Springer

KSME

Journal of Mechanical Science and Technology 27 (6) (2013) 1843~1852
www.springerlink.com/content/1738-494x
DOI 10.1007/s12206-013-0435-x

# Bond graph modeling and simulation of wind turbine systems †

Tore Bakka* and Hamid Reza Karimi

*University of Agder, Department of Engineering, Faculty of Engineering and Science, 4876 Grimstad, Norway*

(Manuscript Received August 9, 2012; Revised December 19, 2012; Accepted January 17, 2013)

**Abstract**

This paper addresses the problem of bond graph methodology as a graphical approach for the modeling of wind turbine generating systems. The purpose of this paper is to show some of the benefits the bond graph approach has, in contributing a model for wind turbine systems. We will present a nonlinear model of a wind turbine generating system, containing blade pitch, drive train, tower motion and generator. All which will be modeled by means of bond graph. We will especially focus on the drive train, and show the difference between modeling with a classical mechanical method and by using bond graph. The model consists of realistic parameters, but we are not trying to validate a specific wind turbine generating system. Simulations are carried out in the bond graph simulation software 20-sim [1].

*Keywords*: Bond graph; Modeling; Wind turbine

## 1. Introduction

The demand for energy world wide is increasing every day. And in these "green times" renewable energy is a hot topic all over the world. Wind energy is currently the most popular energy sector. The growth in wind power industry has been tremendous over the last decade. As of June 2012 the global wind capacity is 254000 MW, according to the World Wind Energy Association [2].

Whenever we are talking about models of wind turbine systems, the turbine model becomes a critical part of the discussion. Over the years it has been some discussion about how to model the wind turbine accurately. In Refs. [3, 4] they perform dynamic analysis on a one-mass-model, in Refs. [5, 6] they examine a two-mass-model. In Ref. [7] they use actual measured data from a wind turbine and compare it with both a one-mass and a two-mass-model. They validate the model using a recorded case obtained in a fixed speed, stall regulated wind turbine. In Ref. [8] a six-, three- and a two-mass model are compared with each other. They argue that a six-mass model is needed for the precise transient analysis of the wind turbine system, and they develop a way to transform a six-mass model into a two-mass model. The goal of that paper is not to use the model in the control scheme, but in the use of transient stability analysis of grid connected system.

The pitching of the blades are usually executed by means of a hydraulic system, but for system modeling purposes it is often considered as a first or second order system. We are here dealing with variable speed generating system, therefore a wound machine or a double fed induction generator is needed. These can be modeled in different ways, ranging from complex electric equivalent circuits to a first order system.

Several advanced wind turbine simulation softwares have emerged during the last decade. HAWC2 [9], Cp-Lambda [10] and FAST [11] are a few examples. They are developed at RISØ in Denmark, POLI-Wind in Italy and NREL in the US, respectively. In these codes the turbine and structure is considered as complex flexible mechanisms, and uses the finite-element-method (FEM) multibody approach. An aero-servo-elastic model is introduced, which consists of aerodynamic forces from the wind, the servo dynamics from the different actuators and the elasticity in the different joints and the structure. Both FAST and HAWC2 can simulate offshore and onshore cases while Cp-Lambda is limited to the onshore case.

As seen above there are many ways to model a wind turbine generating system, some are simple and some are very complex. In a simulation point of view it is desirable that the model is as simple as possible and can capture as much of the dynamics as appear in reality. This is an absolute demand, another important issue is to keep the central processing unit (CPU) labor to a minimum. For example if we are dealing with hardware in the loop (HIL) simulation, then it is necessary to download the model to a programmable logic controller (PLC). This argues in favor of the importance in having a fast C-code. Things that can potentially have a negative effect on the execution of our C-code are for example; algebraic loops and differential causality on the

*Corresponding author. Tel.: +47 37 23 37 83, Fax.: +82 37 23 30 01
E-mail address: tore.bakka@uia.no
† Recommended by Associate Editor Yang Shi

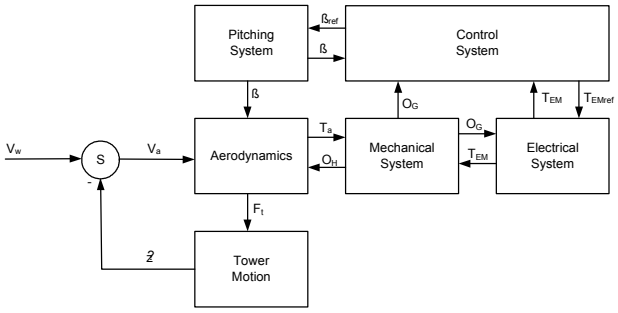



Fig. 1. Setup for a wind turbine system.

different elements in the system. These topics bring us to the use of the bond graph methodology. This is a unified approach to model all types of physical systems, producing both linear and nonlinear mathematical models. Engineers must work and interact in many different disciplines. An understanding of the intersections of these different disciplines is a valuable asset for any engineer. Using the language of bond graphs, one may construct models of electrical-, magnetic-, mechanical-, hydraulic-, pneumatic- as well as thermal systems. It is a systematic way to model these dynamic systems, and there are standard ways to translate them into differential equations or computer simulation schemes. After constructing the bond graph one can easily spot algebraic loops and whether you have integral causality on the dynamic elements by inspecting the bond graph. There are various ways to spot these things in typical simulation software such as MatLab [12], but it is beneficial to spot them before the implementation. It is a quite intuitive way in setting up the bonds and connecting the elements, this will be discussed in a later section. The outcome from the bond graph model is a set of first order differential equations, which afterwards can be used for systems response analysis or for example controller design. After constructing the bond graph one gets a better understanding of what actually happens in the system. In an educational point of view one can easily understand which element decides what in the system. For example in a simple mass-spring-damper system, one can easily see which component decides the speed and which component decides the force. With these arguments in mind we are motivated to explore the possibilities there are with the use of bond graph.

The wind turbine generating system can be divided into several subsystems, see Fig. 1.

The system setup is adopted from Ref. [13], where $V_w$ is the wind speed, $V_a$ is the wind speed for power production, $\dot{Z}$ is the tower speed, $F_t$ is the thrust force acting on the tower, $\beta_{ref}$ is the pitch angle reference, $\beta$ is the actual pitch angle, $T_a$ is the aerodynamic torque, $\Omega_H$ is the hub speed, $\Omega_G$ is the generator speed, $T_{EMref}$ is the generator torque reference and $T_{EM}$ is the actual generator torque.

The expression for power produced by the wind is given by Ref. [14]

$$P_a = \frac{1}{2}\rho\pi R^2 v^3 C_p(\lambda,\beta). \tag{1}$$

The dimensionless tip-speed ratio (TSR) $\lambda$ is defined as

$$\lambda = \frac{v_b}{v} \tag{2}$$

where $v_b$ is the tip speed of the blade and $v$ is the wind speed. From Eq. (1) we can find the aerodynamic torque and the thrust force acting on the tower

$$T_a = \frac{1}{2}\rho\pi R^3 v^2 \frac{C_p(\lambda,\beta)}{\lambda} \tag{3}$$

$$F_t = \frac{1}{2}\rho\pi R^2 v^2 C_T(\lambda,\beta) \tag{4}$$

where $P_a$ is the aerodynamic power, $\rho$ is the air density and $R$ is the blade radius. $C_p$ gives the relationship between how much power is available in the wind and how much can be converted to electrical power. Not all the available power can be converted, this is due to the fact that the wind cannot be completely drained of energy, otherwise the wind speed at the rotor front would reduce to zero and the rotation of the rotor would stop. It can be proven that the theoretical upper limit of $C_p$ is $16/27 \approx 0.59$, this is known as the Betz limit. A general modern wind turbine has a maximum power coefficient of about 0.5. $C_T$ is the thrust force coefficient, both these coefficients are dependent on the TSR $\lambda$ and the pitch angle $\beta$.

This paper is organized as follows. Section II gives a short overview on the bond graph methodology and its different elements. Section III describes the different parts of our system model; aerodynamics, pitch, drive train, tower motion and generator. In section IV the simulation results are presented and section V gives the conclusion and states some suggestions regarding future work.

## 2. Introduction to bond graph

Bond graph is a graphical way of modeling physical systems. All these physical systems have in common the conservation laws for mass and energy. Bond graph, originated by Paynter [15] in 1961, deals with the conservation of energy. This gives a unified approach to model physical systems. This section gives a short introduction to this modeling tool, the interested reader can find more information in Refs. [16, 17]. The bond graph approach has several advantages over conventional methods, i.e.: 1) providing a visual representation of the design; 2) controlling the consistency of the topological settings of the design; 3) providing the hierarchical modeling of designs; 4) extracting the system equations symbolically in a structured way.

Within physical systems, energy is transported form one item to another. This energy is either stored or converted to

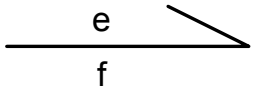



Fig. 2. Power bond with effort and flow.

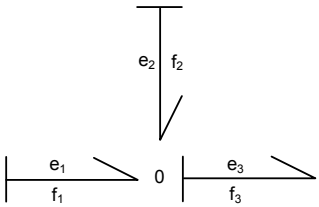



Fig. 3. 0-junction.

other forms. But the important thing is that it does not dissipate. If the energy is changing in one place, it also changes in an opposite way at another location. The definition of power is the change in energy ( $E$ ) with respect to time:

$$P = \frac{d}{dt}(E). \tag{5}$$

The power is transferred between the different parts in the bond graph model with the use of power bonds, see Fig. 2. In bond graph notation the definition of power is effort multiplied with flow. For example, in electric systems this would mean voltage multiplied with current, in mechanical systems it is force multiplied with velocity and in hydraulics it is pressure multiplied with flow.

### *2.1 System elements*

In bond graph modeling there are a total amount of nine different elements. We will here introduce the causality assignments, but first we have to explore the cause and effect for each of the basic bond graph elements. Only elements with its preferred causality will be discussed. The importance of causality will be dealt with later in the paper.

*1. Junctions:* There are two different types of junctions that connects the different parts in a bond graph model, the 0-junction and the 1-junction. The 0-junction is an effort equalizing connection, see Fig. 3 and its corresponding equation in Eq. (6). Since the efforts are the same, only one bond can decide what it is. The 1-junction is a flow equalizing connection, see Fig. 4 and its corresponding equation in Eq. (7). Since the flows are the same, only one bond can decide what it is. Which bond decides the flow and which one decides the effort is indicated with the vertical causality stroke. If the vertical line is closest to the junction, then this element decides the effort, furthest away from the junction decides the flow.

*2. Source element:* We can divide the source elements into two different kinds, effort- and flow-source. The effort source

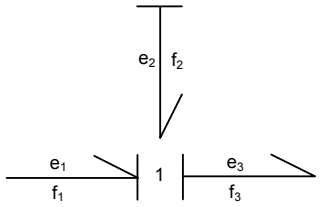



Fig. 4. 1-junction.

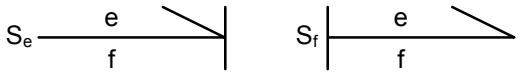



Fig. 5. Effort and flow source with their causality assignment.

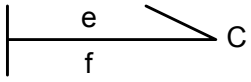



Fig. 6. Example of compliance element with integral causality.

gives an effort into the system, then it is up to the system to decide the flow. This is what is meant with cause and effect, and its vice versa for the flow source. Fig. 5 shows how the causality is indicated on the graphical elements. For the source elements these causality assignments are fixed.

$$e_1 = e_2 = e_3 \tag{6}$$

$$f_3 = f_1 + f_2 \tag{7}$$

$$f_1 = f_2 = f_3$$

$$e_3 = e_1 + e_2.$$

*3. Compliance element:* The causality assignment for the C-element has two possibilities, but one is preferred in contrast to the other. This is discussed at the end of this section. The preferred case is seen in Fig. 6 and its corresponding equation in Eq. (8). We see from both the equation and the figure that flow is given to the element/equation and it gives the effort in return.

$$\mathrm{e} = \frac{1}{\mathrm{C}}\int \mathrm{fdt} = \frac{q}{C}. \tag{8}$$

The variable $q$ is called *the generalized displacement*. For example, this can be rotational position of the rotor in a wind turbine.

*4. Inertia element:* There are two choices for the causality assignment for the I-element, also here one is preferred in contrast to the other. The preferred case is seen in Fig. 7 and its corresponding equation in Eq. (9).

$$\mathrm{f} = \frac{1}{\mathrm{I}}\int \mathrm{edt} = \frac{p}{I} \tag{9}$$

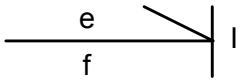



Fig. 7. Example of an inertia element.

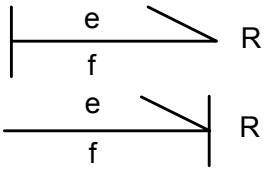



Fig. 8. Example of resistiveelements.

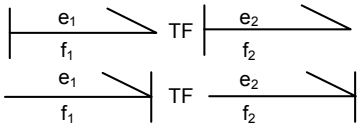



Fig. 9. Example of the two transformers.

The variable $p$ is called *the generalized momentum*. For example, this can be rotor inertia times rotor velocity in a wind turbine.

*5. Resistive element:* It is a bit more freedom when it comes to the causality assignment for the R-element. Its equation do not include any dynamics, it is only an algebraic expression. The two causality choices are shown in Fig. 8 and its corresponding equation in Eq. (10).

$$\begin{aligned} e &= Rf \\ f &= \frac{e}{R}. \end{aligned} \tag{10}$$

*6. Transformer:* The transformer element can work in two ways; either it transforms a flow into another flow or it transforms an effort into another effort. Fig. 9 corresponds to Eqs. (11) and (12), where $m$ is the transformation ratio.

$$\begin{aligned} e_1 &= me_2 \\ f_2 &= mf_1 \end{aligned} \tag{11}$$

$$\begin{aligned} e_2 &= \frac{1}{m}e_1 \\ f_1 &= \frac{1}{m}f_2. \end{aligned} \tag{12}$$

For example, this can represent a mechanical gearing or an electric transformer.

*7. Gyrator:* The gyrator can also work in two ways; either it transforms a flow into an effort or it transforms an effort into a flow. Fig. 10 corresponds to Eqs. (13) and (14), where $r$ is the gyrator ratio.

$$\begin{aligned} e_1 &= rf_2 \\ e_2 &= rf_1 \end{aligned} \tag{13}$$

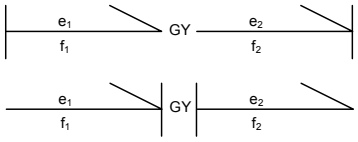



Fig. 10. Example of the two gyrators.

$$\begin{aligned} f_1 &= \frac{1}{r}e_2 \\ f_2 &= \frac{1}{r}e_1. \end{aligned} \tag{14}$$

This can for example be an electric motor, where you have voltage as input and a rotational speed as output.

The importance of integral causality is nicely explained in Ref. [18]. First imagine a step in effort is imposed on a C-element, then the causality assignment will be opposite of what is shown in Fig. 6. This means the flow output is proportional to the derivative of the input effort. From calculus we know that the derivative of the step function at the beginning is infinite, i.e. this do not give any physical meaning. We can imagine a simple electric circuit containing a voltage source coupled with a capacitor, if a step input were to be imposed on the voltage source, the capacitor would experience a very high current and it would blow up. From this we can conclude that nature integrates and only mathematicians differentiate!

On the other hand, the ability to spot algebraic loops is one of the benefits with the use of bond graph as a modeling tool. These loops can be spotted simply by inspection of the bond graph representation, if the causality assignment on the R-elements are different from each other, then we have algebraic loops in the system. If they have the same causality, there are no algebraic loops. These loops occur for example if you have two resistors in series. In this circuit both resistors will try to decide what the current should be, i.e. they depend on each other. This will not necessarily cause problems to the simulation, but it might. Especially if the resistors are nonlinear, then the simulation could easily crash. The simulation program will also be forced to spend time to solve this algebraic loop. If we can easily spot these loops early in the modeling process, then we can try to fix them by simply adding an element. For example, regarding our circuit with two resistors in series, we can add an inductive element to the circuit. Then it would be the inductive element which decides what the current should be and not the resistive elements. The resistive elements would simply have to take what current the inductive element lets through. We can give the inductive element a value such that the voltage drop over the element is very low, i.e. it does not play any major role in the circuit. Now when our model has no algebraic loops and all the dynamic elements have integral causality, the simulation should go smooth. If we have a large set of equations or a large block diagram it is not easy to spot these things right away, but with a bond graph representation of the model we can spot them simply by inspection. To simply remember the

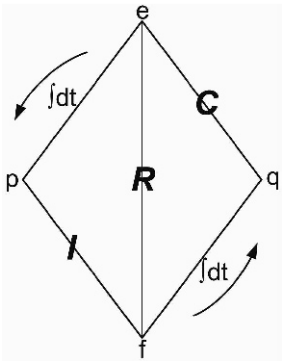



Fig. 11. The tetrahedron of state.

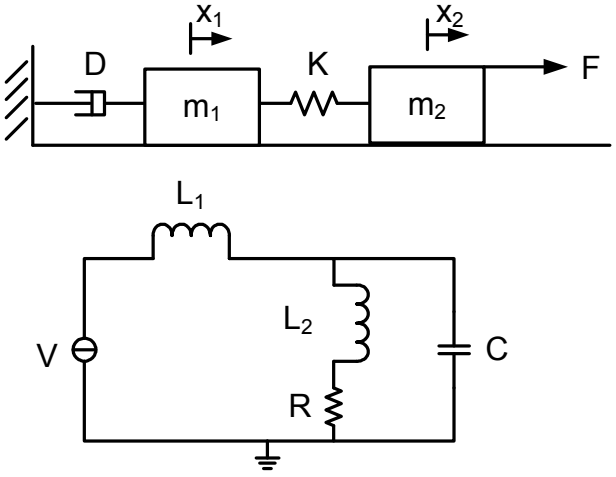



Fig. 12. Bond graph of the two equivalent circuits.

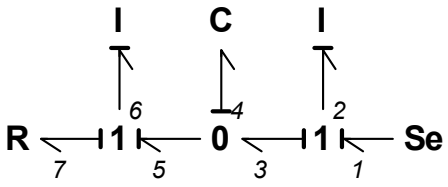



Fig. 13. Two equivalent circuits.

aforementioned relations we can use what Paynter called *the tetrahedron of state*, shown in Fig. 11. The procedure of how to extract the algebraic and dynamic equations from a bond graph model is not included in this short overview, but it can be done in a very systematic way and it will partly be shown in the next section.

We will end this section with a small example. The purpose is to show how to set up a bond graph of a simple system, and also show the difference in relation to block diagrams. Fig. 12 shows two equivalent circuits in two different domains, and they have exactly the same governing equations. The corresponding bond graph is shown in Fig. 13. The easiest way to set up a bond graph when having a mechanical system, is to start with setting up 1-junctions. One junction for each mass, this gives two 1-junctions in our example. We add a 0-junction in between, because we know the speed is different but the force is the same. Force is transferred through the C-element (spring). The right side of the damper has the same speed as m_{1}, R-element and I-element is therefore connected to the left 1-junction.

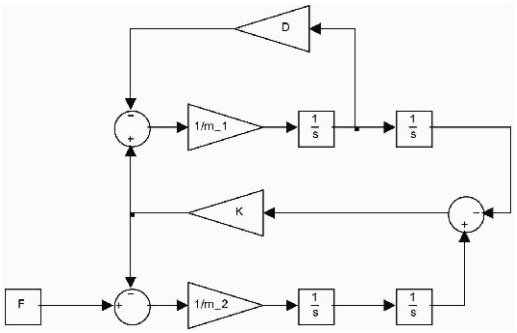



Fig. 14. Block diagram of mechanical example.

Regarding the electric circuit, we know that the source and $L_1$ have the same current $i_1$. We know that $L_2$ and $R$ have the same current $i_2$, and we know that the parallel branches have the same voltage. In this way we end up with the exact same bond graph. We also note that the bond graph has integral causality. The two I-elements receive effort and give flow in return, the C-element receives flow and gives effort in return. We will now find the governing equations. First we find $\dot{p}_i$, second we find $\dot{q}_i$. In mechanical terms this is $m\ddot{x}_i$ and $\dot{x}_i$, respectively. Subscript $i$ corresponds to in which bond we are at.

$$\dot{p}_2 = e_2 = e_1 - e_3 = S_e - e_4 = S_e - \frac{q_4}{C_4}, \tag{15a}$$

$$\dot{p}_6 = e_6 = e_5 - e_7 = e_4 - R_7 f_7 = \frac{q_4}{C_4} - R_7 f_6 = \frac{q_4}{C_4} - R_7 \frac{p_6}{I_6}, \tag{15b}$$

$$\dot{q}_4 = f_4 = f_3 - f_5 = f_2 - f_6 = \frac{p_2}{I_2} - \frac{p_6}{I_6}. \tag{15c}$$

In mechanical domain terms, Eq. (15) correspond to Eq. (16).

$$m_2\ddot{x}_2 = F - kx_0, \tag{16a}$$

$$m_1\ddot{x}_1 = kx_0 - D\dot{x}_1 \tag{16b}$$

$$x_0 = x_2 - x_1. \tag{16c}$$

These are exactly the same equations we will end up with if we do it in the classical Newtons 2*nd* law approach. The block diagram for these equations are shown in Fig. 14. Block diagrams represent the structure of the mathematical model and displays which variables must be known in order to compute others. They do not reflect the physical structure. The reason is that feedback is represented in separate feedback loops.

By using bond graph as the modeling tool we get a good overview of the model's physical structure and we can do

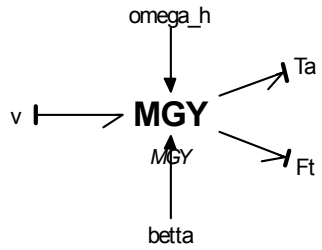



Fig. 15. Modulated gyrator transforming wind speed into aerodynamic torque and thrust force.

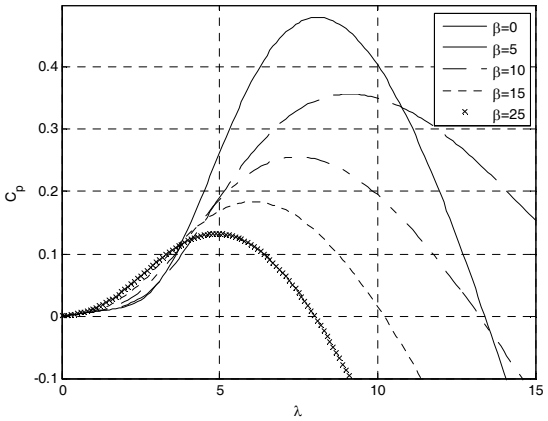



Fig. 16. $C_p$ curve.

simulations in one step, instead of first deriving the equations and then drawing the block diagram.

## 3. Model description

In the following section, the bond graph based modeling for the different subsystems shown in Fig. 1, will be presented. It is shown that the bond graph method provides a hierarchical modeling for the entire wind turbine generating system as well as the system equations can be extracted symbolically in a structured way.

### *3.1 Aerodynamics*

In the aerodynamics part we need to find a way to convert the wind into torque and thrust force, i.e. transform a flow into efforts. This is done by means of a modulated gyrator. We use the torque and thrust equations given in Eqs. (3) and (4). The only difference between a MGY and a GY is that the gyrator ratio is not a constant parameter, but it is a varying parameter. In this case the transformation is dependent on two varying parameters, the pitch angle $\beta$ and the rotor rotational speed $\omega_r$.

A generic equation is used to model $C_p$. This equation and its coefficients, based on the turbine characteristics of Ref. [19], is shown in Eq. (17). A plot of the $C_p$ curve is shown in Fig. 16, the plot is made with different pitch- and $\lambda$ values. Similar formulas can be found regarding the thrust force coefficient $C_T$, in our calculations only a simple relation is used.

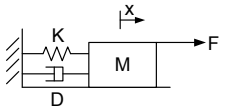



Fig. 17. Mass spring damper.

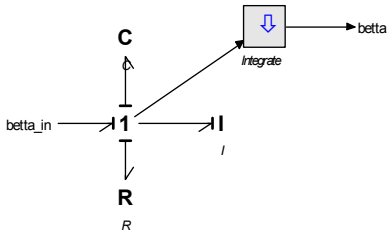



Fig. 18. Bond graph of pitching system.

$$\lambda = \frac{\omega_r R}{v} \tag{17a}$$

$$\lambda_i = \frac{1}{\frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^2 + 1}} \tag{17b}$$

$$C_p = c_1 \left(\frac{c_2}{\lambda_i} - c_3\beta - c_4\right) c^{\frac{c_5}{\lambda_i}} + c_6\lambda \tag{17c}$$

where $c_1 = 0.5176$, $c_2 = 116$, $c_3 = 0.4$, $c_4 = 5$, $c_5 = 21$ and $c_6 = 0.0068$.

### *3.2 Pitching system*

The pitching mechanism can be modeled as a second order system:

$$\omega_n^2\theta_{ref} = \ddot{\theta} + 2\zeta\dot{\theta} + \omega_n^2\theta \tag{18}$$

where $\theta_{ref}$ is the reference pitch angle, $\omega_n$ is the natural frequency and $\xi$ is the damping ratio. By setting up the dynamic equation of the mass-spring-damper system in Fig. 17, we can compare the elements in the equation with Eq. (18). In this way we can set up the bond graph in Fig. 18 with appropriate coefficients.

$$F = \theta_{ref} \qquad M = \frac{1}{\omega_n^2} \qquad D = \frac{2\zeta}{\omega_n^2} \qquad K = 1.$$

### *3.3 Drive train*

A sketch of a two-mass drive train model is seen in Fig. 19. As discussed in the introduction there are many types of drive train models, ranging from for example one- to six mass models. For simplicity we will assume a two-mass-model is enough. To derive the governing equations from a two-mass model is not too hard. If we are talking about a six-mass model the work can be quite extensive, and the possibility of making a mistake in the process is high. This is one of the reasons bond graph is a safer choice. As the complexity of the

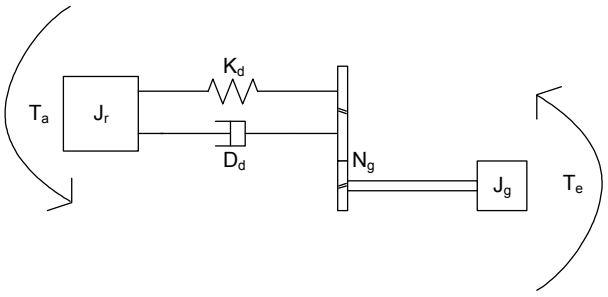



Fig. 19. Sketch of wind turbine.

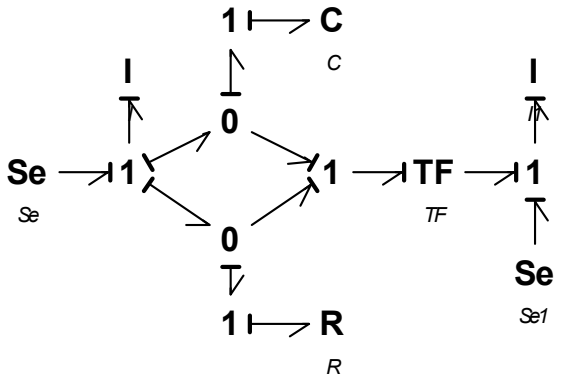



Fig. 20. First bond graph of drive train.

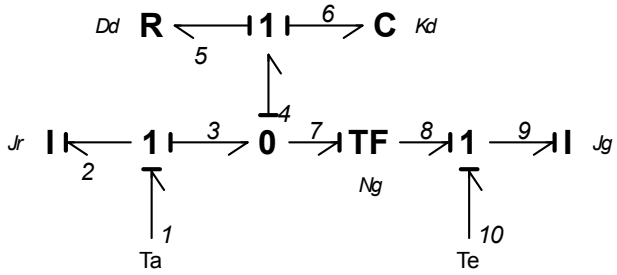



Fig. 21. Final bond graph of drive train.

mechanical system grows, our work as modelers stays about the same. If we have a six-mass model with many springs and dampers, this gives us many equations and to translate this into a block diagram can take quite some time. As for dealing with bond graph, the work is to set up the graphical representation. If we want to see the equations, these can be derived in a very specific way. Or, off course, we can choose to get them from the bond graph simulation program 20-sim.

By utilizing Newton's second law on rotational form of the wind turbine sketch in Fig. 19, we end up with the following differential equations:

$$T_r = I_r\dot{\omega}_r + \dot{\phi}_\Delta D_d + \phi_\Delta K_d \tag{19a}$$

$$-T_g N_g = I_g N_g^2 \frac{\dot{\omega}_g}{N_g} - \dot{\phi}_\Delta D_d - \phi_\Delta K_d, \tag{19b}$$

where

$$\phi_\Delta = \phi_r - \frac{\phi_g}{N_g}, \qquad \dot{\phi}_\Delta = \omega_r - \frac{\omega_g}{N_g}.$$

In a quite intuitive way we can translate the mechanical system in Fig. 19 into a bond graph representation, as shown in Fig. 20. This can again be simplified a bit in order to make a minimal bond graph representation, see Fig. 21. The bond graph model consists of three 1-junctions and one 0-junction. The 1-junction connected to the rotor inertia describes the rotor rotational speed. Since there are dynamics in between the rotor inertia and the generator inertia, they do not have the same speed. This is the reason for the 0-junction, because we know the transferred torque is the same (no loss included in the drive train). The 1-junction connected to the resistive- and the compliance element indicates the rotational speed difference between the two inertias. This connection also indicates that the compliance- and resistive element have the same rotational speed (flow), but different torque (effort). The last 1-junction is connected to the generator inertia and describes the generator rotational speed.

Once the bond graph representation is made, the procedure for extracting its governing equations is quite straight forward. One has to follow some certain rules, and at the end the equations will be the outcome. We can also choose to get the equations from the simulation software. The equations can be derived as follows.

From the bond graph representation we see there are three dynamic elements, two inertias and one spring, i.e. three dynamic equations must exist. These first order differential equations are given in Eq. (20).

$$\begin{aligned}\dot{p}_2 &= e_2 = e_1 - e_3 \\ &= T_a - \frac{q_5}{C_6} - R_5 f_5,\end{aligned} \tag{20a}$$

$$\begin{aligned}\dot{q}_5 &= f_5 = f_3 - f_7 \\ &= \frac{p_2}{I_2} - N_g \frac{p_9}{I_9},\end{aligned} \tag{20b}$$

$$\begin{aligned}\dot{p}_9 &= e_9 = e_8 + e_{10} \\ &= -T_e + \frac{1}{N_g}\left(\frac{q_5}{C_6} + R_5 f_5\right).\end{aligned} \tag{20c}$$

With some manipulations this is exactly the same as in Eq. (19).

### *3.4 Generator*

There are many ways to model the generator dynamics. One of the recurring ways is with an equivalent circuit. In this system we assume that a first order transfer function will capture its dynamics. We do this in the same way as for the pitching system, but since it is first order we do not include the spring.

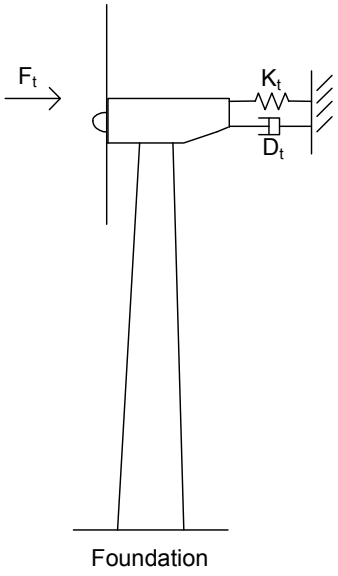



Fig. 22. Sketch of wind turbine structure.

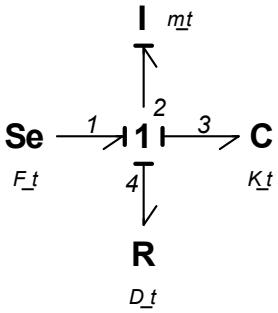



Fig. 23. Bond graph of tower motion.

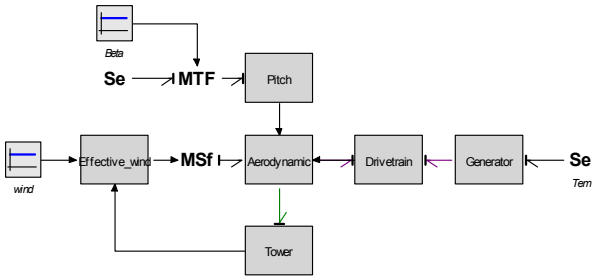



Fig. 24. Bond graph of wind turbine generating system.

$$T_{ref} = \tau \dot{T}_e + T_e, \tag{21}$$

where $T_{ref}$ is the reference torque and $\tau$ is the time constant. In this way we can set up the bond graph similar to Fig. 18 with appropriate coefficients.

$$F = T_{ref}, \quad M = \tau, \quad D = 1.$$

### *3.5 Tower*

In Fig. 22 we see the turbine sketch and where the thrust force is acting on the structure. It is assumed that the tower movement will not influence the mechanical system, it only affects its input, i.e. the wind speed. The bond graph model of the tower can be seen in Fig. 23. Since the deflections of the tower are assumed to be small, we assume tower movement only in horizontal direction.

The dynamic equation from the bond graph model (Fig. 23), is given in Eq. (23).

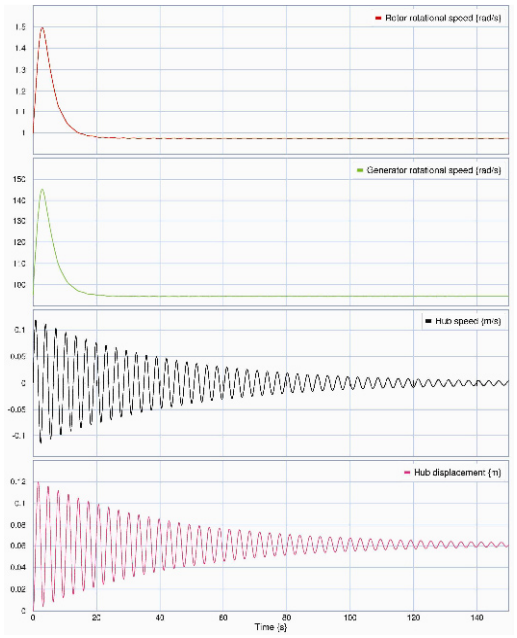



Fig. 25. Time behavior of the selected signals from 20-sim.

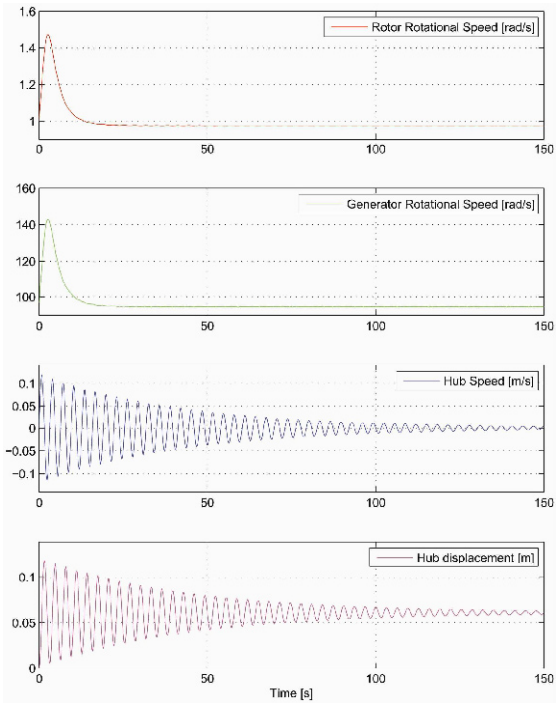



Fig. 26. Time behavior of the selected signals from MATLAB/ Simulink.

$$\dot{p}_2 = S_e - \frac{p_2}{I} - \frac{q_3}{C} \tag{22a}$$

$$\dot{q}_3 = \frac{p_2}{I}. \tag{22b}$$

We can rewrite Eq. (22) in a non bond graph notation:

$$m_t\ddot{z} = F_t - D_t\dot{z} - K_t z, \tag{23}$$

where $m_t$ is the tower mass, $F_t$ is the thrust force acting on the tower, $D_t$ is the tower damping and $K_t$ is the tower stiffness.

## 4. Simulation results

In this section we want to validate the bond graph design. The procedure is to first connected together all the subsystems from section III. Second, the same model is implemented as block diagrams in MATLAB/Simulink. This software is widely established throughout the academic community and the result from MATLAB/Simulink will act as a reference output for validation purpose.

The bond graph representation of the system setup in Fig. 1 is shown in Fig. 24. The inputs to the systems are pitch angle, reference power and wind speed.

The simulations are made with maximum pitch angle, maximum wind condition, maximum power and with initial conditions on the rotor and generator. All wind turbine parameters used in the simulations are found in Ref. [19]. Once the simulations are carried out in the two softwares, time behavior of the most important dynamics are inspected. As seen in Figs. 25 and 26, the behavior of the two systems are identical. This confirms the fact that we eventually end up with the same governing equations whether one uses the classical Newton's $2nd$ law or the bond graph approach.

## 5. Conclusion

The purpose of this paper is to make a nonlinear model of a wind turbine generating system by using the bond graph approach. We are not looking to validate a specific turbine system, but we want to show a simple and suitable way to model it. The nonlinear wind turbine consists of drive train, pitching system, tower and generator. To model dynamic systems in the classical way and the bond graph way is quite different, but the outcome is a model with exactly the same governing equations. We have tried to emphasize that the bond graph approach will give a better understanding of what actually happens in the system. This include; spotting algebraic loops right away and maintaining integral causality, to name a few. The approach is unified, which means one can model all types of physical systems with the same methodology. Today, most engineers must work and interact in many different disciplines. An understanding of the intersections of these different disciplines is a valuable asset for any engineer. Based on the results in this paper, interesting future research include performing control design using bond graph and possibly constructing an offshore wind turbine model. Simulations can be carried out in the software 20-sim, or if one prefer, it is also possible to export the model to MatLab/Simulink via S-function.

Table 1. Wind turbine parameters.

| | |
|---|---|
| Pitch | |
| Natural frequency | $\omega_n = 0.88\left[\frac{rad}{s}\right]$ |
| Damping ratio | $\zeta = 0.9[-]$ |
| Maximum pitch angle | $\beta_{\max} = 25[^\circ]$ |
| Minimum pitch angle | $\beta_{\min} = -5[^\circ]$ |
| Drive train | |
| Nominal power | $P_{nom} = 5\times10^6[W]$ |
| Rotor inertia | $I_r = 5.9154\times10^7[Kg\cdot m^2]$ |
| Generator inertia | $I_g = 500[Kg\cdot m^2]$ |
| Drive train stiffness | $K_d = 8.7354\times10^8\left[\frac{N}{rad}\right]$ |
| Drive train damping | $D_d = 8.3478\times10^7\left[\frac{N}{rad\cdot s}\right]$ |
| Gear ratio | $N_g = 97[-]$ |
| Generator | |
| Time constant | $\tau = 0.1[s]$ |
| Nominal generator speed | $\omega_{g\max} = 122.91\left[\frac{rad}{s}\right]$ |
| Minimum generator speed | $\omega_{g\min} = 70.16\left[\frac{rad}{s}\right]$ |
| Structure/Tower | |
| Rotor speed | $R = 63[m]$ |
| Hub height | $h = 90[m]$ |
| Tower mass | $m_t = 4.2278\times10^5[Kg]$ |
| Tower stiffness | $K_t = 1.6547\times10^6\left[\frac{N}{m}\right]$ |
| Tower damping | $D_t = 2.0213\times10^3\left[\frac{N}{m\cdot s}\right]$ |

## Appendix

The wind turbine parameters used for this study in the model system are given in Table 1.

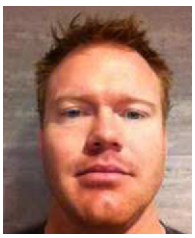
## Acknowledgment

This work has been (partially) funded by Norwegian Centre for Offshore Wind Energy (NORCOWE) under grant

193821/S60 from Research Council of Norway (RCN). NORCOWE is a consortium with partners from industry and science, hosted by Christian Michelsen Research.

## References

[1] C. Kleijn, *20-sim 4.1 Reference manual*, Enschede: Controllab Products B. V. (2009).
[2] http://www.wwindea.org.
[3] J. Tamura, T. Yamajaki, M. Ueno, Y. Matsumura and S. Kimoto, Transient stability simulation of power system including wind generator by PSCAD/EMTDC, *IEEE Porto Power Tech Conference* (2001).
[4] G. Tapia, A. Tapia, I. Zubia, X. Ostolaza and J. R. Saenz, Electrical fault simulation and dynamic response of a wind farm, *International Conference on Power and Energy Systems* (2001).
[5] T. Petru and T. Thiringer, Modeling of wind turbines for power system studies, *IEEE Transactions on Power Systems*, 17 (4) (2002) 1132-1139.
[6] V. Akhmatov and H. Knudsen, Modeling of windmill induction generators in dynamic simulation program, *International Conference on Electric Power Engineering* (1999).
[7] P. Ledesma, E. Agneholm, M. Martins, A. Perdana and O. Carlson, Validation of fixed speed wind turbine dynamic models with measured data, *Renewable Energy*, 32 (8) (2007) 1301-1316.
[8] S. Muyeen, M. Ali, R. Takahashi, T. Murata, J. Tamura, Y. Tomaki, A. Sakahara and E. Sasano, Comparative study on transient stability analysis of wind turbine generator system using different drive train models, *IET Renewable Power Generation*, 1 (2) (2007) 131-141.
[9] T. J. Larsen, *How 2 HAWC2, the user's manual*, Risø-R-1597(ver.3-9)(EN) (2009).
[10] C. L. Bottasso and A. Croce, *Cp-Lambda user manual*, Dipartimento di Ingnegneria Aerospaziale, Politecnico di Milano (2009).
[11] J. Jonkman and M. L. Buhl, *FAST User's Guide*, Technical Report NREL/EL-500-38230 (2009).
[12] *MATLAB. version 7.10.0 (r2010a),* Natick, Massachusetts: The MathWorks Inc. (2010).
[13] P. Brath, K. Hammerum and N. K. Poulsen, A fatigue approach to wind turbine control, *Journal of Physics: Conference Series*, 75 (1) (2007).
[14] D. M. Eggleston and F. S. Stoddard, *Wind Turbine Engineering Design,* New York: Van Nostrand Reinhold Co. (1987).
[15] H. M. Paynter, *Analysis and Design of Engineering Systems*, MIT Press (1961).
[16] D. L. Margolis, D. C. Karnopp and R. C. Rosenberg, *System Dynamics: Modeling and Simulation of Mechatronic Systems*, John Wiley and Sons Ltd. (2006).
[17] W. Borutzky, *Bond Graph Methodology: Development and Analysis of Multidisciplinary Dynamic System Models*, Springer (2010).
[18] E. Pedersen og H. Engja, Mathematical Modeling and Simulation of Physical Systems, *Lecture notes in course TMR4275 Modeling, Simulation and Analysis of Dynamic Systems at NTNU*, Trondheim, Norway (2003).
[19] L. C. Henriksen, Model predictive control of a wind turbine, *Master thesis*, Technical University of Denmark (2007).
[20] S. Heier, *Grid Integration of Wind Energy Conversion Systems*, John Wiley and Sons Ltd. (1998).



**Tore Bakka** was born in Haugesund, Norway, in 1983. He earned his B.S. degree in Electrical Engineering at the University of Stavanger in 2007 and his M.S. degree in Mechatronics at the University of Agder in 2010. Currently, he is pursuing a Ph.D. degree at the University of Agder, working mainly with wind turbine control.

**Hamid Reza Karimi**, born in 1976, is a Professor in Control Systems at the Faculty of Engineering and Science of the University of Agder in Norway. His research interests are in the areas of networked control systems, robust control/filter design and vibration control with an emphasis on applications in engineering. Dr. Karimi is a senior member of IEEE and serves as chairman of the IEEE chapter on control systems at IEEE Norway section. He is also serving as an editorial board member for some international journals, such as Mechatronics, Information Sciences, Neurocomputing, etc.