Repetitive Control of Individual Pitch to Reduce Wake Effect on Wind Turbines

Anders Pedersen & Kåre Engell Nørby

Aalborg University, 2011-2012

Aalborg University
Institute of Electronic Systems
Fredrik Bajers vej 7
9220 Aalborg Ø
Telephone 99 40 86 00
http://www.sict.aau.dk/

Title:

Repetitive Control of Individual Pitch to Reduce Wake Effect on Wind Turbines

Theme:

Control engineering

Project period:

P9-10, September 1st 2011 to May 31th 2012

Project group:

12gr1033

Participants:

Anders Pedersen
Kåre Engell Nørby

Supervisor:

Rafael Wisniewski, AAU
Peter Fogh Odgaard, kk-electronic a/s

Pages: 166

Copies: 5

Attachments: DVD

Finished: May 31, 2012

Synopsis:

As wind turbines get larger, with rotor diameters above 100 m, the blades will sweep a large wind field, containing different wind phenomenas such as, wakes, wind shear and tower shadow; applying a big structural load upon the wind turbine. In this thesis, a lifted repetitive controller is developed which reduces the structural loads by the use of individual pitching. For this purpose a dynamic model of a wind turbine has been developed. The model contains an aerodynamic model, mechanical model, a structural model and a model of the pitch system. The model has been linearized and validated in accordance with simulation code FAST.
From the model a lifted repetitive controller was design, by making a lifted system description where made, and from this a reduced output-feedback formulation was found, making it possible to use LQR design to calculate the controller gain.
In an acceptance test, the lifted repetitive controller was compared to the controller from FAST which was implemented in MATLAB. The results from this were, that the controller designed did not pass, even though the deflection of tower and blades were reduced. It is assumed that it is caused by a mismatch between the model and the implementation in MATLAB.

Aalborg Universitet
Institut for Elektroniske Systemer
Fredrik Bajers vej 7
9220 Aalborg Ø
Telefon 99 40 86 00
http://www.sict.aau.dk/

Titel:

Repetitive regulering af individuel pitch til reduktion af wake effekt på vindmøller

Tema:

Reguleringsteknik

Projektperiode:

P9-10, 1. september 2011 til 31. maj 2012

Projektgruppe:

12gr1033

Deltagere:

Anders Pedersen
Kåre Engell Nørby

Vejleder:

Rafael Wisniewski, AAU
Peter Fogh Odgaard, kk-electronic a/s

Sidetal: 166

Oplag: 5

Bilag: DVD

Afsluttet: 31. maj 2012

Synopsis:

Efterhånden som windmøller bliver større, med rotordiametre over 100 m, vil vingerne dække et større vindfelt som indeholder en række forskellige vindfænomener, såsom wakes, vindforskydning, tårnskygge, hvilket vil påføre store strukturelle laster på vindmøllen.
I denne these er en løftet repetitive regulator udviklet, som reducere de strukturelle laster ved brug af individuel pitching. Til dette formål er en dynamisk model af en vindmølle udledt. Modellen indeholder en aerodynamiskmodel, mekaniskmodel, en strukturelmodel og en model af pitchingsystemet. Modellen er blevet lineariseret og valideret i forhold til simulerings koden FAST.
Fra modellen er en løftet repetitive regulator blevet udviklet, ved at lave en løftet system beskrivelse, og herfra blev en reduceret output-feedback formulering fundet, hvilken lagde grunden til LQR designet af regulatoren.
I en accepttest blev den løftede repetitive regulator samlignet med regulatoren fra FAST som var implementeret i MATLAB. Resultatet herfra var at accepttesten ikke blev godkendt, selvom bøjningen af tårn og vinger blev reduceret. Det formodes at dette skyldes en forskel mellem implementeringen i MATLAB og modellen.

Preface

This thesis is written in the time period 1st of September 2011 to 31th of May 2012 by group 12gr1033, as the documentation of a master project under the specialization Automation and Control at Aalborg University.

The project proposal ”Non-linear repetitive control of wind turbines” was proposed by Peter Fogh Odgaard from kk-electronic and Rafael Wisniewski from Aalborg University, who both assisted as supervisors doing the project.

Furthermore should an acknowledgment go to Dr. Kathryn Johnson from Colorado School of Mines who assisted doing the first half of the project period.

References to the bibliography are denoted by [reference number] and they are included inside the report where needed. The bibliography is located at the end of the report.

A nomenclature is included at the beginning of the report. This contains expressions, abbreviations and variables used in the report. A DVD is attached to the report, and the content of the DVD is listed in Appendix G. Among the attachments is a digital version of this report, MATLAB and FAST files and some of the literature used.

Anders Pedersen

Kåre Engell Nørby

Contents

Nomenclature

The nomenclature includes expressions, abbreviations and variables used throughout the report. Further details on the terms are provided in the respective chapters and sections.

Abbreviations

Name	Description
LSS	Low speed shaft
HSS	High speed shaft
RC	Repetitive control
ILC	Iterative learning control
LQR	Linear-quadratic regulator

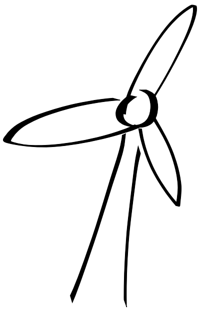
Variables

Symbol	Value	Description
α	rad	Angle of attack
α_{ws}		Wind shear coefficient
β_i	rad	Pitch angle of i^{th} blade
$\beta_{\text{ref},i}$	rad	Pitch angle reference of i^{th} blade
β_{twist}	rad	Pre-pitch of the blade
$\beta_{\text{pre,root}}$	rad	Pre-pitch of linearized root element
$\beta_{\text{pre,tip}}$	rad	Pre-pitch of linearized tip element
η_{G}		Efficiency of the generator
μ		Wake center vector
μ		RC average filter
ϕ	rad	$\alpha + \beta$
ϕ_{M}		Relative wind speed frequency distribution
Σ		Covariance determining width of Gaussian Wake
φ_i	rad	Azimuth angle of the i^{th} blade
φ_{wc}	rad	Angle to center of the wake
ρ	g/m^2	Air density

Symbol	**Value**	**Description**
τ_{G}	Nm	Loading torque from generator
$\tau_{\mathrm{G,ref}}$	Nm	Loading torque reference from generator
τ_{HS}	Nm	Torque acting on high speed shaft
τ_{LS}	Nm	Torque acting on low speed shaft
$\tau_{\mathrm{R},i}$	Nm	Rotational torque on the i^{th} blade element
$\tau_{\mathrm{R,Tot}}$	Nm	Rotational torque on the rotor
$\tau_{\mathrm{R,root}}$	Nm	Rotational torque on the rotor from blade root
$\tau_{\mathrm{R,tip}}$	Nm	Rotational torque on the rotor from blade tip
$\tau_{\mathrm{time,G}}$	s	Time constant for the generator
$\tau_{\mathrm{time,P}}$	s	Time constant for the pitch system
$\tau_{\mathrm{yaw,L}}$	Nm	Yaw torque from left rotor half plane
$\tau_{\mathrm{yaw,R}}$	Nm	Yaw torque from right rotor half plane
$\tau_{\mathrm{yaw,root}}$	Nm	Yaw torque from blade root
$\tau_{\mathrm{yaw,tip}}$	Nm	Yaw torque from blade tip
τ_{yaw}	Nm	Yaw torque
$\tau_{\mathrm{yaw,z},i}$	Nm	Yaw torque around the z-axis from the i^{th} blade
θ_{H}	rad	Angle of blade bending
θ_{R}	rad	Angle of low speed shaft
θ_{G}	rad	Angle of high speed shaft
θ_{δ}	rad	Torsion angle of drive train
$\dot{\theta}_{\mathrm{G}}$	rad/sec	Angular velocity of the generator
$\dot{\theta}_{\mathrm{R}}$	rad/sec	Angular velocity of the rotor
$a_{\mathrm{F,root}}$		Fitting factor for thrust force on root of blade
$a_{\mathrm{F,tip}}$		Fitting factor for thrust force on tip of blade
$a_{\tau,\mathrm{root}}$		Fitting factor for rotor torque on root of blade
$a_{\tau,\mathrm{tip}}$		Fitting factor for rotor torque on tip of blade
$\mathcal{A}'$		Output-feedback formulation system matrix
A		Weibull scaling factor
A_{L}		Linearized system matrix
A_{root}	m^2	Area of linearized root element
A_{tip}	m^2	Linearizedea of linearized tip element
$\mathcal{B}'$		Output-feedback formulation input matrix
B		Number of blades
B_{b}	N/(m/s)	Blade spring component of mass-spring-damper system
B_{t}	N/(m/s)	Tower spring component of mass-spring-damper system
B_{R}	Nm/(rad/s)	Viscous friction of low speed shaft
B_{G}	Nm/(rad/s)	Viscous friction of high speed shaft
B_{DT}	Nm/(rad/s)	Torsion damp coefficient of drive train
B_{L}		Linearized input matrix
c	m	Airfoil chord length
c_{bl1}	m	Linearized chord length
c_{bl2}	m	Linearized chord length
$\mathcal{C}'$		Output-feedback formulation output matrix

Symbol	**Value**	**Description**
C_{D}		Drag coefficient
C_{L}		Lift coefficient
C_{L}		Linearized output matrix
dr	m	Width of blade element
$\mathcal{D}'$		Output-feedback formulation feedforward matrix
D_{L}		Linearized feedforward matrix
e_j		RC error vector
f_j		RC feedforward signal vector
F		Lifted system matrix
F_{D}	N	Drag force
F_{L}	N	Lift force
F_{R}	N	Rotational force
$F_{\text{R,root}}$	N	Rotational force acting on blade root
$F_{\text{R,tip}}$	N	Rotational force acting on blade tip
F_{T}	N	Thrust force
$F_{\text{T,root}}$	N	Thrust force acting on blade root
$F_{\text{T,tip}}$	N	Thrust force acting on blade tip
$F_{\text{T,blade},i}$	N	Thrust force acting on the i^{th} blade
$F_{\text{tip},i}$	N	Thrust force acting on the i^{th} blade tip
$\tilde{F}_{\text{tip},i}$	N	Thrust force acting on the tower from the i^{th} blade tip
$F_{\text{root},i}$	N	Thrust force acting on the i^{th} blade root
G_{p}		Lifted pitch input matrix
G_{w}		Lifted wind input matrix
H		Lifted output matrix
H_0	m	Hub height above ground
$\mathcal{J}$		LQR cost function
J		Lifted feedforward matrix
J_{G}	kgm^2	Moment of inertia of the high speed shaft and generator
J_{R}	kgm^2	Moment of inertia of the low speed shaft and rotor
k		Weibull form factor
K_{DT}	Nm/rad	Torsion stiffness of drive train
K_{b}	N/m	Blade spring component of mass-spring-damper system
K_{t}	N/m	Tower spring component of mass-spring-damper system
L		RC learning filter
l		Outputs from the system
l_{root}	m	Length of blade root
l_{tip}	m	Length of blade tip
m		Number of inputs to the system
M_{b}	kg	Blade mass component of mass-spring-damper system
M_{t}	kg	Tower mass component of mass-spring-damper system
M_{out}		RC output filter
n		Number of states in the system
N		Number of elements in the blade element theory

Symbol	Value	Description
N		Period length i repetitive control
N_{G}		Gear ratio of the drive train
P		Controllability gramian
Q		LQR state weighting matrix
Q		Observability gramian
Q		RC robustness filter
r	m	Distance to the hub center
r_i	m	Distance from rotor axis to the i^{th} element
r_{t}	m	Tower radius
$r_{\mathrm{tip},i}$	m	Real distance to the i^{th} tip element
r_{wc}	m	Distance between the rotor center and wake center
r_{wake}	m	Radius of the wake
r_{w}	m	Distance to widest element of blade
r_{H}	m	Distance to hinge place on blade
r_{root}	m	Distance to linearized root element
r_{tip}	m	Distance to linearized tip element
R		LQR input weighting matrix
R	m	Length of rotor blade
$\mathcal{S}'_{\mathrm{RC}}$		Output-feedback formulation system
T_{red}		Reduction transformation matrix
u_{lin}		Linear input vector
u_{non}		Nonlinear input vector
v_{w}	m/s	Weibull wind speed
V	m/s	Free stream wind velocity
V_{F}	m/s	Full wind field
V_{H}	m/s	Wind speed at hub height
V_{M}	m/s	Wind field from mean wind
V_{S}	m/s	Wind field from wind shear
V_{T}	m/s	Tower shadow wind field
V_{W}	m/s	Wake wind field
V_{rel}	m/s	Relative wind velocity
V_{root}	m/s	Wind speed in linearized root element
V_{tip}	m/s	Wind speed in linearized tip element
W		RC output weighting matrix
W_S		Wind speed disturbance in wind shear expression
x		State vector
$\mathcal{X}'$		Output-feedback formulation state vector
x_N		Lifted state vector
$x_{\mathrm{b},i}$	m	Deflection of the i^{th} blade
x_{t}	m	Deflection of the tower
x_{m}	m	Distance from the tower midline to the rotor blade
y_{j}		RC measured output vector
z_{j}		RC output vector



Chapter 1

Introduction

In the last decades, there has been a lot of focus on changing the source of energy from a basis of coal and fossil fuels to a greener and independent energy source. There exist a lot of alternative solutions to meet this desire, one of the more mature solutions is the wind turbine. As it can be seen in Figure 1.1, the wind turbine technology has been a subject for a huge development and is today a strong competitor in creating sustainable power.

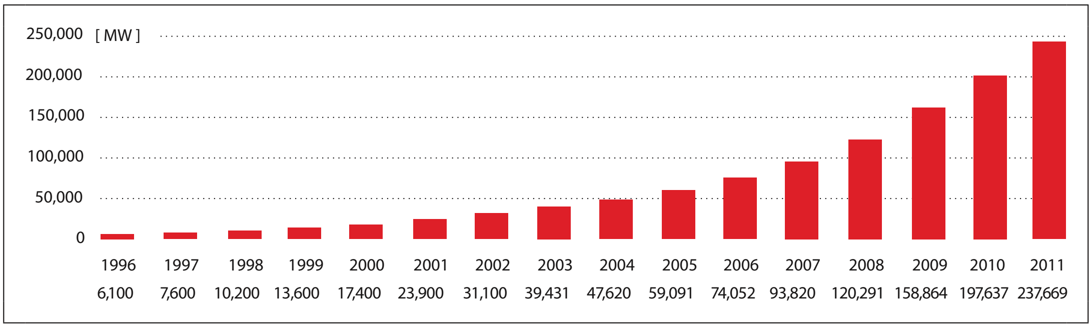


Figure 1.1: Global cumulative installed wind capacity 1996-2011 [11, p. 15]

As the development of the wind turbine continues, the size of the wind turbines have increased, as illustrated in Figure 1.2. Increasing the size of the turbines induces a larger amount of power, torque and stress on the wind turbine.

Placing several wind turbines together in wind farms, has advantage that they can share infrastructure. A downside of the wind farms, is the wind turbines in front can create a turbulent wake for the wind turbines behind. If a turbulent wind hits a wind turbine it can cause an unbalanced structural load, which can with time damage the structure of the wind turbine.

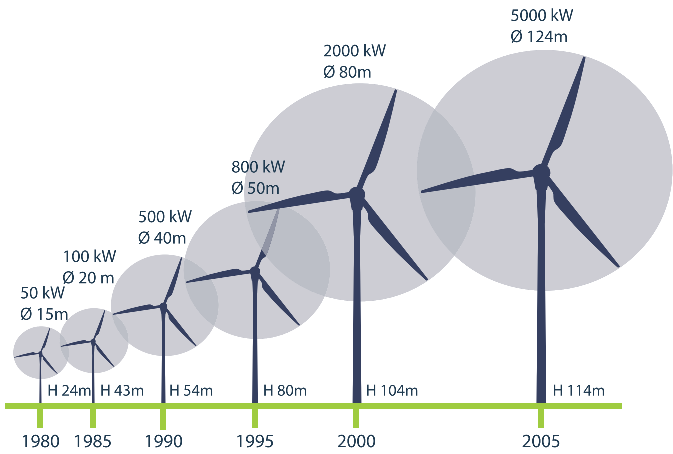


Figure 1.2: Increase in size of wind turbines in recent years [7, p. 51]

Figure 1.3 shows how a wind field can change over the range of the rotor of the wind turbine.

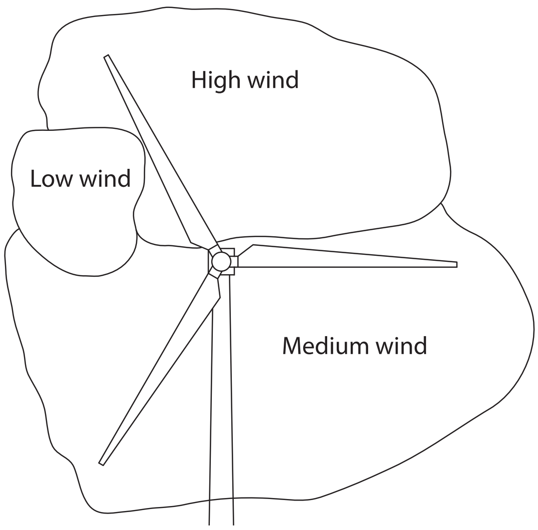


Figure 1.3: Wind turbine in an inhomogen wind field

As the blades rotate in an inhomogeneous wind field, the blades will experience a variation in the wind speed and thereby a variation in rotor torque, yaw torque and thrust force. Variations in the torque from the rotor can cause an unbalanced load on the shaft and tower of the wind turbine. The result of this is more stress on the structural parts of the wind turbine.

A way to reduce the unbalanced load is to use pitch actuators to pitch the single blade out of the wind, when it experience a high load. This reduces the load from the blade in the high wind area and thereby reduces the structure load which implies, that the wind turbine can last for a longer time, or costs of the production can be lowered [26, p. 50]. However, the use of pitch actuators can not be exaggerated as they will wear out to faster.

To reduce the wear of the pitch actuator, a controller, taking this into consideration, is to be designed. Furthermore, this controller can use information from the measurements of the previous blade, which has just passed this area of the wind field, making use of the repetitive information available.

1.1 Simulations using the FAST code

Since a real wind turbine is not available for physical tests in this project, the FAST code is used for simulating a wind turbine [23]. The FAST code is capable of modeling two and three bladed horizontal-axis wind turbines, and the loads affecting it [15]. In this project, both the Simulink in MATLAB and command prompt will be used to run the FAST code, which enables the implementation of advanced controls.
In this project, FAST is used for model validation, parameter extraction and lastly to test the final controller developed. All FAST codes used for testing is attached to the report on a DVD, and references are made to this where used.
In Appendix A, the different FAST files and the procedure to run FAST is described.

The FAST code can though only be used for wind fields without wakes; therefore, it can only be used for test of controller in homogen wind fields.

1.2 Previous work

In recent decades, a lot of papers have been published concerning the modeling and control of wind turbines. At Aalborg University several projects have been carried out as well. Particularly, one of them is of interest for this project, as this group have been working on the same problem:

- *Repetitive Individual Pitch Model Predictive Control for Horizontal Axis Wind Turbine* (2010) [10]
  Part of the FAST simulator was implemented in MATLAB. A repetitive MPC controller was developed and tested

It is this previous project this project is based on, mainly concerning the modeling of the wind turbine.

1.3 Project outline

To get an overview of the structure in this project report, an outline of the report will now be given.

Firstly, the requirements for the controller, the problem statement, and the project delimitation are determined in Chapter 2.

In Chapter 3 to 9, a nonlinear model of the wind turbine is described. The parameters for the nonlinear model is estimated in Chapter 10, and the model is combined in Chapter 11. In order to use the linear controller designs a linear model is required, this is derived in Chapter 12. In Chapter 13, a validation of both the nonlinear and linear model is performed, against the FAST code to see how accurate the nonlinear and linear models are.

The controller will be designed, corresponding to the requirements for the system, in Chapters 14 to 16, and in Chapter 17 the implementation in Matlab is described. Furthermore, in Chapter 18 the controller is compared to the controller from the FAST code to see if the requirements for the project are meet.

Finally, Chapter 19 and 20 comprise the conclusions and discussions of the project.

Chapter 2

Requirements

To investigate, if it is possible to make a controller to individually control the pitch actuators with focus on minimizing the yaw load on the wind turbine, minimizing the pitch actuation and without lowering the power output, the overall requirements of the project must be specified.

In this project, a controller will be made, which have the following requirement, in prioritized order:

- Lower structural load with main focus on yaw torque
- Less pitch actuation
- Similar power output

compared to the WindPACT 1,5 wind turbine in FAST, which is a 1,5 MW 3-bladed upwind baseline turbine [15, p. 121], described in Appendix B.

In brief, the problem statement for this project will be:

> *Can a controller, using individual pitch control, be developed, to reduce the yaw torque under the presence of wake, which is better than the one for the Baseline wind turbine in the FAST code?*

Final and more specific requirements for the controller will be outlined later when the model has been made, and the controller structure is determined in Chapter 14.

2.1 Delimitation

In order to limit the size of the project, some delimitation has been made:

- Yaw freedom has been disabled, as this simplify the equations. Thereby, the wind field is always placed orthogonal on the rotor field
- Wind speed is set to have a basis of 15 m/s. This is above the rated wind speed for the wind turbine, which means the primary function of the controller is to limit the structurally load and keep the rotational velocity

Part I

Modeling

Chapter 3

Model introduction

As described earlier, it is desired to make a controller and simulations based on a model of a wind turbine. The model, which will be developed, must describe the significant behavior of the wind turbine to make sure that the controller is developed on an appropriate base. As this project is a sequel of the project [10], the model is made with inspirations from this.

The model can be divided into minor submodels, as illustrated in Figure 3.1.

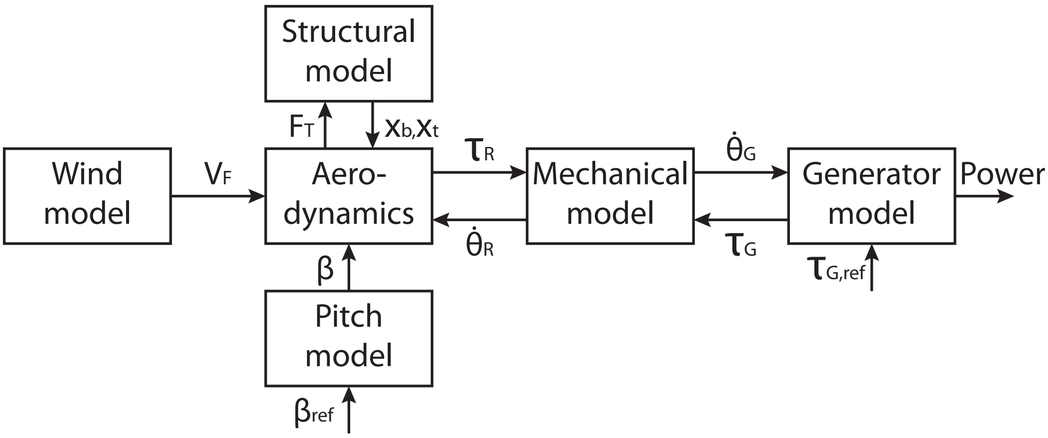


Figure 3.1: Block diagram of overall model structure

In the figure, the inputs and outputs from the different submodels can be seen. And the general structure of the entire model is also shown.

3.1 Model structure

To get insight into the model structure, the submodels will now shortly be described including the inputs and outputs of the different parts.

Wind model

The first part of the overall structure is the wind model, this submodel is not a part of the actual wind turbine, but it generates an input to the wind turbine, being the wind field V_{F}.
The wind model is based on some basic wind phenomena; wakes, which occurs in the shadow of obstacles in front of the wind turbine, wind shear, which describes the variation in wind in correlation with the height, and tower shadow. The wind model will be described in Chapter 4.

Aerodynamic model

The wind field from the wind model is in the aerodynamic model converted into forces acting on the blades. The aerodynamic model is based on blade element theory, which makes it possible to calculate a rotational torque delivered to the drive train τ_{R} and a thrust force F_{T} affecting the blade and thereby the tower. These forces are calculated on basis of the pitch angle of the blades β, which is one of the control inputs that can be made on the wind turbine, the deflection of the blades x_{b}, and the angular velocity of the rotor $\dot{\theta}_{\mathrm{R}}$. The aerodynamic model is presented in Chapter 5.

Structural model

When the wind hits the wind turbine, the wind will induce some forces on the tower and blades forcing them to bend. The deflections, of the tower and blades are modeled using the hinge principle, where a hinge is introduced to model the deflection according to the forces applied, the properties of the tower and blades is then calculated using a mass-spring-damper system.
In addition, the deflection of the blades and tower, x_{b} and x_{t}, is used in the aerodynamic model to find the forces affecting the blades. In Chapter 6, the full model of the structural elements is presented.

Mechanical model

The mechanical model converts the rotational torque, τ_{R}, generated by the aerodynamics into electrical power using the generator. The mechanical part of the wind turbine consists of a gear, which converts the angular velocity of low speed shaft to a higher velocity of the high speed shaft, which drives the generator. The total mechanical system is described in Chapter 7.

Pitch system model

A way to control the wind turbine is by pitching the blades with the angle β. A model of the pitch system is made as a first order model, which describes the pitch angle β, from the reference β_{ref}. The pitch system is described in Chapter 8.

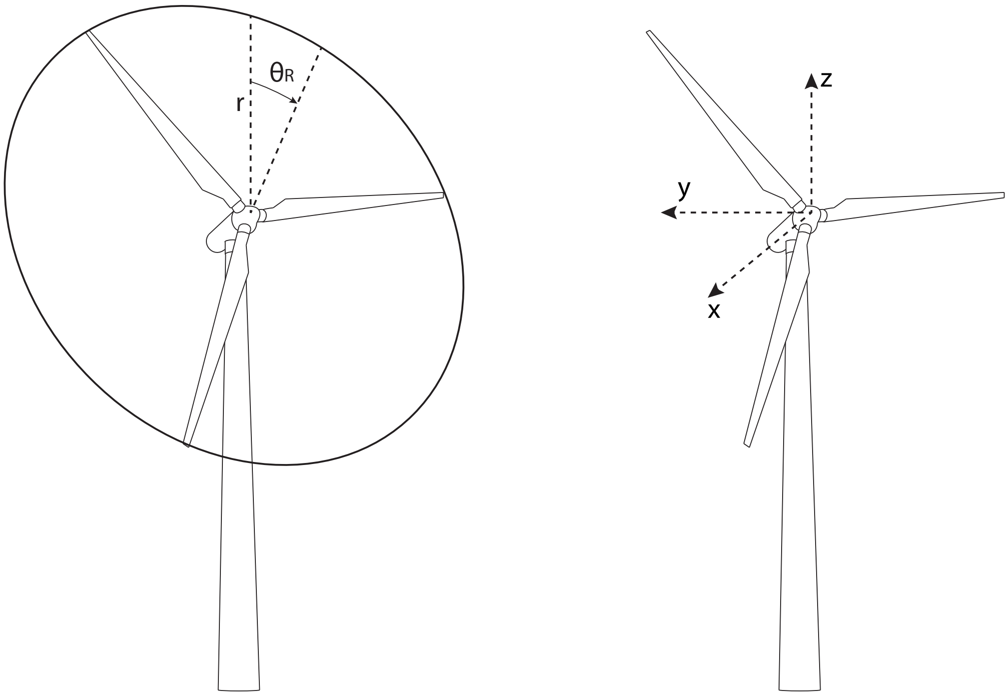
Generator model

Beside pitching the blades, the wind turbine can be controlled by changing the torque of the generator. The generator is controlled from a torque reference $\tau_{\text{G,ref}}$, which leads to the actual torque τ_{G}. The generator model is, as for the pitch system, modeled as a first order system.

The overall structure and the submodels have been described, and the actual modeling of the system can begin. This will be presented in further details in the following chapters, beginning with the wind model. Subsequently, extraction of some of the parameters will be done from FAST. Firstly, the coordinate systems used in modeling the wind turbine will be determined.

3.2 Coordinate systems

In the modeling process, it is necessary to have defined a coordinate system. In Figure 3.2, both a polar and a cartesian coordinate system are defined.



(a) Polar coordinate system **(b)** Cartesian coordinate system

Figure 3.2: Defined coordinate systems

In the polar coordinate system from Figure 3.2a, the coordinate system is the rotor plane and has its center in the hub, from here the length r and the angle θ is defined according to the vertical line up from the hub.
The cartesian coordinate system also has its origin in the hub, and have the axis places as shown in the figure.

Chapter 4

Wind model

A wind model will now be developed. This is used to calculate how the wind behaves. It is complicated to model a wind field as it dependents on a lot of factors such as, time of year, time of day, weather, shape of landscape, and geographical placement, making it hard to validate the model. The wind model is used as an input for the aerodynamic model described in Chapter 5.

The wind field is modeled as a sum of different components, as shown in equation 4.1.

$$V_{\mathrm{F}}(r, \theta_{\mathrm{R}}, t) = V_{\mathrm{M}}(t) + V_{\mathrm{S}}(r, \theta_{\mathrm{R}}, t) + V_{\mathrm{T}}(r, \theta_{\mathrm{R}}, t) + V_{\mathrm{W}}(r, \theta_{\mathrm{R}}, t), \tag{4.1}$$

where V_{M} is the mean wind, V_{S} the wind shear, V_{T} is the effect of the tower shadow, and V_{W} is the wakes from other wind turbines standing in front of the current turbine. The different parts of the wind field will now be described.

4.1 Mean wind

The mean wind is the average wind speed at the moment. It changes depending on time of the season, the weather and the geographical placement. Even though there is a dependence on the surroundings, it has been shown that a Weibull distribution can model the mean wind well [4, p. 14]. A Weibull distribution is illustrated in Figure 4.1, where it can be seen from the graph denoted $d\phi/dv$, that the most frequent wind speed is about 6 m/s.

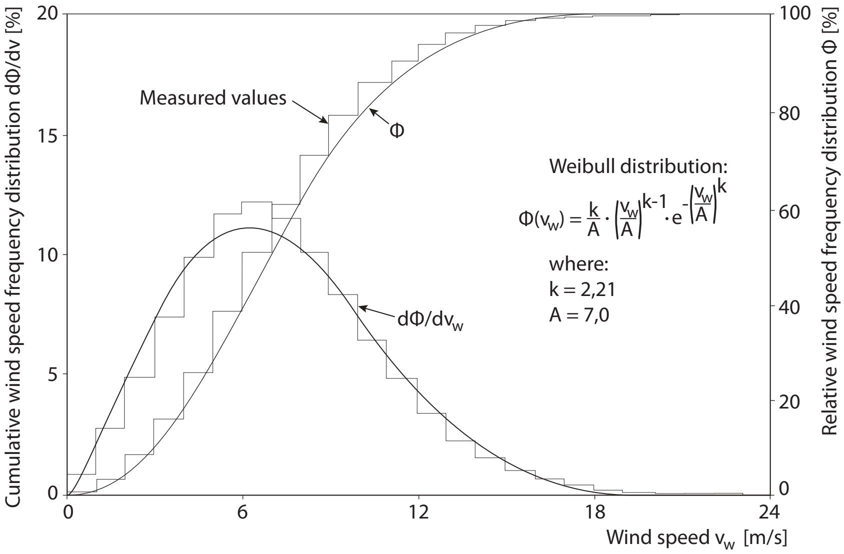


Figure 4.1: Example of a Weibull distribution showing the mean wind on the island of Sylt [12, p. 461]

The formula used to make this Weibull equation is 4.2, as shown in Figure 4.1:

$$\phi_{\mathrm{M}}(v_{\mathrm{w}}) = \frac{k}{A}\left(\frac{v_{\mathrm{w}}}{A}\right)^{k-1} e^{-\left(\frac{v_{\mathrm{w}}}{A}\right)^{k}} \tag{4.2}$$

where v_{w} is the wind speed, A is a scaling factor and k is a form factor.

Now a function has been found making it possible to make a realistic base of the mean wind speed in the wind model.

4.2 Wind shear

Generally the wind speed increases as a function of the height above the ground. This phenomenon is called wind shear, and an illustration is shown in Figure 4.2.

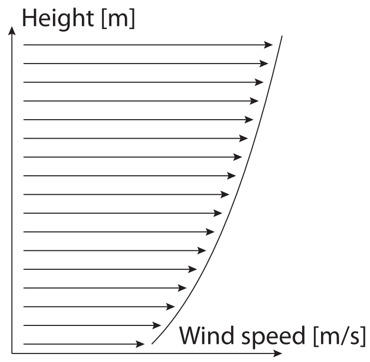


Figure 4.2: Wind shear effect

Wind shear can be modeled with the formula shown in equation 4.3 [6, p. 2].

$$V_{\mathrm{S}}(z) = V_{\mathrm{H}} \left(\frac{z}{H_0} \right)^{\alpha_{\mathrm{ws}}} \tag{4.3}$$

where $V_{\mathrm{S}}(z)$ is the wind shear wind speed, z is elevation above ground, H_0 is the hub height above the ground, V_{H} is the wind speed at hub height, and α_{ws} is the wind shear coefficient.
To analysis the effect of the wind field on a wind turbine, it is convenient to use a polar coordinate system. The transformation of equation 4.3 can be seen in equation 4.4.

$$V_{\mathrm{S}}(r, \theta_{\mathrm{R}}) = V_{\mathrm{M}}(t) \left(\frac{r\cos(\theta_{\mathrm{R}}) + H_0}{H_0} \right)^{\alpha_{\mathrm{ws}}} \tag{4.4}$$

$$= V_{\mathrm{M}}(t) \left(1 + W_{\mathrm{S}}(r, \theta_{\mathrm{R}})\right) \tag{4.5}$$

where $W_{\mathrm{S}}(r, \theta_{\mathrm{R}})$ is a wind speed disturbance, seen from the hub height H_0, and $V_{\mathrm{M}}(t)$ the mean wind found in Section 4.1. W_{S} can be approximated with a third order Taylor series. This is done in [6, p. 2], and the result is shown in 4.6.

$$W_{\mathrm{S}}(r, \theta_{\mathrm{R}}) \approx \alpha_{\mathrm{ws}} \left(\frac{r}{H_0} \right) \cos(\theta_{\mathrm{R}}) + \frac{\alpha_{\mathrm{ws}}(\alpha_{\mathrm{ws}} - 1)}{2} \left(\frac{r}{H_0} \right)^2 \cos^2(\theta_{\mathrm{R}}) + \frac{\alpha_{\mathrm{ws}}(\alpha_{\mathrm{ws}} - 1)(\alpha_{\mathrm{ws}} - 2)}{6} \left(\frac{r}{H_0} \right)^3 \cos^3(\theta_{\mathrm{R}}) \tag{4.6}$$

Three terms is necessary to get the effect of a wind shear. A second order Taylor series will only add wind shear as a constant to the mean wind.

The wind shear coefficients has been investigated by experiment in [27] and there is not at clear value for the wind shear coefficient, so a mean of 0,194 and variance of 0,137 has been picked based on the measurement from these experiments.

4.3 Tower shadow

In an upwind wind turbine, the wind first hits the rotor and afterward the tower, placed behind the rotor. This creates a decrease in wind which is called tower shadow, as illustrated in Figure 4.3

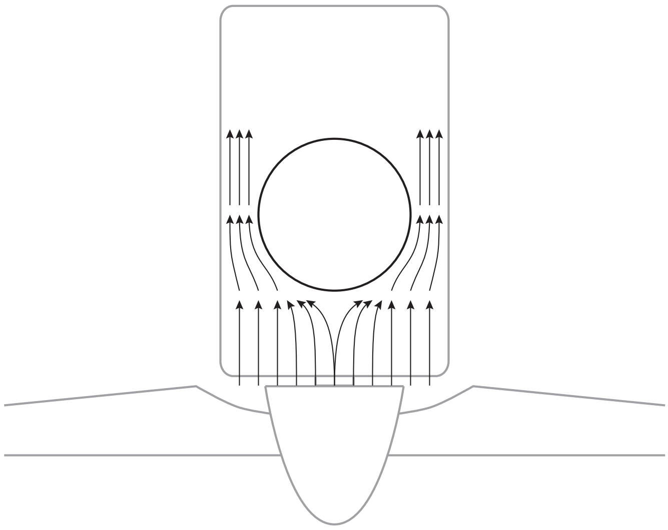


Figure 4.3: Principle of the wind field around the tower

The effect of the tower shadow is less for upwind wind turbines compared to a downwind turbine, but it still has to be considered in the wind model. The tower shadow effect can be modeled as in formula 4.7 [6, p. 3].

$$V_{\mathrm{T}}(r, \theta_{\mathrm{R}}) = m_{\mathrm{t}} r_{\mathrm{t}}^2 \frac{r^2 \sin^2(\theta_{\mathrm{R}}) - x_{\mathrm{m}}^2}{\left(r^2 \sin^2(\theta_{\mathrm{R}}) + x_{\mathrm{m}}^2\right)^2} \tag{4.7}$$

$$m_{\mathrm{t}} = 1 + \frac{\alpha_{\mathrm{ws}}(\alpha_{\mathrm{ws}} - 1) r^2)}{8 H_0^2} \tag{4.8}$$

where r_{t} is the tower radius, r is distance to hub center, θ_{R} is the azimuthal angle and x_{m} is the distance from the tower midline to the rotor blade. Equation 4.8 take the increasing mean wind from the shear effect into account, where α_{ws} is the wind shear coefficient from Section 4.2 and H_0 is the height of the rotor hub.

A graph has been made of the tower shadow equations with different r to see how the effect from the tower shadow varies along the blade, this is shown in Figure 4.3.

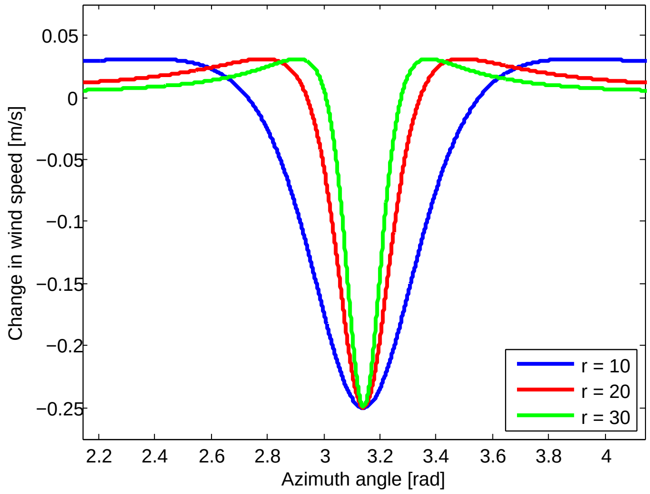


Figure 4.4: Effect of a tower shadow with different distances from the hub

It should be noted that the equations is only valid in the lower half plane, that is $\frac{\pi}{2} \leq \theta_{\mathrm{R}} \leq \frac{3\pi}{2}$ as the tower shadow in the upper half plane should is absent.

4.4 Wakes

As the wind turbine extracts energy from the wind, a wind field behind the wind turbine has decreased speed and increased turbulence. This is called a wake, and if the wake hits another wind turbine, this turbine will meet an irregular load on the full or part of the rotor plane. A wake typically has two states, it can be a near- or a far-wake. In this report, only far-wakes will be considered, because a certain distance between the wind turbines can be assumed. Two wake models has been derived a simple one and a more realistic model.

Simple wake model

A wake can be modeled as a circular part of the rotor plane, in which the wind speed is decreased, and the turbulence increased. An illustration of a wake in a rotor plane is shown in Figure 4.5.

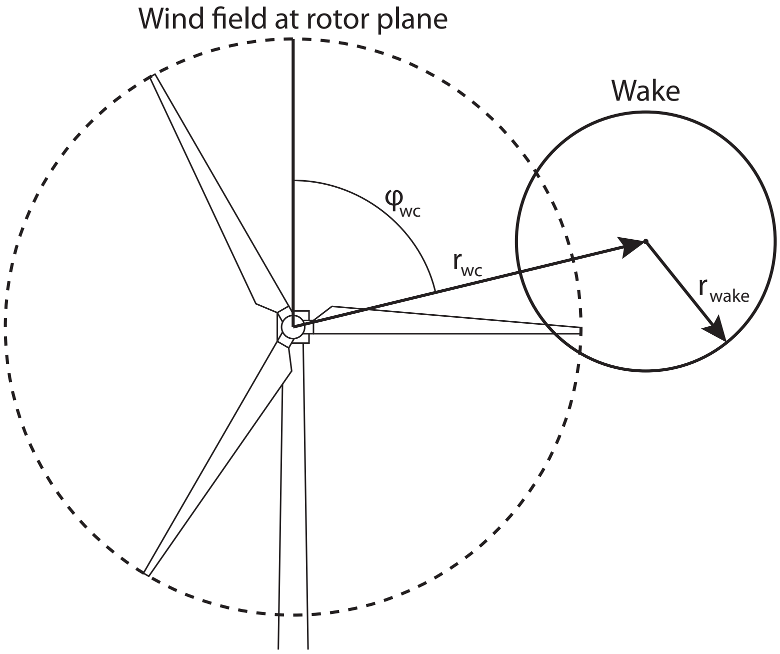


Figure 4.5: Wake in the wind field

The wake as shown in the figure can be described as equation 4.9 [10].

$$\begin{array}{lll} V_{\mathrm{W}} = f(r_{\mathrm{wake}}) & \text{if} & r^2 + r_{\mathrm{wc}}^2 - 2 \cdot r \cdot r_{\mathrm{wc}} \cos(\theta_{\mathrm{R}} - \varphi) - r_{\mathrm{wake}}^2 \leq 0 \\ V_{\mathrm{W}} = 0 & \text{else} & \end{array} \tag{4.9}$$

where

r	is the distance to a point in the wind field
θ_{R}	is the azimuth angle
φ_{wc}	is the angle to center of the wake
r_{wc}	is the distance between the rotor center and wake center
r_{wake}	is the radius of the wake
$f(r_{\mathrm{wake}})$	defines the wind behavior inside the wake

When a part of the wind field meets equation 4.9 some turbulence will be introduced and the wind speed will be decreased.

