Improvement of Turbine Inertia Compensation in Wind Turbine Emulators using Kalman Filter

Mansour Rafiee[1,*], Mahdi Pourgholi[1], Afshin Shahmohamadi[1]

[1] Electrical Engineering Department, Shahid Beheshty University, Tehran, Iran

ARTICLEINFO

ABSTRACT

In this paper, a 3.1kW DC motor is used to simulate the static and dynamic behavior of a horizontal axis wind turbine. The effect of the difference between the wind turbine's inertia and DC motor on the dynamic behavior of the system, torque oscillation due to the effect of the tower shadow and wind shear, and the effect of rotary losses in the mechanical torque of the DC motor are considered. The torque of the DC motor is controlled by the closed-loop PID controller. This closed loop has two feedbacks for the speed and current of the DC motor. The turbine calculations are performed in Matlab/Simulink and the reference signal of the turbine's torque is sent to the chopper using an interface card. Two methods are proposed to estimate the acceleration of the DC motor to improve the compensation of the turbine inertia effect: the Kalman Filter, and the second-order low-pass filter. Adjustable parameters of proposed methods are optimized using the particle swarm optimization algorithm (PSO). A 2.2 kW synchronous generator is coupled with the wind turbine emulator and feeds a constant load. To show the effectiveness of the proposed methods the results are reported. The results indicate that the Kalman filter has a better performance in the compensation of the turbine inertia effect.

1. Introduction

In recent years, the use of wind energy conversion systems has grown significantly and many studies have been done on wind turbines. The use of wind turbine emulators has a beneficial role in the study of wind energy conversion systems. These emulators are used to model the static and dynamic behavior of a wind energy conversion system in a controlled condition without requiring any source of wind power or wind turbine. In addition, these emulators can be used as a training tool to learn how to operate or control a wind turbine.

In the case of the simulation of wind turbine behavior, there have been many studies. In [1], for the first time, a direct current motor was used to simulate a wind turbine behavior. In the following years, [2] and [3] also used DC motors. In [4], an induction motor for the first time was used. Also, in [5] the induction motor was again used, and it can be said that [4] and [5] complement each other. In [6] the synchronous motor is used for simulating wind turbine behavior for the first time. In reference [7] for the first time, a component was added to the torque of DCM for compensating the inertia differences between the wind turbine and electric motor. In [8–12], large turbine inertia compensation component was considered in different ways. These papers filter the speed derivative term with a low-pass filter. In [13] cascaded low-pass and high-pass filters are used to compute the shaft speed

* Corresponding author
  E-mail address: m_rafiee@sbu.ac.ir
  https://www.orcid.org/0000-0002-6973-6062

derivative. In [14] compare the frequency response of real wind turbine and wind turbine emulator.

The effect of tower shadow and wind shear will cause oscillation in the torque of the turbines. These oscillations are considered in [15] for the first time for a wind turbine emulator. In the following years, in [7, 8, 16, 17] This effect was considered in different methods. Regarding the turbine calculations and computing the torque reference, in [18], [19] LabVIEW with a data acquisition board (DAQ) is used, in [3, 7, 14, 15, 20,21] DSP is used, in [9,11]Matlab/Simulink with a DAQ is used, in [6,23] FPGA is used. In [24] Instead of a wind turbine coupled with a synchronous generator, a back-to-back converter has been used for simulation. In [25] a DFIG-based system was implemented.

This paper aims to present a better estimate of the shaft speed derivative with a Kalman Filter. Also, the estimation of DC motor output torque is done with good accuracy. Operation of the Kalman Filter compared with an optimized second-order low-pass filter. In this work, the dynamic and static behavior of a wind turbine is simulated with a DC motor. The DC motor is supplied by a DC chopper. Voltage reference obtained from a closed loop PID controller. The input of the controller is the difference between the torque reference (compute from wind turbine equations) and output torque of the DC motor (estimated with considering rotary losses of the DC motor). Voltage reference converts to real signal and sends to chopper using a DAQ card.

2. Mechanical Model of Wind Turbine Emulator

A wind turbine emulator should be able to simulate the static and dynamic behavior of a real wind turbine.

2.1. Steady-state behavior of wind turbine

Extracted mechanical power by wind turbine from wind is given by well-known expression as follows [26]:

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

where R is the length of blades, V is the wind velocity, $C_p(\lambda,\beta)$ is the power coefficient and ρ is the air density. The C_p describes that fraction of the power in the wind that may be converted by the turbine into mechanical work. The C_p depends on the aerodynamic shape of the blades, the pitch angle (β), and the tip speed ratio (λ). The power coefficient can be modeled by polynomial or exponential functions. In this paper the following expression is used [13]:

$$C_p(\lambda,\beta) = C_1\left(C_2\frac{1}{\lambda_i} - C_3\beta - C_4\right)e^{\left(-\frac{C_5}{\lambda_i}\right)} + C_6\lambda \tag{2}$$

where

$$\lambda_i = \left(\frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^3 + 1}\right)^{-1} \tag{3}$$

$$\lambda = \frac{\omega_t R}{V} \tag{4}$$

ω_t is rotational speed of the turbine, C_1, C_2, C_3, C_4, C_5 and C_6 are constants depends on aerodynamic shape of the blades. The aerodynamic torque of a wind turbine is obtained from the following expression:

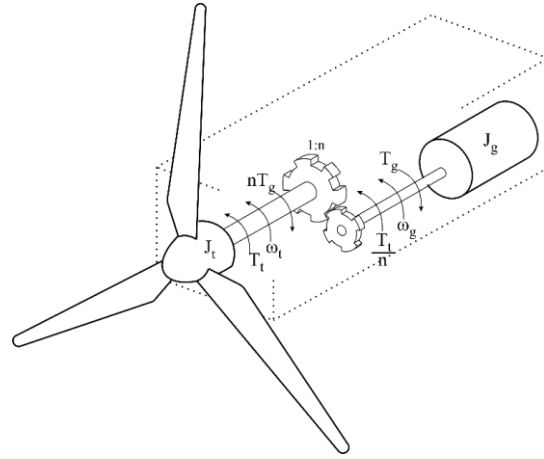$$T_{ae} = P_t/\omega_t \tag{5}$$

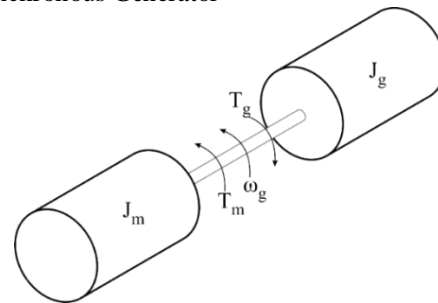**Fig. 1**. Mechanical model of wind turbine coupled with Synchronous Generator

