WEC Development Project

Winter Semester 2024/2025

Draft Version Design Report

Project Name: **Optimus Shakti 5.0**
Sub-Project: **Turbine Controller**

Prepared by:	**Matriculation number:**
Felix Lehmann	670183
Karan Soni	760153
Julius Preuschoff	750203

Supervisor

Prof. Dr.-Ing. David Schlipf

Date: January 17, 2025

Contents

List of Figures

Acronyms

AEP	Annual Energy Production
BMS	Battery Management System
CLC	Close Loop Control
CPC	Collective Pitch Control
DEL	Damage Equvivalent Load
DLC	Design Load Case
DOF	Degree Of Freedom
FAST	OpenFAST
Shakti 5.0	Optimus Shakti 5.0
SLOW	Simplified Low-Order Wind turbine
SSD	Storage System Dummy
TD	Tower Damper
WT	Wind Turbine

List of Symbols

kp	controller gain
F_{a}	aerodynamic force
M_{G}	generator torque
$\Omega_{\text{G,rated}}$	rated generator speed
Ω_{G}	generator speed
R	rotor radius
T	simulation time
x_{T0}	initial tower top deflection
x_{T}	tower top deflection
$C_{p,\text{opt}}$	power coefficient
C_p	power coefficient
k_{Te}	tower equivalent bending stiffness
λ	tip speed ratio
θ_{k}	gain scheduling parameter for pitch controller
θ_{min}	minimum static pitch angle
θ	pitch angle
v_{in}	cut-in wind speed
v_{rated}	rated wind speed

1 Introduction and Motivation (Soni)

The report provides an overview of the dynamic controller for the Optimus Shakti 5.0 (Shakti 5.0) wind turbine. Developed as part of the "Development of a Wind Turbine" module in the Wind Energy Engineering Master's program at the University of Applied Sciences in Flensburg, this prototype aims to prepare students with the skills needed to develop a wind turbine. This year's Optimus turbine has a rated power of $5\,\mathrm{MW}$, a rotor diameter of $178\,\mathrm{m}$ and a $5\,\mathrm{kWh}$ battery storage. Based on the tubular concept with a gearbox drivetrain, it is planned for onshore installation in Karnataka, India.

General objectives and requirements regarding the design of a dynamic controller will be discussed in the following chapters. It is important to highlight the challenges posed by operating in low wind speed areas. One critical aspect is the large rotor disc, which must perform efficiently under these conditions. The varying wind speeds result in dynamic loads that the control system must manage.

For the design of the advanced controller, we used a $3.4\,\mathrm{MW}$ reference wind turbine from the IEA Wind TCP Task 37 [1] as a starting point. The reference model provided a solid foundation for developing a reliable control system that optimizes performance under various conditions.

The report covers essential topics in wind turbine design and optimization, including wind field generation, simple storage system scenarios simulations and control parameter tuning. Different challenges were faced during the development such as design of the rated wind speed and the mismatch of the Simplified Low-Order Wind turbine (SLOW) and OpenFAST (FAST) model.

2 Controller Design Objectives

The motivation of control in Wind Turbine (WT) is to optimize the energy production, avoid overspeed or other constraints and reduce structural loads [5]. In order to manage the above named requirements there are different control tools used. They are organized in an hierarchical order [2]:

1. safety system
2. supervisory control
3. Close Loop Control (CLC)

During the project of the Shakti 5.0 turbine the focus was on the development of the CLC.

2.1 Closed-Loop Control (Soni)

Figure 2.1 illustrates the CLC system of a wind turbine. The main control subject is the rotational speed of the turbine. To obtain the desired behavior two different controllers are used, the pitch controller and the torque controller.

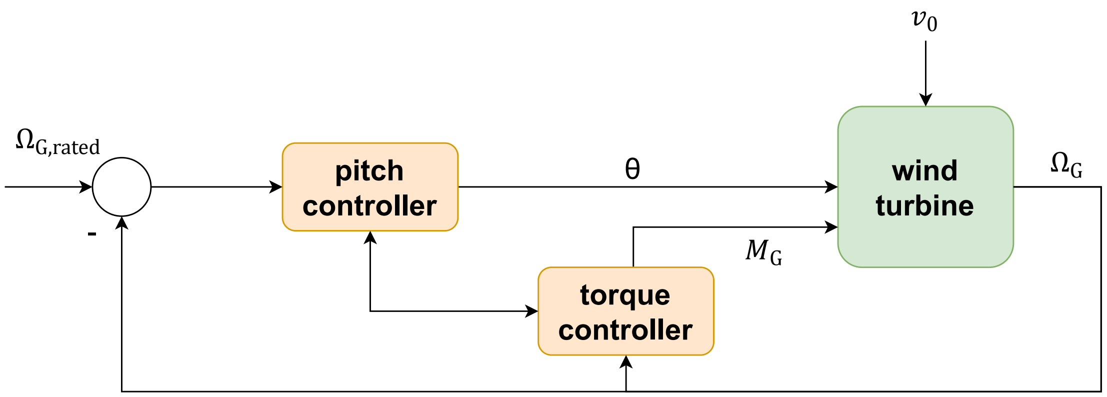

Figure 2.1 Closed-loop wind turbine control scheme

The torque controller optimizes power production below rated wind speed and maintain rated power above rated wind speed by adjusting generator torque (M_G) based on generator speed (Ω_G). The torque controller uses a PI-controller with anti-windup to regulate the generator torque based on the difference between the actual and reference generator speed.

Above rated wind speed, the pitch controller maintains the rated generator speed ($\Omega_{G,rated}$)

by adjusting the blade pitch angle (θ), which ensures the rated power is maintained. With increasing wind speed, the pitch angle increases to reduce the power coefficient (C_p). Once the wind speed reaches the rated value, the generator torque is maintained at its rated value. The control is categorized into different regions, as detailed in section 2.2.

2.2 Control Regions (Soni)

Wind turbine operations are segmented into three primary and two transition regions based on varying wind speed. These regions are explained in detail below and illustrated in Figure 2.2.

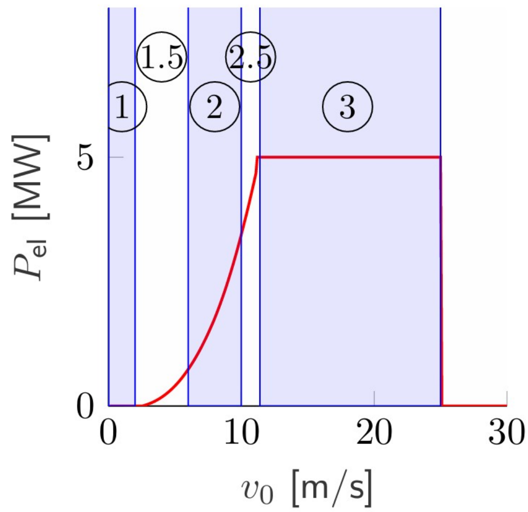

Figure 2.2 Wind turbine control regions [5]

Control Region 1: The wind speed is below the cut-in wind speed (v_{in}). The turbine does not generate any power. There is no pitch activity.

Control Region 1.5: This is a transition to Region 2. The wind speed is sufficient to accelerate rotor and increase the generator torque. The goal is to quickly reach Region 2. There is no pitch activity.

Control Region 2: Wind speeds are above the cut-in speed (v_{in}) but below the rated wind speed (v_{rated}). The primary objective is to maximize energy yield. To achieve this, the power coefficient ($C_{p,opt}$) is kept at its optimum. There is no pitch activity. The control system ensures that the turbine maintains these optimal conditions by regulating the rotational speed through the torque controller. In this region, the generator torque is adjusted based on rotor speed, as shown in Equation 2.1.

$$M_G = \frac{1}{2}\rho R^5 \frac{C_{P,opt}}{\lambda_{opt}^3 r_{GB}^3}\Omega_G^2 \tag{2.1}$$

$$M_\mathsf{G} = k\Omega_\mathsf{G}^2 \tag{2.2}$$

Control Region 2.5: This is a transition to Region 3. The rotational speed and the torque are raised until they reach the rated values. The goal is to quickly and smoothly reach Region 3. There is no pitch activity.

Control Region 3: In this region, wind speeds have reached the rated wind speed. The control goal is to maintain rated power and generator speed as well as reduce structural loads. The pitch controller actively adjusts the blade pitch angle to regulate the speed and keep the power within the turbines rated conditions.

2.3 Advanced Generator Torque Controller (Soni)

Generator torque is one of the two main control inputs for a wind turbine. The performance of an advanced generator torque controller offers significant improvements and greater flexibility compared to a baseline torque controller. The primary goal of the advanced torque controller is to reach the optimal power curve earlier and to maintain it for a longer duration compared to the baseline controller. Additionally, the dynamics in Regions 1.5 and 2.5 are tunable, allowing for more precise control.

Goals of the advanced torque controller:

- Achieve the optimal power curve earlier and maintain it for a longer period.
- Enable tunable dynamics in Regions 1.5 and 2.5.
- Interaction with PI-pitch controller to reduce loads and increase energy yield.

