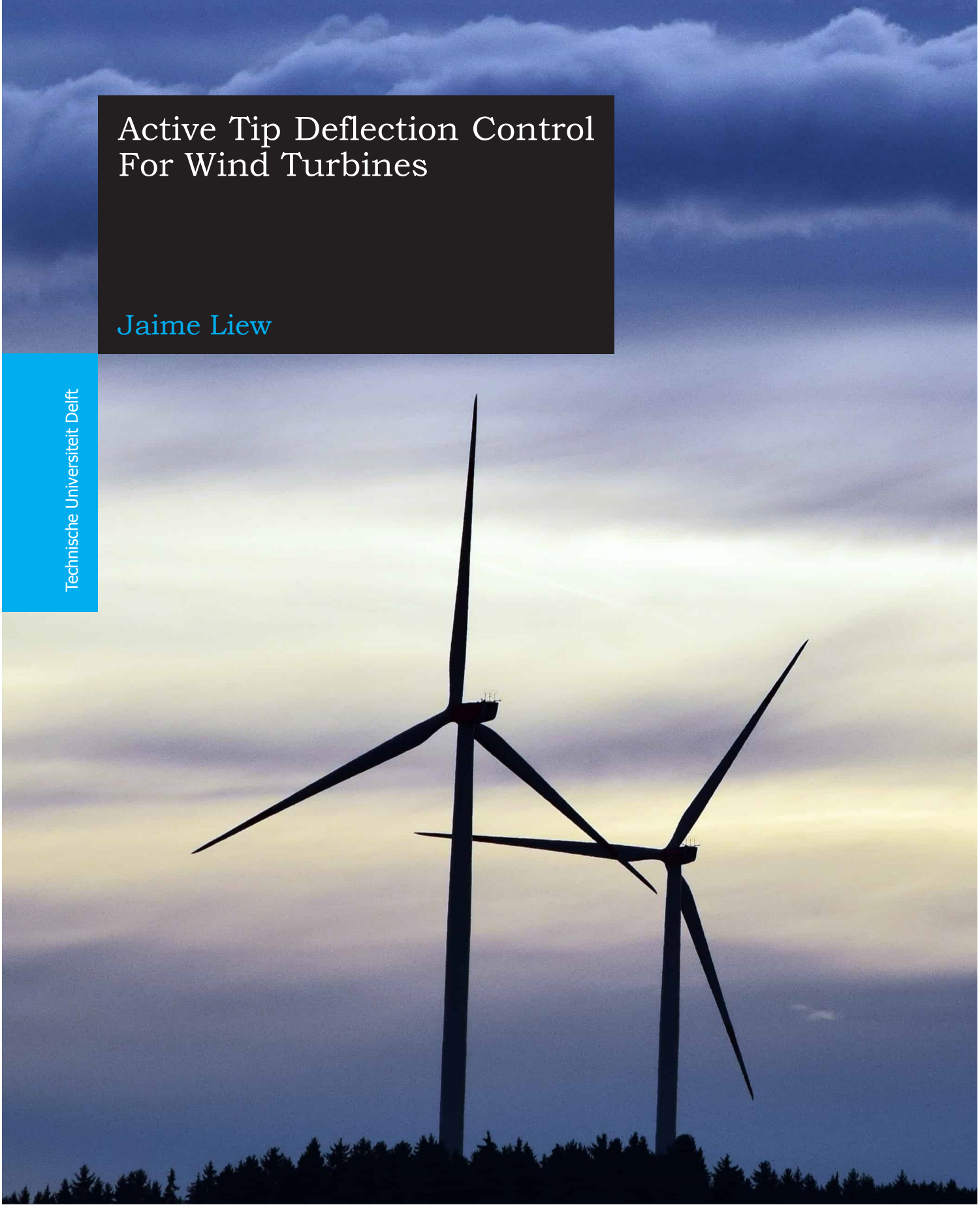
Active Tip Deflection Control For Wind Turbines

Jaime Liew

Technische Universiteit Delft



TU Delft DTU LM WIND POWER EWEM

Active Tip Deflection Control For Wind Turbines

by

Jaime Liew

in partial fulfillment of the requirements for the degree of Master of Science in Aerospace Engineering at Delft University of Technology and Master of Science in Engineering (European Wind Energy) at The Technical University of Denmark.

Student number:	4615409	
Project duration:	November 1, 2017 – August 3, 2018	
Supervisors:	Dr.ir. Jessica Holierhoek	TU Delft
	Prof.Dr. Taeseong Kim	DTU
	Christian Frank Andersen	LM Wind Power
Chairman:	Dr.ir. Carlos Simão Ferreira	TU Delft

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

Abstract

This thesis presents an individual tip control (IPC) system based on blade tip deflection measurements. The controller is based on novel sensor inputs which measure flapwise tip deflection distance at a high sampling rate. IPC plays a key role in reducing fatigue loads in wind turbine components. These fatigue loads are caused by differential loads such as wind shear, yaw misalignment and turbulence. The presented controller is implemented in HAWC2 and high fidelity load measurements are produced using the DTU10MW Reference Wind Turbine. Lifetime equivalent load reductions were seen in both rotating and fixed frame components under extreme turbulence, inverse shear conditions and in normal operating conditions. A novel implementation of IPC is also presented where the blade tips are guided along a fixed trajectory to maximise blade-tower clearance. The motivation of this implementation is to reduce the chance of blade-tower interactions for large diameter turbine rotors. The theoretical background used in this study is presented first along with details of controller discretisation methods. Details of the iterative control design process is presented, and the simulated fatigue loads are compared for a number of control architectures. Finally, the implementation of the tip trajectory tracking control is presented along with an analysis of the pitch rate limits and the effect of IPC on electrical power output.

Acknowledgement

The opportunity to carry out my master thesis on the topic of wind energy control has been an exciting and fulfilling experience for which I have many people to acknowledge.

On the idea of tip deflection sensors, I would like to thank Christian Frank Andersen and Armin Hermes from LM Wind Power for providing me with the inspiration of this thesis as well as support for the duration of the project. Knowledge on the topic of control theory and Fortran is credited to Alan Wai Hou Lio and Albert Meseguer Urban, who's help has saved me countless hours in understanding and applying the theory in this study. A big thank you goes out to my thesis supervisors Taeseong Kim and Jessica Holierhoek for their advice, direction and support throughout the project.

Thank you to the European Wind Energy Master program coordinators for providing such an incredible platform from which bright minds can come together in the field of Wind Energy. Receiving the opportunity to join this community is greatly appreciated, especially the support from the Erasmus Mundus Scholarship.

In my two years in Denmark and the Netherlands, I have made lifelong friends from all over the world, so I would like to thank them, especially my team members in the Aeolus board. Needless to say, I would like to thank my friends and family back in Australia for their encouragement from abroad to follow my dream.

Contents

Nomenclature **1**

1 Introduction **2**
1.1 Background . . . 2
1.2 Motivation . . . 3
1.3 Report Outline . . . 3

2 Simulation Setup **5**
2.1 HAWC2 Simulation Environment . . . 5
2.2 DTU-10MW Reference Turbine . . . 5
2.3 Design Load Case (DLC) Key Parameters . . . 6

3 Theoretical Framework **7**
3.1 Relationship between Tip Deflection and Blade Loads . . . 7
3.2 Quantifying Fatigue Loads . . . 12
3.3 Disturbance Rejection Theory . . . 13
3.3.1 Performance Measures for Disturbance Rejection . . . 14
3.3.2 Robustness Measures for Disturbance Rejection . . . 15
3.4 Blade System Transformations . . . 16
3.5 Filter Design Using Loop Shaping . . . 17
3.6 Controller Discretisation . . . 18
3.6.1 Matched-Pole-Zero Approximation (MPZ) . . . 18
3.6.2 Bilinear Transformation . . . 19
3.6.3 Effect of sampling rate on frequency response . . . 20
3.6.4 Discrete frequency to discrete time domain transformation . . . 21
3.7 Control Algorithm and Properties . . . 21

4 System Identification of Blade System **23**
4.1 Background . . . 23
4.2 System Identification of Plant . . . 25

5 Tip Disturbance Rejection Controller **28**
5.1 Background . . . 28
5.2 Open Loop Output Estimation . . . 29
5.3 Baseline Control Design . . . 30
5.4 Single frequency Control Design . . . 32
5.4.1 Tip Deflection Frequency Analysis . . . 33
5.4.2 Equivalent Load Reduction . . . 35
5.5 Two Frequency Control Design . . . 36
5.5.1 Tip Deflection Frequency Analysis . . . 38

5.5.2 Equivalent Load Reduction . . . 39
5.5.3 Performance in Extreme Turbulence . . . 41
5.6 Strain Gauge versus Tip Deflection Based Control . . . 42
5.7 Influence on Power Output . . . 43

6 Tip Trajectory Tracking Controller **45**
6.1 Background . . . 45
6.2 Considered Wind and Controller Conditions . . . 48
6.3 Results in Normal and Inverse Shear . . . 49
6.3.1 Tracking performance . . . 49
6.3.2 Minimum tower clearance . . . 50
6.3.3 Effect on fatigue loads . . . 52
6.3.4 Effect on blade pitching rate . . . 54

7 Conclusion and Recommendations **56**
7.1 Conclusion . . . 56
7.2 Future Work . . . 57

Appendices **61**

A Transfer Functions **61**
A.1 Plant Transfer Functions . . . 61
A.2 Controller Transfer Functions . . . 62
A.2.1 PI Control . . . 62
A.2.2 Single Frequency Control . . . 62
A.2.3 Two Frequency Control . . . 62
A.3 Controller performance . . . 63

B Code Implementation **64**
B.1 Python-HAWC2 Interface . . . 64
B.2 HAWC2 IPC DLL . . . 66

C Full Blade Deformation Estimation **68**
C.1 Strain Gauge and Tip Deflection Sensor Fusion . . . 68
C.2 Full Blade Deflection Estimation . . . 68

Nomenclature

U	Wind speed
t	Time
s	Laplace variable
$\psi(t)$	Rotor azimuth angle
f_{1p}	Once per revolution frequency
f_{np}	n times per revolution frequency
j	Imaginary unit
C_{PI}	PI Controller (Section 5.3)
C_{fnp}	Single frequency targeting f_{np} (Section 5.4)
C_2	Two frequency controller (Section 5.5)
A_r	Tip Trajectory Tracking amplitude
ω	Angular frequency
ζ	Damping ratio
K_p	Proportional gain
$y(t)$, $y(s)$	Tip deflection distance
$\bar{y}(t)$, $\bar{y}(s)$	Mean tip deflection distance
$\tilde{y}(t)$, $\tilde{y}(s)$	Tip deflection distance perturbation
$\theta(t)$, $\theta(s)$	Blade pitch angle
$\bar{\theta}(t)$, $\bar{\theta}(s)$	Mean blade pitch angle
$\tilde{\theta}(t)$, $\tilde{\theta}(s)$	Blade pitch angle perturbation
$d(t), d(s)$	Blade disturbance
s_m	Stability margin
b_i	The ith feed-forward controller coefficient
a_i	The ith feed-backward controller coefficient
$P(s)$	Plant transfer function
$C(s)$	Controller transfer function

Important and reoccurring symbols are defined above. All other symbols are defined in-text.

Chapter 1

Introduction

1.1 Background

Wind energy is one of the most prominent sources of renewable energy. The technology has seen remarkable growth in recent decades and continues to push boundaries in efficiency and levelised cost of energy. In the battle to reduce the cost of energy, turbine rotors are being upscaled with a larger, lighter, and more flexible design. One of the major limitations of modern turbine design is fatigue stress in turbine components which becomes more significant for larger turbine designs. As the blades sweep through the air, large differential forces are experienced in the blade roots and the rotor shaft [1]. These stresses, caused by wind shear, tower shadow, yaw misalignment, nacelle tilt and turbulence, result in accumulated fatigue damage, and can reduce the operating lifetime of the turbine components [2].

Although these differential loads have been thoroughly investigated in literature, the control systems used in modern wind turbines are typically designed with the assumption of uniform blade loading across the rotor plane. Specifically, collective pitch control (CPC) is the standard approach to regulate rotor speed and power output despite being ineffective at alleviating blade loads. This is the motivation behind individual pitch control (IPC), where each blade is subject to a different pitch angle depending on the rotational position of the rotor. IPC has shown promising results in reducing blade flapwise loads as well as in non-rotating components such as the main rotor shaft. The use of IPC in wind farms could have a significant impact on reducing the levelised cost of energy by extending the lifetime of turbine components.

IPC in literature has shown great reductions in turbine loads using a variety of controller designs. Bossanyi [3] showed an 18% reduction in equivalent fatigue loads in the out-of-plane blade root moments, as well as a noticeable reduction in shaft and yaw bearing moments when using IPC compared to CPC. Larsen [4] shows a 25% reduction in blade flapwise equivalent loads as well as a 9% reduction in shaft equivalent loads using IPC using local blade flow measurements. Trudnowski [5] achieved an 86% reduction in flapwise blade loads using only the rotor angle as an input signal, however, the analysis neglected the effects of turbulence.

Given the success of IPC in experiments and simulations, a range of control methodologies has been adapted with varying degrees of success. Mirzaei et al. [6] showed comparable reductions in out-of-plane blade root bending moments using both PI (proportional-integral) control and model predictive control based on LIDAR measurements. Selvam [7] compares IPC systems using PI control as well as linear quadratic gaussian (LQG) control. PI control achieved load reductions at low frequencies, and the LQG controller was able to achieve load reductions at a higher bandwidth, including f_{2p} and f_{3p} in the rotating frame, therefore able to reduce loads on non-rotating parts such as the nacelle, yaw bearing and tower. IPC using $\mathcal{H}_\infty$ control was addressed in [8], [9], [1] and [10], showing not only reductions in OOP blade root bending moments, but also adequate robustness from unmodelled and stochastic behaviour in the system.

1.2 Motivation

Although numerous IPC strategies exist in literature, the vast majority of the methods share a commonality. That is, IPC is performed with feedback control using strain gauge sensors at the blade roots as the main control input. There are a number of complications which can arise in the use of strain gauges. Strain gauges can be difficult to calibrate for long term operation as they are highly sensitive to external factors such as temperature and humidity. They can experience long term drifting, as well as difficulty in measuring stresses in anisotropic material [11]. With improving sensing technology, alternatives to strain gauge sensors have been explored for use in IPC and other smart rotor technologies. For example, Larsen investigated the use of IPC using local blade flow measurements [4].

In this project, blade tip deflection sensors are investigated for use in IPC. The use of tip deflection sensors in wind turbines is very rare in literature. Bossanyi briefly mentions the possibility of using accelerometers in the blade tips as an alternative to strain gauges, and also mentions the difficulty of maintaining such sensors due to inaccessibility of the blade tips [3]. Berg and Wilson use blade tip deflection and velocity as inputs for their turbine controller designs, however they focus on active flap control [12, 13]. To the best of the author's knowledge, no research has been performed on individual pitch control using tip deflection sensors.

One sensor with potential use in a feedback controller is the iRotor sensor developed by LM Wind Power [14]. The iRotor can provide high-bandwidth flapwise tip deflection measurements for each blade via radio signals. The benefit of using the iRotor over strain gauges is its ability to produce high precision measurements of the blade tip deflection in real time at a high sampling rate. The focus of this project is to show the potential applications of this sensor. In addition to investigating load reduction strategies, methods for increasing blade-tower clearance are also considered.

1.3 Report Outline

The goal of this project is to study the applications of blade tip deflection sensors for IPC applications. The report structure is outlined as follows:

Chapter 2: Simulation Setup

This chapter gives details of how the high-fidelity simulations are set up in HAWC2. The additional tools used to implement IPC in HAWC2 are introduced as well as the DTU10MW reference turbine. Finally, the key simulation parameters are defined.

Chapter 3: Theoretical Framework

The theoretical framework used in this report is outlined in this section. First, a relationship between tip deflection and blade loads is found both analytically and from simulated data. Secondly, the control framework for IPC is outlined including the performance and robustness measures used in this report. Commonly used blade transformations are outlined and the use of single-bladed control is justified, and the considerations regarding control discretisation are discussed.

Chapter 4: System Identification of Blade System

This section describes how the pitch to tip deflection transfer function is obtained for a single blade using system identification. A simplified version of the DTU10MW wind turbine is simulated at various wind speeds subject to a step pitch input. The tip deflection output signal is analysed with the step input signal to estimate the transfer function. This transfer function is used as the plant in designing the control system in Chapter 5.

Chapter 5: Tip Disturbance Rejection Control Design

This section outlines the design process and results of various IPC configurations with the objective of disturbance rejection. First, the behaviour of the open loop turbine system is simulated in HAWC2 and analysed. Next, a baseline PI controller is designed and the performance is analysed. A single frequency controller is designed and implemented separately for once per revolution frequency (f_{1p}) to four times per revolution frequency (f_{4p}) to identify where the greatest load reductions are found. Lastly, a synthesis of the f_{1p} and f_{2p} controllers is created which is found to improve load reductions in both the turbine blades and in the fixed frame compared to the PI and single frequency controllers.

Chapter 6: Tip Trajectory Trajectory Tracking

The IPC controller is modified to perform tracking on the tip position so that the blade tips follow a trajectory away from the tower. The motivation for this is to reduce the chance of tower strikes for highly flexible blades by maximising the distance between blade and tower. Various trajectory amplitudes are tested under both power law and inverse shear conditions. The effects of TTT on turbine loads and blade pitching limits are also investigated.

Chapter 7: Conclusion and Recommendations

In this chapter, the conclusions drawn from the project work are outlined and recommendations for future work on the topic are provided.

Chapter 2

Simulation Setup

This chapter provides details on how the HAWC2 aeroelastic code is used to perform simulations and obtain data for this project. Additionally, the DTU10MW turbine is introduced as well as the modifications made for the purpose of this study. Finally, key parameters for the design load case simulations are discussed.

2.1 HAWC2 Simulation Environment

HAWC2 is an aeroelastic code able to simulate wind turbine responses in the time domain. HAWC2 is used in this project to produce high fidelity simulation data to evaluate and verify the effectiveness of the control system. In particular, time series data of the bending moments of various components, tip deflection of the blades, and blade pitch angle signals are collected [15]. Two Dynamic libraries (DLL) where used in this project to assist in the control design and verification, described as follows:

Python Interface Via TCP/IP

A Python module (Appendix B) was written for the purpose of rapid prototyping and testing on the reference turbine. The module is able to receive sensor data from, and send commands to HAWC2 in real-time. Custom controllers can be implemented with ease using the module which interfaces a Python script with HAWC2 using the TCP/IP protocol DLL. For a number of the simulations in this project, this module is used to send tip deflection signals to a Python script, which then sends back the appropriate blade pitch demands to HAWC2.

Individual Pitch Control Augmentation

A DLLs was written in Fortran for this project which is able to perform individual pitch control. Unlike the Python module, the DLL is able to run in the HAWC2 native environment, allowing for better integration in the simulations, and no dependency on Python. The IPC controller parameters are defined in the main input file. The controller is designed to be augmented over the standard controller and is able to implement discrete single-bladed individual pitch control defined by a list of feed-forward and feed-backward filter coefficients (see Appendix B).

2.2 DTU-10MW Reference Turbine

The DTU-10MW Reference Wind Turbine Model is used in this project for the design and testing of the control systems [16]. The standard controller used for this turbine is the Basic DTU Wind Energy controller described in [17]. The IPC controllers analysed in this project are designed to extend the standard controller. Some of the key specifications used in this project are tabulated in Table 2.1.

Table 2.1: DTU10MW Reference Wind Turbine Model Key Parameters.

Parameter	Value
Wind regime	IEC Class 1A
Rated power output	10MW
Rated wind speed	11.4 m/s
Rotor diameter	178.3m
Hub height	119.0m
Minimum rotor speed	6 RPM (0.1Hz)
Rated rotor speed	9.6 RPM (0.16Hz)

Two modifications are made to the default turbine simulation settings.

1. The saturation limits for the pitch rate and pitch acceleration are deactivated in the pitch actuator DLL. This change ensures the pitch actuator limit does not interfere with the IPC controller. This modification ensures the actuator DLL exhibits linear behaviour only and prevents unexpected controller behaviour due to the nonlinearities introduced due to saturation. A pitch rate analysis is performed in this report to determine the required pitching limits for the designed IPC controller.

2. The pitch deviation monitor in the Basic DTU Wind Energy Controller is deactivated. This monitor shuts down the turbine in the event that a blade's pitch angle deviates from the standard controller reference pitch angle by a given threshold. As it is in the nature of IPC for blade pitch angles to deviate from the standard reference, this safeguard is deactivated.

2.3 Design Load Case (DLC) Key Parameters

The majority of simulations are based on DLC 1.1 as defined in the IEC-61400 standards [18]. This DLC is used for normal power production, and is suited to determine fatigue loads on the turbine. The different controllers in this project are therefore tested in normal power production simulations in order to analyse the controller performance. Additional DLCs are also performed using extreme turbulence (DLC 1.3) and extreme wind shear (DLC 1.5) [18] in Chapters 5 and 6 to investigate the performance and robustness of the controllers in extreme conditions. Some of the key simulation parameters are outlined in table 2.2.

Table 2.2: HAWC2 DLC Key Parameters [16].

Parameter	Value
Simulation sampling frequency	100Hz
Simulation duration	700s (first 100s discarded)
Wind speeds	4, 6, ..., 24, 26 m/s
Turbulence model	Mann
Turbulent seeds	3 per wind speed
Tower shadow method	Potential flow

Unless otherwise specified in the report, the simulations use the following guidelines by default:

- The wind shear profile used is the power law profile with shear coefficient $\alpha = 0.2$.
- The turbulence intensity is determined using the normal turbulence model (NTM) defined in IEC-61400 [18].
- Tip deflection measurements are taken in the flapwise direction which is defined in the blade frame of reference as the direction of least bending resistance.

Chapter 3

Theoretical Framework

This section outlines the theoretical content for the analysis in this project. A relationship between tip deflection and blade root bending moment is first established, both theoretically and through simulation. The control theory and terminology used for the controller design is described, including the choice of blade transformation and the controller discretisation process. Finally, the tip deflection individual pitch control algorithm is provided.

3.1 Relationship between Tip Deflection and Blade Loads

For individual pitch control systems, it is standard practice for wind turbine loads to be measured with strain gauges. Strain gauge measurements can estimate the blade bending moments which are used as the controller input for the majority of IPC literature [1, 3, 7, 8, 19]. However, there are a number of shortcomings to strain gauge measurements, including sensitivity to temperature, humidity, mounting error and long term drift [11]. Tip deflection measurements, although subject to their own sources of error, could be a viable alternative to strain gauges in IPC systems. In order to control blade loads with tip deflection control, a relationship between blade loads and tip deflection must exist. This section explores this relationship by investigating the tip deflection distance and the blade flapwise root bending moment (RBM).

the bending moment at the blade root is of interest as this is where the largest load occurs. In order to analytically show a relationship, a simplified structural model of a wind turbine blade is presented in this section. It should be noted that loads and deflections are only considered in the flapwise direction.

The blade is assumed to behave as an Euler-Bernoulli cantilever beam with length L. The flapwise deflection of the beam, $u(t, z)$, at a transverse distance, z, from the fixed end, and at a time, t, can be expressed a linear combination of the mode shapes:

$$u(t, z) = \sum_{i=1}^{\infty} \alpha_i(t)\gamma_i(z) \tag{3.1}$$

where α_i is the amplitude of the ith mode, and γ_i is the non-dimensional shape of the ith mode such that its value at the tip is $\gamma_i(L) = 1$. Therefore, the deflection at the blade tip is:

$$u(t, L) = \sum_{i=1}^{\infty} \alpha_i(t) \tag{3.2}$$

From Euler-Bernoulli beam theory, the bending moment, $M(z)$, is expressed as:

$$M(t, z) = EI(z)\frac{\partial^2 u}{\partial z^2}(z) \tag{3.3}$$

where $EI(z)$ is the flexural rigidity defined as the product of the area moment of inertia and the Young's Modulus of the beam material, and $\frac{\partial^2 u}{\partial z^2}(z)$ is the curvature at position z along the blade. Substituting Equation 3.1 into 3.3, and noting that α_i is invariant to transverse displacement, the expression for $M(z)$ becomes:

$$M(t,z) = EI(z)\sum_{i=1}^{\infty}\alpha_i(t)\frac{\partial^2\gamma_i}{\partial z^2}(z) \tag{3.4}$$

Now assume only the first flapwise mode is relevant and the contribution from all higher modes is negligible. That is, assume:

$$u(t,z) \approx \alpha_1(t)\gamma_1(z) \tag{3.5}$$

and

$$M(t,z) = EI(z)\alpha_1(t)\frac{\partial^2\gamma_1}{\partial z^2}(z) \tag{3.6}$$

From Equation 3.4, the tip deflection $y(t) = u(t,L)$ and the root bending moment, $M_R(t) = M(t,0)$, can be related by:

$$M(t) = EI(0)\left.\frac{\partial^2\gamma_1}{\partial z^2}\right|_{z=0} y(t) \tag{3.7}$$

Or put another way, with redefined variable names for brevity, the proportionality between tip deflection and root bending moment is:

$$\frac{y(t)}{M_R(t)} = \frac{1}{EI_0\kappa_0} \tag{3.8}$$

where $\kappa_0 = \left.\frac{\partial^2\gamma_1}{\partial z^2}\right|_{z=0}$ and $EI_0 = EI(0)$

It can be seen in Equation 3.8 that the relationship between tip deflection and RBM does not depend on the deflection amplitude, $\alpha_1(t)$.

From the specifications of the DTU10MW turbine, this proportionality constant can be approximated. The first modal shape is determined using the eigenvalue solver in HAWC2, which provides $\gamma_i(z)$. A spline is fit to $\gamma_1(z)$, which is then numerically integrated twice and evaluated at $z = 0$ to get the curvature at the root. Using the parameters in Table 3.1 to evaluate Equation 3.8, the proportionality between tip deflection and RBM is estimated to be $y(t)/M_R(t) = 3.827 \times 10^{-7} m/Nm$.

Table 3.1: Flexural rigidity and curvature values at root of DTU 10MW blades.

Parameter	**Value**
EI_0	$6.101 \times 10^{10} Nm^2$
κ_0	$4.2827 \times 10^{-5} m^{-2}$

This theoretical evaluation of Equation 3.8 is verified against time series data generated by high fidelity simulations. Time series data for tip deflection and blade root bending moment were generated using HAWC2 over a range of windspeeds from 4m/s to 26m/s, using the normal turbulence model defined in IEC 61400. For each wind speed, 30 minutes of simulation data is collected with a sampling frequency of 100Hz.

To demonstrate and quantify the proportionality, $y(t)/M_R(t)$, flapwise tip deflection and flapwise root bending moment data points for all three blades are binned and plotted in 2D histogram in Figure 3.1. Darker regions indicate a larger frequency of occurrence of data points. Linear regression is performed on this data, and the fitted line is represented as the red dashed line. The tip deflection and RBM data is centered about the mean in order to correspond to the theoretical analysis above.

