Real-Time Simulation of Doubly Fed Induction Generator for Wind Turbine Applications

Christian Dufour , Jean Bélanger

Opal-RT Technologies inc.
1751 Richardson, bureau 2525
Montreal Qc H3K 1G6
email: christian.dufour@opal-rt.com
Tel. : 514-935-2323 Fax : 514-935-4994 www.opal-rt.com

Abstract - This paper describes a real-time simulator of wind turbine generator system suitable for controller design and tests. The simulated generator is a grid-connected doubly fed induction machine with back-to-back PWM voltage source vector control of the rotor. The simulator is based on RT-LAB real-time simulation platform that allows for easy model-to-real-time-target design from Simulink models. The paper puts special emphasis on the fixed-step simulation problematic of kHz-range PWM inverter drives and the techniques used in the real-time simulator to accurately simulate those drives.

***Keywords* – real-time simulation, doubly fed inductor machine, wind turbine.**

Introduction

The past decade have seen increased interests in wind energy generation as environment concerns are on the rise. With megawatt-range wind turbine now manufactured by several companies, the need for rapid prototyping and testing of such apparatus is also increasing. The traditional way of testing, integration and validation of complex controlled consists on the systematic analysis of the behavior of individual components, mostly by simulation, before complete integration on real apparatus. At this stage, real cautions are to be taken because of the power levels: a simple controller malfunction can damage the prototype or the real system and create project delays and cost increases. A more advanced testing/integration approach is needed to diminish the probability of damage, personal injuries and time to market.

With rapid advances in computer technologies, it is increasingly advantageous to make a more efficient approach to system prototyping using Hardware-in-the-Loop (HIL) digital simulation, where one or more devices (wind turbine controllers) are tested while connected to a real-time dynamic equivalent of plant (DFIM, inverters, wind-turbine, grid model). If the system is correctly modeled, the wind generator controllers under test will behave as if they were connected to the real system. The tested controllers can therefore be tested over a wide range of parameters without risk to the main system.

Testing of electrical system controllers is particularly challenging because of the high bandwidths required and dynamic “stiffness” of electrical system with regards to the current time step achievable in real-time simulation. Even with modern GHz-speed processor technology, HIL simulation at time steps of 10 μs are just about achievable due to inherent latencies in I/O access and inter-processor communication. In low cost PC-based real-time simulator, a cause of those latencies is the persistence of the CPI bus standard in standard PC architecture which frequency has not increased at fast as processor speed. This time step limitation is unacceptable for such demanding applications as motor-IGBT-drive systems [6] or high-frequency DC-DC converters[7]. In these cases, power electronics switching must be sampled precisely to ensure proper calculation of the motor or inductance fluxes, and most often calls for a switching time resolution well below 1 μs. While using faster I/O and inter-processor communication hardware can shorten the simulation time step, resolving the switching events to sufficient accuracy still remains a problem. The solution presented in this paper is to sample the I/O IGBT gate signals with high frequency counter cards and insert the timing information into the simulation.

In this paper, we describe a doubly fed induction generator wind turbine real-time simulator with advanced hardware-in-the-loop capabilities. The wind-turbine plant, shown in Figure 1, is based on a doubly fed induction machine with back-to-back PWM inverters at the rotor side and is modelled with RT-Events[4] and ARTEMIS, two Simulink blocksets designed for compensated fixed-time step simulation from Opal-RT Technologies and SimPowerSystems[5]. Real-time target code is generated with Real-Time Workshop and RT-LAB[3] real-time simulation software. The simulator is suitable for the design and test of wind-turbine apparatus and controllers.

The paper has the following structure. General principles and objectives of wind energy conversion are first put forward. We will then describe the doubly fed induction generator and the back-to-back PWM Sherbius topology used in this paper. The simulator architecture along with the RT-LAB real-time simulation platform characteristics will follow. This will include some explanations about the importance of precise fixed-time step modelisation tools like RT-Events and in particular some bridge model called Time Stamped Bridges. For the purpose of validating the DFIM plant model, two vector controllers were designed in Simulink for the DC-link voltage control and the rotor PWM control and they are explained next. The last section is about experimental results made with the simulator to demonstrate the effect of the Time Stamping technique for high frequency PWM inverters and the real-time simulation speed.

Variable speed generator and wind energy harnessing

Variable speed generator permits maximum energy capture from a wind turbine. On one part, the mechanical power of a wind turbine that can be extracted highly depends on wind speed. This can be view for example on the wind turbine power curves of Fig. 1. For each wind speed, there is therefore an optimum turbine speed for wind power extraction. This turbine also moves the rotor of the electromechanical conversion device.

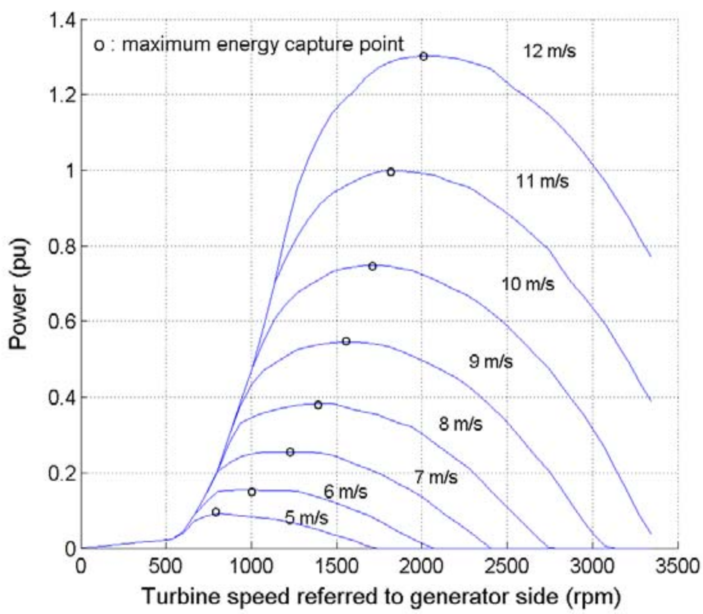

Fig. 1. Example of wind turbine power curves

In the doubly fed induction machine generator, the rotor in coupled to the turbine shaft through a gearbox while the stator is connected directly to the fixed frequency grid. The normalized difference between the stator electrical frequency and the rotor speed (scaled to the number of pair of poles of the machine) is called the induction machine slip and determines in part the power balances in the machine.

$$P_R = slip * P_S \tag{1}$$

$$P_{mech} = (1 - slip) * P_S \tag{2}$$

Depending on slip, some power goes through the rotor circuit and we need not loosing it to keep the benefit of optimum wind power capture. With the proper control, a back-to-back PWM inverter connected to the rotor (through collector rings) can accomplish just that. This technique is also referred as slip energy recovery.

Controlling the machine power flows from the rotor is also advantageous from a cost perspective. By limiting the machine slip, and therefore the wind turbine speed, the power process by the rotor converter is only a fraction of the generator total power (by direct virtue of Equation 1). The rotor inverter ratings and cost will also be down for this reason. Limiting the rotor speed makes senses because there is little energy to be extracted from low wind speed anyway.

In comparison, grid-connected wind turbine system with squirrel cage rotor must limit slip to small value because power transferred to the rotor circuit is simply dissipated in it. Some control must therefore be done to keep the wind turbine speed constant (ex: blade pitch control) and energy extraction is non-optimal for most wind speed. Several other benefits can be obtained from the usage of variable speed generator for wind turbine and the reader is referred to [2] for further details.

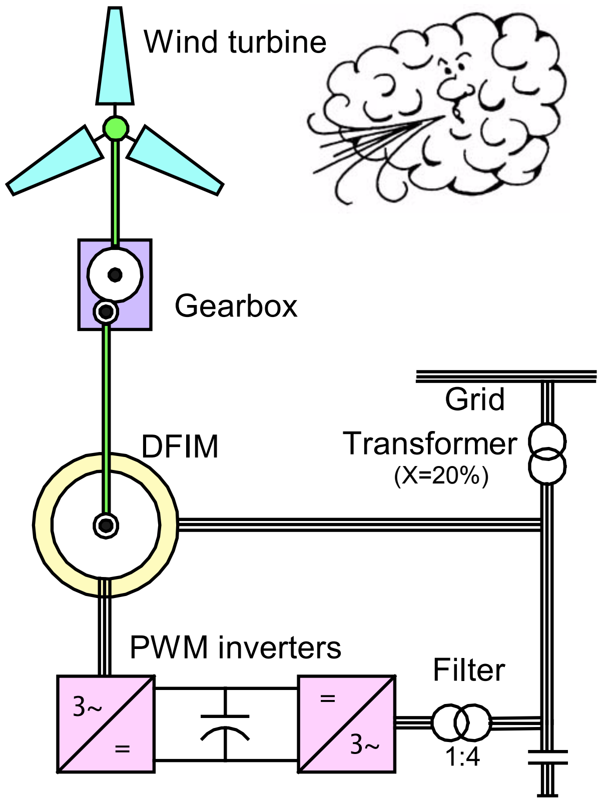

Fig. 2. Doubly-fed induction machine wind generator circuit

The inclusion of rapid controller prototyping and hardware-in-the-loop testing in the design cycle

As power ratings and costs of wind turbine are continually on the rise, validation and testing need follow the same trend. The machine controllers are to be in special scrutiny because some simple malfunction from them could damage the wind turbine apparatus or prototype and cause personal injury and development delays.

Rapid controller prototyping and hardware-in-the-loop testing are two ways to increase the reliability and development speed of machine controllers. Both techniques use the code generation capability of software like Real-Time Workshop from The Mathworks and RT-LAB from Opal-RT Technologies that transform high-level simulation Simulink schematics into executable code to run on a specified target platform.

In rapid controller prototyping, one generates its controller code directly from the simulation software schematic. The code is generated specifically for the type of controller used. Low-cost applications will use fixed-point DSP code and the code generator must produce fixed-point code. Higher cost applications may directly use a real PC as a controller and in those cases floating-point code can be generated and use directly. An example of the later case is Bombardier JetTrain, which uses RT-LAB on a PC under the QNX operating system for the turbine controller. Using rapid controller prototyping can obviously diminish the development time by minimizing manually produce code that is prone to error.

In hardware-in-the-loop testing, one generates this time the plant code for execution into a real-time simulator. In wind turbine applications, the plant is the induction machine, wind turbine and everything that run at high power levels like the grid itself. Real controllers are then interfaced with the simulator with customized I/O ports. Using hardware-in-the-loop, test engineer can thoroughly test the wind turbine and generator controller for a wide variety of operating conditions without risking damaging the real plant.

The challenges of real-time simulation of electrical systems

Making the real-time simulation of electrical systems represent a challenge for several reasons. The main reason is that electrical systems have higher bandwidth than mechanical systems and therefore command smaller simulation time steps. When including I/O access time, the RT-LAB real-time simulator can go under 10 us time step for example and, nevertheless, network equations often still remains stiff. For power network simulation, even such a small time step requires special solvers and interpolation techniques to insure accurate results.

The problem gets most when simulating power converters like PWM motor inverters and DC-DC converters because of their high switching frequencies with regards to the simulation sampling frequency, i.e. the inverse of the time step. In the case of a motor inverter simulated at fixed time step, the sampled IGBT gate signals will have poor resolution if sampled at the simulator sample time and the resulting voltage application time to the machine will be erroneous.

Interpolated fixed-time simulation with RT-Events

From the simulator point of view, fixed-time step simulation of PWM fed wind generator systems presents some challenging technical problems. One of those problem concerns the accurate time sampling of both rotor-side and DC-side inverter IGBT gate signals. Accurate motor flux integration is dependant upon the precise sampling of the IGBT gate signals by the simulator. This sampling must have sub-μs precision but typical real-time simulation time step are more in the 20-50 μs range. There are solutions to this problem.

a) In fully numerical real-time simulation where both controllers and wind power plant are simulated, the IGBT gate signals generation must be made with equivalent sub-µs resolution despites the fixed-time simulation. This requires the use of RT-Events[4], a Simulink blockset designed for interpolation of in-step events in models like the sinus-triangular comparisons occurring in PWM generation.

b) In real-time simulation of the wind power plant with interaction with an external controller, the IGBT gate signals must be sampled by high frequency counter cards and the resulting time stamp incorporated into the simulation process by some interpolation technique.

In both cases, the IGBT bridge model must be able to use this interpolation information to compensate the simulation process.

Take for example the simple chopper drive of Fig. 3 where the load current should be linearly dependant upon the chopper duty-cycle. But the simulation of this drive at 10 µs time step and a 10 kHz chopping frequency leads to gross inaccuracies when the simulation is uncompensated like in SimPowerSystems blockset with discrete simulation option (Fig. 4,curve *r*).

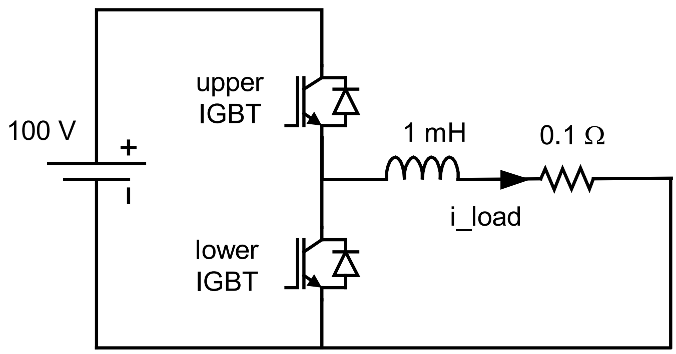

Fig. 3. Simple chopper

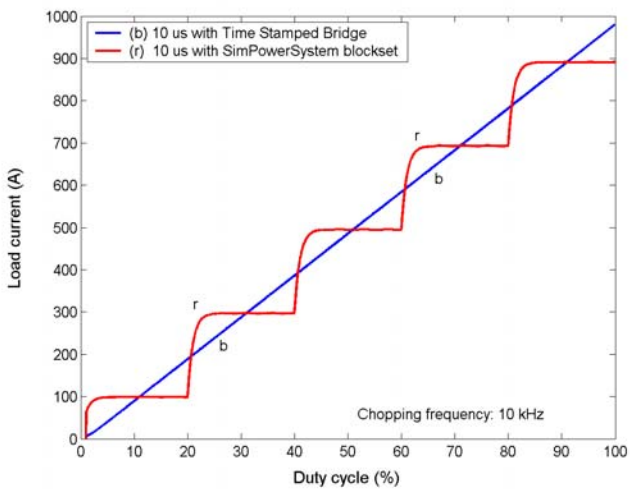

Fig. 4. Simple chopper load current as a function of duty cycle

In contrast, usage of compensated fixed step simulation tools like the RT-Events blockset from Opal-RT technology lead to accurate simulation (Fig. 4,curve *b*). This works because RT-Events blocks are built to propagate zero crossing information at fixed time step and the Time Stamped Bridge are designed to use this information to produce compensated output voltages.

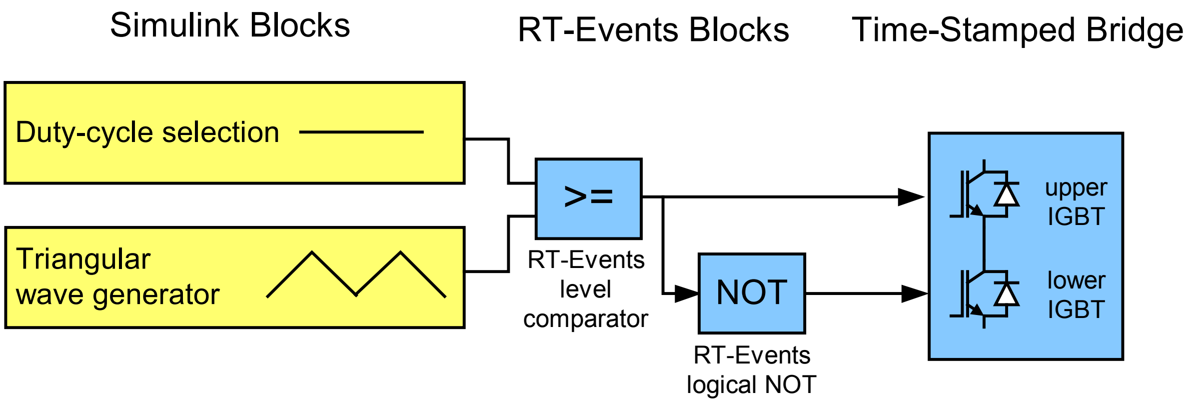

Fig. 5. Simulink blocks, RT-Events logical blocks and Time Stamped Bridge interconnection.

Interpolated Hardware-in-the-Loop simulation with time-stamped digital I/O

When the simulated plant and Time Stamped Bridge are fed from a real controller, the same problematic still remains. If the IGBT gate signals are sampled at the simulator sample time (ex: 10 µs) without compensation, gross error will results.

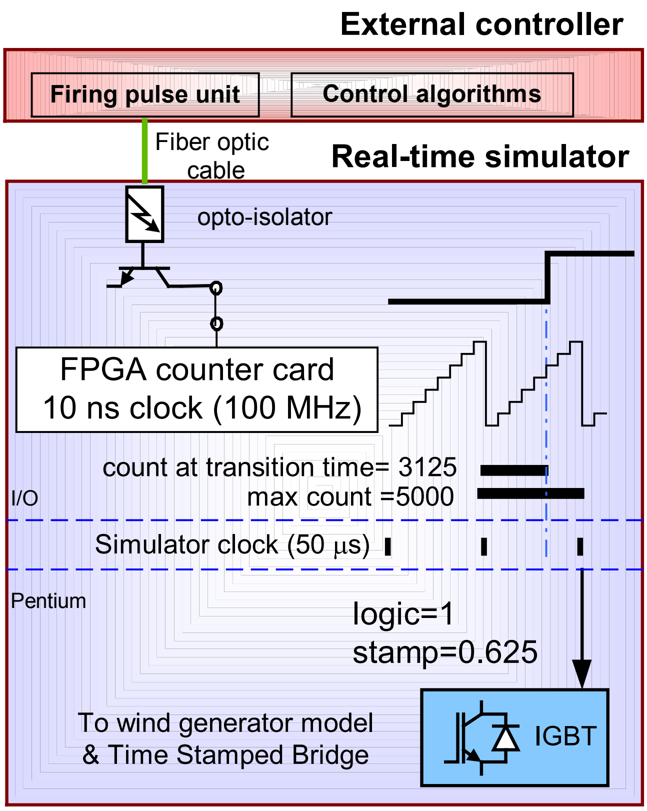

Fig. 6. Time stamping of a real controller gate signal

The solution is explained in Fig. 6. The gate signals coming in the simulator from the real controller are sampled with a high frequency counter card running at much faster rate than the simulation process. In Fig. 6, the simulation rate is 50 µs and the counter card run at 100 MHz so the card can count to 5000 each time step. When some transition occurs on the gate, the counter card stop counting and therefore 'stamps' the time of the transition with 10 ns resolution. The count is then transferred to the Time Stamped Bridge as a normalized ratio (3125/5000=0.625) on the Pentium side of the simulator where the wind generator is simulated.

RT-LAB Electrical Drive Simulator

The RT-LAB Electrical Drive Simulator hardware used for wind turbine generator testing is shown in Fig. 7. Its is composed of a dual Pentium Xeon PC running at 2.4 Ghz under RT-Linux operating system and a console PC running Windows XP. The real-time model of the wind-turbine generator, described in Fig. 2, and its numerical controllers runs on each CPU of the Dual Pentium PC while simulation control and monitoring is made through the console computer.

One CPU of the dual-CPU computer iterates the DFIM wind turbine generator circuit equations while the other one computes the numerical controller values when no external controller is connected to the simulator. In the case of hardware-in-the-loop testing with real external controllers, the CPU in charge of DFIM equation also controls the Opal FPGA card that read PWM signals generated by the real controllers and generates the voltage and current analog signals and the digital position encoder signals necessary to the controllers. The Opal FPGA board is also configured to read the IGBT gate signals generated by the external controllers with a sampling precision near 10 ns. This FPGA card is built around the XILINX VITEX II Pro board and also comes with D/A converters. From the console, the user can set-up various test scenarios or evaluate controller performance in case of network fault or for different wind profiles.

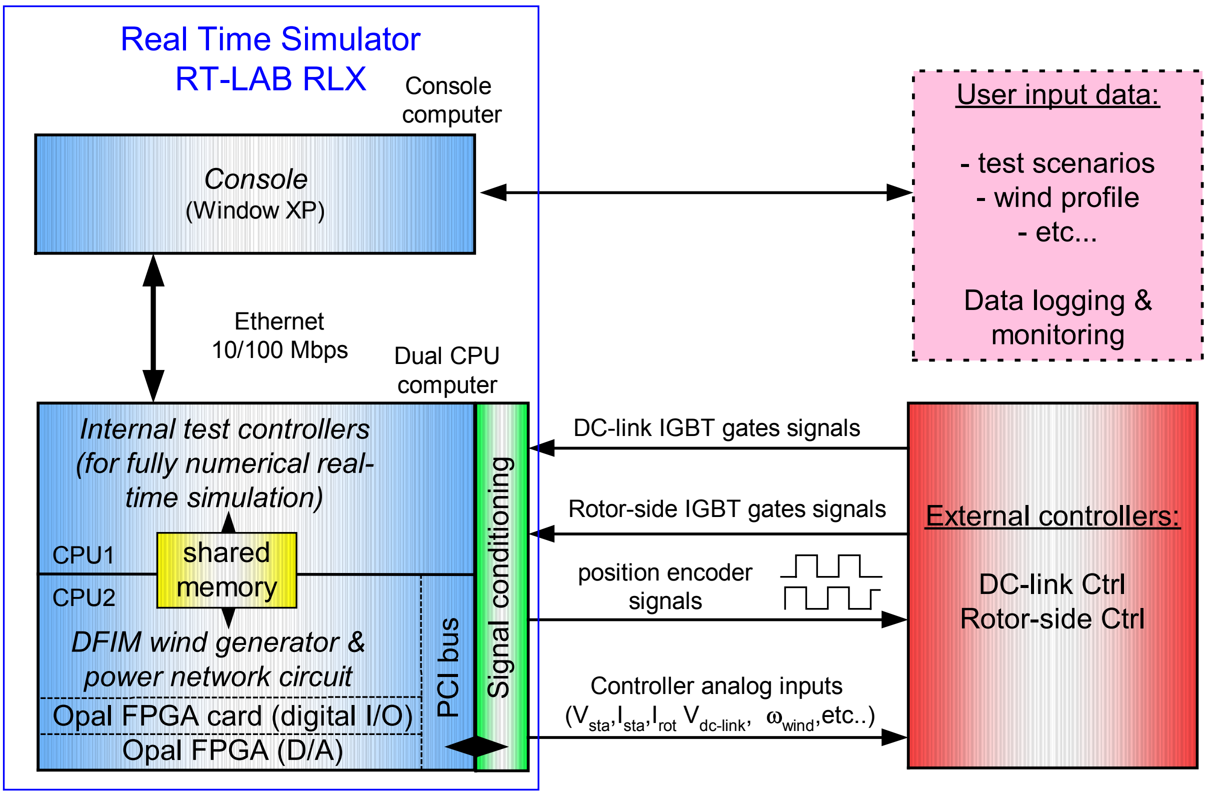

Fig. 7. Structure of the RT-LAB wind turbine generator real-time simulator

Modelling of the DFIM-based wind-turbine generator and controllers

The wind-turbine generator and grid circuit simulated in this paper is described in Fig. 2. The inductive grid, transformers and inductive machine are modelled in SimPowerSystems. The two PWM inverters are modeled with Time Stamped Bridges from RT-Events. The 7.5 kW doubly fed induction machine with back-to-back PWM inverters is based on the paper by Pena[1] and used the exact same parameters except for the controllers sampling time, compensator gains and PWM carrier frequency. Unless otherwise specified, the PWM carrier frequency is set to 2 kHz and the simulation fixed-time step is 50µs.

The doubly-fed induction machine model

The doubly fed induction machine is modelled using SimPowerSystems blockset and implements the well-known Park induction machine model with a stationary reference frame.

$$\begin{aligned}
v_{ds} &= Ri_{ds} + \dot{\psi}_{ds} \\
v_{qs} &= Ri_{qs} + \dot{\psi}_{qs} \\
v_{dr} &= Ri_{dr} + \dot{\psi}_{dr} - \omega_r \psi_{qr} \\
v_{qr} &= Ri_{qr} + \dot{\psi}_{qr} + \omega_r \psi_{dr} \\
\psi_{ds} &= L_s i_{ds} + L_m i_{dr} \\
\psi_{qs} &= L_s i_{qs} + L_m i_{qr} \\
\psi_{dr} &= L_r i_{dr} + L_m i_{ds} \\
\psi_{qr} &= L_r i_{qr} + L_m i_{qs} \\
T_e &= 1.5p * (\psi_{ds} i_{qs} - \psi_{qs} i_{ds})
\end{aligned} \tag{3}$$

The model is discretized by the Forward Euler method. It is worth noting that the SimPowerSystems induction machine model assumes for Y-connected windings with unitary stator-rotor turn ratio. To model an induction machine with unequal stator and rotor turn number, the voltage applied and the currents read at the rotor terminal must be scaled accordingly.

The SimPowerSystems blockset has also been modified by the authors so its speed can by fixed, thus disabling the mechanical equation of the model when this option is selected. Direct control of the rotor speed is sometimes useful for some test scenarios that does not involves the mechanical part of the wind-turbine generator.

Control of the doubly-fed induction machine

The control of doubly-fed induction generator with back-to-back PWM scheme is based on the paper from Pena[1] and is constituted of two main parts: the DC-link voltage regulator and the rotor voltage controller. The DC-link voltage controller is a vector controller that permits keeping constant the DC-link voltage level. The rotor-side controller is a stator flux based vector controller that enables independent control of the induction machine active and reactive powers.

DC-link voltage control

The DC-link voltage control objective is to maintain the DC-link voltage to a constant value despites machine power flow fluctuations. In analogy to the fundamental power network equation,

$$P = \frac{V_1 V_2 \sin\theta}{X},$$

the objective of the control can be viewed as forcing V_{al},V_{bl},V_{cl} voltage through PWM with the required phase shift from the supply voltage that generates the necessary power flow to keep the DC-link voltage constant.

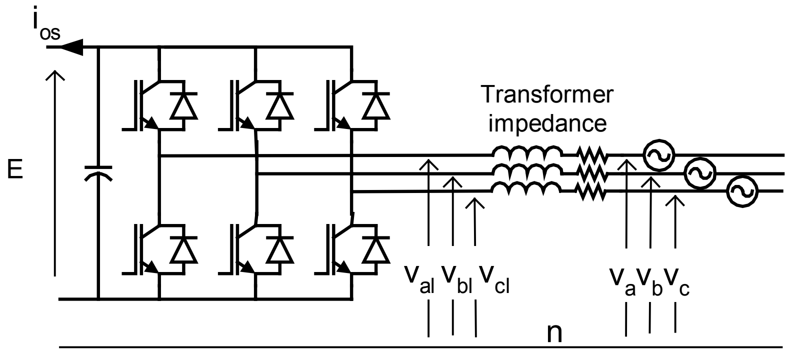

Fig. 8. The DC-link and its network feed

More specifically, the voltage across the inductor is described by the equations:

$$V_{abc} = Ri_{abc} + L\frac{di_{abc}}{dt} + V_{abcl} \tag{4}$$

This equation is then transformed into a *dq* reference frame rotating at the grid frequency ω_e

$$V_{dq} = Ri_{dq} + L\frac{di_{dq}}{dt} + L\begin{bmatrix} 0 & -\omega_e \\ \omega_e & 0 \end{bmatrix} i_{dq} + V_{dql} \tag{5}$$

The active and reactive power flows are then:

$$P = 3(v_d i_d + v_q i_q) \qquad Q = 3(v_d i_q + v_q i_d) \tag{6}$$

The angular position of the supply voltage is computed as

$$\theta_e = \int \omega_e dt = \tan^{-1}(\frac{v_\beta}{v_\alpha}), \tag{7}$$

where v_α and v_β are the α and β components (Clarke transform, stationary 2-axis) of the stator voltage. Aligning the Park reference frame on v_d makes $v_q = 0$. Also, since the supply voltage has constant amplitude, v_d also have constant amplitude. For active power to balance at the DC-link, we have (neglecting harmonics):

$$Ei_{os} = 3v_d i_d \tag{8}$$

and the DC-link voltage can therefore be controlled by i_d. The i_q value is used to control the reactive power flow into the DC-link. The DC-link voltage was regulated at 550V and a transformer with a turn ratio of 4 was used to feed it. The DC-link controller is described in detail in Fig. 9.The DC-link controllers run at 50 µs sample time and continuous time proportional-integral compensators have been used. The current controllers have both proportional and integral gain of 100, while the voltage controller has proportional and integral gains of respectively 3 and 10.

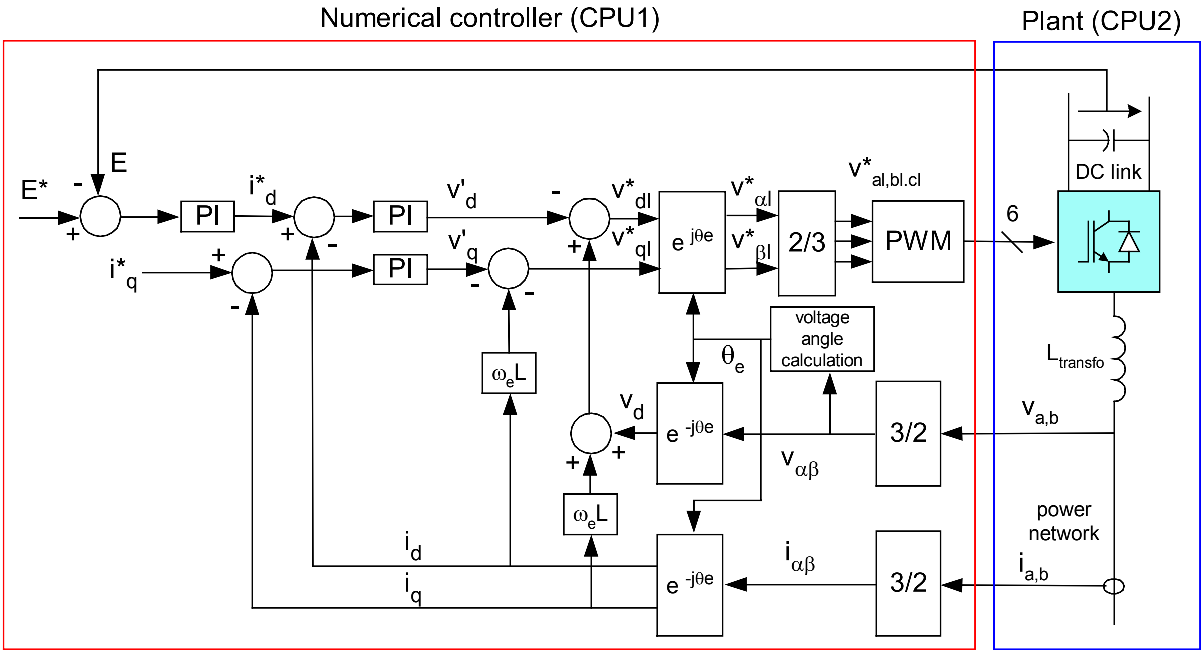

Fig. 9. DC-link controller connected to the DC-link

Induction machine control

The objective of the induction machine controller is to keep the machine rotor speed near the optimum wind energy capture point on the wind turbine characteristic by controlling active and reactive power flows at the stator. The induction machine is controlled in a synchronously rotating *dq* axis frame with the *d*-axis aligned along the stator-flux position. This permits decoupled control of electromagnetic torque and rotor excitation currents. The controller is shown in Fig. 10 and is based on the following equations under the assumption of constant stator flux:

$$v_{dr} = R_r i_{dr} + \sigma L_r \frac{di_{dr}}{dt} - \omega_{slip} \sigma L_r i_{qr} \tag{9}$$

$$v_{qr} = R_r i_{qr} + \sigma L_r \frac{di_{qr}}{dt} + \omega_{slip} (\frac{L_m}{L_s} \lambda_{ds} + \sigma L_r i_{dr}) \tag{10}$$

$$T_e = -1.5P \frac{L_m}{L_s} \lambda_{ds} i_{qr} \tag{11}$$

$$\omega_{slip} = \omega_e - \omega_r \qquad \sigma = 1 - L_m^2 / (L_s L_r) \tag{12}$$

where $v_{dr}, v_{qr}, i_{dr}, i_{qr}$ are the *dq*-axis rotor voltages and currents, R_r, L_r are the rotor resistance and inductance, L_s is the stator inductance (referred to the rotor), P is the number of pair of poles, λ_{ds} is the stator d-axis flux and L_m is the rotor-referred mutual inductance of the machine (note that in Pena this last value is referred as L_0). Details of the controls are explained in depth in Pena [1].

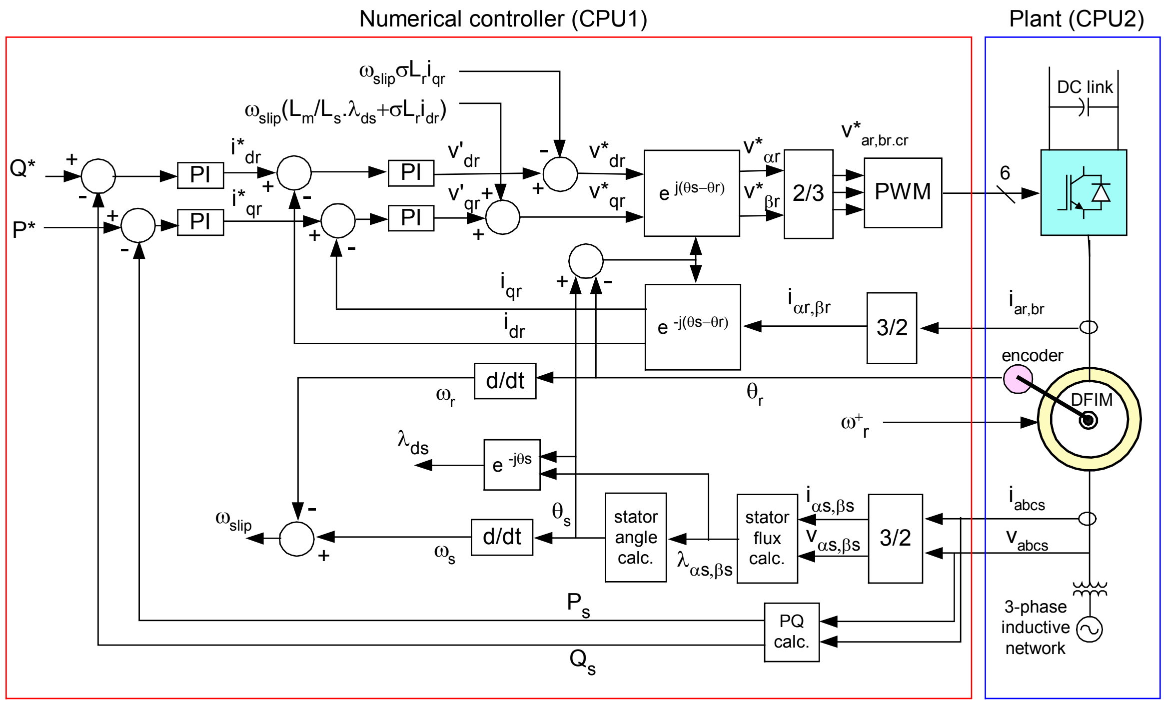

Fig. 10. Doubly fed induction machine rotor PWM controller

The controllers run at 50 µs sample time and continuous time proportional-integral compensators have been used. The current controllers have proportional and integral gains of respectively 50 and 100, while the power controllers have proportional and integral gains of respectively 15 and 100.

Experimental results

This section will show test made with the real-time simulator using the internal controllers. The first test will demonstrate the ability of the rotor-side controller to run across the synchronous speed. The last test will show the effect of the time stamping technique on the simulation accuracy.

TEST 1: Operation across synchronous speed

The first test shows the operation of the wind generator when the rotor speed goes from hypersynchronous to subsynchronous operation. The rotor speed scan has been made in a time lapse of 2 sec. The capability of the DFIM with back-to-back PWM inverters to work easily across the null slip point is an interesting aspect of this topology. Fig. 11 shows the rotor currents during the rotor speed change. While changing the rotor speed, the power controllers try to keep the stator active and reactive powers at their commanded values with some success (Fig. 12).

On Fig. 11, the rotor current for phase A is superposed with the case where the rotor-side inverter is modelled with Simulink SimPowerSystems blockset working in discrete mode. A section of the simulation is zoomed and shows that SimPowerSystems simulation for the rotor-side inverter is the source of inaccuracies. Those inaccuracies could be worst if one would remove rotor inductances (32 mH/phase) that Pena used to minimize rotor current oscillations in hypersynchronous mode. Those inductances are present in the DFIM model simulated in this paper. By contrast, usage of the RT-Events Bridges (also called Time Stamped Bridges), results in smoother rotor currents more compatible with the 2 kHz carrier frequency of the inverter.

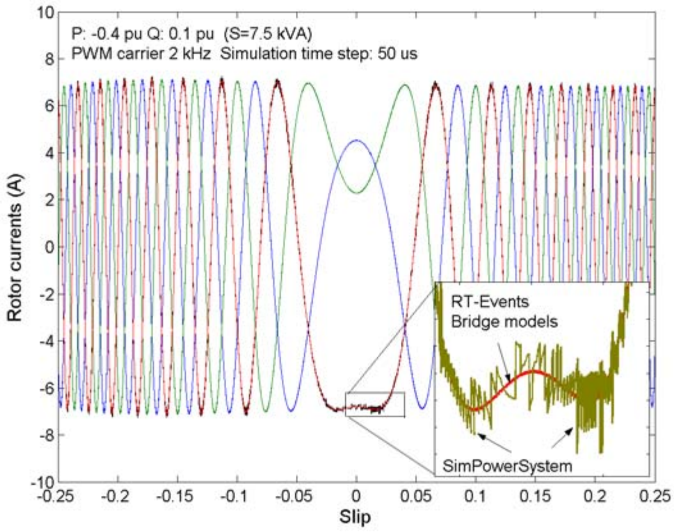

Fig. 11. Rotor currents for operation across synchronous speed

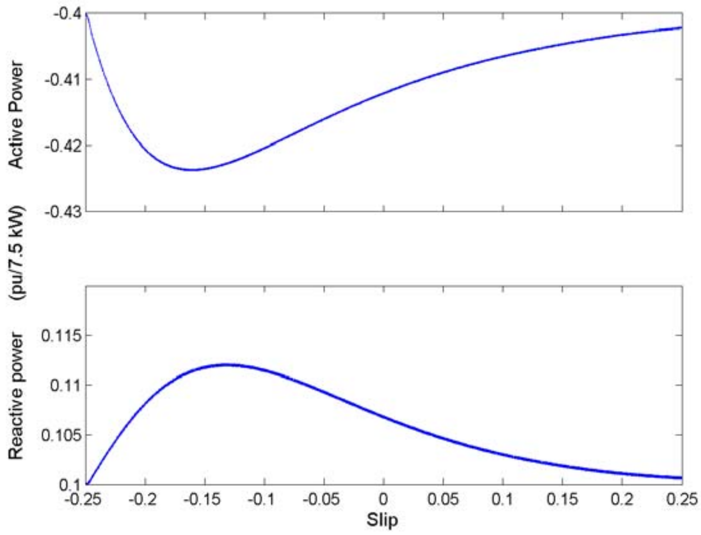

Fig. 12. Powers at stator during slip scan

TEST 2: Accuracy effect of the time stamping technique

This test objective is to show the effect of the time stamping technique on simulation accuracy. The wind generator plant and controllers are run with constant active power and reactive power. The PWM carrier of the rotor-side inverter is still 2 kHz, that is 1/10 the sampling frequency. The test compares the accuracy of the Time Stamped Bridges vs. SimPowerSystems bridge models for the rotor-side and DC-link inverters.

At steady state operation, Fig. 13 compares stator current and active power the Time Stamped Bridges are used or not. The green curve *g* shows the stator active power when a SimPowerSystems is used for the DC-link inverter. The curves show strong jitter-like noise that is absent when a Time Stamped Bridge is used for this inverter (Fig. 13 curves *r* and *b*). The effect of Time Stamps is less impressive for the rotor-side inverter but there is still added noise in the *r* curve vs. *b* curve (in zoom). If the DFIM is run with a separated DC-link feed and therefore *without* filtering capacitor connected to the network, power flickering similar to curve *g* occurs if the rotor-side inverter only is modelled with SimPowerSystems.

The point is that it seems like the complete system is not working properly while in fact, only the numerical modelling technique used is faulty. An inexperienced control engineer would maybe think that his DC-link controller is somewhat unstable but this apparent instability is really caused by the uncompensated fixed step simulation process.

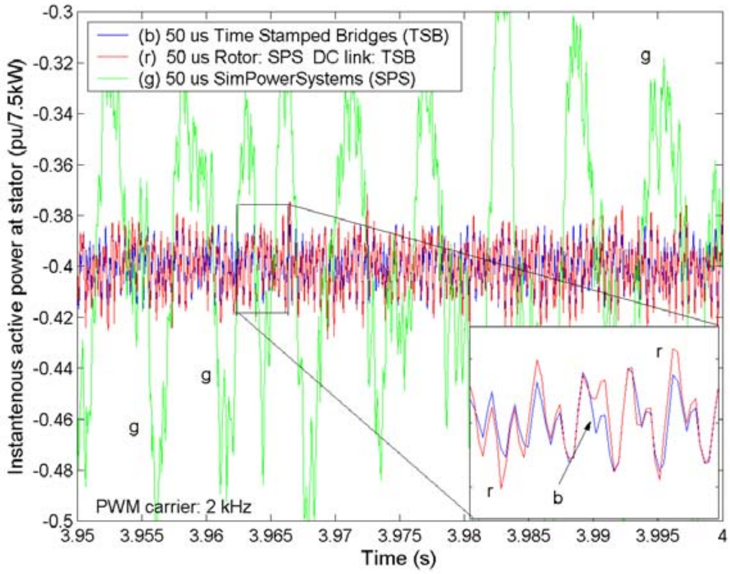

Fig. 13. Active power read at the stator with and without time stamps

Computational speed

Table 1 shows the computational time of the complete wind-turbine generator circuit (including the grid) and numerical controllers. The real-time simulation was made on a dual-CPU 2.4GHz Pentium 4 Xeon target running under RT-LAB-QNX version with the model separation described in Fig. 7. RT-LAB QNX runs on the QNX operating system as opposed to Linux OS for RT-LAB LNX. The test runs without I/O since the internally simulated controllers were used.

Minimum Real-time Step	20 µs
Plant calculation time (DFIM, grid, inverters)	14 µs
Controllers calculation time	11 µs

Table 1: Real-time step size obtainable on 2.4 GHz PC (RT-LAB QNX)

Performance results show that the model could be run at much lower rate than the 50 µs used in the tests. It also shows the simulator ability to model more complex plant models. For example, one could make the simulation of 2 wind generators for example or simulate a wind generator system with more elaborate grid models with fault capability.

References

[1] R.Pena, J.C. Clare, G.M. Asher, "Doubly Fed Induction Generator Using Back-to-Back PWM Converter and its Application to Variable-Speed Wind-Energy Generation", IEE Proc.-Electr. Power Appl., Vol. 143, No. 3, May 1996.

[2] S. Muller, M. Deicke, R.W. De Doncker, "Adjustable Speed Generators for Wind Turbines based on Doubly-fed Induction Machines and 4-Quadrant IGBT Converters Linked to the Rotor", 35th Annual Meeting - IEEE Ind. App. Soc. (IAS), Rome, Italy, Oct. 8-12, 2000

[3] RT-LAB 6.2, Opal-RT Technologies inc. 1751 Richardson, bureau 2525, Montreal, Qc, H3K 1G6 www.opal-rt.com

[4] RT-Events, v2.0, Opal-RT Technologies Inc., Montreal, Qc, Canada

[5] SimPowerSystems Blockset version 2.3 for Simulink, The MathWorks Inc. Nawick, MA, USA

[6] Peter Terwiesch, Thomas Keller, Erich Scheiben, "Rail Vehicle Control System Integration Testing Using Digital Hardware-in-the-Loop Simulation", IEEE Transactions on Control System Technology, Vol. 7, No. 3, May 1999, pp 352-362.

[7] T. Matsumoto, N. Watanabe, H. Sugiura, T. Ishikawa, "Development of Fuel-Cell Hybrid Vehicle", The 18th International Electric Vehicle Symposium, Berlin, 2001