Baseline Generator Torque Controller

Prof. Dr.-Ing. David Schlipf

26.09.2022

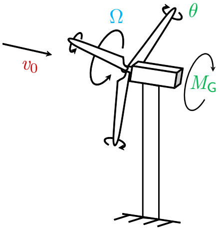
Lecture #2 of Course
"Controller Design for Wind Turbines and Wind Farms"

Hochschule Flensburg
University of Applied Sciences

SWE
Stuttgart Wind Energy
@ Institute of Aircraft Design

Wind Energy Technology Institute

Baseline Generator Torque Controller



Motivation

- Generator torque is one of the two main control inputs.
- Generator toque controller is enabled most of the time.
- Can be implemented in a quite simple form (baseline).

$\rightarrow$ Good point to get started with our controller!

Main questions

- How can we design a baseline generator torque controller?
- How can we and how should we evaluate a controller?

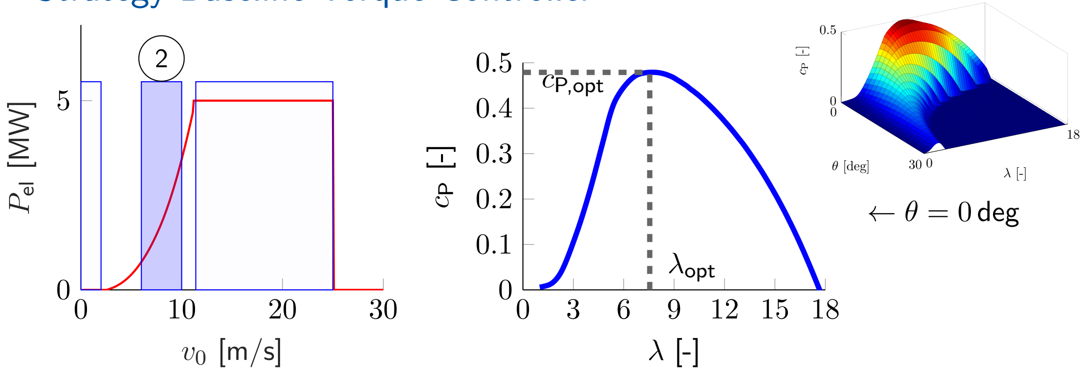
Schedule

19.09.	1	Controller Design Objectives and Modeling
26.09.	2	Baseline Generator Torque Controller (online)
10.10.	3	Collective Pitch Controller (online)
17.10.	4	Filter Design (online)
24.10.	5	Tower Damper
07.11.	6	Advanced Torque Controller
14.11.	7	Wind Field Generation
21.11.	8	Steady State Calculations
28.11.	10	Lidar-Assisted Control I
05.12.	11	Lidar-Assisted Control II
12.12.	12	Wind Farm Effects
19.12.	13	Wind Farm Control
09.01.	14	Floating Wind Control ???

Content

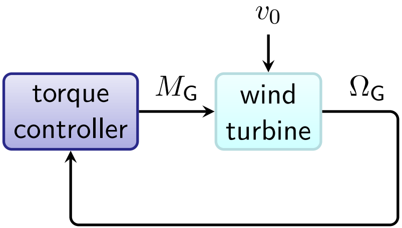
1. Baseline Torque Controller Design
2. Evaluation of a Controller
3. Conclusion and Learning Objectives

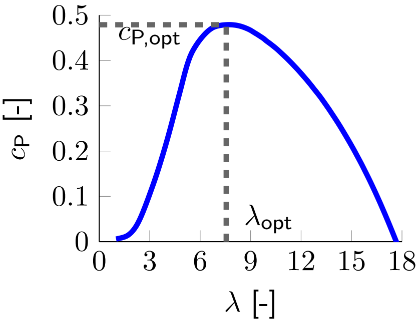
Strategy Baseline Torque Controller



- Torque controller in region 2 aims to maximize $P_{\mathrm{el}} = \frac{1}{2}\rho\pi R^2 c_{\mathrm{P}}(\lambda,\theta)v_0^3\ \eta_{\mathrm{el}}$.
- Optimum $c_{\mathrm{P,opt}}$ is reached for $\lambda = \lambda_{\mathrm{opt}}$ and $\theta = 0\,\mathrm{deg}$.
- Rotor-effective wind speed v_0 is not measurable.
- Indirect Speed Control: Feedback $M_{\mathrm{G}}(\Omega_{\mathrm{G}})$ can be designed such that $\lambda \rightarrow \lambda_{\mathrm{opt}}$.

