Model Predictive Control of a Wind Turbine

Lars Christian Henriksen

Kongens Lyngby 2007
IMM-M.Sc.-2007-41

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

Summary

The increase in size, prize and power production of modern wind turbines continue to improve the overall economy of their installation and maintenance. A suitable place to install these mega wind turbines is on the sea as their is a more stable wind. These water based wind farms are confined to reefs near land where the construction of the foundations doesn't become to expensive and problematic. It has been suggested to build floating wind turbines instead and thus enabling previous unsuited locations to become potential wind farms. This thesis investigates control of both wind turbines mounted on solid foundations and their floating counter parts.

The wind turbine operates over a wide wind speed range and the control objectives changes over that range. It has been investigated how to identify and switch between these different modes of operation.

The turbulent nature of the wind causes the control of the gigantic structures to react within fractions of a second. Such rapid response should cross certain limits otherwise the fatigue of the actuator systems is greatly accelerated leading to uneconomic operation of the wind turbine. Model predictive control have been investigated as a method to keep within these constraints.

Preface

This thesis was prepared at Informatics Mathematical Modelling, the Technical University of Denmark in partial fulfillment of the requirements for acquiring the M.Sc. degree in engineering.

The thesis deals with different aspects of mathematical modeling of wind turbines and control theory methods suited for the control of these. In particular model predictive control has been investigated and its ability to handle constraints of process variables.

I would like to thank my supervisor Assoc. Prof. Niels Kjølstad Poulsen, IMM, DTU, for the inspiration and guidance throughout the project especially concerning the control theory considerations of the thesis. I would also like to my co-supervisor Senior Scientist Morten Hartvig Hansen, Risø, DTU, for his guidance and support especially concerning the more practical parts of implementing the control methods developed on the HAWC2 software and last but least for letting me continue my work as a Ph.D student at Risø, DTU.

Lyngby, April 2007

Lars Christian Henriksen

Nomenclature

v	$[m/s]$	-	Wind speed
v_m	$[m/s]$	-	Mean wind speed
v_t	$[m/s]$	-	Turbulent wind speed
v_r	$[m/s]$	-	Relative wind speed
P_w	$[W]$	-	Wind power in the absence of a rotor disc
P_r	$[W]$	-	Power absorbed from wind to rotor (driveshaft)
P_m	$[W]$	-	Mechanical power from driveshaft to generator
P_e	$[W]$	-	Electrical power of generator from mechanical power
η	$[-]$	-	Generator efficiency
$\dot{m}$	$[kg/s]$	-	Mass flow of air
ρ	$[kg/m^3]$	-	Mass density of air
N_g	$[-]$	-	Gear ratio
R	$[m]$	-	Rotor blade length and rotor disc radius
C_P	$[-]$	-	Quasi-stationary aerodynamic power coefficient
C_T	$[-]$	-	Quasi-stationary aerodynamic thrust coefficient
θ	$[^\circ]$	-	Collective pitch of rotor blades
λ	$[-]$	-	Tip-speed-ratio

Q_r	$[N\ m]$	-	Aerodynamic torque from wind to rotor (driveshaft)
Q_g	$[N\ m]$	-	Mechanical torque from generator to driveshaft
x_t	$[m]$	-	Displacement of nacelle
ϕ_r	$[rad]$	-	Azimuth angle of rotor
ϕ_g	$[rad]$	-	Azimuth angle of generator
ϕ_Δ	$[rad]$	-	Azimuth angular torsion of driveshaft
Ω_r	$[rad/s]$	-	Angular velocity of rotor
Ω_g	$[rad/s]$	-	Angular velocity of generator
I_r	$[kg\ m^2]$	-	Moment of inertia of rotor
I_g	$[kg\ m^2]$	-	Moment of inertia of generator
K_s	$[N/rad]$	-	Driveshaft spring constant
D_s	$[N/rad\ s]$	-	Driveshaft dampening constant
M_t	$[kg]$	-	Mass of part of tower and nacelle
K_t	$[N/m]$	-	Tower spring constant
D_t	$[N/m\ s]$	-	Tower dampening constant
Q_t	$[N]$	-	Thrust force on tower
ω_n	$[rad/s]$	-	Natural frequency (of pitch actuator)
ζ	$[-]$	-	Damping (of pitch actuator)
τ	$[s]$	-	Time constant (of generator torque actuator)

x	$\in \mathbb{R}^{n_x}$	-	State vector
u	$\in \mathbb{R}^{n_u}$	-	Control vector
v	$\in \mathbb{R}^{n_u}$	-	Control deviation vector
z	$\in \mathbb{R}^{n_z}$	-	Optimization vector
y	$\in \mathbb{R}^{n_y}$	-	Measured output vector
$\mathrm{y_r}$	$\in \mathbb{R}^{n_r}$	-	Reference controlled output vector
r	$\in \mathbb{R}^{n_r}$	-	Reference vector
d	$\in \mathbb{R}^{n_d}$	-	State disturbance vector
p	$\in \mathbb{R}^{n_p}$	-	Output disturbance vector
c	$\in \mathbb{R}^{n_c}$	-	Constraints vector
q	$\in \mathbb{R}^{n_z}$	-	Optimization weight vector

$\mathbf{A}$	$\in \mathbb{R}^{n_x \times n_x}$	-	State transition matrix
$\mathbf{B}$	$\in \mathbb{R}^{n_x \times n_u}$	-	Input matrix
$\mathbf{C}$	$\in \mathbb{R}^{n_y \times n_x}$	-	State output matrix
$\mathbf{D}$	$\in \mathbb{R}^{n_y \times n_u}$	-	Direct input output matrix
$\mathbf{E}$	$\in \mathbb{R}^{n_z \times n_x}$	-	Optimization state matrix
$\mathbf{F}$	$\in \mathbb{R}^{n_z \times n_u}$	-	Optimization input matrix
$\mathbf{H}$	$\in \mathbb{R}^{n_r \times n_y}$	-	Reference output matrix
$\mathbf{B_d}$	$\in \mathbb{R}^{n_x \times n_d}$	-	State disturbance matrix
$\mathbf{C_p}$	$\in \mathbb{R}^{n_y \times n_p}$	-	Output disturbance matrix
$\mathbf{\Phi}$	$\in \mathbb{R}^{n_x \times n_x}$	-	Closed loop state transition matrix
$\mathbf{\Psi}$	$\in \mathbb{R}^{n_z \times n_x}$	-	Closed loop optimization transition matrix
$\mathbf{M}$	$\in \mathbb{R}^{n_c \times n_z}$	-	Constraints matrix
$\mathbf{W}$	$\in \mathbb{R}^{n_z \times n_z}$	-	Optimization weight matrix

Contents

Chapter 1

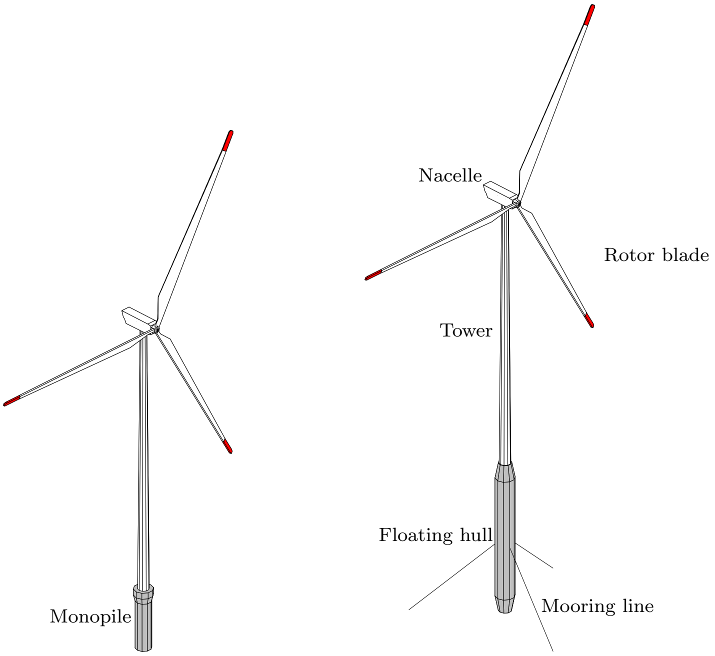
Introduction

The purpose of modern wind energy conversion systems (WECS) is to extract the aerodynamic power from the wind and convert it to electric power. Today the most wide spread version of WECS is the horizontal axis wind turbine (HAWT) with a 3 blade upwind rotor. Before the introduction of variable speed generators, the rotor speed on the HAWT was kept constant. This constraint limited the efficiency of the wind power capture. New wind turbines are able to operate more efficient over a wider range of wind speeds, which has lead to more sophisticated control strategies with the added degrees of freedom.

Modern wind turbines are controlled by the pitch of the rotor blades, the electromagnetic torque of the generator and by the yaw of the nacelle. Although their orientation toward the wind is controlled by a yaw controller this degree of freedom will be omitted in this project. The reason for this simplification will be discussed in section 2.1 on page 8.

Traditionally wind turbines are placed on land or on solid foundations if placed in the water. This limits their deployment to locations of relatively shallow water because the construction costs of an underwater monopile are to expensive or technically impossible. Recently it has been suggested to place floating HAWTs in deep water and anchor them with mooring cables to bottom of the sea.

The concept of a floating HAWT poses new challenges as the vertical stability



(a) Mounted tower

(b) Floating tower

Figure 1.1: Horizontal axis wind turbines

of the HAWT is heavily reduced by the lack of solid foundation. The challenges can be somewhat accommodated mechanically by adding supporting and stabilizing structures with additional construction costs as a drawback. But the changed dynamics of the HAWT can't be completely compensated. Modern control techniques offers the handling of the demanding dynamics within a more systematic framework thus giving better performance and enhances the ability of prioritizing operation parameters from an economic point of view.

The displacement of the nacelle is only modeled in the direction of the wind. Any oscillatory behavior in the other directions and the fact that the motion is not linear is disregarded in this project. Intuition suggests that such a crude assumption significantly diverts from the behavior of a real floating wind turbine, nevertheless to keep focus on control methods this simplification has been decided. The sanity of the simplification will be validated by simulations in the more elaborate model in HAWC2, which is complex wind turbine simulation environment developed Risø.

In reality the drive shaft and rotor are inclined 5° from horizontal, this is also omitted from the model.

Different wind speed means different control objectives, in this project control objectives for the entire operational wind speed range have been developed. Advanced control theory known as model predictive control (MPC) have been implemented to control the wind turbines in all the operating regions. This is contrary to typical projects that solely focus on the top region of the wind spectrum to be controlled by advanced methods and leaves the lower wind speed regions to be controlled by PI(D) controllers or lookup tables.

The report is divided into different parts

- **I - Modeling and analysis** presents linear models of the wind turbine and control strategies for the different operating modes.
- **II - Theory of methods** presents the control theory part of the project.
- **III - Implementation and results** presents the results obtained by implementing the presented controllers and testing them in Matlab and HAWC2.
- **IV - Conclusion and perspectives** discusses the results obtained in the project and which paths could be taken in the future to extend the work of the project.

It assumed that the reader of this report has a solid foundation within linear control theory. More advanced topics such as invariant set theory and static and dynamic optimization are introduced and explained in the report. The fundamentals of wind turbine dynamics is explained in the report and a basic knowledge of physics and mechanical systems is required.

Part I

Modeling and analysis

Chapter 2

Modeling

This chapter will introduce the different part of a wind turbine and present linear model models for the subsystems.

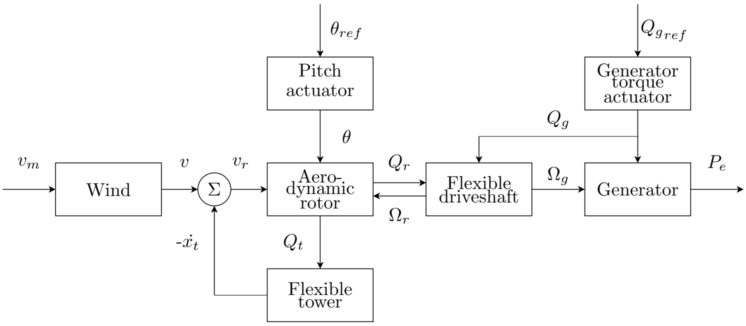


Figure 2.1: Nonlinear dynamic model of a wind turbine and the wind

2.1 The wind

The variations of the wind speed can be divided into different classifications based either on time of geography. Geographic classifications can fx. be:

- **Water** which could be oceans, gulfs etc.
- **Coastal land** which could be the west coast of Denmark, etc.
- **Continental land** which is deep inland and far from water.

The wind speeds are typically higher near water than inland. Due to complex weather systems the wind is also more likely to come from one direction rather than another (fx. north west). This takes us to another classification where long time measurements are used to make statistical charts known as wind roses that show in which directions the usually blows. A change of wind direction usually takes hours or quarters of hours. Variations with time constants on that magnitude are not within the scope of this project and should be modeled as parameter changes or input steps or ramps. Hence the yaw rotation of the wind turbine is omitted from the project and is assumed to be handled by another controller.

As just mentioned time variation is also measure that can be used classify the wind:

- **Annual and seasonal variations** such as El Niño and autumn storms etc.
- **Synoptic and diurnal variations** is the passing of large weather system and the difference between night and day.
- **Turbulence** is caused by friction with surface and temperature differences.

The annual, seasonal, synoptic and diurnal variations are considered to be a constant or slowly varying mean wind speed v_m which is modeled as a constant, a step or a ramp and without any dynamics in this project. The only dynamic components of the wind which should be modeled is thus the turbulent wind v_t. In a cross section perpendicular the wind direction the turbulent wind can be divided into a finite or ideally infinite number of point wind velocities. These are both correlated in time and space. In this project only the time component is considered and the wind field is assumed equally distributed for design modeling purposes.

The wind speed variation can be modeled as a complicated nonlinear stochastic process but for practical purposes it is an approximation based on a more complex model described in (Østergaard, 1994) and (Larsen and Mogensen, 2006)

$$v = v_m + v_t \tag{2.1}$$

where

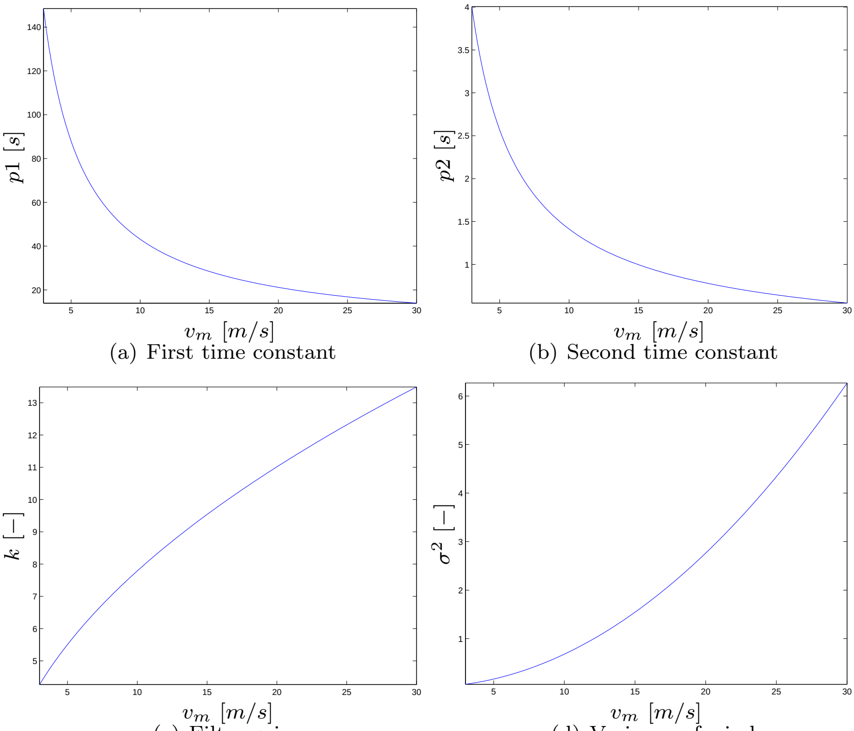
$$v_t = \frac{k(v_m)}{(p_1(v_m)s+1)(p_2(v_m)s+1)}e; \quad e \in N(0,1) \tag{2.2}$$

the turbulent wind model can be formulated in a state space description

$$\begin{pmatrix} \dot{v}_t \\ \ddot{v}_t \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{p_1(v_m)p_2(v_m)} & -\frac{p_1(v_m)+p_2(v_m)}{p_1(v_m)p_2(v_m)} \end{bmatrix} \begin{pmatrix} v_t \\ \dot{v}_t \end{pmatrix} + \begin{bmatrix} 0 \\ \frac{k(v_m)}{p_1(v_m)p_2(v_m)} \end{bmatrix} e \tag{2.3}$$

The coefficients of the filter can be seen in Fig. (2.2)

Another factor omitted from the design modeling is wind shear. Wind shear is the effect that rough terrain has on the turbulence on the wind. The rougher the terrain, the higher the friction between the surface and the wind. This leads to the wind moving slower near ground than farther from ground. This again means that as the rotor passes from top to bottom in its rotation it is subject to different wind speeds giving and effect that is time correlated with the rotation speed of the rotor.



(a) First time constant

(b) Second time constant

(c) Filter gain

(d) Variance of wind

Figure 2.2: Properties of stochastic wind given by existing material

2.2 Wind turbine subsystems

2.2.1 Aerodynamics

From (Burton et al., 2001) and (Hansen, 2000) the following aerodynamic equations of a wind turbine are given. The available power of the wind in a circular cross section with the same area as the rotor disc, but with the absence of the rotor disc is given by

$$P_w = \frac{1}{2}\dot{m}v^2 = \frac{1}{2}\rho\pi R^2 v^3 \tag{2.4}$$

where $\dot{m}$ is the mass flow of the wind, v is the speed of the wind, ρ is the air density and R is the radius of the rotor disc.

Only a fraction of the available power P_w can be converted to rotor power P_r. The ratio is given by the power coefficient C_P

$$P_r = P_w C_P \tag{2.5}$$

C_P have a theoretical upper limit of $16/27 \approx 0.593$ known as the Betz limit. This is due to the fact the wind cannot be completely drained of energy, otherwise the wind speed at the rotor front would reduce to zero and the rotation of the rotor would stop. It can be noted that modern wind turbines have a maximum power coefficient of about 0.5, which is considered to be the optimum for standard design horizontal axis wind turbines. The aerodynamic torque exerted by the wind on the rotor is given by

$$Q_r = \frac{1}{\Omega_r} P_r \tag{2.6}$$

Besides the aerodynamic torque the wind turbine is also influenced by the thrust force Q_t exerted by the wind on the tower and rotor which is given by

$$Q_t = \frac{1}{2}\rho\pi R^2 v^2 C_T \tag{2.7}$$

where C_T is the thrust force coefficient.

The aerodynamic coefficients of power C_P and of thrust C_T are given by complicated measurements and calculations, which shall not be discussed here. In quasi-stationarity, i.e a steady-state mass flow, the coefficients are functions of the rotor blade pitch angle θ, the rotor rotation speed Ω_r and the wind speed. The concept of tip-speed-ratio λ is introduced for simpler a notation

$$\lambda \equiv \frac{v}{\Omega_r R} \tag{2.8}$$

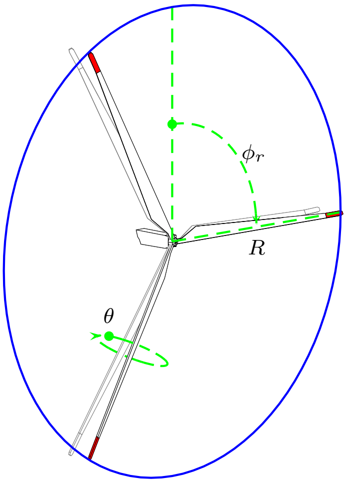


Figure 2.3: Rotor model. The blue circle represents the rotor disc abstraction. The gray blades are out of plane deflections, which are not modeled in this project

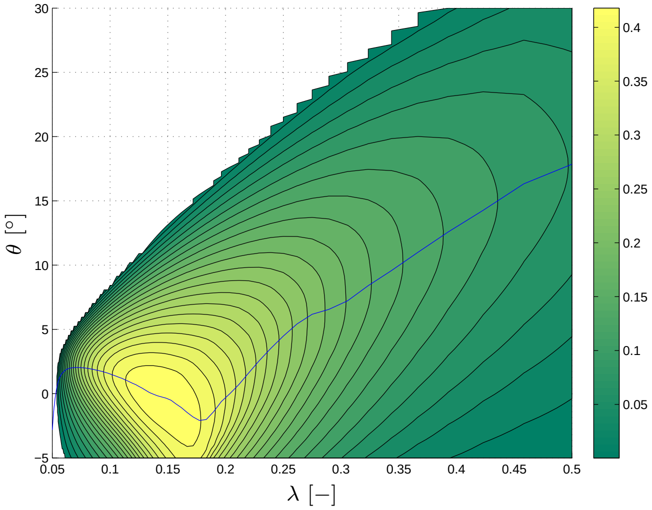


Figure 2.4: Top view of C_P curve. The blue line indicates the maximum of the curve. The region above the blue line is known as the pitch region and the region below the blue line is known as the stall region

Leading to the coefficients variable dependencies can be written as $C_P(\theta, \lambda)$ and $C_T(\theta, \lambda)$. It should be noted that in some parts of the literature the tip-speed-ratio is defined as the inverse of the definition given here.

2.2.1.1 Linear aerodynamic torque

The nonlinear rotor torque is given by eq. (2.6)

$$Q_r = \frac{P_r}{\Omega_r} = \frac{\frac{1}{2}\rho\pi R^2 v_r^3 C_P(\frac{v_r}{\Omega_r R}, \theta)}{\Omega_r} \tag{2.9}$$

The rotor torque has to be linearized, around a linearization point denoted with subscript $_0$, to implement linear control strategies on the HAWT. The notation is the same as in Jannerup and Sørensen (2000). Where the subscript $_0$ denotes the linearization points.

$$Q_{r0} = \frac{P_{r0}}{\Omega_{r0}} = \frac{\frac{1}{2}\rho\pi R^2 v_{r0}^3 C_P(\frac{v_{r0}}{\Omega_{r0} R}, \theta_0)}{\Omega_{r0}} \tag{2.10}$$

The linearization is done with a first order Taylor series expansion of Q_r with respect to its parameters v_r, Ω_r and θ. The Δ denotes the difference between the real variable and the linearization variable (e.g. $\Delta\Omega_r = \Omega_r - \Omega_{r0}$).

$$Q_r \cong Q_{r0} + \left.\frac{\partial Q_r}{\partial \Omega_r}\right|_{\Omega_{r0}} \cdot \Delta\Omega_r + \left.\frac{\partial Q_r}{\partial \theta}\right|_{\theta_0} \cdot \Delta\theta + \left.\frac{\partial Q_r}{\partial v_r}\right|_{v_{r0}} \cdot \Delta v_r \tag{2.11}$$

giving the direct term of a linearized state space model

$$\Delta Q_r \cong \begin{bmatrix} \left.\frac{\partial Q_r}{\partial \Omega_r}\right|_{\Omega_{r0}} & \left.\frac{\partial Q_r}{\partial \theta}\right|_{\theta_0} & \left.\frac{\partial Q_r}{\partial v_r}\right|_{v_{r0}} \end{bmatrix} \begin{pmatrix} \Delta\Omega_r \\ \Delta\theta \\ \Delta v \end{pmatrix} \tag{2.12}$$

and the individual partial derivatives of eq. (2.11) are

$$\left.\frac{\partial Q_r}{\partial \Omega_r}\right|_{\Omega_{r0}} = \frac{1}{\Omega_{r0}} \left.\frac{\partial P_r}{\partial \Omega_r}\right|_{\Omega_{r0}} - \frac{P_{r0}}{{\Omega_{r0}}^2} \tag{2.13}$$

$$\left.\frac{\partial Q_r}{\partial \theta}\right|_{\theta_0} = \frac{1}{\Omega_{r0}} \left.\frac{\partial P_r}{\partial \theta}\right|_{\theta_0} \tag{2.14}$$

$$\left.\frac{\partial Q_r}{\partial v_r}\right|_{v_{r0}} = \frac{1}{\Omega_{r0}} \left.\frac{\partial P_r}{\partial v_r}\right|_{v_{r0}} \tag{2.15}$$

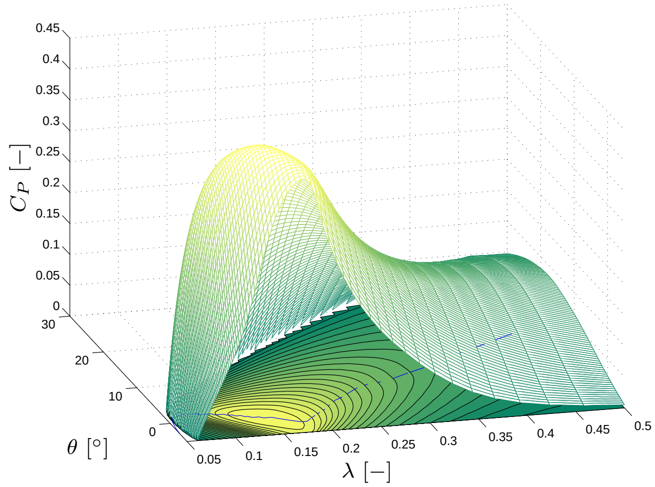


Figure 2.5: Quasi-stationary power coefficient C_P

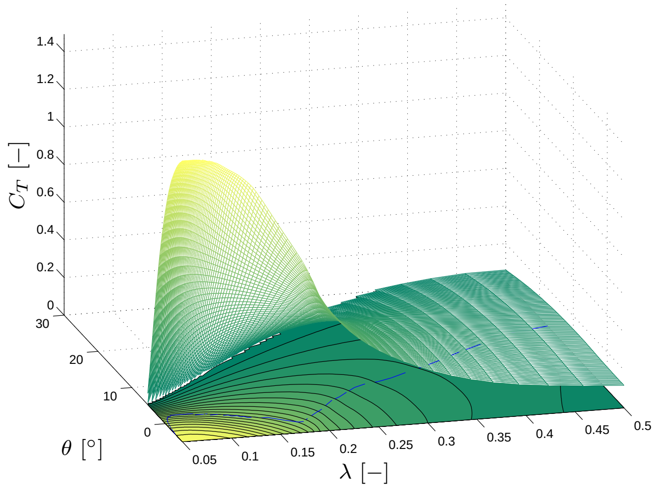


Figure 2.6: Quasi-stationary thrust coefficient C_T. Optimum of C_P curve is depicted as a blue line on the contour plot of C_T at floor of the figure

the partial derivatives of P_r (given in eq. (2.5)) are

$$\left.\frac{\partial P_r}{\partial \Omega_r}\right|_{\Omega_{r0}} = \frac{1}{2}\rho\pi R^2 {v_{r0}}^3 \left.\frac{\partial C_P}{\partial \lambda}\right|_{\lambda_0} \cdot \left.\frac{\partial \lambda}{\partial \Omega_r}\right|_{\Omega_{r0}} \tag{2.16}$$

$$\left.\frac{\partial P_r}{\partial \theta}\right|_{\theta_0} = \frac{1}{2}\rho\pi R^2 {v_{r0}}^3 \left.\frac{\partial C_P}{\partial \theta}\right|_{\theta_0} \tag{2.17}$$

$$\left.\frac{\partial P_r}{\partial v_r}\right|_{v_{r0}} = \frac{1}{2}\rho\pi R^2 3{v_{r0}}^2 C_{P0} + {v_{r0}}^3 \left.\frac{\partial C_P}{\partial \lambda}\right|_{\lambda_0} \cdot \left.\frac{\partial \lambda}{\partial v_r}\right|_{v_{r0}} \tag{2.18}$$

The partial derivatives of λ (given in eq. (2.8)) are

$$\left.\frac{\partial \lambda}{\partial \Omega_r}\right|_{\Omega_{r0}} = -\frac{v_r}{{\Omega_{r0}}^2 R} \tag{2.19}$$

$$\left.\frac{\partial \lambda}{\partial v_r}\right|_{v_{r0}} = \frac{1}{\Omega_{r0} R} \tag{2.20}$$

The partial derivatives (2.21) and (2.22) on the C_P-curve Fig. (2.5) can be found using any number of different numerical interpolation methods.

$$\left.\frac{\partial C_P}{\partial \lambda}\right|_{\lambda_0} \tag{2.21}$$

$$\left.\frac{\partial C_P}{\partial \theta}\right|_{\theta_0} \tag{2.22}$$

2.2.1.2 Linear aerodynamic thrust

The force exerted by the wind on the tower has to be linearized, around a linearization point denoted with subscript $_0$, to implement linear control strategies on the HAWT. This is done with a first order Taylor series expansion of Q_t with respect to its parameters v_r, Ω_r and θ

$$Q_t \cong Q_{t0} + \left.\frac{\partial Q_t}{\partial \Omega_r}\right|_{\Omega_{r0}} \cdot \Delta\Omega_r + \left.\frac{\partial Q_t}{\partial \theta}\right|_{\theta_0} \cdot \Delta\theta + \left.\frac{\partial Q_t}{\partial v_r}\right|_{v_{r0}} \cdot \Delta v_r \tag{2.23}$$

the linerization point Q_{t0} is given by $Q_t(\Omega_{r0}, \theta_0, v_{r0})$ and the partial derivatives of Q_t (given in eq. (2.7)) are

$$\left.\frac{\partial Q_t}{\partial \Omega_r}\right|_{\Omega_{r0}} = \frac{1}{2}\rho\pi R^2 {v_{r0}}^2 \left.\frac{\partial c_t}{\partial \lambda}\right|_{\lambda_0} \cdot \left.\frac{\partial \lambda}{\partial \Omega_r}\right|_{\Omega_{r0}} \tag{2.24}$$

$$\left.\frac{\partial Q_t}{\partial \theta}\right|_{\theta_0} = \frac{1}{2}\rho\pi R^2 {v_{r0}}^2 \left.\frac{\partial c_t}{\partial \theta}\right|_{\theta_0} \tag{2.25}$$

$$\left.\frac{\partial Q_t}{\partial v_r}\right|_{v_{r0}} = \frac{1}{2}\rho\pi R^2 2 v_{r0} c_{t0} + {v_{r0}}^2 \left.\frac{\partial c_t}{\partial \lambda}\right|_{\lambda_0} \cdot \left.\frac{\partial \lambda}{\partial v_r}\right|_{v_{r0}} \tag{2.26}$$

The partial derivatives of λ (given in eq. (2.8)) are already given in eq. (2.19) and eq. (2.19).

