific linearized system is observable the solution of the state space equation can be examined:

$$\begin{aligned} x(t) = e^{At}x(0) + \int_0^{t_1} e^{A(t_1-\tau)}Bu(\tau)\,d\tau \\ y(t) = Cx(t) + Du(t) \end{aligned} \tag{5.3}$$

$e^{At}x(0)$ is the solution of $\dot{x}$ in the case of $u(t) = 0$ often referred to as the natural response or unforced response.

$$\tilde{y} = y(t) - C\int_0^{t_1} e^{A(t_1-\tau)}Bu(\tau)\,d\tau - Du(t) = Ce^{At}x(0) \tag{5.4}$$

$y(t)$ and $u(t)$ is known, and therefore $\tilde{y}$ is also known. To determine $x(0)$ uniquely in the general case, i.e. not limited to the case of zero nullity, the information over a time interval must be used. This means that the unique solution if it exists, can be found by the following steps:

• Premultiplying $\tilde{y}$ by $(Ce^{At})^\intercal$ and integrating over the interval $[0, t_1]$ gives:

$$\begin{aligned} \int_0^{t_1} e^{A^\intercal\tau}C^\intercal\tilde{y}(\tau)\,d\tau = \int_0^{t_1} e^{A^\intercal\tau}C^\intercal Ce^{A\tau}\,d\tau\,x(0) \\ \mathcal{W}_0^{-1}\int_0^{t_1} e^{A^\intercal\tau}C^\intercal\tilde{y}(\tau)\,d\tau = I_n x(0) \end{aligned} \tag{5.5}$$

where I_n is the nxn identity matrix and $\mathcal{W}_0$ as defined:

Definition 5.3 (Observability Gramian)**.**

$$\mathcal{W}_0 \triangleq \int_0^{t_1} e^{A^\intercal\tau}C^\intercal Ce^{A\tau}\,d\tau \tag{5.6}$$

Following Definition 5.2 it can be seen that (5.2) is observable if and only if $\mathcal{W}_0$ is nonsingular for any $t > 0$ [58].

Duality

Now that an understanding of observability is established, another property of an observable system is presented. A related concept is *Controllability*, in fact the controllability Gramian $\mathcal{W}_c$ of the system:

$$\dot{z} = A^\intercal z + C^\intercal u \tag{5.7}$$

given u, A and C being the same as in (5.2) is defined as:

Definition 5.4 (Duality)**.**

$$\mathcal{W}_c^z = \mathcal{W}_0 \tag{5.8}$$

This means that if (5.2) is observable then (5.7) is controllable, and vice versa. This reveals another property:

Theorem 5.1.1 (Eigenvalues)**.** *Consider the pair (A, C). All eigenvalues of $(A - KC)$ can be placed arbitrarily by selecting a real constant vector K if and only if (A, C) is observable.*

Theorem 5.1.1 is proven to be a useful intuition of the importance of observability in closed-loop state estimation.

5.2 Constant estimate

A straightforward method for state estimation involves examining the relationship between two states that appear to be linked by a scaling factor. This factor is used to approximate one variable based on the other. For instance, consider the connection between the yaw error χ and the aerodynamic torque M_z. A rise in yaw error generally leads to a decrease in M_z, indicating a strong coupling.

From a steady-state simulation with a constant yaw error of $\chi(0) = 0.2914$ radians and a free-stream flow of $U_\infty(0) = \sqrt{4^2 + 1.2^2}$, a scaling factor between χ and M_z can be deduced as:

$$\frac{\chi(0)}{M_z(0)} = -4.51 \times 10^{-7} \tag{5.9}$$

This scaling factor can be used to approximate χ as follows:

$$\chi \approx -4.51 \times 10^{-7} \cdot M_z \tag{5.10}$$

This kind of constant estimate can be effective in steady-state conditions, where the system's behavior remains relatively constant over time. However, this approach has the drawback of not having feedback, and contrary to an open-loop estimator it does not utilize system knowledge. Without feedback and system knowledge, this constant estimate will fall short quickly once the system leaves the operating point. This is a key difference between this simple constant estimate approach and more sophisticated state estimation techniques like the Kalman filter and Extended Kalman filter, which incorporate feedback and system information to continuously update and refine their estimates.

To demonstrate the fallacy of the constant estimate the following wind signal was imposed on the wind turbine:

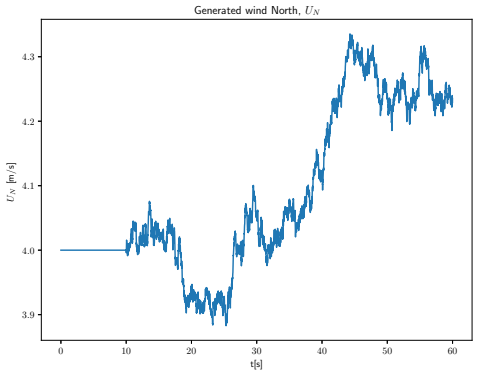


Figure 5.1: Generated North wind.

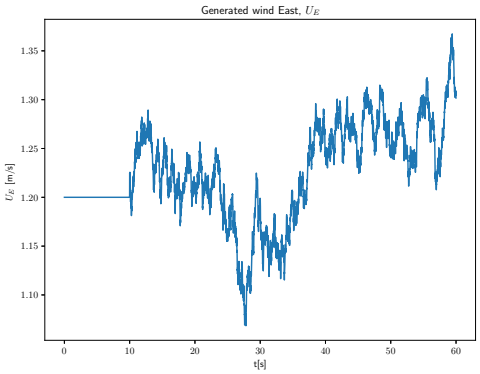


Figure 5.2: Generated East wind.

These figures illustrate that the initial 10 seconds of the simulation remain in a steady-state, consistent with the conditions where the scaling factor was identified. However, after this initial period, turbulent flow is introduced into the wind signal, pushing the system beyond its operating point.

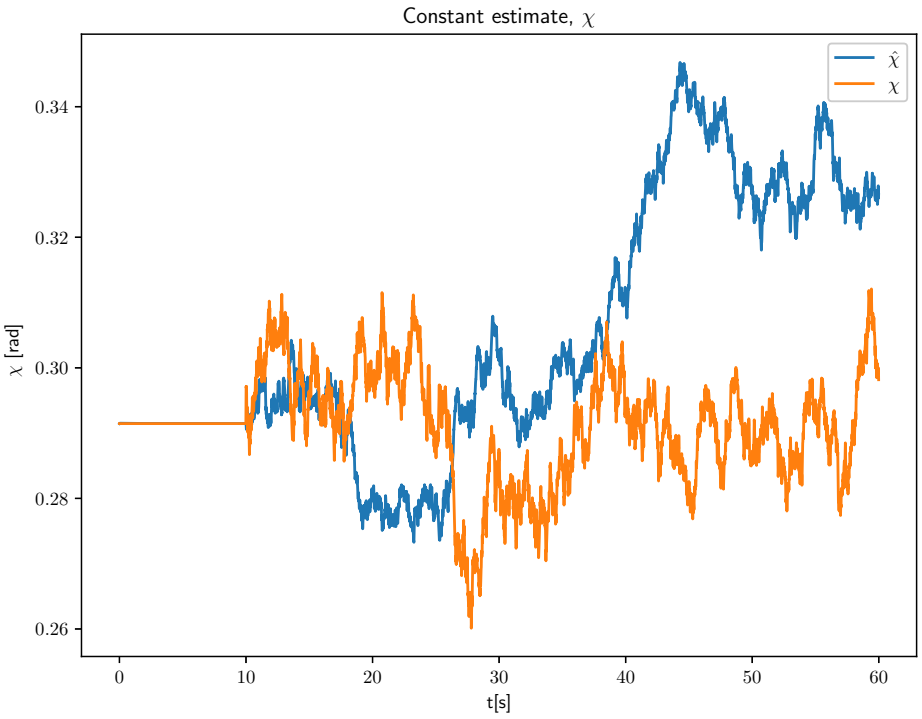


Figure 5.3: Constant estimate of the yaw error χ.

It can be seen from the figure that the constant estimate strays from the real value as soon as it leaves steady state. While the constant estimation method provides a simple and computationally efficient way to predict system behavior in steady-state conditions, its practical applicability is very limited. This emphasizes the significance of incorporating feedback and detailed system knowledge into the estimation process.

5.3 Kalman filter

42212, is the number of citations of the paper introduced by Rudolf E. Kalman in 1960 [59]. The Kalman filter is widely used in sensor and navigation systems [60], object tracking [61], and more. This simple, yet efficient algorithm has passed the test of time. Several variations of the Kalman filter have been derived over the years, such as The Ensemble Kalman filter, The Unscented Kalman filter, the Extended Kalman filter, the Multiplicative Extended Kalman filter, and more [50, 60]. This section covers the ordinary Kalman filter.

5.3.1 Assumptions

It is assumed that the random process can be modelled on the form:

$$x_{k+1} = A_k x_k + \mathrm{w}_k \tag{5.11}$$

Here x_k is the state vector of the random process, A_k is the state transition matrix and w_k is the input white noise.

The measurements vector is assumed to be discrete with the linear relationship:

$$y_k = C_k x_k + \mathrm{v}_k \tag{5.12}$$

where y_k is the measurement vector, C_k is the output matrix and v_k is the white noise measurement error.

Assumption 3 (Covariance of uncorrelated white noise)**.** *w_k and v_k have the known covariance matrices:*

$$E[w_k w_i^\intercal] = \begin{cases} Q_k & \text{for } i = k \\ 0 & \text{for } i \neq k \end{cases} \tag{5.13}$$

$$E[v_k v_i^\intercal] = \begin{cases} R_k & \text{for } i = k \\ 0 & \text{for } i \neq k \end{cases} \tag{5.14}$$

Assumption 4 (Uncorrelated w_k and v_k)**.** *The input white noise and measurement error are assumed to be uncorrelated:*

$$E[w_k v_i^\intercal] = 0 \quad \forall\, k, i \tag{5.15}$$

Assumption 5 (Observability)**.** *The pair (A_k, C_k) is observable.*

Building upon Theorem 5.1.1, this assumption permits the adjustment of the Kalman gain, as shown in (5.20), to arbitrarily manipulate the eigenvalues of the term $(A_k - K_k C_k)$.

5.3.2 Model augmentation

Unlike deterministic approaches, the Kalman filter has the advantage of being able to account for colored noise. This process can be seen in Chapter 3, where the shaping filter is identified. The shaping filter, which transforms white noise into colored noise, contains information about the frequency distribution. This information can be incorporated into the Kalman filter by augmenting the state space. The implication of this is that the information about the frequency distribution can be used to penalize unlikely changes, thereby improving the estimate.

Example 1 (Augmented state-space)**.** *Consider the system:*

$$\dot{x} = x + \Gamma, \quad \Gamma \triangleq H(s)w \tag{5.16}$$

where w is white noise and H(s) is the shaping filter:

$$H(s) = \frac{K}{Ts+1} = \frac{\Gamma}{w} \tag{5.17}$$

The following augmentation can be done:

$$\begin{bmatrix} \dot{x} \\ \dot{\Gamma} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & -\frac{1}{T} \end{bmatrix} \begin{bmatrix} x \\ \Gamma \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K}{T} \end{bmatrix} w \tag{5.18}$$

The example shows that information about Γ *can be extracted instead of treating it as white noise.*

5.3.3 Kalman filter algorithm

An overview of the notation used in the Kalman filter is presented:

Symbol	Definition
x_k	State vector
y_k	measurements
$\bar{x}_{k+1} = A_k\hat{x}_k$	a priori estimate of x_k
$\hat{x}_k = \bar{x}_k + K_k(y_k - C_k\bar{x}_k)$	a posteriori estimate of x_k
$\bar{e}_k = x_k - \bar{x}_k$	a priori estimation error
$\hat{e}_k = x_k - \hat{x}_k$	a posteriori estimation error
$\bar{P}_k = E[\bar{e}_k\bar{e}_k^\intercal]$	a priori error covariance matrix
$\hat{P}_k = E[\hat{e}_k\hat{e}_k^\intercal]$	a posteriori covariance matrix
K_k	Kalman gain

Table 5.1: Notations used in the the Kalman filter.

It can be seen from Table 5.1 that the wording *a priori* and *a posteriori* is used. This notation describes the flow of the algorithm, where a prior estimate $\bar{x}_k$ is used to obtain a new estimate $\hat{x}_k$. The initial iteration of the Kalman filter algorithm utilizes initial estimates of $\bar{x}_0$ and $\bar{P}_0$, where $\bar{x}_0$ is usually set in the operating point and $\bar{P}_0$ is usually set to identity. An alternative initialization method for the error covariance matrix involves using the final value from a preceding simulation. This technique is commonly referred to as a *warm start*, which can give reduced initial transients in the estimate.

Following this initialization, the Kalman gain K_k can be calculated by minimizing the trace of the error covariance matrix, i.e the variances of the state estimate error. This can be done by differentiating the trace of $\hat{P}_k$ with respect to K_k and solving for $\frac{d(tr)}{dK_k} = 0$:

$$\frac{d(tr)}{dK_k} = -2(C_k\bar{P}_k)^\intercal + 2K_k(C_k\bar{P}_kC_k^\intercal + R_k) = 0 \tag{5.19}$$

The Kalman gain is then found to be:

$$K_k = \bar{P}_kC^\intercal(C_k\bar{P}_kC_k^\intercal + R_k)^{-1} \tag{5.20}$$

The *a priori* and *a posteriori* error covariance matrices can now be found as:

$$\begin{aligned} \hat{P}_k &= (I - K_kC_k)\bar{P}_k^-(I - K_kC_k)^\intercal + K_kR_kK_k^\intercal \\ \bar{P}_{k+1} &= A_k\hat{P}_kA_k^\intercal + Q_k \end{aligned} \tag{5.21}$$

note that $\hat{P}_k$ takes on many forms, however, the presented form is valid for any K_k.

The notations used in Table 5.1 along with (5.20) and (5.21) constitutes the Kalman filter, summarized in the following algorithm.

Algorithm 1 Kalman filter

1: $K_k = \bar{P}_k C_k^\intercal (C_k \bar{P}_k C_k^\intercal + R_k)^{-1}$
2: $\hat{x}_k = \bar{x}_k + K_k(y_k - C_k \bar{x}_k)$
3: $\hat{P}_k = (I - K_k C_k)\bar{P}_k^- (I - K_k C_k)^\intercal + K_k R_k$
4: $\hat{x}_{\bar{k+1}} = A_k \hat{x}_k$
5: $\bar{P}_{k+1} = A_k \hat{P}_k A_k^\intercal + Q_k$
6: **return** $\hat{x}_k$

5.3.4 Linearized Kalman filter yaw error estimation using the Cartesian wind model.

Using the preceding formulation of the Kalman filter, and the linearized Cartesian process model in Key Result 6, a way of estimating the yaw error has been established. The linearized process model is repeated formulated on standard form:

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + G\mathbf{w}, \quad \boldsymbol{x} = \begin{bmatrix} \Delta\boldsymbol{\sigma} \\ \Delta\Omega \\ \Delta\xi_1 \\ \Delta\xi_2 \end{bmatrix}, \boldsymbol{x} \in \mathbb{R}^8 \tag{5.22}$$

notice that the linearized process model is time-dependent, therefore discretization is required. Before doing so the covariance matrices must be introduced.

Covariance matrices

The covariance matrices Q and R are tuning variables. An interpretation of the significance of the covariance matrix is made in [58]: "A large Q corresponds to little measurement noise and leads to state estimators that respond fast to changes in the measured output. A large R corresponds to small disturbances and leads to state estimates that respond cautiously (slowly) to unexpected changes in the measured output." Note that "large" in this sense is relative to the other variable, Q is said to be "large" if it is large relative to the corresponding R. In other words the pair Q and R will produce the same state estimate as the pair $10Q$ and $10R$. The dimensional size of Q and R depends on the system. In this case, Q will be a $2x2$ matrix because there is white noise in two states, i.e the white noise vector is:

$$\mathbf{w} = \begin{bmatrix} \mathrm{w}_1 \\ \mathrm{w}_2 \end{bmatrix} \tag{5.23}$$

R stems from the white noise measurement error v_k and is therefore $7x7$ assuming that Ω and $\boldsymbol{\tau}$ is measured. With this established, the continuous time system can be discretized.

$\bar{P}_k$ and $\hat{P}_k$ are both nxn matrices, where $\bar{P}_k$ can be initialized as the $8x8$ identity matrix $I(8)$.

Van Loan's method

The continuous to discrete conversion for a given sampling time dt can be found using Van Loan's method:

$$exp\left(\begin{bmatrix} A & GQG^\intercal \\ 0 & -A^\intercal \end{bmatrix} dt \right) = \begin{bmatrix} M_{11} & M_{12} \\ 0 & M_{22} \end{bmatrix}, A_d = M_{11}, Q_d = M_{12}M_{11}^\intercal \tag{5.24}$$

Here[1] A_d is the discretized version of A, which is the same as A_k used in derivation of the Kalman filter. Q_d is Q discretized and corresponds to Q_k. R can be discretized through an averaging convention:

$$R_d = \frac{R}{dt} \tag{5.25}$$

The discretization can then be summarized with the following definition:

Definition 5.5 (Discrete matrices)**.** *The matrices used in Algorithm 1 are defined as:*

$$Q_k \triangleq Q_d, \quad R_d \triangleq R_k, \quad A_d \triangleq A_k \tag{5.26}$$

Output matrix

The definition of the output matrix y_k is still to be determined. Firstly it is assumed that the airloads and the angular velocity measurements are obtainable, giving the measurements:

$$y = \begin{bmatrix} \Omega \\ \boldsymbol{\tau} \end{bmatrix} \tag{5.27}$$

the linear output matrix C_k can then be constructed using the results from Subsection 4.2.2:

Definition 5.6 (Linear output matrix)**.** *The linear output matrix is defined as:*

$$C_k = \begin{bmatrix} 0_{1x5} & 1 & 0 & 0 \\ \frac{\partial a}{\partial \sigma} & \frac{\partial a}{\partial \Omega} & \frac{\partial a'}{\partial \xi_1} & \frac{\partial a'}{\partial \xi_2} \end{bmatrix} \tag{5.28}$$

Notice that $C_k \bar{x}_k \approx \Delta\boldsymbol{\tau}$, this means that y_k must be defined as follows:

Definition 5.7.

$$y_k \triangleq y - y_0, \quad y_0 \triangleq \begin{bmatrix} \Omega_0 \\ \boldsymbol{\tau}_0 \end{bmatrix} \tag{5.29}$$

Here Ω_0 and $\boldsymbol{\tau}_0$ is the angular velocity and aerodynamic forces and torques at the operating point.

This concludes the section, where the Kalman filter has been introduced along with all the definitions needed for implementation.

[1] Note that *exp* is used for the exponential function for readability.

5.4 Extended Kalman filter

This section presents the Extended Kalman filter as a modification of the Linearized Kalman filter introduced in the previous section. EKF has a wide range of uses from pose tracking [62] to wind speed estimation [63, 64, 65].

