Co-Simulation for Wind Turbines: Using the SIMULIA Co-Simulation Engine for Aiming Towards Fluid-Structure Interaction with Control

Stefan Sicklinger*[1)], Albert Kürkchübasche[2)], and Kai-Uwe Bletzinger[1)]

1) Technische Universität München, Munich, Germany
2) Dassault Systemes SIMULIA, USA
*)presenting author, stefan.sicklinger@tum.de

Abstract: *Co-simulation is a popular method to solve complex multiphysics problems. The co-simulation method has an intrinsic advantage that it allows well-established and specialized simulation tools to be reused and combined with minor adaptations in contrast to the monolithic approach (Sicklinger, 2014). We employ the co-simulation approach by coupling Abaqus, OpenFOAM and a control unit using MATLAB to solve a highly multidisciplinary problem. It is the versatility of the co-simulation approach which allows combining different levels of fidelity during the various development phases and thus helps to improve technical systems in every stage of the design process.*
Within this paper we show a challenging simulation representing problems within the renewable energy sector. The simulation of a wind turbine involving the interaction of the generator/gearbox with the wind is presented (signal field co-simulation). The simulation results are validated against measurement data from the National Renewable Energy Laboratory (NREL) Unsteady Aerodynamic Experiment Phase VI performed in the NASA AMES wind tunnel (Hand, 2001).
In the outlook we present the methodology to extend the co-simulation to a fluid-structure-signal interaction by using the SIMULIA Co-Simulation Engine coupling Abaqus/Standard, OpenFOAM and MATLAB to capture the realist behavior by appropriately modeling the interaction of various components.

Keywords*: Fluid-Structure Interaction, Co-Simulation, Wind Turbine, Multicode Coupling, System Control*

1. Introduction

Many simulations involve the coupling of different physics. An example is the analysis of a wind turbine. Here the structure (blades, tower, generator, gearbox etc.) are interacting with the fluid (air) and with each other. Within the Unsteady Aerodynamic Experiment Phase VI of NREL a real sized wind turbine was measured in the NASA AMES wind tunnel providing experimental data to validate the simulation results.

An interesting and challenging design case is an aerodynamic emergency brake maneuver. We will discuss the simulation of the aerodynamic emergency brake maneuver of the NREL Phase VI two-bladed wind turbine described in Reference (Hand, 2001). The Interface Jacobian-based Co-Simulation Algorithm (IJCSA) (Sicklinger, 2014)was employed between all components for improved accuracy and stability.

2. Simulation

2.1 Software Components

The co-simulation approach allows us to combine and use multiple commercial software tools to solve coupled physics problems. OpenFOAM is used to perform computational fluid dynamic (CFD) analysis of the air flowing around the blades and rotor. OpenFOAM is an open-source Finite-Volume based CFD solver. The wind turbine blades and generator/gearbox are modeled using Abaqus/Standard. In order to couple OpenFOAM with Abaqus/Standard we developed a connector process. This connector process used the SIMULIA Co-Simulation Engine (CSE) to communicate to Abaqus and the open-source tool EMPIRE (see Reference [5]) to communicate with OpenFOAM. OpenFOAM was selected because of the access to the source code.
The SIMULIA Co-Simulation Engine (CSE) API allows us to develop an interface to new simulation tools easily. Furthermore, the CSE allows coupling between two or more codes, being able to couple both physical and logical systems. Within this simulation the IJCSA is employed by the authors. The IJCSA allows to handling mathematical stiff co-simulation problems in a robust and accurate manner.
The presented simulation was executed on the super computer of the Chair of Structural Analysis of TU Munich. The simulations were run on 184 Intel Sandy-Bridge cores using a QDR Infiniband interconnect. In total approximately 85 million unknowns are solved per time step.

1.1 The Co-Simulation

The turbine used within the Unsteady Aerodynamic Experiment Phase VI is described detailed in Reference (Hand, 2001). Figure 1 shows the original and the CAD model of the turbine. The wind turbine has a diameter of 10m and a rated power class of 20kW.

