Method Article

Method for maximum power point tracking and verification by modeling a unified wind energy conversion system

José-Genaro González-Hernández*, Rubén Salas-Cabrera, Roberto Vázquez-Bautista, Luis-Manuel Ong-de-la-Cruz

Tecnológico Nacional de México / Instituto Tecnológico de Ciudad Madero, Mexico

ABSTRACT

Maximum power point tracking in wind turbines is a topic that has attracted many researchers’ interest; however, the studies presented are usually carried out only at the simulation level, so they lack a verification in the system through real measurements. On the other hand, the system’s modeling is usually quite complex, making it challenging to meet the control objectives. There are unified models in which the system is treated in a generalized way according to various research purposes. This work presents a methodology that simplifies the unified system through a series of dynamic tests that applied to obtained a simplified model much easier to handle without sacrificing the system’s dynamic richness.

• An alternative approach for a unified wind energy conversion system is established by employing physical dynamic tests applied to the wind set.
• A maximum power point tracking is verified by real-time measurements managed by an open-source platform.
• Methodology related to electronic instrumentation and programming is described so the tests can be reproduced without much difficulty.

© 2021 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

ARTICLE INFO

Method name: Tracking and verification of maximum power point by modeling a unified wind energy conversion system
Keywords: Wind emulator, Control law, Multilevel converters, Real-time control
Article history: Received 2 January 2021; Accepted 3 March 2021; Available online 9 March 2021

* Corresponding author.
E-mail address: jgenaro.gonzalez@itcm.edu.mx (J.-G. González-Hernández).

https://doi.org/10.1016/j.mex.2021.101298
2215-0161/© 2021 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)
Specifications Table

Subject Area: Engineering
More specific subject area: Wind energy conversion systems
Method name: Tracking and verification of maximum power point by modeling a unified wind energy conversion system
Name and reference of original method: Comprehensive MPPT method to extract maximum power from wind turbine by using a unified system model. Sai Manoj P, Vijayakumari A, and Sasi K Kottayil. Development of a Comprehensive MPPT for grid connected Wind Turbine Driven PMSG, Wind Energy (Chichester, England), vol. 22 no. 6, (2019), pp. 732–744. DOI: 10.1002/we.2318
Resource availability: The database is freely accessible at http://dx.doi.org/10.17632/363d24mcb6.1, Real-time Linux support can also be found at https://www.rtai.org/

Method details

Introduction

Wind power generation has increased considerably in the last years. New and diverse technologies have been developed to improve wind energy conversion systems (WECS) and maximize wind power extraction [1–4]. Several studies have been carried out about maximum power point tracking (MPPT) in WECS [5–7]; nevertheless, complex systems are usually modeled, and commonly no verification in real measurements is made. On the other hand, explanations of technical procedures for obtaining and processing the data are usually not described. In some cases, costly equipment and platforms are used to develop the studies. Some investigations present a methodology to represent a unified system to simplify the treatment of the WECS, keeping the dynamic richness of the system [8]. This work uses the free and open-source Linux-real time platform [9–10] to emulate and control a WECS tracking and verifying MPPT; it also presents a simplified model of the unified system keeping the dynamic richness of the WECS.

General overview of the system

The unified wind energy conversion system used is presented in Fig. 1; as seen, there are three main stages. In the first place, a wind turbine emulator is presented; it was built by using a CPU under the RTAI Linux platform and software Scicos/Scilab. It is responsible for sending a 0–5 V signal proportional to the wind speed written in the program, which is used to control a speed variator that sends a torque signal to an induction motor coupled to a permanent magnet synchronous generator.

Fig. 1. General overview of the unified wind energy conversion system.
A speed sensor was set to send the signal to CPU 1. In the second place, a multilevel converter stage is presented. CPU 2 contains a program responsible for sending a 0–5 V signal to a microcontroller to control the duty cycle for the multilevel boost converter (MBC); previously, an optocoupler is used to isolate the signal. Emulated wind speed is provided by CPU 1, and the speed sensor provides the angular speed of PMSG. In third place, the dc output of MBC is sent to a three-phase inverter, which provides ac voltage to a purely resistive load.