The Extended Kalman filter(EKF) is different from the Kalman filter in the sense that it updates the estimated trajectory in each step. This means that the linearization and discretization of A and C happen in each step k. As a result, it is more convenient to keep track of the total estimates, e.g. $\boldsymbol{\sigma}$ instead of $\Delta\boldsymbol{\sigma}$. Conversely, if incremental estimates are used, then the linearization points have to be saved to reconstruct the estimates, rather than one stationary linearization point in the case of the Kalman filter. This modification necessitates y_k, $\hat{x}_k$ and $\bar{x}_{k+1}$ to be redefined. Conversely, K_k, $\hat{P}_k$ and $\bar{P}_{k+1}$ retain the same equation[2], see Algorithm 1 step 1, 3 and 5, respectively.

Before elaborating further, the state vectors for the two cases are presented:

Definition 5.8 (State vectors)**.**

$$\textbf{\textit{Cartesian wind model:}}\ \boldsymbol{x} = \begin{bmatrix} \boldsymbol{\sigma} \\ \Omega \\ \xi_1 \\ \xi_2 \end{bmatrix} \tag{5.30}$$

$$\textbf{\textit{Yaw-Velocity:}}\ \boldsymbol{x} = \begin{bmatrix} \boldsymbol{\sigma} \\ \Omega \\ U' \\ \psi \end{bmatrix} \tag{5.31}$$

It can be observed that the two cases share notation. When consulting a specific case, the corresponding state vector $\boldsymbol{x}$ and linearization process in Subsection 4.3.1 or Subsection 4.3.2 may be referred to. The notation of the remaining derivation is therefore general.

5.4.1 Linearization in each step

The linearization process described in Chapter 4 takes place after $\hat{x}_k$ is updated in each iteration of the Extended Kalman filter. This leads to the linearization being conducted around $\hat{x}_k$ instead of $\boldsymbol{x}(0)$. As the point of linearization is moved, y_k must be updated:

Definition 5.9 (Measurement update)**.**

$$y'_k \triangleq \begin{bmatrix} \hat{\Omega}_k \\ a(\boldsymbol{v}(\hat{x}_k, \boldsymbol{\tau}, \theta), \boldsymbol{\omega}(\hat{x}_k, \boldsymbol{\tau}, \theta)) \end{bmatrix} \tag{5.32}$$

$$y_k = y - y'_k \tag{5.33}$$

Please note that C_k, A_k and y_k are updated after $\hat{x}_k$. Consequently, C_{k-1} and y_{k-1} are used in the first two steps of Algorithm 1.

5.4.2 Total estimates

The linearized matrices A_k and C_k are designed to interact with an incremental state, a feature stemming from the linearization process. The incremental state can be obtained by subtracting $\hat{x}_k$ from the total estimate, viz.

[2]The changes in the other steps will of course implicitly affect these, however, the equations are not changed.

$$\bar{x}_{k+1} = A_k(\hat{x}_k - \hat{x}_k) \tag{5.34}$$

As stated earlier, it can be seen that $\bar{x}_{k+1}$ has a trivial solution. This issue is resolved in the following definition:

Definition 5.10 (Non trivial projection)**.**

$$\bar{x}_{k+1} = \begin{cases} \textit{Solution of the nonlinear process model at } t = t_{k+1} \\ \textit{subject to the initial condition } x = \hat{x}_k \end{cases} \tag{5.35}$$

The last modification needed is in updating $\hat{x}_k$, where incremental states must be used as follows:

$$\hat{x}_k = \bar{x}_k + K_k\Big(y_k - C_{k-1}(\bar{x}_k - \hat{x}_{k-1})\Big) \tag{5.36}$$

5.4.3 Summary

The Extended Kalman filter can be summarized by an augmented Algorithm:

Algorithm 2 Extended Kalman filter

1: $K_k = \bar{P}_k C_{k-1}^\intercal (C_{k-1}\bar{P}_k C_{k-1}^\intercal + R_k)^{-1}$
2: $\hat{x}_k = \bar{x}_k + K_k(y_{k-1} - C_{k-1}(\bar{x}_k - \hat{x}_{k-1}))$
3: Find A, C_k and y_k
4: Discretize A and Q to obtain A_k and Q_k
5: $\hat{P}_k = (I - K_k C_k)\bar{P}_k^-(I - K_k C_k)^\intercal + K_k R_k$
6: $\hat{x}_{k+1}^- =$ Solution of nonlinear process model at $x = \hat{x}_k$
7: $\bar{P}_{k+1} = A_k \hat{P}_k A_k^\intercal + Q_k$
8: **return** $\hat{x}_k$

Chapter 6

Results

This chapter is dedicated to presenting the results obtained from the methodologies and models discussed in the previous chapters. The results are divided into two main sections. The first section explores yaw error estimation utilizing the Cartesian wind model in its process model. Firstly, the results from using the Kalman filter for state estimation are presented. Then the effectiveness of the Extended Kalman filter is demonstrated by utilizing the yaw error estimate for yaw control. The second section employs the Yaw-Velocity wind model in the process model for yaw error estimation with the Extended Kalman filter. Here, yaw control using the yaw error estimate is implemented and compared to moving average yaw control meant to represent yaw control methods currently in use.

6.1 Cartesian wind

This section presents yaw error estimation using the Kalman filter and the Extended Kalman filter utilizing the Cartesian wind model.

6.1.1 Kalman Filter

The Kalman filter is linearized around zero yaw error and a free-stream flow of 4 m/s. Contrary to the Extended Kalman filter, it was explained in Section 5.3 that the linearized Kalman filter does not update the state and output matrices A and C. This means that once the system strays further from the linearization point it is expected to diverge from the nonlinear process model. Furthermore, the Kalman filter operates in incremental states, which means that the states for which the Kalman filter was linearized about, need to be added to the incremental estimates to obtain the resulting estimate.

The resulting Observability Gramian is invertible when measuring the angular velocity of the rotor Ω and the torques $\boldsymbol{\mathcal{M}}$, meaning that only four measurements are needed for the system to be observable.

The Kalman filter is simulated using the resulting wind shown in Subsection 3.2.3. This wind signal has a mean component that agrees with the linearization, however, highly turbulent flow is imposed on it. The main results from this simulation will be presented and discussed in the subsequent sections.

Resulting Estimates.

The estimates presented in this section are compared to the real values of the simulation. Figures 6.1 and 6.2 show that the estimates of ξ_1 and ξ_2 stray from the real values when the system moves

further from the linearization point. At 40 seconds this is best showcased, where both ξ_1 and ξ_2 are noticeably different from the real values. The yaw error depicted in Figure 6.3 shows the impact of this. This close to the operating point the Kalman filter produces a rough estimate. However, there is a noticeable difference already when the side wind is at 2 m/s and the normal wind is at 5 m/s, which does not bode well as real wind conditions can vary alot. The angular velocity suffers from the same nonlinearity. The internal variables $\boldsymbol{\sigma}$ suffer the most, as it can be seen that nonlinearities in the real system lead to inadequate estimates.

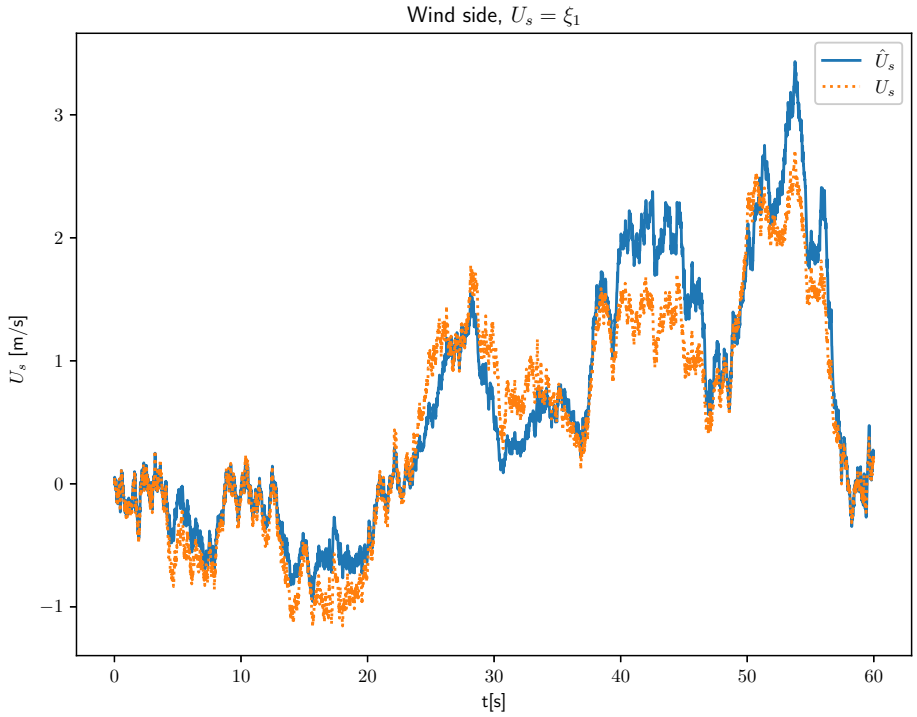


Figure 6.1: Kalman filter estimate of ξ_1.

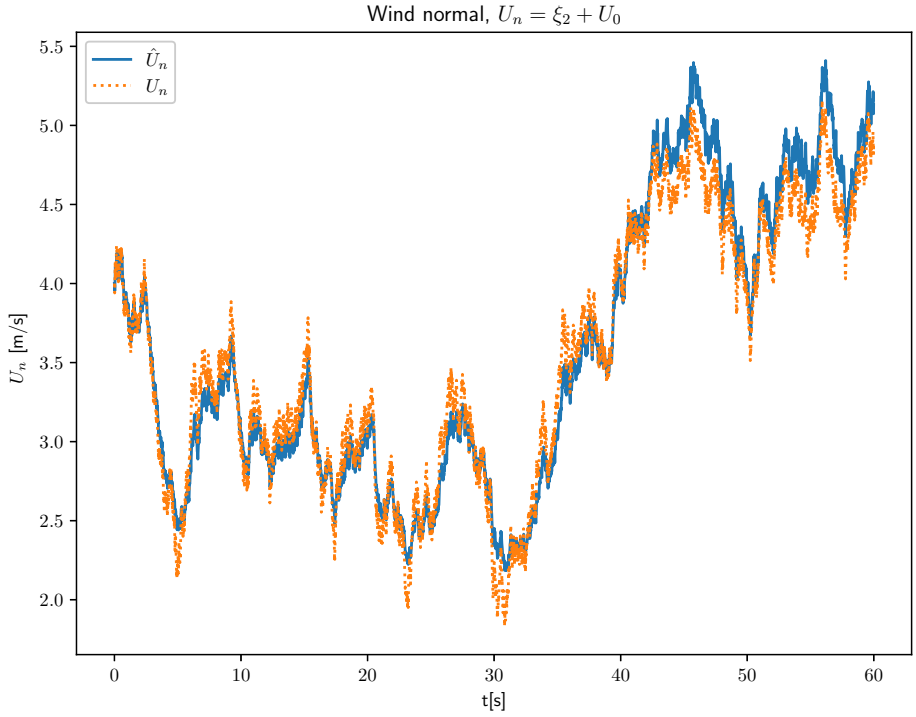


Figure 6.2: Kalman filter estimate of $\xi_2 + U_0$ with $U_0 = 4$ m/s.

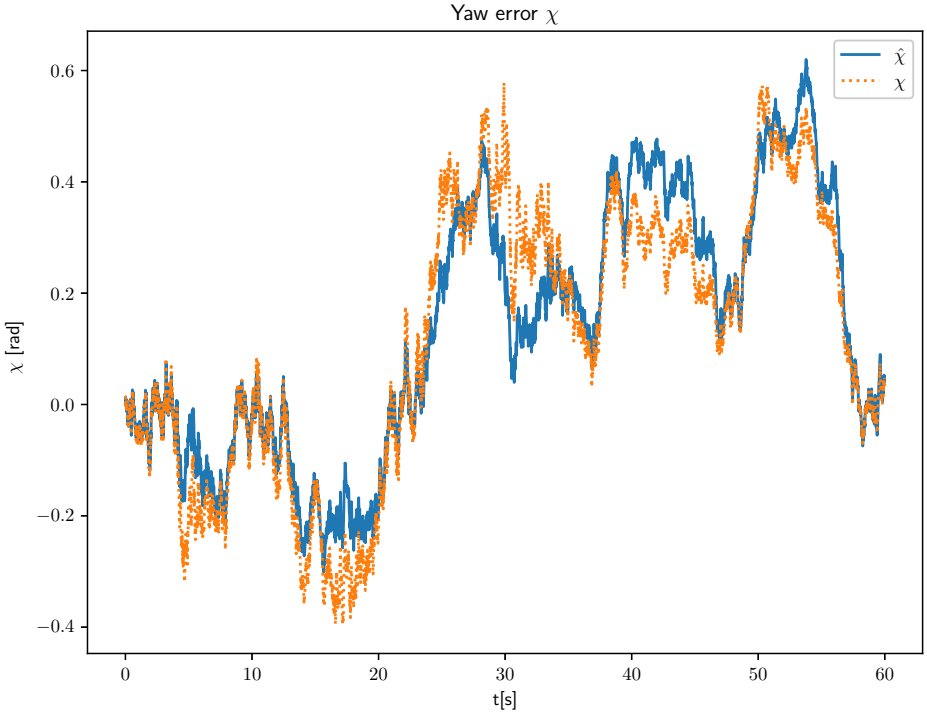


Figure 6.3: Kalman filter estimate of the yaw error χ.

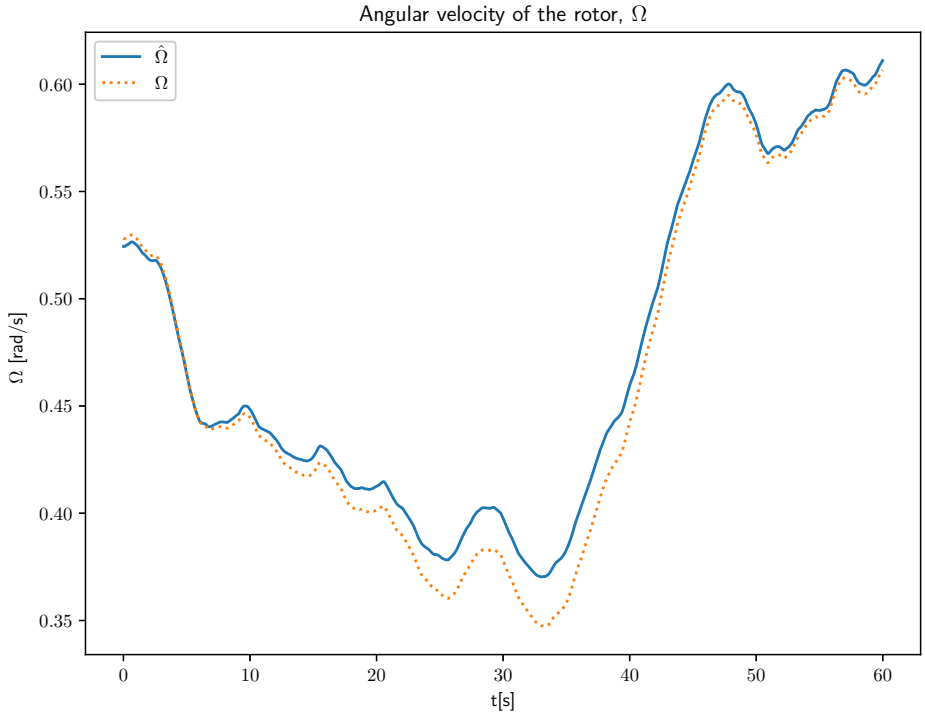


Figure 6.4: Kalman filter estimate of the angular velocity of the rotor Ω.

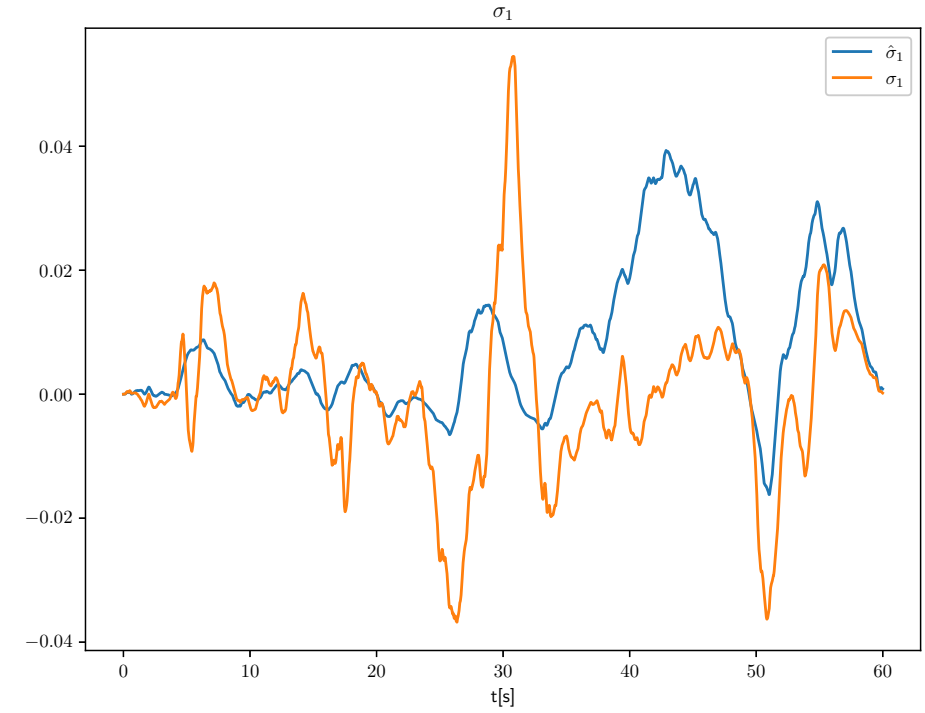


Figure 6.5: Kalman filter estimate of σ_1.

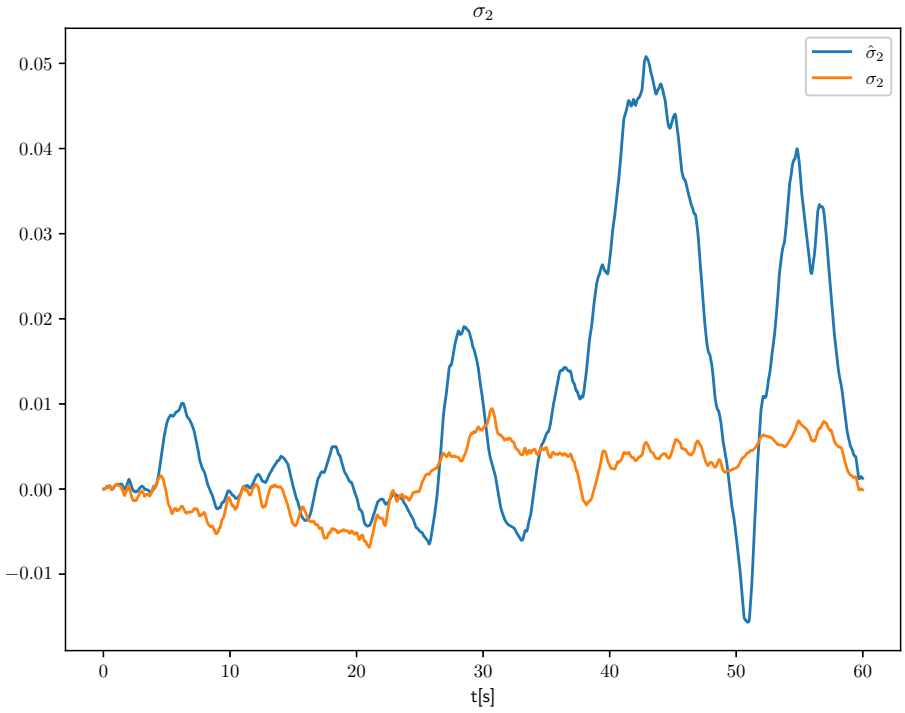


Figure 6.6: Kalman filter estimate of σ_2.