The partial derivatives (2.27) and (2.28) on the C_T-curve Fig. (2.6) can be found using any number of different numerical interpolation methods.

$$\left.\frac{\partial C_T}{\partial \lambda}\right|_{\lambda_0} \tag{2.27}$$

$$\left.\frac{\partial C_T}{\partial \theta}\right|_{\theta_0} \tag{2.28}$$

2.2.1.3 Omitted aerodynamic phenomena

As mentioned earlier the aerodynamic coefficients are only valid under the assumption of a steady-state mass flow of the air. In reality the mass flow does not settle to a new equilibrium infinitely fast during a transition and contributions from the dynamics of the fluid (air) should be added to the coefficients. These contributions are significant and lead to an aerodynamic dampening of the interaction between the wind and the rotor. If the blades are pitching fast or even oscillating the actual coefficients might differ significantly from the quasi-stationary coefficients. Hence, care should be taken not to induce such a situation during control of the wind turbine.

The rotor blades are bended backwards in steady state operation. This means the blades are not rotational symmetric with regards to their masses and when pitched this gives rise to oscillations in both blades and tower. This oscillating behavior also disrupts the quasi-stationary assumptions of the power and thrust coefficients.

Another significant phenomenon is the aerodynamic shadow of the tower. It is especially apparent on new wind turbines with tubular steel towers as opposed to older type lattice towers. As the blades pass by the tower, they enter an area of lower wind speed which gives rise to periodic disturbance correlated in time with the rotor speed.

And finally as mentioned the wind section, the wind shear is also omitted from the model.

2.2.2 Electrical generator

The mechanical power captured by the rotor is transfered via the drive train shaft to electrical generator. The generator impose a electrical counter torque on the drive shaft and thereby extract electrical power.

$$P_r = P_m = \Omega_g Q_g \tag{2.29}$$

However, due to less than perfect efficiency, the generator is only able to convert some of the mechanical power to elecrical power. This is a simplification and any losses in drivetrain bearings, gearbox etc are omitted and simply included in this measure of efficiency

$$P_e = \eta P_m \tag{2.30}$$

The generator is only able to operate within some limited bounds, generators with a wider operating range are available but are increasingly expensive.

$$\Omega_{g_{min}} \leq \Omega_g \leq \Omega_{g_{max}} \tag{2.31}$$

$$0 \leq P_e \leq P_{nom} \tag{2.32}$$

The electrical counter torque is also subjected to constraints. These constraints are discussed in subsection 2.2.6 on page 23.

2.2.2.1 Linear generator

$$P_e \cong P_{e0} + \left.\frac{\partial P_e}{\partial \Omega_g}\right|_{\Omega_{g_0}} \cdot \Delta\Omega_g + \left.\frac{\partial P_e}{\partial Q_g}\right|_{Q_{g_0}} \cdot \Delta Q_g \tag{2.33}$$

where

$$\left.\frac{\partial P_e}{\partial \Omega_g}\right|_{\Omega_{g0}} = Q_{g0} \tag{2.34}$$

$$\left.\frac{\partial P_e}{\partial Q_g}\right|_{Q_{g0}} = \Omega_{g0} \tag{2.35}$$

2.2.3 Flexible drivetrain shaft

The driveshaft transfers power from the rotor to the generator. In steady state operation the gear ratio between the rotor and the generator are given by a constant. The gear is assumed free of losses.

$$\Omega_g = \Omega_r N_g \tag{2.36}$$

The driveshaft on the rotor side is assumed flexible while the driveshaft on the generator side is assumed rigid. This leads to a dynamic angular displacement, between the angle of the rotor ϕ_r and the angle of the generator ϕ_g, in normal operation The angular velocities are derivatives of the angles and to simplify notation, the following definitions are introduced

$$\Omega_r \equiv \dot{\phi}_r \qquad \Omega_g \equiv \dot{\phi}_g \qquad \phi_\Delta \equiv \phi_r - \frac{\phi_g}{N_g} \qquad \dot{\phi}_\Delta \equiv \Omega_r - \frac{\Omega_g}{N_g}$$

The mechanical flexibility of the rotor side driveshaft is modeled as a rotational 2-mass, 1-spring, 1-damper system where the physical properties on the generator side of the gear side translated into physical properties on the rotor side of the gear.

The mechanical equations for the system are (Larsen and Mogensen, 2006)) , see Fig. (2.7)

$$Q_r = I_r\dot{\Omega}_r + \dot{\phi}_\Delta D_s + \phi_\Delta K_s \tag{2.37a}$$

$$-Q_g N_g = I_g N_g^2 \frac{\dot{\Omega}_g}{N_g} - \dot{\phi}_\Delta D_s - \phi_\Delta K_s \tag{2.37b}$$

where the angular displacement in stationary mode ($\ddot{\phi}_r = 0$, $\ddot{\phi}_g = 0$ and $\dot{\phi}_\Delta = 0$) is

$$\phi_{\Delta 0} = \frac{Q_r}{K_s} = \frac{Q_g N_g}{K_s} \tag{2.38}$$

Eq. (2.37) can be written in a state space formulation

$$\begin{pmatrix} \dot{\Omega}_r \\ \dot{\Omega}_g \\ \dot{\phi}_\Delta \end{pmatrix} = \begin{bmatrix} -\frac{D_s}{I_r} & \frac{D_s}{I_r N_g} & -\frac{K_s}{I_r} \\ \frac{D_s}{I_g N_g} & -\frac{D_s}{I_g N_g^2} & \frac{K_s}{I_g N_g} \\ 1 & -\frac{1}{N_g} & 0 \end{bmatrix} \begin{pmatrix} \Omega_r \\ \Omega_g \\ \phi_\Delta \end{pmatrix} + \begin{bmatrix} \frac{1}{I_r} & 0 \\ 0 & -\frac{1}{I_g} \\ 0 & 0 \end{bmatrix} \begin{pmatrix} Q_r \\ Q_g \end{pmatrix} \tag{2.39}$$

To determine the parameters of the driveshaft in Eq. (2.37) the generator side of the driveshaft is be fixed, i.e. $\phi_g = 0, \dot{\phi}_g = 0$ and $\ddot{\phi}_g = 0$. This enables the identification of K_s, D_s and I_r. These conditions in conjunction with Eq. (2.37a) gives

$$Q_r = I_r \ddot{\phi}_\Delta + D_s \dot{\phi}_\Delta + K_s \phi_\Delta \tag{2.40}$$

The simulations on the HAWC2 model seen in Fig. (2.8) is used two determine the parameters of the approximating 2. order model. The approach is elaborated in appendix B.

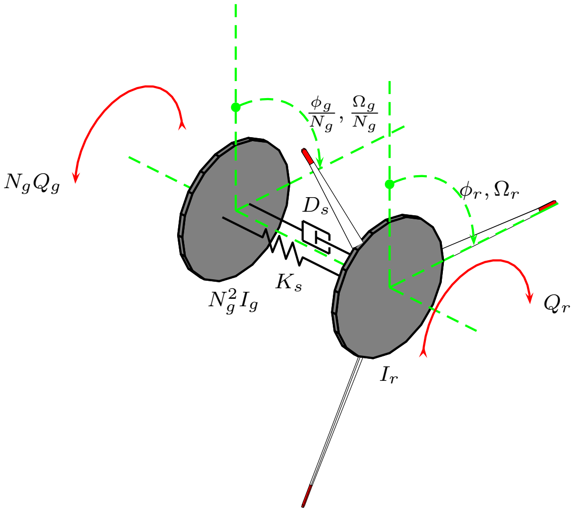


Figure 2.7: Mechanical drive shaft model

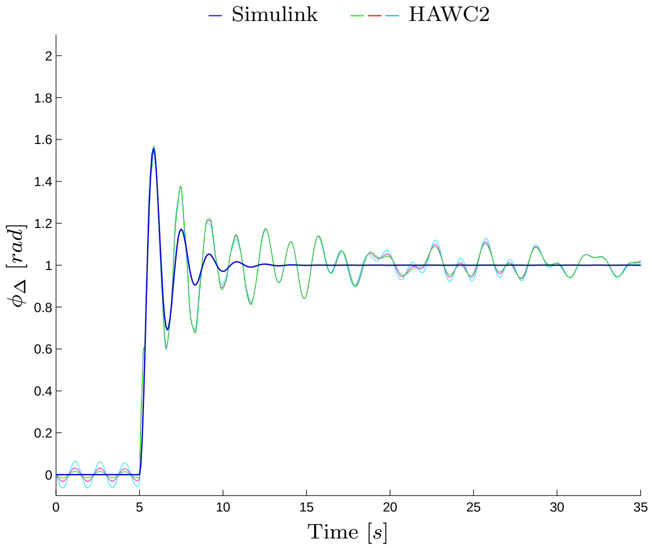


Figure 2.8: Step responses of HAWC2 model compared to approximated 2. order model

2.2.4 Flexible tower

The force (a.k.a thrust) Q_t (eq. (2.7)) exerted by the wind on the wind turbine (mainly the rotor) causes the flexible tower to bend and sway. To simplify the model only the back and forth motion of the nacelle is modeled. The displacement of the nacelle from its original position is denoted x_t.

The tower is modeled as spring-mass-damper system not influenced by gravity (Larsen and Mogensen, 2006), see Fig. (2.9).

$$Q_t = M_t\ddot{x}_t + D_t\dot{x}_t + K_t x_t \tag{2.41}$$

where the displacement of the nacelle in steady state ($\ddot{x}_t = 0$ and $\dot{x}_t = 0$) is

$$x_{t0} = \frac{Q_t}{K_t} \tag{2.42}$$

Eq. (2.41) can be written in a state space formulation

$$\begin{pmatrix} \dot{x}_t \\ \ddot{x}_t \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{K_t}{M_t} & -\frac{D_t}{M_t} \end{bmatrix} \begin{pmatrix} x_t \\ \dot{x}_t \end{pmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M_t} \end{bmatrix} Q_t \tag{2.43}$$

The swaying movement of the nacelle changes the relative wind speed on the rotor. If the nacelle moves forward the relative wind speed is higher than normal and vice versa.

$$v_r = v - \dot{x}_t \tag{2.44}$$

The simulations on the two HAWC2 models seen in Fig. (2.10(a)) and Fig. (2.10(b)) are used two determine the parameters of the approximating 2. order model. The approach is elaborated in appendix B. The fact that they don't match the model indicates that the real towers have more complex dynamics than the approximation but that is expected and will hopefully not prove to be a problem.

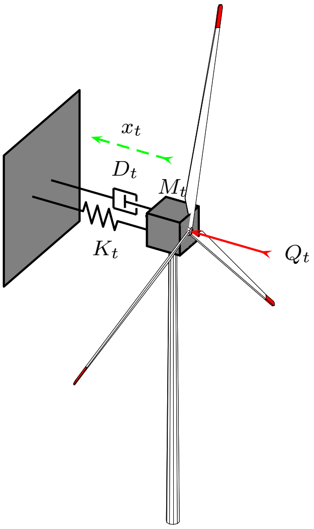
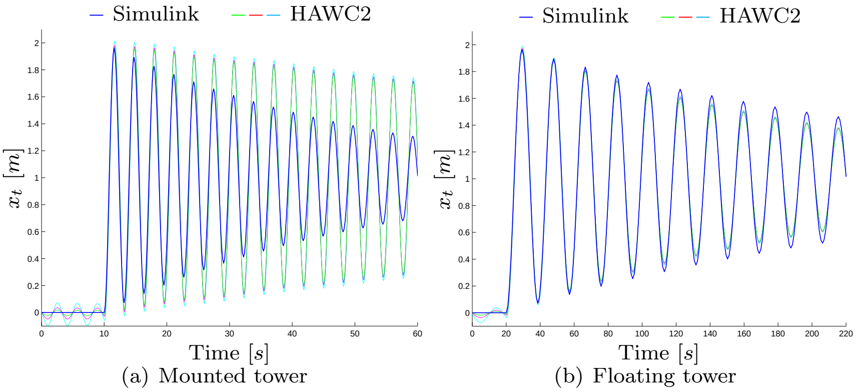


Figure 2.9: Mechanical tower model



(a) Mounted tower

(b) Floating tower

Figure 2.10: Step responses of nacelle displacements approximated to a 2. order system

2.2.5 Collective pitch actuator

(This model is taken from Larsen and Mogensen (2006)). The pitch of the rotor blade is controlled by a hydraulic or electric motor. The actuator can be described by a 2. order transfer function where θ_{ref} is the desired pitch and θ is the actual pitch

$$\omega_n^2 \theta_{ref} = \ddot{\theta} + 2\zeta\omega_n\dot{\theta} + \omega_n^2\theta \tag{2.45}$$

giving the state equation

$$\underbrace{\begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \end{pmatrix}}_{\dot{\mathbf{x}}_\theta} = \underbrace{\begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix}}_{\mathbf{A}_\theta} \underbrace{\begin{pmatrix} \theta \\ \dot{\theta} \end{pmatrix}}_{\mathbf{x}_\theta} + \underbrace{\begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix}}_{\mathbf{B}_\theta} \theta_{ref} \tag{2.46}$$

the pitch actuator is only approximated as a linear system and is in reality subject to several constraints.

$$\theta_{min} \leq \theta \leq \theta_{max} \tag{2.47}$$

$$\dot{\theta}_{min} \leq \dot{\theta} \leq \dot{\theta}_{max} \tag{2.48}$$

$$\ddot{\theta}_{min} \leq \ddot{\theta} \leq \ddot{\theta}_{max} \tag{2.49}$$

The consequences of these constraints are seen in Fig. (2.11).

2.2.6 Generator torque actuator

(This model is taken from Larsen and Mogensen (2006)). The electromagnetic torque of the generator can be described by a 1. order transfer function where $Q_{g_{ref}}$ is the desired torque and Q_g is the actual torque

$$Q_{g_{ref}} = \tau\dot{Q}_g + Q_g \tag{2.50}$$

giving the state equation

$$\underbrace{\left(\dot{Q}_g\right)}_{\dot{\mathbf{x}}_{Q_g}} = \underbrace{\left[-\tfrac{1}{\tau}\right]}_{\mathbf{A}_{Q_g}} \underbrace{Q_g}_{\mathbf{x}_{Q_g}} + \underbrace{\left[\tfrac{1}{\tau}\right]}_{\mathbf{A}_{Q_g}} Q_{g_{ref}} \tag{2.51}$$

the torque actuator is only approximated as a linear system and is in reality subject to several constraints.

$$Q_{g_{min}} \leq Q_g \leq Q_{g_{max}} \tag{2.52}$$

$$\dot{Q}_{g_{min}} \leq \dot{Q}_g \leq \dot{Q}_{g_{max}} \tag{2.53}$$

The consequences of these constraints are seen in Fig. (2.12).

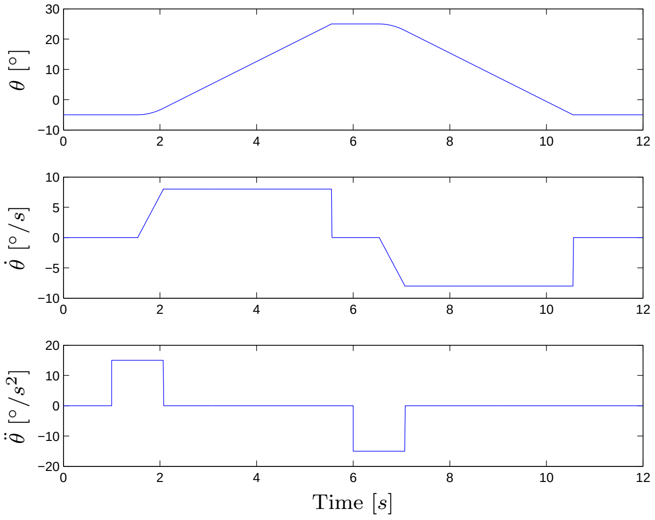


Figure 2.11: Step on constrained pitch actuator, $\theta_{ref}(1) = 1\text{e}7^o$ and $\theta_{ref}(6) = -1\text{e}7^o$

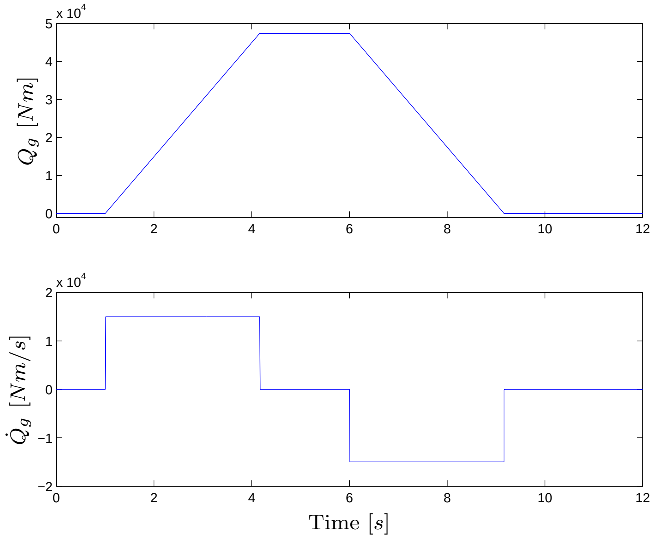


Figure 2.12: Step on constrained generator torque actuator, $Q_{g_{ref}}(1) = 1\text{e}7Nm$ and $Q_{g_{ref}}(6) = -1\text{e}7Nm$

2.3 Level of modeling detail

The controller design model can have various levels of detail. Here is chosen a model with drive shaft torsion and nacelle displacement. That level of detail is denoted wt2, whereas wt1 only includes the drive shaft and wt0 assumes all structure to be rigid. Combing the tower (2.43), the linearized thrust (2.23)

$$\begin{pmatrix} \dot{x}_t \\ \ddot{x}_t \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{K_t}{M_t} & -\frac{D_t}{M_t} \end{bmatrix} \begin{pmatrix} x_t \\ \dot{x}_t \end{pmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial \Omega_r}\Big|_{\Omega_{r0}} \end{bmatrix} \Omega_r + \begin{bmatrix} 0 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial \theta}\Big|_{\theta_0} \end{bmatrix} \theta + \begin{bmatrix} 0 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial v}\Big|_{v_0} \end{bmatrix} v_r \tag{2.54}$$

and the relatve wind speed (2.44) gives

$$\begin{pmatrix} \dot{x}_t \\ \ddot{x}_t \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{K_t}{M_t} & -\frac{D_t}{M_t} - \frac{1}{M_t} \frac{\partial Q_t}{\partial v}\Big|_{v_0} \end{bmatrix} \begin{pmatrix} x_t \\ \dot{x}_t \end{pmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial \Omega_r}\Big|_{\Omega_{r0}} \end{bmatrix} \Omega_r + \begin{bmatrix} 0 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial \theta}\Big|_{\theta_0} \end{bmatrix} \theta + \begin{bmatrix} 0 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial v}\Big|_{v_0} \end{bmatrix} v \tag{2.55}$$

Combing the driveshaft (2.37), the linearized rotor torque (2.11), the tower (2.41), the linearized thrust (2.23) and the relatve wind speed (2.44) gives

$$\underbrace{\begin{pmatrix} \dot{\Omega}_r \\ \dot{\Omega}_g \\ \dot{\phi}_\Delta \\ \dot{x}_t \\ \ddot{x}_t \end{pmatrix}}_{\dot{\mathbf{x}}_{wt2}} = \underbrace{\begin{bmatrix} -\frac{D_s}{I_r} + \frac{1}{I_r} \frac{\partial Q_r}{\partial \Omega_r}\Big|_{\Omega_{r0}} & \frac{D_s}{I_r N_g} & -\frac{K_s}{I_r} & 0 & -\frac{1}{I_r} \frac{\partial Q_r}{\partial v}\Big|_{v_0} \\ \frac{D_s}{I_g N_g} & -\frac{D_s}{I_g N_g^2} & \frac{K_s}{I_g N_g} & 0 & 0 \\ 1 & -\frac{1}{N_g} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial \Omega_r}\Big|_{\Omega_{r0}} & 0 & 0 & -\frac{K_t}{M_t} & -\frac{D_t}{M_t} - \frac{1}{M_t} \frac{\partial Q_t}{\partial v}\Big|_{v_0} \end{bmatrix}}_{\mathbf{A}_{wt2}} \underbrace{\begin{pmatrix} \Omega_r \\ \Omega_g \\ \phi_\Delta \\ x_t \\ \dot{x}_t \end{pmatrix}}_{\mathbf{x}_{wt2}} + \underbrace{\begin{bmatrix} \frac{1}{I_r} \frac{\partial Q_r}{\partial \theta}\Big|_{\theta_0} \\ 0 \\ 0 \\ 0 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial \theta}\Big|_{\theta_0} \end{bmatrix}}_{\mathbf{B}_{wt2}^{\theta}} \theta + \underbrace{\begin{bmatrix} 0 \\ -\frac{1}{I_g} \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{B}_{wt2}^{Q_g}} Q_g + \underbrace{\begin{bmatrix} \frac{1}{I_r} \frac{\partial Q_r}{\partial v}\Big|_{v_0} \\ 0 \\ 0 \\ 0 \\ \frac{1}{M_t} \frac{\partial Q_t}{\partial v}\Big|_{v_0} \end{bmatrix}}_{\mathbf{B}_{wt2}^{v}} v \tag{2.56}$$

the wind turbine model is then augmented with actuator models

$$\underbrace{\begin{pmatrix}\dot{\mathrm{x}}_{wt2}\\ \dot{\mathrm{x}}_{\theta}\\ \dot{\mathrm{x}}_{Q_g}\end{pmatrix}}_{\dot{\mathrm{x}}} = \underbrace{\begin{bmatrix}\mathbf{A_{wt2}} & \begin{bmatrix}\mathbf{B}^{\theta}_{wt2} & \mathbf{0}\end{bmatrix} & \mathbf{B}^{Q_g}_{wt2}\\ \mathbf{0} & \mathbf{A}_{\theta} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{Q_g}\end{bmatrix}}_{\mathrm{A}_c} \underbrace{\begin{pmatrix}\mathrm{x}_{wt2}\\ \mathrm{x}_{\theta}\\ \mathrm{x}_{Q_g}\end{pmatrix}}_{\mathrm{x}} + \underbrace{\begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{B}_{\theta} & \mathbf{0}\\ \mathbf{0} & \mathbf{B}_{Q_g}\end{bmatrix}}_{\mathbf{B}_c} \begin{pmatrix}\theta_{ref}\\ \mathrm{u}_{ref}\end{pmatrix} + \begin{bmatrix}\mathbf{B}^{v}_{wt2}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix} v \quad (2.57)$$

Chapter 3

Full wind range analysis of the HAWT

This chapter will explorer the stationary and dynamic properties of the wind turbine without actuator models over the full operational wind range.

3.1 Operation modes

The objective for controlling a wind turbine is to maximize power production minimizing mechanical stress on the components of the wind turbine. At least between the cut-in wind speed v_1 and the power max wind speed v_4. In the interval

$$v = [v_1; v_4] \tag{3.1a}$$

determine

$$(\Omega_r^*(v), \theta^*(v)) = \underset{(\Omega_r^*(v), \theta^*(v))}{\text{argmax}} \; P_r(\Omega_r, \theta, v) \tag{3.1b}$$

subject to

$$\Omega_{g_{min}} \leq \Omega_r N_g \leq \Omega_{g_{max}} \tag{3.1c}$$

$$0 \leq \eta P_r \leq P_{nom} \tag{3.1d}$$

after v_4 the objective is to keep primary controlled variables at their nominal values until the cut-out wind speed is reached. The wind turbine has different modes of operation depending on the wind speed and the properties of the wind turbine. The limitations of operation are determined by the properties of the generator, it is only able to work within a limited range of generator speed Ω_g and generator power P_e. The wind turbine can operate in continuous mode within these bounds

Region	v	Ω_g	ηP_r
I	(v_1, v_2)	$\Omega_{g_{min}}$	$(0, P_L)$
II	(v_2, v_3)	$(\Omega_{g_{min}}, \Omega_{g_{max}})$	(P_L, P_H)
III	(v_3, v_4)	$\Omega_{g_{max}}$	(P_H, P_{nom})
IV	$(v_4, ...)$	$\Omega_{g_{max}}$	P_{nom}

Table 3.1: Operation modes

The critical wind speeds are the wind speeds where the wind turbine changes form one mode of operation to another.

$$\eta P_r(v_1, \Omega_{rmin}, \theta^*) = 0 \tag{3.2}$$

$$v_2 = \Omega_{rmin} R\lambda^* \tag{3.3}$$

$$v_3 = \Omega_{rmax} R\lambda^* \tag{3.4}$$

$$\eta P_r(v_4, \Omega_{rmin}, \theta^*) = P_{nom} \tag{3.5}$$

v_1	v_2	v_3	v_4
2.7 [m/s]	6.5 [m/s]	11.4 [m/s]	11.6 [m/s]

Table 3.2: Critical wind speed

The values of the table show a very narrow region III that could be widened by lowering the generator speed or increasing the generator power, it as arguable whether or not it should even be included in the control design but for a sense of completion it has been left untouched.

The figures Fig. (3.1), Fig. (3.2), Fig. (3.3) and Fig. (3.4) show the quasi-stationary values of the variables depicted over wind speed sweep for the floating wind turbine the only difference between these and those for the mounted wind turbine is in Fig. (3.3) where the nacelle displacement is significantly smaller. There a two versions one with an optimal pitch and one with a fixed pitch. The fixed pitch version is the one to be used in this project to simplify the control problem, this will also be explained in chapter 4.