Optimal Torque Control





Optimal feedback

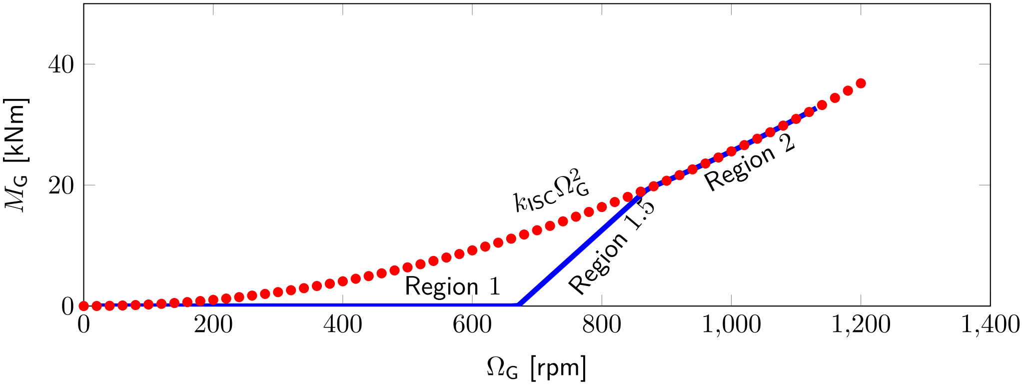
$$J\dot{\Omega} = \boxed{M_a} - \frac{M_G}{i_{GB}} \stackrel{!}{=} 0$$

$$M_a = \frac{1}{2}\rho\pi R^2 \frac{c_{P,opt}}{\boxed{\Omega}} \boxed{v_0}^3, \quad \Omega = \Omega_G\, i_{GB}, \quad v_0 = \frac{\Omega R}{\lambda_{opt}}$$

$$\rightarrow \quad M_G = \underbrace{\frac{1}{2}\rho\pi R^5 \frac{c_{P,opt}}{\lambda_{opt}^3} i_{GB}^3}_{k_{ISC}} \Omega_G^2$$

Proof of convergence and stability, see [1].

Nonlinear State Feedback I

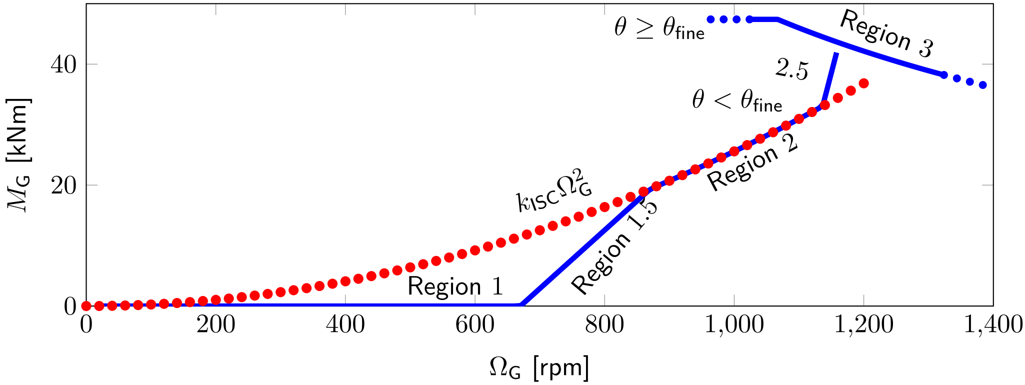


Region 1:
$\Omega_G < \Omega_{1to1.5}$
$M_G = 0$

Region 1.5:
$\Omega_{1to1.5} \leq \Omega_G < \Omega_{1.5to2}$
$M_G = a_{1.5}\ \Omega_G + b_{1.5}$

Region 2:
$\Omega_{1.5to2} \leq \Omega_G < \Omega_{2to2.5}$
$M_G = k_{ISC}\ \Omega_G^2$

Nonlinear State Feedback II



Region 2.5:

$\Omega_{2to2.5} \leq \Omega_G$ and $\theta < \theta_{\text{fine}}$

$M_G = a_{2.5}\ \Omega_G + b_{2.5}$

Region 3:

$\theta \geq \theta_{\text{fine}}$

$M_G = \min(P_{\text{a,rated}}/\Omega_G, M_{\text{G,max}})$ or $M_G = M_{\text{G,rated}}$

Transition Regions

Region 1.5

Necessary to reach region 2 from region 1. Usually $\Omega_{\text{1to1.5}}$ and $\Omega_{\text{1.5to2}}$ are given:

$$0 = a_{1.5}\ \Omega_{\text{1to1.5}} + b_{1.5}$$
$$k_{\text{ISC}}\ \Omega^2_{\text{1.5to2}} = a_{1.5}\ \Omega_{\text{1.5to2}} + b_{1.5}$$

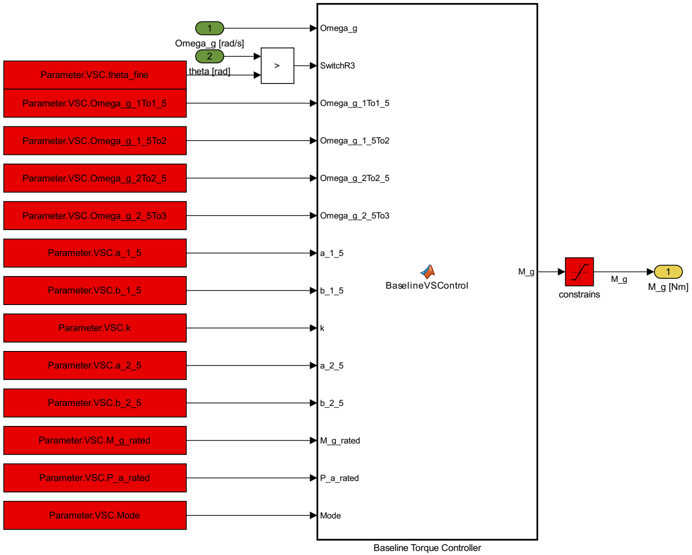
Region 2.5

Necessary to reach region 3, if rated generator speed is reached before rated torque is reached. Otherwise torque is held constant. If $\Omega_{\text{2to2.5}}$ and $\Omega_{\text{G,rated}}$ are given:

$$M_{\text{G,rated}} = a_{2.5}\ \Omega_{\text{G,rated}} + b_{2.5}$$
$$k_{\text{ISC}}\ \Omega^2_{\text{2to2.5}} = a_{2.5}\ \Omega_{\text{2to2.5}} + b_{2.5}$$

Alternatively, $\Omega_{\text{2.5to3}}$ slightly below $\Omega_{\text{G,rated}}$ and a generator-slip percentage can be given [2].

Implementation in Matlab Simulink I



Implementation in Matlab Simulink II

```
function M_g = BaselineVSControl(Omega_g,SwitchR3,...    % signals
    Omega_g_1To1_5,Omega_g_1_5To2,...
    Omega_g_2To2_5,Omega_g_2_5To3,...                    % region limits
    a_1_5,b_1_5,...                                      % Region 1.5 parameters
    k,...                                                % Region   2 parameters
    a_2_5,b_2_5,...                                      % Region 2.5 parameters
    M_g_rated,P_a_rated,Mode)                            % Region   3 parameters

if  Mode == 1                                            % Power constant
    M_g_3 = P_a_rated/Omega_g;
else                                                     % Torque constant
    M_g_3 = M_g_rated;
end

if      Omega_g_2_5To3  <   Omega_g                      % Region 3
    M_g     = M_g_3;

elseif  Omega_g_2To2_5  <   Omega_g                      % Region 2.5
    M_g_2_5 = a_2_5 * Omega_g + b_2_5;
    M_g     = SwitchR3*M_g_3+(1-SwitchR3)*M_g_2_5;

elseif  Omega_g_1_5To2  <   Omega_g                      % Region 2
    M_g_2   = k * Omega_g^2;
    M_g     = SwitchR3*M_g_3+(1-SwitchR3)*M_g_2;

elseif  Omega_g_1To1_5  <   Omega_g                      % Region 1.5
    M_g = a_1_5 * Omega_g + b_1_5;

else                                                     % Region 1
    M_g = 0;
end
```