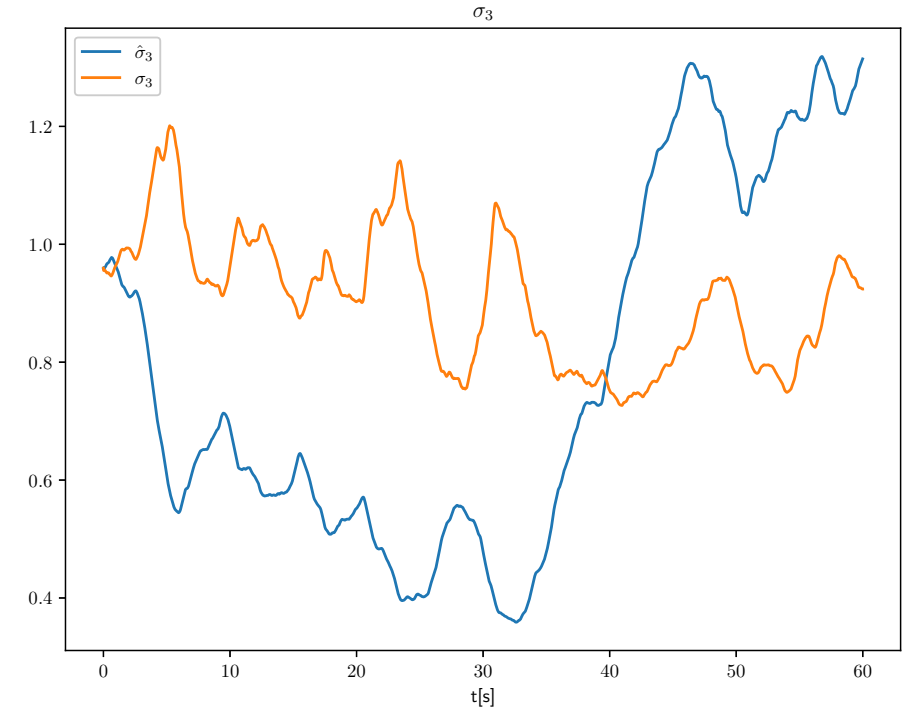


Figure 6.7: Kalman filter estimate of σ_3.

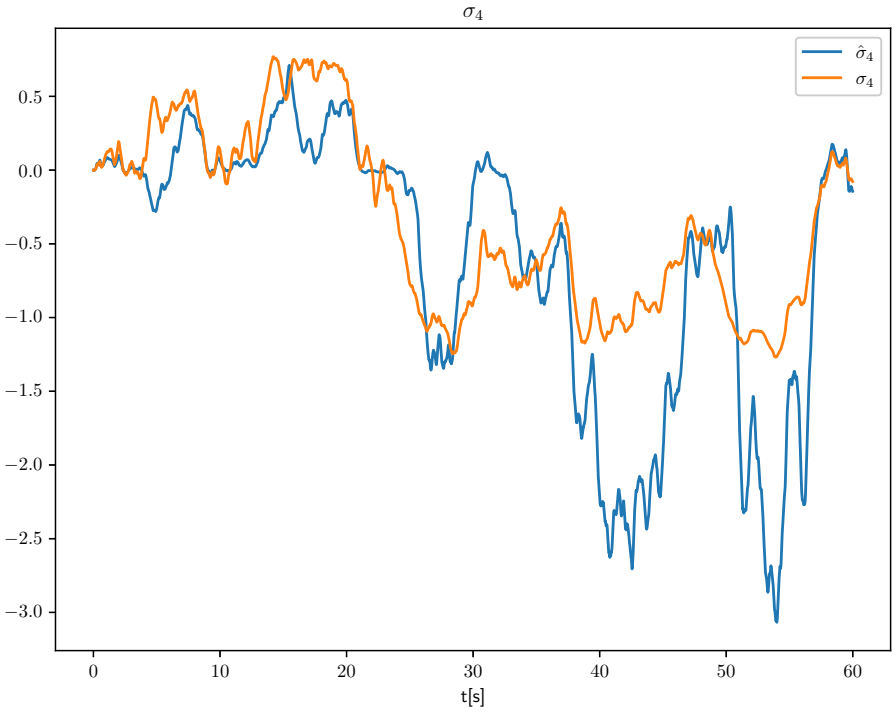


Figure 6.8: Kalman filter estimate of σ_4.

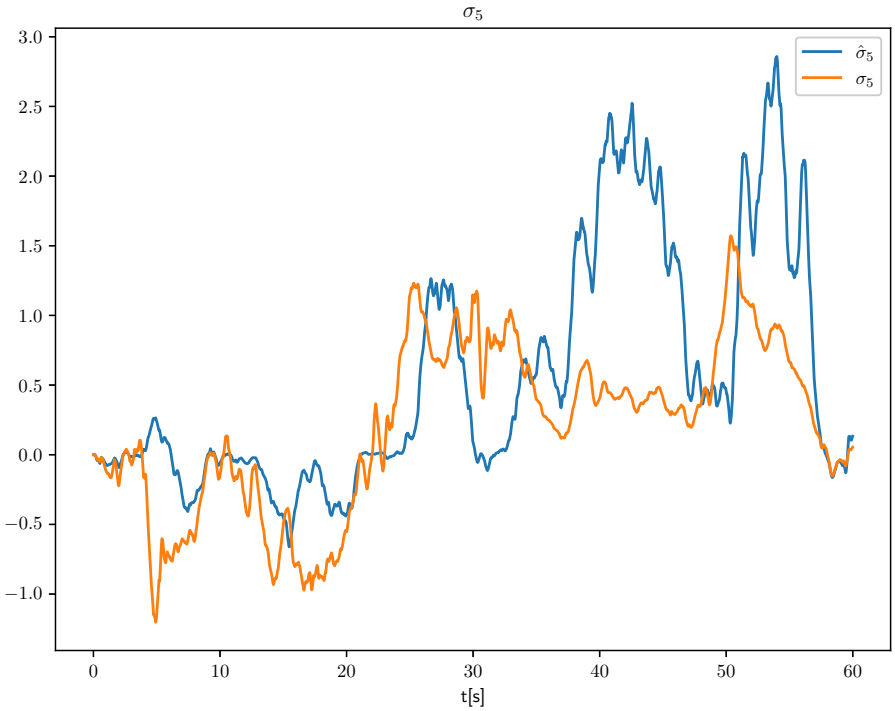


Figure 6.9: Kalman filter estimate of σ_5.

6.1.2 Extended Kalman Filter with Yaw Control

The Kalman filter did not produce satisfactory estimates. A Cartesian wind signal with a mean wind from the North of 4m/s and 1.2m/s from East is employed to demonstrate that the Extended Kalman filter captures the nonlinearities of the model better than the Kalman filter even when it is further from the initial linearization point. The turbulent flow of this wind can then be described by:

$$\begin{aligned} \mathcal{K} &= \sigma_{\mathbf{K}} \sqrt{\frac{19L}{40\bar{U_\infty}}} = 51.42 \\ T &= \frac{L}{\bar{U_\infty}} = 15.42 \\ \begin{bmatrix} T & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} &= - \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} + \begin{bmatrix} \mathcal{K} & 0 \\ 0 & \mathcal{K} \end{bmatrix} \begin{bmatrix} \mathrm{w}_1 \\ \mathrm{w}_2 \end{bmatrix} \end{aligned} \tag{6.1}$$

The resulting North and East wind, free-stream flow, and Yaw of the wind is presented in the following figures:

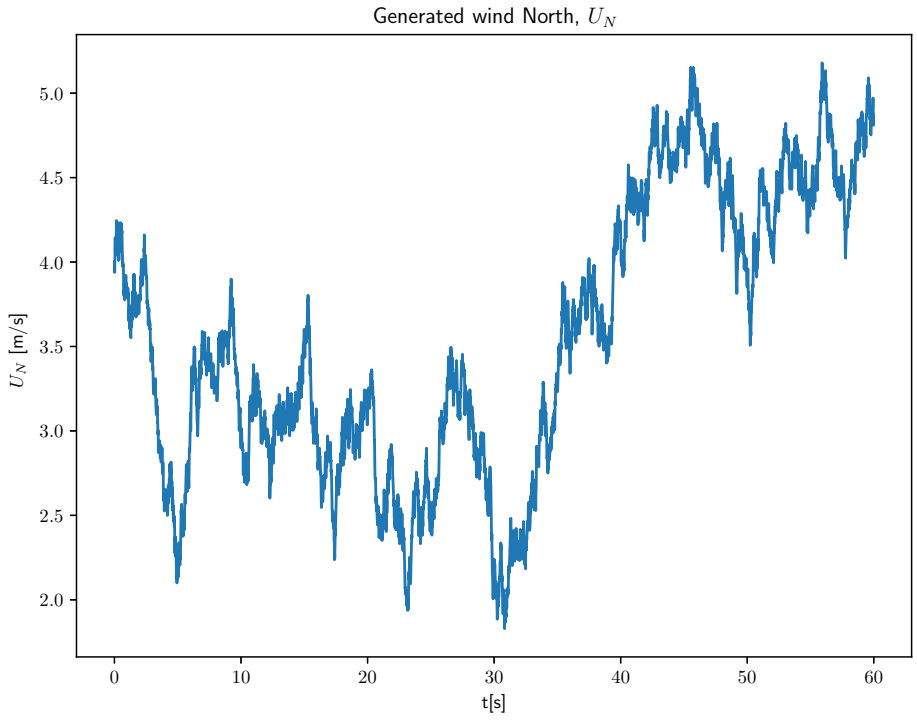


Figure 6.10: North wind generated.

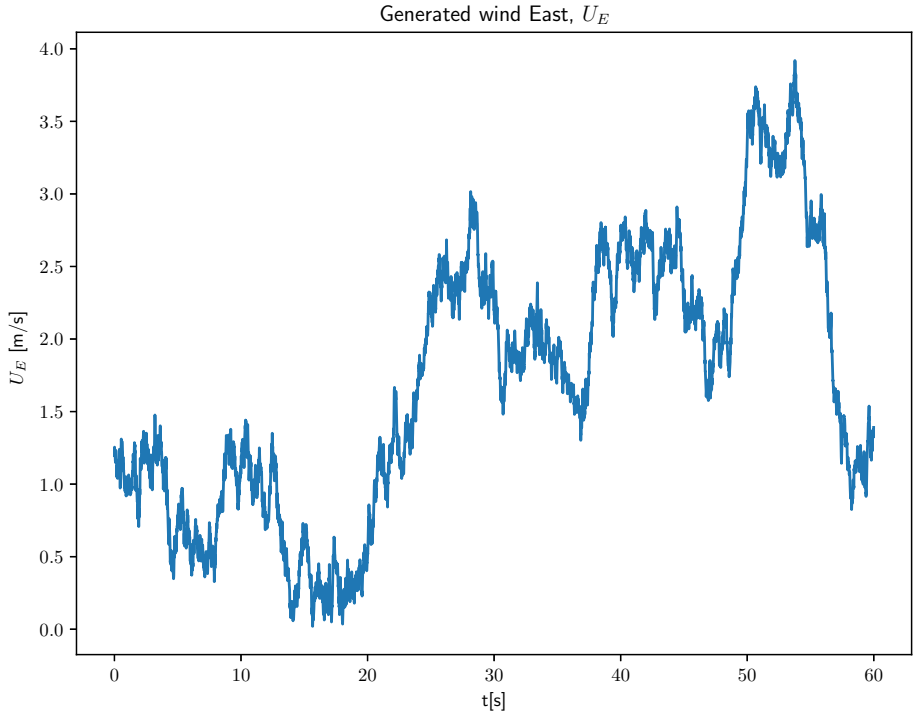


Figure 6.11: East wind generated.

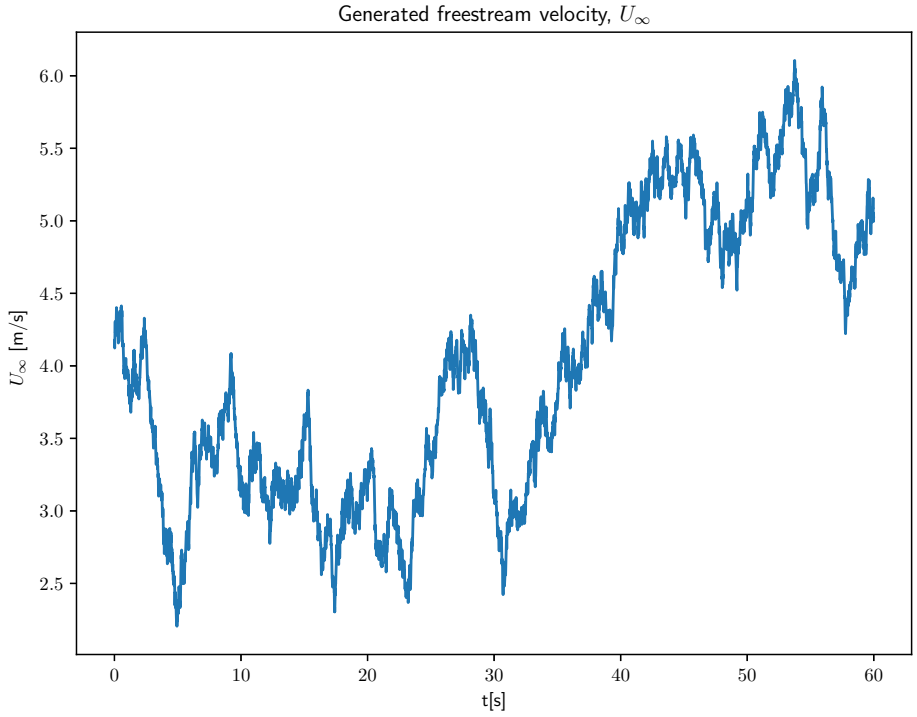


Figure 6.12: Free-stream flow U_∞

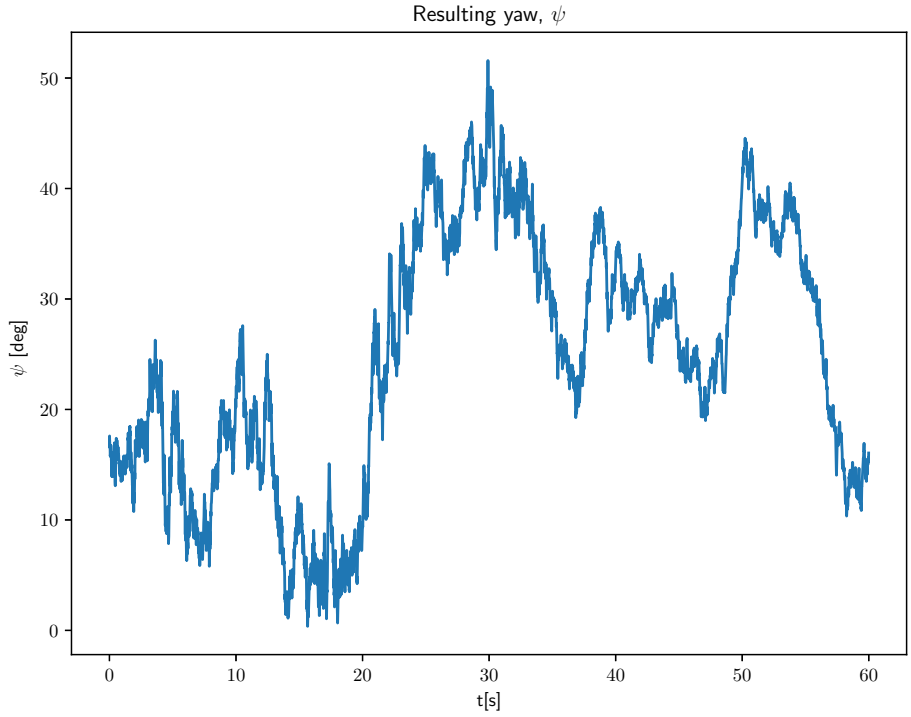


Figure 6.13: Resulting Yaw ψ.

This generated wind is employed on the Extended Kalman filter to demonstrate its effectiveness.

Implementing Yaw control

Yaw control utilizing the yaw error estimate is introduced to show an application of the yaw error estimate. The yaw control is implemented as a simple P-regulator, to exemplify that a good yaw error estimate does not require advanced schemes to work initially.

$$\dot{\theta} = 0.1\,\hat{\chi} \tag{6.2}$$

As it can be seen from the equation, a simple P-regulator with a constant gain is implemented to make the wind turbine yaw such that it aligns with the wind. As mentioned in Section 2.6, the maximum yaw rate of the wind turbine is 0.3 degrees/s. This value is also set as the rate limit in the simulation.

Resulting Estimates.

The estimates presented in this section are compared to the real values of the simulation. Contrary to the Kalman filter, it can be seen that accurate estimates are produced even if the state is far from the initial linearization point.

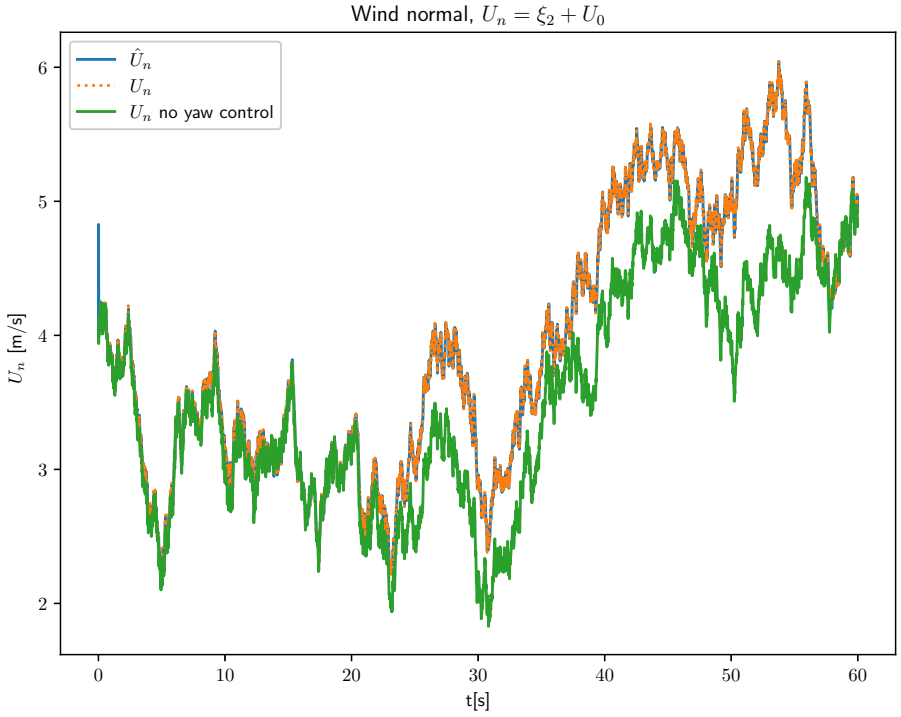


Figure 6.14: Estimate of ξ_2, where $\hat{U}_n = \hat{\xi}_2 + 4$.