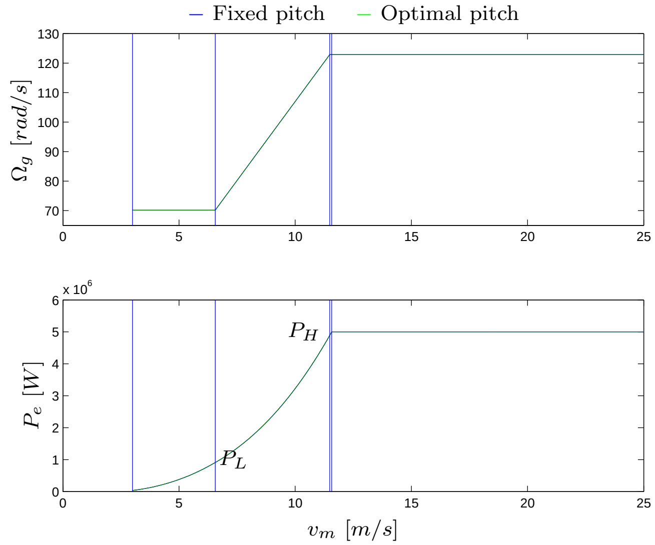


Figure 3.1: Wind speed sweep of primary controlled variables

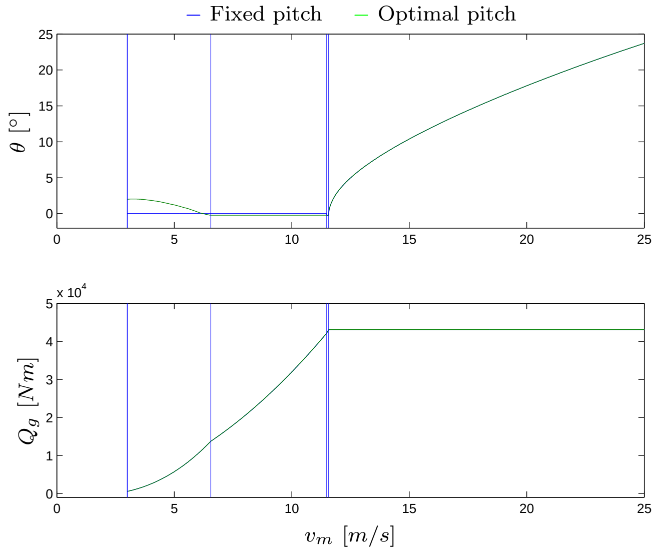


Figure 3.2: Wind speed sweep of input variables

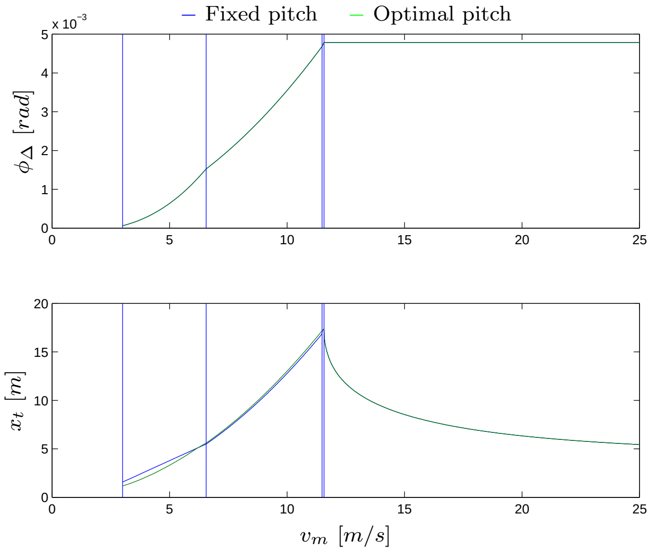


Figure 3.3: Wind speed sweep of secondary structural variables of floating wind turbine

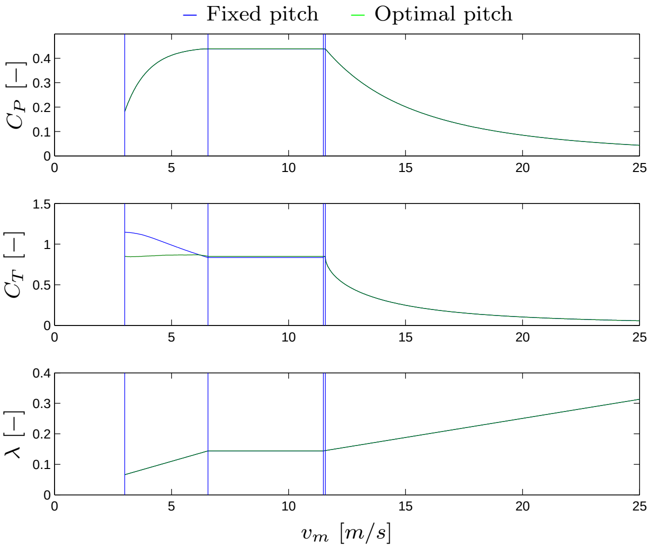


Figure 3.4: Wind speed sweep of aerodynamic coefficients and tip-speed-ratio

3.2 Dynamic analysis of the HAWT

A sweep of the eigenvalues of the time continuous system gives the following results for the two wind turbines. Fig. (3.5) and Fig. (3.6) shows plots of the eigenvalues of the floating wind turbine. For both wind turbines the tables and figures show that λ_1 and λ_2 are fairly constant whereas λ_4 and λ_5 are varying with wind speed to some degree but the real nonlinearity is λ_3 which varies a lot with respect to the wind speed.

v	λ_1, λ_2	λ_3	λ_4, λ_5
5	$-9.5786 \pm 10.4259i$	-0.0220	$-0.0698 \pm 0.3332i$
10	$-9.5796 \pm 10.4250i$	-0.0415	$-0.0982 \pm 0.3461i$
15	$-9.5818 \pm 10.4228i$	-0.1162	$-0.0914 \pm 0.3041i$
20	$-9.5868 \pm 10.4180i$	-0.29575	$-0.0573 \pm 0.2904i$

Table 3.3: Eigenvalues of floating wind turbine

v	λ_1, λ_2	λ_3	λ_4, λ_5
5	$-9.5786 \pm 10.4259i$	-0.0223	$-0.0932 \pm 1.9767i$
10	$-9.5796 \pm 10.4250i$	-0.0470	$-0.1205 \pm 1.9786i$
15	$-9.5818 \pm 10.4228i$	-0.1032	$-0.1231 \pm 1.9713i$
20	$-9.5868 \pm 10.4180i$	-0.2295	$-0.1153 \pm 1.9659i$

Table 3.4: Eigenvalues of fixed wind turbine

The control theory used in this project is time discrete and the continuous time formulation of the linear dynamic systems has to be time-discretized. The sampling time is chosen to be $T_s = 0.02s$ giving a sampling frequency of $f_s = 50Hz$. From (Poulsen, 2007) we have the zero-order-hold time discretization of a continuous time system

$$\mathbf{A} = e^{\mathbf{A_c}T_s} \tag{3.6}$$

$$\mathbf{B} = \int_0^{T_s} e^{\mathbf{A_c}s}\mathbf{B_c}ds \tag{3.7}$$

Considering that the fastest eigenvalue in the model is $\lambda_3 = -0.02$ giving a eigenfrequency of $f = \frac{2\pi}{|-0.02|} \approx 314Hz$ the sampling time might not be sufficiently fast, but its has not been a problem in this project, since the eigenvalue belongs to the aerodynamic pitching dynamics which are not active at the low wind speeds.

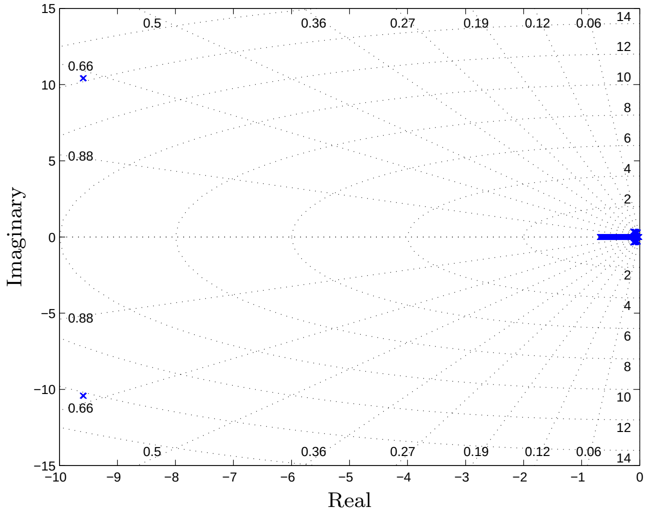


Figure 3.5: Wind speed sweep of system eigenvalues of floating wind turbine

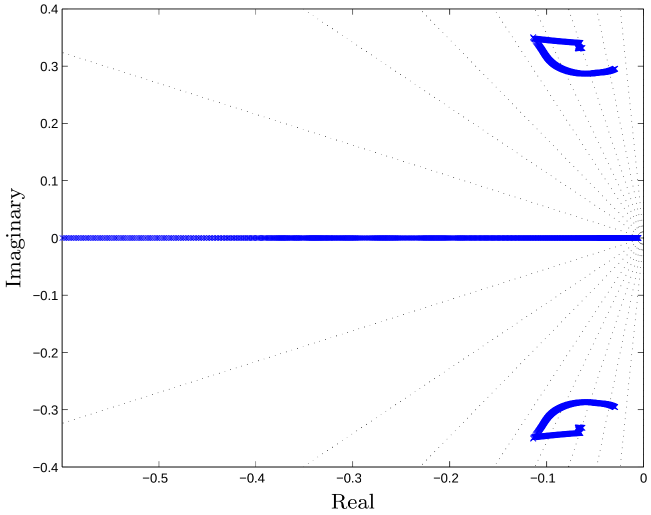


Figure 3.6: Zoomed in on slow eigenvalues of floating wind turbine

Chapter 4

Control strategies

This chapter will discuss the control strategies for the different operation modes. The control problem of a plant can be divided into different levels.

- **The strategic level** is where decisions concerning overall operation are made. This could be if the wind farm or a part of it should shut down due to repairs or the production of electricity is greater than the consumption. The time constants at this level are typically in the range of hours or days.
- **The tactical level** is where local decisions concerning the local operation are made. This could be to shut down the wind turbine or to switch from one mode of operation to another. The time constants at this level are typically in the range of minutes or seconds.
- **The operational level** is where the actual control of the plant is being performed. This could be a PID or LQ controller. The time constant at this level are typically in the range of milliseconds.

In this project attention will only be on the tactical and operational levels of control. Figures showing system blocks at either the tactical or operational level are colored respectively magenta and cyan.

4.1 The tactical level

The tactical supervisor governs which of the control objectives should be active. The supervisor switches between the operation modes with the switching conditions seen in Fig. (4.1). The setup is inspired by (Hammerum, 2006) but additional criteria has been added to avoid unstable switching between the operation modes.

The two power levels P_L and P_H are the power levels at the transition between region I to region II and from region II to region III which can be seen in Fig. (3.1)

The primary objective of controlling the HAWT is of course to capture as much power from the wind as possible. But pursuing this objective without regards to other factors might prove to be impractical. In the power maximizing regions, i.e. I, II and III, the operating point on the C_P curve is placed on the flat top of the C_P curve. It therefore customary to avoid pitching control at these operating modes to prevent the pitch actuators from being used without much effect. Furthermore linear control at the ridge of the C_P curve might prove to be fatal if the wind turbine enters the stall region Fig. (2.4) and the wind turbine thus reacts opposite of what is expected. It is therefore decided that the collective pitching of the rotor blades should not be used in regions I, II and III. Instead the pitch angle is kept constant

$$\theta^* = 0^\circ \tag{4.1}$$

which is used as a switching criteria of the tactical supervisor when the wind turbine goes from region IV to region III.

The power values at the borders of region II are

$$P_L = \eta\frac{1}{2}\rho R^5 \lambda^{*3} \Omega_{r\,min}^{\;3} \tag{4.2}$$

$$P_H = \eta\frac{1}{2}\rho R^5 \lambda^{*3} \Omega_{r\,max}^{\;3} \tag{4.3}$$

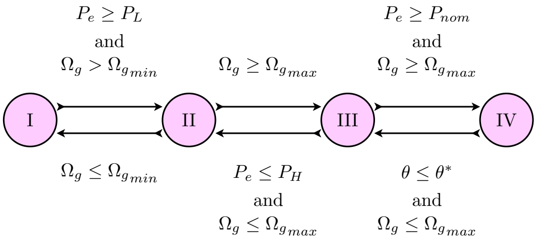

$P_e \geq P_L$
and
$\Omega_g > \Omega_{g_{min}}$
$\Omega_g \geq \Omega_{g_{max}}$
$P_e \geq P_{nom}$
and
$\Omega_g \geq \Omega_{g_{max}}$
I
II
III
IV
$\Omega_g \leq \Omega_{g_{min}}$
$P_e \leq P_H$
and
$\Omega_g \leq \Omega_{g_{max}}$
$\theta \leq \theta^*$
and
$\Omega_g \leq \Omega_{g_{max}}$

Figure 4.1: Mode transition criteria for the tactical supervisor

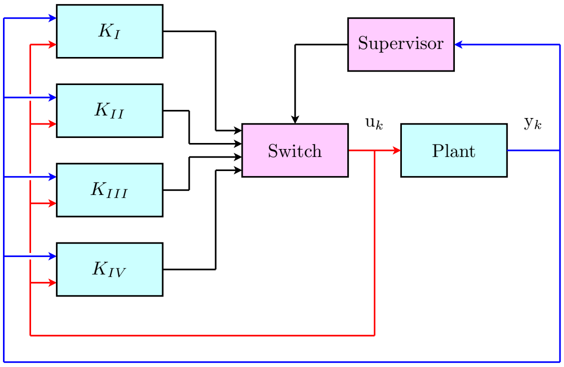


Figure 4.2: Hybrid controller

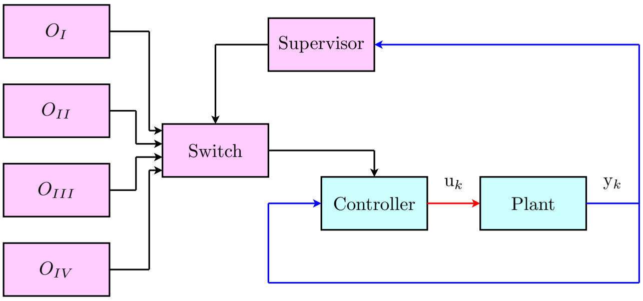


Figure 4.3: Multi objective controller

4.2 The operational level

At the operational level the control objectives for each operation mode has to defined. In this project they are simply defined by solely focusing efficient power capture within the allowed mechanical limits. Other operation modes could focus on these objectives as well as for instance reducing noise or fatigue damage. There could for instance be several region IV modes where slightly different objectives where defined for each of them depending on the time of day etc. The different control objectives could either be controlled different controllers with different control objectives, i.e. hybrid controller Fig. (4.2), or by a single controller with different control objectives, i.e. multi objective controller Fig. (4.3). The latter could be a nonlinear or gain scheduling controller but that path will be not be explored in this project.

4.2.1 Pitch controller for region I, II and III

In region I, II and III the pitch should be fixed. A separate pitch controller will control the pitch and keep it at the fixed set point. This especially useful when the controller switches from region IV to region III, where the pitch is in downward motion which should be carefully stopped and kept steady at the set point.

$$\mathrm{r} = \theta = \theta^* = 0^\circ \tag{4.4}$$

$$\mathrm{u} = \theta_{ref} \tag{4.5}$$

4.2.2 Low region controller - K_I

In region I the wind is to slow to ensure an optimal C_P while keeping the rotational speed of the generator within its allowed limits. Hence the rotational speed should be kept constant at its lower limit to capture as much wind energy as possible. The wind turbine is only controlled with the generator torque.

$$\mathrm{r} = \Omega_g = \Omega_{g_{min}} \tag{4.6}$$

$$\mathrm{u} = Q_{g_{ref}} \tag{4.7}$$

4.2.3 Middle region controller - K_{II}

In region II the primary operational objective is maximize wind power capture by optimizing C_P. Since no measurement of the effective wind speed is available another approach has to implemented.

$$\mathrm{r} = P_e = \eta \; P_r^*(v) \tag{4.8}$$

$$\mathrm{u} = Q_{g_{ref}} \tag{4.9}$$

unless the wind speed is being measured or estimated thats objective is difficult to obtain directly.

A widely used method known as $P\Omega$ control can be used to track the optimal power production. Under optimal operation the following equation holds

$$P_r^*(v) = \frac{1}{2}\rho\pi R^2 v^3 C_P^* \tag{4.10}$$

this imply control at the maximum of the C_P-curve, i.e. $C_P^* = C_P(\lambda^*, \theta^*)$. Since the pitch is fixed at this region only the tip-speed-ratio can be used to maximize C_P, i.e. $\lambda^* = \frac{v}{\Omega_r^* R}$. This results in

$$P_r^*(v) = \underbrace{\frac{1}{2}\rho\pi R^5 \lambda^{*3}}_{K_2} \; \Omega_r^{*3} \tag{4.11}$$

this means that the optimal power reference can be tracked by a function dependent on the rotor speed

$$P_r^*(v) \approx P_r^*(\Omega_r) = K_2 \; \Omega_r^3 \tag{4.12}$$

An adaptive method of this approach have been suggested (Johnson et al., 2004)

Region II is the only region with a variable reference and the method sanity of the method is validated in the illustrative example at the end of chapter 5.

4.2.4 High region controller - K_{III}

In region III the upper limit of the generator rotational speed is reached but the upper limit of generator is not. This means that the controller should keep the generator speed at is maximum to obtain as much power production as possible in order to optimize C_P.

$$\mathrm{r} = \Omega_g = \Omega_{g_{max}} \tag{4.13}$$

$$\mathrm{u} = Q_{g_{ref}} \tag{4.14}$$

4.2.5 Top region controller - K_{IV}

In region IV both the upper power and upper speed limit of the generator is reached and the control objective is to keep both these stable at their desired set points. Both the pitch and the generator torque is now used to control the wind turbine.

$$\mathrm{r} = \begin{pmatrix} \Omega_g \\ P_e \end{pmatrix} = \begin{pmatrix} \Omega_{g_{max}} \\ P_{e_{nom}} \end{pmatrix} \tag{4.15}$$

$$\mathrm{u} = \begin{pmatrix} \theta_{ref} \\ Q_{g_{ref}} \end{pmatrix} \tag{4.16}$$

it is typically in this region that more advanced control methods is developed since it is a multiple-input multiple-output control problem.

Part II

Theory of methods

Chapter 5

Unconstrained Linear Quadratic Control

In this chapter an offset-free unconstrained linear quadratic controller is introduced. The offset-free mechanisms of this particular setup (known as disturbance estimation and origin shifting) requires and state observer to obtain offset-free control. The setup is widely used in MPC implementations since it enables a method to respect constraints even under the influence of disturbances and model/plant mismatch, as seen in chapter 7.

5.1 The standard linear quadratic problem

Linear quadratic control (LQ) offers a systematic approach to control plants with multiple inputs and multiple outputs (MIMO) and with cross coupling states. The control problem is stated as quadratic minimization problem that should resemble the real control objectives of the plant. The minimization problem can be formulated in discrete- or continuous time and describes the plant progress within a given future time horizon. In the discrete time case the cost function

J, with the horizon sample length N, of the minimization problem is

$$J(\mathrm{k}) = \sum_{i=k}^{N+k} \|\mathrm{z}_i\|^2_{\mathbf{W}} = \sum_{i=k}^{N+k} (\mathbf{E}\mathrm{x}_i + \mathbf{F}\mathrm{u}_i)^T \mathbf{W}(\mathbf{E}\mathrm{x}_i + \mathbf{F}\mathrm{u}_i) =$$

$$\sum_{i=k}^{N+k} \mathrm{x}_i^T \underbrace{\mathbf{E}^T\mathbf{W}\mathbf{E}}_{\mathbf{Q_1}} \mathrm{x}_i + \mathrm{u}_i^T \underbrace{\mathbf{F}^T\mathbf{W}\mathbf{F}}_{\mathbf{Q_2}} \mathrm{u}_i + 2\mathrm{x}_i^T \underbrace{\mathbf{E}^T\mathbf{W}\mathbf{F}}_{\mathbf{Q_{12}}} \mathrm{u}_i \tag{5.1}$$

The problem should be formulated such that deviation from the desired operating point of the respective optimization variables is penalized by the cost function. To prioritize the minimization of the individual variables a quadratic weight is placed on of them where a high weight indicates that the variable should be minimized heavily whereas a low weight or even no weight at all indicates the deviation of the variable away from the origo is unimportant. The unconstrained dynamic optimization problem can be solved using dynamic programming. This entails dividing the problem into a sequence of unconstrained minimization problems, one at each sample, starting from the future incident $i = N + 1$ and then iterating through the minimization problems until $i = 0$. This approach is described in (Poulsen, 2007, page 100)

The minimal value of the criterion

$$V(\mathrm{x}_k) = \min_{\vec{\mathrm{u}}_k} J(k) \tag{5.2}$$

is given by the (Bellmans equation) recursion

$$V(\mathrm{x}_i) = \min_{\mathrm{u}_i}\{J(i)\} = \min_{\mathrm{u}_i}\{\|\mathrm{z}_i\|^2_{\mathbf{W}} + V(\mathrm{x}_{i+1})\} \tag{5.3}$$

There is no loss beyond the horizon

$$V(\mathrm{x}_{N+1}) = 0 \tag{5.4}$$

this means that the minimal cost at $i = N$ is given by

$$\mathrm{u}_N = 0 \tag{5.5}$$

thus giving the the terminal cost (terminal costs can also be defined independently, from the rest of the cost function)

$$V(\mathrm{x}_N) = \mathrm{x}_N^T \mathbf{Q_1} \mathrm{x}_N \tag{5.6}$$

a candidate value function is applied

$$V(\mathrm{x}_i) = \mathrm{x}_i^T \mathbf{S}_i \mathrm{x}_i \tag{5.7}$$

inserting the candidate function (5.7) into the recursion (5.3) gives the inner part of the minimization

$$\begin{aligned} J(i) = \|\mathrm{z}_i\|^2_{\mathbf{W}} + \mathrm{x}_{i+1}^T \mathbf{S}_{i+1} \mathrm{x}_{i+1} = \mathrm{x}_i^T \left[\mathbf{A}^T \mathbf{S}_{i+1} \mathbf{A} + \mathbf{Q_1}\right] \mathrm{x}_i \\ + \mathrm{u}_i^T \left[\mathbf{B}^T \mathbf{S}_{i+1} \mathbf{B} + \mathbf{Q_2}\right] \mathrm{u}_i + 2\mathrm{x}_i^T \left[\mathbf{A}^T \mathbf{S}_{i+1} \mathbf{B} + \mathbf{Q_{12}}\right] \mathrm{u}_i \end{aligned} \tag{5.8}$$

the optimal solution to minimization problem is given by

$$0 = \frac{\partial J(i)}{\partial \mathrm{u}_i} = 2\mathrm{u}_i^T \left[\mathbf{B}^T \mathbf{S}_{i+1} \mathbf{B} + \mathbf{Q_2}\right] + 2\mathrm{x}_i^T \left[\mathbf{A}^T \mathbf{S}_{i+1} \mathbf{B} + \mathbf{Q_{12}}\right] \tag{5.9}$$

the control law with the optimal feed back gain $\mathbf{K}_i$ which minimizes $J(i)$ is then

$$\mathrm{u}_i = -\underbrace{\left[\mathbf{B}^T \mathbf{S}_{i+1} \mathbf{B} + \mathbf{Q_2}\right]^{-1} \left[\mathbf{B}^T \mathbf{S}_{i+1} \mathbf{A} + \mathbf{Q_{12}}^T\right]}_{\mathbf{K}_i} \mathrm{x}_i \tag{5.10}$$

inserting the optimal control law (5.10) into the inner part (5.8) gives the minimal cost

$$V(\mathrm{x}_i) = \mathrm{x}_i^T \left[\mathbf{A}^T \mathbf{S}_{i+1} \mathbf{A} + \mathbf{Q_1}\right] \mathrm{x}_i - \mathrm{x}_i^T \left[\mathbf{A}^T \mathbf{S}_{i+1} \mathbf{B} + \mathbf{Q_{12}}\right] \mathbf{K}_i \mathrm{x}_i \tag{5.11}$$

this leads to the recursive discrete-time Riccati equation

$$\mathbf{S}_i = \mathbf{Q_1} + \mathbf{A}^T \mathbf{S}_{i+1} \mathbf{A} - \left[\mathbf{A}^T \mathbf{S}_{i+1} \mathbf{B} + \mathbf{Q_{12}}\right] \mathbf{K}_i \tag{5.12}$$

The minimal total loss of the optimal control strategy is given by the recursion (5.3) and the candidate function (5.7)

$$V(\mathrm{x}_k) = \min_{\vec{\mathrm{u}}_k} J(k) = \mathrm{x}_k^T \mathbf{S}_k \mathrm{x}_k \tag{5.13}$$

For a infinite horizon, i.e. $N \to \infty$, the recursive discrete Riccati equation becomes an algebraic discrete Riccati equation since $\mathbf{S}_N = \mathbf{S}_{N+1}$

$$J(\mathrm{k}) = \sum_{i=k}^{\infty} \|\mathrm{z}_i\|^2_{\mathbf{W}} \tag{5.14a}$$

$$\mathbf{K} = \left[\mathbf{B}^T \mathbf{S} \mathbf{B} + \mathbf{Q_2}\right]^{-1} \left[\mathbf{B}^T \mathbf{S} \mathbf{A} + \mathbf{Q_{12}}^T\right] \tag{5.14b}$$

$$\mathbf{S} = \mathbf{Q_1} + \mathbf{A}^T \mathbf{S} \mathbf{A} - \left[\mathbf{A}^T \mathbf{S} \mathbf{B} + \mathbf{Q_{12}}\right] \mathbf{K} \tag{5.14c}$$

The minimal loss of the optimal control strategy is then

$$V(\mathrm{x}_k) = \min_{\vec{\mathrm{u}}_k} J(k) = \mathrm{x}_k^T \mathbf{S} \mathrm{x}_k \tag{5.15}$$

$$\mathrm{u}_i = -\mathbf{K} \mathrm{x}_i \quad \text{for} \quad i = (k, \ldots, \infty) \tag{5.16}$$

giving the closed loop

$$\mathrm{x}_{k+1} = \mathbf{\Phi}\mathrm{x}_k; \quad \mathbf{\Phi} = \mathbf{A} - \mathbf{B}\mathbf{K} \tag{5.17}$$

The infinite horizon can be shown to be stabilizing either by Lyapunov stability theory or by invariant set theory. The latter approach will be demonstrated here. A set is invariant if, once a state enters that set it can no longer leave. So, for instance, a set $\mathcal{X}$ is invariant if

$$\mathrm{x}_k \in \mathcal{X} \Rightarrow \mathrm{x}_{k+1} \in \mathcal{X} \tag{5.18}$$

this implies that

$$\mathrm{x}_k \in \mathcal{X} \Rightarrow \mathrm{x}_{k+i} \in \mathcal{X}; \quad \forall i > 0 \tag{5.19}$$

a good candidate for such an invariant set is an ellipsoidal set $\mathcal{X}_e \subset \mathcal{X}$ based on the hessian $\mathbf{S}$ determined from the Riccati equation eq. (5.14c)

$$\mathcal{X}_e = \{\mathrm{x}|\mathrm{x}^T\mathbf{S}\mathrm{x} \leq c)\}; \quad \mathbf{S} > 0 \tag{5.20}$$

where c is the outer rim of the invariant set. This is similar to local stability in Lyapunov stability theory. The ellipsoidal set is invariant if

$$\mathbf{\Phi}^T\mathbf{S}\mathbf{\Phi} - \mathbf{S} \leq 0 \tag{5.21}$$

shown by inserting eq. (5.17) into eq. (5.18)

$$\begin{aligned} \mathrm{x}_k^T\mathbf{S}\mathrm{x}_k &= c \Rightarrow \mathrm{x}_{k+1}^T\mathbf{S}\mathrm{x}_{k+1} \leq c \\ \mathrm{x}_k^T\mathbf{S}\mathrm{x}_k &= c \Rightarrow \mathrm{x}_k^T\mathbf{\Phi}^T\mathbf{S}\mathbf{\Phi}^T\mathrm{x}_k \leq c \\ &= c \Rightarrow \mathrm{x}_k^T[\mathbf{\Phi}^T\mathbf{S}\mathbf{\Phi} - \mathbf{S}]\mathrm{x}_k \leq 0; \quad \forall \mathrm{x}_k \in \mathcal{X} \\ &\equiv \mathbf{\Phi}^T\mathbf{S}\mathbf{\Phi} - \mathbf{S} \leq 0 \end{aligned} \tag{5.22}$$

if no constraints are present then the invariant set can be infinitely large, i.e. $c \rightarrow \infty$. Similar to global stability in Lyapunov stability theory. The results presented here are explained in (Rossiter, 2003) and (Blanchini, 1999).

5.2 Offset-free reference tracking

Closed-loop performance of model-based controllers is directly related to model accuracy. In practice, modeling error and unmeasured disturbances can lead to a steady-state offset between the desired output and the measured output unless precautions are taken in the control design. In this section two precautionary measures will be discussed: 1. Reference tracking error integration Fig. (5.1) and 2. Shifting of stationary points based on references and modeled disturbances Fig. (5.2). It will discussed why the latter approach is superior, especially when applied to constrained MPC.