The plot shows a strong correlation between the two measurements. Two key observations can be made from the analysis of Figure 3.1. First, the simulated results show linear proportionality between tip deflection and RBM, where the proportionality constant remains at a similar value (3×10^{-7} to $3.5 \times 10^{-7} m/Nm$) as the wind speed changes. The constant proportionality is supported by the Euler-Bernoulli cantilever analysis above, however it is noted that the simulated results have a lower proportionality constant compared to the predicted value of $y(t)/M_R(t) = 3.827 \times 10^{-7} m/Nm$. This is likely due to the dynamic nature of the simulation, whereas the theoretical analysis considers a statically loaded blade. Secondly, the coefficient of determination (R^2) represents how well the linear regression explains the data points, where a value close to 1 indicates a perfect fit. The coefficient of determination of the experimental fit shows a high similarity between tip deflection and RBM for low wind speeds (0.92), and a decrease in similarity for higher wind speeds (0.84). This is visually indicated by the spreading of the cloud of data points relative to the line. Higher wind speeds tend to have more spread data points, indicating that tip deflection measurements do not completely explain the variations in RBM measurements, likely due to the increased turbulence at these wind speeds. Despite this, the correlation suggests that an IPC which reduces tip deflection fluctuations will also show a reduction in the blade root bending moment.

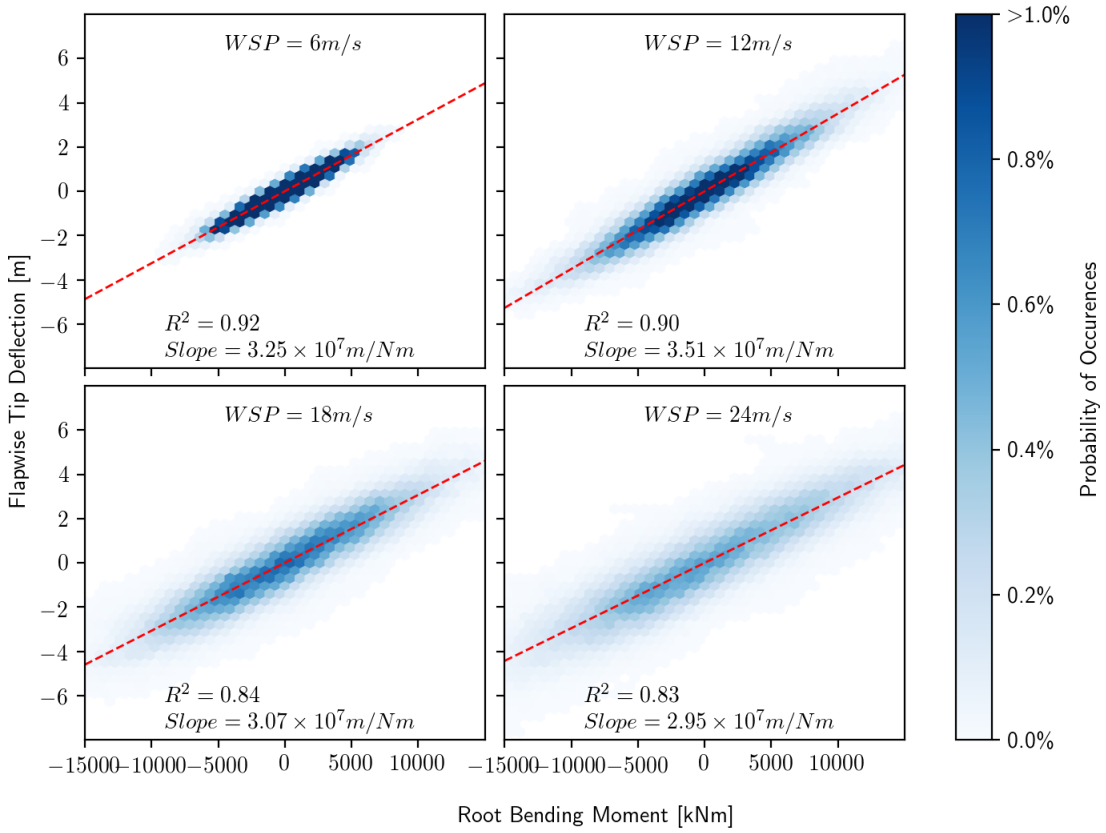


Figure 3.1: Tip deflection vs RBM correlation analysis. Based on HAWC2 simulations of the DTU 10MW Reference Turbine (NTM).

Another way of observing the correlation of tip deflection and RBM is in a time series plot (Figure 3.2). A clear periodicity between the measurements can be observed, which is a result of the oscillating azimuth-dependent loads. To better demonstrate the periodicity, the tip deflection and RBM are bin plotted against

rotor azimuth angle in Figure 3.3, where darker regions indicate a higher level of occurrence. The dominating f_{1p} disturbances are clearly observed as each blade follows an almost sinusoidal distribution for tip deflection and RBM. As expected, each blade measurement is offset by 120^o due to three-fold rotational symmetry of the rotor.

Although the f_{1p} oscillation can clearly be seen in Figure 3.3, there are in fact higher order harmonic components. To better show this, the tip deflection of the blade is represented in the frequency domain. Figure 3.4 shows the power spectral density (PSD) of the tip deflection at different wind speeds. A significant observation is the similarity between the PSD above rated wind speed (11.4m/s). This suggests a single controller targeting f_{1p} should be adequate in the full load region, but insufficient below rated due to the shift in the f_{1p} frequency and its harmonics. The figure confirms the observation in Figure 3.3 that f_{1p} fluctuations dominate, however, it can also be seen that harmonics of f_{1p} are also present in the signal despite being less significant. This indicates that attenuating tip deflection oscillations at multiples of f_{1p} should be adequate in reducing flapwise blade loads.

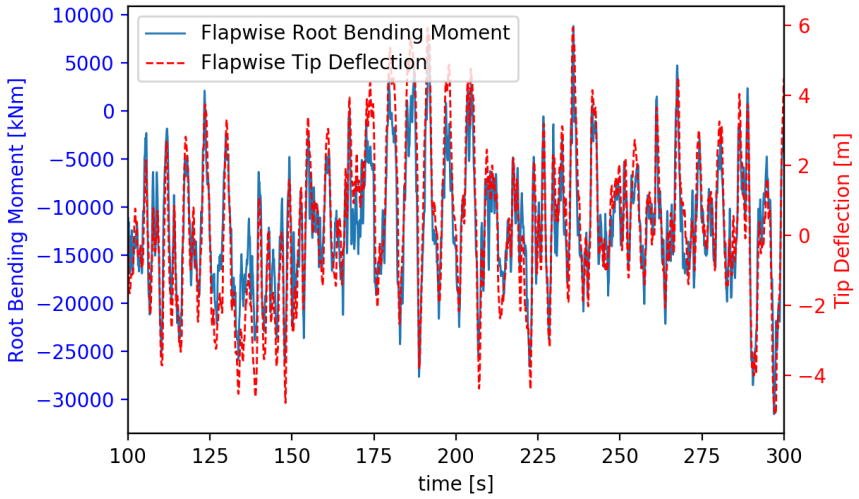


Figure 3.2: Example time series of HAWC2 simulation showing root bending moment and tip deflection. ($U = 6m/s$, NTM)

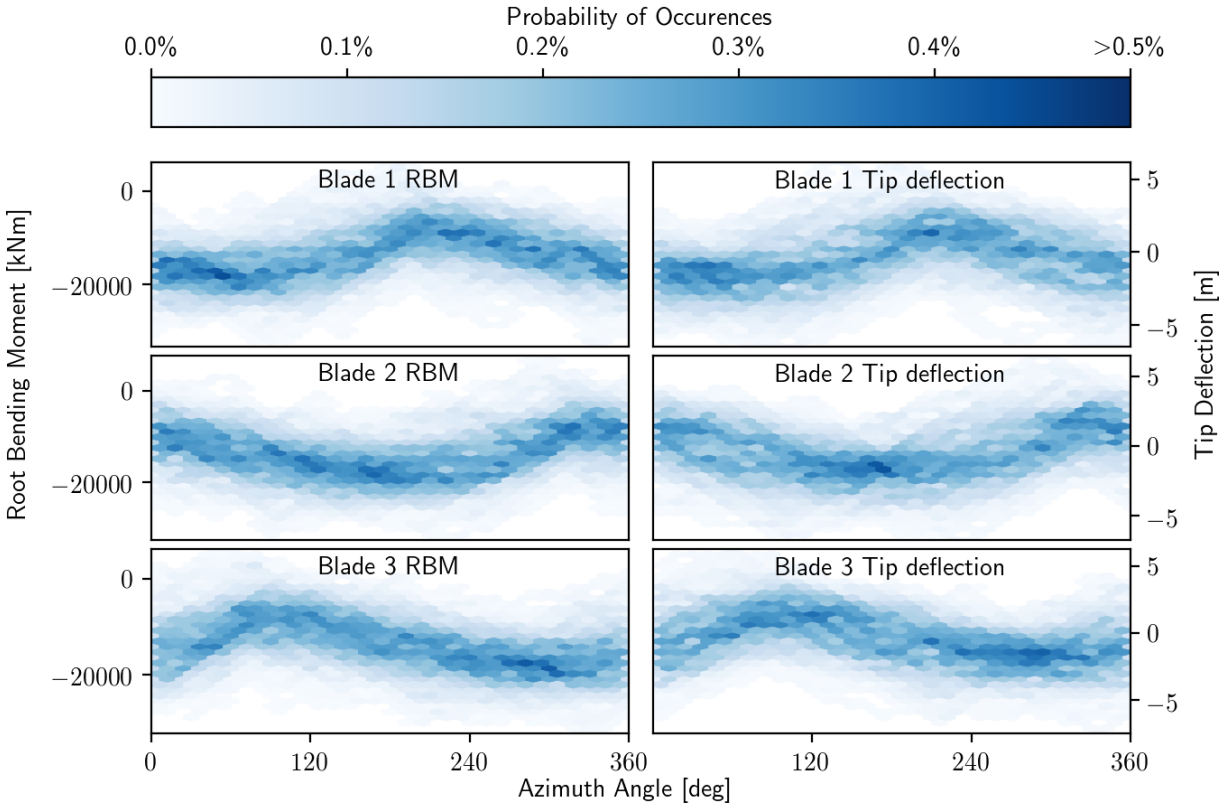


Figure 3.3: Tip deflection, RBM versus azimuth angle ($U = 18m/s$, NTM)

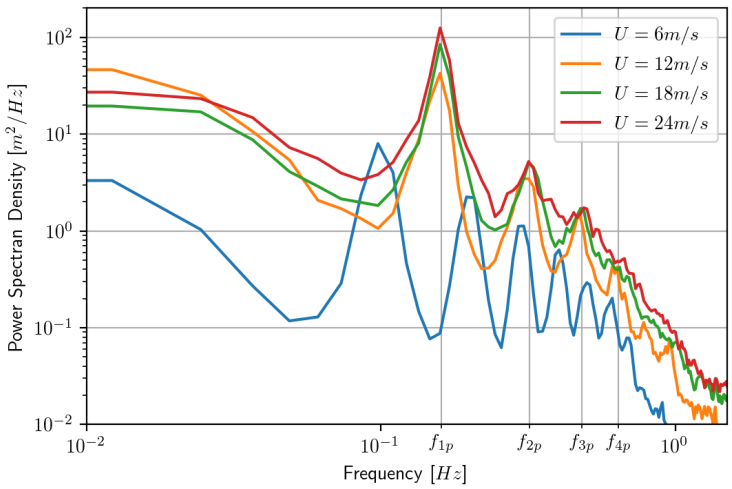


Figure 3.4: Tip deflection power spectral density under normal operating conditions.

The key observations from this section are as follows:

- Flapwise tip deflection and RBM are correlated, and therefore a reduction in tip deflection fluctuations should show a reduction in RBM fluctuations.

- The energy of tip deflection fluctuations is concentrated at harmonics of f_{1p}, and these frequencies should be targeted in order to effectively reduce the fluctuating blade loads.
- The power spectrum retains a similar shape above rated wind speed, suggesting that a single controller could be designed for this operating region.

3.2 Quantifying Fatigue Loads

The turbine simulations produce complicated time histories of the loads experienced in the various components of the turbine. In order to quantify and compare the fatigue damage experienced in these components, a load spectrum is calculated. A load spectrum decomposes a complicated stress history into stress cycles of varying amplitude. This can be achieved using rainflow counting [20]. The fatigue loads in the turbine components can be quantified by calculating the 1Hz equivalent load. The equivalent load is the amplitude of a 1Hz oscillating load which produces the same amount of fatigue damage to a component as a mixed load spectrum. It is a way of comparing different load spectra of the same component. The short term equivalent load, L_{eq}, for a given hub height wind speed, U, is calculated for N_{eq} cycles as follows:

$$L_{eq}(U, N_{eq}) = \left(\frac{\sum_i N_i S_i(U, N_{eq})^m}{N_{eq}} \right)^{\frac{1}{m}} \tag{3.9}$$

where $S_i(U, N_i)$ is the ith load cycle amplitude, N_i is number of full cycles at S_i, m is the material Wöhler curve exponent of the component in question. In this analysis, a Wöhler curve exponent of 4 and 10 is used for steel (tower, shaft, etc) and composite materials (ie. blades) respectively. N_{eq} is the number of cycles experienced of load L_{eq}. For 600 second simulations, which is the case for this analysis, A 1Hz equivalent load requires $N_{eq} = 600$. It can be observed from Equation 3.9 that the damage equivalent load is highly sensitive to large amplitude oscillations due to the exponentiation of the Wöhler curve exponent. Therefore even a small reduction in load amplitudes can lead to a large reduction in lifetime fatigue loads in the turbine components.

Another way of quantifying equivalent loads is the 1Hz lifetime equivalent loads, L_{lt}, which are determined from the short term L_{eq} by integrating:

$$L_{lt} = \int_{U_{in}}^{U_{out}} L_{eq}(U, N_{eq}) p(U) dU \tag{3.10}$$

where U_{in} and U_{out} are the cut-in or cut-out wind speeds, and $p(U)$ is the probability density function for the occurrence of a hub height wind speed. The advantage of using the 1Hz lifetime equivalent loads is that the expected distribution of wind speeds is also taken into account, giving a better estimate of the fatigue damage experienced over the lifetime of the turbine. As the project presented in this report is not site specific, $p(U)$ is defined to follow a Weibull distribution fitted for the specifications of a Class I wind turbine as defined in IEC 61400 [18], shown in figure 3.5.

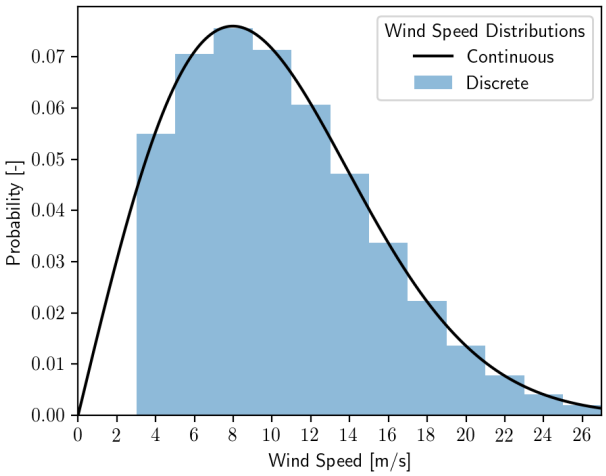


Figure 3.5: Class I Weibull distribution ($k = 2$).

3.3 Disturbance Rejection Theory

The focus of this project is on integrating flapwise tip deflection sensors into an individual pitch control system. Unlike the collective pitch controller, which aims to minimise the error between the power output/torque and a rated power/torque, the task of an IPC is to reject tip deflection perturbations. Consider the linear feedback system in Figure 3.6. The system to be controlled, also known as the plant in control theory, is represented as $P(s)$. The output, $\tilde{y}(s)$, is passed through a controller transfer function $C(s)$ which is to be designed to achieve a certain objective. In Chapter 4, it is shown that the flapwise blade system can be modelled in this form. Frequency components of the disturbance signal, $d(s)$ are passed through to the output, $\tilde{y}(s)$ described by the closed loop transfer function. The closed loop transfer function reveals important characteristics of how the system behaves. A number of different transfer functions made up of $P(s)$ and $C(s)$ are referred to in this report, each with a different application and purpose. These are described as follows:

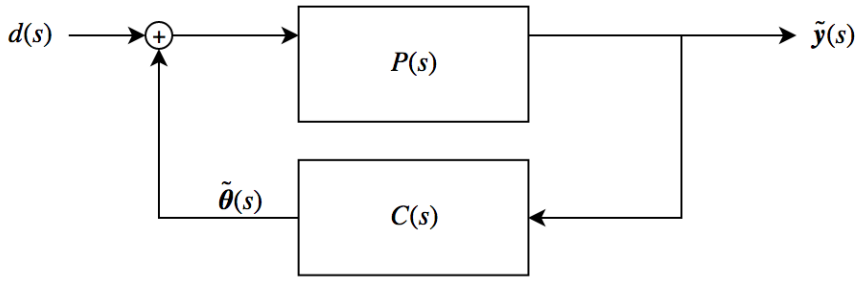


Figure 3.6: Wind turbine system block diagram.

Open Loop Transfer Function (Plant), $P(s)$

$P(s)$ represents the system without any feedback control. $P(s)$ is estimated in Section 4.2. Knowledge of $P(s)$ is required to design an effective controller, and can be used to measure the closed loop system performance.

Closed Loop Transfer Function, $G_{\mathrm{CL}}(s)$

$$G_{CL}(s) = \frac{P(s)}{1 + P(s)C(s)} \tag{3.11}$$

$G_{CL}(s)$ represents the transfer function of the closed loop. It describes the behaviour of the disturbance on the output when the controller is connected in feedback, and can help estimate the controller performance. From Equation 3.11, it can be seen that disturbances are attenuated when the denominator of $G_{CL}(s)$ is large, and disturbances are passed through to the output when the denominator is small

Loop Transfer Function, $L(s)$

$$L(s) = P(s)C(s) \tag{3.12}$$

The loop transfer function is a component of the denominator of $G_{CL}(s)$. Analysing $L(s)$ provides important information about the stability and robustness of the controller. In particular, it can be seen in Equation 3.11 that $G_{CL}(s)$ is unstable when $L(s) = -1$, and therefore the controller should be designed to avoid this condition.

Sensitivity Function, $S(s)$

$$S(s) = \frac{1}{1 + P(s)C(s)} \tag{3.13}$$

The sensitivity function describes how sensitive the closed-loop system is to disturbances in the frequency domain. For any frequency component, the system attenuates when $|S(s)| < 1$ and amplifies when $|S(s)| > 1$.

Complementary Sensitivity Function, $T(s)$

$$T(s) = F(s)\frac{P(s)C(s)}{1 + P(s)C(s)} \tag{3.14}$$

The complementary sensitivity function describes how the system output responds to a reference input. It is based on a different system block diagram introduced later in Figure 6.1. It consists of an additional transfer function block, $F(s)$, which is a precompensator for the reference signal. The use of this transfer function is described in further detail in Chapter 6.

3.3.1 Performance Measures for Disturbance Rejection

The objective of the controller is to attenuate the effect of disturbances, $d(s)$ on the system output, $\tilde{y}(s)$ (see Figure 3.6). One way of measuring the effectiveness of a disturbance rejection control system is to analyse the magnitude of the sensitivity function, $S(s)$.

The sensitivity function is simply the closed loop transfer function (Equation 3.11) divided by the plant, $P(s)$. By looking at the magnitude of the sensitivity function, the level of attenuation or amplification of the closed loop system compared to the open loop system can be determined. Furthermore, if it is assumed that the disturbance spectrum is the same for the open and closed loop system, then the level of attenuation or amplification does not require knowledge of the input signal at all. Instead, it is sufficient to estimate the close loop output spectrum if the open loop output spectrum and the sensitivity function is known. This conjecture is demonstrated below:

Consider an open loop system, $P(s)$ with input $d(s)$ and output $Y_{OL}(s)$, and the closed loop system,

$P(s)/(1+P(s)C(s))$ with the same input, $d(s)$, and output $Y_{CL}(s)$. The magnitudes of $Y_{OL}(s)$ and $Y_{CL}(s)$ are expressed as:

$$|Y_{OL}(s)| = |P(s)||d(s)| \tag{3.15}$$

$$|Y_{CL}(s)| = |\frac{P(s)}{1+P(s)C(s)}||d(s)| \tag{3.16}$$

Dividing Equation 3.16 by 3.15 gives an alternative expression for the sensitivity function:

$$\frac{|Y_{CL}|}{|Y_{OL}|} = \frac{1}{|1+PC|} = |S| \tag{3.17}$$

Therefore, one is able to estimate the closed loop output, $Y_{CL}(s)$ for a given $P(s)$, $C(s)$ and $Y_{OL}(s)$, and this is possible without knowing the disturbance spectrum $d(s)$. Additionally, the level of amplification or attenuation is determined by the magnitude of the sensitivity function. A naive approach of designing a disturbance rejection controller would be to have low sensitivity over a wide bandwidth. However, there are theoretical limitations in doing so. Bode's Integral Formula is a Theorem outlining a fundamental constraint in tuning the sensitivity function of a control system.

Theorem 1 *(Bode's integral formula). Assume that the loop transfer function $L(s)$ of a feedback system goes to zero faster than $1/s$ as $s \to \infty$, and let $S(s) = 1/(1+L(s))$ be the sensitivity function. If the loop transfer function has no poles in the right half-plane, then the sensitivity function satisfies the following integral [21]:*

$$\int_0^\infty log|S(i\omega)|d\omega = 0$$

This is essentially a conservation law of the area under the sensitivity function. If a system attenuates a signal at a certain frequency, it must amplify the signal at another. A trade-off must therefore be made between disturbance attenuation and amplification. This can be taken into account be only attenuating short bands of the signal at f_{1p}, f_{2p} etc, and amplifying frequencies with a small disturbance contribution.

The performance of a controller in this project is quantified by determining the percent reduction or increase in f_{1p}, f_{2p}, f_{3p} and f_{4p} frequency components of tip deflection compared to the open loop system. Given $P(s)$ and $C(s)$, this can be determined by evaluating $|S(s)|$ at these frequencies. Although looking at these four values gives insight into how the controller performs, it is not the whole picture. It does not take into account stability and robustness of the system which is explored further in the following section. In Section 5.4, the effectiveness of attenuating each of these frequencies individually is explored to see which frequencies provide the greatest fatigue load reduction.

3.3.2 Robustness Measures for Disturbance Rejection

As well as achieving performance specifications, the closed loop system must be stable. Stability can be inferred in a number of ways. One such method is to ensure that the closed loop system, $G_{CL}(s)$, has no poles in the right hand plane of the s-plane. Another method is to analyze the bode plot of the loop transfer function to determine the gain and phase margins. An equivalent way of determining stability is to count the encirclements on a Nyquist plot. A Nyquist plot represents $L(s)$ as a 2D contour where the real and imaginary components are plotted on the X and Y axis respectively. The Nyquist stability criterion is elaborated in [21], and is as follows.

Theorem 2 *(Simplified Nyquist criterion). Let $L(s)$ be the loop transfer function for a negative feedback system and assume that $L(s)$ has no poles in the closed right half-plane ($Res \geq 0$) except for single poles on the imaginary axis. Then the closed loop system is stable if and only if the closed contour given by $\Omega = \{L(i\omega) : -\infty < \omega < \infty\} :\subset \mathbb{C}$ has no net encirclements of the critical point, $s = -1$ [21].*

In other words, as long as the Nyquist contour does not encircle the critical point, $s = -1$, then the feed back control system is stable. It is also important to ensure the controller is robust to account for uncertainty in the modelling. Robustness of a system is quantified in this project by determining the stability margin, s_m, as defined in [21]. s_m is the minimum distance between the Nyquist plot and the critical point at $s = -1$. A system is considered more robust for larger values of s_m, where s_m can range from 0 to 1.

Applications of the stability margin are presented in Chapters 5 and 6.

3.4 Blade System Transformations

As IPC controls all three turbine blades simultaneously, it is common for a system transformation to be performed in order to simplify the control system. In literature, typically the Coleman transformation is used in IPC [3], however, in this project, single blade control is used instead. Single blade control assumes all three blades are independent from one another and requires a separate controller for each of blades. This method has the advantage of being simple to implement, assumes each blade is uncoupled, and does not require each blade to communicate with each other. For a three bladed turbine, three identical single-input single-output (SISO) controllers can be cascaded to determine the pitch demands (Figure 3.7). Additionally, the controller does not require the rotor azimuth angle as a measurement. Single blade control requires the input signal to be centred at the mean such that only perturbation measurements are used [19]. Single blade control has been successfully implemented in simulation in [22] and [4].

The use of single blade control over Coleman transform-based control is justified on a theoretical basis. The Coleman transform transforms the stresses or tip deflection measurements from the rotating frame of reference to the stationary frame of reference. When used in a control system, the measurements for the three blades are decomposed into a tilt and yaw component, where the control action is performed. Coleman transform-based control typically assumes the tilt and yaw axes are decoupled, which can cause problems in the control design. Lu provides a mathematical formulation of the tilt-yaw coupling, showing that the assumption that tilt and yaw are decoupled does not hold in certain scenarios [8] and therefore requires further attention in the control design process. Furthermore, the transformation itself is nonlinear, causing a frequency shift in the transformed domain. In particular, the f_{1p} blade loads are shifted to f_{0p} and f_{2p} in the fixed frame. Similarly, f_{3p} oscillations in the fixed frame manifest themselves as f_{2p} and f_{4p} oscillations in the rotating frame [8, 9, 23]. These effects can cause poor control performance if they are not taken into account in the design process.

To overcome these issues, single blade control is performed instead. It is shown in [9] that a single blade control law can be converted to an equivalent Coleman transform-based control law, and that these equivalent controllers yield identical performance. The advantage of designing the controller in this way is that the converted single blade controller already takes into account the effects tilt-yaw coupling, which is often overlooked in Coleman transform-based control design. For this reason, it is chosen to explore the single blade control which treats each blade as an independent system. Careful attention is paid towards the effects of different frequencies on blade loads in Section 5.4.