Fig. 2. Mechanical model of DC motor coupled with Synchronous Generator

2.2. Large inertia effect of wind turbine

Fig. 1 shows the mechanical model of a wind turbine coupled with a synchronous generator (SG). According to this figure, the mechanical equation expressed on the generator side is

$$\frac{T_t}{n} - T_g = \left(\frac{J_t}{n^2} + J_g\right)\frac{d\omega_g}{dt} \tag{6}$$

where the ratio of the gearbox is $1:n$, J_t is the turbine inertia, J_g is the generator inertia, T_t is the turbine output torque, T_g is the generator torque and ω_g is the generator side speed.

The mechanical model of the DC motor coupled with a synchronous generator is shown in Fig. 2. According to this figure J_m is the inertia of the DC motor and T_m is the output torque of the DC motor. The following equation is written according to Fig. 2:

$$T_m - T_g = \left(J_m + J_g\right)\frac{d\omega_g}{dt} \tag{7}$$

By subtraction of Eq. (6) from Eq. (7), the reference torque for the DC motor is obtained:

$$T_m = \frac{T_t}{n} + \left(J_m - \frac{J_t}{n^2}\right)\frac{d\omega_g}{dt} \tag{8}$$

The first term relates to the turbine output torque in steady states and the second term, which includes the generator speed derivative, relates to the difference in the inertia of the DC motor and wind turbine. This term is

produced in transient states and it must be produced by the motor to behave like a wind turbine.

Eq.(5) shows the aerodynamic torque of the wind turbine (T_{ae}) but in Eq. (8) output torque of the wind turbine is used, the relation of these parameters is shown in Eq. (9).

$$T_t = T_{ae} + T_{loss} \qquad (9)$$

where T_{loss} is the torque of mechanical losses of the wind turbine.

2.3. Torque oscillation due to towers shadow and wind shear

The blocking of the airflow by the tower results in regions of reduced wind velocity both upwind and downwind of the tower. This reduction is more severe for tubular towers than for lattice towers and, in the case of tubular towers, is larger on the downwind side because of flow separation. The wind velocity considering the tower shadow effect in the coordinates (*x,y*) is given in Eq. (10) (according to reference frames shown in Fig. 3.

$$V_{ts}(x, y) = m V_H a^2 \frac{y^2 - x^2}{(x^2 + y^2)^2} \qquad (10)$$

where a is the radius of the tower, V_H is wind velocity in hub height and m in according to [26] obtained from Eq. (11).

$$m = 1 + \frac{\alpha(\alpha - 1)}{8} \left(\frac{R}{H}\right)^2 \qquad (11)$$

where H is hub height and α is the experimental constant.
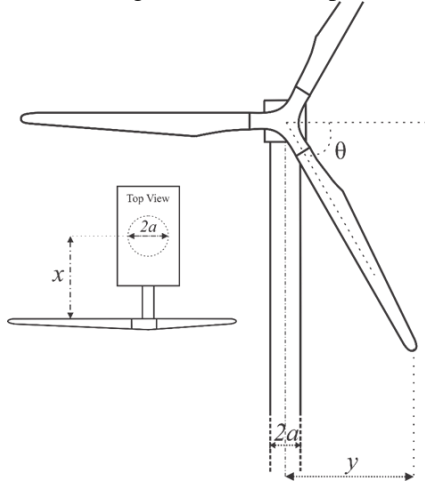

Fig. 3. Dimensions used in the tower shadow formula

Wind velocity generally increases with height and this variation is termed wind shear. Torque oscillation, and therefore power oscillation, are observed due to the periodic variations of wind velocity seen at different heights. The common wind shear model is shown in the following equation:

$$V_{ws}(z) = V_H \left(\frac{z}{H}\right)^\alpha \qquad (12)$$

where α is the experimental constant and z is the height from the ground. According to [27], if Eq. (10) and Eq. (12) are written in polar coordinates, wind velocity is given by Eq. (13).

$$V(t, r, \theta) \approx V_H(t)[1 + V_{ws}(r, \theta) + V_{ts}(r, \theta, x)] \qquad (13)$$

By following the procedure of simplifying the equations from [23], Eq. (14) is obtained.

$$\overline{T_{ae}}(t, \theta) = 1 + \frac{2}{mV_H} \left[V_{eq_{ts}} + V_{eq_{ws}} + (1 - m)V_H \right] \qquad (14)$$

where $\overline{T_{ae}}(t, \theta)$ is per-unit aerodynamic torque of wind turbine, $V_{eq_{ts}}$ and $V_{eq_{ws}}$ is equivalent wind velocity of tower shadow and wind shear.

3. Improvement of turbine inertia compensation

Two methods are proposed to improve the compensation of turbine inertia. In the first method, we optimized a second-order low-pass filter with the PSO algorithm. In the second method, we used the Kalman Filter which this method has a better operation. A similar algorithm has been used to optimize the low pass filter and the Kalman Filter. The number of Iterations and particles in this algorithm is 30. The cost function is the integral of the square error of desired signal and the real signal. In the following section, we describe and compare these proposed methods.

3.1. First method: second-order low-pass filter

According to Eq. (8), we should have the shaft speed derivative of the electrical motor to compute the compensation term of reference torque. The issue that is raised here is how to set the gain and corner frequency of low-pass filters. If the corner frequency is narrow, the system will have a long delay in transient states. Also, if the corner frequency is wide, the system will become unstable. Therefore, we have to set the low pass filter. In this work, the filter is optimized with the PSO algorithm.