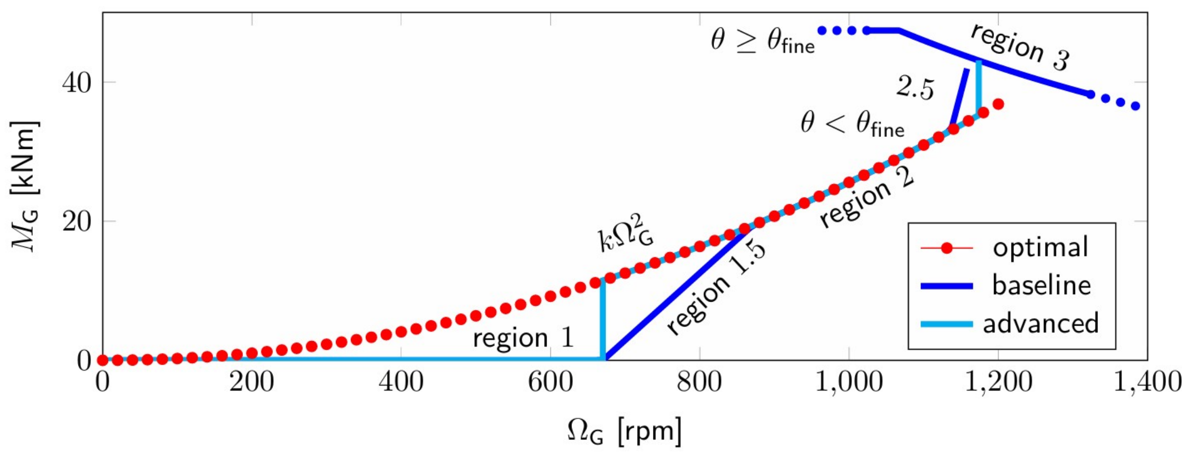

Figure 2.3 Wind turbine control regions [5]

Strategy for the Advanced Torque Controller:

Region 1.5: Lowest generator speed $\Omega_{\mathsf{G},1.5}$ set to avoid the 3P frequency interacting with the tower's eigenfrequency.

Region 2: The controller aims to maximize energy yield while ensuring the turbine operates efficiently.

Region 2.5: Generator speed $\Omega_{G,2.5}$ = $\Omega_{G,rated}$. Switch from $\Omega_{\text{G},1.5}$ to $\Omega_{\text{G},2.5}$ if the measured generator speed Ω_{G} exceeds

$$\Omega_{\text{G,R2switch}} = \frac{1}{2}(\Omega_{\text{G},1.5} + \Omega_{\text{G},2.5}) \tag{2.3}$$

Torque Limits:

The generator torque limits are determined by the measured generator speed Ω_{G} and incorporate Anti-Windup mechanisms.

- if Ω_{G} < $\Omega_{\text{G,R2switch}}$:

$$M_{\text{G,lb}} = 0 \tag{2.4}$$

$$M_{\text{G,ub}} = k\Omega_{\text{G}}^2 \tag{2.5}$$

- ifΩ_{G} > $\Omega_{\text{G,R2switch}}$:

$$M_{\text{G,lb}} = k\Omega_{\text{G}}^2 \tag{2.6}$$

$$M_{\text{G,ub}} = \min\left(M_{\text{G,rated}}\frac{\Omega_{\text{G,rated}}}{\Omega_{\text{G}}}, M_{\text{G,max}}\right) \tag{2.7}$$

Region 3: Adjusted limits for the generator torque based on the generator speed.

The design parameters for the PI controller are derived similar to exercise 3 of [5].

2.4 Collective Pitch Controller (CPC) (Julius)

Collective Pitch Control (CPC) adjusts the pitch for all 3 blades similarly. The pitch control behavior has a high impact on the structural loads therefor on the life time of the wind turbine and thus on costs. CPC can be implemented with a standard PI controller. Main task of the CPC is to make the rotor area more permeable for the wind in order to reduce the power coefficient. This is done by pitching the rotor blades in a less advantageous aerodynamic position. With increasing wind speed the power output increases as well as the loads. In order to keep the loads within an acceptable limit the power output of the wind turbine must be limited.

The pitch controller is only active in region 3, when the wind speed is above the rated wind speed as described in figure 2.2. In region 3 the pitch controller maintains rated speed and the generator torque controller rated torque [5]. In the Shakti 5.0 wind turbine a gain scheduled PI controller is used to control the rotor speed.

The concept of gain scheduling is widely used and a common feature in blade pitch controllers. With the use of gain scheduling the gain parameter kp of the controller is changed based on the operating point of the system. The parameter θ_{k} is used to change the gain of the CPC. The operating point is determined by the pitch angle θ. Based on the operating point the scheduled gain kp_{gs} is derived with 2.8.

$$\frac{kp}{\frac{\theta-\theta_{\text{min}}}{\theta_k}+1} = kp_{\text{gs}} \tag{2.8}$$

The control parameters for the PI controller were based on the model of the IEA reference turbine [1] and the adjustments were brute force optimized.

2.5 Tower Damper (Felix)

This section describes the function and the implementation of a pitch angle based TD. The design follows the description in [2] similar as the workflow in the exercise of the corresponding lecture [5]. The tower dynamics are modeled as in [2] (eq: 8.12 and 8.13). Here referred to as 2.9 and 2.10.

$$M\ddot{x} + D\dot{x} + Kx = F + \Delta F \tag{2.9}$$

$$\begin{aligned} \Delta F = \frac{\partial F}{\partial \theta}\Delta\theta = -D_{\text{TD}}\dot{x} \\ \Delta\theta = \frac{-D_{\text{TD}}}{\partial F/\partial\theta}\dot{x} \end{aligned} \tag{2.10}$$

As described by 2.9 the dynamics of the tower in fore-aft direction are lightly damped if D is small and the force ΔF which is the additional thrust force resulting of a pitch action is equal to zero. The force F is damped by the relative wind speed $v_{\text{rel}} = v_0 - \dot{x}$ and therefore $F = F(\Omega, \theta, v_{\text{rel}})$ [5]. To damp the tower top speed $\dot{x}$ even further [2] proposes an update of the pitch angle with $\Delta\theta$. This will damp the tower motion further as described in 2.10. This leads to a reduction of the tower bottom bending moment. Nevertheless this comes at a cost of higher pitch activity and the damping is only available in control region 3.

The implementation and test of the damper is done in Matlab and Simulink. As in the lecture [5] and corresponding exercise the tower top acceleration $\ddot{x}$ is used because it is a quantity that is measurable in reality as well. What is also taken into account is the existence of a real pitch actuator. This means that the pitch update $\Delta\theta$ can not be applied instantaneously because of the time constant of the pitch actuator. To address this phenomena there are 2 methods tested. First a direct integration 2.11:

$$\dot{x}(t) = \int \ddot{x}(t)\text{d}t \tag{2.11}$$

And second a phase shift of $90°$ of the acceleration signal by a Lag-Compensator. The transfer function in the frequency domain is shown in 2.12. Where the input in the frequency domain is $\ddot{X}(s)$ and the output is $\dot{X}(s)$.

$$\frac{\ddot{X}(s)}{\dot{X}(s)} = \frac{s+z}{s+p} \tag{2.12}$$

$$\dot{x}(t) = \ddot{x}(t) - \int p\dot{x}(t) - z\ddot{x}(t)\text{d}t \tag{2.13}$$

The implementation in Simulink (Figure 2.4) is first following the approach in the exercise [5].

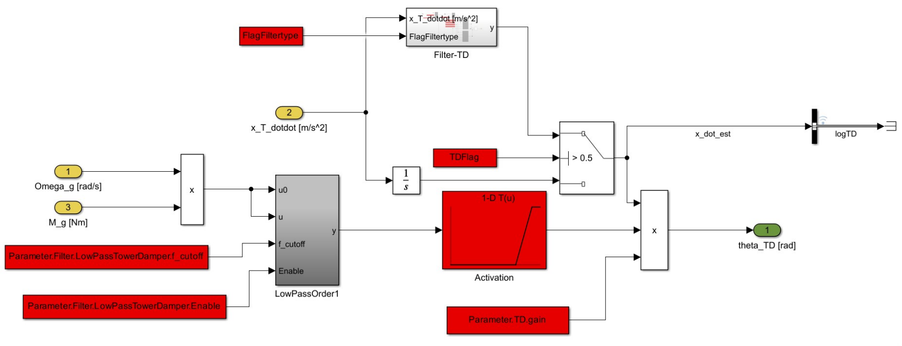

Figure 2.4 Tower Damper in Simulink model

The input *Omega_g* is already a low pass filtered quantity and the blade passing frequency (3P) is notched out. As shown in Figure 2.4 to activate the TD the generator power is used. The *LowPassOrder1* is used to reduce the switching frequency of the TD to ensure that it is not switched on and off if the WT is operating near rated conditions. In the activation the gain is slowly ramped up from $0\,\%$ to $100\,\%$ over a power range from $80\,\%$ of the rated power to $100\,\%$. For higher power values the gain stays at $100\,\%$.