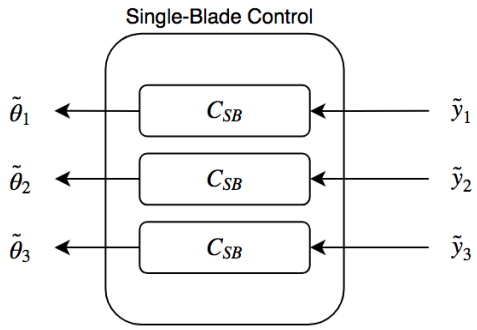
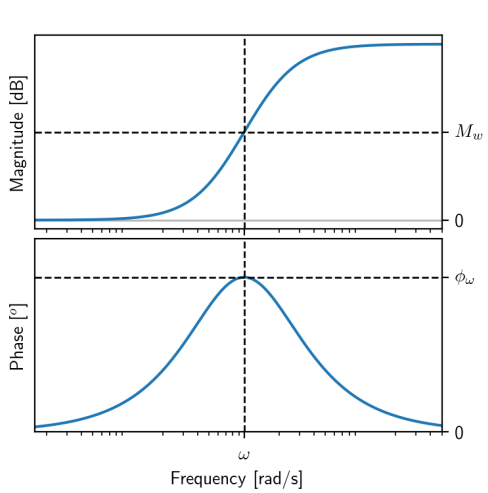


Figure 3.7: Single blade transform block diagram. The pitch demand for each blade, $\tilde{\theta}_i$, is determined only from the measurement of that blade, $\tilde{y}_i$.

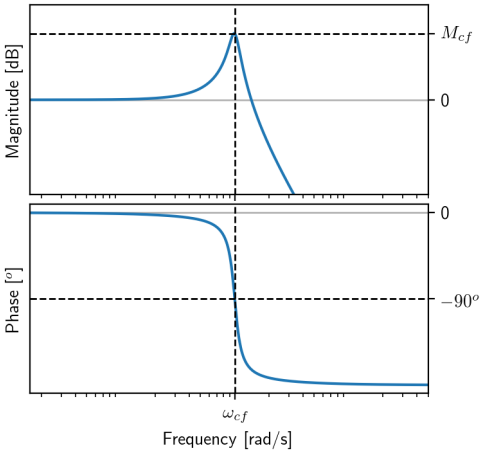
3.5 Filter Design Using Loop Shaping

The controller is designed using loop shaping, which involves adjusting the frequency response of the close-loop systems to achieve certain control objectives. In the case of this project, the close-loop transfer function is desired to have attenuation at harmonics of f_{1p}, while passing low frequency signals. The frequency response of the entire control loop can be adjusted by designing the control transfer function, $C(s)$ to have the required frequency response.

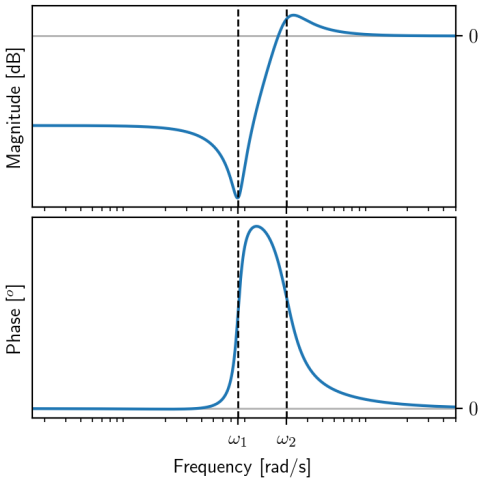
A common method for doing this is called $\mathcal{H}_\infty$ loop shaping. $\mathcal{H}_\infty$ loop shaping is an efficient process of shaping a robust control loop through optimisation. More information of this method can be found in [24]. For the sake of transparency, the controllers in this project are shaped manually instead of using $\mathcal{H}_\infty$ optimisation. At a fundamental level, this involves placing the poles and zeros of the transfer function. To represent this in a more structured way, three basic transfer functions, shown in Figure 3.8, are used to make up the controller transfer functions designed in Chapter 5. These transfer functions are detailed below.



(a) Lead compensator $G(s) = \frac{Tas+1}{as+1}$



(b) Low pass filter $G(s) = \frac{\omega_{cf}^2}{s^2+2\zeta\omega_{cf}+\omega_{cf}^2}$



(c) Band pass filter $G(s) = \frac{s^2+2\zeta_1\omega_1+\omega_1^2}{s^2+2\zeta_2\omega_2+\omega_2^2}$

Figure 3.8: Transfer function building blocks used to construct the controller in this project.

Lead compensator

A lead compensator provides a phase lead angle over a band of frequencies. It was found that adding lead compensators provided better robustness by overcoming the many sources of lag in the system. Lead compensators are able to reduce the lag of a system without introducing as much noise amplification as a derivative controller. For a desired phase lead, ϕ_ω at a frequency, ω, the lead compensator shape and time constant, a and T can be found using:

$$a = \frac{1 + sin(\phi_\omega)}{1 - sin(\phi_\omega)} \tag{3.18}$$

$$T = \frac{1}{\omega\sqrt{a}} \tag{3.19}$$

Low pass filter

A second order low pass filter with a low damping ratio ($\zeta \approx 0.05$) is used to target certain frequencies over a short band of frequencies for attenuation, and is unresponsive to high frequencies. This comes at a cost of introducing phase lag into the system which can lead to instability. This can be overcome by cascading lead compensators. The peak magnitude occuring near the cutoff frequency can be designed by setting the damping ratio using the relation, $M_{cf} = 1/2\zeta$.

Band pass filter

A band pass filter passes frequencies over a small range of frequencies and attenuates frequencies outside this range. Unlike the low pass filter, a band pass filter can introduce a phase lead in the system, which helps provide robustness. Multiple cascaded band pass filters are used for targeting multiple frequencies.

3.6 Controller Discretisation

In order to implement the controllers designed in Chapters 5 and 6, the controller transfer functions must be discretised so that they can be implemented in the time domain. For this reason, the continuous controller must be transformed to be implemented as a discrete time controller.

It is not possible to produce a discrete controller which perfectly matches the frequency and time domain performance of its continuous counterpart. For this reason, many methods exist, each with advantages and disadvantages. The impulse invariant method, zero order hold method, and first order hold method are examples of continuous-to-discrete methods which produce discrete systems which exactly match certain time-domain response behaviours of a continuous system, namely the impulse, step, and ramp response respectively. The zero order hold method, for example, is the default method used in MATLAB's `c2d` function. Despite its common use, time-domain methods do not preserve the frequency domain response of a system as well as other methods. As the frequency response of an IPC controller is of high importance, a different discretisation method is sought after.

Two methods were investigated which better preserve the frequency response of the system: the matched-pole-zero approximation and the bilinear transform. Both methods showed comparable frequency responses due to the high sampling frequency of the controller compared to the frequencies of the wind turbine system. However, it was found that the matched-pole-zero approximation encountered numerical instabilities more easily than the bilinear transform.

3.6.1 Matched-Pole-Zero Approximation (MPZ)

The MPZ method involves directly mapping the continuous poles and zeros, s_p and s_z from the s-domain to equivalent locations in the z-domain, z_p and z_z, using the transformation $z_p = e^{s_p T_s}$ and $z_z = e^{s_z T_s}$ where

T_s is the sampling time of the discrete system. Expressed another way, a continuous transfer function of the form:

$$H_a(s) = K_a \frac{\prod_{i=0}^{n}(s - s_{zi})}{\prod_{i=0}^{m}(s - s_{pi})} \tag{3.20}$$

is transformed to a discrete transfer function:

$$H_d(z) = K_d \frac{\prod_{i=0}^{n}(z - e^{s_{pi}T_s})}{\prod_{i=0}^{m}(z - e^{s_{zi}T_s})} \tag{3.21}$$

Additionally, the continuous zeros at infinity are mapped to discrete zeros at $z = -1$, and the gain of the digital transfer function is chosen such that the low frequency gain of both transfer functions is equal. That is:

$$H_a(s)|_{s=0} = H_d(z)|_{z=1} \tag{3.22}$$

$$\Rightarrow K_a \frac{b_0}{a_0} = K_d \frac{\sum b_i}{\sum a_i} \tag{3.23}$$

where b_i and a_i are the controller transfer function numerator and denominator coefficients.

3.6.2 Bilinear Transformation

The bilinear transformation is a first order approximation of the exact mapping between the s and the z domain: $s \leftarrow \frac{1}{T_s} \ln z$. the bilinear transformation is defined as:

$$s \leftarrow \frac{2}{T_s} \frac{z-1}{z+1} \tag{3.24}$$

Substituting Equation 3.24 into 3.20 gives the following expression for discrete transfer function:

$$H_d(z) \approx H_a(s)|_{s=\frac{2}{T_s}\frac{z-1}{z+1}} = \left(\frac{T_s}{2}(z+1)\right)^r \frac{\prod_{i=1}^{m}[(1 - \frac{T_s}{2}s_{zi})z - (1 + \frac{T_s}{2}s_{zi})]}{\prod_{i=1}^{n}[(1 - \frac{T_s}{2}s_{pi})z - (1 + \frac{T_s}{2}s_{pi})]} \tag{3.25}$$

where r is the relative order of the continuous system (number of poles - number of zeros). The purpose of the first term to the rth power is to place r discrete zeros at $z = -1$ to account for the continuous zeros at infinity [25]. As a result, the discrete transfer function always has the same number of poles and zeros.

Both the bilinear transform and the MPZ method preserve stability between the continuous and discrete systems. Although both methods are found to produce similar discretisations, the bilinear transform is more generally used in literature due to the ability to prewarp the response at a particular frequency [26]. It was also observed in this project that the bilinear transform was more successful in mapping high order filters, whereas the MPZ method tends to produce numerical instabilities. Furthermore, the MPZ method was more likely to deviate in the phase response of the filter compared to the bilinear transform, as shown in Figure 3.9, which can lead to unforeseen instabilities in the closed loop system. For this reason, the bilinear transform is used for the discretisation of the continuous controllers designed in Chapter 5.

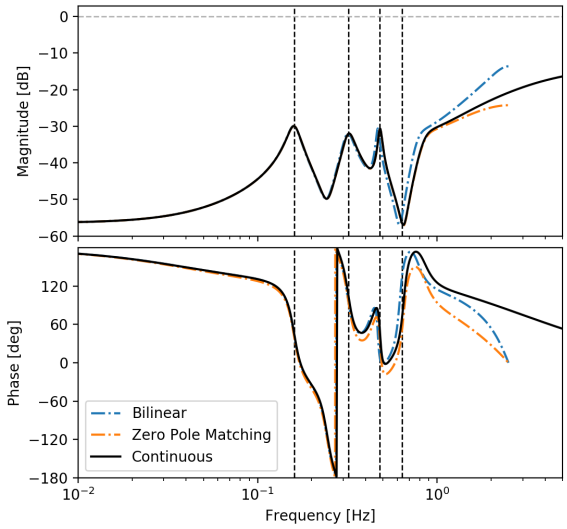


Figure 3.9: Bilinear transformation versus MPZ method for continuous to discrete conversion. The MPZ method can be seen to deviate in the phase response more than the bilinear transform ($f_s = 5Hz$).

3.6.3 Effect of sampling rate on frequency response

The sampling rate of the discrete system should be chosen such that it is high enough to not experience aliasing and frequency response warping while not being too high as to be redundant. The Nyquist-Shannon sampling theorem gives a theoretical boundary stating that the sampling rate should be at least twice the required bandwidth [27]. However, for a better margin of error, a more common and practical guideline is to have a sampling frequency at least ten times greater than the desired bandwidth of the controller [25]. The choice of the controller bandwidth draws from Bergami et al. [28], which shows that the majority of fatigue occurs at frequencies below 2Hz for the NREL 5MW turbine. This frequency limit corresponds to the 10P frequency of the NREL turbine operating at above rated speeds. The corresponding 10P frequency for the DTU10MW turbine used in this project is 1.6Hz. Therefore to be able to account for this full range of load-relevant frequencies as well as having a sufficient amount of room for error, the minimum sampling rate of the sensor should be 16Hz. As an example, the continuous transfer function in Figure 3.10 was discretised using the bilinear transformation with different sampling rates. A sampling rate above 5Hz is seen to be sufficient to address up to f_{4p} with little distortion in the frequency response. frequency of 16Hz would certainly be sufficient for this range of frequencies. In this project, it is assumed that the tip deflection sensor sampling rate is significantly higher than the required bandwidth, and that controller calculations can be performed without delay. To simplify the working code, the sampling rate is set to $100Hz$ in order to match the sampling rate of the HAWC2 simulation. However it should be noted that a lower sampling rate could be used.

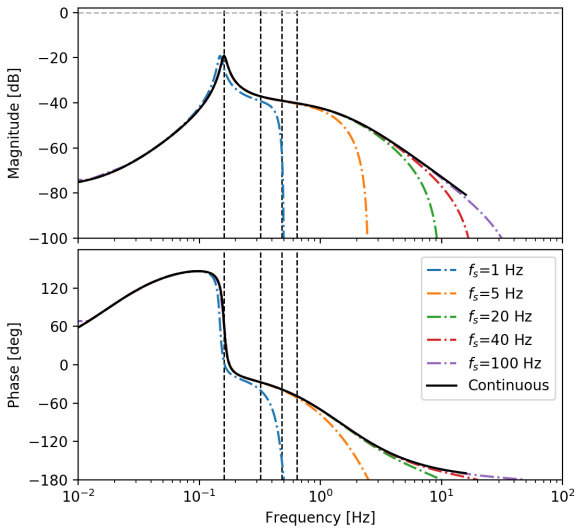


Figure 3.10: Bilinear transform performed for different sampling frequencies. Vertical dotted lines indicate 1p frequencies and its multiples.

3.6.4 Discrete frequency to discrete time domain transformation

In the above sections, methods for transforming a continuous controller transfer function into a discrete controller transfer function is outlined. The final step is to convert the transfer function, which is in the frequency domain, into a time domain function which can be executed and implemented in machine code. For a discrete transfer function of the form:

$$H_d(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1 z^{-1} + ... + b_m z^{-m}}{1 + a_1 z^{-1} + ... + a_n z^{-n}} \tag{3.26}$$

the transformation from the z-domain to the time domain yields the expression:

$$y_k(t) = b_0 x_k(t) + b_1 x_{k-1}(t) + ... + b_m x_{k-m}(t) \tag{3.27}$$

$$- a_1 y_{k-1}(t) - a_2 y_{k-2}(t) - ... - a_n y_{k-n}(t) \tag{3.28}$$

Where $x_k(t)$ and $y_k(t)$ are respectively the discrete input and output signal at the kth time step. It can be seen that the coefficients, b_i and a_j for $i = 0, 1, ...m$ and $j = 0, 1, ..., n$ are the same for both Equations 3.26 and 3.27. This makes it simple to convert a transfer function in the z-domain into the time domain by inspection. Additionally, Equation 3.27 is a linear difference equation which is easily executed in machine code. A control transfer function converted to this form can be executed digitally in a wind turbine.

3.7 Control Algorithm and Properties

In this section, the single blade individual pitch control algorithm outlined. The algorithm follows the form outlined in [19] and has been adapted to use the discrete time execution formulated in this project. The same algorithm is implemented for each blade. A Python implementation of these steps can be found in Appendix B.

1 - Collect measurements

For an individual blade, the discrete tip deflection measurement, $y_k(t)$ as well as the collective pitch control demand is collected. It is assumed that all measurements are obtained at the same time step, k.

2 - Subtract the mean from the measurements

The mean value of the three tip deflection measurements (collected from each blade) is determined and subtracted from each signal to produce a tip deflection perturbation.

$$\tilde{y}_k^i(t) = y_k^i(t) - \bar{y}_k(t) \tag{3.29}$$

for each blade $i = 1, 2, 3$ and where

$$\bar{y}_k(t) = \frac{y_k^1(t) + y_k^2(t) + y_k^3(t)}{3} \tag{3.30}$$

3 - Calculate the pitch demand perturbations, $\tilde{\theta}_k(t)$

The pitch demand perturbation, $\tilde{\theta}_k(t)$, for the blade is a linear combination of new tip deflection measurement as well as the the previous $N-1$ tip deflection measurements and pitch demands:

$$\tilde{\theta}_k(t) = \sum_{j=0}^{N-1} b_j \tilde{y}_{k-j}(t) - \sum_{j=1}^{N-1} a_j \tilde{\theta}_{k-j}(t) \tag{3.31}$$

The constant feed forward and feed-backward coefficients, b_j and a_j are the same as the discretised controller coefficients are found using the method in section 3.6. It is assumed that a and b have the same number of elements. In this project, the bilinear transformation is used for discretisation which enforces that a and b are the same length.

4 - Superimpose IPC signal over CPC signal

The overall blade pitch demand for the blade, $\theta_k(t)$, can then be calculated as:

$$\theta_k(t) = \tilde{\theta}_k(t) + \bar{\theta}_k(t) \tag{3.32}$$

where $\bar{\theta}_k(t)$ is the CPC pitch demand. It is assumed that the calculations can be returned to the pitch actuator within the current time step.

Chapter 4

System Identification of Blade System

This chapter provides details of the linear model used for the wind turbine blades. This is essentially deriving a model for the plant system, $P(s)$, described in Section 3.3, and is achieved using system identification methods. Additionally, details are also provided of how the open loop output spectrum is obtained. A simple analytical turbine model is presented to describe the interaction between blade pitching, flapwise tip deflection and all external disturbances. The linear model is used to estimate the performance and robustness of different controllers. In Chapter 5, the controllers are then run using the high-fidelity HAWC2 turbine model and a comparison is made with the linear model in the following section.

4.1 Background

In order to effectively design an IPC, the wind turbine dynamics must be modelled. A block diagram of a wind turbine system with both CPC and IPC is shown in Figure 4.1. The flapwise tip deflection of the ith blade, $y_i(t)$, is considered an output of this system. The two relevant inputs for system are the blade pitch angle, $\theta_i(t)$, and all other disturbances which influence tip deflection, which is encapsulated in the input term $d(t)$. $d(t)$ represents disturbances from wind shear, tower shadow, yaw misalignment, turbulence and sensor measurements. Although it may seem over-simplistic to encapsulate these disturbances into a single signal, it is shown in the next chapter that the majority of the energy of this disturbance is concentrated at f_{1p} and its harmonics. As described in Section 3.3, $d(t)$ does not need to be directly quantified in order to proceed with the control design. The collective pitch control loop is presented in this figure to show how a typical wind turbine controls pitch angle to achieve a desired power output.

Whereas the CPC responds to changing rotor speed, the IPC responds to the flapwise tip deflection of all three blades. Note how the total pitch demand is a superposition of the collective and individual pitch control loops. This control architecture is common in literature, and is justified by the fact that the bandwidth of the CPC is much lower than the IPC loops, and is therefore essential decoupled [7].

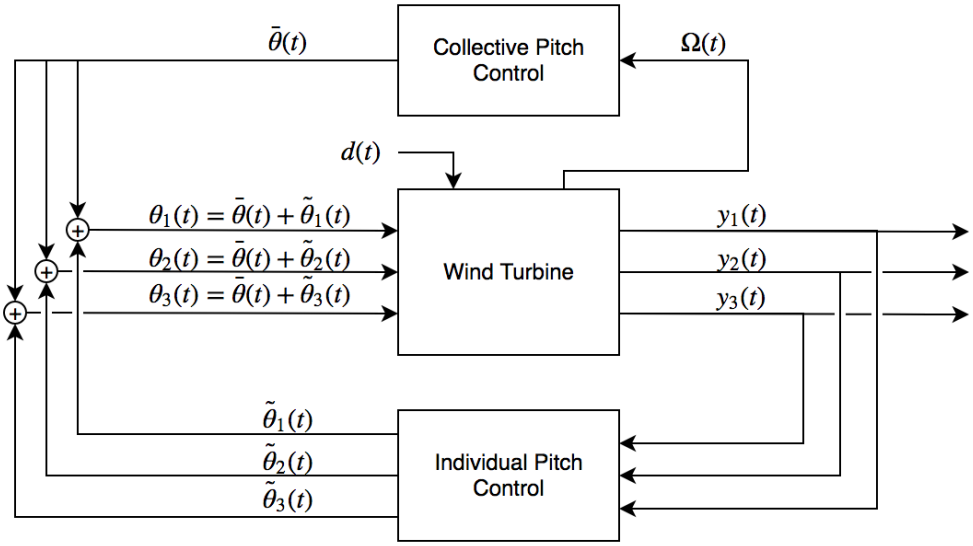


Figure 4.1: Wind turbine system block diagram with IPC control block.

Due to the highly nonlinear nature of the wind turbine system, the models presented in literature vary greatly. It is found in this project that a linear approximation of the turbine system is adequate in designing a controller that provides load reductions using tip deflection inputs. This assumption is supported in literature for IPC systems using strain gauge sensors [3]. By encapsulating and linearising the wind turbine model as well as the CPC loop about a fixed mean rotor speed, tip deflection and blade pitch angle, the block diagram can be split into a plant block, $P(s)$ and a controller block, $C(s)$, with a disturbance input, $d(s)$, and tip deflection of each blade in the vector $\tilde{y}(s)$. Note an additional assumption is put in place in this block diagram. Namely, the disturbance signal and pitch demand signal are additive. Therefore, the effect of any disturbance signal can be rejected by providing an appropriate pitch demand signal.

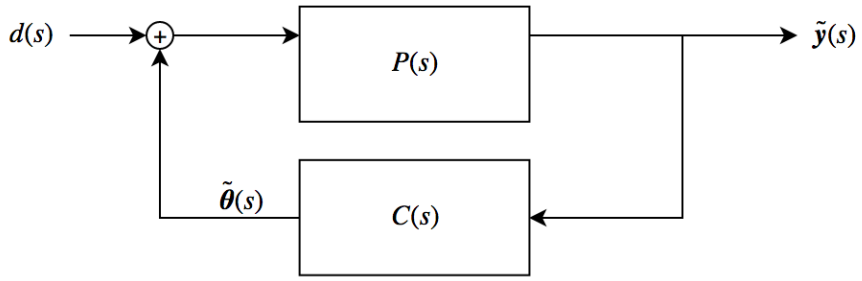


Figure 4.2: Linearised wind turbine block diagram.

The total system is simplified to a single linear block in Figure 4.3. The transfer function between the disturbances and the tip deflection output now match the disturbance rejection model outlined in Section 3.3. In order to use this model to design a controller, the plant, $P(s)$, shall be estimated using system identification.

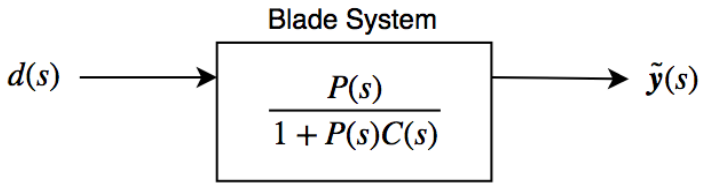


Figure 4.3: Simplified linear wind turbine block diagram.

4.2 System Identification of Plant

The dynamics describing how a changing blade pitch angle affects the flapwise tip deflection for a single blade, referred to as $P(s)$, is complicated. Referring to Figure 4.4, pitch demands are passed through many dynamic systems including the pitching actuator, the structural dynamics of the blade and the interaction with the aerodynamics of the wind. Each of these systems has their own set of nonlinearities and disturbances. One method of estimating the plant would be to model each of these components individually. However, it is not always possible to determine the parameters of each component, especially in a real world situation. In order to overcome potential errors in the modelling, and to remain applicable to real world situations, a different method called system identification is used.

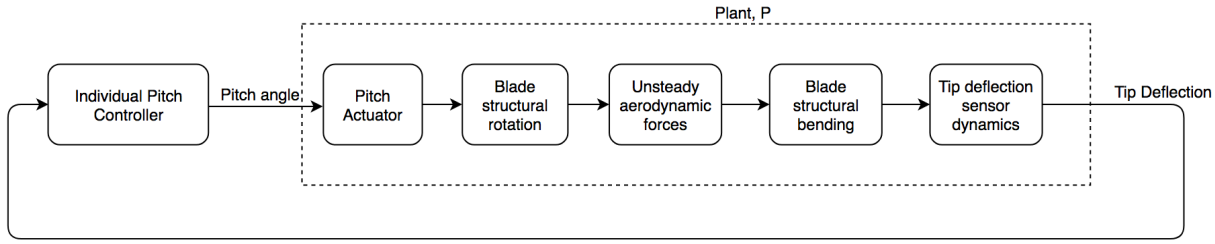


Figure 4.4: Flow chart indicating the systems involved between the transfer of blade pitch signal demands and tip deflection.

System identification is the process of estimating the system dynamics of a black box system only with knowledge of the input and output of the system. One method of performing system identification is to perform a frequency sweep [29]. A frequency sweep involves subjecting the black box system to a sinusoidal input of a particular frequency, and measuring the amplitude and phase response of the output signal. This is repeated for a range of frequencies to estimate the frequency response of the system, from which a transfer function can be fitted. A more efficient way of performing system identification is to subject the system to a step input and to fit a transfer function to the step response of the system. Theoretically, this is valid because a step signal is composed of a wide range frequencies and is therefore typically used in system identification problems [19]. Using the System Identification Toolbox in Matlab, the transfer function of a system can be estimated given an arbitrary input signal and its corresponding output signal.

In this project, the pitch demand of one blade is the input signal, and the flapwise tip deflection is the output, and the transfer function linking these two signals is the unknown system to be found. To isolate the effects of pitching on tip deflection, all components on the turbine are set to stiff apart from the blades. To eliminate disturbances from f_{1p} oscillations, the incoming wind is set to have a constant vertical wind profile with no turbulence, and the nacelle tilt is set to zero. It is assumed that the behaviour of the blade system depends on the turbine operating conditions (rotor speed, pitch angle), and therefore a different transfer function is found for each wind speed, $U = 4, 6, ..., 26m/s$.

Figure 4.5 shows time series data from the HAWC2 simulations used for the system identification at different wind speeds. The tip deflection can be seen to respond only to the step changes in pitch angle, and converge to a constant value at all other times, verifying the simulation is not influenced by other aerodynamic and structural influences.

It should be noted that system identification can only be used for a linear, time invariant system. To verify the assumption of linearity and time invariance, two step inputs are provided at different amplitudes. The first step output is projected (dotted line) and scaled over the second step output. From the figure it can be seen that at above rated wind speed, the step size of the output signal scales with the magnitude of the input step. This supports the underlying assumption that the blade system behaves linearly under small pitch perturbations, and is invariant to the time shift of the step inputs. Below rated wind speed, this linearity does not hold as can be seen by the misfitting projection at $U = 6m/s$. This is likely a result of interactions with the torque control in this region.