The motor's position signal is measured by a sensor and we have to get speed and acceleration signals from the position signal. By twice filtering and deriving from the position signal, the acceleration signal is obtained. The form of transfer functions of position signal to speed signal and speed signal to acceleration signal is shown in the following equations:

$$G_{ps}(s) = \frac{k_1 s}{(s + s_1)(s + s_2)} \qquad (15)$$

$$G_{sa}(s) = \frac{k_2 s}{(s + s_3)(s + s_4)} \qquad (16)$$

where k_1, k_2, s_1, s_2, s_3 and s_4 are constants that are set with the PSO algorithm (the obtained values are shown in Table1), $G_{ps}(s)$ is a transfer function of position to speed signal and $G_{sa}(s)$ is a transfer function of speed to acceleration signal.

3.2. Second method: Kalman Filter

A Kalman Filter will be used for the estimation of speed and acceleration signals using optocoupler measurements. The Kalman Filter is an optimal state estimator for linear dynamical systems of the form:

$$x_{k+1} = \varphi_k x_k + w_k \qquad (17)$$

$$Z_k = H_k x_k + v_k \qquad (18)$$

where x_k is n×1 state vector at t_k, φ_k is $n \times n$ state transition matrix at t_k, Z_k is $m \times 1$ measurement vector at t_k, H_k is m×n measurement matrix at t_k, w_k zero-mean white Gaussian process noise vector at t_k, v_k zero-mean white Gaussian measurement noise vector at t_k. By following the proof of equations from [27], we get the following equations:

Measurement update:

$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1} \tag{19}$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k[Z_k - H_k \hat{x}_k^-] \tag{20}$$

$$P_k^+ = (I_n - K_k H_k)P_k^- \tag{21}$$

Time update:

$$\hat{x}_{k+1}^- = \varphi_k \hat{x}_k^+ \tag{22}$$

$$P_{k+1}^- = \varphi_k P_k^+ \varphi_k^T + Q_k \tag{23}$$

where K_k is Kalman gain, P_k^- is error covariance matrix (before the measurement at t_k), P_k^+ is error covariance matrix (after the measurement at t_k), R_k is measurement noise covariance matrix, Q_k is process noise covariance matrix, $\hat{x}_k^-$ is a prior estimate of x_k (before the measurement at t_k), $\hat{x}_k^+$ is a posterior estimate of x_k (after the measurement at t_k). Using equations (19) to (23), the Kalman Filter is implemented. Using the measured position signal of the motor, the Kalman Filter can estimate the angular speed and acceleration of the system. The value of parameters are considered as follows:

$$\varphi_k = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \tag{24}$$

$$H_k = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \tag{25}$$

$$x_k = \begin{bmatrix} \theta \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} \theta \\ \omega \\ \alpha \end{bmatrix} \tag{26}$$

where T is the sampling time of the Kalman Filter. The adjustable parameters of the Kalman Filter are Q_k and R_k.

$$Q_k = \begin{bmatrix} Q_{11} & 0 & 0 \\ 0 & Q_{22} & 0 \\ 0 & 0 & Q_{33} \end{bmatrix} \tag{27}$$

$$R_k = r_\theta \tag{28}$$

Different methods have been used to set Q_k and R_k [28]. However in this paper, Q_k and R_k are set using the PSO algorithm. This algorithm adjust values of Q_{11}, Q_{22}, Q_{33} and r_θ in optimal point.

The values of these parameters are tabulated in Table 2 in the appendix. In the PSO algorithm, 50 iterations and 50 particles are considered. The cost function of this algorithm was integral of square error for the acceleration signal.

4. Implementation of wind turbine emulator

The hardware side of the wind turbine emulator system contains:

- 3.1kW separately excited DC motor
- 2.2kW synchronous generator
- DC chopper
- DAQ card
- Hall effect sensor and Position sensor
- 3-phase AC voltage source
- 3-phase RL load

In Fig. 4 implemented system is shown. Torque reference is computed by Matlab/Simulink. The input signal of the PID controller is the error signal of torque reference and estimated torque of the DC motor. The output signal of the controller is the voltage reference of the DC motor that is sent to the chopper by a DAQ card. The DAQ card provides possibility of communication between the hardware and software sides.