Wake with Gaussian distribution

In a real wake, the wind decreases as a Gaussian function, that is the wind speed is smallest in the center of the wake and increasing to the edge of the wake [17].

To create a more realistic model of a wake, the probability density function for a multivariate Gaussian distribution has been used to describe the intensity of the wake.

The probability density function of the multivariate Gaussian distribution is shown in equation 4.10. μ is a vector defining the center of the wake. Σ is the covariance, which determine the wide of the wake. k is the dimension of the distribution. Furthermore, a constant has to be multiplied to the probability density function in order to get a proper amplitude and direction of the wake.

$$f(x) = (2\pi)^{-\frac{k}{2}}|\Sigma|^{-0,5}e^{-0,5(x-\mu)^{\mathsf{T}}\Sigma^{-1}(x-\mu)} \tag{4.10}$$

x is the input $x = [\text{Radius Angle}]^{\mathsf{T}}$ in reference to the center of the rotor plane.
Σ has been estimated to $\left[\begin{smallmatrix} 80 & 0 \\ 0 & 80 \end{smallmatrix}\right]$, and k to 2. The probability density function is multiplied by -1000 in order to get the amplitude of the wake center to be around -1,9 m/s and the wind speed to be decreasing in the wake. In Figure 4.6, the Gaussian wake is illustrated.

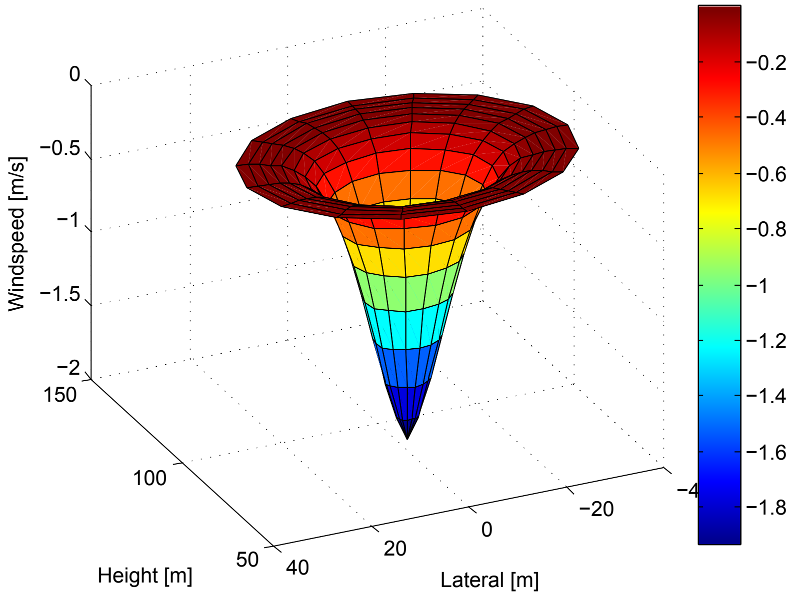


Figure 4.6: The Gaussian wake

Wake meandering

A wake behind a wind turbine is not moving in a linear manner, instead it is twisting both laterally and vertically as shown in Figure 4.7. The twisting behavior is called wake meandering [17].

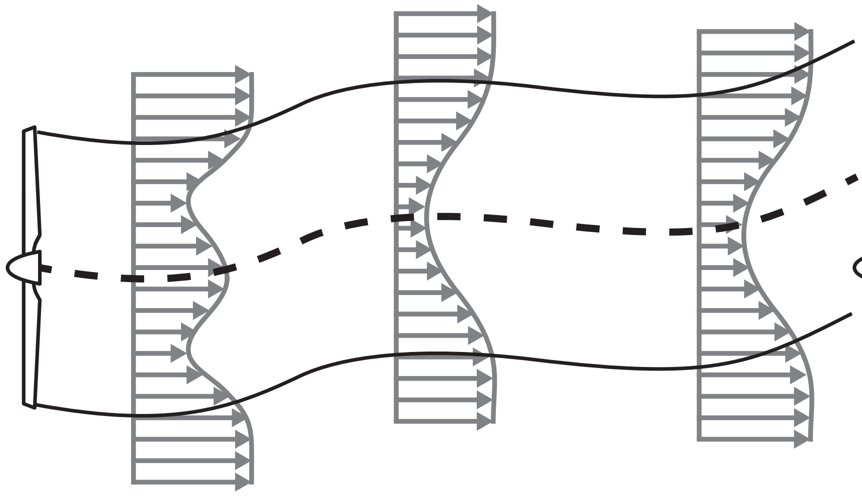


Figure 4.7: Wake meandering principal sketch [9]

Wake meandering is caused by turbulence in the air stream making the wake to move. When including a wake in the modeling it will be moving in a stochastic manner lateral and vertical, with a speed approximately the same as the wind speed in hub height. Though the movement of an actual wake is more complex, this is assumed to be appropriate for this wind model.

No test or validation data is available for the wake behavior, so the turbulent terms has been estimated by Gaussian noise with mean 0. The variance for the separate parts are: 0,01 for the distance to wake, 0,01 for the angle to the wake center, 0,01 for the radius for wake and 0,1 for the turbulent in the wake area.

4.5 Turbulence

Beside the stochastic elements, which have been added to the different parts of the wind model, some turbulence should be added. The overall turbulence is added to the whole wind field by a Gaussian noise with mean value 0 and variance 0,01. No literature have been found which determines the actual behavior of the turbulence, so for this wind model this has been found to be suitable.

4.6 Combined wind field

A wind model has been made in MATLAB and a surface plot is shown in Figure 4.8, the MATLAB code is included in folder `['Wind model']`.

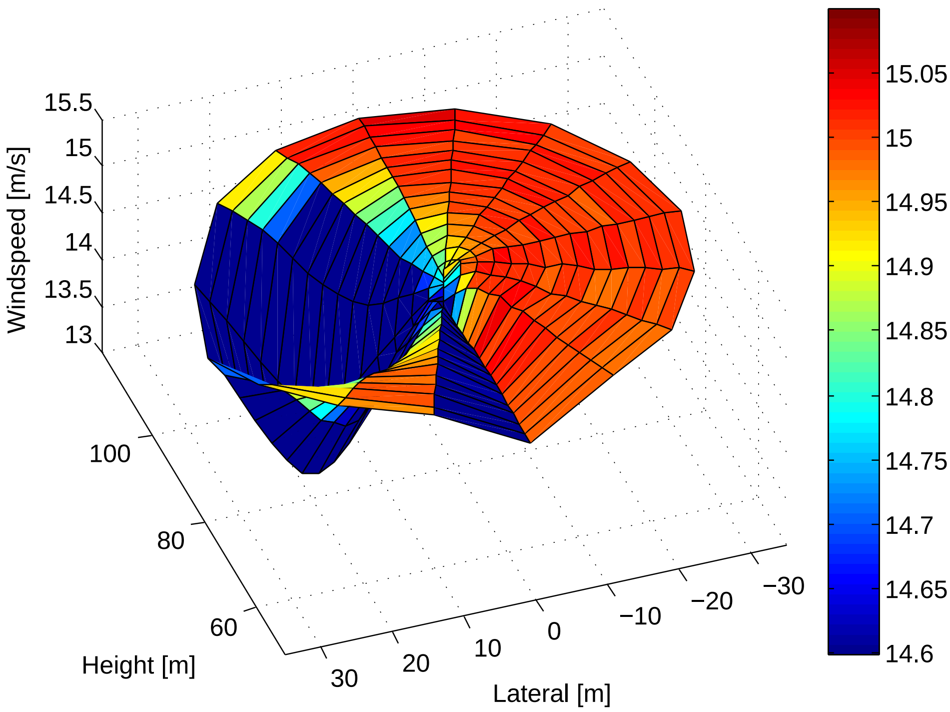
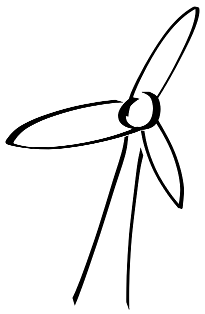


Figure 4.8: Plot of windfield with all elements included

In the plot, the tower shadow is shown in the lower part of the wind field, and a wake is generated to the left of the center. Further more is some turbulence present, which is also shown.

A wind model has now been designed for testing the wind turbine model, the wind model will not be validated as there is no data to compare the model with. In the control of the wind turbine, it can be preferred to ignore some parts of the wind field, this will be possible by removing single parts when the full wind field is summarized.

The modeling of the wind turbine will commence with the aerodynamic model.



Chapter 5

Aerodynamic model

In this chapter, it will be described how the incoming wind field will affect the blade aerodynamics, and induce a rotational torque for the drive train, which will be described in Chapter 7. Furthermore, the forces into the wind turbine will be used to compute the deflecting of the tower and the blades, described in Chapter 6. The inputs and outputs of the aerodynamic model can be seen in Figure 5.1.

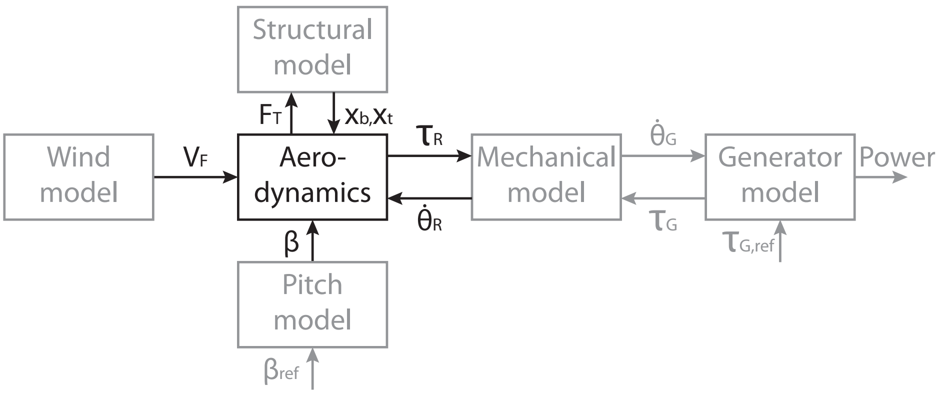


Figure 5.1: Block diagram showing the aerodynamic model inputs and outputs in the overall model structure

The forces induced by the wind will be found using the blade element theory which is described in the following section and afterwards it will be described how the wind and the acting forces on the blade causes the blade to bend.

5.1 Blade element theory

In this section, it is described how the forces generated by the wind is determined using the blade element theory. This theory is among others described in [20]. In this method, the blade is divided into N elements, and the forces affecting those elements is examined.

Firstly the blade is divided into minor elements of length dr as shown in Figure 5.2, where a part of the blade is divided into smaller elements.

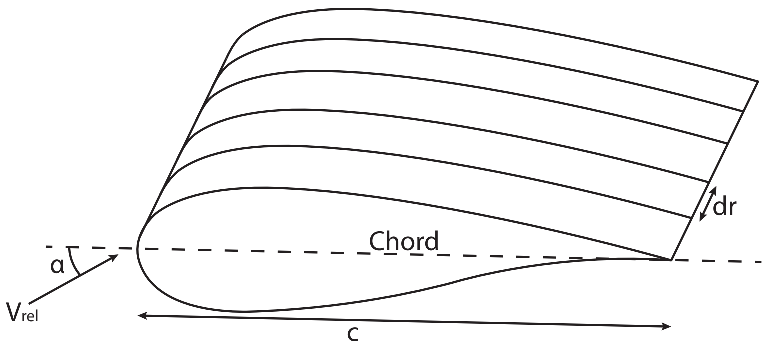


Figure 5.2: Division of the blade

The chord is the dotted line from end to tip of the blade, the length of this is denoted by c. The wind affecting the blade is V_{rel}, with the angle of attack α being the angle that the flow makes with the chord.
The relative wind velocity V_{rel} is the composition from the actual wind flow V and the air flow that appears from the rotation of the blade $\dot{\theta}_{\text{R}}r$, where $\dot{\theta}_{\text{R}}$ is the angular velocity of the rotor, and r is the distance from the center of the rotor to the blade element. The calculation of V_{rel} is shown in equation 5.1.

$$V_{\text{rel}}^2 = V^2 + (\dot{\theta}_{\text{R}}r)^2 \tag{5.1}$$

When the relative wind hits the blade two perpendicular forces are generated. The lift force F_{L} and the drag force F_{D}. These are illustrated in Figure 5.3, shown from a blade element viewed from the end, with the different wind flows illustrated as well.

The drag force is moving in the same direction as the air flow, and the lift force is perpendicular to these. The two forces for an element of width dr and length c can be calculated as shown in equations 5.2 and 5.3 [20, p. 120].

$$F_{\text{L}} = \frac{\rho c}{2} V_{\text{rel}}^2 C_{\text{L}}(\alpha) dr \tag{5.2}$$

$$F_{\text{D}} = \frac{\rho c}{2} V_{\text{rel}}^2 C_{\text{D}}(\alpha) dr \tag{5.3}$$

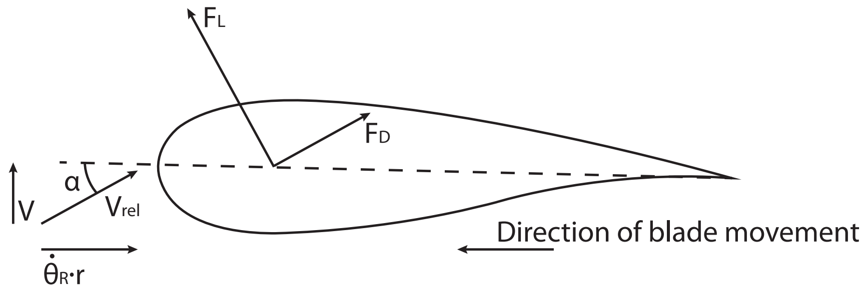


Figure 5.3: The lift and drag forces affecting the blade

Besides the dimension parameters the forces depends of air density ρ and the lift and drag coefficients, C_L and C_D. These coefficients varies depending on the shape of the airfoil and the angle of attack.

When the blade is pitched the lift and drag forces will not be respectively parallel and perpendicular to the rotational plane; therefore, they can be resolved into an tangential and axial component, F_R and F_T, as shown in Figure 5.4.

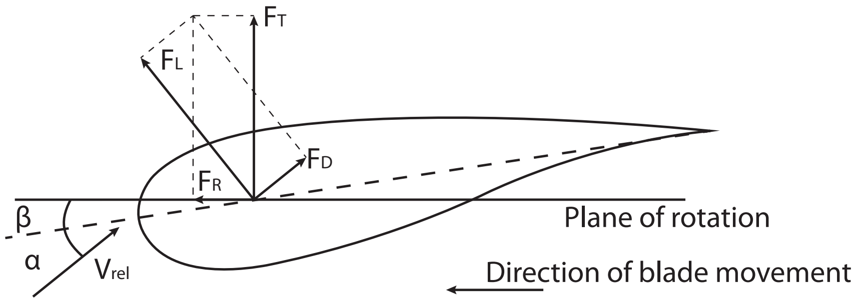


Figure 5.4: The rotational and thrust forces affecting the blade

It can be seen that the blade is pitched with an angle of β, compared to the plane of rotation, and furthermore the two forces F_R, which make the rotor rotate, and F_T, which is making a thrust at the structure, is shown. The forces are combined from F_L and F_D as shown in equations 5.4 and 5.6.

$$F_R = F_L \sin(\phi) - F_D \cos(\phi) \tag{5.4}$$
$$= \frac{\rho c}{2} V_{rel}^2 dr (C_L(\alpha)\sin(\phi) - C_D(\alpha)\cos(\phi)) \tag{5.5}$$

$$F_T = F_L \cos(\phi) + F_D \sin(\phi) \tag{5.6}$$
$$= \frac{\rho c}{2} V_{rel}^2 dr (C_L(\alpha)\cos(\phi) + C_D(\alpha)\sin(\phi)) \tag{5.7}$$

where $\phi = \alpha + \beta$, which can be calculated as

$$\phi = \tan^{-1}\left(\frac{V}{\dot{\theta}_{\mathrm{R}} r}\right) \tag{5.8}$$

$$\Leftrightarrow \quad \alpha = \tan^{-1}\left(\frac{V}{\dot{\theta}_{\mathrm{R}} r}\right) - (\beta + \beta_{\mathrm{twist}}) \tag{5.9}$$

where β_{twist} is the pre-pitching of the blade, which varies according to the position on the blade, r.

As it were seen from Figure 5.4, and the equations 5.4 and 5.6, both the lift and drag force contributes to the thrust force F_{T}. Whereas, only the lift force contributes to the rotational torque, and the drag force opposes it. It is therefore desirable for the ratio between the lift and drag coefficient, $\frac{C_{\mathrm{L}}}{C_{\mathrm{D}}}$, to be as high as possible to get a high efficiency. An example is shown in Figure 5.5.

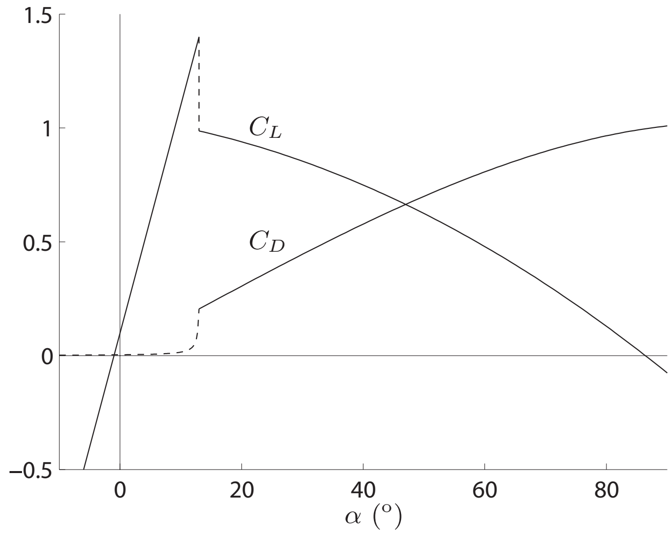


Figure 5.5: Typical drift and drag coefficients of an airfoil [2, p. 18]

The ratio increases as the angle of attack gets larger, but when the blade begins to stall, this makes an sudden drop in the ratio, which causes a significant drop in the rotational force. In the figure, it occurs around $\alpha = 13°$.
Stalling is when the airflow on the upper side of the blade is no more laminar and separates from the blade, this causes a wake to form above the blade.

The forces affecting a blade induces a torque of the rotor, $\tau_{\mathrm{R},i}$, this torque from the single element i, of the blade can be expressed as in equation 5.10.

$$\tau_{\mathrm{R},i} = F_{\mathrm{R},i} \cdot r_i \tag{5.10}$$

Where $F_{\mathrm{R},i}$ is the forces on the i^{th} element, and r_i the distance from the rotor center to the element.
To get the total torque from the whole rotor, the torque from the elements will be summarized, which is shown in equation 5.11.

$$\begin{aligned}
\tau_{\mathrm{R,blade},k} &= \sum_{i=1}^{N} F_{\mathrm{R},i} \cdot r_i &(5.11)\\
&= \sum_{i=1}^{N} \frac{\rho c_i}{2} V_{\mathrm{rel}}^2(r_i) dr (C_\mathrm{L}(\alpha)\sin(\phi) - C_\mathrm{D}(\alpha)\cos(\phi)) \cdot r_i &(5.12)
\end{aligned}$$

Where N is the number of elements the blade is divided into and c_i is the chord length of the i^{th} element. Because the wind field is not homogeneous, $\tau_{\mathrm{R,blade},k}$ has to be found for each blade, and summarized as in equation 5.13.

$$\tau_{\mathrm{R,tot}} = \sum_{k=1}^{B} \tau_{\mathrm{R,blade},k} \tag{5.13}$$

with B being the number of blades.

Likewise can the thrust force affecting the turbine structure be calculated as shown in equation 5.14.

$$\begin{aligned}
F_{\mathrm{T,blade},k} &= \sum_{i=1}^{N} F_{\mathrm{T},i} &(5.14)\\
&= \sum_{i=1}^{N} \frac{\rho c_i}{2} V_{\mathrm{rel}}^2(r_i) dr (C_\mathrm{L}(\alpha)\cos(\phi) + C_\mathrm{D}(\alpha)\sin(\phi)) &(5.15)
\end{aligned}$$

Now the forces affecting the blades has been described. However, in the used method, it is assumed that the blade is a rigid body. Due to the thrust forces the blade will bend, it will now be described how this bending can be modeled.

5.2 Bending of the blade

To model the bending of the wind turbine blade, the bending motion can be modeled by a hinge in the blade as shown in Figure 5.6.

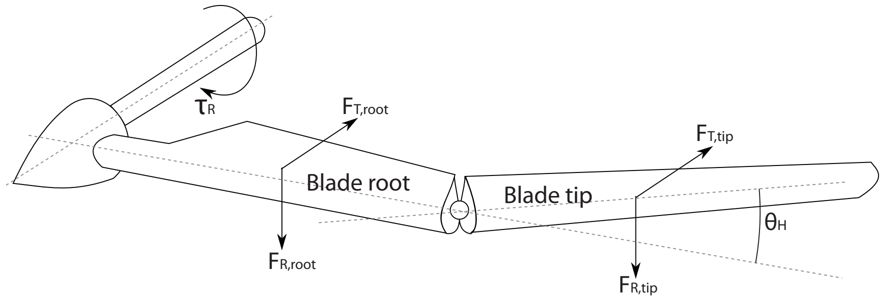
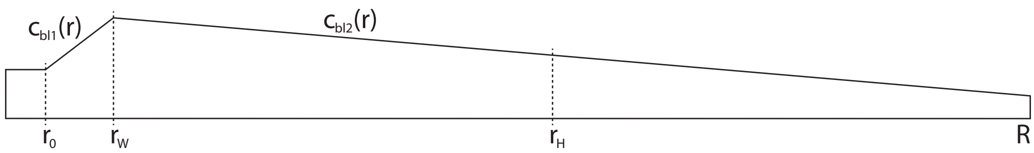


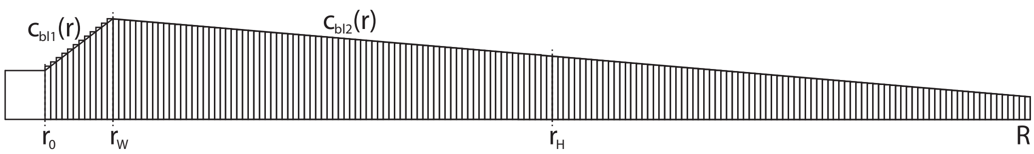
Figure 5.6: Hinge model [10]

This means that the model found in the previous section, can be used directly for the root, which is the part of the blade placed before the hinge, but because the blade is bended the wind will have a different effect on the tip of the blade which is bended.
To calculate the forces acting on the different parts, of the blade, it is divided into three regions as shown in Figure 5.7a, where the hinge is inserted at the r_{H} mark. The force affecting the blade will then be found by calculating three sums; from r_0 to r_{W}, r_{W} to r_{H} and r_{H} to R.

The part of the blade in the left of r_0 is the part connecting the blade to the hub, and it is assumed that the forces acting on this part are insignificant.



(a) Blade divided into main regions



(b) Blade divided into summation elements

Figure 5.7: Blade division

In Figure 5.7b, the blade is divided further in the elements that will be summarized when the forces are calculated.

If it is assumed that the blade is not bended the forces can be expressed as in equation 5.16.

$$\begin{bmatrix} F_{\mathrm{R}} \\ F_{\mathrm{T}} \end{bmatrix} = \left(\sum_{i=r_0}^{r_{\mathrm{W}}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) c_{\mathrm{bl1}}(r_i) dr + \sum_{i=r_{\mathrm{W}}+1}^{R} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) c_{\mathrm{bl2}}(r_i) dr \right) \cdot \begin{bmatrix} C_{\mathrm{L}}(\alpha)\sin(\phi) - C_{\mathrm{D}}(\alpha)\cos(\phi) \\ C_{\mathrm{L}}(\alpha)\cos(\phi) + C_{\mathrm{D}}(\alpha)\sin(\phi) \end{bmatrix} \tag{5.16}$$

The equation elements c_{bl1} and c_{bl2} are used to calculate the chord length by means of the length r. The equations can be written as:

$$c_{\mathrm{bl1}}(r_i) = a_1 r_i + b_1 \quad \text{and} \quad c_{\mathrm{bl2}}(r_i) = a_2 r_i + b_2 \tag{5.17}$$

In these equations, it is assumed that the sides of the blade are linear, but these equations can be changed to get nonlinear blade sides, if it wanted later in the modeling.

When the tip of the blade bends the relative wind will have a different effect on this part than it will on the not bending part, this is because the incoming wind V is no longer perpendicular on the blade tip. An illustration of this can be seen in Figure 5.8.

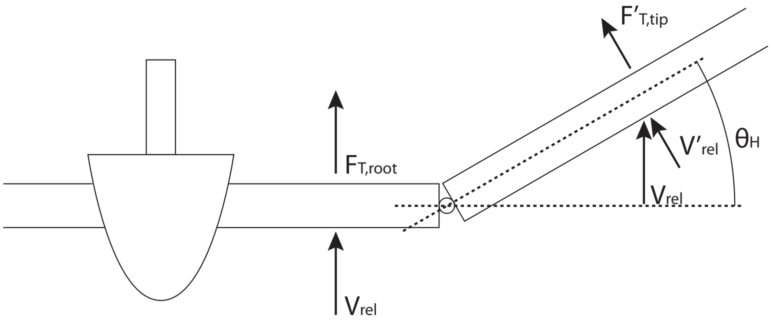


Figure 5.8: Wind turbine top view showing the changed relative wind V'_{rel}

In the figure, a new relative wind V'_{rel} is shown, this will be a function of the hinge angle θ_H. Furthermore, a new tip thrust force is drawn, $F'_{\mathrm{T,tip}}$, which is induced by the new relative wind. The calculation of the relative wind V'_{rel} is shown in equation 5.19, and an extended equation for V_{rel} is shown as well.

$$V_{\mathrm{rel}}^2(r_i) = (V - \dot{x}_{\mathrm{t}})^2 + (\dot{\theta}_{\mathrm{R}} \cdot r_i)^2 \qquad \text{for } r_i \leq r_{\mathrm{H}} \tag{5.18}$$

$$V_{\mathrm{rel}}'^2(r_i) = (\cos(\theta_{\mathrm{H}}) \cdot (V - \dot{x}_{\mathrm{b}} - \dot{x}_{\mathrm{t}}))^2 + (\dot{\theta}_{\mathrm{R}} \cdot r_i)^2 \qquad \text{for } r_i > r_{\mathrm{H}} \tag{5.19}$$

As it can be seen, the movement of the tower $\dot{x}_{\mathrm{t}}$ and the blade $\dot{x}_{\mathrm{b}}$ have been included in the equations because this movement can have a significant influence on the variables. $\dot{x}_{\mathrm{t}}$ and $\dot{x}_{\mathrm{b}}$ are described further in Chapter 6.

With the bending of the blade taken into account, α and ϕ, can be redefined as well, this is shown in equation 5.20 and 5.21.

$$\begin{aligned} \phi &= \tan^{-1}\left(\frac{V-\dot{x}_{\mathrm{t}}}{\dot{\theta}_{\mathrm{R}} r}\right) \\ \alpha &= \tan^{-1}\left(\frac{V-\dot{x}_{\mathrm{t}}}{\dot{\theta}_{\mathrm{R}} r}\right) - (\beta + \beta_{\mathrm{twist}}) \end{aligned} \qquad \text{for } r_i \leq r_{\mathrm{H}} \tag{5.20}$$

$$\begin{aligned} \phi' &= \tan^{-1}\left(\frac{V-\dot{x}_{\mathrm{b}}-\dot{x}_{\mathrm{t}}}{\dot{\theta}_{\mathrm{R}} r}\right) \\ \alpha' &= \tan^{-1}\left(\frac{V-\dot{x}_{\mathrm{b}}-\dot{x}_{\mathrm{t}}}{\dot{\theta}_{\mathrm{R}} r}\right) - (\beta + \beta_{\mathrm{twist}}) \end{aligned} \qquad \text{for } r_i > r_{\mathrm{H}} \tag{5.21}$$

where α' and ϕ' is variables for the tip of the blade.
Furthermore, the thrust forces for the root and tip of the blade can be calculated as in equation 5.22 and 5.23.

$$\begin{aligned} F_{\mathrm{T,root}} &= \left(\sum_{i=r_0}^{r_{\mathrm{W}}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{bl1}(r_i) dr + \sum_{i=r_{\mathrm{W}}+1}^{r_{\mathrm{H}}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr\right) \\ &\quad \cdot (C_{\mathrm{L}}(\alpha)\cos(\phi) + C_{\mathrm{D}}(\alpha)\sin(\phi)) \end{aligned} \tag{5.22}$$

$$\begin{aligned} F'_{\mathrm{T,tip}} &= \left(\sum_{i=r_{\mathrm{H}}+1}^{R} \frac{\rho}{2} V_{\mathrm{rel}}'^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr\right) \\ &\quad \cdot \left(C_{\mathrm{L}}(\alpha')\cos(\phi') + C_{\mathrm{D}}(\alpha')\sin(\phi')\right) \end{aligned} \tag{5.23}$$

As it were seen in Figure 5.8, the force $F'_{\mathrm{T,tip}}$ is no longer parallel to the wind inflow, which have to be taken into account when the total thrust force is found. This is done as in equation 5.24 and 5.25.

$$F_{\mathrm{T,tip}} = F'_{\mathrm{T,tip}} \cdot \cos(\theta_{\mathrm{H}}) \tag{5.24}$$

$$F_{\mathrm{T,blade}} = F_{\mathrm{T,root}} + F_{\mathrm{T,tip}} \tag{5.25}$$

If the force is found for all the blades, the total force affecting the tower can be calculated.

Finding the rotation force can be done in a similar way by summarizing the root and tip forces, this is done in equation 5.26 and 5.27.

$$F_{\mathrm{R,root}} = \left(\sum_{i=r_0}^{r_{\mathrm{W}}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{\mathrm{bl1}}(r_i) dr + \sum_{i=r_{\mathrm{W}}+1}^{r_H} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr \right) \cdot \left(C_L(\alpha)\sin(\phi) - C_D(\alpha)\cos(\phi)\right) \tag{5.26}$$

$$F'_{\mathrm{R,tip}} = \left(\sum_{i=r_{\mathrm{H}}+1}^{R} \frac{\rho}{2} V_{\mathrm{rel}}^{\prime 2}(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr \right) \cdot \big(C_L(\alpha')\sin(\phi') - C_D(\alpha')\cos(\phi')\big) \tag{5.27}$$

As done in equation 5.13, the torque induced by the blades can be calculated by multiplying the force on each element by the distance from the rotational center to the element.
Because of the bending of the blade, it is not possible to use the blade radius r_i directly. Therefore, it is necessary to find the real length to the element of the blade. If it is known how much the blade it bended, it is possible to find the angle θ_{H} and from this calculate the real distance form the center of the rotor to the i^{th} element, $r_{\mathrm{tip,i}}$. Firstly, the angle θ_{H} is found as:

$$\theta_{\mathrm{H}} = \sin^{-1}\left(\frac{x_{\mathrm{b}}}{l_{\mathrm{tip}}}\right) \tag{5.28}$$

When θ_{H} is found the real distance to the tip elements $r_{\mathrm{tip},i}$ can be calculated according to the distances illustrated in Figure 5.9.

$$r_{\mathrm{tip},i} = l_{\mathrm{root}} + l_{\mathrm{tip},i} \cdot \cos(\theta_{\mathrm{H}}) \tag{5.29}$$

Using the found $r_{\mathrm{tip},i}$ the rotational torque induced by the root and tip of the blade is found in equation 5.30 and 5.31.

$$\tau_{\mathrm{R,root}} = \left(\sum_{i=r_0}^{r_{\mathrm{W}}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{\mathrm{bl1}}(r_i) dr \cdot r_i + \sum_{i=r_{\mathrm{W}}+1}^{r_{\mathrm{H}}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr \cdot r_i \right) \cdot \left(C_{\mathrm{L}}(\alpha)\sin(\phi) - C_{\mathrm{D}}(\alpha)\cos(\phi)\right) \tag{5.30}$$

$$\tau_{\mathrm{R,tip}} = \left(\sum_{i=r_{\mathrm{H}}+1}^{R} \frac{\rho}{2} V_{\mathrm{rel}}^{\prime 2}(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr \cdot r_{\mathrm{tip},i} \right) \cdot \big(C_{\mathrm{L}}(\alpha')\sin(\phi') - C_{\mathrm{D}}(\alpha')\cos(\phi')\big) \tag{5.31}$$

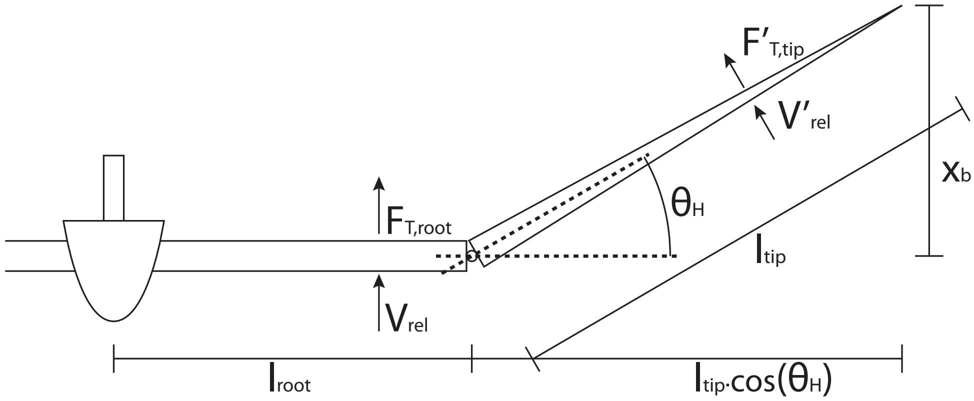


Figure 5.9: Illustration of the bended blade distances

The total torque for one blade is then calculated in equation 5.32.

$$\tau_{\mathrm{R,blade},k} = \tau_{\mathrm{R,root}} + \tau_{\mathrm{R,tip}} \tag{5.32}$$

In this chapter, the thrust force acting on the tower and blade structure was found, this will be used in Chapter 6 to calculate the deflection of the tower and the blades. Furthermore, the aerodynamic torque has induced by the wind flow found, this will be driving the generator through a drive train which will be described in Chapter 7.

As there are no parameters to fit in the nonlinear aerodynamic model, four factors will be multiplied to the output from the model to make it more similar to the output from the more complex FAST model, these parameters are $a_{\mathrm{F,root}}$, $a_{\mathrm{F,tip}}$, $a_{\tau,\mathrm{root}}$, and $a_{\tau,\mathrm{tip}}$, which is for the thrust force for the root and tip, and the torque for root and tip respectively. The fitting of these parameters will be implemented in Section 10.1.

Chapter 6

Structural model

In this chapter, some of the structural behaviors of the wind turbine will be described, namely the deflection of the tower and the blades due to the thrust force from the wind described in Chapter 5. The placement of the tower structures in the overall system, and the input and output from the system can be seen in Figure 6.1.

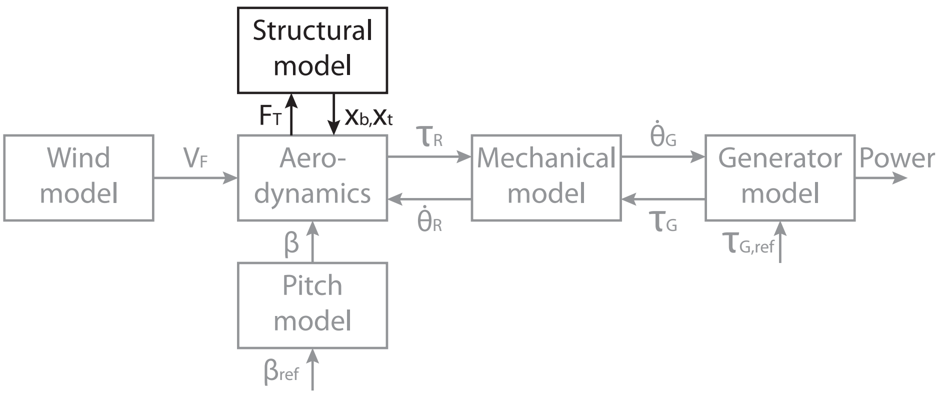


Figure 6.1: Block diagram showing the tower structures inputs and outputs in the overall model structure

The main input to the model of the tower is the thrust force, F_{T} from the blades, this force will make the tower and the blades deflect. This deflection of the blades, x_{b}, and tower x_{t} is the output of the tower model.
In this submodel, only the flap-wise bending of the blades, and the fore-aft movement of the tower is considered, because this have a significance to the model.

By inserting a hinge in each of the blades, as described in the Section 5.2, and one in tower as illustrated in Figure 6.2, the deflecting of the structures can be modeled as mass-spring-damper systems.

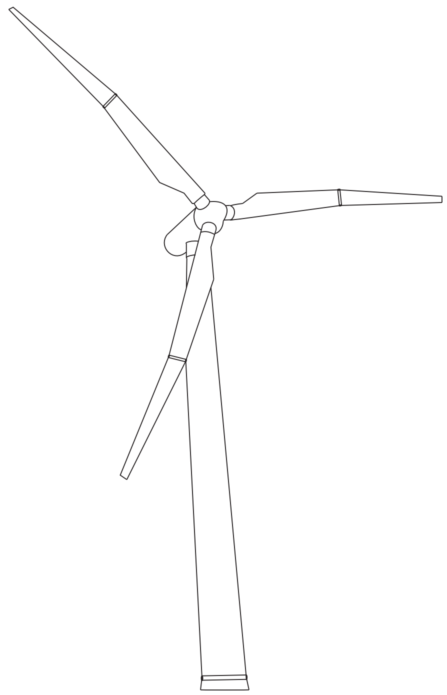


Figure 6.2: Wind turbine with virtual hinges inserted

The fore-aft movement of the tower consist of the mass of the whole wind turbine, this is divided into four masses, one for each blade tip $M_{\mathrm{b},i}$ and one for the tower including the masses of the blade roots, M_{t}. These masses are connected in a parallel mass-spring-damper system as shown in Figure 6.3.

The mass-spring-damper system for the blades is assumed to be the same for all three blades, thus the calculations for these are made for only one blade. The equations for the system will now be found. Firstly the equations for the blades is shown in 6.1, and is rearranged to find the force, $\tilde{F}_{\mathrm{tip},i}$, by which the tower is affected by each blade.

$$F_{\mathrm{tip},i} = M_{\mathrm{b},i}\ddot{x}_{\mathrm{b},i} + K_{\mathrm{b},i}(x_{\mathrm{b},i} - x_{\mathrm{t}}) + B_{\mathrm{b},i}(\dot{x}_{\mathrm{b},i} - \dot{x}_{\mathrm{t}}) \tag{6.1}$$

$$M_{\mathrm{b},i}\ddot{x}_{\mathrm{b},i} = F_{\mathrm{tip},i} - K_{\mathrm{b},i}(x_{\mathrm{b},i} - x_{\mathrm{t}}) - B_{\mathrm{b},i}(\dot{x}_{\mathrm{b},i} - \dot{x}_{\mathrm{t}}) \tag{6.2}$$

$$\tilde{F}_{\mathrm{tip},i} = F_{\mathrm{tip},i} - K_{\mathrm{b},i}(x_{\mathrm{b},i} - x_{\mathrm{t}}) - B_{\mathrm{b},i}(\dot{x}_{\mathrm{b},i} - \dot{x}_{\mathrm{t}}) \tag{6.3}$$

In the equations $F_{\mathrm{tip},i}$ is the tip thrust force from each blade, found in equation 5.23, $K_{\mathrm{b},i}$ and $B_{\mathrm{b},i}$ is the spring and damper constants, $x_{\mathrm{b},i}$ and x_{t} is the displacement of the blade elements and the tower respectively, also denoting the deflection of the blade and tower.