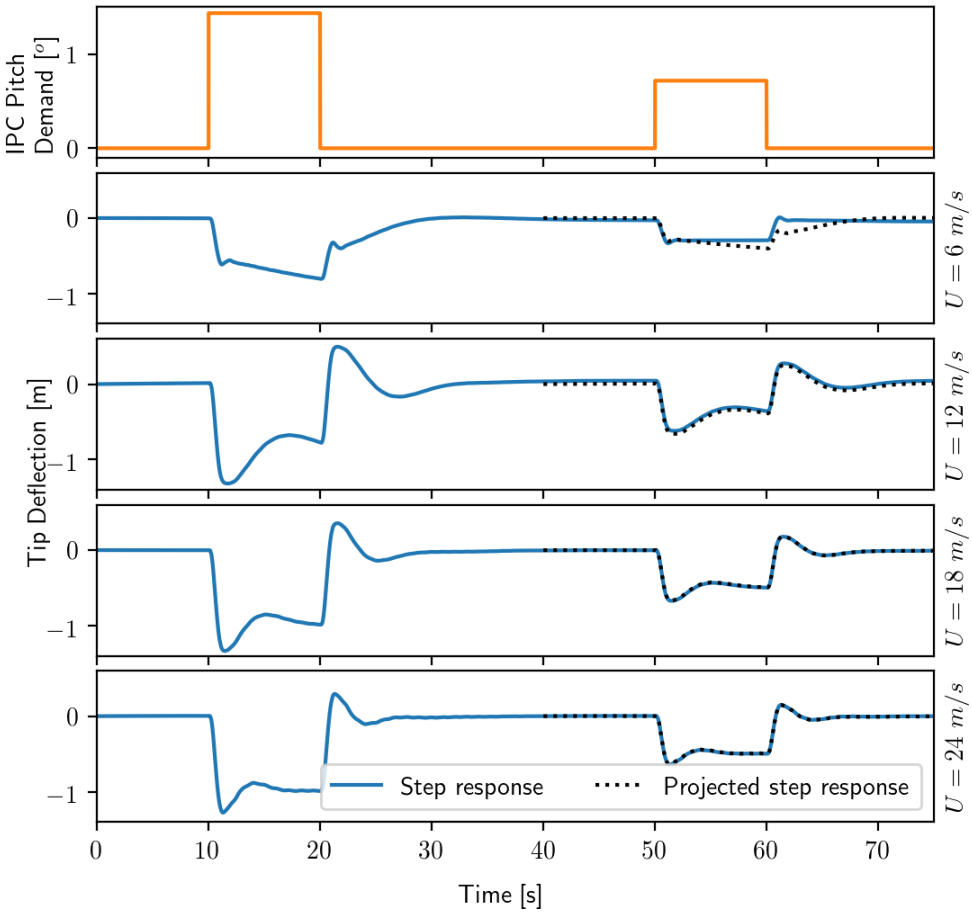


Figure 4.5: Pitch step input and output data. The dotted lines are a projection of the first step response onto the second in order to demonstrate linearity and time invariance.

The transfer function estimation is performed in Matlab using the time series data plotted above. The time series input and output are run through the `spafdr` function (spectral analysis with frequency-dependent resolution) to obtain an estimate for the frequency response of each time series. This allows the system identification to be performed in the frequency domain instead of the time domain. As an IPC is targeting

frequency components, it is important to have a good estimate of the plant in the frequency domain.

The `spafdr` function uses a similar method outlined by Ljung [19]. First, the autospectrum of the discrete input and output signal, $\Phi_{xx}(\omega)$ and $\Phi_{yy}(\omega)$, as well as the cross spectrum, $\Phi_{xy}(\omega)$ is estimated. The frequency response of blackbox system, $G(z)$ is estimated as:

$$G(e^{j\omega}) = \frac{\Phi_{xy}(\omega)}{\Phi_{xx}(\omega)} \tag{4.1}$$

The estimated frequency response is run through the `tfest` function to find the best fitting transfer function coefficients. A transfer function with four poles and three zeros was found to fit most appropriately. Figure 4.6 shows the bode plot corresponding to the fitted plant transfer functions for various wind speeds. The system behaves similar to a low pass filter, implying that high frequency pitching is attenuated due to the system dynamics. It can also be seen that the blade models above rated wind speed closely match, whereas the below rated wind speed models show some deviations. This once again suggests that an IPC suitable above rated will not be as suitable below rated.

The continuous transfer functions are tabulated in Appendix A.

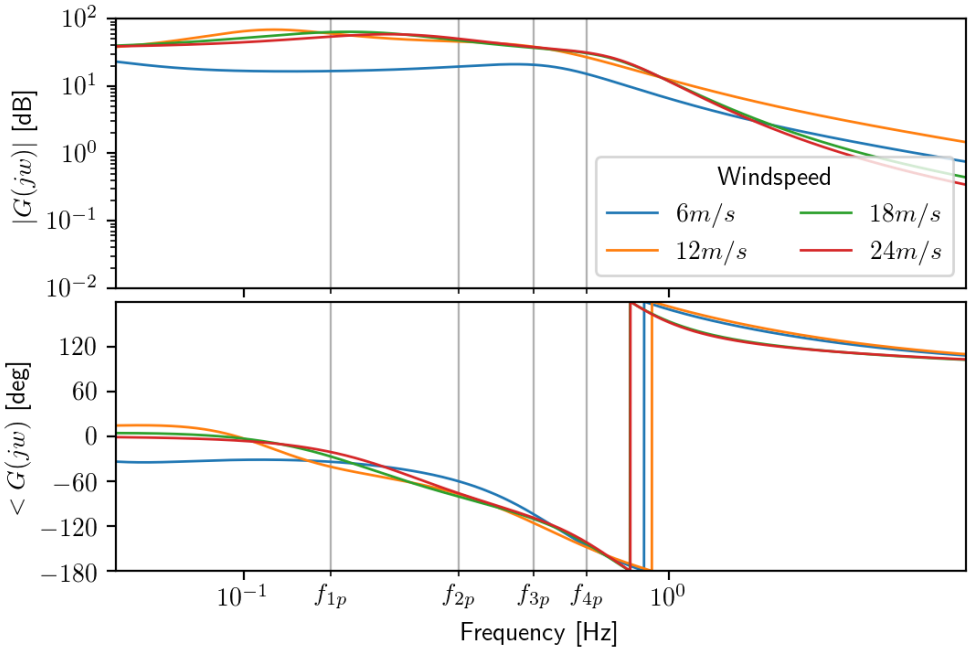


Figure 4.6: Blade transfer functions derived using system identification.

Chapter 5

Tip Disturbance Rejection Controller

In this section, the iterative design process used to design an effective IPC is detailed. This chapter draws from the theory and modelling outlined in the previous chapters. In particular, the plant system, $P(s)$ is used to determine the robustness and linear performance of each controller. This prediction is then compared to high-fidelity simulation results by implementing each IPC in HAWC2.

The chapter is arranged as follows. First, the open loop system output is analysed in Section 5.2. Next, a baseline PI controller is designed in Section 5.3 which is used as a benchmark controller for comparison purposes. In Section 5.4, four single frequency controllers are then designed to specifically target frequencies which are harmonics of f_{1p}. The load reduction capabilities of each of these controllers is investigated. The final control design iteration is carried out in Section 5.5, where a controller which targets f_{1p} and f_{2p} disturbances is designed. Finally, the tip deflection sensor based control is compared to strain gauge based control, and the effect of IPC on power output is investigated.

5.1 Background

The unique aspect of this project is the use of tip deflection measurements in the implementation of IPC. This is contrary to many of the control designs in literature which use strain gauge measurements instead [3, 8, 9, 22, 30]. Nevertheless, the control design process remains similar. Instead of rejecting disturbances in the blade root bending moments, tip deflection disturbances are rejected. Due to the high correlation of the two signals, minimising tip deflection oscillations will also minimise blade loads as outlined in Section 3.1.

The controllers in this section are designed to attenuate frequencies between f_{1p} and f_{4p}. This is achieved by performing loop shaping, where the controller transfer function is designed in the frequency domain (refer to Chapter 3). In literature, loop shaping is typically performed using $\mathcal{H}_\infty$ optimisation to improve robustness and performance ([8], [9], [1] and [10]). For clarity in this section, $\mathcal{H}_\infty$ optimisation is not performed and the controller is tuned manually.

IPC typically operates in conjunction with CPC, so it is important to ensure both controllers do not interfere with each other. This is achieved in two ways. Firstly, the IPC controller ensures that the average pitch of each blade is equal to the CPC pitch demand, ensuring that the power output of the turbine remains unaffected. Secondly, IPC operates at a significantly higher bandwidth than CPC. Therefore the pitch demand of the two controllers can be superimposed without noticeable interference [7].

One consequence of this decoupling that the control algorithm is unable to respond to multiples of f_{3p} frequency components [23]. This is due to the three-fold rotational symmetry of the rotor, and appears in many of the results in this paper (Figures 5.3a, 5.5, 5.6). To demonstrate this analytically, let ω be the

turbine rotor angular frequency. If each blade tip deflection signal is offset by 120^o then a f_{3p} tip deflection signal for all three blades is:

$$y_1(t) = A\sin(3\omega t + \phi) \tag{5.1}$$

$$y_2(t) = A\sin\left(3\left(\omega t + \frac{2\pi}{3}\right) + \phi\right) = A\sin(3\omega t + \phi + 2\pi) \tag{5.2}$$

$$y_3(t) = A\sin\left(3\left(\omega t + \frac{4\pi}{3}\right) + \phi\right) = A\sin(3\omega t + \phi + 4\pi) \tag{5.3}$$

for an amplitude and phase, A and ϕ. Due to periodicity, $y_1(t) = y_2(t) = y_3(t)$, and therefore, centering about the mean gives $\tilde{y}_1 = \tilde{y}_2 = \tilde{y}_3 = 0$. Therefore when applying Equation 3.31 in step 3 of the control algorithm to these inputs, zero IPC pitch action will be produced. This insensitivity at multiples of f_{3p} is observed in Section 5.4. It should also be noted that this phenomenon is not unique to single blade control. The same insensitivity is present for Coleman based control.

An issue with IPC is the increased pitch rate of the blades. As IPC requires the blades to continuously pitch, the rate at which they pitch is significantly higher than CPC. IPC pitch rates required to achieve decent reductions in fatigue load are around ± 10 deg s^{-1}, which is considered quite high [2]. However, the required pitching rate decreases with rotor diameter due to the decrease in rotational frequency. This justifies the use of IPC for larger wind turbine models. Higher order harmonic control may not meet the limiting pitch rate requirements, which explains why most papers consider low frequency oscillations (usually up to 3P) [10].

It is worth mentioning that IPC is just one of many methods for performing disturbance rejection in turbine blades, especially in the higher frequency range where IPC is unable to operate. Active aerodynamic load control (AALC) devices such as trailing edge flaps and micro tabs have also been an active area of research. The advantage of AALC devices is their high bandwidth, allowing for controllability of high frequency dynamics. Berg [12] and Wilson [13] have researched the effects of load reduction using trailing edge flaps, showing a 20-32% reduction in blade root stress, which can allow for a 10% increase in blade length without exceeding the original equivalent fatigue damage. Lackner [31] explores this concept and proposes a hybrid control system using both IPC and active flaps, showing excellent blade load reductions over a larger bandwidth than IPC alone. Bergami performed a similar study in HAWC2, showing 15% reduction in lifetime equivalent loads using active flaps alone, and up to 30% life time equivalent load reductions using both active flaps and pitch control [32].

5.2 Open Loop Output Estimation

In order to compare the performance of the controllers designed in this chapter, results are collected for the open loop system (CPC only). The tip deflection frequency response is of interest for two reasons. First, it can be used to compare open loop and output simulation results. Second, the closed loop frequency response can be estimated with the open loop response by using the theory developed in 3.3.

The open loop output frequency response refers to the frequency decomposition of the flapwise tip deflection for a turbine without tip deflection control. This is found by running simulations using the DTU10MW turbine with full structural flexibility, realistic aerodynamic effects including wind shear, tower shadow and turbulence. A different frequency response is produced for each wind speed. Three simulations of 10 minutes each with different turbulent seeds were run for each wind speed $U = 4, 6, ..., 26m/s$. From these simulations, time series data of blade flapwise tip deflection is collected. A variation of the Welch method is used to generate a smooth frequency response. The one sided Fourier transform is taken of each of these time series. For a given wind speed, the average Fourier transform is taken over all three blades and turbulent seeds to produce an averaged frequency response. Furthermore, each time series is split into six segments of 10000 data points each, from which the Fourier transform is taken and averaged. The result is a smoothed

frequency response as demonstrated in Figure 5.1. The discrete one sided Fourier transform is defined and normalized as:

$$\mathcal{F}\{y[n]\} = \frac{2}{N}\sum_{m=0}^{N-1} y[m]\cdot e^{-2\pi i\frac{mk}{N}} \qquad \text{for } k = 0, 1, ..., \frac{N-1}{2} \tag{5.4}$$

where N is the length of the time series, k/N is the per sample frequency. This particular normalization is chosen so that the frequency component amplitude matches the tip deflection amplitude in meters. The shape of the frequency responses above rated wind speed show a similar shapes to each other, having a large peak at f_{1p}, and smaller peaks at harmonics of this frequency. Below rated wind speed, this peak shifts due to the changing rotor speed.

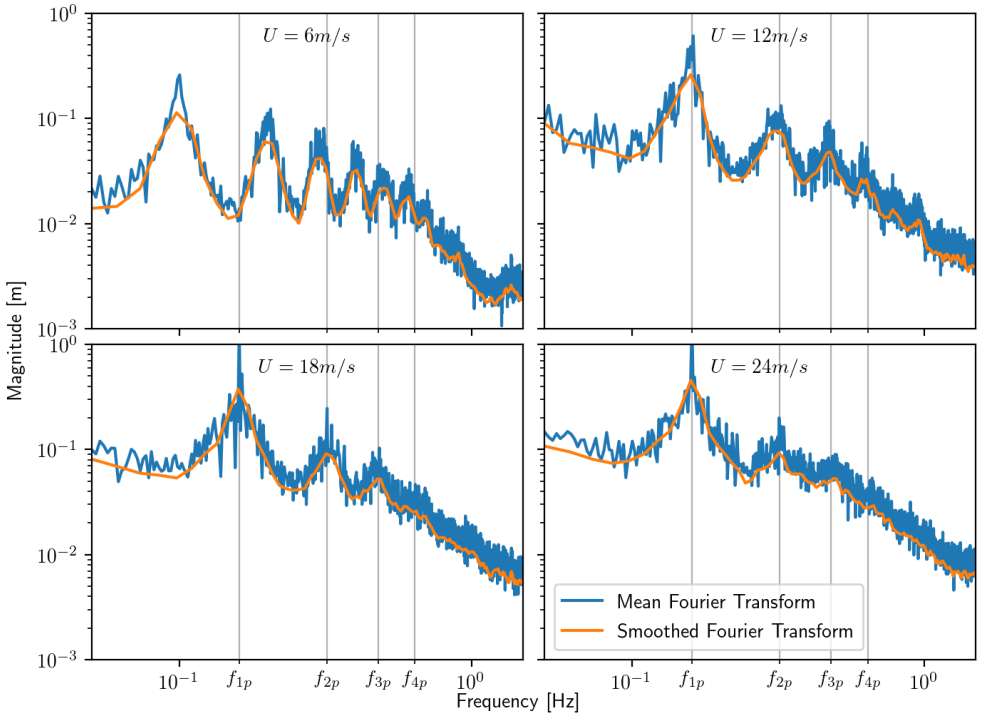


Figure 5.1: Open loop frequency magnitude response of the high-fidelity turbine model tip deflection (NTM).

5.3 Baseline Control Design

IPC control is notoriously difficult to compare in literature. There is great variation between the aeroelastic code used, control methodology and wind turbine model. In order to better compare the controllers designed in this project, a baseline proportional-integral (PI) controller is designed. By doing so, it is assured that the same model, software and assumptions are used in comparing the performance. Additionally, the theoretical tools outlined in Chapter 3 will be used to demonstrate the disadvantages of using PI control for single-bladed IPC.

A PI controller has a transfer function of the form:

$$C_{PI}(s) = K_p\frac{s + \frac{1}{T_i}}{s} \tag{5.5}$$

where K_p is the proportional gain, T_i is the controller time constant. From the form of Equation 5.5, it is clear that the controller has a single pole and a single zero, which is a lower order than the other controllers analysed in this project.

The PI control parameters, K_p and T_i were chosen using an iterative process such that a balance is struck between the attenuation level at f_{1p} and the system robustness. Figure 5.2 shows the Nyquist plots for varying controller parameters, K_p and T_i at a wind speed of $U = 18m/s$. The linear sensitivity at f_{1p} is indicated in the lower left corner of each plot. The key insight from these plots is the distance between the critical point (marked as a red cross) and the Nyquist plot itself. If the Nyquist plot encircles the critical point, then the system is unstable. To account for uncertainties in the system modelling, the distance between the Nyquist plot and the critical point should be as large. it can be seen that there is a trade-off between the stability margin, indicated as the length of the red dotted line, and the f_{1p} attenuation. That is, the plots towards the lower left have high robustness but poor tip deflection attenuation, whereas the plots towards the upper right have low robustness but better attenuation. The parameters $K_p = 0.015$ and $T_i = 1.0$ were chosen as a compromise between the two parameters.

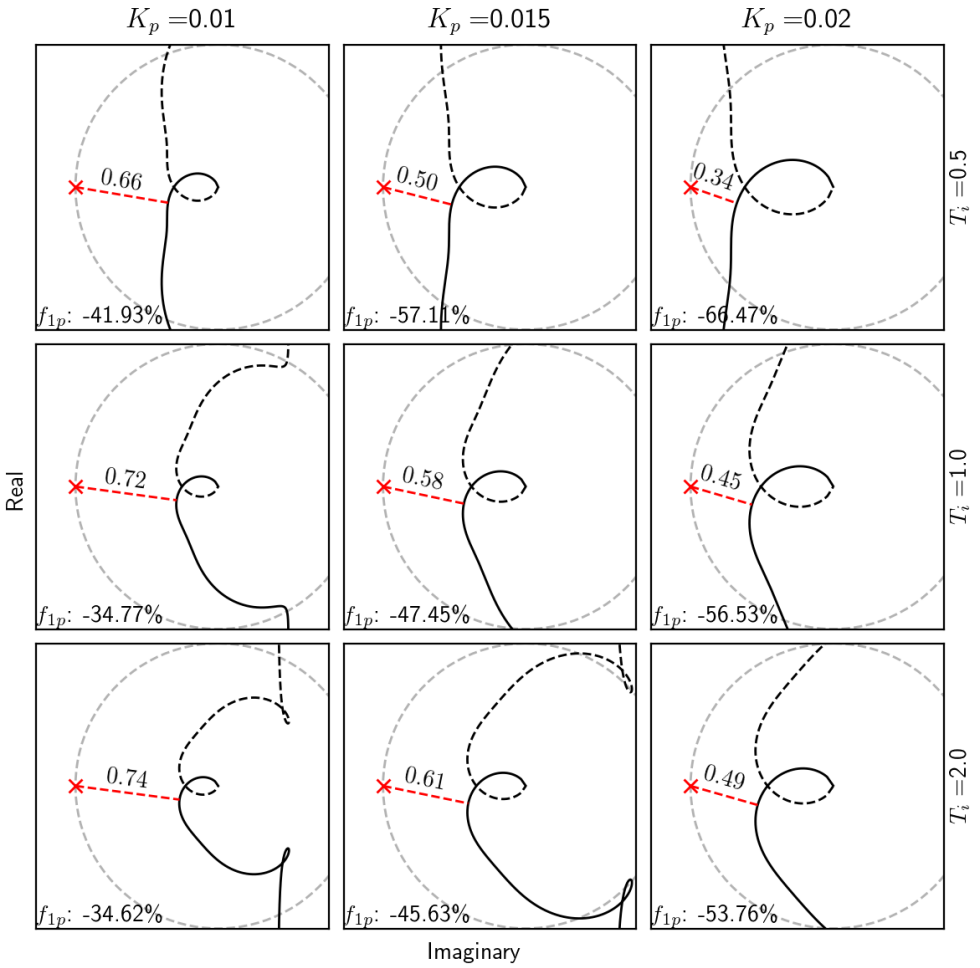


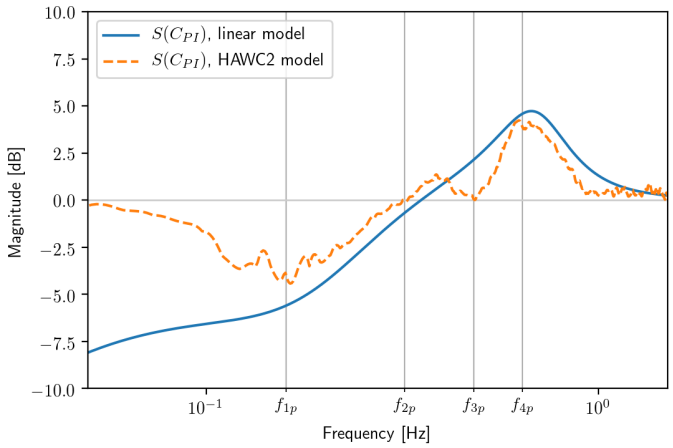
Figure 5.2: Nyquist plots the linear turbine model with different PI controller variations ($U = 18m/s$).

The PI controller is tested on the nonlinear turbine model in HAWC2 under NTM and the results are compared to the linear model results. Figure 5.3a shows the sensitivity of using the PI controller on both the

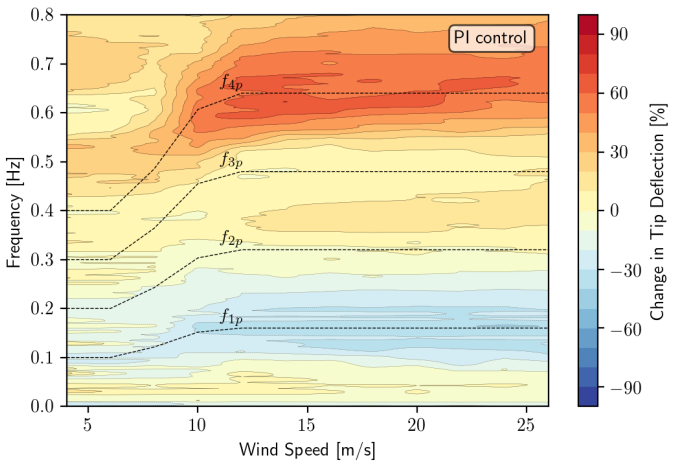
linear and the HAWC2 model. A number of weaknesses in the PI controller can be identified. Firstly, the controller is expected to attenuate frequencies near f_{0p} based on the linear model results. However, frequency components at this frequency cannot be suppressed in the rotating frame of reference due to the rotational symmetry of the rotor as outlined earlier in this chapter. This is verified in the high-fidelity HAWC2 simulations (dotted orange line), which shows insensitivity near at as the frequency approaches zero, and also at f_{3p}. Secondly, it can also be seen that there is a large amplification at f_{4p}. Although this is not desirable, the result is present in both the linear and the HAWC2 model, verifying that the blade modelling and the control design function as expected.

Another way of representing the frequency response of the controller is in the spectral contour plot in Figure 5.3b, which shows the percent change in tip deflection frequency components over a range of operating wind speeds. The results from this plot are from HAWC2 simulations. The advantage of representing the frequency data in this way is that the amplification and attenuation of each frequency component can be easily compared over all operating conditions. The f_{1p} attenuation and f_{4p} amplification can be clearly seen at operating conditions above rated wind speed (11.4m/s). The performance drops off below rated wind speed, where the rotor speed deviates from the controller target frequencies. This is found to be the case for all controllers investigated in this chapter.

Due to the conservation law outlined in Bodes Integral Formula (Theorem 1), signal attenuation at some frequencies require signal amplification at others. The controller could be better designed by suppressing different frequencies. For example, less attenuation could be performed at low frequencies (¡f_{1p}) as this frequency range is uncontrollable. This would allow for further attenuation at the target frequencies, and would reduce the amplification at f_{4p}.



(a) Sensitivity function of linear and HAWC2 turbine model ($U = 18m/s$).



(b) Frequency sensitivity contour indicating the increase/decrease in tip deflection frequency components by using C_{PI} on HAWC2 model.

Figure 5.3: Sensitivity plots for PI controller.

5.4 Single frequency Control Design

One clear shortfall of the PI controller is that it is unable to target particular frequencies without influencing other frequencies. To address this, a controller which has high levels of attenuation at a single frequency is designed. The motivation for this section is to investigate the load reductions of different turbine components when different blade frequencies are attenuated.

A controller targeting f_{1p} frequencies has the transfer function shown in Equation 5.6. The controllers for f_{2p} to f_{4p} have a slightly different form shown in Equation 5.7. These controllers consist of a second order low pass or band pass filter with a cutoff frequency set at the frequency component of interested, as well as two identical lead compensators in order to increase the phase response which helps increase the system robustness.

$$C_{1p}(s) = K_p \underbrace{\frac{1}{s^2 + 2\zeta_2\omega_2 s + \omega_2^2}}_{\text{Low pass filter}} \underbrace{\frac{(1-aTs)^2}{(1-Ts)^2}}_{\text{Lead compensator}} \tag{5.6}$$

$$C_{np}(s) = K_p \underbrace{\frac{s^2 + 2\zeta_1\omega_1 s + \omega_1^2}{s^2 + 2\zeta_2\omega_2 s + \omega_2^2}}_{\text{Band pass filter}} \underbrace{\frac{(1-aTs)^2}{(1-Ts)^2}}_{\text{Lead compensator}} \quad \text{for } n = 2, 3, 4 \tag{5.7}$$

where K_p is the proportional gain, ω and ζ refer to cut-off frequencies and damping ratios respectively, and a and T are the lead compensator shape and time parameters as described in Section 3.5. The values of these parameters are defined in Appendix A.

These controllers were tested in HAWC2 in order to determine the fatigue load reductions. From these results it is able to be determined which frequency components influence the loads of particular turbine components, and to what degree these loads can be reduced. Figure 5.4 shows the linear sensitivity functions of each single frequency controller, showing high levels of attenuation at a single multiple of f_{1p} as per the design. Unlike the PI controller (orange dotted line), the single frequency controllers show little action at frequencies between f_{0p} and f_{1p}, which allows for more flexibility in targeting specific frequencies.

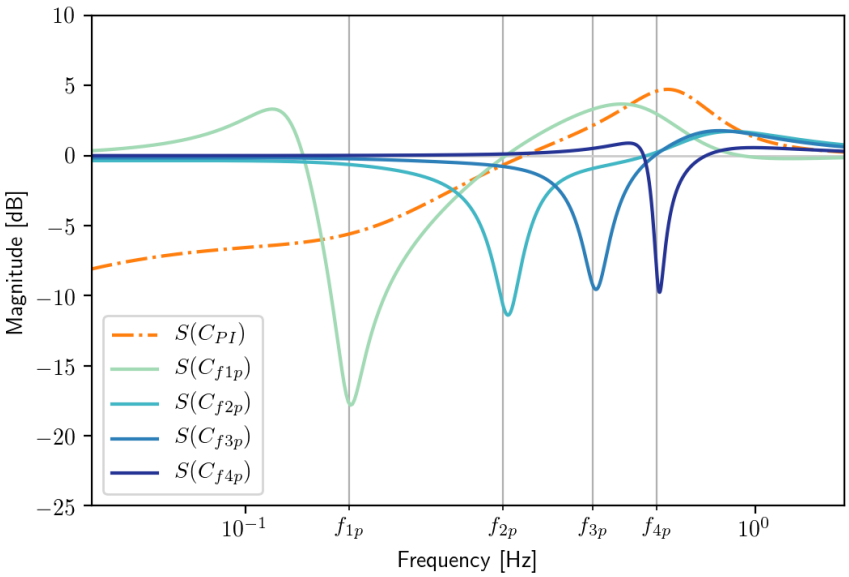


Figure 5.4: Sensitivity functions of single frequency controllers ($U = 18m/s$).