The current and speed of the DC motor are feedbacks of the system that measured by sensors. By using the current signal, the output torque of the DC motor is estimated. Using the well-known equation for DC motor torque $T_e = L_{af} I_f I_a$, the electromagnetic torque of the motor is calculated by considering the effect of the armature reaction and rotary losses. The losses of DC motor (rotary losses) are considered as a factor of rotational speed. With this estimation method, the output torque of the motor and the produced torque by the turbine's model are equal with good accuracy. Parameters of the wind turbine, DC motor, and synchronous generator are tabulated in Tables 3, 4, and 5.
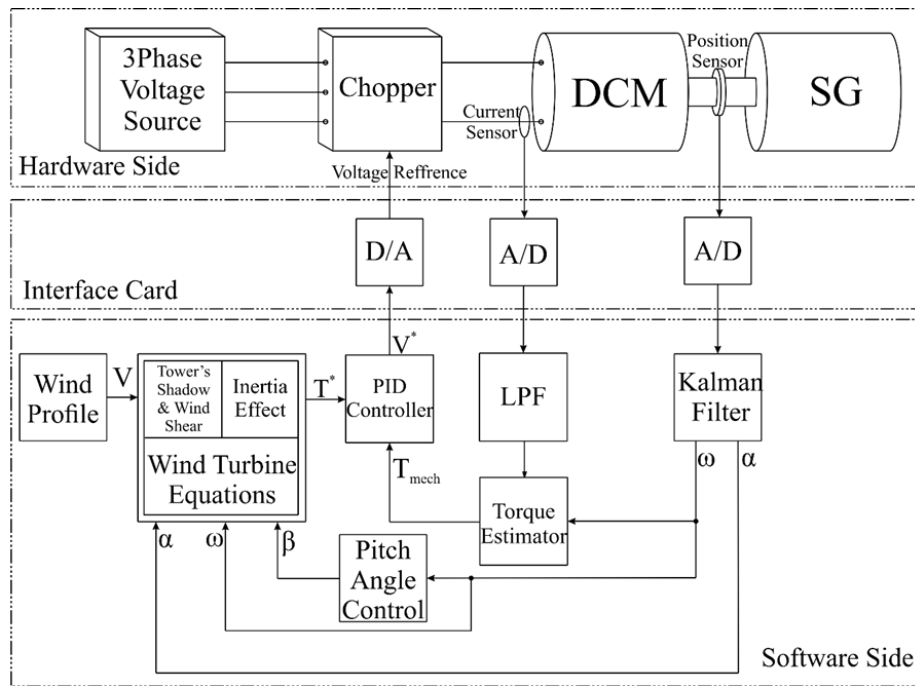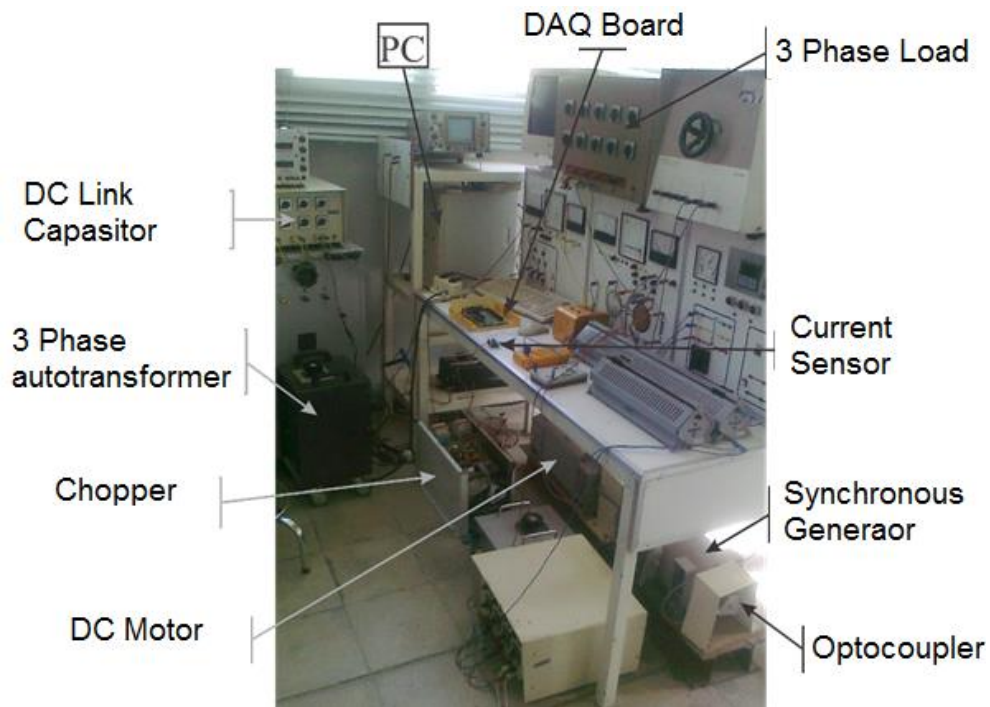
Fig. 4. Scheme of the implemented system
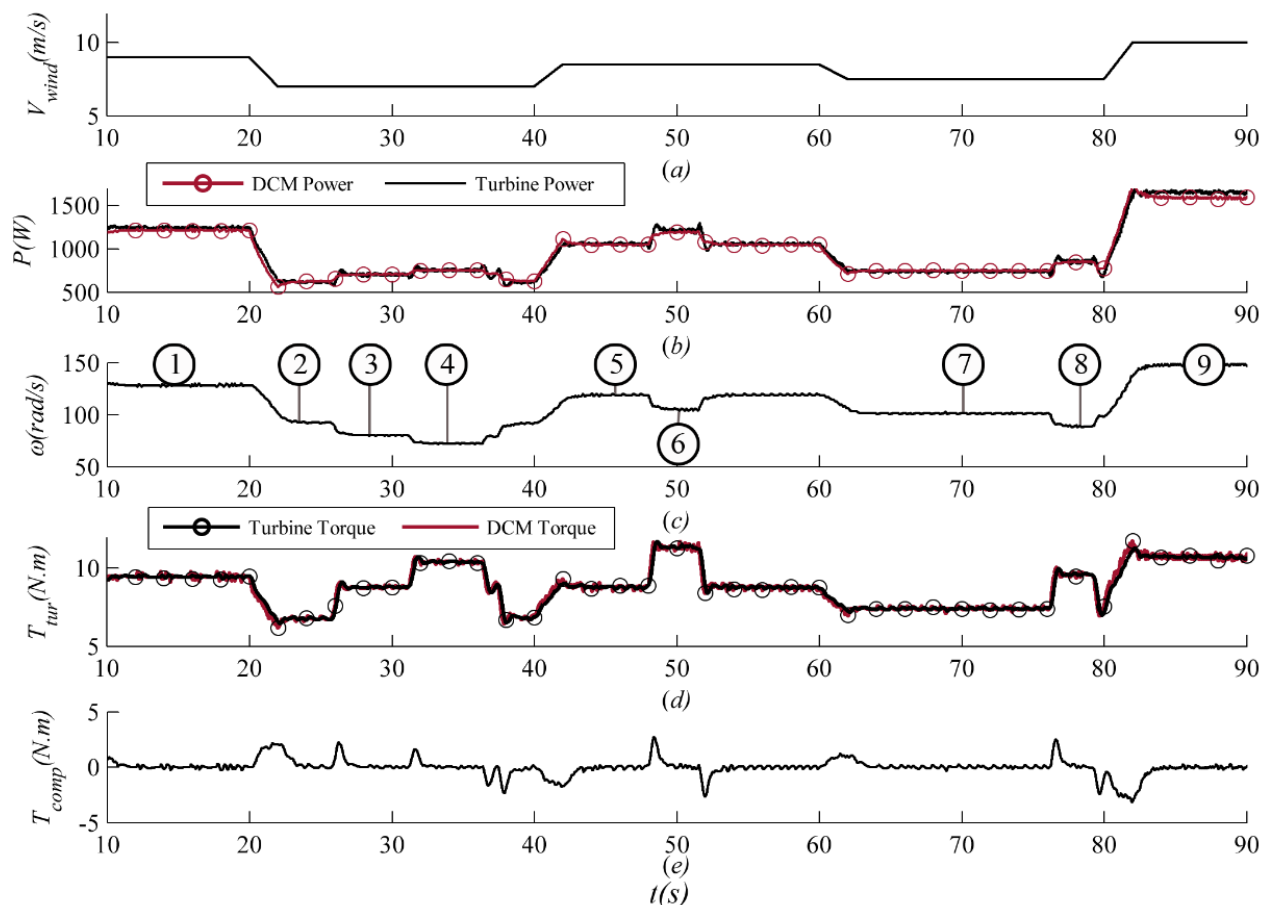