We use a body-fitted URANS approach for modelling the incompressible Navier-Stokes Equations in OpenFOAM. A block-structured mesh is used around the blades to gain high-quality results. The first boundary layer is chosen such that the max. y+ on the blades is below 7. OpenFOAMs implementation of the SST k-ω turbulence model is selected. The Arbitrary Mesh Interface (AMI) within OpenFOAM is used to model the blade-tower interaction within the CFD solver.
A challenge in current simulation models of wind turbines is to model the interaction of the wind and the generator i.e. the rotor is mostly rotating with a prescribed rotational velocity.
As this case is widely investigated in literature (see Reference (Hsu, 2011)) we use this case first. NREL and NASA provided also measurement data for this case. Hence, the CFD model is validated with the provided experimental results of NREL and NASA. The measurement case

S0700000 (Hand, 2001) is chosen for the validation. For the S0700000 the flow around the blades is fully attached. The rotational velocity for the validation case is set to 72 rpm and the approaching free stream velocity to 7m/s.

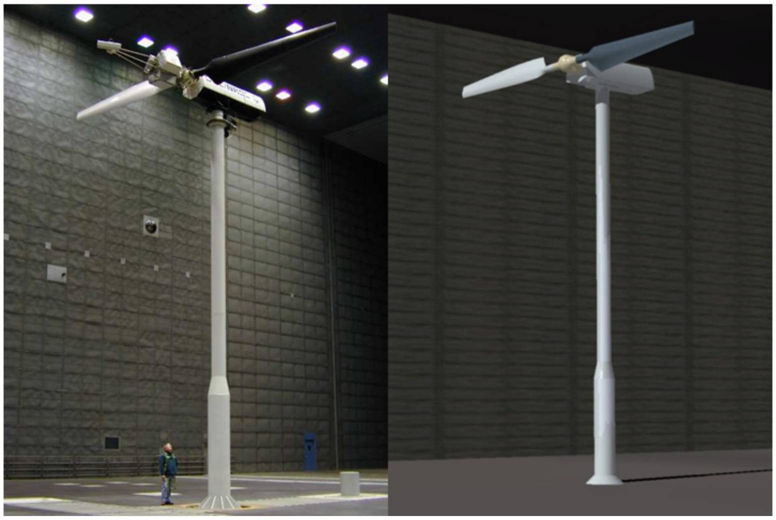

Figure 1: Left: Original wind turbine in the NASA Ames wind tunnel (Hand, 2001). Right: CATIA CAD model (Sicklinger, 2014b).

Figure 2 gives an illustration of the three dimensional flow field in the wake zone of the turbine. The pressure coefficients (cp), defined in equation (1), for different sections of the blade (Figure 3) agree very well with the measurements. The pressure coefficient is a dimensionless number which describes the relative pressures throughout a flow. With this we have validated our CFD model. With the help of the validated simulation model test cases can be investigated which were not measured in the experiment e.g. an emergency brake maneuver.

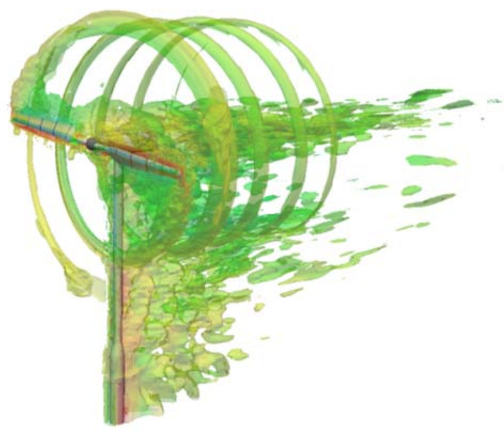

Figure 2: Vorticity isosurface of the CFD simulation.