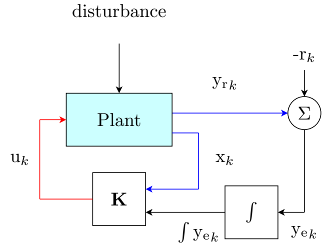


Figure 5.1: LQI control

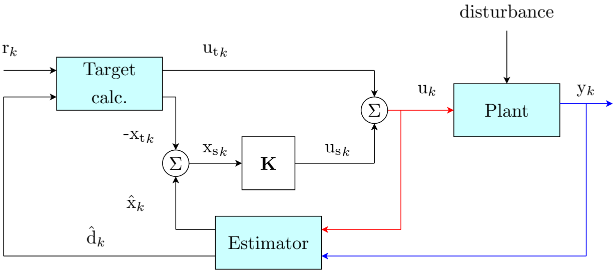


Figure 5.2: Origin shifting unconstrained LQ control with state and disturbance estimation (ULQ)

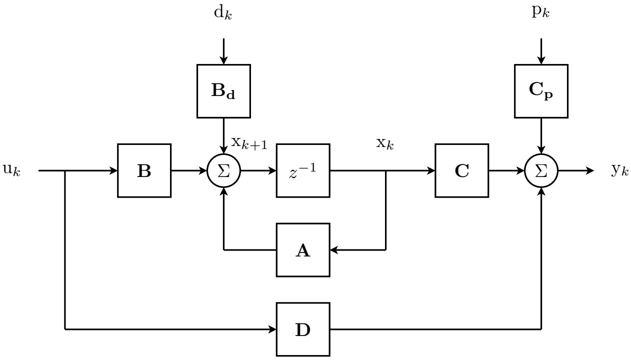


Figure 5.3: Model augmented with unmeasured constant step disturbances (without noise)

5.2.1 Reference error integration (LQI)

A simple way of ensuring offset-free reference tracking is to integrate the error between the reference signal and the output signal, augmenting the system with the reference error integration state and minimize the augmented state in the controller cost function. This type of LQ controller is sometimes called an LQI controller where I means integration of the reference error. The LQI approach is similar to that of the integral part of a traditional PI(D) controller. Reference tracking error integration has several major shortcomings: Integrator wind-up when the reference is unreachable, e.g. when another controller is active. Furthermore, the computational burdens that the augmented system system puts on the control algorithms increase with the order of the model.

5.2.2 Disturbance modeling and origin shifting controller

This approach was introduced in (Kwakernaak and Sivan, 1972) and have recently been discussed in (Muske and Badgwell, 2002) and (Pannocchia and Rawlings, 2003) where disturbance modeling has been applied to MPC. Disturbance modeling offers a different approach to achieve offset-free control. The method involves augmenting the plant model to include a constant step disturbance model. The unmeasured disturbances are estimated with an estimator and their undesired influence on the plant is negated by shifting the origin of the controller to a new operating point that ensures stable offset-free control of the plant.

5.2.2.1 Nonzero reference tracking

Before the disturbance rejection concept is introduced, the concept of origin shifting is presented with nonzero reference tracking. This method works on a nominal model that matches an undisturbed plant perfectly but does not guarantee offset-free set point tracking otherwise. The variables are denoted with subscript $_\mathrm{t}$ to indicate which steady state target value they should have to be at a equilibrium point for the desired reference. The steady state equations of the system are

$$\mathrm{x}_{\mathrm{t}k} = \mathbf{A}\mathrm{x}_{\mathrm{t}k} + \mathbf{B}\mathrm{u}_{\mathrm{t}k} \tag{5.23}$$

$$\mathrm{y}_{\mathrm{t}k} = \mathbf{C}\mathrm{x}_{\mathrm{t}k} + \mathbf{D}\mathrm{u}_{\mathrm{t}k} \tag{5.24}$$

$$\mathrm{r}_{\mathrm{t}k} = \mathrm{y}_{\mathrm{t}k} \tag{5.25}$$

Not all outputs can be steered to any number of linear independents references. In general it is not possible to steer more outputs than there is linear independent inputs. Thus only a subset of the outputs should be considered when deciding to track references. The reference controlled outputs $\mathrm{y_r}$ should be steered toward the references r. The subset of outputs is defined by the transformation matrix **H**

$$\mathrm{r}_k = \mathrm{y_r}_k = \mathbf{H}(\mathbf{C}\mathrm{x_t}_k + \mathbf{D}\mathrm{u_t}_k) \tag{5.26}$$

giving the linear reference tracking problem known as target calculation or origin shifting calculation

$$\begin{bmatrix} \mathbf{A}-\mathbf{I} & \mathbf{B} \\ \mathbf{HC} & \mathbf{HD} \end{bmatrix} \begin{pmatrix} \mathrm{x_t}_k \\ \mathrm{u_t}_k \end{pmatrix} = \begin{pmatrix} 0 \\ \mathrm{r}_k \end{pmatrix} \tag{5.27}$$

which can only be solved the if problem has full rank and thus as many inputs and controlled outputs (Pannocchia and Rawlings, 2003, eq. (11))

$$\operatorname{rank} \begin{bmatrix} \mathbf{A}-\mathbf{I} & \mathbf{B} \\ \mathbf{HC} & \mathbf{HD} \end{bmatrix} = n_x + n_r \tag{5.28}$$

The shifted variables are denoted with a subscript $_\mathrm{s}$ are defined as the difference between the real variables and the target variables giving the shifted optimization variable

$$\underbrace{(\mathrm{z}-\mathrm{z_t})}_{\mathrm{z_s}} = \mathbf{E}\underbrace{(\mathrm{x}-\mathrm{x_t})}_{\mathrm{x_s}} + \mathbf{F}\underbrace{(\mathrm{u}-\mathrm{u_t})}_{\mathrm{u_s}} \tag{5.29}$$

5.2.2.2 Disturbance model and estimator

The concept of constant step disturbance modeling can be used to counter actual disturbances or model/plant mismatch. The disturbances can be modeled as either constant input/state disturbances, i.e. d

$$\mathrm{x}_{k+1} = \mathbf{A}\mathrm{x}_k + \mathbf{B_d}\mathrm{d}_k + \mathbf{B}\mathrm{u}_k + \mathrm{w_x}_k \tag{5.30a}$$
$$\mathrm{d}_{k+1} = \mathrm{d}_k + \mathrm{w_d}_k \tag{5.30b}$$

or constant output disturbances, i.e. p

$$\mathrm{p}_{k+1} = \mathrm{p}_k + \mathrm{w_p}_k \tag{5.30c}$$
$$\mathrm{y}_k = \mathbf{C}\mathrm{x}_k + \mathbf{C_p}\mathrm{p}_k + \mathbf{D}\mathrm{u}_k + \mathrm{w_y}_k \tag{5.30d}$$

which can be formulated as the nominal model augmented with the disturbances

$$\underbrace{\begin{pmatrix} \mathrm{x}_{k+1} \\ \mathrm{d}_{k+1} \\ \mathrm{p}_{k+1} \end{pmatrix}}_{\mathrm{e}_{k+1}} = \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{B_d} & \mathbf{0} \\ 0 & \mathbf{I} & \mathbf{0} \\ 0 & \mathbf{0} & \mathbf{I} \end{bmatrix}}_{\hat{\mathbf{A}}} \underbrace{\begin{pmatrix} \mathrm{x}_k \\ \mathrm{d}_k \\ \mathrm{p}_k \end{pmatrix}}_{\mathrm{e}_k} + \underbrace{\begin{bmatrix} \mathbf{B} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\hat{\mathbf{B}}} \mathrm{u}_k + \underbrace{\begin{pmatrix} \mathrm{w}_{\mathrm{x}k} \\ \mathrm{w}_{\mathrm{d}k} \\ \mathrm{w}_{\mathrm{p}k} \end{pmatrix}}_{\mathrm{w}_{\mathrm{e}k}} \tag{5.31a}$$

$$\mathrm{y}_k = \underbrace{\begin{bmatrix} \mathbf{C} & \mathbf{0} & \mathbf{C_p} \end{bmatrix}}_{\hat{\mathbf{C}}} \begin{pmatrix} \mathrm{x}_k \\ \mathrm{d}_k \\ \mathrm{p}_k \end{pmatrix} + \underbrace{\begin{bmatrix} \mathbf{D} \end{bmatrix}}_{\hat{\mathbf{D}}} \mathrm{u}_k + \mathrm{w}_{\mathrm{y}k} \tag{5.31b}$$

where the state and output noise is assumed zero-mean Gaussian distributed white noise with the variances $\mathrm{v_x}$, $\mathrm{v_d}$, $\mathrm{v_p}$ and $\mathrm{v_y}$

$$\mathrm{w}_{\mathrm{e}k} \in N(0, \mathbf{R_e}); \quad \mathbf{R_e} = \mathrm{diag}\,(r_x\mathbf{I}\ r_d\mathbf{I}\ r_p\mathbf{I}) \tag{5.32}$$

$$\mathrm{w}_{\mathrm{y}k} \in N(0, \mathbf{R_y}); \quad \mathbf{R_y} = r_y\mathbf{I} \tag{5.33}$$

Deciding how to structure the augmented model can prove to be difficult. (Pannocchia and Rawlings, 2003, Lemma 3) states that there should be as many disturbances as there are measurement

$$n_d + n_p = n_y \tag{5.34}$$

In order to obtain estimates leading to offset-free control the pair $(\mathbf{A}, \mathbf{C})$ should be observable and the augmented system should have full rank (Pannocchia and Rawlings, 2003, Theorem 1)

$$\mathrm{rank} \begin{bmatrix} \mathbf{A} - \mathbf{I} & \mathbf{B_d} & \mathbf{0} \\ \mathbf{C} & \mathbf{0} & \mathbf{C_p} \end{bmatrix} = n_x + n_y \tag{5.35}$$

Even though this requirement ensure offset-free control, the task of structuring a disturbance model still remains a problem. It should be noted that state disturbances trust the measured state itself but not its relation to other states and inputs. Output disturbances trust measured output but not the relation to the states and inputs. These relations leads to a rule of thumb for disturbance modeling

- **State disturbances** should be included in the disturbance model for each directly measured state.
- **Output disturbances** should be included in the disturbance model for each directly measured output that is not a state directly but a function of states and inputs.

Since the disturbances can't be measured they have to be estimated. Ordinary or predictive Kalman filters are appropriate candidates for such an estimation. In this project outputs which are dependent on control signals are present and the direct term in the output equation is thus not equal to zero, i.e. $\mathbf{D} \neq 0$. This leads to the predictive Kalman filter as the appropriate estimator since the ordinary Kalman filter would have a problem regarding causality.

The steady-state predictive Kalman filter has the form of an optimal observer

$$\hat{\mathrm{e}}_{k+1} = \hat{\mathbf{A}}\hat{\mathrm{e}}_k + \hat{\mathbf{B}}\mathrm{u}_k + \mathbf{L}(\mathrm{y}_k - \hat{\mathrm{y}}_k) \tag{5.36}$$

$$\hat{\mathrm{y}}_k = \hat{\mathbf{C}}\hat{\mathrm{e}}_k + \hat{\mathbf{D}}\mathrm{u}_k \tag{5.37}$$

and it seeks to minimize the covariance $\mathbf{P} = E\{\tilde{\mathrm{e}}^T\tilde{\mathrm{e}}\}$ of the one-step-ahead estimation error $\tilde{\mathrm{e}}$

$$\underbrace{(\mathrm{e}_{k+1} - \hat{\mathrm{e}}_{k+1})}_{\tilde{\mathrm{e}}_{k+1}} = (\hat{\mathbf{A}} - \mathbf{L}\hat{\mathbf{C}})\tilde{\mathrm{e}}_k + \mathrm{w}_{\mathrm{e}k} + \mathbf{L}\mathrm{w}_{\mathrm{y}_k} \tag{5.38}$$

The predictive Kalman gain is given by

$$\mathbf{L} = \hat{\mathbf{A}}\mathbf{P}\hat{\mathbf{C}}^T(\hat{\mathbf{C}}\mathbf{P}\hat{\mathbf{C}}^T + \mathbf{R_y}) \tag{5.39}$$

where the covariance of the estimation error is given by the discrete-time algebraic Riccati equation

$$\mathbf{P} = \mathbf{R_e} + \hat{\mathbf{A}}\mathbf{P}\hat{\mathbf{A}}^T - \hat{\mathbf{A}}\mathbf{P}\hat{\mathbf{C}}^T(\hat{\mathbf{C}}\mathbf{P}\hat{\mathbf{C}}^T + \mathbf{R_y})^{-1}\hat{\mathbf{C}}\mathbf{P}\hat{\mathbf{A}}^T \tag{5.40}$$

The design parameters of the Kalman filter is the state, disturbance and output covariance. Ideally information about state and output noise variance should be available but the noise variance of the unmeasured disturbances is not available. So another rule of thumb is given instead. The values are relative and if information about real state and output variance is available then the disturbance variances should be dimensioned according to that.

- **State variance** should be small.
- **State disturbance variance** should be small to smooth out the disturbance compensation.
- **Output disturbance variance** should be larger than the state variance if the state measurement are to trusted more than the linearized output functions dependent of the states and inputs.
- **Output variance** should be small.

The tuning rules presented here might be not make sense if the goal of the estimator was to estimate a constant step disturbance. But that is not the purpose of this estimator. The purpose is to compensate plant/model mismatch where the plant might be nonlinear.

The influence of the estimated disturbances can be rejected by taking them into account in the target calculation.

$$\mathrm{x}_{\mathrm{t}k} = \mathbf{A}\mathrm{x}_{\mathrm{t}k} + \mathbf{B_d}\hat{\mathrm{d}}_k + \mathbf{B}\mathrm{u}_{\mathrm{t}k} \tag{5.41}$$

$$\mathrm{y}_{\mathrm{t}k} = \mathbf{C}\mathrm{x}_{\mathrm{t}k} + \mathbf{C_p}\hat{\mathrm{p}}_k + \mathbf{D}\mathrm{u}_{\mathrm{t}k} \tag{5.42}$$

$$\mathrm{r}_k = \mathrm{y}_{\mathrm{r}k} = \mathbf{H}\mathrm{y}_{\mathrm{t}k} \tag{5.43}$$

$$\begin{bmatrix} \mathbf{A}-\mathbf{I} & \mathbf{B} \\ \mathbf{HC} & \mathbf{HD} \end{bmatrix} \begin{pmatrix} \mathrm{x}_{\mathrm{t}k} \\ \mathrm{u}_{\mathrm{t}k} \end{pmatrix} = \begin{pmatrix} -\mathbf{B_d}\hat{\mathrm{d}}_k \\ \mathrm{r}_k - \mathbf{HC_p}\hat{\mathrm{p}}_k \end{pmatrix} \tag{5.44}$$

The unmeasured integrating disturbances are not controllable and there is no point in including them in the dynamic optimization problem. Instead the original controllable system should be object of optimization and the disturbances should only be used to determine an offset-free equilibrium.

Giving the estimated and target optimization variable

$$\hat{\mathrm{z}}_k = \mathbf{E}\hat{\mathrm{x}}_k + \mathbf{E_p}\hat{\mathrm{p}}_k + \mathbf{F}\hat{\mathrm{u}}_k \tag{5.45}$$

$$\mathrm{z}_{\mathrm{t}k} = \mathbf{E}\mathrm{x}_{\mathrm{t}k} + \mathbf{E_p}\hat{\mathrm{p}}_k + \mathbf{F}\mathrm{u}_{\mathrm{t}k} \tag{5.46}$$

The origin shifted variables are redefined as the difference between the estimates and the targets

$$\underbrace{(\hat{\mathrm{z}}_k - \mathrm{z}_{\mathrm{t}k})}_{\mathrm{z}_{\mathrm{s}k}} = \mathbf{E}\underbrace{(\hat{\mathrm{x}}_k - \mathrm{x}_{\mathrm{t}k})}_{\mathrm{x}_{\mathrm{s}k}} + \mathbf{F}\underbrace{(\hat{\mathrm{u}}_k - \mathrm{u}_{\mathrm{t}k})}_{\mathrm{u}_{\mathrm{s}k}} \tag{5.47}$$

5.3 Illustrative example

An illustrative example is given by a Matlab simulation in region II. The controller is the full sensor K_{II} controller given in chapter 8. The estimator is tuned with 4 different tuning sets:

	r_x	r_d	r_p	r_y
s_1	1e-6	0	1e6	1e-6
s_2	1e-6	0	0	1e-6
s_3	1e-6	1e-6	0	1e-6
s_4	1e-6	1e-6	1e6	1e-6

Table 5.1: Estimator tuning sets

The figures Fig. (5.4), Fig. (5.5) and Fig. (5.6) indicates that all of the tuning sets behave similar, on the surface at least. Fig. (5.4) also shows that C_P is maximized quickly after the step and the reference control law eq. (4.12) that governs K_{II} works as intended.

A closer look at the generator torque velocity Fig. (5.7) shows that s_3 and s_4 with low weights on the state disturbance estimations gives a slower/smoother response unlike the dead beat response of s_1 and s_2. This is also reflected in Fig. (5.8(b)) where the disturbance corresponding to the generator speed reacts with a dead beat and smooth behavior for the same combination of tuning sets.

Fig. (5.9) shows that the sets s_1 and s_4 have the smallest overall estimation error on the selected variables. This is because the output disturbance weights are high and the output disturbance estimations are thus allows to react fast to changes.

Overall s_4 is chosen as the best suited tuning set since is gives good smooth estimated which avoids dead beat control but is still able to track output disturbances which is important in chapter 7 in order to respect output constraints.

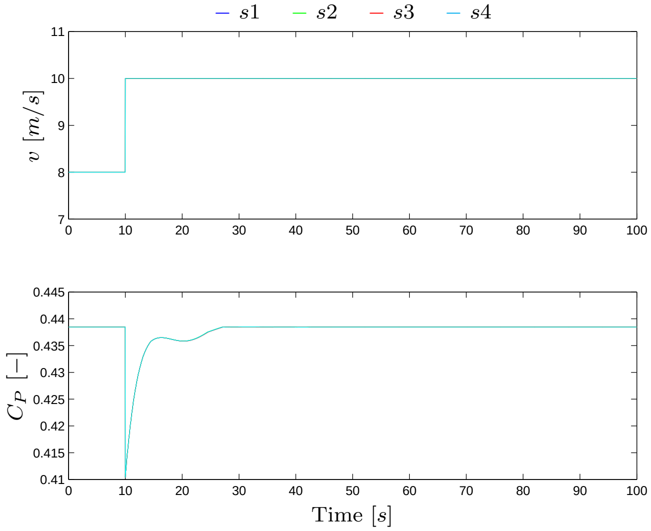


Figure 5.4: Wind speed and power coefficient

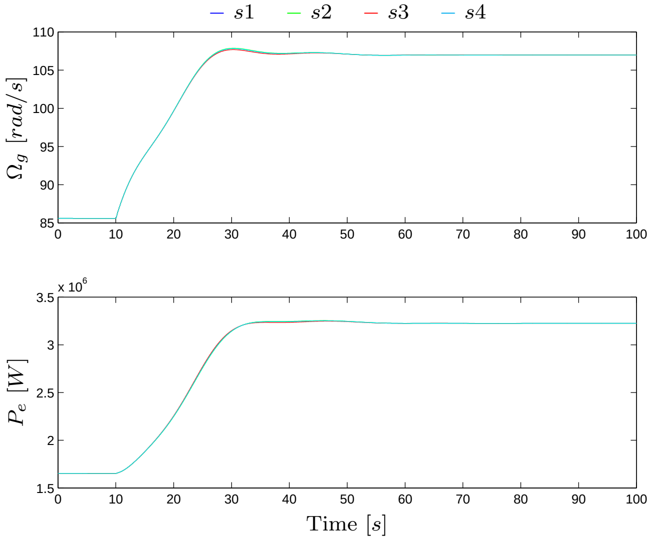


Figure 5.5: Primary controlled variables

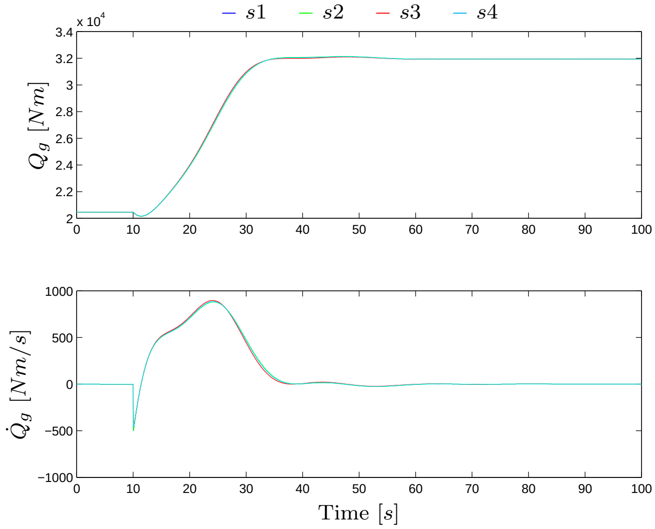


Figure 5.6: Generator torque and derivative

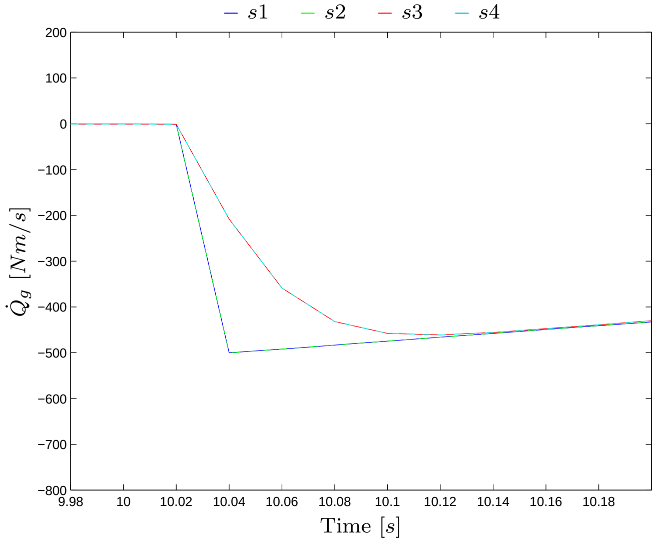
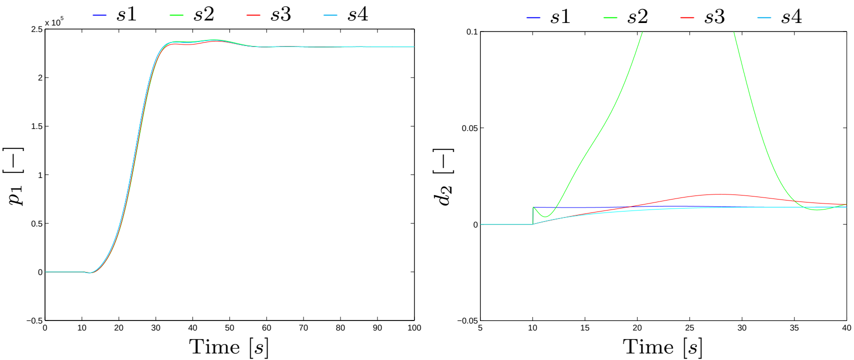


Figure 5.7: Zoom on generator torque velocity



(a) Generator power output disturbance (b) Generator speed state disturbance

Figure 5.8: Disturbances

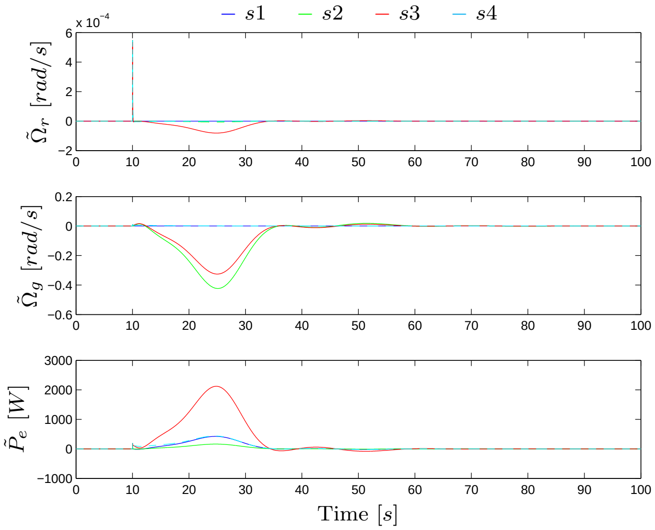


Figure 5.9: Estimation errors

Chapter 6

Model Predictive Control

In this chapter the concept of model predictive control (MPC) is presented. MPC has traditionally found is application at the strategic level of plant control (see chapter 4) because of its heavy computing load. But with the ongoing development of powerful computers MPC can be implemented at the lower levels of control which has faster time constants.

A model is either obtained by physical equations or by system responses from e.g. step inputs. Known respectively as white- and black box modeling. The model can also be a combination of the two, which naturally is called gray box modeling. The complexity or order of the model is usually lower than the actual plant to lighten the computational burden. Only the significant states of the plant are modeled.

The behavior of the model is predicted based on past measurements and computed future inputs.

6.1 Receding Prediction Horizon

In discrete time the receding prediction horizon constitutes the predictions of the system for the next N samples, in continuous time the prediction horizon have a length T seconds. Only the discrete time case will be dealt with in this project. The predictions are based on the model and current state properties as well as any control inputs from the controller. In a time discrete and open-loop paradigm the predictions would be given by the following equations

$$\hat{\mathrm{x}}_{k+i+1|k} = \mathrm{f}(\hat{\mathrm{x}}_{k+i|k}, \mathrm{u}_{k+i}), \qquad 0 \le i \le N \tag{6.1}$$

$$\hat{\mathrm{y}}_{k+i|k} = \mathrm{g}(\hat{\mathrm{x}}_{k+i|k}, \mathrm{u}_{k+i}), \qquad 0 \le i \le N \tag{6.2}$$

where

$$\hat{\mathrm{x}}_{k|k} = \mathrm{x}_k \tag{6.3}$$

The actual and predicted progress of a state x can be seen in Fig. (6.1). The state x have until time $t = k$ evolved by the influences of the system dynamics and any inputs. The future predictions of the state should be controlled by a sequence of control signals to a desired set-point, the computation of this sequence of control signals is of course the objective of Model Predictive Control.

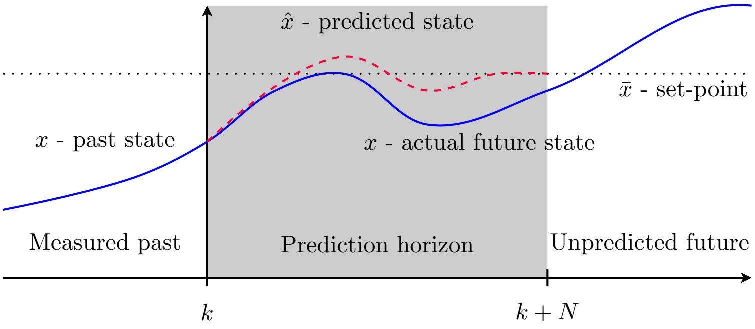


Figure 6.1: Prediction horizon

The actuation of that particular sequence of control signals might reveal that the predicted states and the actual states in the future would differ. This could be due to model/plant mismatch, unknown disturbances etc. With an adequate model at least the initial predictions should correspond to the actual state progress. The solution to this problem is to only actuate the first control signal of the sequence and then at the next sample calculate a new sequence of control signals based on the state at that time, hence using a receding horizon. There exist many variations of this principle, e.g. the prediction horizon is divided into a controlled horizon and an uncontrolled horizon and if any time delays exist in the system these can also be incorporated into the framework.

