



Control of a Microturbine Using Neural Networks

Helbert Espitia¹(✉), Iván Machón², and Hilario López²

¹ Universidad Distrital Francisco José de Caldas, Bogotá, Colombia
heespitiac@udistrital.edu.co

² Universidad de Oviedo, Oviedo, Spain
{machonivan,hilario}@uniovi.es

Abstract. In many distributed generation applications, microturbines are used for energy conversion. On the other hand, neural networks are a suitable option for the control of complex non-linear systems. Thus, in this article is shown the speed control of a microturbine using neural networks. For this process, the identification of the microturbine using a neural network is carried out in order to subsequently perform the optimization of the other neural network used for control.

Keywords: Control · Neural networks · Microturbine

1 Introduction

Distributed resources of energy are an important alternative to supply electric power, the approach of distributed generation may allow the costs of generators and the production of electricity to be cheaper [1]. Some distributed resources of energy are: eolian generation, fuel cells, microturbines and photovoltaic systems [2]. According to [3] and [4], using appropriated control schemes is possible to improve the reliability and the power quality.

A microturbine (MT) generation system belongs to the type of thermal generation due to characteristics like lower cost, higher efficiency, more reliability and convenience; thus, MT system is considered one of the best forms of distributed generation. However, a microturbine is a complex thermal dynamic system which makes complicated to establish a precise model because of the manifested non-linearities and uncertainties [2, 5]. Some investigations addressing proper modeling for a MT can be seen in [2, 5–7] and [8].

Conventionally, a MT shows controls of velocity, acceleration, temperature, and fuel. According to [5], a microturbine is mainly adjusted by velocity which is regulated to a constant value close to nominal velocity in conditions of normal operation.

Meanwhile, neural networks (NN) offer an acceptable control alternative when having a highly complex system given by non-linearities, parameter variance, and saturation, among others [9].

A system of control to a microturbine using NN is proposed in this paper; the work is mainly focused on velocity controls, leaving the possibility of studying other control variables in further papers.

2 Microturbine Model

In relation of development models for a microturbine, in [6] a modular model of sixth order to a microturbine is presented. Moreover, [7] describes a simple model based on a transfer function.

According to [8], assuming the microturbine operates always close to its nominal velocity, and ignoring the start and off processes, it is then possible the construction of a simple model. In this regard, it is necessary to bear in mind that suppositions may arise uncertainties in the model [5].

2.1 Microturbine Operation

In general, the following process are involved in the operation of a microturbine:

1. Air at atmospheric pressure enters to the gas turbine through the compressor.
2. Air is compressed to achieve the conditions to combustion.
3. The fuel (gas) is mixed with the air in the combustion chamber.
4. Then, combustion takes place. Hot gases are expanded through the turbine to generate mechanic energy.

During the combustion process, the chemical energy present in the combustion reagents is transferred to the gas flux. This energy is given by gas enthalpy such that it becomes into mechanic work by the gas expansion through the turbine. It also occurs that in the last stage of the combustion process a part of the flux is derivate to activate the compressor [10]. Figure 1 displays the interaction of these systems. From this view three components can be seen to a MT:

1. Combustion system.
2. Turbine and compressor system.
3. Thermal system.

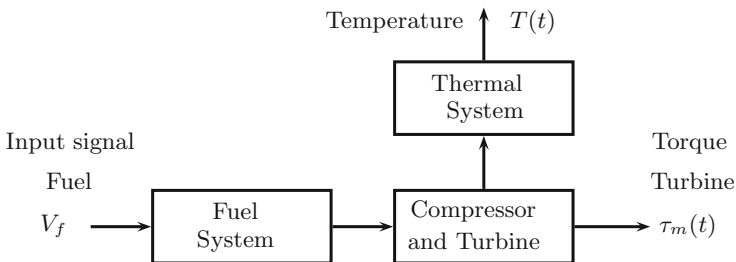


Fig. 1. Microturbine systems.

Fuel System. Figure 2 shows the fuel system parts. Main components are the valve positioner and the flux dynamic behavior. In this figure V_f is the fuel input signal, this is amplified via K_f (gain) and compensated using F_0 which are the fuel flux without load in nominal velocity. Then, the signal passes through the valve and actuator positioner to produce the fuel flow signal W_f [2].

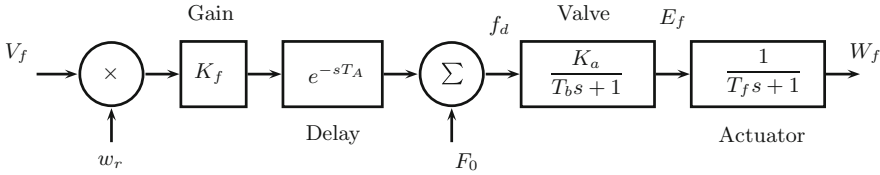


Fig. 2. Fuel subsystem.