In the first method, here named integrator, the tower top acceleration signal is integrated. The signal is than multiplied with the damping gain *Parameter.TD.gain* and the activation signal as shown in Figure 2.4. The resulting quantity is the pitch offset mentioned in 2.10. This offset $\Delta\theta$ is added to the pitch angle control value of the CPC and this sum is the new input for the SLOW-model. In this first ideal method no pitch actuator is considered.

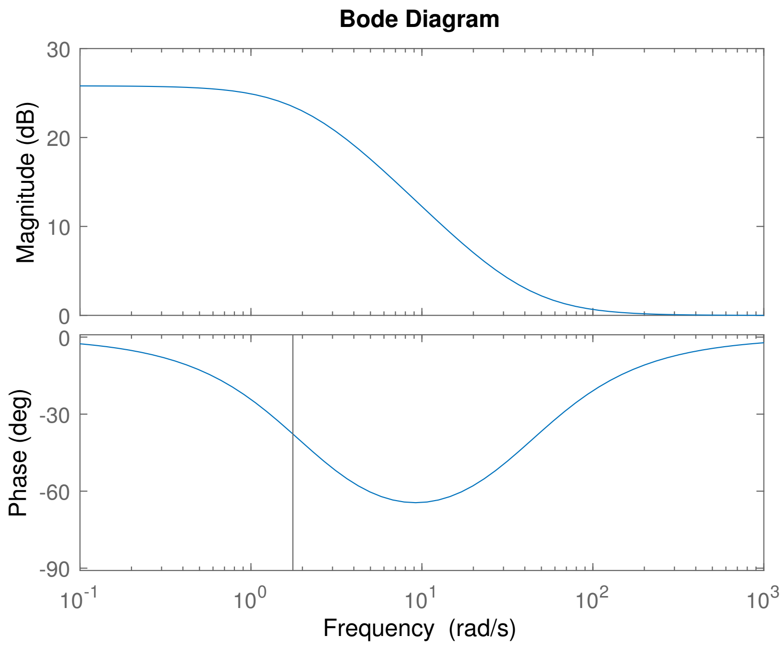

Figure 2.5 Bode plot of the Lag-Compensator with tower eigenfrequency (black)

The second method, here named Lag-Compensator, isolates the tower eigenfrequency by passing the acceleration signal first through a low pass filter and than a high pass filter. The cutoff frequency of both filters is the eigenfrequency of the tower. Now the signal contains mainly the isolated eigenfrequency of the tower with which the tower is oscillating in the fore-aft direction. To phase shift the signal a Lag-Compensator is used.

As shown in Figure 2.5 the frequency is phase shifted by nearly $37°$ and the magnitude is increasing by $23\,\mathrm{dB}$. This leads with an included pitch actuator that behaves like a PT2-Filter to an overall system shift of $90°$ and a magnitude increase of $16.5\,\mathrm{dB}$ see Figure 2.6. The higher magnitude is adjusted by a different gain value.

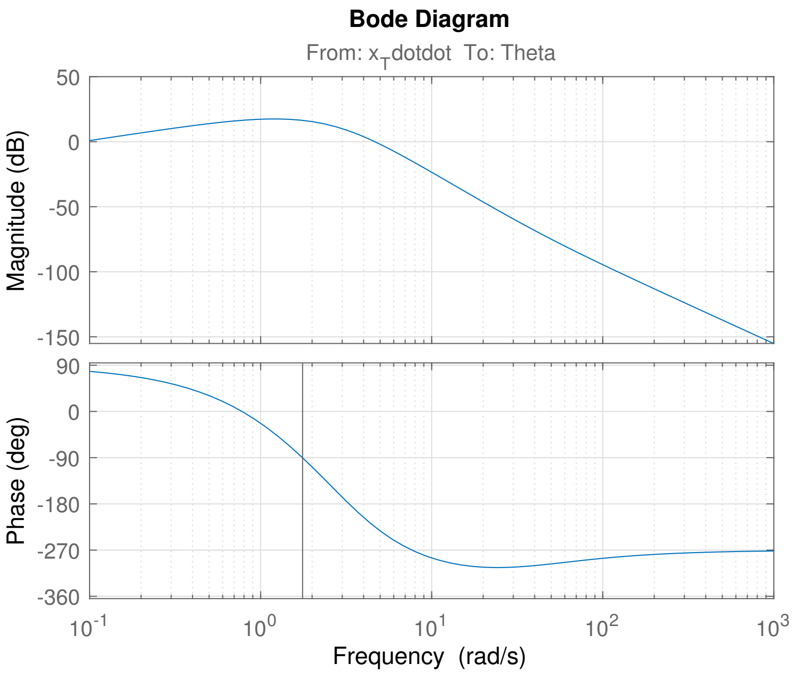

Figure 2.6 Bode plot of the Pitch-Actuator and Filter-Chain with tower eigenfrequency (black)

The design is tested with a wind step from $v_0 = 20\,\mathrm{m/s}$ to $v_1 = 21\,\mathrm{m/s}$. The results are shown in Figure 2.7.

As shown in Figure 2.7 the result is not matching the design and expectations above. The tower top speed is not in phase with the integrator method. It is to expect, that the implementation is containing an error. To debug the system the filter and the integrator method are disturbed by a $\cos$-signal with the eigenfrequency of the tower as shown in Figure 2.8. The two methods are not perfectly aligned. The difference at the beginning is not the issue but that the signals are not converting into the same is causing the issue. First fine tuning iterations of the Lag-Compensator does not lead to a successful result. This issue remains unsolved by the end of the project end requires a deeper investigation.

The expected result with the filter result is, that the loads at the eigenfrequency of the tower are reduced the same way as they are with an integrator. But due to the fact that the filter is considering the signal delay by the pitch actuator the loads around the eigenfrequency are not increased as with a pure integrator.

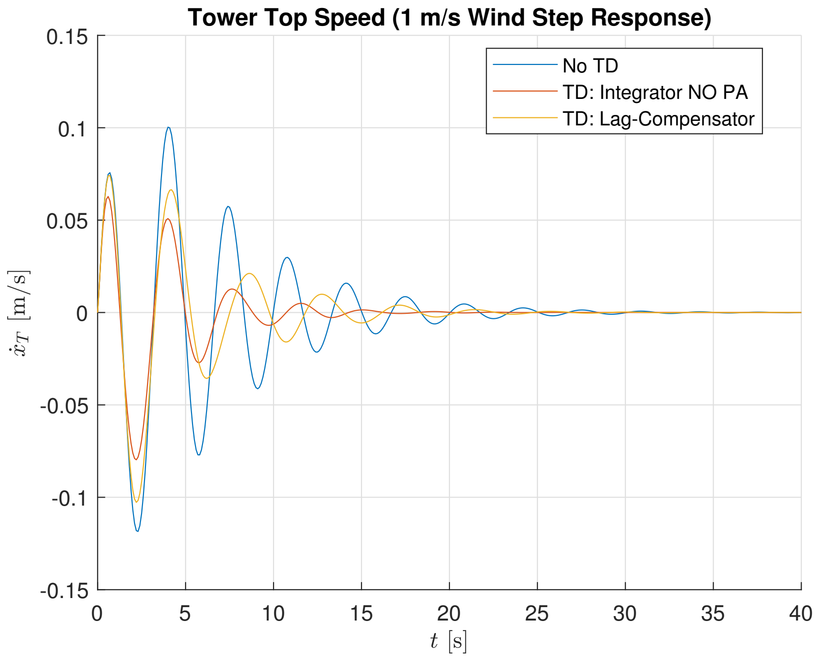

Figure 2.7 Tower-Top-Speed

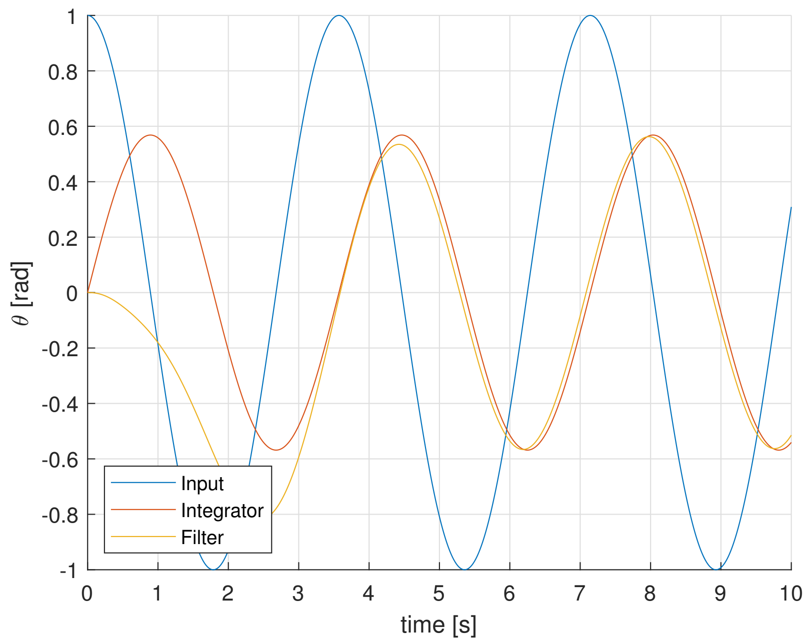

Figure 2.8 Debugging of the TD with Lag-Compensator

3 Support Tasks

This chapter explains the generation of a turbulent wind field, the Simulink model of Storage System Dummy (SSD) including various scenarios, and the tower bending stiffness.