From equation 6.2 the acceleration of the blade is found by dividing the equation by $M_{\mathrm{b},i}$, yielding equation 6.4

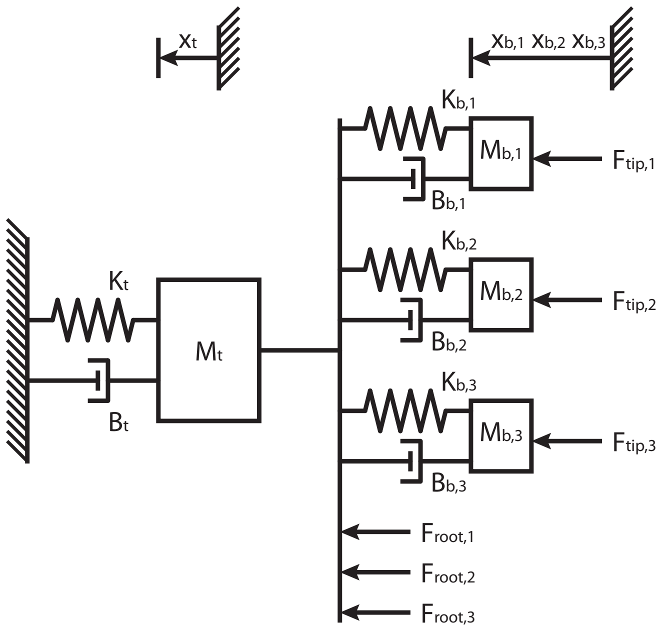


Figure 6.3: Mass-spring-damper system of the tower structure

$$\ddot{x}_{\mathrm{b},i} = \frac{F_{\mathrm{tip},i}}{M_{\mathrm{b},i}} - \frac{K_{\mathrm{b},i}}{M_{\mathrm{b},i}} x_{\mathrm{b},i} + \frac{K_{\mathrm{b},i}}{M_{\mathrm{b},i}} x_{\mathrm{t}} - \frac{B_{\mathrm{b},i}}{M_{\mathrm{b},i}} \dot{x}_{\mathrm{b},i} + \frac{B_{\mathrm{b},i}}{M_{\mathrm{b},i}} \dot{x}_{\mathrm{t}} \tag{6.4}$$

For the tower the mass-spring-damper system can be expressed as in equation 6.5.

$$\begin{aligned} F_{\mathrm{root},1} + F_{\mathrm{root},2} + F_{\mathrm{root},3} + \tilde{F}_{\mathrm{tip},1} + \tilde{F}_{\mathrm{tip},2} + \tilde{F}_{\mathrm{tip},3} \\ = M_{\mathrm{t}} \ddot{x}_{\mathrm{t}} + K_{\mathrm{t}} x_{\mathrm{t}} + B_{\mathrm{t}} \dot{x}_{\mathrm{t}} \end{aligned} \tag{6.5}$$

where $F_{\mathrm{root},i}$ is the root thrust force from each blade, found in equation 5.22, and K_{t} and B_{t} the spring and damper constants.
$\ddot{x}_{\mathrm{t}}$ will now be isolated by inserting equation 6.3 into equation 6.5 and rearranging, the result is seen in equation 6.6.

$$\begin{aligned} \ddot{x}_{\mathrm{t}} = & \frac{F_{\mathrm{root},1}}{M_{\mathrm{t}}} + \frac{F_{\mathrm{root},2}}{M_{\mathrm{t}}} + \frac{F_{\mathrm{root},3}}{M_{\mathrm{t}}} + \frac{F_{\mathrm{tip},1}}{M_{\mathrm{t}}} + \frac{F_{\mathrm{tip},2}}{M_{\mathrm{t}}} + \frac{F_{\mathrm{tip},3}}{M_{\mathrm{t}}} \\ & + \frac{K_{\mathrm{b},1} + K_{\mathrm{b},2} + K_{\mathrm{b},3} - K_{\mathrm{t}}}{M_{\mathrm{t}}} x_{\mathrm{t}} + \frac{B_{\mathrm{b},1} + B_{\mathrm{b},2} + B_{\mathrm{b},3} - B_{\mathrm{t}}}{M_{\mathrm{t}}} \dot{x}_{\mathrm{t}} \\ & - \frac{K_{\mathrm{b},1}}{M_{\mathrm{t}}} x_{\mathrm{b},1} - \frac{K_{\mathrm{b},2}}{M_{\mathrm{t}}} x_{\mathrm{b},2} - \frac{K_{\mathrm{b},3}}{M_{\mathrm{t}}} x_{\mathrm{b},3} - \frac{B_{\mathrm{b},1}}{M_{\mathrm{t}}} \dot{x}_{\mathrm{b},1} - \frac{B_{\mathrm{b},2}}{M_{\mathrm{t}}} \dot{x}_{\mathrm{b},2} - \frac{B_{\mathrm{b},3}}{M_{\mathrm{t}}} \dot{x}_{\mathrm{b},3} \end{aligned} \tag{6.6}$$

The input to this tower structure was the thrust force from the aerodynamics in Chapter 5 and the output will be the deflection of the blades and tower.

Furthermore, two state equations in 6.4 and 6.6 were found. These state equation has been rewritten into the state space representation in 6.7 and 6.8.

$$A_{\mathrm{str}} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{K_{\mathrm{b},1}}{M_{\mathrm{b},1}} & -\frac{B_{\mathrm{b},1}}{M_{\mathrm{b},1}} & 0 & 0 & 0 & 0 & \frac{K_{\mathrm{b},1}}{M_{\mathrm{b},1}} & \frac{B_{\mathrm{b},1}}{M_{\mathrm{b},1}} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{K_{\mathrm{b},2}}{M_{\mathrm{b},2}} & -\frac{B_{\mathrm{b},2}}{M_{\mathrm{b},2}} & 0 & 0 & \frac{K_{\mathrm{b},2}}{M_{\mathrm{b},2}} & \frac{B_{\mathrm{b},2}}{M_{\mathrm{b},2}} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{K_{\mathrm{b},3}}{M_{\mathrm{b},3}} & -\frac{B_{\mathrm{b},3}}{M_{\mathrm{b},3}} & \frac{K_{\mathrm{b},3}}{M_{\mathrm{b},3}} & \frac{B_{\mathrm{b},3}}{M_{\mathrm{b},3}} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -\frac{K_{\mathrm{b},1}}{M_{\mathrm{t}}} & -\frac{B_{\mathrm{b},1}}{M_{\mathrm{t}}} & -\frac{K_{\mathrm{b},2}}{M_{\mathrm{t}}} & -\frac{B_{\mathrm{b},2}}{M_{\mathrm{t}}} & -\frac{K_{\mathrm{b},3}}{M_{\mathrm{t}}} & -\frac{B_{\mathrm{b},3}}{M_{\mathrm{t}}} & a_{8,7} & a_{8,8} \end{bmatrix} \tag{6.7}$$

$$a_{8,7} = \frac{K_{\mathrm{b},1}+K_{\mathrm{b},2}+K_{\mathrm{b},3}-K_{\mathrm{t}}}{M_{\mathrm{t}}}$$

$$a_{8,8} = \frac{B_{\mathrm{b},1}+B_{\mathrm{b},2}+B_{\mathrm{b},3}-B_{\mathrm{t}}}{M_{\mathrm{t}}}$$

$$B_{\mathrm{str}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{M_{\mathrm{b},1}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{M_{\mathrm{b},2}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{M_{\mathrm{b},3}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} \end{bmatrix} \quad C_{\mathrm{str}} = I_8 \quad D_{\mathrm{str}} = 0_{8,6} \tag{6.8}$$

The state vector and input vector to the structural state space model are 6.9 and 6.10.

$$x_{\mathrm{str}} = \begin{bmatrix} x_{\mathrm{b},1} & \dot{x}_{\mathrm{b},1} & x_{\mathrm{b},2} & \dot{x}_{\mathrm{b},2} & x_{\mathrm{b},3} & \dot{x}_{\mathrm{b},3} & x_{\mathrm{t}} & \dot{x}_{\mathrm{t}} \end{bmatrix}^{\mathrm{T}} \tag{6.9}$$

$$u_{\mathrm{str}} = \begin{bmatrix} F_{\mathrm{root},1} & F_{\mathrm{tip},1} & F_{\mathrm{root},2} & F_{\mathrm{tip},2} & F_{\mathrm{root},3} & F_{\mathrm{tip},3} \end{bmatrix}^{\mathrm{T}} \tag{6.10}$$

6.1 Yaw torque

When a wake hits the rotor plane this will cause an uneven load on the wind turbine. The difference in the loads will induce an yaw torque on the wind turbine, as illustrated in Figure 6.4.

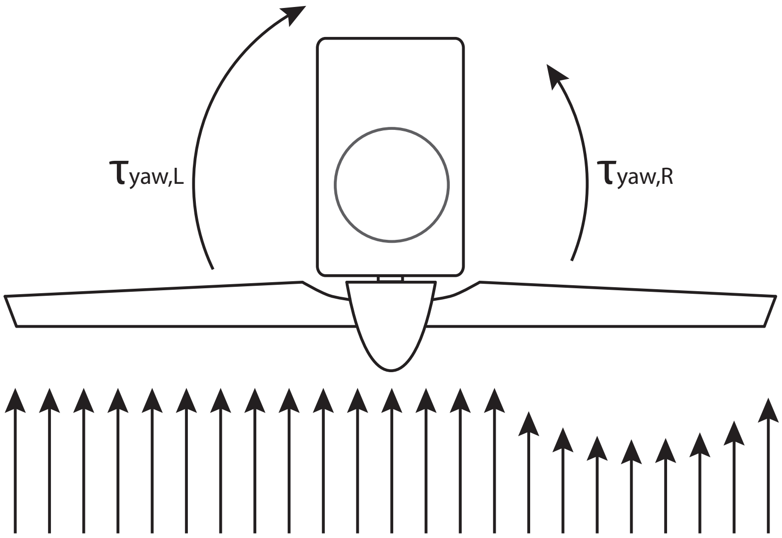


Figure 6.4: Principal sketch of an uneven wind load inducing an yaw torque

The total yaw torque can be defined as in equation 6.11.

$$\tau_{\text{yaw}} = \tau_{\text{yaw,L}} + \tau_{\text{yaw,R}} \tag{6.11}$$

In order to identify if there is an uneven load on the rotor plane, a model for the yaw torque is derived. This model can later be used to limit the yaw torque affecting the wind turbine.

As the wind pushes upon a blade, the blade generates an yaw torque on the tower, as in equation 6.12. $\tau_{\text{yaw},k}$ is the torque around the tower from the k^{th} blade. Each blade is divided into N pieces, for each piece the force from the wind $F_{\text{T},i}$, and the distance to center r_i is multiplied.

$$\tau_{\text{yaw},k} = \sum_{i=1}^{N} F_{\text{T},i} \cdot r_i \tag{6.12}$$

For the root of blade, which does not bend, the force upon the blade is $\tau_{\text{yaw,root}}$, which is shown in equation 6.13.

$$\tau_{\text{yaw,root}} = \Bigg(\sum_{i=r_0}^{r_{\text{W}}} \frac{\rho}{2} V_{\text{rel}}^2(r_i) \cdot c_{\text{bl1}}(r_i) dr \cdot r_i + \sum_{i=r_{\text{W}}+1}^{r_{\text{H}}} \frac{\rho}{2} V_{\text{rel}}^2(r_i) \cdot c_{\text{bl2}}(r_i) dr \cdot r_i \Bigg) \cdot (C_{\text{L}}(\alpha)\cos(\phi) + C_{\text{D}}(\alpha)\sin(\phi)) \quad (6.13)$$

For the tip of the blade both the force and distance to the tower are changed, this is illustrated in Figure 6.5.

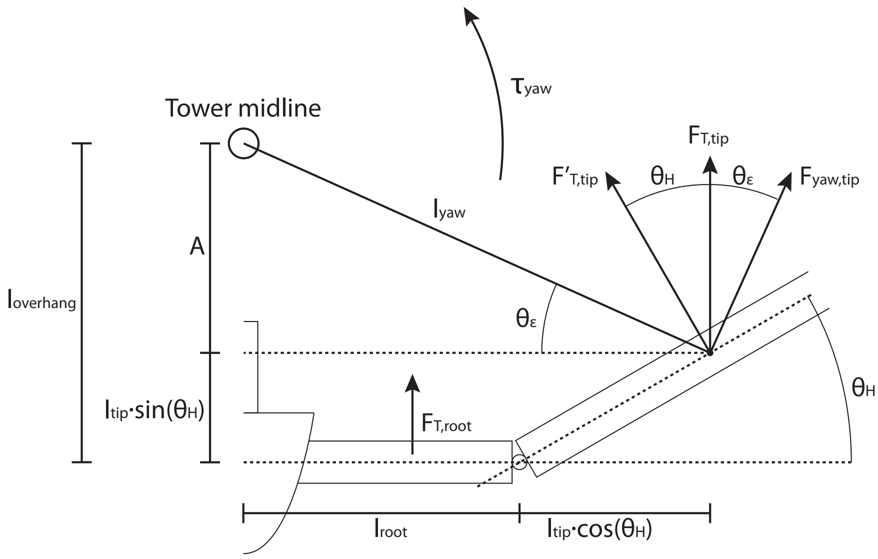


Figure 6.5: How to find the yaw torque

Equations 6.14 to 6.17 shows the calculation of $F_{\text{yaw,tip}}$.

$$A = l_{\text{overhang}} - l_{\text{tip}}(r_i)\sin(\theta_H) \quad (6.14)$$

$$l_{\text{yaw}}(r_i) = \sqrt{A^2 + (l_{\text{root}} + l_{\text{tip}}(r_i)\cos(\theta_H))^2} \quad (6.15)$$

$$\theta_\epsilon = \tan^{-1}\left(\frac{A}{l_{\text{root}} + l_{\text{tip}}(r_i)\cos(\theta_H)}\right) \quad (6.16)$$

$$F_{\text{yaw,tip}} = F'_{\text{T,tip}} \cdot \cos(\theta_H + \theta_\epsilon) \quad (6.17)$$

When $F_{\text{yaw,tip}}$ is found, $\tau_{\text{yaw,tip}}$ can be found as well by including the distance from blade segment to the tower midline l_{yaw} in the equation, this is done in equation 6.18.

$$\tau_{\text{yaw,tip}} = \left(\sum_{i=r_{\text{H}}+1}^{R} \frac{\rho}{2} V_{\text{rel}}'^2(r_i) \cdot c_{\text{bl2}}(r_i) dr \cdot l_{\text{yaw}}(r_i) \right) \cdot \big(C_{\text{L}}(\alpha')\cos(\phi') + C_{\text{D}}(\alpha')\sin(\phi')\big) \cdot \cos(\theta_H + \theta_\epsilon) \tag{6.18}$$

The total yaw torque for each blade is then calculated in equation 6.19.

$$\tau_{\text{yaw,blade,}k} = \tau_{\text{yaw,root}} + \tau_{\text{yaw,tip}} \tag{6.19}$$

As the yaw torque of the interest is around the vertical axis, the azimuth angle of the blade has to be taken into account. The angles are defined in Figure 6.6.

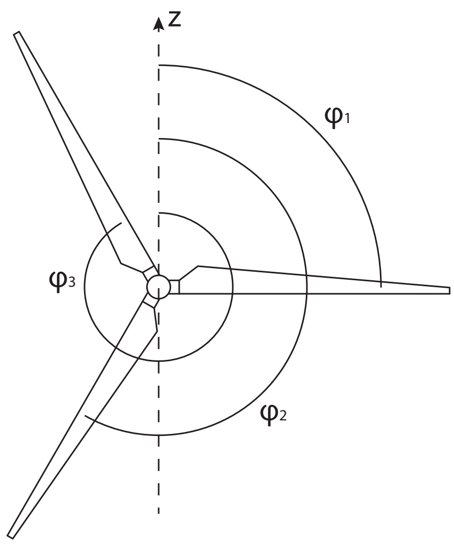


Figure 6.6: The angle used to find the yaw torque

The yaw torque for one blade around the vertical axis is in equation 6.20, where φ_i is the angle of the blade, k is the blade number, and z indicates that it is around the z-axis.

$$\tau_{\text{yaw,z,}k} = \sin(\varphi_k) \cdot \tau_{\text{yaw,blade,}k} \tag{6.20}$$

Applying a homogeneous wind field the single blade generates an yaw torque from each blade, this is illustrated in Figure 6.7, where the summarized yaw torque is shown as well.

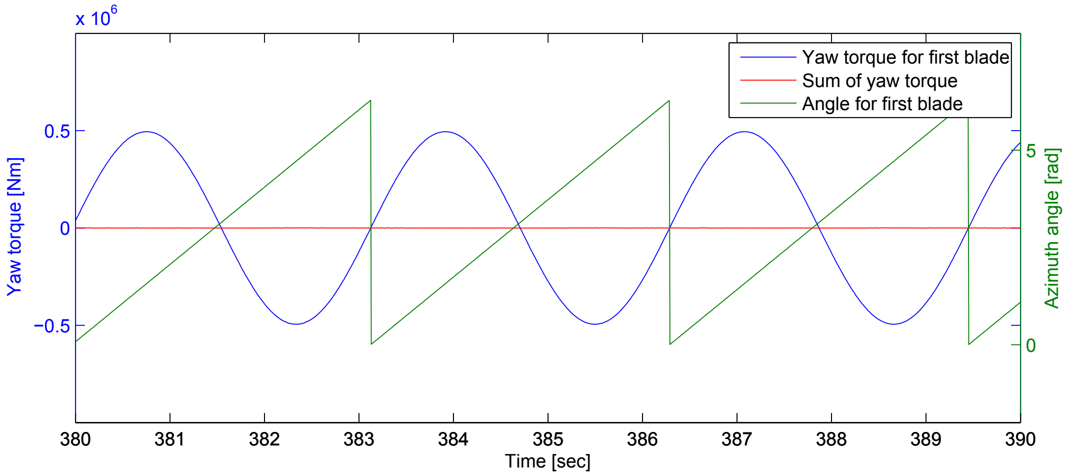


Figure 6.7: Simple simulation with a homogeneous wind field

If a wake is present in the wind field an uneven yaw torque occurs, this is illustrated in Figure 6.8.

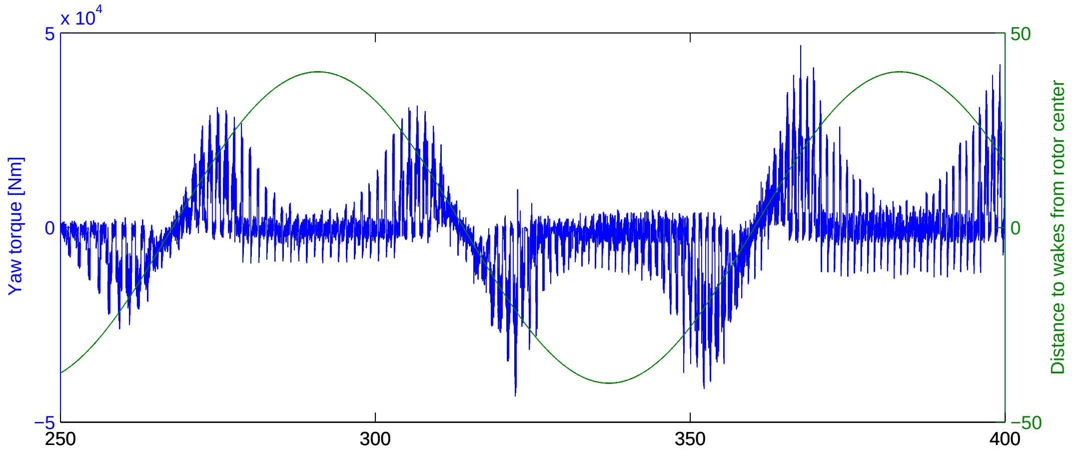


Figure 6.8: Yaw simulation with a wake moving in a sine pattern

As the wake moves in the wind field the yaw torque increases, and when it move out of the rotor plane the yaw torque decreases again.

Now it will be examined how the rotational force from Chapter 5, drives the generator, this is done in the following chapter.

Chapter 7

Mechanical model

In this chapter, the main mechanics of the wind turbine is considered, namely the drive train. The objective of the drive train is to connect the rotor to the generator, and thereby convert the rotational torque to electrical power. The inputs and outputs of the mechanical model can be seen in Figure 7.1.

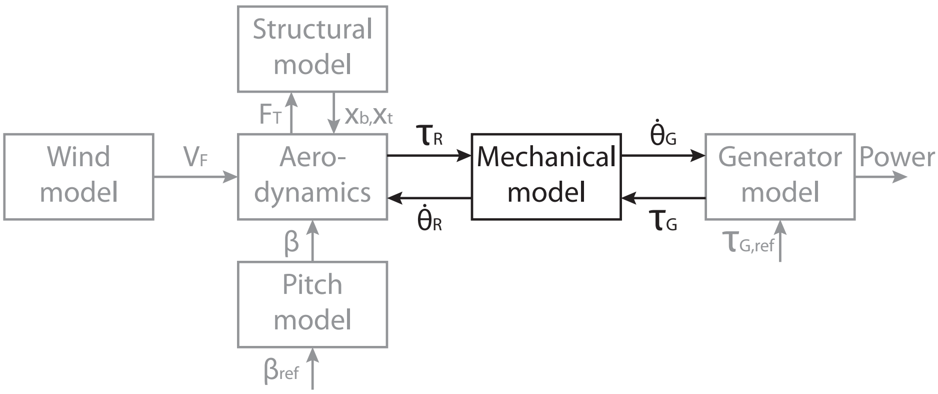


Figure 7.1: Block diagram showing the mechanical model inputs and outputs in the overall model structure

The torque from the rotor τ_R, found in Chapter 5, is transfered to the generator through a low speed shaft, which typically has a speed of 20-50 rpm [2, p. 30], into a gearbox giving the high speed shaft the speed 1000-1500 rpm, and lastly into the generator, this is illustrated in Figure 7.2.

Gearless wind turbines with direct connection between rotor and generator is making an entrance into the market, but this will not be considered in this model.

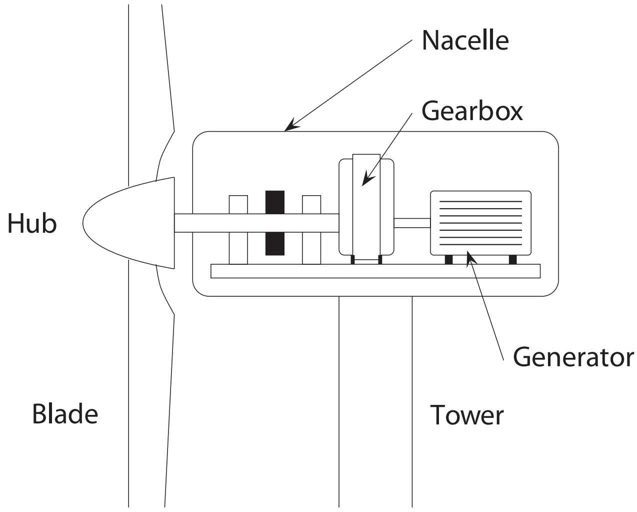


Figure 7.2: Mechanical system of the wind turbine [2, p. 30]

The different parts of the mechanical system will now be examined and a model is made inspired by [10]. The mechanical system from Figure 7.2 can be turned into the model illustration in Figure 7.3.

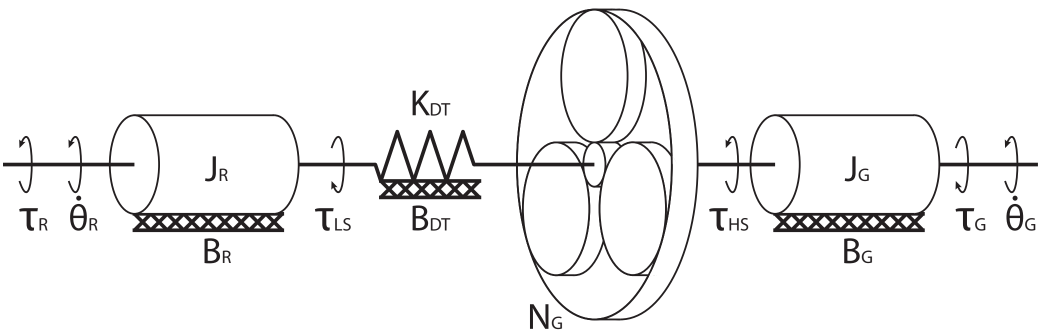


Figure 7.3: Mechanical model

In left side of the figure, the torque from the rotor is applied as τ_{R}, this makes the low speed shaft rotate with a speed of $\dot{\theta}_{\mathrm{R}}$. The low speed shaft drives a gear, here illustrated as a planetary gear, this transfers a speed of $\dot{\theta}_{\mathrm{G}}$, and the torque τ_{HS} to the high speed shaft which is fed into the generator. The equations of the transfer speed and torque is shown in equation 7.1 and 7.2 where N_{G} is the gear ratio.

$$\dot{\theta}_{\mathrm{G}} = N_{\mathrm{G}} \cdot \dot{\theta}_{\mathrm{R}} \tag{7.1}$$

$$\tau_{\mathrm{HS}} = \frac{\tau_{\mathrm{LS}}}{N_{\mathrm{G}}} \tag{7.2}$$

J_R in the model, is the inertia of both the low speed shaft and the rotor, and J_G includes the inertia of the generator and gearbox. Furthermore, the torsion of the shaft and the flexibility of the drive train are modeled by an inserted spring. Frictions from the gear and the bearings are included on the spring and the low and high speed shafts.
The torques, that is induced by the inertias and frictions of the drive train, are included in the model as τ_LS and the torque from the generator is included as τ_G.

7.1 Dynamic equations

The dynamics of the shafts can now be expressed, in equation 7.3 is the low speed shaft firstly described.

$$J_\mathrm{R}\ddot{\theta}_\mathrm{R} = \tau_\mathrm{R} - \tau_\mathrm{LS} - B_\mathrm{R}\dot{\theta}_\mathrm{R} \tag{7.3}$$

Beside the known elements this equation contains the friction of the bearing denoted as B_R.
The torsion on the drive train is included as a torsion spring, which is described in equation 7.4.

$$\tau_\mathrm{LS} = K_\mathrm{DT}\theta_\delta + B_\mathrm{DT}\dot{\theta}_\delta \tag{7.4}$$

$$\theta_\delta = \theta_\mathrm{R} - \frac{\theta_\mathrm{G}}{N_\mathrm{G}} \tag{7.5}$$

Where B_DT is a torsion damping coefficient, K_DT the stiffness, and θ_δ the torsion angle of the drive train.

The absolute angles of the system are of no interest for the model, except the torsion angle θ_δ. This leads to the rewriting in equation 7.6 where 7.5 is differentiated and inserted into 7.4.

$$\tau_\mathrm{LS} = K_\mathrm{DT}\theta_\delta + B_\mathrm{DT}\left(\dot{\theta}_\mathrm{R} - \frac{\dot{\theta}_\mathrm{G}}{N_\mathrm{G}}\right) \tag{7.6}$$

Now the dynamics of the low speed shaft is expressed, and is found for the high speed shaft in equation 7.7.

$$J_\mathrm{G}\ddot{\theta}_\mathrm{G} = \tau_\mathrm{HS} - \tau_\mathrm{G} - B_\mathrm{G}\dot{\theta}_\mathrm{G} \tag{7.7}$$

The dynamic equations have been found for the drive train, and they can now be rewriten into three equations.
Firstly 7.6 is substituted into 7.3 giving equation 7.8. Then 7.9 is found in a similar way, by first inserting 7.6 into 7.2 and these are inserted into 7.7. The last equation 7.10 is made by differentiating 7.5.

$$J_{\mathrm{R}}\ddot{\theta}_{\mathrm{R}} = \tau_{\mathrm{R}} - K_{\mathrm{DT}}\theta_\delta - (B_{\mathrm{DT}} + B_{\mathrm{R}})\,\dot{\theta}_{\mathrm{R}} + \frac{B_{\mathrm{DT}}}{N_{\mathrm{G}}}\dot{\theta}_{\mathrm{G}} \tag{7.8}$$

$$J_{\mathrm{G}}\ddot{\theta}_{\mathrm{G}} = \frac{K_{\mathrm{DT}}}{N_{\mathrm{G}}}\theta_\delta + \frac{B_{DT}}{N_{\mathrm{G}}}\dot{\theta}_{\mathrm{R}} - \left(\frac{B_{\mathrm{DT}}}{N_{\mathrm{G}}^2} + B_{\mathrm{G}}\right)\dot{\theta}_{\mathrm{G}} - \tau_{\mathrm{G}} \tag{7.9}$$

$$\dot{\theta}_\delta = \dot{\theta}_{\mathrm{R}} - \frac{\dot{\theta}_{\mathrm{G}}}{N_{\mathrm{G}}} \tag{7.10}$$

In the final state space model, desired states is $\ddot{\theta}_{\mathrm{R}}$, $\ddot{\theta}_{\mathrm{G}}$ and $\dot{\theta}_\delta$, these variables is isolated in equation 7.11 to 7.13.

$$\ddot{\theta}_{\mathrm{R}} = \frac{\tau_{\mathrm{R}}}{J_{\mathrm{R}}} - \frac{K_{\mathrm{DT}}}{J_{\mathrm{R}}}\theta_\delta - \frac{(B_{\mathrm{DT}} + B_{\mathrm{R}})}{J_{\mathrm{R}}}\dot{\theta}_{\mathrm{R}} + \frac{B_{\mathrm{DT}}}{N_{\mathrm{G}}J_{\mathrm{R}}}\dot{\theta}_{\mathrm{G}} \tag{7.11}$$

$$\ddot{\theta}_{\mathrm{G}} = \frac{K_{\mathrm{DT}}}{N_{\mathrm{G}}J_{\mathrm{G}}}\theta_\delta + \frac{B_{DT}}{N_{\mathrm{G}}J_{\mathrm{G}}}\dot{\theta}_{\mathrm{R}} - \left(\frac{B_{\mathrm{DT}}}{N_{\mathrm{G}}^2 J_{\mathrm{G}}} + \frac{B_{\mathrm{G}}}{J_{\mathrm{G}}}\right)\dot{\theta}_{\mathrm{G}} - \frac{\tau_{\mathrm{G}}}{J_{\mathrm{G}}} \tag{7.12}$$

$$\dot{\theta}_\delta = \dot{\theta}_{\mathrm{R}} - \frac{1}{N_{\mathrm{G}}}\dot{\theta}_{\mathrm{G}} \tag{7.13}$$

Three first order differential equations have now been derived, in order to describe the behavior of the drive train.

The equations found have been written into a state space representation in equation 7.14 to 7.16

$$x_{\mathrm{mech}} = \begin{bmatrix}\dot{\theta}_{\mathrm{R}} & \dot{\theta}_{\mathrm{G}} & \theta_\delta\end{bmatrix}^{\mathsf{T}} \quad u_{\mathrm{mech}} = \begin{bmatrix}\tau_R & \tau_G\end{bmatrix}^{\mathsf{T}} \tag{7.14}$$

$$A_{\mathrm{mech}} = \begin{bmatrix} \frac{-(B_{\mathrm{DT}}+B_{\mathrm{R}})}{J_{\mathrm{R}}} & \frac{B_{\mathrm{DT}}}{N_{\mathrm{G}}J_{\mathrm{R}}} & \frac{-K_{\mathrm{DT}}}{J_{\mathrm{R}}} \\ \frac{B_{\mathrm{DT}}}{N_{\mathrm{G}}J_{\mathrm{G}}} & -\left(\frac{B_{\mathrm{DT}}}{N_{\mathrm{G}}^2 J_{\mathrm{G}}} + \frac{B_{\mathrm{G}}}{J_{\mathrm{G}}}\right) & \frac{K_{\mathrm{DT}}}{N_{\mathrm{G}}J_{\mathrm{G}}} \\ 1 & \frac{-1}{N_{\mathrm{G}}} & 0 \end{bmatrix} \tag{7.15}$$

$$B_{\mathrm{mech}} = \begin{bmatrix} \frac{1}{J_{\mathrm{R}}} & 0 \\ 0 & \frac{-1}{J_{\mathrm{G}}} \\ 0 & 0 \end{bmatrix} \quad C_{\mathrm{mech}} = I_3 \quad D_{\mathrm{mech}} = 0_{3,2} \tag{7.16}$$

Chapter 8

Pitch system model

A part of the control will consist of blade pitching. This pitching will be performed by three pitch actuators placed in the hub of the wind turbine. It is necessary to model these actuators in order to include the pitching control in the overall model.
The input to the pitch system model β_{ref} will come from the controller, as illustrated in Figure 8.1, this input consists of three different pitch references, to make individual pitching possible, and the output will be the actual pitch of the blade used by the aerodynamic model in Chapter 5.

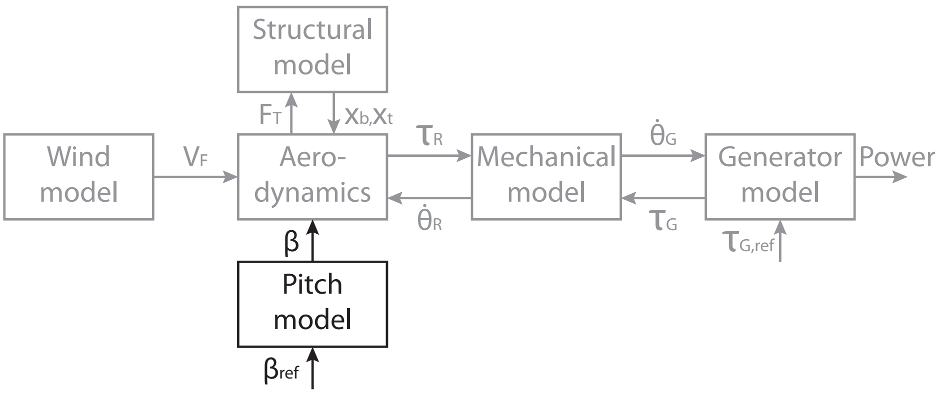


Figure 8.1: Block diagram showing the pitch system model inputs and outputs in the overall model structure

In closed loop, the pitch actuators can be modeled as a first-order system [2, p. 42], where there is a limitations in speed of the actuators and amplitude. This has been considered to be sufficient to this project where the wind turbine will have three independent and identical pitch systems.

The input for the pitch actuator will as described be the pitch reference β_{ref} and the output will be the actual pitch of the blade β. The dynamic behavior of the pitch system is expressed in equation 8.1.

$$L\left(\dot{\beta}\right) = -\frac{1}{\tau_{\mathrm{time,P}}} L\left(\beta\right) + \frac{1}{\tau_{\mathrm{time,P}}} \beta_{\mathrm{ref}} \tag{8.1}$$

where

$$L(x) = \begin{cases} x & \text{if} \quad x_{\text{min}} < x < x_{\text{max}} \\ x_{\text{min}} & \text{if} \quad x < x_{\text{min}} \\ x_{\text{max}} & \text{if} \quad x > x_{\text{max}} \end{cases} \tag{8.2}$$

where $\tau_{\text{time,P}}$ is the time constant of the pitch system. In Figure 8.2, a block diagram of the first-order pitch actuator model is shown, where the limitations of the actuator is inserted as well.

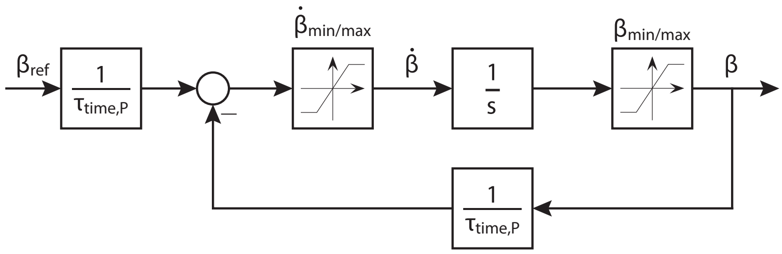


Figure 8.2: Pitch actuator block diagram [2, p. 43]

As the properties for pitch actuator is not included in FAST, the limitations shown in Table 8.1 will be used.

β_{min}	0°	$\dot{\beta}_{\text{min}}$	-10°/s
β_{max}	90°	$\dot{\beta}_{\text{max}}$	10°/s

Table 8.1: Pitch actuator limitations [18, p. 25][19, p. 6]

From [22, p. 5] $\tau_{\text{time,P}}$ is set to 0,2 seconds.

The model is written in state space as shown in equations 8.3 to 8.5

$$x_\beta = \begin{bmatrix} \beta_1 & \beta_2 & \beta_3 \end{bmatrix}^\mathsf{T} u_\beta = \begin{bmatrix} \beta_{\text{ref,1}} & \beta_{\text{ref,2}} & \beta_{\text{ref,3}} \end{bmatrix}^\mathsf{T} \tag{8.3}$$

$$A_\beta = \begin{bmatrix} \frac{-1}{\tau_{\text{time,P}}} & 0 & 0 \\ 0 & \frac{-1}{\tau_{\text{time,P}}} & 0 \\ 0 & 0 & \frac{-1}{\tau_{\text{time,P}}} \end{bmatrix} B_\beta = \begin{bmatrix} \frac{1}{\tau_{\text{time,P}}} & 0 & 0 \\ 0 & \frac{1}{\tau_{\text{time,P}}} & 0 \\ 0 & 0 & \frac{1}{\tau_{\text{time,P}}} \end{bmatrix} \tag{8.4}$$

$$C_\beta = I_3 \quad D_\beta = 0_{3,3} \tag{8.5}$$

The model of the pitch actuators have now been modeled as a first order system. The output β of this pitch system model will be an input to the aerodynamic model described in Chapter 5.

Chapter 9

Generator model

Besides the blade pitch, the controller can use the generator torque to control the system. It is therefore necessary to make a model of the loading torque from the generator which shows the relation between the input and output of the generator. The placement of the generator model in the overall structure can be seen in Figure 9.1.

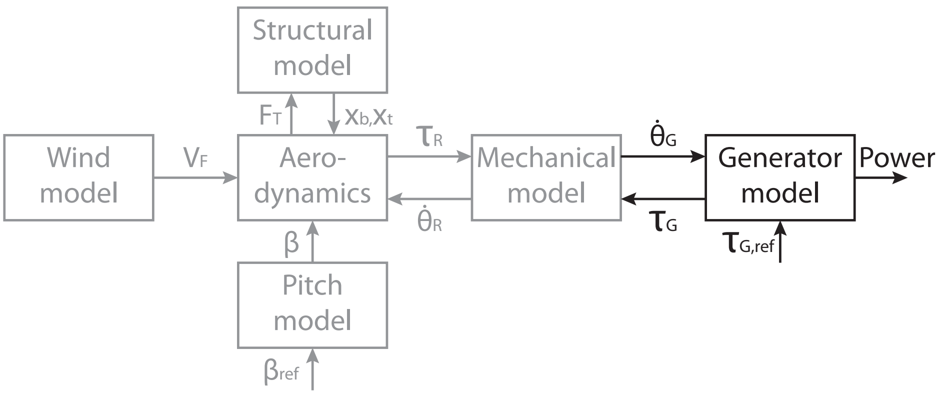


Figure 9.1: Block diagram showing the generator model inputs and outputs in the overall model structure

The input to the model of the generator is the angular velocity of the high speed shaft from Chapter 7 where the mechanical model is described, and a control signal consisting of a torque reference $\tau_{G,ref}$ and the output will be the actual torque from the generator τ_G, and an electrical power output.
The dynamics of the electric part of the wind turbine are much faster than the mechanical parts of the system [2, p. 37], and therefore the generator model can be modeled as a first order system [8].

The dynamic equation describing the generator is expressed in equation 9.1.

$$L(\dot{\tau}_G) = -\frac{1}{\tau_{time,G}} L(\tau_G) + \frac{1}{\tau_{time,G}} \tau_{G,ref} \tag{9.1}$$

where

$$L(x) = \begin{cases} x & \text{if} \quad x_{\text{min}} < x < x_{\text{max}} \\ x_{\text{min}} & \text{if} \quad x < x_{\text{min}} \\ x_{\text{max}} & \text{if} \quad x > x_{\text{max}} \end{cases} \tag{9.2}$$

where $\tau_{\text{time,G}}$ is the time constant of the first-order system. A block diagram of the generator model is illustrated in Figure 9.2, where the limitations of the generator is inserted.

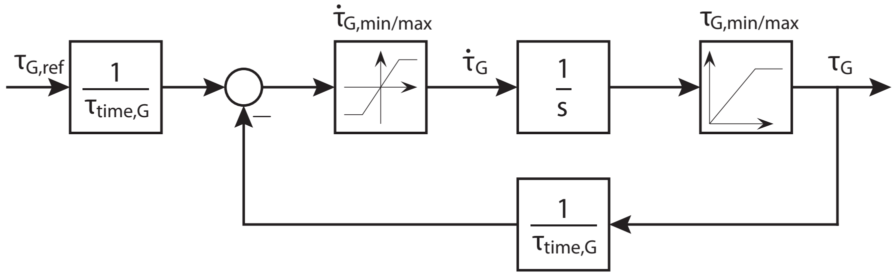


Figure 9.2: Block diagram of generator model

In the final model, the generator torque is said to be equal to the torque reference because the generator unit is assumed to be significantly faster than the rest of the system.

Besides the response on a reference torque, the incoming angular velocity from the high speed shaft, $\dot{\theta}_{\text{G}}$, will combined with the torque give a power output, P_{G}, as shown in equation 9.3.

$$P_{\text{G}} = \eta_{\text{G}} \dot{\theta}_{\text{G}} \tau_{\text{G}} \tag{9.3}$$

where η_{G} is the efficiency of the generator.

In this chapter, a model of the generator has been modeled as a first order system. The outputs of this generator model is the generator torque τ_{G} which is used in the mechanical model in Chapter 7, and a power output P_{G} which will also be the output from the overall model.

Chapter 10

Parameter estimation

Different parts of the model has been derived in previous chapters. To complete the submodels, some of the parameters will now be determined, these are parameters that can be estimated by implementing the submodels in MATLAB, and comparing the results with the FAST outputs.

Ahead of the parameter estimation the placement of the hinge has been determined by inspiration from [10], and use of MATLAB simulations. The hinge is placed between the 8^{th} and the 9^{th} blade element, this gives a r_{H} of 19,5 m.

MATLAB scripts and FAST files used for the parameter estimation is attached in the folder `['Parameter estimation']`.

10.1 Aerodynamic parameter estimation

In the following, the aerodynamic model is compared to the output from AeroDyn. The drive train, tower, and blade freedom has been disabled to avoid undesired influence.

The aerodynamic model for the blades from Chapter 5 has no parameters to fit. Instead four factors has been added for this purpose, $a_{\mathrm{F,root}}$ and $a_{\mathrm{F,tip}}$, which is multiplied to the thrust forces F_{root} and F_{tip} respectively, and $a_{\tau,\mathrm{root}}$ and $a_{\tau,\mathrm{tip}}$, which is multiplied to the rotational torques τ_{root} and τ_{tip} respectively.