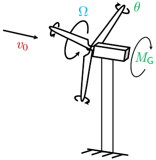
Evalution Steps and Purpose

1. Step response of controller design model (e.g. Simulink):
   Does the controller exactly what it should or is there an implementation error?
2. Step response of aero-elastic tool (DLL: Dynamic Link Library):
   How does model uncertainties impact the behavior?
3. Load Simulations following standards (DLL):
   How does the controller perform over a wide range of conditions?
4. Software/Hardware-in-the-loop tests (PLC: Programmable Logic Controller):
   How does the controller perform under realistic conditions?
5. Field Testing (PLC):
   How does the controller perform in reality?

- Controller should be initialized with current states to avoid long settling times (simulations) and bump-free switching (reality).
- If possible use one source code for all steps, e.g. libraries to avoid translation errors.

Why do we need load simulations?

Simulation



Reality

Load simulations help during all design phases:

- Which tower height should be used? → Conceptual design
- Which loads affect the components? → Design
- Are the loads within design limits? → Certification
- How can the system be optimized? → Optimization

Benefits over real tests

- simulations have low costs
- simulations are often fast
- simulations allow us to study easily many configuration
- simulations are reproducible
- however: models and tools need to be validated

Simulation Process in General

Pre-Processing

- specification of all model, controller and simulation parameters
- specification of all external input signals (wind and sea states)
- performing calculations, which are needed as input for simulations

→

Processing

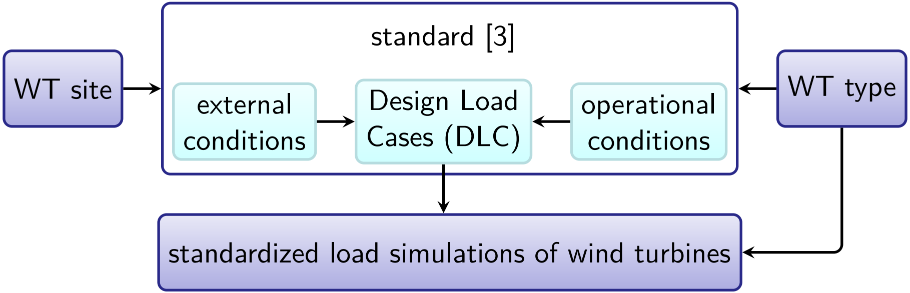
performing simulations using the defined algorithms, parameters and input signals

→

Post-Processing

- calculation of statistics and fatigue loads
- calculation of spectra
- plotting results etc.