The predictions could be calculated using dynamic (Belman) programming as in chapter 5 or the prediction problem can be formulated to a static problem where the iterative predictions are gathered in one prediction vector denoted with a $\underset{\rightarrow}{\ }$. For a linear system the open-loop predictions can be written as

$$\hat{\mathrm{x}}_{k+1|k} = \mathbf{A}\mathrm{x}_k + \mathbf{B}\mathrm{u}_k \tag{6.4}$$

$$\hat{\mathrm{x}}_{k+2|k} = \mathbf{A}\hat{\mathrm{x}}_{k+1|k} + \mathbf{B}\mathrm{u}_{k+1} = \mathbf{A}^2\mathrm{x}_k + \mathbf{AB}\mathrm{u}_k + \mathbf{B}\mathrm{u}_{k+1} \tag{6.5}$$

$$\vdots \tag{6.6}$$

$$\hat{\mathrm{x}}_{k+N|k} = \mathbf{A}\hat{\mathrm{x}}_{k+N-1|k} + \mathbf{B}\mathrm{u}_{k+N-1} = \mathbf{A}^N\mathrm{x}_k + \mathbf{A}^{N-1}\mathbf{B}\mathrm{u}_k + \ldots + \mathbf{B}\mathrm{u}_{k+N-1} \tag{6.7}$$

where the predicted states in the prediction horizon can be written in matrix form

$$\underbrace{\begin{pmatrix} \hat{\mathrm{x}}_{k|k} \\ \hat{\mathrm{x}}_{k+1|k} \\ \hat{\mathrm{x}}_{k+2|k} \\ \vdots \\ \hat{\mathrm{x}}_{k+N|k} \end{pmatrix}}_{\underset{\rightarrow}{\mathrm{x}}_k} = \underbrace{\begin{bmatrix} \mathbf{I} \\ \mathbf{A} \\ \mathbf{A}^2 \\ \vdots \\ \mathbf{A}^N \end{bmatrix}}_{\underset{\rightarrow}{\mathbf{A}}} \mathrm{x}_k + \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{B} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{AB} & \mathbf{B} & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{N-1}\mathbf{B} & \mathbf{A}^{N-2}\mathbf{B} & \ldots & \mathbf{0} \end{bmatrix}}_{\underset{\rightarrow}{\mathbf{B}}} \underbrace{\begin{pmatrix} \mathrm{u}_k \\ \mathrm{u}_{k+1} \\ \mathrm{u}_{k+2} \\ \vdots \\ \mathrm{u}_{k+N} \end{pmatrix}}_{\underset{\rightarrow}{\mathrm{u}}_k} \tag{6.8}$$

If the system has unstable modes then the computations of the open-loop predictions might give cause to numerical problems for large prediction horizons since the stable modes moves toward origo and the unstable modes move toward infinity, giving a ill-conditioned problem. This can be remedied by the introduction of the closed-loop paradigm where the progress of states is computed under the assumption that they are controlled by a stabilizing feedback controller and any desired deviation in the control signal is given by the addition of control deviation signal v.

$$\hat{\mathrm{u}}_{k+i|k} = -\mathbf{K}\hat{\mathrm{x}}_{k+i|k} + \mathrm{v}_{k+i} \tag{6.9}$$

$$\hat{\mathrm{x}}_{k+i+1|k} = \mathbf{A}\hat{\mathrm{x}}_{k+i|k} + \mathbf{B}\hat{\mathrm{u}}_{k+i|k} \tag{6.10}$$

giving the closed-loop description

$$\hat{\mathrm{x}}_{k+i+1|k} = \mathbf{\Phi}\hat{\mathrm{x}}_{k+i|k} + \mathbf{B}\mathrm{v}_{k+i}; \quad \mathbf{\Phi} = (\mathbf{A} - \mathbf{BK}) \tag{6.11}$$

and the closed-loop paradigm state predictions

$$\underset{\rightarrow}{\mathrm{x}}_k = \underbrace{\begin{bmatrix} \mathbf{I} \\ \mathbf{\Phi} \\ \mathbf{\Phi}^2 \\ \vdots \\ \mathbf{\Phi}^N \end{bmatrix}}_{\underset{\rightarrow}{\mathbf{\Phi}}} \mathrm{x}_k + \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{B} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{\Phi B} & \mathbf{B} & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{\Phi}^{N-1}\mathbf{B} & \mathbf{\Phi}^{N-2}\mathbf{B} & \ldots & \mathbf{0} \end{bmatrix}}_{\underset{\rightarrow}{\mathbf{\Gamma}}} \underbrace{\begin{pmatrix} \mathrm{v}_k \\ \mathrm{v}_{k+1} \\ \mathrm{v}_{k+2} \\ \vdots \\ \mathrm{v}_{k+N} \end{pmatrix}}_{\underset{\rightarrow}{\mathrm{v}}_k} \tag{6.12}$$

and the closed-loop paradigm control signal predictions

$$\begin{aligned}\underrightarrow{\mathrm{u}}_k = -\mathbf{K}^{\mathrm{diag}}\underrightarrow{\mathrm{x}}_k + \underrightarrow{\mathrm{v}}_k = -\mathbf{K}^{\mathrm{diag}}[\underrightarrow{\mathbf{\Phi}}\mathrm{x}_k + \underrightarrow{\mathbf{\Gamma}}\,\underrightarrow{\mathrm{v}}_k] + \underrightarrow{\mathrm{v}}_k = \\ -\mathbf{K}^{\mathrm{diag}}\underrightarrow{\mathbf{\Phi}}\mathrm{x}_k - [\mathbf{K}^{\mathrm{diag}}\underrightarrow{\mathbf{\Gamma}} - \mathbf{I}]\underrightarrow{\mathrm{v}}_k \qquad (6.13)\end{aligned}$$

the closed-loop paradigm optimization variable predictions are given by the state (6.12) and control (6.13) predictions

$$\begin{aligned}\underrightarrow{\mathrm{z}_1}_k = \mathbf{E}^{\mathrm{diag}}\underrightarrow{\mathrm{x}}_k + \mathbf{F}^{\mathrm{diag}}\underrightarrow{\mathrm{u}}_k = \\ \mathbf{E}^{\mathrm{diag}}[\underrightarrow{\mathbf{\Phi}}\mathrm{x}_k + \underrightarrow{\mathbf{\Gamma}}\,\underrightarrow{\mathrm{v}}_k] + \mathbf{F}^{\mathrm{diag}}[-\mathbf{K}^{\mathrm{diag}}\underrightarrow{\mathbf{\Phi}}\mathrm{x}_k - (\mathbf{K}^{\mathrm{diag}}\underrightarrow{\mathbf{\Gamma}} - \mathbf{I})\underrightarrow{\mathrm{v}}_k] = \\ \underbrace{[\mathbf{E}^{\mathrm{diag}}\underrightarrow{\mathbf{\Phi}} + \mathbf{F}^{\mathrm{diag}}\mathbf{K}^{\mathrm{diag}}\underrightarrow{\mathbf{\Phi}}]}_{\underrightarrow{\mathbf{E_1}}}\mathrm{x}_k + \underbrace{[\mathbf{E}^{\mathrm{diag}}\underrightarrow{\mathbf{\Gamma}} - \mathbf{F}^{\mathrm{diag}}[\mathbf{K}^{\mathrm{diag}}\underrightarrow{\mathbf{\Gamma}} - \mathbf{I}]]}_{\underrightarrow{\mathbf{F_1}}}\underrightarrow{\mathrm{v}}_k \qquad (6.14)\end{aligned}$$

6.2 Constrained Predictive Control

The static formulation of the prediction horizon leads to an alternative to dynamic programming to solve the control problem given by eq. (5.1). To simpilfy notation the following definition is introduced $\underrightarrow{\mathbf{W}} \equiv \mathbf{W}^{\mathrm{diag}}$.

$$\begin{aligned}J_1 = \sum_{i=k}^{k+N}\|\mathrm{z}_i\|^2_{\mathrm{W}} = \|\underrightarrow{\mathrm{z}}_k\|^2_{\underrightarrow{\mathbf{W}}} = \underrightarrow{\mathrm{z}_1}_k^T\underrightarrow{\mathbf{W}}\underrightarrow{\mathrm{z}_1}_k = \\ (\mathrm{x}_k^T\underrightarrow{\mathbf{E_1}}^T + \underrightarrow{\mathrm{u}}_k^T\underrightarrow{\mathbf{F_1}}^T)\underrightarrow{\mathbf{W}}(\underrightarrow{\mathbf{E_1}}\mathrm{x}_k + \underrightarrow{\mathbf{F_1}}\underrightarrow{\mathrm{v}}_k) = \\ \mathrm{x}_k^T\underrightarrow{\mathbf{E_1}}^T\underrightarrow{\mathbf{W}}\underrightarrow{\mathbf{E_1}}\mathrm{x}_k + 2\mathrm{x}_k^T\underbrace{\underrightarrow{\mathbf{E_1}}^T\underrightarrow{\mathbf{W}}\underrightarrow{\mathbf{F_1}}}_{\underrightarrow{\mathrm{q_1}}^T}\underrightarrow{\mathrm{v}}_k + \underrightarrow{\mathrm{v}}_k^T\underbrace{\underrightarrow{\mathbf{F_1}}^T\underrightarrow{\mathbf{W}}\underrightarrow{\mathbf{F_1}}}_{\underrightarrow{\mathbf{Q_1}}}\underrightarrow{\mathrm{v}}_k \qquad (6.15)\end{aligned}$$

Since the prediction horizon is given by the closed-loop paradigm the cost function variable is the control signal deviation variable $\underrightarrow{\mathrm{v}}_k$ instead of the control signal $\underrightarrow{\mathrm{u}}_k$ itself. In an open-loop paradigm the cost function variable would be $\underrightarrow{\mathrm{u}}_k$. In the unconstrained case the optimal deviation control sequence $\underrightarrow{\mathrm{v}}_k$ is of course zero. In the open-loop paradigm the optimal control sequence would be determined by differentiation of the optimization problem with regards to the control sequence.

6.2.1 Constraints and feasibility

So far all control problems in the project has been assumed unconstrained. But as described in chapter 2 the actuators have constraints on their states and outputs. Constraints can be divided into two categories of priority: Soft and hard constraints. Soft constraints are constraints which can be violated but shouldn't. A soft constraint could be a limit of temperature, pressure etc. which would cause an greatly increased damage to the system if crossed. Hard constraints are constraints which can't be crossed under any circumstances. A hard constraint could be the mechanical limits of the pitch of a rotor blade on a wind turbine. The constraints of the system can be expressed as limits on optimization variable

$$\mathrm{z}_{min} \preceq \mathrm{z}_i \preceq \mathrm{z}_{max}; \quad \forall i \in \mathbb{N} \tag{6.16}$$

which can be formulated as single linear inequality

$$\begin{bmatrix} \mathbf{I} \\ -\mathbf{I} \end{bmatrix} \mathrm{z}_i \preceq \begin{pmatrix} \mathrm{z}_{max} \\ -\mathrm{z}_{min} \end{pmatrix}; \quad \forall i \in \mathbb{N} \tag{6.17}$$

only a subset of the optimization variable might be constrained and a more compact formulation can be used which only contains actual constraints

$$\mathbf{M}\mathrm{z}_i \preceq \mathrm{c}; \quad \forall i \in \mathbb{N} \tag{6.18}$$

the linear inequality given in eq. (6.18) can be represented as a polyhedral set

$$\mathrm{z}_i \in \mathscr{Z}; \quad \mathscr{Z} = \{\mathrm{z}|\mathbf{M}\mathrm{z} \preceq \mathrm{c}\}; \quad \forall i \in \mathbb{N} \tag{6.19}$$

The optimization variable can expressed in terms of the deviation control signals and the states and the constraints will only considered within the prediction horizon

$$\mathbf{M}[\mathbf{E}\mathrm{x}_i - \mathbf{F}\mathbf{K}\mathrm{x}_i + \mathrm{v}_i] \preceq \mathrm{c}; \quad k \leq i \leq N \tag{6.20}$$

The static prediction formulation of constraints and optimization variable enables the use of quadratic programming (QP) algorithms to solve the constrained convex quadratic optimization problem. For an elaboration on constrained optimization see appendix C

$$\underrightarrow{\mathrm{v}}_k = \arg\min_{\underrightarrow{\mathrm{v}}_k} \frac{1}{2} \underrightarrow{\mathrm{v}}_k^T \underrightarrow{\mathbf{Q_1}} \underrightarrow{\mathrm{v}}_k + (\mathrm{x}_k^T \underrightarrow{\mathrm{q}_1}^T) \underrightarrow{\mathrm{v}}_k \tag{6.21a}$$

subject to

$$[\mathbf{M}^{\mathrm{diag}} \underrightarrow{\mathbf{F_1}}] \underrightarrow{\mathrm{v}}_k \preceq [\mathrm{c}^{\mathrm{col}} - \mathbf{M}^{\mathrm{diag}} \underrightarrow{\mathbf{E_1}} \mathrm{x}_k] \tag{6.21b}$$

The first control signal of the computed sequence can then be actuated

$$\mathrm{u}_k = -\mathbf{K}\mathrm{x}_k + \mathrm{v}_k \tag{6.22}$$

Sometimes the problem can be so strict that it is infeasible. This means that it can't be solved within the given constraints. This could be due to the fact the constraints are in fact violated at the beginning of the problem or that some constraints are contradictive e.g. $1 \leq z \leq -1$. If optimization of the cost function is infeasible due to overly stringent constraints one method of handling this is to introduce a slack variable which try to minimize the violation of the constraint but removes the violated constraint from the constrains. This won't be discussed further since the constraints in this project are assumed to be hard constraints.

6.2.2 Dual Mode Horizon and nominal stability

The finite horizon problem given by eq. (6.21) has a major flaw: It doesn't consider the systems behavior after the prediction horizon and could inadvertently be sending the system to a undesired state. This can be remedied by dividing the problem into a dual mode horizon problem (more modes could be added to suit a specific problem)

$$J = J_1 + J_2 + \dots \tag{6.23}$$

where each cost function covers a specific part of the prediction horizon and each cost function can be subjected to different constraints.

In a dual mode setup the first part of the horizon is called the control horizon is subjected to constraints and is similar to the finite horizon problem in eq. (6.21). The second mode is the prediction mode where the system is assumed to inside an invariant and unconstrained set. This could be obtained by adding a dead-beat constraint to the final control signal, i.e. $\mathrm{u}_{k+N} = 0$. But that requires a long control horizon and result in a very aggressive control.

$$J = \underbrace{\sum_{i=k}^{k+N} \|\mathrm{z}_i\|^2_{\mathbf{W}}}_{J_1} + \underbrace{\sum_{i=k+N+1}^{N_2} \|\mathrm{z}_i\|^2_{\mathbf{W}}}_{J_2} \tag{6.24}$$

A softer approach is to append the unconstrained infinite horizon problem from eq. (5.14a) which draws the optimization towards the unconstrained invariant set the equilibrium point

$$J = \underbrace{\sum_{i=k}^{k+N} \|\mathrm{z}_i\|^2_{\mathbf{W}}}_{J_1} + \underbrace{\mathrm{x}^T_{k+N+1}\mathbf{S}\mathrm{x}_{k+N+1};}_{J_2} \quad N_2 \to \infty \tag{6.25}$$

this of means that the prediction horizon is assumed unconstrained. That assumption is not entirely bulletproof which will be illustrated in the example at the end of this chapter.

The static formulation of the constrained control mode J_1 is already given in eq. (6.21). The static formulation of the prediction mode is

$$\begin{aligned} J_2 = \mathrm{x}^T_{k+N+1}\mathbf{S}\mathrm{x}_{k+N+1} = \hat{\mathrm{x}}^T_{k+N+1|k}\mathbf{S}\hat{\mathrm{x}}_{k+N+1|k} = \\ (\mathrm{x}_k^T\boldsymbol{\Phi}_\mathbf{2}{}^T + \underrightarrow{\mathrm{v}}_k^T\boldsymbol{\Gamma}_\mathbf{2}{}^T)\mathbf{S}(\boldsymbol{\Phi}_\mathbf{2}\mathrm{x}_k + \boldsymbol{\Gamma}_\mathbf{2}\underrightarrow{\mathrm{v}}_k) = \\ \mathrm{x}_k^T\boldsymbol{\Phi}_\mathbf{2}{}^T\mathbf{S}\boldsymbol{\Phi}_\mathbf{2}\mathrm{x}_k + 2\mathrm{x}_k^T\underbrace{\boldsymbol{\Phi}_\mathbf{2}{}^T\mathbf{S}\boldsymbol{\Gamma}_\mathbf{2}}_{\underrightarrow{\mathrm{q}_2}^T}\underrightarrow{\mathrm{v}}_k + \underrightarrow{\mathrm{v}}_k^T\underbrace{\boldsymbol{\Gamma}_\mathbf{2}{}^T\mathbf{S}\boldsymbol{\Gamma}_\mathbf{2}}_{\underrightarrow{\mathbf{Q}_\mathbf{2}}^T}\underrightarrow{\mathrm{v}}_k \end{aligned} \tag{6.26}$$

where

$$\hat{\mathrm{x}}_{k+N+1|k} = \underbrace{\boldsymbol{\Phi}^{N+1}}_{\boldsymbol{\Phi}_\mathbf{2}}\mathrm{x}_k + \underbrace{\begin{bmatrix}\boldsymbol{\Phi}^N\mathbf{B} & \boldsymbol{\Phi}^{N+1}\mathbf{B} & \dots & \mathbf{B}\end{bmatrix}}_{\boldsymbol{\Gamma}_\mathbf{2}}\underrightarrow{\mathrm{v}}_k \tag{6.27}$$

$$\hat{\mathrm{z}}_{k+N+1|k} = \underbrace{\mathbf{E}\boldsymbol{\Phi}_\mathbf{2}}_{\mathbf{E}_\mathbf{2}}\mathrm{x}_k + \underbrace{\mathbf{F}\boldsymbol{\Gamma}_\mathbf{2}}_{\mathbf{F}_\mathbf{2}}\underrightarrow{\mathrm{v}}_k \tag{6.28}$$

The dual mode prediction horizon optimization variable is then

$$\underbrace{\begin{pmatrix}\underrightarrow{\mathrm{z}_1}_k \\ \hat{\mathrm{z}}_{k+N+1|k}\end{pmatrix}}_{\underrightarrow{\mathrm{z}}_k} = \underbrace{\begin{bmatrix}\underrightarrow{\mathbf{E}_\mathbf{1}} \\ \mathbf{E}_\mathbf{2}\end{bmatrix}}_{\underrightarrow{\mathbf{E}}}\mathrm{x}_k + \underbrace{\begin{bmatrix}\underrightarrow{\mathbf{F}_\mathbf{1}} \\ \mathbf{F}_\mathbf{2}\end{bmatrix}}_{\underrightarrow{\mathbf{F}}}\mathrm{v}_k \tag{6.29}$$

The constrained infinite horizon optimization problem can now be written

$$\underrightarrow{\mathrm{v}}_k = \arg\min_{\underrightarrow{\mathrm{v}}_k}\frac{1}{2}\underrightarrow{\mathrm{v}}_k^T\underbrace{[\underrightarrow{\mathbf{Q}_\mathbf{1}} + \underrightarrow{\mathbf{Q}_\mathbf{2}}]}_{\underrightarrow{\mathbf{Q}}}\underrightarrow{\mathrm{v}}_k + (\mathrm{x}_k^T\underbrace{(\underrightarrow{\mathrm{q}_1}^T + \underrightarrow{\mathrm{q}_2}^T)}_{\underrightarrow{\mathrm{q}}^T})\underrightarrow{\mathrm{v}}_k \tag{6.30a}$$

subject to

$$\begin{bmatrix}\mathbf{M}^{\mathrm{diag}}\underrightarrow{\mathbf{F}}\end{bmatrix}\underrightarrow{\mathrm{v}}_k \preceq \begin{bmatrix}\mathrm{c}^{\mathrm{col}} - \mathbf{M}^{\mathrm{diag}}\underrightarrow{\mathbf{E}}\mathrm{x}_k\end{bmatrix} \tag{6.30b}$$

Unfortunately the problem is now constrained and $\mathcal{X}$ can't be infinitely large. A terminal invariant set can be computed via linear matrix inequalities (LMI). A sanity check can be performed after the calculation of the QP and if the final state is not within the terminal invariant set then a new problem should be formulated with a longer horizon and so on until the final predicted state is within the terminal invariant set.

$$\mathrm{x}_{k+N+1} \in \mathcal{X}_{terminal} \tag{6.31}$$

That is however not implemented is this project. The prediction horizon is simply expected to be long enough.

The terminal invariant set can be any type of invariant set suited for the problem usually a polyhedral set will be the appropriate choice. A conservative candidate of the terminal invariant set is an ellipsoidal invariant set based on **S** and with a outer rim determined off line via LMI's. If the cost of the second mode, i.e. J_2, is larger than the outer rim of the maximal allowed invariant ellipsoidal set then it can't be guaranteed that constraints won't be violated in the future.

6.3 Illustrative example

This example will illustrate the behavior of a plant in the autonomous mode 2 of the dual mode control horizon. As an example the linear pitch actuator from subsection 2.2.5 is considered, this is suitable as the state vector of the pitch actuator is 2-dimensional and thus presentable on a 2-dimensional paper. The optimization variable is

$$\underbrace{\begin{pmatrix}\theta\\ \dot{\theta}\\ \ddot{\theta}\end{pmatrix}}_{\mathrm{z}_k} = \underbrace{\begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\\ -\omega_n^2 & -2\zeta\omega_n\end{bmatrix}}_{\mathbf{E}} \underbrace{\begin{pmatrix}\theta\\ \dot{\theta}\end{pmatrix}}_{\mathrm{x}_k} + \underbrace{\begin{bmatrix}\mathbf{0}\\ -\omega_n^2\end{bmatrix}}_{\mathbf{F}} \underbrace{\begin{pmatrix}\theta_{ref}\end{pmatrix}}_{\mathrm{u}_k} \tag{6.32}$$

The optimization weights are

$$\mathbf{W} = \mathrm{diag}\left(\begin{pmatrix}\frac{1}{10} & \frac{1}{8} & \frac{1}{15}\end{pmatrix}^2\right) \tag{6.33}$$

The gain **K** and hessian **S** are calculated with eq. (5.14b) and eq. (5.14c). Closing the loop gives

$$\mathrm{x}_{k+1} = \mathbf{\Phi}\mathrm{x}_k; \quad \mathbf{\Phi} = \mathbf{A} - \mathbf{BK} \tag{6.34}$$

$$\mathrm{z}_k = \mathbf{\Psi}\mathrm{x}_k; \quad \mathbf{\Psi} = \mathbf{E} - \mathbf{FK} \tag{6.35}$$

The polyhedral set $\mathcal{Z}$ given in eq. (6.19) can be expressed in terms of the state variable since the control law is fixed and a closed-loop description can be formed.

$$\mathrm{x}_i \in \mathcal{X}; \quad \mathcal{X} = \{\mathrm{x}|\mathbf{M\Psi}\mathrm{x} \preceq \mathrm{c}\} \tag{6.36}$$

The boundary of $\mathcal{X}$ is defined by the innermost vertices of the inequality.

$$\mathrm{c}_{\{i\}} = [\mathbf{M\Psi}]_{\{i,1\}}\ x_1 + [\mathbf{M\Psi}]_{\{i,2\}}\ x_2; \quad 1 \leq i \leq n_c \tag{6.37}$$

A conservative invariant ellipsoidal set can dimensioned to be inside the constraints. A good candidate for the invariant set is the invariant set given by eq. (5.20)

$$\mathcal{X}_e = \{\mathrm{x}|\mathrm{x}^T\mathbf{S}\mathrm{x} \leq c\} \tag{6.38}$$

due to constraints the outer rim c can no longer be infinitely large and can be determined by a rather complicated linear matrix inequality (LMI) shown in (Rossiter, 2003). To keep things simple, c has instead been adjusted manually by trial and error and determined to be $c \approx 4.01$.

The maximal admissible set (MAS) is the set largest possible terminal invariant set that ensures the autonomous system stays within the constraints for all future incidents. This implies

$$\mathrm{x}_k \in \mathcal{X} \Rightarrow \mathrm{x}_{k+1} \in \mathcal{X} \Rightarrow \mathrm{x}_{k+2} \in \mathcal{X} \Rightarrow \ldots \tag{6.39}$$

which can be written as a polyhedral set

$$\mathcal{X}_p = \left\{ \mathrm{x} \left| \begin{bmatrix} \mathbf{M\Psi\Phi} \\ \mathbf{M\Psi\Phi}^2 \\ \vdots \\ \mathbf{M\Psi\Phi}^n \end{bmatrix} \mathrm{x} \preceq \mathrm{c}^{\mathrm{col}} \right. \right\} \tag{6.40}$$

the size of n need not be infinite, an iteration of linear programs (see (Rossiter, 2003)) can determine when the set is sufficient. Any redundant constraints can be removed from the set. Such clever modifications have not been implemented in this project, instead a polyhedral set with $n = 100$ is shown in Fig. (6.2). This gives $n(+1) \times n_c$ (including the original constraints) vertices where not all are the innermost of the set and are hence redundant.

Three different trajectories with three different starting points are shown. The three different starting point illustrate where the prediction horizon is at time $N + 1$ when mode 2 of the dual mode horizon takes over the control. They are all within the actual constraints given by $\mathcal{X}$.

The first trajectory s_1 is within the constraints but not within either of the invariant sets, it is seen on the figures that the trajectory violates constraints. The second trajectory s_2 starts within the conservative invariant ellipsoidal set $\mathcal{X}_e$ and remains inside the set. The third trajectory starts just within the boundary of the MAS $\mathcal{X}_p$ and remains within that set.

	s_1	s_2	s_3
x_0	$(20, -7)$	$(-3, -2)$	$(17.1, -6)$

Table 6.1: Different starting point of autonomous system

The ellipsoidal set $\mathcal{X}_e$ is a simple formulation. But the fact that it is quadratic means that it cannot be implemented directly into the constraints of the quadratic problem. The MAS $\mathcal{X}_p$ is very large and can be added to the constraints of quadratic problem. Neither will however be implemented since the actual implementation of MPC in the project involves disturbance detection and the constraints and origo are therefore not static (see chapter 7). That would require a robust control formulation of the terminal invariant set with uncertainty perturbations included in the set (see (Blanchini, 1999) and (Rossiter, 2003)).

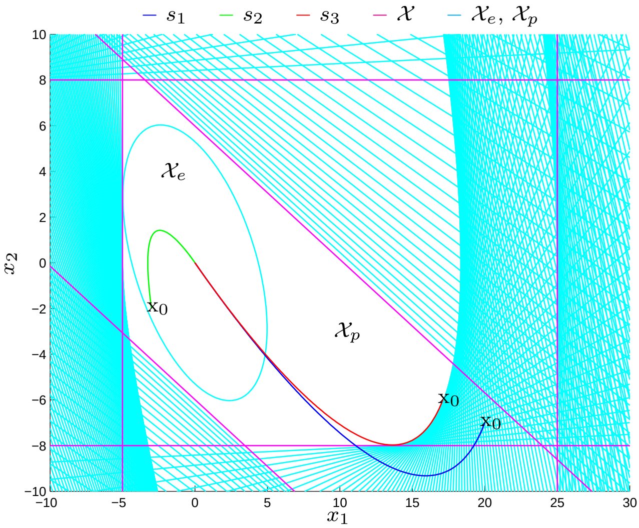


Figure 6.2: Trajectories of autonomous system in the vector field of x.

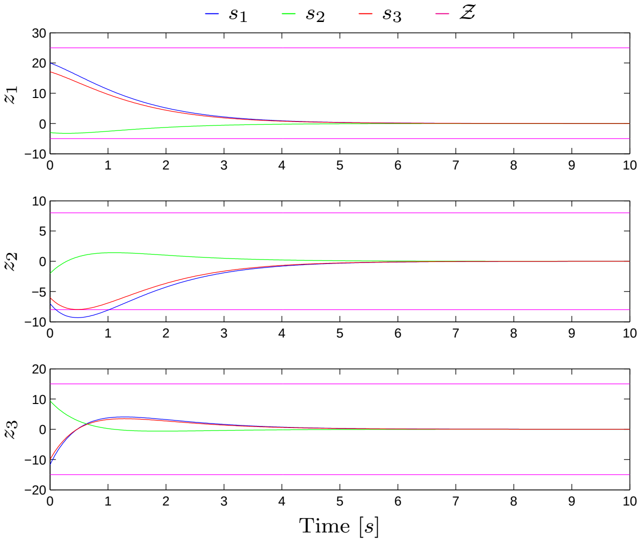


Figure 6.3: Trajectories of autonomous system in the vector field of z.

Chapter 7

Constrained Linear Quadratic Control

In this chapter the disturbance estimation and origin shifting from chapter 5 and the constraint handling from chapter 6 is combined into the constrained linear quadratic controller (CLQ) which is described in (Pannocchia et al., 2005) among many other papers and books.

