# Robust Control [SC42145]

# Practical Assignment: Control Design for a Floating Wind Turbine

Delft Center for Systems and Control
Delft University of Technology

26 October 2020

## Introduction

This document gives a description of the requirements associated with the practical assignment for the *Robust Control* course [SC42145]. In order to complete the assignment you need to submit three reports, each giving a comprehensive overview of the results together with the interpretation of these results. Groups of a maximum of two students are allowed for completing this assignment. You should register your group on Brightspace. Each of the reports have to be submitted together with the necessary MATLAB files (please create a RAR/ZIP archive) via Brightspace as a group submission before the deadline given on Brightspace. Each of the reports should also be uploaded to `https://peer.tudelft.nl` for the peer review a week after the deadline. Please, do not email questions directly to the instructors, but use Discord, to avoid duplicate questions.

### Outline

In this assignment you will investigate the challenges and limitations in designing a control system for a floating wind turbine (Figure 1) for normal operation. In this assignment you will be asked to design several controllers and to evaluate them in the frequency and time domain. On Brightspace you will find a MATLAB file which contains the state-space model of a floating wind turbine. Wthin this MATLAB file you will also find 10 minutes of wind data, this is needed for the exercises in Section 2.1. A full description of the operational parameters of the system is given in Table 1.

Figure 1: An example of a Floating Wind Turbine (FWT)

## Model Description

The linear state-space model has been derived using a first principles modeling procedure and it corresponds to a specific linearisation point. The parameter with respect to which the linearisation is performed is the wind speed ($V_{lin} = 16$ $[m/s]$). We refer to [3] for a more detailed model description. The model has three inputs (namely the blade pitch angle $\beta$ $[rad]$, the generator torque $\tau_e$ $[Nm]$ and the wind speed $V$ $[m/s]$), of which only the first two are control inputs. The third input will be used as a disturbance input around the linearisation point and it can be further considered that its variation is best described by a unit step signal. The model also has two output channels which correspond to a specific choice of measured quantities in this floating wind turbine, namely the generator speed $\omega_r$ $[rad/s]$ and the fore-aft tower top displacement $z$ $[m]$. In the provided MATLAB model the internal states employed to capture the dynamics of the system are $x_1 = \omega_r$, $x_2 = \dot{z}_1$, $x_3 = z_1$, $x_4 = \dot{z}_2$, $x_5 = z_2$, where $z_{1,2}$ are a decomposition of the fore-aft tower displacement, corresponding to the tower bending and platform tilting modes respectively. Figure 2 display a schematic view of fore-aft movement for a floating wind turbine. The inputs of the MATLAB model are ordered as $u_1 = \beta$, $u_2 = \tau_e$, $u_3 = d = V$.