Standardized load simulations for wind turbines



External conditions

- wind conditions
- other environmental inputs
- electrical power network

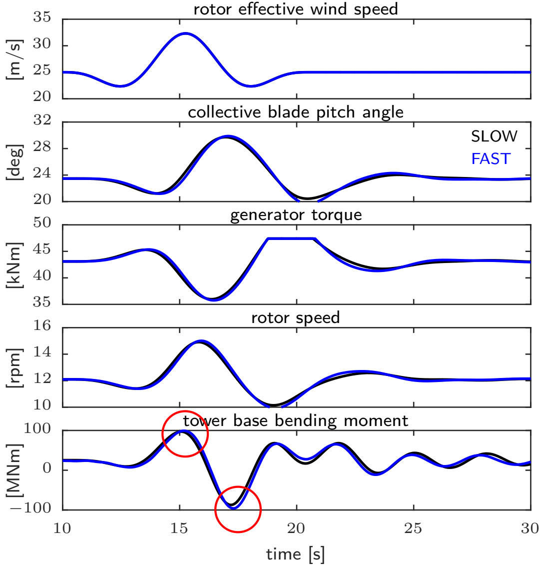
Operational conditions

- power production (+ fault)
- start up, shut down
- parked, transport

Type of analysis

- fatigue load analysis
- ultimate load analysis

Example: ultimate load simulation



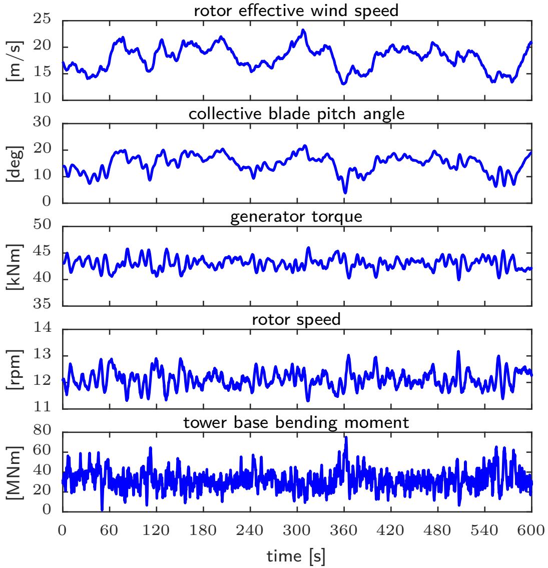
Extreme Operating Gust (EOG)

- coherent at cut-out wind speed
- often used for controller testing
- here NREL 5 MW reference wind turbine
- over speed and tower motion
- absolute values count

SLOW to FAST comparision

- main dynamic well captured
- differences mainly due to unmodeled blades and dynamic inflow / stall

Example: fatigue load simulation 1/2



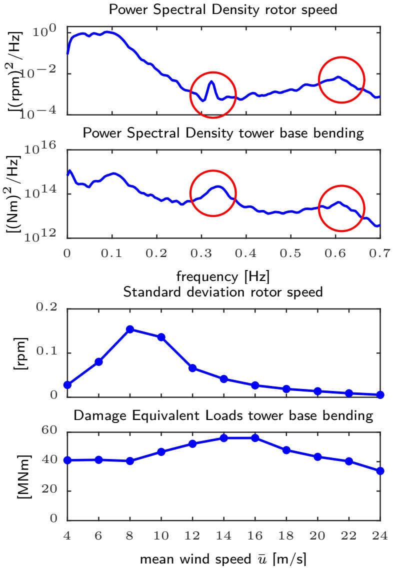
Normal Operation DLC 1.2

- 6×10 min per mean wind speed
- from cut-in to cut-out, 2 m/s steps
- turbulence according to turbine class

Damage Equivalent Loads (DEL)

DEL is an equivalent periodic load with constant amplitude and a predefined number of cycles which causes the same damage as stochastic loads.

Example: fatigue load simulation 2/2



Power Spectral Density (PSD)

- distribution of energy content over frequency
- tower resonance and 3P excitation visible
- useful to identify vibration issues at certain frequencies

Lifetime weighted load distribution

- weighting of values from each wind bin according to their relative frequency of occurrence over the lifetime
- useful to identify issues at certain operation points

Conclusion

Main questions

- How can we design a baseline generator torque controller?
- How can we and how should we evaluate a controller?

Nonlinear-state feedback controller $M_\mathsf{G} = f(\Omega_\mathsf{G}, \theta)$

- Region 2: $k_\mathsf{ISC}\ \Omega_\mathsf{G}^2$ brings turbine to chosen power coefficient
- Region 1.5 and 2.5: linear functions for transition
- Region 3: onshore usually constant power
- pitch limit θ_fine to determine, whether Region 3 or 2.5 should be applied

- Various steps to focus on specific questions starting with nominal model.
- Use same source code and initialize controller.

Quick check on learning objectives

After this lectures you should be able to...

- to describe the main tasks of a generator torque controller
- to design a baseline torque controller for all control regions
- to describe how the interaction with the pitch controller works
- to name useful steps to evaluate a controller
- to name conditions and type of analyses for standardized load simulations

References

[1] K. Johnson, L. Pao, M. Balas, and L. Fingersh. "Control of Variable-Speed Wind Turbines". In: IEEE Control Systems Magazine 06 (2006), pp. 70–81.

[2] J. Jonkman, S. Butterfield, W. Musial, and G. Scott. Definition of a 5-MW Reference Wind Turbine for Offshore System Development. Tech. rep. TP-500-38060. NREL, 2009. DOI: 10.2172/947422.

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

Please let me know if you have further questions!

Prof. Dr.-Ing. David Schlipf
David.Schlipf@HS-Flensburg.de
www.hs-flensburg.de/go/WETI