**Equipment**

- Two desktop computers
- Hall effect sensors
- Two generic data acquisition card connectors
- Induction motor
- Permanent magnet synchronous generator including incremental encoder
- BK power source triple output accuracy
- Multilevel Boost Converter
- Frequency to voltage conversion card for encoder connection
- 4 to 20 mA voltage to current converter card
- Wind turbine generator power cabinet
- Two data acquisition cards PCI6024E
- Digital scope
- Digital multimeters
- Microcontroller
- Optocoupler
- Three-phase inverter
- Resistive load

**RTAI-Lab as a real-time tool**

RTAI (Real-time Application Interface) was used in CPU 1 and 2; it is a patch hosted within the Linux kernel; it can modify the kernel in-depth so that processes are executed with the highest priority. In this way, lower priority processes can be interrupted, thus accessing the necessary resources. We can understand that lower priority processes are those applications of the user’s operating system and programs installed. These processes are alien to the task we want to carry out. RTAI-Lab is a set of tools that includes:

- Scilab/Scicos, which is used to create simulations and to generate a compilation code
- Comedi. A set of drivers for data acquisition cards in Linux.
- Xrtailab, which is a scope that is connected virtually to the executable programs in real-time. It also allows adjusting program parameters during execution.

**Turbine emulation in Scicos**

The program reads the signal sent by frequency to voltage card, and then it converts that signal to rad/s, so tip speed ratio ($\lambda$) can be calculated using Eq. (1) in a Scicos mathematical expression block.

Scicos step blocks are useful to specify and adjust rotor radius and blades pitch angle.

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

Where:

- $\lambda$ is the tip speed ratio
- $\omega_r$ is the angular speed of PMSG
- $R$ is the radius of the wind turbine
- $v$ is the wind speed
Once the program obtains the value of $\lambda$, then it determines the wind power coefficient by using Eq. (2)

$$C_p(\lambda, \beta) = [a_0 + a_1(b_0\beta + a_2)] \sin \left[ \frac{\pi (\lambda + a_3)}{a_4 + a_5(b_1\beta + a_6)} \right] + a_7(\lambda + a_8)(b_2\beta + a_9)$$  \hspace{1cm} (2)$$

Where coefficients used were (Tables 1 and 2):

After obtaining the power coefficient, then turbine torque ($T_m$) is calculated using Eq. (3)

$$T_m = \frac{1}{2} C_t \rho \pi R^2 v^2 = \frac{1}{2} \frac{C_p \rho \pi R^3 v^2}{\lambda}$$  \hspace{1cm} (3)$$

Where:

$C_t$ is the torque coefficient

**Speed variator**

The program stored in CPU 1 demands an analog signal corresponding to the emulated turbine’s torque: this signal is a 0–5 dc voltage with which the induction motor is operated. In order to rotate the motor with that voltage signal, a variable speed drive of the ABB brand model ACS350 was used, as shown in Fig. 2. Its function is to regulate the motor’s speed so that the signal adjusts to the application’s real demand. The variable speed drive is installed in a cabinet already wired, ready to operate. Inside this cabinet are the protection fuses, a thermomagnetic switch, an electromechanical contactor, and the start and stop buttons. The variable speed drive was configured so that with a 4–20 mA signal, it could regulate the induction motor’s torque. In order to achieve that, some motor characteristics such as the rated speed range, the operating frequency, and the maximum torque were entered into the drive. Then, to obtain the 4–20 mA signal, a voltage to current converter with a voltage divider at its input was used. This small module is shown in Fig. 3, it has an operating voltage of 7–30 V, an input voltage of 0–5 V, and the current it delivers is 4–20 Ma.