5.4.1 Tip Deflection Frequency Analysis

To see if the four single frequency controllers perform as expected, the frequency response generated from HAWC2 simulations using these controllers is compared to the linear model case as well as open loop case. Figure 5.5 shows the frequency response for the four controllers at a wind speed of $U = 18m/s$. It was found

that the predictions from the linear model (black dotted line) fit quite closely with the simulated output (red line). With the exception of the f_{3p} controller, the single frequency controllers show similar reductions in tip deflection fluctuations between the linear and the HAWC2 model at the target frequencies.

To visualise the level of amplification and attenuation at all operating wind speeds, Figure 5.6 shows the relative frequency response as a function of wind speed as a contour plot. Blue regions represent tip deflection signal attenuation, whereas red regions represent signal amplification. Above rated wind speed (11.4m/s), the behaviour of the system outputs does not vary significantly. This agrees with the findings in Chapter 4 where the blade models show similarities above rated wind speed. Below rated, the controllers do not attenuate the the target rotor frequency. This is a limitation of the controller architecture which is unable to adjust its frequency response to the changing rotor speed of the turbine. It is possible to overcome this issue by introducing gain scheduling.

The particular behaviour of the f_{3p} controller can also clearly be observed. As mentioned in Section 5.1, the IPC algorithm is unresponsive to f_{3p} fluctuations. This is shown to indeed be the case in Figure 5.5 and 5.6 which shows no change in the magnitude response at f_{3p}.

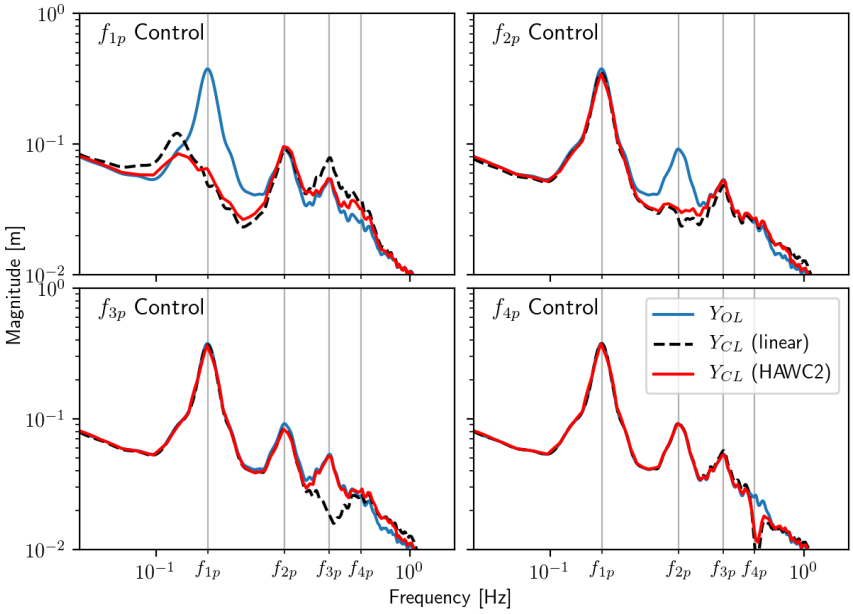


Figure 5.5: Tip deflection frequency magnitude response of closed loop single frequency control (linear and HAWC2 models) versus open loop system.

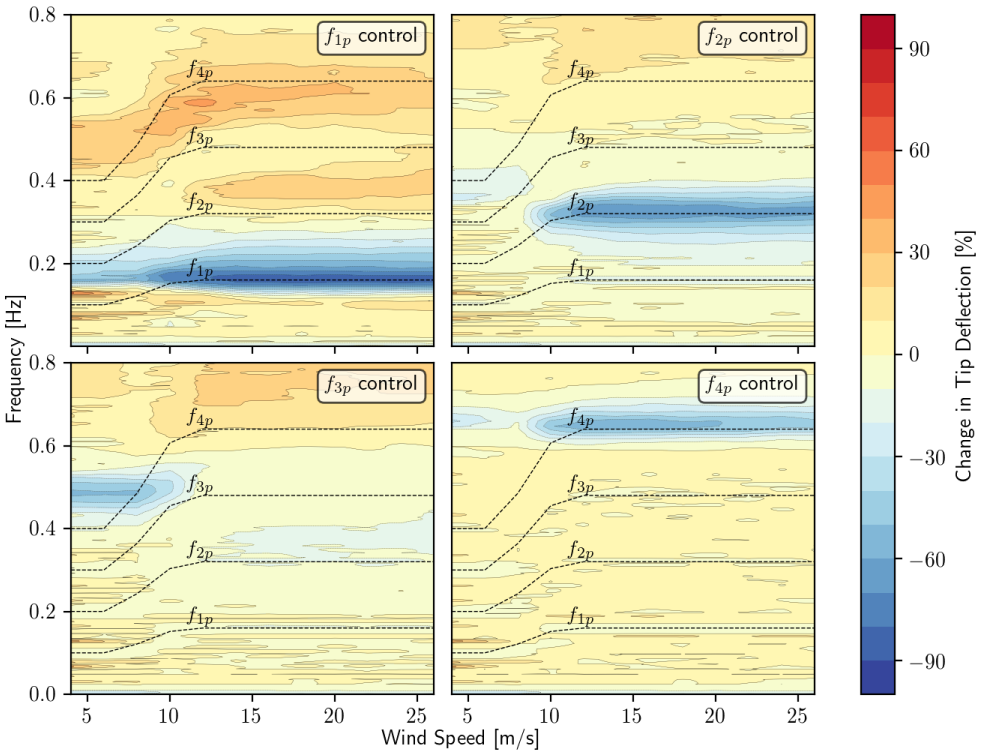


Figure 5.6: Frequency sensitivity contour of single frequency controllers. Each controller successfully attenuates their target frequency except for the f_{3p} controller.

5.4.2 Equivalent Load Reduction

Figure 5.7 and Table 5.1 show the lifetime equivalent loads for the first three single frequency controllers with a comparison against the no control case and the PI controller. The largest reduction in loads can be seen in the blade flapwise root bending moment. The f_{1p} controller has a 19.97% reduction in lifetime equivalent loads, which outperforms the PI controller which shows an 11.89% reduction. The only controller which shows reductions in the main bearing loads is the f_{2p} controller, showing a 10.98% reduction in tilt equivalent loads, and a 12.84% reduction in yaw equivalent loads. This result agrees with the work of Bossanyi [3]. As the main bearing is in the fixed frame, the frequency components of interest in the rotating frame are f_{2p} and f_{4p}. For this reason, the f_{1p} controller shows negligible bearing load reductions. Furthermore, the PI controller shows increasing loads in the main bearing compared to the open loop case. This can be attributed to the PI controller's sensitivity function (Figure 5.4) where amplification can be seen at f_{4p}. There are no significant changes in edgewise loads for all controllers as expected. Furthermore, for the change in loads due to the f_{4p} controller are negligible compared to the f_{1p} and f_{2p} controllers.

Table 5.1: Lifetime equivalent loads of single frequency controllers on HAWC2 turbine model. Percentage change is relative to the no-IPC case.

	PI Control		f_{1p} Control		f_{2p} Control		f_{3p} Control	
	$L_{lt}[kNm]$	% chng	$L_{lt}[kNm]$	% chng	$L_{lt}[kNm]$	% chng	$L_{lt}[kNm]$	% chng
Blade (flap)	17579	-11.89	15966	-19.97	18726	-6.14	19572	-1.90
Blade (edge)	15535	-1.80	15814	-0.03	15865	+0.29	15883	+0.41
Main Bearing (tilt)	12355	+3.96	11860	-0.21	10579	-10.98	11673	-1.78
Main Bearing (yaw)	11333	+2.50	11006	-0.47	9638	-12.84	10802	-2.31

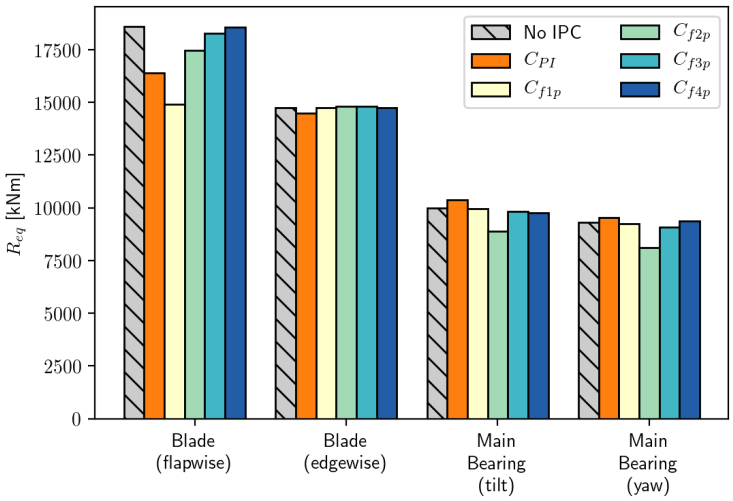


Figure 5.7: Lifetime equivalent loads of single frequency controllers on HAWC2 turbine model.

5.5 Two Frequency Control Design

From the results in the previous section, it can be seen that the largest blade load reduction can be achieved by targeting f_{1p}, whereas the largest main bearing load reduction can be achieved by targeting f_{2p}. This suggests that a hybrid controller targeting both these frequencies could show sufficient reductions in both the fixed and rotating frame. This strategy matches that used by Leithead and Dominguez [22] who also used single blade control to target f_{1p} and f_{2p} frequencies. A hybrid controller is designed to have the following form:

$$C_2(s) = K_p \left(\frac{s^2 + 2\zeta_1\omega_1 s + \omega_1^2}{s^2 + 2\zeta_2\omega_2 s + \omega_2^2} \right) \left(\frac{s^2 + 2\zeta_3\omega_3 s + \omega_3^2}{s^2 + 2\zeta_4\omega_4 s + \omega_4^2} \right) \left(\frac{1 - aTs}{1 - Ts} \right)^2 \tag{5.8}$$

The values of these parameters is tabulated in Appendix A.

The controller consists of two band pass filters targeting f_{1p} and f_{2p}, as well as two identical lead compensators. The magnitude plot of the transfer function of this controller (hereon referred to as C_2) is plotted and compared to the PI and C_{1p} controllers in Figure 5.8. The parameters have been chosen to have large control action only near the two target frequencies while passing through frequencies near f_{0p}, f_{3p}, and f_{4p}. A slight amplification can be seen above f_{4p}, a result of Bode's integral formula (Theorem 1). As the majority of the tip deflection energy is before f_{4p}, this amplification has negligible effects.

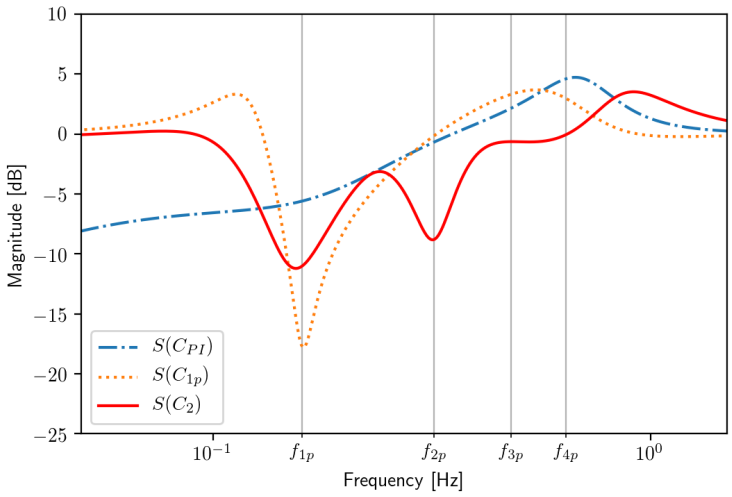
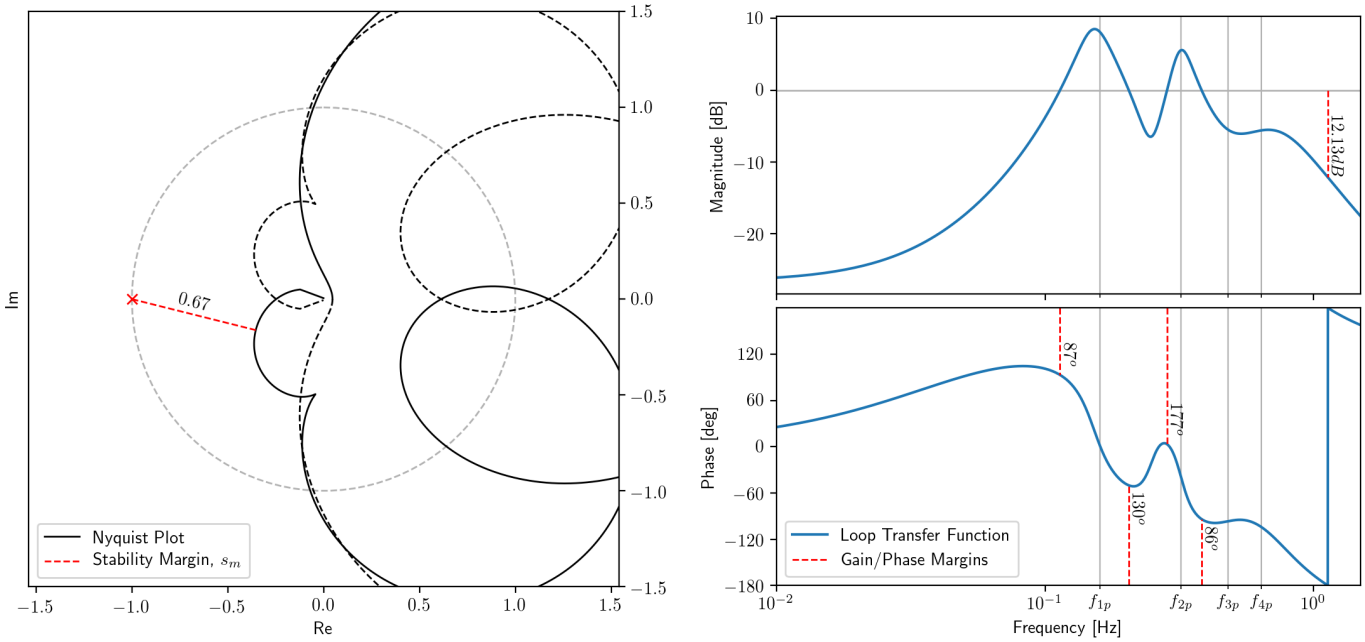


Figure 5.8: C_2 controller transfer function compared against C_{PI} and C_{f1p} controllers.

In addition to shaping the control loop for adequate performance, the robustness of the controller is also considered. The Nyquist plot is shown in Figure 5.9a for $U = 18m/s$, showing a stability margin of $s_m = 0.67$. Figure 5.9b shows the standard measures for stability in a bode plot, showing a gain and phase margin of $12.13dB$ and 87^o. The stability margin is compared for various controllers on the following section.



(a) Nyquist Plot.

(b) Bode plot.

Figure 5.9: Stability plots of C_2 controller ($U = 18m/s$).

5.5.1 Tip Deflection Frequency Analysis

The tip deflection frequency response was calculated from HAWC2 simulations using the C_2 controller, shown in Figure 5.10. The C_2 controller successfully attenuates f_{1p} and f_{2p} tip deflection oscillations at above-rated wind speed, which is visible in the plots as a reduction in sensitivity at f_{1p} and f_{2p} compared to the open loop case. It can also be seen that the frequency response shows similar behavior above rated wind speed, and is unable to attenuate the peak frequencies effectively below rated ($U = 6m/s$). This is a result of the reduced rotor speed in this operating region for which C_2 has not been designed.

The HAWC2 frequency response matches closely with the linear approximation (dashed lines). This is also demonstrated in Table 5.2 where the sensitivity and stability margin of various controllers is compared at key frequencies. It can be seen in this table that the linear approximation for the controller sensitivity over estimates the attenuations at all frequencies.

The controller sensitivity at different frequencies varies based on the controllers in Table 5.2. The C_2 controller shows a 67.88% reduction in f_{1p} oscillations, which outperforms the PI controller which has a reduction of 37.24%. The C_{f1p} controller shows better attenuation at this frequency (83.10%) than the C_2 controller, however, this comes at a cost of lower robustness ($s_m = 0.65$) than the C_2 controller ($s_m = 0.67$), as well as worse attenuation at higher frequencies. The C_2 is able to attenuates f_{2p} frequencies to a higher degree than both the C_{PI} and C_{f1p} controllers. At f_{3p}, the HAWC2 model shows negligible attenuation and amplification compared to the predicted attenuation/amplification by the linear model. This is a result of the f_{3p} cancellation mentioned earlier in this chapter which is not take into account in the linear model.

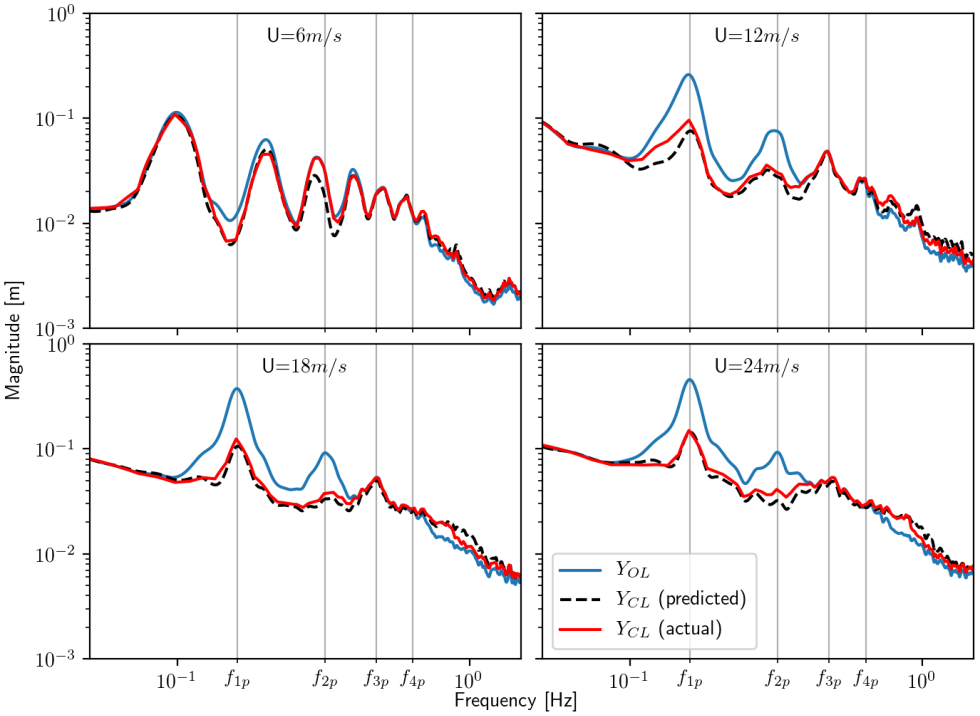


Figure 5.10: Tip deflection frequency magnitude response of closed loop C_2 control (linear and HAWC2 models) versus open loop system.

Table 5.2: Controller sensitivity and stability margin, s_m ($U = 18m/s$).

Controller	Turbine Model	Sensitivity [%]				s_m
		f_{1p}	f_{2p}	f_{3p}	f_{4p}	
C_{PI}	Linear	-47.45	-7.46	28.05	69.84	0.58
	HAWC2	-37.24	-0.67	0.78	56.10	-
C_{f1p}	Linear	-87.07	-1.68	46.47	41.15	0.65
	HAWC2	-83.10	3.98	1.71	23.57	-
C_2	Linear	-71.82	-63.70	-7.02	-0.77	0.67
	HAWC2	-67.88	-59.00	-1.18	4.44	-

5.5.2 Equivalent Load Reduction

The load reductions for the C_2 controller are plotted and tabulated in Figure 5.11, and Table 5.3 respectively. The 1Hz equivalent load results are compared to the C_{PI} controller, the f_{1p} controller, and the open loop system. The C_2 is able to reduce loads in both the rotating and non-rotating by a larger factor than both the C_{PI} and F_{1p} controllers. The blades experience a 25.10% reduction in flapwise RBM lifetime equivalent loads, and the main bearing experiences a 13.19% and a 17.88% reduction in the tilt and yaw directions respectively. It can be seen that the C_2 has the greatest degree of load reduction even though the controller has a lower attenuation level at f_{1p} compared to the C_{f1p} controller. This is a result of the f_{2p} attenuation which also has a contribution to flapwise blade load reduction. The C_2 controller therefore is able to achieve better load reductions with less aggressive control action at f_{1p}, and a higher level of robustness than either C_{PI} or C_{f1p}.

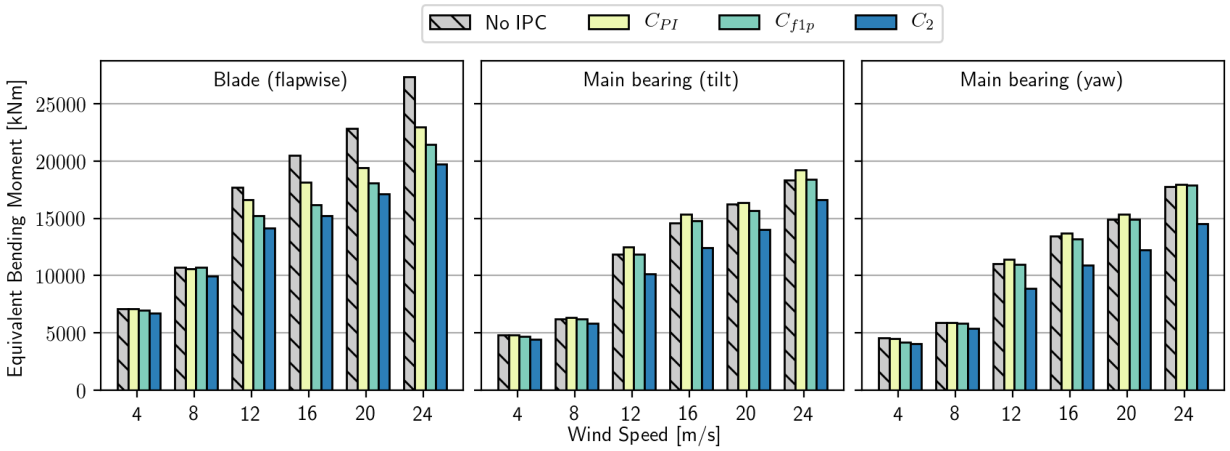


Figure 5.11: Short term equivalent loads (HAWC2, NTM).

Table 5.3: Lifetime equivalent loads of C_2 control.

	PI Control		f_{1p} Control		$C2$ Control	
	$L_{lt}[kNm]$	% chng	$L_{lt}[kNm]$	% chng	$L_{lt}[kNm]$	% chng
Blade (flap)	17579	-11.89	15966	-19.97	13942	-25.10
Main Bearing (tilt)	12355	+3.96	11860	-0.21	8675	-13.19
Main Bearing (yaw)	11333	+2.50	11006	-0.47	7636	-17.88

Comparing these results to literature, the load reductions in this project are in a similar range depending on the implementation. Single blade IPC control using local inflow measurements as a control input as explored by Larsen [4]. Larsen demonstrated a 25% reduction in 20 year equivalent blade flapwise loads, and a 9% reduction in the shaft loads on a smaller turbine model (2MW). Bossanyi shows a 17.8% reduction in blade flapwise 1Hz equivalent loads using a Coleman-based approach. The performance of the C_2 controller shows comparable blade load reductions to the Larsen implementation, and an improved reduction in shaft loads.

To visually demonstrate the operation of the IPC controller, Figures 5.12 and 5.13 show the tip deflection as a function of time and rotor azimuth respectively. In Figure 5.12, it can be seen that the fluctuation levels with C_2 IPC are lower than CPC alone, however there still exists some oscillations. It is possible to further reduce these oscillations at the cost of decreasing controller robustness. Plotting the tip deflection perturbations against azimuth angle gives a clearer representation of the f_{1p} attenuation as the IPC data shows a flatter tip deflection response than CPC alone. Furthermore, the perturbations show less deviation when IPC is present, represented as the thinner spread of data points.

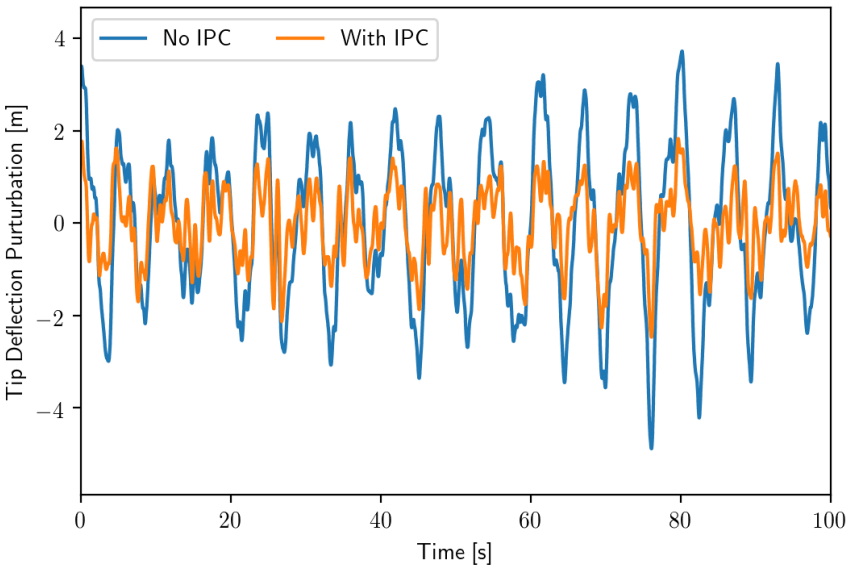


Figure 5.12: Tip deflection time series with and without IPC ($U = 20m/s$, NTM)

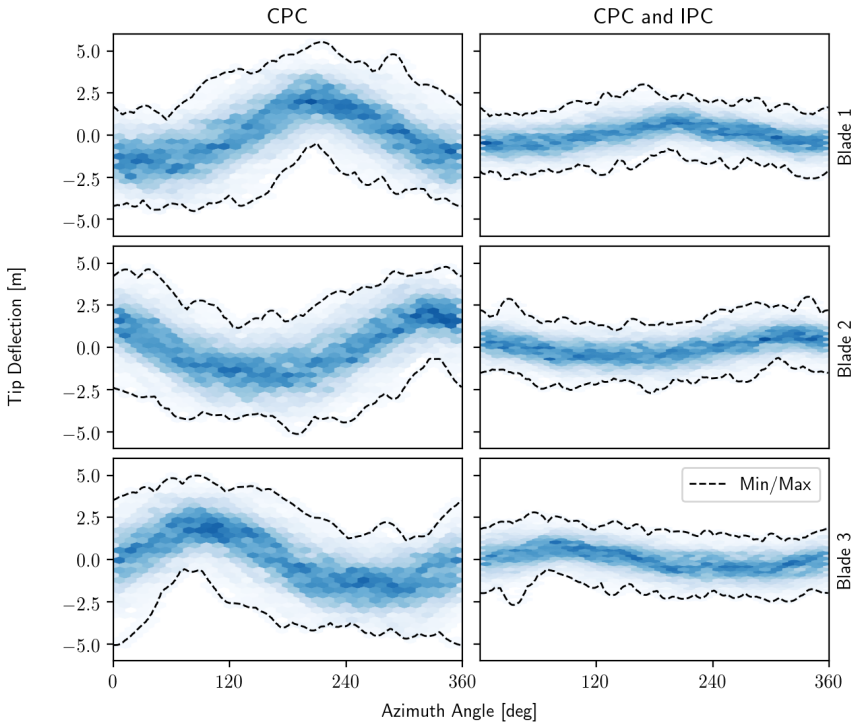


Figure 5.13: Tip deflection versus rotor azimuth angle ($U = 20m/s$, NTM).

