Practical Considerations regarding Implementation of Wind Power Applications into Real-Time Hardware-In-The-Loop Framework

L. Petersen[1], F. Iov[2]
Department of Energy Technology, Aalborg University, Pontoppidanstræde 101, 9220 Aalborg Øst, Denmark
[1]lep@et.aau.dk, [2]fi@et.aau.dk

Abstract

This paper addresses the system implementation of voltage control architecture in wind power plants into a Real-Time Hardware-In-The-Loop framework. The increasing amount of wind power penetration into the power systems has engaged the wind power plants to take over the responsibility for adequate control of the node voltages, which has previously been accomplished by conventional generation. Voltage support at the point of common coupling is realized by an overall wind power plant controller which requires high-performance and robust control solution. In most cases the system including all controls is reproduced in continuous-time domain using Laplace transform, while in practical implementation digital control systems are employed. The scope of this paper is to elaborate on the practical implementation of the voltage control architecture into a Real-Time Hardware-In-The-Loop framework, where the focus is laid on the model development in a real-time simulator. It enables to verify the functionality of developed controls, which is one of the research priorities due to the increased complexity of large wind power plants requiring high level of communication between plant control and a large number of assets such as wind turbines and FACTS devices.

1 Introduction

The increasing amount of wind power penetration into the power systems has engaged the wind power plants (WPPs) to take over the responsibility for adequate control of the node voltages, which has previously been accomplished by conventional generation. Voltage support at the point of common coupling (PCC) is realized by an overall WPP controller, which dispatches reference signals to the wind turbines (WTs), thereby controlling the voltage at the PCC (Fig. 1).

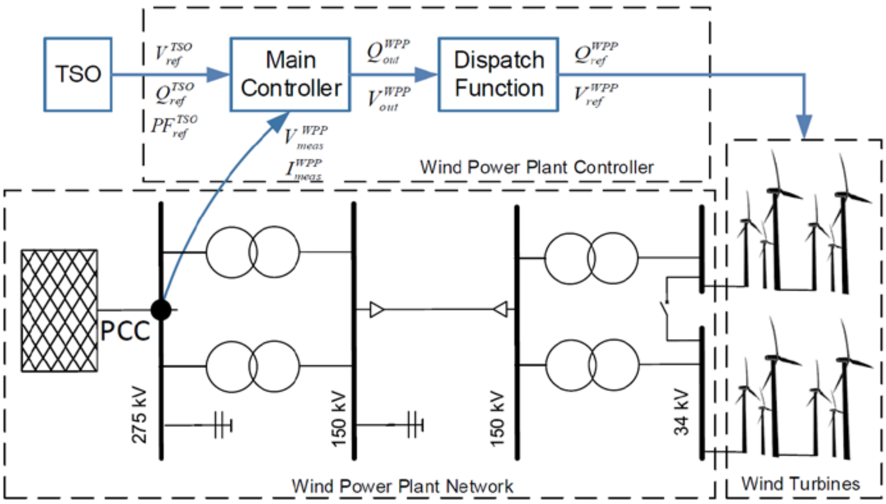

Fig. 1 Overview Wind Power Plant Voltage Control Architecture

Therefore the overall WPP controller requires high-performance and robust control solution. Recent research activities such as in [1] and [2] have focused on the design and tuning of voltage control for large-scale WPPs. In most cases the system including all controls is reproduced in continuous-time domain using Laplace transform, while in practical implementation digital control systems are employed. Moreover, the complexity of large WPPs involving a huge number of WTs and possibly FACTS devices (e.g. STATCOM) requires high level of communication between plant control and assets. Taking into account those aspects the system needs to tested prior to on-site implementation in order to verify the functionality of developed controls. However, this applies generally for more sophisticated controls in WPPs being required due to more stringent grid code requirements stipulated by ENTSO-E in near future. Verifying enhanced ancillary services is one of the research priorities defined by the Danish Research Consortium for Wind Energy [3].

