"D8.7.3 - Final demonstrator report"

"WP 8.7: Wind turbines"

"WP 8: Demonstrators"

MODRIO (11004)

Version	3.0 (M36)
Date	22/03/2016

Authors

Henning Schmitt	SIMPACK GmbH
Franciscus van der Linden	DLR-SR, Germany
Rüdiger Franke	ABB AG, Mannheim

Summary

This report describes the final status of the deliverable D 8.7.3 "Final demonstrator report" (at M36). The objective is to give a short description of the Simpack wind turbine model, and show the export of the model as an FMU. The Simpack gearbox submodel of the wind turbine model is numerically linearized and utilized to design a health monitoring system of the gearbox that is able to detect faults in all of its three stages. The health monitoring system is demonstrated at hand of tooth failures introduced in all three gearbox stages and these failures are successfully detected. The results of the detailed wind turbine simulation and health monitoring are exploited to set up an online optimization of the control of multiple wind turbines in a wind farm. The online optimization itself uses simplified wind turbine models along with aerodynamic interactions due to wake effects. The model is exported with FMI 2.0 and imported into the ABB control system. A numerical online optimization loads the binary FMU and obtains optimized wind turbine set points basing on current measurements.

1. Model description

1.1. Wind Turbine

The wind turbine is a Simpack model based on a publicly available 5MW reference wind turbine. A detailed gearbox and drivetrain has been added. The wind turbine is driven by the aerodynamic forces modeled by the Simpack interface to the aerodynamic code AeroDyn, which is based on the blade element momentum theory (BEM). These forces are applied to the flexible rotor blade modeled with the SIMBEAM module. A variable speed torque/pitch controller is integrated in the model by means of the Simpack wind turbine controller interface. This element interfaces with an industry standard controller DLL.

Figure 1: 5MW wind turbine – complete model

The input shaft of the gearbox system is a flexible main shaft that is connected to the hub on the one side and the planet carrier of the first stage on the other. The high speed shaft of the gearbox is connected to the generator via torsional coupling that allows for small tilt movements and displacements. The gearbox itself consists of two planetary stages (4 planets and 3 planets) and one spur stage on the high speed side. All gear pairs are modelled with Simpack's advanced gear pair element, which accounts for detailed teeth meshing forces and load distributions along the tooth flanks. The sun shafts and the high speed shaft are modeled as flexible beams build up with the SIMBEAM element, which is a modal beam representation integrated in Simpack. The considered eigenmodes for every shaft have been selected based on maximum rotational velocity, e.g. for the high speed shaft modes up to 4200Hz have been taken into account.

A more detailed description of the model structure and setup is given in the MODRIO deliverable report D 8.7.2.

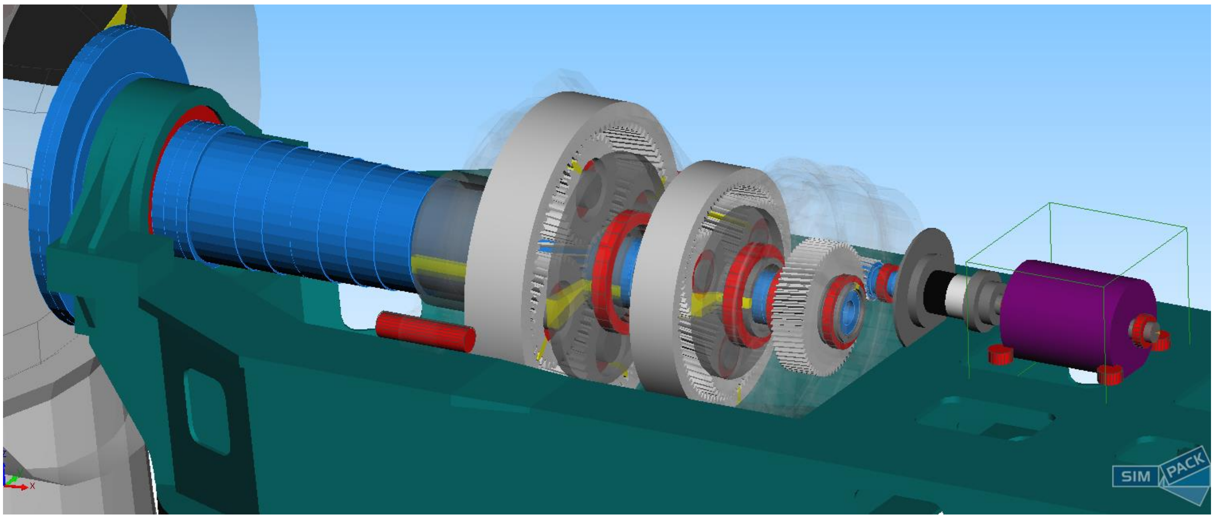

Figure 2: 5MW wind turbine – detailed drivetrain

1.2. Controller

The controller regulates the generator torque and the collective pitch angle of the rotor blades to yield an optimal energy output below rated wind velocity and to maintain a constant power output of 5MW above rated wind speed. Before the cut-in wind speed, the generator torque and the pitch angle are held to 0 to accelerate the rotor. In the region between cut-in and rated wind speed, the collective pitch angle of the rotor blades stays zero, while the generator torque is used to regulate the variable rotor/generator speed to yield an optimal energy output. Above rated wind speed the collective pitch angle of the rotor blades is the main mechanism to control the rotor/generator speed with the intention to hold it constant. The generator torque is subsequently adjusted to maintain a constant power output of 5MW. The figures below show the integration of the controller algorithm in the model and its basic behavior in a run-up of the wind turbine from a stand still state to the end of the operating range. The turbine is driven by a linear increasing laminar wind in the velocity range of $0m/s<v_{wind}<25m/s$.