Estimation of the parameters will be carried out as shown in Figure 10.1, by having a wind field as input to the aerodynamic model and AeroDyn, the parameters can now be fitted in the model, to make the output from AeroDyn and the model to match.

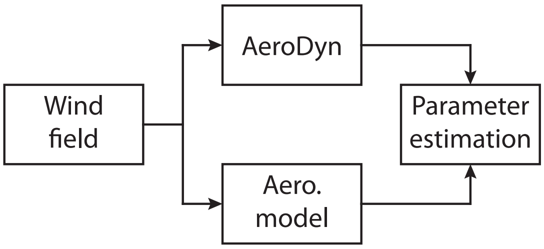


Figure 10.1: Block diagram of aerodynamic model parameter estimation

In Figure 10.2, the wind input for the aerodynamic model and FAST is shown.

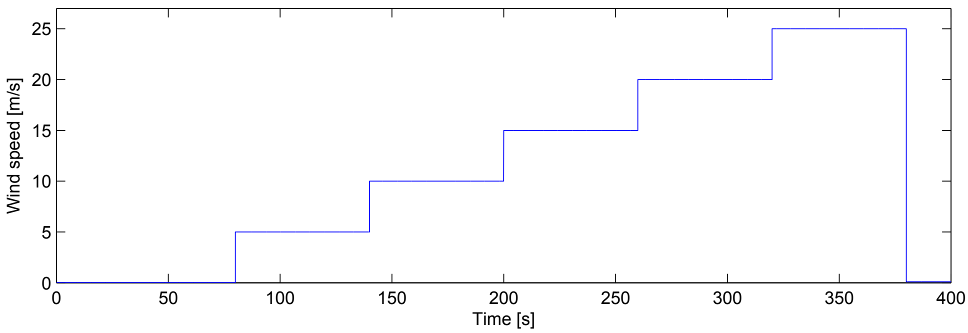
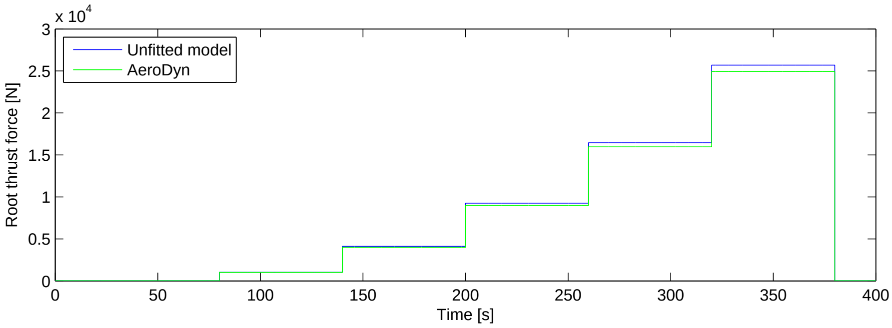
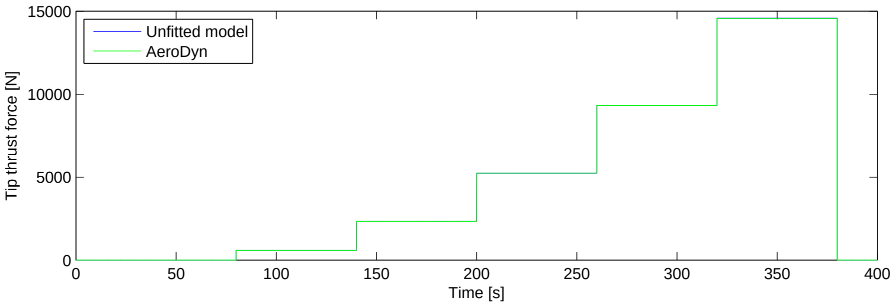


Figure 10.2: Wind input for the aerodynamic model and FAST

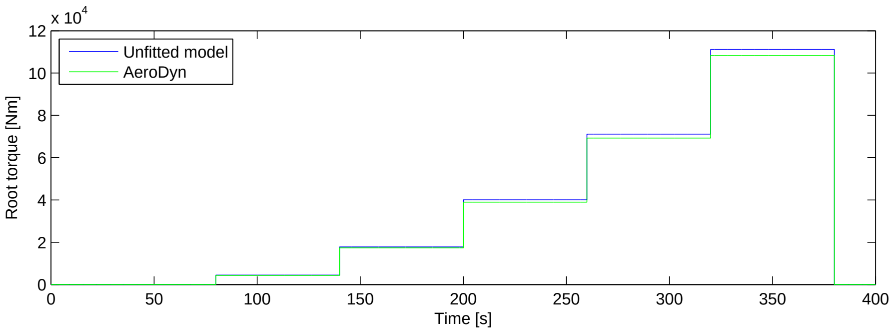
The output from AeroDyn can be seen in Figure 10.3 along with the output from the aerodynamic model. The fitted model it is the same as the AeroDyn output as the factors make a perfect fit.



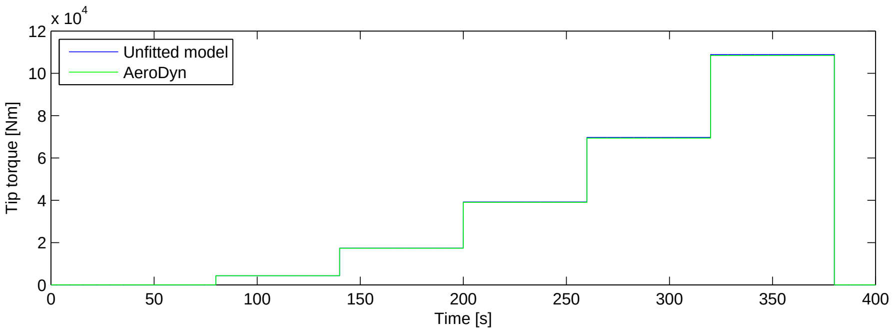
(a) Root thrust force



(b) Tip thrust force



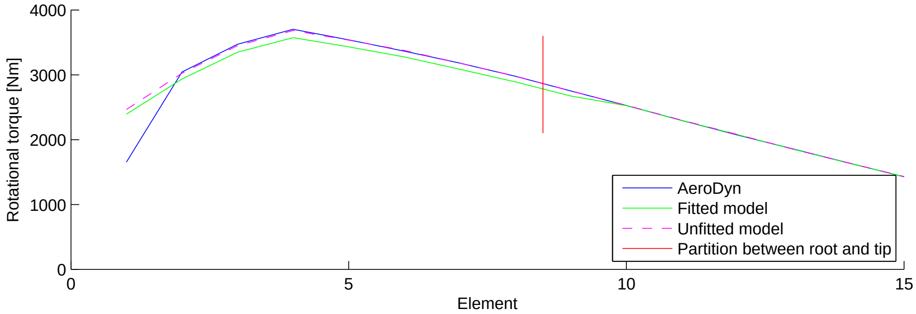
(c) Root torque



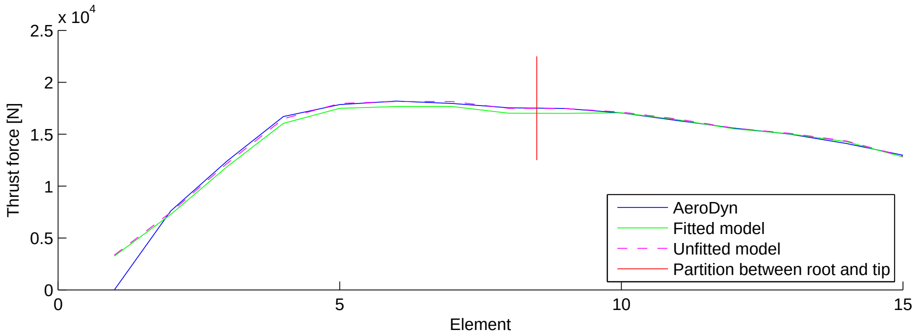
(d) Tip torque

Figure 10.3: Thrust force and rotational torque from AeroDyn, along with the unfitted torque from the aerodynamic model

These plots show how the total force and torque of a blade are behaving according to the wind input. In Figure 10.4, the forces and torques output from AeroDyn for each blade element are shown. Furthermore, the unfitted and fitted elements from the aerodynamic model are plotted as well.



(a) Rotational torque



(b) Thrust force

Figure 10.4: Element-wise thrust forces and rotational torques from AeroDyn, along with the unfitted and fitted forces and torques from the aerodynamic model

As it can be seen from the figures, there is not a perfect fit between AeroDyn and original model output. By multiplying the different parts by a fitting factor it is possible to get a perfect fit between the two outputs. These fitting factors is shown in Table 10.1.

Parameter	**Value**
$a_{\mathrm{F,root}}$	0,9704
$a_{\mathrm{F,tip}}$	0,9989
$a_{\tau,\mathrm{root}}$	0,9735
$a_{\tau,\mathrm{tip}}$	0,9959

Table 10.1: Fitting factors for the aerodynamic model

The factors has been estimated using `fminsearch` to minimize the quadratic error between AeroDyn and the model, the error is therefore as small as possible.

10.2 Yaw torque

It is not possible to fit the yaw torque to FAST, as it is not possible to use wind field including a wake as input to FAST. The equations used to calculate the yaw torque, is almost similar to the equations in the aerodynamic model and therefore the same parameters, $a_{\mathrm{F,root}}$ and $a_{\mathrm{F,tip}}$, is used on the yaw torque.

10.3 Mechanical parameter estimation

The set up for the parameter estimation of the mechanical model found in Chapter 7 can be seen in Figure 10.5.

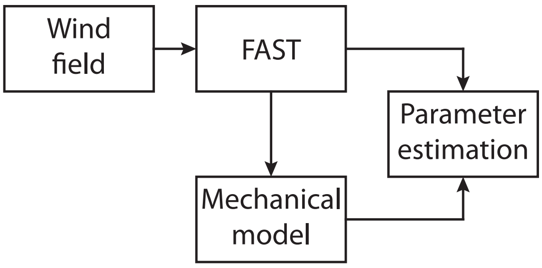


Figure 10.5: Block diagram of mechanical model parameter estimation

The input to the drive train model are the rotor and generator torque. The output are the velocity of the low and high speed shafts. The two torques are extracted from FAST and loaded into the drive train model. The output from drive train model is compared to the output from FAST.

To estimate the mechanical parameters, the MATLAB function `pem` has been used.

Previous to the estimation is some data made in FAST with the wind input shown in Figure 10.6.

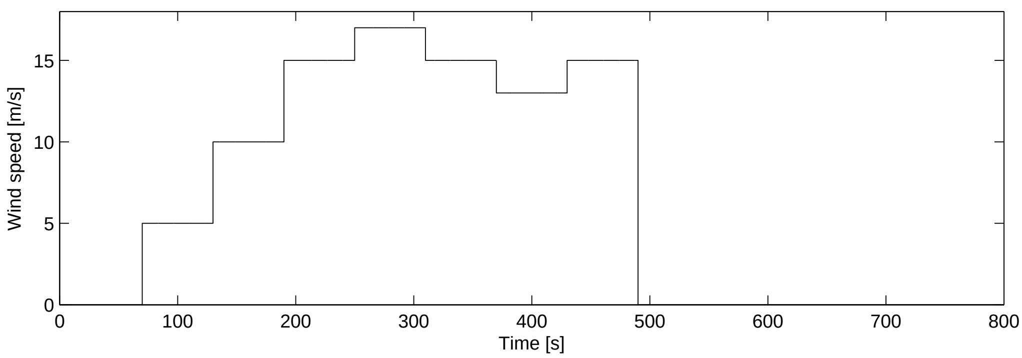


Figure 10.6: Wind input for FAST

This wind input generates the torques shown in Figure 10.7.

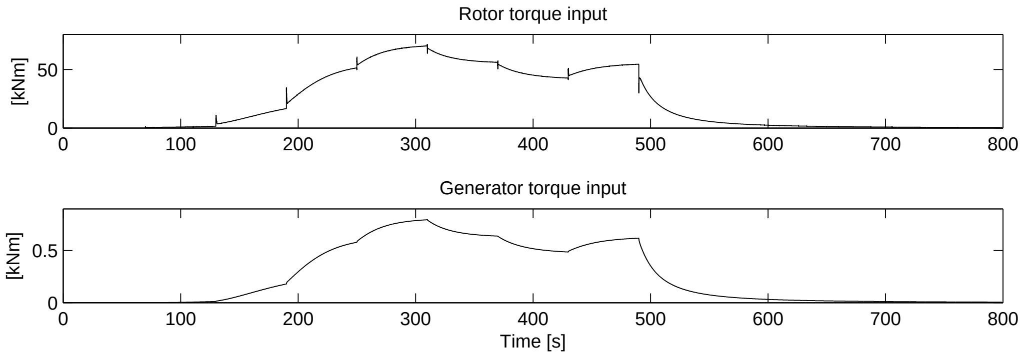


Figure 10.7: Torque input for `pem` estimation

These two torques will be the input for the `pem` estimation, along with the velocities of the low and high speed shafts.

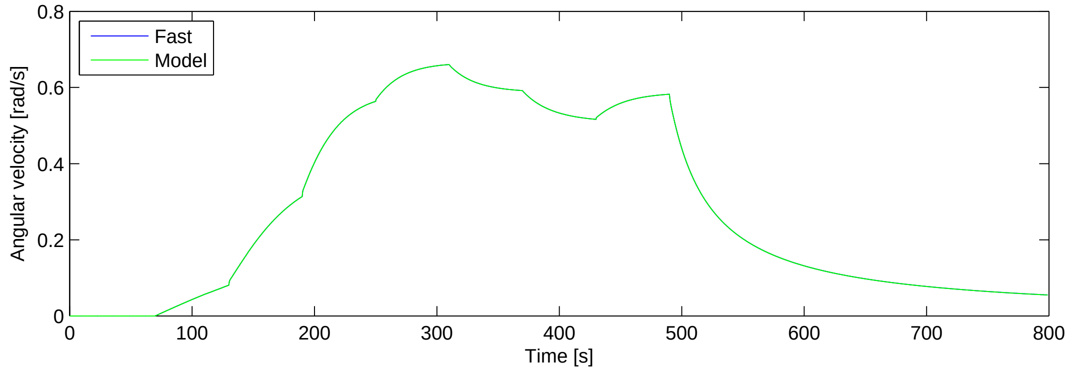
`pem` needs some initial parameter guesses and ends out with the best fitting parameter. Both the initial guesses and the final estimated parameter is listed in Table 10.2. The initial guesses is based on the project [10] and FAST.

Parameter		**Initial guess**	**Estimation**
K_{DT}	[Nm/rad]	5,60E9	5,60E9
B_{DT}	[Nm/(rad/s)]	1,00E7	1,00E7
B_{R}	[Nm/(rad/s)]	2,50E1	7,72E-5
B_{G}	[Nm/(rad/s)]	0,00E0	2,79E-5
J_{G}	[kg m^2]	5,30E1	4,51E1
J_{R}	[kg m^2]	2,96E4	6,13E4

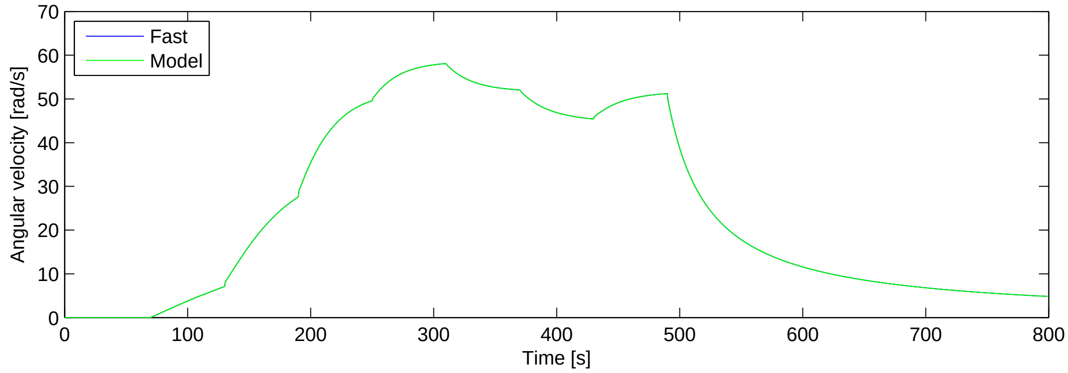
Table 10.2: Estimated drive train parameters

Other than these parameters is the gear ratio, which is not estimated but read off from the FAST files. The gear ratio is found to be 87,965.

The output from the parameter estimation is shown in the figures 10.8a and 10.8b.



(a) Low speed shaft velocity



(b) High speed shaft velocity

Figure 10.8: Result from parameter estimation of the drive train

As it can be seen from these two figures, the parameters are estimated perfectly, giving a perfect fit of both the low and high speed velocity.

Besides the estimation according to the velocities of the two shafts was the twist of the low speed shaft also originally included in the estimation. It turned out though that the output from FAST, was not suitable for estimation. The twist found in the model is shown in Figure 10.9, where only the model output is shown.

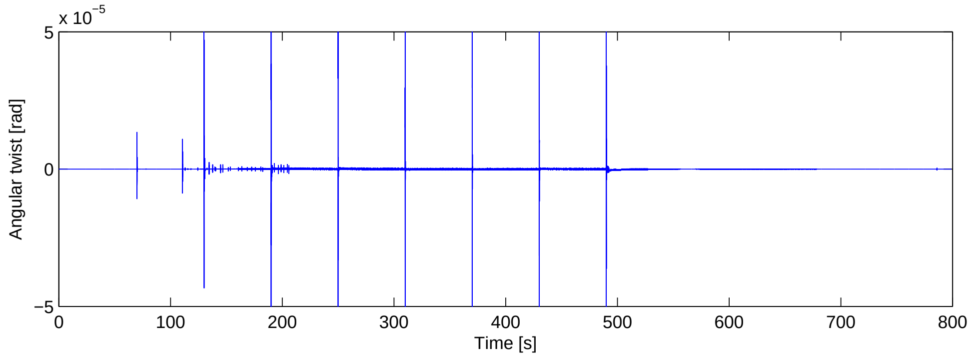


Figure 10.9: Low speed shaft twist from the model

Even though the twist of the low speed shaft have been omitted the parameter estimation of the drive train is still supposed to be suitable.

10.4 Structural parameter estimation

In Chapter 6, a model for the structural elements on the wind turbine were found, that is the bending of tower and blades.
In this section, the parameters for this model will be estimated. The set up for the estimation is shown in Figure 10.10.

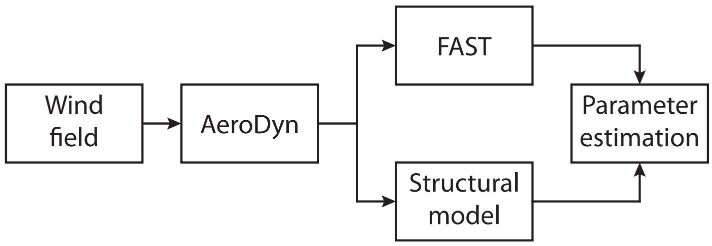


Figure 10.10: Block diagram of structural model parameter estimation

AeroDyn calculates the forces acting on each element of the blade, which are used as an input to FAST and the structural model. The bending of the tower and the blades calculated in FAST is then compared to the bendings from the model, and the parameters is estimated to give the best result.

Implementation of this parameter estimation is done by the MATLAB function `pem`.

The wind input for AeroDyn is shown in Figure 10.11. The wind field consists of steps of 5 m/s until 25 m/s where the wind speed returns to 0 m/s.

Normally the wind will not change that steep, but in this parameter estimation it will

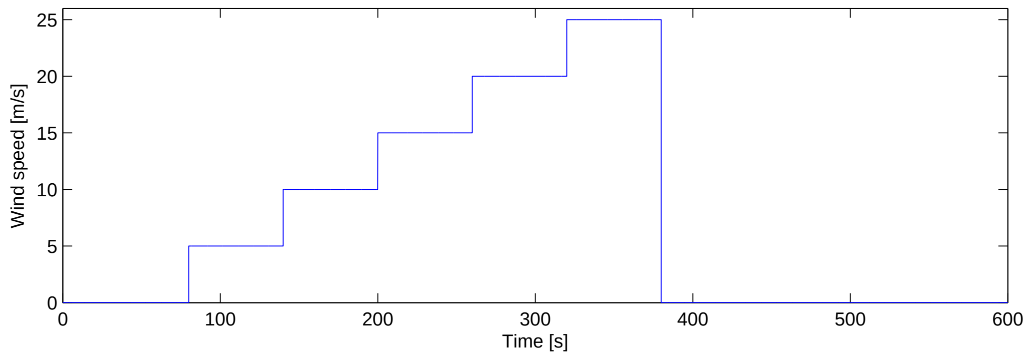


Figure 10.11: Wind input for estimation of parameters in the structural model

give a clear change in deflection of the blades and tower.
The pitch has been set to 0° , and the hinge inserted in the blades in placed between 8th and 9th blade element. Furthermore, all degrees of freedom is turned of in FAST, besides the ones for tower and blade deflection.

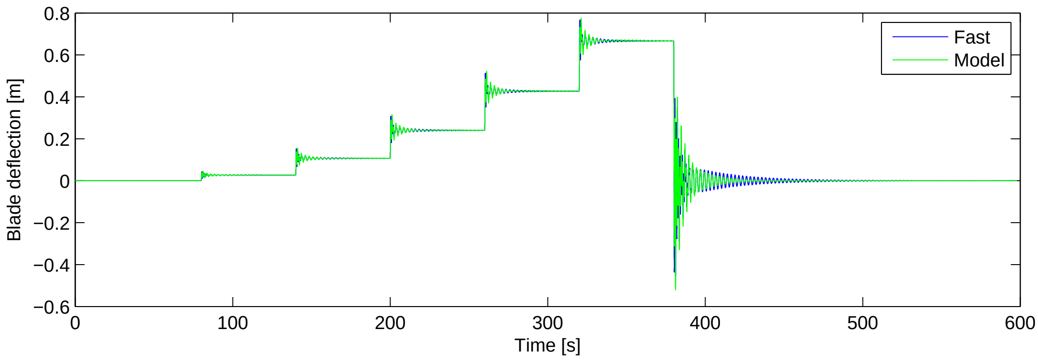
Running the estimation results in the parameters presented in Table 10.3, where also the initial guesses are listed. These guesses are based on the project [10, p. 119].

Parameter		**Initial guess**	**Estimation**
Blade spring	[N/m]	2,32E4	2,59E4
Blade damper	[N/(m/s)]	3,26E2	2,11E2
Blade mass	[kg]	3,61E2	4,06E2
Tower spring	[N/m]	9,30E5	4,76E5
Tower damper	[N/(m/s)]	2,41E4	1,12E4
Tower mass	[kg]	1,38E5	7,30E4

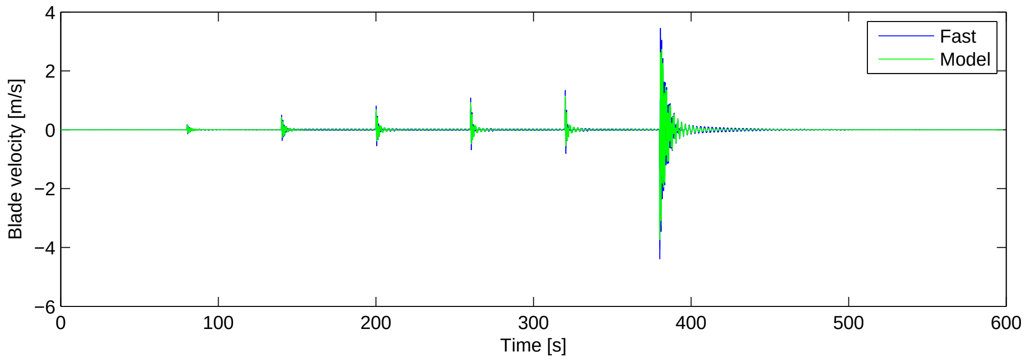
Table 10.3: Estimated structural parameters

In Chapter 6, different spring, damper and mass coefficients where given for each blade, but it is assumed that these coefficients are equal.

When inserting the parameters into the structural model this yields deflection and velocity of the blades presented in Figure 10.12. Originally the fitting process should take both deflection and velocity into account, but it turned out that the fit were best when only the deflection were considered. This is probably because the velocity is not an output from FAST, but were found be differentiating the deflections.



(a) Blade deflection

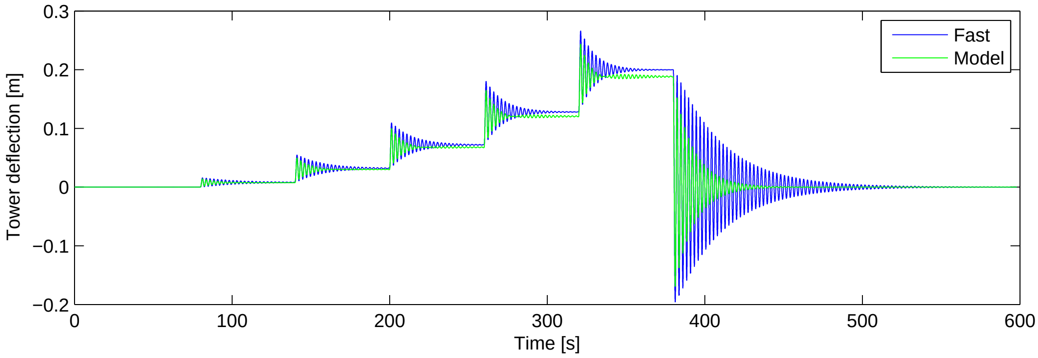


(b) Blade velocity

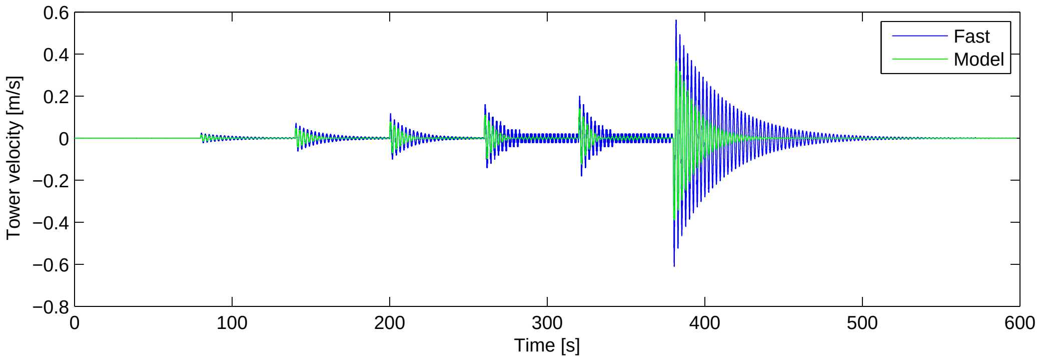
Figure 10.12: Comparison of deflection and velocity of blade from FAST and structural model with the estimated parameters

As it can be seen, is the fit satisfactory, it is not perfect but it is definitely suitable for this project.

The deflection and velocity of the tower where estimated as well, this is shown in Figure 10.13.



(a) Tower deflection



(b) Tower velocity

Figure 10.13: Comparison of deflection and velocity of tower from FAST and structural model with the estimated parameters

As it can be seen in the figures, the fit with the FAST output is not exact. But the result is close enough to be used in this project.

Chapter 11

Combined model

In the previous chapters, the different submodels for the project have been derived. In this chapter, the equations from the previous chapters are collected into a complete model of the wind turbine.

The model is illustrated in Figure 11.1, where it can be seen that the model is divided into two parts, where the nonlinear model serves as an input to the linear model.

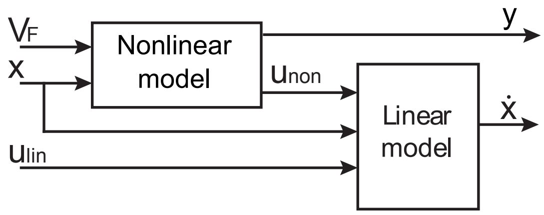


Figure 11.1: Complete model with linear and nonlinear part

V_{F} is the wind input for the aerodynamic model, x is the state vector, u_{non} and u_{lin} are input vectors to the linear block, these are listed in Table 11.1. y is an output vector, which contains the yaw torque.

	Input	Description
u_{non}	$F_{\mathrm{root},1}$	Thrust force from the root of blade 1
	$F_{\mathrm{tip},1}$	Thrust force from the tip of blade 1
	$F_{\mathrm{root},2}$	Thrust force from the root of blade 2
	$F_{\mathrm{tip},2}$	Thrust force from the tip of blade 2
	$F_{\mathrm{root},3}$	Thrust force from the root of blade 3
	$F_{\mathrm{tip},3}$	Thrust force from the tip of blade 3
	τ_{R}	Torque from the rotor
u_{lin}	τ_{G}	Generator torque reference
	$\beta_{\mathrm{ref},i}$	Blade pitch reference for blade i, i = $\{1, 2, 3\}$

Table 11.1: Inputs for the linear model

The nonlinear model consists of the aerodynamic model, and the linear consists of the structural model, mechanical model, pitch system model and the generator model. The state vector for the full model is shown in Table 11.2.

State	Description
$x_{\mathrm{b},1}$	Deflection of blade 1
$\dot{x}_{\mathrm{b},1}$	Velocity of blade 1
$x_{\mathrm{b},2}$	Deflection of blade 2
$\dot{x}_{\mathrm{b},2}$	Velocity of blade 2
$x_{\mathrm{b},3}$	Deflection of blade 3
$\dot{x}_{\mathrm{b},3}$	Velocity of blade 3
x_{t}	Deflection of the tower
$\dot{x}_{\mathrm{t}}$	Velocity of the tower
$\dot{\theta}_{\mathrm{R}}$	Angular velocity of the rotor
$\dot{\theta}_{\mathrm{G}}$	Angular velocity of the generator
θ_{δ}	Torsion angle of drive train
β_i	Angle of the i^{th} pitch angle, $\mathrm{i} = \{1, 2, 3\}$

Table 11.2: State vector of the system model

The first part of the model combination is the nonlinear model which will now be summarized.

11.1 Nonlinear model

The nonlinear model contains the aerodynamic model of the blade. It takes the state vector and the wind field as input and the output is the thrust force, yaw torque and rotational torque. A block diagram representation of the nonlinear model is shown in Figure 11.2.

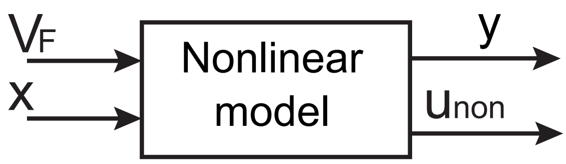


Figure 11.2: Inputs and outputs to the nonlinear model

V_{F} is the wind field, which is a 16x16 matrix and y is a 3x1 vector describing the yaw torque for each blade, furthermore is x the state vector and u_{non} the input for the linear model described earlier containing the elements shown in the vector in 11.1.

$$u_{\mathrm{non}} = \begin{bmatrix} \mathrm{F}_{\mathrm{root},1} & F_{\mathrm{tip},1} & F_{\mathrm{root},2} & F_{\mathrm{tip},2} & F_{\mathrm{root},3} & F_{tip,3} & \tau_{\mathrm{R,tot}} \end{bmatrix}^{\mathsf{T}} \tag{11.1}$$

Aerodynamic model

The equations describing the aerodynamic model will now be listed. The thrust forces affecting each blade and thereby the tower is described in equation 11.2 and 11.3.

$$\begin{aligned}
F_{\mathrm{T,root}} &= \left(\sum_{i=r_0}^{r_\mathrm{W}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{bl1}(r_i) dr + \sum_{i=r_\mathrm{W}}^{r_\mathrm{H}} \frac{\rho}{2} V_{rel}^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr \right) && (11.2)\\
&\cdot \left(C_\mathrm{L}(\alpha)\cos(\phi) + C_\mathrm{D}(\alpha)\sin(\phi) \right)\\
F_{\mathrm{T,tip}} &= \left(\sum_{i=r_\mathrm{H}}^{R} \frac{\rho}{2} V_{\mathrm{rel}}'^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr \right) && (11.3)\\
&\cdot \left(C_\mathrm{L}(\alpha')\cos(\phi') + C_\mathrm{D}(\alpha')\sin(\phi') \right)
\end{aligned}$$

These are the equations that are used as input in the model of the structural model, causing tower and blades to bend. The torque, from a single blade, that is transfered to the mechanical part of the system is shown in equation 11.4 to 11.6.

$$\begin{aligned}
\tau_{\mathrm{R,blade},}k &= \tau_{\mathrm{R,root}} + \tau_{\mathrm{R,tip}} && (11.4)\\
\tau_{\mathrm{R,root}} &= \left(\sum_{i=r_0}^{r_\mathrm{W}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{\mathrm{bl1}}(r_i) dr \cdot r_i \right. && (11.5)\\
&\left. + \sum_{i=r_\mathrm{W}}^{r_\mathrm{H}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr \cdot r_i \right) \cdot \left(C_\mathrm{L}(\alpha)\sin(\phi) - C_\mathrm{D}(\alpha)\cos(\phi) \right)\\
\tau_{\mathrm{R,tip}} &= \left(\sum_{i=r_\mathrm{H}}^{R} \frac{\rho}{2} V_{\mathrm{rel}}'^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr \cdot r_{\mathrm{tip},i} \right) && (11.6)\\
&\cdot \left(C_\mathrm{L}(\alpha')\sin(\phi) - C_\mathrm{D}(\alpha')\cos(\phi') \right)
\end{aligned}$$

In both the expression of the thrust force and rotor torque, the relative wind velocity is used, this is expressed in equation 11.7 and 11.8.

$$\begin{aligned}
V_{\mathrm{rel}} &= \sqrt{(V - \dot{x}_\mathrm{t})^2 + (\dot{\theta}_\mathrm{R} \cdot r)^2} && \text{for } r_i \leq r_\mathrm{H} && (11.7)\\
V_{\mathrm{rel}}' &= \sqrt{(\cos(\theta_\mathrm{H})(V - \dot{x}_\mathrm{b} - \dot{x}_\mathrm{t}))^2 + (\dot{\theta}_\mathrm{R} \cdot r)^2} && \text{for } r_i > r_\mathrm{H} && (11.8)\\
\theta_\mathrm{H} &= \mathrm{asin}\left(\frac{x_\mathrm{b}}{l_{\mathrm{tip}}} \right) && && (11.9)
\end{aligned}$$

Furthermore, the angle of attack and the angle of the relative wind, in proportion to the rotor plane, are expressed in equation 11.10 and 11.11.

$$\begin{aligned} \phi &= \tan^{-1}\left(\frac{V-\dot{x}_\mathrm{t}}{\dot{\theta}_\mathrm{R} r}\right) \\ \alpha &= \tan^{-1}\left(\frac{V-\dot{x}_\mathrm{t}}{\dot{\theta}_\mathrm{R} r}\right) - (\beta + \beta_\mathrm{twist}) \end{aligned} \qquad \text{for } r_i \leq r_\mathrm{H} \tag{11.10}$$

$$\begin{aligned} \phi' &= \tan^{-1}\left(\frac{V-\dot{x}_\mathrm{b}-\dot{x}_\mathrm{t}}{\dot{\theta}_\mathrm{R} r}\right) \\ \alpha' &= \tan^{-1}\left(\frac{V-\dot{x}_\mathrm{b}-\dot{x}_\mathrm{t}}{\dot{\theta}_\mathrm{R} r}\right) - (\beta + \beta_\mathrm{twist}) \end{aligned} \qquad \text{for } r_i > r_\mathrm{H} \tag{11.11}$$

The lift and drag coefficients and the length of the chord are found from table references in FAST.

The output of the this part of the nonlinear model will now be the rotational torque $\tau_\mathrm{R,tot}$ and the thrust force of the root and tip of the blades, $F_\mathrm{T,root}$ and $F'_\mathrm{T,tip}$ which is used for the last part of the non linear model, the yaw torque.

Yaw torque model

Equations for the yaw torque where derived in Section 6.1. Here the thrust force from the aerodynamic model where used to describe how an yaw torque where affecting the wind turbine.

The yaw torque contribution from the k^th blade is calculated in equation 11.12.

$$\tau_{\mathrm{yaw,z},k} = \sin(\varphi_k) \cdot (\tau_{\mathrm{yaw,root},}k + \tau_{\mathrm{yaw,tip},}k) \tag{11.12}$$

φ_k is the azimuth angle of the k^th blade, which has to be supplied to the formula. With the contributions from the root and tip, $\tau_\mathrm{yaw,root}$ and $\tau_\mathrm{yaw,tip}$ described in equations 11.13 and 11.14.

$$\begin{aligned} \tau_\mathrm{yaw,root} = &\left(\sum_{i=r_0}^{r_\mathrm{W}} \frac{\rho}{2} V_\mathrm{rel}^2(r_i) \cdot c_\mathrm{bl1}(r_i) dr \cdot r_i + \sum_{i=r_\mathrm{W}+1}^{r_\mathrm{H}} \frac{\rho}{2} V_\mathrm{rel}^2(r_i) \right. \qquad (11.13) \\ &\left. \cdot\, c_\mathrm{bl2}(r_i) dr \cdot r_i \right) \cdot (C_\mathrm{L}(\alpha)\cos(\phi) + C_\mathrm{D}(\alpha)\sin(\phi)) \\ \tau_\mathrm{yaw,tip} = &\left(\sum_{i=r_\mathrm{H}+1}^{R} \frac{\rho}{2} V_\mathrm{rel}'^2(r_i) \cdot c_\mathrm{bl2}(r_i) dr \cdot l_\mathrm{yaw}(r_i) \right) \qquad (11.14) \\ &\cdot \big(C_\mathrm{L}(\alpha')\cos(\phi') + C_\mathrm{D}(\alpha')\sin(\phi')\big) \cdot \cos(\theta_H + \theta_\epsilon) \end{aligned}$$

where most of the coefficients is the same as for the aerodynamic model just described, and the remaining described in equations 11.15 to 11.17.

$$A = l_{\text{overhang}} - l_{\text{tip}}(r_i)\sin(\theta_H) \tag{11.15}$$

$$\theta_\epsilon = \tan^{-1}\left(\frac{A}{l_{\text{root}} + l_{\text{tip}}(r_i)\cos(\theta_H)}\right) \tag{11.16}$$

$$l_{\text{yaw}}(r_i) = \sqrt{A^2 + (l_{\text{root}} + l_{\text{tip}}(r_i)\cos(\theta_H))^2} \tag{11.17}$$

The output of the nonlinear model are the rotational torque $\tau_{\text{R,tot}}$ and the thrust force of the root and tip of the blades, $F_{\text{T,root}}$ and $F'_{\text{T,tip}}$, all of these variables will be used as inputs to different parts of the linear model. Furthermore, can the yaw torque τ_{yaw} be used as an input to the controller.

11.2 Linear model

After the examining the nonlinear model, the linear model will now be combined to one complete state space representation. A block diagram of the linear model is shown in Figure 11.3.

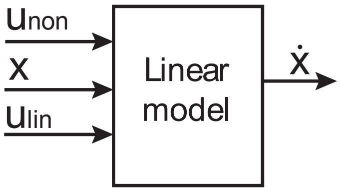


Figure 11.3: Inputs and output of the linear model

The linear model consists of the structural dynamics, the mechanical dynamics and the pitch system. These subsystems are merge into one state space model by the equations 11.18 to 11.22.

$$\begin{aligned}\dot{x} &= A_{\mathrm{s}}x + B_{\mathrm{s}}u \\ y &= C_{\mathrm{s}}x + D_{\mathrm{s}}u\end{aligned} \tag{11.18}$$

where

$$u = \begin{bmatrix} u_{\mathrm{non}} & u_{\mathrm{lin}} \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} u_{\mathrm{str}} & u_{\mathrm{mech}} & u_{\beta} \end{bmatrix}^{\mathsf{T}} \tag{11.19}$$

$$x = \begin{bmatrix} x_{\mathrm{str}} & x_{\mathrm{mech}} & x_{\beta} \end{bmatrix}^{\mathsf{T}} \tag{11.20}$$

$$A_{\mathrm{s}} = \begin{bmatrix} A_{\mathrm{str}} & 0_{8\mathrm{x}3} & 0_{8\mathrm{x}3} \\ 0_{3\mathrm{x}8} & A_{\mathrm{mech}} & 0_{3\mathrm{x}3} \\ 0_{3\mathrm{x}8} & 0_{3\mathrm{x}3} & A_{\beta} \end{bmatrix} \quad B_{\mathrm{s}} = \begin{bmatrix} B_{\mathrm{str}} & 0_{8\mathrm{x}2} & 0_{8\mathrm{x}3} \\ 0_{3\mathrm{x}6} & B_{\mathrm{mech}} & 0_{3\mathrm{x}3} \\ 0_{3\mathrm{x}6} & 0_{3\mathrm{x}2} & B_{\beta} \end{bmatrix} \tag{11.21}$$

$$C_{\mathrm{s}} = I_{14\mathrm{x}14} \quad D_{\mathrm{s}} = 0_{14\mathrm{x}11} \tag{11.22}$$

Each of these state space models will now be described.