According to [11] and [12], the transfer function associated to the valve is:

$$E_f = \frac{K_a}{T_b s + 1} f_d \tag{1}$$

Regarding the flux dynamics the respective transfer function is:

$$W_f = \frac{1}{T_f s + 1} E_f \tag{2}$$

where K_a represents the gain associated with the valve positioner, T_b and T_f are time constant values for the valve positioner and the fuel system. Meanwhile, f_d , E_1 are the input and output positioner of the valve, and W_f is the flux signal fuel.

Turbine and Compressor Systems. Figure 3 displays the block diagram of the gas turbine. Turbine input signals are the fuel flux W_f and the angular velocity $w_r(t)$. Output signals are the turbine torque $\tau_m(t)$. According to [11] and [12], the transfer for the gas turbine is:

$$W_C = \frac{1}{T_{CD} s + 1} W_{fC} \tag{3}$$

where T_{CD} is the constant time value associated with the dynamics of the gas turbine. The torque of the microturbine is described by:

$$F_C = a_C - b_C W_C + c_C (1 - w_r) \tag{4}$$

As observable, F_C is a function of the fuel flux and the turbine velocity.

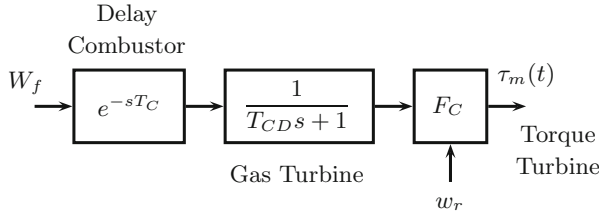


Fig. 3. Turbine and compressor systems.

Thermal System. The fuel burnt (fuel flux) in the combustion chambers generates torque of the turbine and the gas escape temperature [11]. Figure 4 shows the block diagram of the thermal system.

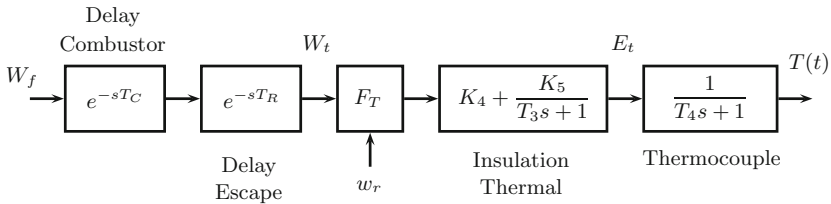


Fig. 4. Thermal system.

According to [11] and [12], the escape temperature of the gas is given by:

$$F_T = T_{ref} - a_t(1 - W_t) + b_t(1 - w_r) \tag{5}$$

where F_T is a function whose entries consist of fuel flux with delay W_t , and the turbine velocity $w_r(t)$. Escape temperature is measured using a set of thermocouples which are assigned in the radiators.

On the other hand, K_4 and K_5 are constant values associated with the insulation transfer function (opposing to the radiation). T_3 and T_4 are values of constant time of the transfer functions of both the insulation and the thermocouple. In addition, T_{ref} represents the escape temperature (reference temperature) [5, 8].

3 Permanent Magnet Synchronous Generator

In energy generation systems, the Permanent Magnet Synchronous Generator (PMSG) is widely used to obtain transformation from mechanical to electrical energy [13]. According to [14] and [15], permanent magnet synchronous generators need no feeding of direct current (DC) for the excitation circuit, neither have contact brushes.

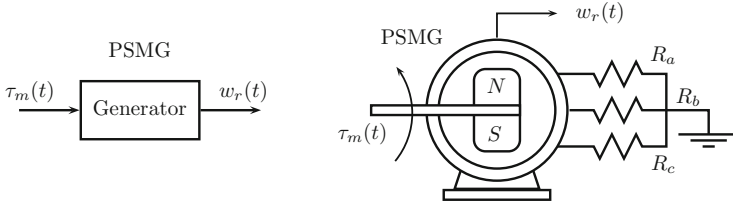


Fig. 5. Scheme of a PMSG.

The scheme for PMSG is shown in Fig. 5, where the input of the generator is the torque $\tau_m(t)$ and the output the angular velocity $w_r(t)$. Related works using PMSG and neural systems are presented in [16] and [17].

