Nonlinear Control of a Wind Turbine

Sven Creutz Thomsen

Kongens Lyngby 2006

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

Abstract

This thesis describes analysis of various nonlinear control methods for controlling a wind turbine. High speed wind conditions are considered, in which the rotational speed of the rotor and the generated power need to be stabilized at nominal values.

The methods investigated include gain scheduling, feedback linearization, sliding mode control and nonlinear inverse control. The gain scheduled controller has served as a basis of performance for the pure nonlinear methods.

Incomplete state knowledge and noisy measurements have been introduced in the simulation model. To provide state estimates, an extended Kalman filter has therefore been designed. The feedback linearization controller and the sliding mode controller, both proved advantageous compared to the gain scheduled controller. The sliding mode control law, however, resulted in the actuator limits being exceeded. Therefore, only feedback linearization, resulted in a control law, which could surpass the performance achieved with the gain scheduled controller.

Preface

This thesis was written to fulfill the requirements for obtaining the Masters degree in electrical ingeneering at the Technical University of Denmark (DTU). The project was supervised by Niels Kjølstad Poulsen, Section of Mathematical Statistics, Institute of Informatics and Mathematical Modeling. The project was conducted in the period from March 6$^{\text{th}}$ to September 6$^{\text{th}}$.

Prior to this project my background in nonlinear control includes an individual course covering the basics of Lyapunov theory and a short introduction to the basics of feedback linearization and sliding mode. This project has given me the opportunity to acquire thorough understanding of the nonlinear designs through practical application and theoretical study. Consequently, I feel that it has provided me with a launch pad for doing research in this area.

I would like to thank my supervisor Niels Kjølstad Poulsen for inspiration, for always being helpful and for giving me the possibility to continue the present study in a Ph.d.

Finally, I would like to thank my girlfriend Heidi Aino Hansen for being supportive throughout my path of attaining the Masters degree.

Lyngby, September 2006

Sven Creutz Thomsen
s011344

Nomenclature

θ	$[^\circ]$	-	Rotor pitch angle
δ	$[rad]$	-	Twist of rotor shaft
λ	[-]	-	Tip speed ratio
ρ	$[kg/m^3]$	-	Air density
ω_r	$[rad/s]$	-	Angular speed of rotor
ω_g	$[rad/s]$	-	Angular speed of generator
D_g	$[Nm \cdot s]$	-	Generator damping
D_s	$[Nm \cdot s]$	-	Shaft damping
J_g	$[kg \cdot m^2]$	-	Inertia of generator
J_r	$[kg \cdot m^2]$	-	Inertia of rotor
K_s	$[Nm]$	-	Spring constant of shaft
N_g	[-]	-	Gear ratio
P_e	$[W]$	-	Power generated by generator
P_r	$[W]$	-	Mechanical power absorbed from the wind
R_r	$[m]$	-	Radius of rotors
T_g	$[Nm]$	-	Generator load torque
T_r	$[Nm]$	-	Torque transfered from rotor to the gear
T_{sg}	$[Nm]$	-	Rotor torque transfered to generator
T_{sr}	$[Nm]$	-	Generator torque transfered to rotor
v	$[m/s]$	-	Effective wind speed
v_f	$[m/s]$	-	Fluctuating wind speed
v_m	$[m/s]$	-	Mean wind speed
c_p	[-]	-	Rotor efficiency coefficient

Contents

Chapter 1

Introduction

As a result of increasing environmental concern, more and more electricity is being generated from renewable sources. In Denmark wind turbines have experienced a wide popularity since the introduction in the early 1970ties, now covering 20 percent of the Danish energy consumption. Ongoing research is being focused at increasing the efficiency of the individual turbines and the parks of wind turbines as entities.

The basic operation of a wind turbine can be explained briefly: Wind turbines harness the power of the wind by using rotors, fitted with aerodynamic blades, to turn a shaft. The shaft rotates inside a generator which will then produce electricity. However, if left uncontrolled in the ever changing wind conditions, the stress on the structure could in worst case cause it to break. Furthermore, controlling the turbine makes it possible to optimize the power extraction from the wind or limit the extraction under strong wind conditions.

Modern turbines rely on complex control systems to maximize efficiency and ensure safe operation. Control of wind turbines can be divided into three levels (Johnson et al. 2006) as illustrated in figure 1.1. At the top level is the supervisory control which monitors the turbine and wind resource to determine, when to startup the turbine, shut down the turbine and shift between control strategies. At the middle level is the turbine controller which controls the blade pitch angle and generator torque, thereby controlling the power extraction from the wind. The turbine controller also controls the yaw of the nacelle, such that the nacelle points into the wind. However, this is a relative slow motion as compared to the dynamics of the turbine and is hence not of particular interest for a control engineer. At the lowest level are the controllers for the power electronics, internal generator and pitch actuator.

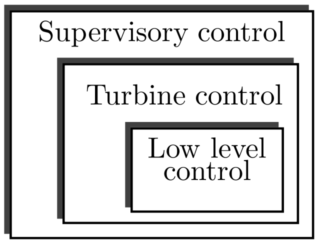


Figure 1.1: The levels of a wind turbine control system

In this project the turbine controller has been considered. However, the objective of the turbine controller changes according to the present wind speed. As illustrated in figure 1.2 the objective can be sorted according to four wind regions. Only the

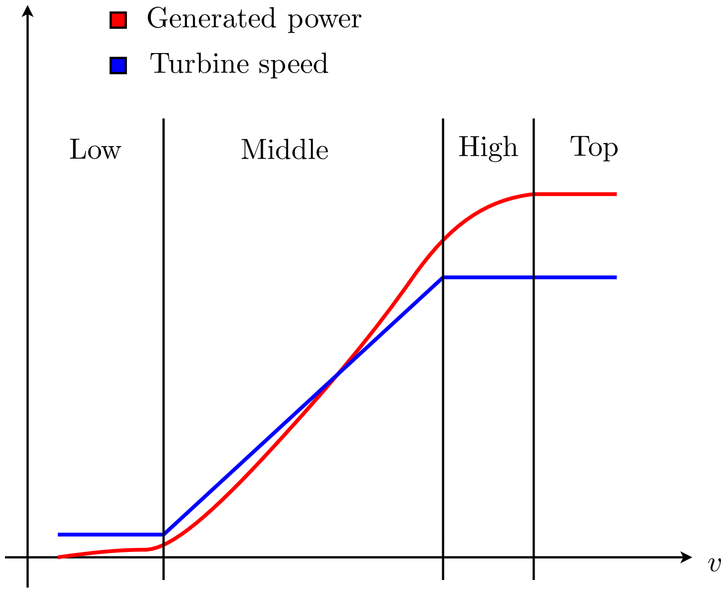


Figure 1.2: Control regions for a turbine controller

top region has been considered in this project. In the top region, the available power in the wind, exceeds the limit for which the generator and the turbine mechanics have been designed. Hence, to avoid failure the speed and output power should be kept constant at nominal values. The consequent control problem is the classical stabilization problem. A description of the other regions can be found in (Larsen and Mogensen 2006).

The type of wind turbine, which is considered in this project, is a horizontal axis pitch regulated wind turbine with three blades. Moreover the turbine includes a variable speed asynchronous generator which allows for more efficient control of the turbine. 'Variable speed' simply means that it is possible to vary the relative speed of the generator, compared to the speed/frequency of the electrical grid. In effect, changing the relative speed changes the generator torque and thereby provides a useful control parameter.

Much research has gone into designing efficient turbine controllers. A commonly used method is the PID controller which requires little knowledge of underlying mathematical model. These PID controllers are gain scheduled to accommodate for the varying wind conditions (Hansen et al. 2005). Other, more theoretical studies, include linear design methods such as LQ control (Larsen and Mogensen 2006) and stochastic control (Xin 1993). Common for most of the previous work is that the controllers do not consider the nature of the nonlinearities directly and thereby accommodate for these.

In the present work, nonlinear design tools have been investigated for controlling a wind turbine in the top wind speed region. The methods include feedback linearization, sliding mode control and inverse optimal control. Furthermore, a gain scheduled linear controller has been designed to provide a basis of comparison. Gain scheduling is basically just an extension of linear control and will *not* be regarded as a pure nonlinear method. However, it is a method that has proven its worth in a wide range of application, one of these being the wind turbine. Feedback linearization and sliding mode control are also methods that have proven their worth in practical applications. Inverse nonlinear optimal control, is the most recent method of the four and its appeal is more mathematical than it is practical.

Outline

The report is divided into three parts:

Part 1: Modeling and analysis The first part introduces the wind turbine models and the wind model which have been used in the project. The dynamic response of the uncontrolled turbine is illustrated through deterministic and stochastic simulations, and the eigenvalues of the corresponding linear model is analyzed. Finally, the development framework which have been used, is presented.

Part 2: Control designs The second part describes the objectives which the nonlinear controllers have to meet. The control methods are investigated individually in an idealized setup. More specifically, it is assumed that all measurements are available, and that there are no model uncertainties.

Part 3: Control with incomplete state knowledge The third part introduces a more realistic setting. It is assumed that limited measurements are available, furthermore noise is added to the measurements. To supply estimates for the controller an extended Kalman filter is designed. The controllers are tested in the perturbed setting, and their performances are evaluated and compared with each other.

Part 4: Conclusion and perspectives The last part, summarizes the objectives which have been attained in the project, and concludes on the results. Furthermore, different perspectives are discussed.

prerequisites

It is assumed that the reader of this report has a background in linear control theory, but not necessarily familiar with the nonlinear design tools, presented in this report. Therefore, all nonlinear methods that are introduced will be explained thoroughly, before they are applied to the wind turbine. If necessary Appendix B contains an introduction to nonlinear systems and the concept of Lyapunov stability.

Part I

Modeling and analysis

CHAPTER 2

Model descriptions

The wind turbine considered in this project is a 3-rotor horizontal axis pitch-regulated wind turbine (See figure 2.1) with an asynchronous generator. The model introduced

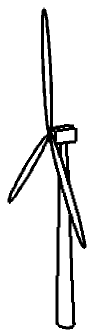


Figure 2.1: Horizontal axis wind turbine

in this chapter, applies to a wide range of horizontal axis wind turbines, however the parameters used in the model are (partly) taken from a Vestas v29 225 kW wind turbine. The wind turbine parameters can be seen in appendix A.

Two mathematical models for the wind turbine are introduced in this chapter: The primary model is a variable speed (multi-input) 5th order model that has been used as a benchmark model for the designs. The secondary model is a constant speed (single-input) 2nd order model, the purpose of which is to exemplify the nonlinear designs and to perform initial investigations with reduced complexity.

The adjectives 'constant' and 'variable' indicates whether it is possible to vary the relative speed between the generator and electrical grid. In effect, changing the relative speed, changes the torque of the generator. The inputs of a variable speed turbine are hence pitch angle as well as the generator torque, whereas the constant speed turbine only has the pitch angle as input.

Another important aspect when modeling a wind turbine, is the influence of the wind. The wind is regarded as a stochastic process and will in this project be modeled using a gain scheduled 2nd order linear filter driven by a white noise process.

The outline of this chapter is as follows: In section 2.1 the dynamic model for the variable speed turbine is derived. Section 2.2 briefly derives the constant speed model. Section 2.3 explains how the wind has been modeled and in section 2.4 the variable speed model is augmented with the wind model. In section 2.5 the wind augmented variable speed model is linearized.

2.1 Variable speed wind turbine

In this section the mathematical model will be derived for the variable speed wind turbine. The dynamic relations presented in this chapter has previously been introduced in (Horsbøl 2003).

A model is a simplification of a real world problem, and should capture the main characteristics of the problem. The wind turbine characteristics which will be considered are:

- Aerodynamics
- Turbine mechanics
- Generator dynamics
- Actuator dynamics

The modeling perspective can be separated into sub-models describing the characteristics above. The interconnections between these sub-models are seen in figure 2.2. A detailed descriptions of the sub-blocks, and the associated mathematical models, are given in the following sections.

2.1.1 Mechanics

Figure 2.3 shows a schematic of the wind turbine mechanics. The turbine is split

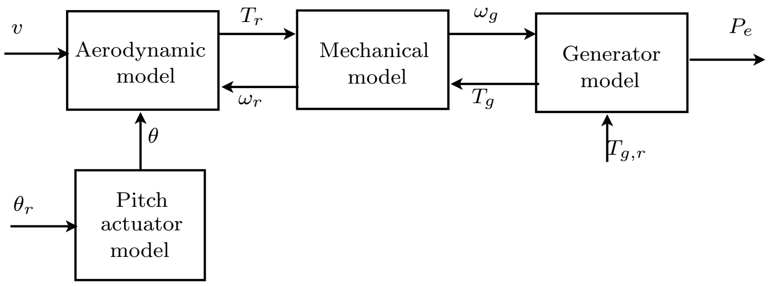


Figure 2.2: Interconnection of sub-models describing the characteristics of the wind turbine

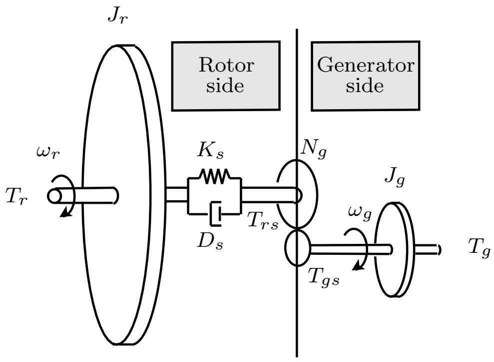


Figure 2.3: Schematic of the wind turbine mechanics

into two parts separated by the transmission: The rotor side and the generator side. The inertia on the rotor side J_r and generator side J_g are illustrated by the leftmost and the rightmost disc respectively. The shaft connecting the rotor to the transmission is subject to immense torques that cause it to twist, consequently the shaft is appropriately modeled as a damped spring. The dynamic nature of the shaft (the drive train dynamics) is illustrated by the damping D_s and the spring constant K_s. The gear ratio N_g is illustrated by the discs in the middle. On the left the model is exited by the rotor torque T_r and on the right the generator torque T_g. The torques T_{sr} and T_{sg} are the torques at each side of the transmission, which are related by the gear ratio:

$$T_{sg} = \frac{T_{sr}}{N_g} \tag{2.1}$$

The equations describing the dynamics are obtained using Newton's second law for rotating bodies. This results in two equations: one for the rotor side and one for the generator side.

$$\begin{aligned} \dot{\omega}_r J_r &= T_r - T_{sr} && (2.2) \\ \dot{\omega}_g J_g &= T_{sg} - T_g && (2.3) \end{aligned}$$

Introducing a variable δ [rad] describing the twist of the shaft, leads to the following equation describing the twist of the flexible shaft:

$$T_{sr} = D_s\dot{\delta} + K_s\delta \tag{2.4}$$

where

$$\delta = \Omega_r - \frac{\Omega_g}{N_g} \quad , \quad \dot{\delta} = \omega_r - \frac{\omega_g}{N_g} \tag{2.5}$$

Ω_r and Ω_g are naturally the shaft angle at the rotor and the generator respectively.

2.1.2 Aerodynamics

The aerodynamic blades on the rotor converts the kinetic energy of the wind into mechanical energy, effectively providing the torque T_r on the rotor:

$$T_r = \frac{P_r}{\omega_r} \tag{2.6}$$

where the power P_r is given by the following relation (DNV/Risø 2002):

$$P_r = \frac{1}{2}\rho\pi R^2 v^3 c_p(\lambda, \theta) \tag{2.7}$$

ρ is the air density, R the wing radius and v the effective wind speed. c_p is the efficiency coefficient which is a function of the blade pitch angle θ and the tip speed ratio λ. In the context of this project, λ is the ratio between the blade tip speed and the wind speed.

$$\lambda = \frac{v}{v_{tip}} = \frac{v}{R\omega_r} \tag{2.8}$$

however the inverse ratio is also commonly met in the literature. The performance coefficients c_p for wind turbines are obtained through numerical calculations and are hence given as a look-up tables. A three dimensional plot of the efficency coefficient for the Vestas v29 is shown in figure 2.4.

The objective in this project has been to stabilize the power extraction and the rotor speed at:

$$P_{nom} = 225 \text{ kW} \qquad \omega_{r,nom} = 4.29 \text{ rad/s}$$

Iso-power curves (kW) of the power extracted by the turbine, at the nominal rotor speed, is seen figure 2.5. The curves are plotted as functions of the pitch angle and the wind speed. It is seen that the wind speed accordingly has to be greater than 11 m/s to extract the nominal power. An interesting characteristic revealed by the iso-power curves, is that a given power can be extracted by two values of the pitch[1]. The power can hence be limited by either pitching the blades in the positive

[1]This is not easily seen in the plot 2.5, since the table based c_p factor is not defined for pitch angles below $-1°$. However, this is a general characteristic of the aerodynamic blades (DNV/Risø 2002)

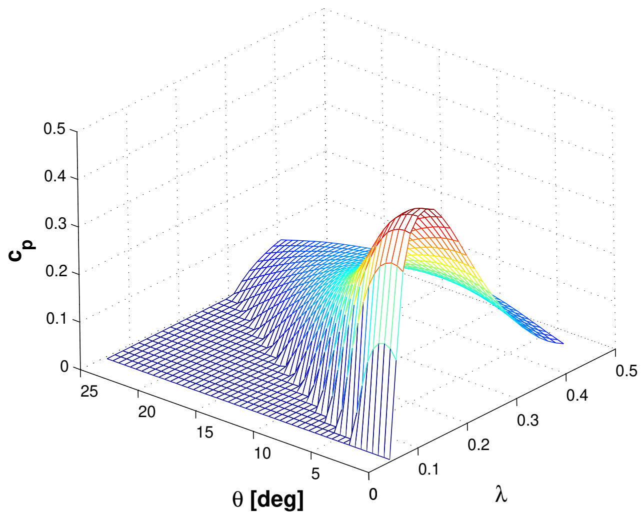


Figure 2.4: The efficiency coefficient c_p for the blades on a Vestas v29 wind turbine. The coefficient is a function of the tip speed ratio λ and the pitch angle θ. Negative values have been truncated.

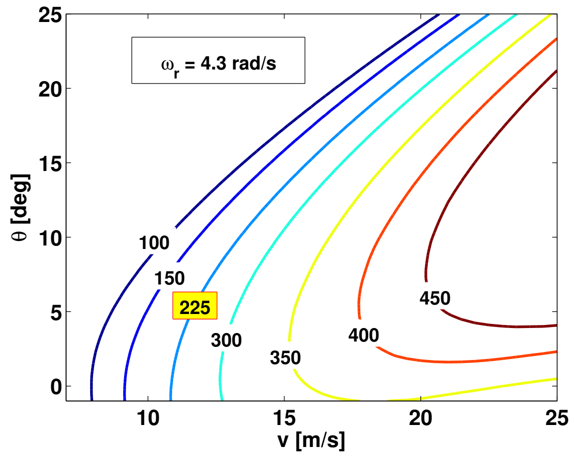


Figure 2.5: Iso-power curves (kW) based on the efficiency coefficient for the Vestas v29 wind turbine. The curves are generated at the nominal rotational speed. The control objective is to stabilize the output of the (lossless) generator at 225kW, hence the associated iso-power curve is of particular interest.

direction or in the negative direction. Negative pitching, is called active-stall control and positive pitching is called pitch control. Only pitch control has been investigated in this project, hence the pitch angle will be increased when the wind speed increases.

The nonlinear methods which will be presented in part 2 make heavy use of directional derivatives. The derivative of c_p with respect to both λ and θ are therefore used extensively in the control designs. In previous work, numerical derivatives of c_p with respect to λ and θ have been successfully obtained (See for example (Klingenberg 1991)) and utilize in linear control designs. However, the nonlinear controllers have proven sensitive towards the noise introduced by the numerical derivations, making it difficult to validate the responses obtained by e.g. a feedback linearizing controller. Consequently, it has been chosen to base the aerodynamic model on an analytic expression of c_p. The analytical approximation used in this project has been taken from (Slootweg et al. 2001).[2]

$$c_p(\lambda, \theta) = 0.22 \left(\frac{116}{\lambda_t} - 0.6\theta - 5 \right) \exp \left(-\frac{12.5}{\lambda_t} \right) \tag{2.9}$$

where

$$\frac{1}{\lambda_t} = \frac{1}{\lambda^{-1} + 0.12\theta} - \frac{0.035}{(1.5\theta)^3 + 1}$$

The iso-power curves, obtained with the analytical efficiency coefficient, are seen in figure 2.6 for the nominal rotor speed. At low pitch angles the analytical iso-power

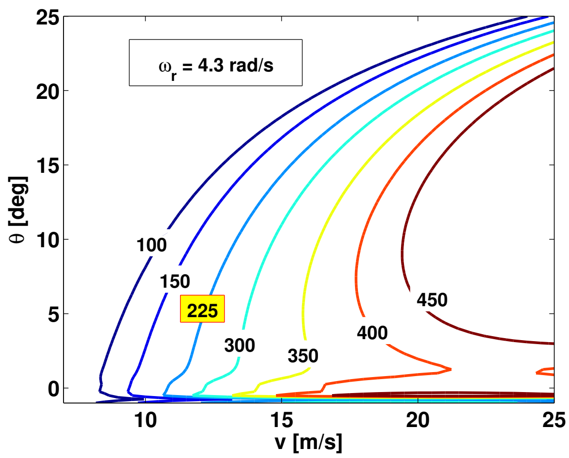


Figure 2.6: Iso-power curves based on the analytical efficiency coefficient.

curves exhibit an unwanted behavior. However, if focusing on the iso-power curve corresponding to the nominal power $P_{nom} = 225$ kW, it is seen that the behavior is acceptable when working at wind speeds above approx. 12 m/s. Accordingly, only wind speeds above 12m/s will be considered in the following chapters.

[2]The original formulation uses the inverse tip speed ratio mentioned earlier. Furthermore, the parameters have been tuned slightly.

2.1.3 Generator dynamics

It has been assumed that the asynchronous generator is ideal (lossless), hence the generator power is given by:

$$P_e = T_g \omega_g$$

As mentioned, the generator torque T_g can be controlled, however, it can not be changed instantaneously. The dynamic response of the generator has therefore been modeled by a first order linear model with time constant τ_T:

$$\dot{T}_g = -\frac{1}{\tau_T} T_g + \frac{1}{\tau_T} T_{g,r} \tag{2.10}$$

2.1.4 Pitch actuator

The pitch of the blades is changed by a hydraulic/mechanical actuator. A simplified model of the dynamics is presented by the following first order linear model:

$$\dot{\theta} = -\frac{1}{\tau_\theta}\theta + \frac{1}{\tau_\theta}\theta_r \tag{2.11}$$

2.1.5 Nonlinear state space description

Combining all the previous equations results in the following set of coupled first order differential equations (state-space model)

$$\begin{aligned}
\dot{\omega}_r &= \frac{P_r(\omega_r, \theta, v)}{\omega_r J_r} - \frac{\omega_r D_s}{J_r} + \frac{\omega_g D_s}{J_r N_g} - \frac{\delta K_s}{J_r} \\
\dot{\omega}_g &= \frac{\omega_r D_s}{N_g J_g} - \frac{\omega_g D_s}{N_g^2 J_g} + \frac{\delta K_s}{N_g J_g} - \frac{T_g}{J_g} \\
\dot{\delta} &= \omega_r - \frac{\omega_g}{N_g} \\
\dot{\theta} &= -\frac{1}{\tau_\theta}\theta + \frac{1}{\tau_\theta}\theta_r \\
\dot{T}_g &= -\frac{1}{\tau_T} T_g + \frac{1}{\tau_T} T_{g,r}
\end{aligned}$$

Defining the state and input vector as

$$\begin{aligned}
x &= \begin{bmatrix} \omega_r & \omega_g & \delta & \theta & T_g \end{bmatrix}^T \\
u &= \begin{bmatrix} \theta_r & T_{g,r} \end{bmatrix}^T
\end{aligned}$$

allows for the compact notation:

$$\dot{x} = f(x, u) \tag{2.12}$$

or, since the system is affine in the input (also commonly denoted: linear in input), it can be put in the following form which comes in handy when dealing with the nonlinear control theory.

$$\dot{x} = f(x) + Gu \tag{2.13}$$

$$= \begin{bmatrix} \frac{P_r(x_1,x_4,v)}{x_1 J_r} - \frac{x_1 D_s}{J_r} + \frac{x_2 D_s}{J_r N_g} - \frac{x_3 K_s}{J_r} \\ \frac{x_1 D_s}{J_g N_g} - \frac{x_2 D_s}{N_g^2 J_g} + \frac{x_3 K_s}{N_g J_g} - \frac{x_5}{J_g} \\ x_1 - \frac{x_2}{N_g} \\ -\frac{1}{\tau_\theta} x_4 \\ -\frac{1}{\tau_T} x_5 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{\tau_\theta} & 0 \\ 0 & \frac{1}{\tau_T} \end{bmatrix} u \tag{2.14}$$

The wind speed is naturally a disturbance input to the model. However, to conform with later notation, it is not shown explicitly in equation (2.13).

2.2 Constant speed wind turbine model

This section presents a simple constant speed model. The model has been used to exemplify and make preliminary analysis of the nonlinear methods, without having to deal with multi-input theory. The model is a nonlinear second order system which makes it possible, to illustrate the state evolution in a phase plane.

What differentiates a constant speed turbine from a variable speed turbine, is that the generator torque cannot be controlled. Hence, the model diagram in figure 2.2, describes the dynamics of the constant speed model, if leaving out the input $T_{g,r}$. Furthermore the drive-shaft dynamics have been left out, such that the resulting model is a second order nonlinear system. In conclusion, only the mechanical model and the generator need to be reformulated to obtain the constant speed model.

2.2.1 Mechanics

The constant speed model has been rendered a second order system by excluding the drive-shaft dynamics in the mechanical model. Consequently the inertia of the rotor is merged with the generator side inertia (taking into account the gear ratio):

$$J = \frac{J_r}{N_g^2} + J_g$$

Newton's second law for rotating bodies is set up on the generator side:

$$\dot{\omega}_g J = T_{sg} - T_g$$

T_{sg} is simply the rotor torque transfered to the generator side

$$T_{sg} = \frac{T_r}{N_g}$$

The equation describing the simplified turbine mechanics is consequently:

$$\dot{\omega}_g J = \frac{T_r}{N_g} - T_g \tag{2.15}$$

2.2.2 Generator

The generator torque is no longer controlled but given by the following expression:

$$T_g = D_g s \tag{2.16}$$

where D_g is the generator damping and s is the difference between the rotor angular velocity and the net frequency divided by the number of poles in the generator:

$$s = \omega_g - \frac{\omega_0}{p} \tag{2.17}$$

2.2.3 Nonlinear state space description

Combining the equations (2.15)-(2.17) with the equations describing the aerodynamics and the pitch actuator (from the variable speed model) the following nonlinear state space model is obtained.

$$\begin{aligned}\dot{x}_1 &= \frac{P(x_1, x_2, v)}{Jx_1} - \frac{D_g x_1}{J} + \frac{D_g \omega_0}{Jp} \\ \dot{x}_2 &= -\frac{1}{\tau_\theta} x_2 + \frac{1}{\tau_\theta} u\end{aligned}$$

where the state and input vector are

$$x = \begin{bmatrix}\omega_g & \theta\end{bmatrix}^T \quad u = \theta_r$$

For convenience the system is given in the affine form:

$$\begin{aligned}\dot{x} &= f(x) + Gu \\ &= \begin{bmatrix}\frac{P(x_1,x_2,v)}{Jx_1} - \frac{D_g x_1}{J} + \frac{D_g \omega_0}{Jp} \\ -\frac{1}{\tau_\theta} x_2\end{bmatrix} + \begin{bmatrix}0 \\ \frac{1}{\tau_\theta}\end{bmatrix} u\end{aligned} \tag{2.18}$$

It is seen that the constant speed model incorporates the same nonlinearity as the variable speed model making it suitable for preliminary analysis.

2.3 Wind model

A model of the effective wind speed v is derived in this section. This model is used in the simulations to provide a realistic wind disturbance. Furthermore, the model structure has been utilized in the controller designs, thereby obtaining more efficient control.

The effective wind speed v is modeled as a mean wind speed v_m superimposed by a turbulent wind speed v_f.

$$v = v_m + v_f$$

The mean wind speed is assumed constant (or at least slowly varying as compared to the dynamics of the wind turbine). The turbulent wind speed will however be modeled as a filter driven by a white noise process. The mean wind speed has been assumed measurable throughout the project.

The model of the fluctuating wind speed is established in the frequency domain. It is derived in two steps: First the frequency contents (power spectrum density) of the wind in a single point, on the disc swept by the rotor, is derived. This quantity is denoted the *point wind*. However, the turbine is not subject to a single wind speed, but rather a spatially distributed wind field swept by the entire rotor disc. Therefore the power spectrum density is filtered by a suitable filter such that the resulting power spectrum density describes the effective (fluctuating) wind speed.

2.3.1 The point wind

Following the approach of (Højstrup 1982) the frequency contents of the point wind can be described by the following power spectrum density:

$$S_p(f)\frac{f}{v_{fr}^2} = \underbrace{\frac{0.5f_i}{1+2.2f_i^{\frac{5}{3}}}\left(\frac{h_i}{-L}\right)}_{\text{stable}} + \underbrace{\frac{105f_{ru}}{(1+33f_{ru})^{\frac{5}{3}}}\frac{(1-\frac{h}{h_i})}{(1+15\frac{h}{h_i})^{\frac{2}{3}}}}_{\text{unstable}} \tag{2.19}$$

where

$$f_i = f\frac{h_i}{v_m} \quad , \qquad f_{ru} = \frac{f}{1+15\frac{h}{h_i}}$$

the nomenclature is:

f	:	Frequency (Hz)
h	:	The height under consideration i.e. the heigh of the nacelle (m)
h_i	:	The height of the lowest inversion (m)
L	:	Monin-Obukhov length
v_{fr}	:	Friction velocity

For further details on the parameters in the expression refer to (Larsen and Mogensen 2006). The expression (2.19) is divided into two parts: The first part is dominant in relative stable wind conditions and the second is dominant in unstable wind conditions. The wind turbine operates in relative unstable conditions (Knudsen 1983) and the first part is therefore not considered in the following derivation:

$$S'_p(f)\frac{f}{v_{fr}^2} = \frac{105 f_{ru}}{(1+33f_{ru})^{\frac{5}{3}}}\frac{(1-\frac{h}{h_i})^2}{(1+15\frac{h}{h_i})^{\frac{2}{3}}} \tag{2.20}$$

The only unknown parameter in (2.20) is the friction velocity v_{fr}. To derive v_{fr} the variance of the point wind is calculated:

$$\sigma_p^2 = \int_{-\infty}^{\infty} S'_p(f)df = \frac{105}{22}\frac{\left(1-\frac{h}{h_i}\right)^2}{\left(1+15\frac{h}{h_i}\right)^{\frac{2}{3}}}v_{fr}^2 \tag{2.21}$$

According to (Knudsen 1983) the variance can also be expressed as a turbulent intensity τ_t multiplied by the mean wind velocity:

$$\sigma_p^2 = (\tau_t \cdot v_m)^2 \tag{2.22}$$

The turbulent intensity is a known factor which dependents on the terrain and the observation height, hence the friction velocity v_{fr} is found by combining (2.21) and (2.22):

$$v_{fr}^2 = (\tau_t \cdot v_m)^2 \frac{22}{105}\frac{\left(1+15\frac{h}{h_i}\right)}{\left(1-\frac{h}{h_i}\right)} \tag{2.23}$$

the final expression for the point wind power spectrum becomes:

$$S'_p(f)f = (\tau_t \cdot v_m)^2 \frac{22 f_{ru}}{(1+33f_{ru})^{\frac{5}{3}}} \tag{2.24}$$

2.3.2 Effective wind speed

Having established a spectrum for the point wind the next step is to obtain a spectrum for the effective wind speed. According to (Knudsen 1983), the spectrum of the effective wind speed can be approximated by filtering the point wind with the following filter:

$$F(f) = \frac{1}{\left(1+\frac{8\sqrt{\pi}R}{3v_m}f\right)\left(1+\frac{4\sqrt{\pi}R}{v_m}f\right)} \tag{2.25}$$

The spectrum of the effective wind speed is accordingly:

$$S_e(f) = S'_p(f)F(f) \tag{2.26}$$

2.3.3 Design and simulation model

The filter which is derived above is a nonlinear model in the frequency domain. For the controller designs and the simulations a model is needed in the time domain. Following the example of (Larsen and Mogensen 2006) the model is therefore approximated by a second order linear filter for a given mean wind velocity. Consequently the wind model becomes a gain scheduled linear filter, scheduled according to the mean wind speed v_m.