Structural part

The structural model where found in Chapter 6. It takes the thrust forces from the nonlinear model as input, and has the deflection of tower and blades as output. In equation 11.23 to 11.26, the state space representation of the structural model is shown.

$$A_{\mathrm{str}} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{K_{\mathrm{b},1}}{M_{\mathrm{b},1}} & -\frac{B_{\mathrm{b},1}}{M_{\mathrm{b},1}} & 0 & 0 & 0 & 0 & \frac{K_{\mathrm{b},1}}{M_{\mathrm{b},1}} & \frac{B_{\mathrm{b},1}}{M_{\mathrm{b},1}} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{K_{\mathrm{b},2}}{M_{\mathrm{b},2}} & -\frac{B_{\mathrm{b},2}}{M_{\mathrm{b},2}} & 0 & 0 & \frac{K_{\mathrm{b},2}}{M_{\mathrm{b},2}} & \frac{B_{\mathrm{b},2}}{M_{\mathrm{b},2}} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{K_{\mathrm{b},3}}{M_{\mathrm{b},3}} & -\frac{B_{\mathrm{b},3}}{M_{\mathrm{b},3}} & \frac{K_{\mathrm{b},3}}{M_{\mathrm{b},3}} & \frac{B_{\mathrm{b},3}}{M_{\mathrm{b},3}} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -\frac{K_{\mathrm{b},1}}{M_{\mathrm{t}}} & -\frac{B_{\mathrm{b},1}}{M_{\mathrm{t}}} & -\frac{K_{\mathrm{b},2}}{M_{\mathrm{t}}} & -\frac{B_{\mathrm{b},2}}{M_{\mathrm{t}}} & -\frac{K_{\mathrm{b},3}}{M_{\mathrm{t}}} & -\frac{B_{\mathrm{b},3}}{M_{\mathrm{t}}} & a_{8,7} & a_{8,8} \end{bmatrix} \tag{11.23}$$

$$a_{8,7} = \frac{K_{\mathrm{b},1}+K_{\mathrm{b},2}+K_{\mathrm{b},3}-K_{\mathrm{t}}}{M_{\mathrm{t}}}$$

$$a_{8,8} = \frac{B_{\mathrm{b},1}+B_{\mathrm{b},2}+B_{\mathrm{b},3}-B_{\mathrm{t}}}{M_{\mathrm{t}}}$$

$$B_{\mathrm{str}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{M_{\mathrm{b},1}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{M_{\mathrm{b},2}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{M_{\mathrm{b},3}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} & \frac{1}{M_{\mathrm{t}}} \end{bmatrix} \tag{11.24}$$

$$x_{\mathrm{str}} = \begin{bmatrix} x_{\mathrm{b},1} & \dot{x}_{\mathrm{b},1} & x_{\mathrm{b},2} & \dot{x}_{\mathrm{b},2} & x_{\mathrm{b},3} & \dot{x}_{\mathrm{b},3} & x_{\mathrm{t}} & \dot{x}_{\mathrm{t}} \end{bmatrix}^{\mathsf{T}} \tag{11.25}$$

$$u_{\mathrm{str}} = \begin{bmatrix} F_{\mathrm{root},1} & F_{\mathrm{tip},1} & F_{\mathrm{root},2} & F_{\mathrm{tip},2} & F_{\mathrm{root},3} & F_{\mathrm{tip},3} \end{bmatrix}^{\mathsf{T}} \tag{11.26}$$

Mechanical part

The states and inputs for mechanical part of the model, found in Chapter 7 are shown in 11.27. And the system can be described in a state space system with the matrices in 11.28 and 11.29.

$$x_{\mathrm{mech}} = \begin{bmatrix} \dot{\theta}_{\mathrm{R}} & \dot{\theta}_{\mathrm{G}} & \theta_{\delta} \end{bmatrix}^{\mathsf{T}} \quad u_{\mathrm{mech}} = \begin{bmatrix} \tau_{\mathrm{R}} & \tau_{\mathrm{G}} \end{bmatrix}^{\mathsf{T}} \tag{11.27}$$

$$A_{\mathrm{mech}} = \begin{bmatrix} \frac{-(B_{\mathrm{DT}}+B_{\mathrm{R}})}{J_{\mathrm{R}}} & \frac{B_{\mathrm{DT}}}{N_{\mathrm{G}} J_{\mathrm{R}}} & \frac{-K_{\mathrm{DT}}}{J_{\mathrm{R}}} \\ \frac{B_{\mathrm{DT}}}{N_{\mathrm{G}} J_{\mathrm{G}}} & -\left(\frac{B_{\mathrm{DT}}}{N_{\mathrm{G}}^2 J_{\mathrm{G}}} + \frac{B_{\mathrm{G}}}{J_{\mathrm{G}}}\right) & \frac{K_{\mathrm{DT}}}{N_{\mathrm{G}} J_{\mathrm{G}}} \\ 1 & \frac{-1}{N_{\mathrm{G}}} & 0 \end{bmatrix} \tag{11.28}$$

$$B_{\mathrm{mech}} = \begin{bmatrix} \frac{1}{J_{\mathrm{R}}} & 0 \\ 0 & \frac{-1}{J_{\mathrm{G}}} \\ 0 & 0 \end{bmatrix} \tag{11.29}$$

Pitch system model

The last part of the linear model is the pitch system which were found in Chapter 8.

The pitch system can be expressed as a state space system with the state and input vector and system matrixes shown in equations 11.30 to 11.31.

$$x_\beta = \begin{bmatrix}\beta_1 & \beta_2 & \beta_3\end{bmatrix}^\mathsf{T} \quad u_\beta = \begin{bmatrix}\beta_{\text{ref},1} & \beta_{\text{ref},2} & \beta_{\text{ref},3}\end{bmatrix}^\mathsf{T} \tag{11.30}$$

$$A_\beta = \begin{bmatrix} \frac{-1}{\tau_{\text{time,P}}} & 0 & 0 \\ 0 & \frac{-1}{\tau_{\text{time,P}}} & 0 \\ 0 & 0 & \frac{-1}{\tau_{\text{time,P}}} \end{bmatrix} \quad B_\beta = \begin{bmatrix} \frac{1}{\tau_{\text{time,P}}} & 0 & 0 \\ 0 & \frac{1}{\tau_{\text{time,P}}} & 0 \\ 0 & 0 & \frac{1}{\tau_{\text{time,P}}} \end{bmatrix} \tag{11.31}$$

Before the system model can be validated the nonlinear model, consisting of the aerodynamics and the yaw torque, will be linearized, this is done in Chapter 12.

Chapter 12

Linearization of aerodynamics

In Chapter 5, the equations 12.1 to 12.3 were found, as it can be seen, these contains nonlinear expressions, table references, and sums, which have to be linearized in order to use linear methods for controller design.

The procedure in linearizing the aerodynamic model will be as follows. Firstly the operation points for the linearization is found. The equations are then approximated to continuously and differentiable functions. Subsequently the Taylor approximation can be computed.

$$\begin{aligned}
F_{\mathrm{T,root}}\Big(&r_i, V_{\mathrm{rel}}^2(V,\dot{\theta}_{\mathrm{R}},r_i,\dot{x}_{\mathrm{t}}), C_{\mathrm{L}}(\alpha), C_{\mathrm{D}}(\alpha), \alpha(V,\dot{\theta}_{\mathrm{R}},r_i,\beta),\\
&\phi(V,\dot{\theta}_{\mathrm{R}},r_i), c_{\mathrm{bl1}}(r_i), c_{\mathrm{bl2}}(r_i)\Big) =\\
&a_{\mathrm{F,root}}\left(\sum_{i=r_0}^{r_{\mathrm{W}}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) c_{\mathrm{bl1}}(r_i) dr + \sum_{i=r_{\mathrm{W}}}^{r_{\mathrm{H}}} \frac{\rho}{2} V_{\mathrm{rel}}^2(r_i) \cdot c_{\mathrm{bl2}}(r_i) dr\right)\\
&\cdot \left(C_{\mathrm{L}}(\alpha)\cos(\phi) + C_{\mathrm{D}}(\alpha)\sin(\phi)\right)
\end{aligned} \tag{12.1}$$

$$\begin{aligned}
F_{\mathrm{T,tip}}\Big(&r_i, V_{\mathrm{rel}}'^2(V,\dot{\theta}_{\mathrm{R}},r_i,\dot{x}_{\mathrm{b}},\dot{x}_{\mathrm{t}},\theta_{\mathrm{H}}), C_{\mathrm{L}}(\alpha'), C_{\mathrm{D}}(\alpha'), \alpha'(V,\dot{\theta}_{\mathrm{R}},r_i,\\
&\dot{x}_{\mathrm{b}},\dot{x}_{\mathrm{t}},\beta), \phi'(V,\dot{\theta}_{\mathrm{R}},r_i,\dot{x}_{\mathrm{b}},\dot{x}_{\mathrm{t}}), c_{\mathrm{bl1}}(r_i), c_{\mathrm{bl2}}(r_i)\Big) =\\
&a_{\mathrm{F,tip}}\left(\sum_{i=r_{\mathrm{H}}}^{R} \frac{\rho}{2} V_{\mathrm{rel}}'^2(r_i) c_{\mathrm{bl2}}(r_i) dr\right)\\
&\cdot \big(C_{\mathrm{L}}(\alpha')\cos(\phi') + C_{\mathrm{D}}(\alpha')\sin(\phi')\big)
\end{aligned} \tag{12.2}$$

$$\tau_{\mathrm{R,blade}}\Big(r_i, r_{\mathrm{tip},i}, V_{\mathrm{rel}}(V, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \dot{\theta}_{\mathrm{R}}, \theta_{\mathrm{H}}), V'^2_{\mathrm{rel}}(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \theta_{\mathrm{H}}),$$
$$C_{\mathrm{L}}(\alpha), C_{\mathrm{D}}(\alpha), C_{\mathrm{L}}(\alpha'), C_{\mathrm{D}}(\alpha'), \alpha(V, \dot{\theta}_{\mathrm{R}}, r_i, \beta), \phi(V, \dot{\theta}_{\mathrm{R}}, r_i),$$
$$\alpha'(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \beta), \phi'(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}), c_{\mathrm{bl1}}(r_i), c_{\mathrm{bl2}}(r_i)\Big) =$$
$$a_{\tau,\mathrm{root}}\left(\sum_{i=r_0}^{r_{\mathrm{W}}} \frac{\rho}{2} V^2_{\mathrm{rel}}(r_i) c_{\mathrm{bl1}}(r_i) dr \cdot r_i + \sum_{i=r_{\mathrm{W}}+1}^{r_{\mathrm{H}}} \frac{\rho}{2} V^2_{\mathrm{rel}}(r_i)\right.$$
$$\left. c_{\mathrm{bl2}}(r_i) dr \cdot r_i\right) \cdot (C_{\mathrm{L}}(\alpha)\sin(\phi) - C_{\mathrm{D}}(\alpha)\cos(\phi)) +$$
$$a_{\tau,\mathrm{tip}}\left(\sum_{i=r_{\mathrm{H}}+1}^{R} \frac{\rho}{2} V'^2_{\mathrm{rel}}(r_i) c_{\mathrm{bl2}}(r_i) dr \cdot r_{\mathrm{tip},i}\right)$$
$$\cdot \big(C_{\mathrm{L}}(\alpha')\sin(\phi') - C_{\mathrm{D}}(\alpha')\cos(\phi')\big) \tag{12.3}$$

12.1 Operation points

As it has been chosen to focus on a wind field of 15 m/s, this will be used as basis for the operation points. A simulation of the nonlinear model with the FAST controller, has been running for 600 seconds to get a steady values and the state vectors has been used as operation points along with the wind speed of 15 m/s.

The operation points are:

$$x_b = 0{,}908 \text{ m} \tag{12.4}$$
$$\dot{x}_b = 0 \text{ m/s} \tag{12.5}$$
$$x_t = 0{,}199 \text{ m} \tag{12.6}$$
$$\dot{x}_t = 0 \text{ m/s} \tag{12.7}$$
$$\dot{\theta}_{\mathrm{R}} = 2{,}143 \text{ rad/s} \tag{12.8}$$
$$\beta = 0{,}251 \text{ rad} \tag{12.9}$$

These operation points are used later, in the linearization.

12.2 Preparation for linearization

The wind input contains summations, which must be redefined in order to make these differentiable. The wind field is contracted into two parts for each blade, one for the root of the blade and one for the tip. These are shown in 12.10 and 12.11.

$$V_{\text{root}}(\theta_{\text{blade}}) = \sum_{i=r_0}^{r_{\text{H}}} v(\theta_{\text{blade}}, r_i) \tag{12.10}$$

$$V_{\text{tip}}(\theta_{\text{blade}}) = \sum_{i=r_{\text{H}}+1}^{R} v(\theta_{\text{blade}}, r_i) \tag{12.11}$$

where the elements are divided as $r_0 = 1$, $r_{\text{H}} = 8$ and $R = 15$.

As described, the blades are divided into two parts, divided at the hinge placement, the calculation of forces and torques are made for each part. The radius to the two points where the calculations are made are found by taking the mean of each region, root and tip, which yields the following:

$$r_{\text{root}} = 10{,}6125 \text{ m} \tag{12.12}$$

$$r_{\text{tip}} = 27{,}2296 \text{ m} \tag{12.13}$$

In the nonlinear equations, there are some functions and table references which is needed change to a differentiable description, this will now be done starting with the chord length.

Element chord length

In the aerodynamic equation, the length of the airfoil chords is included, this can be found by a table reference in FAST. To avoid this table lookup, the area of the blade regions can be found, the root and tip area. The areas are calculated by multiplying width and length of each blade element and summarize these.

$$A_{\mathrm{root}}(c_{\mathrm{bl1}}(r_i), c_{\mathrm{bl2}}(r_i)) = dr \left(\sum_{i=r_0}^{r_{\mathrm{W}}} c_{\mathrm{bl1}}(r_i) + \sum_{i=r_{\mathrm{W}}+1}^{r_{\mathrm{H}}} c_{\mathrm{bl2}}(r_i) \right) \tag{12.14}$$
$$= 2{,}21667\ \mathrm{m} \cdot 18{,}7593\ \mathrm{m} = 41{,}5832\ \mathrm{m}^2$$

$$A_{\mathrm{tip}}(c_{\mathrm{bl2}}(r_i)) = dr \sum_{i=r_{\mathrm{H}}+1}^{R} c_{\mathrm{bl2}}(r_i) \tag{12.15}$$
$$= 2{,}21667\ \mathrm{m} \cdot 10{,}1156\ \mathrm{m} = 22{,}4229\ \mathrm{m}^2$$

where $c_{\mathrm{bl1}}(r_i)$ and $c_{\mathrm{bl2}}(r_i)$ are the width of the element r_i.

Lift and drag factor

The lift and drag factors are as well linearized, to determine the lift and drag force in proportion to a certain angle of attack.

As for the blade width, the lift and drag factors for the specific airfoil can be found in the source code for FAST [23]. The factors is plotted in Figure 12.1, where the lift factor C_{L} is expressed as the mean of the three different lift factors, and as the drag factor C_{D} is the same for the three airfoil it is not necessary to take the mean of those. It is assumed that taking the mean of the lift factors this does not affect the result noticeable, because they are as similar as they are.

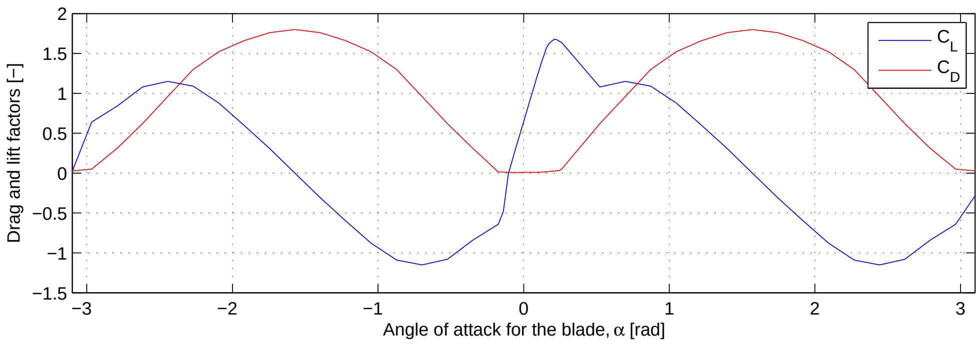


Figure 12.1: Lift and drag factor for the blade

To get a linear expression for these factors, a first order approximation is wanted, this can be acceptable in a limited area around an angle of attack of 0 rad in the interval from -0,1 to 0,2 rad. The first order approximation is made by equations 12.16 and 12.17.

$$C_{\mathrm{L}}(\alpha) = a_{\mathrm{L}} \cdot \alpha + b_{\mathrm{L}} \tag{12.16}$$
$$C_{\mathrm{D}}(\alpha) = a_{\mathrm{D}} \cdot \alpha + b_{\mathrm{D}} \tag{12.17}$$

In Figure 12.2, MATLAB's `polyfit` function has been used to find two approximations.

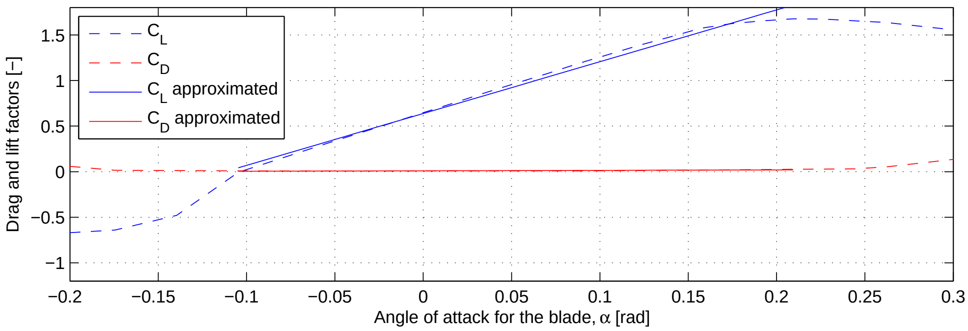


Figure 12.2: Drag and lift factor for the blade

The coefficients used in equations 12.16 and 12.17 is shown in Table 12.1.

Coefficient	Value
a_{L}	5,6827
b_{L}	0,6374
a_{D}	0,0495
b_{D}	0,0092

Table 12.1: List of approximated lift and drag coefficients

This approximation is as for the approximation of the blade width used when calculating the forces acting on the blades of the wind turbine.

It should be noted that this approximation only holds within the boundary of -0,1 and 0,2 rad, outside this another approximation is necessary to get a valid solution.

Pre-pitch of the blade

A wind turbine blade have a pre-pitch, β_{twist}, which implies that the angle of attack on each blade element is optimal. It is wanted to linearize this pre-pitch within the two given regions.

To linearize the pre-pitch, the mean of the root and tip pre-pitch is calculated which yields:

$$\beta_{\text{pre,root}} = 0{,}16456 \text{ rad} \tag{12.18}$$
$$\beta_{\text{pre,tip}} = 0{,}01522 \text{ rad} \tag{12.19}$$

A plot of the original pre-pitch is shown in Figure 12.3, with the approximation plotted as well.

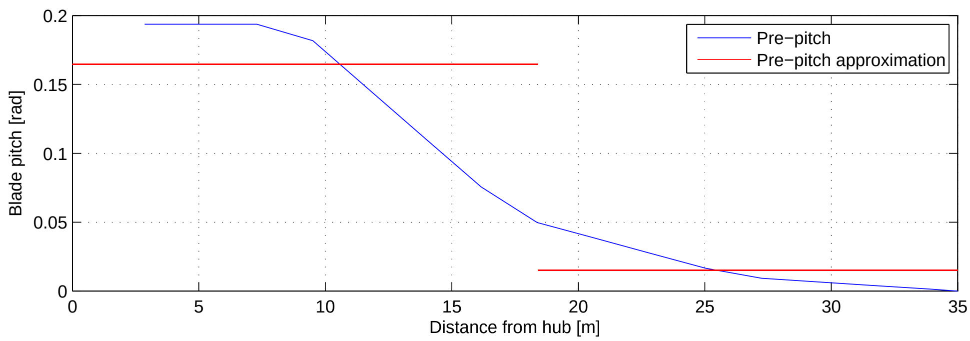


Figure 12.3: Pre-pitch of the blade, shown with the approximation

12.3 Combined equations

The equations expressing the aerodynamic model can now be written in a differential form:

$$
\begin{aligned}
&F_{\mathrm{T,root}}\Big(V_{\mathrm{rel}}^2(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{t}}), \alpha(V, \dot{\theta}_{\mathrm{R}}, r_i, \beta), \phi(V, \dot{\theta}_{\mathrm{R}}, r_i)\Big) = \\
&\quad a_{\mathrm{F,root}}\frac{\rho}{2}V_{\mathrm{rel}}^2 A_{\mathrm{root}}((5,6827 \cdot \alpha + 0,6374)\cos(\phi) \\
&\quad + (0,0495 \cdot \alpha + 0,0092)\sin(\phi)
\end{aligned}
\tag{12.20}
$$

$$
\begin{aligned}
&F_{\mathrm{T,tip}}\Big(V_{\mathrm{rel}}'^2(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \theta_{\mathrm{H}}), \alpha'(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \beta), \\
&\quad \phi'(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}})\Big) = \\
&\quad a_{\mathrm{F,tip}}\frac{\rho}{2}V_{\mathrm{rel}}'^2 A_{\mathrm{tip}}\,\big((5,6827 \cdot \alpha' + 0,6374)\cos(\phi') \\
&\quad +(0,0495 \cdot \alpha' + 0,0092)\sin(\phi')\big)
\end{aligned}
\tag{12.21}
$$

$$
\begin{aligned}
&\tau_{\mathrm{R,blade}}\Big(V_{\mathrm{rel}}(V, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \dot{\theta}_{\mathrm{R}}, \theta_{\mathrm{H}}), V_{\mathrm{rel}}'^2(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \theta_{\mathrm{H}}), \\
&\quad \alpha(V, \dot{\theta}_{\mathrm{R}}, r_i, \beta), \phi(V, \dot{\theta}_{\mathrm{R}}, r_i), \alpha'(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \beta), \\
&\quad \phi'(V, \dot{\theta}_{\mathrm{R}}, r_i, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}})\Big) = \\
&\quad a_{\tau,\mathrm{root}}\frac{\rho}{2}V_{\mathrm{rel}}^2 A_{\mathrm{root}} r_{\mathrm{root}}\,((5,6827 \cdot \alpha + 0,6374)\cos(\phi) \\
&\quad -(0,0495 \cdot \alpha + 0,0092)\sin(\phi)) \\
&\quad + a_{\tau,\mathrm{tip}}\frac{\rho}{2}V_{\mathrm{rel}}'^2 A_{\mathrm{tip}} r_{\mathrm{tip}}\,\big((5,6827 \cdot \alpha' + 0,6374)\cos(\phi') \\
&\quad -(0,0495 \cdot \alpha' + 0,0092)\sin(\phi')\big)
\end{aligned}
\tag{12.22}
$$

where

$$\phi(V_{root}, \dot{\theta}_{\mathrm{R}}) = \tan^{-1}\left(\frac{V_{root}}{\dot{\theta}_{\mathrm{R}} r_{root}}\right) \tag{12.23}$$

$$\phi'(V_{tip}, \dot{\theta}_{\mathrm{R}}, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}) = \tan^{-1}\left(\frac{V_{tip} - \dot{x}_{\mathrm{b}} - \dot{x}_{\mathrm{t}}}{\dot{\theta}_{\mathrm{R}} r_{tip}}\right) \tag{12.24}$$

$$\alpha(V_{root}, \dot{\theta}_{\mathrm{R}}, \beta) = \tan^{-1}\left(\frac{V_{root}}{\dot{\theta}_{\mathrm{R}} r_{root}}\right) - (\beta + \beta_{\mathrm{pre,root}}) \tag{12.25}$$

$$\alpha'(V_{tip}, \dot{\theta}_{\mathrm{R}}, \dot{x}_{\mathrm{b}}, \dot{x}_{\mathrm{t}}, \beta) = \tan^{-1}\left(\frac{V_{tip} - \dot{x}_{\mathrm{b}} - \dot{x}_{\mathrm{t}}}{\dot{\theta}_{\mathrm{R}} r_{tip}}\right) - (\beta + \beta_{\mathrm{pre,tip}}) \tag{12.26}$$

$$V_{\text{rel}}^2(V_{root}, \dot{\theta}_{\text{R}}, \dot{x}_{\text{t}}) = (V_{root} - \dot{x}_{\text{t}})^2 + (\dot{\theta}_{\text{R}} \cdot r_{root})^2 \tag{12.27}$$

$$V_{\text{rel}}'^2(V_{tip}, \dot{\theta}_{\text{R}}, \dot{x}_{\text{b}}, \dot{x}_{\text{t}}, \theta_{\text{H}}) = (\cos(\theta_{\text{H}}) \cdot (V_{tip} - \dot{x}_{\text{b}} - \dot{x}_{\text{t}}))^2 + (\dot{\theta}_{\text{R}} \cdot r_{tip})^2 \tag{12.28}$$

and the fitting parameters from Section 10.1 as shown in Table 12.2.

Parameter	**Value**
$a_{\text{F,root}}$	0,9704
$a_{\text{F,tip}}$	0,9989
$a_{\tau,\text{root}}$	0,9735
$a_{\tau,\text{tip}}$	0,9959

Table 12.2: Fitting factors for the aerodynamic model

12.4 The Taylor approximation

The first order Taylor expression is stated in equation 12.29. x and u are the state and input vector. $\overline{x}$ and $\overline{u}$ are the operation points for the Taylor approximation found in Section 12.1.

$$f(x) \approx f(\overline{x}, \overline{u}) + \frac{\partial f(\overline{x})}{\partial x}(x - \overline{x}) + \frac{\partial f(\overline{u})}{\partial u}(u - \overline{u}) \tag{12.29}$$

The MATLAB function `taylor` has been used to find the linearized functions, which is shown in equation 12.30 to 12.32

$$\begin{aligned} F_{\text{T,root},i} \approx &- 3189\, \dot{x}_{\text{t}} + 556,77\, \dot{\theta}_R - 87524\, \beta_i \\ &+ 3189\, V_{\text{root}}(\theta_{\text{blade},i}) - 2545,8 \end{aligned} \tag{12.30}$$

$$\begin{aligned} F_{\text{T,tip},i} \approx &- 99,516\, x_{\text{b},i} - 4525,5\, \dot{x}_{\text{b},i} - 4525,5\, \dot{x}_{\text{t}} - 6666,7\, \dot{\theta}_R \\ &- 274244\, \beta_i + 4525,5\, V_{\text{tip}}(\theta_{\text{blade},i}) + 42127 \end{aligned} \tag{12.31}$$

$$\begin{aligned} \tau_{\text{R},i} \approx &- 71,685\, x_{\text{b},i} - 42858\, \dot{x}_{\text{b},i} - 76124\, \dot{x}_{\text{t}} - 212255\, \dot{\theta}_R \\ &- 2,4506\text{E}6\, \beta_i + 42858\, V_{\text{tip}}(\theta_{\text{blade},i}) \\ &+ 33266\, V_{\text{root}}(\theta_{\text{blade},i}) + 271666 \end{aligned} \tag{12.32}$$

These equations are linearized functions calculating $F_{\text{T,root},i}$, $F_{\text{T,tip},i}$ and $\tau_{\text{R},i}$ for each blade.

In order to connect these to the linear model some matrixes are introduced. A B_1 matrix which take the state vector x as input, and returns the thrust forces and rotor torque in the input vector u_{non}. The B_2 matrix has the u_{wind}, used equation 12.33, as input and returns thrust forces and rotor torque as well.

$$\begin{aligned} u_{\text{wind}} = [&V_{\text{root}}(\theta_{\text{blade},1})\ V_{\text{tip}}(\theta_{\text{blade},1})\ V_{\text{root}}(\theta_{\text{blade},2})\ V_{\text{tip}}(\theta_{\text{blade},2}) \\ &V_{\text{root}}(\theta_{\text{blade},3})\ V_{\text{tip}}(\theta_{\text{blade},3})\ ]^{\mathsf{T}} \end{aligned} \tag{12.33}$$

The constant terms that are not contained in these matrixes are collected in a vector called *opv*. The three vectors can now be summarized to get the nonlinear input vector u_{non}.

$$u_{\text{non}} = B_1 x + B_2 u_{\text{wind}} + opv \tag{12.34}$$

12.5 Combining state space systems

Collecting the state space system with the new matrixes is shown in equation 12.35.

$$\dot{x} = A_{\text{s}} x + B_{\text{s}} \left[\begin{smallmatrix} u_{\text{non}} \\ u_{\text{lin}} \end{smallmatrix}\right] \tag{12.35}$$

$$= A_{\text{s}} x + B_{\text{s}} \left[\begin{smallmatrix} B_1 x + B_2 u_{\text{wind}} + opv \\ u_{\text{lin}} \end{smallmatrix}\right] \tag{12.36}$$

where A_{s} and B_{s} is the linear state space matrixes from Section 11.2.

B_{s} is split into two parts, B_{s1} and B_{s2}. B_{s1} is the first seven columns having u_{non} as input, and B_{s2} is the last four columns of B_{s} with the input u_{lin}, which is the generator torque and pitch reference angles. This is shown in

$$B_{\text{s}} = \begin{bmatrix} B_{\text{s1}} & B_{\text{s2}} \end{bmatrix} \tag{12.37}$$

This leads to the following state space system

$$\dot{x} = A_{\text{s}} x + B_{\text{s1}}(B_1 x + B_2 u_{\text{wind}} + opv) + B_{\text{s2}} u_{lin} \tag{12.38}$$

$$= (A_{\text{s}} + B_{\text{s1}} B_1) x + \begin{bmatrix} B_{\text{s1}} B_2 & B_{\text{s2}} \end{bmatrix} \left[\begin{smallmatrix} u_{\text{wind}} \\ u_{\text{lin}} \end{smallmatrix}\right] + B_{\text{s1}} opv \tag{12.39}$$

The full linearized model of the wind turbine can now be described as a state space system.

$$\dot{x} = A_\mathrm{L} x + B_\mathrm{L} u \tag{12.40}$$
$$y = C_\mathrm{L} x + D_\mathrm{L} u \tag{12.41}$$

where

$$A_\mathrm{L} = A_\mathrm{s} + B_\mathrm{s1} B_1 \qquad B_\mathrm{L} = \begin{bmatrix} B_\mathrm{s1} B_2 & B_\mathrm{s2} \end{bmatrix} \tag{12.42}$$
$$C_\mathrm{L} = I_{14} \qquad D_\mathrm{L} = 0_{14,10} \tag{12.43}$$

and

$$x = \begin{bmatrix} x_\mathrm{b,1}\ \dot{x}_\mathrm{b,1}\ x_\mathrm{b,2}\ \dot{x}_\mathrm{b,2}\ x_\mathrm{b,3}\ \dot{x}_\mathrm{b,3} x_\mathrm{t}\ \dot{x}_\mathrm{t}\ \dot{\theta}_\mathrm{R}\ \dot{\theta}_\mathrm{G}\ \theta_\delta\ \beta_1\ \beta_2\ \beta_3 \end{bmatrix}^\mathsf{T} \tag{12.44}$$
$$u = \begin{bmatrix} u_\mathrm{wind} \\ u_\mathrm{lin} \end{bmatrix} \tag{12.45}$$
$$= \Big[V_\mathrm{root}(\theta_\mathrm{blade,1})\ V_\mathrm{tip}(\theta_\mathrm{blade,1})\ V_\mathrm{root}(\theta_\mathrm{blade,2})\ V_\mathrm{tip}(\theta_\mathrm{blade,2}) \ V_\mathrm{root}(\theta_\mathrm{blade,3})\ V_\mathrm{tip}(\theta_\mathrm{blade,3})\ \tau_\mathrm{G}\ \beta_\mathrm{ref,1}\ \beta_\mathrm{ref,2}\ \beta_\mathrm{ref,3} \Big]^\mathsf{T} \tag{12.46}$$

The term $B_\mathrm{s1} opv$ is omitted to remove the offset from the Taylor equations.

The full linearized matrixes are shown in Appendix D.

Chapter 13

Model validation

A complete model of a wind turbine has been collected, and the nonlinear parts of the model have been linearized. To ensure, that the models is a representation of the reference wind turbine from FAST, it will now be validated by comparing the full model with data generated from FAST.

The submodels found in the previous chapters, will all be validated, with a few exceptions, this is the wind model which cannot be validated as FAST can only generate a homogeneous wind field and the pitch system because there is no dynamics in the pitch system in FAST. Beside these validations that can be compared with FAST will the yaw torque output from the simulation be described in the end of the validation.

In Figure 13.1, an overall block diagram of the validation process can be seen. A wind field is used as input, and the different outputs from FAST and the two models are compared.

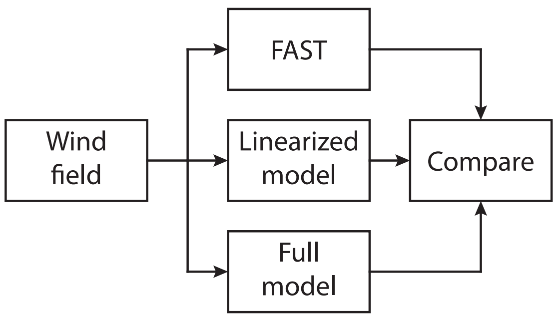


Figure 13.1: Block diagram of the model validation

Though it is only the aerodynamic model which has been linearized, the output from here will be an input for other parts of the model, therefor the linearized outputs from the structural and mechanical model will also be compared.

A homogeneous wind field with a basis speed of 15 m/s has been chosen to form the basis of the wind field which will be used in this model validation. The reason for this is that it is above the rated wind speed of 12 m/s, it is in this region the main operation of the wind turbine will be as a starting point.

Around the mean wind of 15 m/s the wind speed decreases and increases in steps of 1-3 m/s. A graph of the wind input can be seen in Figure 13.2. This will act as input to the FAST simulation, and for the model of the wind turbine model.

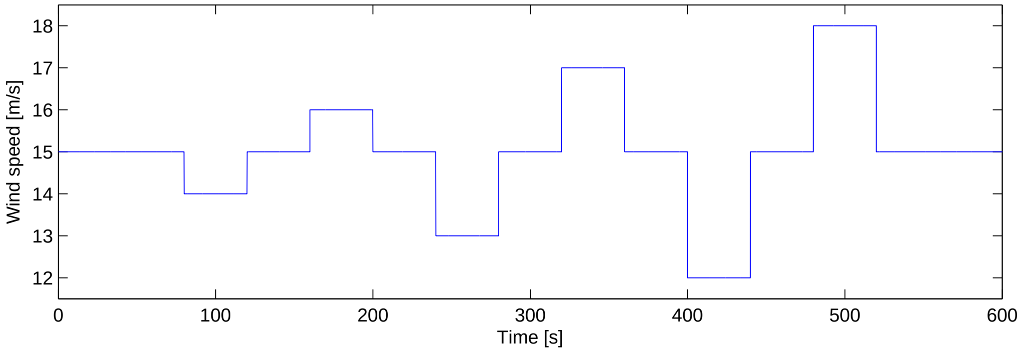


Figure 13.2: Wind input for model validation

As the validation will be conducted above the rated wind speed of 12 m/s, the generator torque τ_G, will be hold constant since only the pitch is used for control above rated wind speed. The pitch controller is described in Appendix E, and the output from the pitch controller is shown in Section 13.5.

13.1 Aerodynamic model

As the submodel, which has the wind field as input, the aerodynamic model will be validated first. This is done as shown in Figure 13.3, by comparing the output from the models to the output from AeroDyn.

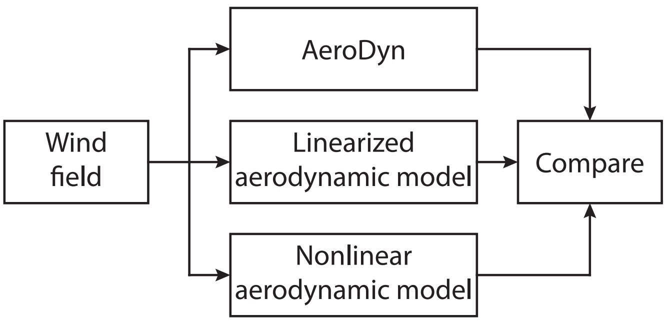
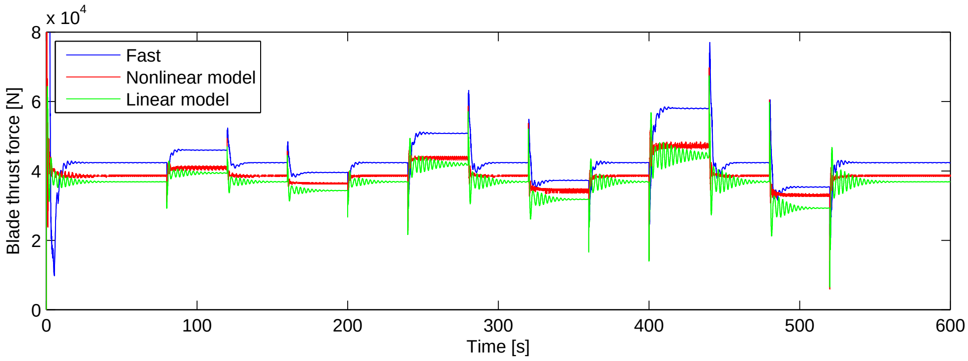


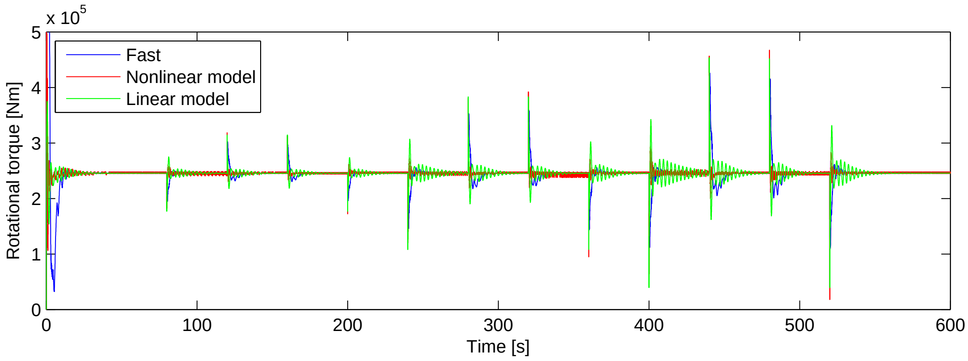
Figure 13.3: Block diagram of the validation of the aerodynamic model

The outputs that will be compared is the thrust force and the rotational torque that is generated by the two regions of one blade.

Simulating the linear and nonlinear model in MATLAB and comparing to the FAST data yields the result shown in Figure 13.4.



(a) Thrust force



(b) Rotational torque

Figure 13.4: Comparison of thrust force and rotational torque from MATLAB model and FAST

As it can be seen in Figure 13.4a, is there an offset in the thrust force, this offset is caused by a difference in the pitch controller which can be seen in Section 13.5, where the pitch controller is described. Apart from this offset, the behavior of the two is somewhat the same for both the nonlinear and linear model.

From Figure 13.4b it can be seen that there is a good fit between the model and FAST, as the level for the three is all alike, and the behavior is the same except for some oscillations in the model outputs.

Both considered it can be concluded that there is a reasonable fit between FAST and the models, it is though needed that the first 50 seconds of the simulation is not considered as both the model and FAST needs some time to settle, this will moreover be the case for the entire validation.

13.2 Structural model

From the aerodynamic model, a thrust force making a deflection of blades and tower is passed to the structural model, this model will now be validated by comparing the deflection of blades and tower, with the output from FAST as illustrated in Figure 13.5.

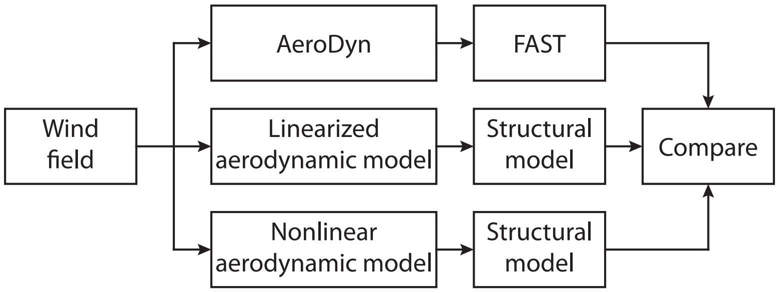
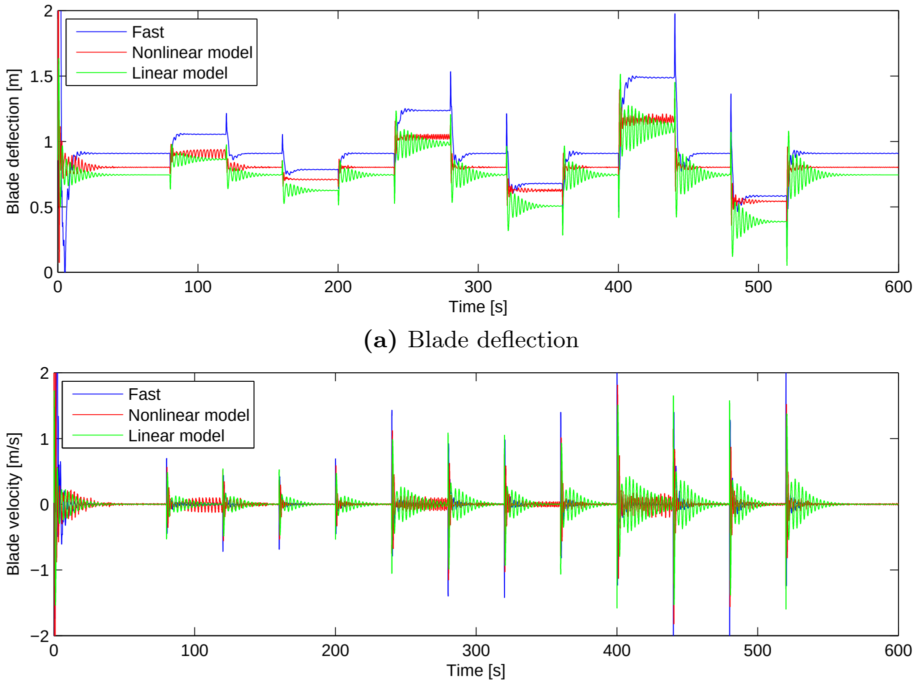


Figure 13.5: Block diagram of the validation of the structural model

Comparing the structural outputs from MATLAB and FAST simulations, shown in Figures 13.6 and 13.7, can it be seen that there, as for the thrust force, is an significant offset for both deflection of tower and blades, this is because the offset from the thrust force is passed on from the aerodynamic model to the structural model.