7.1 Constrained target calculation

The target calculation in eq. (5.44) does not consider violation of constraints which might lead to an unreachable target, it is apparent that in the case of constraint violations in steady state the offset-free guarantee given by the disturbance estimation is no longer valid. The target calculation is bounded by the constraints

$$\mathbf{M}\mathrm{z}_{\mathrm{t}k} \preceq \mathrm{c} \tag{7.1}$$

which can be rewritten to the linear inequality

$$\mathbf{M}\begin{bmatrix}\mathbf{E} & \mathbf{F}\end{bmatrix}\begin{pmatrix}\mathrm{x}_{\mathrm{t}k}\\ \mathrm{u}_{\mathrm{t}k}\end{pmatrix} \preceq \left(\mathrm{c} - \mathbf{M}\mathbf{E}_{\mathbf{p}}\hat{\mathrm{p}}_k\right) \tag{7.2}$$

Instead of the target calculation given eq. (5.44) the difference between the reference and the reference controlled output eq. (5.43) should be minimized

$$\min_{(\mathrm{x}_{\mathrm{t}k},\mathrm{u}_{\mathrm{t}k})} \|\mathrm{r}_k - \mathbf{H}\mathrm{y}_{\mathrm{t}k}\|^2 =$$

$$\begin{pmatrix}\mathrm{x}_{\mathrm{t}k}^T & \mathrm{u}_{\mathrm{t}k}^T\end{pmatrix} \underbrace{\begin{bmatrix}\mathbf{C}^T\mathbf{H}^T\mathbf{H}\mathbf{C} & \mathbf{C}^T\mathbf{H}^T\mathbf{H}\mathbf{D} \\ \mathbf{D}^T\mathbf{H}^T\mathbf{H}\mathbf{C} & \mathbf{D}^T\mathbf{H}^T\mathbf{H}\mathbf{D}\end{bmatrix}}_{\mathbf{Q_t}} \begin{pmatrix}\mathrm{x}_{\mathrm{t}k} \\ \mathrm{u}_{\mathrm{t}k}\end{pmatrix} +$$

$$2\underbrace{\begin{pmatrix}-\mathrm{r}_k^T\mathbf{H}\mathbf{C} + \hat{\mathrm{p}}_k^T\mathbf{C_p}^T\mathbf{H}^T\mathbf{H}\mathbf{C} & -\mathrm{r}_k^T\mathbf{H}\mathbf{D} + \hat{\mathrm{p}}_k^T\mathbf{C_p}^T\mathbf{H}^T\mathbf{H}\mathbf{D}\end{pmatrix}}_{\mathrm{q_t}} \begin{pmatrix}\mathrm{x}_{\mathrm{t}k} \\ \mathrm{u}_{\mathrm{t}k}\end{pmatrix} +$$

$$\mathrm{r}_k^T\mathrm{r}_k - \mathrm{r}_k^T\mathbf{H}\mathbf{C_p}\hat{\mathrm{p}}_k - \hat{\mathrm{p}}_k^T\mathbf{C_p}^T\mathbf{H}^T\mathrm{r}_k + \hat{\mathrm{p}}_k^T\mathbf{C_p}^T\mathbf{H}^T\mathbf{H}\mathbf{C_p}\hat{\mathrm{p}}_k \tag{7.3}$$

giving the constrained quadratic problem

$$(\mathrm{x}_{\mathrm{t}k}, \mathrm{u}_{\mathrm{t}k}) = \arg\min_{(\mathrm{x}_{\mathrm{t}k},\mathrm{u}_{\mathrm{t}k})} \frac{1}{2}\begin{pmatrix}\mathrm{x}_{\mathrm{t}k}^T & \mathrm{u}_{\mathrm{t}k}^T\end{pmatrix} \mathbf{Q_t} \begin{pmatrix}\mathrm{x}_{\mathrm{t}k} \\ \mathrm{u}_{\mathrm{t}k}\end{pmatrix} + \mathrm{q_t}\begin{pmatrix}\mathrm{x}_{\mathrm{t}k} \\ \mathrm{u}_{\mathrm{t}k}\end{pmatrix} \tag{7.4}$$

which is convex if $\mathbf{Q}_t > 0$ and is subject to the steady state requirement in eq. (5.41) and the constraints in eq. (7.2).

7.2 Constrained dynamic optimization

The constrained dynamic optimization is origin shifted as in eq. (5.47)

$$\underrightarrow{\mathrm{z}_{\mathrm{s}}}_k = \underrightarrow{\mathrm{z}}_k - \mathrm{z}_{\mathrm{t}k}^{\mathrm{col}} \tag{7.5}$$

where the control signal is given by

$$\mathrm{u}_k = \mathrm{u}_{\mathrm{t}k} + \mathrm{u}_{\mathrm{s}k}; \quad \mathrm{u}_{\mathrm{s}k} = -\mathbf{K}\underbrace{(\hat{\mathrm{x}}_k - \mathrm{x}_{\mathrm{t}k})}_{\mathrm{x}_{\mathrm{s}k}} + \mathrm{v}_{\mathrm{s}k} \tag{7.6}$$

subject to

$$\mathrm{z}_{min}^{\mathrm{col}} \preceq \underrightarrow{\mathrm{z}_{\mathrm{s}}}_k + \mathrm{z}_{\mathrm{t}k}^{\mathrm{col}} \preceq \mathrm{z}_{max}^{\mathrm{col}} \tag{7.7}$$

The control signal deviation is determined by the quadratic problem seen below where the first incident of the deviation prediction vector is used

$$\underrightarrow{\mathrm{v}_{\mathrm{s}}}_k = \arg\min_{\underrightarrow{\mathrm{v}_{\mathrm{s}}}_k} \frac{1}{2}\underrightarrow{\mathrm{v}_{\mathrm{s}}}_k^T \underrightarrow{\mathbf{Q}}\,\underrightarrow{\mathrm{v}_{\mathrm{s}}}_k + (\mathrm{x}_{\mathrm{s}k}^T\,\underrightarrow{\mathrm{q}}^T)\underrightarrow{\mathrm{v}_{\mathrm{s}}}_k \tag{7.8a}$$

subject to

$$\left[\mathbf{M}^{\mathrm{diag}}\underset{\rightarrow}{\mathbf{F}}\right]\underset{\rightarrow}{\mathrm{v}_{\mathrm{s}}}{}_{k} \preceq \left[\mathrm{c}^{\mathrm{col}} - \mathbf{M}^{\mathrm{diag}}\underset{\rightarrow}{\mathbf{E}}\mathrm{x}_{\mathrm{s}k} - \mathrm{z}_{\mathrm{t}k}^{\mathrm{col}}\right] \tag{7.8b}$$

it can be noted that

$$\mathrm{z}_{\mathrm{t}k}^{\mathrm{col}} = \mathbf{E}^{\mathrm{col}}\mathrm{x}_{\mathrm{t}k} + \mathbf{E}_{\mathbf{p}}{}^{\mathrm{col}}\hat{\mathrm{p}}_{k} + \mathbf{F}^{\mathrm{col}}\mathrm{u}_{\mathrm{t}k} \tag{7.9}$$

it is apparent that the terminal invariant sets described in the previous chapter can't be directly implemented here. Instead robust formulations of such sets should be formulated, but that has not been done in this project.

7.3 Illustrative example

An illustrative example is presented. It is the K_{IV} controller designed in chapter 8 but with a different set of tuning weights for the controller. These weights prioritize generator speed higher than generator power. This is opposite to the tuning of K_{IV}. In the case of a positive wind step in region IV the following occurs depending on the tuning priority

- **High priority generator power** leads to a decrease in generator torque to compensate for the increase in generator speed. This leaves the speed control more or less in the hands of the pitch actuation.
- **High priority generator speed** leads to an increase in generator torque to dampen the generator speed increase. This leads to a generator power increase, which under normal circumstances is unacceptable. This priority is more or less the same as the control objective for K_I and K_{III}.

The prediction horizon length is $N = 50$ in this example whereas the prediction horizon length is only needs to be $N = 20$ in chapter 8 because the wind turbine doesn't operate on the constraint limits as long as in this example.

$$\mathrm{q} = (\text{1e3 1e3 1e-1 1e-3 1e-1 1e-1 1e1 1e-1 1e0 1e2}) \tag{7.10}$$

Fig. (7.2), Fig. (7.3) and Fig. (7.4) shows the results for simulations done in Simulink with a full sensors CLQ controller. The simulations are done both on the mounted and floating wind turbine. It is seen that the constraints are not violated and that the wind turbines and controllers react as expected. The different tower structures give noticeably different responses.

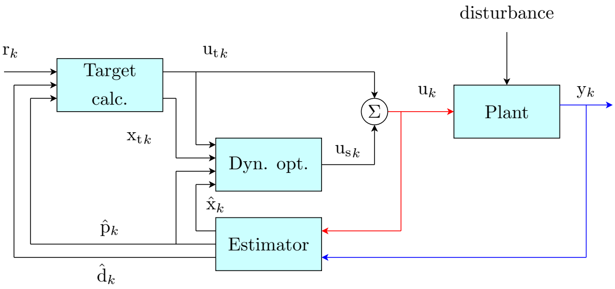


Figure 7.1: Origin shifting constrained LQ control with state and disturbance estimation (CLQ)

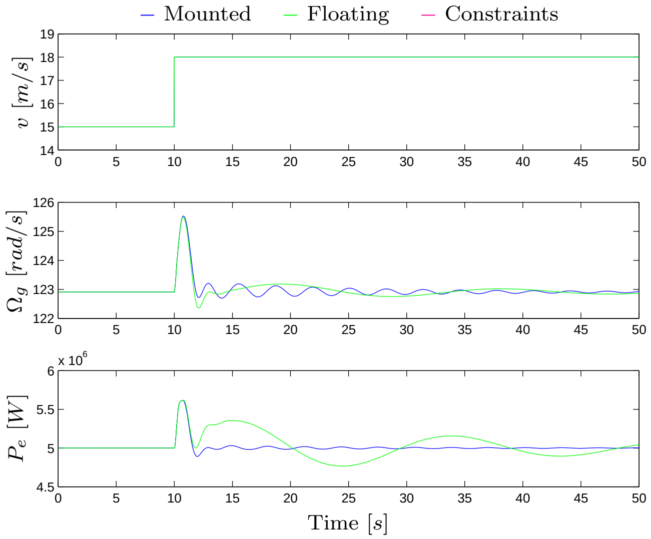


Figure 7.2: Step in wind speed and resulting primary controlled variables

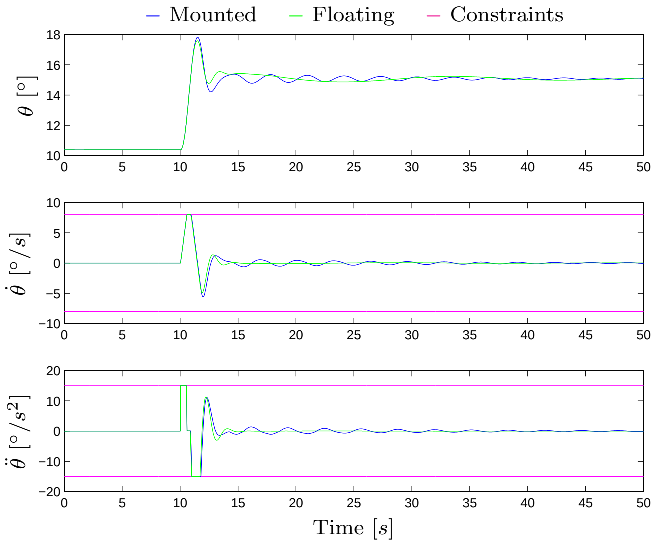


Figure 7.3: Collective pitch and derivatives

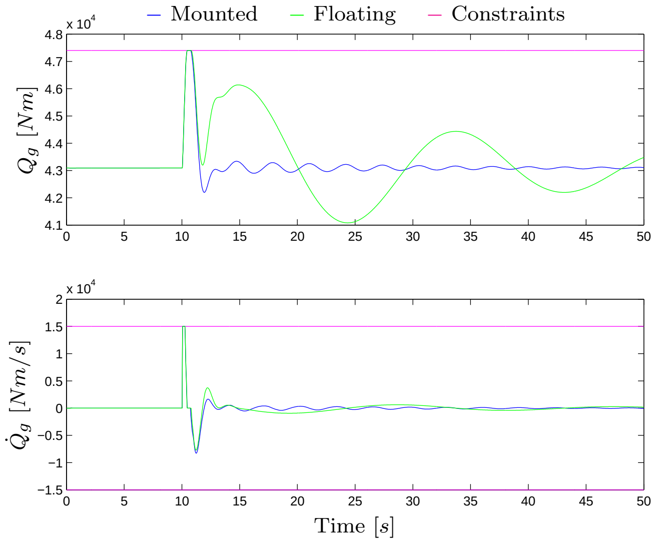


Figure 7.4: Generator torque and derivative

Part III

Implementation and results

Chapter 8

Controller designs

A tactical supervisor is implemented as suggested in chapter 4 and the different controllers are designed. Any design requirements outlined in the previous chapters such as controllability and observability checks are honored.

Each controller consist of a origin shifting controller and a state and disturbance estimator. They are tuned more softly than they would have been if transition between the operating modes where not an issue. The disturbance estimations approach ensures somewhat bumpless transfer as the generator switches in since no integrator wind-up is present. But as it will be seen in chapter 9 the switching criteria makes bumpless transfer impossible since the different controllers have different control objectives. That is due to the fact that the wind turbine is subjected to stochastic wind that means that the switching criteria designed on stationary operation considerations are not sufficient under more difficult operating conditions. This problem becomes especially apparent with the floating wind turbine. The pitch of the controllers is also designed more conservative than needed for the Simulink simulations. That is because fast pitching in HAWC2 leads to a unstable behavior and the simulation fails. In order for the simulations to be comparable the same tuning parameters have been used both the Simulink controller and for the HAWC2 controller.

8.1 Controllers

A common design rule of thumb is to use the inverted square of the linearized operating point or the maximal value of a variable as tuning parameter and then multiply the initial guess with an additional weight. The maximal stationary values of the variables through the wind speed sweep are used as tuning weights. Except for constrained variables in that case the maximal constraint limit is used as the initial weight. The weight matrix **W** contains these basic weights and the tuning vector q is used to adjust them manually to obtain the desired performance.

$$\mathrm{q} = (q_1 \dots q_{n_z}) \tag{8.1}$$

$$\mathbf{W} = \operatorname{diag}\left(\frac{q_1}{z_{1max}} \ \dots \ \frac{q_{n_z}}{z_{n_z max}} \right)^2 \tag{8.2}$$

The pitch controller is only subjected to constraints concerning the collective pitch and its derivatives. The low level controllers K_I, K_{III} and K_{III} are only subjected to constraints concerning the generator torque and its derivative. The top level controller K_{IV} is subjected to constraints concerning both collective pitch and generator and there respective derivatives. The constraints naturally only apply for the constrained linear quadratic controller. The prediction horizon length of the CLQ controllers is $N = 20$.

8.1.1 Pitch controller

The pitch controller has the the following control variables

$$\mathrm{r} = (\theta) \tag{8.3}$$

$$\mathrm{u} = (\theta_{ref}) \tag{8.4}$$

$$\mathrm{z} = (\theta \ \dot{\theta} \ \ddot{\theta})^T \tag{8.5}$$

The pitch controller should slow down and stop the pitch movement when the wind turbine comes from region IV to region III. A high weight has been placed on the pitch velocity to obtain that objective. The tuning parameters of the controller are

$$\mathrm{q} = (\text{1e-1 1e1 1e-1}) \tag{8.6}$$

The optimization matrices are

$$\mathbf{E} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \quad \mathbf{E_p} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} 0 \\ 0 \\ \omega_n^2 \end{bmatrix} \tag{8.7a}$$

8.1.2 K_I and K_{III}

The K_I controller is linearized at a wind speed of $v = 6\ m/s$. The K_{III} controller is linearized at a wind speed of $v = 11.5\ m/s$. They both have the following control variables

$$\mathrm{r} = (\Omega_g) \tag{8.8}$$

$$\mathrm{u} = (Q_{g_{ref}}) \tag{8.9}$$

$$\mathrm{z} = (\Omega_r\ \Omega_g\ \phi_\Delta\ \dot{x}_t\ \ Q_g\ \dot{Q}_g)^T \tag{8.10}$$

The fixed speed controllers should keep the generator speed fixed at its desired set point using the generator torque as the controlling variable. High weights have been placed on the generator and rotor speeds variables to obtain that objective. A high weight has also been placed on the generator torque velocity to keep it somewhat within its limits. The tuning parameters of the controller of the mounted tower are

$$\mathrm{q} = (\text{1e3 1e3 1e-1 1e-3 1e-1 1e1}) \tag{8.11}$$

$$\tag{8.12}$$

The tuning parameters of the controller of the floating tower are

$$\mathrm{q} = (\text{1e2 1e2 1e-1 1e-3 1e-1 1e1}) \tag{8.13}$$

The optimization matrices are

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} \end{bmatrix} \quad \mathbf{E_p} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{\tau} \end{bmatrix} \tag{8.14}$$

8.1.3 K_{II}

The K_{II} controller is linearized at a wind speed of $v = 8\ m/s$ and has the following control variables

$$\mathrm{r} = (P_e) \tag{8.15}$$

$$\mathrm{u} = (Q_{g_{ref}}) \tag{8.16}$$

$$\mathrm{z} = (\Omega_r\ \Omega_g\ \phi_\Delta\ \dot{x}_t\ \ Q_g\ \dot{Q}_g\ P_e)^T \tag{8.17}$$

The power maximizing controller should steer towards the optimal power production. A high weight has been placed on the generator power to obtain that objective. A high weight has also been placed on the generator torque velocity to keep it somewhat within its limits. The tuning parameters of the controllers are

$$\mathrm{q} = (\text{1e-3 1e-3 1e-1 1e-3 1e-1 1e1 1e2}) \tag{8.18}$$

The optimization matrices are

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} \\ 0 & Q_{g_0} & 0 & 0 & 0 & 0 & 0 & \Omega_{g_0} \end{bmatrix} \quad \mathbf{E_p} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{\tau} \\ 0 \end{bmatrix} \tag{8.19}$$

8.1.4 K_{IV}

The K_{IV} controller is linearized at a wind speed of $v = 15\ m/s$ and has the following control variables

$$\mathrm{r} = (\Omega_g \ P_e)^T \tag{8.20}$$

$$\mathrm{u} = (\theta_{ref} \ Q_{g_{ref}})^T \tag{8.21}$$

$$\mathrm{z} = (\Omega_r \ \Omega_g \ \phi_\Delta \ \dot{x}_t \ \theta \ \dot{\theta} \ \ddot{\theta} \ Q_g \ \dot{Q}_g \ P_e)^T \tag{8.22}$$

The top level controller should keep the primary controlled variables at their nominal values. High weight have been placed on the primary controlled variables to obtain that objective. High weights have also been placed on the pitch acceleration and generator torque velocity to keep those somewhat within their limits. The tuning parameters of the controllers are

$$\mathrm{q} = (\text{1e2 1e2 1e-1 1e-3 1e-1 1e-1 1e2 1e-1 1e0 1e3}) \tag{8.23}$$

The optimization matrices are

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\omega_n^2 & -2\zeta\omega_n & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} \\ 0 & Q_{g0} & 0 & 0 & 0 & 0 & 0 & \Omega_{g0} \end{bmatrix} \tag{8.24a}$$

$$\mathbf{E_p} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \omega_n^2 & 0 \\ 0 & 0 \\ 0 & \frac{1}{\tau} \\ 0 & 0 \end{bmatrix} \tag{8.24b}$$

8.2 State and disturbance estimator

All the controllers are disturbance rejecting origin shifting controllers and thus require state and disturbance estimations. The estimators are constructed in accordance with the design rules outlined chapter 5. They are all linearized around the same wind speed as their corresponding controller. They are all tuned with the values given in s_4 in table 5.1.

8.2.1 Pitch controller estimator

The pitch controller has a separate estimator that is linear in it self and hence not linearized around anything.

$$\mathrm{y} = \left(\theta \; \dot{\theta} \; \ddot{\theta}\right)^T \tag{8.25}$$

The output and disturbance matrices are

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} 0 \\ 0 \\ \omega_n^2 \end{bmatrix} \tag{8.26a}$$

$$\mathbf{C_p} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{B_d} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{8.26b}$$

8.2.2 Full sensor

The full sensor estimator has available measurements of all the variables being estimated.

$$\mathrm{y} = \Big(\Omega_r\ \Omega_g\ \phi_\Delta\ x_t\ \dot{x}_t\ \ddot{x}_t\ \theta\ \dot{\theta}\ \ddot{\theta}\ Q_g\ \dot{Q}_g\ P_e\Big)^T \tag{8.27}$$

The output and disturbance matrices are

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \frac{1}{M_t}\frac{\partial Q_t}{\partial \Omega_r}\Big|_{\Omega_{r0}} & 0 & 0 & -\frac{K_t}{M_t} & -\frac{D_t}{M_t} - \frac{1}{M_t}\frac{\partial Q_t}{\partial v}\Big|_{v_0} & \frac{1}{M_t}\frac{\partial Q_t}{\partial \theta}\Big|_{\theta_0} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\omega_n^2 & -2\zeta\omega_n & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} \\ 0 & Q_{g0} & 0 & 0 & 0 & 0 & 0 & \Omega_{g0} \end{bmatrix} \tag{8.28a}$$

$$
\mathbf{C_p} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad
\mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \omega_n^2 & 0 \\ 0 & 0 \\ 0 & \frac{1}{\tau} \\ 0 & 0 \end{bmatrix} \qquad
\mathbf{B_d} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{8.28b}
$$

8.2.3 Reduced sensor

The reduced sensor estimator lacks measurements of drive shaft torsion, nacelle displacement and nacelle velocity.

$$
\mathrm{y} = \left(\Omega_r \; \Omega_g \; \ddot{x}_t \; \theta \; \dot{\theta} \; \ddot{\theta} \; Q_g \; \dot{Q}_g \; P_e \right)^T \tag{8.29}
$$

The output and disturbance matrices are

$$
\mathbf{C} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{M_t} \left. \frac{\partial Q_t}{\partial \Omega_r} \right|_{\Omega_{r0}} & 0 & 0 & -\frac{K_t}{M_t} & -\frac{D_t}{M_t} - \frac{1}{M_t} \left. \frac{\partial Q_t}{\partial v} \right|_{v_0} & \frac{1}{M_t} \left. \frac{\partial Q_t}{\partial \theta} \right|_{\theta_0} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & -\omega_n^2 & -2\zeta\omega_n & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} \\
0 & Q_{g0} & 0 & 0 & 0 & 0 & 0 & \Omega_{g0}
\end{bmatrix} \tag{8.30a}
$$

$$\mathbf{C_p} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \omega_n^2 & 0 \\ 0 & 0 \\ 0 & \frac{1}{\tau} \\ 0 & 0 \end{bmatrix} \quad \mathbf{B_d} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{8.30b}$$

8.3 Implementation

The calculated controller and estimator parameters are loaded to the controllers in the simulation. In HAWC2 the parameters are imported from an file generated by Matlab.

At each sample the following actions are taken by the hybrid controller

1. All estimators are updated with current measurements
2. Tactical supervisor determines which controller is active
3. Active controller computes and actuates control signal (if this is not K_{IV} then the active controller computes the generator torque while the pitch controller controls the collective pitch)

Chapter 9

Simulations in Simulink

In this chapter simulations done in Matlab/Simulink are presented.

9.1 Full sensor ULQ/CLQ

In this section a turbulent wind with a up and down going mean wind speed (5 m/s to 15 m/s to 5 m/s) is used to test hybrid controller performance with regards to operations mode switching and general performance of the individual controllers. Both an the ULQ and CLQ hybrid controllers are tested. Both the CLQ and ULQ controllers have full sensor estimators.

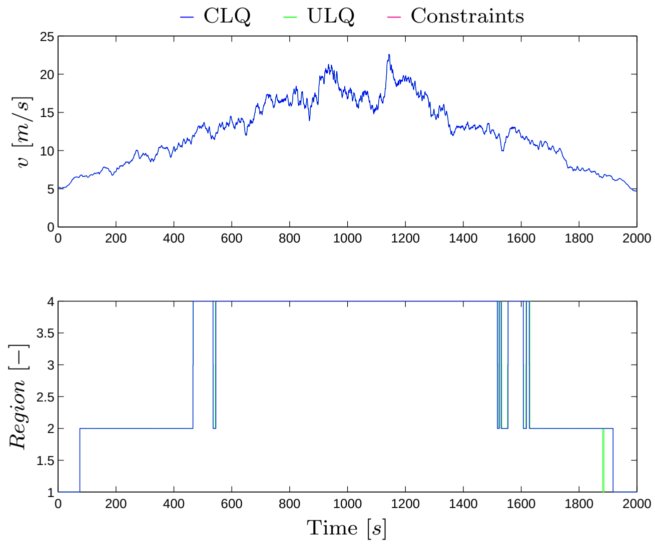


Figure 9.1: Point wind speed at hub height and operation modes of hybrid controller (Mounted tower)

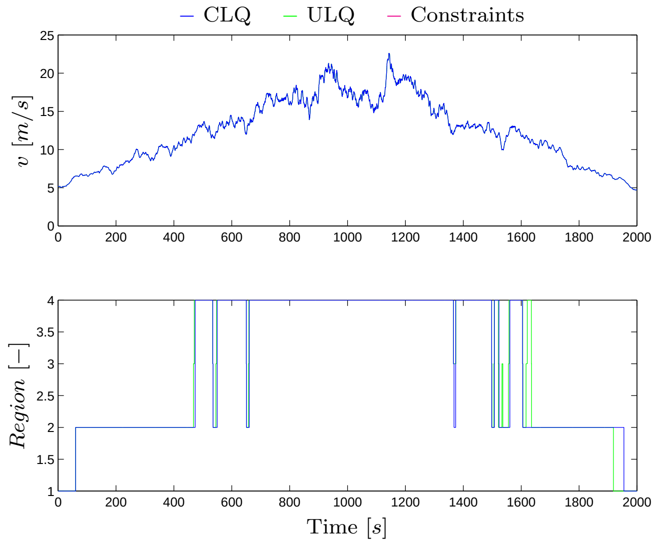


Figure 9.2: Point wind speed at hub height and operation modes of hybrid controller (Floating tower)

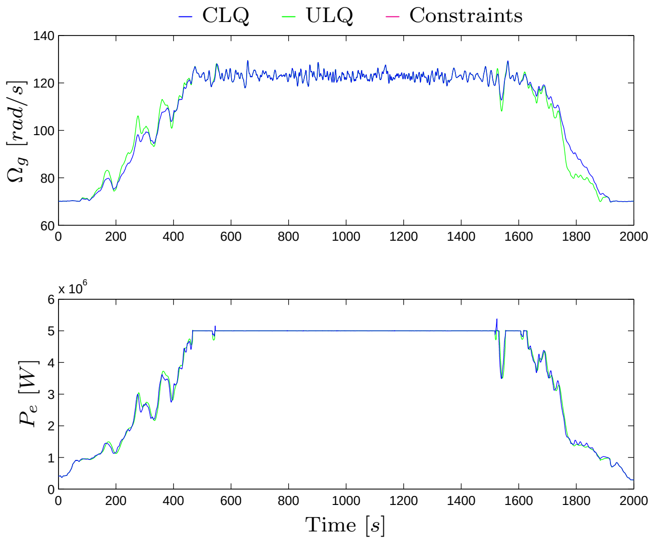


Figure 9.3: Primary controlled variables (Mounted tower)

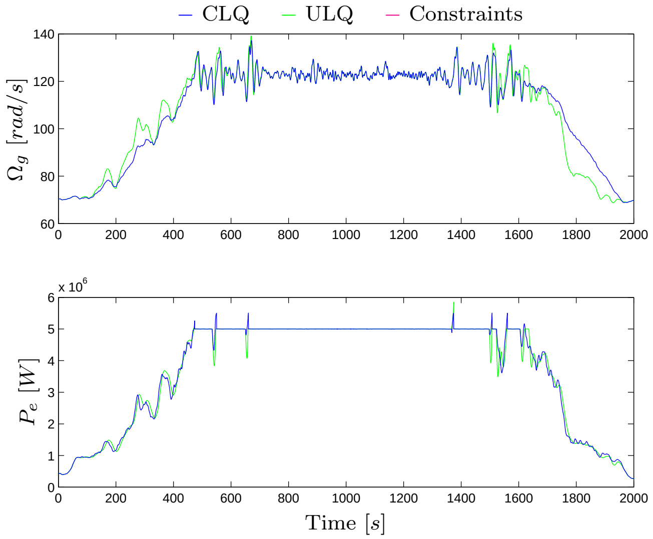


Figure 9.4: Primary controlled variables (Floating tower)

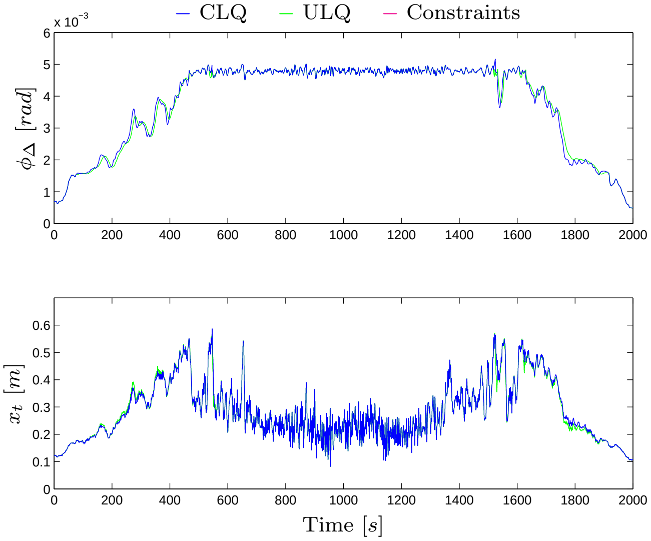


Figure 9.5: Secondary structural controlled variables (Mounted tower)