The transfer function of the linear filter is:

$$H(s) = \frac{K_v}{s^2 + a_2 s + a_1}$$

and the corresponding power density spectrum is

$$S_H(f) = H(j2\pi f)H(j2\pi f)^*$$

The poles of the linear filter for a given mean velocity is then established by numerically minimizing the cost:

$$J = \int_{f_1}^{f_2} \left[\log\left(\frac{S_H(f)}{S_e(0)}\right) - \log\left(\frac{S_e(f)}{S_e(0)}\right) \right]^2 df$$

The spectrums are scaled with the $S_e(0)$ to ensure that the approximated linear filter has the correct DC value. Figure 2.7 shows the frequency characteristics of the nonlinear model versus the approximated linear model for the mean wind speed $v_m = 18\text{m/s}$.

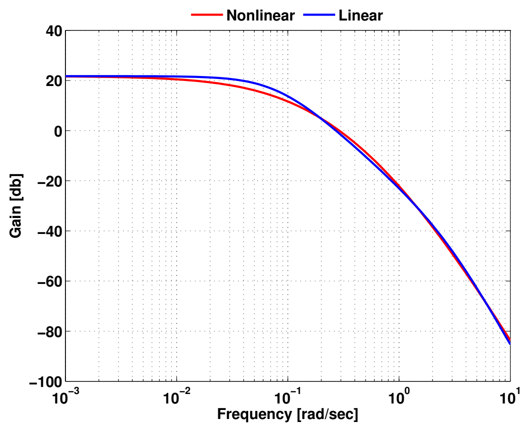


Figure 2.7: Frequency characteristics of the nonlinear model versus the approximated linear model.

The equivalent state-space formulation of the wind model is:

$$\begin{aligned}\dot{w} &= f_w(w) + G_w e \\ &= \begin{bmatrix} w_2 \\ -a_2 w_2 - a_1 w_1 \end{bmatrix} + \begin{bmatrix} 0 \\ K_v \end{bmatrix} e \qquad (2.27) \\ v &= h(w) \\ &= w_1 + v_m \qquad (2.28)\end{aligned}$$

where $e \in \mathcal{N}(0, 1^2)$ is a white noise process. The model is written in the general affine form to conform with later notation.

2.4 Wind augmented variable speed model

The known structure of the wind is utilized in the controller and observer designs. Formally, this is done by combining the turbine model and the wind model in the design process. To reduce the notation in the following chapters, the augmentation is described here.

Augmenting the variable speed turbine (2.14) with the wind model (2.27) results in the following state-space model which is driven by a white noise process $e \in \mathcal{N}(0, 1^2)$.

$$\begin{aligned}\begin{bmatrix} \dot{x} \\ \hline w \end{bmatrix} &= \begin{bmatrix} \frac{P_r(x_1,x_2,w_1)}{x_1 J_r} - \frac{x_1 D_s}{J_r} + \frac{x_2 D_s}{J_r N_g} - \frac{x_3 K_s}{J_r} \\ \frac{x_1 D_s}{J_g N_g} - \frac{x_2 D_s}{N_g^2 J_g} + \frac{x_3 K_s}{N_g J_g} - \frac{x_5}{J_g} \\ x_1 - \frac{x_2}{N_g} \\ -\frac{1}{\tau_\theta} x_4 \\ -\frac{1}{\tau_T} x_5 \\ \hline w_2 \\ -a_2 w_2 - a_1 w_1 \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{\tau_\theta} & 0 \\ 0 & \frac{1}{\tau_T} \\ \hline 0 & 0 \\ 0 & 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \hline 0 \\ K_v \end{bmatrix} e \qquad (2.29) \\ &\equiv f(x, w) + Gu + G_w e\end{aligned}$$

The mean wind speed has been absorbed into the model and is hence not shown explicitly. It is seen that the resulting model is not only affine in the input but also in the disturbance. This makes it easier to deal with the disturbance in the nonlinear designs.

2.5 Linear wind augmented turbine model

Although this report deals with nonlinear control, the linear model corresponding to the wind augmented variable speed model (2.29) is needed for the gain scheduling controller and the state estimator. Therefore to reduce the notation in the following chapters, the linear model is introduced here.

The linear state space description is trivially obtained by calculating the Jacobians of the system in the point of linearization x^*. The following notation

$$\bar{x} = x - x^*$$

will be used to describe the deviation of the state from the point of linearization. Equivalent notation is used for the input, disturbance and output.

Calculating the Jacobians of the wind augmented variable speed model (2.29) results in the following linear state space description

$$\begin{aligned}
\begin{bmatrix} \dot{\bar{x}} \\ \dot{\bar{w}} \end{bmatrix} &= A \begin{bmatrix} \bar{x} \\ \bar{w} \end{bmatrix} + B\bar{u} + B_w \bar{w} \\
&= \begin{bmatrix}
A_{11} & \frac{D_s}{J_r Ng} & -\frac{K_s}{J_r} & A_{14} & 0 & A_{16} & 0 \\
\frac{D_s}{J_g N_g} & -\frac{D_s}{N_g^2 J_g} & \frac{K_s}{J_g N_g} & 0 & -\frac{1}{J_g} & 0 & 0 \\
1 & -\frac{1}{N_g} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{1}{\tau_\theta} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{1}{\tau_T} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & a_1 & a_2
\end{bmatrix}_{x=x_0} \begin{bmatrix} \bar{x} \\ \bar{w} \end{bmatrix} + \\
& \begin{bmatrix}
0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{\tau_\theta} & 0 \\ 0 & \frac{1}{\tau_T} \\ 0 & 0 \\ 0 & 0
\end{bmatrix} \bar{u} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ K_v \end{bmatrix} e
\end{aligned} \tag{2.30}$$

where

$$\begin{aligned}
A_{11} &= \frac{\frac{\partial}{\partial x_1} P_r(x_1, x_2, w_1)}{x_1 J_r} - \frac{P_r}{x_1^2 J_r} - \frac{D_s}{J_r} \\
A_{14} &= \frac{\frac{\partial}{\partial x_4} P_r(x_1, x_2, w_1)}{x_1 J_r} \\
A_{16} &= \frac{\frac{\partial}{\partial x_6} P_r(x_1, x_2, w_1)}{x_1 J_r}
\end{aligned}$$

Chapter 3

Analysis of turbine dynamics

This chapter presents a preliminary study of the variable speed wind turbine model. Both deterministic simulations and simulations with the stochastic wind disturbance are presented.

Section 3.1 outlines how the stationary conditions for the turbine model are found. Section 3.2 analyses the eigenmodes of the linearized system and section 3.3 studies the step response of the system. Section 3.4 shows the response of the nonlinear system when subject to a stochastic wind sequence.

3.1 Stationary conditions

As mentioned the objective of the controllers is to keep the rotational speed constant and at the same time provide a constant power output. Consequently three parameters are known:

$$\omega_{r,nom} = 4.29 \text{ rad/s} \quad \omega_{g,nom} = 105.6 \text{ rad/s} \quad P_{e,nom} = 225000 \text{ W}$$

Hence, for a given value of the effective wind speed v, the stationary values of the remaining variables are calculated by setting $\dot{x} = 0$ in the state equation of the turbine. The solution cannot be found analytically due to the intricate expression of c_p, however by using a numerical solver the stationary points are easily obtained. It turns out that only the stationary value of the pitch angle θ and the corresponding reference θ_r changes for different values of v in the top region. The stationary values of the states are shown in table 3.1 for different values of the wind speed:

v	[m/s]	14	16	18	20	22
ω_r	[rad/s]	4.29				
ω_g	[rad/s]	105.6				
δ	[rad]	$6.55 \cdot 10^{-3}$				
θ	[deg]	11.9	16.2	19.1	21.2	22.8
T_g	[Nm]	2130				

Table 3.1: Stationary values of the state for different values of the effective wind speed.

3.2 Eigenmodes

This section studies the change of the wind turbine eigenmodes for different wind speeds in the top wind region. The linearization method provides an excellent tool, for analyzing how the system properties change when parameters in the model are varied. The eigenmodes give an insight into the behavior of the system, such as the speed of convergence, damping and simply whether a system is stabile. The linearized variable speed turbine is trivially obtained as a subset of the wind augmented linear turbine 2.30.

The variable speed wind turbine has five eigenvalues. However, the nonlinearities in the system, only have an effect on three of the eigenvalues, namely the ones associated with the turbine mechanics:

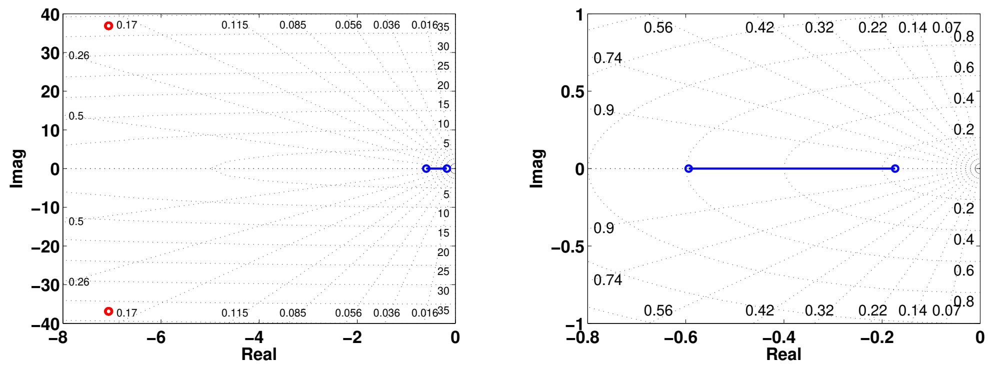
$$x_m = \begin{bmatrix} \omega_r & \omega_g & \delta \end{bmatrix}^T$$

Intuitively, this is because the actuator dynamics are independent of the rest of the system. Consequently only the eigenmodes associated with x_m are of interest in the following.

Figure 3.1(a) shows a root locus plot of the eigenvalues as the wind speed is varied from 14 to 22. 3.1(b) takes a closer look at the slowest eigenvalue. It is seen that there are two complex conjugated eigenvalues and a real eigenvalue, all stable throughout the range of interest. The values of the eigenvalues at integer values of the wind speed is seen in table 3.2. Some insight into the behavior of the wind turbine can be

v	λ	
14	-0.1733	$-7.0601 \pm 36.8918i$
16	-0.2726	$-7.0634 \pm 36.8911i$
18	-0.3736	$-7.0668 \pm 36.8904i$
20	-0.4803	$-7.0704 \pm 36.8897i$
22	-0.5941	$-7.0742 \pm 36.8888i$

Table 3.2: Eigenvalues at integer values of the wind speed.



(a) Overview

(b) Zoom view

Figure 3.1: Root locus of the eigenvalues associated with x_s, as the wind speed is varied from 14 to 22 m/s. The rightmost plot takes a closer look at the real eigenvalue.

deduced from the location of the eigenvalues: The imaginary component of the complex conjugated pair is large relative to the real part. Consequently this eigenvalue pair is associated with poorly damped oscillating dynamics. The real eigenvalue is associated with a well damped response (no oscillations). Comparing the absolute values of the eigenvalues, it is seen that the real eigenvalue is much smaller than the complex conjugated pair. Hence, the dynamics of the complex conjugated pair will die out quickly relative to the dynamics associated will the real eigenvalue. The fast oscillating dynamics are intuitively an effect of the drive train dynamics, which is very poorly damped. The slow dynamics are due to the large inertia of the turbine mechanics.

The system is obviously nonlinear, since the location of the eigenvalues changes when the operating conditions change. A significant change is however only seen on the real eigenvalue, which is increased significantly when the mean wind speed increases. hence, the stronger the wind conditions, the faster the response of the turbine.

3.3 Deterministic simulations

This section presents deterministic simulations illustrating the dynamic response of the wind turbine. Both the linear and the nonlinear model is considered.

Figure 3.2 shows the step response of the linearized wind turbine, associated with four different values of the wind speed. The step is performed on the wind disturbance. It is clearly seen that the speed of the system changes as the point of linearization is varied.

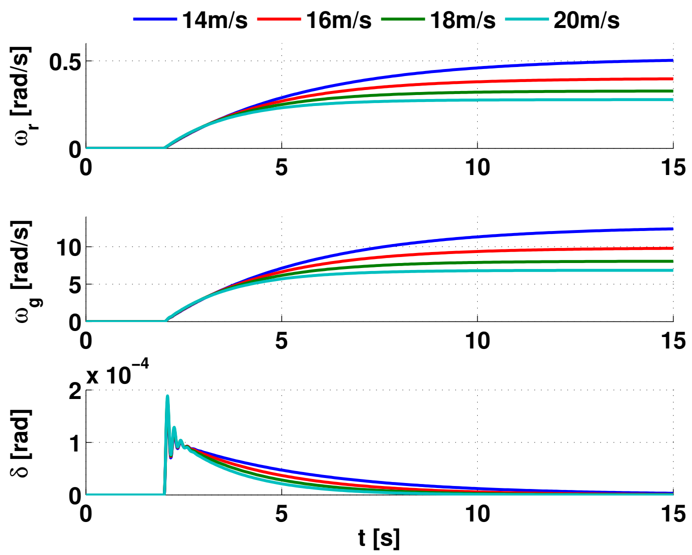


Figure 3.2: Disturbance step response of the linear wind turbine model, linearized about 4 different values of v_m. The wind disturbance is stepped with unity magnitude. Note that the system dynamics is faster at high wind speeds.

Figure 3.3 shows the response of the nonlinear system and the linear system, associated with $v = 14$. The systems are exited by a step-sequence which is also shown in the figure. As expected, the difference between the response if the linear and the nonlinear model increases as the operating conditions move away from the point of linearization.

3.4 Simulation with stochastic wind

This section presents the open loop response of the nonlinear system when subjected to the stochastic wind disturbance. The simulation has been made with the stochastic wind model introduced in chapter 2 with a mean wind speed of $v_m = 18\text{m/s}$. The wind sequence is shown in figure 3.4. The corresponding output power P_e and the states of the turbine mechanics are shown in figure 3.5. It is seen that the fluctuations are relatively large compared to the stationary values associated with $v_m = 18\text{m/s}$. The maximum values are as much as 50% larger than the nominal values. The controllers in the following chapters will reduce these fluctuations significantly.

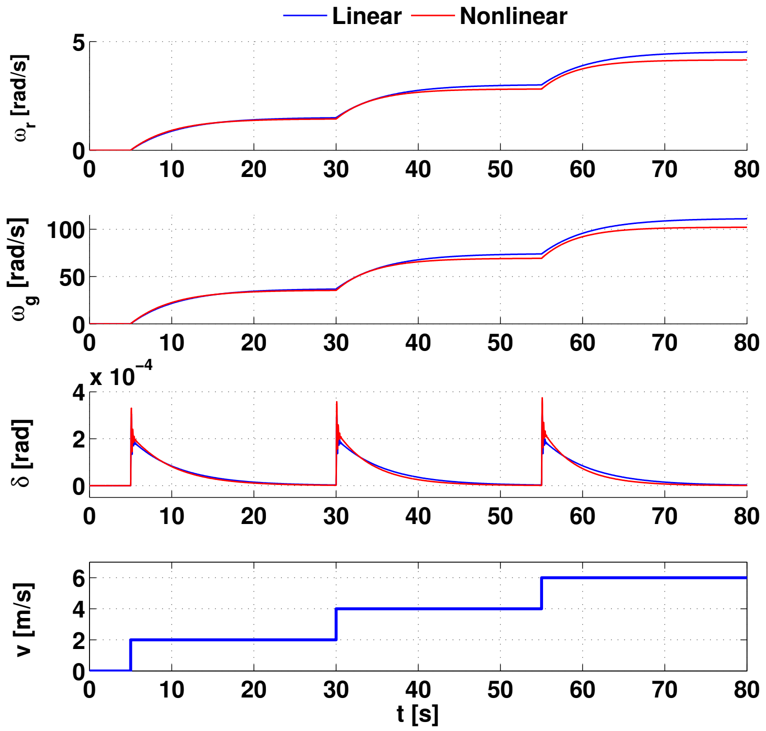


Figure 3.3: Disturbance step response of linearized wind turbine model and nonlinear wind turbine model. The linear model is linearized at $v = 14$m/s. It is seen that the linear system becomes less descriptive as the operating conditions are changed.

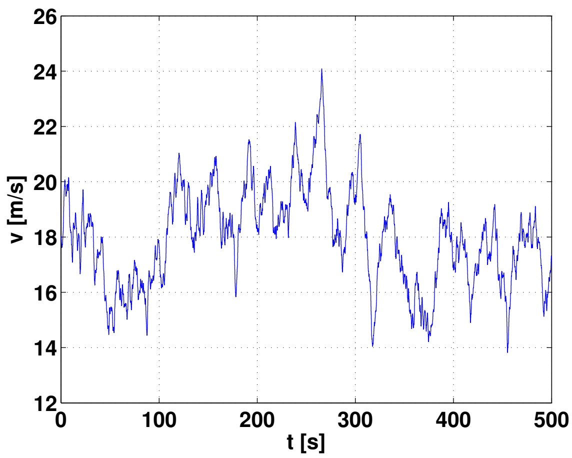


Figure 3.4: Stochastic wind sequence with mean value $v_m = 18$m/s.

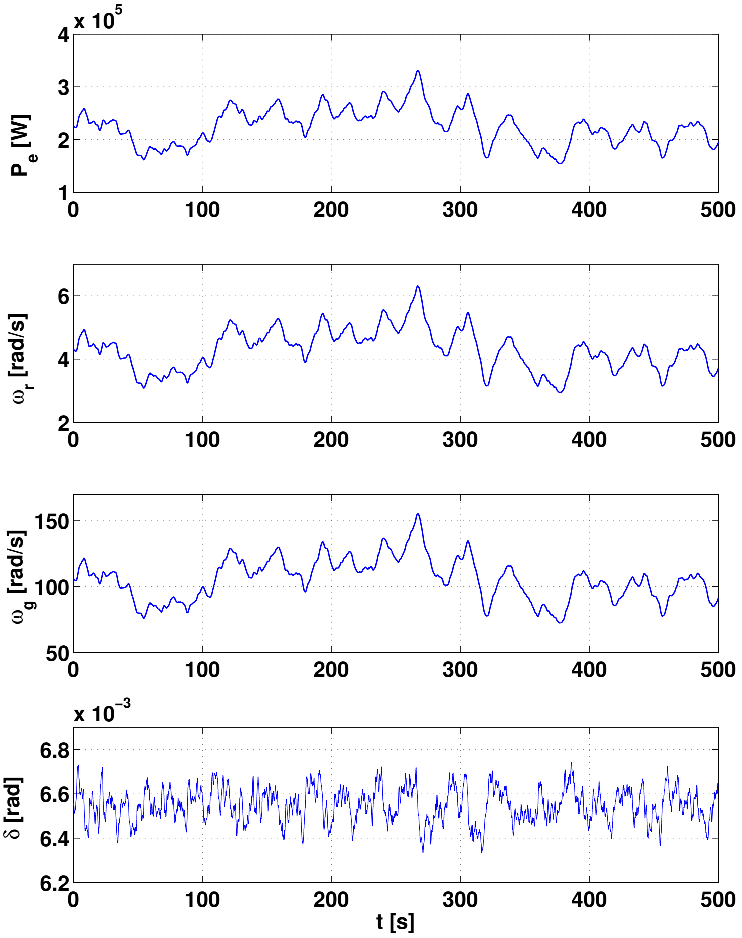


Figure 3.5: Evolution of the output P_e and the states of the turbine mechanics x_m when subject to the stochastic wind sequence in figure 3.4.

Chapter 4

Development environment

The project has involved complex algebraic calculations as well as extensive numerical simulations, therefore the development-platform, used in the project, has not been confined to either an algebraic program nor a numerical computing environment.

Common software environments chosen when dealing with nonlinear control systems are Matlab and Maple. Matlab is widely used in the control theory community due to its wast number of functions related to control theory, and the ease with which a dynamic system can be simulated. Maple is a symbolical calculation software which is used mainly by mathematicians. Dealing with linear control Matlab provides what is necessary. However nonlinear controllers cannot be designed solely using numerical computations. The nonlinear functions make it impossible to generalize the system structures. Algebraic programs such as Maple can, on the other hand, be used for the symbolic calculations needed to design the nonlinear controllers, however numerical simulations cannot be done efficiently.

One of the obstacles in this project, was therefore how to perform the analytical calculations and the numerical computations, without having to transfer data back and forth from one development environment to another. This would slow down the development process significantly.

The choice made was to take advantage of the symbolic toolbox in Matlab which uses the Maple kernel. In this way both the symbolic manipulations and numerical computations can be done on the same platform. However, this does not imply that it is easy to go from the symbolic expressions of nonlinear systems to data structures appropriate for numerical computations. To bridge the gab a specific toolbox has been developed in the project which simplifies analysis, design and simulations. This chapter will shortly describe the toolbox and give an overview of the capabilities provided. For details about the toolbox the reader is referred to the source files on the enclosed CD-rom.

4.1 Nonlinear control toolbox

The Nonlinear control toolbox developed in the project has two main purposes. First and foremost, it bridges the gab between the symbolic functions/system definitions and the numeric simulation environment i.e. it provides export capabilities of the symbolic systems and functions, so that they can be used in the numeric environment. The principle is shown in figure 4.1. Secondly, the toolbox provides various functions

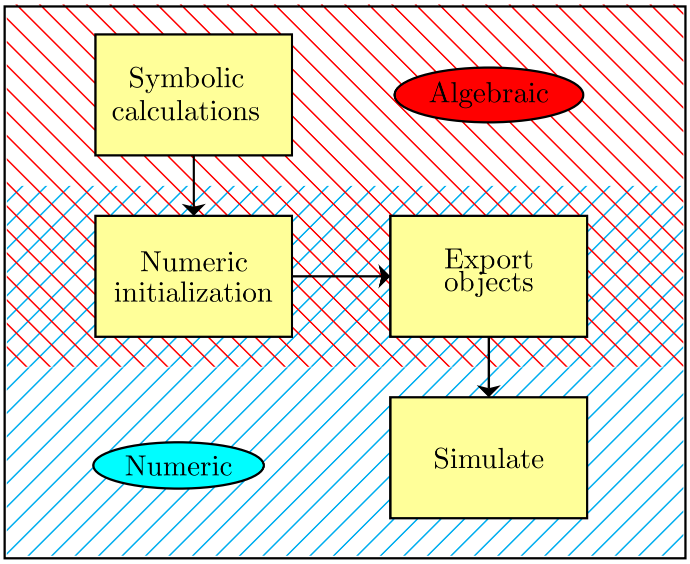


Figure 4.1: Principle of the toolbox. The toolbox bridges the gab between the symbolic expression of nonlinear systems and controllers (top layer) and the numerical environment (bottom layer) by providing export capabilities (middle layer).

for symbolic analysis and design of the nonlinear controllers covered in the project. The design and analysis functions will not be described here. However, the next section will give a brief description of the bridging functionality.

4.1.1 Interfacing algebraic expression and numerical simulations

The bridging functionality of the toolbox has been realized by making a set of objects associated with symbolic definitions of systems, controllers, etc. These (symbolic) objects have member functions that exports the definitions to Matlab functions (m-functions) which can be used in numerical simulations. The toolbox provides the following set of exportable objects: [1]

@fl_sys: System definition.

@fl_diffeo: Diffeomorphism

@fl_CLF: Control Lyapunov Function

@fl_comp: Nonlinear compensation (feedback linearization)

@fl_smc: Sliding mode controller

@fl_sontag: Inverse Optimal Controller (based on Sontag's formula)

@fl_matrix: Matrix definition

When the objects are exported all symbolic parameters used in the expressions are substituted with numerical counterparts. More specifically if an object has a symbolic constant K the export function searches the global variable space for a numeric constant with the same name. Therefore, at the time an export function is executed, the numerical parameters must be defined and stated global.

4.1.2 Simulation

The m-files that are obtained when exporting the objects are simple, in that signals are directly feed through. For example: a system definition, that is exported contains a state equation $\dot{x} = f(x) + G(x)u$ and the observation equation $y = h(x)$. Consequently, using the m-files in Simulink, the user needs to supply a feedback loop with an integrator. This is illustrated in figure 4.2.

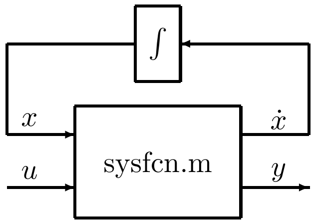


Figure 4.2: Block-diagram showing how dynamic systems created with the non-linear toolbox are included in Simulink.

[1] The at-sign @ is used in Matlab to denote objects. The 'fl' prefix has been used to avoid collision with other functions in Matlab. It is an abbreviation for 'feedback linearization' since the toolbox was develop in connection to feedback linearization originally.

Since system definitions and controller designs etc. are exported to m-files, Simulink has basically only been used as a connection tool between these m-files and for providing dynamics (integrators in feedback loops). As a consequence, block-diagrams are rarely used in this report.

4.2 Toolbox example

This section provides a short example showing how to define and export a nonlinear pendulum system with the toolbox. The system is given by the following state-space description:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -a\sin(x_1) - bx_2 + Ku \\ y &= x_1 \end{aligned}$$

```
%------------------------------------------------------------
%
%    Nonlinear pendulum example
%
%------------------------------------------------------------


%----  Definition of symbolic variables --------

syms a b K x1 x2

%----  System definition -----------------------------

x = [x1; x2];

f = [ x2                    ;
     -a*sin(x1) - b*x2      ];

g = K;

h = x1;

sys = fl_sys(x,f,g,h); % makes fl_sys object

%----  Export system -----------------------------------

global a b K;
a=1; b=2; K=3  % numeric definition with global scope

fl_export('pendulum', sys); % export sys definition
```

Part II

Control designs

CHAPTER 5

Control strategy

The control objective in the top wind speed region was briefly introduced in the introduction. However additional details need to be described, before the controller designs are introduced.

5.1 Wind region

The controllers are designed such that they can operate in wind speeds above 12 m/s. This ensures that the analytic efficiency coefficient, used in the simulations, is well behaved. Wind speeds above 24 m/s will not be considered. Consequently, the controllers are designed for the following wind speed interval:

$$\mathcal{V} = (12; 24) \tag{5.1}$$

The interval is explicitly considered only by the gain scheduled controller.

5.2 Objectives

The primary objective is to minimize the effect of the stochastic wind variations on the output power P_e and the rotational speeds ω_r and ω_g. The nominal values at which they should be stabilized are:

$$P_{e,nom} = 225000 \text{ W} , \qquad \omega_{g,nom} = 105.62 \text{ rad/s} , \qquad \omega_{r,nom} = 4.29 \text{ rad/s}$$

Furthermore the controllers should reduce the stress on the drive train, i.e. minimize the effect on the δ.

These objectives cannot be met at any cost. Naturally there are some restrictions to the pitching motion of the blades i.e. limitations of the angles which can be attained and the speed at which the blades are pitched. The limitations are summarized below:

$$\begin{aligned}\theta_{min} &= -1 \text{ deg}\\ \theta_{max} &= 25 \text{ deg}\\ \left|\dot{\theta}_{max}\right| &= 10 \text{ deg/s}\end{aligned}$$

$\dot{\theta}$ is allowed to peek above ± 10deg/s for short periods of time.

Chapter 6

Gain Scheduling

This chapter describes the design of a gain scheduled controller for the variable speed wind turbine. Based on linear point designs, the purpose of the gain scheduled controller is to provide a benchmark of performance for the nonlinear designs. The method can be used when it is possible to parameterize, how the operating conditions change by one or more variables (scheduling parameters). In such situations one can

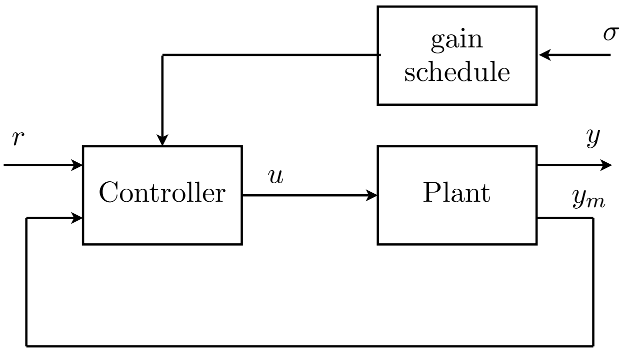


Figure 6.1: General gain scheduling setup. σ denotes the scheduling variable. The scheduler block changes the parameters of the controller according to the current value of σ.

design linear controllers for operating conditions, throughout the region of interest, which are indexed by the scheduling parameters. The family of linear controllers are then implemented as a single controller, whose parameters are changed by monitoring the scheduling variable. This is illustrated in figure 6.1. The method stretches the limits of the basic 'design via linearization' approach which is valid only in a neighborhood of a single operating point.

The outline of the chapter is as follows: Section 6.1 describes the LQ controller which has been used for the individual point designs. Section 6.2 outlines the procedure which has been used to design the gain scheduled controller. Section 6.3 describes the gain scheduling design for the variable speed wind turbine model. Simulations of the closed loop system response, with the gain scheduled controller is shown in section 6.4.

6.1 The LQ controller

This section describes the infinite horizon LQ controller which has been used for the linear point designs. The LQ controller has been chosen, due to the systematic nature with which it can be designed for multi-input systems. Furthermore, the quadratic cost which is to be minimized, is easily designed such that it reflects the control objectives for the wind turbine.

An infinite horizon LQ controller minimizes the quadratic performance index:

$$J = \int_0^\infty \begin{bmatrix} \psi \\ \nu \end{bmatrix}^T \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} \begin{bmatrix} \psi \\ \nu \end{bmatrix} dt \tag{6.1}$$

subject to the state-equation:

$$\dot{\psi} = A\psi + B\nu \tag{6.2}$$

where Q is a weighting matrix, used to manipulate the behavior of the closed loop system. In effect the weighting matrix Q penalize the closed loop system for deviating from the point of linearization.

The control law that minimizes (6.1) can be shown to be the feedback law $\nu = -K\psi$ where K is found through the solution of the continuous algebraic Riccati equation:

$$\begin{aligned} K &= Q_{22}^{-1}(B^T P + Q_{11}^T) && (6.3) \\ 0 &= A^T P + PA + Q_{11} - (PB + Q_{12})Q_{22}^{-1}(B^T P + Q_{12}^T) && (6.4) \end{aligned}$$

Q must be chosen such that the closed loop system has a desired behavior. This is in general an iterative approach, where in each step the Q matrix is changed and system performance is evaluated through simulations.