The scope of this paper is to elaborate on the practical implementation of the mentioned voltage control architecture into a Real-Time Hardware-In-The-Loop (RT-HIL) framework, where the focus is laid on the model development in a RT simulator. One important aspect is the model discretization. Both WT and WPP control need to be modeled in z-domain, where different discretization techniques (i.e. Euler, Tustin) affect the system stability and performance. Furthermore, various model domains such as WT, WPP network and WPP control (Fig. 1) are simulated with different sampling times depending on their requirements (e.g. system bandwidth). Last but not least, some practical issues need to be regarded, i.e. splitting model parts into different CPU cores of the simulator to meet the requirements for RT performance (no overruns) and configuring communication links for the data exchange with an industrial controller.

2 Architecture of Smart Energy Systems Laboratory

The *Smart Energy Systems Laboratory* (*SES Lab*) at Aalborg University facilitates a range of assets (see Fig. 2) to capture two main aspects of intelligent energy systems: the ICT component and the energy networks and systems itself [4]. In this way, this laboratory enables also testing in a RT-HIL framework. A *Real-Time Digital Simulator* based on *Opal-RT* technology simulates large scale energy networks and the *Renewable Plant Control* based on *Bachmann* technology hosts typical control functionalities implemented in renewable based generation plants such as wind or PV. The backbone for the setup is the ICT layer linking all assets and aiming to emulate different technologies and topologies for the communication networks including realistic data traffic.

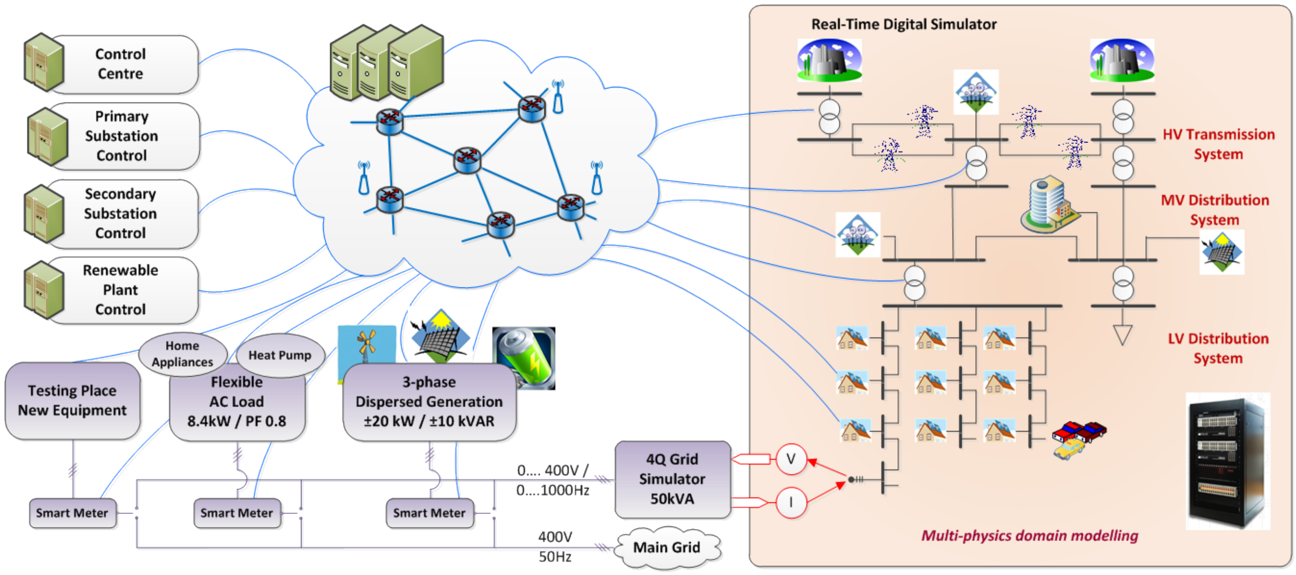

Fig. 2 Overview of *Smart Energy Systems Laboratory* at Aalborg University

The main goal of the *Real-Time Simulator* is to capture the electrical system from the transmission level (TS) down to low voltage distribution grids (DS). The RT transient stability simulation tool *ePHASORsim*, simulating power system dynamic phenomena within fundamental frequency, is the appropriate software product for validating algorithms for complex global control schemes. The relevant main features of *Opal-RT* simulator in *SES Lab* are summarized in following table:

Tab. 1 Selected main features of *Opal-RT* simulator in *SES Lab*

Real-Time Simulator Overview	
Operating System	Linux Redhat
CPU	4 x Intel 8-core, Xeon 2.7GHz LGA2011, 20MB cache
Total Core #	32
Memory	8 x 8 GB
Support for Third-Party I/O	IEC 61850 (Goose & SV), IEC 60870 104, UDP
OP5600 I/O Expansion Unit	
Platform	Spartan 3 FPGA
Inputs / Outputs	16 Analogue In, 16 Analogue Out, 32 Digital In, 32 Digital Out
Real-Time Transient Stability Simulation Tool *ePHASORsim*	
Application	Phasor-domain simulations for large-scale power systems, Advanced HIL and Model-in-the-Loop testing of power system components
Maximum # of busses	20.000
Guaranteed Time Step	10 ms
Embedded Power System Components	Synchronous Generator, Exciter, PSS, Turbine & Governor, Line, Load, Transformer, Shunt, Switch
Third-Party Connectivity	MATLAB/Simulink, SimPowerSystems, SimScape, Excel, PSS/E

The modular system for *Renewable Plant Control* by *Bachmann*, consisting of processor module and grid measurement module, contains functions such as closed-loop control, networking and monitoring. The complete integration of the controller as target system for *MATLAB/Simulink* enables convenient and efficient implementation of *MATLAB* functions on the control system, thereby facilitating the model-based design approach. Communication in RT is realized by open and standardized systems. The integrated MMS Server is tailored to the energy industry for standardized communication. The key features of the *Renewable Plant Control* system in *SES Lab* are summarized in following table:

Tab. 2 Selected main features of *Bachmann Renewable Plant Control* system in *SES Lab*

Processor Module MC210	
CPU	1.6 GHz ATOM E680
RAM	1 GB DRAM DDR2
Data Memory	512 kB
Interfaces	2x Eth100/1000; 1x RS232; 1x RS232/422/485; 1x USB2.0; 1x USB 2.0
Grid Measurement Module GMP232	
Measurements	Current, voltage, frequency, power, power factor, phase angle, grid harmonics up to 50^{th} order
Sampling Rate	50 µs
Additional features	Monitoring /Protection functions for grid and generator protection; direct relay outputs for circuit-breaker/trip circuits; integrated real-time data recorder and event logging
Communication Capabilities	
OPC Server	Manufacturer-neutral interface
MMS Server	Communication in accordance with IEC 61400-25 or IEC 61850

3 Requirements for Real-Time Simulations of Wind Power Plant Control

3.1 Model Discretization

In many cases continuous-time domain or frequency domain models serve as a basis to be implemented in a RT simulation platform to accomplish controller validation and HIL testing. E.g. in [2] a state-space model of a WPP system has been developed for analytical design and tuning of the WPP voltage controller in frequency domain. When it comes to discrete-time systems, the simulation model needs take on fixed-step sampling times. Their parametrization depends on both the performance capabilities of the simulation platform and the characteristics of the modeled system.

The former aspect relates to the real-time requirement that the amount of real-time required to compute all equations and functions during a given time-step should be equal or shorter than the duration of the selected simulation time-step [5]. Hence, too small sampling times may cause overruns in the RT simulator.

On the other hand, the modeled system can exhibit various sampling times dependent on the simulated components. In a WPP control system the RMS voltages and currents are normally sampled with 10 ms step size. The wind turbine (WT) system can be simulated with various sampling times dependent on the level of detail. For instance the IEC 61400-27 standard, representing electrical simulation models for wind turbines being appropriate for FRT studies, specifies the smallest sampling time to be 5 ms [6].