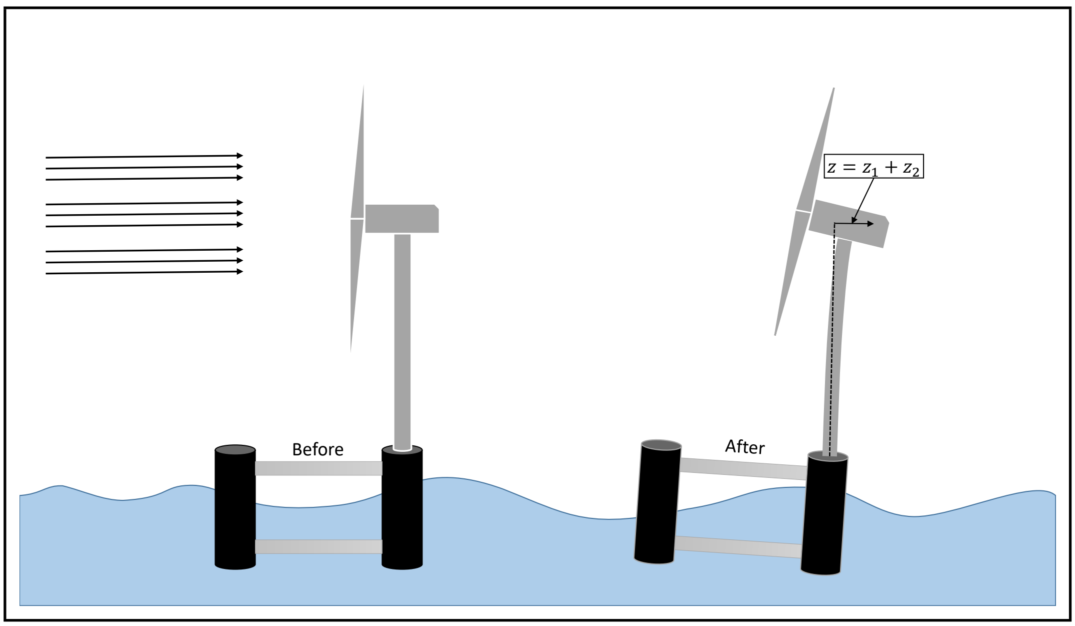


Figure 2: Schematic depiction of fore-aft movement.

The regulation problem when the rated power of 1.8 $MW$ is produced deals with keeping the output power $P_o = \tau_e \cdot \omega_r$ constant by pitching the blades. We are also interested in scenarios where the power demand changes and use blade pitching to drive the generator to a new operating point. More explicitly [4], we can pitch the turbine blades in order to control the rotor speed. An attempt will be made at also using another control approach, namely a multi variable one, as it will be described in the subsequent steps.

| | |
|---|---|
| Rated Power | 1.8 $[MW]$ |
| Rated Torque | $10^4$ $[Nm]$ |
| Rated Speed | 180 $[rad/s]$ |

Table 1: Operational parameters for the FWT

# Part 1 SISO Analysis and Control Design

**Due: 27-Nov-2020**

For the first part of this assignment you will design a SISO controller for the floating wind turbine. As the wind turbine is in operation the power demand is increased and a new rotational velocity target ($\omega_r$) for the generator is set. The goal of the SISO controller is to pitch the blades to increase the rotational velocity of the turbine and acquire the new desired generator speed. To this end, your task is to design a reference tracking controller that achieves the highest possible bandwidth whilst remaining within the design requirements. You can use any of the control actions covered in lecture 1 (P, PI, PD, PID, Lead-Lag, Notch, etc). The design requirements are:

1. Small settling-time.
2. Overshoot $< 1\%$.
3. No steady state error.

Your report should cover the design process you went through and explain how you arrived at your controller. *Please make sure that you design a physically realizable controller (i.e., not a textbook version with an ideal D term). Furthermore, completely automated, brute force, or random search for the controller parameters will not be appreciated.*

The following steps will help you on your way in your control design.

1. Take a look at the open loop Bode plot and the pole-zero map of the SISO system. Are there any features of the system that might hinder achieving a high bandwidth ($> 0.1Hz$), and if so why do they make it difficult?

2. Translate the design requirements into requirements on the open loop bode diagram and/or complementary sensitivity function.

3. Perform time-domain simulations of your closed-loop system (please see a list of useful Matlab commands at the end of this document, you can also use Simulink if you see fit) and test the performance of your SISO controller. For the reference input use a step input. Plot and evaluate the simulation results and highlight your achieved performance specifications (bandwidth, phase margin, settling time etc.). The Matlab command `minreal()` can be useful to increase simulation performance.

4. Consider that we use your SISO controller for disturbance rejection on the output of the system. Without changing the controller, perform the same time-domain simulations as in the previous question. For this simulation apply the disturbance step-input on the third input channel of the state space. Would you change your controller for the disturbance rejection scenario?

# Part 2 Multi variable Mixed-Sensitivity

**Due: 18-Dec-2020**

Now we consider both channels as control inputs: ($\beta$ and $\tau_e$), and both outputs of the system: $\omega_r$ and $z$. Even though now you can use $\tau_e$ as a second controllable input, we want to keep the static power curve unchanged. Therefore, this control input is not allowed to have a static (steady-state) contribution (i.e. a unit step change in the wind disturbance channel should still result in $\lim_{t\to\infty} \tau_e(t) \to 0$). It is also important to look at Table 1 to understand that the physical system is subject to some limitations.