Choosing Q can be done partly systematic. If there are some additional terms which should be penalized, these can be extracted by making the following partitioning of Q.

$$Q = \begin{bmatrix} C_{\varphi\psi} & C_{\varphi\nu} \end{bmatrix}^T O \begin{bmatrix} C_{\varphi\psi} & C_{\varphi\nu} \end{bmatrix} \tag{6.5}$$

where the $C_{\varphi\psi}$ and $C_{\varphi\nu}$ are linear transformations that extract quantities from ψ and ν. Consequently O is the associated weight on φ.

6.2 Gain scheduling procedure

This section briefly describes the design procedure, which has been used to design the gain scheduled controller. The procedure is stated below:

- Identify a suitable scheduling parameter.
- Linearize the nonlinear model about a family of operating points, resulting in a discrete set of models, indexed by the scheduling parameter.
- Design a family of linear controllers to obtain specified performance in each operating point.
- Interpolate the discrete set of linear controllers to obtain, at least, continuous control or even better a smooth control.

The approach to gain scheduling presented above is *adhoc* in nature and simulations should be used to verify the nonlocal performance of the gain scheduled controller. More analytical approaches to gain scheduling have been formulated based on the so called Linear Parameter Invariant formulation (LPV) and the quasi LPV formulation of a dynamic model (Rugh and Shamma 2000).

6.3 Gain scheduling control of the variable speed model

The gain scheduled controller has been designed such that it covers wind speeds in the wind interval (5.1). This section will describe the choice of scheduling variable, how the family of wind turbine models have been chosen and the corresponding controller designs.

6.3.1 Scheduling parameter

The scheduling variable should be chosen, such that it is descriptive of the current operating conditions in which the wind turbine is situated. Being driven by the wind, the mean wind speed v_m is naturally a descriptive parameter of the current operating conditions. Furthermore the parameters of the wind model (2.27) are changed according to the current mean wind speed. The results obtained in e.g. (Larsen and Mogensen 2006) further motivates choosing v_m, since it is seen that high performance can be obtained, using linear controllers which are designed for the current mean wind speed.

6.3.2 Family of linear wind turbine models

The nonlinear wind augmented model (2.29) has been used as the design model. Including the model of the wind in the design makes it possible for the resulting control design, to take into account the structure of the wind and thereby counteract the evolution of the wind more efficiently.

To reject disturbances and avoid stationary offsets in the power P_e and the rotational speeds ω_g and ω_r the system has been augmented with integrators.

$$\begin{aligned} \int \bar{P}_e dt &= \int (P_e - P_{e,nom}) dt \\ \int \bar{\omega}_g dt &= \int (\omega_g - \omega_{g,nom}) dt \end{aligned}$$

which results in the following nonlinear state space model.

$$\begin{bmatrix} \dot{x} \\ \dot{w} \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} f(x,w) \\ f_i(x) \end{bmatrix} + \begin{bmatrix} G \\ 0 \end{bmatrix} u + \begin{bmatrix} G_w \\ 0 \end{bmatrix} e$$

where $\left[\begin{smallmatrix} \dot{x} \\ \dot{w} \end{smallmatrix} \right] = f(x,w) + Gu + G_w e$ is the wind augmented model (2.29) and:

$$f_i(x) = \begin{bmatrix} x_2 x_5 - P_{e,0} \\ x_2 - \omega_{g,0} \end{bmatrix} = \begin{bmatrix} P_e - P_{e,nom} \\ \omega_g - \omega_{g,nom} \end{bmatrix}$$

The final augmented system is illustrated graphically in figure 6.2. Excluding the

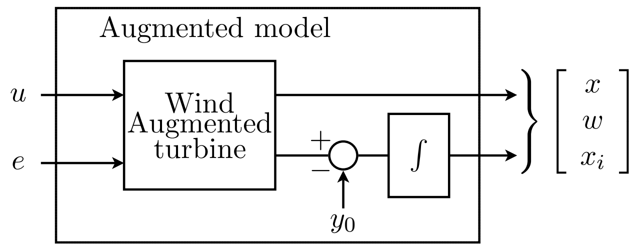


Figure 6.2: The design model for the linear controller. The wind augmented turbine augmented with integral states, such that disturbances are rejected.

disturbance e, the corresponding linear model is:

$$\begin{bmatrix} \dot{\bar{x}} \\ \dot{\bar{w}} \\ \dot{\bar{x}}_i \end{bmatrix} = \begin{bmatrix} A & 0 \\ A_i & 0 \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{w} \\ \bar{x}_i \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} \bar{u} \tag{6.6}$$

where $\left[\begin{smallmatrix}\dot{\bar{x}}\\ \dot{\bar{w}}\end{smallmatrix}\right] = A\left[\begin{smallmatrix}\bar{x}\\ \bar{w}\end{smallmatrix}\right] + B\bar{u}$ is given for arbitrary operating conditions in equation (2.30) and

$$A_i = \left.\frac{\partial f_i}{\partial x}\right|_{x_0} = \begin{bmatrix} 0 & x_{5,0} & 0 & 0 & x_{2,0} & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The indexed family of linear design models are obtained, by calculating the linear model (6.6) for stationary condition corresponding to discrete values of the mean wind speed. The following subset of (5.1) has been chosen.

$$\begin{aligned} \mathcal{S}_{v_m} &= \{v_1,\ v_2\ ,\dots,\ v_j\} \subset \mathcal{V} \\ &= \{12,\ 13,\ \dots,\ 24\} \end{aligned} \tag{6.7}$$

The subsequent spacing between the models and the resolution has been proven appropriate through simulations.

6.3.3 Family of linear controller

An LQ controller has been designed for each of the linear models corresponding to the mean wind speeds in the discrete set (6.7).

By defining $\psi \equiv \left[\begin{smallmatrix}\bar{x}\\ \bar{w}\\ \bar{x}_i\end{smallmatrix}\right]$ and $\nu \equiv \bar{u}$, the linear controllers have been designed such that they minimize the quadratic cost (6.1) subject to the linear state equation (6.6). The terms that are desirable to manipulate by the controller are the physical states of the wind turbine model including the integral states, the inputs and additionally the output P_e and the rate at which the drive shaft twists.

$$\varphi = \begin{bmatrix} \bar{\omega}_r & \bar{\omega}_g & \bar{\delta} & \bar{\theta} & \bar{T}_g & \int \bar{P}_e & \int \bar{\omega}_g & \bar{\theta}_r & \bar{T}_{g,r} & \bar{P}_e & \dot{\bar{\delta}} \end{bmatrix}^T$$

Using the decomposition of the weight Q defined in (6.5) it is possible to extract φ explicitly by the following linear mappings from ψ to φ and ν to φ:

$$C_{\varphi\psi} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & T_{g0} & 0 & 0 & \omega_{g0} & 0 & 0 & 0 & 0 \\ 1 & -N_g^{-1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad C_{\varphi\nu} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Although the wind states are included in the design model, they are deliberately not included in the index since they are uncontrollable

The next step is to find appropriate weights of the variables in the φ vector i.e. to find a reasonable choice of O in equation (6.5). A natural first choice is simply to scale the quantities of φ with their squared stationary values (the time derivatives and integral terms inherits the stationary values of there non-differentiated and non-integrated counterparts).

$$O = \operatorname{diag}\left(\frac{1}{\omega_{r0}} \quad \frac{1}{\omega_{g0}} \quad \frac{1}{\delta_0} \quad \frac{1}{\theta_0} \quad \frac{1}{T_{g0}} \quad \frac{1}{P_{e0}} \quad \frac{1}{\omega_{g0}} \quad \frac{1}{\theta_0} \quad \frac{1}{T_{g0}} \quad \frac{1}{P_{e0}} \quad \frac{1}{\delta_0} \right)^2$$

Until now the approach has been completely systematic. However, the previous choice of O is merely a rule of thumb and O should be adjusted until the closed loop system response is desirable. Acceptable performance has been obtained with the following choice of O:

$$O = \operatorname{diag}\left(\frac{1}{\omega_{r0}} \quad \frac{1}{\omega_{g0}} \quad \frac{50}{\delta_0} \quad \frac{1}{\theta_0} \quad \frac{1}{T_{g0}} \quad \frac{50}{P_{e0}} \quad \frac{1}{\omega_{g0}} \quad \frac{1}{\theta_0} \quad \frac{0.01^2}{T_{g0}} \quad \frac{50}{P_{e0}} \quad \frac{1}{\delta_0} \right)^2$$

This weight has been used for all point designs.

The LQ gain has been calculated for the linear models corresponding to each of the mean wind speeds in the set (6.7):

$$\mathcal{S}_K = \{K_1,\ K_2,\ \ldots,\ K_j\} \quad , \qquad K_i \in \mathbb{R}^{2\times 9} \tag{6.8}$$

Besides the discrete set of LQ gains shown above, there is an additional set of stationary offsets that need to be subtracted/added in the linear controller.

$$\mathcal{S}_X = \{X0_1,\ X0_2\ \ldots,\ X0_j\} \quad , \quad X0_i \in \mathbb{R}^9 \tag{6.9}$$
$$\mathcal{S}_U = \{U0_1,\ U0_2\ \ldots,\ U0_j\} \quad , \quad U0_i \in \mathbb{R}^2 \tag{6.10}$$

As was seen in chapter 3 only the stationary values associated with θ and θ_r changes when the mean wind change. Consequently only a subset of $\mathcal{S}_X$ and $\mathcal{S}_U$ needs to be scheduled.

6.3.4 Controller interpolation

To obtain the final gain scheduled controller the discrete set of controller parameters (6.8)-(6.10) needs to be interpolated according to the mean wind speed. The interpolation method which has been used in this project, is a simple linear interpolation. Figure 6.3 illustrates the setup of the controller.

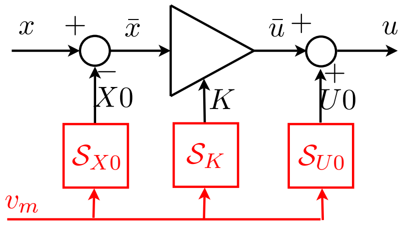


Figure 6.3: Gain scheduled controller. The parameters of the gain and stationary values are interpolated according to the scheduling parameter v_m.

6.4 Simulations

This section illustrates the closed loop performance of the wind turbine with the gain scheduled controller. Two types of simulations are presented:

- Control with constant mean wind speed
- Gain scheduling versus point design with drifting mean wind speed.

The first simulation illustrate the effeciency of the linear design in itself, since the mean wind speed is not the varied. The second simulation illustrates the advantage of using the gain scheduled controller as compared to a point design.

6.4.1 Closed loop response with constant mean wind speed

The wind sequence shown in figure 6.4 with mean wind speed $v_m = 18\text{m/s}$ has been used in the simulation. Figure 6.5 shows the corresponding state evolution of the closed loop system and figure 6.6(a) shows the output P_e. The pitch speed is seen in figure 6.6(b).

Compared to the open-loop response in figure 3.5, it is verified that the controller reduces the fluctuations of the output P_e and the states associated with the wind turbine mechanics (ω_r, ω_g and δ). It is seen that the controller does not compromise the physical limits of the actuator.

6.4.2 Gain scheduling versus point design

To check that an improvement has been gained with the gain scheduled controller, its performance is compared to a point design. The point design is designed for the

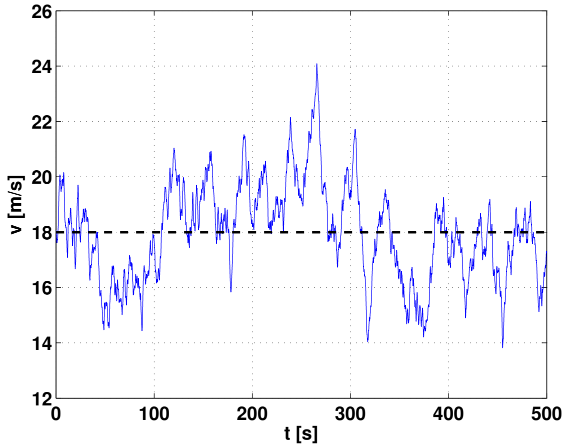


Figure 6.4: Wind speed. $v_m = 18\text{m/s}$

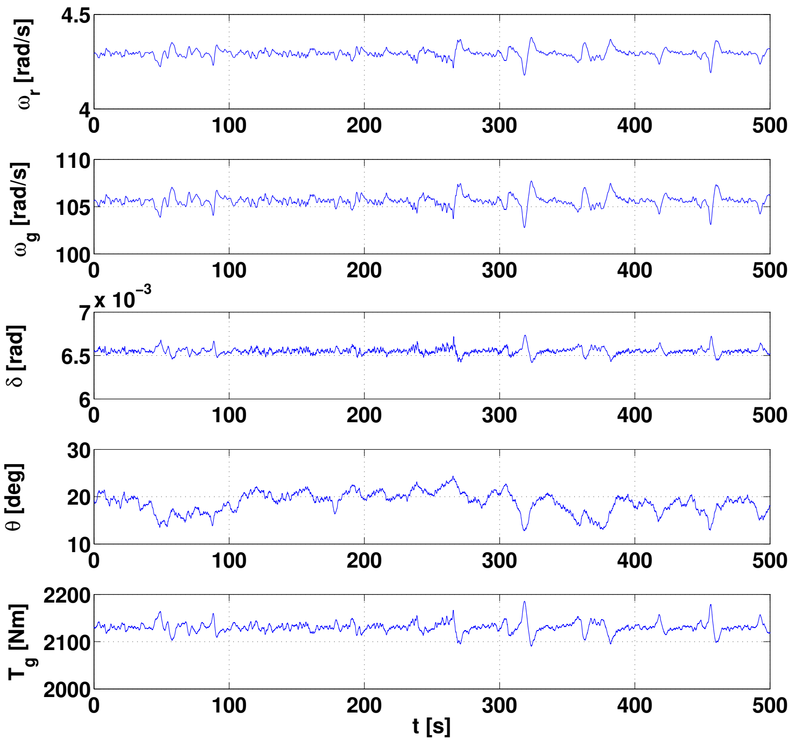
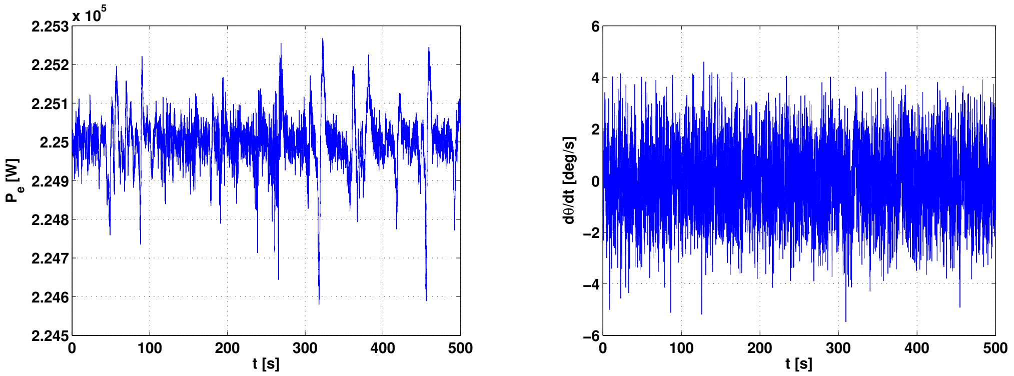


Figure 6.5: State evolution of closed loop system when subject to the stochastic wind in figure 6.4.



(a) P_e (b) $\dot{\theta}$

Figure 6.6: Generated power P_e and the pitching speed $\dot{\theta}$ associated with the state evolution in figure 6.5.

mean wind speed $v_m = 16\text{m/s}$. In the simulations the mean wind speed is changed over a period of 1500 seconds from 16 to 19 m/s. Figure 6.7 shows the stochastic wind sequence. Figure 6.8 shows the state evolution of the gain scheduled controller

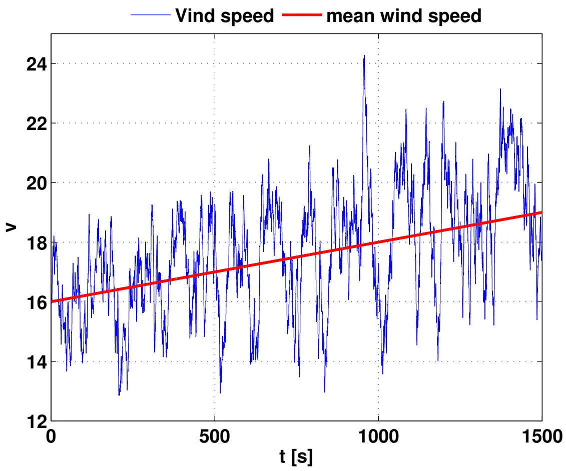


Figure 6.7: Wind sequence with mean wind speed increased linearly from 16m/s to 19m/s3 over a period of 1500 seconds.

and point design. Figure 6.9 shows the corresponding output P_e and the speed of the pitching motion. It is seen that the closed loop response with the point design is degraded as the operating condition change from the point of linearization. The gain scheduled controller reconfigures as the conditions change, and is therefore seemingly unaffected by the changing conditions.

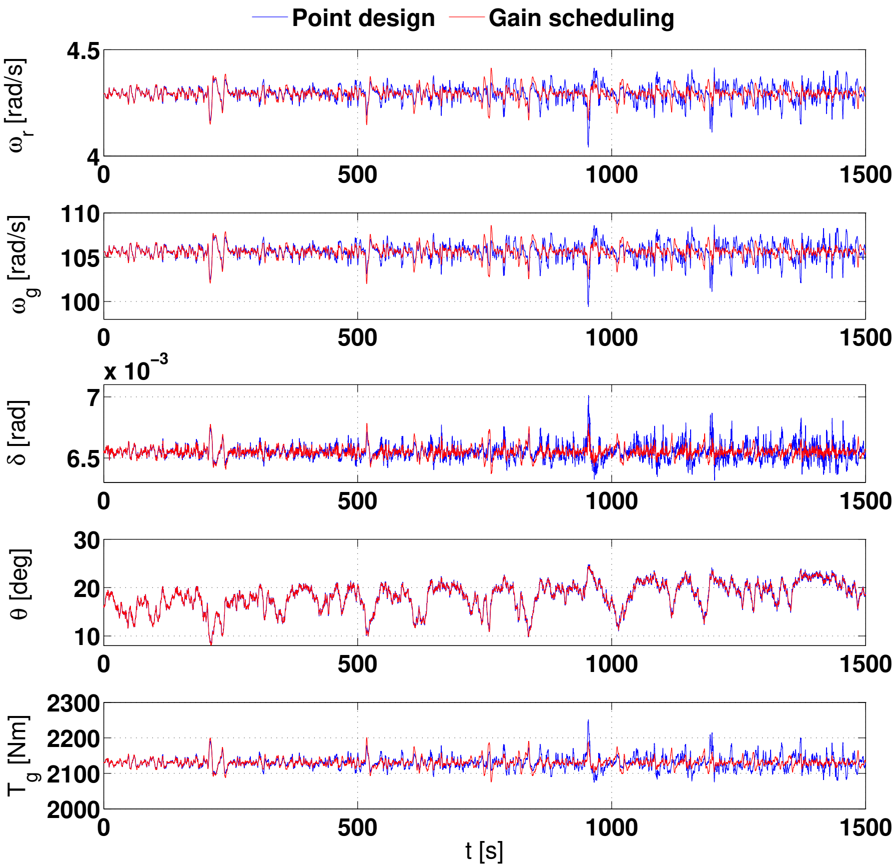
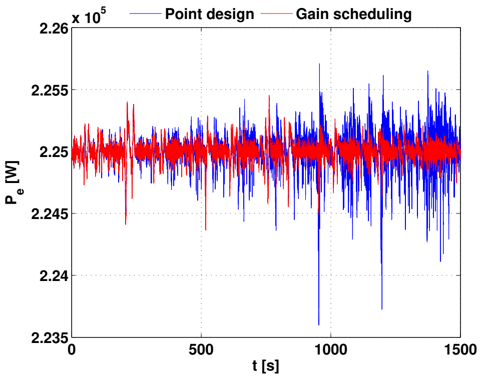
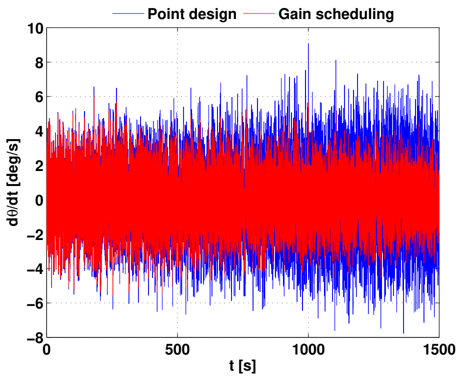


Figure 6.8: State evolution associated with the gain scheduled controller and the linear point design ($v_m = 16$m/s).



(a) P_e



(b) $\dot{\theta}$

Figure 6.9: Evolution of P_e and $\dot{\theta}$.

Chapter 7

Feedback linearization

This chapter will analyze the possibility of controlling the wind turbine using feedback linearization theory. Feedback linearization theory provides methods that, through feedback, cancels the nonlinearities of the system. A controller can then be designed for the system using the powerful linear design tools. The theory can be divided into two parts, namely:

- Input-Output linearization.
- Input-state linearization.

Input-Output linearization is very easy to obtain and requires little more than differentiating the output a number of times. However, this may result in internal dynamics which cannot be controlled from an input-output point of view.

Input-state linearization is not generally possible with a given system. Whether it is possible, can be verified by solving a set of partial differential equations. If the solution exists, a state transformation and a linearizing feedback can be found, such that the transformed system is rendered linear.

The methods are systematic and relatively simple, however the validity of the approach relies on exact knowledge of the physical system at hand. Robustness and effectiveness of the resulting control are hence important factors which should be analyzed, analytically if possible, or simply through extensive simulations.

Only input-state linearization will be applied to the wind turbine. However, input-state linearization is a natural extension of input-output linearization, therefore both methods will be covered in the theoretical presentation. The following presentation is meant to be intuitive with a mathematical level which is limited to what is needed to understand the methods in connection to the wind turbine. To reduce the overhead the single-input theory will be introduced first and applied to the constant speed wind turbine. The more complex multi-input theory will be applied to the variable speed turbine.

The outline of the chapter is as follows: Feedback linearization for single-input systems is introduced in Section 7.1. Section 7.2 shows how the theory can be applied to the constant speed turbine. Section 7.3 extends the theory to multi-input systems. In section 7.4 the theory is applied to the variable speed turbine, taking the stochastic nature of the wind into account. Simulations of the variable speed wind turbine with the feedback linearizing controller is presented in section 7.5.

7.1 SISO Feedback linearization

This section gives a short introduction to the most essential concepts regarding single input feedback linearization. Input-Output linearization will be presented first to provide the basis for which input-state linearization can be introduced. The presentation is based on (Khalil 2002).

The theory applies to nonlinear systems which are affine in the input:

$$\begin{aligned} \dot{x} &= f(x) + G(x)u && (7.1) \\ y &= h(x) && (7.2) \end{aligned}$$

For the following discussion it is necessary to introduce the *Lie derivative* which is a mathematical operator used extensively, to reduce the notational burden. The Lie derivative of h with respect to f, is simply the directional derivative of h along the direction of the vector f.

$$L_f h \equiv \frac{\partial h(x)}{\partial x} f \tag{7.3}$$

When e.g. differentiating the output $h(x)$ of a system n times with respect to time, it can be formalized by recursive use of the Lie derivative. For example differentiating the output of an unforced system can be written in the following manner:

$$\begin{aligned} \dot{y} &= \frac{\partial h(x)}{\partial x} f(x) = L_f h(x) \\ \ddot{y} &= L_f^2 h(x) \\ &\vdots \\ y^{(n)} &= L_f^n h(x) \end{aligned}$$

7.1.1 Input-Output linearization

As mentioned earlier the purpose of input-output linearization is to establish a linear relation between input and output. The means of doing so, is first to establish a direct nonlinear relation between the input and output. This is easily obtained by differentiating the output until the input occurs. Differentiating the output of the affine system (7.1)-(7.2) results in:

$$\dot{y} = \frac{\partial h}{\partial x}(f(x) + G(x)u) = L_f h(x) + L_g h(x)u$$

If $L_g h(x) = 0$, the system needs to be differentiated further to obtain a direct relation between the input and the output. If the input appears after r differentiations, the following relation is obtained:

$$\begin{aligned} y &= h(x) \\ \dot{y} &= \frac{\partial h}{\partial x}(f(x) + G(x)u) = L_f h(x) \\ \ddot{y} &= L_f^2 h(x) \\ &\vdots \\ y^{(r)} &= L_f^r h(x) + L_G L_f^{r-1} h(x)u \end{aligned}$$

It is seen that the following control law will linearize the input-output description of the system:

$$\begin{aligned} u &= \alpha(x) + \beta(x)\upsilon \\ &= -\frac{L_f^r h(x)}{L_G L_f^{r-1} h(x)} + \frac{1}{L_G L_f^{r-1} h(x)}\upsilon \end{aligned}$$

rendering the system a cascade of integrators

$$y^{(r)} = \upsilon$$

which is trivially controlled using linear control theory. If the number of differentiations r is smaller than the order of the nonlinear system n, the resulting closed loop system will have internal dynamics. This will be illustrated shortly by transforming the system (7.1)-(7.2) to the so-called normal form. r is consequently called the relative degree of the input-output system. From the previous derivation of the linearizing control law it is seen that the relative degree is given by the smallest integer r such that:

$$L_g L_f^{r-1} h(x) \neq 0$$

To illustrate the concept of internal dynamic, the so-called normal form needs to be introduced. Therefore, without loss of generality, it is assumed that the affine system

(7.1)-(7.2) with relative degree r has been transformed to the normal form:

$$\begin{aligned}
\dot{\eta} &= \zeta(\eta, z, u) \\
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_r &= f_c(x) + G_c(x)u \\
y &= h(x) = z_1
\end{aligned}$$

where

$$f_c(x) = L_f^r h(x) \quad , \qquad G_c(x) = L_g L_f^{r-1} h(x)$$

The normal form can be written more compactly

$$\begin{aligned}
\dot{\eta} &= \zeta(\eta, z, u) && (7.4) \\
\dot{z} &= A_c z + B_c(f_c(x) + G_c(x)u) && (7.5) \\
y &= C_c z && (7.6)
\end{aligned}$$

where (A_c, B_c, C_c) is a canonical form representation of r integrators ie.:

$$A_c = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}_{r \times r} \qquad B_c = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}_{r \times 1} \qquad C_c = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}^T_{1 \times r}$$

and $z \in \mathbb{R}^r$ and $\eta \in \mathbb{R}^{n-r}$. The characteristic of the normal form is the direct relation between the input and the output i.e. they are related through a cascade of integrators. Hence the normal form is associated with a specific choice $y = h(x) = z_1$. The transformation to the normal form is given by

$$T_c(x) = \begin{bmatrix} \phi(x) \\ --- \\ h(x) \\ L_f h(x) \\ \vdots \\ L_f^{r-1} h(x) \end{bmatrix} = \begin{bmatrix} \eta \\ --- \\ z \end{bmatrix} \tag{7.7}$$

where ϕ should be chosen such that the $T_c(x)$ is a diffeomorphism, which essentially means that the transformation is invertible. Choosing the input-output linearizing feedback:

$$u = -G_c(x)^{-1} f_c(x) + G_c(x)^{-1} \upsilon \tag{7.8}$$

clearly illustrates the inability to control the dynamics $\dot{\eta} = \zeta(\eta, z, u)$ from an input-output point of view. Accordingly they are called internal dynamics. One consequently needs to analyze the stability of the internal dynamics when input-output linearizing a system. If the diffeomorphic transformation renders the internal dynamics independent of the input[1] ie.

$$\dot{\eta} = \zeta_0(\eta, z)$$

stability of the internal dynamics can be found by analyzing stability of $\dot{\eta} = \zeta_0(\eta, 0)$ which are called zero dynamics. A nonlinear system with unstable zero dynamics is analog to a linear system with zeros in the right half plane. Consequently a nonlinear system with unstable zero dynamics is called non-minimum phase and minimum phase if the zero dynamics is stable.

As seen above, the normal form is associated with a given choice of output function. Choosing another output function yields another normal form. Therefore, one might obtain an input-output relation without internal dynamics by choosing a different output function. The problem of choosing an output equation $h(x)$ which yields relative degree $r = n$ is actually the same as finding the diffeomorphic transformation that leads to input-state linearization of the system. A problem that does not necessarily hold a solution. This will be clarified in the next section.

7.1.2 Input-state linearization

Having introduced input-output linearization it is straight forward to extend the theory to input-state linearization. In input-state linearization the purpose is to obtain a linear state space description, rather than a linear input-output description. As implied above full-state linearization is obtained by choosing the output function in a cleaver way, such that the relative degree of the system equals the order of the system. Consequently the system will be rendered linear in the transformed coordinates, by the input-output linearizing control (7.8). Equivalently, this corresponds to finding a solution $h(x)$ to the set of partial differential equations:

$$L_g L_f^{i-1} h(x) = 0 \ , \quad i = 1, 2, \ldots, n-1 \tag{7.9}$$

subject to the restriction

$$L_g L_f^{n-1} h(x) \neq 0 \tag{7.10}$$

naturally the output function does not need to be the actual output of the system but simply a “dummy output” that leeds to full state linearization. A solution does not exist in general, but a solution can be shown to exist if certain conditions are fulfilled (Khalil 2002). These conditions can be seen as an extension of the controllability conditions for linear systems.

Fortunately it is not always nessesary to solve the partial differential equations needed to find the solution $h(x)$. The cleaver choice of $h(x)$ might simply be obtained through physical insight in the system at hand or simply by guessing.

[1] The diffeomorphic transformation renders the internal dynamics independent of the u, by imposing the restriction $\frac{\partial \phi_i}{\partial x} g(x) = 0$, $i = 1, 2, \ldots, n-r$

From the presentation above it is not clear when to choose input-output linearization and when to choose input-state linearization. If the objective is to track a given reference, the output will be given on beforehand, and consequently input-output linearization is appropriate. If the objective is stabilization it is natural to seek the transformation, from which the entire state vector can be linearized, thereby obtaining larger flexibility in the control.

A last note about diffeomorphisms is necessary for the next sections. That a transformation $z = T(x)$ is a diffeomorphism *locally* in a neighbourhood of a point x^* in state space can be seen by checking that the jacobian:

$$\left.\frac{\partial T(x)}{\partial x}\right|_{x=x^*} \tag{7.11}$$

is nonsingular[2].

7.2 Preliminary investigation on the constant speed model

This section will briefly show how to input-state linearize the constant speed turbine. This will serve as an example of the previously introduced SISO theory while providing the basis for feedback linearizing the variable speed wind turbine.

The nonlinear state space description is repeated here for convenience:

$$f(x) = \begin{bmatrix} \frac{P_r}{x_1 J} - \frac{x_1 D_g}{J} + \frac{D_g \omega_0}{Jp} \\ -\tau_\theta^{-1} x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \tau_\theta^{-1} \end{bmatrix} u$$

To feedback linearize the system, the output that yields relative degree $r = n = 2$ needs to be found. It is easily seen that choosing the dummy output x_1, requires that the output is differentiated twice to obtain a direct relation between the input and the output. Equivalently x_1 satisfies the partial differential equation (7.9) under the restriction (7.10):

$$\begin{aligned} L_g x_1 &= 0 \\ L_g L_f h x_1 &= \frac{\partial P_r}{\partial x_2} \frac{D_g \omega_0}{x_1 p J \tau_\theta} \end{aligned}$$

The transformation that brings the system into the normal form is accordingly:

$$z = T_c(x) = \begin{bmatrix} x_1 \\ L_f x_1 \end{bmatrix} = \begin{bmatrix} x_1 \\ \frac{P_r}{x_1 J} - \frac{x_1 D_g}{J} + \frac{D_g \omega_0}{Jp} \end{bmatrix} = \begin{bmatrix} \omega_g \\ \dot{\omega}_g \end{bmatrix}$$

[2] A diffeomorphism is global if the Jacobian is nonsingular for all $x \in R^n$ and $\lim_{x\to\infty} ||T(x)|| = \infty$.