Fig. 5. Experimental Setup

Fig. 6. Results of wind turbine emulator under wind velocity and generator's load changes**. (a)** Wind velocity profile, **(b)** output power of DCM and Wind Turbine, **(c)** rotational speed of DCM, **(d)** output torque of DCM and Wind Turbine, **(e)** compensation torque of inertia.

5. Experimental result

According to the presented model in Fig. 4, the wind turbine emulator has been implemented and the behavior of this emulator has been investigated under different conditions. Fig.5 shows the experimental setup. In Fig. 6, the simulation results are investigated while the system is under changes in generator load and wind velocity. Fig. 6(a) shows the wind velocity profile. This wind profile is optional and is used to give a better vision of the turbine's operating points. In 10 seconds, the wind velocity is 9 m/s, and at times 30, 50, z70, and 90 seconds, the values are 7, 8.5, 7.5, and 10 m/s, respectively. Fig. 6(b) shows the wind turbine power and output power of the DC motor.

The output power of the motor is obtained by subtracting the copper and rotary losses of the motor from its input power. It can be seen that these two powers have very little difference. For example, in the 50s, the difference between these two curves is approximately 40 watts or 4% and this is the result of a good estimation of DC motor output torque. Power changes in this figure because of the changes in generator load and wind velocity. These changes are explained in Fig. 7. Fig. 6(c) shows the DC motor speed. By dividing this speed by the gearbox ratio, the speed of the wind turbine model will be obtained. The numbers in the figure related to Fig. 7. In Fig. 6(d) the output torque of the turbine model and output torque of the DC motor are shown. The controller's task is to set the motor output torque equal to the output torque

of large turbine inertia. This component is non-zero in transient states and steady states is almost equal to zero. torque of the turbine. According to this figure, the controller works well. Fig. 6(e) shows compensation

Fig.7 shows the power curves of the purposed wind turbine and power-speed curves of the implemented system. The figure is shown the operating points of Fig. 5. The system operating point in the 20th second in point 1, with a change in wind velocity from 9 to 7 m/s, reaching point 2. Then, by stepping up the generator's load in the 26th second, the operating point moves on the same power curve and reaches point 3. With stepping up the generator's load in 31.5th seconds, the operating point is located at point 4. At 37th and 38th seconds, the generator's load stepped down to its initial value, with this action the operating point of the system returned to point 2. In the 40th second, the wind velocity changed from 7 to 8.5 m/s, and the operating point was located at point 5. In the 47th and 52nd seconds, the generator's load stepped up and then stepped down to the previous value. During these seconds operating point of the system is located in point 6 and then returned to point 5. In the 60th second, the wind velocity changed from 8.5 to 7.5 m/s, and the operating point is located at point 7. In the 77th second, the generator's load stepped up, and in the 79th second it returned to its previous value, and finally in the 80th-second wind velocity changed from 7.5 to 10 m/s which resulted in point 9. It is can be seen in all the operating points that the output power of the motor was following

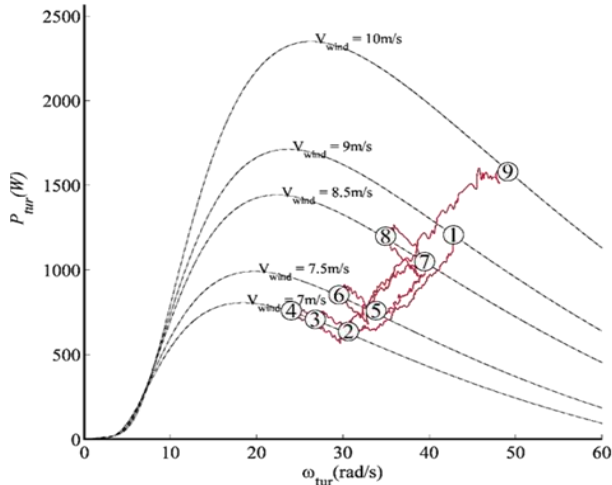wind turbine curves, which indicates the desired performance of the system.

Fig. 7. Operating points of wind turbine emulator under wind velocity and generator load changes
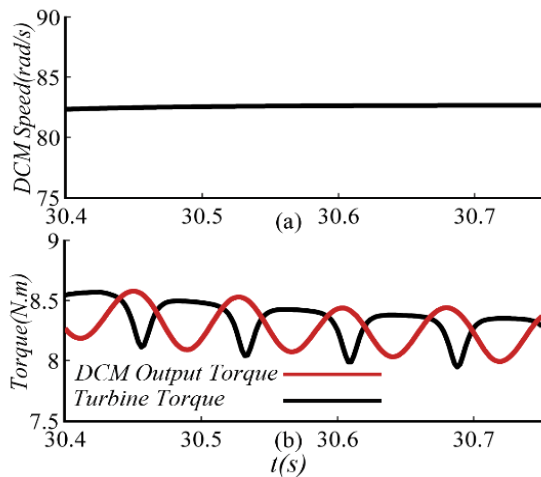