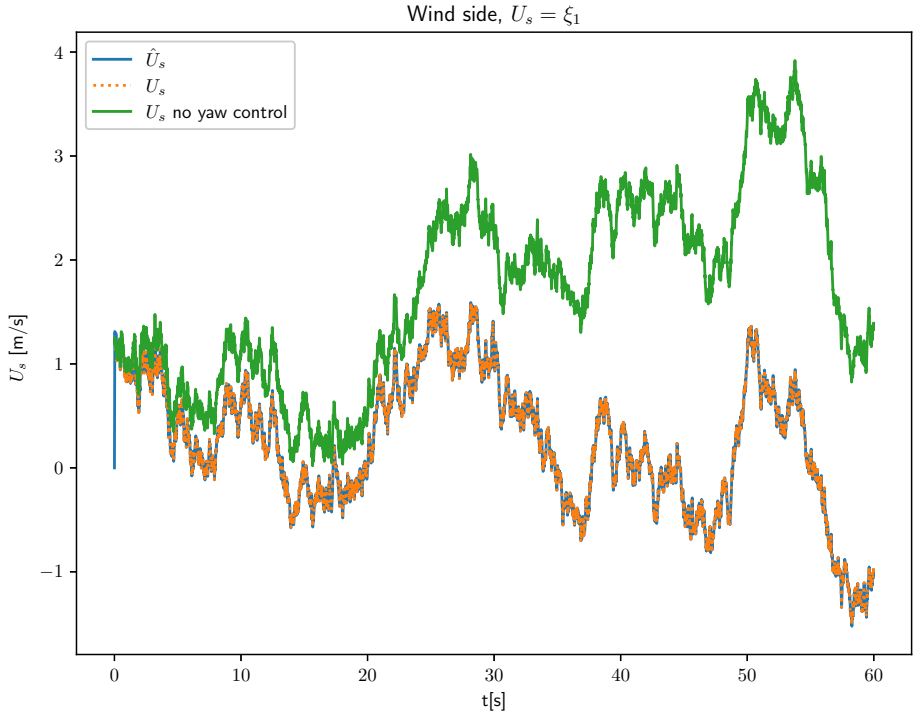


Figure 6.15: Estimate of ξ_1, where $\hat{U}_s = \hat{\xi}_1 + 1.2$

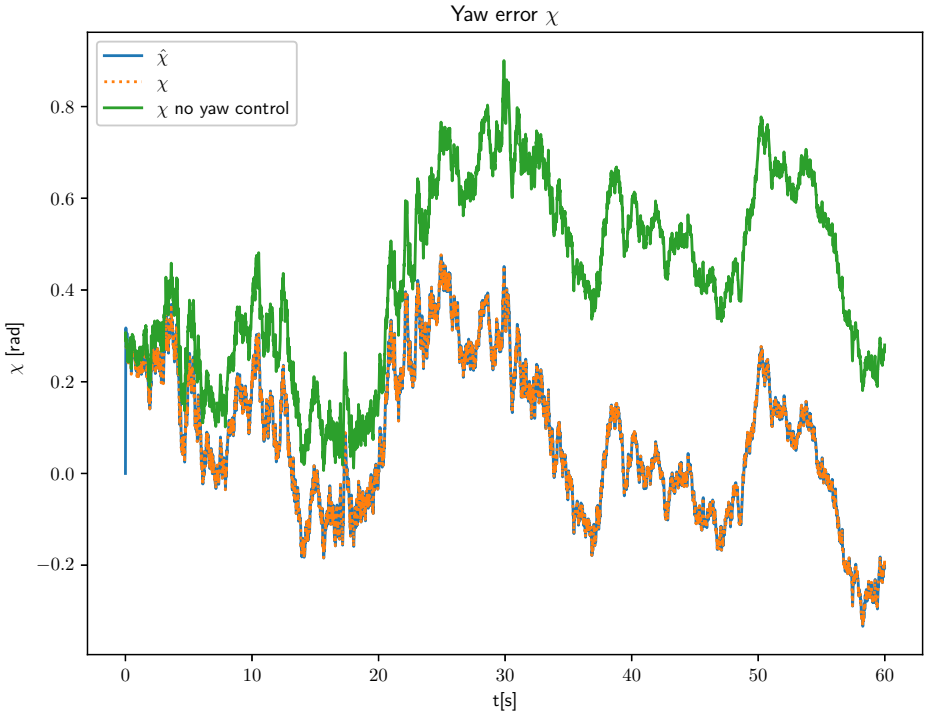


Figure 6.16: Estimate of yaw error.

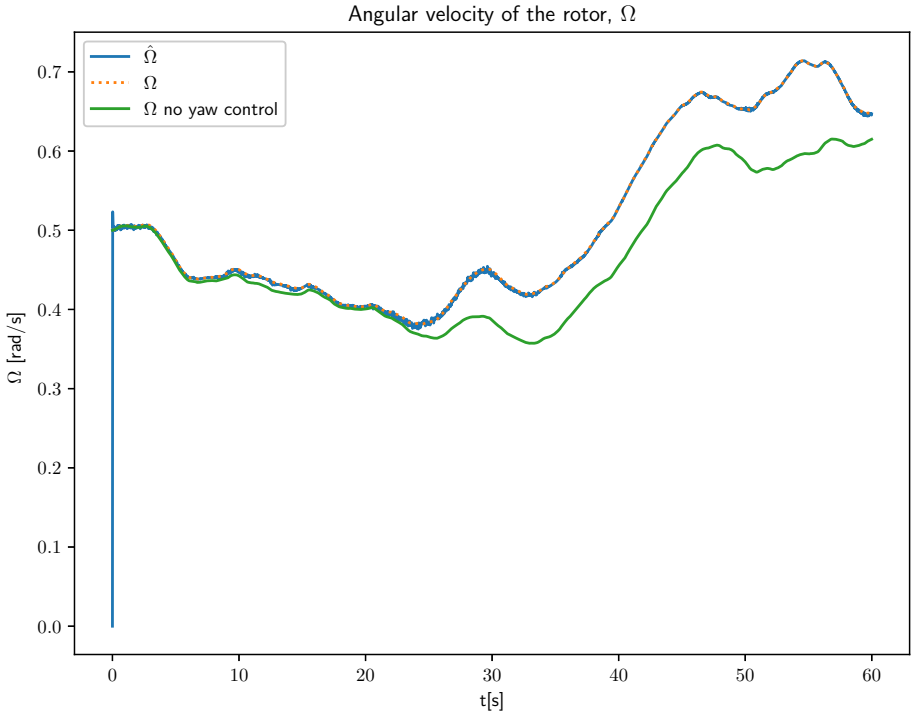


Figure 6.17: Estimated angular velocity.

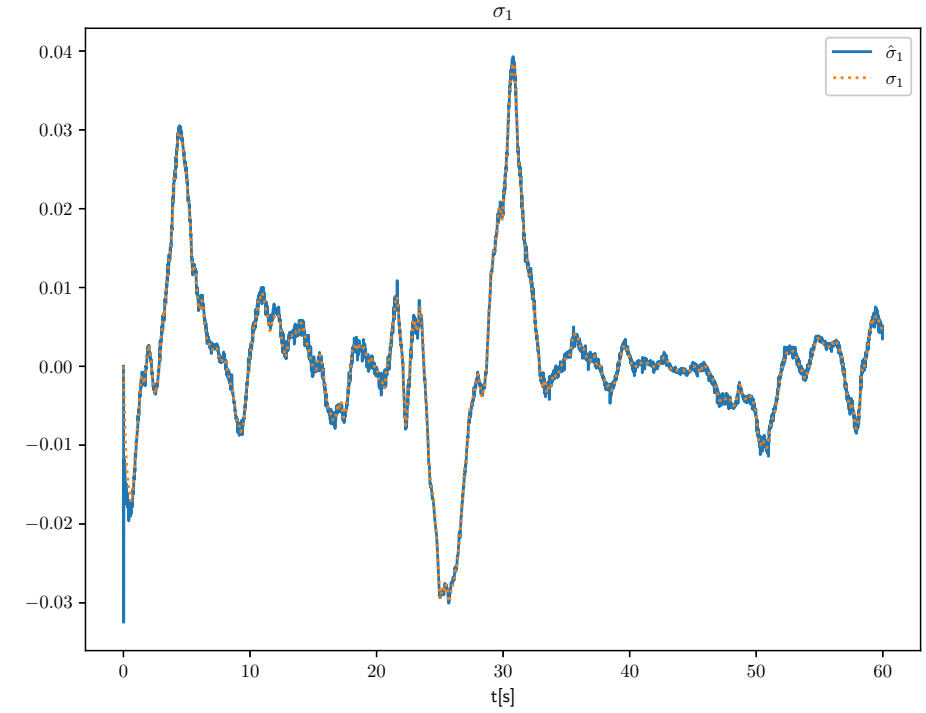


Figure 6.18: Estimate of σ_1.

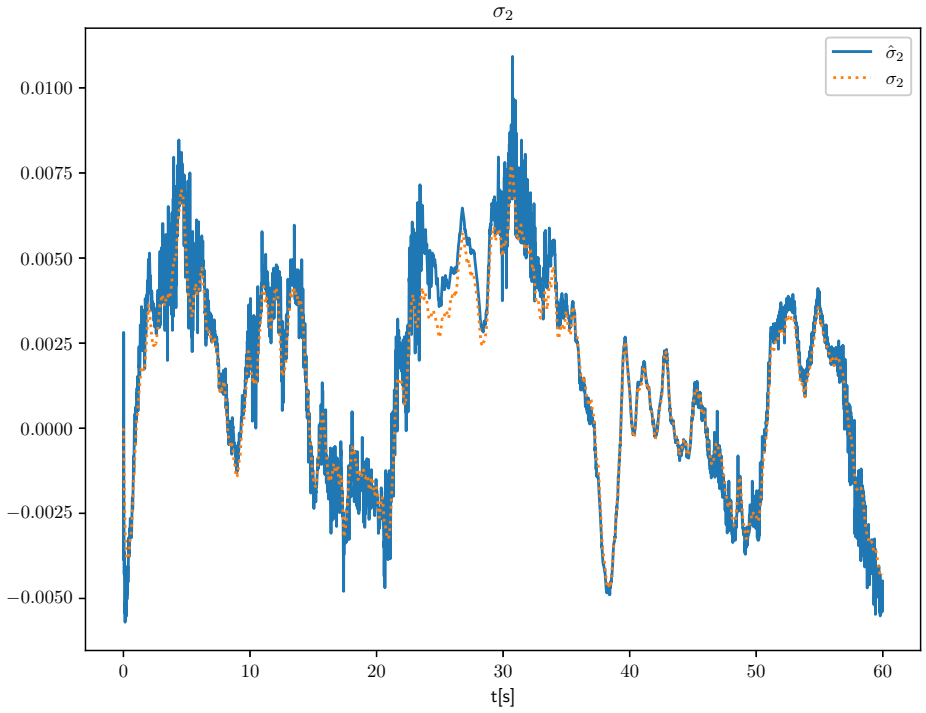


Figure 6.19: Estimate of σ_2.

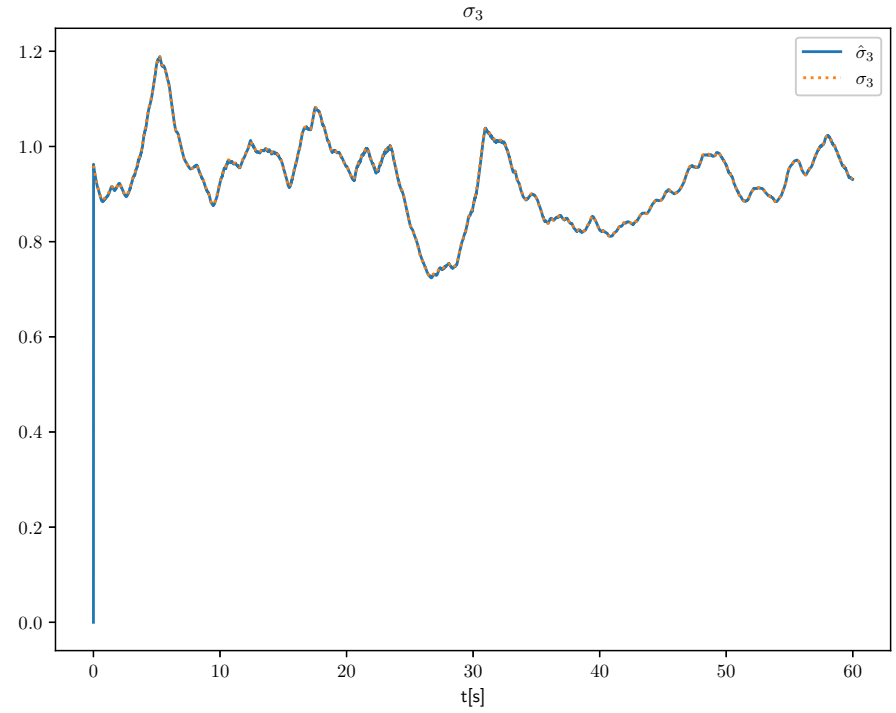


Figure 6.20: Estimate of σ_3.

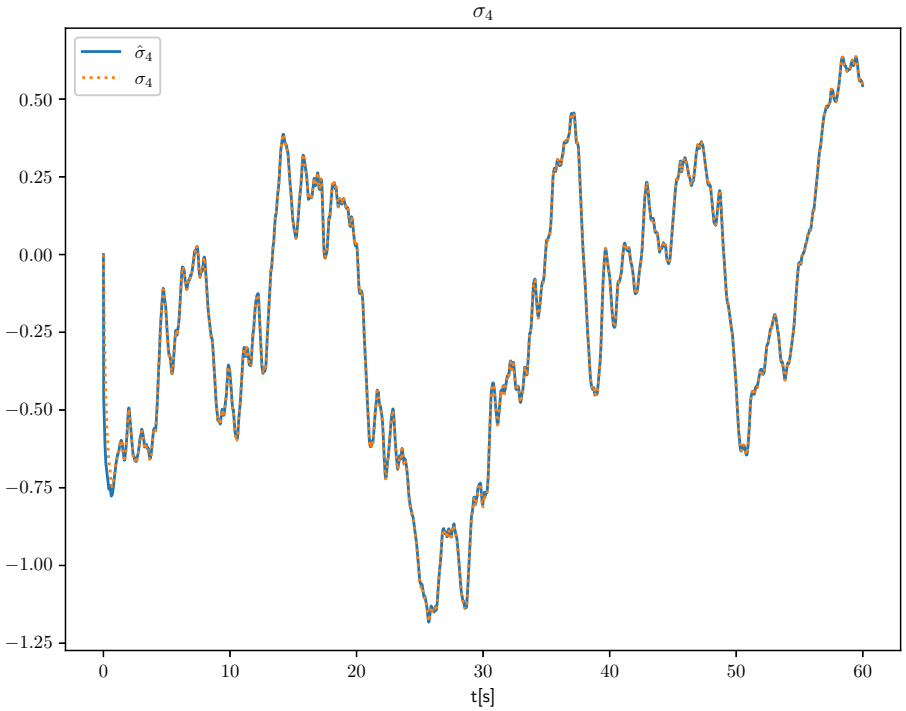


Figure 6.21: Estimate of σ_4.

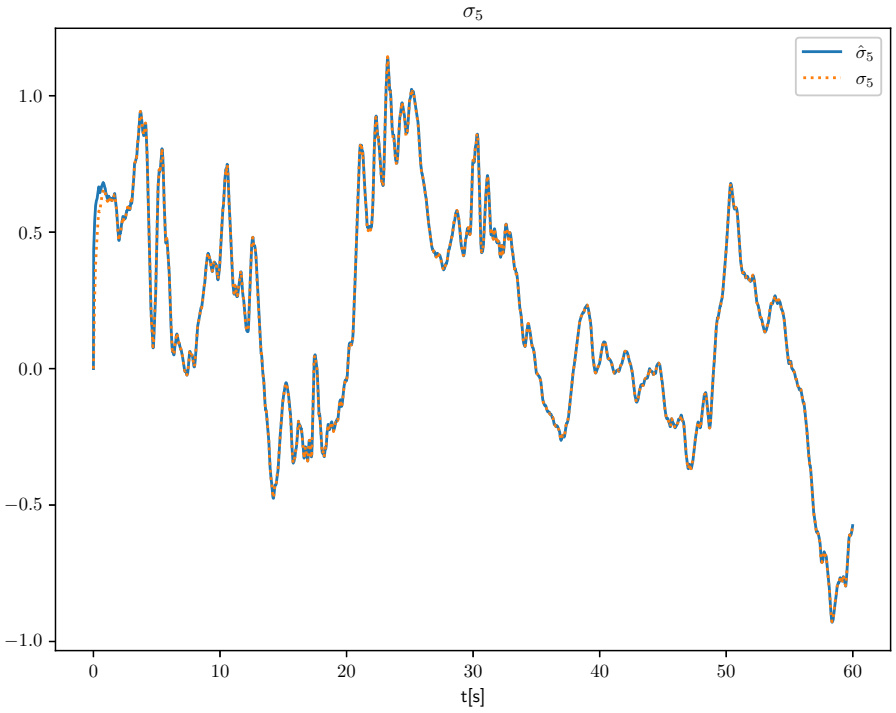


Figure 6.22: Estimate of σ_5.

Evaluating the Effects of Introducing Yaw Control.

The yaw control shown in (6.2) is implemented to align the wind turbine with the wind. The effects of introducing yaw control are shown in Figures 6.14, 6.15, 6.16, and 6.17. The green plot shows the values of a simulation without yaw control exposed to the same conditions. It can be observed that the yaw controller reduces the yaw error and, as a side effect, increases the angular velocity and normal wind while decreasing the side wind.

Ideally, for zero yaw error, the normal wind will be the same as the free-stream flow. Figure 6.23 shows how the yaw controls "lift" the normal wind of the wind turbine towards the free-stream flow. This is achieved by yawing the wind turbine into the wind, the resulting yaw of the wind turbine is shown in Figure 6.24.

The implication of a smaller yaw error is that more power can be extracted from the same conditions. This is evident in the Maximum Power Point Tracking (MPPT) control law from Definition 2.9, which shows that the rotor's angular velocity is cubically related to the power extracted. Gaining extra angular velocity under the same conditions and control law can therefore significantly improve power extraction. This is reflected in Figure 6.25, where the generated power in both cases is shown. The resulting power gained from yaw control under these conditions is shown in Figures 6.26 and 6.27.