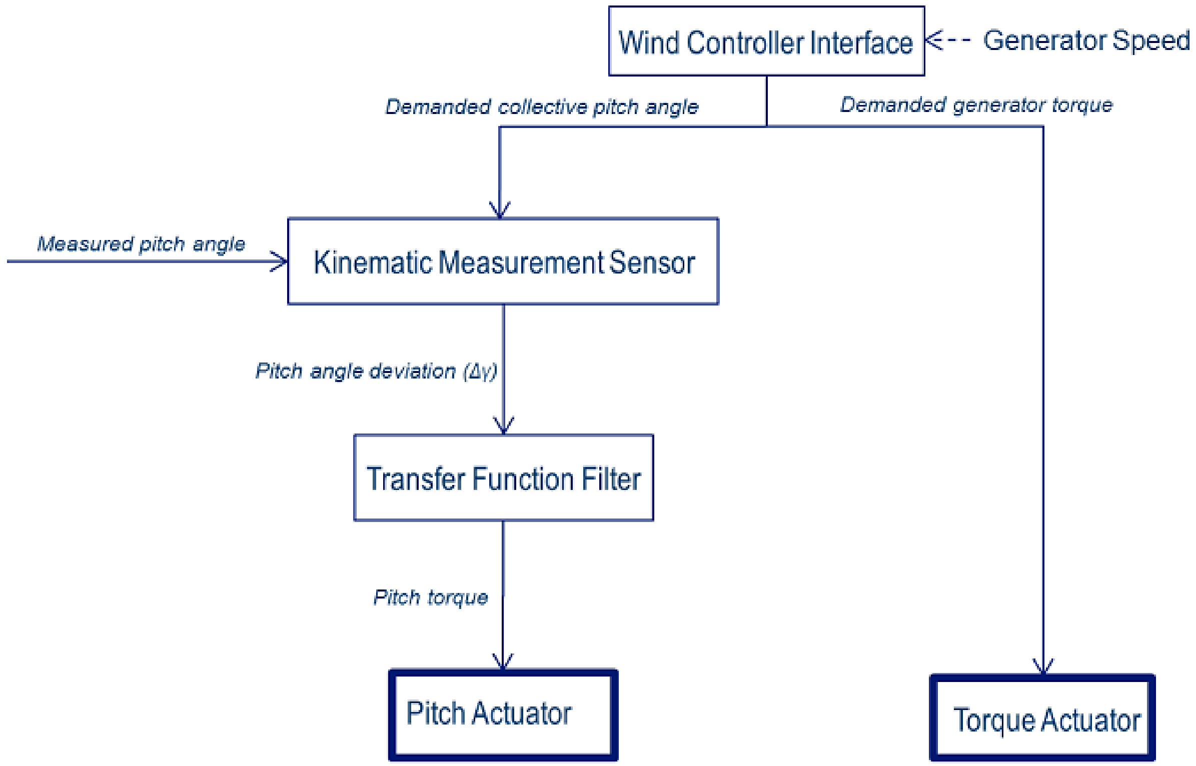

Figure 3: Wind controller setup

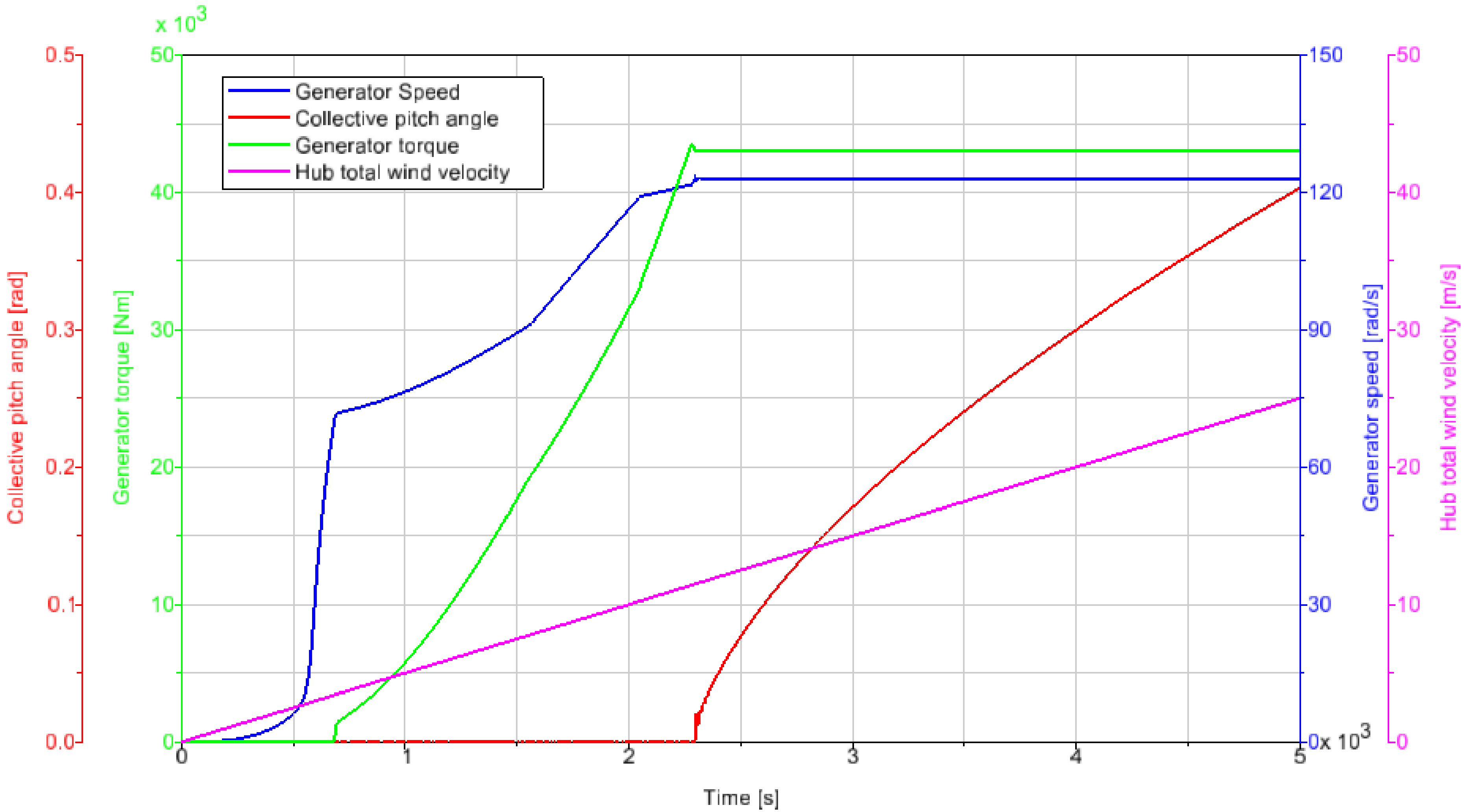

Figure 4: Wind controller and turbine behaviour

1.3. FMU Export

The whole wind turbine model has been exported according to the FMU 1.0 standard for co-simulation. This includes the whole detailed drivetrain model including gear wheel contact as well as the aerodynamic and controller element. The model is completely functional as a standalone simulation without the need for any additional modules that need to be connected. Since no external influences are setup in the model, no input channels are defined for the FMU. To communicate the main measurements of the generator, the controller and the wind conditions to other tools, additional output channels have been created. These include rotational velocity of the generator rotor, the generator torque, the collective rotor blade pitch angle, the hub height wind velocity, the hub height wind direction, the rotor thrust, the rotor moment and the rotor power.