5.5.3 Performance in Extreme Turbulence

The three key controllers in this project comparison are subjected to extreme turbulence as defined in the IEC standards for two main reasons. First, to determine by what degree these IPC controllers can reduce turbine loads in high levels of turbulence, and second, to test the limits of the controller stability. HAWC2 simulations are run in a similar fashion to the results in the section above with the turbulence model changed to the extreme turbulence defined in [18].

As it can be seen in Figure 5.14 and Table 5.5, the C_{PI}, C_{f1p} and C_2 controllers remain operational under extreme turbulence and show slightly lower load reductions compared to the normal turbulence case. The C_2 controller shows a 20.74% reduction in lifetime equivalent blade loads compared to the 25.1% reduction in normal turbulence 5.5. A similar performance drop can be seen for the bearing loads.

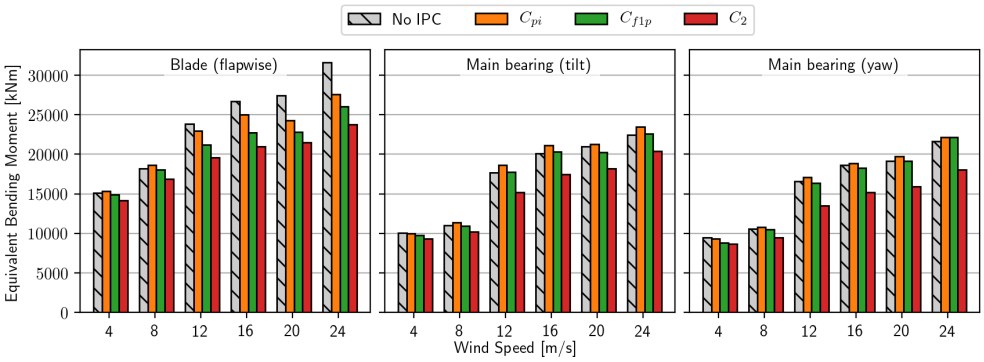


Figure 5.14: Short term equivalent loads of controllers in extreme turbulence.

Table 5.4: Lifetime equivalent loads in extreme turbulence.

	PI Control		f_{1p} Control		C_2 Control	
	$L_{lt}[kNm]$	%	$L_{lt}[kNm]$	% chng	$L_{lt}[kNm]$	% chng
Blade (flap)	21322	-7.34	19816	-13.89	18238	-20.74
Main Bearing (tilt)	14805	+4.76	14165	+0.23	12386	-12.35
Main Bearing (yaw)	13570	+3.11	13143	-0.14	10968	-16.66

5.6 Strain Gauge versus Tip Deflection Based Control

The performance of IPC using strain gauge sensors is outlined in this section to compare with the tip deflection implementation earlier in this chapter. In order to compare the strain gauge and tip deflection based controllers, the C_2 controller has been adapted for flapwise root bending moment measurements instead of tip deflection measurements. The HAWC2 simulations are adapted to take RBM measurements as inputs instead, and the controller transfer functions are scaled by a factor of $3.4 \times 10^{-4} m/kNm$ to account for the change in input signal. This scaling factor was determined from the proportionality analysis performed in Section 3.1. The HAWC2 simulation results are shown in Figure 5.15, and Table 5.5, showing comparable load reductions when using either tip deflection sensors or strain gauge sensors. Slight discrepancies in the results exist due to the differences in the feedback system, variations in the proportionality between tip deflection and RBM, as well as statistical errors which could be mitigated by increasing the seed count in the simulations. The conclusion from these results show that tip deflection sensors could be used as an alternative to strain gauge sensors for the use in IPC without significant differences in the turbine load reductions.

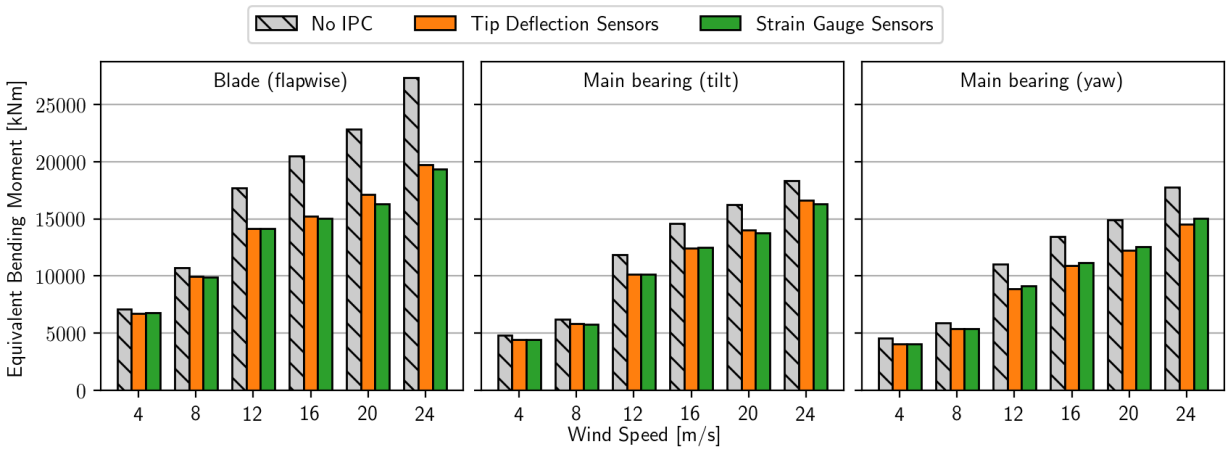


Figure 5.15: Short term equivalent loads of IPC with tip deflection sensors versus strain gauge sensors. The same turbulent seeds are used for both simulation sets.

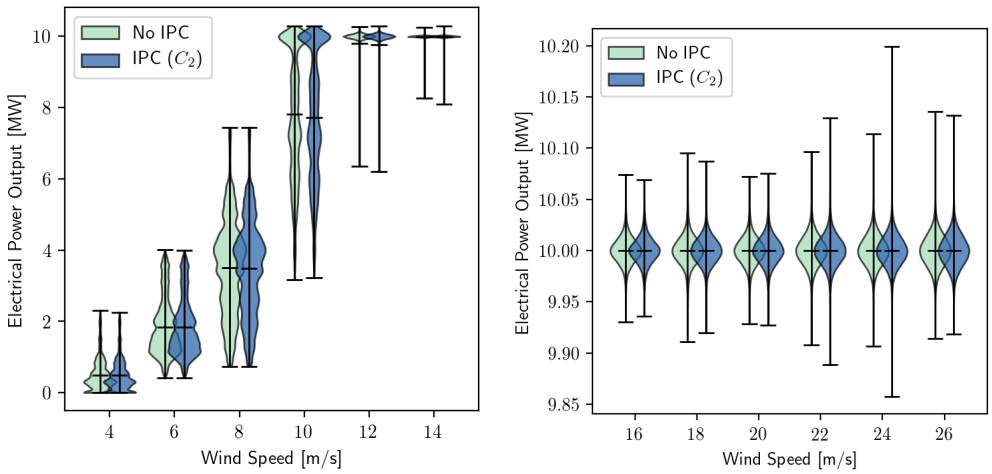
Table 5.5: Lifetime equivalent loads for strain gauge based IPC.

	Tip Deflection Sensors		Strain Gauge Sensors	
	$L_{lt}[kNm]$	% chng	$L_{lt}[kNm]$	% chng
Blade (flap)	14942	-25.10	14682	-26.40
Main Bearing (tilt)	10317	-13.19	10231	-13.92
Main Bearing (yaw)	9081	-17.88	9308	-15.82

It should be noted that it would be beneficial to use both tip deflection and strain gauge sensors together to provide better control performance. A combination of both sensors could provide better estimates of root bending moment or tip deflection than using a single sensor. The topic of sensor fusion is not explored in this study, however a Kalman filter is a typical approach to fuse the measurements from two noisy sensors. Kalman filtering has been explored for IPC applications in [7]. Another application for using both strain gauges and tip deflection sensors together is provided in Appendix C. In the appendix, a theoretical framework for estimating the full flapwise blade deformation is provided. This is achieved by estimating the excitation of the first two blade modes. As this topic falls outside the scope of the project, it is not included in the main report.

5.7 Influence on Power Output

One underlying assumption of IPC is that the electrical power output of the turbine remains unaffected. The reasoning for this is that the IPC is decoupled from the CPC by ensuring the average pitch angle of all three blades is equal to the CPC pitch demand. To verify this assumption, the influence on power output due to tip deflection control has been investigated in this section. The distribution of electrical power output with and without tip deflection control is displayed in a violin plot for each wind speed in Figure 5.16. Two plots are provided at different zoom levels for clarity. Not only are the mean and range of the power output almost identical, the distribution itself also follows comparable shapes. This observation confirms that the IPC is able to operate without significant influence on the power output, especially above rated wind speed which is where the controller is designed to operate.



(a) $U = 4$ to $14m/s$

(b) $U = 16$ to $26m/s$

Figure 5.16: Electrical power output distributions with and without IPC (C_2 controller, NTM).

Chapter 6

Tip Trajectory Tracking Controller

Blade-tower clearance is an important factor in wind turbine design. The nacelle tilt angle, the rotor cone angle and the blade prebend can be optimally chosen in the design phase to leave enough tower clearance to accommodate for extreme blade bending and possible tower strikes [33]. In this chapter, an innovative approach to increasing tower clearance using IPC is introduced.

The work in this chapter outlines a control method to guide the blade tips away from the tower as the blades pass the tower. To do so, the single blade controllers introduced in the previous chapter are modified to include a tip deflection reference signal. The controller aims to minimise the distance between the measured tip deflection and the reference tip deflection. That is, to guide the blade tips along a desired path around the rotor. This control strategy is referred to in this chapter as Tip Trajectory Tracking (TTT).

6.1 Background

A straightforward way of achieving TTT is by modifying the tip disturbance rejection controller from the previous chapter to include a tracking reference signal, $r(t)$ (or $r(s)$ in the s-domain). The system block diagram in Figure 4.2 is modified in Figure 6.1 to include a reference signal, $r(s)$ which is precompensated by the transfer function $F(s)$. $F(s)$ is designed such that the closed loop system tracks the reference with the correct phase and magnitude.

The sensitivity function $S(s)$, which is the transfer function between the disturbance, $d(s)$ and the output, $\tilde{y}(s)$ remains unchanged. There is an additional transfer function to consider called the complimentary sensitivity function, $T(s)$ [21]. The complimentary sensitivity function describes the connection between $r(s)$ and $\tilde{y}(s)$, and is the focus of this chapter. From the block diagram, $T(s)$ can be shown to be:

$$T(s) = \frac{F(s)P(s)C(s)}{1 + P(s)C(s)} \tag{6.1}$$

The following analysis describe a way of integrating $F(s)$ directly into the reference signal to simplify the implementation of the TTT controller.

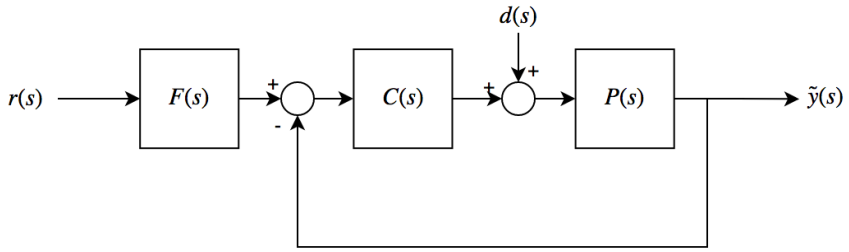


Figure 6.1: IPC control block diagram with TTT.

The simplest reference signal is a sinusoid with a frequency of f_{1p}, a desired amplitude of A_r, and with a phase offset of 120 degrees for each blade. The phase of the signal is chosen such that the tip deflection distance is furthest from the tower when the blade faces downwards. To enforce the correct phase, the azimuth angle of the rotor, $\psi(t)$, is used to generate the reference signal. The mean of the three reference signals is zero to ensure the tracking does not interfere with the CPC. Using three phase shifted sinusoids meets this condition:

$$r_1(t) = A_r \cos\left(\psi(t)\right) \tag{6.2}$$

$$r_2(t) = A_r \cos\left(\psi(t) + \frac{2\pi}{3}\right) \tag{6.3}$$

$$r_3(t) = A_r \cos\left(\psi(t) + \frac{4\pi}{3}\right) \tag{6.4}$$

$$r_1(t) + r_2(t) + r_3(t) = 0 \tag{6.5}$$

In order for the tip deflection to track the reference sinusoid, a precompensator, $F(s)$, must be implemented so that the gain and phase of the output matches the gain and phase of the reference signal. As the reference signal is a sinusoid oscillating at a frequency of f_{1p}, it is sufficient to design $F(s)$ such that only f_{1p} frequency components are compensated correctly.

Consider the complimentary sensitivity function in Equation 6.1 split into two factors: $F(s)$, and $P(s)C(s)/(1+P(s)C(s))$:

$$\frac{\tilde{y}(s)}{r(s)} = F(s)\frac{P(s)C(s)}{1 + P(s)C(s)} \tag{6.6}$$

The latter transfer function is known, and is plotted in Figure 6.2 for the C_{f1p} and C_2 controllers from the previous chapter. To ensure the output tracks the reference, let $\tilde{y}(s) = r(s)$, which sets the left hand side to 1. Evaluating the right hand side of Equation 6.6 at the reference signal frequency ($s = j2\pi f_{1p}$) yields:

$$1 = F(j2\pi f_{1p})ae^{jb} \tag{6.7}$$

where a and b are real numbers and ae^{jb} is a known complex number in exponential form such that $ae^{jb} = P(j2\pi f_{1p})C(j2\pi f_{1p})/(1 + P(j2\pi f_{1p})C(j2\pi f_{1p}))$. The values of a and b can be obtained directly from Figure 6.2. a equals the magnitude at f_{1p}, and b equals the phase at f_{1p} (red crosses).

Rearranging Equation 6.7 gives $F(j2\pi f_{1p}) = e^{-jb}/a$. If this condition is true, then $y(s)$ will track $r(s)$ with the same magnitude and phase assuming $r(s)$ is of the form of Equations 6.2 - 6.4.

The reference signals in Equations 6.2 - 6.4 can be adjusted for phase and magnitude to directly take into account this precompensation of $F(s)$. The adjusted reference signals are:

$$\tilde{r}_1(t) = \frac{1}{a} A_r \cos\left(\psi(t) - b\right) \tag{6.8}$$

$$\tilde{r}_2(t) = \frac{1}{a} A_r \cos\left(\psi(t) + \frac{2\pi}{3} - b\right) \tag{6.9}$$

$$\tilde{r}_3(t) = \frac{1}{a} A_r \cos\left(\psi(t) + \frac{4\pi}{3} - b\right) \tag{6.10}$$

In other words, by using a reference signals $\tilde{r}_1(t)$, $\tilde{r}_2(t)$, and $\tilde{r}_3(t)$, the tip deflection of the three blades will track the signals $r_1(t)$, $r_2(t)$, and $r_3(t)$ with correct phase and amplitude. The purpose of the analysis in this section is to accommodate digital implementation. The compensated reference signals are implemented in HAWC2 using the DLL described in Appendix B by specifying a desired tracking amplitude, A_r, and the compensation terms, a and b.

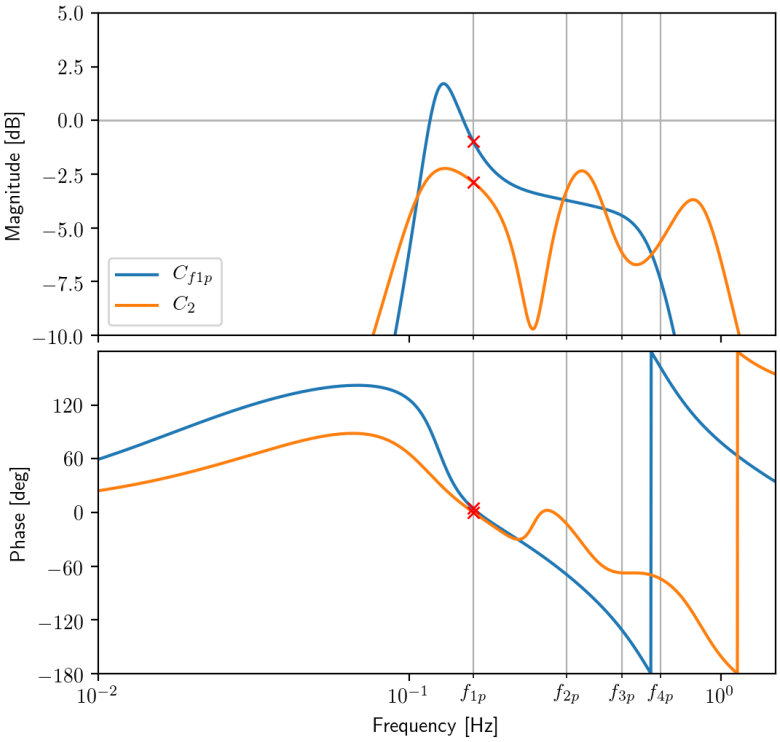


Figure 6.2: Transfer functions of PC/(1+PC) ($U = 18m/s$).

Table 6.1: Precompensator terms.

Controller	a [-]	b [^{o}]
C_{f1p}	0.891	4.56
C_2	0.719	0.40

6.2 Considered Wind and Controller Conditions

Simulations are set up under two load case regimes. The first being the normal turbulence level simulations with power law shear profile as defined in the DLC 1.1. The second case also uses the normal turbulence model, but with an inverted shear profile defined in the DLC 1.5. To investigate the long term effects of extreme shear, the inverse shear profile is modified from the IEC standards [18] to be persistent instead of a gust. Equation 6.11 shows the profile used.

$$V(z) = V_{hub}\left(\frac{z}{z_{hub}}\right)^{\alpha} - 2\left(\frac{z - z_{hub}}{D}\right)\left(2.5 + 0.2\beta\sigma_1\left(\frac{D}{\Lambda_1}\right)^{\frac{1}{4}}\right) \tag{6.11}$$

where $\alpha = 0.2$, $\beta = 6.4$, $\Lambda = 42$ is in accordance to [18] for the DTU10MW turbine, σ_1 is in accordance with the normal turbulence model, and z_{hub} and D are set to the hub height and rotor diameter of the turbine respectively.

An example of the inverted shear profile at $U = 18m/s$ is demonstrated in Figure 6.3. Inverted shear profiles are generated for all simulation wind speeds and implemented in HAWC2 using the `user_defined_shear` option.

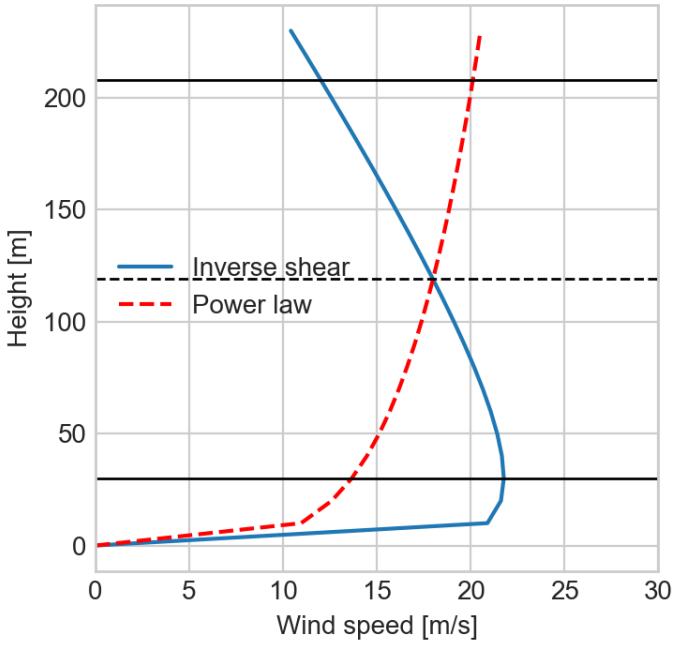


Figure 6.3: Inverted vertical shear profile ($U = 18m/s$

Two controllers were tested in this section: C_{f1p} and C_2, which have been adjusted to include TTT, are simulated using the tracking reference path defined in Equation 6.8, and for different sinusoid amplitudes, A_r. Simulations are run for both inverted and normal shear over a range of wind speeds, V=4, 6, ..., 26m/s. Three 10 minute seeds are run for each wind speed and the turbulence model used is NTM. Although both C_{f1p} and C_2 controllers were successfully implemented in simulation, only the results for C_2 are shown in this chapter for clarity.

6.3 Results in Normal and Inverse Shear

In this section, the performance of the TTT controller in HAWC2 simulations is analysed. The simulations are first run without any IPC to ensure the inverted shear functions correctly. Next, the performance of the tracking is analysed for different tracking amplitudes (A_r) and different shear profiles. The increase in tower clearance is then quantified, and the limitations of the controller is analysed by inspecting the equivalent loads and the blade pitch rate.

Figure 6.4 shows the tip deflection as a function of rotor azimuth angle in a 2D histogram. The vertical dashed lines indicate the angles at which the blades are facing downwards, and therefore are closest to the tower. The normal and inverted shear cases show approximately 180^o of azimuth phase difference in the tip deflection oscillation, and the inverted shear cases show a closer proximity to the tower. This is expected as the inverted shear has a higher wind speed at lower altitudes compared to the normal shear case, causing more deflection when the blades face downward.

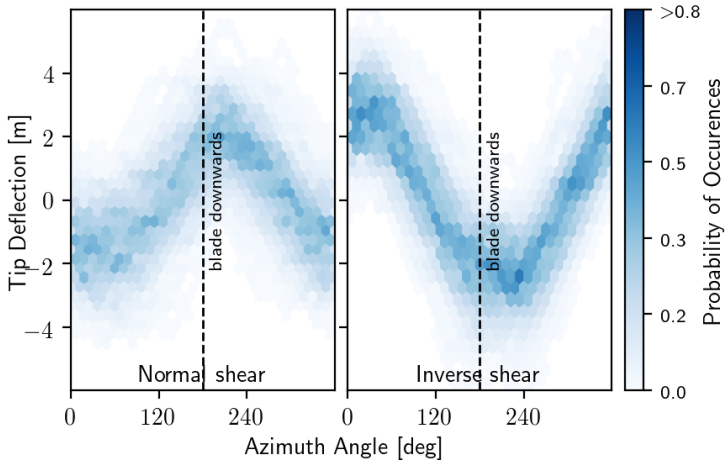


Figure 6.4: Tip deflection versus azimuth angle for normal and inverse shear. Positive tip deflection is away from the tower (no IPC, $U = 18m/s$).

6.3.1 Tracking performance

This section quantifies how well the controller is able to guide the blade tips along the set trajectory. Different reference signals were tested by changing the sinusoid tracking amplitude, A_r between 0m and 4m. Figure 6.5 shows tip deflection results of the HAWC2 simulations for the first blade under both normal (upper plot) and inverted shear (lower plot) cases. The C_2 controller is able to successfully track the reference signal (red dotted line) with comparable phase and amplitude. The TTT controllers with a tracking amplitude set to $A_r = 0$ is equivalent to the disturbance rejecting controller in the previous chapter as it aims to bring to the tip deflection to zero at all azimuth angles.

The controller is unable to track the reference signal as accurately in inverse shear conditions compared to normal shear. The controller must act more aggressively to reach the target trajectory in inverted shear conditions, leading to worse tracking. To better quantify the tracking performance the standard deviation of the tracking error is investigated in Figure 6.6. The C_2 controller is able to track approximately 15% lower standard deviation than the C_{f1p} controller in normal shear conditions, and 3% to 11% lower standard deviation in inverse shear. The tracking error experiences a minimum error near $A_r = 2m$ for the normal shear case. The reason for this is because the tracking path matches closely with the open loop tip deflection path. As a result, the controller requires less pitch action to guide the tip trajectory and can perform with

a lower error. This is in contrast to the tracking error in inverse shear conditions, where the error increases monotonically with tracking amplitude.

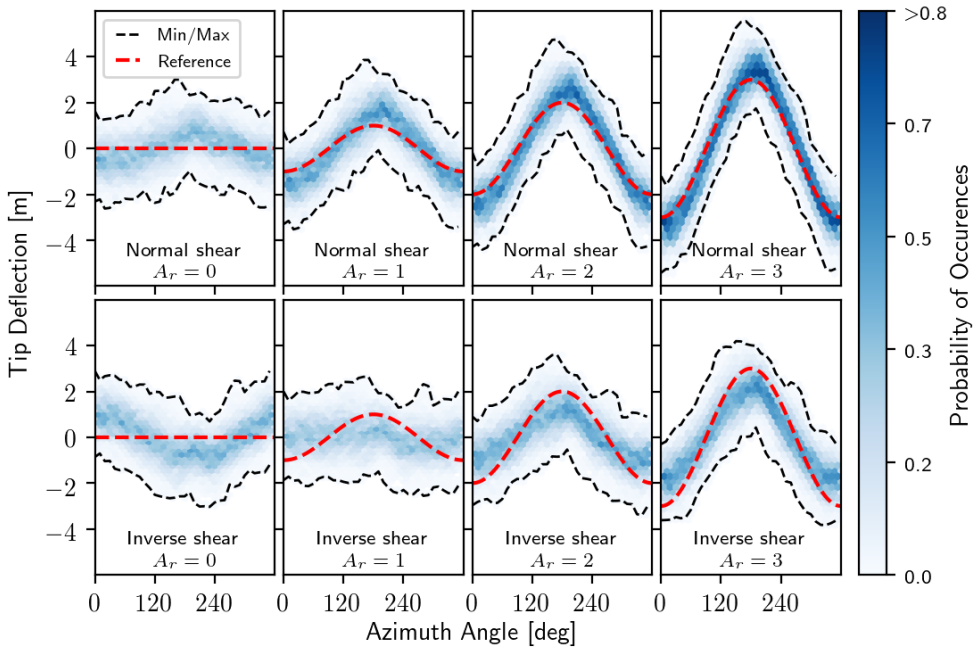


Figure 6.5: Tip deflection of first blade versus azimuth angle for different tracking amplitudes. (C_2 controller, $U = 18m/s$).