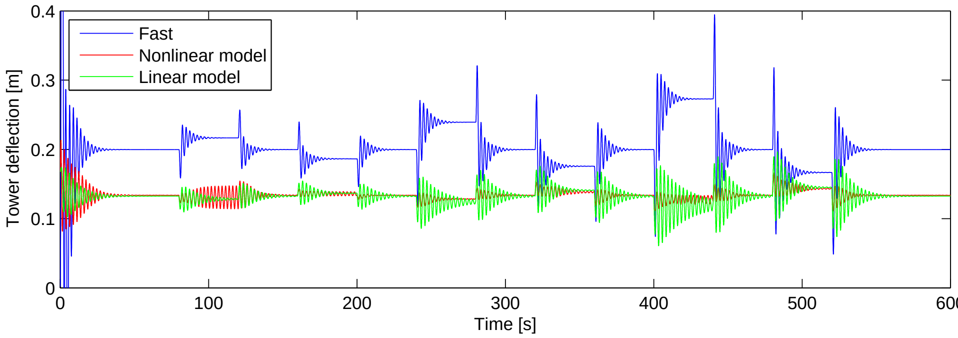


(a) Blade deflection

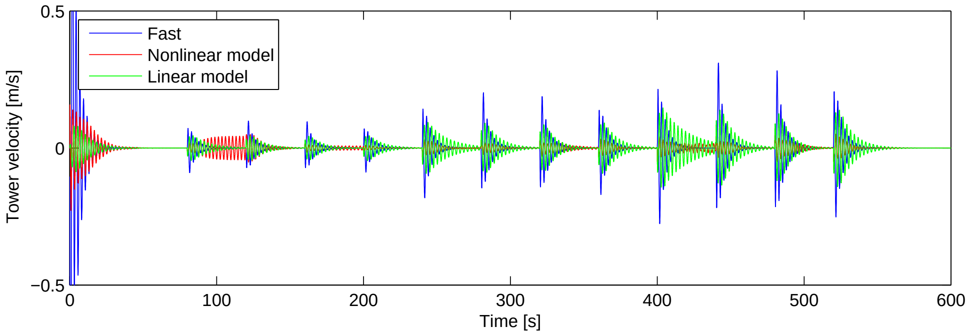
(b) Blade velocity

Figure 13.6: Comparison of blade deflection and velocity from MATLAB model and FAST

In Figure 13.6a, the blade deflection is shown. Disregarding the offset is the behavior almost the same, except for some smaller oscillations in both the nonlinear and linear model. The blade velocity can be seen in Figure 13.6b where the behavior is alike for the three plot, except of some oscillations in the two models.



(a) Tower deflection



(b) Tower velocity

Figure 13.7: Comparison of tower deflection and velocity from MATLAB model and FAST

For the tower the offset is quite more significant, and it is hard to see a behavior that can be linked to the output from FAST; however, there is some deflection of the tower, but it do not vary particularly much.

From the structural model validation it can be seen that a controller probably have to based on the blade deflections, as the tower deflection do not change with the steps in the wind field.

13.3 Mechanical model

As the last part, the mechanical model will be validated. The validation is carried out as shown in Figure 13.8, where a torque is passed on from the two aerodynamic models to the low speed shaft from where it affects the rest of the mechanical system.

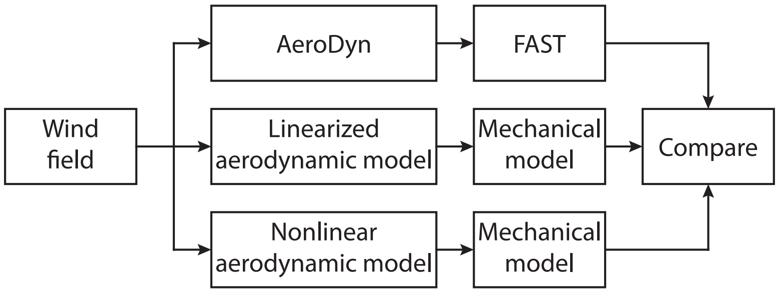
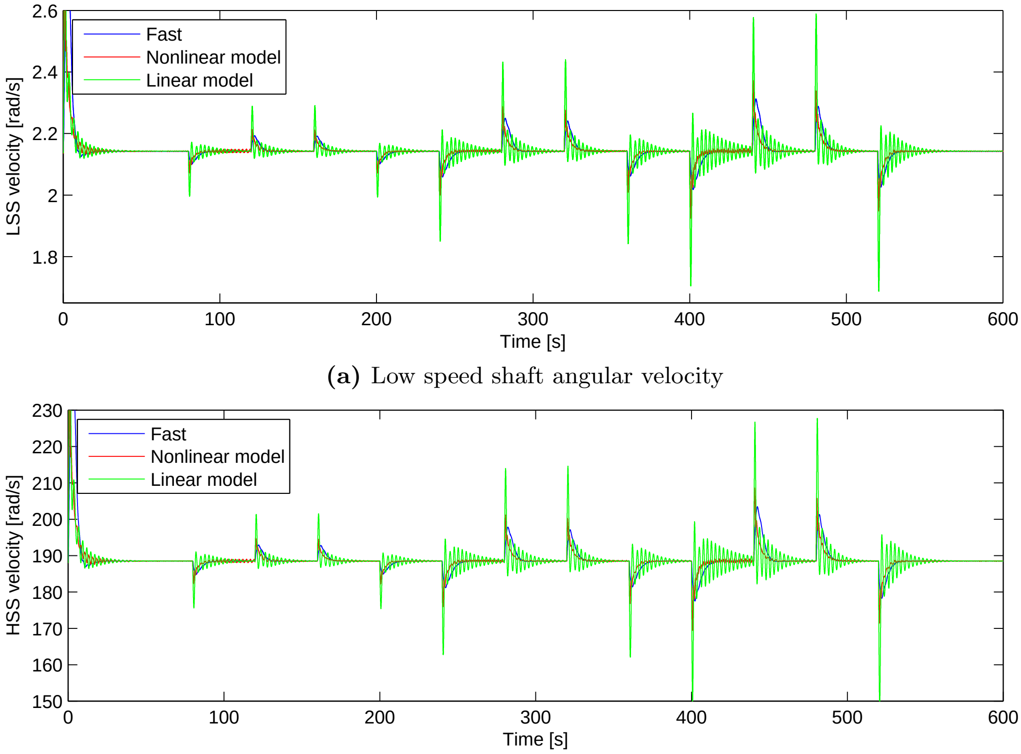


Figure 13.8: Block diagram of the validation of the mechanical model

The result of the comparison between the models and FAST is shown in Figure 13.9.



(a) Low speed shaft angular velocity

(b) High speed shaft angular velocity

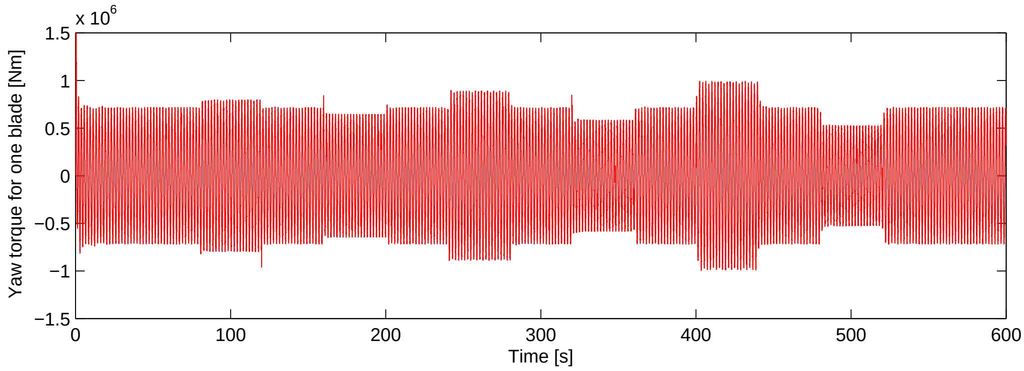
Figure 13.9: Comparison of angular velocities of low and high speed shafts from MATLAB models and FAST

As it can be seen, the behavior of the low and high speed shaft in the nonlinear model are more or less alike the output from FAST, the output from the linearized model is though containing some oscillating. There are small differences in the levels when a deviation from the mean level occurs, but the main behavior is acceptable.

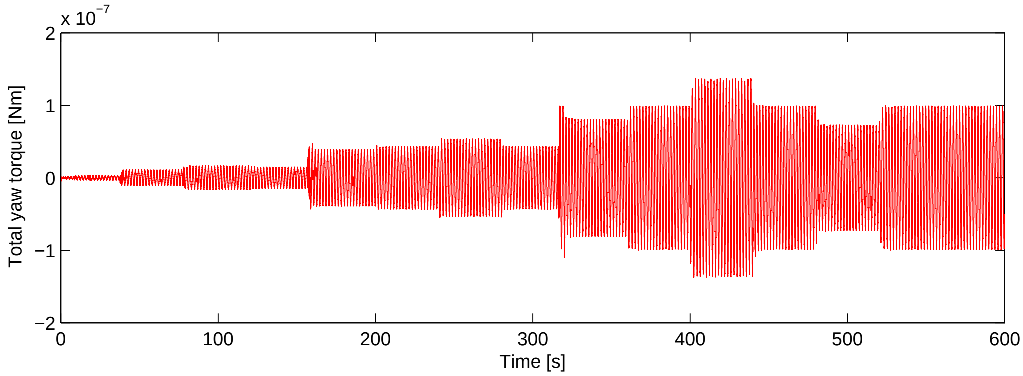
13.4 Yaw torque

The yaw torque found in the simulation will now be plotted even though a validation according to FAST is not possible.

In Figure 13.10, the yaw torque output from the simulation is shown.



(a) Yaw torque contribution from a single blade



(b) Total yaw torque

Figure 13.10: Yaw torque from simulation of nonlinear and linear model in MATLAB

As it can be seen in Figure 13.10a, the contribution for the yaw torque from a single blade is symmetric around zero, oscillating like a sine wave.

In Figure 13.10b, the total yaw torque from the simulations is shown. This yaw torque can be seen as zero, as the maximal value is about $1,25\text{E-}7$ Nm.

13.5 Pitch control in model validation

As previously described, the pitch control in FAST is enabled in the model validation. This pitch control have been implemented in the model, to be able to make a suitable comparison, the pitch controller is the same for the nonlinear and linear model. The pitch control is described in Appendix E.

Simulating the models in MATLAB with the pitch control implemented gives the result shown in Figure 13.11, where the pitch from FAST is shown as well.

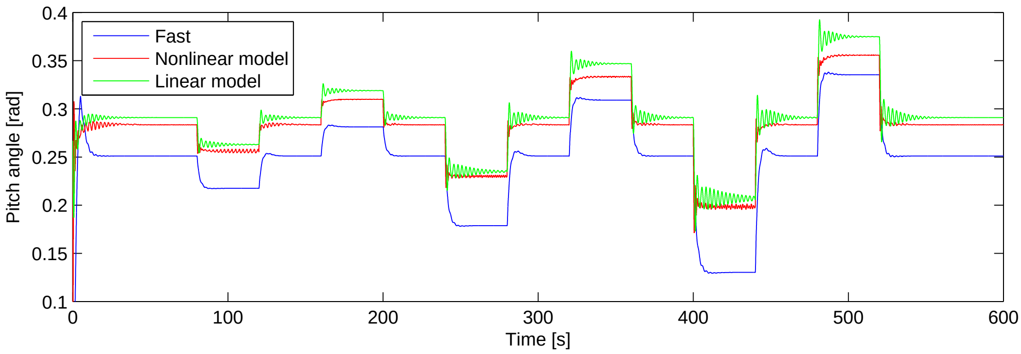


Figure 13.11: Pitch controller output in model validation

As mentioned earlier, there is an offset between the controller in the models and FAST, but the behavior is somewhat the same. The offset in the controller is the main cause in the earlier results in the validation, where offset have occurred in the comparison of the submodels.

13.6 Conclusion of model validation

A validation of the full model have now been conducted, and a conclusion can be made. In all the submodels, the outputs from FAST and the model is somewhat alike.

In parts of both the nonlinear and linear aerodynamic model and the structural model an offset is present, caused by the pitch controller. Apart from this is the outputs the same, with the exception of the tower deflection where a more significant difference is present.

All in all, the derived models are found to be suitable to use for developing a controller in this project, this controller will later be compared with the controller included in FAST.

Part II

Control

Chapter 14

Control introduction

A model of a wind turbine have been found, and this model will now form the basis of the controller design.

In this introduction, the control objective for the project will firstly be described. A standard wind turbine controller will then be described, as the controller for this project will have some of the same properties as a standard controller. After this description, the choice of controller strategy will be substantiated, before the actual controller design is carried out. The implementation in MATLAB will now be described before a simulation will show how the controller performs compared with the standard controller.

Firstly the control objective will be determined.

14.1 Control objective

Control objectives for this project is primarily based on the requirements determined in Chapter 2. These were:

- Lower structural load with main focus on yaw torque
- Less pitch actuation
- Similar power output

compared to the WindPACT 1,5 wind turbine in FAST.

More specifically the control objective can be described as:

- Reduction of yaw torque by 80 % in both fatigue and extreme loads
- Similar energy capture, mean ±0,1 % and a variance of 0,01 %
- Similar or lower pitch actuation, measured mean of the pitch actuators and there derivatives
- The blade and tower deflection must be similar or less

The above control objectives are all in comparison to the FAST controller.

Furthermore, the following constraints be must respected:

- The velocity of the pitch actuator must be below $\pm 10^o/s$
- Blades under deflection may not hit the tower
- Rated power should not be exceeded

A standard wind turbine controller will now be described as this will be used as a baseline for the controller design.

14.2 Standard controller

How the wind turbine normally is controlled can be seen from Figure 14.1.

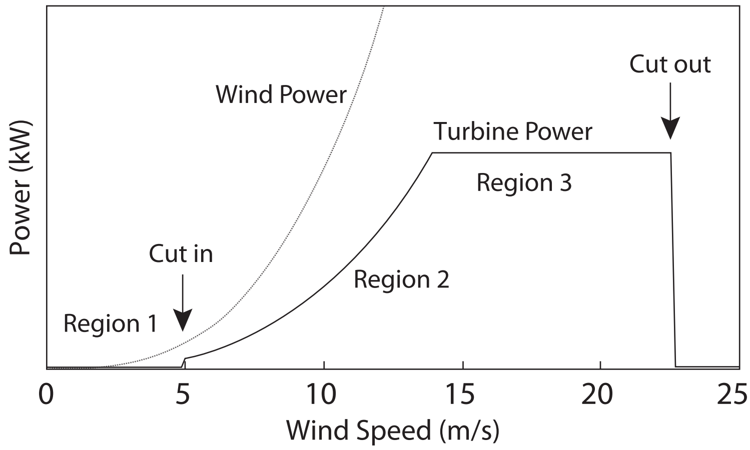


Figure 14.1: Power curves [26, p. 46]

This figure shows the power curves, which are the power generated by the rotor torque and the available power in the wind defined by equations 14.1 and 14.2 respectively [14].

$$P_\tau = \tau_\mathrm{R}\dot{\theta}_\mathrm{R} \tag{14.1}$$

$$P_\mathrm{Wind} = \frac{\rho}{2}AV^3 \tag{14.2}$$

τ_R is the torque applied to the rotor by the wind and $\dot{\theta}_\mathrm{R}$ is the angular velocity of the rotor, ρ is the air density, A is the area that is covered by the rotor and V is the wind speed.

`Cut in` is when the wind stream contains enough power for the wind turbine to generate power, the wind turbine starts without any torque from the generator, when the speed of the rotor is sufficiently large a torque is applied to the generator, subsequently, the turbine starts producing electricity. If the wind speed becomes too high there is a risk of structural damage of the wind turbine, as a consequence, it will be stopped at the `Cut out` point in the right of Figure 14.1.

Control regions

The control of the wind turbine can be divided into three different regions, as shown in Figure 14.1.

Furthest to the left is region 1, here the wind turbine is stopped because the energy in the wind is not powerful enough to induce a rotation of the rotor. In region 2, the objective is to get as much power out of the wind as possible, this is often done by keeping a constant pitch angle, and controlling the generator torque to maximize aerodynamic efficiency. Region 3 is where the wind turbine captures the amount of power it is rated to, here collective pitch control is used. Power is among others limited to avoid structural damage on the wind turbine in high wind speeds [26].

Control of the wind turbine is as described often done by two different control strategies, one for region 2 and one for region 3, but it is in the transition between these controllers some of the largest loads is affecting the wind turbines. A way to overcome this problem is to incorporate an additional controller which switches between region 2 and 3, often called region 2,5 [25]. In [31], the purpose of the region 2,5 controller is to extracted more energy from wind in the transition from region 2 to region 3. If the region 2 controller was used, the speed of the rotor would have to increase too much before going into region 3. The region 2,5 controller secures the rotor reaches rated velocity at the same time as the generator reaches the rated torque.
Region 2,5 controllers is the field of research which is still in development, and the optimal structure of this controller is not yet identified [16].

FAST controller

The starting point for the control design is to use the FAST controller structure, shown in Figure 14.2 and described further in Appendix E, and then improve this.

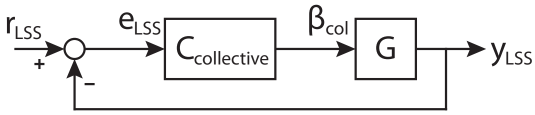


Figure 14.2: FAST pitch controller for region 3

This part of the FAST controller is used in region 3, where the objective is to keep a constant speed, which is achieved by changing the collective pitch.

In the figure, r_{LSS} is the reference for the low speed shaft velocity, the error is denoted by e_{LSS}, β_{col} is the collective control signal for all three pitch actuators and y_{LSS} is the current velocity of the low speed shaft.

The control objective along with a standard controller have been described, and the actual controller design can begin, with a description of the control strategy.

Chapter 15

Control strategy

Before designing the controller, some of the reasons for the choice of controller strategy will shortly be considered, whereafter the actual controller will be designed.

15.1 Repetitive control

The control strategy used will be repetitive control (RC), which can improve a control signal according to periodic disturbances in the system.

In Figure 15.1, the blade deflection for at single blade is plotted, along with a division into periods by dashed lines. A period is defined by one rotation of the azimuth angle. The wind field applied in simulations contains a stationary wake, which can be seen at time instances where the deflection decreases just before every period ends.

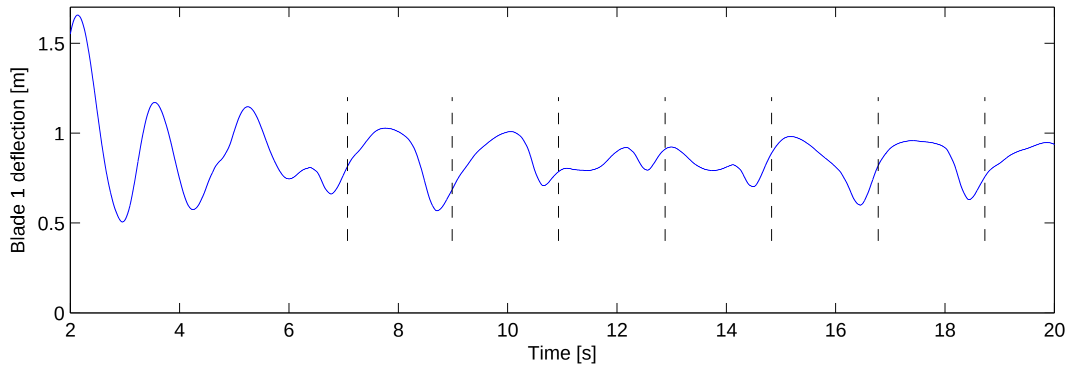


Figure 15.1: Periodic blade deflection

Using either RC or iterative learning control (ILC) could help to reduce the effect from the periodic disturbances a wake would cause, when it is only covering a part of the rotational plane.

The difference between RC and ILC is the initial conditions and time duration, which are the same for every period in ILC whereas it can change in RC, which is the case in the project. Therefore, RC is seen as the most suitable controller.

Chapter 16

Repetitive control

If there are periodic disturbances to the system, these can, as just described, be suppressed by using repetitive control [29]. The repetitive control is integrated by adding another control loop to the system that already contains the collective pitch controller, this is illustrated in Figure 16.1.

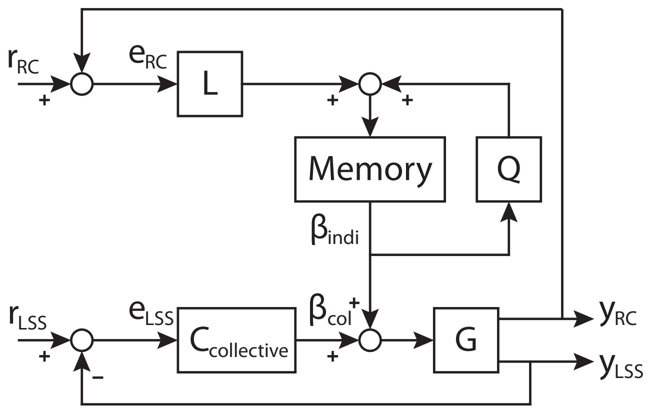


Figure 16.1: Repetitive controller added to the standard controller

where e_{LSS} and e_{RC} are the error in rotor velocity and the error for the repetitive control respectively, and β_{col} and β_{indi} is the collective and individual pitch respectively. The reference for the repetitive control, r_{RC} will be determined in Section 16.1.

The repetitive control will add a control signal, to the collective control signal, which is individual for the each blade, and only considering the yaw torque requirement. The collective pitch controller is still present in the system to ensure stability performance of the system in time domain by keeping the rotor speed at the rated level.

A more detailed exposition of the repetitive controller will know be given.

16.1 Reference

The references for the system is defined by the requirements for the system, described in Section 14.1. The power is the primary reference as this is the original purpose of the wind turbine. In region 3, this is respected by keeping the generator torque at a rated value and control the speed of the low speed shaft by the use of the rotor pitch. This is the same method as the FAST controller. Furthermore, is the pitch actuator used to respect the other requirements, but the use of the pitch actuator should be kept small to reduce wear of the actuators.

Yaw torque

The value for the yaw torque reference is set to zero, as a minimal yaw torque is desired. However, the yaw torque is very difficult to control directly, as it is a sum of the yaw torque from each blade according to equation 16.1.

$$\tau_{\text{yaw}} = \tau_{\text{yaw,z,1}} + \tau_{\text{yaw,z,2}} + \tau_{\text{yaw,z,3}}, \tag{16.1}$$

where $\tau_{\text{yaw,z},k}$ is the yaw torque around the z-axis for the k^{th} blade.

In Figure 16.2, the yaw torque for a single blade and for the entire wind turbine can be seen. It is shown that the wake effect has a small impact on the each blade. However, when the yaw torque is summed, the wake effect is clear, as an offset, which is the yaw torque applied upon the tower around the z-axis.

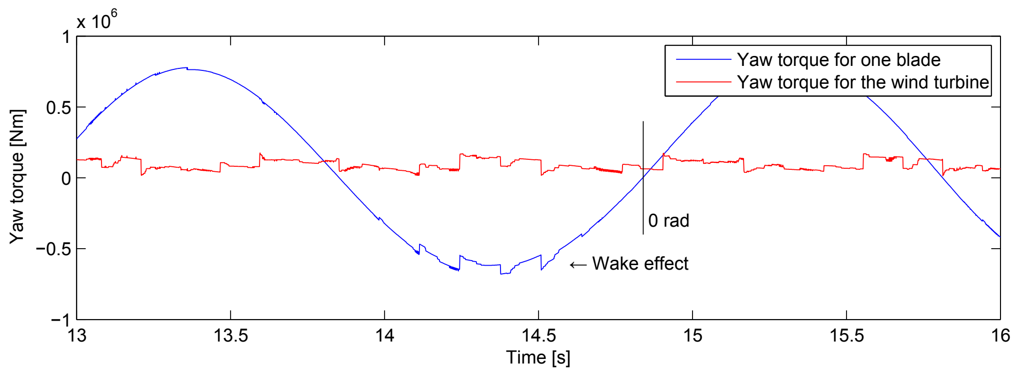


Figure 16.2: Yaw torque for the wind turbine and one blade

There exist two solutions to reduce the τ_{yaw}, which will now be described. The trivial solution, which is not feasible to the problem, as is to apply no forces on the blade and thereby no energy output from the wind turbine. The other solution is to make the sum of the yaw torques zero by reducing the effect of the wake.

The yaw torque of each blade is a result of the thrust force affecting the blade, multiplied by the radius and a sine of the azimuth angle of the blade,

$$\tau_{\mathrm{yaw,z},k} = \sin(\varphi_{\mathrm{k}}) \cdot F_{\mathrm{T,blade},k} \cdot r_{\mathrm{blade}} \tag{16.2}$$

Radius of the blade is a constant and $\sin(\varphi_{\mathrm{k}})$ is can not be changed as a constant rotation is desired. The $F_{\mathrm{T,blade},k} \cdot r_{\mathrm{blade}}$ has been illustrated in Figure 16.3 in two versions, one with a wind field containing wake and one without a wake.

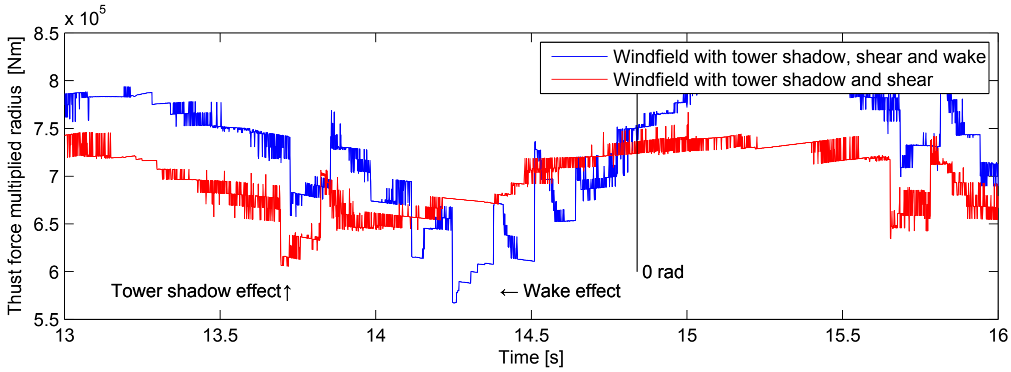


Figure 16.3: Thrust force for a single blade with and without a wake in the wind field

In the figure, the effect from the wake is marked, as well as the tower effect. From this, it can be seen that the thrust force is a more obvious choice for control base.

Only the wake effect is desired reduced as the effect from the tower do not contribute to the yaw torque, and the part of the thrust force not affected by the wake, always will be present when a rotational torque is applied to the wind turbine. The effect from the wake be can emphasised by using the $\sin(\varphi_{\mathrm{k}})$ part of equation 16.2. Furthermore, is it desired to have a reference at zeros as the thrust force offset change dependent on the wind speed, as a higher wind speed will cause a higher pressure on the turbine blades, and a reference can not be set according to that. Both of these are solved by introducing an output filter.

Output filtering

As described, it is not possible to set a constant reference of the thrust force. Therefore, in order to reduce the wake effect, an output filter have been made, which shall ensure the controller only will compute an input for the system based on the wake effect in the thrust force. Furthermore, it will be possible to set the reference to zero, as the wake effect must be reduced as much as possible.

By subtracting the thrust force mean value from the thrust force, the offset will be set to zero. This is followed by a weighting matrix W, which depends on the azimuth angle.

This W matrix shall make the effect from the wake in the signal more clear, as it is desired. In Figure 16.4, it is graphically shown how this filter will work.

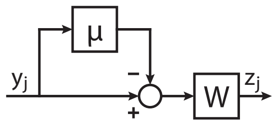


Figure 16.4: Output filter M_{out}

The mathematical realization of this filter is shown in equation 16.3, where the output vector from the system y_j is applied to the filter, which leads to z_j, this is the output that should be reduced to remove the wake effect. The matrix W takes sine of the azimuth angle and weight the output; as only a weight is desired, the sign of sine is removed.

$$M_{\text{out}} = \left(I_{N\text{x}N} - \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}_N \cdot \frac{1}{N} \cdot \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}_N \right) W \tag{16.3}$$

$$W = \begin{bmatrix} |\sin(\frac{0}{N}2\pi)| & 0 & \cdots & 0 \\ 0 & |\sin(\frac{1}{N}2\pi)| & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & |\sin(\frac{N-1}{N}2\pi)| \end{bmatrix} \tag{16.4}$$

$$z_j = M_{\text{out}} \cdot y_j \tag{16.5}$$

N is the length of the period. Because of the weighting matrix W, the y_j vector has to start with the element corresponding to an azimuth angle of zero. Figure 16.5 shows the effect of filters. The simulation parameters are the same as used in Figure 16.3; however, the sample rate has been changed as the filter runs with a different frequency.

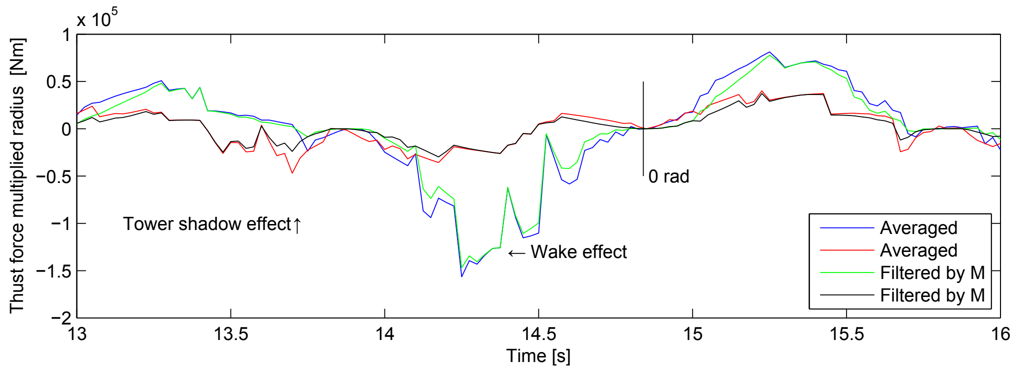


Figure 16.5: The effect of the output filter with and without the W filter. The original signals are in Figure 16.3

The offset has been removed and the tower shadow effect has been reduced, as desired.

16.2 Periodicity

In the repetitive control design, the trial length N has to be defined. There are different opportunities in the length of a period. In this design, it is chosen to be one third of a revolution, because there are three blades. With this choice, the error signal for the repetitive control is updated three times as often, compared with a trial length of a whole revolution.

A period time of one third revolution means that the measured signal from one blade is used for calculating the repetitive control signal for the next. By multiplying the feedforward signals f_j, for the j^{th} period, by a matrix Q, as in equation 16.6, the single signal is passed on to the next blade.

$$f_{j+1} = Qf_j + Le_j \tag{16.6}$$

$$Q = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \tag{16.7}$$

L is a learning filter multiplied with the error signal e_j, which will be described later.

16.3 Lifted repetitive control

By making a lifted system of the closed loop system, it is possible to get the full relation between the system input and states, and the input and the output of the system for a period of N [28].

As an example, can a discrete system be described by taking the lifted input for a full period $\bar{u}_j$ and mapping it to a lifted output $\bar{y}_j$ as in equation 16.8 [5, p. 44], temporarily disregarding internal dynamics.

$$\bar{y}_j = J\bar{u}_j \tag{16.8}$$

where J is a matrix, describing the full system without the initial conditions for the period. This is called a lifted system, because input and output vectors are lifted into column vectors creating a matrix map of the system [5, p. 45][30].

A lifted representation is made, of the discretized original periodically varying system, found in Section 12.5 as

$$x_{k+1} = A_{\mathrm{L}}x_k + B_{\mathrm{L}}u_k \tag{16.9}$$

$$y_k = C_{\mathrm{L}}x_k + D_{\mathrm{L}}u_k \tag{16.10}$$

which is made closed loop with the collective pitch controller K

$$x_{k+1} = A_{\mathrm{cl}}x_k + B_{\mathrm{cl}}u_k \tag{16.11}$$
$$y_k = C_{\mathrm{cl}}x_k + D_{\mathrm{cl}}u_k \tag{16.12}$$

where

$$A_{\mathrm{cl}} = A_{\mathrm{L}} - B_{\mathrm{L}}K \tag{16.13}$$
$$B_{\mathrm{cl}} = B_{\mathrm{L}}, \quad C_{\mathrm{cl}} = C_{\mathrm{L}}, \quad D_{\mathrm{cl}} = D_{\mathrm{L}} \tag{16.14}$$

The lifting results in the equations 16.15 and 16.16 [3, 28].

$$x_{Nj+N} = Fx_{Nj} + G_{\mathrm{p}}\bar{f}_j + G_{\mathrm{w}}\bar{d}_j \tag{16.15}$$
$$\bar{y}_j = Hx_{Nj} + J\bar{f}_j \tag{16.16}$$

where the lifted matrixes is defined as

$$F = A_{\mathrm{cl}}^N, \qquad G_{\mathrm{p}} = \begin{bmatrix} A_{\mathrm{cl}}^{N-1}B_{\mathrm{cl}} & A_{\mathrm{cl}}^{N-2}B_{\mathrm{cl}} & \cdots & A_{\mathrm{cl}}B_{\mathrm{cl}} \end{bmatrix} \tag{16.17}$$

$$G_{\mathrm{w}} = \begin{bmatrix} A_{\mathrm{cl}}^{N-1}B_{\mathrm{cl}} & A_{\mathrm{cl}}^{N-2}B_{\mathrm{cl}} & \cdots & A_{\mathrm{cl}}B_{\mathrm{cl}} \end{bmatrix} \tag{16.18}$$

$$H = \begin{bmatrix} C_{\mathrm{cl}} \\ C_{\mathrm{cl}}A_{\mathrm{cl}} \\ \vdots \\ C_{\mathrm{cl}}A_{\mathrm{cl}}^{N-1} \end{bmatrix}, \qquad J = \begin{bmatrix} D_{\mathrm{cl}} & 0 & \cdots & 0 \\ C_{\mathrm{cl}}B_{\mathrm{cl}} & D_{\mathrm{cl}} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ C_{\mathrm{cl}}A_{\mathrm{cl}}^{N-1}B_{\mathrm{cl}} & C_{\mathrm{cl}}A_{\mathrm{cl}}^{N-2}B_{\mathrm{cl}} & \cdots & D_{\mathrm{cl}} \end{bmatrix} \tag{16.19}$$

It should be noted that matrix B_{cl} in G_{p} is the last three columns of B_{cl} containing the pitch input, and in G_{w} is the first six columns of B_{cl} containing the wind input.

Furthermore, x_{Nj} is the initial conditions of the system states at the beginning of the j^{th} period, defined by the final states in the trial $Nj - N$, $\bar{y}_j$ the measured thrust forces, $\bar{f}_j$ the individual feedforward pitch signal and $\bar{d}_j$ the wind input in the j^{th} period.

Some of the vectors in the lifted system is

$$\bar{y}_j = \begin{bmatrix} \bar{y}_{Nj} \\ \bar{y}_{Nj+1} \\ \vdots \\ \bar{y}_{Nj+N-1} \end{bmatrix}, \qquad \bar{f}_j = \begin{bmatrix} \bar{f}_{Nj} \\ \bar{f}_{Nj+1} \\ \vdots \\ \bar{f}_{Nj+N-1} \end{bmatrix} \tag{16.20}$$

16.4 Full repetitive control system

Collecting the entire repetitive control structure, with the elements just described, yields the illustration in Figure 16.6 [3].

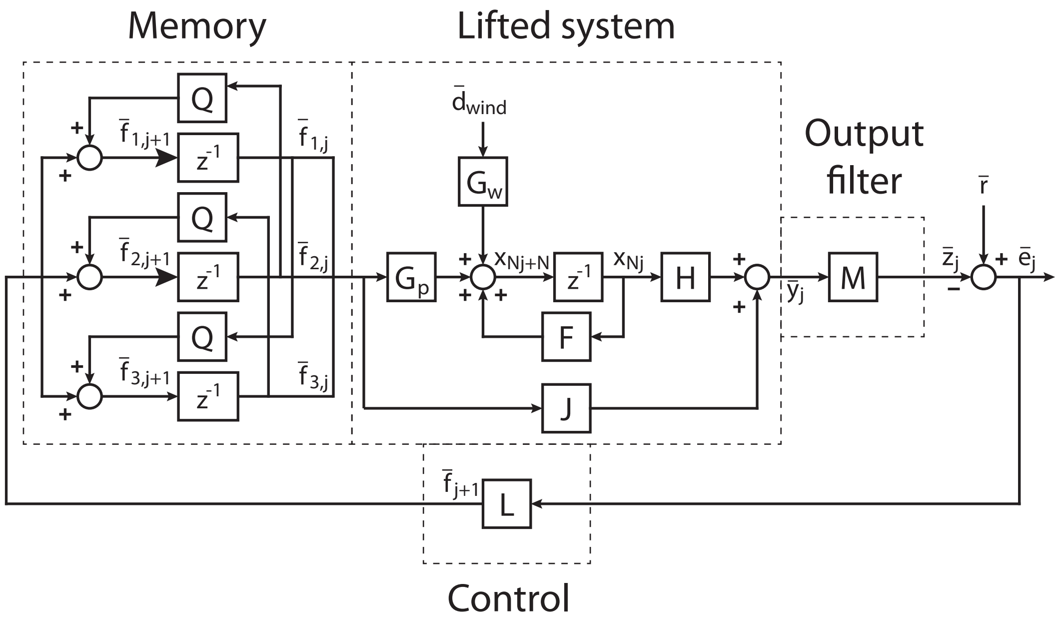


Figure 16.6: Block diagram of the lifted repetitive control system

The found lifting system will now be the base of designing the controller.

16.5 Design of repetitive control using lifted LQR

The repetitive controller design problem will be formulated as an output-feedback problem [28, 13]. This makes it possible to calculate the lifted repetitive controller as a state feedback controller by solving one Riccati equation.

Output-feedback formulation

Only the periodic disturbances shall have influence on the repetitive controller. Equation 16.16 from the lifted system, repeated in equation 16.21 shows the output.

$$\bar{y}_j = Hx_{Nj} + J\bar{f}_j \tag{16.21}$$

and the lifted error can then be defined as [28]:

$$\bar{e}_j = \bar{y}_j - \bar{r}_j \tag{16.22}$$

For this system, the reference is zero, which implies that the error is equal to the output. The update of the error passed on from one trial to the next as shown in equation 16.23.

$$\bar{e}_j = \bar{e}_{j-N} - H\Delta x_{jN} - J\Delta\bar{f}_j \tag{16.23}$$

where the operator Δ defines the difference between two periods as:

$$\begin{align} \Delta\bar{f}_j &= \bar{f}_j - \bar{f}_{j-N} \tag{16.24}\\ \Delta x_{jN} &= x_{jN} - x_{jN-N} \tag{16.25}\\ &= F\Delta x_{jN-N} + G\Delta\bar{u}_j + G_{\mathrm{w}}\bar{d}_j \tag{16.26} \end{align}$$

The output-feedback formulation can now be described as the linear system $\mathcal{S}'_{\mathrm{RC}}$ [28, 13]:

$$\mathcal{S}'_{\mathrm{RC}} \left\{ \begin{array}{rl} \mathcal{X}_{j+N} & = \mathcal{A}'\mathcal{X}_N + \mathcal{B}'\Delta\bar{f}_j + \mathcal{B}'_{\mathrm{w}}\bar{d}_j \\ \bar{e}_j & = \mathcal{C}'\mathcal{X}_N + \mathcal{D}'\Delta\bar{f}_j \end{array} \right. \tag{16.27}$$

where the state vector $\mathcal{X}_N \in \mathbb{R}^{n+N\cdot l}$ for this system is:

$$\mathcal{X}_N = \begin{bmatrix} \Delta x_{jN}^{\mathsf{T}} & \bar{e}_{j-N}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \tag{16.28}$$

with $\Delta x_{jN} \in \mathbb{R}^n$ and $\bar{e}_j \in \mathbb{R}^{Nl}$. The input is defined as $\Delta \bar{f}_j \in \mathbb{R}^{Nm}$.

The matrixes of suitable dimensions are given as:

$$\mathcal{A}' = \begin{bmatrix} F & 0 \\ -H & I_{Nl} \end{bmatrix}, \quad \mathcal{B}' = \begin{bmatrix} G \\ -J \end{bmatrix}, \tag{16.29}$$

$$\mathcal{C}' = \begin{bmatrix} -H & I_{Nl} \end{bmatrix}, \quad \mathcal{D}' = [-J], \tag{16.30}$$

$$\mathcal{B}'_{\mathrm{w}} = \begin{bmatrix} G_{\mathrm{w}} \\ 0 \end{bmatrix} \tag{16.31}$$

An output-feedback formulation for controller design have been set, but before the actual controller design, the observability and controllability conditions will be examined.

Observability

The observability of the system $\mathcal{S}'_{\mathrm{RC}}$ will be determined, using point 4 in Theorem 4.26 from [1, p. 78]:

No right eigenvector v of A is in the right kernel of C: $Av = \lambda v \Rightarrow Cv \neq 0$

which is a condition for observability.