**Rotor speed measurement**

An encoder coupled to the Pepperl + Fuchs brand’s rotor was used to measure the induction motor’s rotor speed, as shown in Fig. 4; it converts 1 rpm to 17.066 Hz. The maximum rotor speed for
Fig. 2. Speed variator used to energize induction motor.

the induction motor is 1800 rpm, so 30.719 kHz is the encoder’s maximum frequency. However, the DAC’s maximum sampling frequency in this project is 10 KHz; because of that, a frequency to voltage converter was used to adjust the signal. This F/V converter was built using two microcontrollers and a digital to analog converter model AD7846, as shown in Fig. 5.

Induction motor/permanent magnet synchronous generator

For the power generation stage, a permanent magnet synchronous generator was used directly coupled to the induction motor from the WindBlue Power brand model DC-520. This generator has its built-in three-phase rectifier so that it can generate alternating and direct current. It manages to generate 12 V at a speed of 240 rpm; its design is brushless, it has Neodymium magnets N50, and it can be directly coupled without the need for a gearbox as it was set in this project. The induction motor used in this system was the Dayton 5N784A Inverter Duty Motor; its specifications are 1HP, rated voltage 230/460 V, rated current 3.12/1.56 A, and 1755 rpm. This element represents the real blades and the rotor, which rotate due to the wind’s action that falls directly on them and is mechanically coupled to the generator through the gearbox. The angular velocity values in which it operates range between 200 RPM for wind speeds of 3 m/s and 1200 RPM for 8 m/s. Fig. 6 shows the direct coupling between induction motor and PMSG.

Multilevel boost converter

A multilevel boost converter was used to amplify the dc voltage delivered by the permanent magnet generator. The duty cycle of the boost converter is controlled through an ATMega 328p
Fig. 3. Voltage to current converter.

Fig. 4. Encoder coupled to induction motor.

microcontroller, which adjusts the duty cycle through a reference voltage of 0–5 V where 0 volts correspond to 0% and 5 V to 100%; this voltage is sent through the RTAI software by CPU 2 via the data acquisition card. Fig. 7 shows the MBC.

Three-phase inverter

The inverter converter used for the project was the Lab-Volt IGBT Chopper/Inverter module. It has seven insulated gate bipolar transistors, of which six are used to implement the inverter. These IGBT’s are protected against various severe operating conditions, such as short circuits, overvoltage, overcurrent, and overheating. The seventh IGBT, together with a discharge resistor, allows for smooth dissipation of excess power on the DC bus. To control the switching of the IGBT’s, the Chopper/Inverter Control module of the same brand Lab-Volt was used, which generates six pulse signals of 0–5 V through a 9-pin connector, as shown in Figs. 8 and 9. Various control modes can be selected on the front panel, but for this specific project, 180° modulation was selected.
Obtaining dynamic unified model

When considering the system with all the stages and involved elements, several difficulties can be presented because of the non-linearities inherent to the WECS. A high-order non-linear model would probably be derived from the analysis. To simplify the model, a set of experimental dynamic tests applied to the input of MBC were made based on a standard state-space dynamic model, as shown in Eq. (4).

\[
\dot{x}(t) = Ax(t) + Bu(t) \quad \text{state equation}
\]

\[
y(t) = Cx(t) + Eu(t) \quad \text{output equation}
\]

Where:
A, B, C and E are matrices of nxn, nxp, qx n and qxp dimension respectively

$x(t)$ is the state vector

$u(t)$ is the input

$y(t)$ is the output
Fig. 9. Front panel of IGBT Chopper/Inverter control unit module.

$x(t)$ was defined as the output voltage $V_o$, $u(t)$ is the duty cycle $D$, and $y(t)$ is also $V_o$. Seven steps from 10 to 70% were applied from CPU 2; the wind speed was set as 5 m/s, and the turbine radius was established as 0.6 m by CPU 1. Fig. 10 presents the response of the system to the steps.

The procedure for obtaining the parameters of the unified system was:

- Supply a step signal in $u(t)$ and store the output $y(t)$
- Obtaining the system time constant based on the transient response previously-stored and then determine matrix $A$
- Calculate matrix $B$ based on matrix $A$ and $u(t)$.