The associated normal form is:

$$\begin{aligned}\dot{z}_1 &= z_2\\ \dot{z}_2 &= f_c(x) + G_c(x)u\end{aligned}$$

where

$$\begin{aligned}f_c(x) &= L_f^2 x_1\\ &= \left(\frac{\frac{\partial}{\partial x_1}P_r(x_1,x_2)}{x_1 J} - \frac{P_r(x_1,x_2)}{x_1^2 J} - \frac{D_g}{J}\right)\cdot\\ &\quad \left(\frac{P_r(x_1,x_2)}{x_1 J} - \frac{x_1 D_g}{J} + \frac{D_g\omega_0}{Jp}\right) - \frac{\frac{\partial}{\partial x_2}P_r(x_1,x_2)x_2}{x_1 J\tau_\theta} && (7.12)\\ G_c(x) &= L_g L_f x_1\\ &= \frac{\frac{\partial}{\partial x_2}P_r(x_1,x_2)}{x_1 J\tau_\theta} && (7.13)\end{aligned}$$

The feedback linearizing control is accordingly given by:

$$u = -G_c^{-1}(x)f_c(x) + G_c^{-1}(x)\upsilon \tag{7.14}$$

which renders the system a cascade of two integrators:

$$\begin{aligned}\dot{z} &= A_c z + B_c\upsilon\\ &= \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix} z + \begin{bmatrix}0\\ 1\end{bmatrix}\upsilon\end{aligned}$$

To check the validity of the resulting control (7.14) (and the implementation) a pole placement controller has been designed such that the poles of the feedback linearized closed loop system are placed at:

$$p = -2 \pm 8i \tag{7.15}$$

The design and simulation has been made with a constant wind disturbance $v =$ 18m/s. Figure 7.1 shows the trajectory of the closed loop system in the linearized coordinates z, when the initial state of the system differs from the equilibrium point: $x_0 - x^* = [-0.1\ 2]^T$. The gradient field of a system with eigenvalues (7.15) is included in the figure. It is seen that the trajectory of the closed loop system follows the gradient field. Figure 7.2 shows the control signals for the entire nonlinear system system $u = \theta_{ref}$ and the control signal for the feedback linearized system $\upsilon = \dot{\omega}_{g,r}$.

Evaluating the Jacobian of $T_c(x)$ (see equation (7.11)) it has been verified that $T_c(x)$ is a diffeomorphism, in a neighborhood of equilibrium points, throughout the top wind speed region. However, an analytical validation for $T_c(x)$ throughout the region of interest, is complicated by the intricate analytical expression of c_p. Instead an intuitive justification will be given. As mentioned in chapter 2 the nominal power

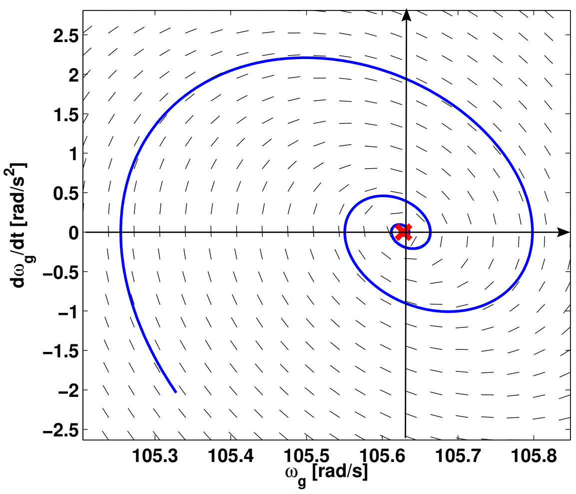


Figure 7.1: Trajectory of feedback linearized system. The trajectory follows the gradient field of a system with poles: $p = -2 \pm 8i$.

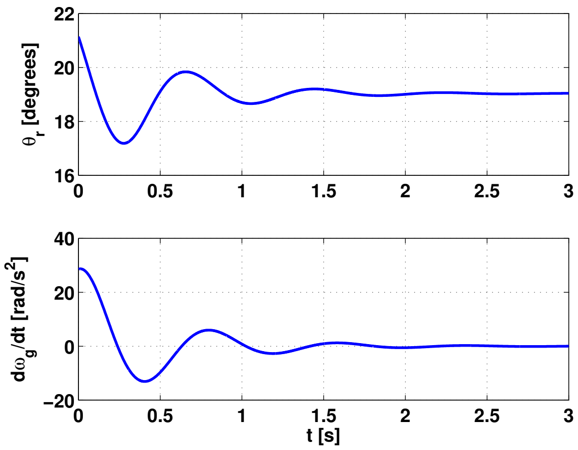


Figure 7.2: Control signal for the nonlinear system, and the feedback linearized system respectively.

$P_{e,nom} = 225kW$ can both be extracted at negative and positive pitch angles when working in wind speeds above approx. 12 m/s (See figure 2.6). Consequently the transformation from $x_2 = \theta$ to $z_2 = \dot{\omega}_g$ is not globally invertible and therefore not a global diffeomorphism. However, the pitch will always be above zero to limit the power extraction to 225 kW. Hence, it is proposed that the transformation is a diffeomorphism throughout the region of interest. As will be seen in the next sections, the discussion above will apply equally, to the transformation associated with variable speed turbine.

7.3 MIMO feedback linearization

The theory is now extended to multi-input systems so that the variable speed wind turbine can be feedback linearized. The review of MIMO feedback linearization will in this report be limited to systems with an equal number of inputs and outputs (as is the case in most literature). Furthermore only basic MIMO linearization will be considered in which the so-called decoupling matrix is nonsingular. The review is based on (Slotine and Li 1991) and (Isidori 1995). As was the case for the SISO theory, input-output linearization is presented first, to provide the basis for understanding input-state linearization.

7.3.1 Input-output linearization

Similar to SISO input-output linearization, the objective is to find the functions $\alpha(x)$ and $\beta(x)$ which through multiplication and addition linearizes the input-output description. For SISO systems the functions could be found knowing the relative degree of the system. This also goes for MIMO system, but the concept of relative degree needs to be extended to MIMO systems.

A MIMO system has a relative degree associated with each of its outputs $h_i(x)$. These relative degrees form the so-called vector relative degree:

$$r = \begin{bmatrix} r_1 & r_2 & \dots & r_m \end{bmatrix}$$

The total relative degree is found by summing the relative degrees. A system has no internal dynamics, if the total relative degree equals the order of the system. Just as for SISO systems this results in a input-state feedback linearizable system.

A relative degree associated with an output $h_i(x)$ is equal to the number of times the output can be differentiated until at least *one* of the inputs appears. More specific, the relative degree r_i associated with $h_i(x)$ is the smallest integer so that:

$$L_{g_j} L_f^{r_i-1} h_i(x) \neq 0 \tag{7.16}$$

for at least one j. The input-output relations that are obtained when performing these differentiation are given by:

$$\begin{aligned}
y_1^{(r_1)} &= f_1(x) + G_{11}(x)u_1 + G_{12}(x)u_2 + \cdots + G_{1m}(x)u_m \\
y_2^{(r_2)} &= f_2(x) + G_{21}(x)u_1 + G_{22}(x)u_2 + \cdots + G_{2m}(x)u_m \\
&\vdots \\
y_m^{(r_m)} &= f_m(x) + G_{m1}(x)u_1 + G_{m2}(x)u_2 + \cdots + G_{mm}(x)u_m
\end{aligned}$$

Introducing the following functions:

$$\begin{aligned}
f_c(x) &= \begin{bmatrix} L_f^{r_1} h_1(x) \\ \vdots \\ L_f^{r_m} h_m(x) \end{bmatrix} \\
G_c(x) &= \begin{bmatrix} L_{g_1} L_f^{r_1-1} h_1(x) & \dots & L_{g_m} L_f^{r_1-1} h_1(x) \\ \vdots & \ddots & \vdots \\ L_{g_1} L_f^{r_m-1} h_m(x) & \dots & L_{g_m} L_f^{r_m-1} h_m(x) \end{bmatrix}
\end{aligned}$$

It is easily seen that the following control law linearizes the input output description:

$$\begin{aligned}
u &= \alpha(x) + \beta(x)v \\
&= -G_c(x)^{-1} f_c(x) + G_c(x)^{-1} v
\end{aligned} \tag{7.17}$$

This choice is actually a decoupling control since it renders the channels through the system independent, hence the input-output map will be given by:

$$y = \begin{bmatrix} 1/s^{r_1} & 0 & \dots & 0 \\ 0 & 1/s^{r_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1/s^{r_m} \end{bmatrix} v \tag{7.18}$$

where s is the Laplace operator. If the total relative degree of the system is less than the order of the system, there will be internal dynamics. Just as for SISO systems this is best visualized by transforming the system to the normal form. For multivariable systems the normal form is given by:

$$\begin{aligned}
\dot{\eta} &= \zeta(\eta, z, u) \\
\dot{z} &= \underbrace{\begin{bmatrix} A_c^1 & 0 & \cdots & 0 \\ 0 & A_c^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_c^m \end{bmatrix}}_{A_c} z + \underbrace{\begin{bmatrix} B_c^1 & 0 & \cdots & 0 \\ 0 & B_c^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & B_c^m \end{bmatrix}}_{B_c} (f_c(x) + G_c(x)u) \\
y &= \underbrace{\begin{bmatrix} C_c^1 & 0 & \cdots & 0 \\ 0 & C_c^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C_c^m \end{bmatrix}}_{C_c} z
\end{aligned}$$

where (A_c^i, B_c^i, C_c^i) is a canonical form representations of r_i integrators.

It is seen that choosing the control law (7.17) will render the sub-dynamics $\dot{\eta} = \zeta(\eta, z, u)$ uncontrollable from an input-output point of view.

The transformation that brings the system into the normal form is given by:

$$T_c(x) = \begin{bmatrix} \phi(x) \\ --- \\ T_c^1(x) \\ T_c^2(x) \\ \vdots \\ T_c^i(x) \end{bmatrix} \tag{7.19}$$

where

$$T_c^i = \begin{bmatrix} h_i(x) \\ L_f h_i(x) \\ \vdots \\ L_f^{r_i-1} h_i(x) \end{bmatrix} = \begin{bmatrix} z_1^i \\ z_2^i \\ \vdots \\ z_{r_i}^i \end{bmatrix}$$

$\phi(x)$ is of course chosen such that $T_c(x)$ is a diffeomorphism.

7.3.2 Input-state linearization

As was the case for SISO systems, input-state linearization amounts to finding dummy outputs so that the transformation (7.19) renders the system without internal dynamics. I.e. finding the solutions $h_1(x), \dots, h_m(x)$ to the equations:

$$L_{g_i} L_f^k h_i(x) = 0 \quad \text{forall} \quad 0 \le k \le r_i - 2, \quad 1 \le j \le m$$

with the restriction that the total relative degree equals the order of the system $r_1 + r_2 + \ldots + r_i = n$.

As for single-input systems, the solution might simply be obtained by making a qualified guess.

7.4 Feedback linearizing control of the variable speed model

The ultimate goal is to obtain full state linearization of the variable speed wind turbine. This section will show that this is possible and that the linearized state vector is physically meaningful and appropriate for controlling the turbine. The section is organized such that the 'deterministic' problem is considered first i.e. the wind speed is assumed constant and consequently considered a static parameter in the design. Having solved the input-state linearizing problem for the deterministic scheme, it is shown how the disturbance can be decoupled from the final design.

The variable speed wind turbine has two inputs, hence two dummy outputs need to be determined. These outputs must be chosen such that the relative degree of the system equals the system order. A simple choice is to check whether a combination of the states results in the total relative degree $r = n = 5$. Table 7.1 shows the relative degree associated with each of the states. Naturally the physical output of the turbine

h(x)	Relative degree
$x_1 = \omega_r$	2
$x_2 = \omega_g$	2
$x_3 = \delta$	3
$x_4 = \theta$	1
$x_5 = T_g$	1

Table 7.1: Relative degree associated with the states as dummy outputs

P_e only attains a relative degree $r = 1$ since it is a function of T_g. Therefore using P_e as one dummy output, would require that another dummy output with relative degree $r = 4$ should be found. Considering the structure of the variable speed model, it is not obvious how this should be chosen and it might not even exist. Hence, in the interest of attaining full input-state feedback linearization, the dummy outputs should either be $h(x) = [x_1\ x_3]^T$ or $h(x) = [x_2\ x_3]^T$. The relative degree vector associated with both choices is:

$$r_v = \begin{bmatrix} 2 & 3 \end{bmatrix}$$

the state vectors which are associated with the corresponding normal forms are actually related simply by a (non-singular) linear transformation, which is easily seen by noting that $\dot{x}_3$ is a linear combination of x_1 and x_2.

Choosing $h(x) = [x_2\ x_3]^T$ the diffeomorphic transformation to the normal form is given by:

$$\begin{aligned} T_c(x) &= \begin{bmatrix} \omega_g \\ \dot{\omega}_g \\ \delta \\ \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} x_2 \\ L_f x_2 \\ x_3 \\ L_f x_3 \\ L_f^2 x_3 \end{bmatrix} \\ &= \begin{bmatrix} x_2 \\ \frac{x_1 D_s}{J_g N_g} - \frac{x_2 D_s}{N_g^2 J_g} + \frac{x_3 K_s}{J_g N_g} - \frac{x_5}{J_g} \\ x_3 \\ x_1 - \frac{x_2}{N_g} \\ \frac{P_r}{x_1 J_r} - \frac{x_1 D_s}{J_r} + \frac{x_2 D_s}{J_r N_g} - \frac{x_3 K_s}{J_r} - \frac{\frac{x_1 D_s}{J_g N_g} - \frac{x_2 D_s}{N_g^2 J_g} + \frac{x_3 K_s}{J_g N_g} - \frac{x_5}{J_g}}{N_g} \end{bmatrix} \end{aligned} \tag{7.20}$$

The associated normal form is:

$$\dot{z} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{A_c} z + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}}_{B_c} (f_c(x) + G_c(x)u)$$

where $z = T_c(x)$ and

$$\begin{aligned} f_c(x) &= \begin{bmatrix} L_f^2 h_1(x) \\ L_f^3 h_2(x) \end{bmatrix} \\ &= \begin{bmatrix} \frac{f_1(x)D_s}{J_g N_g} - \frac{f_2(x)D_s}{N_g^2 J_g} + \frac{f_3(x)K_s}{N_g J_g} - \frac{f_5(x)}{J_g} \\ \hdashline \frac{P_r(f_1(x),f_4(x))}{f_1(x)J_r} - \frac{f_1(x)D_s}{J_r} + \frac{f_2(x)D_s}{J_r N_g} - \frac{f_3(x)K_s}{J_r} \\ -\frac{\frac{f_1(x)D_s}{J_g N_g} - \frac{f_2(x)D_s}{N_g^2 J_g} + \frac{f_3(x)K_s}{N_g J_g} - \frac{f_5(x)}{J_g}}{N_g} \end{bmatrix} \end{aligned} \tag{7.21}$$

and

$$\begin{aligned} G_c(x) &= \begin{bmatrix} L_{g_1} L_f x_2(x) & L_{g_2} L_f x_2(x) \\ L_{g_1} L_f x_3(x) & L_{g_2} L_f x_3(x) \end{bmatrix} \\ &= \begin{bmatrix} 0 & -\frac{1}{J_g \tau_T} \\ \frac{\partial P_r(x_1,x_2)}{x_1 J_r \tau_\theta} & \frac{1}{N_g J_g \tau_T} \end{bmatrix} \end{aligned} \tag{7.22}$$

The feedback linearizing feedback should hence have the following structure:

$$u = \alpha(x) + \beta(x)\upsilon \tag{7.23}$$

$$= -G_c(x)^{-1} f_c(x) + G_c(x)^{-1}\upsilon \tag{7.24}$$

where υ is the new input. The feedback linearized system is simply a cascade of integrators:

$$\begin{aligned} \dot{z} &= A_c z + B_c \upsilon \\ &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} z + \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \upsilon \end{aligned}$$

Although it is easy to design a *stabilizing* control law for the feedback linearized system above, it is not obvious how to design the control law such that the closed loop system has desirable performance. The system is basically unrecognizable after the nonlinear transformation and feedback linearizing component. All the cross-couplings have been removed, and the new state vector z does not explicitly supply the physical insight into the system behavior, as the original x. An alternative transformation has therefore been found which is presented in the next section.

7.4.1 Alternative transformation

Input-state linearization is not restricted to systems on the normal form but applies to systems which can be transformed to the more general form (Khalil 2002):

$$\dot{\xi} = A_\xi \xi + B_\xi(f_\xi(x) + G_\xi(x)u)$$

where $f_\xi(x) : \mathbb{R}^n \to \mathbb{R}^m,\ G_\xi(x) : \mathbb{R}^n \to \mathbb{R}^{m\times m},\ u \in \mathbb{R}^m,\ x \in \mathbb{R}^n$ and the matrix pair (A_ξ, B_ξ) is controllable. Hence, instead of rushing headlong into the stabilization problem, using the established transformation to the normal form, a more simple solution has been derived. Inspired by the solution found for the constant speed turbine, the alternative transformation targets the nonlinearity in $\dot{x}_1 = f_1(x)$ directly. Consequently the transformation should have the form $T_\xi(x) = \begin{bmatrix} x_1 \\ L_f x_1 \\ \phi(x) \end{bmatrix}$, where $\phi(x)$ should be chosen such that $T_\xi(x)$ is a diffeomorphism. The somewhat surprising result is that the transformation

$$T_\xi(x) = \begin{bmatrix} x_1 \\ L_f x_1 \\ --- \\ x_2 \\ x_3 \\ x_5 \end{bmatrix} \tag{7.25}$$

is indeed a diffeomorphism which leeds to feedback linearizable coordinates. Since the dynamics associated with $[x_2\ x_3\ x_5]^T$ is linear, feedback linearizing the resulting system, amounts to linearizing the path between the pitch input θ_r and the rotor speed ω_r which is exactly what was done for the constant speed turbine. That T_ξ is an alternative transformation which leeds to feedback linearization actually means that it is related to the transformation $T_c(x)$ through an additional (non-singular) linear transformation (Khalil 2002). The linear transformation is given below:

$$\underbrace{\begin{bmatrix} \omega_r \\ \dot{\omega}_r \\ \omega_g \\ \delta \\ T_g \end{bmatrix}}_{T_\xi(x)} = \underbrace{\begin{bmatrix} \frac{1}{N_g} & 0 & 0 & 1 & 0 \\ 0 & \frac{1}{N_g} & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -J_g & \frac{K_s}{N_g} & \frac{D_s}{N_g} & 0 \end{bmatrix}}_{M_{\xi z}} \underbrace{\begin{bmatrix} \omega_g \\ \dot{\omega}_g \\ \delta \\ \dot{\delta} \\ \ddot{\delta} \end{bmatrix}}_{T_c(x)} \tag{7.26}$$

Given the new state vector the system can be written in the following form:

$$\begin{aligned}
\dot{\xi} &= A_\xi \xi + B_\xi (f_\xi(x) + G_\xi(x)u) \\
&= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{D_s}{J_g N_g} & 0 & -\frac{D_s}{J_g N_g^2} & \frac{K_s}{J_g N_g} & -\frac{1}{J_g} \\ 1 & 0 & -\frac{1}{N_g} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{\tau_T} \end{bmatrix} \xi \\
&+ \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{1}{\tau_T} \end{bmatrix} \left(\begin{bmatrix} L_f^2 x_1 \\ 0 \end{bmatrix} + \begin{bmatrix} L_g L_f x_1 & 0 \\ 0 & 1 \end{bmatrix} u \right)
\end{aligned} \tag{7.27}$$

The linearizing input is:

$$\begin{aligned}
u &= \alpha_\xi(x) + \beta_\xi(x)\upsilon \\
&= -\begin{bmatrix} (L_g L_f x_1)^{-1} L_f^2 x_1 \\ 0 \end{bmatrix} + \begin{bmatrix} (L_g L_f x_1)^{-1} & 0 \\ 0 & 1 \end{bmatrix} \upsilon
\end{aligned}$$

All in all the complexity has been significantly reduced as compared to the normal form.

The transformation $T_\xi(x)$ has been verified, to be a diffeomorphism in the neighborhood of equilibrium points, throughout the top wind speed region. It is not immediately possible to analytically justify that the transformation is a diffeomorphism throughout the region of interest. However, the intuitive justification which was proposed in section 7.2 applies equally for $T_\xi(x)$ since it essentially targets the same state as $T_c(x)$ for the constant speed turbine.

7.4.2 Disturbance decoupling

Until now the wind disturbance has been considered a constant parameter in the design process. One could simply try and use the design obtained so far and hope that it is sufficiently robust, towards deviations in the wind speed. However, it turns out it is possible to decouple the wind from the states ξ if the wind augmented turbine model (2.29) is considered in the design instead of the wind turbine model (2.14). This section explains how the decoupling control is obtained.

Considering the wind augmented model (2.29), it is seen that the wind augmentation does not change the relative degree of the input-output description. This means that the total relative degree of the system is unchanged $r = 5$, whereas the order of the system has been increased to $n = 7$. When transforming the system to the normal form, a description of the internal dynamics is therefore needed. Naturally the internal dynamics are simply given by the wind model, which means that the transformation to the normal form becomes:

$$\begin{bmatrix} T_c(x,w) \\ -- \\ w \end{bmatrix} = \begin{bmatrix} x_2 \\ L_f x_2 \\ x_3 \\ L_f x_3 \\ L_f^2 x_3 \\ -- \\ w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} \omega_g \\ \dot{\omega}_g \\ \delta \\ \dot{\delta} \\ \ddot{\delta} \\ -- \\ w_1 \\ w_2 \end{bmatrix}$$

where the Lie derivatives have been calculated on the basis of the wind augmented turbine model (2.29). The wind augmented version, of the *alternative* transformation T_ξ is obtained through the linear transformation:

$$\begin{bmatrix} T_\xi(x,w) \\ -- \\ w \end{bmatrix} = \left[\begin{array}{c|c} M_{\xi z} & 0 \\ \hline 0 & I \end{array}\right] \begin{bmatrix} T_c(x,w) \\ -- \\ w \end{bmatrix} = \begin{bmatrix} \omega_r \\ \dot{\omega}_r \\ \omega_g \\ \delta \\ T_g \\ -- \\ w_1 \\ w_2 \end{bmatrix}$$

where $M_{\xi z}$ is the linear transformation which was used to transform the normal form to the alternative form, in the last section (see equation (7.26)).

Transforming the original system (2.29) by the transformation $T_\xi(x,w)$ results in the following nonlinear system description:

$$\dot{\xi} = A_\xi \xi + B_\xi (f_\xi(x,w) + G_\xi(x,w)u) \tag{7.28}$$
$$\dot{w} = f_w(w) + G_w e \tag{7.29}$$

where A_ξ and B_ξ are given in equation (7.27) and:

$$f_\xi(x,w) = \begin{bmatrix} L_f^2 x_2 \\ 0 \end{bmatrix}$$
$$G_\xi(x,w) = \begin{bmatrix} L_g L_f x_1 & 0 \\ 0 & 1 \end{bmatrix}$$

It is again stressed that the Lie derivatives are calculated on the basis of the wind augmented turbine model (2.29). $f_\xi(x,w)$ and $G_\xi(x,w)$ are not given in detail above, since they are notationally exhaustive, however, it is noted that the disturbance e does not appear in either $f_\xi(x,w)$ nor $G_\xi(x,w)$, which means that the wind model can be decoupled from the feedback linearized system by choosing the linearizing control:

$$\begin{aligned} u &= \alpha_\xi(x,w) + \beta_\xi(x,w)\upsilon \\ &= -G_\xi(x,w)^{-1} f_\xi(x,w) + G_\xi(x,w)^{-1}\upsilon \end{aligned} \tag{7.30}$$

That e does not appear in f_ξ and G_ξ, is because the relative degrees associated with the disturbance-output relations, exceeds the corresponding relative degrees, associated with the input-output relations, by at least 1. Put in another way: when the outputs are differentiated to obtain a direct input-output relation, the disturbance e does not appear in the relations. General conditions under which disturbance decoupling is possible, can be seen in (Isidori 1995). The disturbance decoupling control which has been derived above, is simply a special case.

Because the gain scheduled parameters of the wind model (2.27) is in the same sub-state-equation as e, they do not appear, in the feedback linearizing control (7.30), either.

7.4.3 Linear controller for feedback linearized system

To conclude the feedback linearization design, what remains is to design a controller for the feedback linearized (disturbance decoupled) system. Following the example of chapter 6, an infinite horizont linear quadratic controller has been designed for the feedback linearized system. The definition of a LQ controller is introduced in chapter 6.1 and will not be repeated here.

Analog to the LQ design for the original system in chapter 6, the feedback linearized (disturbance decoupled) system is augmented with the integral states:

$$\xi_{i1} = \int (\xi_3\xi_5 - P_{e,nom})\,dt \qquad \xi_{i2} = \int (\xi_3 - \omega_{g,nom})\,dt$$

The integral state ξ_{i1} introduces a nonlinearity in the feedback linearized system because $P_e = \omega_g T_g = \xi_3\xi_5$. However, since the closed loop system will be designed, such that the deviations from the nominal value $P_{e,nom}$, are relatively small, a linear design for the augmented system is justified. The model used for the controller design is consequently:

$$\begin{bmatrix} \dot{\bar{\xi}} \\ \dot{\bar{\xi}}_i \end{bmatrix} = \begin{bmatrix} A_\xi & 0 \\ A_i & 0 \end{bmatrix} \begin{bmatrix} \bar{\xi} \\ \bar{\xi}_i \end{bmatrix} + \begin{bmatrix} B_\xi \\ 0 \end{bmatrix} \upsilon \tag{7.31}$$

where

$$\begin{aligned} A_i &= \left(\frac{\partial}{\partial \xi} \begin{bmatrix} \xi_3\xi_5 - P_{e,nom} \\ \xi_3 - \omega_{g,nom} \end{bmatrix} \right) \Bigg|_{\xi=\xi_0} \\ &= \begin{bmatrix} 0 & 0 & T_{g,nom} & 0 & \omega_{g,nom} \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{aligned}$$

Defining $\psi \equiv \begin{bmatrix} \bar{\xi} \\ \xi_i \end{bmatrix}$ and $\nu \equiv \upsilon$ the feedback law is found by minimizing the quadratic cost (6.1) subject to the state equation (7.31). The following quantities are weighted in the quadratic cost:

$$\varphi = \begin{bmatrix} \bar{\omega}_r & \dot{\bar{\omega}}_r & \bar{\omega}_g & \bar{\delta} & \bar{T}_g & \int \bar{P}_e & \int \bar{\omega}_g & \upsilon_1 & \upsilon_2 & \bar{P}_e & \dot{\bar{\delta}} \end{bmatrix}^T$$

compared to the LQ design in chapter 6 the state θ and input θ_r are no longer explicitly given in the criteria. This is no longer possible due to the nonlinear transformation $T_\xi(x)$. The linear transformations that extract φ from ψ and ν are:

$$C_{\varphi\psi} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & T_{g,0} & 0 & \omega_g & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$C_{\varphi\nu} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Appropriate weights on the terms are found to be:

$$O = diag\begin{bmatrix} \frac{1}{x_{10}} & \frac{1}{x_{10}} & \frac{50}{x_{20}} & \frac{1}{x_{30}} & \frac{0.001}{x_{50}} & \frac{50}{P_{e0}} & \frac{1}{x_{20}} & \frac{10}{\upsilon_{10}} & \frac{0.001}{\upsilon_{20}} & \frac{50}{P_{e0}} & \frac{1}{x_{30}} \end{bmatrix}^2$$

Consequently the weight in the quadratic cost is:

$$Q = \begin{bmatrix} C_{\varphi\psi} & C_{\varphi\nu} \end{bmatrix}^T O \begin{bmatrix} C_{\varphi\psi} & C_{\varphi\nu} \end{bmatrix}$$

Finding the LQ gain $K \in \mathbb{R}^{2\times 7}$ the final feedback linearizing disturbance decoupling control law is given by:

$$\begin{aligned} u &= \alpha_\xi(x,w) + \beta_\xi(x,w)\upsilon \\ &= -G_\xi^{-1}(x,w) f_\xi(x,w) + G_a^{-1}(x,w)\upsilon \\ &= -G_\xi^{-1}(x,w) f_\xi(x,w) - G_\xi^{-1}(x,w) K T_\xi(x,w) \end{aligned}$$

7.5 Simulations

Naturally decoupling is compromised if model uncertainties are present - in particular uncertainties in the wind model. The states of the wind model are impossible to measure therefore a practical controller would rely on estimates of these states. Simulations have been made to see the effect of a stochastic uncertainty in the wind speed. Figure 7.3 shows the true wind states versus the perturbed wind states. The stochastic uncertainty is a band-limited white noise process with a significant larger band-width than the wind model. Figure 7.4 shows the corresponding evolutions of

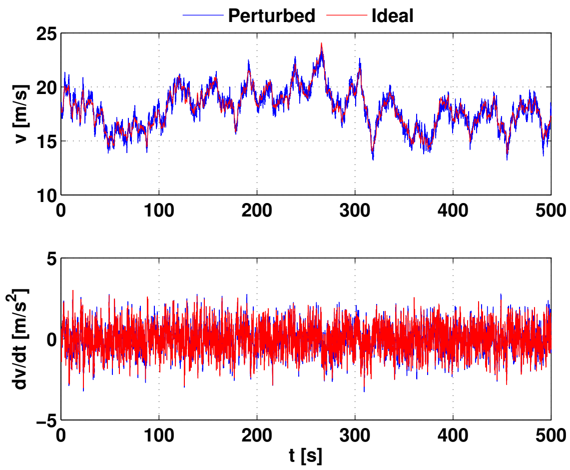


Figure 7.3: True and perturbed wind speed and wind acceleration. The mean wind speed is 18 m/s.

the states. Figure 7.5 shows the generated power P_e and the speed of the pitching motion.

It is verified that the wind disturbance is indeed decoupled from the linearized state vector rendering ω_r, ω_g, δ and T_g constant (and consequently also P_e). Introducing the stochastic perturbation of the wind states naturally compromises the decoupling, however it does not destabilize the system. Being a nonlinear function of the feedback linearized state vector, The output P_e only exhibits fluctuation in one direction, given the stochastic uncertainties. This interesting behavior is not easily explained, but is somehow connected to the fact that P_e is a product of the feedback linearized states. It is only observed, when the uncertainties are restricted to the wind states.

Obviously the performance of the feedback linearized controller is superior to the gain scheduled controller in ideal conditions. Simulations in a more realistic setup will be given in chapter 12.