- y-Outputs
  - $Y_GEN_speed
  - $Y_GEN_Torque
  - $Y_PCH_angle
  - $Y_HH_wind_velocity
  - $Y_HH_horizontal_wind_direction
  - $Y_ROT_thrust
  - $Y_ROT_moment
  - $Y_ROT_power

To get access to more detailed information about a finished simulation and the corresponding time series, the FMUs have been set up with the possibility to export Simpack binary result files (.sbr). These can be opened and used for post processing purposes with Simpack Post.

2. Validation

To validate the model exported according to the FMU standard, a comparison between the simulation results of the original Simpack model and exported model has been done. Three different scenarios have been tested. The first scenario is a run up from a standstill state of the wind turbine to an operating point above rated wind speed. The simulation runs through all controller regions described in chapter 1.2. The second scenario shows the system behavior under an extreme operating gust above rated wind speed, while in the third scenario a full field turbulent wind field is applied to the turbine.

All scenarios show an identical behavior between the original Simpack model and the FMU, which validates the success of the Simpack FMU export and its simulation capabilities.

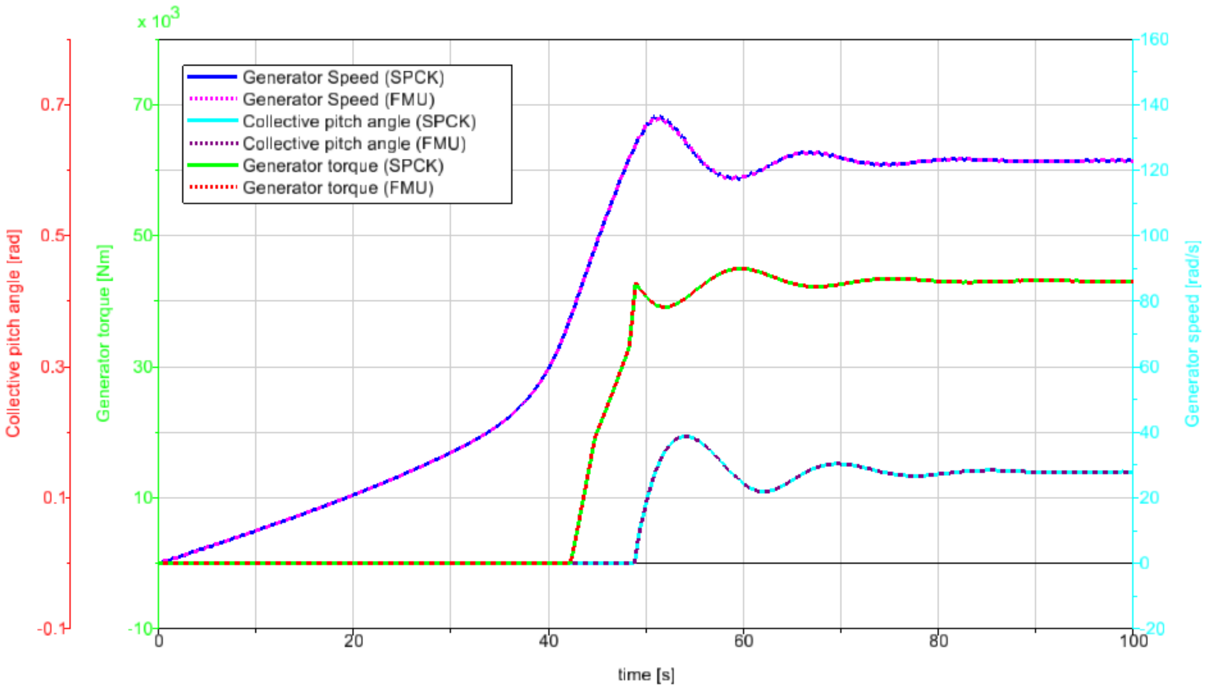

Figure 5: Validation of simulation results in a run up scenario

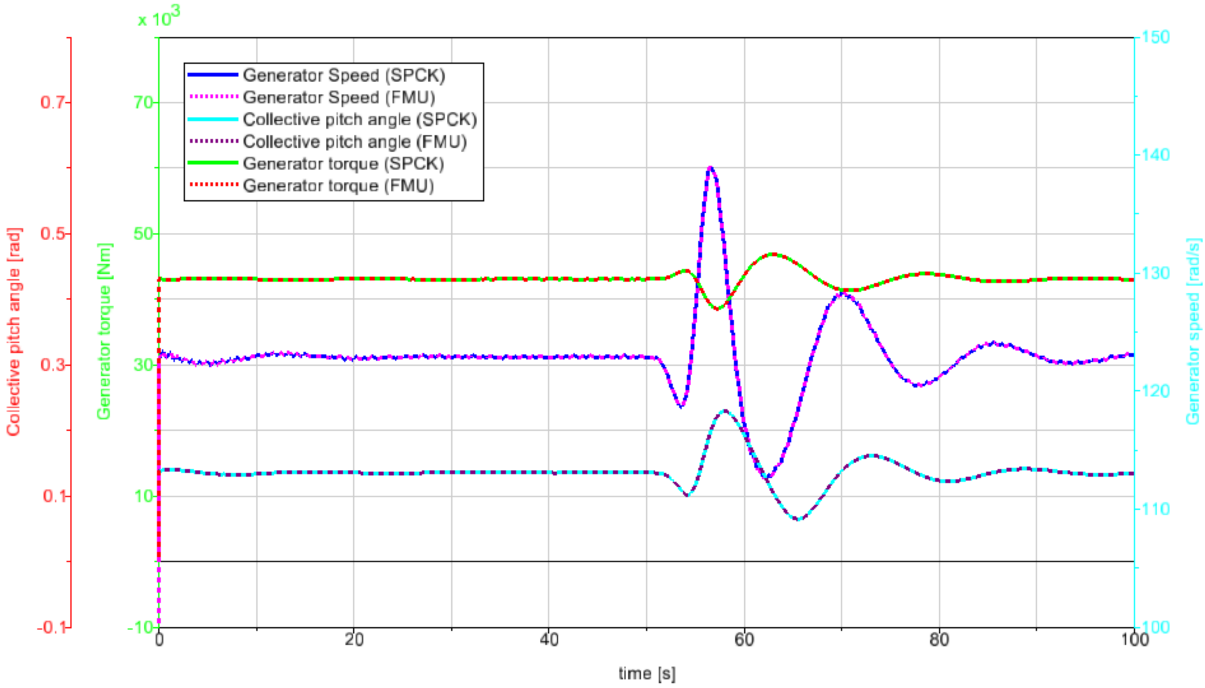

Figure 6: Validation of simulation results in under an extreme operating gust

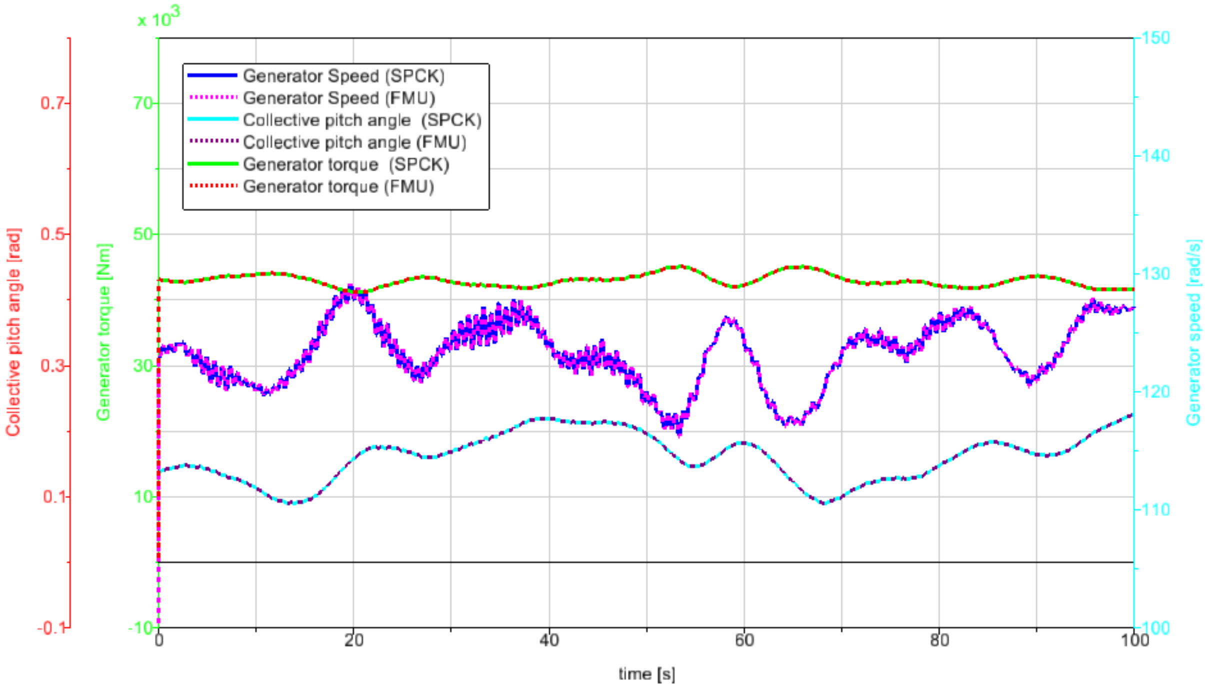

Figure 7: Validation of simulation results under turbulent wind conditions

3. Health Monitoring

The wind turbine gearbox model presented in Section **Error! Reference source not found.** has been used for the design and testing of health-monitoring algorithms to detect tooth faults.

The figure below shows a schematic overview of the applied detection method.

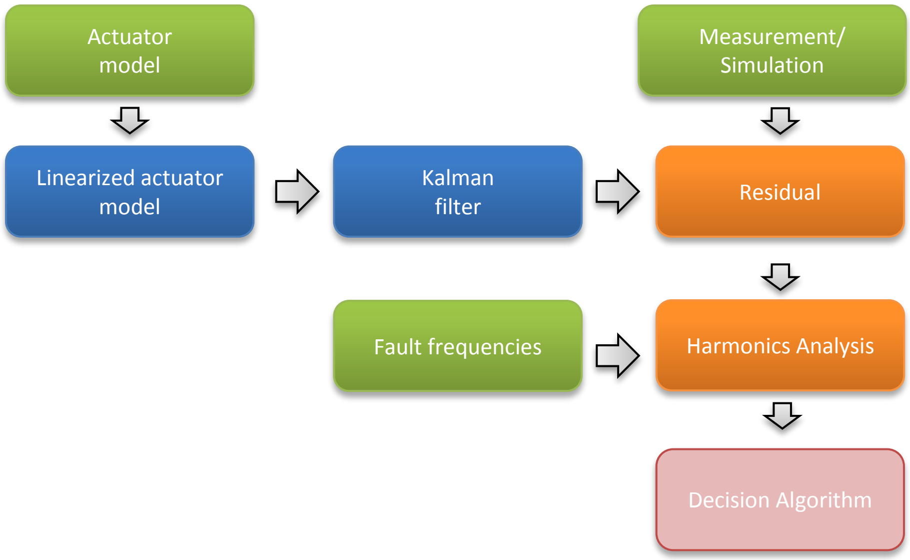

Figure 8: Block diagram of the Health Monitoring Algorithms.

3.1. Fault Description

The failure modes considered most relevant for wind turbine gear boxes are pitting, scuffing, and tooth breakage (Takoutsing et al., 2014). To simulate these failures, flank pitch errors are introduced that can emulate the following faults:

- Tooth base crack: reduction of the flank pitch of a single gear tooth, simulating a lower stiffness due to tooth bending.
- Gear Pitting: reduction of the flank pitch of a single gear tooth, simulating a smaller contact between the gears due to missing material.