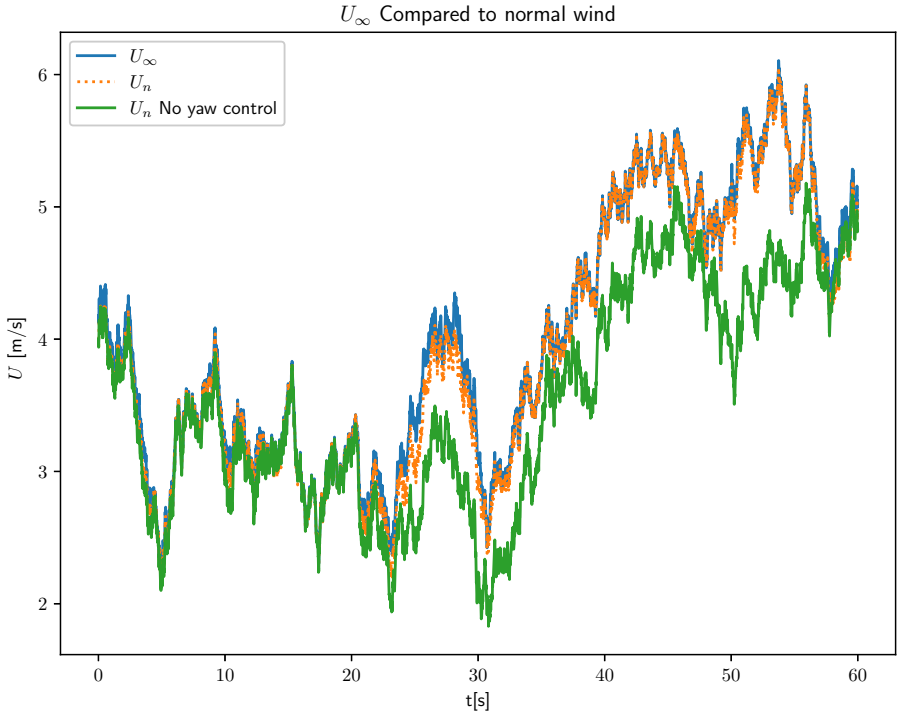


Figure 6.23: Free-stream flow compared to wind normal to the wind turbine.

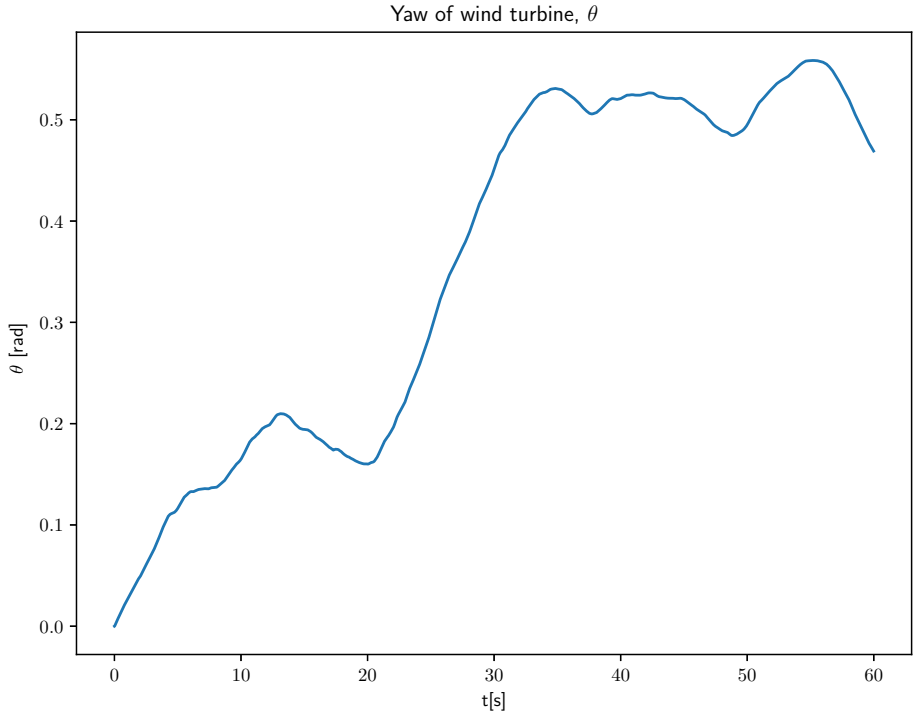


Figure 6.24: Yaw of the wind turbine from yaw control.

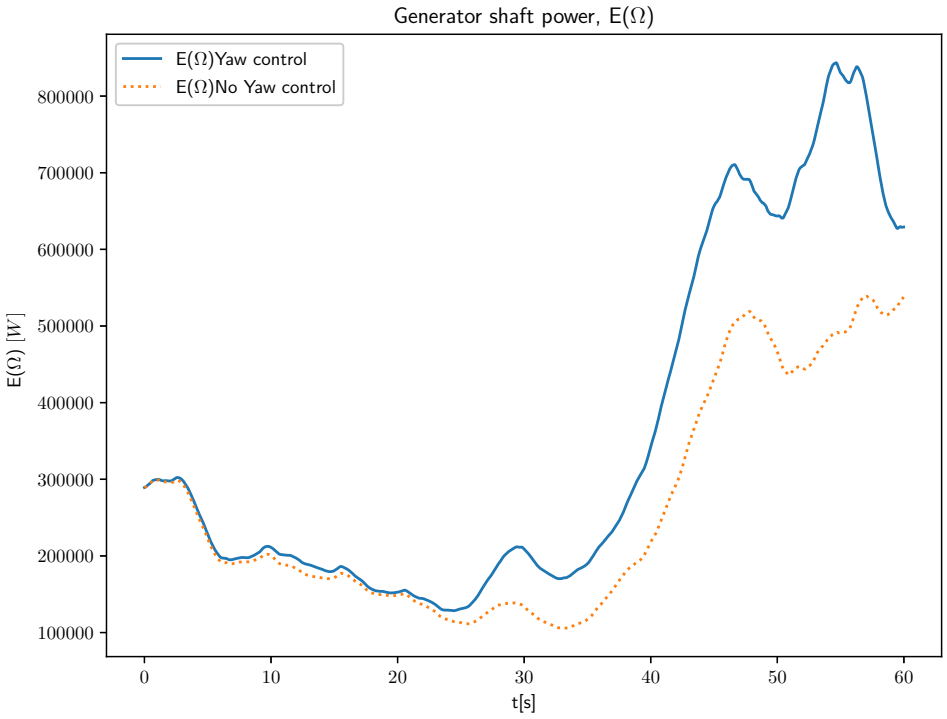


Figure 6.25: Power extraction yaw control vs no yaw control.

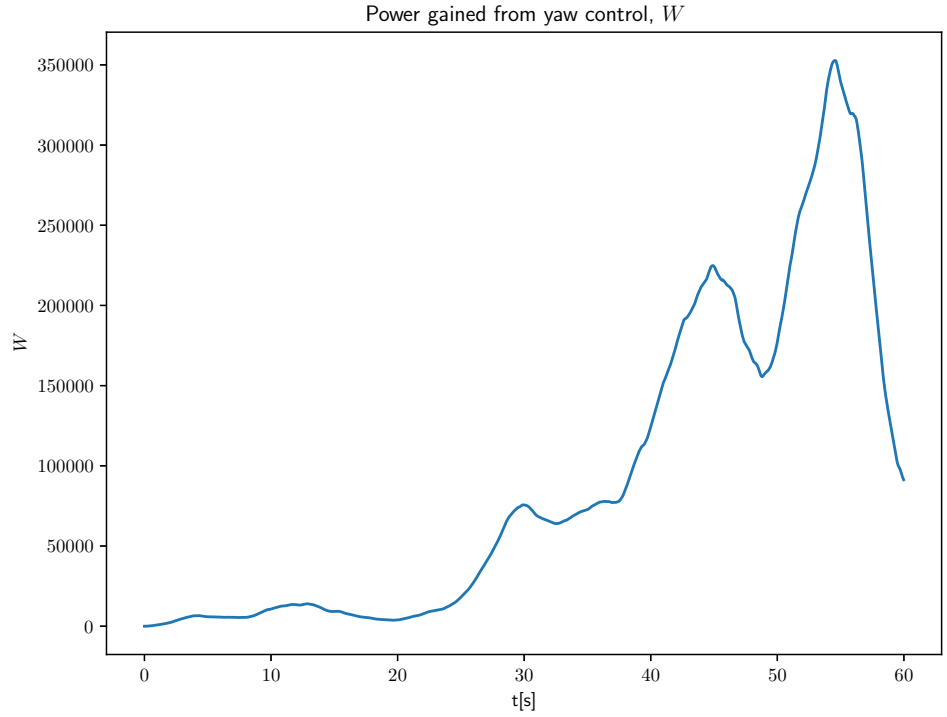


Figure 6.26: Power extraction gain W.

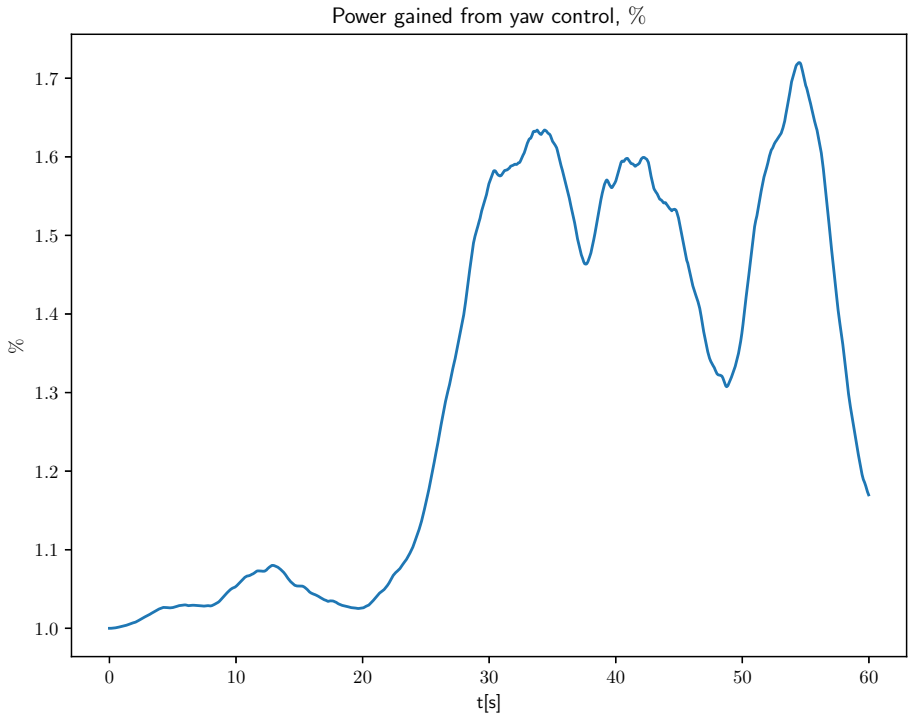


Figure 6.27: Power extraction gain, where 1.0 represents 100% of the power extracted without yaw control.

6.2 Yaw-Velocity

In this section, the Extended Kalman filter is employed on the system utilizing the Yaw-Velocity model to estimate yaw error. Similar to the previous section, this estimate is used in yaw control to steer the wind turbine into the wind. In addition to this, the following section provides a comparison between this yaw control and presentable strategies employed in a real wind turbine.

The resulting Observability Gramian is invertible when measuring the angular velocity of the rotor θ and the torques $\boldsymbol{\mathcal{M}}$, however, the full force-torque vector $\boldsymbol{\tau}$ is utilized in the simulations in this section. Note that no measurement noise was added to the measurements, however, introducing measurements can be resolved by re-tuning the measurement noise covariance matrix R.

Yaw control in wind turbines is commonly implemented using a threshold-based algorithm, where the turbine yaws whenever the moving average yaw error over a certain time period reaches a defined threshold [66]. The threshold can typically range from 7 to 15 degrees yaw error and the time period is often 10 minutes[15, 67]. This approach is partly adopted because yaw error measurements, usually obtained through wind vanes, are often heavily filtered over a given time period [68].

It's worth noting that frequent yaw adjustments can cause unnecessary wear and tear on the turbine [9]. On the other hand, infrequent yaw can lead to forced shutdowns and unnecessary loads on the blades, also leading to wear and tear [69]. However, finding this balance is beyond the scope of this thesis.

The yaw control in this thesis utilizes a simple P-regulator driven by the yaw error estimate as in (6.2). This approach is compared to a generous representation of the existing yaw controllers. Two separate yaw controllers representing existing approaches are employed, they utilize perfect yaw error measurements to correct for the moving average yaw error. The two scenarios are 5 and 10 minute control intervals. The 5 minute control interval corrects for the 5 minute moving

average yaw error every 5 minutes, whereas the 10 minute control does the same with 10 minute intervals and a 10 minute moving average. This is provided to demonstrate the effectiveness of the yaw error estimation, with further applications left for future work.

The moving average yaw control is then:

Algorithm 3 Moving average yaw control

if $t \mod 300 == 0$ **then**
  $\theta_{ref5} \leftarrow$ 5 minute moving average of χ
end if
$\dot{\theta}_{5m} = \theta_{ref5} - \theta_{5m}$
if $t \mod 600 == 0$ **then**
  $\theta_{ref10} \leftarrow$ 10 minute moving average of χ
end if
$\dot{\theta}_{10m} = \theta_{ref_1} - \theta_{10m}$

Here θ_{5m} and θ_{10m} is the yaw of the wind turbine with the 5 and 10 minute moving average controls, while θ is the yaw of the wind turbine using the P-regulator in (6.2).

The first 30 minutes of the wind yaw and speed from Subsection 3.4.1 was used in the simulation to demonstrate the different yaw controls. The wind yaw and speed used in the simulation are reiterated below.

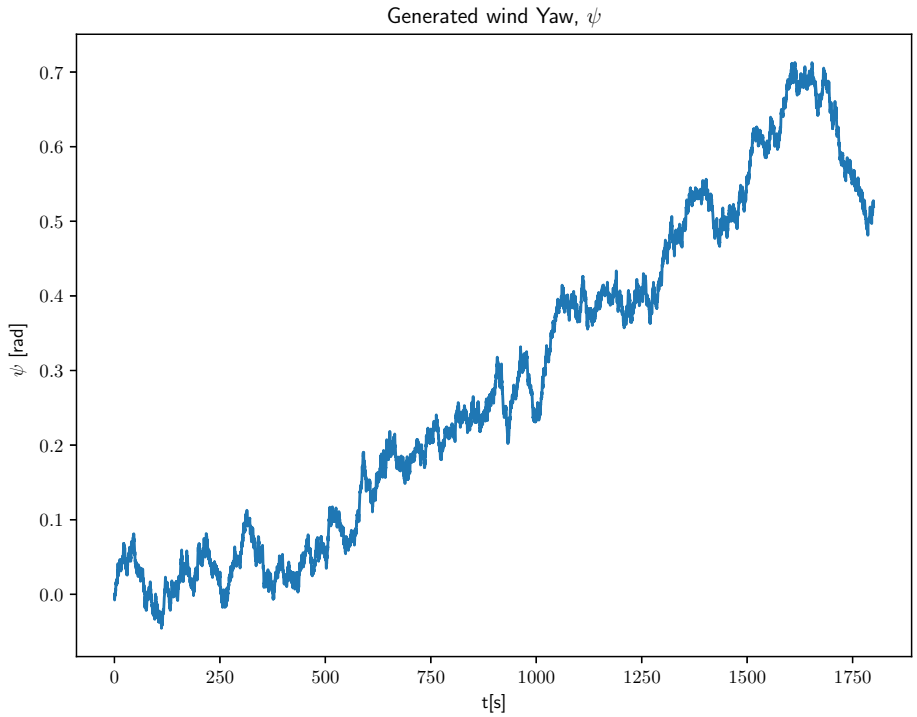


Figure 6.28: Generated yaw of the wind.

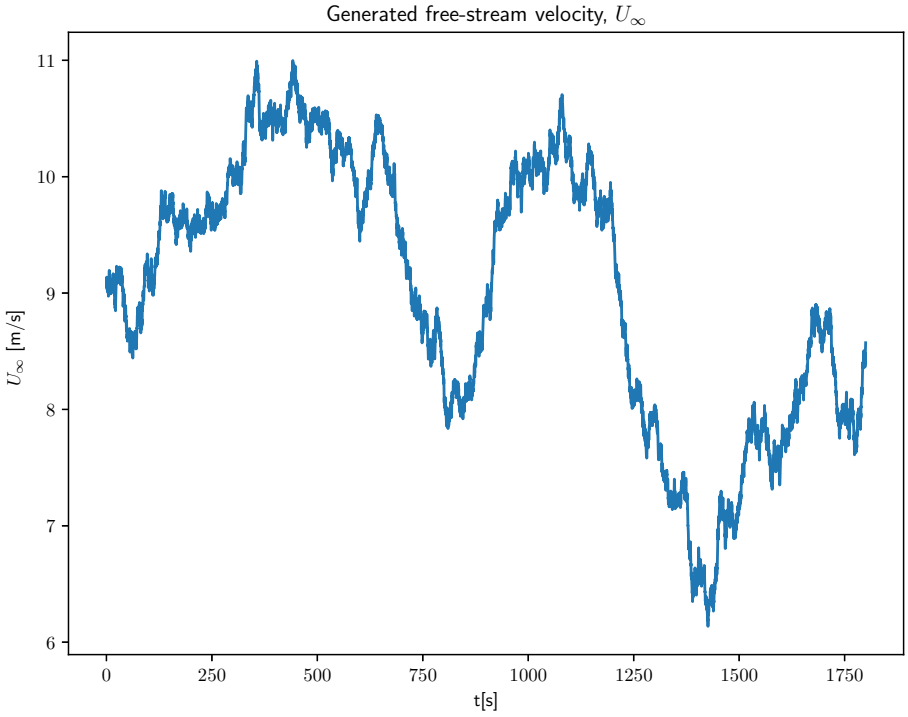


Figure 6.29: Generated wind speed.

6.2.1 Resulting Estimates.

The estimates presented in this section are compared to the real values of the simulation. It can be seen from the figures below that the Extended Kalman filter produces estimates that follow the real value very closely. This also leads to a good estimate of the yaw error.

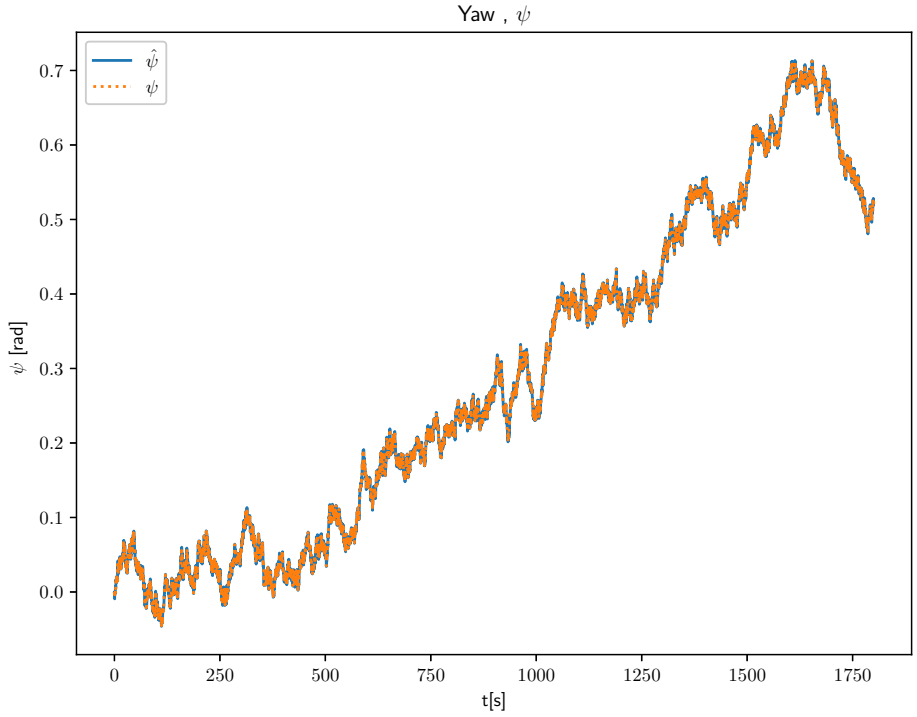


Figure 6.30: Estimated yaw angle of the wind ψ.

