Open Journal of Modelling and Simulation, 2014, 2, 77-89
Published Online April 2014 in SciRes. http://www.scirp.org/journal/ojmsi
http://dx.doi.org/10.4236/ojmsi.2014.22010

Physical Network Approach Applied to Wind Turbine Modeling with Simscape Language

Elhaini Jamila, Saka Abdelmjid*

Mechanical Engineering Laboratory, Faculty of Sciences and Technology FST, Road Immouzer, Fez, Morocco
Email: j.elhaini@gmail.com, *abdelmjid.saka@gmail.com

Received 10 February 2014; revised 19 March 2014; accepted 27 March 2014

Abstract

Model-Based Design is an efficient and cost-effective way to develop controls, signal processing, image processing, communications, mechatronics, and other embedded systems. Rather than relying on physical prototypes and textual specifications, Model-Based Design uses a system model as an executable specification throughout development. It supports system- and component-level design and simulation, automatic code generation, and continuous test and verification. This paper is focused firstly on the so-called model-based design and aims at presenting an up-to-date state of the art in this important field. Secondly, it develops a model based design for wind energy systems. Mathematical formulations and numerical implementations for different components of wind energy systems are highlighted with Simscape language. Finally, results are derived from simulations.

Keywords

Mathematical Modeling; Model-Based Design; Simscape; Physical Network Approach; Wind Energy Systems

1. Introduction

For a system as complex as a wind turbine, the ability to simulate the physical systems (mechanical, electrical, hydraulic, etc.) and control systems in a single environment is crucial to the development process. Model-Based Design approach, the best method for modeling complex systems, enables engineers to incorporate requirements into the development process, to design at the system level, and to predict and optimize overall system performance without relying only on hardware prototypes. It reduces development time, improves product quality and helps to lower development costs.

The contribution of this paper is three-fold. Firstly, an up-to-date of the state of the art of Model-Based-De-

*Corresponding author.

How to cite this paper: Jamila, E. and Abdelmjid, S. (2014) Physical Network Approach Applied to Wind Turbine Modeling with Simscape Language. *Open Journal of Modelling and Simulation*, **2**, 77-89. http://dx.doi.org/10.4236/ojmsi.2014.22010

sign approach is presented, especially the Physical Network approach. According to this approach, each system is represented as consisting of functional elements that interact with each other by exchanging energy through their ports. Then, the powerful language for physical modeling simscape is presented. In the third part, we firstly give general information about the wind energy system and secondly, we model this system with MATLAB/ Simscape software, a powerful tool for physical modeling. Finally, the simulation results are presented.

2. Model Based Design-State of Art

In traditional design processes, design information is communicated and managed as text based documentation. Frequently this documentation is difficult to comprehend [1]. Model-based design can be used as a specification that contains greater detail than text-based specifications. In real-time applications, it enables developers to evaluate multiple options, predict systems performance, test systems functionality by imposing I/O conditions that might be operationally expected (before product deployment), and test designs. It allows building virtual prototypes or model to converge to an optimal solution much faster than traditional methods [1].

Modeling system can be done in different ways. Firstly, a mathematical model can be developed from natural laws that describe the evolution of the system. Then it is either linearized to form a transfer function or represented directly using block diagram and the signals are transmitted through the relationship between the individual blocks; the signals are used to transfer the values of individual variables from the output of an input block to other blocks. The mathematical model is dependent upon location in system making it difficult to reuse [2] [3]. It is the input-output or signal-based methods which were useful for control engineers because they were written in a language they could easily understand and integrated with their control system models [4]. The models built by this method typically execute very quickly because the modeler has specified the exact calculations that the solver must complete [5].

However, while this method works very well for control systems, it has some disadvantages when modeling physical systems. Physical systems are often expressed in the form of DAEs, which are composed of sets of equations that must be solved simultaneously. Another disadvantage is that the models that are created depend upon which elements they are connected to. It is necessary to know which inputs are available and which outputs must be calculated in order to connect it with the rest of the system. This type of model is difficult to reuse in other models or applications, for it requires that the other components in the system are modeled in the same manner. This becomes especially complex when modeling components that cross multiple physical domains, like a DC motor or a hydraulic cylinder [4].

Due to these reasons, engineers began looking for a better method for modeling these types of systems. For purely electrical systems, Kirchoff's laws have been used for quite a long time to express the equations for a entire system by applying a few basic mathematical rules to a network of electrical components represented by their individual mathematical models [5]. For example, the component model of a resistor was represented by v = iR, and this component model of an ideal resistor was identical for all resistors in the electrical network, independent of where the resistors were placed in the circuit. The equations for the entire system could be derived by applying Kirchoff's laws at the nodes of the electrical circuit. This method permits the component models to be modular and reusable while also being able to mathematically represent an entire circuit [4]. It was seen that other physical domains were analogous to electrical networks, and that similar rules could be applied to systems composed of one or more other physical domains.

Recently, the new approach, Physical Network provides for multidomain physical systems the same benefits that Kirchoff's laws provided for electrical systems. Using this method, an engineer could build up a library of component models that could be reused in a variety of models in a variety of applications which facilitate multidomains modeling.

Languages such as Simscape [6] and Modelica [7] are based on physical this approach. Modelica generates many redundant variables and equations, which are later removed by the solver. Contrariwise, Simscape language generates a fewer equations .Also, it has no need for co-simulation environment. That is why we focus in this thesis in Simscape language.

3. Strategy Based on Simscape Language

Simscape is a toolbox for physical modeling developed by the MathWorks for Simulink, and it has been available since version R2007A of the MATLAB suite [8]. It includes a foundation library, which contains basic components for electrical, hydraulic, mechanical and thermal systems, and an utilities library, which contains utility blocks, such as:

- Solver Configuration block, which contains parameters relevant to numerical algorithms for Simscape simulations. Each Simscape diagram (or each topologically distinct physical network in a diagram) must contain a Solver Configuration block.
- Simulink-PS Converter block and PS-Simulink Converter block, to connect Simscape and Simulink blocks. Use the Simulink-PS Converter block to connect Simulink outports to Physical Signal inports. Use the PS-Simulink Converter block to connect Physical Signal outports to Simulink inports [6].
- There are also more specialized toolboxes for physical modeling (such as SimDriveline, SimHydraulics, SimElectronics and SimMechanics) that now are considered as parts of the Simscape product family.
- In R2008b, a major upgrade of Simscape was made, introducing the Simscape language which allows the users to create their own physical models, and even new physical domains. The language is based on MATLAB syntax, but it is possible to define acausal equations [8]. This language offers three important advances:
- Non-causal modeling based on differential and algebraic equations instead of assignment statements as in traditional input/output block abstractions. Direct use of equations significantly increases re-usability of model components, since components adapt to the data flow context in which they are used. This generalization enables both simpler models and more efficient simulation.
- Multidomain modeling capability wich enables us to combine electrical, mechanical, thermodynamic, hydraulic within the same application model.
- Object-oriented modeling: oriented-object means that you can build a model similar to a real system by taking the components and connecting them into a model. These physical components can be connected through exactly defined interfaces (connectors) in which variables used for connection are defined, so that only connectors falling in the same connector classes can be connected. which are employed to support hierarchical structuring, reuse, and evolution of large and complex models covering multiple technology domains [9].

In Simscape each block in the system consists of functional elements that interact with each other by exchanging power or energy trough their ports. These ports, named physical conserving ports, are bidirectional and each different type of ports represents a physical domain. We can only connect to conserving ports of the same type. There are several types of conserving ports in SIMSCAPE, each with effort and flow associated variables.

The lines between these ports carry physical variables (trough and across) instead of signals. Through variables are measured with a gauge connected in series to an element. While Across variables are measured with a gauge connected in parallel to an element [10]. The product of these variables represents the power (energy flow) in watts, with the exception of air and magnetic field, where their product is energy [11].

According to Pecheux and Al (2005), these two variables obey the generalized Kirchhoff first law.

Another kind of ports in simscape is Physical signal ports. these ports are one-way directional and transfers signals between blocks with units associated with them. They permit to connect physical blocks developped in simscape to control model built using signal flows in Simulink or to other MathWorks application and this by means of converter blocks Simulink-PS Converter block and PS-Simulink Converter block [12].

4. Wind Energy System

4.1. Wind Energy Basics

Wind turbines are systems that harness the kinetic energy of the wind for useful power. Wind flows over the rotor of a wind turbine, causing it to rotate on a shaft. The resulting shaft power can be used for mechanical work, like pumping water, or to turn a generator to produce electrical power.

Wind turbines span a wide range of sizes, from small rooftop turbines generating less than 100 kilowatts up to large commercial wind turbines in the megawatt power range, many of which operate in large clusters called wind farms.

There are many different wind turbine designs, they are broadly grouped in two categories based on the orientation of the axis of rotation: Horizontal Axis Wind Turbines and Vertical Axis Wind Turbines. Most wind turbines are built today with the horizontal-axis design, which offers a cost-effective turbine construction, installation, and control by varying the blade pitch.

The vertical-axis machine has different advantages. First of all, it is omnidirectional and requires no yaw mechanism to continuously orient itself toward the wind direction. Secondly, its vertical drive shaft simplifies the installation of the gearbox and the electrical generator on the ground, making the structure much simpler. On the negative side, it normally requires guy wires attached to the top for support. This could limit its applications, particularly at offshore sites. Overall, the vertical-axis machine has not been widely used, primarily because its output power cannot be easily controlled in high winds simply by changing the blade pitch [13].

4.2. System Components

The wind power system comprises one or more wind turbine units operating electrically in parallel. Each turbine is made of the following basic components as shown in **Figure 1**:

- Tower structure
- Rotor with blades attached to the hub. The three-bladed concept is the most common concept for modern wind turbines. It is dynamically more stable and has better visual impact. It converts the energy in the wind to rotational shaft energy.

The mechanical power available at the output of the rotor is given by Equation (1).

$$Pm = 1/2 \cdot \rho \cdot \pi \cdot R^2 \cdot v^3 \cdot Cp(\lambda, \beta) \quad (1)$$

Cp is the power coefficient or performance coefficient which indicates the efficiency with which the turbine converts the mechanical energy of the wind into electricity. This coefficient differs according to the turbines.

In our case, the coefficient is given by the relation in the Equation (2). It is the most used formula.

$$Cp = 0.22(116/\lambda' - 0.4a - 5) * e(-12.5/\lambda') \quad (2)$$

With a is the angle of attack of wind turbine and λ' depends on λ and a as shown in Equation (3).

$$1/\lambda' = 1/(\lambda + 0.08a) - 0.035/(a^3 + 1)\rho \quad (3)$$

And λ is the specific speed witch is calculated by Equation (4).

$$\lambda = U/v = \omega \cdot R/v \quad (4)$$

with U is the tip speed of the blades, v is wind speed, R is the radius or length of the blades and ω is the rotational frequency of the rotor (in rad/s). β is the pitch angle of the blades.

- Shaft with mechanical gear: The power from the rotation of the wind turbine rotor is transferred to the generator through the power train. With a gearbox you convert between slowly rotating, high torque power which you get from the wind turbine rotor, and high speed, low torque power, which you use for the generator.
- Electrical generator: The generator is what converts the turning motion of a wind turbine's blades into electricity. Inside this component, coils of wire are rotated in a magnetic field to produce electricity. The two types of electric machines commonly used in the wind turbine industry are synchronous and asynchronous machines.

The mechanical equation that present the machine is given by Equation (5)

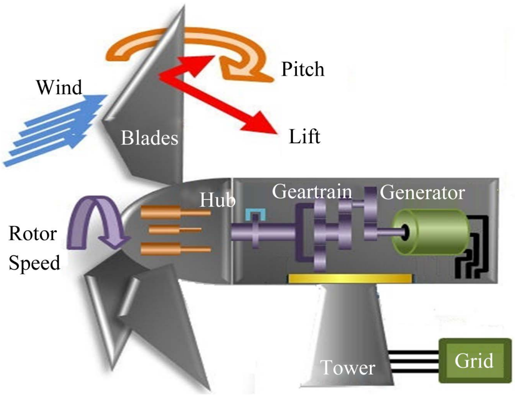

Figure 1. Wind turbine system [6].

$$J\frac{d\Omega m}{dt} = Cem - f\Omega m - Cr \qquad (5)$$

where Ω is the mechanical rotational speed, Cr is the load torque, J is the moment of inertia of the rotating part and f is the coefficient of viscous friction.

- Control mechanisms: The wind turbine technology has changed significantly in the last 25 years. Large wind turbines being installed today tend to be of variable-speed design, incorporating pitch control and/or power electronics(electronic converters).

Pitch angle control system consists of varying the pitch angle (from a certain value to 0˚) of the blades arround the hub using hydraulic cylinders around a longitudinal axis to stop the blades if the wind speed is below the inflow speed (generally 6 m/s) and if it is above the rated speed in order not to destroy the system [14]. The hydraulic system depicted in **Figure 2** performs positioning of the blades according to the wind speed. It consists of crank-rod system actuated by a hydraulic generator [13] [15].

The second solution which can be used to vary the pitch angle is composed of a DC motor which rotates the blade when it gets an electrical command. It is the solution treated in this paper.

The power electronics function is to catch the maximum of instantaneous wind.

5. Physical Modeling of Wind Turbine in Simscape

5.1. Interest for Multiphysics Modeling of Wind Energy System

The rapid expansion of wind power is partly the result of increased turbine size and the addition of new wind farms. It is also due to technological improvements in the wind turbines themselves. Better rotor-blade aerodynamics, more efficient generators, and improved supervisory control systems make today's wind turbines more efficient than ever before.

This enhanced technology presents engineers with new challenges. A wind turbine is a complex system in which a variety of subsystems must work together as efficiently as possible. The subsystems include mechanical devices such as rotor blades, gearboxes, hydraulic or electric drives for setting blade pitch angles, electrical yaw drives, generators, and the connection to the electrical grid. All of these are monitored by a complex supervisory control system that must respond in a specific way to varying environmental conditions, such as changing wind speeds.

In a traditional development process, the various subsystems are created in separate software and simulation environments. The result is that subsystems can only be tested together when hardware prototypes have been produced.

In physical modeling, we can assemble system-level models that span multiple physical domains and include the control system in a single environment and extend the analysis with 3-D visualization.

5.2. Wind Turbine Model

Figure 3 represents diffent blocks of physical systems that constitute the wind energy system.

The block 'wind couple' depicted in **Figure 4**, calculates the aerodynamic torque applied to the blades witch depends on the wind speed, the speed of the generator and the pitch angle. this torque turns the blades and

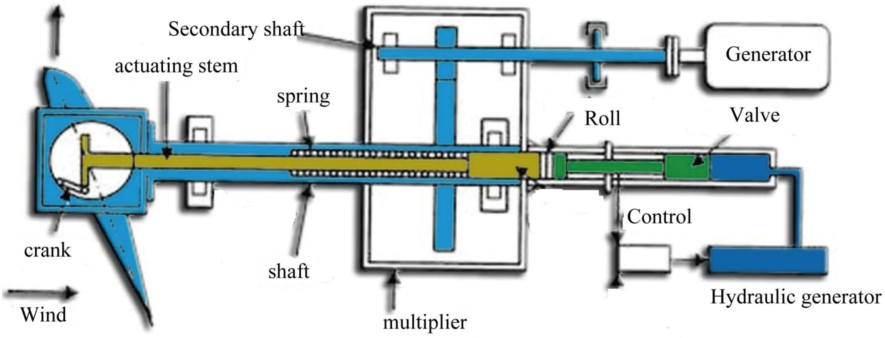

Figure 2. Hydraulic pitch actuator [14].

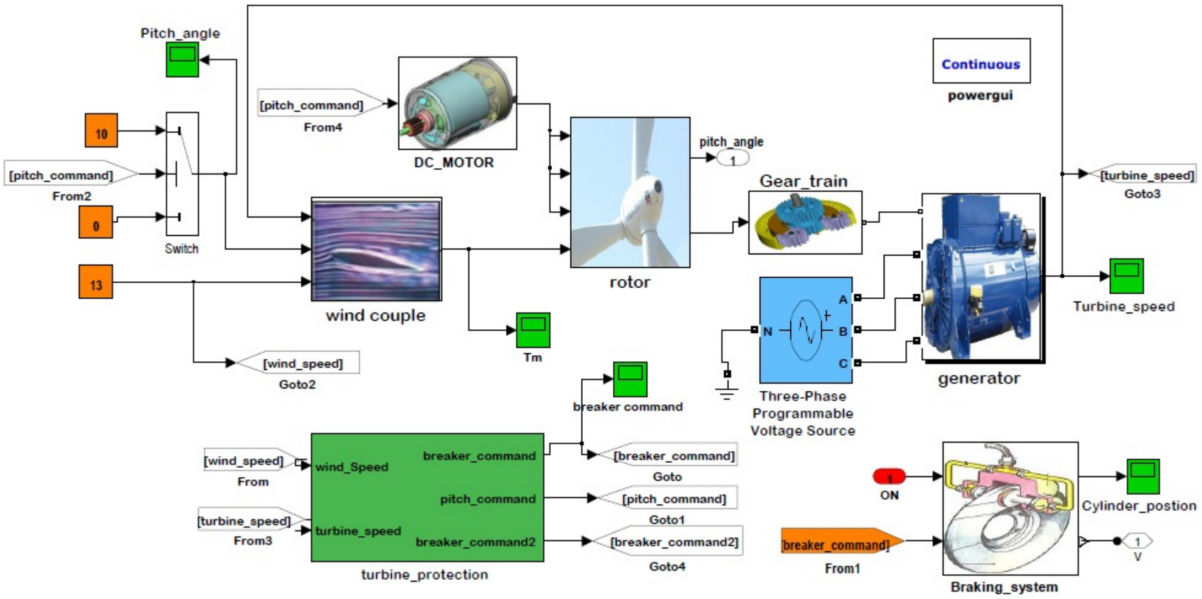

Figure 3. Model of wind turbine in SIMSCAPE.

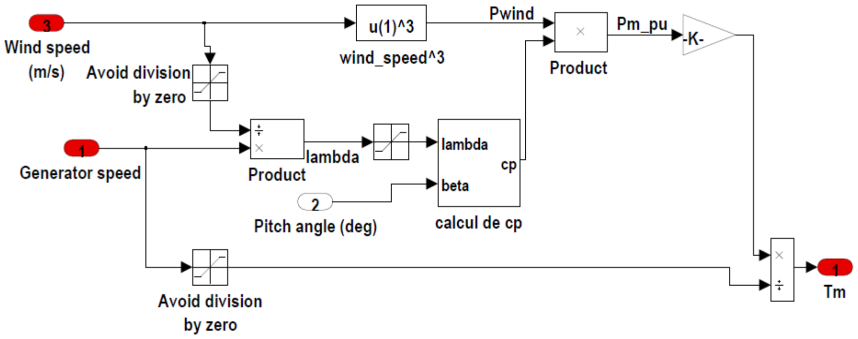

Figure 4. Model of wind couple subsystem.

therefore the hub which constitute the rotor. This motion can be visualized in 3-D animation provided by Sim-mechanics environment in which the block 'rotor' is modeled. The block's rotor model is depicted in Figure 5.

We can multiply the speed of rotation of the wind turbine using a gearbox as shown in Figure 6. The Sim-Driveline tool enable modeling driveline systems which consist of one or more inertias and masses, rotating about or translating along one or more axes, constrained to rotate or translate together by gears, which transfer torque and forces to different parts of the driveline. The torque at the output of the block gear-train is that applied to the generator's shaft.

DC motor block permits to vary the pitch angle by rotating the blade around its longitudinal axis. Each blade has its own pitch command actuating but for simplification, only one DC motor model is presented.

The braking system block models the hydromechanical command to brake the rotor in case of a high wind speed.

5.2.1. Model of Wind Couple Subsystem

The calculation of the wind couple is based on Equation (1) which is linearized to form a transfer function.

5.2.2. Model of Rotor Subsystem

The blades and the rotor are represented by bodies connected by joints representing the possible motions of bodies relative to one another.

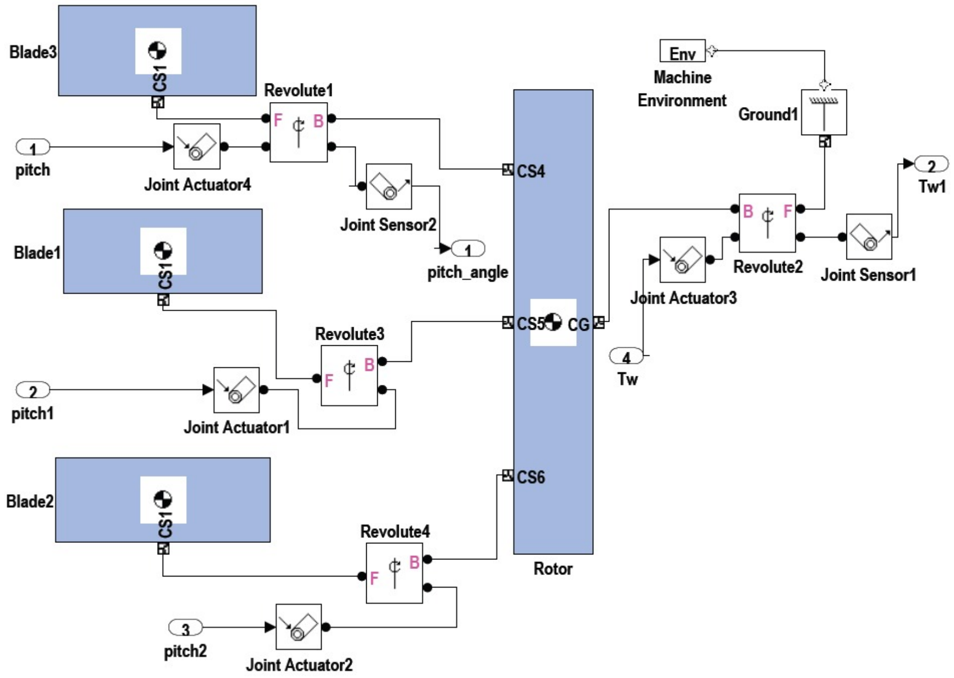

Figure 5. Model of rotor subsystem in simmechanics.

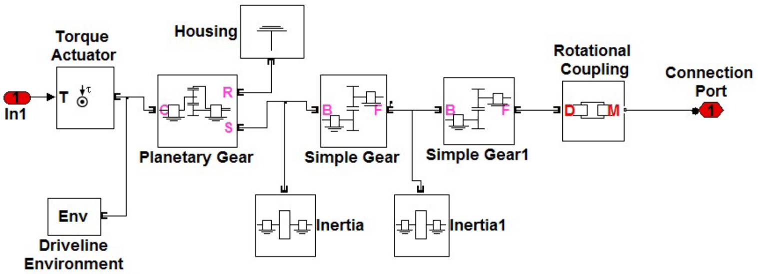

Figure 6. Model of gear train.

The rotor and blades can rotate relative to ground. Also, the blades can make a rotational movement relative to the hub to vary thereby the pitch angle. The bodies are modeled with Body blocks specified by their masses, inertia tensors, and attached Body coordinate systems (CSs).

Instead of typing these characteristics, we can use the SimMechanics importer to automatically generate a SimMechanics model based on an external data file previously exported from a supported CAD platform.

5.2.3. Model of Gear Train

In the model depicted in **Figure 6**, simple and planetary gears are used to transfer torque up and down the driveline axes. The Inertia block represents a rotating body specified by its moment of inertia. The choice of planetary gear and simple gear is made to have a greater gear ratio. Otherwise, we can settle for a simple gear.

5.2.4. Model of Asynchronous Machine

The model depicted in **Figure 7** consists of an asynchronous machine block ready to use and the loop speed command of the generator based on Equation (6):

$$J\frac{\mathrm{d}\Omega m}{\mathrm{d}t} = \Gamma\mathbf{e} - \mathbf{f}\boldsymbol{\Omega}\mathbf{m} - \Gamma\mathbf{r} \tag{6}$$

Ideal Torque Source: the representation of the force acting on the machine.

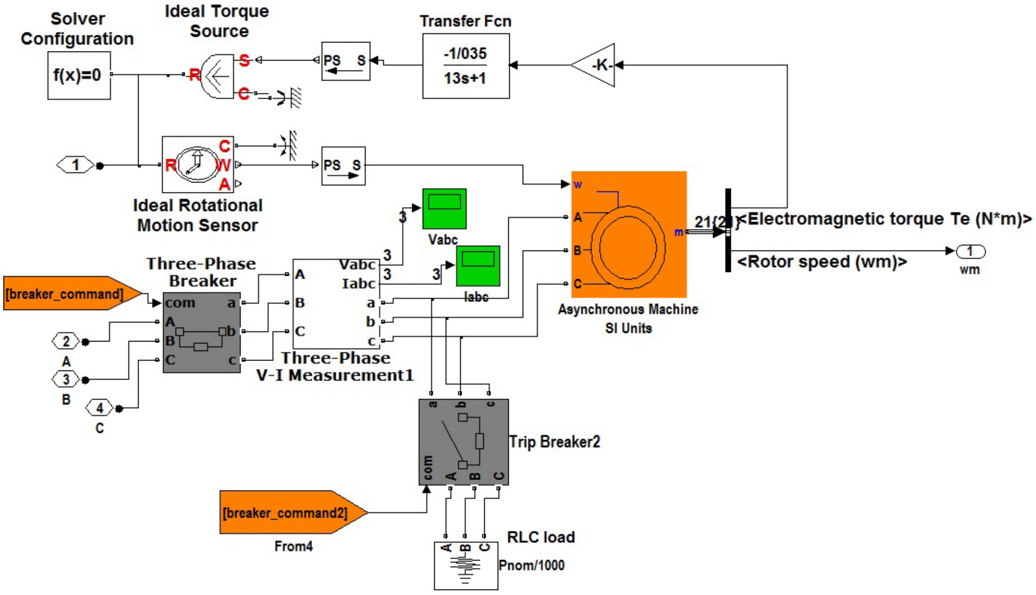

Figure 7. Model of generator subsystem.

PS-Simulink Converter: converts a physical signal into a Simulink output signal.

Solver Configuration: specifies global environment information for simulation and provides parameters for the solver that model needs before we can begin simulation.

Mechanical References: blocks represent a reference point, or frame, for all mechanical translational and rotational ports.

5.2.5. Model of Turbine Protection

Protection system modeled in **Figure 8**, permits immobilizing the wind turbine when the wind reaches a certain strength or when the wind speed is below a certain value by changing the pitch angle of the blades for zero engine torque (beta = 0) (shown in **Figure 9**). Also, in case of over speed's generator, it allows triggering the breaker in order to disconnect the wind turbine from grid (shown in **Figure 10**) and varying the pitch angle to 0 by means of the DC motor block depicted in **Figure 11**.

If the wind speed is less than 4 m/s or above 25 m/s for a time of 0.05 seconds, the pitch angle value switch from 10 to 0.

We switch the pitch angle to 0 and trigger the circuit breaker if the generator's speed exceeds the rated speed by 20% for a period of 0.05 seconds.

5.2.6. Model of Pitch Angle Command Block

In the block depicted in **Figure 11**, the pitch command output of the blade protection block switches on or off the DC motor in order to vary the pitch angle by rotating the blade. In our case, to pass from beta = 10 to beta = 0, we need 0.945 s and then we stop again the motor. 0.945 s is needed because the rotation speed is equal to 30 tr/min.

1400 N.m/s is the value of the load torque represented by the weight of the blade. For our wind turbine, the weight of the nacelle and the rotor is equal to 123 tons.

5.2.7. Model of Braking System Block

The block depicted in **Figure 12**, models the hydromechanical command applied to the disc brake.

When we want to stop the wind turbine if the wind speed reaches maximum speed, the hydraulic command provides some pressure sliding thereby a cylinder which in turn pushes the plates of the disc. The other end of the disc brake is mounted on the rotor shaft.

For restarting, the input ON commands the flow of the fluid in the opposite direction. So, the double acting cylinder returns to its initial position.

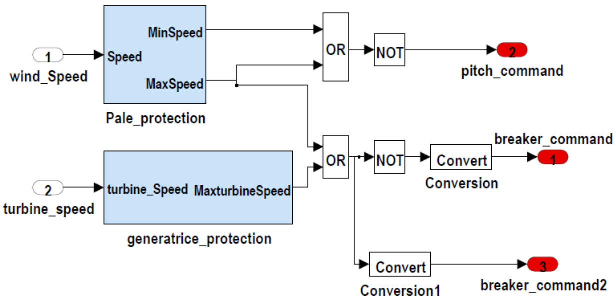

Figure 8. Model of wind protection.

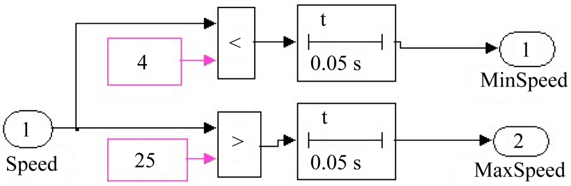

Figure 9. Model of blade protection.

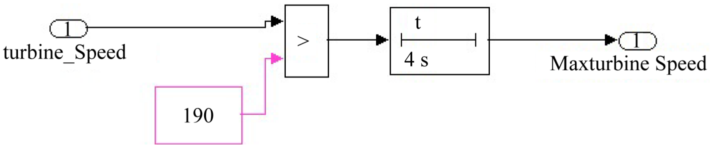

Figure 10. Model of generator protection.

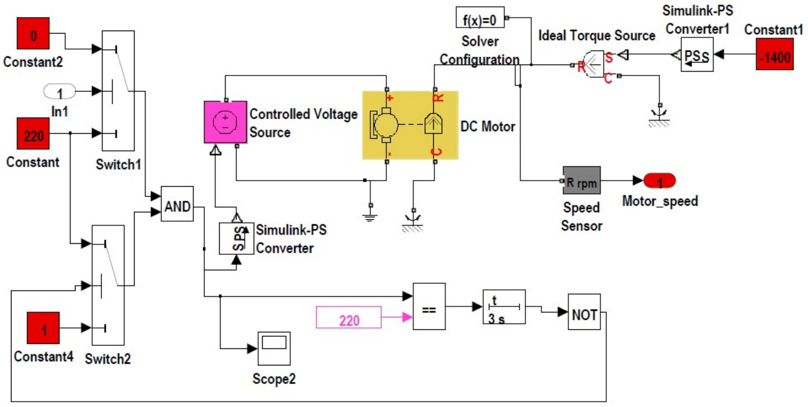

Figure 11. Pitch angle command block.

The value of the applied pressure is calculated based on the following dynamic Equation (7):

$$\omega(t) = -P*\pi*S/J*t+\omega 0 \qquad (7)$$

$\omega 0$ is the rotational speed of rotor and S is the surface of brake pad which is equal to 1.8 m in our case.

The duration of braking is equal to 3 s, then $w\ (t = 3) = 0$.

J is the moment of inertia of the rotor. It is calculated by equating the rotor to a hollow cylinder and considering the weight equal to 56 tons.

6. Simulation Results and Discussion

6.1. Mechanical Torque Applied to Blades

For a wind speed equal to 13 m/s, the mechanical torque applied to blades has the value 15.5 N.rad/s as shown

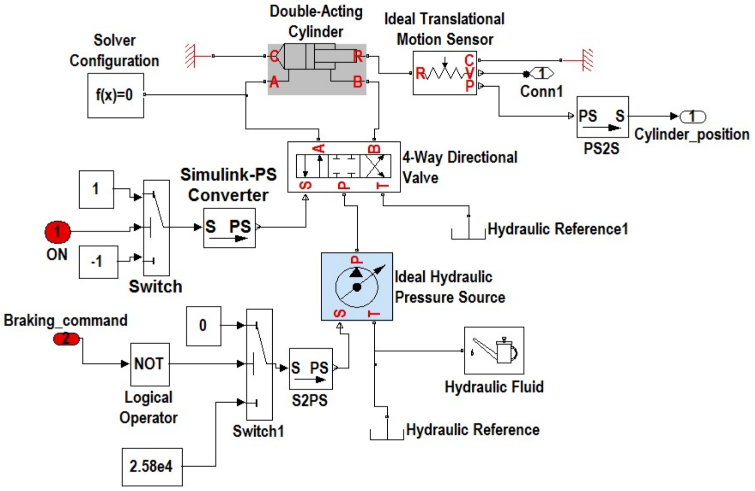

Figure 12. Braking system block.

in **Figure 13**.

6.2. Generator Speed

As shown in **Figure 14**, the speed of the generator is equal to the rated speed; it is obtained at the wind speed equal to 13 m/s.

6.3. Active and Reactive Power of the Wind Turbine

As depicted in **Figure 15**, the active power knows fluctuations in the beginning before stabilizing at the value 400 kW. These fluctuations are due to the transient period of the generator.

6.4. Results of Simulation in Case of a Wind Speed of 27 m/s

To simulate the role of protection, we took a wind speed equal to 27 m/s. We observe that after 0.05 seconds, the voltages and currents and torque are set to 0 (as shown in **Figure 16** and **Figure 17**) because the pitch angle and breaker command pass to 0 as shown in **Figure 18** and **Figure 19**.

Then, the active and reactive powers passes to 0 at the time of switching of the generator breaker (**Figure 20**).

6.5. Rotor Visualization

SIMMECHANICS allows visualizing the motion of system created in Simmechanics or imported from a CAO software.

You can use this environment to validate the 3-D configuration of your model and visually analyze the behavior of your system under different conditions [15].

Figure 21 visualizes the motion of rotor in Simmechanics.

7. Conclusions

In our work, we had presented the state of the art of the physical network approach, a more developed method for multiphysics and multidomain modelling than the classical method of mathematic-physical analysis. This method enables modelling realistic models and facilitates the reuse of these models.

Also, we were able to model the horizontal wind turbine and simulate this model in MATLAB/SIMSCAPE, a powerful tool for modelling physical systems.

The model developed shows how we can build component models and assemble these models into complete systems which facilitate the reuse of models in the new context.

Also, the results prove that it is possible to build and simulate physical models with simscape software.

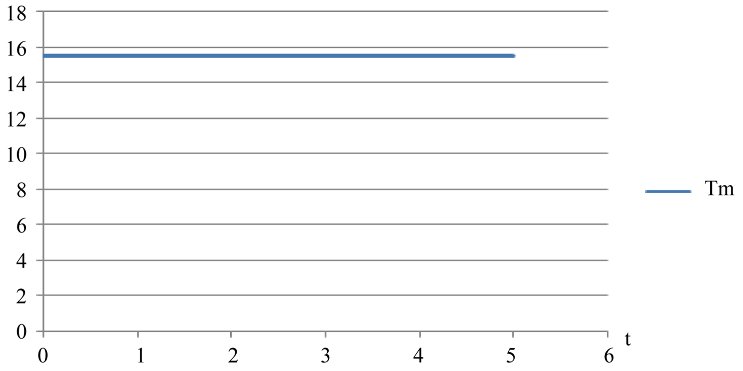

Figure 13. Mechanical torque Tm.

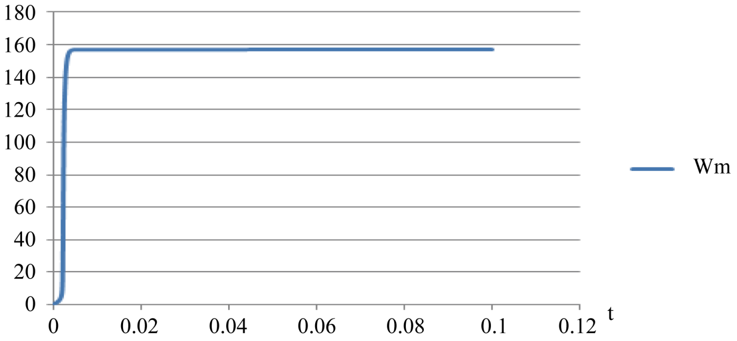

Figure 14. Generator speed.

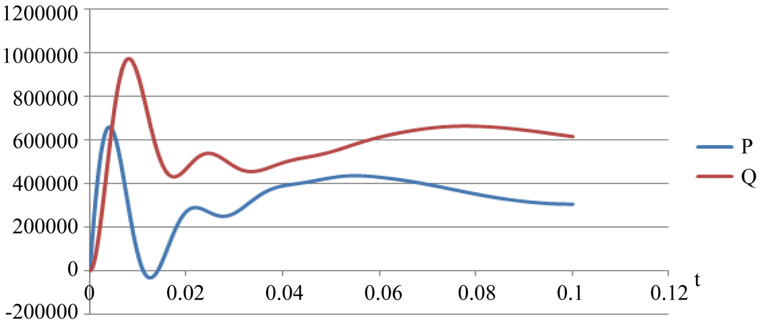

Figure 15. Active and reactive power of the wind turbine.

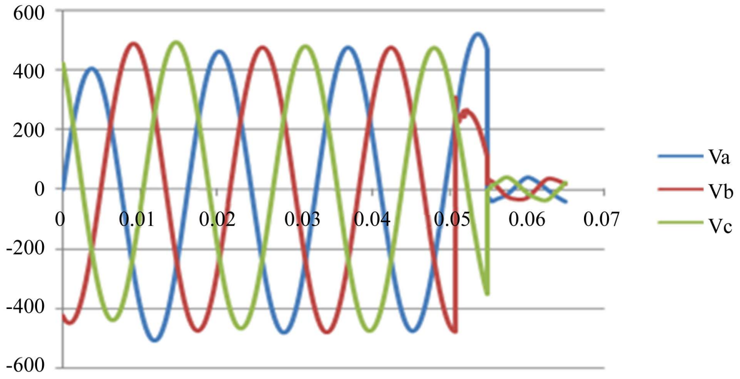

Figure 16. Voltages of the wind turbine for a wind speed equal to 27 m/s.

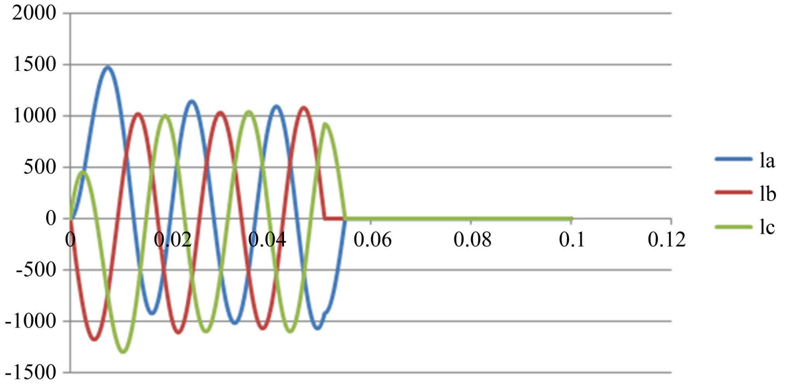

Figure 17. Currents of the wind turbine for a wind speed equal to 27 m/s.

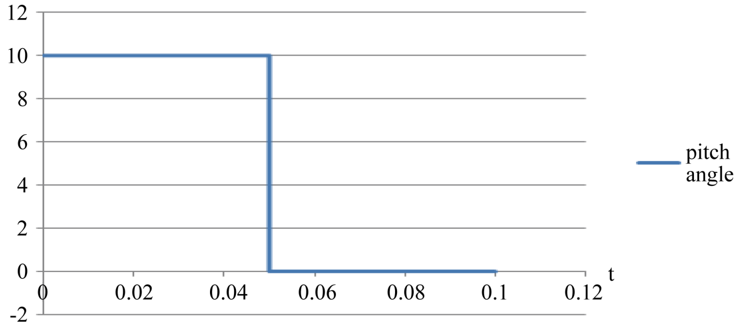

Figure 18. Variation of the pitch angle for a wind speed equal to 27 m/s.

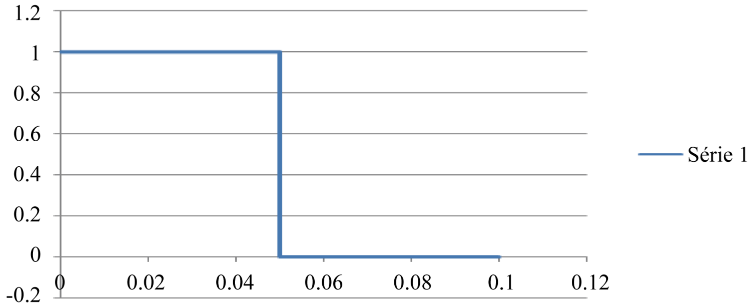

Figure 19. Variation of breaker command for a wind speed equal to 27 m/s.

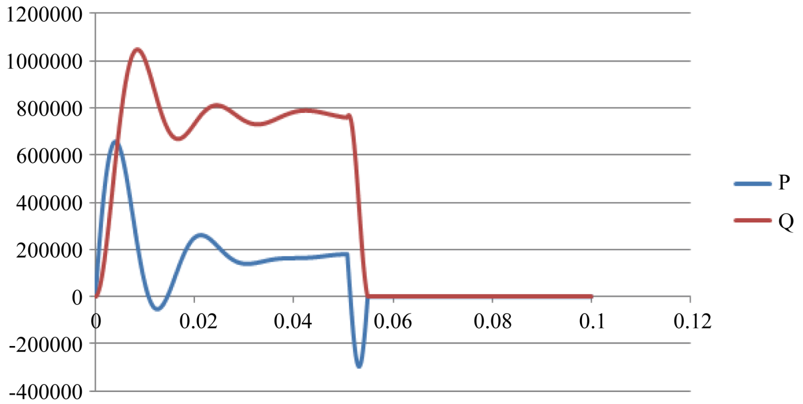

Figure 20. Active and reactive power for a wind speed equal to 27 m/s.

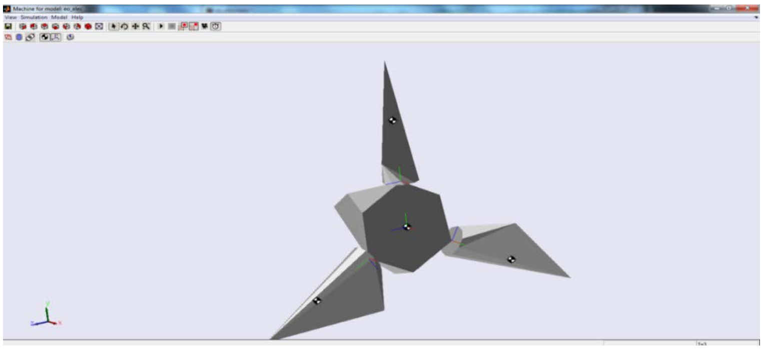

Figure 21. Rotor visualization.

My suggestions for further works would be the following:

1. Enrich the Simscape's library with more blocks which cover more physical domains.
2. Build a more realistic and complex wind turbine model.
3. Develop a model of generator based on physical network approach rather than physical signal modeling.

References

[1] Krasner, J. (2004) Model-Based Design and Beyond: Solutions for Today's Embedded Systems Requirements.

[2] Broman, D. (2010) Meta-Languages and Semantics for Equation-Based Modeling and Simulation.

[3] Kofráne, J., Mateják, M., Privitzer, P. and Tribula, M. (2004) Causal or A Causal Modeling: Labour Ytytt for Humans or Labour for Machines. Technical Computing Prague, Conference Proceedings.

[4] Popinchalk, S., Glass, J., Shenoy, R. and Aberg, R. (2007) Working in Teams: Modeling and Control Design within a Single Software Environment. *AIAA Modeling and Simulation Technologies Conference and Exhibit* [*CD-ROM*], Hilton Head, 20-23 August 2007.

[5] Nage, L.W., Pederson, D.O. and SPICE (1973) Simulation Program with Integrated Circuit Emphasis. Memorandum No. ERL-M382, University of California, Berkeley.

[6] Mathworks Home Page. www.mathworks.com

[7] Elmqvist, H., Mattsson, S.E. and Otter, M. (1998) Modelica: The New Object-Oriented Modeling Language. *The* 12*th European Simulation Multiconference*, Manchester, 16-19 June 1998, 127-131.

[8] Sjostedt, C.J. (2009) Modeling and Simulation of Physical Systems in a Mechatronic Context. Doctoral Thesis Stockholm.

[9] Fernandez, J. (2013) Object-Oriented Modeling and Simulation of the Closed Loop Cardiovascular System by Using SIMSCAPE. *Computers in Biology and Medicine*, **43**, 323-333. http://dx.doi.org/10.1016/j.compbiomed.2013.01.007

[10] Jirstrand, M. and Gunnarsson, J. (1999) Math Modelica—A New Modeling and Simulation Environment for Mathematica. 3*rd International Mathematica Symposium*, Linz, 23-25 August 1999.

[11] Dragan, P., Novak, N. and Ljubiša, D. (2012) Bond Graph Modeling in Simscape. *International Tyhu Journal of Mathematics and Computers in Simulation*, **6**, 2012.

[12] Enocksson, S. (2011) Modeling in Math Works Simscape by Building a Model of an Automatic Gearbox.

[13] Dubois, C. (2009) Le guide de L'éolien, techniques et pratiques.

[14] Heliciel Home Page. http://www.heliciel.com/helice/helice-pas-variable.htm

[15] Hamdi, N. (2008) Modélisation et commande des génératrices éoliennes.