Faults on 3 gear wheels have been investigated:

1. The gear wheel of the high speed stage (HSS) (Fault 1 in Figure **9**)
2. The sun of the intermediate stage (IMS) (Fault 2 in Figure **9**)
3. The sun of the low speed stage (LSS) (Fault 3 in Figure **9Error! Reference source not found.**)

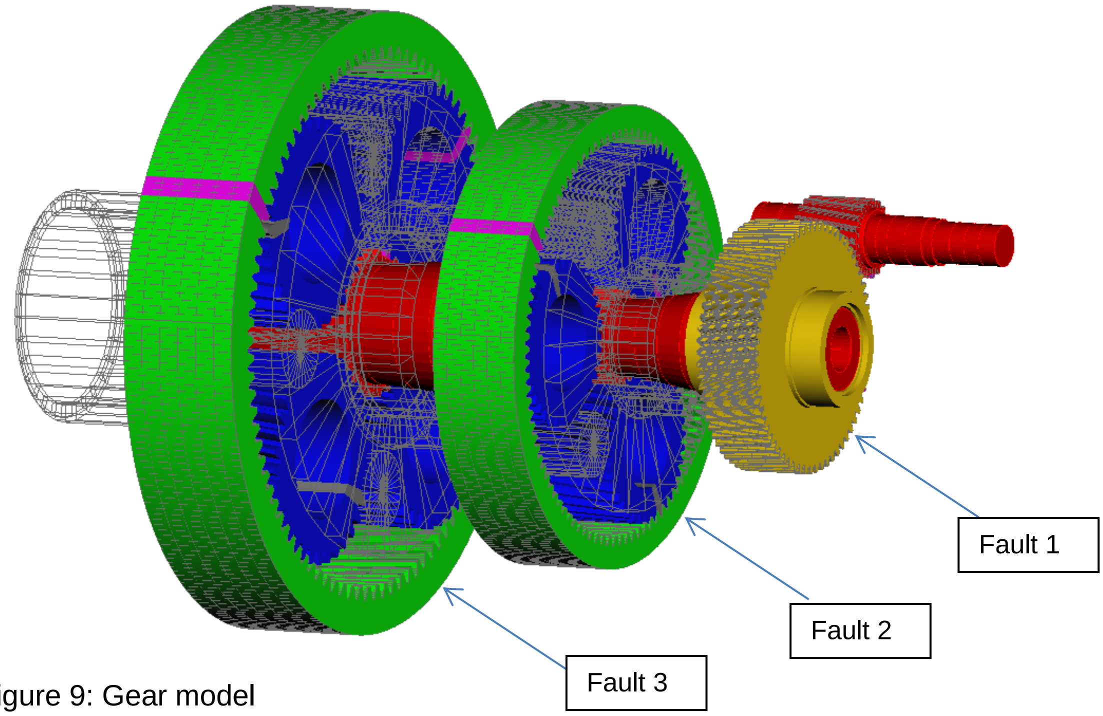

Figure 9: Gear model

The harmonics of these faults are shown in Table **1**.

Table 1: Description of Faults with harmonics and multiplicity

Fault	Fault description	Harmonics of Motor speed	Multiplicity
1	HSS Gear	0.29	4,5,6,8
2	IMS Carrier	0.73	1,2,3,4
3	LSS Carrier	0.16	1,2,3

The multiplicity of the faults has been set to represent the peakedness of the residual signal.

3.2. Kalman Filter

Simpack and the corresponding Gear-Toolbox have been used to generate a linearized model of the presented gear train. To model further disturbances, a torque input has been added to the sun of the low speed stage. The list of inputs to the model with the initial values are:

Table 2: Inputs to the system model.

Input	Initial value
Torque HSS Pin	43.5kN
Torque LSS Carrier	4393.6kN
Torque IMS Sun	0N

The used model outputs are:

Table 3: Outputs of the system model.

Input	Unit	Unit
HSS Pin	Position	rad
HSS Pin	Speed	rad/s
HSS Pin	Acceleration	rad/s^2
LSS Carrier	Position	rad
LSS Carrier	Speed	rad/s
LSS Carrier	Acceleration	rad/s^2

The linearized Simpack model has 122 states. For an efficient detection, the system has been reduced to 10 states with Matlab. Using the reduced system, a Kalman filter has been synthetized with the inputs and outputs shown in Table **2** and Table **3**.

Since in practice no speed sensors and acceleration sensors are available, the speed and acceleration of the motor and blade side is obtained by lowpass-filtering the position signal and differentiating thereafter.

3.3. Fault Simulations

The gearbox model described in Section **Error! Reference source not found.** has been imported in Dymola using a co-simulation FMU. The system has been simulated with constant load and speed. The faults in the HSS, IMS and LSS have been simulated and the results have been transferred to Matlab for further analysis.

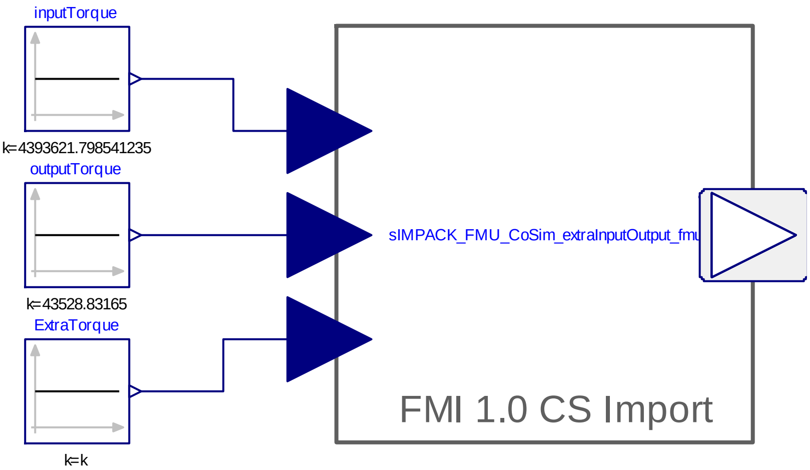

Figure 10: FMU of the wind turbine in Dymola

3.4. Fault Analysis

A health monitoring system has been created and implemented in Matlab according to the block

diagram shown in Figure **10**. Uniform realistic noise with a magnitude of a reasonable position sensor with 4096 increments has been added to position sensors of in- and output of the gearbox. A sampling rate of 2000 Hz has been used. The developed algorithms are suitable for online monitoring of the faults on low performance hardware.

Pre- and post-filtering of the signals has been applied for a further optimization of the ability to detect faults.

Note that the Fault analysis only relies on an input and output position sensor. As the motor position sensor is in most cases already available for motor control, the developed monitoring needs only a single extra sensor.

3.5. Kalman filter optimization

An optimization setup has been created to tune the measurement noise and process noise needed for the Kalman filter synthetis. These parameters have been tuned using MOPS (Joos, 2002). The optimization setup analyzes the 3 defined failure cases. As a criteria, the amplitude of the harmonics have been compared resulting to the detection capability. Furthermore, the stability of the Kalman filter is a tuning criterion. The result of the optimization is a single set of that can be applied for all three detection cases. For the final results, 226 evaluations have been simulated.

3.6. Detection results.

In the following figures, the detection of the faults is presented. They show the optimization steps of the optimization algorithm together with the final optimization result. The detection amplitude is the amplitude of the residual that is detected by the algorithm at the first fault harmonic. The rejection amplitude is the amplitude of one of the other signals. The detection capability is the relationship between the detection amplitudes and the rejection amplitudes.