Fig. 8. Speed and output torque of the motor by considering the effect of tower shadow. **(a)** DC motor speed **(b)** Torque of DC motor and Wind Turbine
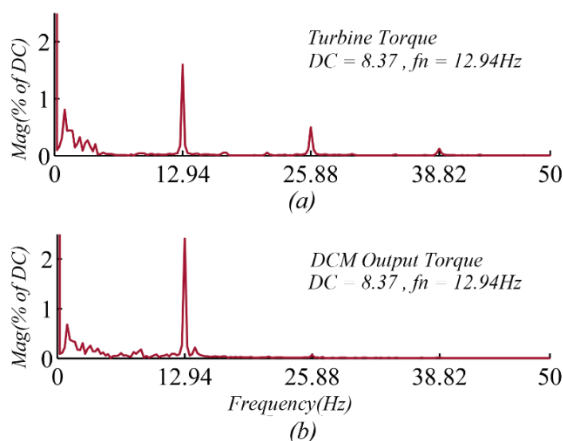

Fig. 9. Frequency analysis of turbine and DC motors torques by considering the effect of tower shadow. **(a)** Frequency analysis of Wind Turbine's Torque, **(b)** frequency analysis of DC Motor's Torque.

In Fig. 8, the behavior of the wind turbine emulator is shown by considering the effect of tower shadow. In this figure, the wind velocity has been 7 m/s. In Fig. 8(a)

speed of the DC motor and in Fig. 8(b) output torque of the DC motor vs produced torque of the wind turbine is shown. The tower shadow has caused oscillations in the output torque of the wind turbine. Fig.9 shows Fourier analysis for the curves of Fig. 8(b).

Fig. 9(a) is related to wind turbine torque and Fig. 9(b) is related to DC motor torque. Both curves have the same offset values (8.37N.m) and their first harmonics frequency is 12.94Hz. The first harmonic values of signals are approximately equal but due to the low-pass filter in the motor current feedback signal, high-frequency harmonics have not appeared in the current and electrical torque of the DC motor.

On the other hand, because of the inertia in the motor and generator coupling system, the high-frequency components of the torque oscillations caused by the towers shadow, will not appear in the mechanical torque of the motor. For more accurate results, the mechanical torque of the generator shaft should be measured.

The speed of the DC motor, as shown in Fig. 8(a), is approximately equal to 81.5 rad/s. By dividing the DC motor speed by the gearbox ratio, the speed of the turbine is obtained, which is 27.17 rad/s. the oscillations frequency due to the tower shadow effect is three times bigger than the turbine rotating frequency. According to this statement, the torque oscillations frequency is 12.96Hz (3×27.17 rad/s $= 81.5$rad/s $= 12.96$Hz), which corresponds to the number obtained for the first harmonic of oscillations frequency (12.94 Hz).

Regarding the improvement of turbine inertia compensation, the Kalman Filter and second-order optimized low-pass filter are compared and the results are shown in Fig. 10. In this figure the wind velocity profile (Fig. 10(a)) and the compensated torque (Fig. 10(b)) is shown. Fig. 10(b) consists of three curves (the desired signal, Kalman Filter's signal, and optimized low-pass filter's signal). In seconds 20, 40, 60, and 80, the wind velocity changed and in seconds 28, 32, 37, 46, 68, 72, and 92 generator's load changed. When the compensation torque is positive, the generator load is stepped up and when the compensation torque is negative, the generator load is stepped down.

In Fig. 10(c), (d), and (e), the curves of Fig. 10(b) are magnified. It is clearly seen in these figures that the Kalman Filter performs better than the optimized low-pass filter. Over 80 seconds, the value of the error square integral of the turbine inertia compensation component in the case of the Kalman Filter is equal to 11.88 and in the case of an optimized low-pass filter, it is 23.5. The Kalman Filter has higher accuracy and lower latency compared to the optimized low-pass filter. In Table 6 summary of the comparison between the two methods is tabulated. In this table Values of error square integration of torque compensation component are denoted. According to this comparison, The Kalman Filter has higher accuracy and lower latency compared to the optimized low-pass filter.

A graphical user interface is designed for easier user interaction with the wind turbine simulator. In this user interface, the simulation results can be seen and it is possible to change the parameters of the wind turbine. Parameters such as rotor radius, turbine inertia, gearbox coefficient, and also coefficients C_1 to C_6 can be changed

manually. The proposed graphical user interface can be seen in Fig.11.

6. Conclusion

This paper describes how to build a wind turbine emulator with high accuracy. The machines used in this paper were a DC motor and a synchronous generator.

The computation of wind turbine torque was done on a PC. With a DAQ card, the software side and hardware side were interfaced. The input voltage of the DC motor was supplied by a DC chopper. The PC computes the reference voltage of DC copper.

By controlling the output torque of a DC motor, the behavior of a wind turbine was emulated. The output torque of the DC motor was estimated with good accuracy

and the turbine's output power with good approximation appeared on the DC motor output.

Two methods were suggested for the improvement of turbine inertia compensation. A comparison was done between the two methods and the result was that the Kalman Filter performance was better than the optimized low-pass filter.