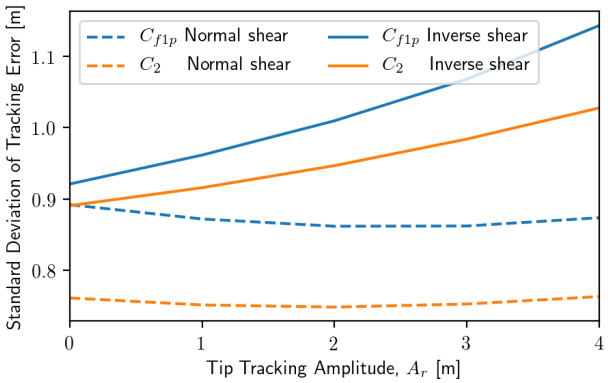


Figure 6.6: Standard deviation of tracking error for C_{f1p} and C_2 controllers in normal and inverse shear ($U = 18m/s$).

6.3.2 Minimum tower clearance

In this section, the increase in tower clearance due to TTT control is investigated. This is in contrast to the previous section which focuses on how well the blade tips follow a tracking signal. To more accurately measure the passing distance between the tower and the blades, the `tower_clearance_mblade` DLL in HAWC2 is used to provide a tower clearance signal. This module measures the shortest distance between all three turbine blades and the tower, and it takes into account the diameter of the tower, the tower motion, and the motion

and pitch of the blade. Figure 6.7 shows an example of the tower clearance signal. To summarise the data better, only the magnitude of the lower peaks is collected (indicated with red crosses) which represent the minimum distance between the tower and blade each time a blade passes. These data points are collected and used to generate a probability distribution of the closest tower passes.

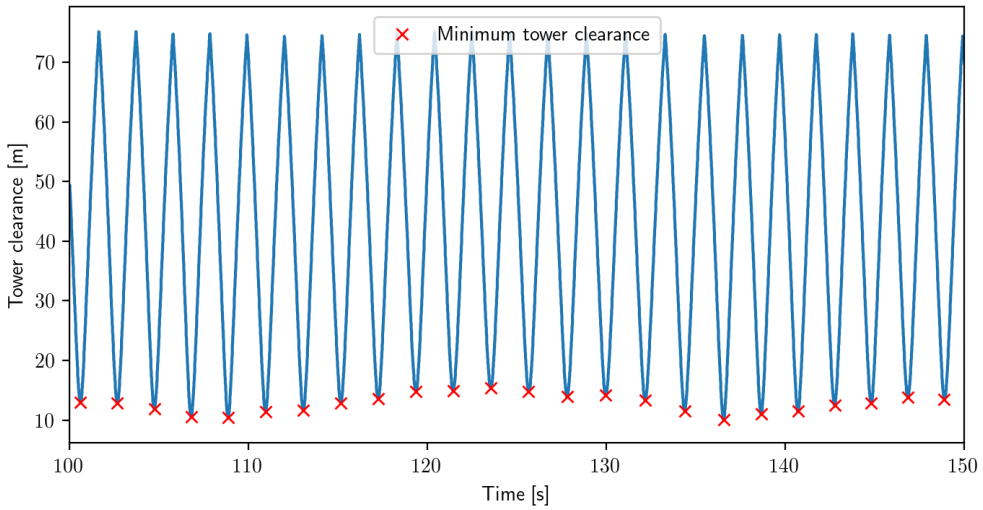
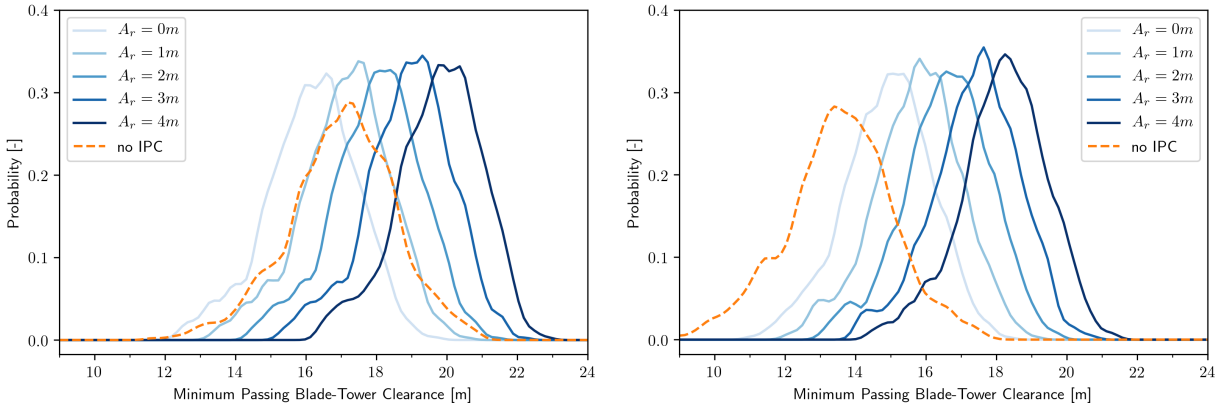


Figure 6.7: Minimum distance between tower and all three blades (Inverse shear, no IPC, $U = 18m/s$). The distribution of the lower peaks is analysed.

Figure 6.8 and Table 6.2 show the tower clearance statistics for normal and inverted shear conditions over a range A_r. Looking at the normal shear case (left plot), it can be seen that the IPC controller with zero tracking amplitude actually decreases the blade-tower clearance. This is a result of the reduction in f_{1p} fluctuations. Without IPC, the blade will tend to move away from the tower due to a reduced wind speed at this azimuth angle due to wind shear and tower shadow. The IPC controller without tracking counteracts this effect. Nevertheless, when a tracking amplitude is introduced in 1 meter increments ($A_r = 0$, 1, 2, 3, 4), the minimum tower clearance increases by corresponding 1 meter increments (12.30m, 13.33m, 14.46m, 15.37m, and 16.11m respectively) indicating that the tracking controller is indeed reducing the chance of blade-tower interactions. The mean and minimum tower clearance statistics are summarised in Table 6.2 which shows the increased clearance.

In the inverted shear case in Figure 6.8b, it can first be seen that the minimum tower clearance without IPC (9.25m) is approximately 4 meters less than with normal shear (13.03m). Introducing IPC, even with a tracking amplitude of zero, the tower clearance is already greatly increased (11.34), with the minimum clearance increasing by 2.1m. The average tower clearance in inverted shear conditions increases in accordance with A_r. The minimum tower clearance also increases with A_r, but at a lesser rate. For example, the minimum tower clearance at $A_r = 4$ (14.11m) is only 2.77m greater than for $A_r = 0$ (11.34m) instead of the expected 4m. This is likely a result of the larger tracking error in the inverse shear cases compared to the normal shear cases.



(a) Normal shear

(b) Inverse shear

Figure 6.8: Probability distribution of minimum tower clearance for varying TTT amplitudes and wind shear profiles ($U = 18m/s$). Kernal density estimation is used on HAWC2 data to produce distribution.

Table 6.2: Minimum tower clearance statistics under inverse shear ($U = 18m/s$).

Shear Condition	Statistic	Tower Clearance [m]					
		No IPC	$A_r = 0$	$A_r = 1$	$A_r = 2$	$A_r = 3$	$A_r = 4$
Normal Shear	mean	17.85	16.87	17.80	18.72	19.60	20.45
	min	13.03	12.30	13.33	14.46	15.37	16.11
Inverse Shear	mean	14.28	15.60	16.45	17.28	18.08	18.86
	min	9.25	11.34	12.18	12.74	13.35	14.11

6.3.3 Effect on fatigue loads

TTT introduces a tip deflection oscillation with a frequency of f_{1p} which can increase blade loads. This is contrary to the work in Chapter 5 which goes into detail of rejecting tip deflection oscillations to reduce blade loads. In this section, the loads on the turbine components are analysed when different tracking amplitudes are used to identify what amplitude of TTT can be implemented without increasing blade loads.

The 1Hz equivalent flapwise bending loads in normal shear conditions (Figure 6.9) increase as tracking amplitude increases due to the induced oscillation. It can be seen that a tracking amplitude of $A_r = 2m$ can be introduced and still maintain a lower blade load level than if no IPC were to be used at all. This observation can also be seen in the lifetime equivalent loads shown in Table 6.3. Introducing a tracking amplitude of $A_r = 2m$ still yields a reduction in 1Hz lifetime equivalent loads by 2.13%. In inverse shear conditions, the load trends show a similar pattern (Table 6.4). Blade loads increase with increasing A_r, and furthermore, a tracking amplitude of $A_r = 4m$ can be introduced while still maintaining a reduction in blade loads (1.26%).

Main bearing loads are mostly unaffected when TTT is introduced in normal shear conditions. The tilt moment in the main bearing increase by $233kNm$, when A_r is increased from 0m to 4m, which is only a 2.6% increase in lifetime equivalent loads. Nevertheless, the bearing loads still remain significantly lower than using no IPC at all.

In inverted shear conditions, the effect of TTT on main bearing loads is more significant than normal shear conditions, showing a 9.4% increase in main bearing life time equivalent loads in the tilt direction

when A_r is increased from 0m to 4m (Table 6.4). The source of this load increase can be seen in the lower plots of Figure 6.9 at $U = 12m/s$. At this wind speed, both the yaw and tilt bearing loads increase with A_r. It is possible that the TTT is interfering with the CPC controller in this region as it is in a transition point between rated and below rated operating conditions.

The negligible influence of TTT on bearing loads compared to blade loads can be explained due to the change in frame of reference. As the tracking introduces a f_{1p} oscillation in the rotating frame, there are only small fluctuations introduced in the fixed frame due to cancellation between all three blades. For this reason, the main bearing loads are insensitive to this form of tip tracking. This is also the case in inverse shear conditions.

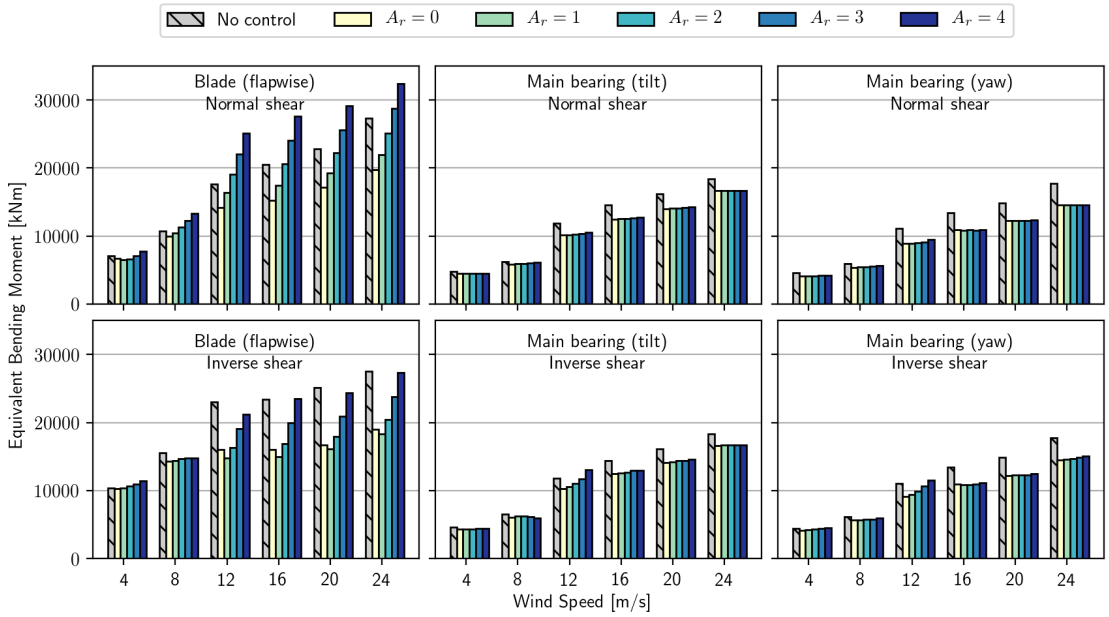


Figure 6.9: Short term equivalent loads under normal and inverse shear conditions and varying tracking amplitudes (C_2 controller).

Table 6.3: Lifetime equivalent loads [kNm] of TTT controller under normal shear conditions. Percent change is based on the no-IPC case.

	$A_r = 0$		$A_r = 1$		$A_r = 2$		$A_r = 3$		$A_r = 4$	
	L_{lt}	% chng	L_{lt}	% chng	L_{lt}	% chng	L_{lt}	% chng	L_{lt}	% chng
Blade (flap)	13942	-25.10	15744	-15.42	18218	-2.13	20965	+12.63	23872	+28.25
Main Bearing (tilt)	8675	-13.19	8706	-12.89	8762	-12.33	8799	-11.96	8908	-10.87
Main Bearing (yaw)	7636	-17.88	7633	-17.91	7649	-17.73	7680	-17.40	7763	-16.51

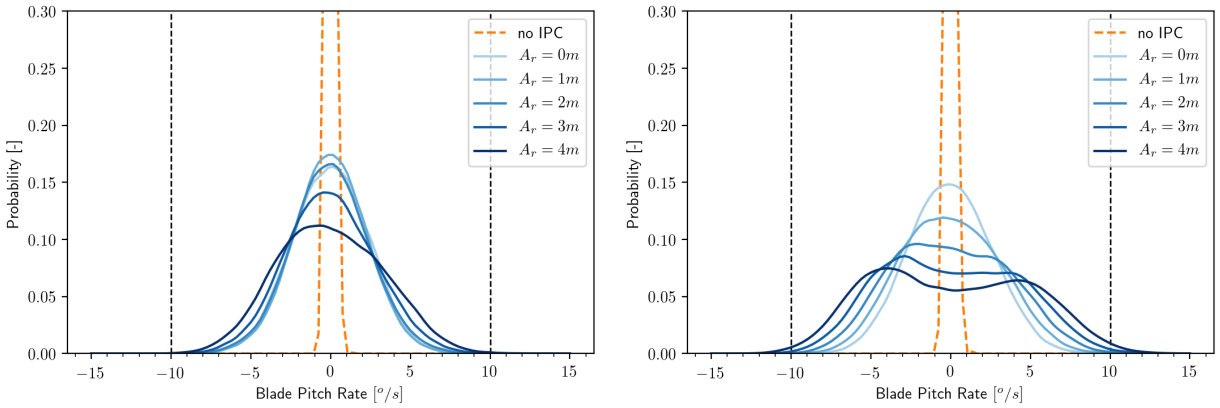
Table 6.4: Lifetime equivalent loads $[kNm]$ of TTT controller under inverse shear conditions. Percent change is based on the no-IPC case.

	$A_r = 0$		$A_r = 1$		$A_r = 2$		$A_r = 3$		$A_r = 4$	
	L_{lt}	% chng	L_{lt}	% chng	L_{lt}	% chng	L_{lt}	% chng	L_{lt}	% chng
Blade (flap)	14795	-28.09	14254	-30.72	15621	-24.07	17790	-13.53	20315	-1.26
Main Bearing (tilt)	8762	-11.79	8861	-10.80	9014	-9.26	9231	-7.07	9590	-3.45
Main Bearing (yaw)	7693	-17.32	7758	-16.62	7865	-15.47	8053	-13.45	8354	-10.21

6.3.4 Effect on blade pitching rate

Another effect to consider is the blade pitch rate and pitch angle. TTT greatly increases the movement of a bearing, which can wear the bearings at a faster rate. With IPC and tip tracking, the rate at which the blades pitch increases with tracking amplitude (Figure 6.10). Although no specific bearing model is investigated in this analysis, a pitch rate of $10^o/s$ is used as an indicator for an upper limit and is based on [2]. It can first be seen that the TTT controller shows a broader distribution of pitch rates than the no-IPC case. Furthermore, the TTT controller predominately remains within the pitch limit of $10^o/s$ in normal shear conditions, and slightly exceeds the limit in inverse shear. For the most aggressive case ($A_r = 4m$), the pitch rate remains within the limit 99.86% of the time in normal shear, and 98.44% of the time in inverse shear based on the estimated probability density functions in Figures 6.10. The large pitch rates in the C_2 controller are a result of the f_{2p} control action which requires faster pitching that the f_{1p} control action. One way to ensure the controller does not exceed this limit is to make the controller less aggressive at this frequency at the cost of reducing disturbance rejection performance. Another method is to set a saturation limit for the pitch rate. However, this can cause instability in the controller as a saturation limit is a nonlinear phenomenon.

The difference between normal and inverse shear can also be seen. Under inverse shear conditions, there is a broader distribution of pitch rates. Additionally, the pitch rate shows a stronger bimodal distribution compared to the normal shear case. This distribution is typical of a sinusoidal signal subject to noise, indicating that the f_{1p} tracking signal dominates the control action.



(a) Normal shear

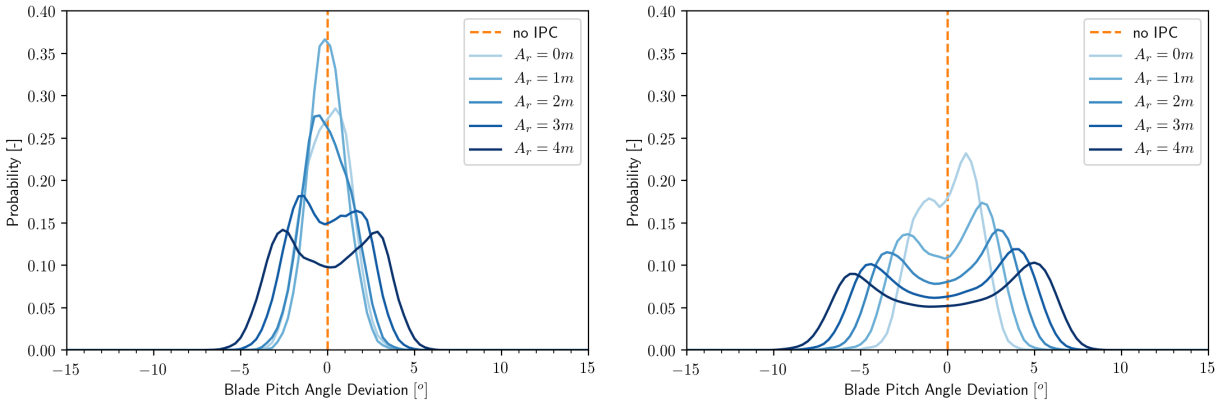
(b) Inverse shear

Figure 6.10: Pitch rate probability distributions for varying TTT amplitudes. Calculated using kernel density estimation. (C_2 controller, $U = 18m/s$).

In addition to investigating the blade pitch rate, the pitch angle deviation from the mean is also investigated in Figure 6.11. Without IPC, the pitch angle shows no deviation from the CPC pitch demand by definition. The introduction of IPC increases the variation from the CPC pitch angle deviation. Larger deviations also

occur with larger tip tracking amplitudes and in inverted shear. Again, the bimodal shape is more prominent for higher tracking amplitudes and for inverse shear conditions.

IPC causes specific types of damage to the pitch bearing due to more frequent small oscillation in the blade pitch angle. Fretting and false brinelling are causes of surface contact damage. These effects are seen to be significant hurdles in applying IPC in practice [34]. It is beyond the scope of this project to investigate the specific types of damage in the bearings and more research is required in this field. Nevertheless, the pitch angle and rate investigation in this section provide an insight into what the pitch requirements are to perform different degrees of TTT.



(a) Normal shear

(b) Inverse shear

Figure 6.11: Pitch angle probability distribution for varying TTT amplitudes. Calculated using kernel density estimation. (C_2 controller, $U = 18m/s$).

Chapter 7

Conclusion and Recommendations

In this study, applications for blade tip deflection sensors in IPC systems was investigated. The conclusions drawn from the results of this report as well as a summary of future work is presented in this chapter.

7.1 Conclusion

An individual pitch controller (IPC) based on tip deflection sensors is presented in this study. The motivation of doing so is to verify the performance of tip deflection sensors in IPC systems compared to the more typical strain gauge sensors at the blade roots. A preliminary study in this report demonstrates the highly correlated nature between blade flapwise bending moment and flapwise tip deflection, which indicates that the two measurements can be substituted for IPC applications.

The individual pitch controller was designed for the DTU10MW Reference Turbine Model. A linear approximation of the blade pitch angle to flapwise tip deflection dynamics was estimated using system identification techniques. HAWC2 simulations were run over a range of operating wind speeds and the turbine model was subject to a step pitch signal in order to identify the system dynamics of the blade. From this estimate, a loop shaping design approach was performed to design various individual pitch controllers. To deal with model uncertainty and nonlinearities, a Nyquist-based stability margin (s_m) was maximised to robustify the controller.

The IPC follows a single-blade control approach. That is, each blade is controlled by three independent single-input single-output controllers. This is in contrast to Coleman-based control which is more common in IPC. Single blade control avoids issues of tilt-yaw coupling which is often overlooked in Coleman-based control. First, four prototype controllers were designed to attenuate key frequencies over a narrow bandwidth. In particular, f_{1p} (once per revolution) to f_{4p} (four times per revolution) frequencies were targeted and tested for their attenuation and load reduction capabilities. As the controller operates in the rotating frame, tip deflection attenuations at f_{1p} showed the greatest level of load reductions in the blades, whereas fixed frame loads (ie. main shaft loads) were attenuated when f_{2p} and f_{4p} frequencies were targeted. Harmonics of f_{3p} frequencies could not be attenuated as the IPC controller architecture is decoupled form the collective pitch controller. Based on this analysis, a hybrid controller, referred to as the C_2 controller, was produced which targets f_{1p} and f_{2p} tip deflection frequencies.

The C_2 disturbance rejection controller showed promising fatigue load reductions in both the rotating and non-rotating frame. Lifetime equivalent flapwise blade loads, main bearing tilt loads and yaw loads were reduced by 25.10%, 13.19% and 17.88% respectively compared to the no-IPC case. Furthermore, this controller greatly outperformed a PI controller architecture by 14.4% in lifetime equivalent blade load reductions. The PI controller was unable to attenuate fixed frame loads which was a result of performing control in the rotat-

ing frame. The C_2 controller has a stability margin of $s_m = 0.67$, which exceeds the stability margin of the PI controller ($s_m = 0.58$), showing that the design approach in this study produced better performing and more robust controllers than a PI based approach. The controller showed negligible effects on the electrical power output of the turbine. Furthermore, the C_2 showed comparable performance when either tip deflection or strain gauge measurements was used as an inputs.

The C_2 controller was further modified to perform tip trajectory tracking (TTT). The motivation of this is to reduce the chance of blade-tower interactions. To the best of the authors knowledge, tower clearance has not been addressed using active control techniques in previous studies. TTT control was successfully able to guide the blade tips along a sinusoidal path away from the tower in both normal and inverted shear conditions. The controller was implemented by introducing a sinusoidal reference signal to the C_2 controller. The blade tips were able to track the reference signal with a standard deviation of less than 0.76m in normal shear conditions. In normal shear, up to 2 meters of average tower clearance could be introduced while still maintaining a lower lifetime equivalent load than the standard CLC controller, and up to 4 meters of mean tower clearance could be introduced while keeping the pitch limit below $10^o/s$ for 99.86% of operating time.

The control design process presented lends itself to real world applications, and has an emphasise on digital implementation. The controller was discretised using the bilinear transformation with a sampling frequency of 100Hz, although a sampling frequency as low as 16Hz could have been used with comparable performance. Two software implementations of the controller are presented in Appendix B in Python and HAWC2.

The research in this study assumed ideal tip deflection measurements with no noise or low pass filtering. Therefore the results presented shall be considered as best case scenarios. Nevertheless, it is straight forward to modify the control design process to take into account tip deflection sensor dynamics and measurement noise.

7.2 Future Work

Sensor Measurement Error Modelling and Sensor Fusion

The analysis in this project assumes an ideal tip deflection sensor operating at a high sampling rate (100Hz). In reality, there will be some degree of measurement error and noise due to low pass filtering and noise. The use of a Kalman filter can be investigated as a state estimator to optimally estimate the tip deflection based on noisy measurements and knowledge of the blade dynamics. Furthermore, sensor fusion can be performed using a Kalman filter to combine measurements from both strain gauges and tip deflection sensors.

Active Aerodynamic Load Control (AALC)

A major limitation of IPC is its response time. There are limits to how fast a blade can pitch which limits the bandwidth of IPC. An alternative to pitching is using AALC devices such as trailing edge flaps or microtabs. a fusion of IPC and AALC shows has shown promising results in [31] as it can target both low and high frequency components effectively. The use of tip deflection sensors could benefit such a control system by providing an additional measurement input.

Robust Stabilisation

The controllers in this project are designed using an manual loop shaping approach. More sophisticated optimisation methods exist for designing such a controller and to provide additional robustness. One such method is $\mathcal{H}_\infty$ loop shaping such as the Glover-Macfarlane method. $\mathcal{H}_\infty$ methods are often used in literature for IPC design [35] and could provide improved performance and robustness than the method presented in this project.

Pitch Bearing Damage Mechanisms

The use of IPC introduces more frequent small oscillation in the blade pitch angle. This can lead to additional wear in the pitch bearing by means of fretting and false brinelling. The lifetime damage that this can cause on pitch bearings has not been thoroughly investigated, and is a necessary area of research befor IPC becomes widely adapted in the wind industry.

Gain Scheduling

The presented controllers are designed at the rated rotor speed. This causes a performance drop when the turbine is operating below this speed. Although the presented controllers do not significantly increase the loads in this region, they do not provide a noticeable added benefit. Gain scheduling could be implemented to adjust the target frequencies of the controller to the actual rotor speed. An alternative to gain scheduling could be to use adaptive control techniques.

Reactive Tip Deflection Control

It is assumed that the IPC controller operates at all times, however this does not need to be the case. An alternative is to perform the necessary pitch action only during high risk situations. For example, the tip deflection sensors could be used to detect increased tower strike risk, or abnormally high blade loads. During such events, the controller could react by performing IPC or by simply pitch the blades out to mitigate such risks. Such a controller could reduce pitch bearing wear, potentially increasing pitch bearing lifetime.

Bibliography

[1] Stoyan Kanev and Tim van Engelen. Wind turbine extreme gust control. *Wind Energy*, 13(1):18–35, 2010.

[2] E Bossanyi and A Ervin. The design of closed loop controllers for wind turbines. *Wind energy*, 3(3):149–163, 2000.

[3] EA Bossanyi. Individual blade pitch control for load reduction. *Wind energy*, 6(2):119–128, 2003.