$$c_p = \frac{p - p_\infty}{\frac{1}{2}\varrho\, u_\infty^2} \qquad (1)$$

ϱ: fluid density

u_∞: free stream flow velocity

p_∞: free stream pressure

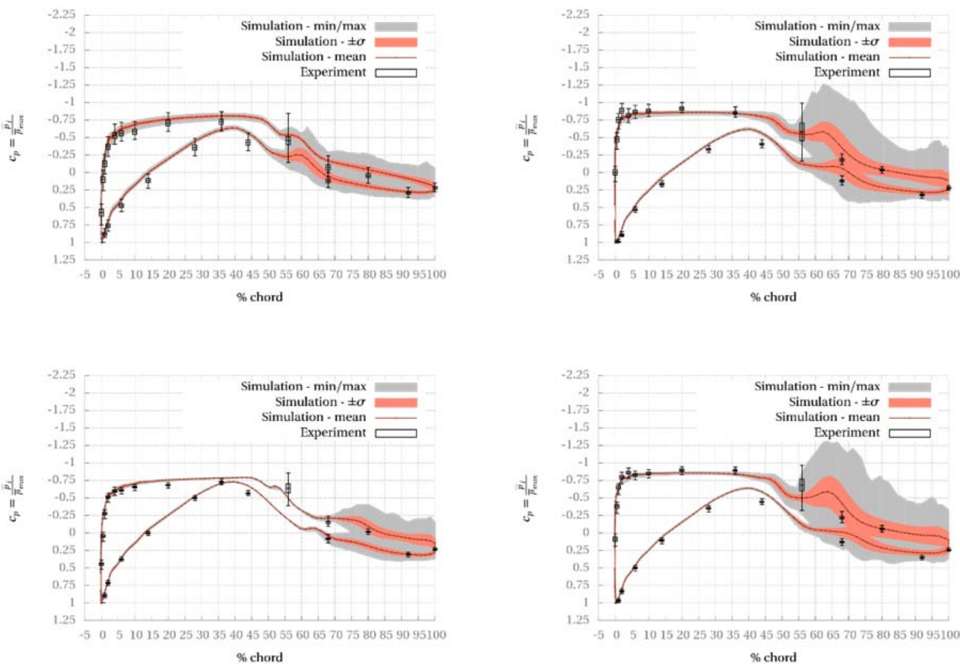

Figure 3: Cp coefficients of simulation and measurement for 30%, 47%, 63% and 95% blade radius (mean, standard deviation and min/max)

As a next evolution of the simulation we want to ensure that the rotational velocity is more realistic by including the coupling of the generator with the rotor. In addition a pitch control unit is added. The pitch control unit is modeled using MATLAB. Connectors are used to represent the Generator/Gearbox in Abaqus/Standard.

Stiffness and rotational inertia of the wind turbines powertrain provided in Reference (Hand, 2001) are used. Figure 4 shows a block diagram of the software components in the co-simulation.

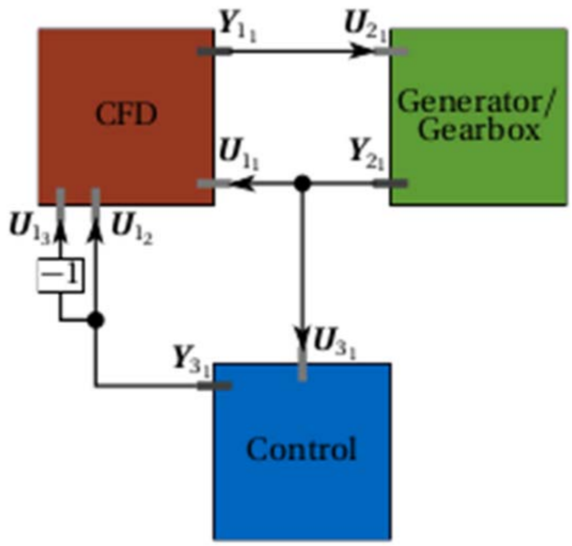

Figure 4: Block diagram of aerodynamic emergency brake maneuver

The aerodynamic torque is send from OpenFOAM to the generator modeled using Abaqus. The generator computes the corresponding rotational velocity and sends it to the CFD (Y_2→U_1). The control unit monitors the rotational velocity and executes a 90° pitching of the blades if the critical rotational velocity of 500°/sec is exceeded (see right axis of Figure 5).