3.1 Wind Field Generation (Felix)

To test the turbine model in Simulink with a reasonable disturbance a turbulent wind field was created. This was done based on [4] and [5].
As method the IEC Kaimal Spectral Model [4] was used. The Auto-spectrum of the rotor effective wind speed v_0 is calculated according to equation 3.1.

$$S_{\text{RR}} = \frac{S_{ii,u}}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \gamma_{ij,u} \tag{3.1}$$

The time series of the wind speed is calculated with the following simulation parameters: total simulation time T, time step Δt, random seed for reproducibility and reference wind speed U_{Ref}. The frequency range is determined based on the total simulation time and time step:

$$\begin{aligned} f_{\min} &= \frac{1}{T}, \\ f_{\max} &= \frac{1}{2\Delta t}, \\ \Delta f &= f_{\min}, \\ f &= \{f_{\min}, f_{\min} + \Delta f, \ldots, f_{\max}\}. \end{aligned}$$

With the Auto-spectrum according to equation 3.1 the amplitude $A(f)$ for each frequency component is calculated in equation 3.2.

$$A(f) = \sqrt{2 S_{\text{RR}}(f) \Delta f} \tag{3.2}$$

Random phase angles Φ are generated for each frequency component using a uniform distribution in the range $[0, 2\pi]$. The inverse Fourier transform is used to generate the time series as shown in equation 3.3.

$$\begin{aligned} t &= \{0, \Delta t, 2\Delta t, \ldots, T - \Delta t\}, \\ U(f) &= \begin{cases} 0 & \text{(DC component)}, \\ A(f) e^{i\Phi} & \text{(frequency components)}. \end{cases} \\ v_0(t) &= U_{\text{Ref}} + \mathcal{F}^{-1}(U(f)). \end{aligned} \tag{3.3}$$

For the rotor area and for a duration of $T = 3600\,\mathrm{s}$ for different reference mean wind speeds U_{Ref} the time series are calculated. To allow the storage team reasonable simulations on a timescale of several hours or days these time series are combined after a pattern of one hour mean wind speeds as input values. The corresponding turbulent wind series are then selected and combined by circular shifting the time series until the beginning of the new series is matching the end of the previous one to avoid bigger jumps in the wind speed than expected by the turbulence itself. To achieve this an algorithm is looking in the next time series for values that are near to the end of the old series within a threshold and a number of consecutive numbers to ensure the gradient of next series is not to steep.
One example of a combined turbulent time series is shown in Figure 3.1.

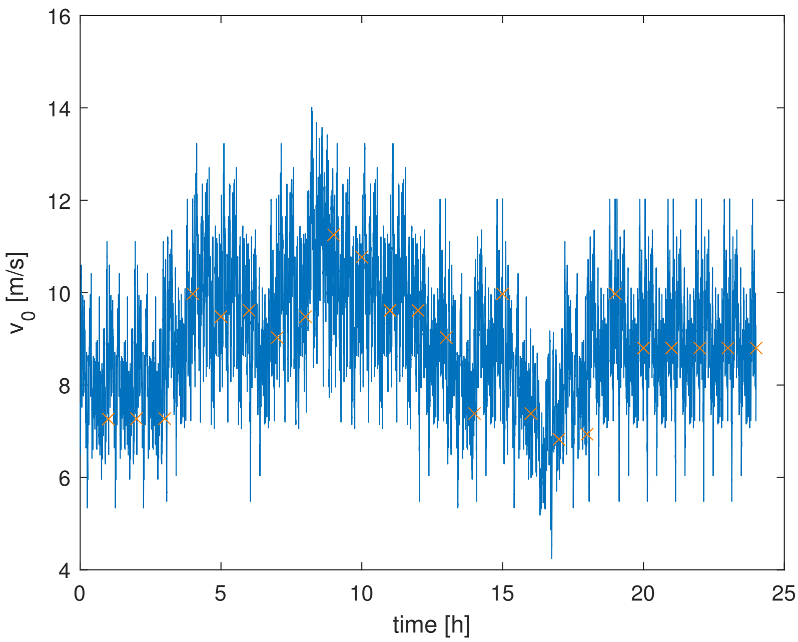

Figure 3.1 Combined time series with turbulence class B

3.2 Simple Storage System Dummy (Julius)

In order to assist the storage development team in the beginning of the project phase a simple energy Storage System Dummy (SSD) was developed and integrated into the used IEA Wind Task 37 $3.4\,\mathrm{MW}$ [1] reference wind turbine Simulink model (Figure 3.2). The further development of the storage model was executed by the storage development team.

Description

The SSD was realized by a simple Battery Management System (BMS) in combination with a integrator block. The BMS is capable to simulate the storage in 3 states *standby*, *charge* and *discharge*.

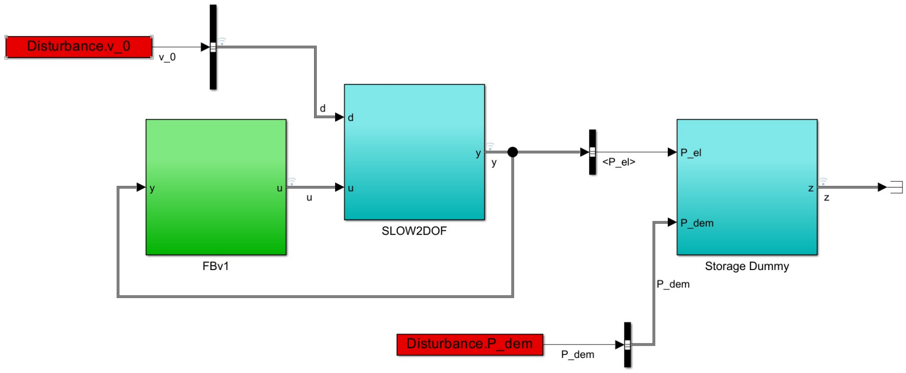

Figure 3.2 SSD in Simulink model

Scenarios

In order to explore different possibilities in which the storage system could be applied multiple scenarios where implemented into the simulation model. A curtailment event as described in scenario 5 is shown in figure 3.3.

1. **No grid power demand:** The storage system is in working condition. The storage system is not at full capacity. There is no power infeed into the grid. The storage is getting charged.
2. **Rated power demand from grid:** The storage system is in working condition. There is rated power feed into the grid. The storage is not getting charged.
3. **50 % of rated power demand from grid:** The storage system is in working condition. The storage system is not at full capacity. There is a power demand from the grid of $50\,\%$ of rated power. The storage is getting charged with a reduced rate of charge.
4. **Turbine operated below rated power, grid demand is exceeding production:** The storage system is in working condition. The storage is charged to $50\,\%$ of its maximum capacity. The WT is operating below rated power and the grid demand is higher than the power production of the WT. The storage system is getting discharged.
5. **Curtailment scenario of 4h in 25h period:** The storage system is in working condition. The storage system is not at full capacity. The WT is operating at rated power. The power must be reduced for a certain amount of time because of a curtailment order from the grid operator. The storage system is getting charged.

3.3 Tower Bending Stiffness (Julius)

For the implementation of the tower damper in the SLOW-model the tower equivalent bending stiffness k_{Te} and the initial tower top deflection x_{T0} is needed. With FAST the steady states calculations (section 4.1) with a wind speed range from $3\,\mathrm{m/s}$ to $9\,\mathrm{m/s}$ where done. The bending stiffness of the tower can be calculated with 3.4.

$$k_{\mathrm{Te}} = \frac{F_{\mathrm{a}}}{x_{\mathrm{T}} - x_{\mathrm{T0}}} \tag{3.4}$$

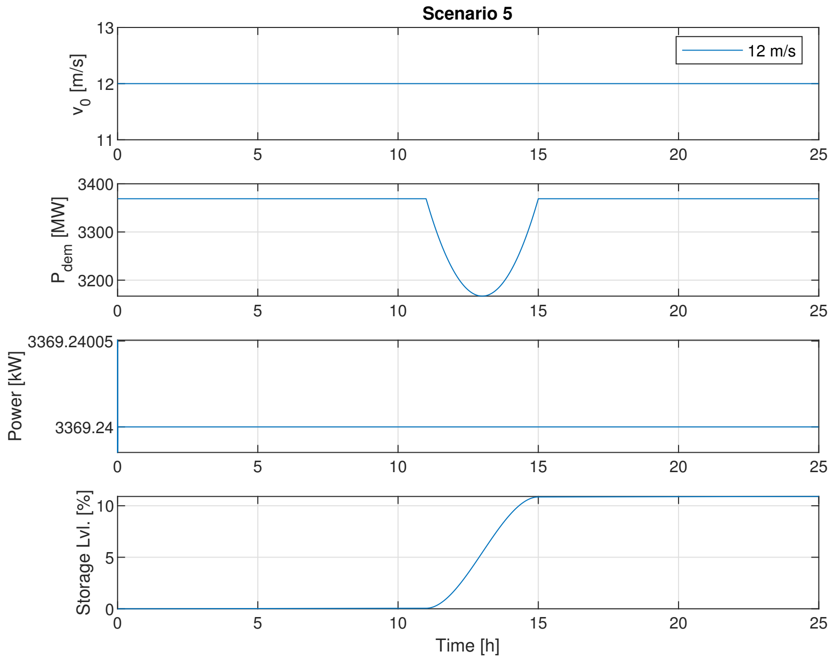

Figure 3.3 Curtailment scenario for 4h duration, a storage capacity of $5\,\mathrm{MWh}$ and a curtailment rate of $6\,\%$.