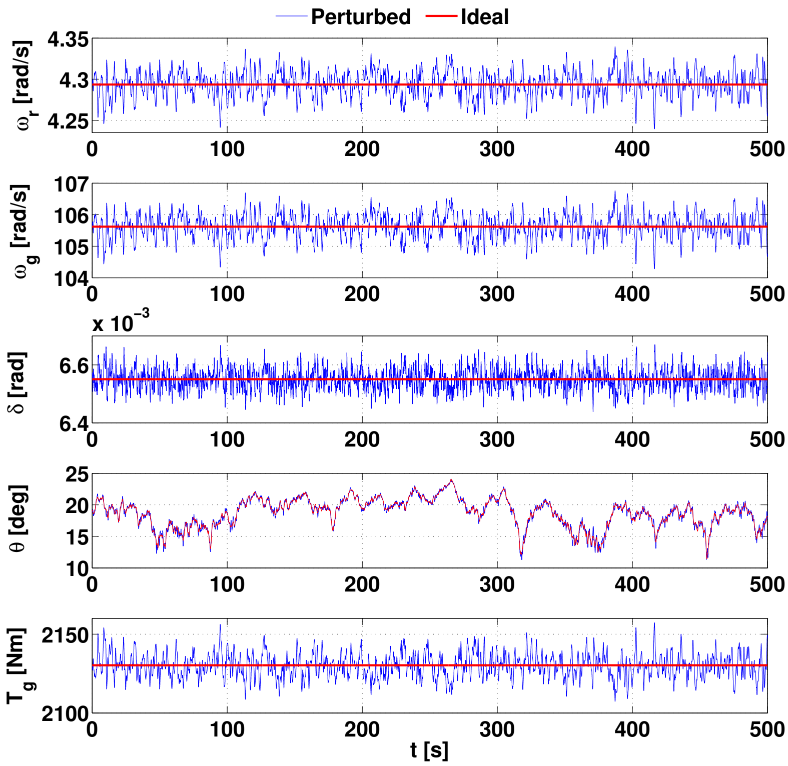
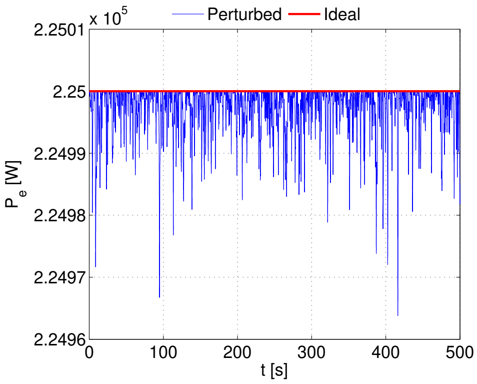
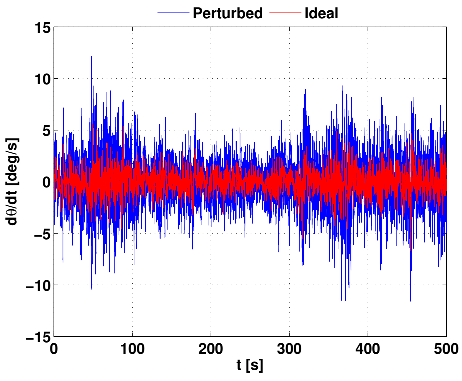


Figure 7.4: Evolution of the states with the true and perturbed wind evolution seen in figure 7.3.



(a) P_e



(b) $\frac{d\theta}{dt}$

Figure 7.5: Evolution of output P_e and the pitching speed $\dot{\theta}$ associated with the state evolution in figure 7.4.

Chapter 8

Sliding mode control

Sliding mode control aims at stabilizing a system using a fast switching control law. The purpose of the switching control law, is to restrict the trajectory of the system, to a manifold in state space in which system properties are desirable. An advantage of the sliding controllers is that they can be made robust towards model uncertainties which occur at the input, so-called matched uncertainties. The Achilles heel of sliding mode is however the discontinuous switching control law.

In section 8.1 sliding mode control is shortly introduced for both single- and multi-input systems. In section 8.2 the simple constant speed model is used for preliminary analysis of sliding mode control. In section 8.3 two sliding mode controllers are designed for the variable speed turbine. Section 8.4 present simulations with the sliding mode control laws.

8.1 Theory

As mentioned, the idea in sliding mode control, is to restrict the trajectories of the system to a certain hyper-surface in state space. This surface is chosen such that the system behavior is desirable, when restricted to the surface. When the surface has been chosen, the next step is to design a control law that makes the trajectory intersect the surface and furthermore restricts the trajectory to the surface for all future times. In ideal sliding control this is done using an infinitely fast switching control law.

As for feedback linearization the presentation will be restricted to systems which are affine in the inputs and time invariant i.e. can be written in the following form.

$$\dot{x} = f(x) + G(x)u \quad , \ x \in \mathbb{R}^n, \ u \in \mathbb{R}^m \tag{8.1}$$

The derivation of a sliding mode control law can be made systematic and greatly simplifies, when the system is given in the so-called regular form, therefore the focus will be on this system structure[1]. The theoretical presentation in this chapter is mainly based on (Khalil 2002) and (DeCarlo et al. 1988). Although restricted to the normal form, (Slotine and Li 1991) has also been a source of inspiration.

8.1.1 Choosing the surface

The sliding surface $s(x) = 0$ is a $(n-m)$-dimensional manifold in $\mathbb{R}^n$, determined by the intersection of m, $(n-1)$-dimensional surfaces $s_i(x) = 0$, ie.:

$$s(x) = \begin{bmatrix} s_1(x) & s_2(x) & \dots & s_m(x) \end{bmatrix}^T = 0$$

The surface should be chosen such that, when restricted to the surface, the system has a desired behavior. It is generally not possible to give a systematic procedure for choosing the surface, but having chosen a surface one can easily determine the system behavior on it. A straight forward procedure is to use the method of equivalent control (DeCarlo et al. 1988) which will be used throughout rest of the presentation. The equivalent control u_{eq} is the feedback law which makes the system trajectory stay on the surface $s = 0$ when started on the surface:

$$s(x(t_0)) = 0 \Rightarrow s(x(t)) = 0 \quad \text{for} \quad t > t_0$$

This implies that $\dot{s}(x(t)) = 0$. Calculating the time derivative of s the equivalent control is derived:

$$\begin{aligned} \frac{\partial s}{\partial x}\dot{x} &= \frac{\partial s}{\partial x}\left(f(x) + G(x)u_{eq}\right) = 0 \\ &\Updownarrow \\ u_{eq} &= -\left(\frac{\partial s}{\partial x}G(x)\right)^{-1}\frac{\partial s}{\partial x}f(x) \end{aligned} \tag{8.2}$$

assuming that the matrix product $\frac{\partial s}{\partial x}G(x)$ is nonsingular. The dynamics of the system on $s = 0$, is now found by combining (8.1) and (8.2).

$$\begin{aligned} \dot{x} &= f(x) + G(x)u_{eq} \\ &= \left(I - G(x)\left(\frac{\partial s}{\partial x}G(x)\right)^{-1}\frac{\partial s}{\partial x}\right)f(x) \end{aligned} \tag{8.3}$$

[1]Restricting the focus to the regular form is restrictive in general but is suitable for the wind turbine

The dynamics above can then be analyzed using e.g. Lyapunov theory to determine stability.

As mentioned, this procedure does not hint on how to choose the surface for a general system structure, but only provides a method for validating a specific choice. However if the system is given in the so-called regular form and restricting the surface to a special structure a systematic procedure can be deduced. A system on the regular form has the following structure:

$$\begin{aligned}\dot{x}_a &= f_a(x) \\ \dot{x}_b &= f_b(x) + G_b(x)u \qquad (8.4)\end{aligned}$$

where $x_a \in R^{n-m}$, $x_b \in R^m$, $u \in R^m$ i.e. f_b is in the range space of G_b. The sliding surface is restricted to the following structure:

$$s = x_b - \phi(x_a) = 0 \qquad (8.5)$$

The advantage obtained by the restrictions (8.4) and (8.5) is seen by first calculating the equivalent control

$$u_{eq} = -G_b(x)^{-1}\left(-\frac{\partial\phi(x_a)}{\partial x_a} f_a(x) + f_b(x)\right)$$

from which the behavior on the surface is found:

$$\begin{aligned}\dot{x}_a &= f_a(x_a, \phi(x_a)) \qquad (8.6) \\ \dot{x}_b &= \frac{\partial\phi(x_a)}{\partial x_a} f_a(x_a, \phi(x_a)) \qquad (8.7)\end{aligned}$$

It is seen from the dynamics above that designing $\phi(x_a)$ is equivalent to designing a feedback law for the reduced order system f_a, using x_b as input. When f_a is linear, the problem of designing a sliding surface, is equivalent to designing a linear controller for the reduced order system f_a using linear control methods [2]. Examining equations (8.6)-(8.7) it is seen that the behavior on the surface is dependent only on f_a, hence the system is robust towards model uncertainties in f_b when restricted to the surface.

8.1.2 Choosing the control law

Given a sliding manifold, the next step is to find a control law which makes the system trajectory intercept the manifold (in finite time) and restricts the state trajectory to the manifold for all future times. As will be shown, this is done using a discontinuous switching control law.

[2]This fact will be utilized in the subsequent design on the wind turbine

For single input systems on the regular form it is straightforward to establish a pure discontinuous control law which intersects the surface:

$$s = x_b - \phi(x_a) = 0 \quad , \qquad x_a \in R^{n-1}, \quad x_b \in R$$

Utilizing Lyapunov theory, the Lyapunov function candidate is chosen as:

$$V = \frac{1}{2}s^2$$

which is positive definite. The derivative is given by

$$\begin{aligned}
\dot{V} &= s\dot{s} = s\frac{\partial s}{\partial x}\dot{x} \\
&= s(-\frac{\partial \phi(x_a)}{x_a}\dot{x}_a + \dot{x}_b) \\
&= s(-\frac{\partial \phi(x_a)}{x_a}f_a(x) + f_b(x) + G_b(x)u)
\end{aligned}$$

Now, to simplify the derivation the following auxiliary function is introduced (Assuming that $G_b(x)$ is positive in the region of interest).

$$\varrho(x) \geq \left| \frac{-\frac{\partial \phi(x_a)}{\partial x_a}f_a(x) + f_b(x)}{G_b(x)} \right|$$

Using the auxiliary function, the following inequality is obtained

$$\dot{V} \leq G_b(x)|s|\varrho(x) + G_b(x)su \tag{8.8}$$

From this expression it can be seen that choosing the following discontinuous control law will render $\dot{V}$ negative definite:

$$u = -k(x)\text{sgn}(s) \quad , \qquad k(x) = \varrho(x) + \varrho_0 \quad \varrho_0 > 0 \tag{8.9}$$

since combining (8.8) and (8.9) results in

$$\dot{V} < -G_b(x)\varrho_0|s| \quad , \qquad s \neq 0$$

Furthermore the discontinuous control (8.9) will make the trajectory reach the surface in finite time (Khalil 2002). For multi-input systems the solution is not so easily found because of the cross-couplings in the input-state description. (DeCarlo et al. 1988) describes various methods for dealing with multi-input systems. For example the so-called diagonalization method, which solves the problem analog to the single-input problem, through a nonlinear transformation. The method that has been chosen in this project involves using the equivalent control u_{eq}. This method has the advantage that the resulting control law is the sum of a continuous and discontinuous signal, which means that the discontinuous signal is reduced in size. With the restrictions on the system structure and manifold imposed earlier the method is easily described: Having found a surface on the form

$$s = x_b - \phi(x_a) \quad , \qquad x_a \in R^{n-m}, \quad x_b \in R^m$$

The following Lyapunov candidate is chosen:

$$V = \frac{1}{2} s^T s$$

Calculating the time derivative and splitting the control into a superposition of the equivalent control and a auxiliary input the problem simplifies significantly:

$$\begin{aligned}
\dot{V} &= s^T \dot{s} \\
&= s^T(-\frac{\partial \phi(x_a)}{\partial x_a}\dot{x}_a + f_b(x) + G_b(x)(u_{eq} + u_d)) \\
&= s^T(-\frac{\partial \phi(x_a)}{\partial x_a} f_a + f_b(x) + G(x)u_{eq}) + s^T G_b(x) u_d \\
&= 0 + s^T G(x) u_d
\end{aligned}$$

It is seen that the auxiliary input u_d renders $\dot{V}$ negative definite if chosen as

$$u_d = -G(x)_b^{-1} k(x) \mathrm{sgn}(s)$$

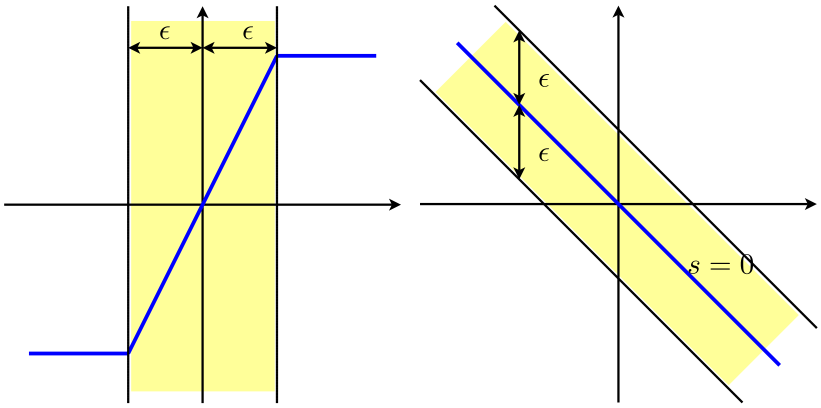
where the only restriction is that $k(x)$ should be larger than 0. $k(x)$ could for example be chosen such that the system remains on the surface given matched uncertainties. A derivation of the control including a certain type of matched disturbances, is given in (Khalil 2002). However, it is not straight forward to analytically quantize the disturbances considered in this project, moreover, not only matched uncertainties are considered. Therefore, $k(x)$ is chosen as a static gain $k \equiv k(x)$ and simulations are used to verify that the gain is appropriate.

8.1.3 Smoothing the control law

Increasing the gain in the discontinuous control k will increase the robustness of the system. Equivalently it will make the trajectory intercept the surface faster when started of the the surface. However, in practice the switching control law will introduce chattering e.g. due to a delay in the switching component, and a large gain will make the chattering more profound. There are two common ways of dealing with chattering: The first is to use the equivalent control u_{eq} in the control law. The second method is to make a continuous approximation to the discontinuous control law. This significantly suppresses the chattering but decreases the theoretical performance of the controller. The continuous approximation can be made by smoothing the discontinuity in a thin boundary layer neighboring the manifold, making this layer an invariant set in state space. However, the precision of the control is worsened, as the boundary layer is made thicker. Figure 8.1(a) shows the most common continuous approximation; namely the saturation function.

$$\mathrm{sat}\left(\frac{s}{\epsilon}\right) = \begin{cases} \frac{s}{\epsilon} & \text{if } \left|\frac{s}{\epsilon}\right| \leq 1 \\ \mathrm{sgn}\left(\frac{s}{\epsilon}\right) & \text{if } \left|\frac{s}{\epsilon}\right| > 1 \end{cases}$$

Figure 8.1(b) shows the resulting boundary layer for a second order system(Slotine and Li 1991).



(a) Saturation function: sat $\left(\frac{s}{\epsilon}\right)$ (b) Boundary layer around s.

Figure 8.1: Left: Continuous approximation the signum function. Right: corresponding boundary layer around the surface.

Chattering is not explicitly considered in this project, rather it is assumed that the switching component is ideal. However, the continuous approximation has been used in the simulation environment, since the numerical solvers in Matlab experience problems with the stiffness of the dynamics. The boundary layer ϵ that has been used in the simulation, is sufficient thin, such that the inherent im-precisions are of no practical concern.

8.2 Preliminary investigations on the constant speed model

This section provides preliminary analysis of sliding mode control in connection with the simple constant speed wind turbine. Using this model it is possible to illustrate the modes (reaching phase and sliding phase) graphically.

As mentioned in the section 8.1 it is advantageous to have the model on the regular form since this simplifies analysis. The original description of the constant speed model (2.18) is already on the regular form. This is explicitly shown below:

$$\begin{aligned}\dot{x}_1 &= \underbrace{\frac{P_r(x_1,x_2)}{Jx_1} - \frac{D_g x_1}{J} + \frac{D_g\omega_0}{Jp}}_{f_a(x)} \\ \dot{x}_2 &= \underbrace{-\frac{1}{\tau_\theta}x_2}_{f_b(x)} + \underbrace{\frac{1}{\tau_\theta}}_{G_b} u\end{aligned}$$

Newer the less, using (2.18) as a basis for sliding mode control is not convenient, since finding a sliding surface, essentially requires that a stabilizing control law is found for the sub-dynamics:

$$\dot{x}_a = \dot{x}_1 = \frac{P_r(x_1,x_2)}{Jx_1} - \frac{D_g x_1}{J} + \frac{D_g\omega_0}{Jp}$$

using $x_b = x_2$ as the control input. First of all, the sub-dynamics are nonlinear, secondly it is non-affine in the control $x_b = x_2$, which makes systematic derivation of a control law difficult. Consequently, the system needs to be transformed to obtain a more suitable regular form.

The normal form of the constant speed model, which was derived in chapter 7, is a special case of the regular form. It has the advantage that the sub-dynamics are linear which makes it painfully simple to design the sliding surface. Therefore, a linear sliding surface will be designed in the transformed coordinate system and a control law will be designed, such that the closed loop system will intersect this surface and stay on it for all future times.

The normal form associated with the simple wind turbine is repeated below:

$$\begin{aligned}\dot{z}_1 &= z_2 \\ \dot{z}_2 &= f_c(x) + G_c(x)u\end{aligned}$$

where f_c and G_c are given in equations (7.12)-(7.13) and the transformation to the normal form is

$$T_c(x) = \begin{bmatrix} x_1 \\ L_f x_1 \end{bmatrix} = \begin{bmatrix} x_1 \\ \frac{P(x_1,x_2)}{Jx_1} - \frac{D_g x_1}{J} + \frac{D_g\omega_0}{Jp} \end{bmatrix}$$

8.2.1 Sliding surface

Designing the sliding surface, amounts to finding a stabilizing control law for the dynamics $\dot{z}_1 = z_2$, using z_2 as the control input. Shifting the equilibria to the origin, the linear feedback law/surface becomes:

$$s = \bar{z}_2 + \lambda\bar{z}_1 = 0\ , \quad \bar{z} = z - z^*$$

This yields the following dynamics when the system is restricted to the surface:

$$\begin{aligned}\dot{\bar{z}}_1 &= -\lambda \bar{z}_1 \\ \dot{\bar{z}}_2 &= \lambda^2 \bar{z}_1\end{aligned}$$

It is seen that choosing λ^{-1} corresponds to choosing the desired time constant for the state z_1 i.e. choosing how fast $\omega_g = z_1$ should converge.

Note, although linear in the transformed coordinate system, the sliding surface is nonlinear in the original states x. The definition in the original states, is shown below:

$$\begin{aligned}s &= \bar{T}_{c2}(x) + \lambda \bar{T}_{c1}(x) \\ &= \frac{P(x_1, x_2)}{Jx_1} - \frac{D_g x_1}{J} + \frac{D_g \omega_0}{Jp} + \lambda(x_1 - x_{10})\end{aligned}$$

8.2.2 Control law

Since the constant speed model is a single input system, a pure discontinuous control law could easily be designed for the system, following the approach in section 8.1. However, the approach chosen in this project, is to use the method of equivalent control to minimize the discontinuous component.

Given the choice of surface, the equivalent control is:

$$\begin{aligned}u_{eq} &= -G_c(x)^{-1} \left(\lambda T_2(x) + f_c(x)\right) \\ &= -(L_g L_f x_1)^{-1} \left(\lambda L_f x_1 + L_f^2 x_1\right)\end{aligned}$$

The discontinuous part of the control is:

$$\begin{aligned}u_d &= -k G_c^{-1}(x) \mathrm{sgn}(s) \\ &= -k(L_g L_f x_1)^{-1} \mathrm{sgn}(s)\end{aligned}$$

When using the Lie derivative notation, the control law does not look especially complex. To illustrate how intricate the expression actually is, it is written in detail below.

$$
\begin{aligned}
u &= u_{eq} + u_d \\
&= -\left(\frac{x_1 J \tau_\theta}{\frac{\partial}{\partial x_2} P_r(x_1, x_2)}\right)\left(\lambda\left(\frac{P(x_1,x_2)}{Jx_1} - \frac{D_g x_1}{J} + \frac{D_g \omega_0}{Jp}\right) + \right. \\
&\underbrace{\left(\left(\frac{\frac{\partial}{\partial x_1} P_r(x_1,x_2)}{x_1 J} - \frac{P_r(x_1,x_2)}{x_1^2 J} - \frac{D_g}{J}\right) \cdot \left(\frac{P_r(x_1,x_2)}{x_1 J} - \frac{x_1 D_g}{J} + \frac{D_g \omega_0}{Jp}\right) - \frac{\frac{\partial}{\partial x_2} P_r(x_1,x_2) x_2}{x_1 J \tau_\theta}\right)\Bigg)}_{u_{eq}} - \\
&\underbrace{k\left(\frac{x_1 J \tau_\theta}{\frac{\partial}{\partial x_2} P_r(x_1,x_2)}\right) \operatorname{sgn}\left(\frac{P(x_1,x_2)}{Jx_1} - \frac{D_g x_1}{J} + \frac{D_g \omega_0}{Jp} + \lambda(x_1 - x_{10})\right)}_{u_d}
\end{aligned}
$$

It is seen that the control is relatively complex (and therefore also notationally exhaustive), even for the simple turbine model. Therefore, when dealing with the variable speed turbine, the notation will be restricted to that of Lie derivatives.

8.2.3 Simulations

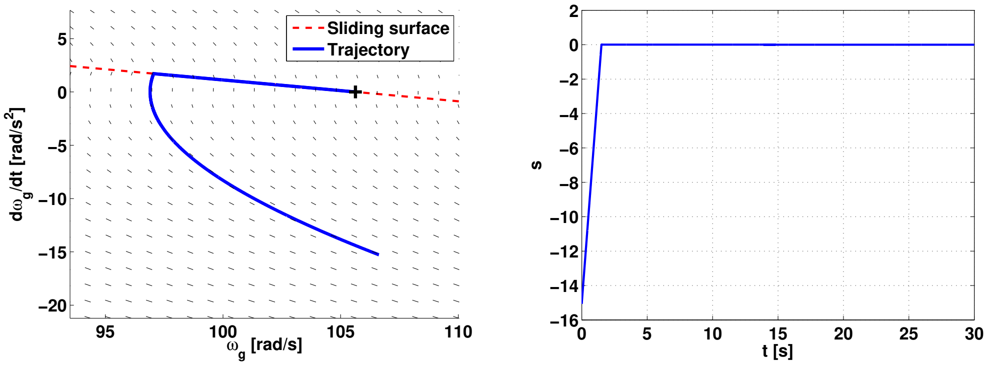
To illustrate the sliding mode controller the parameters have been chosen arbitrarily as $k = 10$ and $\lambda = 0.2$. The system is initiated at a distance from its equilibria and off the surface:

$$x_0 - x^* = [1\ 0]^T$$

Figure 8.2(a) shows the trajectory of the system, and the sliding surface, in the phase plane (transformed coordinates). Figure 8.2(b) shows the value of s as it approaches the surface. The trajectory follows the the gradient field (the reaching phase) for 1.5 seconds after which it collides with the surface and slides into the equilibria (the sliding phase).

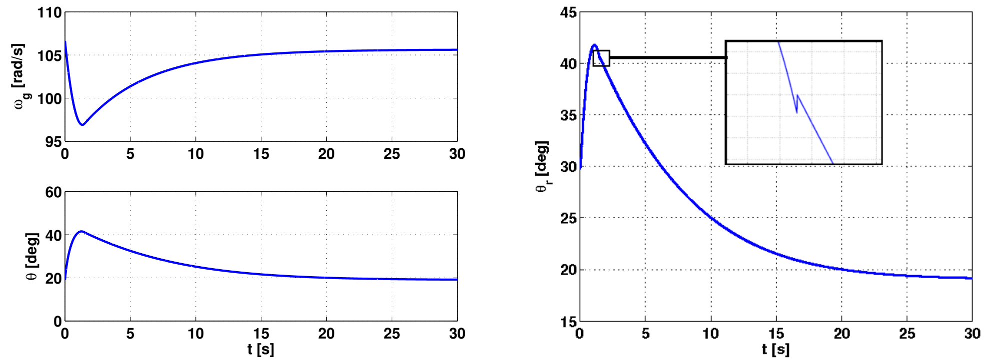
The evolution of the original coordinates x is seen in figure 8.3(a) and figure 8.3(b) shows the control actions. Note the small discontinuity in the control when the surface $s = 0$ is reached.

The controller has been chosen for demonstrational purposes and is not meant as a serious choice. However, this section clearly shows that it is possible to design a sliding surface controller which removes the effect of the nonlinearities.



(a) Phase portrait (b) Evolution of s

Figure 8.2: Phase portrait of closed loop system and evolution of s. Note that the phase portrait is in the transformed coordinates.



(a) Evolution of states (b) Evolution of control

Figure 8.3: Evolution of states and control. Note the small discontinuity in the control when the surface is reached

8.3 Sliding mode control of the variable speed model

It was seen in section 8.2 that the constant speed model is natively given in a regular form. The same is seen to apply, for the variable speed model (2.14) by defining $x_a = [x_1\ x_2\ x_3]^T$ (wind turbine mechanics) and $x_b = [x_4\ x_5]^T$ (actuator states). However, as for the constant speed turbine, this would require that a nonlinear control problem is solved to obtain the surface (see the discussion in the previous section).

Another possibility is to transform the system into the normal form, derived in chapter 7 and repeated below:

$$\dot{z} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{A_c} z + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}}_{B_c} (f_c(x) + G_c(x)u)$$

where $f_c(x)$, $G_c(x)$ and the state transformation $z = T_c(x)$ are given in equations (7.21)-(7.22) and (7.20). Finding the sliding surface for this structure is simply a matter of finding a (linear) stabilizing control law for the integrator system:

$$\dot{z}_a = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} z_a + \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} z_b \tag{8.10}$$

where $z_a = [z_1\ z_3\ z_4]^T$ is the state vector and $z_b = [z_2\ z_5]^T$ is the control input. Consequently the surface design problem is basically reduced to a linear control problem. Although it is painfully simple to find a stabilizing control law for (8.10), it is not obvious how the problem should be solved such that appropriate performance is obtained for the original system.

Instead the representation of the dynamic system obtained through the transformation:

$$T_\xi(x) = \xi = \begin{bmatrix} x_1 \\ L_f x_1 \\ x_2 \\ x_3 \\ x_5 \end{bmatrix} = \begin{bmatrix} \omega_r \\ \dot{\omega}_r \\ \omega_g \\ \delta \\ T_g \end{bmatrix}$$

has been considered. The resulting system is repeated below:

$$\dot{\xi} = A_\xi \xi + B_\xi (f_\xi(x) + G_\xi(x)u)$$

The model details can be seen in equations (7.27). This transformation was introduced in chapter 7, to simplify the design of a feedback linearizing controller. As will be seen it also simplifies the surface design significantly. To see that the transformed system is on the regular form, and to simplify notation in the following, the system is subjected to yet another transformation:

$$\begin{bmatrix} \xi_a \\ -- \\ \xi_b \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{M} T_\xi(x) = \begin{bmatrix} \xi_1 \\ \xi_3 \\ \xi_4 \\ -- \\ \xi_2 \\ \xi_5 \end{bmatrix} = \begin{bmatrix} \omega_r \\ \omega_g \\ \delta \\ -- \\ \dot{\omega}_r \\ T_g \end{bmatrix}$$

transforming the matrixes A_ξ and B_ξ results in:

$$\left[\begin{array}{c|c} A_{a1} & A_{a2} \\ \hline A_{b1} & A_{b2} \end{array}\right] = MA_\xi M^{-1}$$

$$= \left[\begin{array}{ccc|cc} 0 & 0 & 0 & 1 & 0 \\ \frac{D_s}{J_g N_g} & -\frac{D_s}{N_g^2 J_g} & \frac{K_s}{N_g J_g} & 0 & -\frac{1}{J_g} \\ 1 & -\frac{1}{N_g} & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{\tau_T} \end{array}\right]$$

$$\left[\begin{array}{c} B_a \\ \hline B_b \end{array}\right] = MB_\xi$$

$$= \left[\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \hline 1 & 0 \\ 0 & \frac{1}{\tau_T} \end{array}\right]$$

The system can now be divided into two equations, which clearly shows that the system is on the regular form:

$$\dot{\xi}_a = \underbrace{A_{a1}\xi_a + A_{a2}\xi_b}_{f_a}$$

$$\dot{\xi}_b = \underbrace{A_{b1}\xi_a + A_{b2}\xi_b + B_b f_\xi(x)}_{f_b} + \underbrace{B_b G_\xi(x)}_{G_b} u$$

It is seen that finding an appropriate surface corresponds to finding a suitable stabilizing control law for the sub-dynamics:

$$\dot{\xi}_a = A_{a1}\xi_a + A_{a2}\xi_b$$

where ξ_a are the states to be controlled and ξ_b are the inputs. This is where the advantage is gained as compared to the normal form $z = T_c(x)$. It is seen that ξ_a are exactly the states of the turbine mechanics, furthermore the dynamics $\dot{\xi}_a = A_{a1}\xi_a$ are roughly the same as for the original structure.

A positive feature of the transformation $T_\xi(x)$ is that the wind disturbance has been transformed into $f_b(x)$ i.e. it satisfies the matching condition. Consequently, if restricting the trajectory to the surface, the dynamics of the closed loop system will be robust towards fluctuations in the wind.

8.3.1 Sliding surface

The surface is found by designing a control law for the sub-dynamics such that the performance of the 'closed loop' sub-dynamics is desirable. To design the controller it has been chosen to use the LQ controller which was introduced in section 6.1.

To ensure zero stationary error, when on the surface, the system structure is augmented with integrators analog to the designs in chapter 6 and 7 i.e. integrators on $\bar{P}_e = \xi_3\xi_5 - P_{e,0}$ and $\bar{\omega}_g = \xi_3 - \omega_{g,0}$. Augmenting the system with the integral states naturally adds a nonlinearity to the design, since P_e is a nonlinear function of the 'state' ξ_3 and the 'input' ξ_5. The resulting nonlinear system is shown below:

$$\begin{bmatrix} \dot{\xi}_a \\ -- \\ \xi_i \end{bmatrix} = \begin{bmatrix} f_a \\ -- \\ f_i \end{bmatrix} = \begin{bmatrix} A_{a1}\xi_a + A_{a_2}\xi_b \\ ------ \\ \xi_3\xi_5 - P_{e0} \\ \xi_3 - \omega_{g0} \end{bmatrix}$$

where $\xi_i = [\int \bar{P}_e \ \int \bar{\omega}_g]^T$. However, since the controller is designed such that fluctuations around the nominal power $P_{e,0}$ are small, this nonlinearity is appropriate approximated with its linearized counterpart. This results in the following augmented linear model:

$$\begin{bmatrix} \dot{\bar{\xi}}_a \\ \dot{\bar{\xi}}_i \end{bmatrix} = \begin{bmatrix} A_{a1} & 0 \\ A_{i1} & 0 \end{bmatrix} \begin{bmatrix} \bar{\xi}_a \\ \bar{\xi}_i \end{bmatrix} + \begin{bmatrix} A_{a2} \\ A_{i2} \end{bmatrix} \bar{\xi}_b \tag{8.11}$$

where

$$\begin{aligned} A_{i1} &= \left.\frac{\partial f_i}{\partial \xi_a}\right|_{\xi_a=\xi_{a0}} = \begin{bmatrix} 0 & T_{g0} & 0 \\ 0 & 1 & 0 \end{bmatrix} \\ A_{i2} &= \left.\frac{\partial f_i}{\partial \xi_b}\right|_{\xi_b=\xi_{b0}} = \begin{bmatrix} 0 & \omega_{g,0} \\ 0 & 0 \end{bmatrix} \end{aligned}$$

Defining $\psi \equiv \begin{bmatrix} \bar{\xi}_a \\ \bar{\xi}_i \end{bmatrix}$ and $\nu \equiv \xi_b$ the sliding surface is found by minimizing the quadratic cost (6.1) subject to the state equation (8.11). The following quantities are weighted in the quadratic cost:

$$\varphi = \begin{bmatrix} \bar{\omega}_r & \bar{\omega}_g & \bar{\delta} & \int \bar{P}_e & \int \bar{\omega}_g & \dot{\bar{\omega}}_r & \bar{T}_g & \bar{P}_e & \dot{\bar{\delta}} \end{bmatrix}$$

The linear transformation that extract φ from ψ and ν are:

$$C_{\varphi\psi} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & T_{g,0} & 0 & 0 \\ 1 & 0 & -\frac{1}{N_g} & 0 & 0 \end{bmatrix} \quad C_{\varphi\nu} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & \omega_{g,0} \\ 0 & 0 \end{bmatrix}$$

the appropriate weight on φ has been found to be:

$$O = \text{diag}\left[\frac{1}{x_{10}} \quad \frac{1}{x_{20}} \quad \frac{1}{x_{30}} \quad \frac{50}{P_{e0}} \quad \frac{50}{x_{20}} \quad \frac{4}{x_{10}} \quad \frac{0.0001}{x_{50}} \quad \frac{50}{P_{e0}} \quad \frac{1}{x_{30}}\right]^2$$

This leads to the following weight Q in the quadratic criteria:

$$Q = [C_{\varphi\psi} C_{\varphi\nu}]^T O [C_{\varphi\psi} C_{\varphi\nu}]$$

Having calculated the LQ gain K, the surface is given by:

$$s = \begin{bmatrix} \bar{\xi}_a \\ \bar{\xi}_i \end{bmatrix} + K\bar{\xi}_b \ , \quad K \in \mathbb{R}^{2\times 5} \tag{8.12}$$

8.3.2 Control law without wind knowledge (SM1)

As mentioned previously the wind disturbance satisfies the matching condition. This is an intriguing fact, since choosing a suitable high gain in the discontinuous control law, will make the system stay on the surface given matched uncertainties. I.e. the dynamics of the closed loop system remains unchanged.

A pure discontinuous control law is not desirable from a practical point of view. Therefore the control law is designed as a superposition of the equivalent control and a discontinuous switching control.

The equivalent control is first established

$$u_{eq} = G_\xi(x)^{-1} \left(-f_\xi(x) - K \begin{bmatrix} f_a(x) \\ f_i(x) \end{bmatrix} \right)$$

The discontinuous control is given by:

$$u_d = -G_\xi^{-1}(x) k \mathrm{sgn}(s)$$

where k should be chosen such that the trajectory stays on the surface, when subject to the wind disturbance. In chapter 11 measurement noise and a state estimator is introduced, hence not only matched disturbances are considered in the project. k has therefore been found by tuning the closed loop system, for these conditions.

$$k = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix}$$

The gain above makes the trajectory stay on the surface, when the only uncertainty is the wind fluctuations. Furthermore, it ensures that the trajectory is forced onto the surface under the perturbed conditions introduced in chapter 11.

8.3.3 Control law with wind knowledge (SM2)

Instead of letting a high gain discontinuous component deal with the fluctuating wind speed, the wind will be dealt with more intelligently in this section. Effectively it will incorporate the disturbance decoupling control, as shown in chapter 7, but have the additional advantage of being robust towards uncertainty in the wind model.

To take advantage of the known structure of the wind the augmented model including integral states and wind dynamics is considered. Calculating the equivalent control on the basis of the augmented wind turbine model (7.28) results in the following control law:

$$u_{eq} = G_\xi(x,w)^{-1}\left(-f_\xi(x,w) - K\begin{bmatrix} f_a \\ f_i \end{bmatrix}\right)$$

It is seen that the equivalent control incorporates exactly the disturbance decoupling control used in chapter 7, to feedback linearize the turbine (see equation (7.30)). The discontinuous component is chosen to be:

$$u_d(x) = -G_\xi^{-1}(x,w)k\mathrm{sgn}(s)$$

where, without model uncertainties, the only restriction is that k should be larger than zero. Taking into account the perturbations which will be introduced in chapter 11, the gain k has been chosen as:

$$k = \begin{bmatrix} 100 & 0 \\ 0 & 1000 \end{bmatrix}$$

The gain above forces the trajectory onto the surface, under the perturbed conditions. It is seen that k_{11} has been reduces by a factor 10, compared to the gain associated with SM1.

8.4 Simulations

This section present simulations of the closed loop system response with the two control laws found in the previous section. Assuming complete state knowledge and no model uncertainties, the second control law (SM2) naturally decouples the disturbance perfectly. Therefore, SM2 has been simulated with ideal and perturbed conditions.

8.4.1 Closed loop performance with SM1

To illustrate the design the equivalent control has been dimensioned for a constant wind speed of $v = 18$m/s. The wind sequence that has been used for the simulation is shown in figure 8.4. Figure 8.6 shows the evolution of the states and figure 8.5 shows the output P_e. Comparing the signals with those for the gain scheduled controller in chapter 6 it is seen that SM1 has superior performance, even though it has no knowledge of the wind. The control signals and the speed of the pitching motion is

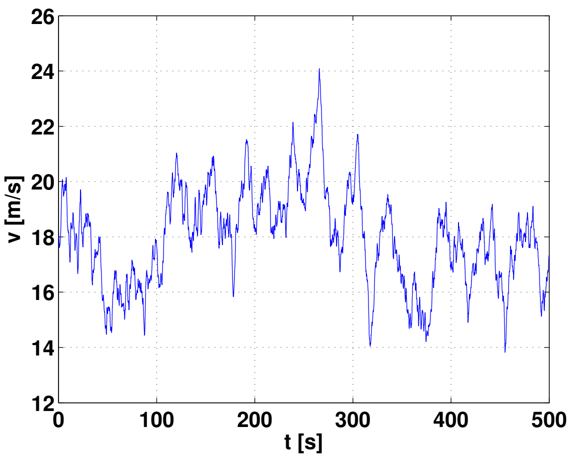


Figure 8.4: Wind sequence with constant mean wind speed v_m=18 m/s

seen in figure 8.7. Even though the control signal is partly discontinuous, it does not make the pitch exceed its physical limits.

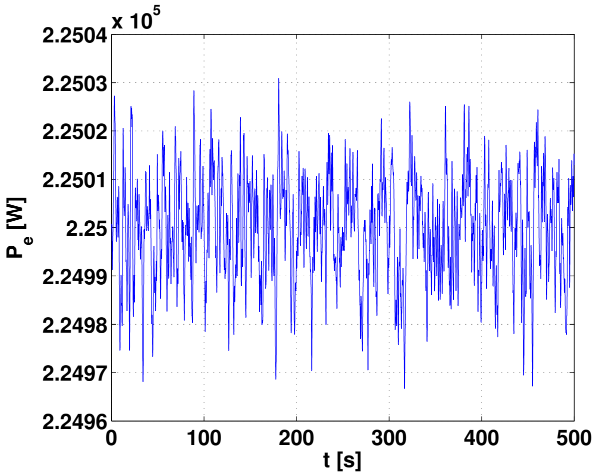


Figure 8.5: Output P_e associated with the state evolution in figure 8.6.

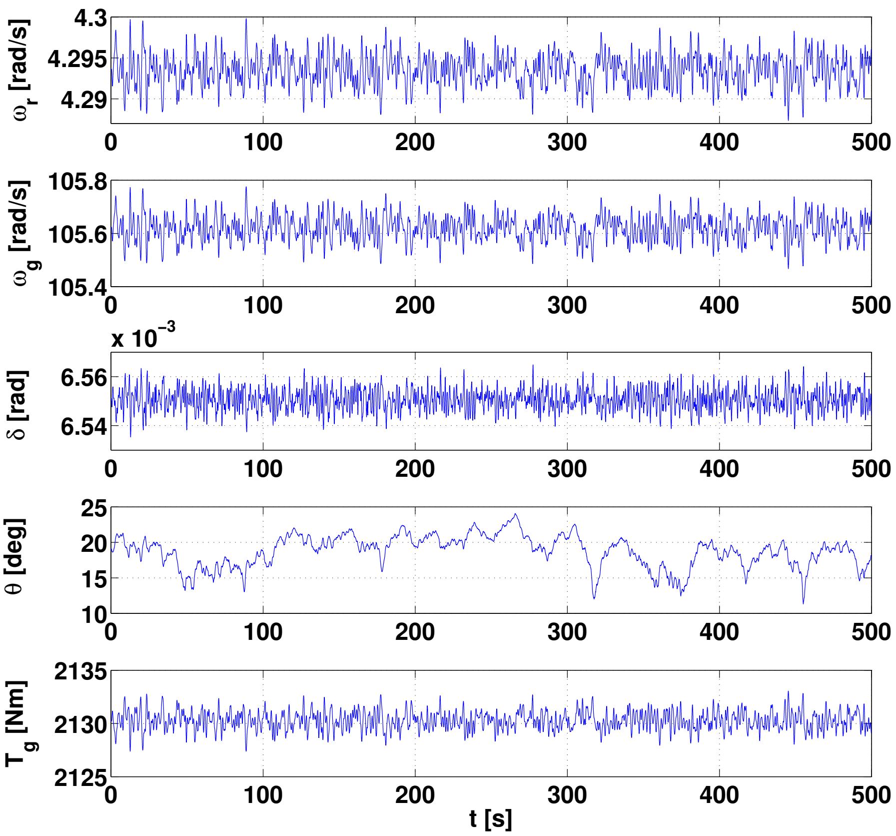
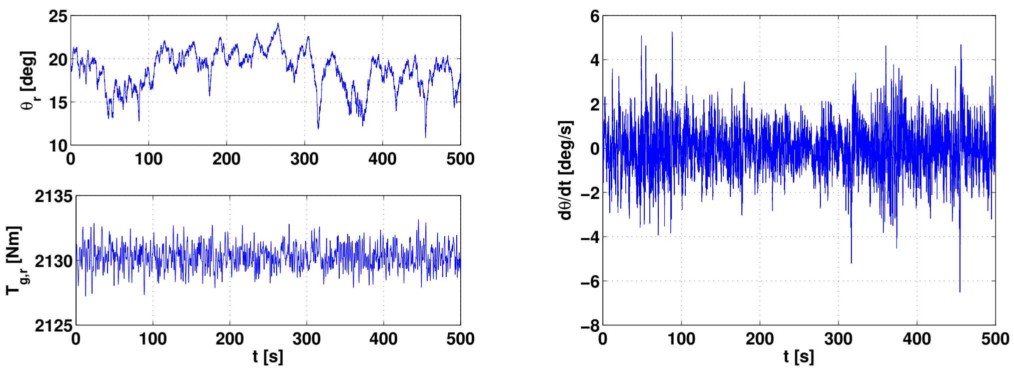


Figure 8.6: Evolution of the states x when subjected to the stochastic wind disturbance in figure 8.4.



(a) input u (b) Pitching speed $\dot{\theta}$

Figure 8.7: Evolution of input and the pitching speed associated with the state evolution in figure 8.6.

8.4.2 Closed loop performance with SM2

Closed loop simulations have been made with SM2 both under ideal conditions and perturbed conditions. The perturbations are the same as the ones introduced for the feedback linearizing (disturbance decoupling) control law in chapter 7. The perturbed and ideal wind states are shown in figure 8.8. Figure 8.9 shows the evolution of the

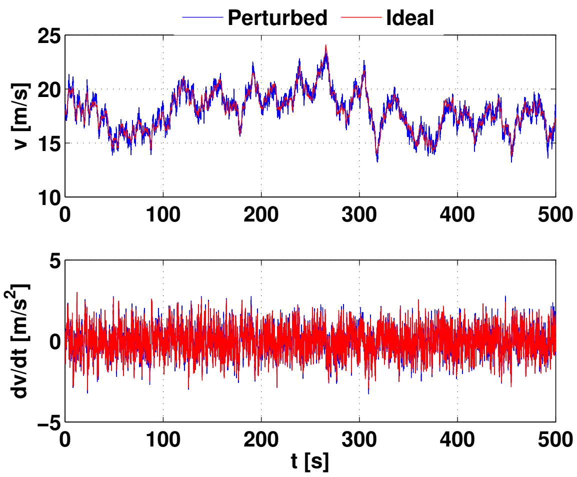


Figure 8.8: Perturbed and ideal wind state measurements. $v_m = 18\text{m/s}$

states, given ideal and perturbed conditions. As expected the transformed state vector has been completely decoupled from the disturbance under ideal conditions, whereas under perturbed conditions the response is degraded. The same is seen to apply for the output in figure 8.10.

The controller design ensures that the trajectory is forced onto the surface under the perturbed conditions. Therefore it is expected that the high frequency perturbation cause an increased control effort. The evolution of the control signal is shown in figure 8.11(a) and illustrates exactly that. The control signal is extremely aggressive when introducing the perturbation. Although the control signals are low pass filtered by the dynamics of the pitch actuator, the pitching speed shown in figure 8.11(b) is still higher than what is physically possible.

Chapter 12 will further analyses the consequence of the more realistic disturbances, which are introduced in connection to the state estimator design in chapter 11.

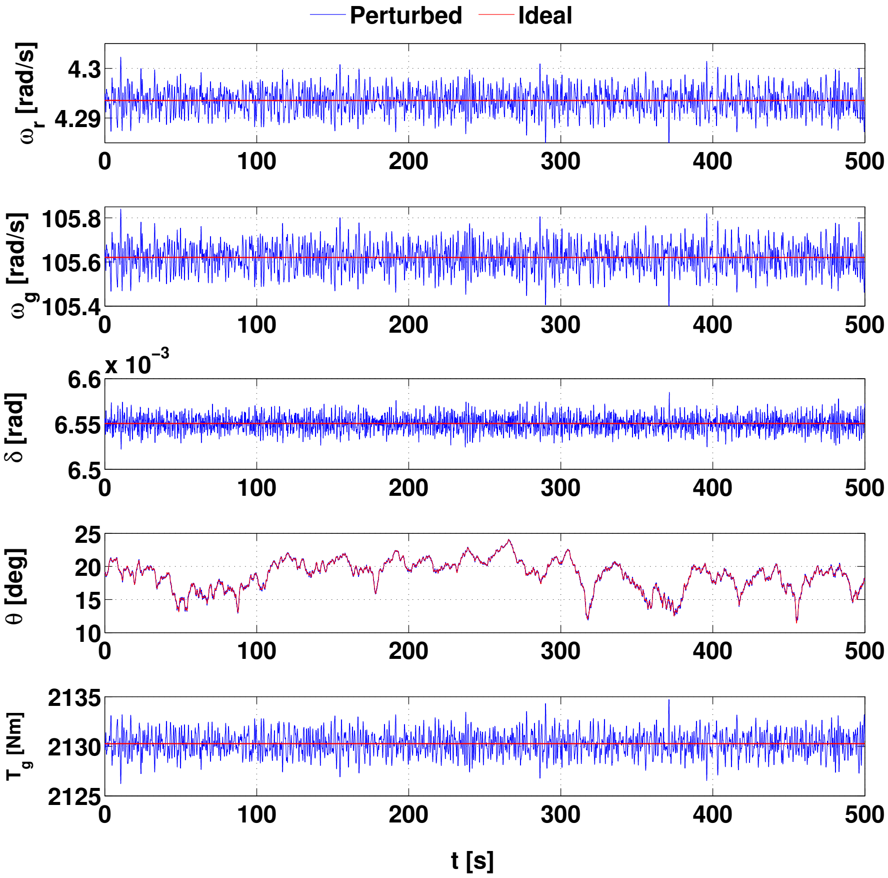


Figure 8.9: Evolution of state when the system is subjected to the true and perturbed wind state measurements in figure 8.8.

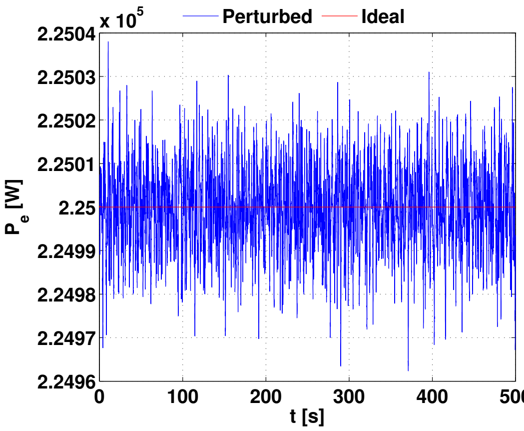
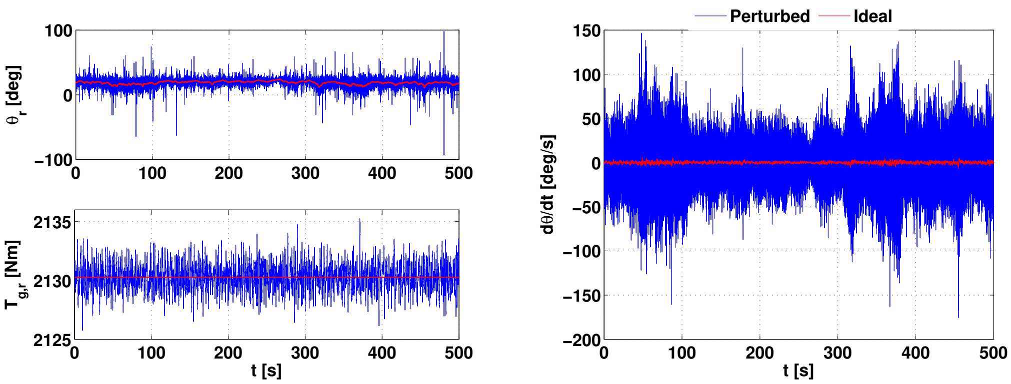


Figure 8.10: Output P_e associated with the perturbed and ideal state evolution in figure 8.9.



(a) Control signals (b) $\frac{d\theta}{dt}$

Figure 8.11: Evolution of input and pitching speed associated with the state evolutions in figure 8.9.

Chapter 9

Inverse nonlinear optimal control

In the previous chapters the linear quadratic (LQ) control method, has been used in connection with the nonlinear control methods. However, although the method has been used to design an optimal controller for e.g. the feedback linearized wind turbine, it is not necessarily optimal with respect to any reasonable criteria, since the control effort that is applied to linearize the system is not considered in the optimization problem. Consequently the robustness properties of the closed loop system might be very poor (Freeman and Kokotovic 1995).

In this chapter it is shown how to obtain a nonlinear optimal controller which is optimal with respect to a *reasonable* criteria, meaning that the resulting control has acceptable robustness properties. However, instead of solving the nonlinear optimization problem directly (which could be done by solving the Hamilton-Jacobi-Bellman (HJB) equation), the *inverse* problem is solved instead i.e. by using a ready-made control law, with an associated cost. The control law that is investigated in this chapter is called Sontag's formula which, on the basis of a so-called Control Lyapunov Function (CLF), supplies an optimal stabilizing control law (Sepulchre et al. 1997). The method is equally straightforward for single-input and multi-input systems. Hence, unlike the other pure nonlinear methods (feedback linearization and sliding mode), the method is applied to the variable speed turbine, without exemplifying the method on the simple constant speed turbine first.

The inverse optimal method has not resulted in an acceptable control law for the turbine. Although the method stabilizes the system, it has not been possible to manipulate the parameters of the control law, such that the the closed loop performance is acceptable.

The outline of the chapter is: Section 9.1 describes the method. This includes a description of the CLF concept, the control law (Sontag's formula) and the associated criteria which is minimized. The method is applied to the variable speed turbine in section 9.2. Simulations with the controller is shown in section 9.3.

9.1 Theory

This section introduces the inverse optimal control method which has been considered in the project. More specifically the method can be described as *inverse optimal control of feedback linearizable systems*. The method takes its basis in (Sepulchre et al. 1997), where it is shown that a stabilizing control law, which minimizes a reasonable criteria, can be found for feedback linearizable systems in the feedback linearizable coordinates. The idea is to find a Control Lyapunov Function for the feedback linearized system (which is always possible). This CLF is then shown to be a CLF for the nonlinear system as well, and can therefore be used to deduce an optimal stabilizing control law for the system.

This section will first give a general description of CLF's and Sontag's formula, before providing the exact details of the method.

9.1.1 Control Lyapunov functions and Sontag's formula

CLF's are an extension of the Lyapunov function concept covered in Appendix B. Whereas Lyapunov functions provide descriptive means of assessing the stability properties of a system, CLF's provide constructive means for establishing a stabilizing control law. Consequently the existence of a CLF is equivalent to stabilizability of a system rather than stability. The mathematical definition of CLF's is:

Definition 9.1 (From (Krstic et al. 1995)) A smooth positive definite and radially unbounded function $V : \mathbb{R}^n \rightarrow \mathbb{R}_+$ is called a Control Lyapunov Function (CLF) for the system

$$\dot{x} = f(x, u) \ , \quad x \in \mathbb{R}^n, \ u \in \mathbb{R}^m \tag{9.1}$$

if

$$\inf_{u \in \mathbb{R}^m} \left\{ \frac{\partial V(x)}{\partial x} f(x, u) \right\} < 0, \quad \forall x \neq 0 \tag{9.2}$$

The definition simply states that $V(x)$ is a CLF if along every trajectory of the system (9.1) the time derivative $\dot{V}(x)$ can be made negative by an appropriate choice of u. As for the previous chapters, only systems affine in the control are considered:

$$\dot{x} = f(x) + G(x)u$$

For systems affine in the control, the equivalent statement is that $V(x)$ is a CLF if and only if

$$L_g V(x) = 0 \Rightarrow L_f V(x) < 0 \tag{9.3}$$

Just like a Lyapunov function it may be difficult to find a CLF or even to determine whether one exits for a system. Fortunately, it is possible for several significant classes of systems to construct CLF's in a systematic way. A CLF can e.g. be obtained through recursive algorithms. One example is the backstepping algorithm; this algorithm both results in a CLF and a stabilizing control law, but is restricted to certain kinds of system structures. As mentioned previously, another procedure is to utilize the structure of feedback linearizable systems.

Given a CLF $V(x)$ for a (nonlinear) system a particular optimal control law which can be derived from the CLF is given by Sontag's formula (Sepulchre et al. 1997):

$$u(x) = \begin{cases} -\left(\frac{a(x)+\sqrt{a(x)^2+(b(x)^T b(x))^2}}{b(x)^T b(x)}\right) b(x) & , \quad b(x) \neq 0 \\ 0 & , \quad b(x) = 0 \end{cases} \tag{9.4}$$

where

$$a(x) = L_f V(x) \qquad b(x) = (L_g V(x))^T$$

Naturally the CLF will be a Lyapunov function for the resulting closed loop system. The criteria function which is minimized, using this control law is:

$$J = \int_0^\infty (\frac{1}{2} p(x) b^T(x) b(x) + \frac{1}{2p(x)} u^T u) dt \tag{9.5}$$

where

$$p(x) = \begin{cases} \frac{a(x)+\sqrt{a(x)^2+(b(x)^T b(x))^2}}{b(x)^T b(x)} & , \quad b(x) \neq 0 \\ 0 & , \quad b(x) = 0 \end{cases}$$

A consequence of the optimality is that the control law guarantees certain robustness properties. A discussion of these can be seen in (Sepulchre et al. 1997), and will not be detailed here.

9.1.2 Inverse optimal control for feedback linearizable systems

In this project it has been utilized that a CLF can be found in a systematic way for the class of feedback linearizable systems in the feedback linearizable coordinates. The idea was initially proposed in (Freeman and Kokotovic 1995), however the exact procedure which is presented below is from (Sepulchre et al. 1997).

The procedure can be described by the following steps:

1. Transform the system to the feedback linearizable coordinates.
2. Find a CLF for the transformed system.
3. Use Sontag's formula to derive an optimal control law for the transformed system.

Transforming a system to the feedback linearizable coordinates, has already been described in chapter 7, and can generally be written in following way:

$$\begin{aligned} \dot{x} &= f(x) + G(x)u \\ &\Downarrow \ \ z = T(x) \\ \dot{z} &= Az + B(f_z(x) + G_z(x)u) \end{aligned} \tag{9.6}$$

where $f_z(x)$: $\mathbb{R}^n \rightarrow \mathbb{R}^m$ and $G_z(x)$: $\mathbb{R}^n \rightarrow \mathbb{R}^{m\times m}$ and $T(x)$ is a diffeomorphic transformation. z are general feedback linearizable coordinates in this context, and should not be confused with the states of the normal form introduced in chapter 7.

A CLF for the transformed system can be found by noting that a Lyapunov function for a closed-loop system, is a CLF for the corresponding open-loop system. Hence, by finding a particular stabilizing control law for the transformed system (9.6), a CLF is attained by finding a Lyapunov function for the controlled system. Naturally no advantage has been gained if the closed-loop system is nonlinear. However, since the transformed system is feedback linearizable, a natural choice is simply to choose a feedback linearizing control law which renders the closed loop system linear.

$$\dot{z} = Az + B\upsilon \ , \quad \upsilon = -Kz$$

As is proven in appendix B a Lyapunov function can always be found for a stable linear system.

Instead of finding the stabilizing linear control law $\upsilon = -Kz$ explicitly, a CLF is actually found in the process of obtaining a linear quadratic control law. Recalling that the solution to the algebraic Richatti equation P is a symmetric positive definite matrix, it is easily shown that

$$V(z) = z^T P z$$

is a CLF for the system (9.6) i.e. a Lyapunov function for the LQ controlled feedback linearized system. Therefore, following the approach of (Sepulchre et al. 1997) the CLF will be found through the solution to the algebraic Riccati equation:

$$A^T P + PA - PBB^T P < 0 \tag{9.7}$$

Calculating the directional derivatives of the CLF and inserting these into Sontag's formula, finally results in the nonlinear optimal inverse control law. It is stressed that the directional derivatives are calculated on the basis of the system (9.6) i.e. the system in the *feedback linearizable coordinates*, therefore the directional derivatives are:

$$\begin{aligned} L_p V(z) &= \frac{\partial V(z)}{\partial z} \underbrace{(Az + Bf_z(x))}_{p(x)} \\ L_Q V(z) &= \frac{\partial V(z)}{\partial z} \underbrace{BG_z(x)}_{Q(x)} \end{aligned}$$

where

$$\dot{z} = p(x) + Q(x)u$$

It is seen that, given a feedback linearizable system, it is straightforward to obtain the stabilizing inverse optimal control law. However, there is no systematic procedure for obtaining adequate closed loop performance, since the cost cannot be manipulated. Hence, it is simply a matter of recursively calculating a specific CLF and simulating the resulting system performance.

9.2 Inverse optimal control of the variable speed model

As mentioned in the previous section, the inverse approach for solving the nonlinear optimal control problem, has the obvious drawback that there is no systematic procedure for manipulating the criteria function (9.5) associated with the control law (9.4). Therefore, designing an inverse optimal control law for the variable speed turbine has been a matter of recursively choosing a CLF for the feedback linearizable system and simulating the performance of the resulting closed loop system. Within the scope of this project, this has not resulted in a closed loop system with acceptable performance.

Following the procedure of the previous chapters the control law has been designed for the system in the feedback linearizable coordinates:

$$\xi = T_\xi(x) = \begin{bmatrix} x_1 \\ L_f x_1 \\ x_2 \\ x_3 \\ x_5 \end{bmatrix}$$

which results in the dynamic system

$$\begin{aligned} \dot{\xi} &= A_\xi \xi + B_\xi (f_\xi(x) + G_\xi(x) u) \\ &= \underbrace{A_\xi \xi + B_\xi f_\xi(x)}_{p(x)} + \underbrace{B_\xi G_\xi(x)}_{Q(x)} u \end{aligned} \tag{9.8}$$

where A_ξ, B_ξ, $f_\xi(x)$ and $G_\xi(x)$ are given in equation (7.27) in chapter 7. Experiments have been made, in which the system has been augmented with the wind model and the integral state $\int \bar{\omega}_g$. However, the means for finding a CLF is based on the linear feedback linearized system, hence $\int \bar{P}_e$ cannot be included in the design since it introduces a nonlinearity. Newer the less, it has not been accomplished to obtain an acceptable control law for the variable speed turbine. Therefore only a basic design for the system (9.8) will be shown, such that the limitations of the method is readily illustrated.

Shifting the origin to the equilibrium of the system, the CLF is chosen as:

$$V = \bar{\xi}^T P \bar{\xi} \ , \quad \bar{\xi} = \xi - \xi^*$$

where P is chosen such that it fulfills the algebraic Riccati equation:

$$-1000I = A_\xi^T P + P A_\xi - P B_\xi B_\xi^T P$$

where $I_{5\times 5}$ is a unity matrix. The Lie derivatives for the controller design are consequently:

$$\begin{aligned} L_f V(x) &= \frac{\partial V}{\partial \xi}(A_\xi \xi + B_\xi f_\xi(x)) \\ &= 2\xi^T P(A_\xi \xi + B_\xi f_\xi(x)) \\ L_g V(x) &= \frac{\partial V}{\partial \xi}(B_\xi G_\xi) \\ &= 2\xi^T P B_\xi G_\xi \end{aligned}$$

Inserting these in Sontag's formula results in the stabilizing control law. As with the other nonlinear control laws the resulting control law is notationally exhaustive, hence it will be written in no further detail.

9.3 Simulations

This section presents a deterministic and a stochastic simulation with the nonlinear inverse control law, derived in the previous section.

9.3.1 Deterministic simulation

A deterministic simulation with constant wind speed $v = 18$m/s has been made to illustrate that the resulting control law stabilizes the system and renders the time derivative of the CLF negative. The system has been started outside the equilibrium point. More specifically with the initial states:

$$\bar{x}_0 = \begin{bmatrix} 0 & 0 & 0 & 2 & 100 \end{bmatrix}^T \tag{9.9}$$

Figure 9.1 shows the evolution of the states and figure 9.2(a) show the control effort. The Corresponding values of the CLF and its derivative is seen in figure 9.2(b)

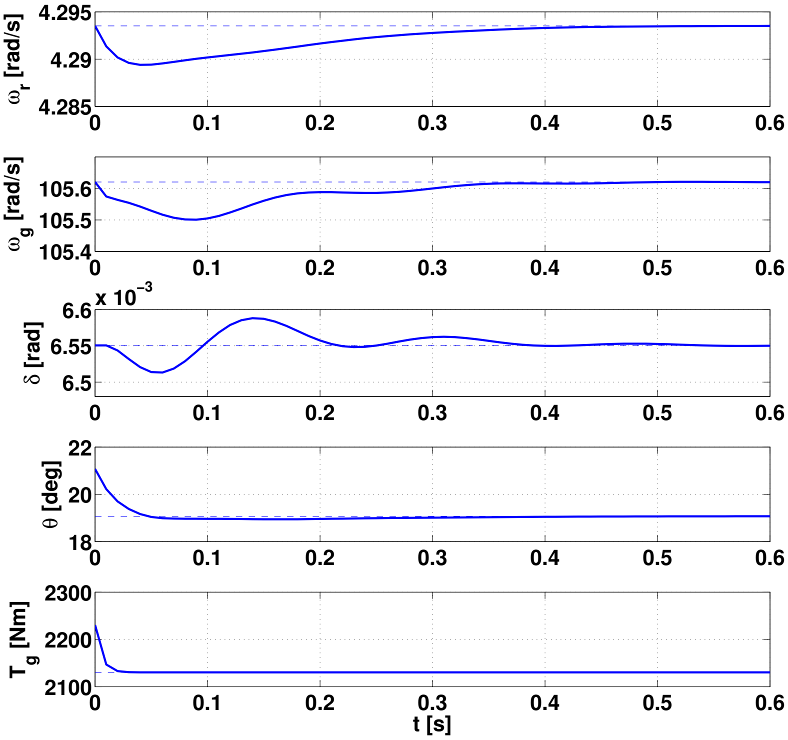
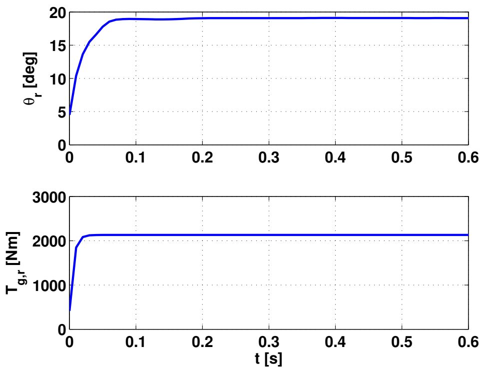
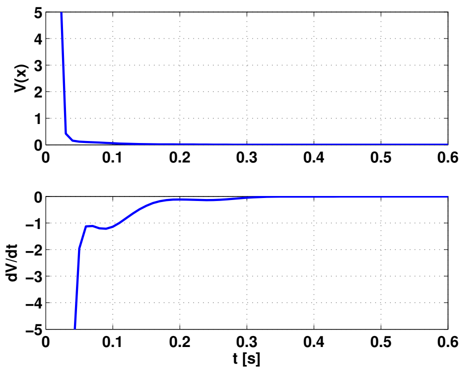


Figure 9.1: State evolution when the closed loop system is started with the initial conditions in equation (9.9).