During the startup stages when the turbine accelerates from standstill to operational rotational velocity the control unit rotates the blades slowly till operational conditions are reached.

We employ the IJCSA (Sicklinger, 2014) algorithm between OpenFOAM and Abaqus to achieve an accurate and stable solution. Moreover, the IJCSA allows for parallel execution of all involved software components. Thus the runtime is only determined by the overall maximum runtime of the individual component. Furthermore, the IJCSA uses a Newton method at interface level to solve the coupled problem, this results in a small number of interface iterations. For the aerodynamic emergency brake maneuver three interface iterations where enough to converge to the coupled solution. Figure 5 shows a comparison of the shaft torque between loose coupling and IJCSA coupling for the aerodynamic emergency brake maneuver. Within this simulation the rotational velocity is a result of the moment extracted by the turbine rotor and the inertia of the turbine power train. The error introduced by loose coupling is depending on the actual state in which the turbine is in. The IJCSA is the best choice if the behavior of the coupled simulation is not a priori known.

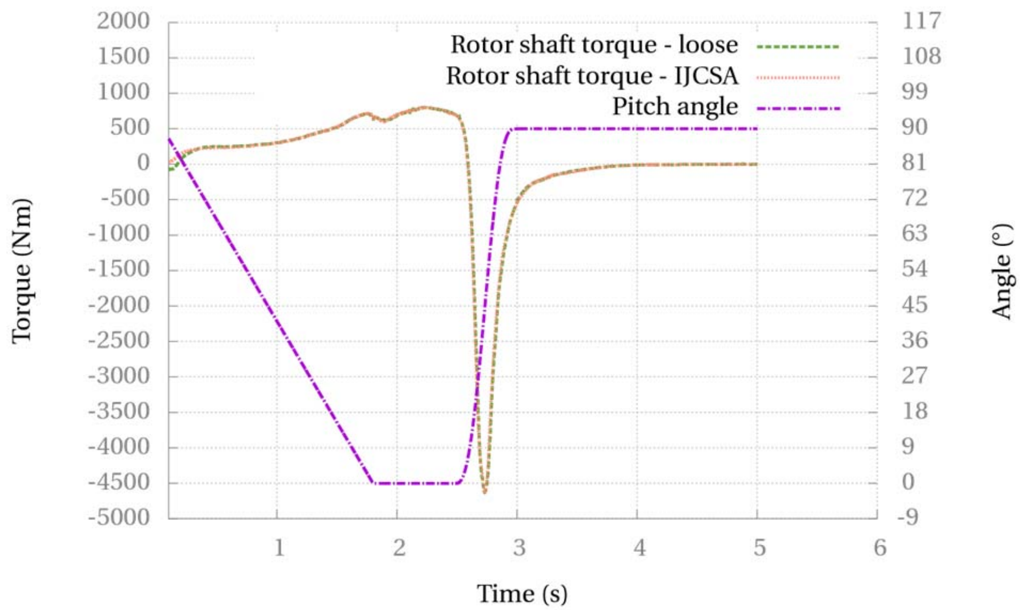

Figure 5: Rotational shaft torque and pitch angle for the aerodynamic emergency brake maneuver

The evolution of vorticity isosurface within the aerodynamic emergency brake maneuver is shown in Figure 6. In this case the blades were assumed to be rigid. Our next refinement will include deformable blades with realistic composite layups and material properties. .