Where F_a is the aerodynamic thrust force onto the rotor plane and x_T the deflection of the tower top. The initial tower top deflection x_T0 is determined from the calculated steady states via a polyfit of the curve to get the deflection of the tower at $F_\mathsf{a} = 0\,\mathrm{N}$. With 3.4 the bending stiffness in every steady state is determined and averaged over the number of points. This leads to an tower equivalent bending stiffness of $k_\mathsf{Te} = 2.185\,\mathrm{MN/m}$ with a initial tower top deflection of $x_\mathsf{T0} = -0.021\,\mathrm{m}$.

4 Controller tuning

In this chapter the steady state calculations and the control parameter optimization are discussed.

4.1 Steady States (Soni)

The steady states are influenced by the design of the controller and turbine. The static behavior in every operating point can be shown by the calculation of steady states. Such calculated steady states provide an crucial overview of the turbines behavior early in the design process. Additionally, these steady states play an important role in the overall wind turbine design and can be utilized to initialize simulations.

The general procedure we followed to find the steady state parameters is as follows:

1. Determine the wind speeds for the control regions.
2. Calculate the steady states below the rated wind speed separately for regions 1, 1.5, 2, and 2.5. With the determined wind speeds from step 1.
3. Calculate the steady states above the rated wind speed.

With the help of the differential equation 4.1 for the rotor motion the steady states are derived.

$$\dot{\Omega} = \frac{M_a(v_0, \Omega, \theta) - M_G}{J} \tag{4.1}$$

The condition to reach a steady state is $\dot{\Omega} = 0$. The determined parameter is the variable and the other parameters are fixed. The conditions to which the fixed parameters are set depending on the control region hence the behavior of the turbine itself.

In the Shakti 5.0 simulation model, simulations are performed for wind speeds ranging from $3\,\mathrm{m/s}$ to $25\,\mathrm{m/s}$. The steady state values for wind speed, pitch angle, rotor speed, tip speed ratio, power coefficient, generator torque and tower top displacement are derived by 4.1. Additionally, the results are visually assessed through plots. The wind speeds of the control regions are $v_{\text{rated}} = 9.3531\,\mathrm{m/s}$, $v_{\text{1to1.5}} = 2.7070\,\mathrm{m/s}$, $v_{\text{1.5to2}} = 6.0652\,\mathrm{m/s}$ and $v_{\text{2to2.5}} = 8.6631\,\mathrm{m/s}$. Figure 4.1 demonstrates the model's power curve and Figure 4.2 demonstrates the model's tower displacement.

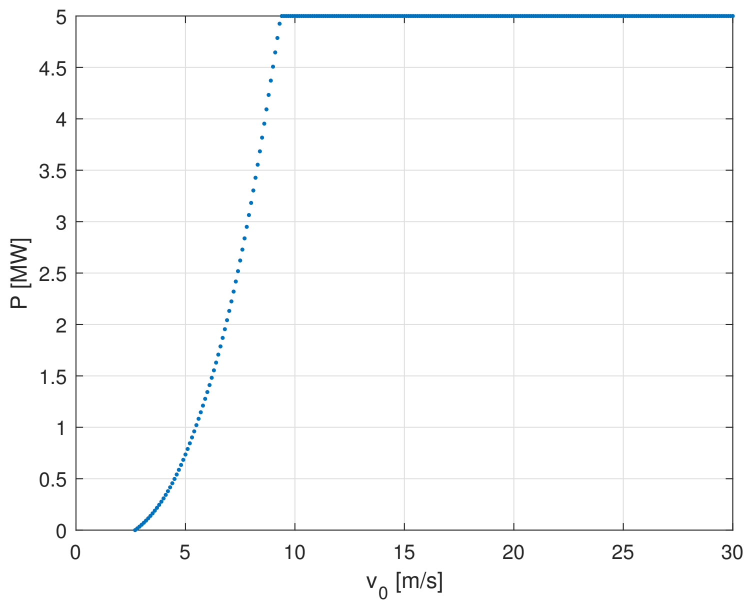

Figure 4.1 Power curve calculated with steady state calculations

4.2 DEL Calculations and Parameter Optimization (Felix)

To optimize the controller and its parameters such as k in Control-Region 2, kp for the pitch control in Control-Region 3, ΔP in Control-Region 2.5 and θ_k in Control-Region 3, a brute-force optimization pattern has been followed.

The used method runs the Design Load Case (DLC) 1.2 for the selected control parameter for a specified range of the control value. Used wind disturbance is created beforehand with the use of **TurbSim** and the *GenerateTurbSimWindFields.m* file from the *LAC SummerGames 2024* [6]. The following input parameters to **TurbSim** have been changed: The turbulence class is set to *B* and the wind-field-grid values are set to match the dimensions of the Shakti 5.0. The wind time series are created in a range of $[4:2:24]\frac{m}{s}$ with 6 different seeds per wind speed. The length of the series are $T = 600\,\mathrm{s}$. In the DLC calculation the simulation is done 6 times per wind speed for all different seeds. And for all 12 wind speeds. The total simulation time per wind speed ads up to $T_{\text{Simulation}} = 3600\,\mathrm{s}$. With this setup the over speed, life time weighted Damage Equvivalent Load (DEL) and the Annual Energy Production (AEP) is computed. The used Weibull parameter for the lifetime weighting are $C = \frac{2}{\sqrt{\pi}} \cdot 7.5$, this corresponds to the wind class III [4] and a $k = 2$. 4.2 shows how the Weibull distribution is computed in this case.

$$f(V_{\text{ref}}) = \frac{k}{C}\left(\frac{V_{\text{ref}}}{C}\right)^{k-1} \exp\left(-\left(\frac{V_{\text{ref}}}{C}\right)^{k}\right) \tag{4.2}$$

The weighting function is derived to 4.3

$$w(V_{\text{ref}}) = \frac{f(V_{\text{ref}})}{\sum f(V_{\text{ref}})} \tag{4.3}$$

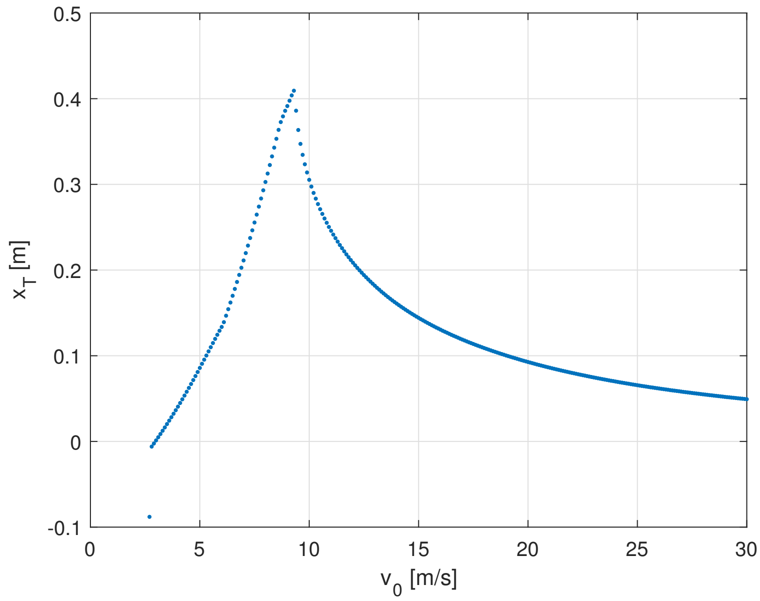

Figure 4.2 Tower displacement calculated with steady state calculations

The AEP is calculated as 4.4

$$AEP = \sum \left(\bar{P}_{\text{el}} w(V_{\text{ref}})\right) \cdot 8760\,\text{h} \tag{4.4}$$

The DEL calculation is using the parameters of the Woehler exponent as $m = 4$ hence this is the typical value for steel a reference number of $N_{\text{ref}} = \frac{2\cdot10^6}{20\cdot8760}$ as a value for 20 years for 1 hour simulations. The DEL is calculated per V_{ref} with the use of the rainflow count as DEL(V_{ref}) and the life time weighted DEL is than calculated in 4.5.

$$\text{DEL}_{\text{LTW}} = (w(V_{\text{ref}})\text{DEL}(V_{\text{ref}})^m)^{\frac{1}{m}} \tag{4.5}$$

The parameters ΔP in Control-Region 2.5 and k in Control-Region 2 are optimized first hence they are not depending on another control parameter. Figure 4.3 shows one example of how the optimization results are displayed. The chosen control parameter is $\Delta P = 4.9\,\text{MW}$ hence the AEP is higher than for $\Delta P = 5\,\text{MW}$ without a change in the life time weighted DEL.