In order to illustrate the previous procedure, the transient response to a duty cycle from 10 to 50% is considered, as shown in Fig. 11. First, $\Delta_{otr}$ is obtained by Eq. (5).

$$\Delta_{otr} = \omega_r^2 - \omega_r^1 = 56.18 \frac{\text{rad}}{s} - 98.08 \frac{\text{rad}}{s} = -39.9 \frac{\text{rad}}{s}$$

(5)

This result is multiplied by one time constant, that is, when system obtains 63.2% of its steady state. That results in:

$$0.632 \left(-39.9 \frac{\text{rad}}{s}\right) = -24.857 \frac{\text{rad}}{s}$$
If $\omega_{r1}$ is added to this result, correspondent rotor speed to 63.3% of steady state can be calculated.

$$\Delta\omega_{63.3\%} = 98.08 \frac{\text{rad}}{s} - 24.857 \frac{\text{rad}}{s} = 71.233 \frac{\text{rad}}{s}$$

\(\tau\) is determined considering \(\Delta r\) as follows:

$$\tau = \Delta t = t_{63.2\%} - t_1 = 42.205s - 41.648s = 0.577s$$

Then, matrix \(A\) can be obtained by using Eq. (6).

$$A = -\frac{1}{\tau}$$

$$A = -\frac{1}{0.557} = -1.794$$

Finally, \(B\) can be calculated considering the steady-state in Eq. (4) as follows:

$$B = -\frac{Ax(t)}{u(t)} = -\frac{(-1.794)(56.18)}{50\%} = 2.016$$

This procedure is repeated for every one of the duty cycles applied. An average of each matrix (\(A\) and \(B\)) is obtained; finally, a PI discrete controller was implemented to optimize wind power extraction. Four input channels were used:

- Channel Al 0: rotor wind speed signal $\omega_r$. 

Fig. 12. Discrete PI control in RTAI.

- Channel Al 1: signal for set point of rotor wind speed.
- Channel Al 2: voltage sensor signal.
- Channel Al 3: current sensor signal.

On the other hand, an output channel was required:

- Channel AO 0: analog signal 0–5 V correspondent to duty cycle.

Five signals were stored in this work by using FIFO blocks of RTAI.

- FIFO-1: duty cycle.
- FIFO-2: set point of rotor wind speed.
- FIFO-3: rotor wind speed.
- FIFO-4: output voltage of MBC.
- FIFO-5: output current of MBC.

Implementation of control in discrete-time was obtained by using a 1/z RTAI block. Scilab contains useful gain, mathematical, and addition blocks used for programming the discrete PI controller. The upper sum block shown in Fig. 12 represents the integral variable in discrete time, this is \( X_I(k+1) \), the integral variable in \((k + 1)\) requires itself in time \((k)\), the state variable \((x(k))\), and the reference signal or set-point \((r)\). Adding these terms gives Eq. (7):

\[
X_I(k+1) = X_I(k) + x(k) - r
\]  

Similarly, the lower sum block represents the input of the closed-loop system. By obtaining the integral variable and multiplying it by the gain \( G_I \), the term \( x_I(k)G_I \) is calculated. Taking the state variable and multiplying it by \( G_x \), the term \( x(k)G_x \) is obtained. When adding them in the block
Eq. (8) is obtained:

\[ u(k) = x_I(k)G_I + x(k)G_x \]  

In this way, we have both equations needed for the PI controller in discrete time, and the analog input/output channels necessary for reading and recording real variables of interest in the system.