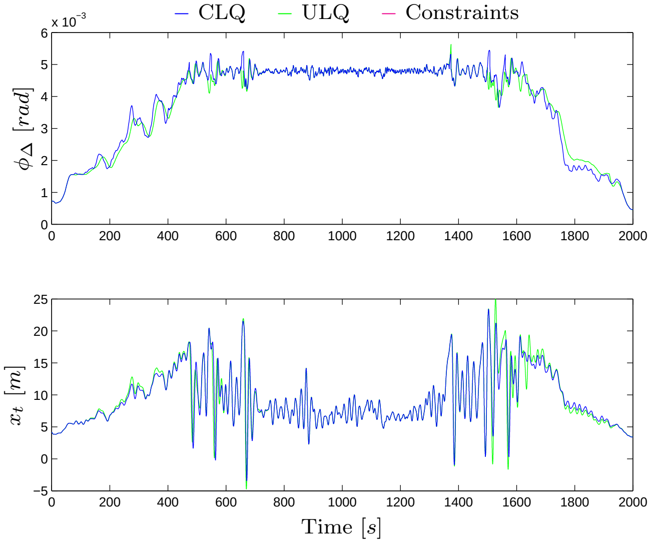


Figure 9.6: Secondary structural controlled variables (Floating tower)

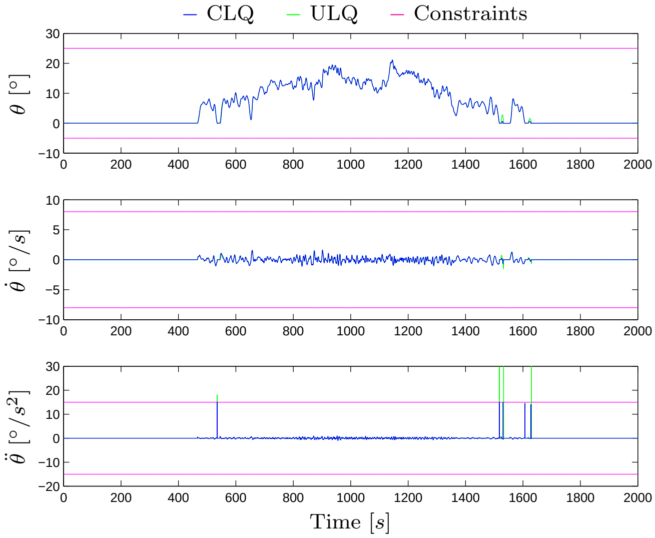


Figure 9.7: Collective pitch and derivatives (Mounted tower)

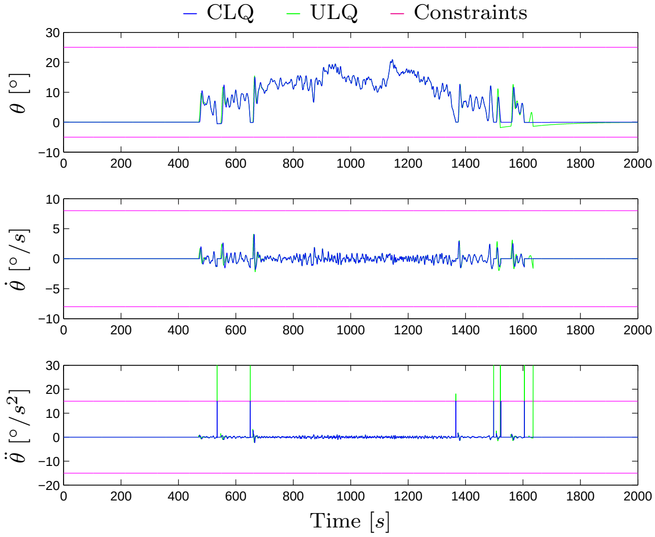


Figure 9.8: Collective pitch and derivatives (Floating tower)

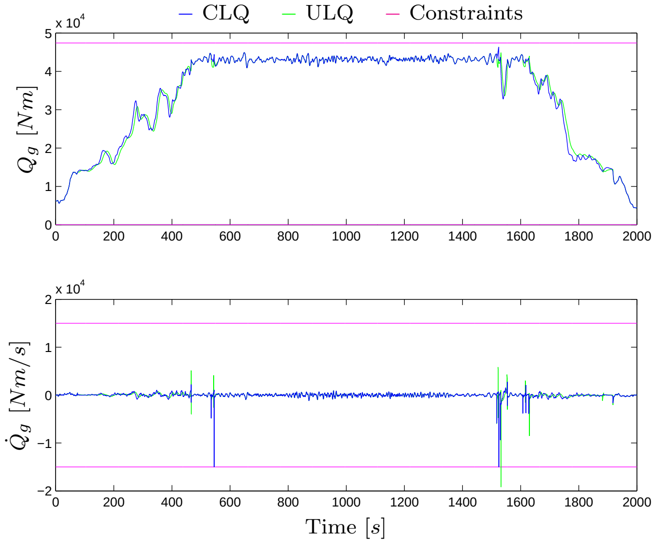


Figure 9.9: Generator torque and derivative (Mounted tower)

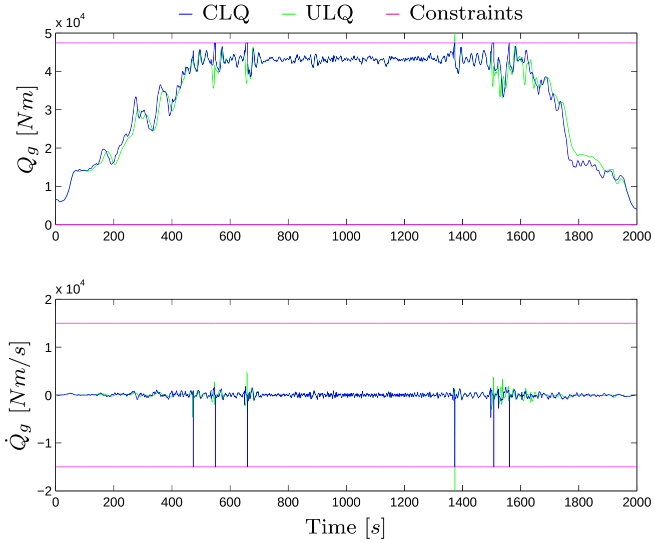


Figure 9.10: Generator torque and derivative (Floating tower)

The figures show that both of the hybrid controllers perform acceptable. The floating tower is performs worse than the mounted tower. This is as expected.

The figures also show switching criteria doesn't work optimal. The only time constraints are active are when the hybrid controller switches from on controller to another. If for instance K_IV is active and the wind speed drops, then it takes some time before the pitch reaches the switching limit. In that period of time the generator speed is falling below its nominal value. That causes the generator torque to increase to keep the generator power stable. When the hybrid controller finally switches to K_{III} it directly switches again to K_{II} because of the small wind range of K_{III}. Now the generator speed is low but the generator torque is high. K_{II} sets its generator torque according the generator speed and the generator torque drops to its new set point as fast as possible. This conflict of objectives leads to a less than bumpless transfer between the controllers. Not because of the controllers themselves but because of the switching conditions.

The switching criteria are determined by stationary considerations and are not sufficient in a dynamic situation, especially for the floating tower. The tuning weights of the controllers could be changed to prioritize this concern, but that would give worse normal performance. Additional controllers suited for the different situations could be designed or another set of switching criteria could be defined, fx. based on each controllers controllers control signal.

Chapter 10

Simulations in HAWC2

In this chapter simulations done in HAWC2 are presented. Sadly is has not been possible to carry out extensive stochastic simulations with the CLQ controller. The program continuously fails at some point before the simulation finishes. That has not been a problem with the ULQ controller. HAWC2 is not able to start the simulation at any operating point except a total stands still. This fact has given cause to a high time consumption when trying to find errors in the implementation.

10.1 Stationary comparison of HAWC2 vs Simulink model

In this simulation a deterministic wind speed staircase from 3 m/s to 22 m/s is used to compare the stationary values of the Simulink and HAWC2 models. Furthermore the dynamic responses can be compared also be compared. The simulations are performed with the full sensor ULQ controller.

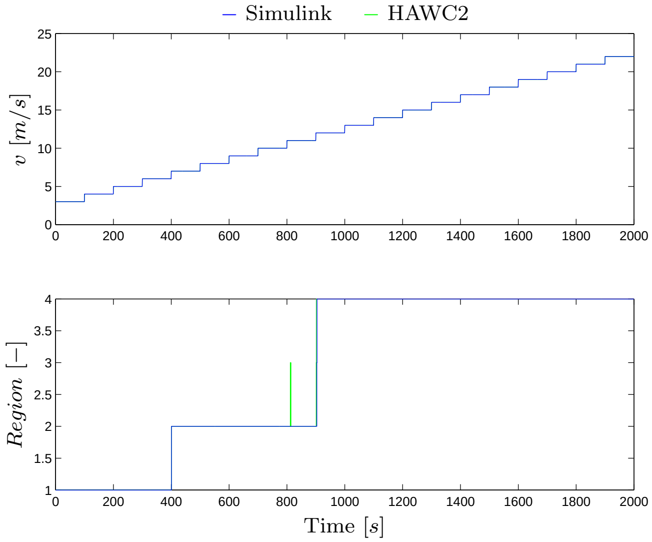


Figure 10.1: Wind speed and operations modes of hybrid controller (Mounted tower)

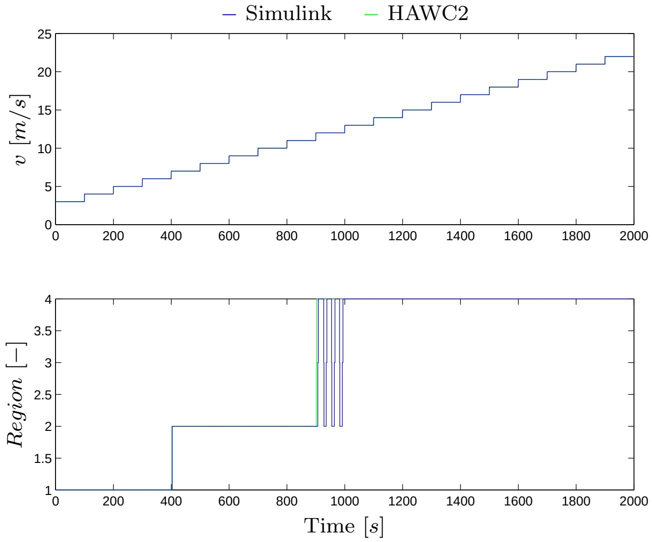


Figure 10.2: Wind speed and operations modes of hybrid controller (Floating tower)

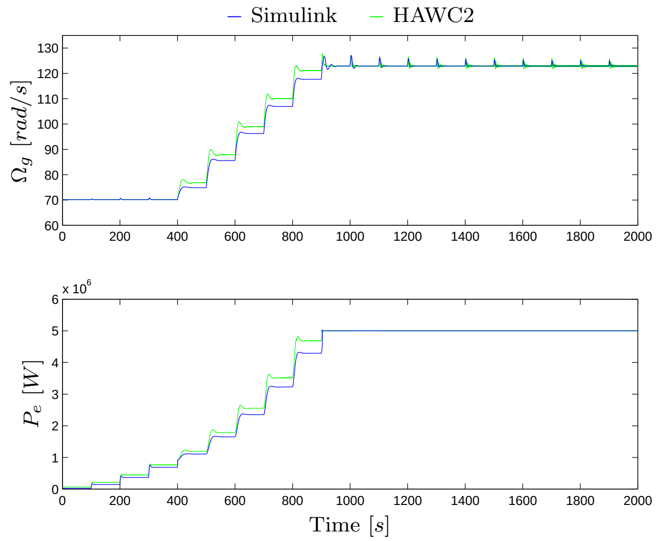


Figure 10.3: Primary controlled variables (Mounted tower)

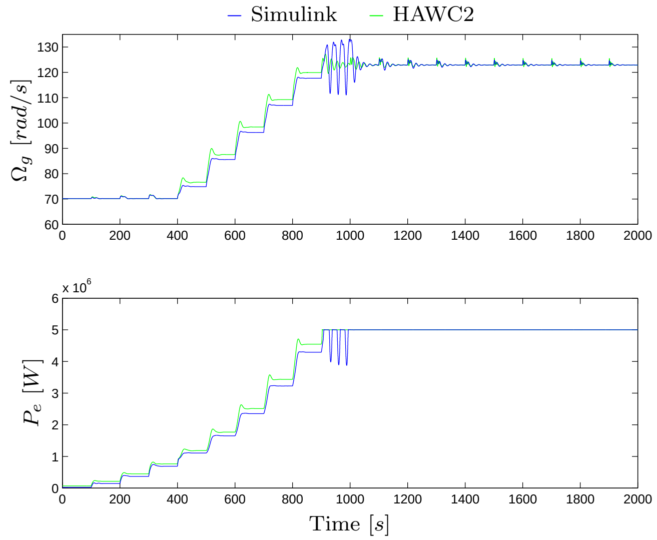


Figure 10.4: Primary controlled variables (Floating tower)

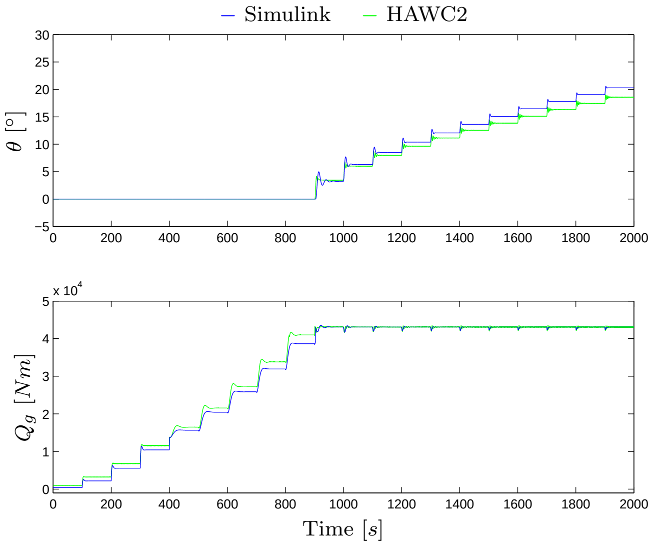


Figure 10.5: Main actuator variables (Mounted tower)

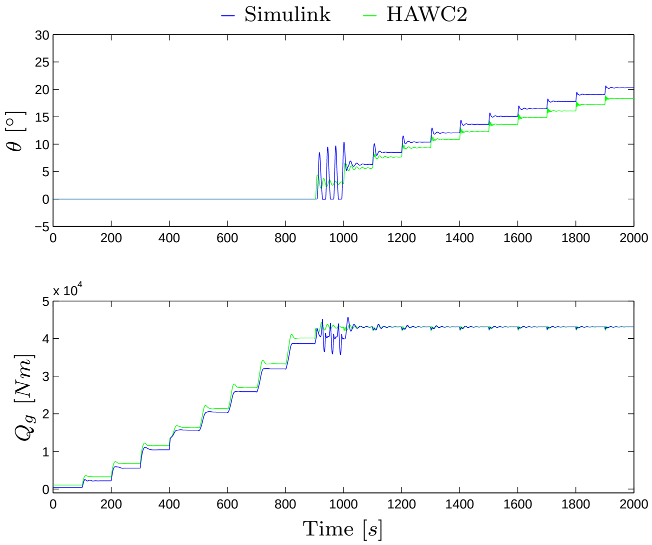


Figure 10.6: Main actuator variables (Floating tower)

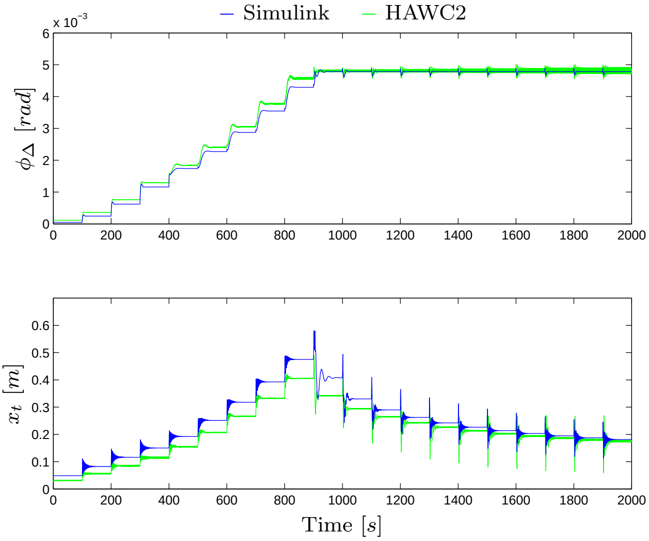


Figure 10.7: Secondary structural controlled variables (Mounted tower)

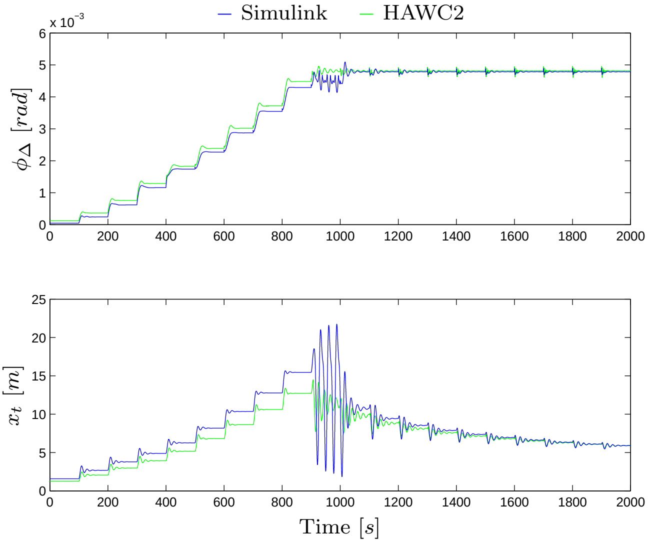
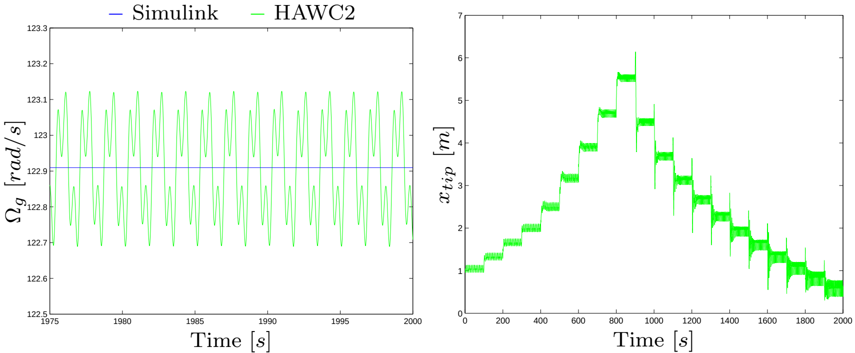
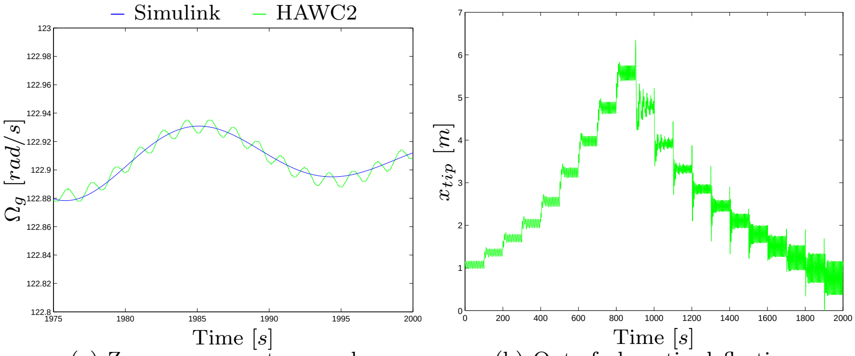


Figure 10.8: Secondary structural controlled variables (Floating tower)



(a) Zoom on generator speed (b) Out of plane tip deflection

Figure 10.9: Mounted tower



(a) Zoom on generator speed (b) Out of plane tip deflection

Figure 10.10: Floating tower

The behavior of the Simulink and HAWC2 simulations is quite similar, but a offset in stationary values of the collective pitch Fig. (10.5) and Fig. (10.6) is observed. This discrepancy is correlated with the unmodeled out-of-plane tip deflection of the rotor that occurs in HAWC Fig. (10.9(b)) and Fig. (10.10(b)). The discrepancy affects the rest of variables in the simulations and leads to offset in nacelle displacements etc.

In the HAWC2 mounted tower simulation both wind shear and tower shadow is enabled. In the HAWC2 floating tower simulation tower shadow is disabled due to simulation errors. This is clear to see in Fig. (10.9(a)) and Fig. (10.10(a)).

10.2 Full/reduced sensor ULQ

In this section a turbulent wind with a up and down going mean wind speed (5 m/s to 15 m/s to 5 m/s) is used to test hybrid controller performance with regards to operations mode switching and general performance of the individual controllers. Since it hasn't been possible to perform full scale simulations with the CLQ controller only the ULQ controller is tested in this section. It is tested both with a full sensor estimator and a reduced sensor estimator.

The simulations show generally good performance both by the full sensor ULQ and by the reduced sensor ULQ hybrid controllers. The reduced sensor floating tower simulation shows however a quite oscillating behavior of the wind turbine as it switches from K_{II} to K_{IV}. And again the issue of timely switching between the operation modes is not working satisfactory. Especially in the reduced sensor floating tower simulation. It is seen in Fig. (10.14) where the generator power is a good deal above the nominal value before the tactical controller switches from K_{II} to K_{IV}.

Fig. (10.19) and Fig. (10.20) shows the unmeasured estimates of the reduced sensors estimator belonging K_{IV}. It is clear that the estimates are more accurate when the wind turbine is actually in the K_{IV} region.

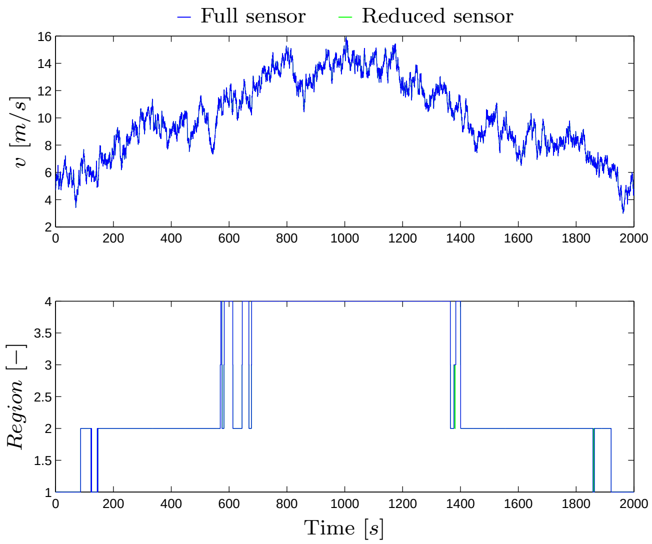


Figure 10.11: Point wind speed at hub height and operation modes of hybrid controller (Mounted tower, ULQ)

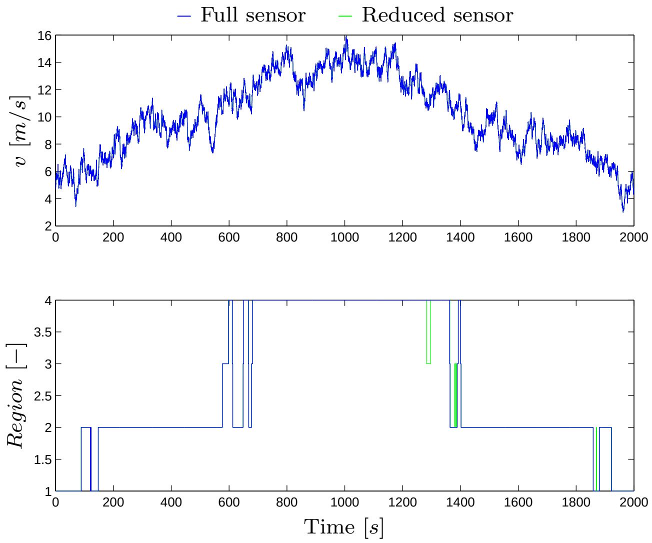


Figure 10.12: Point wind speed at hub height and operation modes of hybrid controller (Floating tower, ULQ)

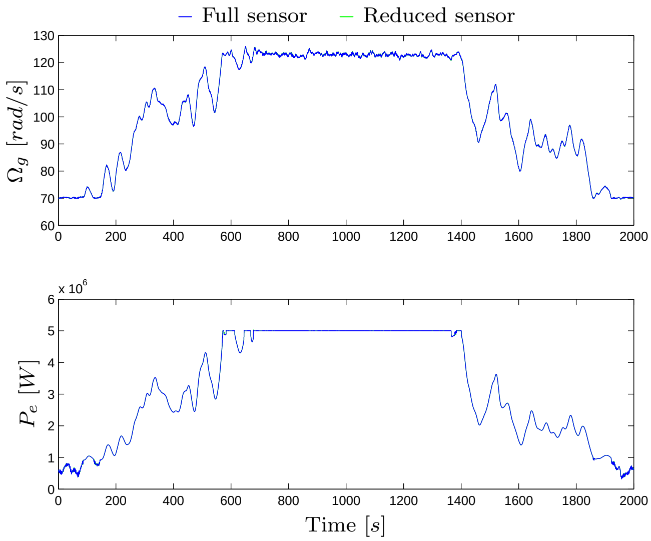


Figure 10.13: Primary controlled variables (Mounted tower, ULQ)

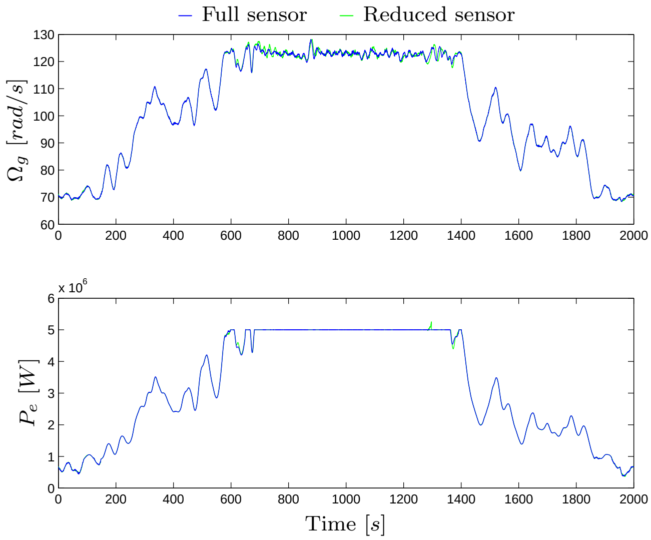


Figure 10.14: Primary controlled variables (Floating tower, ULQ)

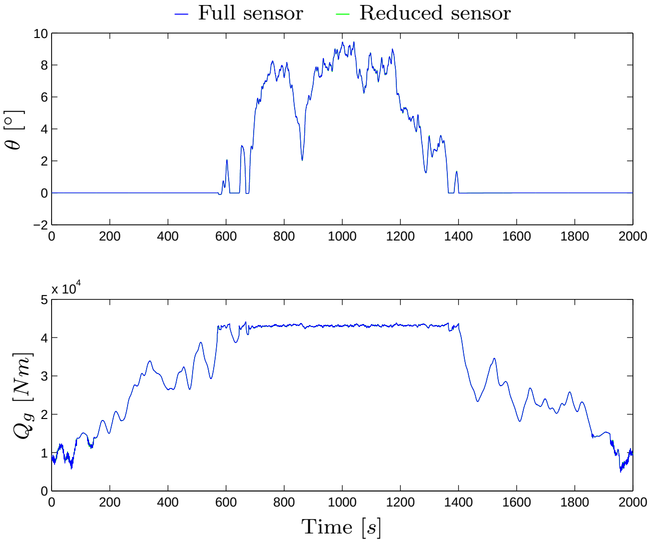


Figure 10.15: Main actuator variables (Mounted tower, ULQ)

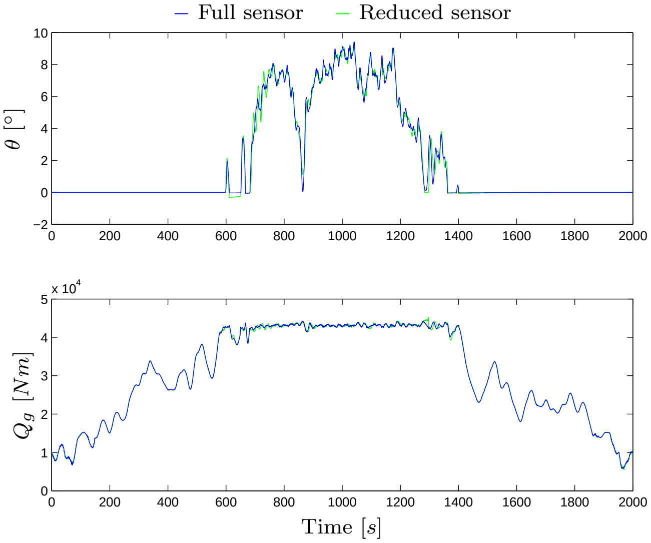


Figure 10.16: Main actuator variables (Floating tower, ULQ)