The design value of k is derived first with the help of 4.6 based on [5].

$$k = \frac{1}{2}\rho\pi R^5 \frac{c_{\text{P,opt}}}{\lambda_{\text{opt}}^3 r_{\text{GB}}^3} \tag{4.6}$$

The value out of 4.6 is $k_{\text{design}} = 46.3035\,\text{Nm}/(\text{rad/s})^2$. This is than brute force optimized with a step size of 0.1 as described above. The resulting value k can next to the other optimized parameters be found in 7.2.

The CPC gain kp and the gain scheduling parameter θ_k are effecting not only the AEP and the DEL of the tower but also the over speed. This introduces another optimization layer. The over

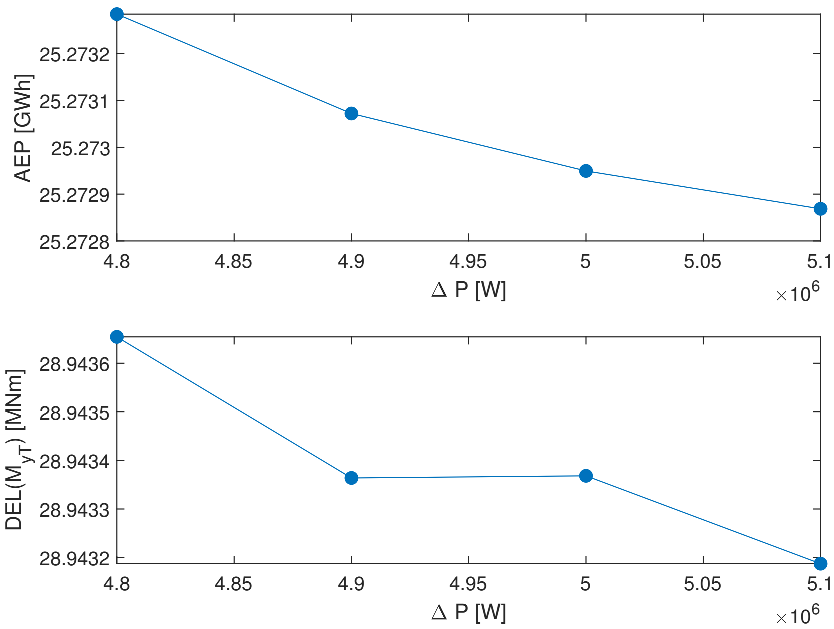

Figure 4.3 Brute-Force optimization of ΔP

speed is computed as 4.7.

$$\overset{+}{\Omega} = \frac{\hat{\Omega}}{\Omega_{\text{rated}}} \tag{4.7}$$

Due to the late design freeze the final Shakti 5.0 design leads to low over speeds (less then 10%). A new iterative optimization needs to be done in the future. The current values are all listed in 7.2.

4.3 Minimum Pitch Angle Optimization (Julius)

The optimization of the minimum pitch angle is a simple adjustment which leads to a small increase in the AEP. The optimization was done with a brute force approach and the steady states calculations (section 4.1). In control region 2 the WT should work at optimum C_p and λ. In order to operate at optimal C_p the optimal pitch angle has to be found. For different pitch angles the steady states where calculated. As optimum min. pitch angle the angle which leads to the highest C_p was chosen. As a result the min. pitch angle of $0.5°$ was determined and is shown in Figure 4.4. During the calculation the pitch angle was optimized in a range of $0°$ to $5°$ with a step size of $0.1°$.

The determined min. static pitch angle of $0.5°$ leads to an increase in AEP of $0.29\,\%$ compared to min. pitch angle of $0°$. (Calculated with Weibull parameters of TC III and $k = 2$.)

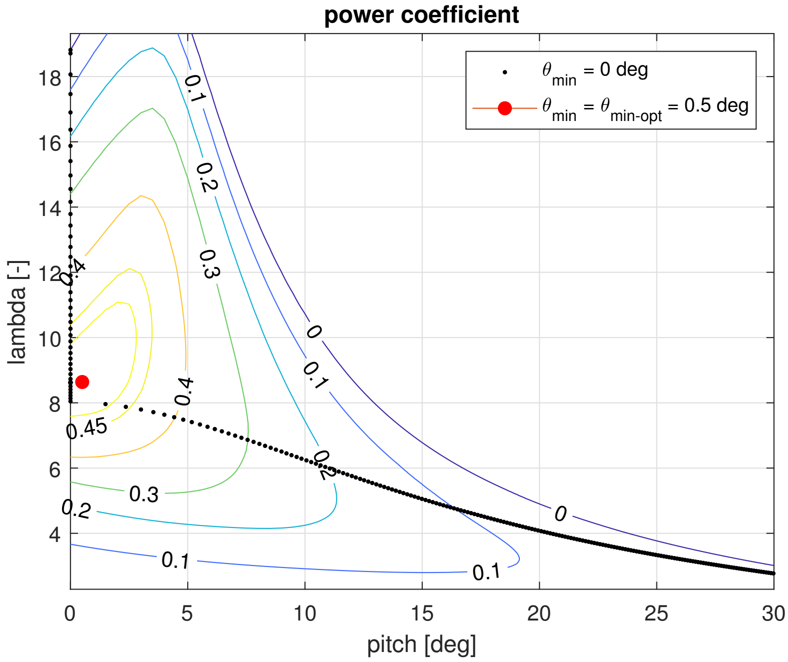

Figure 4.4 Brute force optimization for minimum pitch angle θ_{min}

4.4 Minimum Pitch Angle Optimization for Control Region 1.5 (Julius)

Since the control region 1.5 has a large wind speed range of $3.36\,\mathrm{m/s}$ the optimization of the pitch angle could lead to an increase in AEP. As optimization process a brute force approach was used in order to find the optimum pitch angle for every operating wind speed in region 1.5. During the calculation the pitch angle was optimized in a range of $0°$ to $5°$ with a step size of $0.1°$. The results of the optimization can be seen in Figure 4.5. The result shows, that keeping a static pitch angle through region 1.5 is not leading to the optimal power production. A calculation of the AEP with a dynamic pitch adjustment for region 1.5 leads to an increase of $0.19\,\%$ compared to a static minimum pitch angle of $0.5°$ as shown in section 4.3. (Calculated with Weibull parameters of TC III and $k = 2$.) Since the calculation is done without transition regions for the adjustment of the pitch the increase in AEP after implementation of the control behavior is to be expected less than the named $0.19\,\%$. The approach of changing the pitch angle dynamically in region 1.5 was not implemented in the Shakti 5.0 project but could be interesting for further optimization of the developed WT.

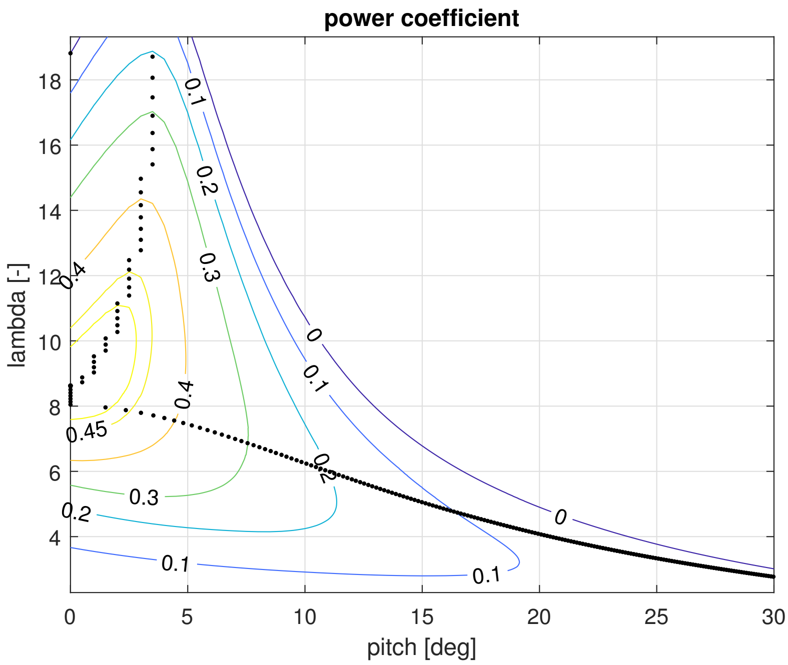

Figure 4.5 Brute force optimization for minimum pitch angle θ in region 1.5

5 Challenges, Teamwork and Lessons Learned

This chapter describes the challenged that were faced during the development project of the Shakti 5.0 wind turbine. Furthermore the lessons learned from the project and the workflow of the controls team are listed.

5.1 Rated Wind Speed (Julius)

In order to get started with the project some specifications of the turbine had to be decided really quick. In this particular case the value of the rated wind speed lead to some confusion during the development phase of the Shakti 5.0 WT. After the rotor blades where aerodynamically designed the steady state calculations revealed a problem with the rated wind speed. It was discovered that the design value of the rated wind speed was not fitting to the aerodynamic behavior of the rotor.

An investigation in cooperation with the project management lead to the result, that the decision regarding the rated wind speed was not based on the same source as rotor diameter and rated power. This lead to large mismatch in between the listed values.