(a) u



(b) $V(x)$ and $\dot{V}(x)$

Figure 9.2: Control signal and CLF evolution.

9.3.2 Stochastic simulation

The resulting closed loop system has been simulated under the influence by the stochastic wind disturbance in figure 9.3 ($v_m = 18$m/s). Figure 9.4 shows the evolu-

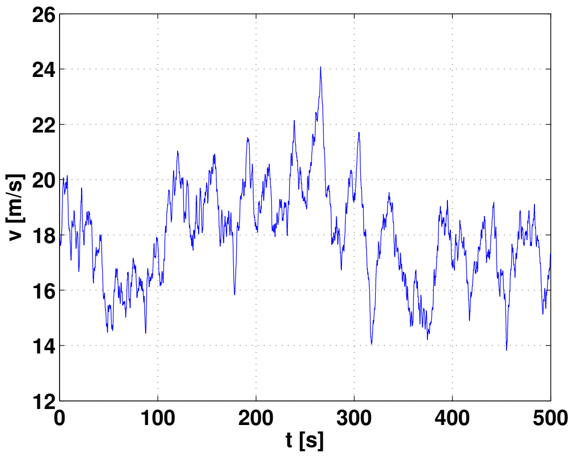


Figure 9.3: Wind disturbance with constant mean wind speed v_m=18 m/s

tion of the states when controlled by the optimal controller while 9.5(a) and 9.5(b) shows the corresponding control actions and output P_e.

Comparing these plots with the simulations in chapters 6-8, it is seen that the performance is very poor. This can most likely be attributed to the almost non-existing control effort on the generator torque. The inability to target the weight on T_g in the criteria function, makes it impossible to deal with this problem (At least in a structured way). As long as this weight is not targeted and reduced in size, there is little idea in adding further complexity in the design. All this being said it is worth mentioning that an advantage has indeed been gained as compared to the open loop system. Furthermore, the method is, at first glance, robust towards the wind disturbance.

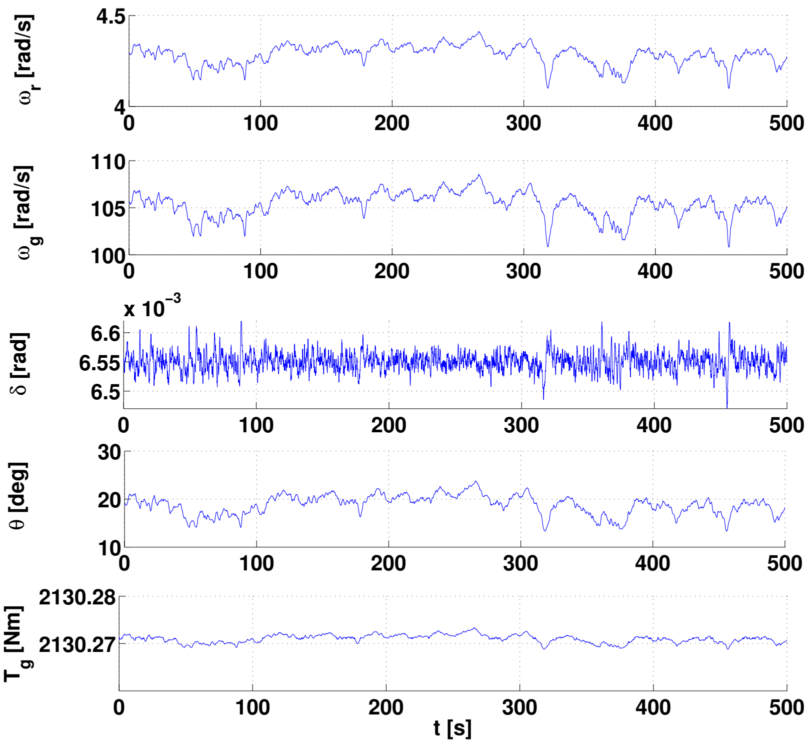
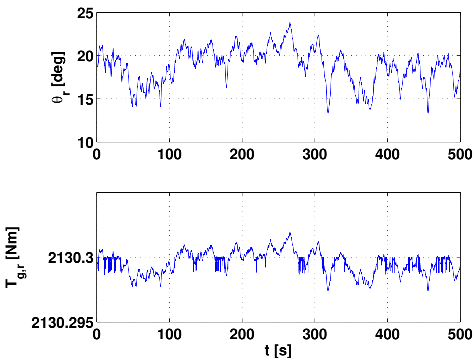
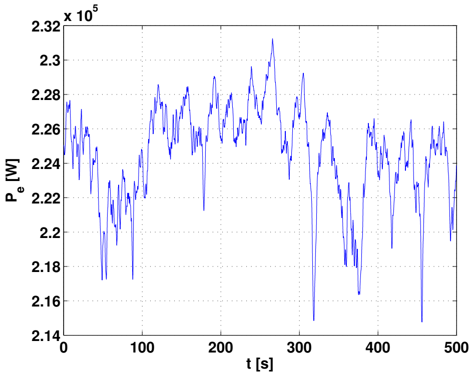


Figure 9.4: State evolution closed loop system, when subjected to the stochastic disturbance in figure 9.3.



(a) Control signal



(b) Output P_e

Figure 9.5: Control input to model 2 when controlled by the nonlinear inverse controller.

CHAPTER 10

Summary

This chapter briefly summarizes the results, attained the in chapters 6 to 9. This is only an intermediate summary, based on the idealized control setup which has been presented so far.

It was seen that 'perfect' performance could be obtained with the disturbance decoupling control laws (feedback linearization and sliding mode control (SM2)). Naturally, decoupling can only be attained in the idealized setting. This, was readily illustrated by introducing a small pertubation in the wind measurements. The results obtained in the previous chapters has therefore not been used to compare the controllers. However, they merely indicate a potential advantage in using these control laws. The next chapter will introduce a more realistic control setup, for the controllers, and will consequently serve as the primary setup for comparing the nonlinear designs.

Besides the disturbance decoupling sliding control law (SM2), a simple sliding mode controller (SM1) was derived which assumed no knowledge of the wind. However, by restricting the trajectory to the sliding surface, it was seen to provide better performance than the gain scheduled controller which uses wind knowledge in the control.

The inverse optimal control law did not result in an acceptable control law. Although it stabilizes the system, it has the serious drawback, that the criteria associated with the control law, cannot be manipulated in a systematic manner.

Part III

Control with incomplete state knowledge

Chapter 11

State estimation

Due to practical and economical reasons it cannot be expected that all states of the variable speed wind turbine are measured in real life. Furthermore the measurements available are bound to be corrupted by a certain level of noise. To provide a more realistic setting when evaluating the control designs a nonlinear observer has therefore been designed for the wind turbine. More specifically an Extended Kalman Filter (EKF), which takes into account the assumed levels of the process and measurement noise.

The outline of this chapter is as follows: Section 11.1 introduces the EKF. Section 11.2 describes what measurements which are assumed to be available, the models of the noise sources and the consequent EKF design for the wind turbine. Section 11.3 validates the design through simulations.

11.1 The Extended Kalman Filter

The linear Kalman filter is the optimal state predictor (in a least square sense) for linear models:

$$\dot{x} = Ax + Bu + w \ , \quad y = Cx + v$$

where w and v are process and measurement noise signals (not to be confused with the wind model states or the wind speed), which are assumed to be Gaussian distributed white noise processes. Briefly stated, the Kalman filter consists of a state-update and a covariance-update which are based on the linear model and the assumed intensity of the noise sources R_w and R_v.

$$\begin{aligned}\dot{\hat{x}}(t) &= A\hat{x}(t) + Bu(t) + K(t)(y_m(t) - C\hat{x}(t)) \\ \dot{P} &= AP(t) + P(t)A^T + R_w - P(t)C^T R_w^{-1} CP(t)\end{aligned}$$

where $K(t) = P(t)A^T R_v^{-1}$ is denoted the Kalman gain.

The EKF is basically a nonlinear extension of the Kalman filter where the model is allowed to be nonlinear i.e.:

$$\begin{aligned}\dot{x} &= f(x,u) + w \\ y &= h(x) + v\end{aligned}$$

However, the nonlinear expressions cannot be used in covariance-update equation. Instead, the Jacobians are computed from which the covariance and consequently the Kalman gain can be found. The resulting state-update and covariance-update are shown below (Gelb et al. 1974).

$$\dot{\hat{x}}(t) = f(\hat{x}(t), u(t)) + K(t)\left(y_m - h(\hat{x}(t), u(t))\right) \tag{11.1}$$

$$K(t) = P(t)F^T R_v^{-1} \tag{11.2}$$

$$\dot{P} = FP(t) + P(t)F^T + R_w - P(t)H^T R_w^{-1} HP(t) \tag{11.3}$$

where

$$F = \left.\frac{\partial f}{\partial x}\right|_{\substack{x=\hat{x}\\u=u}} \qquad H = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}}$$

Equation (11.3) is known as the Riccati equation. It is common practice to assume that this equation is static resulting in the algebraic Riccati equation:

$$0 = FP + PF^T + R_w - PH^T R_w^{-1} HP \tag{11.4}$$

The algebraic Riccati equation has been used in the project to simplify the implementation. Figure 11.1 shows a block diagram of the continuous EKF.

11.2 EKF for the variable speed model

This section describes the EKF design for the variable speed wind turbine. This includes a description of noise models, the nonlinear state and observation equations and the associated Jacobians.

11.2.1 State and observation equations

Being driven by a white noise sequence $e \in \mathcal{N}(0, 1^2)$, the nonlinear wind augmented turbine (2.29) is naturally used as the state equation in the state-update equation

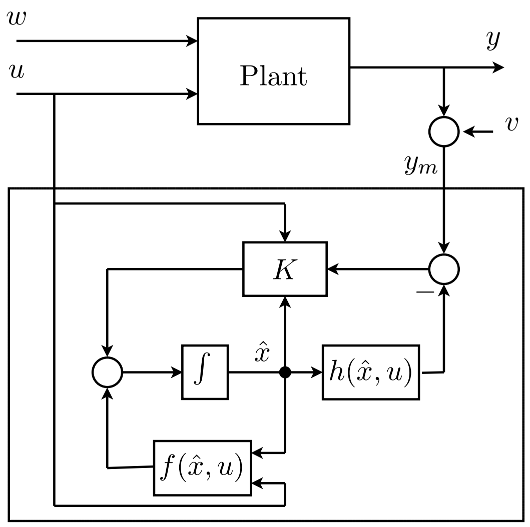


Figure 11.1: Block diagram of the extended Kalman filter. Note that the Kalman gain is dependent of the current state estimate $\hat{x}$ and input u.

(11.1). The corresponding linear model (2.30) is accordingly used in the covariance-update (11.3).

The states of the wind turbine model are all reasonably easy to measure, given the adequate hardware. However due to economical or practical reasons it is not likely that all the states are measured in reality. The states of the wind model are on the other hand impossible to measure. In fact, the effective wind speed introduced in chapter 2 is an abstraction, the purpose of which is to simplify the modeling perspective. All in all, to increase the credibility, it will be assumed that the generator speed and the pitch θ are the only states which are measured. Furthermore the output power P_e which is a nonlinear function of the states is assumed to be measured. The resulting observation equation used in the design is therefore:

$$h(x) = \begin{bmatrix} x_2 x_5 \\ x_2 \\ x_4 \end{bmatrix} = \begin{bmatrix} P_e \\ \omega_g \\ \theta \end{bmatrix} \tag{11.5}$$

The corresponding Jacobian used in the covariance-update is:

$$H = \left.\frac{\partial h(x)}{\partial x}\right|_{x=\hat{x}} = \begin{bmatrix} 0 & \hat{x}_5 & 0 & 0 & \hat{x}_2 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The mean wind speed is a physical quantity which is easily measured. Hence, perfect knowledge of the mean wind speed will still be assumed.

11.2.2 Process noise

The white noise process with intensity $\sigma = 1^2$ which drives the wind model is naturally viewed as process noise. Using this as the only noise source, would result in zero Kalman gain associated with the actuator states, since the wind does not affect the actuator states. From a practical point of view this is undesirable since model uncertainties in the actuators are inevitable. Therefore the noise covariance associated with the actuator states has been chosen nonzero in the design. Actuator noise will not be introduced, when controlling the turbine in chapter 12, hence it is purely a design parameter.

Looking at the linear wind augmented turbine (2.30) it is seen that the white noise signal e which drives the wind model, affects the states through the input matrix B_w. Equivalently, the actuator noise signal ε_u is chosen to affect the states through the input matrix B. The resulting process noise signal is consequently:

$$w = \begin{bmatrix} B_w & B \end{bmatrix} \begin{bmatrix} e \\ \varepsilon_u \end{bmatrix}$$

The intensity matrix associated with the process noise is:

$$\begin{aligned} R_w &= \begin{bmatrix} B_w & B \end{bmatrix} \begin{bmatrix} R_e & 0 \\ 0 & R_{\varepsilon_u} \end{bmatrix} \begin{bmatrix} B_w & B \end{bmatrix}^T \\ &= \begin{bmatrix} B_w & B \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0.1 \end{bmatrix} \begin{bmatrix} B_w & B \end{bmatrix}^T \end{aligned}$$

The level of the noise intensities associated with the actuators have no physical interpretation, but simply chosen such that they are small relative to the measurement noise intensities (which are presented shortly). Doing this, the EKF is designed such, that it has more faith in the actuator model than in the measurements.

11.2.3 Measurement noise

It is assumed that the maximum values of the measurement noise signals are approximately 3% of the stationary values at $v_m = 18\text{m/s}$. Given a Gaussian distribution 99.7% of the values will be within 3 standard deviation. Consequently the covariance matrix is appropriately chosen as:

$$Q = \begin{bmatrix} \left(\frac{0.03 \cdot P_{e0}}{3}\right)^2 & 0 & 0 \\ 0 & \left(\frac{0.03 \cdot \omega_{g0}}{3}\right)^2 & 0 \\ 0 & 0 & \left(\frac{0.03 \cdot \theta_0}{3}\right)^2 \end{bmatrix} \tag{11.6}$$

where it has been assumed that the noise signals are uncorrelated.

Since white noise has infinite power/variance the covariance matrix above does *not* equal the intensity matrix R_v. The intensity has been obtained by assuming that the measurement noise has been generated by filtering white noise with the following first order low pas filter:

$$\dot{x} = -\omega_{bw} x + \omega_{bw} e_t \tag{11.7}$$

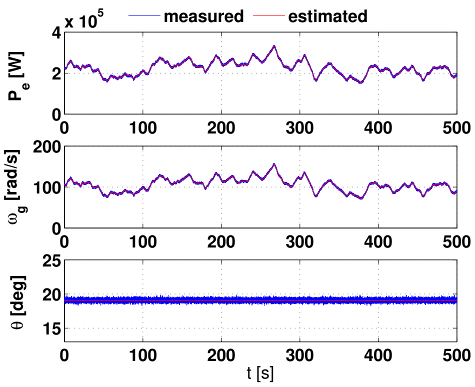
where the bandwidth ω_{bw} is much larger than the bandwidth of the system. The underlying intensity of the white noise model can be calculated e.g. by using the continuous Lyapunov equation (Hendricks et al. 2004).

$$R_v = \frac{2}{\omega_{bw}} Q$$

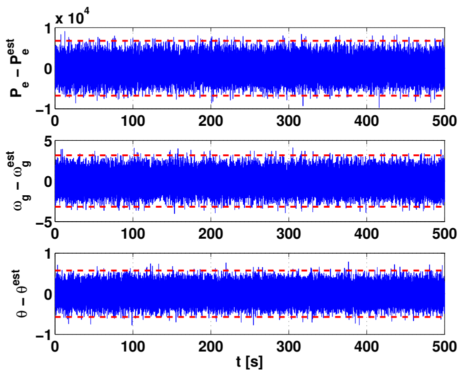
The bandwidth has been set at ω_{bw} = 500 rad/s which is approximately 25 times higher than the bandwidth of the open loop system.

11.3 Simulations

This section presents an open-loop simulation, which has been used to validate the EKF design. The validation is mainly based on whiteness tests of the innovation process (estimation error). As mentioned, process noise has been added to the actuator states in the EKF design, such that the associated elements in the Kalman gain are nonzero. The system has therefore been simulated with process noise on the actuator states such that appropriate validation is possible. The simulation has been made with a stochastic wind disturbance with mean wind speed v_m = 18m/s.



(a) Measurements and estimates



(b) Innovation processes

Figure 11.2: Plots showing the measurements,estimates and the innovation processes. The horizontal line in the innovation plots indicates the 3σ level

Figure 11.2(a) shows the evolution of the outputs P_e, ω_g and θ together with the estimated evolution. Figure 11.2(b) shows the evolution of the corresponding estimation errors (the innovation processes). The innovation process of a regular Kalman filter is a white noise process with variance equaling that of the measurement noise process. A quick check whether the variance is correct is to see whether three standard deviations approximately corresponds to the maximum values (indicated by the horizontal lines in figure 11.2(b)). To consolidate the hypothesis that the innovation process is white, the auto-covariance of the sampled process has been calculated. The result is seen in figure 11.3. The stippled lines indicate the limits of a 99% confidence

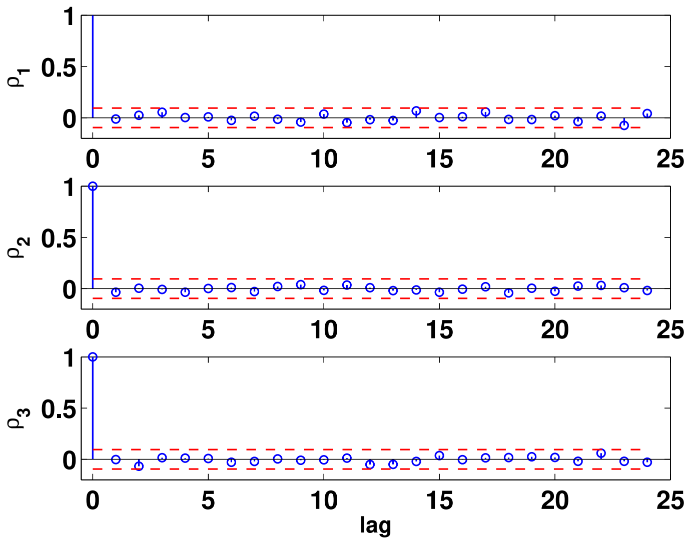


Figure 11.3: Auto-covariance of the innovations processes. A 99% confidence interval is indicated by the horizontal lines.

intervals, which have been made under the assuming that the distribution of the estimated autocorrelations is given by (Madsen 2001):

$$\hat{\rho}(k) \in_{approx} N(0, 1/N), \qquad k \neq 0$$

where N is the number of samples used to calculate the estimates. All in all, the simulations are in consistence with theory (although the EKF inherently is an ad hoc solution) and it is concluded that the nonlinear filter works and is correctly dimensioned.

To illustrate the precision of the estimates, the state estimation errors ($\tilde{x} = x - \hat{x}$) are shown in table 11.1. The wind states exhibit the largest estimation error, compared to the fluctuations of the true states. Especially the estimate of the wind acceleration is seriously affected. This is readily seen in Figure 11.4 which shows the true and estimated evolution of the wind states. The next chapter illustrates the consequences of controlling the turbine based, on the state estimates obtained with the EKF.

State	$\tilde{\omega}_r$	$\tilde{\omega}_g$	$\tilde{\delta}$	$\tilde{\theta}$	$\tilde{T}_g$	$\tilde{v}$	$\tilde{\dot{v}}$
Var	2.964e-5	1.780e-2	4.284e-10	2.902e-4	4.83e-1	3.83e-2	5.67e-1

Table 11.1: Variance of state estimation error.

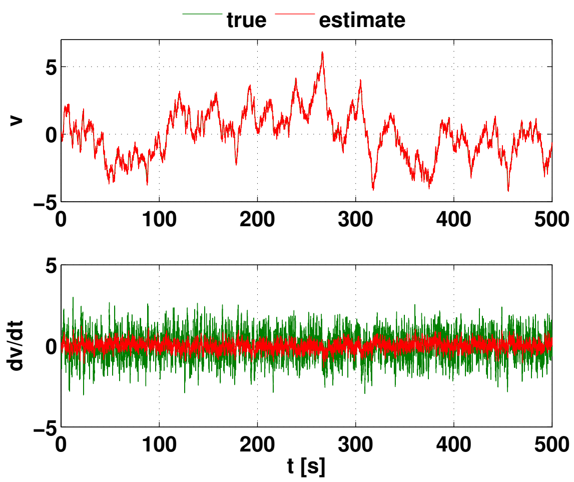


Figure 11.4: True and estimated evolution of the wind states.

Chapter 12

Simulation with incomplete state knowledge

Chapter 6 to 9 clearly showed that superior performance could be obtained with feedback linearization and sliding mode control. However, these results where obtained on the basis of a perfect model and assumed perfect knowledge of the states. This chapter illustrates the consequence of having incomplete state knowledge and therefore controlling the wind turbine on the basis of estimated states.

The setup that has been used, is seen in figure 12.1. As illustrated, the states are

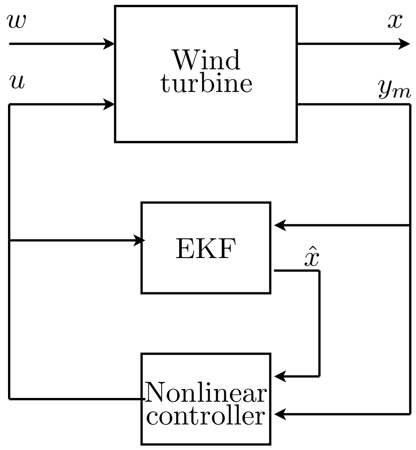


Figure 12.1: Control setup with nonlinear controller and state estimator

estimated, based on the noise-corrupted measurements of P_e, ω_g and θ using the EKF introduced in chapter 11. The control decisions are based on the state estimates and integral control is done directly on the measurements.

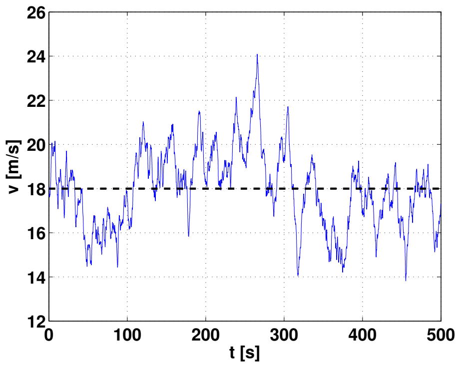
The controllers that are tested in this chapter includes:

- Gain scheduling (• GS)
- Feedback linearization (• FL)
- Sliding mode without wind measurements (• S1)
- Sliding mode with wind measurement (• S2)

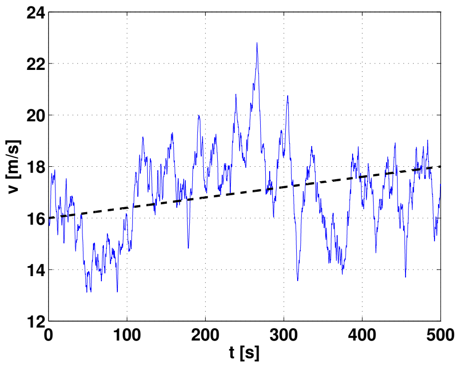
In the parentheses are shown the abbreviations and colors that are associated with the methods in the following. Since acceptable performance has not been obtained with the inverse optimal control method in chapter 9, this method will not be considered in this chapter. Two types of simulations have been made:

- Simulations with a stochastic wind field with constant mean $v_m = 18\text{m/s}$. The simulation time is $t_{sim} = 500\text{s}$.
- Simulations with a stochastic wind field with mean wind speed drifting from 16m/s to 18m/s. The simulation time is $t_{sim} = 500\text{s}$.

The associated stochastic wind sequences are shown in figure 12.2. Simulating with



(a) Constant mean 18 m/s



(b) Drifting mean 16-18 m/s

Figure 12.2: Stochastic wind disturbances used in the simulations. The stippled line shows the evolution of the mean wind speed.

constant mean wind speed, naturally means that the gain-scheduled controller effectively is a point design.

To reduce the overhead, the simulation data associated with the methods are merged together. Although this hides the evolution of the individual signals, it makes it easier to identify differences between the methods. The variance of the signals will be used as a basis of comparison, since this reflect the power in the signal.

12.1 Simulations with constant mean wind speed

The following pages present plots and key figures from the simulations with constant mean wind speed. The figures are summarized below.

Figure 12.3 shows the evolution of the output and the states. Figure 12.4 on page 111 shows the estimated variance of the signals. Naturally, the disturbance decoupling control laws FL and S2 are unable to decouple the disturbance given the perturbated conditions. It is seen that the states associated turbine mechanics exhibit a significant larger variance with the linear controller than the nonlinear designs. Consequently the stress on the turbine mechanics is smaller when controlled by the nonlinear designs. By increasing the control effort on the generator torque, the gain scheduled controller however manages to obtain almost equal output power P_e as compared to the nonlinear designs.

The variance of the states does not tell the whole story. Figure 12.5(a) shows that the performance of the SMC designs has been attained at the cost of unrealistic fast pitching motion. Table 12.5(b) on page 111 shows the key figures associated with the plot. The feedback linearizing controller stays within the actuator limits 99.9%, however, since peaking is allowed, the performance will not be drastically affected by the actuator limits. The pitch speed of the linear design is well within the limits. All in all, it is concluded that only the gain scheduled and feedback linearizing controller are realistic from a practical point of view.

The aggressive pitch with the SMC designs, is a result of an extremely aggressive pitch reference. The pitch and torque reference is seen in figure 12.6 on page 112 which also shows the variance of the control signals. The pitch control signals associated with the sliding mode controllers contain significantly more power than the other designs. This is a consequence of the discontinuous control signal, which forces the trajectory onto the surface whenever it leaves it. There are several ways of fixing the problem: For example filtering the control to remove the high frequency contents or by introducing a wide boundary layer around the sliding surface. However, unmodeled actuator dynamics will introduce chattering, and widening the boundary layer will reduce the mathematical justification of the sliding mode concept. Neither approach, has been considered explicitly in the context of this project.

It is worth noting that the performance of the two sliding mode controllers is almost equal. Consequently, under the perturbated conditions, the sliding mode controller works equally well with and without knowledge of the wind. Since the wind model used in this project does not truly reflect wind conditions in real life, this is an interesting result.

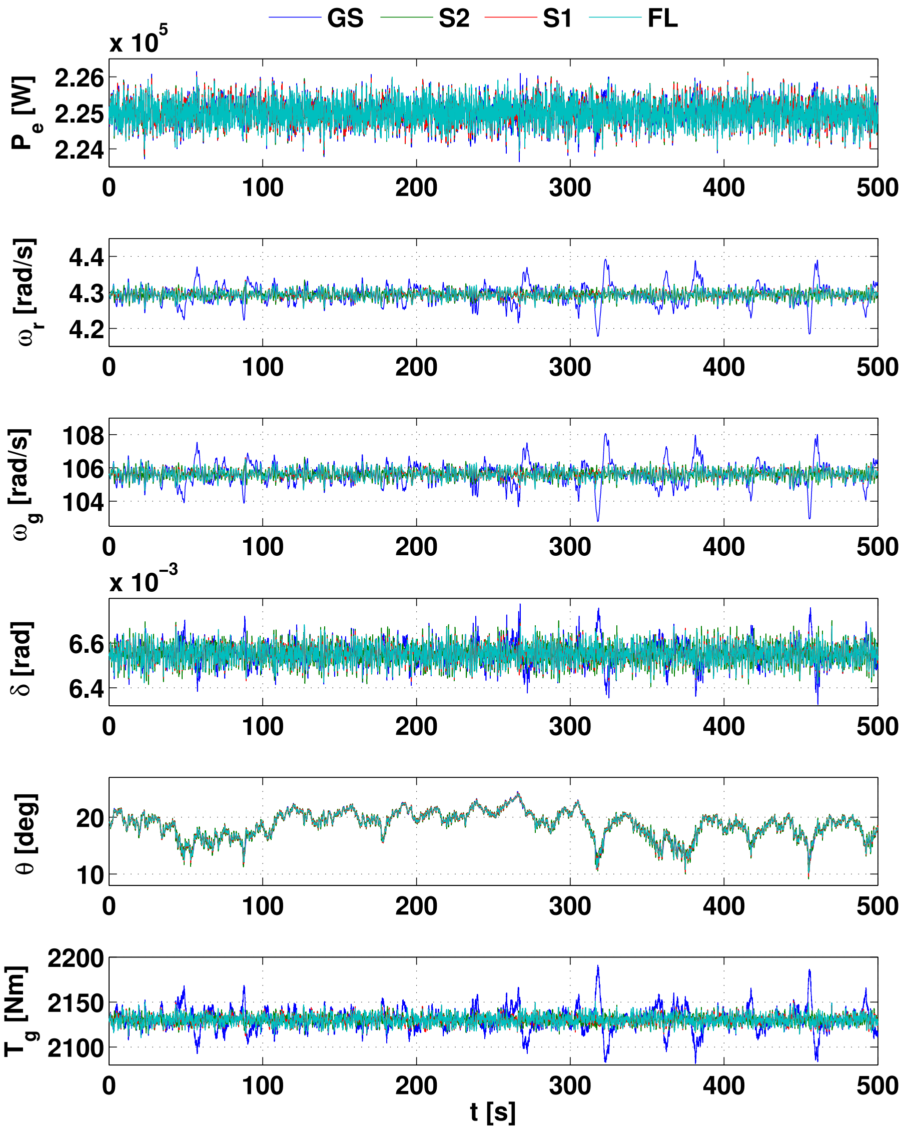


Figure 12.3: Output and state evolution when subjected to the stochastic wind disturbance in figure 12.2(a).

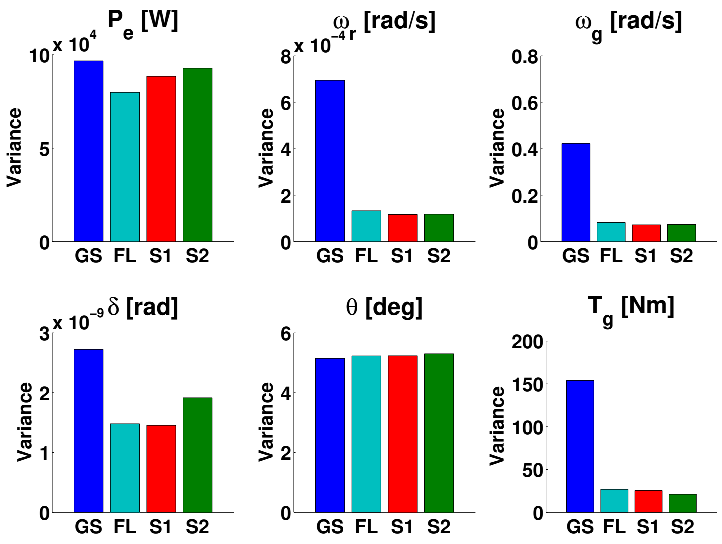
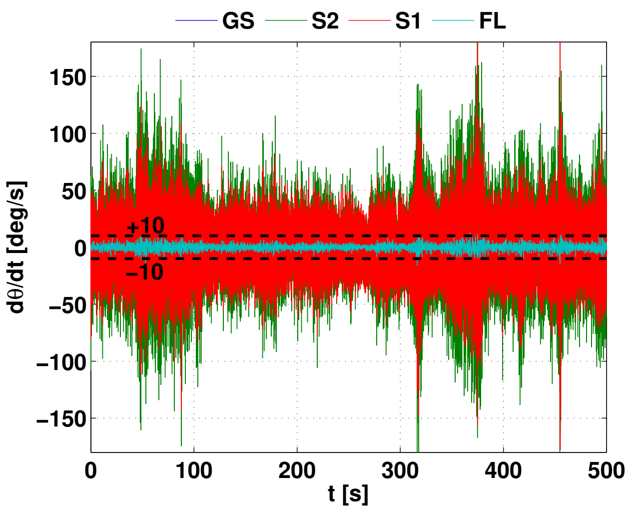


Figure 12.4: Variances of output and states associated with the signal evolution in figure 12.3.

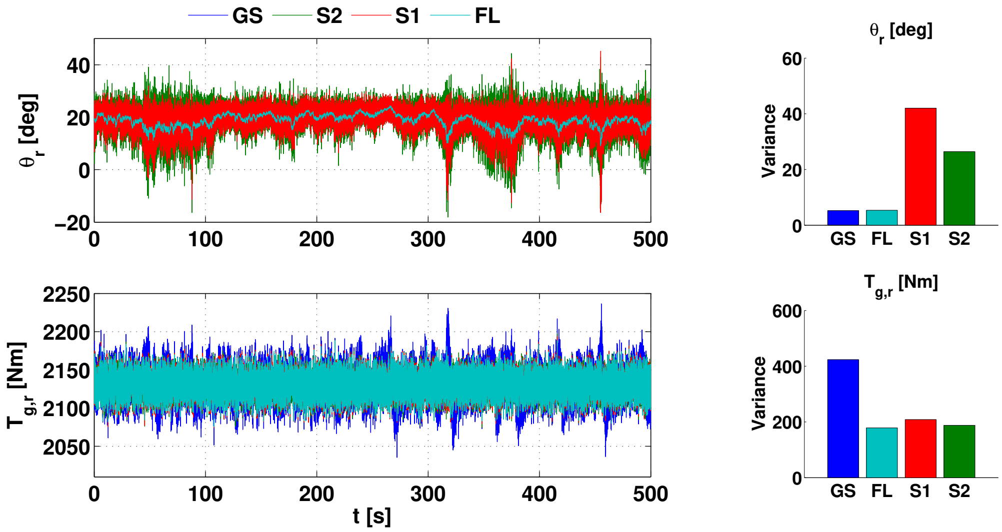


(a) $\dot{\theta}$

$\dot{\theta}$	min / max	OK%
GS	-7.70 / 8.37	100
FL	-15.8 / 12.72	99.9
S1	-178 / 217	39.5
S2	-188 / 215	31.1

(b) Key figures

Figure 12.5: Left: Pitch speed associated with the pitch evolution in figure 12.3. Right: Extremum values for the pitch speed and the relative period that the signal stays within the limits.



(a) u (b) Variance

Figure 12.6: Left: Evolution of control signals associated with the state evolution in figure 12.3. Right: Variance of control signals.

12.2 Simulations with drifting mean wind speed

The following pages present plots and key figures from the simulations with drifting mean wind speed. The plots are analog to those presented in the previous section and only reveals little additional knowledge. The figures will therefore not be commented individually.

Figure 12.8 on the next page shows the state evolution and the corresponding output power P_e, for the various control methods. Figure 12.7 shows the variance of the states and output. Figure 12.9 shows the speed of the pitching motion and a table with associated key figures. Figure 12.10 shows the control signals and the variance of the signals.

Only small details differentiates the plots in this section from the plots in the previous section. Most notable is a general increase in the pitch speed and an increased variance of the pitch. However, the increased pitch variance, is a natural consequence of the turbine being subjected to a wider range of wind speeds, than was the case in the previous section.

All considered, none of the controllers seems to be significantly affected by the changing wind conditions i.e. changes in the mean wind speed.

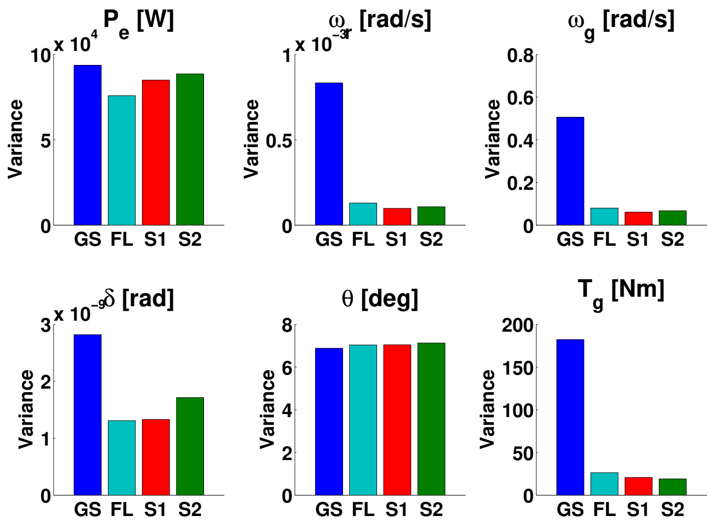


Figure 12.7: Variances of output and states associated with the signal evolution in figure 12.8.

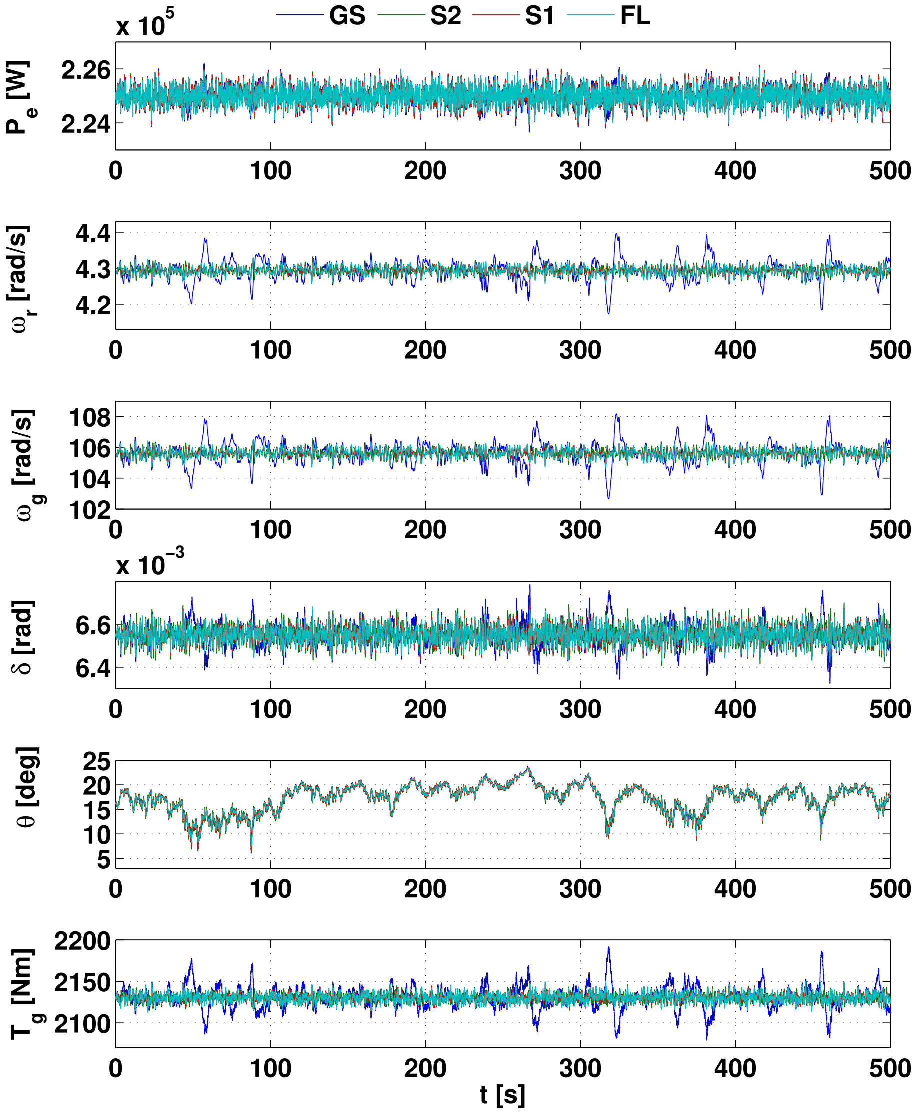
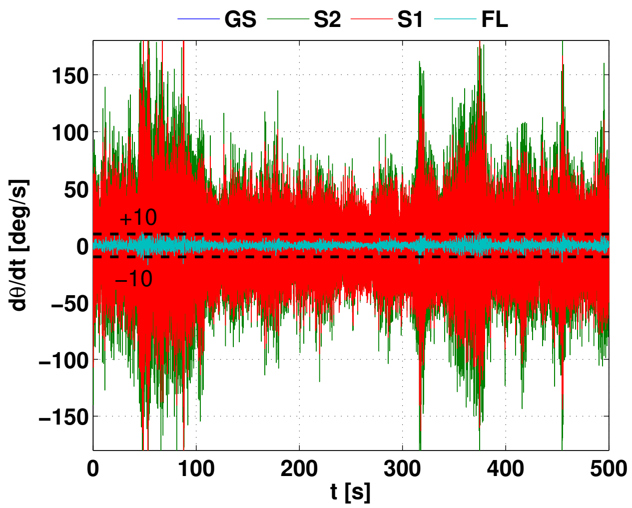


Figure 12.8: Output and state evolution when subjected to the stochastic wind disturbance in figure 12.2(b).

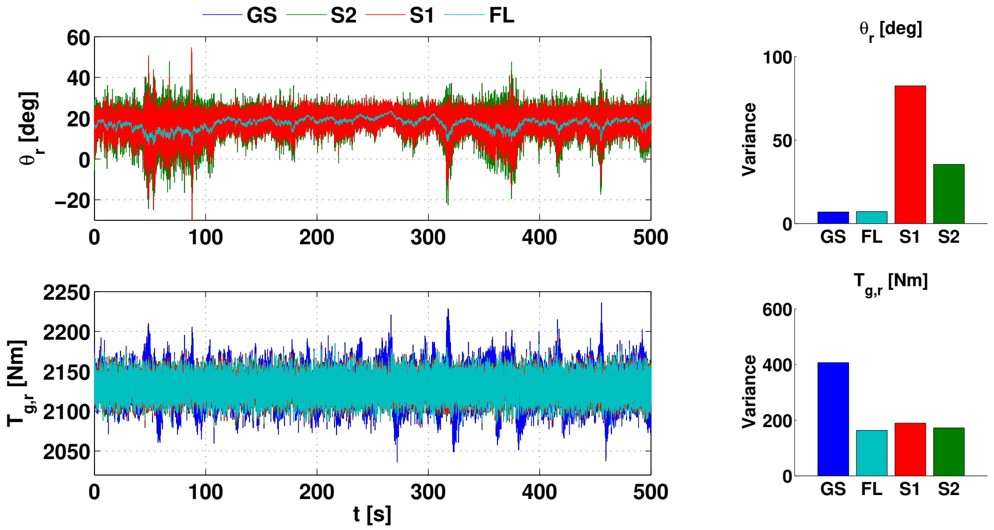


(a) $\dot{\theta}$

$\dot{\theta}$	min / max	OK%
GS	-8.11 / 9.49	100
FL	-17.13 / 12.93	99.8
S1	- 307/ 313	32.5
S2	-257 / 263	27.6

(b) Key figures

Figure 12.9: Left: Pitch speed associated with the state evolution in figure 12.8. Right: Extremum values for the pitch speed and the relative period that the signal stays within the limits.



(a) u (b) Variance

Figure 12.10: Left: Evolution of control signals associated with the state evolution in figure 12.3. Right: Variance of control signals.

12.3 Conclusion

This section briefly summarizes the conclusions which were drawn in the chapter. The previous sections presented simulations with constant and drifting mean wind speed. The overall picture in both situations was seen to be the same. The following summary, therefore applies for both simulation types.

The controllers reduced the fluctuations in the output power P_e almost equally well, whereas the fluctuations of the mechanical states (ω_r, ω_g, δ) were reduced more, with the nonlinear designs, than with the gain scheduled design. However, the sliding control laws did this at the expense, of a highly aggressive pitch action. Consequently, the pitch speed was seen to exceed the physical limits. The feedback linearizing controller, is therefore the only pure nonlinear design, which can be considered from a practical point of view.

Part IV

Conclusion and perspectives

Chapter 13

Conclusion

This chapter summarizes what have been accomplished in the project and the results obtained. The chapter is divided into three sections associated with the parts of the report.

Modeling and analysis

Two nonlinear wind turbine models have been derived: A variable speed turbine model used as benchmark model and a constant speed turbine model used for exemplification and preliminary investigations. Furthermore a stochastic wind model has been derived. The nonlinear dynamics of the variable speed wind turbine has been analyzed in an open loop setting. A toolbox for nonlinear control design has been developed in Matlab.

Control designs

The following four control methods have been implemented on the variable speed turbine: Gain scheduling, feedback linearization, sliding mode control and inverse optimal control. The gain scheduled controller was implemented as a basis of performance for the other controller designs. It consists of a discrete set of linear quadratic controllers, which are interpolated according to the current mean wind speed. The feedback linearizing controller consists of a nonlinear compensation, which renders the system linear. Furthermore, it decouples the stochastic fluctuations of the wind. The resulting linear system is controlled by a linear quadratic controller. Two gain scheduled controllers have been designed: The first is robust towards the wind fluctuations, whereas the other effectively decouples the wind fluctuations. The sliding surface is the same for both methods, and has been designed by minimizing a quadratic cost. The last control method that has been investigated is an inverse optimal control law. The method takes advantage of the feedback linearizable structure of the system, from which a stabilizing optimal control law has been derived.

Comparing the results obtained with perfect system knowledge, it was obvious that the feedback linearizing and the sliding mode controllers resulted in superior performance as compared to the gain scheduled linear controllers. The inverse optimal control law did not result in acceptable performance within the scope of this project.

Control with incomplete state knowledge

To make the control setup more realistic, incomplete state knowledge has been assumed and measurement noise has been introduced. Consequently, an extended Kalman filter (EKF) has been designed for the system, to provide estimates of the states. This illustrated that the feedback linearizing controller and the sliding mode strategies, still gave superior performance. However, the sliding mode strategies resulted in an extremely aggressive control law, which is unrealistic from a practical point of view. Hence, without exceeding the physical limits of the system, only the feedback linearizing controller resulted in a control law, which could surpass the performance achieved with the gain scheduled controller.

Chapter 14

Perspectives

This chapter presents a few subjects, which could be interesting to look at in future studies. The ideas presented are mainly concerned with the control perspectives.

The sliding mode method is intriguing, since it has the ability to make the system robust towards wind fluctuations. It would therefore be interesting to analyze, whether it is possible to avoid the aggressive behavior, without compromising the analytical justification of the method.

The controllers described in the report have been formulated in continuous time. In real life, a wind turbine is controlled by a digital computer. Consequently, the implementation issues regarding the controller design are of great interest. It is not obvious how to solve the discretization problem due to the nonlinearities in the controllers. Generally, the controllers are very complex and it is questionable whether it is possible to implement them on a real wind turbine. Therefore, it is of interest to analyze additional nonlinear control methods.

The controllers in this project have been designed for the top wind speed region, where the control objective is the classical stabilization problem. Hence, to obtain a fully operational nonlinear turbine controller the other wind regions should be considered.

Part V

Appendices

Appendix A

Model data

Wind turbine data

Nominal output power	$P_{e,0}$	$=$	225	[kW]
Nominal rotor speed	$\omega_{g,0}$	$=$	4.3	[rad/s]
Rotor inertia	J_r	$=$	90000	[kg·m^2]
Generator inertia	J_g	$=$	10	[kg·m^2]
Spring coefficient of drive train	K_s	$=$	$8 \cdot 10^6$	[N/m]
Damping coefficent of drive train	D_s	$=$	$8 \cdot 10^4$	[s^{-1}]
Damping coefficient of generator	D_g	$=$	2365	[s^{-1}]
Gear ratio	N_g	$=$	24.6	[-]
Generator pole pairs	p	$=$	3	[-]
Blade length	R	$=$	14.5	[m]
Pitch actuator time constant	τ_θ	$=$	0.15	[s]
Generator time constant	τ_T	$=$	0.1	[s]
Minimum pitching angle	θ_{min}	$=$	-1	[deg]
Maximum pitching angle	θ_{max}	$=$	25	[deg]
Maximum pitching rate	$\dot{\theta}_{max}$	$=$	10	[deg/s]

Wind model data

Nacelle height	z	$=$	45	[m]
Altitude of lowest inversion	z_i	$=$	1000	[m]
Turbulent intensity	τ_t	$=$	0.12	[%]

Additional data

Air density	ρ	$=$	1.2	[kg·m^3]
Grid frequency	ω_0	$=$	$2\pi 50$	[rad/s]

Appendix B

Nonlinear systems

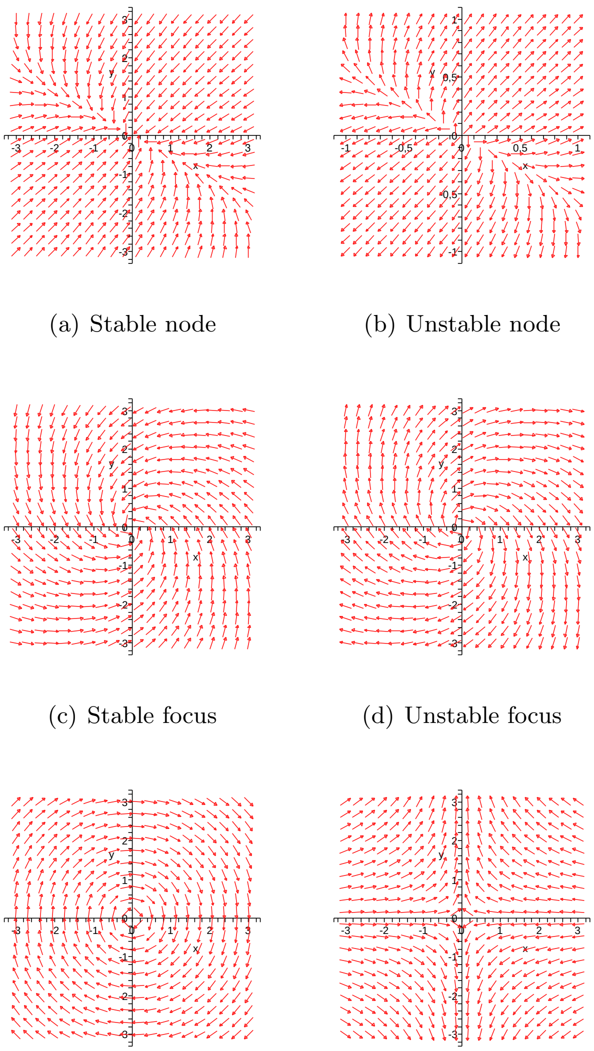
In this chapter a brief introduction to nonlinear systems will be given. The main purpose is to describe the differences between the characteristics of linear and nonlinear systems. The description will mainly be restricted to second order systems, this allows for visualization of the system characteristics through phase plane analysis. Another purpose of this chapter is to introduce the notion of Lyapunov stability and Lyapunov function which will be used extensively throughout the report.

B.1 Linear versus nonlinear systems

A linear system has one equilibrium point. For a second order system the equilibrium point can take one of the following 6 shapes:

- Stable node (poles in LHP)
- Unstable node (poles in RHP)
- Saddle point (one pole in LHP and one pole in RHP)
- Stable focus (Complex conj. poles in LHP)
- Unstable focus (Complex conj. poles in RHP)

The equilibrium points are illustrated in figure B.1 Except for the saddle point these



(a) Stable node (b) Unstable node

(c) Stable focus (d) Unstable focus

(e) Center (f) Saddle point

Figure B.1: Types of equilibria for second order linear systems

equilibria are either attractive or not. The saddle point is attractive when having the correct initial conditions, however in practice it is considered unstable because the slightest disturbances will make the trajectory go into the unstable region.

A nonlinear system on the other hand can have several equilibrium points which means that the system can be stable when operated in some parts of state space and unstable in other parts. A simple illustration of a system with several equilibrium points is shown in the following example:

Example B.1 (Nonlinear pendulum) *A simple system which has multiple equi-*

libria is the pendulum. A simple model for a pendulum is the following nonlinear 2. order differential equation:

$$\ddot{\theta} + \dot{\theta} + \sin\theta = 0 \tag{B.1}$$

Where θ is the angle relative to the downwards vertical position of the pendulum. The phase portrait of B.1 is seen in Figure B.2. It is seen that the pendulum has two

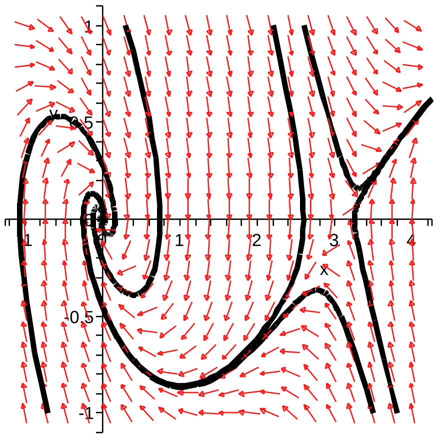


Figure B.2: Phase portrait for pendulum system

equilibrium points: One in the vertical downwards position and one in the vertical upwards position. When looking locally at the individual equilibria it is seen that the equilibrium in $(0, 0)$ resembles that of a stable focus and the equilibria in $(+-\pi, 0)$ resembles the saddle point.

The fact that the behavior of a nonlinear system can be described locally by the characteristics of a linear system provides fundamental justification for using linear methods on nonlinear plants. The method of linearizing a nonlinear plant around a chosen operating point is often denoted Lyapunov's linearization method or Lyapunov's first method.

When the range of operation is large the linearization method quickly becomes inadequate for describing a nonlinear system (Depending on the nonlinearities the 'large' is a relative measure). Instead one should seek to analyze the system using other methods. For systems of 1. and 2. order phase portraits as the ones above can be constructed to examine stability. Another approach which applies to systems of arbitrary order is Lyapunov's direct method which will be explained in the next section.

B.2 Lyapunov theory

In this section Lyapunov's direct method will be described which uses so called Lyapunov functions as a method for assessing stability of a system; linear as well as nonlinear. Actually Lyapunov theory covers both Lyapunov's linearization method (described shortly in the previous section) and the direct method. However in general Lyapunov theory is associated with the direct method and this presentation is no exception. Only what is needed to understand the rest of report is included in this section, hence only Lyapunov theory for time invariant systems will be presented.

B.2.1 Stability concepts

Before presenting Lyapunov direct method the stability concepts used in the presentation needs to be introduced.

Definition B.1 (Stability) An equilibrium state $x = 0$ is stable in the sense of Lyapunov if, for any $R > 0$, there exists $r > 0$, such that if $||x(0)|| < r$ then $||x(t)|| < R$ for all $t \geq 0$. Otherwise it is unstable.

Definition B.2 (Asymptotic stability) $x = 0$ is asymptotically stable if in addition there exists some $r > 0$ such that $||x(0) < r||$ implies that $x(t) \to 0$.

Definition B.3 (Exponential stability) $x = 0$ is furthermore exponentially stable if there exist two strictly positive numbers α and γ such that $||x(t)|| \leq \alpha||x(0)||e^{-\lambda t}$.

Definition B.4 (Global stability) If the asymptotic or exponential stability holds for any initial states, the equilibrium is said to be globally asymptotically or exponentially stable.

B.2.2 Lyapunov's direct method

Lyapunov's direct method provides a number of theorems for establishing local or global stability of systems. To proceed with these the concept of a Lyapunov function need to be presented.

Definition B.5 (Lyapunov function) If, in a ball B_{R_0}, the function $V(x)$ is positive definite and has continuous partial derivatives, and if its time derivative along any state trajectory of the system $\dot{x} = f(x)$ is negative semi-definite. Then $V(x)$ is said to be a Lyapunov function for the system.

Theorem B.6 (Local stability) *An equilibrium state exhibits local stability if*

- *$V(x)$ is positive definite locally in B_{R_0}*
- *$\dot{V}(x)$ is negative semidefinite locally in B_{R_0}*

The equilibrium is asymptotically stable if $\dot{V}(x)$ is negative definite.

Theorem B.7 (Global stability) *For the equilibrium state to hold global (asymptotic) stability the ball B_{R_0} must include the whole state-space. Furthermore $V(x)$ must be radially unbounded.*

- *$V(x)$ is positive definite.*
- *$\dot{V}(x)$ is negative definite.*
- *$V(x) \to \infty$ as $||x|| \to \infty$*

The above theorems provides a conceptually simple method for analyzing a system for stability. The drawback is however that there generally is no systematic method for finding such a Lyapunov function. Furthermore the theorems does not provide any information about the instability of an equilibrium point. [1]

Linear systems are an exception. A Lyapunov function can be created for stable linear systems in a systematic way as seen in the next example.

Example B.2 (Lyapunov function for linear systems) *A closed loop linear system is considered*

$$\dot{x} = Ax \tag{B.2}$$

A simple quadratic form for the Lyapunov function is chosen:

$$V(x) = x^T P x$$

The directional derivative of $V(x)$ is

$$\begin{aligned} \dot{V}(x) &= x^T P \dot{x} + \dot{x}^T P x \\ &= x^T \left(A^T P + P A \right) x \\ &= -x^T Q x \end{aligned}$$

Choosing P as an arbitrary positive definite matrix does not generally lead to conclusive results. On the other hand, it can be shown that choosing Q as a symmetric positive definite matrix (which means that $\dot{V}(x)$ is negative definite) and solving for P will lead to a Lyapunov function for the (stable) system B.2. The equation

$$A^T P + P A = -Q$$

is of course the well known Lyapunov equation.

[1] Theorems exists that provides such information see for example (Khalil 2002)

Appendix C

Contents of CD

The CD includes a report repository and a Matlab repository. A directory tree is seen below:

```
/
├─ report
└─ matlab
   ├─ fltoolbox
   ├─ gainscheduling
   ├─ feedlin
   ├─ slidingmode
   ├─ optimal
   └─ share
      └─ systems
```

The report folder contains the report in PDF format and in PS format. The Matlab folder is described in detail in the following section

C.1 Matlab repository

The Matlab folder includes:

- The sub-folder `fltoolbox` containing the functions and objects which constitutes the nonlinear toolbox.
- Scripts which designs the nonlinear controllers and Simulink simulation models which are placed in the sub-folders `gainscheduling`, `feedlin`, `slidingmode` and `optimal`.

- Additional functions and scripts which are used in the design of the nonlinear controllers placed in the sub-folder `share`. The system descriptions of the constant speed turbine, the variable speed turbine and the wind augmented variable speed turbine are also contained in this folder.

It is stressed that the simulation/design scripts cannot be executed directly from the CD, since they generate simulation files (a CD is read-only). Therefore, they should be moved to a write-enabled environment before executed. The simulation files are divided into two categories:

- Initialization files (initmdl*.m): Derives the control laws. Furthermore, generates the m-files, containing system and controller descriptions, which are used in the simulations.

- Model files (wtmodel*.mdl): Simulink models which interconnects the systems and controllers.

The names of the initialization/model have numbers which indicate whether they are used in connection to the constant speed turbine or the variable speed turbine: '0' is associated with the constant speed turbine and '1' is associated with the variable speed turbine.
The control design folders additional contain `starup` files which need to executed before running the initialization scripts.

References

DeCarlo, R., S. Zak, and G. Matthews (1988). Variable structure control of nonlinear multivariable systems: a tutorial. *Proceedings of the IEEE 76*(3), 212 –232.

DNV/Risø(2002). *Guideline for design of wind turbines* (second ed.). DNV/Risø.

Freeman, R. and P. Kokotovic (1995). Optimal nonlinear controllers for feedback linearizable systems. *American Control Conference, Proceedings of the 1995 4*, 2722–2726 vol.4.

Gelb, A., J. F. Kasper, R. A. Nash, C. F. Price, and A. A. Sutherland (1974). *Applied Optimal Estimation*. The M.I.T. Press.

Hansen, M., A. Hansen, T. Larsen, Øye S., P. Sørensen, and P. Fuglsang (2005). Control design for a pitch-regulated, variable speed wind turbine. Technical report, Risø.

Hendricks, E., O. Jannerup, and P. H. Sørensen (2004). *Linear Systems Control: Deterministic and stochastic methods*. Ørsted DTU.

Horsbøl, O. (2003). Bumpless transfer og regulering af vindmøller. Master's thesis, Technical University of Denmark.

Højstrup, J. (1982). Velocity spectra in the unstable planetary layer. *Journal of the Atmospheric Sciences 39*(10), 2239–2248.

Isidori, A. (1995). *Nonlinear Control Systems* (Third ed.). Springer.

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

Khalil, H. K. (2002). *Nonlinear Systems* (Third ed.). Prentice Hall.

Klingenberg, B. (1991). Adaptive pitch control for wind turbines. Master's thesis, IMSOR, Technical University of Denmark.

Knudsen, T. (1983). Regulering af vindmøller. Master's thesis, Danmarks Tekniske Højskole.

Krstic, M., I. Kanellakopoulos, and P. Kokotovic (1995). *Nonlinear and Adaptive Control*. John Wiley & Sons.

Larsen, A. J. and T. S. Mogensen (2006). Individuel pitchregulering af vindmølle. Master's thesis, Technical University of Denmark.

Madsen, H. (2001). *Time Series Analysis*. Technical University of Denmark.

Poulsen, N. K. (1997). *Stokastisk Adaptiv Regulering*. IMM DTU. v. 1.01.

Poulsen, N. K. (2004). *Dynamic Optimization*. IMM DTU.

Rugh, W. J. and J. S. Shamma (2000). Research on gain scheduling. *Automatica 36*(10), 1401–1425.

Sepulchre, R., M. Jankovic, and P. Kokotovic (1997). *Constructive Nonlinear Control*. Springer.

Slootweg, J., H. Polinder, and W. Kling (2001). Dynamic modelling of a wind turbine with doubly fed induction generator. *Power Engineering Society Summer Meeting, 2001 1*, 644–649.

Slotine, J.-J. E. and W. Li (1991). *Applied Nonlinear Control*. Prentice Hall.

Xin, M. (1993). Modelling and control of a wind turbine. Master's thesis, DTU.

Xin, M. (1997). *Adaptive Extremum Control and Wind Turbine Control*. Ph. D. thesis, Informatics and Mathematical Modelling, Technical University of Denmark, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby. Supervisor: Niels K. Poulsen.