You are now asked to design a Mixed-Sensitivity centralized controller [1, pages 94-96]. You should **perform and explain** the following exercises:

1. Compute the RGA for $\omega = 0$ and $\omega = 0.4 \times 2 \times \pi$. What can you conclude?

2. Compute the MIMO poles and zeros. What can you conclude?

3. Design an appropriate $W_{p11}$ (see Table 2), for a cut-off frequency of 0.4 Hz, to get an attenuation of low-frequency disturbances of $10^{-4}$, and an $\mathcal{H}_\infty$ norm of the sensitivity function of 1.8.

4. Draw a block diagram with the model of the floating wind turbine and controller, and include the performance weights $W_p$ and $W_u$ (see Table 2) in this block diagram. Define all signals that are relevant for the mixed sensitivity problem (use negative feedback). **Define the performance signals $z_1$ and $z_2$ as the signals coming out of $W_p$ and $W_u$ respectively**

| | |
|---|---|
| $W_p$ | $\begin{bmatrix} W_{p11} & 0 \\ 0 & 0.2 \end{bmatrix}$ |
| $W_u$ | $\begin{bmatrix} 0.01 & 0 \\ 0 & \frac{5\cdot10^{-3}s^2+7\cdot10^{-4}s+5\cdot10^{-5}}{s^2+14\cdot10^{-4}s+10^{-6}} \end{bmatrix}$ |

Table 2: Performance weights

5. Derive a generalized plant of the previously designed block diagram, and analyse it, i.e. assign the variables and give the mathematical description. You can use Matlab, to check if the number of states are as expected. When using Matlab make sure to use the command `minreal()` to get a minimal realization.

6. Give an interpretation of the given performance weights. Try to relate this to the control design targets explained in the introductory text of this part.

7. Using the generalized plant you found, design a mixed-sensitivity generalized controller (hint: use `hinfsyn` in MATLAB). Is the closed-loop system internally stable while using this controller (look at the generalized Nyquist plot)? How many states does the controller have? and the generalized plant?

8. Perform the same time-domain (reference tracking and disturbance rejection) simulations as you did before. Based on the results discuss the fundamental limitations of the design.

## 2.1 MIMO Weighting Design

In this section we will explore an alternative way to design controllers using the robust control toolbox. For this section we consider the wind turbine running at its rated power, that is, the maximum power it can output. Any change in rotational velocity that is caused by a change in wind speed needs to be counteracted by either a change in blade pitch angle $\beta$ or generator torque $\tau_e$. Figure 3 shows a typical windprofile the wind turbine encounters during operation. We recognise a low frequent part (a sin wave with a period of around 1000 seconds) and some high frequency turbulence.

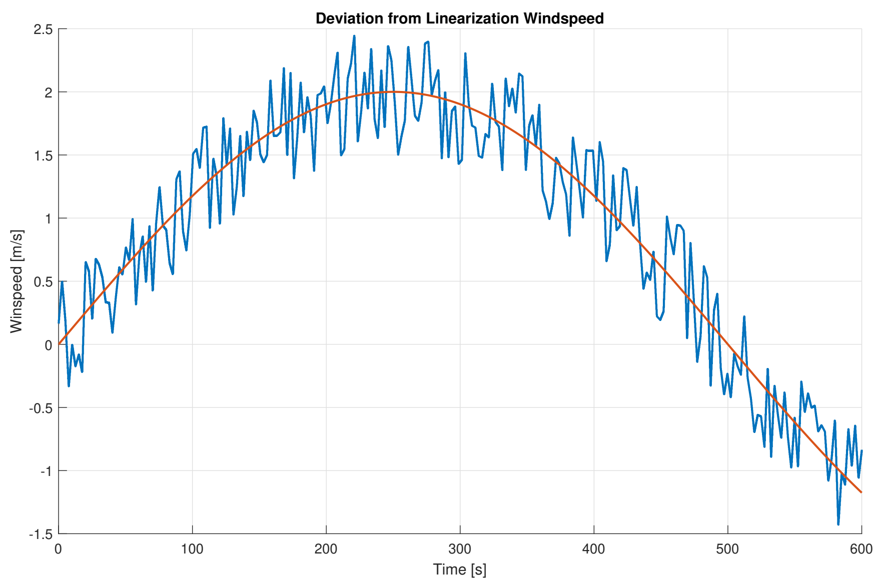


Figure 3: Theoretical wind profile for 10 minutes as experienced by the wind turbine.