This means if the equations

$$\mathcal{A}'v = \lambda v \tag{16.32}$$

$$\mathcal{C}'v \neq 0 \tag{16.33}$$

is respected, the system is observable. The eigenvectors for the system is $v = \begin{bmatrix} v_1 & v_2 \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^n \times \mathbb{R}^{Nl}$.

Calculating the claim in 16.32 for the system yields:

$$\begin{bmatrix} F & 0 \\ -H & I_{Nl} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} Fv_1 \\ -Hv_1 + v_2 \end{bmatrix} = \lambda \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{16.34}$$

There are to types of eigenvector combinations, the first is $(0, v_2)$, which from equations 16.32 and 16.33 yields:

$$-H \cdot 0 + v_2 = \lambda v_2 \tag{16.35}$$
$$\Rightarrow \lambda = 1 \tag{16.36}$$

$$\begin{bmatrix} -H & I_{Nl} \end{bmatrix} \begin{bmatrix} 0 \\ v_2 \end{bmatrix} = v_2 \neq 0 \tag{16.37}$$

From this it can be seen that the eigenvector $(0, v_2)$ respects the conditions.
Another eigenvector is (v_1, w), inserting this into equation 16.32 yields:

$$\begin{bmatrix} F & 0 \\ -H & I_{Nl} \end{bmatrix} \begin{bmatrix} v_1 \\ w \end{bmatrix} = \begin{bmatrix} Fv_1 \\ -Hv_1 + w \end{bmatrix} = \lambda \begin{bmatrix} v_1 \\ w \end{bmatrix} \tag{16.38}$$

Equation 16.33 yields:

$$\begin{bmatrix} -H & I_{Nl} \end{bmatrix} \begin{bmatrix} v_1 \\ w \end{bmatrix} \neq 0 \tag{16.39}$$

It is assumed that the system is not observable, and thereby:

$$-Hv_1 + w = 0 \tag{16.40}$$
$$\Leftrightarrow$$
$$w = Hv_1 \tag{16.41}$$

As the lifted system in equation 16.15 and 16.16 is observable, the following is valid.

$$Fv_1 = \lambda v_1 \tag{16.42}$$
$$Hv_1 \neq 0 \tag{16.43}$$

hence

$$w = Hv_1 \neq 0 \tag{16.44}$$

From the lower part of equation 16.38 comes

$$Hv_1 + w = w \tag{16.45}$$
$$\Rightarrow w = 0 \tag{16.46}$$

As this violates the assumption of the system being not observable, implies this that the system is observable.

Remark on observability

Even though it has been proven that the system is observable can it be seen from the plot in Figure 16.7, that the observability of the last elements are very small, but not zero.

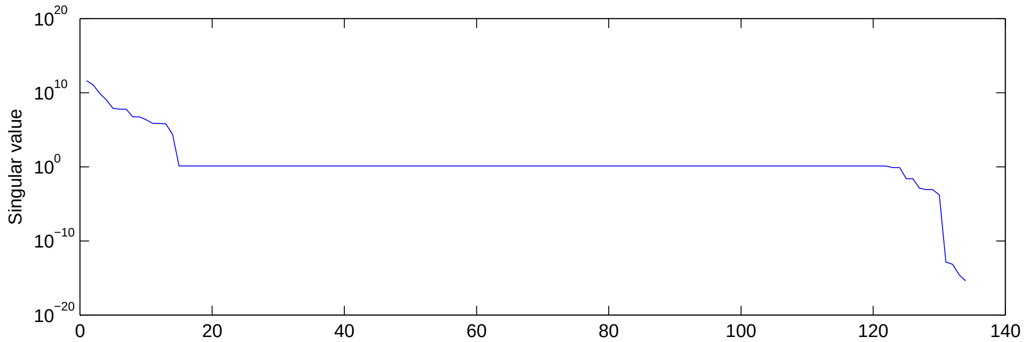


Figure 16.7: Observability of the system $(\mathcal{A}', \mathcal{C}')$, for $N = 40$

The reason for the low observability is the lifted matrixes F and G_p, in equation 16.15, contains a high power of the matrix A_cl from equation 16.10, which means that the eigenvalues of A_cl is raised up to N times. It can therefore be difficult to use these for control. The output from the system can though be measured so this will not have an effect on the controller design.

Controllability

Similar to observability, the controllability of the system $\mathcal{S}'_\mathrm{RC}$ will be examined. This will be done using point 4 in Theorem 4.15 from [1, p. 72]:

No left eigenvector v of A is in the left kernel of B: $v^*A = \lambda v^* \;\Rightarrow\; v^*B \neq 0$

which is a condition for controllability.

This means if the equations

$$v^*\mathcal{A}' = \lambda v^* \tag{16.47}$$
$$v^*\mathcal{B}' \neq 0 \tag{16.48}$$

is respected, the system is observable. The eigenvectors for the system is $v^* = \begin{bmatrix} v_1^* & v_2^* \end{bmatrix} \in \mathbb{R}^n \times \mathbb{R}^{Nl}$.

Calculating the claim in 16.47 for the system yields:

$$\begin{bmatrix} v_1^* & v_2^* \end{bmatrix} \begin{bmatrix} F & 0 \\ -H & I_{Nl} \end{bmatrix} = \begin{bmatrix} v_1^*F - v_2^*H & v_2^* \end{bmatrix} = \lambda \begin{bmatrix} v_1^* & v_2^* \end{bmatrix} \tag{16.49}$$

A possible eigenvector for examining controllability is $v^* = \begin{bmatrix} v_1^* & 0 \end{bmatrix}$, which substituted into equations 16.48 and 16.47 yields:

$$\begin{bmatrix} v_1^* & 0 \end{bmatrix} \begin{bmatrix} F & 0 \\ -H & I_{Nl} \end{bmatrix} = \begin{bmatrix} v_1^*F & 0 \end{bmatrix} = \lambda \begin{bmatrix} v_1^* & 0 \end{bmatrix} \tag{16.50}$$
$$\begin{bmatrix} v_1^* & 0 \end{bmatrix} \begin{bmatrix} G \\ -J \end{bmatrix} = v_1^*G \neq 0 \tag{16.51}$$

which respect the conditions in equation 16.47 and 16.48 as the system in 16.15 (F, G) is controllable.

Another eigenvector for the system is on the form $v^* = \begin{bmatrix} w^* & v_2^* \end{bmatrix}$, inserting this in equation 16.47 and 16.48 yields.

$$\begin{bmatrix} w^* & v_2^* \end{bmatrix} \begin{bmatrix} F & 0 \\ -H & I_{Nl} \end{bmatrix} = \begin{bmatrix} w^*F - v_2^*H & v_2^* \end{bmatrix} = \lambda \begin{bmatrix} w^* & v_2^* \end{bmatrix} \tag{16.52}$$
$$\begin{bmatrix} w^* & v_2^* \end{bmatrix} \begin{bmatrix} G \\ -J \end{bmatrix} = w^*G - v_2^*J \tag{16.53}$$

If it is assumed that the system is uncontrollable the following is valid

$$w^*G - v_2^*J = 0 \tag{16.54}$$
$$\Leftrightarrow$$
$$w^*G = v_2^*J \tag{16.55}$$

From equation 16.52 the following two terms is obtained.

$$w^* F - v_2^* H = \lambda w^* \tag{16.56}$$
$$v_2^* = \lambda v_2^* \Rightarrow \lambda = 1 \tag{16.57}$$

these can be combined to

$$w^*(F - I) - v_2^* H = 0 \tag{16.58}$$

It is assumed that $(F - I_n)$ is invertible, which is allowable as F is stable, and the norm $\|F\|$ is below one [24]. This yields

$$w^* = v_2^* H (F - I_n)^{-1} \tag{16.59}$$

Equation 16.59 can be inserted into equation 16.55:

$$v_2^* H (F - I_n)^{-1} G = v_2^* J \tag{16.60}$$
$$\Leftrightarrow$$
$$v_2^* \left(H(F - I_n)^{-1} G - J \right) = 0 \tag{16.61}$$

If the matrix $H(F - I_n)^{-1}G - J$ has maximal row rank, the system is controllable.

Inspecting the singular values for this system results in the plot in Figure 16.8.

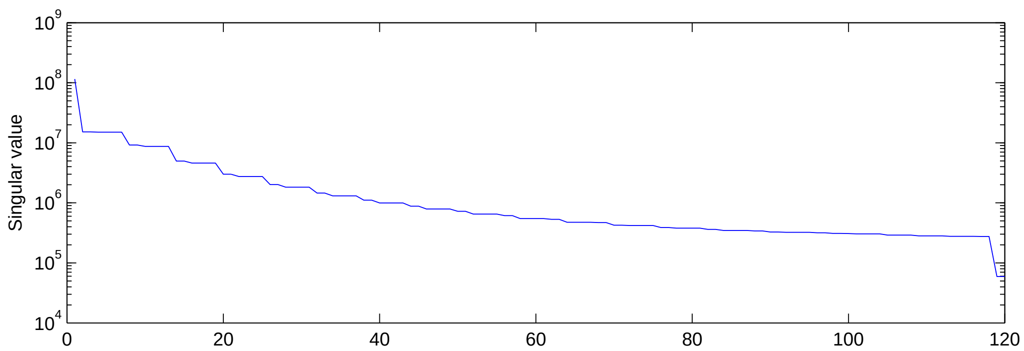


Figure 16.8: Singular value decomposition for the matrix $H(F - I_n)^{-1}G - J$, for $N = 40$

From this is can be seen that the matrix is nonsingular, as non of the singular values are zero. Therefore, is the system $(\mathcal{A}', \mathcal{B}')$ controllable.

A plot of the controllability is shown in Figure 16.9.

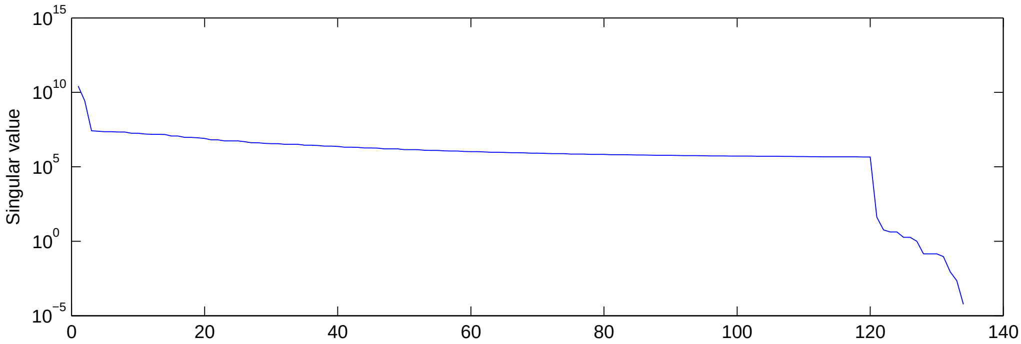


Figure 16.9: Controllability of the system $(\mathcal{A}', \mathcal{B}')$, for $N = 40$

This shows that, as for the observability, that the last values is the plot is very poor, this means that some parts of the system is hardly controllable.

16.6 Model reduction of the output-feedback formulation

It was found, that the system is both observable and controllable, it has also been found that parts of the system only have small observability or controllability. This causes some errors, which can not be observed and states that can not be controlled.

By reducing the model, the states with small controllability or observability can be removed. This is done by performing a model reduction by balanced truncation [1, p. 211]. The procedure is carried out by making a balanced realization, and keeping the states with the highest Hankel singular values [21]. For this purpose, a transformation matrix will be found.

The first part of the balanced model reduction is to find the controllability gramian P and the observability gramian Q, which can be done by solving the following Lyapunov equations

$$\mathcal{A}'P + P\mathcal{A}'^{\mathsf{T}} + \mathcal{B}'\mathcal{B}'^{\mathsf{T}} = 0 \tag{16.62}$$
$$\mathcal{A}'^{\mathsf{T}}Q + Q\mathcal{A}' + \mathcal{C}'^{\mathsf{T}}\mathcal{C}' = 0 \tag{16.63}$$

for example by using the MATLAB function `gram`. The SVD of each of the gramians is found

$$P = U_{\mathrm{p}}\Sigma_{\mathrm{p}}V_{\mathrm{p}}^{\mathsf{T}} \tag{16.64}$$
$$Q = U_{\mathrm{q}}\Sigma_{\mathrm{q}}V_{\mathrm{q}}^{\mathsf{T}} \tag{16.65}$$

The Hankel singular values are now to be found, firstly by taking the square roots of the gramians:

$$L_{\mathrm{p}} = U_{\mathrm{p}}\sqrt{\Sigma_{\mathrm{p}}} \tag{16.66}$$
$$L_{\mathrm{q}} = U_{\mathrm{q}}\sqrt{\Sigma_{\mathrm{q}}} \tag{16.67}$$

From these square roots, the Hankel singular values are calculated by finding the SVD of $L_{\mathrm{q}}^{\mathsf{T}} L_{\mathrm{p}}$:

$$L_{\mathrm{q}}^{\mathsf{T}} L_{\mathrm{p}} = U_{\mathrm{H}}\Sigma_{\mathrm{H}} V_{\mathrm{H}}^{\mathsf{T}} \tag{16.68}$$

Now the transformation matrix T_{red} and its inverse T_{red}^{-1} are found:

$$T_{\mathrm{red}} = L_{\mathrm{q}} U_{\mathrm{H}} \Sigma_{\mathrm{H}}^{\frac{1}{2}} \tag{16.69}$$
$$T_{\mathrm{red}}^{-1} = L_{\mathrm{p}} V_{\mathrm{H}} \Sigma_{\mathrm{H}}^{\frac{1}{2}} \tag{16.70}$$

A control of the transformation matrix can be performed as

$$T_{\mathrm{red}} T_{\mathrm{red}}^{-1} = I \tag{16.71}$$

The transformation matrixes can transform the original system as

$$\begin{bmatrix} \hat{\mathcal{A}}' & \hat{\mathcal{B}}' \\ \hat{\mathcal{C}}' & \hat{\mathcal{D}}' \end{bmatrix} = \begin{bmatrix} T_{\mathrm{red}}^{\mathsf{T}} \mathcal{A}' T_{\mathrm{red}}^{-1} & T_{\mathrm{red}}^{\mathsf{T}} \mathcal{B}' \\ \mathcal{C}' T_{\mathrm{red}}^{-1} & \mathcal{D}' \end{bmatrix} \tag{16.72}$$

The transformed system is now arranged in such a way that the states with the highest Hankel singular value is first, and the ones with a lower value last in the matrixes. With this arrangement it is possible to reduce the system by removing the lower state, as shown in equation 16.73, until the wanted order k is reached.

$$\hat{\mathcal{S}}'_{\mathrm{RC,red}} = \begin{bmatrix} \hat{\mathcal{A}}'(1:k, 1:k) & \hat{\mathcal{B}}'(1:k, :) \\ \hat{\mathcal{C}}'(:, 1:k) & \hat{\mathcal{D}}' \end{bmatrix} \tag{16.73}$$

The transform of the state vector can be done as in equation 16.74, and the other way by 16.75.

$$\hat{\mathcal{X}} = T_{\mathrm{red}} \mathcal{X} \tag{16.74}$$
$$\mathcal{X} = T_{\mathrm{red}}^{-1} \hat{\mathcal{X}} \tag{16.75}$$

Model reduction applied to the output-feedback formulation

The result from making the model reduction of the system, can be seen in Figure 16.10. Here the Hankel singular values of the unreduced system is plotted along with two systems with the order 60 and 100.

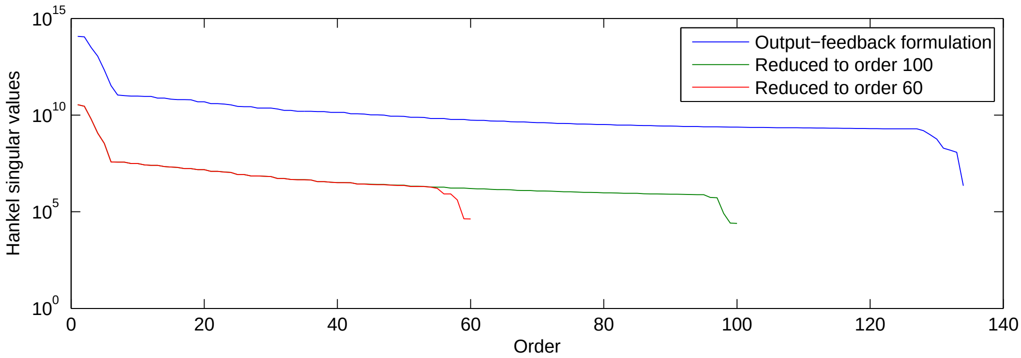


Figure 16.10: Reduction of the output-feedback formulation

As it can be seen, the model reduction not removing the smallest singular values, which where desired.

Looking at Figure 16.11 is the Hankel singular values of the original system, the lifted system and the output-feedback formulation.

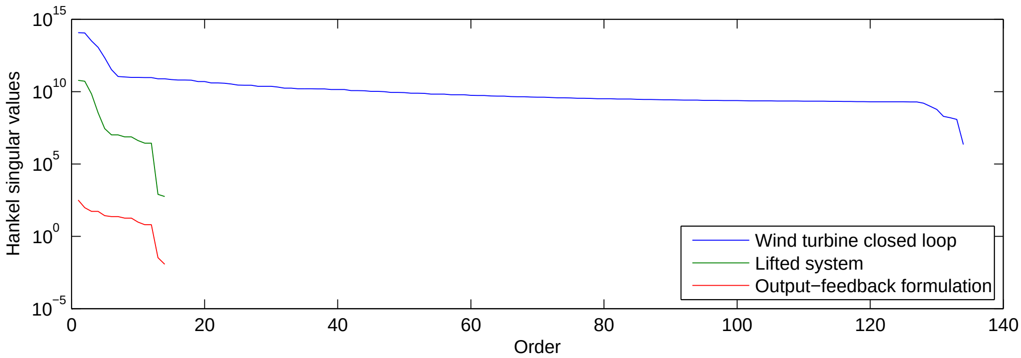


Figure 16.11: Hankel singular values

From this figure, the coherence between the closed loop system, and the lifted system can be seen, as the rightmost part of them are quite similar. Comparing the lifted system with the output-feedback formulation, it can be seen that the leftmost singular values of them have the same shape, to a certain extent is the rightmost part of the two also alike.

Examining the controllability and observability of the reduced systems reveal how the reduction is affecting the system. The controllability singular values is plotted in Figure

16.12. Here it can be seen, that the smallest values are removed when the system is reduced from order 134, which is the original system, to an order of 100. But if the system is reduced further to an order of 60, the lowest singular value is almost the same.

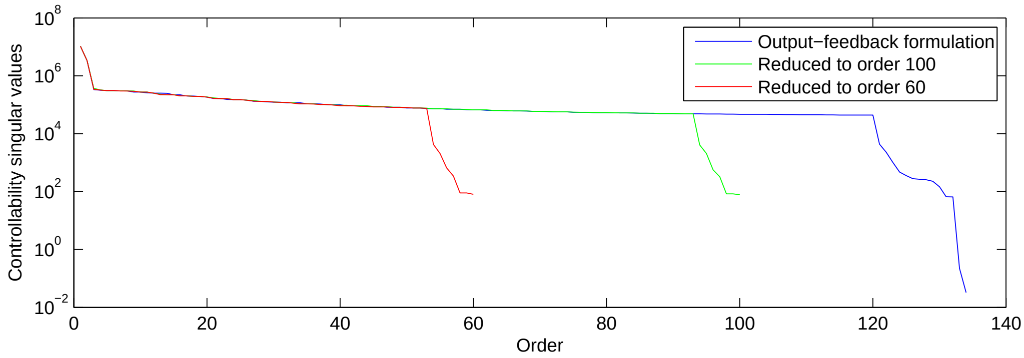


Figure 16.12: Singular values of the controllability matrix

Likewise, is the observability singular values is plotted in Figure 16.13. Here it is shown that the smallest values are removed, when the system is reduced to a order of 100, but reducing it further does not have any effect of the smaller values.

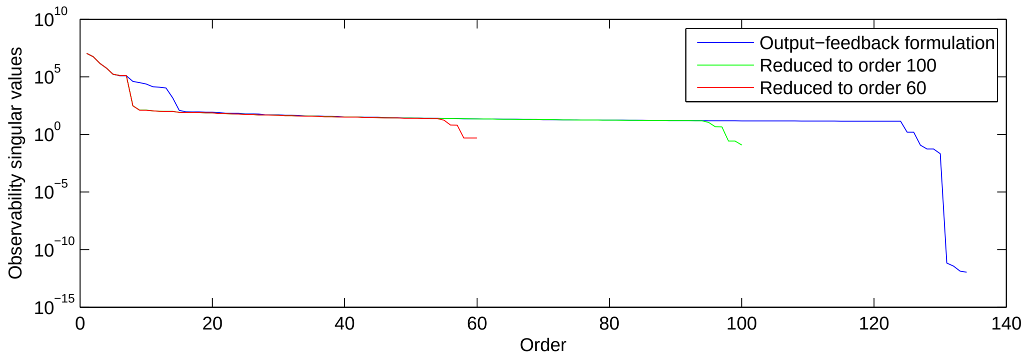


Figure 16.13: Singular values of the observability matrix

From the figures shown in this section, it could be seen that a reduction to a order of 100 would remove the smallest singular values, but reducing the system further will not have any desired effect.

16.7 Lifted LQR controller design

A reduced output-feedback formulation has been made, repeated in equation 16.76.

$$\hat{\mathcal{S}}'_{\mathrm{RC}} \begin{cases} \hat{\mathcal{X}}_{j+N} & = \hat{\mathcal{A}}'\hat{\mathcal{X}}_N + \hat{\mathcal{B}}'\Delta\bar{f}_j \\ \bar{e}_j & = \hat{\mathcal{C}}'\hat{\mathcal{X}}_N + \hat{\mathcal{D}}'\Delta\bar{f}_j \end{cases} \tag{16.76}$$

A controller L is now found, which can create a change in the feedforward signal as in equation 16.77.

$$\Delta\bar{f}_j = -L\hat{\mathcal{X}}_N \tag{16.77}$$

The controller L is found using LQR design.

In LQR design the controller must minimize the quadratic cost function $\mathcal{J}$ in equation 16.78.

$$\mathcal{J}(\Delta\bar{f}_j) = \int_0^\infty \hat{\mathcal{X}}_N^{\mathsf{T}} Q \hat{\mathcal{X}}_N + \Delta\bar{f}_j^{\ \mathsf{T}} R \Delta\bar{f}_j \, dt \tag{16.78}$$

Q and R are quadratic weighting matrixes, with the dimensions of the state vector and input vector, respectively. The weighting matrixes can be chosen to give different states or inputs larger or smaller influence on the controller design and thereby lastly the controlled system.

As the state vector has been truncated by a balanced realization, the states with the best conditions is placed first. It is desired that errors in the yaw torque is reduced, so an identity matrix has been picked as an initial Q matrix; recall that the state vector $\mathcal{X}_N$ contains all yaw torque errors for one period. The R weight penalize the actuator action, initially R has been set as an identity matrix and if the constrains for the actuator is violated, the value of R can be raised to give the pitch actuation a higher penalty.

For computing the L controller, the MATLAB function `lqr` has been used. This function returns a state-feedback gain L, which is used in equation 16.77.
Using `lqr` in MATLAB makes necessary to multiply the identity matrix in $\mathcal{A}'$ by 0,99, as the function otherwise will make an error.

The control design has now been completed, and a implementation in MATLAB will follow. This implementation is described in the next chapter.

Chapter 17

Implementation

In order to test the designed controller against the standard controller from FAST, the system has been implemented in MATLAB.

There are three different sampling time in the implemented system, these are:

- System: 1 ms
- Collective pitch control: 25 ms
- The lifted repetitive control runs three times per rotation: The frequency is the time it takes for a third rotation divided by collective pitch control sample time

17.1 System and collective pitch controller

The wind turbine system runs with a sample time of 1 ms, where all the states and variables are updated, and the collective pitch controller with a sample time of 25 ms, where the pitch angle input to the system is set. These two sample time is taken directly from the FAST code which works with the same times.

17.2 Repetitive controller

The lifted repetitive controller is executed three times for every rotation. However, it samples the errors from each blade every time the collective pitch controller is executed. These error samples are collected into the vector `errorvec`, which is reshaped, to be used in the lifted repetitive controller.

The repetitive controller calculates an output for the next third rotation and saves it in a vector. The individual pitch output is now added to the collective pitch signal every time it is executed.

Remarks

The rotational speed of the rotor has to be kept steady, as the repetitive controller memory has a fixed finite length. This should be maintained by the collective pitch controller, which has the rotor speed as reference. In the implementation the collective controller is designed using a LQR i MATLAB.

Furthermore, has no noise been added to the simulation, as it has been prioritized to focus on the development of repetitive controller. Thereby, all state are sent directly to the controller. By not taking noise in consideration, the result for the simulation will be better than in the real world.

17.3 Test of implementation

Testing the implemented controller in MATLAB, with a wind field containing all the phenomena described in Chapter 4, yields the yaw torque in Figure 17.1, and the pitch input from the controller in Figure 17.2.

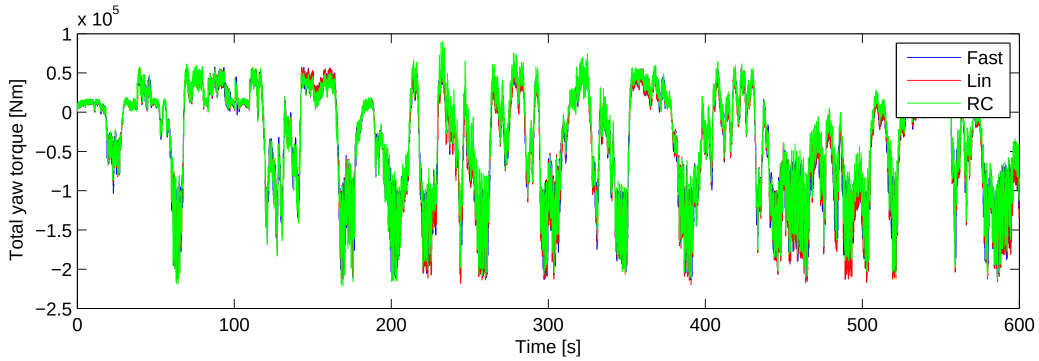


Figure 17.1: Summarized yaw torque from developed RC controller

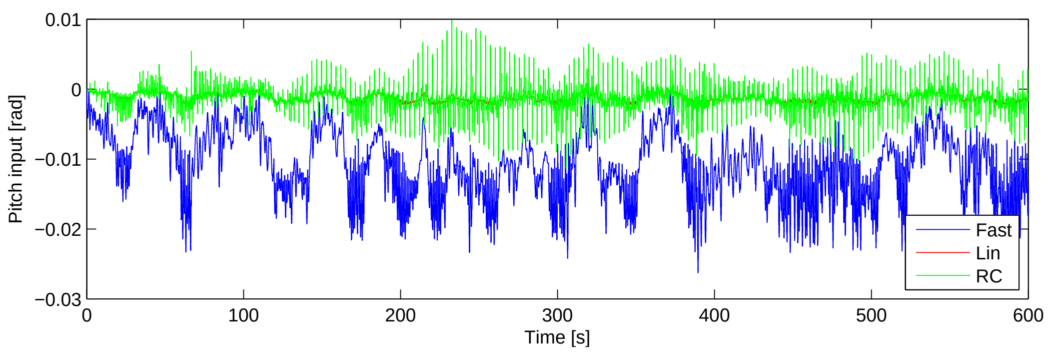


Figure 17.2: Pitch input from developed RC controller

From these two figures, it can be seen that the developed lifted repetitive control behavior is stable in the plots. However, the pitch actuator has a lot of peaks which are undesired. This is even when the R matrix in LQR function has been adjusted to a very high value, to limit the control signal. For these plots R is an identity matrix multiplied by 10E12.

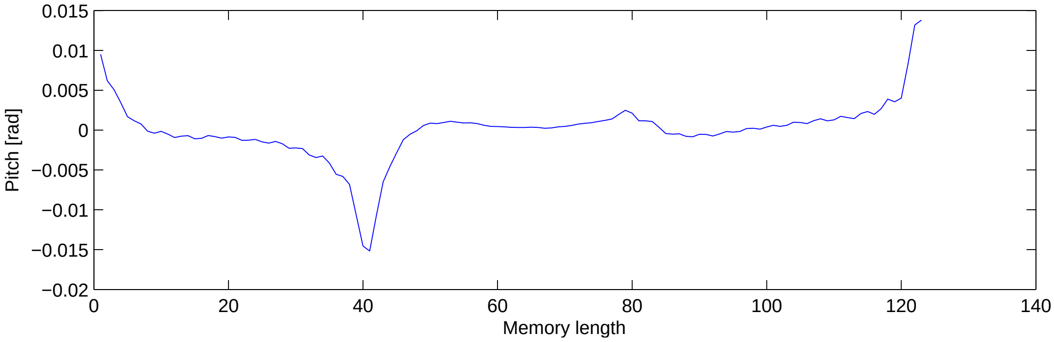


Figure 17.3: A memory vector with the pitch angle, which is added to the collective pitch. The value added is dependent on the blade angle. 1 to 123 represents one rotation, where 1 and 123 is the top of the wind turbine

Figure 17.3 shows the memory vector for one period. There are two peaks in the plot, in the points at 41 and 123. These peaks continues to be in the memory as the new memory is based on the old memory, plus the change in input signal Δu. These peaks have some undesired effect on the simulations and should not be there. As no error occurs, when generating the LQR controller, it should be stable. The error is properly arising because of a difference in the model and the implementation.

A number of different controller variables have been tried for the LQR controller design, and wind fields containing fewer phenomena, but the results from simulations are the same, the system does not behave as desired. To test if the simulation works a simple test controller has been implemented.

Simplified repetitive controller

As the developed controller does not behave as expected, a simplified repetitive controller, with a proportional control gain has been implemented. The control gain was found by trial and error as well as the factor in the robustness filter Q. The applied wind field has a nonmoving wake and no turbulence or tower shadow, in order to see the wake effect more clearly.

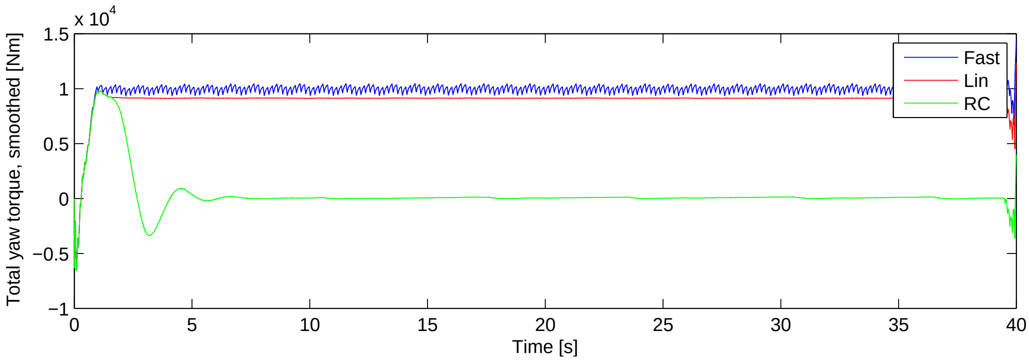


Figure 17.4: Total yaw torque from simplified controller, smoothed with a moving average filter

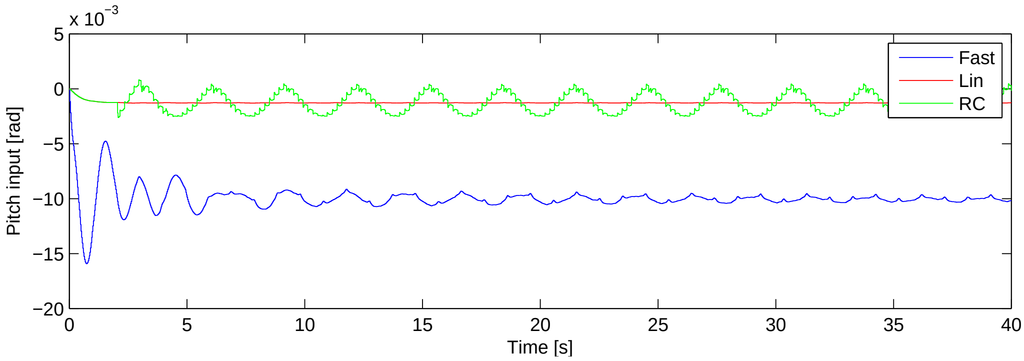


Figure 17.5: Pitch input from simplified controller

Figure 17.4 and 17.5 shows that it is possible to reduce the yaw torque with a pitch input of ±2,5E-3 rad. The small ripples on the control signal is from the discretization of the wind field.

Part III

Evaluation

Chapter 18

Acceptance test

To evaluate the designed repetitive controller an acceptance test will now be carried out. The controller will be evaluated according to the requirements from Section 14.1, where the controller objective were determined. The requirements are:

- Reduction of yaw torque by 80 % in both fatigue and extreme loads
- Similar energy capture, mean ±0,1 % and a variance of 0,01 %
- Similar or lower pitch actuation, measured mean of the pitch actuators and there derivatives
- The blade and tower deflection must be similar or less

all in comparison to the FAST controller.

The acceptance test will be carried out by applying a wind field to the a wind turbine with the developed controller, and the FAST controller, and making a simulation in MATLAB. Data saved from the two controllers can then be compared to see if the requirements are respected. The MATLAB files used for acceptance test is to be found in `'Acceptance test'`

The wind field for the acceptance test will contain all wind phenomena described in Chapter 4, being a mean wind, tower shadow, wind shear, turbulence and a moving Gaussian wake.

18.1 Yaw torque reduction

The first requirement to be evaluated is the yaw torque. The requirement for the yaw torque, was a reduction of 80 %.

In Figure 18.1, a plot of the yaw torque doing the simulation.

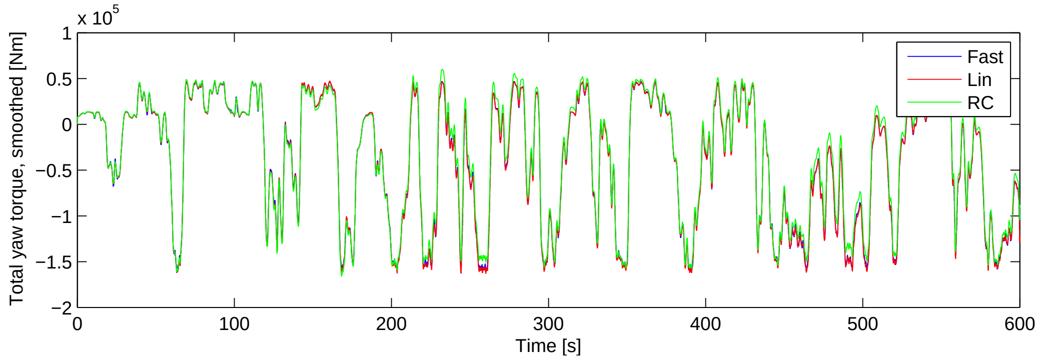


Figure 18.1: Total yaw torque smoothed with a moving average filter

From this it can be seen that the yaw torque have been lowered a bit, as there is a minor difference between the graphs in the plot. But the requirement was a reduction of 80 % which is clearly not respected.

18.2 Energy capture

The energy production of the wind turbine should, according to the requirements, be the rated for the wind turbine, in other words the same as the FAST controller.

The power output in the acceptance test is shown in Figure 18.2.

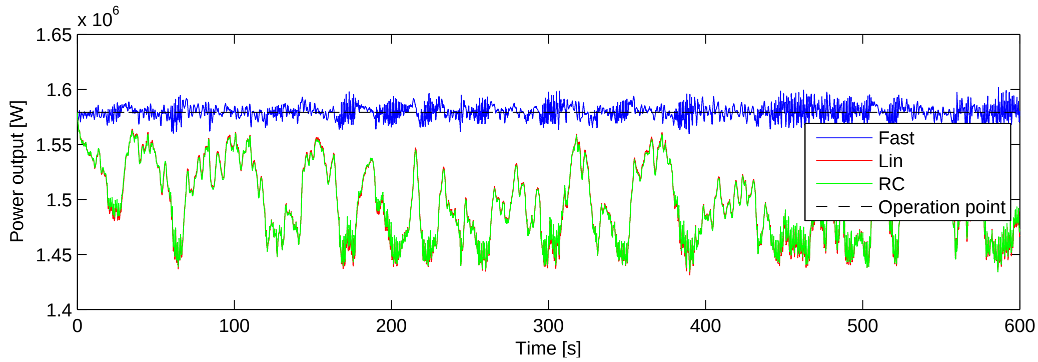


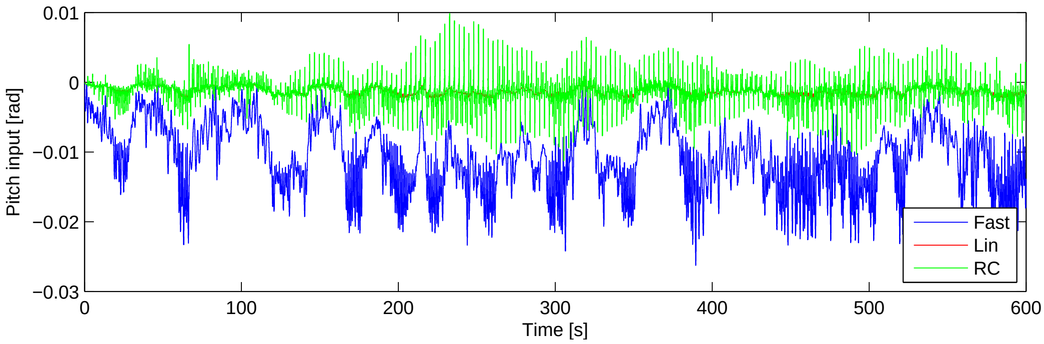
Figure 18.2: Output power

The level of the energy production of the developed controller, is as it can be seen lower than the rated level, marked in the figure. The FAST controller is moving about the rated level as required, which implies that the requirement is not respected.

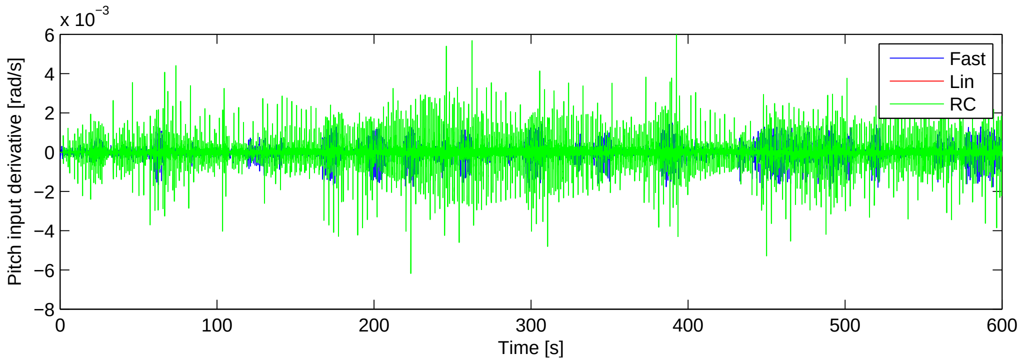
18.3 Pitch actuation

The level of the pitch actuation should be reduced with the developed controller, as a higher level of actuation will cause the actuator to wear out faster.

The pitch signal and its derivative is shown in Figure 18.3.



(a) Pitch input



(b) Pitch input derivation

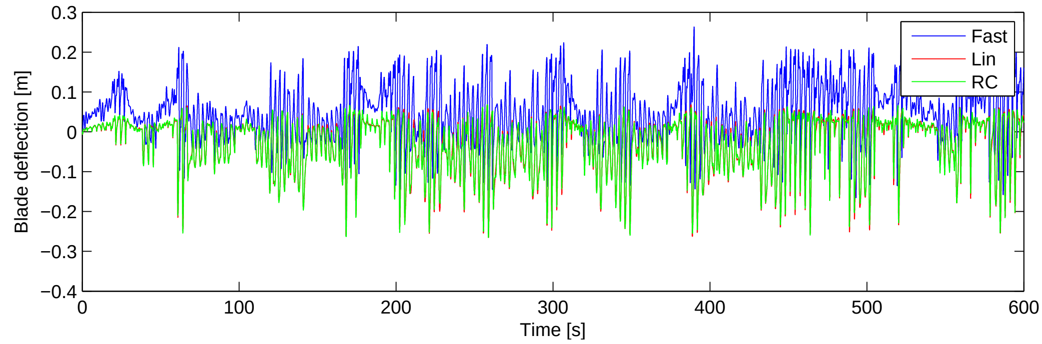
Figure 18.3: Acceptance test of pitch actuation

It can be seen that the level of the pitch input from the repetitive controller is higher than the signal from the FAST controller, which not necessarily says anything of the level of actuation. The actual requirement is about the pitch actuation which can be examined from the derivative of the pitch input, shown in Figure 18.3b. The level of the derivative for the repetitive controller is quite bigger than for the FAST controller, meaning that the pitch actuation is larger as well. Thereby is the requirement not respected.