The decision of the rotor diameter was based on the fixed rated power. This was calculated by using a data base of multiple WTs with similar technical specifications. From the database a power per square meter value was derived from which the rotor diameter was calculated. The C_p value and the rated wind speed was calculated based on the scaling of a power curve from a Senvion WT.

The unchecked use of the calculated power per square meter value which already contained an unknown averaged C_p value in combination with the chosen C_p and rated wind speed from the scaled power curve lead to the mismatch. In absolute numbers the difference was a calculated rated wind speed value of $9.3\,\mathrm{m/s}$ compared to the design value of $10.61\,\mathrm{m/s}$.

In order to fix the issue the 4 following options where proposed: The fixed design values at that time where: rated wind speed $10.61\,\mathrm{m/s}$, rotor radius R = $89\,\mathrm{m}$, rated power $5\,\mathrm{MW}$, C_p = 0.48.

1. Keep rated wind speed and **power**, but **reduce** rotor radius. $C_p = 0.48$ for new $R = 70\,\mathrm{m}$
2. Keep rated wind Speed and **rotor** radius, but **increase** rated power.
3. Keep rated power and rotor radius and accept new rated wind speed at ca. $9.3\,\mathrm{m/s}$.
4. Keep the design values and use the "peak shaving" method to start pitching already before region 3 in order to only reach rated power at a higher wind speed.

Since the project aim is to build a WT for low wind speed regions in collaboration with the project management and the project owner the decision was made to go with the option number 3 and accept the new rated wind speed.

5.2 Generator Speed and Control Region 2.5 (Felix)

In addition to the challenge described in the previous section 5.1 another one was discovered during the first iteration of the steady states calculation. The rated conditions where met before the control region 2.5 is reached. The cause of this was the rated generator speed of $\Omega_{\text{G,rated}} = 458\,\mathrm{rpm}$ was set too high with the aerodynamic efficiency at this time. After discovering this issue and addressing it with the electrical drive train team, a new design value was provided (see Steady States 4.1). This was done by increasing the number off poles in the generator.

The issue was not fully resolved because an infinite design loop was triggered. Hence the mechanical drive train team was not calculating the gearbox ratio r_{GB} with the $\lambda_{\text{opt}} = 8.75$ but with another design value proposed by the management: The tip speed at λ_{opt}. Which is defined as $v_{\text{Tip}} = 80\,\mathrm{m/s}$.

So after the changes of the generator speed $\Omega_{\text{G,rated}}$ a new gearbox ratio r_{GB} was calculated this would lead to the same issue described in the paragraph above. After clarification with the management the issue was resolved because the used tip speed was only a maximal value $v_{\text{Tip}} = \hat{v}_{\text{Tip}}$. The tip speed was not fixed. The resulting solution is to keep the original gearbox ratio and adjust the generator speed as described above.

5.3 Mismatch of SLOW and FAST Model (Julius)

During development and the release of the first FAST version of the WT controller the following problem was encountered. In order to validate the new controller version for FAST a simulation was done and the results were compared to the expected behavior of the turbine with the SLOW model of the developed WT. The findings were inconclusive.

The simulations showed for the same conditions, such as constant wind, no pitch activity and a matching number of DOFs a difference in power production of $16\,\%$. The FAST model showed for rated wind conditions a $16\,\%$ lower power output compared to the SLOW model.

After investigating several possibilities in SLOW and FAST the aerodynamics in FAST were identified as the cause of the mismatch. The rotor blades team provided a C_p lookup table with the software QBlade which is used by the SLOW model for deriving the correct aerodynamic power from the rotor. Since FAST was used for the load simulations the blade design in QBlade had to be exported. FAST in this case is not using a C_p look up table and is instead calculating the aerodynamics directly from the blade design. The design is input as airfoil data and the corresponding position along the blade. The load simulations in FAST were based on the AerodynV15 module. Because QBlade is only supporting the export up to version AerodynV13 the conversion from AerodynV13 to V15 was done separately by the rotor blades team. During this conversion an unidentified error occurred and lead to the faulty aerodynamics which lead to a difference in power production.

As a solution the change from the AerodynV15 module to the AerodynV14 module was approached. This was done because the conversion from the QBlade output in AerodynV13 is compatible with the AerodynV14. The change from one aerodyn version to another revealed that faulty airfoil data was exported by QBlade. The files contained random NaN values as certain key values. This problem was solved by correcting the wrong values after the export. The change from AerodynV15 to AerodynV14 were able to solve the issue and make the SLOW and FAST simulation match.

The faulty exported data was not the cause for the mismatch of AerodynV15. This problem is still unclear.

As in section 5.1 described the rated wind speed was reduced during the design process from $10.61\,\mathrm{m/s}$ to $9.3\,\mathrm{m/s}$. Because of the higher wind speed used for the FAST simulations up to that point the low aerodynamic power output was only identified quite late during the design process. With the incorrect rated wind speed the WT was apparently providing the rated power. Which in truth the turbine was operating way above wind conditions to reach rated power. Nevertheless the problem was identified in time and solved as a team effort of the loads, blades and control team. The described issue shows, that the validation of a model is from high importance and mistakes in totally different areas of responsibilities can lead to unexpected behavior later in the development phase.

5.4 Blade Masses and Generator Inertia (Julius)

Generator inertia and blade masses have significant influence on the dynamic behavior of the wind turbine. For the tuning of the controller and updating the used SLOW model the exact knowledge of above mentioned values are necessary.

During the project development phase the inertias of the wind turbine were obtained from the FAST model. The controller tuning was done with these values and showed in the weekly presentation. During the presentation the dynamic behavior of the turbine was criticized because the turbine took $90\,\mathrm{s}$ to recover from a wind step of $7.0\,\mathrm{m/s}$ to $7.1\,\mathrm{m/s}$. Investigating the cause lead to the finding that the mass of the rotor blades were not correct in the FAST model. The updated masses from the first rotor blades draft were 17-times higher than the ones before the update. These high masses caused the long reaction time ($90\,\mathrm{s}$) of the simulation model. Correcting the masses resolved the described issue and lead to a more realistic reaction time of $30\,\mathrm{s}$.

One update of the generator inertia lead to confusion because the received value was about 45-times higher than before. Checking different references lead to the result, that the given value is not correct. The developed wind turbine is equipped with a $5\,\mathrm{MW}$ generator. Similar wind turbines have a inertia of $500\,\mathrm{kgm}^2$, the first provided value was $24\,000\,\mathrm{kgm}^2$. For comparison a value for the inertia of a $33\,\mathrm{MW}$ generator is $2293\,\mathrm{kgm}^2$ [3]. Contacting the responsible departments resolved the issue quickly and lead to the final value of $782.44\,\mathrm{kgm}^2$.

Both incidents show that a plausibility check of values after receiving and before sending is important to avoid mistakes in presentations. Communication between responsible groups is key to resolving such normal issues quickly and correctly.

5.5 Lessons Learned (Julius)

In conclusion all challenges encountered during the development of the Shakti 5.0 wind turbine were properly resolved. Some key aspects that were learned from solving the described challenges in the section before are the following ones:

- **Model validation** is crucial before releasing the model/information to other participants of the project or working with a given model.
- **Plausibility check** of provided values and results either by comparing with previous or researching similar ones.
- Using **small steps** to apply changes in order to be able to keep an overview what changed in case unexpected behavior occurs.
- **Making the problem smaller** such as deactivating parts of a simulation model that are not necessary in order to isolate the fault. (e.g. the controller or multiple Degree Of Freedom (DOF))

5.6 Teamwork (Felix)

This section is describing how our team worked together during the duration of the Shakti 5.0 Project. At the beginning it is worth to mentioning that we are also the same team in the lecture "Controller Design for Wind Turbines and Wind Farms" [5] which helped a lot hence we were able to split the tasks of the lecture and the project together and to focus also on the lecture tasks because they are the main source and baseline on which we build up this project work.

Our weekly approach was to solve the lecture tasks first and as a team. The main benefit of this is that we all understanding the control theory of the task we are facing and being able to check each others work. The process in this project is not really different from the lecture work. We first meet after the weekly assembly meeting together and discuss what could be the next steps and what problems where phased during the last week. This internal meeting follows normally the weekly meeting with our supervisor Prof. Dr. David Schlipf. In this meeting new tasks and open questions were discussed. To solve the tasks the team sat together, often in front of a white board. After we understand the problem and a possible solution to it as far that the coding can begin we split the tasks and the coding is done mainly by a specific member of the team at home until someone is facing any issues. Than the problem solving loop regarding the coding starts again and is done by the whole team.

As described above it is hard to assign specific roles to specific members of the team. Nevertheless we all had our niche areas in the project where each member spent more time and effort to:

- **Julius Preuschoff:** As the elected team lead of our group Julius has to attend the team leaders meetings on a weekly basis and is therefore mainly responsible for the communication with other teams and the management specially in a case when we figured out as design issue that could not be solved only by our group. During the controller design itself Julius had his focus mainly on the steady states calculation and the SLOW to FAST comparison.
- **Karan Soni:** Soni focused mainly on the Simulink implementation of the controller and the designing and testing of the parameters as well as the baseline TD.
- **Felix Lehmann:** Felix focused mainly on the Brute-force optimization approach for the parameters and the TD using the Lag-Compensator.