Of the two control elements on the wind turbine, the torque $\tau_e$ is best used to counteract high frequency changes in wind speed as blade pitching is too slow. This has the downside (as we saw previously) that the power of the wind turbine is altered for a short period of time. Blade pitch control is ideal for low frequency control and doesn't affect power production of the wind turbine. Your task is to design the control weights (2 weights) and error weight (1 weight) such that when your system is subjected to the input of Figure 3 the torque control input is only used for the high frequency inputs. *For this part of the assignment we will only take $\omega_r$ as the output of the system.* The following exercises will help you on your way:

1. Draw the generalized plant for this system. How many inputs and outputs does the plant have?

2. Design the weight on the error (a first order weight suffices). Explain how you arrived at your weight design. (hint: Try to relate the dB gain of your weight to how the input signal of the transfer function is related to the output signal.)

3. Do the same for the controller weights (for these weights a first order weight is also sufficient), where each of the control inputs is weighted independently.

4. Synthesize your controllers with your weights using the `hinfsyn()` command. Plot the controllers bodes versus their weights. Also plot the error transfer function with respect to its weight.

5. Run a simulation with the windspeed input given on Brightspace, plot the control action of both blade pitch and generator torque. Do the controllers work as intended (blade pitch for low frequency, generator torque for high frequency)? You can check this by applying a noiseless version of the sin wave to your system.

# Part 3 Robust Analysis and Controller Design

**Due: 15-Jan-2021**

In the third part of the assignment we start from the supplied (nominal) model and derive a perturbed (uncertain) model, followed by an analysis of the robustness associated with the previous controller design. The perturbations are characterised by $\Delta$ which belongs to a certain set, i.e. $\Delta \in \underline{\Delta}$.

Including this uncertainty in the model results in a (infinite) set of models describing the behaviour of the floating wind turbine. Stability and performance analysis should hence be done for a set of models and not for one single (nominal) mode, like you have done in the first computer session. We call stability and performance analysis of a set of models analysis of the Robust Stability (RS) and analysis of the Robust Performance (RP) respectively.

In this third part you will use the Robust Control Toolbox in MATLAB to perform a robustness analysis on an uncertain model of the floating wind turbine when using the Mixed-Sensitivity controller you designed in the first computer session.

## 3.1 Analysis of Robust Stability (RS) and Robust Performance (RP)

In this part of the assignment we are first going to add input and output multiplicative uncertainty to the nominal model describing the dynamical behaviour between the inputs $\beta$ and $\tau_e$ and the outputs $\omega_r$ and $z$. You will use the numerical uncertainty weights given in Table 3. We define the weight on the input uncertainty as $W_i = \texttt{diag}(W_{i_1}, W_{i_2})$ and the weight on the output uncertainty as $W_o = \texttt{diag}(W_{o_1}, W_{o_2})$. Their corresponding perturbation blocks are $\Delta_i = \texttt{diag}(\delta_{i_1}, \delta_{i_2})$ and $\Delta_o = \texttt{diag}(\delta_{o_1}, \delta_{o_2})$ respectively. Furthermore, we have that $\delta_{i_1}, \delta_{i_2}, \delta_{o_1}, \delta_{o_2}$ are all complex scalars and their absolute value is bounded by 1. Define the uncertainty block $\Delta = \texttt{diag}(\Delta_i, \Delta_o)$. You should **perform and explain** the following steps:

1. Draw a block diagram with the model of the floating wind turbine and controller and include the performance weights $W_p$ and $W_u$ (Table 2) and uncertainty weights $W_i$ and $W_o$ (Table 3) in this block diagram. Define all signals and use negative feedback.

2. Derive a mathematical expression for the new generalised plant with performance and uncertainty weights as given in Table 2 and Table 3 respectively [1, section 8.3].