The sampling time of the plant controller depends primarily on the employed communication technology, i.e. protocols and signal delays. As for instance, in the case of STATCOM application within the plant for reactive power compensation, *Modbus/TCP* as commonly used communication protocol for the parameter exchange with STATCOMs processes the data query within 60 ms, without considering wire delays [7]. Hence, the control sampling time needs to take on a larger value. However, the upper boundary is determined by the system dynamics. Grid codes specify certain requirements (such as delay, rise & settling time and overshoot) for the dynamic response of reactive power for voltage control [8]. Considering the resulting system bandwidth, the sampling time must not violate the Nyquist criterion stating that the sampling rate needs to be at least two times of the highest frequency component in the system in order to avoid aliasing effect.

Furthermore, the analog system in s-domain needs to be transformed to a digital system in z-domain with $z = e^{sT_s}$. The discrete-time equivalents can be derived by different approximation methods. The most commonly applied are Forward Euler, Backward Euler and Trapezoidal (Tustin) method. When mapping to the z-plane (Fig. 3), some remarks can be stated regarding the stability of each approximation methods which should be regarded for the system implementation: For Forward Euler it is possible that a stable continuous-time system is mapped into an unstable discrete-time system. For Backward Euler a stable continuous-time system will always give a stable discrete time-system. The advantage of using Tustin's approximation is that the left half s-plane is transformed into the unit disc in the z-plane [9].

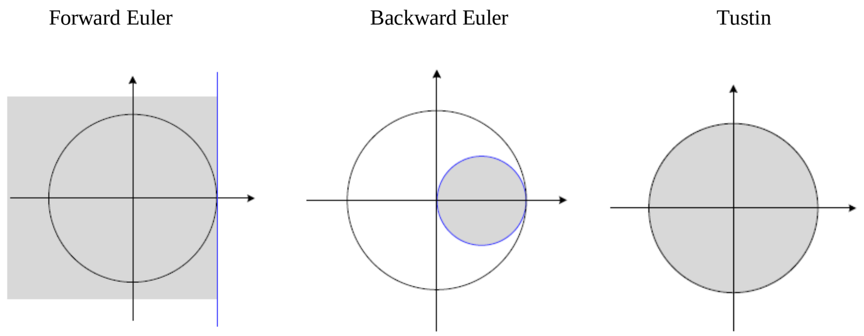

Fig. 3 Possible regions for discrete approximation methods in z-plane [9]

3.2 Asset Communication

For HIL testing signals need to be exchanged between *Real-Time Simulator* and *Renewable Plant Control*. Voltage and current values obtained by *ePHASORsim* simulations are provided to the controller, which computes and sends the reactive power reference signals to the WTs simulated in RT. Two popular transport protocols, TCP and UDP, are available for the signal exchange. They offer different level of data transportation service quality to the application.
TCP includes mechanisms to acknowledge and retransmit data, such that data is reliably transmitted and received in causal order. However, due to retransmissions of lost data packets, TCP generally suffers in delays when packets are dropped [10]. Standardized protocols such as IEC 61850 can run over TCP networks.
UDP is simpler and faster due to the lack of functionality and can be easily implemented in *MATLAB/Simulink* based simulation platforms. Transportation using UDP are the best effort and the application has to accept that packets (or data) may very well be lost in the network, e.g. dropped by a router or arrive in different order than being sent from the source [10].
The general requirement for realizing smooth signal exchange between the assets is to avoid queuing effects of received data. Attention should be paid when selecting the sampling rates of data sending and receiving process respectively, so that at a certain instant of time asset 1 is ready to receive and process the data being send from one asset 2.