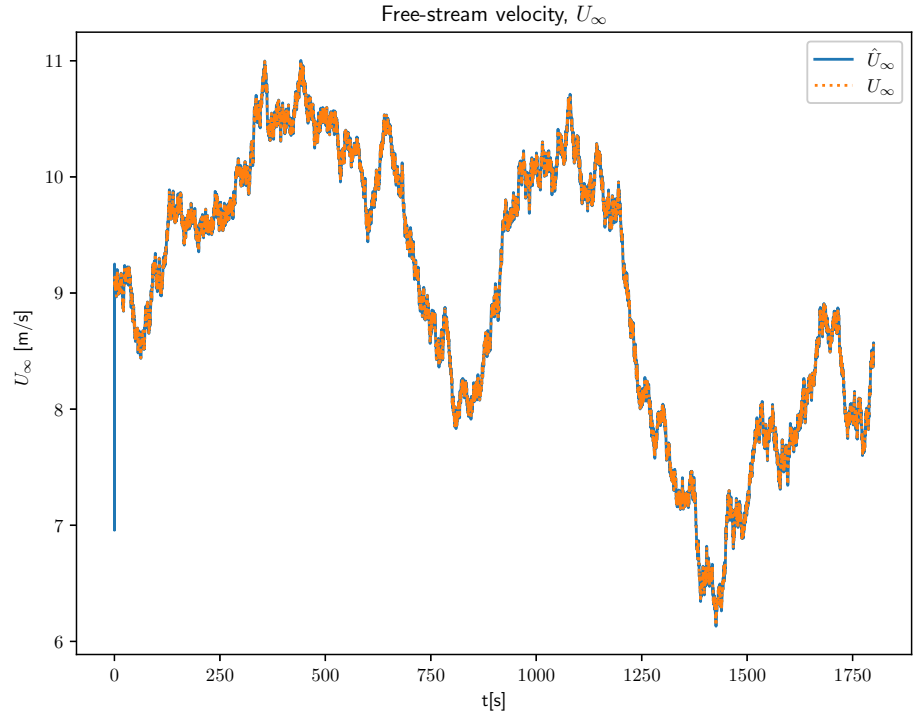


Figure 6.31: Estimated free-stream flow.

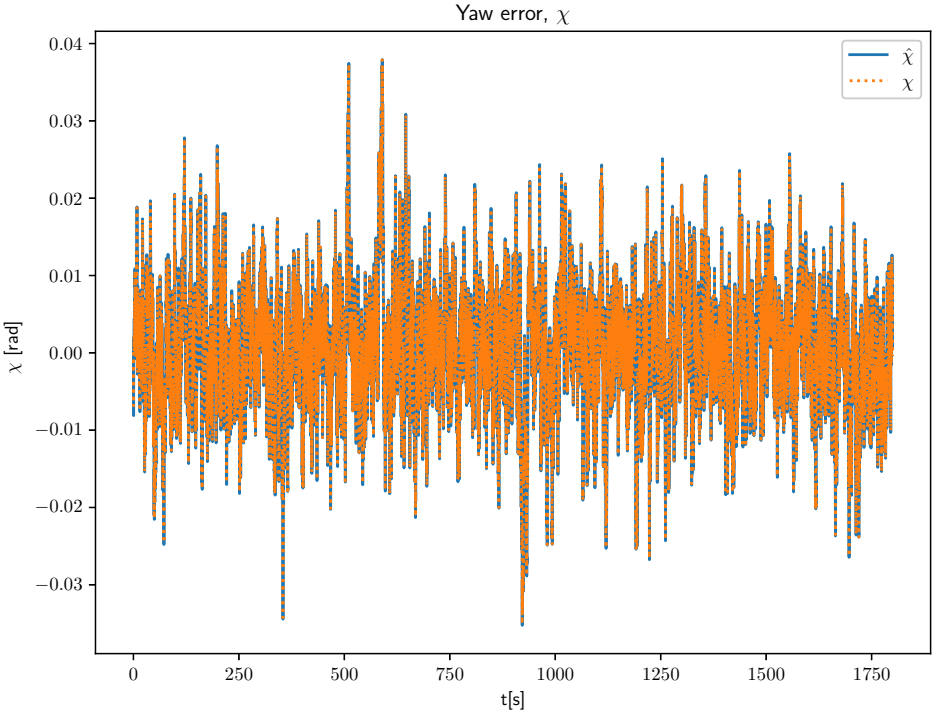


Figure 6.32: Estimated Yaw error.

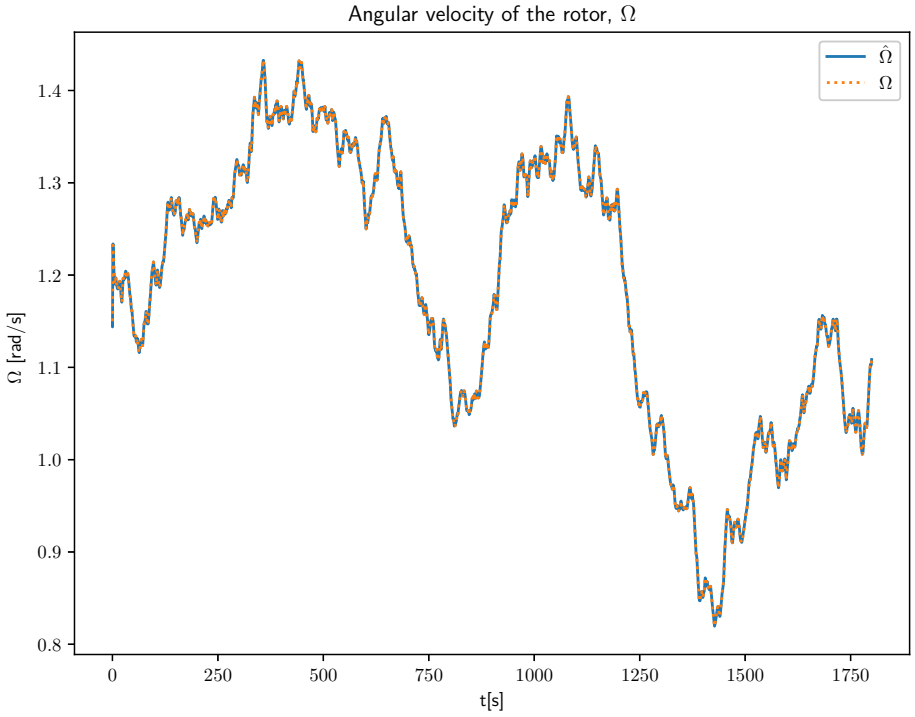


Figure 6.33: Estimated angular velocity.

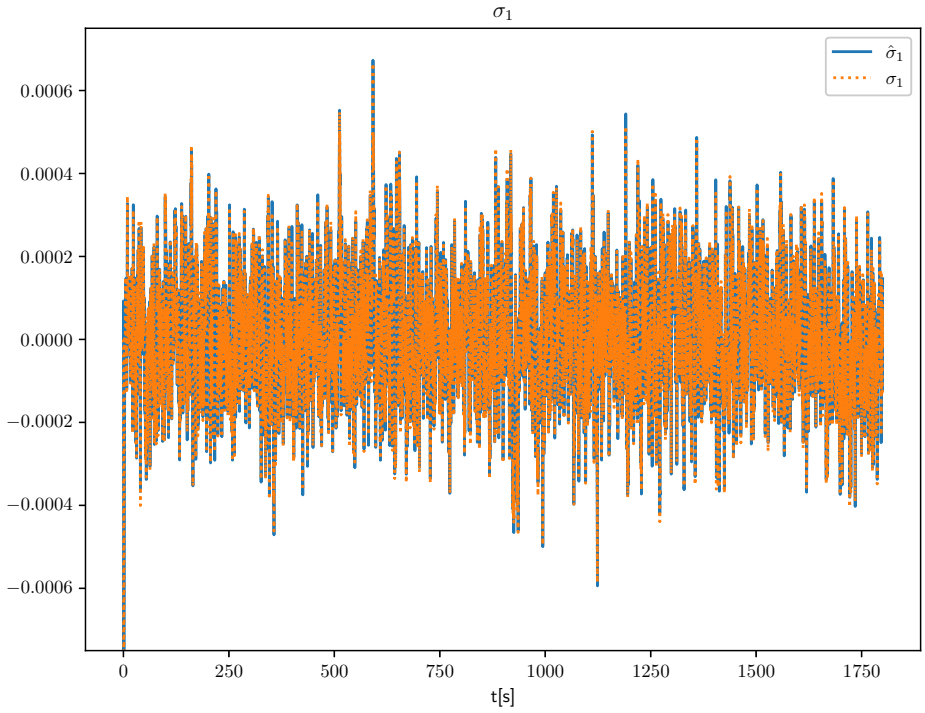


Figure 6.34: Estimate of σ_1.

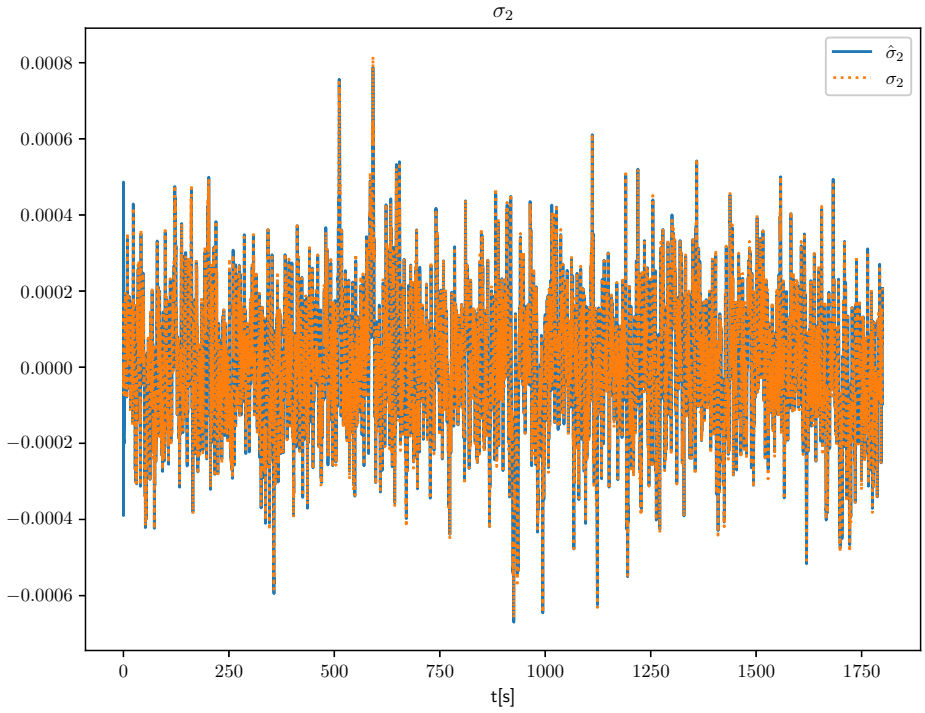


Figure 6.35: Estimate of σ_2.

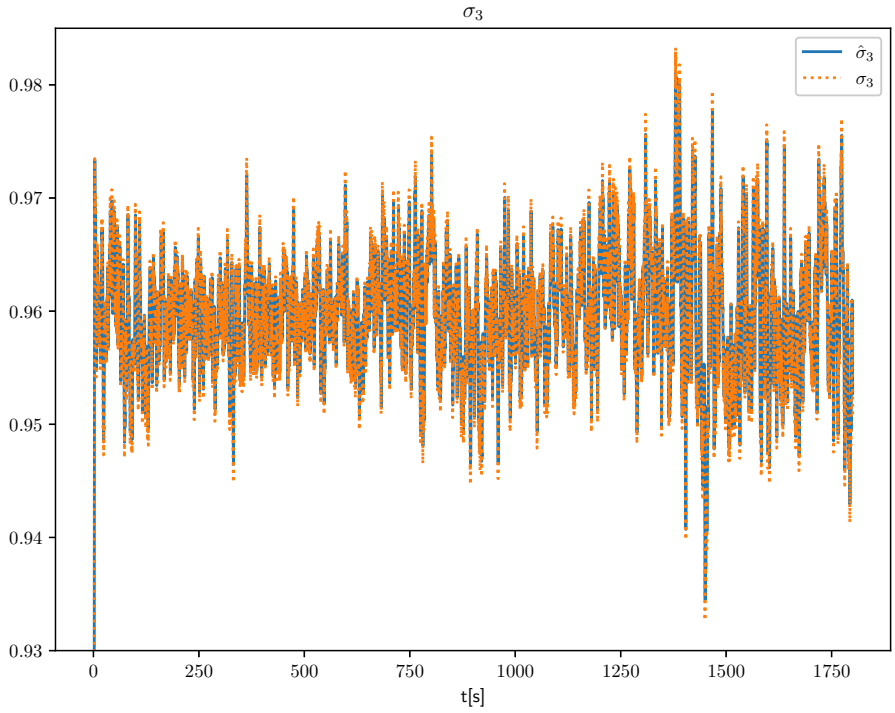


Figure 6.36: Estimate of σ_3.

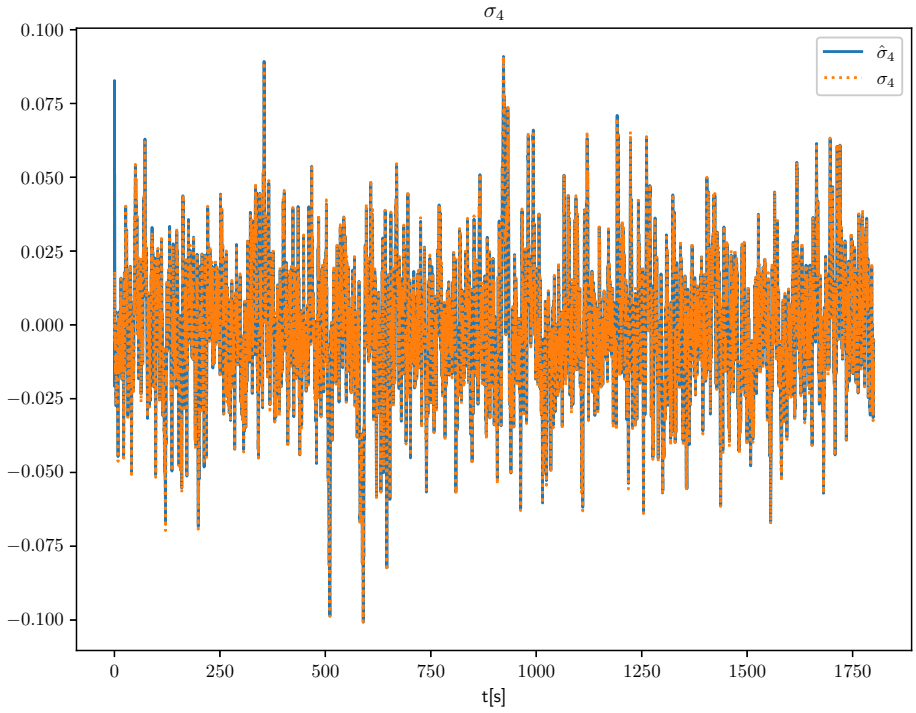


Figure 6.37: Estimate of σ_4.

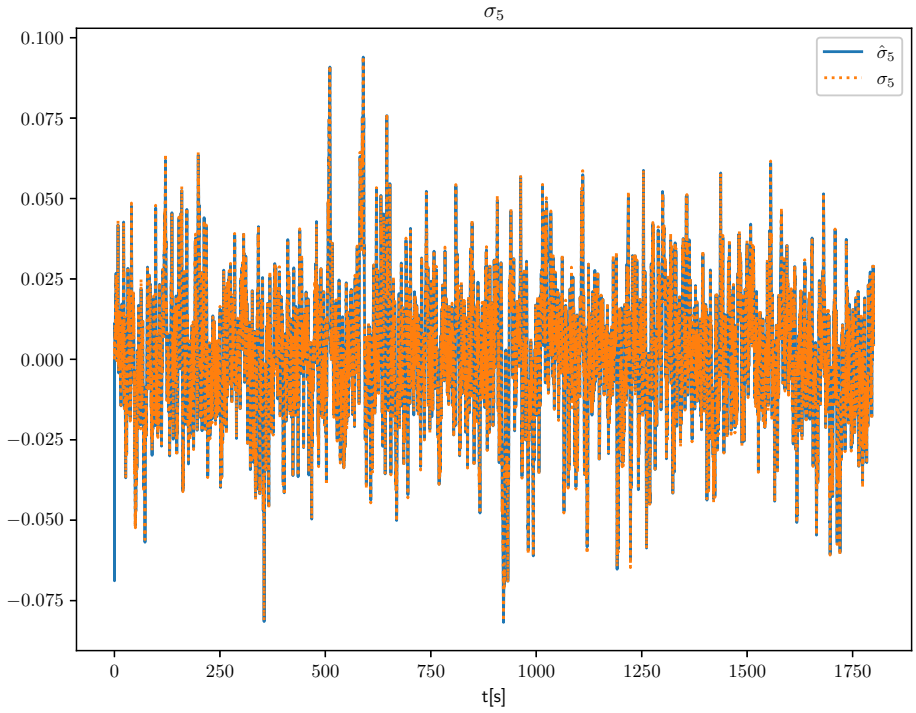


Figure 6.38: Estimate of σ_5.

6.2.2 Evaluating the Performance of Different Yaw Control Methods.

In order to compare the different yaw controls, three separate wind turbines were set up. Each turbine was subjected to the same initial conditions and wind signal, as shown in Figures 6.28 and 6.29. The P-regulator used the estimated yaw error $\hat{\chi}$ from Figure 6.32, while the 5 and 10 minute moving average controls used the actual yaw error χ. The estimated yaw error $\hat{\chi}$ was found to closely follow the actual yaw error χ, making the difference between them negligible.

The P-regulator performed near-continuous yaw actuation, while the moving average controls made one yaw correction every 5 and 10 minutes, respectively.

Figure 6.39 illustrates the yaw error for the three cases, as well as for an unactuated case where $\psi = \chi$. Figure 6.40 shows the wind turbine yaw for the three cases. Given the definition of yaw error as $\chi = \psi - \theta$, the ideal scenario would be $\theta = \psi$. As seen in Figure 6.39, the P-regulator achieves close to zero yaw error. This is reflected in Figure 6.40, where θ closely resembles ψ. The 5 and 10 minute yaw controls, on the other hand, struggled to keep up with the changes in wind conditions.