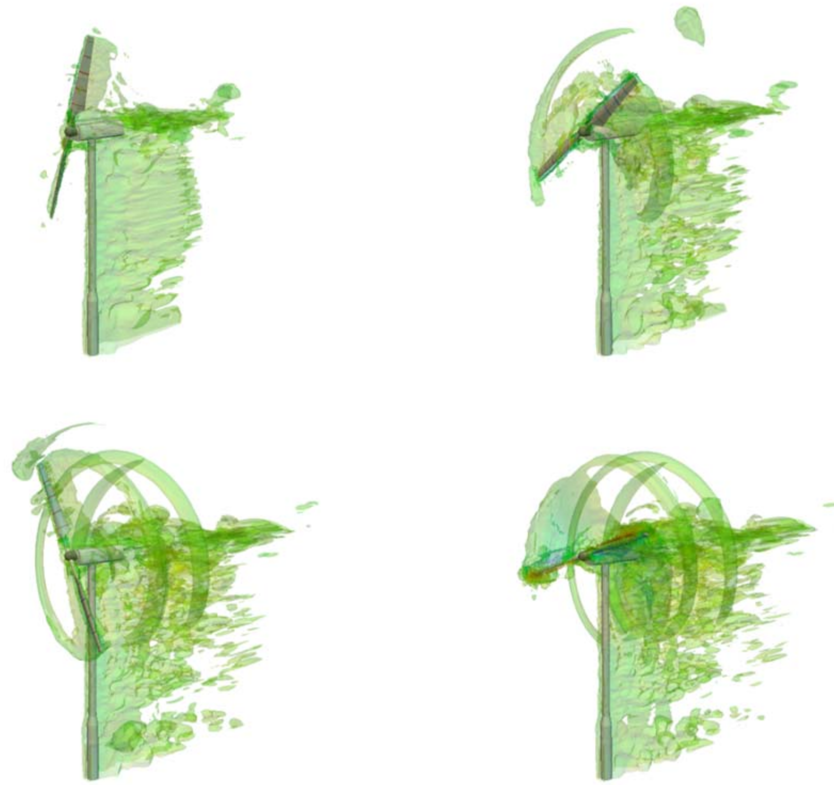

Figure 6: Time series of aerodynamic emergency brake maneuver (vorticity isosurface)

3. Outlook

As mentioned above, our next refinement will include realistic composite layup and material properties for the blades. With this a fluid-structure analysis coupled to a control system will be performed accounting for the large deformation and finite rotation of the blade angles. We hope to present preliminary results during the presentation.

4. Conclusions

Within this paper we presented the co-simulation of an emergency brake maneuver of wind turbine. We employed the Interface Jacobian-based Co-Simulation Algorithm and verified the results against data provided by Unsteady Aerodynamic Experiment Phase VI NREL.
In this work three codes were coupled, two physical codes modelling the fluid-structure interaction and a logical code modelling the control unit that adjust the pitch angle of the rotor blades.

The approach we took in the co-simulation was to gradually build upon the complexity of the model to obtain a more accurate and realistic solution. First a CFD analysis was performed which was validated against experimental data, then we added a generator/gearbox model and a control unit. The next phase we will be to provide realistic blade properties in a complete fluid-structure-control simulation.

5. Acknowledgement

The authors would like to thank Dr. Scott Schreck from NREL for providing the measurement data.

6. References

1. Sicklinger S., Belsky V., Engelmann B., Elmqvist H., Olsson H., Wüchner R., Bletzinger K.-U. (2014). Interface-Jacobian based Co-Simulation: *International Journal of Numerical Methods in Engineering. (DOI: 10.1002/nme.4637)*
2. Sicklinger S., Wang T., Wüchner R., Bletzinger K.-U. (2014). Co-Simulation for Renewable Energy Simulations: A Framework for Field-Signal Coupling for Fluid-Structure Interaction with Control: *International Symposium on Computational Wind Engineering*
3. Hand, M.M., Simms D.A., Fingersh L.J., Jager D.W., Cotrell J.R., Schreck S., Larwood S. (2001).
   Unsteady Aerodynamics Experiment Phase VI: Wind Tunnel Test Configurations and Available Data Campaigns: *NREL Technical Report.*
4. Hsu M.-C., Akkerman I.,·Bazilevs Y. (2011) Wind turbine aerodynamics using ALE–VMS: validation and the role of weakly enforced boundary conditions: *Computational Mechanics (Springer)*
5. http://www.empire-multiphysics.com