[4] Torben Juul Larsen, Helge A Madsen, and Kenneth Thomsen. Active load reduction using individual pitch, based on local blade flow measurements. *Wind Energy*, 8(1):67–80, 2005.

[5] Daniel Trudnowski and David LeMieux. Independent pitch control using rotor position feedback for wind-shear and gravity fatigue reduction in a wind turbine. In *American Control Conference, 2002. Proceedings of the 2002*, volume 6, pages 4335–4340. IEEE, 2002.

[6] Mahmood Mirzaei, Lars Christian Henriksen, Niels Kjølstad Poulsen, Hans Henrik Niemann, and Morten H Hansen. Individual pitch control using lidar measurements. In *Control Applications (CCA), 2012 IEEE International Conference on*, pages 1646–1651. IEEE, 2012.

[7] Kausihan Selvam, Stoyan Kanev, Jan Willem van Wingerden, Tim van Engelen, and Michel Verhaegen. Feedback–feedforward individual pitch control for wind turbine load reduction. *International Journal of Robust and Nonlinear Control*, 19(1):72–91, 2009.

[8] Q Lu, R Bowyer, and B Ll Jones. Analysis and design of coleman transform-based individual pitch controllers for wind-turbine load reduction. *Wind Energy*, 18(8):1451–1468, 2015.

[9] Wai Hou Lio, Bryn Ll Jones, Qian Lu, and JA Rossiter. Fundamental performance similarities between individual pitch control strategies for wind turbines. *International Journal of Control*, 90(1):37–52, 2017.

[10] M Geyler and P Caselitz. Robust multivariable pitch control design for load reduction on large wind turbines. *Journal of solar energy engineering*, 130(3):031014, 2008.

[11] K Papadopoulos, E Morfiadakis, TP Philippidis, and DJ Lekou. Assessment of the strain gauge technique for measurement of wind turbine blade loads. *Wind Energy*, 3(1):35–65, 2000.

[12] Dale E Berg, David G Wilson, Brian Ray Resor, Matthew Franklin Barone, Jonathan Charles Berg, Sridhar Kota, and Gregory Ervin. Active aerodynamic blade load control impacts on utility-scale wind turbines. Technical report, Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States); Sandia National Laboratories, Albuquerque, NM, 2009.

[13] David G Wilson, Dale E Berg, Mathew F Barone, Jonathan C Berg, Brian R Resor, and Don W Lobitz. Active aerodynamic blade control design for load reduction on large wind turbines. In *European Wind Energy Conference, Marseille, France*, volume 26, pages 643–678, 2009.

[14] irotor research project: intelligent blade technology to reduce energy costs.

[15] Torben J Larsen and Anders Melchior Hansen. How 2 hawc2, the user's manual. 2007.

[16] C Bak, F Zahle, R Bitsche, T Kim, A Yde, LC Henriksen, M Hansen, J Blasques, M Gaunaa, and A Natarajan. Description of the DTU 10 MW reference wind turbine. DTU wind energy report-i-0092. *Technical University of Denmark, Fredericia (Denmark)*, 2013.

[17] Morten Hartvig Hansen and Lars Christian Henriksen. Basic DTU wind energy controller. 2013.

[18] International Electrotechnical Commission et al. IEC 61400-1: Wind turbines part 1: Design requirements. *International Electrotechnical Commission*, 2005.

[19] Lennart Ljung. System identification. In *Signal analysis and prediction*, pages 163–173. Springer, 1998.

[20] M Matsuishi and T Endo. Fatigue of metals subjected to varying stress. *Japan Society of Mechanical Engineers, Fukuoka, Japan*, 68(2):37–40, 1968.

[21] Karl Johan Aström and Richard M Murray. *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.

[22] WE Leithead, Victoria Neilson, S Dominguez, and Arkadiusz Dutka. A novel approach to structural load control using intelligent actuators. In *Control and Automation, 2009. MED'09. 17th Mediterranean Conference on*, pages 1257–1262. IEEE, 2009.

[23] Stoyan Kanev and TG Van Engelen. Exploring the limits in individual pitch control. In *European Wind Energy Conference*, pages 1–10, 2009.

[24] Glenn Vinnicombe. *Uncertainty and Feedback: H [infinity] Loop-shaping and the [nu]-gap Metric*. World Scientific, 2001.

[25] Elbert Hendricks, Ole Jannerup, and Paul Haase Sørensen. *Linear systems control: deterministic and stochastic methods*. Springer Science & Business Media, 2008.

[26] Leland B Jackson. *Digital Filters and Signal Processing: With MATLAB® Exercises*. Springer Science & Business Media, 2013.

[27] Steven W Smith et al. The scientist and engineer's guide to digital signal processing. 1997.

[28] Leonardo Bergami and Mac Gaunaa. Analysis of aeroelastic loads and their contributions to fatigue damage. In *Journal of Physics: Conference Series*, volume 555, page 012007. IOP Publishing, 2014.

[29] Rik Pintelon and Johan Schoukens. *System identification: a frequency domain approach*. John Wiley & Sons, 2012.

[30] Kausihan Selvam. Individual pitch control for large scale wind turbines. *ECN, The Netherlands*, 2007.

[31] Matthew A Lackner and Gijs van Kuik. A comparison of smart rotor control approaches using trailing edge flaps and individual pitch control. *Wind Energy*, 13(2-3):117–134, 2010.

[32] Leonardo Bergami. *Adaptive trailing edge flaps for active load alleviation in a smart rotor configuration*. PhD thesis, DTU Wind Energy, 2013.

[33] L Sartori, P Bortolotti, A Croce, and CL Bottasso. Integration of prebend optimization in a holistic wind turbine design tool. In *Journal of Physics: Conference Series*, volume 753, page 062006. IOP Publishing, 2016.

[34] M Stammler and A Reuter. Blade bearings: damage mechanisms and test strategies, conference for wind power drives, aachen. 2015.

[35] D McFarlane and Keith Glover. Robust controller design using normalized coprime factor plant descriptions (lecture notes in control and information sciences). 1990.

Appendix A

Transfer Functions

A.1 Plant Transfer Functions

The continuous plant transfer functions used to design the controller are tabulated in Table A.1 for each wind speed.

Table A.1: Plant transfer functions.

Wind speed [m/s]	P(s)
4	$\frac{-20.8462s^3+137.9195s^2+1.4947s+-5.5565}{s^4+2.9405s^3+11.0112s^2+-1.3023s+0.1142}$
6	$\frac{-22.8369s^3+155.6219s^2+72.1321s+18.4033}{s^4+3.1049s^3+11.7084s^2+3.2239s+0.6901}$
8	$\frac{3.5201s^3+-90.1913s^2+1374.1849s+833.0069}{s^4+7.7312s^3+33.3743s^2+80.4849s+25.2398}$
10	$\frac{3.0869s^3+-165.5213s^2+2045.5058s+765.3664}{s^4+8.1594s^3+35.4788s^2+76.7278s+17.0833}$
12	$\frac{-44.4296s^3+303.6120s^2+375.6548s+146.0925}{s^4+4.0627s^3+12.4423s^2+7.6577s+4.9489}$
14	$\frac{-23.6444s^3+101.7049s^2+1179.8646s+483.0086}{s^4+4.3583s^3+22.0046s^2+22.6862s+15.5734}$
16	$\frac{-15.7581s^3+46.0359s^2+1488.9400s+753.3842}{s^4+4.7121s^3+25.6926s^2+30.0381s+22.5229}$
18	$\frac{-11.9307s^3+23.9174s^2+1621.7506s+1018.0156}{s^4+4.9132s^3+27.6913s^2+35.0812s+29.1484}$
20	$\frac{-9.9619s^3+16.3604s^2+1665.8556s+1276.9854}{s^4+5.0184s^3+28.8681s^2+38.9551s+35.6853}$
22	$\frac{-9.1095s^3+16.6020s^2+1655.4116s+1547.3149}{s^4+5.0579s^3+29.6321s^2+42.1713s+42.5659}$
24	$\frac{-8.9044s^3+21.2896s^2+1613.4809s+1834.6691}{s^4+5.0591s^3+30.1636s^2+45.0344s+49.8681}$
26	$\frac{-8.9127s^3+26.3289s^2+1561.0250s+2143.1836}{s^4+5.0482s^3+30.7345s^2+47.9828s+57.9452}$

A.2 Controller Transfer Functions

A.2.1 PI Control

The PI controller has a transfer function of the form:

$$C_{PI}(s) = K_p \frac{s + \frac{1}{T_i}}{s} \tag{A.1}$$

The parameters used in this report are $T_i = 1s$ and $K_p = 0.015rad/m$

A.2.2 Single Frequency Control

The single frequency controller targeting f_{1p} frequencies has a transfer function of the form:

$$C_{nP}(s) = K_p \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \frac{(1 - aTs)^2}{(1 - Ts)^2} \tag{A.2}$$

where the parameters are set to $K = 0.9217$, $\omega_n = 1.005rad/s$, $\zeta = 0.05$, $a = 83.3739$ and $T = 0.1089$.
The single frequency controller targeting higher harmonics of f_{1p} has a different transfer function form:

$$C_{nP}(s) = K_p \frac{s^2 + 2\zeta_1\omega_1 s + \omega_1^2}{s^2 + 2\zeta_2\omega_2 s + \omega_2^2} \frac{(1 - aTs)^2}{(1 - Ts)^2} \tag{A.3}$$

Table A.2: Parameters for C_{f2p}, C_{f3p}, and C_{f4p}.

	f_{2p}	f_{3p}	f_{4p}
K	0.03712	0.02858	0.01328
ω_1	2.8148	3.116459	1.00530
ω_2	2.0608	3.06619	4.07150
ζ_1	0.7	0.7	0.07
ζ_2	0.05	0.05	0.02
a	7.5486	7.5486	7.5486
T	0.1207	0.1207	0.1207

A.2.3 Two Frequency Control

The two frequency controller has a transfer function of the form:

$$C_2(s) = K_p \left(\frac{s^2 + 2\zeta_1\omega_1 s + \omega_1^2}{s^2 + 2\zeta_2\omega_2 s + \omega_2^2} \right) \left(\frac{s^2 + 2\zeta_3\omega_3 s + \omega_3^2}{s^2 + 2\zeta_4\omega_4 s + \omega_4^2} \right) \left(\frac{1 - aTs}{1 - Ts} \right)^2 \tag{A.4}$$

where the parameters are set to $K = 0.042539$, $\omega_1 = 0.95504rad/s$, $\omega_2 = 1.56828rad/s$, $\omega_3 = 2.01061rad/s$, $\omega_4 = 3.015928rad/s$, $\zeta_1 = 0.15$, $\zeta_2 = 0.1$, $\zeta_3 = 0.1$, $\zeta_4 = 0.3$, $a = 13.9282$ and $T = 0.2665$.

A.3 Controller performance

Table A.3: tip deflection controller sensitivity at key frequencies and stability margin, s_m ($U = 18m/s$, NTM).

	Model	Sensitivity [%]				s_m
		f_{1p}	f_{2p}	f_{3p}	f_{4p}	
C_{PI}	Linear	-47.45	-7.46	28.05	69.84	0.58
	HAWC2	-37.24	-0.67	0.78	56.10	-
C_{f1p}	Linear	-87.07	-1.68	46.47	41.15	0.65
	HAWC2	-83.10	3.98	1.71	23.57	-
C_{f2p}	Linear	-7.06	-70.68	-10.05	2.69	0.82
	HAWC2	-11.49	-65.36	-2.28	4.40	-
C_{f3p}	Linear	-2.54	-8.50	-65.38	0.98	0.81
	HAWC2	-6.60	-10.45	-3.67	9.46	-
C_{f4p}	Linear	0.17	1.33	6.09	-59.47	0.90
	HAWC2	-3.55	-0.81	-1.57	-47.85	-
C_2	Linear	-71.82	-63.70	-7.02	-0.77	0.67
	HAWC2	-67.88	-59.00	-1.18	4.44	-

Appendix B

Code Implementation

B.1 Python-HAWC2 Interface

Python-HAWC2 Interface is a tool written for the purpose of rapid control prototyping in HAWC2. The repository allows a custom Python function to run in HAWC2 in realtime. The repository is available at `github.com/jaimeliew1/Python-HAWC2-Interface`.

An implementation of the tip deflection single blade individual pitch controller using the Python-HAWC2 Interface is provided below:

```
"""
Institution : DTU
Course      : Master Thesis Project
Date        : 20-Jan-2018
Author      : Jaime Liew - S1417777
Email       : Jaimeliew1@gmail.com
Description : Demonstration if individual pitch control. This script
contains the controller code which is interfaced in realtime with a
HAWC2 simulation using the HAWC2Interface Module.
demand to each blade.
"""

from HAWC2_TCP import HAWC2Interface
import numpy as np


class IPC(HAWC2Interface):
    # A class which executes your custom python function.
    def __init__(self, modeldir, K, b, a):
        HAWC2Interface.__init__(self, modeldir)

        self.N = len(b) # order of filter
        self.K = K # Proportional gain of filter
        self.b = b # Numerator coefficients of filter
        self.a = a # Denominator coefficients of coefficients
        #note. len(b) == len(a) == N

        # placeholders for storing the past N inputs and outputs.
        # Assumes 3 blades
        self.x_ = np.zeros([3, self.N])
```

```
        self.y_ = np.zeros([3, self.N])



    def update(self, array1):
        theta       = array1[1:4] # Power pitch demand
        x           = array1[4:7] # tip deflection [m]

        x           = x - np.mean(x)   # Center about the mean

        # shift index of past states by 1. eg: [1,2,3,4] -> [4,1,2,3]
        N = self.N
        self.x_[:, 1:N] = self.x_[:, 0:N-1]
        self.y_[:, 1:N] = self.y_[:, 0:N-1]
        self.x_[:, 0]     = x
        self.y_[:, 0]     = [0, 0, 0]

        #apply filter to x_ and y_ to find newest y_
        for i in [0,1,2]:       # For each blade...
            for j in range(N):  # for each past input and output value...
                                # apply filter coefficients.
                self.y_[i,0] += self.b[j] * self.x_[i, j] - self.a[j] * self.y_[i, j]

        #Calculate control feedback action
        theta_ = -self.K*self.y_[:,0]

        # Superimpose IPC control action (theta_) over power pitch
        # demand (theta)
        out = list(theta + theta_)

        return out



if __name__ == '__main__':
    # IPC filter coefficients
    K = -1
    b = [0.04139865635722332, -0.24724984420650856, 0.6153247291080708,
        -0.8167730078549886, 0.6098892568550152, -0.24290112363655772,
        0.04031133337781191]
    a = [1.0, -5.918976177119697, 14.597287405477822, -19.19933888234148,
         14.204054827782327, -5.6043617399861, 0.9213345662369582]

    simTime, sampleTime = 100, 0.01
    N_iters = int(simTime/sampleTime)

    # Run HAWC2 with Python interface.
    HAWC2 = IPC('DTU10MW_Turbine/', K, b, a)
    HAWC2.run('htc/IPCExample.htc', N_iters)
```

B.2 HAWC2 IPC DLL

This section describes how the IPC DLL is integrated into a HAWC2 simulation. First, the file `IPC_HAWC2.dll` should be placed in the control folder of the wind turbine model. In the dll section of the .htc file, the IPC DLL should be defined. An example of a definition is as follows:

```
begin type2_dll;
name IPC ;
filename  ./control/IPC_HAWC2.dll ;
dll_subroutine_init init_ipc;
dll_subroutine_update update_ipc ;
arraysizes_init  100 1 ;
arraysizes_update 100 100 ;
begin init ;
constant 1       5    ; N  Length of filter
constant 2       -1   ; Kp Gain of controller [rad/m]
constant 3       7.166307797783242e-09     ; Feed-Forward Coefficients, b0
constant 4       1.577142916498056e-11     ; b1
constant 5       -1.4316835489060785e-08   ; b2
constant 6       -1.5754074483536844e-11   ; b3
constant 7       7.150545045958984e-09     ; b4
constant 8       1.0                         ; Feed-Backward Coefficients, a0
constant 9       -3.82336227575735         ; a1
constant 10      5.4780848051918065        ; a2
constant 11      -3.486056534625313        ; a3
constant 12      0.831334783268205         ; a4
 end init ;
begin output;
   dll type2_dll dtu_we_controller inpvec 2;      #1 Pitch1 demand angle [rad]
   dll type2_dll dtu_we_controller inpvec 3;      #2 Pitch2 demand angle [rad]
   dll type2_dll dtu_we_controller inpvec 4;      #3 Pitch3 demand angle [rad]
   mbdy state pos blade1  26 1.0 blade1 only 2;    #4 Blade 1 Flapwise tip deflection [m]
   mbdy state pos blade2  26 1.0 blade2 only 2;    #5 Blade 2 Flapwise tip deflection [m]
   mbdy state pos blade3  26 1.0 blade3 only 2;     #6 Blade 3 Flapwise tip deflection [m]
   constraint bearing1 shaft_rot 2 only 1;          #7 Rotor azimuth angle [deg]
   general constant 0;               #8 TTT Amplitude [m]
   general constant 0;               #9 TTT Phase [deg]
   general constant 0;   #debug mode. 1= debug. 0=do not debug
end output;
end type2_dll;
```

In the `init` section, the controller gains are defined as described in Table B.1. In this example, the C_{f1p} controller is implemented (Appendix A). The `output` section defines the inputs of the controller, which are the CPC pitch demand for each blade (output 1, 2, and 3), and the flapwise tip deflection measurement (output 4, 5, 6). Output 7 to 9 are used for tip trajectory tracking, and can be set to zero if not in use.

Table B.1: `init` parameter definition

Constant	**Description**
1	N, the number of coefficients in the filter, or the order of the discrete controller transfer function. Note that it is assumed that the number of feed forward and feed-backward coefficients is the same and equal to N.
2	K_p, the proportional gain of the controller.
3 to 3+N	b_0, b_1, ..., b_{N-1}, the feed-forward coefficients.
$4+N$ to $4+2N$	a_0, a_1, ..., a_{N-1}, the feed-backward coefficients.

Appendix C

Full Blade Deformation Estimation

C.1 Strain Gauge and Tip Deflection Sensor Fusion

Although the analysis so far compares the use of either root strain gauges or tip deflection sensors, it is also valuable to investigate using both sensors in conjunction with each other. Using both sensors provides information about the root and the tip of the blade, which allows for better approximation of the state of the blade system. Whereas using one sensor allows for the first blade mode to be observed, using two sensors allows for the first two blade modes to be observed. To demonstrate this, consider Equations 3.4 and 3.5 derived in Section 3.1. These equations relate root bending moment, $M(t)$, tip deflection, $y(t)$ and the flapwise blade mode excitation $\alpha_i(t)$:

$$M(t) = EI_0 \sum_{i=1}^{\infty} \alpha_i(t) \left. \frac{\partial^2 \gamma_i}{\partial z^2} \right|_{z=0} \tag{C.1}$$

$$y(t) = \sum_{i=1}^{\infty} \alpha_i(t) \tag{C.2}$$

Now, considering only the first two modes ($i = 1, 2$), a linear set of equations connecting tip deflection, root bending moment and the first two mode amplitudes can be formed. For brevity, the following notation is introduced: $\gamma_i'' = \left. \frac{\partial^2 \gamma_i}{\partial z^2} \right|_{z=0}$

$$\begin{bmatrix} M(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} EI_0\gamma_1'' & EI_0\gamma_2'' \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \alpha_1(t) \\ \alpha_2(t) \end{bmatrix} \tag{C.3}$$

As the mode shapes are orthogonal, the curvature terms are not equal. Therefore the matrix is invertible, and therefore a transformation between state space and modal space can be found:

$$\begin{bmatrix} \alpha_1(t) \\ \alpha_2(t) \end{bmatrix} = \begin{bmatrix} \frac{1}{EI_0(\gamma_1'' - \gamma_2'')} & -\frac{\gamma_2''}{\gamma_1'' - \gamma_2''} \\ -\frac{1}{EI_0(\gamma_1'' - \gamma_2'')} & \frac{\gamma_1''}{\gamma_1'' - \gamma_2''} \end{bmatrix} \begin{bmatrix} M(t) \\ y(t) \end{bmatrix} \tag{C.4}$$

Therefore

$$u(t, z) \approx \frac{\gamma_1 - \gamma_2}{EI_0(\gamma_1'' - \gamma_2'')} M_0(t) + \frac{\gamma_1''\gamma_2 - \gamma_2''\gamma_1}{\gamma_1'' - \gamma_2''} y(t) \tag{C.5}$$

C.2 Full Blade Deflection Estimation

By measuring the amplitude of the first two modes, the flapwise blade deformation along the entire blade can be estimated if the first two mode shapes are known. The deformation of the entire blade could be useful

in estimating the wind field, the force distribution along the blade, and for better control system modelling. Although the implementation of these suggestions is not carried out in this project, a brief comparison of the different estimation methods is outlined in this section.

$$\tilde{u}_{\text{TD, RBM}}(t,z) \approx \frac{\gamma_1 - \gamma_2}{EI_0(\gamma_1'' - \gamma_2'')}(M_0(t) + C_1) + \frac{\gamma_1''\gamma_2 - \gamma_2''\gamma_1}{\gamma_1'' - \gamma_2''}(y(t) - y_p) \tag{C.6}$$

It was found that adding an constant offset, C_1, to the RBM measurement to account for steady error yielded better results. The tip deflection measurement is also offset by the amount of prebend in the blade, y_0. This estimation is compared to using a single tip deflection sensor and a single RBM sensor:

$$\tilde{u}_{\text{TD}}(t,z) \approx \gamma_1(y(t) - y_p) \tag{C.7}$$

$$\tilde{u}_{\text{RBM}}(t,z) \approx \frac{\gamma_1}{EI_0\gamma_1''}(M(t) - C_2) \tag{C.8}$$

To test the accuracy of this estimation, a HAWC2 simulation is run on the DTU10MW turbine model at a windspeed of 12m/s under with NTM (check if this abbreviation has been defined) for 30 minutes with position sensors along the entire length of the blade.

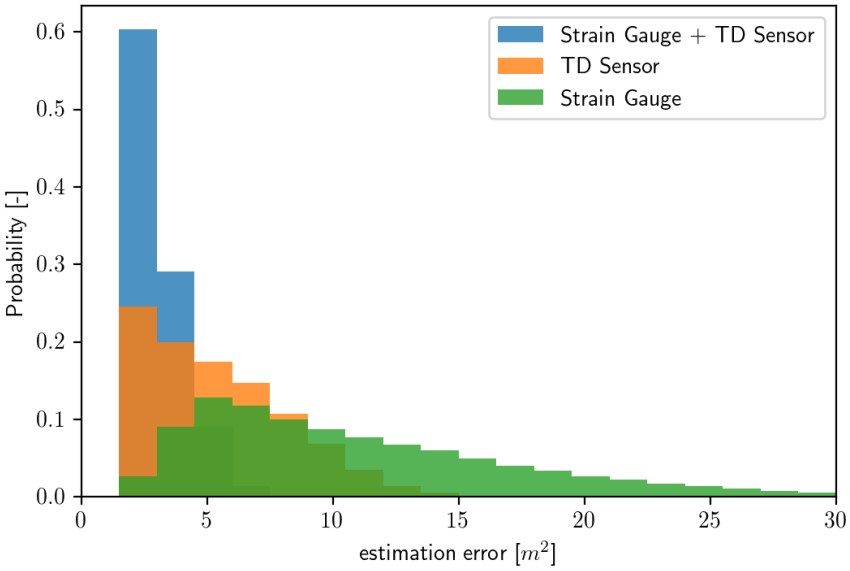
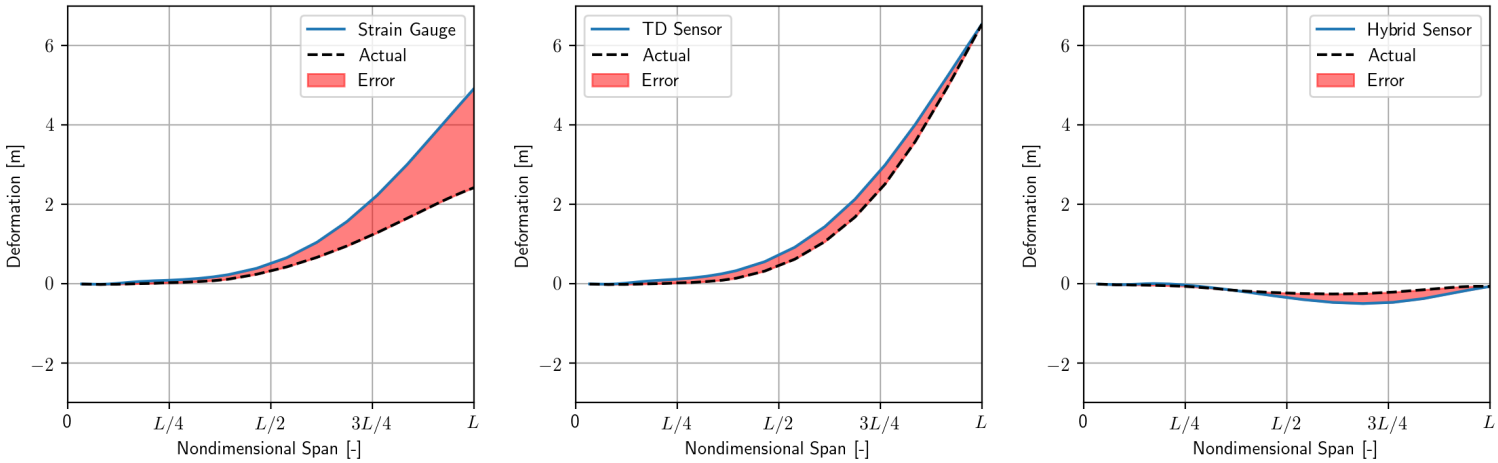


Figure C.1: Full blade deflection estimation error distribution for different combinations of sensors.

To evaluate the performance, the mean square error, E, is determined for each estimation. That is,

$$e(t) = \int_0^L |u(z,t) - \tilde{u}(z,t)|\, dz \tag{C.9}$$

The constant offsets, C_1 and C_2 were found such that the mean square error is minimised. The Mean square error of these estimations are shown in Table C.1. A tip deflection sensor is able to provide a better estimate of the blade deformation than an RBM sensor, however, using both sensors together provides a better overall result. Figures C.2 show the worst case estimations for each of the sensors. The RBM sensor tends to poorly estimate the deformation behaviour towards the blade tip. Discrepancies in the estimation are likely from higher mode excitation and non-linear bending. These factors could be accounted for by including additional sensors along the span of the blade or by using an alternative estimation method.



(a) Strain gauge

(b) Tip deflection sensor

(c) Hybrid sensing

Figure C.2: Worst case (highest error) estimates of each sensor combination ($U = 12m/s$, NTM).

Table C.1: Mean estimation error for each sensing method.

	Strain Gauge	TD Sensor	Hybrid Sensor
mean e $[m^2]$	11.2054	5.4354	2.9653