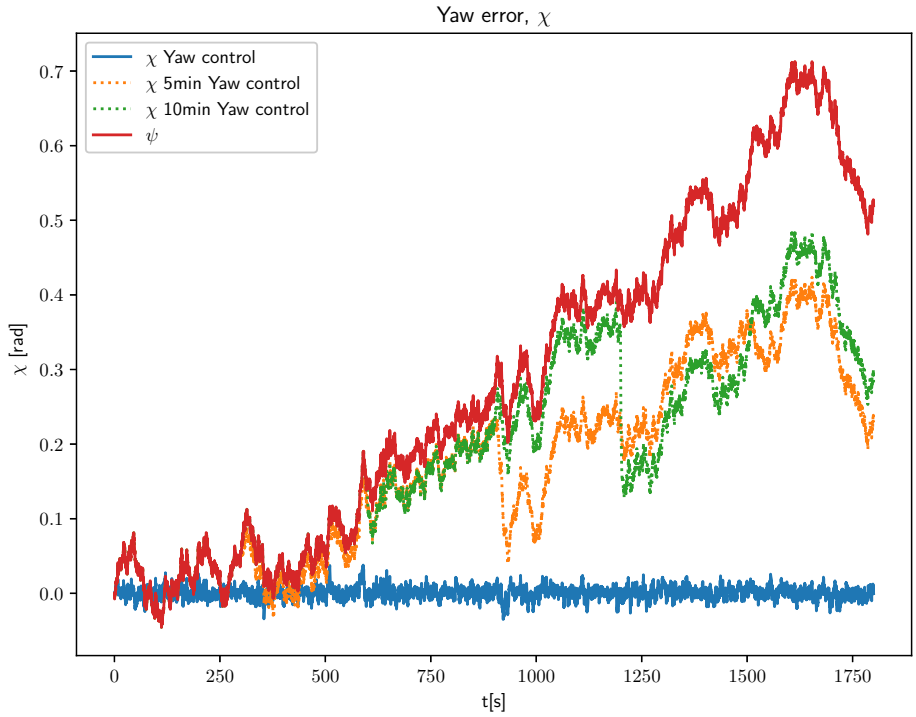


Figure 6.39: Comparison of yaw error across three cases, including the unactuated scenario where $\chi = \psi$.

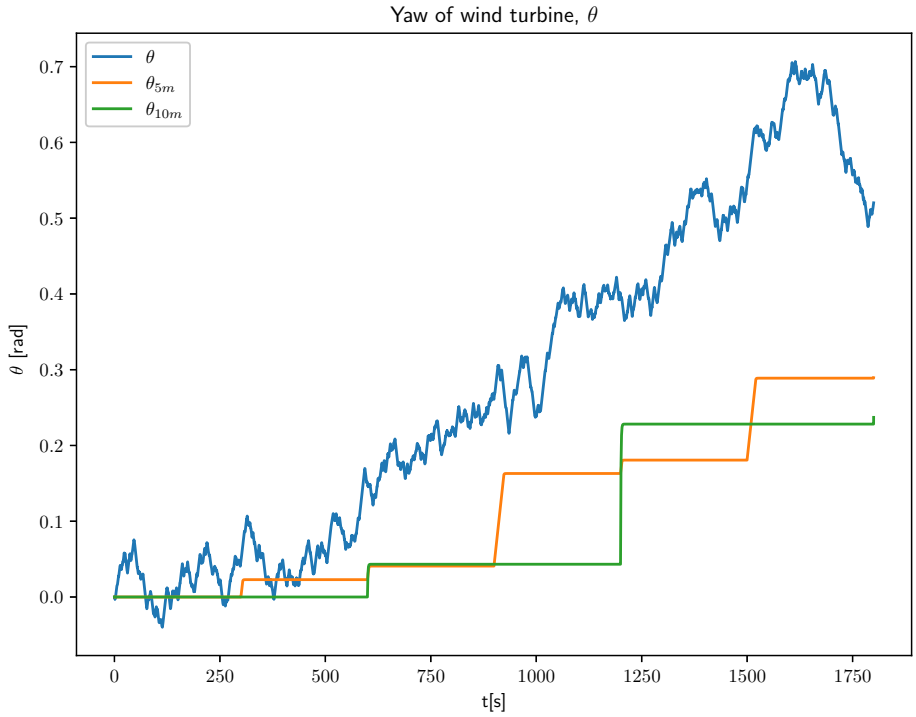


Figure 6.40: Comparison of wind turbine yaw.

To compare the yaw control methods a simulation with no yaw control is used as a reference.

Figures 6.41 and 6.42 illustrate the difference in angular velocity across the three yaw control methods. It's clear that the difference is substantial, particularly when the yaw error is large.

The power gained from using yaw control is shown in Watts and percentages in Figures 6.43 and 6.44, respectively.

To better understand the impact of these differences, Figure 6.45 presents the total energy produced in kWh for each control method, while Figure 6.46 shows the average energy gain in percentage.

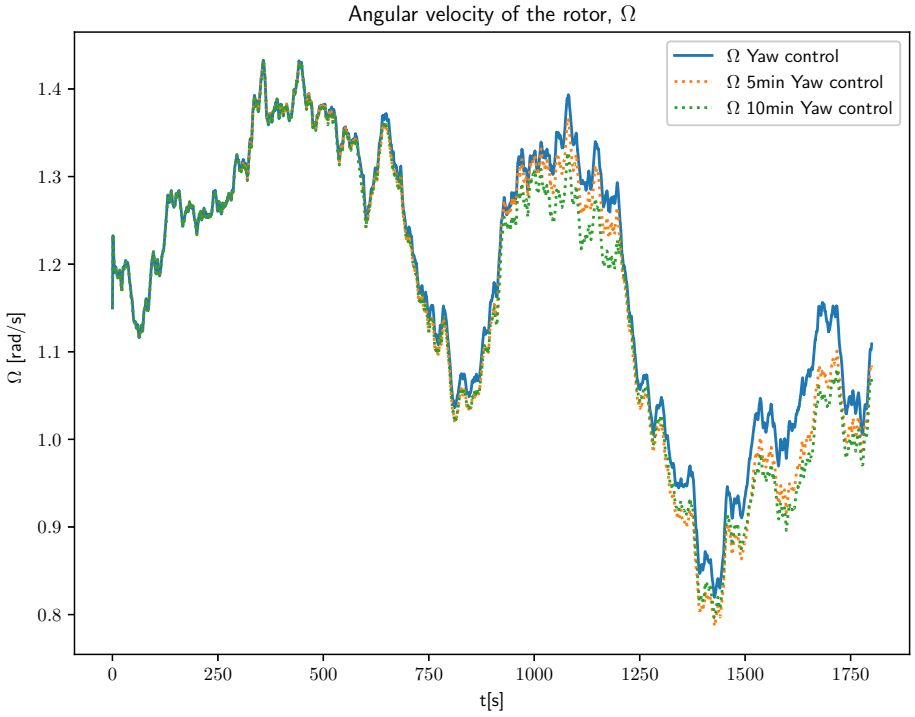


Figure 6.41: Angular velocity in the different yaw controls.

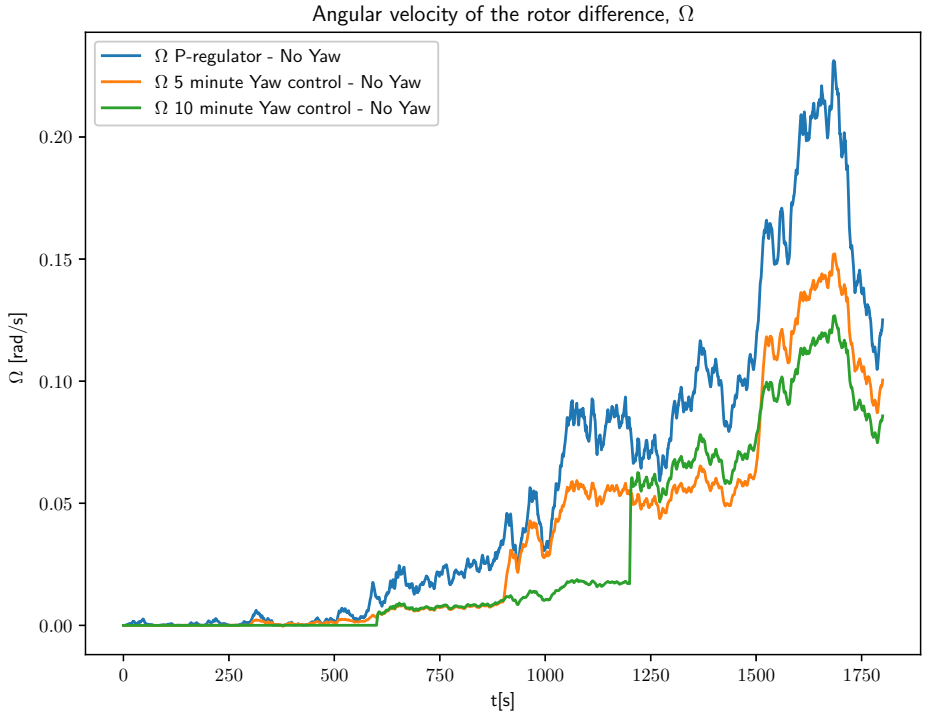


Figure 6.42: Gained angular velocity from yaw control.

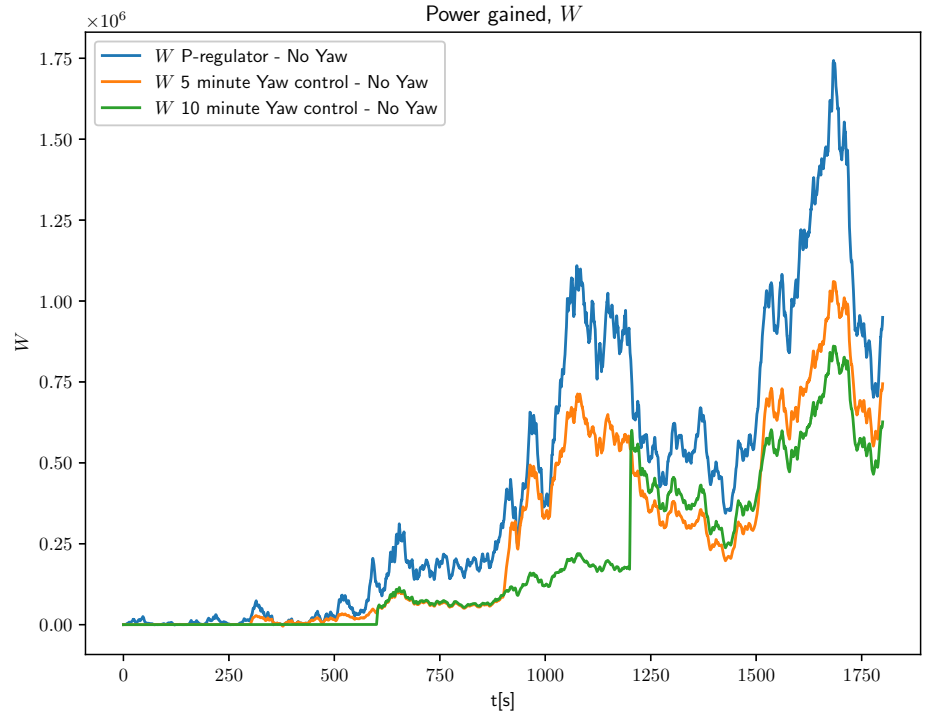


Figure 6.43: Increase in power from implementing yaw control.

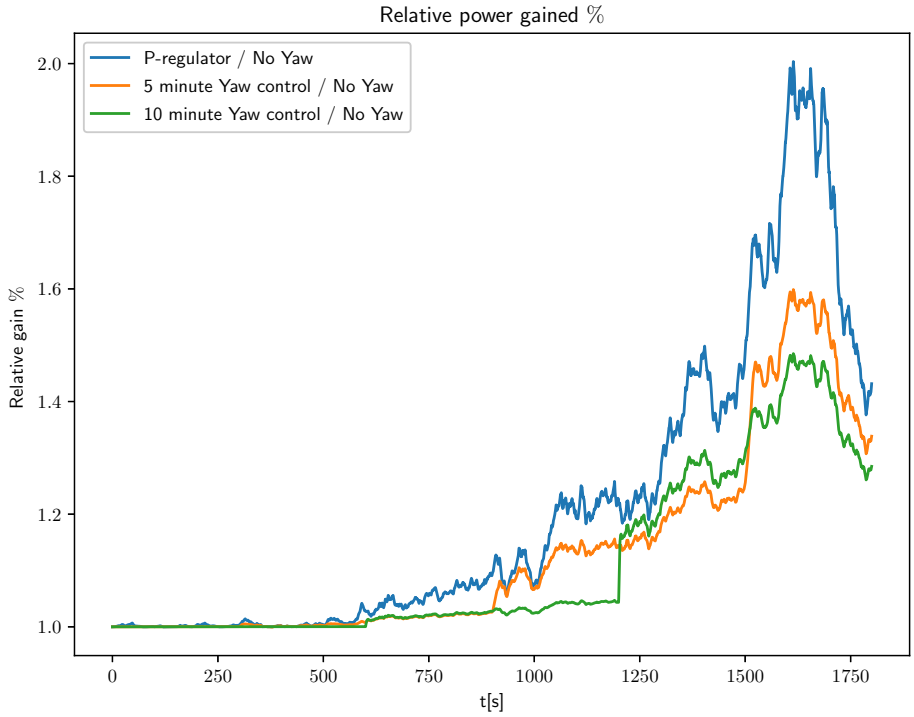


Figure 6.44: Relative % increase in power gain from utilizing yaw control.

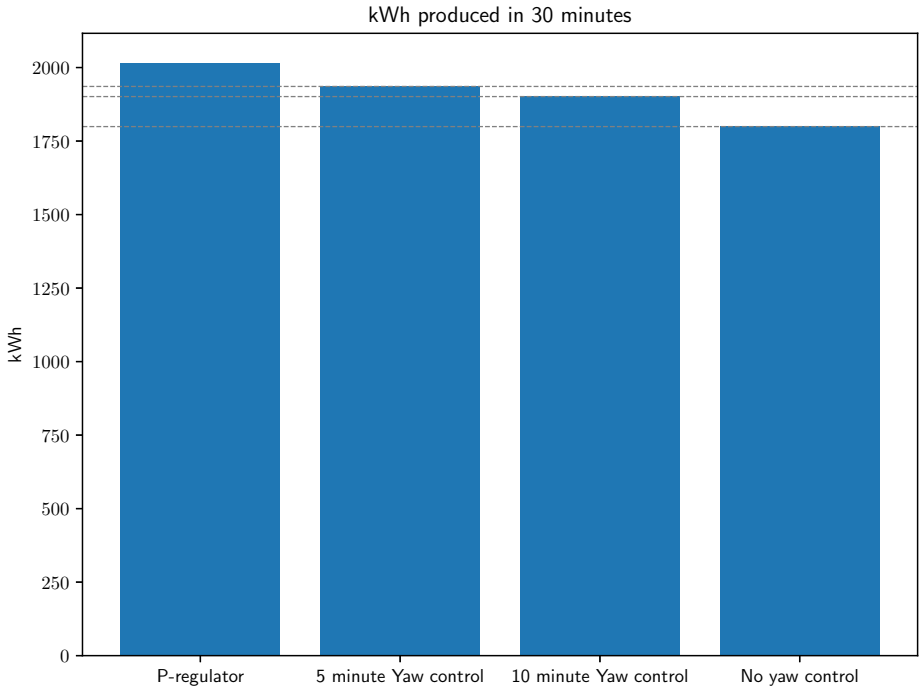


Figure 6.45: Bar graph illustrating the difference in produced Kwh over 30 minutes with 2015, 1935, 1901 and 1799 kWh produced, respectively.

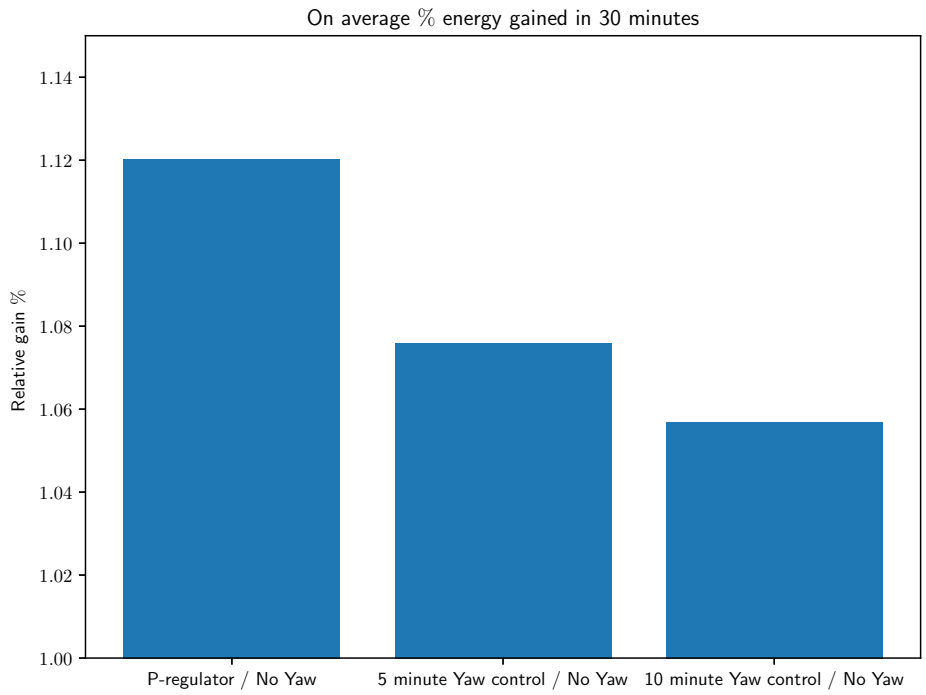


Figure 6.46: Bar graph illustrating relative % increase in power gained compared to no yaw control.

Chapter 7

Conclusion

7.1 Yaw-Velocity wind model

A new type of wind model was derived from measurement data using a Gauss-Markov style model identification strategy. The results in Subsection 3.4.1 validate the Yaw-Velocity model's ability to accurately estimate autocorrelations and generate simulations that align with observed data. The model's performance remains robust even when extended beyond the initial fitting range, indicating it has not overfitted the data. This suggests that limiting the simulations to 50 minutes is a conservative approach, which is deemed appropriate for ensuring the robustness and reliability of the results.

7.2 Simulation Results

In the previous chapter the simulation results were presented. This section discusses the results.

Kalman Filter on the Cartesian Wind Model

The Kalman filter, while able to provide a basic estimate of the yaw error, falls short in accurately capturing the system's nonlinearities, especially when conditions deviate from the point of linearization. Given the dynamic nature of wind conditions, which can change significantly over short time periods, this limitation is particularly problematic for yaw error estimation. Consequently, the Kalman filter is deemed unsuitable for this specific application.