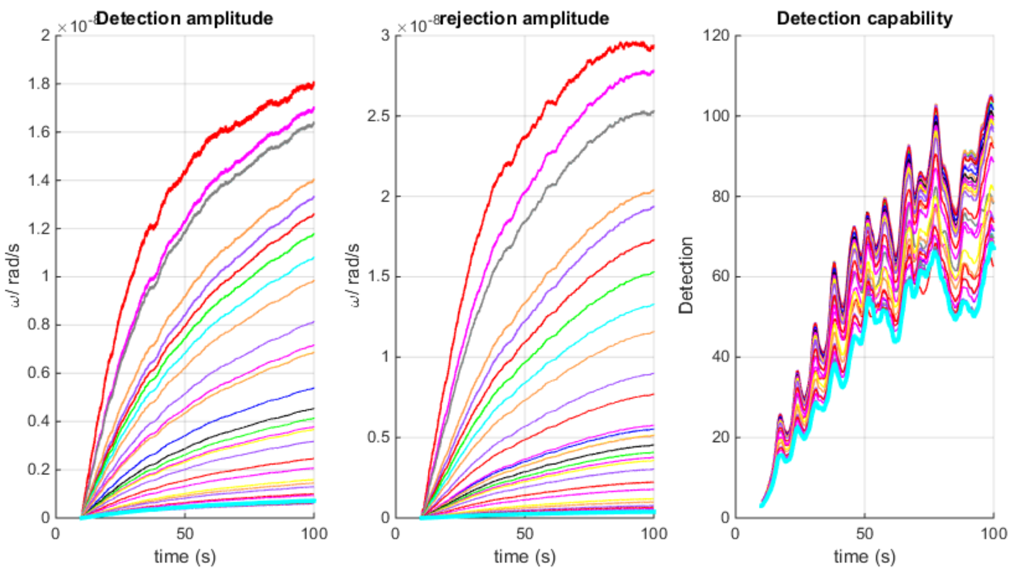

Figure 11: Fault detection of the HSS Drive. A detection capability > 3 means a good detectability. The thick blue result is the final optimization result; the other lines are optimization steps.

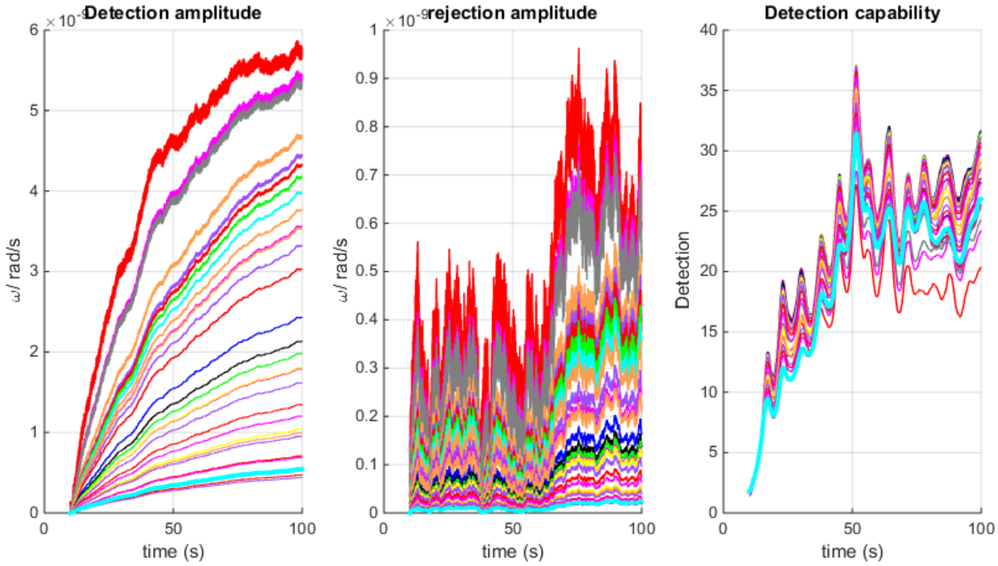

Figure 12: Fault detection of the IMS Drive. A detection capability > 3 means a good detectability. The thick blue result is the final optimization result; the other lines are optimization steps.

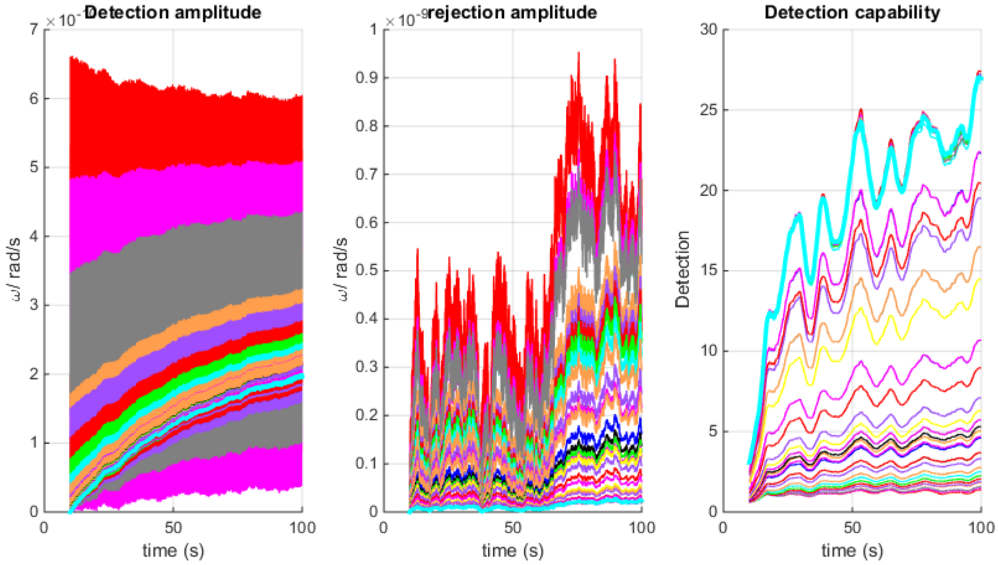

Figure 13: Fault detection of the LSS Drive. A detection capability > 3 means a good detectability. The thick blue result is the final optimization result; the other lines are optimization steps.

3.7. Conclusions

Using the presented algorithm, it is possible to detect faults in all stages of the wind turbine gear train. Using a global optimization of the detection problem, it is possible to create a single set of parameters that allow for a detection of all faults, simplifying the detection algorithms.