3. Plot the frequency responses of the uncertainty weights and interpret them accordingly. Use [1, section 7.2, 7.4, 8.2] to see how these weights might have been derived and explain your findings. In addition, plot the singular values of the uncertain transfer matrix describing the dynamic

behaviour between the inputs $\beta$ and $\tau_e$ and the outputs $\omega_r$ and $z$. You can use `ultidyn` in Matlab to create the perturbation blocks.

4. Use the uncertainty and performance weights with numerical values as given in Table 3 and 2 respectively and program the generalized plant in MATLAB (hint: you can augment the generalized plant code from the nominal design).

5. Use the mixed-sensitivity controller found in the first computer session and the formulated generalized plant found in this computer session and check for nominal stability (NS) using the generalized Nyquist criterion, nominal performance (NP), robust stability (RS) and robust performance (RP) using the Structured Singular Value (SSV). See [1, sections 8.5-8.10] for the necessary background information. For the NP, RS and RP tests, you should make corresponding statements about the obtained values for $\mu$.

6. How does the value of $\mu$ for NP relate to the $\mathcal{H}_\infty$ norm of the objective function in the mixed-sensitivity design? Write down your results and explain them.

| | |
|---|---|
| $W_{i_1} = \dfrac{\frac{1}{16\pi}s + 0.3}{\frac{1}{64\pi}s + 1}$ | $W_{i_2} = \dfrac{\frac{1}{16\pi}s + 0.3}{\frac{1}{64\pi}s + 1}$ |
| $W_{o_1} = \dfrac{0.05s + 0.2}{0.01s + 1}$ | $W_{o_2} = \dfrac{0.05s + 0.2}{0.01s + 1}$ |

Table 3: Uncertainty weights

## 3.2 Robust Controller Design

In this final section of the assignment you are asked to use the same settings as in the previous one and to synthesize a controller that achieves robust stability (RS) and robust performance (RP). For this use the D-K iteration procedure as shown in [1, section 8.12]. The MATLAB script from the corresponding lecture slides [2] is also useful for implementing this procedure. You can use `dksyn` in MATLAB. We expect the following items in your report

1. Give and explain the symbolic expression for the $D$ matrices used in the DK-synthesis for this particular problem.

2. Design a robust controller using DK-iterations. Once synthesized, check for NS, NP, RS and RP using the structured singular values (SSV).

3. Perform the same time-domain simulation scenarios again with the synthesized controller and compare the results with the previously designed MIMO controllers. Explain the differences you see in the results using relevant frequency-domain plots.

| | |
|---|---|
| **lsim** | simulate time response of dynamic system to arbitrary inputs |
| **minreal** | minimal realization or pole-zero cancellation in a linear system |
| **lft** | generalized feedback interconnection of two models |
| **series** | series interconnection for two models |
| **sysic** | create interconnection structure for certain and uncertain matrices/systems |
| **append** | group models by appending their inputs and outputs |
| **norm** | norm of linear model |
| **ultidyn** | create uncertain LTI object |
| **hinfsyn** | compute $H_\infty$-optimal controller for LTI plant |
| **mussv** | calculate upper and lower bounds for the structured singular value of a block |
| **fitmagfrd** | fit frequency response magnitude data |
| **mussvextract** | extract *muinfo* structure returned by *mussv* |
| **frd** | create frequency response data model |
| **fnorm** | pointwise peak gain of FRD model |

Table 4: Recommended MATLAB functions

# References


[1] S. Skogestad and I. Postlethwaite - *Multivariable Feedback Control: Analysis and Design*, John Wiley and Sons, Chichester, England, 2nd Edition, 2005.

[2] J.W. van Wingerden - *Lecture Slides on Robust and Multivariable Control [SC4015]*, Technische Universiteit Delft, 2015.

[3] G.J. van der Veen, I.A. Couchman and R.O. Bowyer - *Control of Floating Wind Turbines*, American Control Conference, Montreal, Canada, 2012.

[4] J.M. Jonkman - *Influence of Control on the Pitch Damping of a Floating Wind Turbine*, ASME Wind Energy Symposium, Reno, Nevada, U.S.A., 2008.