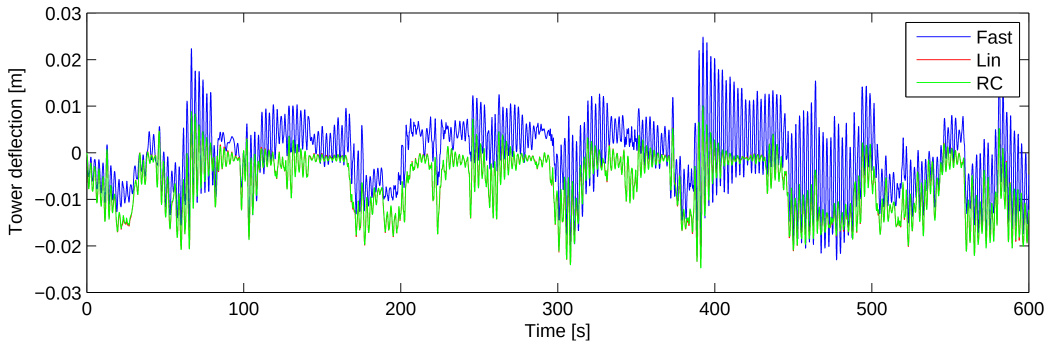
18.4 Blade and tower deflection

The last requirement will now be evaluated. This is that the deflection of both blade and tower should not be more than for the FAST controller.

The result from the simulation is shown in Figure 18.4.



(a) Blade deflection



(b) Tower deflection

Figure 18.4: Acceptance test of blade and tower deflection

In Figure 18.4a, it can be seen that the deflection of the blade is somewhat the same, the deflection is though displaced a little down in the plot.

Deflection of the tower is shown in Figure 18.4b, where it can be seen that the deflection of the tower is lowered by the repetitive, as the FAST controller is oscillating quite more.

The requirement of a similar or lower deflection of blades and tower is thus respected.

More figures from the acceptance test is to be found in Appendix F. These were considered to have no importance for the acceptance test.

After the acceptance test a conclusion on the controller development will now be given.

Chapter 19

Conclusion

In the project a wind field and the Baseline wind turbine have successfully been modeled. The wind model includes a mean wind, tower shadow, shear effect, turbulence and a Gaussian formed wake. In the aerodynamic model, the blade element theory was used to describe the effect of the wind on the blades as thrust force and rotation torque. The deflection of the blades and tower has been modeled by introducing a hinge in each blade and in the tower, and express them as mass-spring-damper systems. The drive train and pitch actuator has been modeled as a two linear state space systems. In order to make use of the linear control the model has been linearized to one single state space model. The linear and nonlinear model were compared to the FAST wind turbine and the model was found suitable to control.

In order to control the yaw torque of the wind turbine the reference has been investigated. It was found that, by adding an output filter to the thrust force output, a reference was generated which could be used to reduce the yaw torque.

As the wake affecting the wind turbine has been assumed to be quasi static, the response of the blades are assumed to be similar when they pass through the wake. Thereby the response from the blades can be see as repetitive, and repetitive control can be applied.

In order to use repetitive control on the multi variable state space model, the system has been lifted. In the lifted domain the output of one whole rotation is taken in consideration, which makes it possible to use a LQR controller which generates an output that minimizes the output for a rotation.

The found controller was implemented in a MATLAB simulator based on the linear model. The response from the controller was not as expected, as the yaw torque was not minimized. In order to test the simulator, an simple proportional controller was tested successfully, showing the potential in the method. Thereby the error in the controller can possibly be a difference between the model and the simulator, which was not found.

In the acceptance test one goal was to reduce the yaw torque by 80 %, this was not achieved as the repetitive controller was not implemented in a functional way. Due to this two other requirements, the level of power production, and pitch actuation, were not respected either. But even though the controller did not work as intended the deflection of the tower was lowered to some degree.

From Chapter 2, the following problem statement was defined for the project:

> *Can a controller, using individual pitch control, be developed, to reduce the yaw torque under the presence of wake, which is better than the one for the Baseline wind turbine in the FAST code?*

It could be possible, can the conclusion be, as the simple implementation showed the potential. But in this project, the developed repetitive controller, was not implemented properly and thereby it can not be concluded, that it is not possible.

Chapter 20

Discussion

In the project, the designed lifted repetitive control has been implemented in a Matlab simulator. However, the benefit of the repetitive controller was not obtained, but some of the properties were shown. From this it can be seen that more work, could lead to a repetitive controller, which is able to reduce the yaw torque effecting the wind turbine significantly.

The project is based on the AeroDyn and FAST wind turbine model and controller. The FAST controller is a PID, however in this project a state feedback controller of the collective pitch controller were used instead. The collective pitch controller did not keep the rotational speed as steady or at the correct level as it was require to get the use of memory vectors to work properly. This could be seen from the acceptance test, where the power output was less smooth than the FAST controller, and the level of the power output was significantly lower as well. The collective pitch controller ought to be stable such that the power output is stable, and an integration term could pull the output up to the desired level.

In a simple simulator test, it was shown that it is possible to control the yaw torque. But the solution did not show any robustness and there were no noise on the input to the controller, so further work would be needed to make the a better controller performance.

In the simulations no noise was added, as it has been prioritized to get the repetitive controller implemented before this was added. However, when the controller is implemented and working properly, the next step would be to add noise to the output of the model. In order to use noisy measurements for control purpose, a state estimator should be added.

Bibliography

[1] A.C. Antoulas. *Approximation of Large-Scale Dynamical Systems.* Advances in Design And Control. Society for Industrial and Applied Mathematics, 2005. ISBN 9780898715293.

[2] F.D. Bianchi, H.D. Battista, and R.J. Mantz. *Wind turbine control systems: Principles, Modelling and Gain Scheduling Design.* Advances in industrial control. Springer, 2007. ISBN 1-84628-492-9.

[3] Okko H. Bosgra and Maarten Steinbuch. *Lifted approach to ILC-Repetitive Control.* TUD Delft Centre for Systems and Control, 2003.
Filename on DVD: "Lifted approach to ILC-Repetitive Control.pdf".

[4] Tony Burton, David Sharpe, Nick Jenkins, and Ervin Bossanyi. *Wind Energy: Handbook.* J. Wiley, 2001. ISBN 0-471-48997-2.

[5] B.G. Dijkstra. *Iterative Learning Control, with applications to a wafer stage.* Delft University of Technology, January 2003.
Filename on DVD: "Iterative Learning Control, with applications to a wafer stage.pdf".

[6] D.S.L. Dolan and P.W. Lehn. Simulation model of wind turbine 3p torque oscillations due to wind shear and tower shadow. *Energy Conversion, IEEE Transactions on*, 21(3):717–724, sept. 2006. ISSN 0885-8969. doi: 10.1109/TEC.2006.874211.
Filename on DVD: "Simulation Model of Wind Turbine 3p Torque.pdf".

[7] Enova. *Renewable energy.* `www.renewableenergy.no`, 2011.
Filename on DVD: "Renewable energy.pdf".

[8] Thomas Esbensen, Brian Thorarins Jensen, Michael Odgaard Niss, and Christoffer Sloth. *Joint Power and Speed Control of Wind Turbines.* Aalborg University, 2008.

[9] Institut für Flugzeugbau. *Operation of wind turbines.* Universität Stuttgart, `http://www.ifb.uni-stuttgart.de/en/forschung/windenergie/betriebsverhalten`, 2012.

[10] Johannes Friis, Ebbe Nielsen, and Jeppe Bønding. *Repetitive Individual Pitch Model Predictive Control for Horizontal Axis Wind Turbine.* Aalborg University, 2010.

[11] Global Wind Energy Council. *Global wind report - Annual Market update 2011*. GWEC, 2011.
Filename on DVD: "GWEC - Annual report 2011.pdf".

[12] E. Hau. *Wind turbines: fundamentals, technologies, application, economics*. Springer, 2006. ISBN 3540242406.

[13] Ivo Houtzager. *Towards Data-Driven Control for Modern Wind Turbines*. Technische Universiteit Delft, 2011.
Filename on DVD: "Houtzager.pdf".

[14] Kathryn E. Johnson, Lucy Y. Pao, Mark J. Balas, and Lee J. Fingersh. Control of variable-speed wind turbines: standard and adaptive techniques for maximizing energy capture. *Control Systems, IEEE*, 26(3):70–81, June 2006. ISSN 1066-033X. doi: 10.1109/MCS.2006.1636311.
Filename on DVD: "Control of variable-speed wind turbines.pdf".

[15] Jason M. Jonkman and Marshall L. Buhl Jr. *FAST User's Guide*. National Renewable Energy Laboratory, 3.01 edition, August 2005.
Filename on DVD: "FAST userguide.pdf".

[16] J. Laks, L. Pao, and A. Alleyne. Comparison of wind turbine operating transitions through the use of iterative learning control. *American Control Conference (ACC), 2011*, pages 4312–4319, July 2011. ISSN 0743-1619.
Filename on DVD: "Comparison of Wind Turbine Operating Transitions Through the Use of Iterative Learning Control.pdf".

[17] Gunner C. Larsen, Helge Aa. Madsen, Kenneth Thomsen, and Torben J. Larsen. Wake meandering: a pragmatic approach. *Wind Energy*, 11(4):377–395, 2008. ISSN 1099-1824. doi: 10.1002/we.267. URL `http://dx.doi.org/10.1002/we.267`.

[18] D.J. Malcolm and A.C. Hansen. *WindPACT Turbine Rotor Design Study*. National Renewable Energy Laboratory, April 2006.
Filename on DVD: "WindPACT Turbine Rotor design study.pdf".

[19] Andreas Manjock. *Design Codes FAST and ADAMS for Load Calculations of Onshore Wind Turbines*. Germanischer Lloyd WindEnergie GmbH, May 2005.
Filename on DVD: "Design Codes FAST and ADAMS for Load Calculations of Onshore Wind Turbines.pdf".

[20] J.F. Manwell, J.G. McGowan, and A.L. Rogers. *Wind Energy Explained: Theory, Design and Application*. Wiley, 2009. ISBN 978-0-47001-500-1.

[21] *balancmr - Balanced model truncation via square root method*. The MathWorks, Inc., `http://www.mathworks.se/help/toolbox/robust/ref/balancmr.html`, 2012.
Filename on DVD: "Balanced model truncation via square root method.pdf".

[22] Timothy J. McCoy. *XCEL RDF Project CWO2 - Advanced Methods for Development of Wind Turbine Models for Control Design. Final Report.* Xcel Energy, 2003. Filename on DVD: "XCEL RDF Project CWO2 .pdf".

[23] National Wind Technology Center. *FAST*. National Renewable Energy Laboratory, http://wind.nrel.gov/designcodes/simulators/fast/, 2010.

[24] Encyclopedia of Mathematics. *Neumann Series*. http://www.encyclopediaofmath.org/index.php/Neumann_series, 2012. Filename on DVD: "Neumann Series.pdf".

[25] Lucy Y. Pao and Kathryn E. Johnson. *A tutorial on the dynamics and control of wind turbines and wind farms*. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5160195, June 2009. Filename on DVD: "A tutorial on the dynamics and control of wind turbines and wind farms.pdf".

[26] Lucy Y. Pao and Kathryn E. Johnson. Control of wind turbines. *Control Systems, IEEE*, 31(2):44–62, April 2011. ISSN 1066-033X. doi: 10.1109/MCS.2010.939962. Filename on DVD: "Control of wind turbines.pdf".

[27] Shafiqur Rehman and Naif M. Al-Abbadi. Wind shear coefficients and their effect on energy production. *Energy Conversion and Management*, 46(15-16):2578–2591, 2005. ISSN 0196-8904. doi: 10.1016/j.enconman.2004.12.005. URL http://www.sciencedirect.com/science/article/pii/S0196890405000142.

[28] Justin K. Rice and Michel Verhaegen. A structured matrix approach to efficient calculation of lqg repetitive learning controllers in the lifted setting. *International Journal of Control*, 83(6):1265–1276, 2010. doi: 10.1080/00207171003682671. URL http://www.tandfonline.com/doi/abs/10.1080/00207171003682671.

[29] Professor Maarten Steinbuch, Dr. Tom Oomen, and M.Sc. Joost Bolder. *PhD course: Repetitive and Iterative Learning Control*. Eindhoven University of Technology, AAU, 11.-13. April 2012. Folder on DVD: "ILC & RC course ".

[30] R. Tousain, E. van der Meche, and O. Bosgra. Design strategy for iterative learning control based on optimal control. In *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, volume 5, pages 4463–4468, 2001. doi: 10.1109/.2001.980905. Filename on DVD: "Design strategy for Iterative Learning Control based on Optimal control.pdf".

[31] A. D. Wright and L. J. Fingersh. *Advanced Control Design for Wind Turbines Part I : Control Design , Implementation , and Initial Tests*. National Renewable Energy Laboratory, http://www.osti.gov/bridge/product.biblio.jsp?osti_id=927269, March 2008.

Part IV

Appendix

Appendix A

How to run FAST

In this appendix, it will be described how the FAST simulator works, and furthermore the procedure to run FAST in MATLAB Simulink and in the command prompt will be described. The files used in this chapter is placed in `['FAST files'\'How to run FAST\']` on the attached DVD.

An overview of how the different processors is interacting in FAST is shown in Figure A.1.

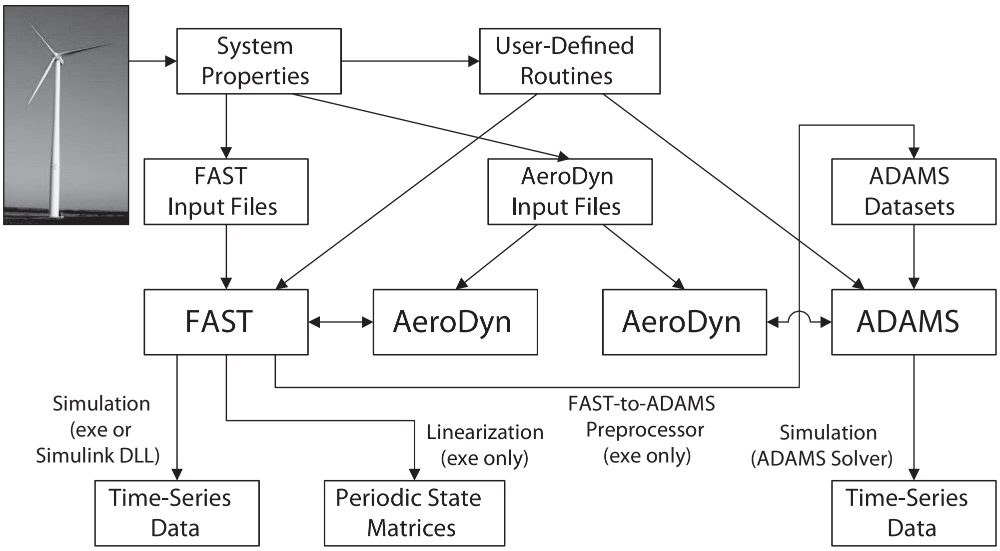


Figure A.1: Overview of the FAST processors [15]

As it can be seen, the FAST simulator consists of a number of processors which defines the properties of the simulated wind turbine.

A.1 Matlab

To run a FAST simulation, the Matlab file `Run_Self_Made.m` is opened, the content of this file is:

Listing A.1: Run_Self_Made.m

```
input_fast ='Self_Made.fst'
Read_FAST_Input
sim('OpenLoop')
```

The primary FAST file is `Self_Made.fst` in this file all the parameters for the wind turbine is defines e.g. rotor diameter and hub height, furthermore the simulation time is defined and files describing the properties of the wind turbine is called. It is also in this file the outputs of the simulation is defined, a list of the possible outputs is present in the FAST User's Guide [15]. It is important that the file structure as shown in the attached folder is kept, as Matlab otherwise will not be able to run and will report an error.

Also the AeroDyn file `Self_Made_AD.ipt` is called, in this input file the aerodynamics of the wind turbine is defined by calling files containing properties of i.e. the blades. From this file another file `Wind\Wind_file.wnd` describing the incoming wind flow is called.

When the input to FAST is defined the function `Read_FAST_Input.m` is called. This script reads the FAST input file, here `Self_Made.fst`, and from this creates Matlab workspace variables which are used FAST.

Finally, the Simulink model `OpenLoop.mdl` is executed, this could also be done using the Simulink interface. `OpenLoop.mdl` contains a FAST S-Function which runs in open loop with the inputs defined.

The output of FAST simulation in Matlab is saved in the file `Self_Made_SFunc.out`, from where it can be imported and analysed further in Matlab.

A.2 Command prompt

Running FAST in Matlab Simulink excludes some parameters from the simulation, this could for example be the initial conditions for the blade deflection. By running FAST in command prompt more opportunities is present. This will now be described.

Listing A.2: Run FAST in command prompt

```
C:\Users\...\FAST>fast Self_Made.fst
```

As for Matlab, the primary FAST file is `Self_Made.fst`, and furthermore is AeroDyn file `Self_Made_AD.ipt` and wind file `Wind\Wind_file.wnd` determining the parameters for the simulation. In the folder where simulation is ran the file `FAST.exe` shall be located, this file is the FAST program, which have been downloaded from [23].

The output of FAST simulation in command prompt is saved in the file `Self_Made.out`, from where it can be imported and analysed further in Matlab.

When an output from FAST is analysed it should be noticed that the first 50 seconds, dependent of the individual simulation, should be disregarded, because FAST needs some time to settle when starting a simulation, causing the first part of the simulation to be invalid.

Appendix B

Baseline wind turbine

The model and controller in this project are based on the upwind turbine included in the FAST test files. This turbine is a WindPACT 1,5 wind turbine, which is a 1,5 MW 3-bladed upwind baseline turbine. A full description of the wind turbine can be found in [18] and [19]. The relevant data used for simulations is listed in Table B.1.

Variable	**Value**
Rated power	1,5 MW
Rotor diameter	70,0 m
Hub height	84,0 m
Rotor speed range	10,0-20,0 rpm
Rated rotor speed	20,0 rpm
Blade mass	3912 kg
Hub mass without blades	15148 kg
Distance rotor to center tower axis	3,3 m
Nacelle mass without hub and blades	51170 kg
Gear ratio	87,97
Distance rotor plane to main bearing	0,99 m
Cut-in wind speed	4,0 m/s
Rated wind speed	12,0 m/s
Cut-out wind speed	25,0 m/s
Blade pitch range	0-90°

Table B.1: Principal data of WindPACT 1,5 MW wind turbine [19, p. 6]

Appendix C

Sensor model

A number of sensors are places on the wind turbine to measure some of the variables included in the model. In this appendix, these sensors will be listed. It is not necessary to model the dynamics of the sensors since it is assumed that these dynamics are significantly faster than the dynamics of the wind turbine.

In Table C.1, the used sensors are listed, and the symbols of the variables they are measuring are furthermore shown.

Measured variable	**Sensor type**	**Symbol**
Generator angular velocity	Speed encoder	$\dot{\theta}_{\mathrm{G}}$
Rotor angular velocity	Speed encoder	$\dot{\theta}_{\mathrm{R}}$
Generator torque	Soft sensor	τ_{G}
Pitch angle	Encoder	β_i
Tower acceleration	Accelerometer	$\ddot{x}_{\mathrm{t}}$
Root moment	Strain gage	$\tau_{\mathrm{b},i}$
Wind speed	Anemometer	V

Table C.1: Sensors from the wind turbine [10]

A list of measured variables have here been presented. These can be used in the overall model to be compared with the states in the full state space model.

Appendix D

Matrixes from linearization

In Chapter 12, the nonlinear parts of the derived model is found. A linearization of the model have been accomplished and combined with the linear parts. The full linear system description is shown in the matrixes found in the following pages.

$$A = \begin{bmatrix}
1,0 & 9,9\mathrm{E}\text{-}4 & -3,2\mathrm{E}\text{-}11 & -5,8\mathrm{E}\text{-}12 & -3,2\mathrm{E}\text{-}11 & -5,8\mathrm{E}\text{-}12 & 3,2\mathrm{E}\text{-}5 & -5,3\mathrm{E}\text{-}6 & -8,2\mathrm{E}\text{-}6 & 0 & 0 & -3,4\mathrm{E}\text{-}4 & -4,4\mathrm{E}\text{-}10 & -4,4\mathrm{E}\text{-}10 \\
-0,064 & 0,99 & -9,6\mathrm{E}\text{-}8 & -1,8\mathrm{E}\text{-}8 & -9,6\mathrm{E}\text{-}8 & -1,8\mathrm{E}\text{-}8 & 0,064 & -0,011 & -0,016 & 0 & 0 & -0,68 & -1,3\mathrm{E}\text{-}6 & -1,3\mathrm{E}\text{-}6 \\
-3,2\mathrm{E}\text{-}11 & -5,8\mathrm{E}\text{-}12 & 1,0 & 9,9\mathrm{E}\text{-}4 & -3,2\mathrm{E}\text{-}11 & -5,8\mathrm{E}\text{-}12 & 3,2\mathrm{E}\text{-}5 & -5,3\mathrm{E}\text{-}6 & -8,2\mathrm{E}\text{-}6 & 0 & 0 & -4,4\mathrm{E}\text{-}10 & -3,4\mathrm{E}\text{-}4 & -4,4\mathrm{E}\text{-}10 \\
-9,6\mathrm{E}\text{-}8 & -1,8\mathrm{E}\text{-}8 & -0,064 & 0,99 & -9,6\mathrm{E}\text{-}8 & -1,8\mathrm{E}\text{-}8 & 0,064 & -0,011 & -0,016 & 0 & 0 & -1,3\mathrm{E}\text{-}6 & -0,68 & -1,3\mathrm{E}\text{-}6 \\
-3,2\mathrm{E}\text{-}11 & -5,8\mathrm{E}\text{-}12 & -3,2\mathrm{E}\text{-}11 & -5,8\mathrm{E}\text{-}12 & 1,0 & 9,9\mathrm{E}\text{-}4 & 3,2\mathrm{E}\text{-}5 & -5,3\mathrm{E}\text{-}6 & -8,2\mathrm{E}\text{-}6 & 0 & 0 & -4,4\mathrm{E}\text{-}10 & -4,4\mathrm{E}\text{-}10 & -3,4\mathrm{E}\text{-}4 \\
-9,6\mathrm{E}\text{-}8 & -1,8\mathrm{E}\text{-}8 & -9,6\mathrm{E}\text{-}8 & -1,8\mathrm{E}\text{-}8 & -0,064 & 0,99 & 0,064 & -0,011 & -0,016 & 0 & 0 & -1,3\mathrm{E}\text{-}6 & -1,3\mathrm{E}\text{-}6 & -0,68 \\
-1,8\mathrm{E}\text{-}7 & -3,2\mathrm{E}\text{-}8 & -1,8\mathrm{E}\text{-}7 & -3,2\mathrm{E}\text{-}8 & -1,8\mathrm{E}\text{-}7 & -3,2\mathrm{E}\text{-}8 & 1,0 & 1,0\mathrm{E}\text{-}3 & -1,3\mathrm{E}\text{-}7 & 0 & 0 & -2,5\mathrm{E}\text{-}6 & -2,5\mathrm{E}\text{-}6 & -2,5\mathrm{E}\text{-}6 \\
-3,6\mathrm{E}\text{-}4 & -6,5\mathrm{E}\text{-}5 & -3,6\mathrm{E}\text{-}4 & -6,5\mathrm{E}\text{-}5 & -3,6\mathrm{E}\text{-}4 & -6,5\mathrm{E}\text{-}5 & -5,5\mathrm{E}\text{-}3 & 1,0 & -2,5\mathrm{E}\text{-}4 & 0 & 0 & -5,0\mathrm{E}\text{-}3 & -5,0\mathrm{E}\text{-}3 & -5,0\mathrm{E}\text{-}3 \\
-1,1\mathrm{E}\text{-}6 & -6,4\mathrm{E}\text{-}4 & -1,1\mathrm{E}\text{-}6 & -6,4\mathrm{E}\text{-}4 & -1,1\mathrm{E}\text{-}6 & -6,4\mathrm{E}\text{-}4 & 0 & -3,4\mathrm{E}\text{-}3 & 0,8 & 2,1\mathrm{E}\text{-}3 & -82,0 & -0,036 & -0,036 & -0,036 \\
-1,6\mathrm{E}\text{-}6 & -9,8\mathrm{E}\text{-}4 & -1,6\mathrm{E}\text{-}6 & -9,8\mathrm{E}\text{-}4 & -1,6\mathrm{E}\text{-}6 & -9,8\mathrm{E}\text{-}4 & 0 & -5,2\mathrm{E}\text{-}3 & 2,9 & 0,97 & 1,3\mathrm{E}3 & -0,056 & -0,056 & -0,056 \\
-5,4\mathrm{E}\text{-}10 & -3,3\mathrm{E}\text{-}7 & -5,4\mathrm{E}\text{-}10 & -3,3\mathrm{E}\text{-}7 & -5,4\mathrm{E}\text{-}10 & -3,3\mathrm{E}\text{-}7 & 0 & -1,7\mathrm{E}\text{-}6 & 8,9\mathrm{E}\text{-}4 & -1,0\mathrm{E}\text{-}5 & 0,95 & -1,9\mathrm{E}\text{-}5 & -1,9\mathrm{E}\text{-}5 & -1,9\mathrm{E}\text{-}5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1,0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1,0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1,0
\end{bmatrix}$$

$$B = \begin{bmatrix}
3,899\mathrm{E}\text{-}12 & 5,572\mathrm{E}\text{-}6 & 3,899\mathrm{E}\text{-}12 & 5,533\mathrm{E}\text{-}12 & 3,899\mathrm{E}\text{-}12 & 5,533\mathrm{E}\text{-}12 & 0 & 0 & 0 & 0 \\
1,181\mathrm{E}\text{-}8 & 0,01114 & 1,181\mathrm{E}\text{-}8 & 1,676\mathrm{E}\text{-}8 & 1,181\mathrm{E}\text{-}8 & 1,676\mathrm{E}\text{-}8 & 0 & 0 & 0 & 0 \\
3,899\mathrm{E}\text{-}12 & 5,533\mathrm{E}\text{-}12 & 3,899\mathrm{E}\text{-}12 & 5,572\mathrm{E}\text{-}6 & 3,899\mathrm{E}\text{-}12 & 5,533\mathrm{E}\text{-}12 & 0 & 0 & 0 & 0 \\
1,181\mathrm{E}\text{-}8 & 1,676\mathrm{E}\text{-}8 & 1,181\mathrm{E}\text{-}8 & 0,01114 & 1,181\mathrm{E}\text{-}8 & 1,676\mathrm{E}\text{-}8 & 0 & 0 & 0 & 0 \\
3,899\mathrm{E}\text{-}12 & 5,533\mathrm{E}\text{-}12 & 3,899\mathrm{E}\text{-}12 & 5,533\mathrm{E}\text{-}12 & 3,899\mathrm{E}\text{-}12 & 5,572\mathrm{E}\text{-}6 & 0 & 0 & 0 & 0 \\
1,181\mathrm{E}\text{-}8 & 1,676\mathrm{E}\text{-}8 & 1,181\mathrm{E}\text{-}8 & 1,676\mathrm{E}\text{-}8 & 1,181\mathrm{E}\text{-}8 & 0,01114 & 0 & 0 & 0 & 0 \\
2,184\mathrm{E}\text{-}8 & 3,099\mathrm{E}\text{-}8 & 2,184\mathrm{E}\text{-}8 & 3,099\mathrm{E}\text{-}8 & 2,184\mathrm{E}\text{-}8 & 3,099\mathrm{E}\text{-}8 & 0 & 0 & 0 & 0 \\
4,368\mathrm{E}\text{-}5 & 6,197\mathrm{E}\text{-}5 & 4,368\mathrm{E}\text{-}5 & 6,197\mathrm{E}\text{-}5 & 4,368\mathrm{E}\text{-}5 & 6,197\mathrm{E}\text{-}5 & 0 & 0 & 0 & 0 \\
0,0004934 & 0,0006356 & 0,0004934 & 0,0006356 & 0,0004934 & 0,0006356 & -2,276\mathrm{E}\text{-}8 & 0 & 0 & 0 \\
0,0007572 & 0,0009756 & 0,0007572 & 0,0009756 & 0,0007572 & 0,0009756 & -2,182\mathrm{E}\text{-}5 & 0 & 0 & 0 \\
2,524\mathrm{E}\text{-}7 & 3,252\mathrm{E}\text{-}7 & 2,524\mathrm{E}\text{-}7 & 3,252\mathrm{E}\text{-}7 & 2,524\mathrm{E}\text{-}7 & 3,252\mathrm{E}\text{-}7 & 1,173\mathrm{E}\text{-}10 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,004988 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,004988 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,004988
\end{bmatrix}$$

$$C = I_{14}$$

$$D = 0_{14,10}$$

$$B1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3189{,}0 & 556{,}8 & 0 & 0 & -87522{,}0 & 0 & 0 \\ -99{,}52 & -4526{,}0 & 0 & 0 & 0 & 0 & 0 & -4526{,}0 & -6667{,}0 & 0 & 0 & -2{,}742\text{E}5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3189{,}0 & 556{,}8 & 0 & 0 & 0 & -87522{,}0 & 0 \\ 0 & 0 & -99{,}52 & -4526{,}0 & 0 & 0 & 0 & -4526{,}0 & -6667{,}0 & 0 & 0 & 0 & -2{,}742\text{E}5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3189{,}0 & 556{,}8 & 0 & 0 & 0 & 0 & -87522{,}0 \\ 0 & 0 & 0 & 0 & -99{,}52 & -4526{,}0 & 0 & -4526{,}0 & -6667{,}0 & 0 & 0 & 0 & 0 & -2{,}742\text{E}5 \\ -71{,}68 & -42866{,}0 & -71{,}68 & -42866{,}0 & -71{,}68 & -42866{,}0 & 0 & -2{,}284\text{E}5 & -6{,}367\text{E}5 & 0 & 0 & -2{,}451\text{E}6 & -2{,}451\text{E}6 & -2{,}451\text{E}6 \end{bmatrix}$$

$$B2 = \begin{bmatrix} 3189{,}0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4526{,}0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3189{,}0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4526{,}0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3189{,}0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4526{,}0 \\ 33277{,}0 & 42866{,}0 & 33277{,}0 & 42866{,}0 & 33277{,}0 & 42866{,}0 \end{bmatrix}$$

$$opv = \begin{bmatrix} -2546{,}0 & 42133{,}0 & -2546{,}0 & 42133{,}0 & -2546{,}0 & 42133{,}0 & 8{,}15\text{E}5 & 0 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$$

Appendix E

FAST reference controller

FAST has an integrated controller, which is used under the validation of the model and as a reference controller when evaluating the controller. In this project, the focus is primarily on region 3; therefore, the controller for this region will be of main focus, the controller for region 2 and 2,5 will though be briefly described.

E.1 Region 2 and 2,5

As it is described in Section 14.2, only the generator torque is used for control in region 2, and the pitch is kept constant. In FAST, the region 2 and 2,5 control is made according to Figure E.1 where the different input parameters is defined in the primary FAST file [15, p. 26].

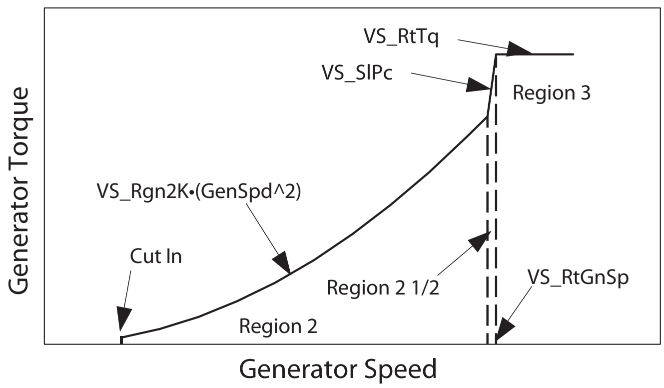


Figure E.1: Torque/speed curve for simple variable-speed control [15, p. 26]

E.2 Region 3

In region 3, the wind turbine has reached its rated power and the controller has to ensure that this power is not exceeded. Secondary the controller should try to limit the load on the blades and tower, in the FAST controller this option can though be switched off.

The rated power is obtained by keeping the generator torque at a constant level and adjusting the pitch angle in order to keep rotor speed at the rated level. The FAST controller is written in FORTRAN code and can be found in the FAST files: `Source\PitchCntrl_ACH.f90` and `CertTest\Pitch.ipt`.

In Figure E.2, a block diagram of the FAST pitch controller in region 3 controller is shown.

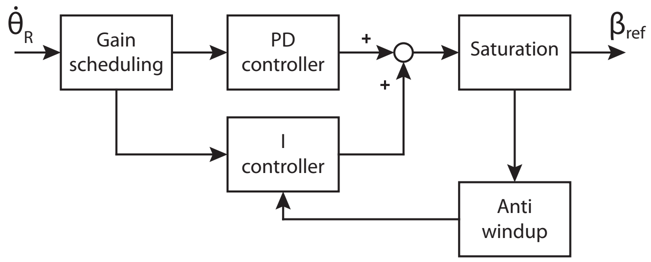


Figure E.2: Block diagram of the controller from FAST in region 3

The velocity of the low speed shaft $\dot{\theta}_R$ acts as an input for the controller, a gain scheduling is applied to this signal, which is sent through a proportional-derivative controller and an integral controller, containing anti-windup. These two is summarized and saturated, and used as pitch reference β_{ref}.

The FAST controller is used for estimating parameters in Chapter 10, to validate the model in Chapter 13, and in the end used in the acceptance test in Chapter 18 for comparison with the developed controller.

The FAST controller have been translated into MATLAB code, which makes it possible to run both controllers in MATLAB, and analyze the data here. The translated FAST controller is included in `'FAST controller'\main.m` in a MATLAB script.

Appendix F

Acceptance test appendix

In the appendix additional figure from the acceptance test is included. These had no significance for the actual acceptance test, but can be seen here.

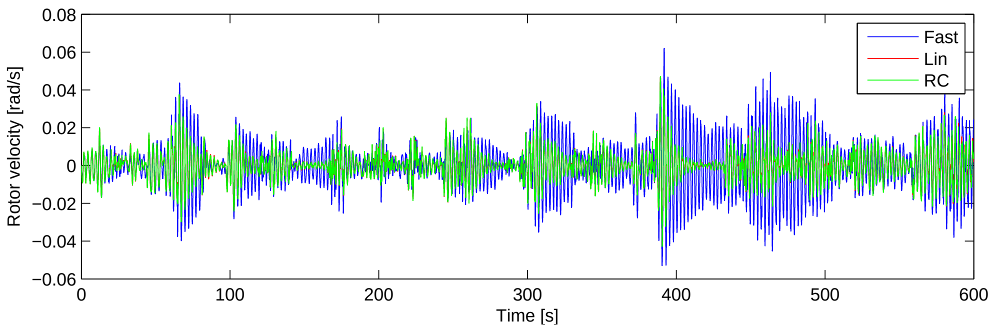


Figure F.1: Low speed shaft angular velocity

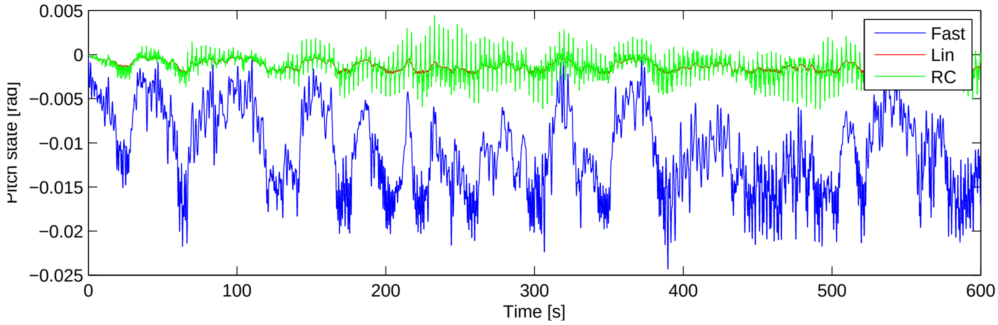


Figure F.2: Pitch angle on wind turbine

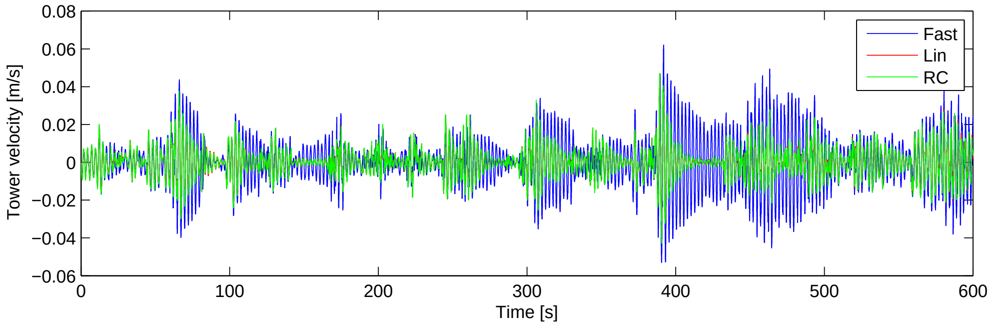


Figure F.3: Tower deflection velocity

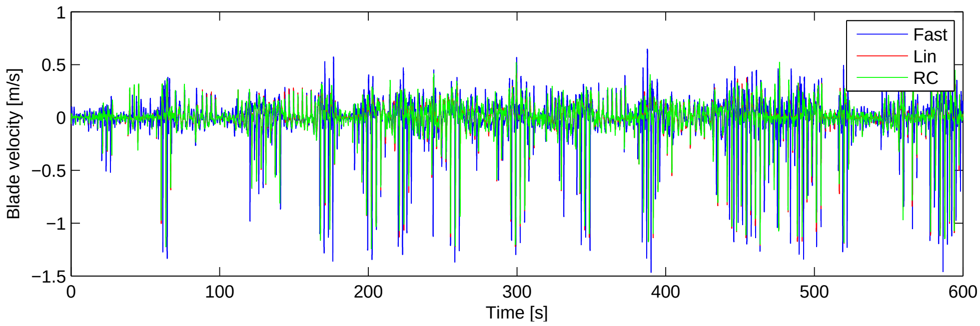


Figure F.4: Blade deflection velocity

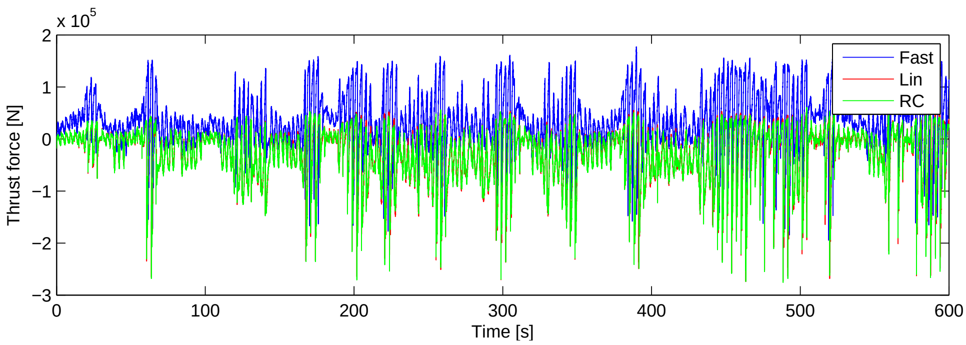


Figure F.5: Thrust force on single blade

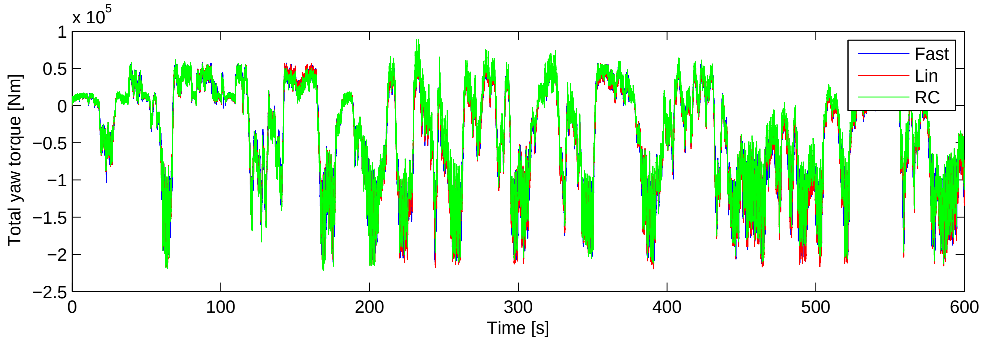


Figure F.6: Total yaw torque

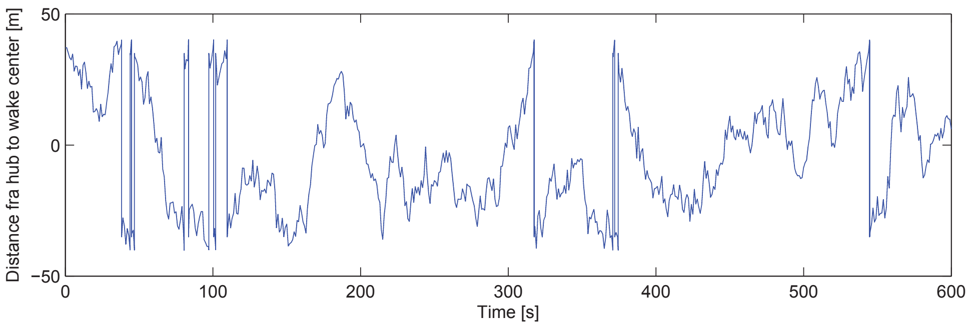


Figure F.7: Wake movement in wind field, at the fixed angle 0,5π. Sharper vertical lines is caused by the wake moving across the edge of the wind field and entering in the opposite side

Appendix G

Content of DVD

The included DVD have the following contents:

- Literature from the bibliography
- Digital copy of the report
- MATLAB/FAST files:
  - Acceptance test
  - FAST controller
  - FAST files

    How to run FAST
    Original FAST directory
  - Linearization

    Lift and drag factor
    Pre-pitch of the blade
    Width of the blade
  - Parameter estimation

    Aerodynamic
    Mechanical
    Structural
  - Validation
  - Wind model

    Simple wind field creation
    Gaussian wind field creation