4. Optimal control of turbines considering wake effects

Wind turbines are typically erected collectively in wind farms. This results in aerodynamic interactions between subsequently arranged wind turbines, the so called wake effect. Furthermore the collection grid causes losses. When attempting to maximize the overall production, it may pay off to reduce the power of an upstream turbine to reduce its wake and thus increase the power of downstream turbines. This results in the need to optimize individual set points for each wind turbine. The main optimization variables are axial induction factor and yaw angle per turbine.

ABB has developed a model-based optimal control of set points of individual wind turbines in wind farms (Orth, 2016). The wind farm model uses simplified wind turbine models along with an electrical model of the collection grid and an overall wind field model. The MODRIO demonstrator builds on this development, the contributions by the partners in WP 8.7 and the achievements in WP4.

Individual wind turbine controllers adjust generator torque and possibly pitch angles along with yaw angles to achieve the requested output power (see section 1.2). The operation of a wind turbine away from its individual optimum may result in increased mechanical stress that must be analyzed using a detailed wind turbine model (section 1) and monitored (section 3).

These topics are covered with the contributions of the three partners in WP8.7.

4.1. Wind farm model

The figure below shows a model diagram of an exemplary wind farm.

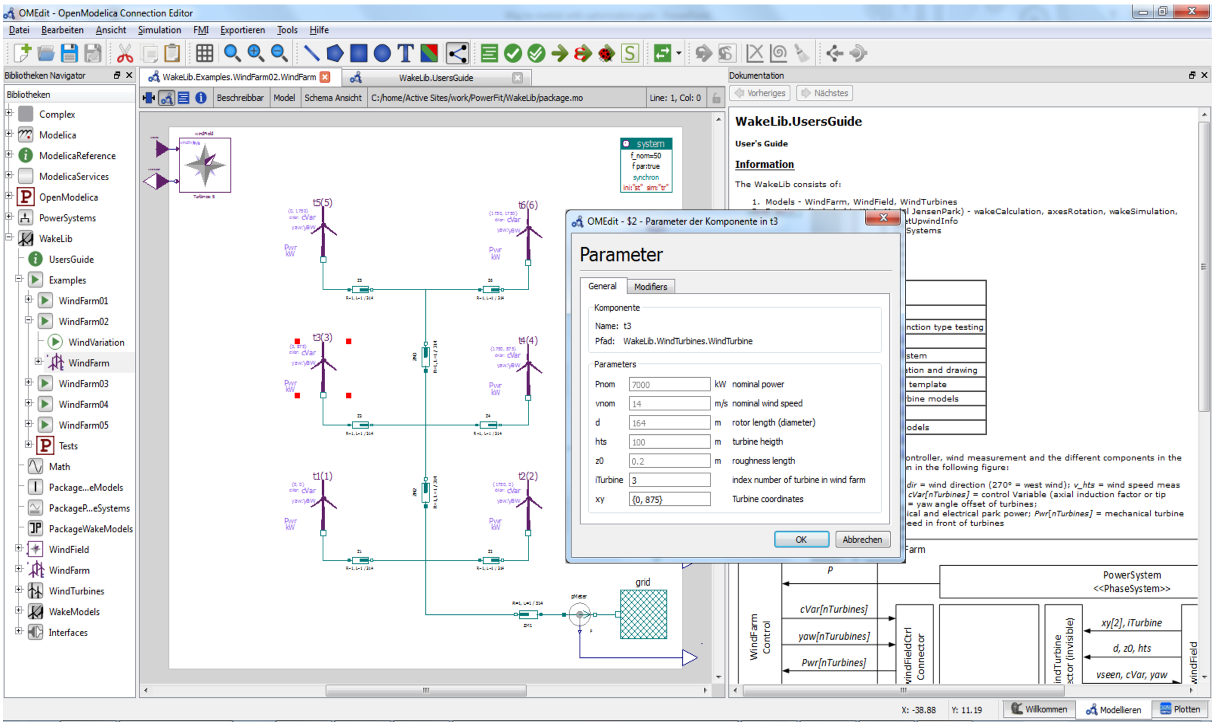

Figure 14: Screenshot of the OpenModelica Connection Editor showing a wind farm model with embedded documentation and parameterization dialog.

Each wind turbine connects to the wind field model (upper left component). The wind field model obtains the effective wind speed per wind turbine basing on overall wind speed, direction and wakes of individual wind turbines.

The collection grid is modeled using the PowerSystems library (Franke, Wiesmann, 2014).

The wind farm model is exported with the FMI 2.0 interface that has been implemented by project partners in WP4. The exported FMU typically contains two binary representations: one for test simulations on a Windows PC and a second for the control system. New cross compilation features of OpenModelica enable the generation of both binary representations with one call.

4.2. Setup of the demonstrator

The following figure gives an overview of the demonstrator.