The effect of tower shadow and wind shear was considered. For a better investigation of the DC motor output torque, a Fourier analysis was done. The first harmonics of the torques are equal. The accuracy of the tower shadow effect was better than in previous works.
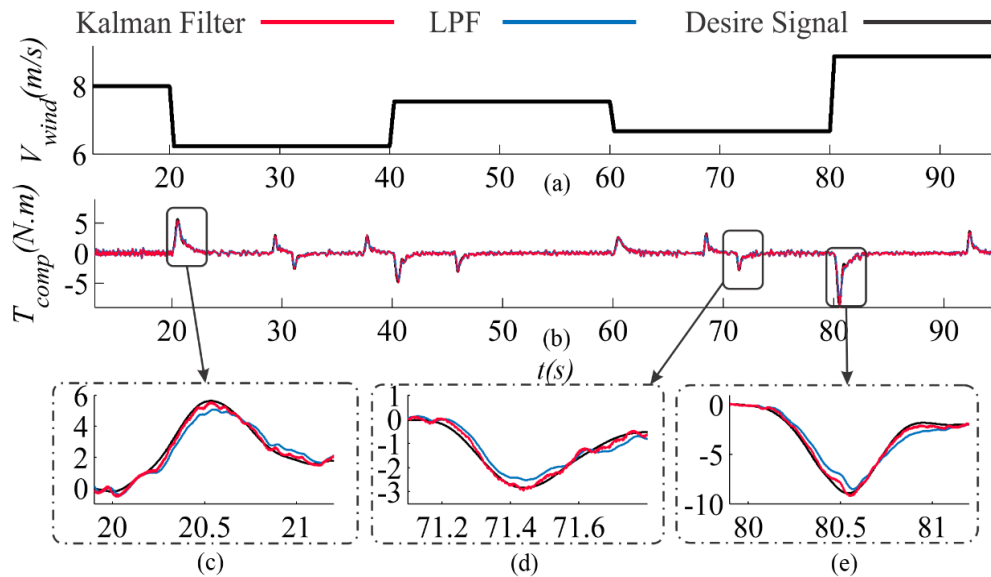

Fig. 10. Improvement of turbine inertia compensation component. (a) Wind velocity profile, (b-e) compensation torque of inertia.
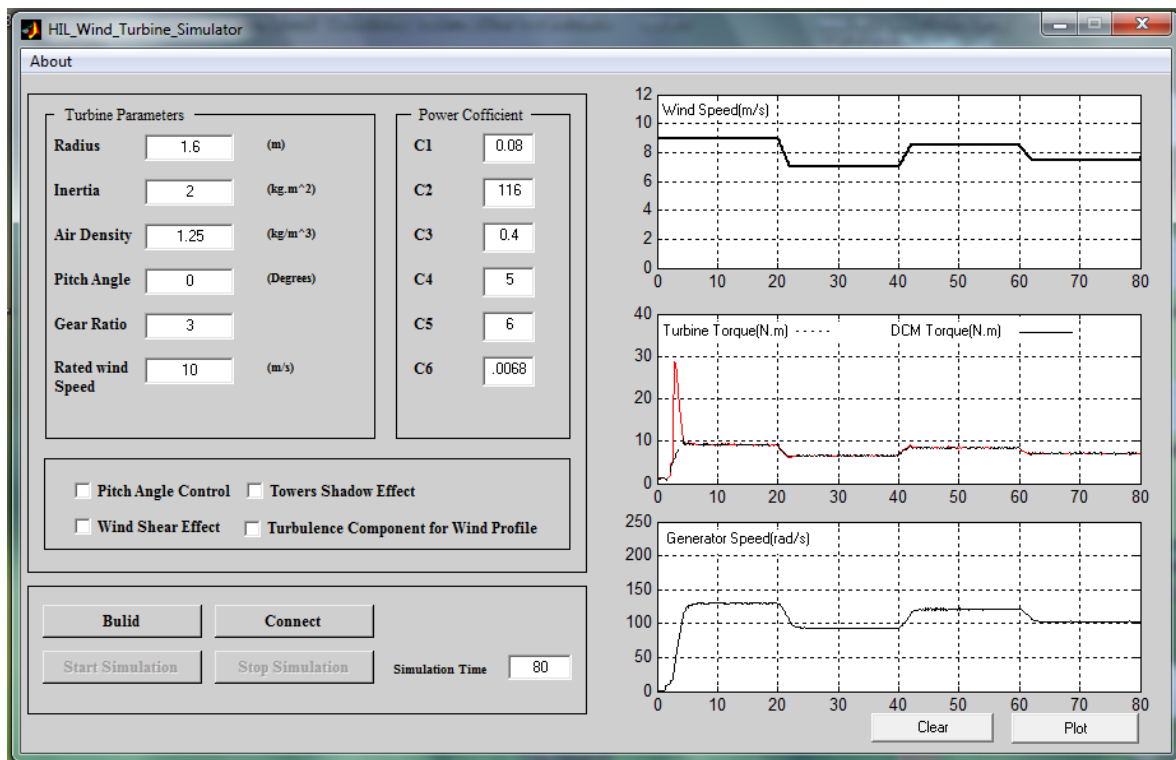

Fig. 11. Graphical user interface in the hardware simulator

7. Appendix

Table.1. Parameters of G_{ps} and G_{sa}

Parameters	Value
k_1	183.3
k_2	456.4
s_1	13.5
s_2	13.3
s_3	21.2
s_4	21.5

Table.2. Parameters of Q_k and R_k

Parameters	Value
Q_{11}	0.001
Q_{22}	1.54
Q_{33}	76.9
r_θ	15

Table.3. Wind Turbine Parameters

Parameters	Value	Unit
Rated power	2	kW
Radius	1.6	m
Gearbox ratio	3	-
C_1	0.08	-
C_2	116	-
C_3	4	-
C_4	5	-
C_5	16.5	-
C_6	0.0068	-

Table.4. DC Motor Parameters

Parameters	Value	Unit
Rated power	3.1	kW
Rated voltage	230	V
Rated armature current	14	A
Field current	0.7	A
Rated speed	1500	RPM

Table.5. Synchronous Generator Parameters

Parameters	Value	Unit
Rated power	2.2	kW
Rated voltage	380	V
Rated current	3.2	A
Poles	4	-
Rated speed	1500	RPM
Field current	1.5	A

Table.6. Comparison between two methods

Method	Integral of square speed error	Integral of square compensated torque error
Kalman Filter	21.92	11.88
Optimized low-pass filter	61.20	23.50

References

[1] J.H.R. Enslin, J.D. Van Wyk, A study of a wind power converter with micro-computer based maximal power control utilizing an over-synchronous electronic scherbius cascade, Renew. Energy. 2 (1992) 551–562.

[2] F.A. Farret, R. Gules, J. Marian, Micro-turbine Simulator Based on Speed and Torque of a DC motor to Drive Actually Loaded Generators, Devices, Circuits Syst. (1995) 89–93.