4 Model Performance and Implementation Challenges in RT Simulator

The model development and implementation for the *Real-Time Simulator* is realized via *RT-LAB*, an open RT simulation software environment using the model-based design approach. It is fully integrated with *MATLAB/Simulink* by linking code generated with the *Simulink* coder to highly-optimized runtime libraries. When developing the discrete model, there are many aspects to be considered based on the requirements outlined in the previous chapter to ensure proper performance after transferring the executable file to the RT target.

4.1 Sampling Times and CPU's Model allocation

In order to capture the relevant dynamics for each part of the WPP voltage control system (see Fig. 1), the multi-rate function is applied where the model contains an algorithm running at sampling time T_s and other algorithms running at integer multiple of T_s. In *Simulink* this is realized by using so called *Rate Transition* blocks which transfer data from the output of a block operating at one rate to the input of a block operating at a different rate. The protected / non-deterministic option should be used. In this mode, data integrity is protected by double-buffering data transferred between rates. The blocks downstream from the Rate Transition block always use the latest available data from the block driving the Rate Transition block [11]. Deterministic data transfer should be avoided, since the involved signal delay would alter the desired model performance significantly for the transition from large step sizes to small step sizes.
Following sampling times are chosen for the WPP voltage control model:

Tab. 3 Model Sampling Times for Wind Power Plant Voltage Control

Wind Power Plant Controller	Wind Power Plant Network	Wind Turbines
$T_{s,ctrl} = 100\ ms$	$T_{s,grid} = 10\ ms$	$T_s = 1\ ms$

The WTs are represented by a current source model according to [2], where the grid-side converter of a type-4 WT and its control loops are captured being sufficient for voltage control analysis. The chosen sampling time is related to the smallest time constant of the model, i.e. the inner current control loop with $\tau_{min} = 4\ ms$. Then, according to [12] the sampling time should be as per Eq. 1.

$$T_s \approx \frac{\tau_{min}}{3} \approx 1\,ms \qquad (1)$$

For such small sampling times the simulation model may violate the real-time requirement leading to overruns, if the complete system is computed on single CPU core of the *Real-Time Simulator*. In this case, the computation blocks can be split into different computation subsystems. Each of the computation subsystems will be executed on one CPU core of the RT target. Fig. 4 shows the system implementation of the WPP voltage control architecture presented in Fig. 1.

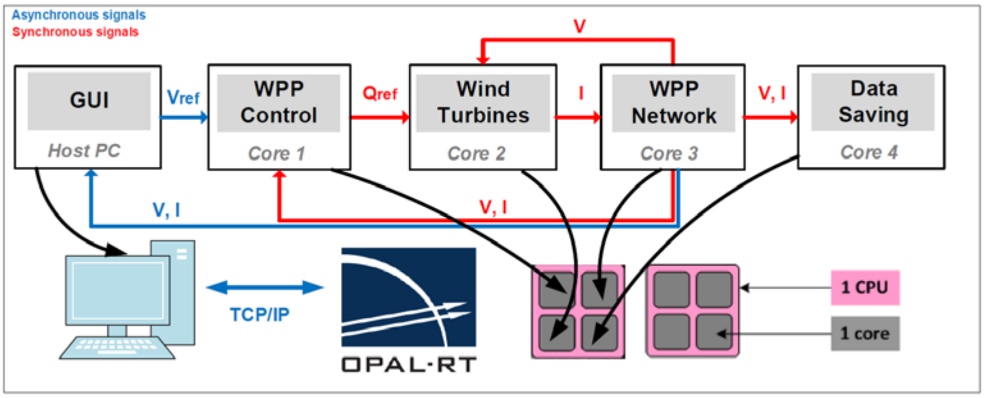

Fig. 4 Wind Power Plant Voltage Control - Implementation on *Real-Time Simulator*