Extended Kalman Filter on the Cartesian Wind Model

The Extended Kalman filter was tested on a wind signal that deviated significantly from the initial linearization point, demonstrating its ability to accurately capture the system's nonlinear dynamics. Unlike the Kalman filter, the Extended Kalman filter was able to produce precise state estimates even when conditions strayed from the initial linearization. To emphasize the importance of accurate yaw error estimation, a simple P-regulator was implemented to align the wind turbine with the wind. The power gain from this yaw control was then compared to a similar setup without yaw control. The results showed a substantial increase in power under these conditions. The potential power gain is quite dramatic in these conditions, however, this does not come as a surprise as it is not really a fair comparison.

Extended Kalman Filter on the Yaw-Velocity Wind Model

The Extended Kalman filter showcases its strength particularly well when employed on the Yaw-Velocity wind model. It demonstrates its ability to produce good state estimates when exposed to varying conditions. An attempt at reproducing yaw control employed on real wind turbines was then made by making two yaw controllers correcting a moving average yaw error based on exact measurements. There obviously exists ways of improving these yaw controls, however they resemble yaw controllers used in the literature as stated in Section 6.2. The results exemplify that there is a significant amount of power that can be gained through effective yaw control. In conditions where pitch control has to be activated to not exceed the rated maximum power, this sees less benefit. It's also worth noting that while the P-regulator was implemented to highlight potential gains, continuous yawing of the wind turbine may cause more harm than good in the long run.

Yaw error estimation

The findings in Chapter 6 can be condensed into the conclusion that accurate yaw error estimation can be achieved by utilizing the Extended Kalman filter on either wind model.

7.3 Future work

The scope of this thesis is inherently limited and several areas for improvement have been identified. These areas present opportunities for further research. The following list outlines potential paths for a continuation of this work:

- Explore the potential usage of yaw error estimation, for instance, accurate wake deflection or to obtain data for machine learning approaches.
- In terms of yaw control there is potential for utilizing these measurements, however, the balance between power optimization and equipment longevity must be considered when developing the yaw control.
- Explore the effect of measurement noise on the estimates.
- Implement the Extended Kalman filter on a real wind turbine. The results shown here, while efficient in simulations, might not be in a practical setting.

Bibliography

[1] J. Jonkman, S. Butterfield, W. Musial, and G. Scott. Definition of a 5-mw reference wind turbine for offshore system development. Technical Report NREL/TP-500-38060, National Renewable Energy Laboratory, Golden, CO, 2009.

[2] Finn Matras. Ba8607 - design of offshore wind turbines: Rotor airloads. Unpublished manuscript, 2021.

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

[4] Joyce Lee and Feng Zhao. Global wind report 2022, 2022.

[5] Albert Betz. Das maximum der theoretisch möglichen ausnutzung des windes durch windmotoren. *Die Naturwissenschaften*, 7(23):561–564, 1919.

[6] Peter J. Schubel and Richard J. Crossley. Wind turbine blade design. *Energies*, 5(9):3425–3449, 2012.

[7] Justin Creaby, Yaoyu Li, and John E. Seem. Maximizing wind turbine energy capture using multivariable extremum seeking control. *Wind Engineering*, 33(4):361–387, 2009.

[8] Sura Zuheir, Oday Ibraheem Abdullah, and Mustafa Al-Maliki. Stress and vibration analyses of the wind turbine blade (a nrel 5mw). *Journal of mechanical engineering research and developments*, 42(4):14–19, 2019.

[9] Det Norske Veritas, Forskningscenter Risø, Risø National Laboratory (Copenhague), and Risø National Lab. (DK). Wind Energy Department. *Guidelines for design of wind turbines*. Contract ENS-51171/98-0036. Det Norske Veritas : Wind Energy Department, Risø National Laboratory, 2001.

[10] Paul A. Fleming, Andrew Ning, Pieter M. O. Gebraad, and Katherine Dykes. Wind plant system engineering through optimization of layout and yaw control. *Wind Energy*, 19(2):329–344, 2016.

[11] Marc Bromm, Andreas Rott, Hauke Beck, Lukas Vollmer, Gerald Steinfeld, and Martin Kühn. Field investigation on the influence of yaw misalignment on the propagation of wind turbine wakes. *Wind Energy*, 21(11):1011–1028, 2018.

[12] J. Ribrant and L. Bertling. Survey of failures in wind power systems with focus on swedish wind power plants during 1997-2005. In *2007 IEEE Power Engineering Society General Meeting*, pages 1–8, 2007.

[13] T. Mikkelsen, N. Angelou, K. Hansen, M. Sjöholm, M. Harris, C. Slinger, P. Hadley, R. Scullion, G. Ellis, and G. Vives. A spinner-integrated wind lidar for enhanced wind turbine control. *Wind Energy*, 16(4):625–643, 2013.

[14] Yunong Bao and Qinmin Yang. A data-mining compensation approach for yaw misalignment on wind turbine. *IEEE Transactions on Industrial Informatics*, 17(12):8154–8164, 2021.

[15] P A Fleming, A K Scholbrock, A Jehu, S Davoust, E Osler, A D Wright, and A Clifton. Field-test results using a nacelle-mounted lidar for improving wind turbine power capture by reducing yaw misalignment. *Journal of Physics: Conference Series*, 524(1):012002, jun 2014.

[16] Torben Mikkelsen, Kasper Hjorth Hansen, Nikolas Angelou, Mikael Sjöholm, Michael Harris, Paul Hadley, Richard Scullion, Gary Ellis, and G Vives. Lidar wind speed measurements from a rotating spinner. In *European Wind Energy Conference and Exhibition*, 2010.

[17] E A Bossanyi, A Kumar, and O Hugues-Salas. Wind turbine control applications of turbine-mounted lidar. *Journal of Physics: Conference Series*, 555(1):012011, dec 2014.

[18] Knud A. Kragh, Morten H. Hansen, and Torben Mikkelsen. Precision and shortcomings of yaw error estimation using spinner-based light detection and ranging. *Wind Energy*, 16(3):353–366, 2013.

[19] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM review*, 59(1):65–98, 2017.

[20] National Renewable Energy Laboratory (NREL). National wind technology center (nwtc) live data feed, 2023. Accessed: 2023-05-19.

[21] Knud A. Kragh, Paul A. Fleming, and Andrew K. Scholbrock. Increased Power Capture by Rotor Speed–Dependent Yaw Control of Wind Turbines. *Journal of Solar Energy Engineering*, 135(3), 06 2013. 031018.

[22] Bo Jing, Zheng Qian, Yan Pei, Lizhong Zhang, and Tingyi Yang. Improving wind turbine efficiency through detection and calibration of yaw misalignment. *Renewable Energy*, 160:1217–1227, 2020.

[23] Nida Sheibat-Othman, Sami Othman, Raoaa Tayari, Anis Sakly, Peter F. Odgaard, and Lars F.S. Larsen. Estimation of the wind turbine yaw error by support vector machines. *IFAC-PapersOnLine*, 48(30):339–344, 2015. 9th IFAC Symposium on Control of Power and Energy Systems CPES 2015.

[24] Evangelos Tsioumas, Nektarios Karakasis, Nikolaos Jabbour, and Christos Mademlis. Indirect estimation of the yaw-angle misalignment in a horizontal axis wind turbine. In *2017 IEEE 11th International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives (SDEMPED)*, pages 45–51. IEEE, 2017.

[25] Theodore Von Karman. Progress in the statistical theory of turbulence. *Proceedings of the National Academy of Sciences*, 34(11):530–539, 1948.

[26] E. Welfonder, R. Neifer, and M. Spanner. Development and experimental identification of dynamic models for wind turbines. *Control Engineering Practice*, 5(1):63–73, 1997.

[27] D.A. Peters, D. Barwey, and A. Su. An integrated airloads-inflow model for use in rotor aeroelasticity and control analysis. *Mathematical and Computer Modelling*, 19(3):109–123, 1994.

[28] Finn. Matras and Morten Dinhoff. Pedersen. Inflow as feedback - the multirotor case. Submitted for review, 2023.

[29] James F Manwell, Jon G McGowan, and Anthony L Rogers. *Wind energy explained: theory, design and application*. John Wiley & Sons, 2010.

[30] Jan van der Tempel and David-Pieter Molenaar. Wind turbine structural dynamics – a review of the principles for modern power generation, onshore and offshore. *Wind Engineering*, 26(4):211–220, 2002.

[31] R. E. Froude. On the part played in propulsion by difference in pressure. *Transaction of the Institute of Naval Architects*, 30:390–423, 1889.

[32] W. Johnson. *Helicopter Theory*. Dover Books on Aeronautical Engineering Series. Dover Publications, 1994.

[33] ANSYS, Inc. ANSYS Fluent, Version 19.2. `http://www.ansys.com`, 2021.

[34] Morten Dinhoff Pedersen. Stabilization of floating wind turbines. 2017.

[35] D. M. Pitt and D. A. Peters. Theoretical prediction of dynamic inflow derivatives. In *SIXTH EUROPEAN ROTORCRAFT AND POWERED LIFT AIRCRAFT FORUM*, 1980.

[36] M.O.L. Hansen, J.N. Sørensen, S. Voutsinas, N. Sørensen, and H.Aa. Madsen. State of the art in wind turbine aerodynamics and aeroelasticity. *Progress in Aerospace Sciences*, 42(4):285–330, 2006.

[37] Morten Dinhoff Pedersen. Fractional inflow dynamics. In *AHS Technical Conference on Aeromechanics Design for Transformative Vertical Flight*, pages 1–12, San Francisco, CA, January 2018.

[38] H. Glauert. A general theory of the autogyro. Reports and Memoranda 1111, Air Ministry, 11 1926. Presented by the Director of Scientific Research.

[39] Robert T. N. Chen. A survey of nonuniform inflow models for rotorcraft flight dynamics and control applications. In *European Rotorcraft Forum*, Amsterdam, 11 1989. NASA Ames Research Center. NASA-TM-102219, NAS 1.15:102219, A-89220, Document ID: 19900006622.

[40] Snorri Gudmundsson. Chapter 14 - the anatomy of the propeller. In Snorri Gudmundsson, editor, *General Aviation Aircraft Design*, pages 581–659. Butterworth-Heinemann, Boston, 2014.

[41] Temesgen Batu and Hirpa Lemu. *Comparative Study of the Effect of Chord Length Computation Methods in Design of Wind Turbine Blade*, pages 106–115. 01 2020.

[42] M. O. L. Hansen. *Aerodynamics of Wind Turbines*. Earthscan, London, 2 edition, 2008.

[43] Jens Nørkær Sørensen. *General momentum theory for horizontal axis wind turbines*, volume 4. Springer, 2016.

[44] K.E. Johnson, L.Y. Pao, M.J. Balas, and L.J. Fingersh. Control of variable-speed wind turbines: standard and adaptive techniques for maximizing energy capture. *IEEE Control Systems Magazine*, 26(3):70–81, 2006.

[45] Mike T. van Dijk, Jan-Willem van Wingerden, Turaj Ashuri, Yaoyu Li, and Mario A. Rotea. Yaw-misalignment and its impact on wind turbine loads and wind farm power output. *Journal of Physics: Conference Series*, 753(6):062013, sep 2016.

[46] Galih Bangga. Modeling Turbulence for Wind Turbine Simulations. In *Wind Turbine Aerodynamics Modeling Using CFD Approaches*. AIP Publishing LLC.

[47] Finn Matras, Dirk Peter Reinhardt, and Morten Dinhoff Pedersen. Parametrization of airloads using a homogeneity-based orthogonal decomposition. In *2022 26th International Conference on System Theory, Control and Computing (ICSTCC)*, pages 553–558, 2022.

[48] Weibin Gu, Dewen Hu, Liang Cheng, Yabing Cao, Alessandro Rizzo, and Kimon P. Valavanis. Autonomous wind turbine inspection using a quadrotor. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 709–715, 2020.

[49] Zhengru Ren, Roger Skjetne, Zhiyu Jiang, Zhen Gao, and Amrit Shankar Verma. Integrated gnss/imu hub motion estimator for offshore wind turbine blade installation. *Mechanical Systems and Signal Processing*, 123:222–243, 2019.

[50] Robert Grover Brown and Patrick Y C Hwang. *Introduction to random signals and applied kalman filtering: with MATLAB exercises; 4th ed.* Wiley, New York, NY, 2012.

[51] Pierre Lamon. The gauss–markov process. In *3D-Position Tracking and Control for All-Terrain Robots*, pages 93–95. Springer, 2008.

[52] Paul Veers. Three-dimensional wind simulation. 01 1988.

[53] Amadou D. Diop, Emil Ceanga, Jean-Louis Rétiveau, Jean-François Méthot, and Adrian Ilinca. Real-time three-dimensional wind simulation for windmill rig tests. *Renewable Energy*, 32(13):2268–2290, 2007.

[54] Miles Lubin, Oscar Dowson, Joaquim Dias Garcia, Joey Huchette, Benoît Legat, and Juan Pablo Vielma. Jump 1.0: Recent improvements to a modeling language for mathematical optimization. *Mathematical Programming Computation*, 2023. In press.

[55] Andreas Wächter and Lorenz T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106:25–57, 2006.

[56] Xue Ying. An overview of overfitting and its solutions. *Journal of Physics: Conference Series*, 1168(2):022022, feb 2019.

[57] C.T. Chen. *Linear System Theory and Design*. The Oxford Series in Electrical and Computer Engineering. Oxford University Press, 2014.

[58] J.P. Hespanha. *Linear Systems Theory: Second Edition*. Princeton University Press, 2018.

[59] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, 03 1960.

[60] T.I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2021.

[61] Xin Li, Kejun Wang, Wei Wang, and Yang Li. A multiple object tracking method using kalman filter. In *The 2010 IEEE International Conference on Information and Automation*, pages 1862–1866, 2010.

[62] Simon Lynen, Bernhard Zeisl, Dror Aiger, Michael Bosse, Joel Hesch, Marc Pollefeys, Roland Siegwart, and Torsten Sattler. Large-scale, real-time visual–inertial localization revisited. *The International Journal of Robotics Research*, 39(9):1061–1084, 2020.

[63] Dongran Song, Jian Yang, Zili Cai, Mi Dong, Mei Su, and Yinghua Wang. Wind estimation with a non-standard extended kalman filter and its application on maximum power extraction for variable speed wind turbines. *Applied Energy*, 190:670–685, 2017.

[64] Sung ho Hur. Short-term wind speed prediction using extended kalman filter and machine learning. *Energy Reports*, 7:1046–1054, 2021.

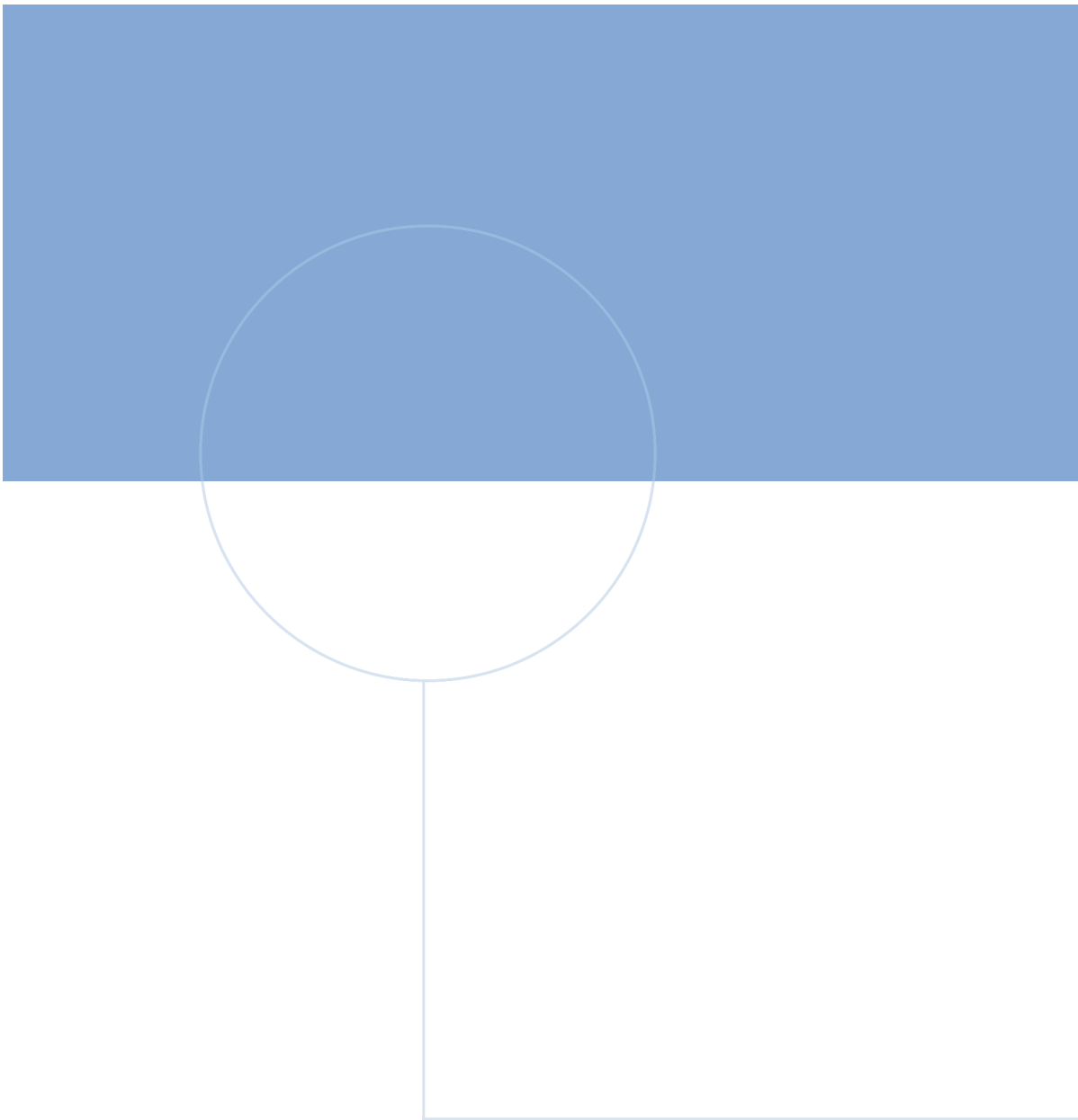
[65] Mahmood Mirzaei, Hans Henrik Niemann, and Niels Kjølstad Poulsen. A mu-synthesis approach to robust control of a wind turbine. In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 645–650, 2011.

[66] Wei Tong. *Wind power generation and wind turbine design*. WIT press, 2010.

[67] Xinghui Dong, Yizhou Wang, Hui Wang, and Lingxing Kong. Optimised yaw model for adaptive wind direction characteristic based on a data-driven approach. *IET Renewable Power Generation*, 15(1):237–250, 2021.

[68] Troels Friis Pedersen, Niels N Soerensen, Luca Vita, and Peder Enevoldsen. Optimization of wind turbine operation by use of spinner anemometer. 2008.

[69] Roozbeh Bakhshi and Peter Sandborn. The effect of yaw error on the reliability of wind turbine blades. In *Energy Sustainability*, volume 50220, page V001T14A001. American Society of Mechanical Engineers, 2016.



NTNU

Norwegian University of
Science and Technology