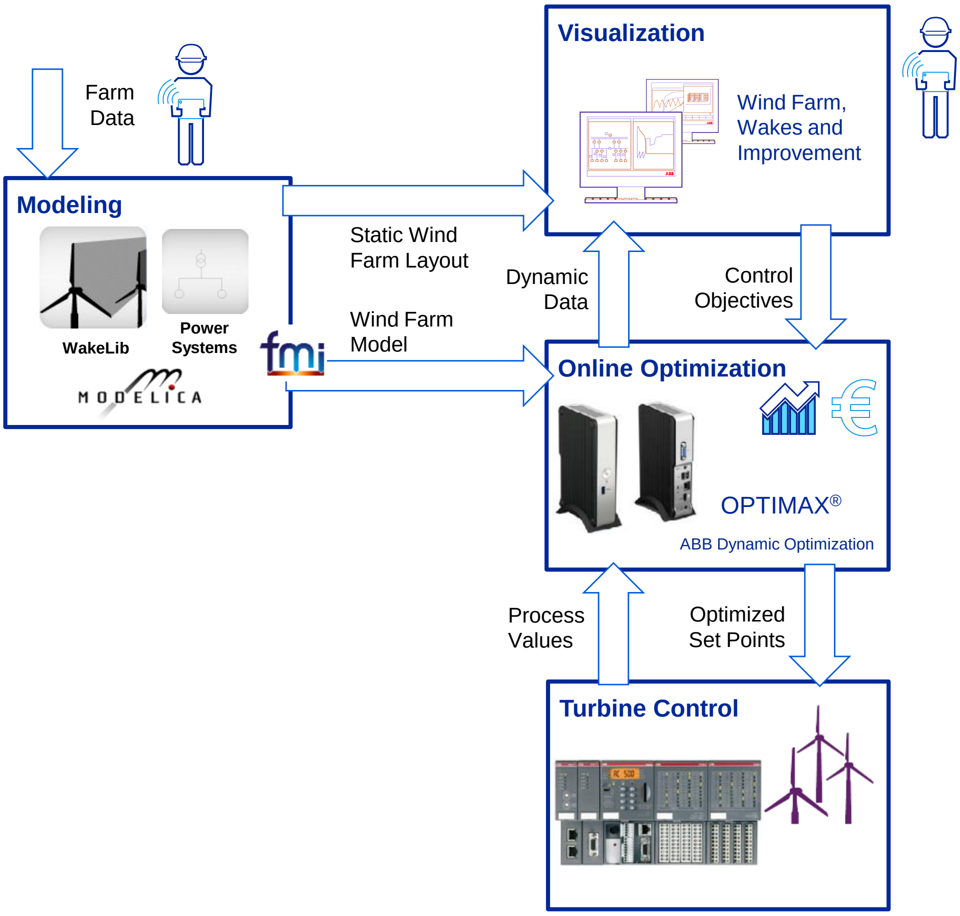

Figure 15: Overview of the demonstrator

The wind farm model is imported as binary FMU into the online optimization running on an embedded device with sufficiently high numerical CPU power. The online optimization connects to the turbine controllers to fetch measurements and to feedback optimized set points.

An operator graphics is set up for the static wind farm layout. It gives access to all relevant process values, set points and optimization results, including also a detailed visualization of wake effects and optimization results.

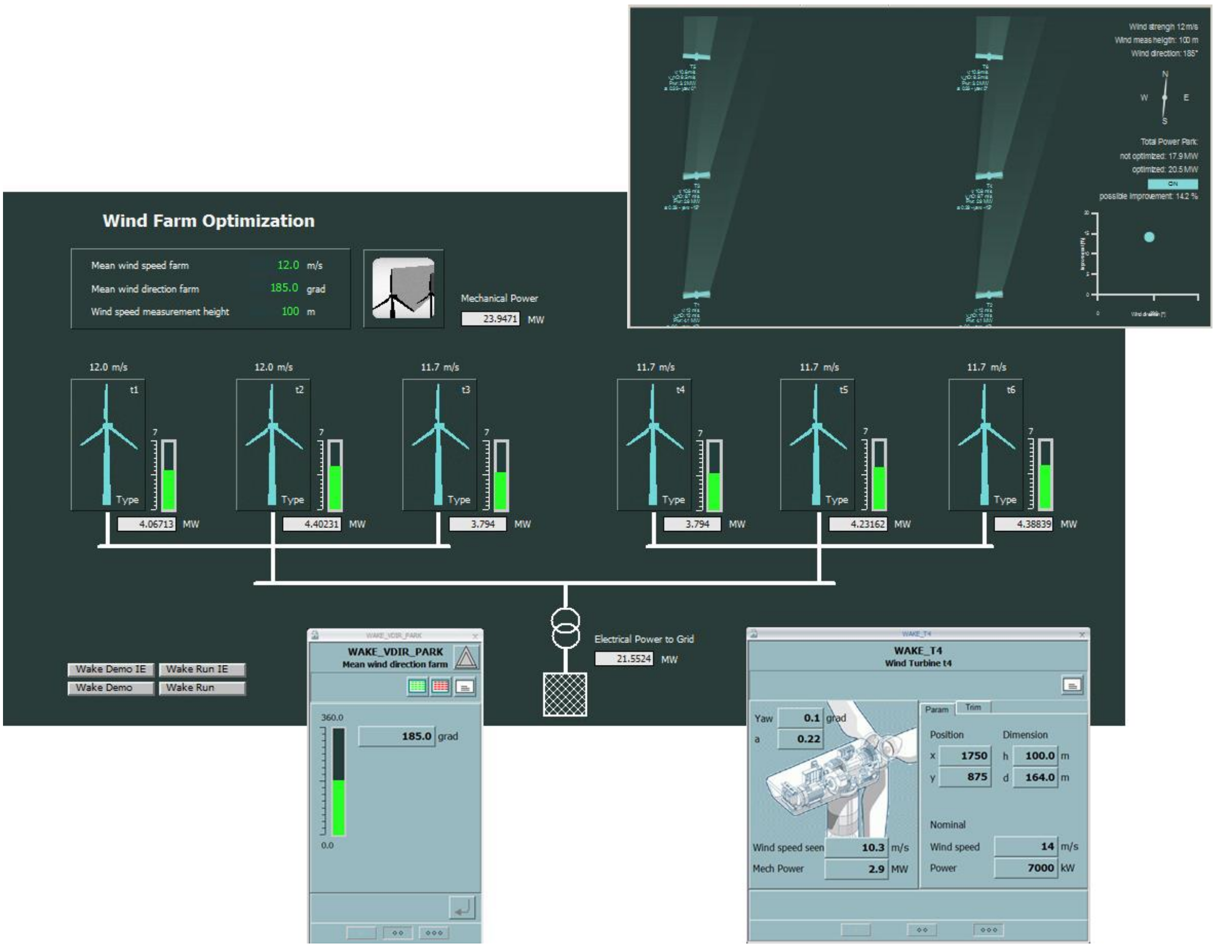

Figure 16: Operator screen with faceplates and visualization of the wake effect. Mind the different output power per wind turbine (green bars) due to wake effects.

5. Literature

Franke, R., Wiesmann, H.-J. (2014). Flexible modeling of electrical power systems -- the Modelica PowerSystems Library, 10th International Modelica Conference, Lund, Sweden

Joos, H.-D. (2002). A multiobjective optimisation-based software environment for control systems design. In *Proceedings. IEEE International Symposium on Computer Aided Control System Design* (pp. 7–14). IEEE. doi:10.1109/CACSD.2002.1036921

Orth, F. (2016). Modellbasierte Steuerung von Windfarmen unter Berücksichtigung aerodynamischer Interaktionen. Master Thesis. University of Applied Sciences, Darmstadt, Germany.

Takoutsing, P., Wamkeue, R., Ouhrouche, M., Slaoui-Hasnaoui, F., Tameghe, T., & Ekemb, G. (2014). Wind Turbine Condition Monitoring: State-of-the-Art Review, New Trends, and Future Challenges. *Energies*, *7*(4), 2595–2630. doi:10.3390/en7042595