Voltage and current measurements were made at the output of MBC. Hall HAS 100-S sensors and the IC ISO-122 were used for measuring the current and voltage, respectively. A purely resistive \( Y \) configuration load was connected to the three-phase inverter, as shown in Fig. 13, while Fig. 14 presents the whole system’s connection diagram.
**Results**

Dataset to validate this project is freely accessible at [http://dx.doi.org/10.17632/363d24mcb6.1](http://dx.doi.org/10.17632/363d24mcb6.1). In it, a MATLAB program can be downloaded along with five *.dat files which are read by *.m file to generate the correspondent graphics to:

- Experimental power coefficient as a function of tip speed ratio in closed loop operation
- Experimental tip speed ratio in closed-loop operation
- Experimental turbine-generator rotational speed in closed-loop operation
- Experimental duty cycle of the 2N-MBC
- Experimental turbine power and electrical power at the 2N-MBC terminals in closed-loop operation

The most important result is shown in Fig. 15. The red line represents the curve of power coefficient generated by Eq. (2), and the blue asterisks represent the closed-loop transient behavior before wind speed changes. The PI controller's action is remarkable to modify the duty cycle for MBC for changing rotor speed so that maximum $C_p$ can be reached, as is shown in the green square depicted in Fig. 15.

**Conclusions**

The methodology described in this paper provides to those interested in wind energy, an alternative to simplify a WECS using a unified model based on several dynamic tests. The results for the maximum power point tracking show the reliability of the model. The PI control designed and implemented in RTAI can maintain the state variable, that is, the rotor speed at a value dependent on the incident wind speed with which a constant tip speed ratio ($\lambda$) is associated. The real-time control action on the duty cycle injected into the MBC causes the power coefficient to always be the maximum for any incident wind speed. The maximum power transfer is satisfactorily fulfilled for different wind speed values, achieving effective control from synthesizing a much simpler model that works experimentally.

**Declaration of Competing Interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.
Acknowledgments

To Tecnológico Nacional de México (Instituto Tecnológico de Ciudad Madero) for make available laboratories, materials, instruments and all support related to this work.

References

[1] Global Wind Energy Council 2019. Global wind report, annual market update, GWEC
[2] R. Mittal, K.S. Sandhu, D.K. Jain, An overview of some important issues related to wind energy conversion systems (WECS), Int. J. Environ. Sci. Dev. 1 (4) (2010) 351–363.
[3] Y. Venkata, W. Bin, C.S. Paresh, K. Samir, M. Mehdi, High power wind energy conversion systems: state-of-the-art and emerging technologies, Proc. IEEE 103 (5) (2015) 740–778.
[4] S.Y.A. De-Falani, M.O.A. Gonzalez, F.M. Berreto, J.C. Toledo, A.L.V Torkomian, Trends in the technological development of wind energy generation, Int. J. Technol. Manag. Sustain. Dev. 19 (1) (2020) 43–68.
[5] L. Hu, F. Xue, Z. Qin, J. Shi, W. Qiao, W. Yang, et al., Sliding mode extremum seeking control based on improved invasive weed optimization for MPPT in wind energy conversion system, Appl. Energy 248 (2019) 567–575.
[6] H.H.H Mousa, A. Yousef, E.E.M. Mohamed, Variable step size P&O MPPT algorithm for optimal power extraction of multi-phase PMSG based wind generation system, Int. J. Electr. Power Energy Syst. 108 (2019) 218–231.
[7] A. Asri, Y. Mihoub, S. Hassaine, P. Logerais, T. Allaoui, Intelligent maximum power tracking control of a PMSG wind energy conversion system, Asian J. Control 21 (4) (2019) 1980–1990.
[8] P. Sai Manoj, A. Vijayakumari, S.K. Kottayil, Development of a comprehensive MPPT for grid connected wind turbine driven PMSG, Wind Energy 22 (2019) 732–744, doi:10.1002/we.2318.
[9] Dipartimento di scienze e tecnologie aerospaziali del Politecnico di Milano, real time application interface, 2018. URL: https://www.rtai.org/.
[10] R. Bucher, S. Mannori, T. Netter, Rtai-lab tutorial: scilab, comed, and real-time control”, https://www.rtai.org/rtailab/rtai-lab-tutorial.pdf [retrieved 12, 2009.]