To conclude and summarize the teamwork we worked really well as a team and the problems we phased where purely related to the technical work of the project and never personally.

6 Summary

This chapter contains the overall process of the Shakti 5.0 design project. Improvements and possible future work is also contained.

6.1 Conclusion (Soni)

The implementation and tuning of the WT controller is important for the overall performance of a WT. Hence it effects efficiency and lifespan of the whole WT.

The parameters for the advanced torque controller and the CPC were derived for normal operation of the turbine. Used were learned methods from the "Controller Design" lecture and optimization techniques.

Furthermore the controller tuning involves the optimization for the different control regions. This is realized via brute force optimization of the static and dynamic control parameters. For example the parameters k in control region 2, kp for the pitch control in control region 3, ΔP in control region 2.5 and θ_k in control region 3 were all brute force optimized. As shown by the example of ΔP for control region 2.5 an increase in AEP was achieved. The optimization of a static control parameter such as the minimum pitch angle θ_{min} lead to an increase of AEP as well.

In order to assist other teams the development of an SSD was done that was used by the storage team for their simulations of the battery storage system development. For the SSD it was possible to choose different behavior such as curtailment scenarios. For more realistic simulations the generation of wind field was carried out. The generated wind fields were also used by the storage team to simulate long term effects on their storage system.

During the project the control team was facing not only tasks regarding the tuning of the controller but also many interface issues during the design process of the Shakti 5.0. As described in Chapter 5 the mismatch of models, wrong values or unrealistic wind speeds were part of the development process. Resolving such challenges took more time than expected but were an important part of the learning process.

6.2 Improvements and Future Workflow (Felix)

This section begins with the possible improvements regarding the general workflow of the Shakti 5.0 project and the controller design. Due to the short time span of the project, main components like the tower or the structural design of the blades are finished late in the project. This caused that the processes implemented for optimization of control parameters could not be applied to the final design of the WT. The methods were applied instead to an earlier version of the design or a reference turbine. In order to be able to optimize the control parameters properly the design freeze of the developed turbine should be earlier in the project phase. Another important change in the future would be to start comparing models early in the development process (e.g. FAST, SLOW).

In order to continue the development of the Shakti 5.0 turbine and improve its behavior testing the tower damper within the FAST environment is necessary. Another important step would be to enter a second design loop in order to improve the interconnections between different parts of the turbine further. A really interesting feature would be to approach the connection of storage system and controller further. The interaction between controller and storage offers great opportunities to stabilize the grid with the current increase of renewable energies.

7 Appendix

7.1 Steady States (Julius)

Wind Speed [m/s]	Pitch Angle [deg]	Rotor Speed [rpm]	Power [kW]
3	0.50	5.62	52
4	0.50	5.62	309
5	0.50	5.62	735
6	0.50	5.62	1343
7	0.50	6.49	2132
8	0.50	7.41	3182
9	0.50	8.03	4507
10	4.81	8.03	5000
11	7.81	8.03	5000
12	10.06	8.03	5000
13	12.00	8.03	5000
14	13.71	8.03	5000
15	15.29	8.03	5000
16	16.79	8.03	5000
17	18.21	8.03	5000
18	19.55	8.03	5000
19	20.84	8.03	5000
20	22.09	8.03	5000
21	23.31	8.03	5000
22	24.49	8.03	5000
23	25.64	8.03	5000
24	26.77	8.03	5000
25	27.86	8.03	5000

7.2 Control Parameter (Felix)

Static Parameters		
Parameter	**Value**	**Unit**
CPC.Omega_g_rated	44.8724	[rad/s]
VSC.M_g_rated	115350	[Nm]
CPC.theta_min	0.008726646259972	[rad]
VSC.Omega_g_1d5	31.4159	[rad/s]
VSC.k	42.6	[Nm/(rad/s)^2]
Dynamic Parameters		
Parameter	**Value**	**Unit**
CPC.theta_K	0.3491	[rad]
CPC.kp	0.025	[s]
CPC.Ti	10	[s]
VSC.kp	27928.661741	[Nm/(rad/s)]
VSC.Ti	2.196201	[s]
VSC.Delta_Omega_g	4.4872	[rad/s]
VSC.Delta_theta	0.3491	[rad]
VSC.Delta_P	4900000	[W]
Other Parameters		
Parameter	**Value**	**Unit**
VSC.i_G_max	1.1	[-]
Filter.LowPass.Enable	1	[-]
Filter.LowPass.f_cutoff	1	[Hz]
Filter.LowPass2.f_cutoff	0.02	[Hz]
Filter.LowPass3.f_cutoff	0.01	[Hz]
Filter.NotchFilter.Enable	1	[-]
Filter.NotchFilter.Gain	0.030928	[-]
Filter.NotchFilter.BW	0.1	[Hz]
Filter.NotchFilter.D	0.01	[-]
Pitch Actuator		
Parameter	**Value**	**Unit**
PitchActuator.omega	3.1416	[rad/s]
PitchActuator.xi	0.8	[-]
PitchActuator.theta_dot_max	0.069813170079773	[rad/s]
PitchActuator.theta_max	1.5708	[rad]
PitchActuator.theta_min	0	[rad]

7.3 Project Order (Julius)

Project Order			
Project Name:	Optimus Shakti 5.0	Project Number:	1.1
Sub-Project	Controller Design	Project Manager	Mostafa Elbanna
Customer:	David Schlipf	Deputy / SI:	Jannik Stegert
Date:	26.10.2024	Team leader:	Julius Preuschoff

Problem Description (Reason for the Project, Strategic Purposes):	
The sub-project target is to develop a control system for a wind turbine in the range of up to 5 MW rated power for the Indian market. The Indian market requires a turbine for relatively low wind speed conditions while using a relatively high rotor area to harvest more energy from the wind. In addition, the Wind Turbine will be coupled with an energy storage unit. The to-be-designed controller of the future WT needs to be developed in accordance with these pre-set conditions to assure optimal control of the WT, which serves the highest yield of energy with the lowest loads on the turbine itself.	
Project Objectives:	
Design and implementation of the controller for the developed onhore wind turbine integrating the storage control unit's controller (will not design the storage controller)	
Provide an interface to be able to connect the wind turbine controller to the energy storage unit	
Coordinate with the energy storage development team the development of the interface connection	
Interface work between energy storage dev. Team and load & simulation team	
Improving base controller model in regards to the special requirements of the project	
Organisation (Committees, People, Responsabilities):	
Steering Committee:	David Schlipf, Mostafa Elbanna, Jannik Stegert
Project Team:	3rd semester 'Master Wind engineering' 2024/2025
Team leaders:	Julius Preuschoff
Sub-Project Team:	Julius Preuschoff, Felix Lehmann, Karan Soni
Dates, Milestones	
Start of Project:	24.9.2024
Overview and research	15.10.2024
Supplying storage group with simple WT model	29.10.2024
Improvement of base controller	11.11.2024
Simulation and testing	20.12.2024
End of Project	10.1.2024
Final report	23.12.2024
Final presentation / end of project:	28.1.2024
Restrictions:	
No strategy for the energy storage system is chosen yet -> ontroller design effort therefore is not fully predictable	
Risk Management (Which risks may occur, how to manage it?)	
Risk: Lack of knowledge	Measure: Research on topic
Risk: Unmotivated team member	Measure: Encourage the team
Risk: Lack of research and documentation	Measure: Ask supervisors for references
Reporting:	
Weekly presentation	
Weekly meeting with tutor	
Weekly meeting with group members	
Final report	
Final presentation	
Team leader:	**Head of Project:**
Date: 28.10.2024	Date: 05.11.2024
Signature: J. Preuschoff	Signature:

References

[1] P. Bortolotti, H. C. Tarres, K. Dykes, K. Merz, L. Sethuraman, D. Verelst, and F. Zahle. “IEA Wind TCP Task 37: Systems Engineering in Wind Energy - WP2.1 Reference Wind Turbines”. In: *IEA Wind* (2019), p. 136.

[2] T. Burton, N. Jenkins, D. Sharpe, and E. Bossanyi. *Wind Energy Hanbook*. New York, USA: John Wiley a Sons, 2011.

[3] A. Gloe, P. D. C. Jauch, and T. Räther. “Grid Support with Wind Turbines: The Case of the 2019 Blackout in Flensburg”. In: *Energies* 14.6 (2021), p. 1697. ISSN: 1996-1073. DOI: 10.3390/en14061697. URL: https://www.mdpi.com/1996-1073/14/6/1697.

[4] IEC 61400-1. *Wind turbines - Part 1: Design requirements*. International Electrotechnical Commission, 2005.

[5] D. Schlipf. *Lecture: Controller Design for Wind Turbines and Wind Farms*. 2024.

[6] D. Schlipf, F. Guo, Z. Liang, M. Pitter, J. L. Queija, J. Rinker, A. Venkatachalapathy, and S. Weich. *IEA Wind Task 52 LAC SummerGames 2024*. [Online; from 4. January 2024]]. 2024. URL: https://github.com/IEAWindTask52/LidarAssistedControl.