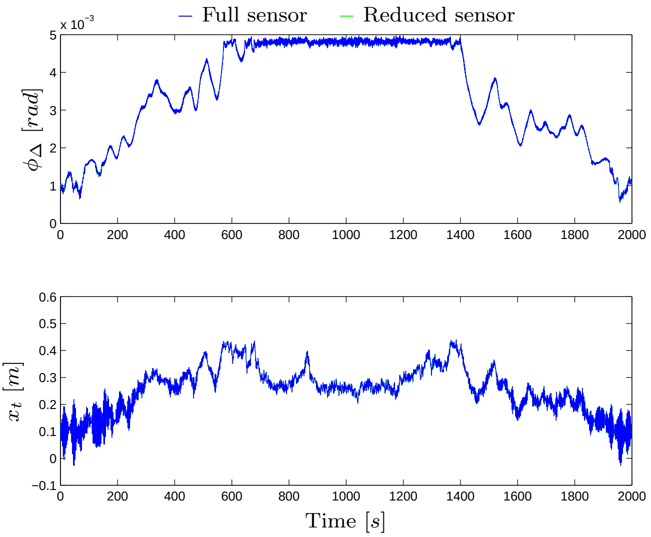


Figure 10.17: Secondary structural controlled variables (Mounted tower, ULQ)

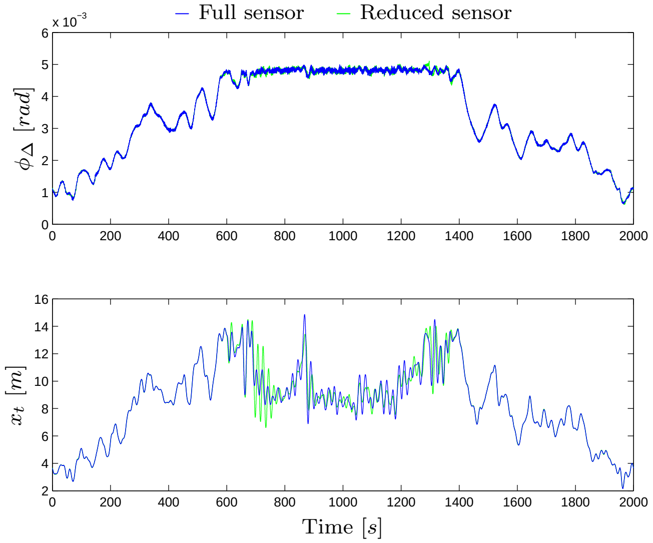


Figure 10.18: Secondary structural controlled variables (Floating tower, ULQ)

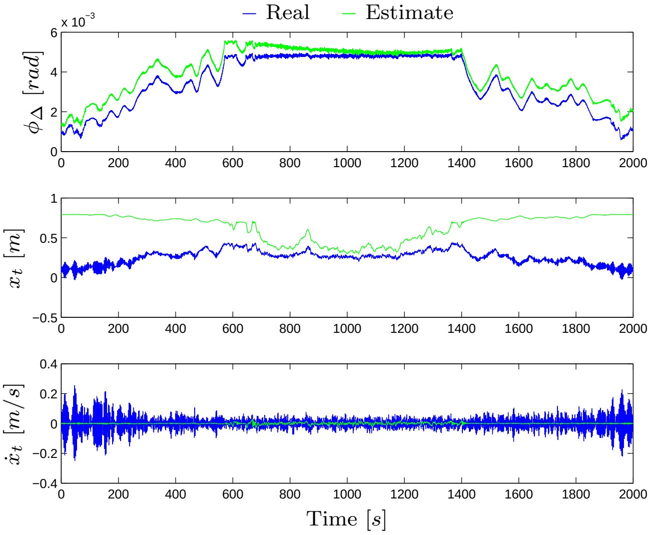


Figure 10.19: Unmeasured variables (Mounted tower, Reduced sensor ULQ)

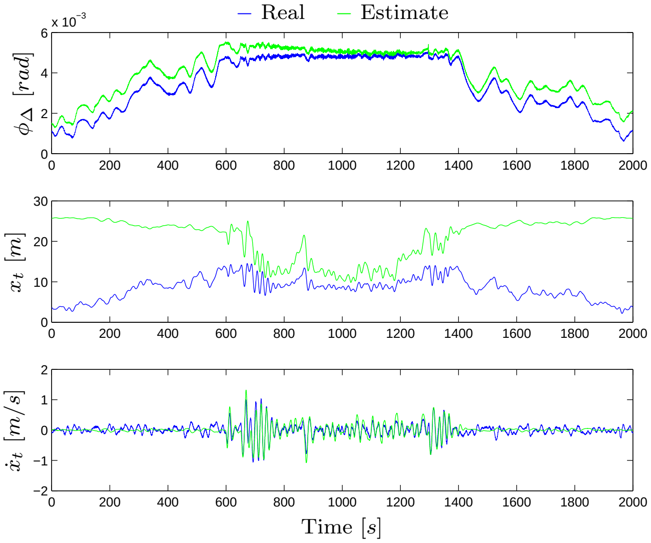


Figure 10.20: Unmeasured variables (Floating tower, Reduced sensor ULQ)

Part IV

Conclusion and perspectives

Chapter 11

Conclusion

In this chapter conclusions and results will be drawn from the different parts of the report.

11.1 Modeling and analysis

Two models have been developed: A mounted tower model and a floating tower model. They have been approximated as linear 2. order dynamic systems dimensioned through experiments in HAWC2. The nonlinear dynamics of an aerodynamic rotor have been linearized at the appropriate operating points. The driveshaft of the wind turbine have been modeled as a linear 2. order dynamic system and it has been dimensioned through experiments in HAWC2. Different operating modes have been identified to ensure maximal power production and honoring requirements to the mechanical limits of the generator.

Tactical operations mode switching criteria have been established to enable switching between the individual controllers of a hybrid controller. All the considerations on these issues have been on stationary conditions and the later chapters have shown that more elaborate switching criteria are required.

11.2 Theory of methods

The origin shifting and disturbance rejecting controller have been presented. It enables set point tracking and offset-free control. The infinite horizon model predictive controller have been presented. There are unsolved issues concerning constraint violations in the infinite horizon but this issues has not proofed to be a serious problem in the implementation of an constrained linear quadratic controller. The constraints in this project have however only been on linear well-known subsystems. How the CLQ controller behaves if constraints where on the nonlinear variables have not been subject to attention.

The entire CLQ formulation is time-discrete and has a built-in observer. This makes the CLQ suited for real work applications and enables implementation in other simulation suites such as HAWC2.

11.3 Implementation and results

CLQ and ULQ controllers have been designed and tested in Simulink and HAWC2 with satisfactory results. It has sadly not been possible to perform long stochastic simulations with the CLQ controller in HAWC2 due to software problems. But the controllers have nonetheless proofed to work over the entire wind range and with reduced sensor information available. The Simulink simulations have also shown that CLQ controller is able to honor constraints under stochastic conditions with switching of controllers.

The designed switching criteria have been seen not to be sufficient for bumpless transfer of controllers at certain situations.

Comparative simulations between Simulink and HAWC2 have shown minor discrepancies between the two models. Tower shadow and out-of-plane blade tip deflection are among the more significant ones resulting in slightly different behavior of the two models.

It has been established that control of floating wind turbines poses new problems but it is definitely possible to overcome these problems and establish good robust control via MPC and more detailed investigations in terms of bumpless transfer between controllers.

Chapter 12

Perspectives

On the modeling side of the project attention should be paid to tower shadow and rotor dynamics as these are seen to differ from the model developed in this project. A better model of the rotor dynamics might also enable pitching at lower wind speeds to ensure better power capture.

The control design model could also be augmented with a wind model to give better performance and enable a gain schedueling controller. The CLQ formulation is based on linear models. Instead nonlinear and even robust MPC-methods could be investigated to give better performance. A nonlinear robust formulation might enable a multi objective controller that was able to switch been the different operating modes internally and thus giving a smoother transition from one operating mode to another. Adaptive or system identification methods could also be incorporated into the controller framework to give better performance.

Implementation of controllers on HAWC2 have been difficult. The libraries used to program the CLQ have not been optimal and debug is made difficult since HAWC2 is unable to start the simulation at any given different operating point. Integration between Matlab and HAWC2 would be desirable and thus only one controller development was required. It is possible to create dll-files with Matlab but it has not been explored in this project, that could perhaps be a shortcut in terms of controller development.

On a more general perspective in can be noted that advanced control methods such as MPC is definitely the future of wind turbine control. Many other objectives could integrated into the framework such component fatigue prioritization, accoustic silencing prioritization etc. depending on the location of the wind turbine and on the time of day.

Part V

Appendices

APPENDIX A

Notation

Scalars are typed in lower case italic style: x_1 where $x_1 \in \mathbb{R}$

Vectors are typed in lower case roman style: $\mathrm{x} = (x_1, x_2, \ldots x_n)^T$ where $\mathrm{x} \in \mathbb{R}^n$ and encased in parenthesis brackets $(\ldots)$

A column of repeated vectors is denoted with an super $^{\mathrm{col}}$:

$$\mathrm{x}^{\mathrm{col}} = \begin{pmatrix} \mathrm{x} \\ \vdots \\ \mathrm{x} \end{pmatrix} \tag{A.1}$$

A matrix, i.e. $\mathbf{A} \in \mathbb{R}^{n \times m}$ is typed in upper case boldface style and the scalar elements are type like seen below

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & & a_{1,m} \\ & \ddots & \\ a_{n,1} & & a_{n,m} \end{bmatrix} = \begin{bmatrix} [\mathbf{A}]_{\{1,1\}} & & [\mathbf{A}]_{\{1,m\}} \\ & \ddots & \\ [\mathbf{A}]_{\{n,1\}} & & [\mathbf{A}]_{\{n,m\}} \end{bmatrix} \tag{A.2}$$

A matrix of diagonal matrices is denoted with an upper $^{\text{diag}}$

$$\mathbf{A}^{\text{diag}} = \begin{bmatrix} \mathbf{A} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{A} \end{bmatrix} \tag{A.3}$$

A positive definite matrix is written as

$$\mathbf{A} > 0 \tag{A.4}$$

A row-wise linear inequality is written as

$$\mathbf{A} \succ 0 \tag{A.5}$$

Appendix B

System parameters

B.1 Parameter identification of a mechanical 2. order system

This approach is described in (Jannerup and Sørensen, 2000) and (Poulsen, 2007) Newtons second law applied to a classical linear motion spring-mass-damper mechanical system gives the following differential equation

$$Q(t) = M\frac{d^2}{dt^2}x(t) + D\frac{d}{dt}x(t) + Kx(t) \tag{B.1}$$

which can be transformed from the time domain to the Laplace domain

$$Q(s) = Ms^2x(s) + Dsx(s) + Kx(s) \tag{B.2}$$

and reformed to a transfer function

$$K\frac{x(s)}{Q(s)} = H(s) = \frac{1}{\frac{M}{K}s^2 + \frac{D}{K}s + 1} \tag{B.3}$$

if subjected to a step at time t_0

$$Q(t) = \begin{cases} 0 & \text{if } t < t_0, \\ Q_{ss} & \text{if } t \geq t_0. \end{cases} \tag{B.4}$$

in Laplace

$$Q(s) = \frac{Q_{ss}}{s} \tag{B.5}$$

In steady state which given by $t \rightarrow \infty$ in the time domain and equals $s \rightarrow 0$ in the Laplace domain

$$Kx_{ss} = \lim_{s \rightarrow 0} sH(s)Q(s) = Q_{ss} \tag{B.6}$$

giving the spring constant

$$K = \frac{Q_{ss}}{x_{ss}} \tag{B.7}$$

this can be compared to the standard formulation of a 2. order system given in Jannerup and Sørensen (2000)

$$\frac{Y(s)}{U(s)} = \frac{1}{\frac{1}{\omega_n^2}s^2 + \frac{2\zeta}{\omega_n}s + 1} \tag{B.8}$$

where

$$\zeta = \frac{|\ln(M_p)|}{\sqrt{\pi^2 + \ln^2(M_p)}} \tag{B.9}$$

$$\omega_n = \frac{\omega_d}{\sqrt{1 - \zeta^2}} \tag{B.10}$$

which can be determined from the step response overshoot ratio M_p and damped frequency f_d

$$M_p = \frac{y_{max} - y_{ss}}{y_{ss}} \tag{B.11}$$

$$\omega_d = 2\pi f_d \tag{B.12}$$

giving

$$M = \frac{K}{\omega_n^2} \tag{B.13}$$

$$D = 2\zeta\omega_n M \tag{B.14}$$

No.	1	2	3
Q	1e6	2e6	4e6
X_0	4e-7	4e-7	4e-7
X_{max}	1.8e-3	3.6e-6	7.1e-3
X_{ss}	1.1e-3	2.3e-6	4.6e-3
M_p	0.57	0.56	0.55
f_d	0.60	0.60	0.60
ζ	0.18	0.18	0.18
ω_n	3.84	3.84	3.84
K	8.7e8	8.7e8	8.7e8
M	5.9e7	5.9e7	5.9e7
D	8.0e7	8.3e7	8.3e7

Table B.1: Drive shaft step responses

No.	1	2	3
Q	2e5	4e5	8e5
X_0	-1.6e-2	-1.6e-2	-1.6e-2
X_{max}	2.4e-1	4.8e-1	9.5e-1
X_{ss}	1.2e-1	2.4e-1	4.8e-1
M_p	1.01	0.97	0.96
f_d	0.31	0.31	0.31
ζ	4.4e-3	6.5e-3	1.2e-2
ω_n	1.98	1.98	1.98
K	1.6e6	1.6e6	1.6e6
K	4.3e5	4.2e5	4.2e5
D	7.4e3	1.1e4	2.0e4

Table B.2: Fixed tower displacement step responses

No.	1	2	3
Q	2e5	4e5	8e5
X_0	-2.7e-1	-2.7e-1	-2.7e-1
X_{max}	7.7	15.4	30.9
X_{ss}	3.9	7.8	15.7
M_p	0.99	0.97	0.96
f_d	5.3e-3	5.3e-3	5.3e-3
ζ	3.2e-3	8.9e-3	1.1e-2
ω_n	0.34	0.34	0.34
K	5.1e5	5.1e5	5.1e5
M	4.5e5	4.5e5	4.5e5
D	9.7e2	2.7e3	3.5e3

Table B.3: Floating tower displacement step responses

B.2 Mechanical data for NREL 5MW wind turbine

P_{nom}	5e6 W	-	Nominal generator power
$\Omega_{g_{max}}$	122.91 rad/s	-	Nominal generator speed
$\Omega_{g_{min}}$	70.16 rad/s	-	Minimum generator speed
θ_{max}	25 °	-	Maximum collective pitch angle
θ_{min}	-5 °	-	Minimum collective pitch angle
$\dot{\theta}_{max}$	8 $°/s$	-	Maximum collective pitch angular velocity
$\dot{\theta}_{min}$	-8 $°/s$	-	Minimum collective pitch angular velocity
$\ddot{\theta}_{max}$	15 $°/s^2$	-	Maximum collective pitch angular acceleration
$\ddot{\theta}_{min}$	-15 $°/s^2$	-	Minimum collective pitch angular acceleration
ρ	1.25 kg/m^3	-	Mass density of air
N_g	97	-	Gear ratio
R	63 m	-	Rotor blade length and rotor disc radius
H	90 m	-	Height from surface to center of rotor
I_r	5.9154e7 $kg\ m^2$	-	Moment of inertia of rotor
I_g	500 $kg\ m^2$	-	Moment of inertia of generator
K_s	8.7354e8 N/rad	-	Driveshaft spring constant
D_s	8.3478e7 $N/rad\ s$	-	Driveshaft dampening constant
M_t	4.4642e5 kg	-	Mass of part of floating tower and nacelle
K_t	5.0871e4 N/m	-	Floating tower spring constant
D_t	3.5159e3 $N/m\ s$	-	Floating tower dampening constant
M_t	4.2278e5 kg	-	Mass of part of mounted tower and nacelle
K_t	1.6547e6 N/m	-	Mounted tower spring constant
D_t	2.0213e3 $N/m\ s$	-	Mounted tower dampening constant
ω_n	0.88 rad/s	-	Natural frequency of pitch actuator
ζ	0.9	-	Damping of pitch actuator
τ	0.1 s	-	Time constant of generator torque actuator

Appendix C

Constrained optimization

C.1 Convexity

In order for a minimum to be determined the function to be optimized should convex (and concave for maximization). Convexity means that between any two points within the function a straight line can be drawn without the line gets outside the interior of the function.

The minimum is a global minimum if the function is positive definite (and a maximum if the function is negative definite). From (Slotine and Li, 1991) we have the following definition

Definition C.1 (Positive definite function) A scalar continuous function V(x) is said to be locally positive definite if if $V(0) = 0$, and in a ball B_{R0}

$$x \neq 0 \quad \Rightarrow V(x) \quad > 0$$

If $V(0) = 0$ and the above property hold overt the hole state space, then $V(x)$ is said to be globally positive definite.

The definition means that a function a positive definite if $V(x) \rightarrow \infty$ for $x \rightarrow \infty$.

In this project only linear control theory is applied and all of the functions to be optimized are matrices. A matrix is positive definite if the eigenvalues of the matrix are positive.

The following notation is used to indicate positive definiteness in this report: $\mathbf{A} > 0$

C.2 Linear and Quadratic Programming

Quadratic programming is particular type of optimization capable of handling linear matrix equalities and inequalities. (Boyd and Vandenberghe, 2004, Page 152)

$$\mathrm{x}^* = \arg\min_{\mathrm{x}} \frac{1}{2}\mathrm{x}^T\mathbf{Q}\mathrm{x} + \mathrm{q}^T\mathrm{x} \tag{C.1a}$$

s.t.

$$\mathbf{A}\mathrm{x} = \mathbf{b} \tag{C.1b}$$
$$\mathbf{C}\mathrm{x} \preceq \mathbf{d} \tag{C.1c}$$

where $\mathrm{x} \in \mathbb{R}^n$, $\mathbf{Q} \in \mathbb{R}^{n\times n}$ If the quadratic matrix $\mathbf{Q}$ is positive definite, then the problem is convex and a global minimum can be determined. If the problem is only subjected to equality constraints then the problem can be reformulated to an unconstrained optimization problem.

If $\mathbf{Q} = 0$ the problem becomes a LP problem instead. As a side note the Simplex algorithm is a LP-method.

If the size of decision variables increases so does the required number of iterations to determine a constrained suboptimal solution. That means that the QP-algorithm usually have a upper limit of iterations, but that limit should be increased if the QP problem itself is increased.

The QP problem is solved either an interior-points method that utilizes the fact the function is convex or active-sets method travels along the constraints towards the optimum. The underlying method is in it self not interesting but is still debated which underlying methods are best suited for MPC.

C.2.1 Implementional issues

To ensure the symmetry of the quadratic matrix due to numerical issues the following method can be used

$$\mathbf{Q} = \frac{1}{2}\left[\mathbf{Q} + \mathbf{Q}^T\right] \tag{C.2}$$

Appendix D

HAWC2

Hawc2 is FEM simulation program developed by Risø. The controllers are implemented as dll-files and their interaction with HAWC is defined in a htc-file, which also defines the HAWC2 model.

`http://risoe-staged.risoe.dk/business_relations/Products_Services/consultancy_service/VEA_aeroelastic_simulation.aspx`

D.1 Models

The mounted tower is modeled as a tower of 10 elements based on a monopile. On the floating tower model an extra element is added between the tower and monopile. This elements is extremely flexible and emulates the floating motion of floating wind turbine.

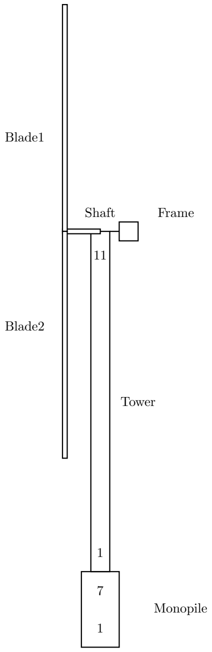


Figure D.1: Structural composition of c32f (Floating tower)

D.2 Programming

Unfortunately the controller developed for HAWC is not sufficiently stable (seen from a programming point of view) and time and time again the program stops halve way into the simulation with an error message saying that no memory could be allocated to a GSL routine.

D.2.1 IDE

The dll files used by Hawc2 are programmed in C and the integrated development environment (IDE) is Bloodshed Dev-C++ and it is available for free under the GNU General Public License (GPL)

`http://www.bloodshed.net/devcpp.html`

D.2.2 Used libraries

Besides the standard libraries: The GNU Scientific Library (GSL) is a numerical library for C and C++ programmers. It is free software under the GNU General Public License.

`http://www.gnu.org/software/gsl/`

It is however a Windows port of GSL that have been used

`http://gnuwin32.sourceforge.net/packages/gsl.htm`

In addition an GSL extension has been used to solve the quadratic programming problems developed by Ewgenij Hübner it is called CQP

`http://www.mathematik.uni-trier.de/~huebner/software.html`

D.3 Implementation

The controller parameters are calculated in Matlab and exported to a txt-file which the dll-file reads when it initializes.

The CQP package by Ewgenij Hübner requires equality constraints in its function call so the QP problem have been augmented with and additional decision variable which should be equal to zero, i.e. $v_{k+N+1} = 0$.

Appendix E

HAWC2 CLQ simulation

This appendix shows that a CLQ controller have been developed for HAWC2. The simulation only activated the generator torque velocity constraints.

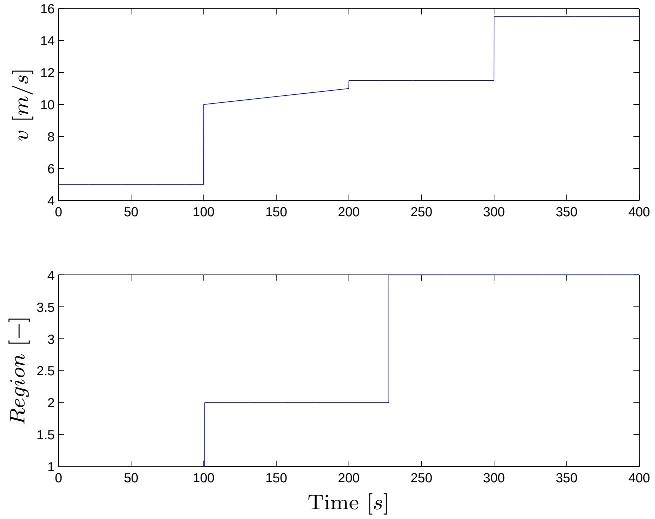


Figure E.1: Point wind speed at hub height and operation modes of hybrid controller (Mounted tower)

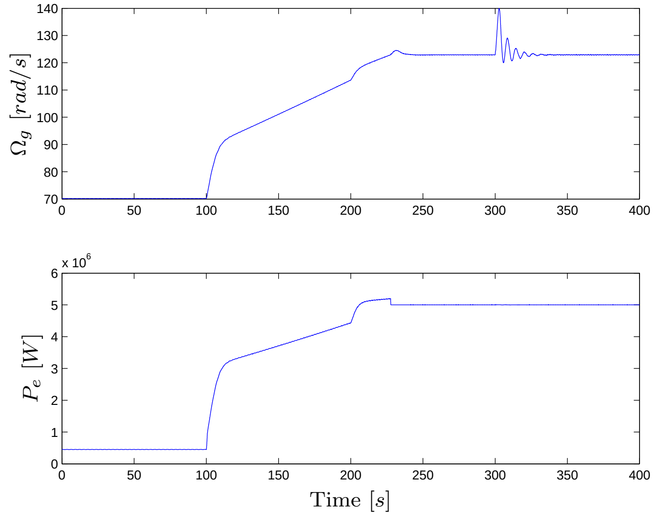


Figure E.2: Primary controlled variable (Mounted tower)

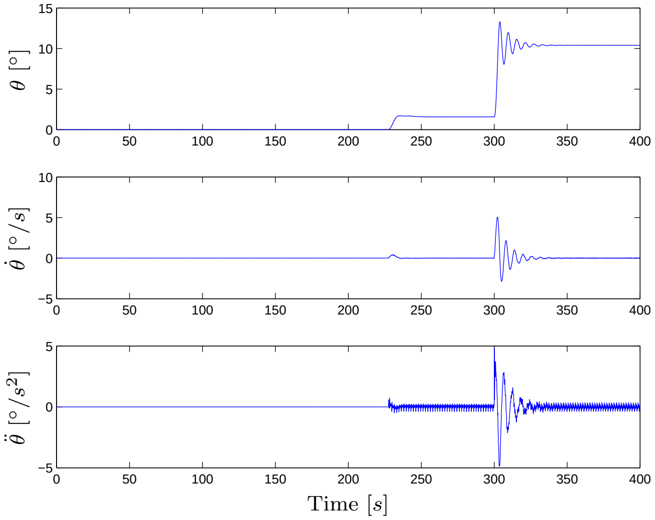


Figure E.3: Collective pitch and derivatives (Mounted tower)

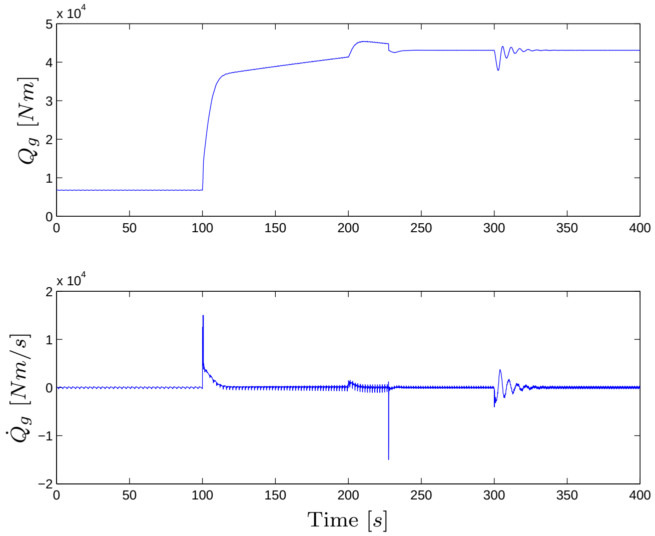


Figure E.4: Generator torque and derivative (Mounted tower)

Bibliography

Blanchini, F. (1999). Set invariance in control. *Automatica 35*(11), 1747–1767.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization.* Cambridge University Press.

Burton, T., D. Sharpe, N. Jenkins, and E. Bossanyi (2001). *Wind Energy Handbook.* John Wiley And Sons Ltd.

Hammerum, K. (2006). A fatigue approach to wind turbine control. Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby. Supervised by Assoc. Prof. Niels Kjølstad Poulsen, IMM, DTU, in collaboration with the Loads, Aerodynamics and Control group at Vestas Wind Systems, Denmark.

Hansen, M. O. L. (2000). *Aerodynamics of Wind Turbines.* James & James (Science Publishers) Ltd.

Jannerup, O. and P. H. Sørensen (2000). *Introduktion til reguleringsteknik.* Politeknisk Forlag.

Johnson, K. E., L. J. Fingersh, M. J. Balas, and L. Y. Pao (2004). Methods for increasing region 2 power capture on a variable-speed wind turbine. *Wind Energy and Journal of Solar Energy Engineering, Transactions of the ASME 126*(4), 1092–1100.

Kwakernaak, H. and R. Sivan (1972). *Linear Optimal Control System.* John Wiley & Sons, Inc.

Larsen, A. J. and T. S. Mogensen (2006). Individuel pitchregulering af vindmølle. Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby. Supervised by Assoc. Prof. Niels Kjølstad Poulsen, IMM.

Muske, K. R. and T. A. Badgwell (2002). Disturbance modeling for offset-free linear model predictive control. *Journal of Process Control 12*(5), 617–632.

Pannocchia, G., N. Laachi, and J. B. Rawlings (2005). A candidate to replace pid control: Siso-constrained lq control. *AIChE Journal 51*(4), 1178–1189.

Pannocchia, G. and J. B. Rawlings (2003). Disturbance models for offset-free model-predictive control. *AIChE Journal 49*(2), 426–437.

Poulsen, N. K. (2007). *Stokastisk adaptiv regulering.* IMM Trykkeri.

Rossiter, J. (2003). *Model-Based Predictive Control.* Taylor & Francis Ltd.

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

Østergaard, P. (1994). Pitchregulering af en vindmølle. Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby.