Emulation of a Low Power Wind Turbine with a DC motor in Matlab/Simulink

R. II. Ovando, J. Aguayo, M. Cotorogea
Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET)
Departamento de Ingeniería Electrónica
Interior Internado Palmira s/n Apdo. 5-164.
C.P. 62050, Cuernavaca, Morelos, México
Tel.: +52 (777) 3 62 77 70

Abstract - In this paper the experimental implementation of a wind turbine emulator (WTE) for wind energy conversion systems (WECS), using a separately excited DC motor are presented. The model of the wind turbine (WT) and the control of the DC motor are implemented in MATLAB/SIMULINK. The developed WTE allows a WECS to be analyzed without the need of real wind turbines. The WECS considered in this work is composed by a DC motor coupled with a double fed induction machine (DFIM). The DC motor and its control actuate as the WT and the DFIM actuates as the generator. The simulations and experimental results are obtained for a 300W prototype.

I. INTRODUCTION

Electric energy has been the booster of world-wide industry development in the last two hundred years. It has been commonly generated by means of combustion of fossil fuels. However, because this generation type is polluting, expensive and in some years it will be insufficient; it becomes necessary to find alternative sources of electric generation, as the renewable energy.

Among the different types of renewable energies, the main ones are: hydroelectric, solar and wind energy. Today, the cost of energy (CoE) from wind generation at good wind sites (>8.5 m/s annual wind velocity) is lower than the CoE from the best fossil fuel generation technology. It is clear that with further technology advancements, CoE for alternative energy sources will continue to decline, and demand for alternative energy generation equipment will continue to grow rapidly [1].

For the massive use of wind energy conversion systems (WECS) the study of the grid connected systems is important. However, the development and tests of new technologies must be performed using an isolated and controlled test bench. In the literature different systems of this kind have been presented, but the majority of them are focused on the control and modelling of the generator. In [2] a brief description of the different kinds of WECS is presented, but this document refers to the initialization problem and do not report the implementation of any prototype. In [3] and [4] a review of the main kinds of generators and static converters is presented. In [5] and [6] use a DC motor and an induction motor respectively, the implementation in both documents is rather complex and the dynamics of the wind turbine (WT) in [6] it is not modeled.

A Wind Turbine Emulator (WTE) using a DC motor to characterize the real WT is presented in [7] where both the static and dynamic characteristics are modeled. The experimental implementation is the best ones of all the reviewed documents, but it is possible to improve the performance using a DSP.

This paper proposes a novel scheme of implementation in a DSP using MATLAB/SIMULINK to generate the processor code. In this way, a robust implementation is obtained, since MATLAB/SIMULINK offers a friendly and a powerful graphic environment. When it is required to change the parameters in the model of the WT or in the DC motor control any user can realize it without problem.

II. WIND TURBINE MODEL

The WT model is divided in two parts. The first part is the static model, which consists in the CP (Current vs. Power) curve and the generated torque. The inputs for this model are the wind speed $V_w$, the angular speed $\omega_r$, and the pitch angle $\beta$ which in this case remain fixed. The output is the rotor torque $T_m$.

The second part is the dynamic model, which takes into account the inertia, damping and stiffness in the shafts of the WT and the double fed induction machine (DFIM). A gearbox is considered to multiply the speed rate between the low and high speed shaft. The inputs for this model are the rotor torque $T_m$ and the generator torque $T_g$. The outputs are the speed in the low speed shaft $\omega_r$ and the speed in the high speed shaft $\omega_g$.

A. Static Model

The three most important currently applied wind generation systems are the following:
- A constant speed wind turbine.
- A variable speed wind turbine with DFIG (this case is presented in the paper).
- A variable speed wind turbine with a direct drive synchronous generator [2].
The static model of the WT presented in this paper corresponds to the horizontal lift based designs. The equations that represent the static behavior of the WT are included below.

Equation (1) defines the ratio of the angular speed of the blade tip and the shaft rotating speed.

$$\lambda = \frac{\omega_r}{\omega_w}$$  \hspace{1cm} (1)

Where: \(r_r\) is the rotor radius in meters (m), \(\omega_r\) is the angular speed in the low speed shaft in (rad/s) and \(V_w\) the wind speed in (m/s).

The power extracted from the wind is limited by the aerodynamic design. The power curve of a turbine is defined by (2) where \(C_p\) is the power coefficient and \(\beta\) the pitch angle of the WT blades. \(c_i, c_2, c_3, c_4, c_5\) and \(c_6\) are power coefficients and depend on each WT design.

$$C_p(\lambda, \beta) = c_i \left( \frac{c_2}{\lambda_i} - c_3\beta - c_4 \right) e^{\frac{-c_5}{\lambda}} + c_6\lambda$$  \hspace{1cm} (2)

$$\frac{1}{\lambda_i} = \frac{1}{\lambda + 0.08\beta - \beta^3 + 1}$$  \hspace{1cm} (3)

The torque coefficient is defined in (4), and is necessary to determine the mechanic torque \(T_m\) of the WT given in (5).

$$C_M = \frac{C_p}{\lambda}$$  \hspace{1cm} (4)

$$T_m = \frac{1}{2} \rho A r_r V_w^2 C_M$$  \hspace{1cm} (5)

Where \(\rho\) is air density in (kg/m³) and \(A\) is the area spanned by the rotor blades in (m²) [1]-[6]. Fig. 1 shows the static characteristic of the modeled WT.

B. Dynamic Model

All existing systems changing with a rate of time and when this rate is significant they are called dynamic systems. The main characteristic is that the output at any time instant depends of the previous state and not only on the current input.

In this work two dynamic models are presented. Equation (6) models the inertia-damping-stiffness of the WT shaft \(J_r, B_r\) and \(K_r\) respectively and the inertia-damping-stiffness of the DFIM \(J_g, B_g\) and \(K_g\) respectively. Where \(N\) is the gearbox rate, this model will be further referred to as case I. Fig. 2 shows the dynamic model with the considered parameters.

When case 1 is modeled a set of complex equations is obtained. Therefore certain basic equations must be considered in order to reduce the complexity of the system. Finally, the set of equations (6) is obtained that describes the behavior of the dynamic system.

$$T_{re} = J_r \frac{d}{dt} \omega_r + B_r (\omega_r + \omega_g) + K_r (\theta_r + \theta_g)$$  \hspace{1cm} (6)

$$-T_g = J_g \frac{d}{dt} \omega_g + B_g (\omega_g + \omega_{re}) + K_g (\theta_g + \theta_{re})$$  \hspace{1cm} (6)

$$\omega_g = \frac{d}{dt} \theta_g$$

$$\omega_{re} = \frac{d}{dt} \theta_{re}$$

For simulation purposes and for cases where it is necessary to emulate the dynamic characteristics of the DFIM the equations system (6) is used. However, in the experimental implementation developed in this work only the dynamic of the WT is considered. Neglecting the DFIM and the stiffness of the shaft the reduced set of equations (7) is obtained.

$$T_m = J_r \frac{d}{dt} \omega_r + B_r \omega_r$$

$$\omega_r = \frac{d}{dt} \theta_r$$  \hspace{1cm} (7)

Where \(T_m\) is the mechanic torque by the WT, \(J_r\) is the inertia of the WT rotor; \(B_r\) is the damping coefficient of the WT shaft and \(\omega_r\) is the angular speed of the WT shaft [8].

Since a commercial WT of the low power required by the double-fed induction generator considered in this work does not exist, a power scaling must be performed in accord to WT’s commercially available. Table I presents the parameters used in the experimental implementation.

Fig. 1. Static characteristics of the WT.

Fig. 2. Dynamic characteristics of the WT.
### TABLE I. WT PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_r$</td>
<td>1 m</td>
</tr>
<tr>
<td>$\rho$</td>
<td>1.225 kg/m$^3$</td>
</tr>
<tr>
<td>$J_r$</td>
<td>0.3 Nms$^2$</td>
</tr>
<tr>
<td>$B_r$</td>
<td>0.024 Nms</td>
</tr>
<tr>
<td>$N$</td>
<td>4</td>
</tr>
<tr>
<td>$P_n$</td>
<td>300 W</td>
</tr>
</tbody>
</table>

### III. DC MOTOR CONTROL

The DC motor with its control is the basic stage of the WTE system, since it is the responsible to reproduce the behavior of a real WT. The model of the DC motor used in this work is omitted, since it can be found in any basic book of electrical machinery. To drive the DC motor two chopper converters (class “A”) are used.

Prior to selecting the control scheme of the DC motor, its operating region must be defined. It is known that the nominal speed of the DFIM is 3600 rpm and the DC motor must be able to reach a 30% higher speed.

However, the nominal operating region of the motor is hardly 1800 rpm. In order to achieve the required speed, the motor must be operated in the field weakening region. This operating scheme allows achieving two or three times the nominal speed.

To obtain this kind of operation the armature circuit and the field circuit must be controlled independently. A class “A” chopper converter was designed for the armature circuit and a second chopper converter for the field circuit.

The DC motor control uses a cascaded scheme (Fig. 3) with an inner current loop and an outer speed control loop, for the nominal controller region. The field weakening controller uses only a speed control loop. Both of them are PI controllers.

The DC motor parameters were obtained by a motor characterization using the standard IEEE 113 [9]. Table II presents the required parameters of the DC motor used to emulated the behavior of a real WT. Discrete

As can be seen in Fig. 3, the control scheme contains a block called Embedded MATLAB function, which is implemented in SIMULINK. It is mandatory to use this kind of block, since the application contains a program algorithm and it is the only supported block to generate executable code for the TI board (TI=Texas Instruments) and by means of which we can develop any program algorithm.

In this particular case, the block Embedded MATLAB Function has an algorithm that enables the motor to operate below or above its nominal speed, i.e. in the interval from 0 to 1800 rpm and from 1800 to 4600 rpm, respectively. In the first interval, the algorithm selects the converter that drives the armature circuit, and sets the required conditions in the converter that drives the field circuit. In the second interval the algorithm selects the converter that drives the field circuit, and sets the required conditions in the converter that drives the armature circuit.

The root locus method was used to calculate the gain of each controller. The calculated parameters and summarizes the Table III.

### TABLE II. DC MOTOR PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Armature resistance, $R_a$</td>
<td>1.276Ω</td>
</tr>
<tr>
<td>Field resistance, $R_f$</td>
<td>163.39Ω</td>
</tr>
<tr>
<td>Armature inductance, $L_a$</td>
<td>23.62 mH</td>
</tr>
<tr>
<td>Field inductance, $L_f$</td>
<td>23.13 H</td>
</tr>
<tr>
<td>Mutual inductance, $L_{af}$</td>
<td>821.6 mH</td>
</tr>
<tr>
<td>Inertia, $J_m$</td>
<td>25e-3 Nms$^2$</td>
</tr>
<tr>
<td>Damping, $B_m$</td>
<td>1e-3 Nms</td>
</tr>
<tr>
<td>Power</td>
<td>¾ HP</td>
</tr>
</tbody>
</table>

### TABLE III. DC MOTOR CONTROLLER PARAMETERS

<table>
<thead>
<tr>
<th>Controller</th>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$t_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Armature current controller</td>
<td>0.0732</td>
<td>43.92</td>
<td>157 ms</td>
</tr>
<tr>
<td>Armature speed controller</td>
<td>1.15</td>
<td>23</td>
<td>387 ms</td>
</tr>
<tr>
<td>Field speed controller</td>
<td>0.0181</td>
<td>0.2715</td>
<td>2.98 s</td>
</tr>
</tbody>
</table>

![Fig. 3. DC motor control scheme.](image3.png)

![Fig. 4. Test bench with the DC motor.](image4.png)
IV. EXPERIMENTAL RESULTS

To understand the structure of the WTE should be kept in mind the parts that compose it and the operation they should have. The system shown in Fig 5 consists in a personal computer with the real wind profile, a DSP card with the wind turbine model and the DC motor control, a DC motor and the power converter to drive it, and signal conditioning block.

Each block that composes the WTE should fulfill certain function:

- The vector of data that contains the profile of wind are like one of the entrances that excites the pattern of the turbine, will be a correspondent to the card eZdspF2812 by means of the tool of communication Real-Time Exchange it Dates (RTDX).
- The turbine model generates the reference for the DC motor control.
- The DC motor control sends the appropriate control signs to the power converter.
- The power converter drives the DC motor such that it reproduces the behavior of a wind turbine.
- The eZdspF2812 card carries out the digital signals processing of the turbine model, the DC motor control and the MATLAB wind profile.
- Finally, the signal conditioning block modifies the signs of the current sensors and of the DC motor speed at the level required by the analog-digital converters (ADC) of the eZdspF2812 card.

An important characteristic for in the implementation of the WTE is that the system should be robust and of easy handling for the user. In this way, if it is required to make some changes in any of the turbine model, the controllers gain or the wind profile data, the user has not to be an expert in the topic. This is achieved by using MATLAB/SIMULINK which is a powerful software and of common use among students.

The tool of SIMULINK that allows the implementation of the wind turbine emulator in the eZdspF2812 card is the Embedded Target for TI C2000 DSP that MATLAB/SIMULINK integrates with the tools eXpressDSP of TI.

The Embedded Target for the TI C2000 DSP uses the Real-Time Workshop (RTW) tool to automatically generate, pack and compile the source code of the SIMULINK model, creating software applications in real time in a variety of systems; RTW compiles a project Code Composer Studio (CCS) starting from the C code generated from the SIMULINK pattern, where the whole system should be discretized, since the RTW does not support continuous models.

The implementation of the wind turbine emulator in MATLAB/SIMULINK is represented in Fig. 6, where the speed control block that can be observed corresponds to the diagram in Fig. 3.

One of the reaches obtained in this work was the implementation of a WTE able to accept real wind data profiles. Therefore, it was investigated what institutions in Mexico are devoted to carry out measurements of the natural wind resource. It was found that the Mexican Electric Research Institute (IIE) has a database where information of wind measurements is available. These measurements come from the different stations installed along the Mexican Republic.

The evaluation of the wind resource is carried out based on international recommendations that indicate that wind speed data should be acquired starting from samples each 2s as minimum which are averaged in a 10min period[10].

In the Fig. 7 is shown the wind profile corresponding to the month of January 2006 (744 hrs. duration) in La Venta, Oaxaca in Mexico. It represents the average wind at 32m of height.

Due to the low power characteristics of the wind turbine emulated in this work (Table I), the averaged wind profile is not appropriate, since the dynamic behavior of the turbine is faster than the changes in the wind speed values of the averaged profile acquired by the IIE, whose characteristics are adapted for large wind turbines.

![Fig. 5. Block diagram of the WTE system.](image)

![Fig. 6. Implemented WTE system.](image)
In order to observe the dynamic behavior of the systems under realistic wind conditions, it was necessary to adjust the wind profile of Fig. 7 that has a sampling period of 10min (600s), to a smaller sampling rate of 5s. The durations of the experiment was shorten to 1hr. with the wind profile shown in the dotted area of Fig. 7. The new scale corresponds to the duration of the test carried out in the laboratory.

The re-sampled wind profile was used to carry out a simulation with the turbine model implemented in MATLAB/SIMULINK as well as experimental test with the WTE in the laboratory. Both results are illustrated in Fig. 8, representing the speed of the high speed shaft, dotted line for the model, bold line for the experiment.

At the time scale of Fig. 8, the presented results of the WTE are in excellent agreement with the model, however by analyzing standard deviation a certain error can be detected due to different factors.

A zoom in the interval between 2100s and 2300s is shown in Fig. 9 to observe the error in a smaller time scale. It can be seen that the obtained results are satisfactory. In Table IV and V the quality of the system is evaluated, according to the results in Fig. 8 and 9, respectively.

The standard deviation is a measure of the degree of dispersion of the data of the stocking. Said otherwise, the standard deviation is simply the prospective variation regarding the arithmetic stocking and therefore, it is measured in the same units that the variable. A large standard deviation indicates that the points are far from the stocking, and a small deviation indicates that the data are contained near the stocking. The Table IV presents the evaluation of the standard deviation for the laboratory test (Fig. 8). The Table V shows the calculation for the approach of the Fig 8 (Fig. 9). By means of the presented data one can observe a good acting of the system WTE.

<table>
<thead>
<tr>
<th>Table IV: Standard Deviation of the Fig 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>Proposed</td>
</tr>
<tr>
<td>Real</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table V: Standard Deviation of the Fig 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>Proposed</td>
</tr>
<tr>
<td>Real</td>
</tr>
</tbody>
</table>

V. CONCLUSIONS

The work presented in this paper contributes to the study of alternative energy generation, which has a boom nowadays at a world-wide level; in the particular case of wind energy generation, this work will allow to carry out studies in regions where the natural resource is unavailable, with test benches at a low power scale.
A wind turbine emulator (WTE) has been developed that reproduces the behavior of a WT using a real wind profile as input. By implementing the emulator in a test bench for wind energy conversion, it is possible to validate and analyze different control strategies in this kind of systems. The MATLAB-SIMULINK software offers through the RTW toolbox a viable form to develop and probe models in a simple way.

In the laboratory of electrical machines in CENIDET it was possible to operate the WTE with a lowest dynamic behavior than usually presents the real wind turbines. In spite of the bad speed sensor quality an acceptable following of the speed reference is achieved.

Since the DC motor used to emulate the WT was unable to achieve the nominal speed of the double-fed induction generator, an alternative operating scheme was developed. In this way, the performance of the motor was satisfactory in both operating regions (nominal and field weakening).

References