There is one GUI subsystem being displayed on the Host PC, which in this framework acts as control centre, sending reference signals to the WPP Controller, and simultaneously enabling the user to monitor the simulation output. The data between computation subsystem and GUI subsystem is exchanged asynchronously through the TCP/IP link. The data between two computation subsystems is exchanged synchronously through shared memory. In such multi-core operation there is always one Master subsystem and one or several Slave subsystems. When a synchronous signal is transmitted between Master and Slave subsystem, a starting point for the real-time computation must be given by adding a delay block on the signal. In this way, parallel computation is maximized, since both subsystems can execute their computation at the same time within the time-step [13]. However, as mentioned previously, such delay signal should not alter the model performance. Hence, it should be implemented for the smallest step size $T_s = 1\,ms$, so that the overall voltage control performance running with $T_{s,ctrl} = 100\,ms$ is not affected.
Opal-RT provides an alternative or additional approach to prevent overruns in the system by activating the extreme high performance mode (XHP mode) for specific cores. It disables some unnecessary background tasks by deactivating the default operating system scheduler and thereby allowing smaller step sizes of the simulation model.

4.2 Discretization Method

Different discretization methods are implemented for the WPP Controller, as its largest sampling time determines the overall performance of the system. It has been detected that Forward Euler method constitutes an unstable system leading to termination of the RT simulation. However, Backward Euler and Tustin method has been implemented successfully and their impact of the dynamic performance is illustrated in Fig. 5.

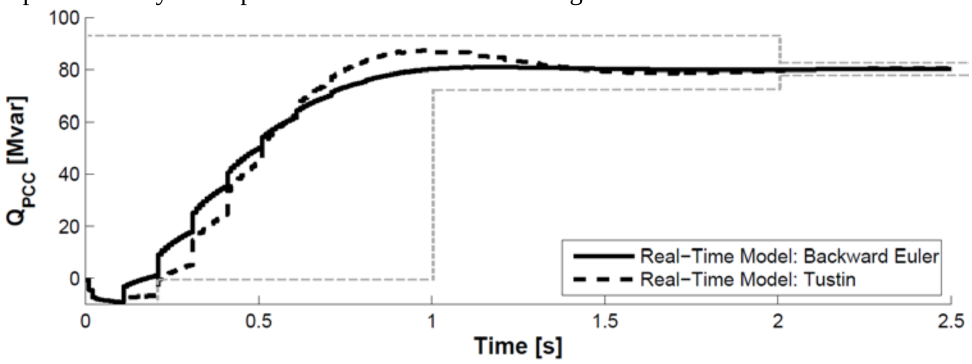

Fig. 5 Reactive power response for a grid voltage disturbance for different discretization methods

It depicts a case where a sudden grid voltage drop leads to reactive power injection by the WTs for supporting the grid voltage to reestablish closed to its nominal value. The WPP controller determines the dynamic response which underlies certain performance criteria (such as delay, rise & settling time and overshoot) indicated by the grey dashed envelopment. It can be observed that the discretization technique has significant influence on the performance, as Tustin method tends to overshoot in contrast to Backward Euler.

4.3 Asset Communication

Fig. 6 describes the system architecture, when the entire WPP voltage control system is implemented in the *Real-Time Simulator* and *Renewable Plant Control*. For realizing a RT-HIL system, the WPP Control subsystem (Core 1) is replaced by a subsystem containing UDP/IP links to communicate with *Renewable Plant Control*, now incorporating the WPP controller. Data reception (Q_{ref} signals) and transmission (V & I signals) is managed by UDP Send / UDP Receive blocks executing asynchronous processes, whose configuration needs to account for the following aspects:

- **Sampling rate of data input / output:** The sampling rate of the receiving end (*Real-Time Simulator*) should be larger than at the sending end (*Renewable Plant Control*) and vice versa, as queuing effects at the receiving end should be avoided to ensure smooth data transmission without signal delays.
- **Data format:** The network byte order (endianess) and data representation in memory (16-bit, 32-bit etc.) should be harmonized on both ends of UDP-IP link to ensure correct network data exchange.

Further attention has to be paid in case of activated XHP mode, since it limits asynchronous applications such as UDP/IP communication. Hence, XHP mode should be deactivated for a subsystem containing UDP/IP links.