[3] P.E. Battaiotto, R.J. Mantz, P.F. Puleston, A wind turbine emulator based on a dual DSP processor system, Control Eng. Pract. 4 (1996) 1261–1266.

[4] L. Chang, R. Doraiswarmi, T. Boutot, H. Kojabadi, Development of a Wind Turbine Simulator for Wind Energy Conversion Systems, Can. Conf. Electr. Comput. Eng. IEEE CCECE2000. (2000) 550–554.

[5] H.M. Kojabadi, L. Chang, T. Boutot, Development of a Novel Wind Turbine Simulator for Wind Energy Conversion Systems Using an Inverter-Controlled Induction Motor, IEEE Trans. Energy Convers. 19 (2004) 547–552.

[6] D.S.L. Dolan, P.W. Lehn, Real-Time Wind Turbine Emulator Suitable for Power Quality and Dynamic Control Studies, Int. Conf. Power Syst. Transients. (2005) 1–6.

[7] S. Song, B. Jeong, H. Lee, J. Kim, Emulation of Output Characteristics of Rotor Blades Using a Hardware-In-Loop Wind Turbine Simulator, Annu. IEEE Appl. …. (2005) 1791–1796.

[8] L.A.C. Lopes, J. Lhuilier, A. Mukherjee, M.F. Khokhar, A wind turbine emulator that represents the dynamics of the wind turbine rotor and drive train, PESC Rec. - IEEE Annu. Power Electron. Spec. Conf. 2005 (2005) 2092–2097.

[9] M. Monfared, H. Madadi Kojabadi, H. Rastegar, Static and dynamic wind turbine simulator using a converter controlled dc motor, Renew. Energy. 33 (2008) 906–913.

[10] B. Gong, D.W. Xu, Real-Time Wind Turbine Simulator for Wind Energy Conversion System, Ieee. Power. Electron. (2008) 1110–1114.

[11] I. Munteanu, A.I. Bratcu, S. Bacha, D. Roye, J. Guiraud, Hardware-in-the-loop-based simulator for a class of variable-speed wind energy conversion systems: Design and performance assessment, IEEE Trans. Energy Convers. 25 (2010) 564–576.

[12] L. Hui, M. Steurer, K.L. Shi, S. Woodruff, Z. Da, Development of a Unified Design, Test, and Research Platform for Wind Energy Systems Based on Hardware-in-the-Loop Real-Time Simulation, Ind. Electron. IEEE Trans. 53 (2006) 1144–1151.

[13] J. Castelló, J.M. Espí, R. García-Gil, Development details and performance assessment of a Wind Turbine Emulator, Renew. Energy. 86 (2016) 848–857.

[14] J. Chen, J. Chen, C. Gong, H. Wang, Design and Analysis of Dynamic Wind Turbine Simulator for Wind Energy Conversion System TL TD, IECON 2012-38th Annu. (2012) 971–977.

[15] M. Chinchilla, S. Amaltes, J.L. Rodriguez-Amenedo, Laboratory set-up for wind turbine emulation, in: 2004 IEEE Int. Conf. Ind. Technol. 2004. IEEE ICIT '04., 2004: pp. 553–557.

[16] E. Mohammadi, R. Fadaeinedjad*, H. Naji , Using a new wind turbine emulator to analyze tower shadow and yaw error effects, Energy Conversion and Management 174 (2019) 378–387.

[17] A. Sajadi, Ł. Rosłaniec, M. Kłos, P. Biczel, K.A. Loparo, An emulator for fixed pitch wind turbine studies, Renew. Energy. 87 (2016) 391–402.

[18] T.D. Hardy, Emulation of a Ge 1.5Mw Wind Turbine With a Dc Motor, 2011 IEEE Power Energy Soc. (2010).

[19] J. Neely, S. Glover, J. Finn, F.E. White, B. Loop, O. Wasynczuk, Wind turbine emulation for intelligent microgrid development, Proc. - 2012 IEEE Int. Conf. Cyber Technol. Autom. Control. Intell. Syst. CYBER 2012. (2012) 28–33.

[20] R. Patel, C. V. Patki, V. Agarwal, Armature and field controlled dc motor based wind turbine emulation for wind energy conversion systems operating over a wide range of wind velocity, Sustain. Energy Build. - Proc. Int. Conf. Sustain. Energy Build. SEB'09. (2009) 117–125.

[21] S. Tammaruckwattana, K. Ohyama, Modeling and simulation of permanent magnet synchronous generator wind power generation system using boost converter circuit, Power Electron. Appl. (EPE), 2013 15th Eur. Conf. (2013) 1–10.

[22] B. Rabelo, W. Hofmann, M. Glück, Emulation of the static and dynamic behavior of a wind turbine with a DC-machine drive, in: PESC Rec. - IEEE Annu. Power Electron. Spec. Conf., 2004: pp. 2107–2112.

[23] I. Moussa , A. Bouallegue , A. Khedher ,New wind turbine emulator based on DC machine: hardware implementation using FPGA board for an open-loop operation, IET Circuits Devices Syst. 2019, 13, 896–902.

[24] D. H. Wollz , S. A.Oliveira , L. P. Sampaio, Real-time monitoring of an electronic wind turbine emulator based on the dynamic PMSG model using a graphical interface. Renewable Energy 155 (2020) 296e308.

[25] D. Zouheyr, B. Lotfi, B. Abdelmadjid , Improved hardware implementation of a TSR based MPPT algorithm for a low cost connected wind turbine emulator under unbalanced wind speeds. Energy 232 (2021) 121039.

[26] T. Burton, D. Sharpe, N. Jenkind, E. Bossanyi, Wind Energy Handbook, 2001.

[27] Rigatos, G.G., 2009. Particle and Kalman filtering for state estimation and control of DC motors. *ISA transactions*, *48*(1), pp.62-72.

[28] S.Y. Chen, Kalman filter for robot vision: A survey, IEEE Trans. Ind. Electron. 59 (2012) 4409–442.