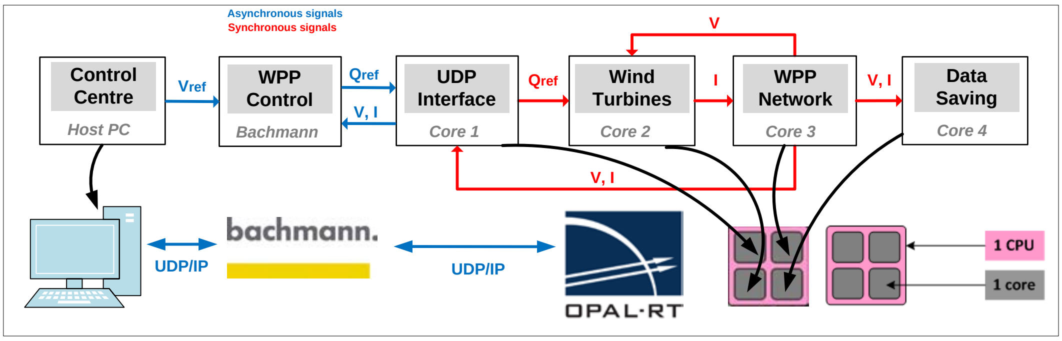

Fig. 6: Wind Power Plant Voltage Control - RT-HIL Implementation

5 Conclusion

This paper has summarized the crucial aspects regarding the model development and practical system implementation of a WPP voltage control architecture into a RT-HIL framework. The authors intend to address two main messages to the audience: a) that one needs to contemplate a discrete time-domain simulation model differently and with additional attention compared to well-known continuous time-domain models and b) that real-time requirements are crucial for HIL testing with regard to computational performance of the hardware equipment and the communication infrastructure between *Real-Time Simulator* and *Renewable Plant Control*.

Future work is intended for testing the performance of WPP voltage control when being subject to communication delays or packet losses, in this way aiming to validate further the analytical results for a tuned control system in [2]. Moreover, a large focus will be laid on the implementation of IEC 61850 communication standard as being one of the promising candidates for future standardized supervisory control and data acquisition in the context of increased demand of asset communication (i.e. distributed generation, DSO, TSO) in today's power systems.

6 References

[1] J. Martinez, P. C. Kjær, P. Rodriguez, and R. Teodorescu, "Design and analysis of a slope voltage control for a dfig wind power plant," IEEE Transactions on Energy Conversion, vol. 27, no. 1, pp. 11–20, 2012.

[2] L. Petersen and F. Kryezi, Wind Power Plant Control Optimisation with Embedded Application of Wind Turbines and STATCOMs. Aalborg University, Department of Energy Technology, ISBN: 978-87-92846-55-6, 2015.

[3] Wind Energy Research Strategy, Danish Research Consortium of Wind Energy, 2015

[4] AAU, Smart Energy Systems Laboratory, Aalborg University. http://www.smart-energy-systems-lab.et.aau.dk.

[5] J. Bélanger, P. Venne, J.-N. Paquin, "The What, Where and Why of Real-Time Simulation", Planet RT, 2010

[6] IEC 61400-27-1:2015, Wind Turbines – Part 27-1: Electrical Simulation Models

[7] M. Hewitt, "Technote 42 – Modbus RS-485 Timing Issues", Obvius, 2012

[8] "National Grid Electricity Transmission The Grid Code", Issue 5, Revision 13 2015, National Grid Electricity Transmission Std.

[9] W. S. Levine, "The Control Handbook", CRC press, 1996

[10] Deliverable D1.2 "Technical Feasibility of Ancillary Services provided by ReGen Plants", RePlan Project *Ancillary Services from Renewable Power Plants*, AAU – DTU – Vestas, 2015

[11] *MATLAB/Simulink* Help Documentation, TheMathworks Inc.

[12] Y. Zhu, "Multivariable System Identification", Pergamon, 2001

[13] Opal-RT Knowledge Base. http://www.opal-rt.com/KMP