The dynamic model of a PMSG is composed of two parts, one mechanical and the other electrical that is obtained of the mobile synchronous framework of two signals $d-q$, where the axis q is 90° ahead axis d regarding the direction of rotation [15]. The mechanical system is composed by the following equations:

$$\frac{dw_r}{dt} = \frac{1}{J}(\tau_e - Fw_r - \tau_m) \tag{6}$$

$$\frac{d\theta}{dt} = w_r \tag{7}$$

where w_r rotor angular velocity, θ angular position, J is the rotor inertia, F rotor viscous friction, and τ_m mechanical torque. For the electrical part the model equations in the coordinates $d-q$ are:

$$\frac{di_d}{dt} = \frac{1}{L_d}v_d - \frac{R}{L_d}i_d + \frac{L_q}{L_d}pw_r i_q \tag{8}$$

$$\frac{di_q}{dt} = \frac{1}{L_q}v_q - \frac{R}{L_q}i_q - \frac{L_d}{L_q}pw_r i_d - \frac{\lambda pw_r}{L_q} \tag{9}$$

$$\tau_e = 1.5p[\lambda i_q + (L_d - L_q)i_d i_q] \tag{10}$$

in these equations i_d, i_q are the currents, v_d, v_q voltages, L_d, L_q the inductances, R stator resistance, λ amplitude of the flux induced by the permanent magnets, p number of pole pairs, and τ_e electromagnetic torque. For current and voltage conversions is used the parameter $\gamma = 2\pi/3$. The current conversion from $d-q$ to abc axis is:

$$\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \cos(\theta - \gamma) & -\sin(\theta - \gamma) \\ \cos(\theta + \gamma) & -\sin(\theta + \gamma) \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} \tag{11}$$

The voltage conversions from abc to $d-q$ corresponds to:

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos(\theta) & \cos(\theta - \gamma) & \cos(\theta + \gamma) \\ -\sin(\theta) & -\sin(\theta - \gamma) & -\sin(\theta + \gamma) \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \tag{12}$$

4 Neural Networks

According to [18,19] neural networks are a suitable alternative for the identification and control of dynamic systems when the system presents non-linearities. One of the advantages of neural networks is the flexibility of adaptation, association, evaluation and recognition of patterns [20].

The information is processed through inputs, a weight per connection and an output associated with an activation function as corresponds to equation (13), where each input x_n is weighted using w_n and added to be later evaluated by an activation function f to obtain the output y . Some activation functions are: sigmoid, gaussian, linear, saturated linear, and hard limit. The addition of network layers allows the increase of the adaptation capabilities [21].

$$y = f \left(\sum_{n=1}^N w_n x_n \right) \quad (13)$$

In applications of identification and control can be used feed forward or backward networks. In recurrent networks (feed-backward networks) feedback is present among neurons as well as from the output to the inputs. In feed-forward the signals move forward only, therefore, it is necessary an additional feedback from the output to the inputs [22].

5 Neural Control System

Neural control system is proposed as an alternative to identify and to control dynamic systems, using its capacity as approximator for general functions [9]. Figure 6 shows the neural network control model.

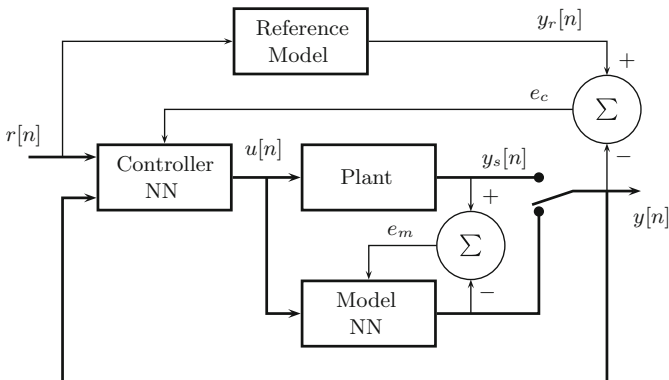


Fig. 6. Neural network control reference model.

In Fig. 6 it is shown two neural networks used for the neural control system, where $y[n]$ is the measured signal of the process, $u[n]$ is the control signal, $r[n]$

the reference for the controller, $y_s[n]$ the simulated plant output, and $y_r[n]$ the desired output (reference-model).

The considered architecture uses two neural networks, one for the control and another to estimate the plant model. Thus, the identification of the plant is first carried out and subsequently the controller training. The plant identification is made using the Backpropagation algorithm, after carrying out the identification, the controller training is performed using the algorithm Dynamic Backpropagation in which it incorporates in a closed loop the dynamics of the neural-model plant and the dynamic behavior of the controller, thus, the parameters of the controller are optimized so that the output has the desired behavior (reference-model). After being trained the controller is put into operation with the plant [23, 24].

5.1 Identification Using Neural Networks

According to [25], when using neural networks, the classic methods for identifying dynamic systems are series-parallel and parallel, shown in Fig. 7. In the series-parallel identification scheme are directly used the inputs and outputs of the plant; under this approach, the Backpropagation method can be used as a training algorithm [26]. On the other hand, in the parallel identification scheme, inputs of the plant are used and the output of the network being feedback to the inputs, with this perspective, the Dynamic Backpropagation is employed as training algorithm [24]. In this paper is used series-parallel identification.

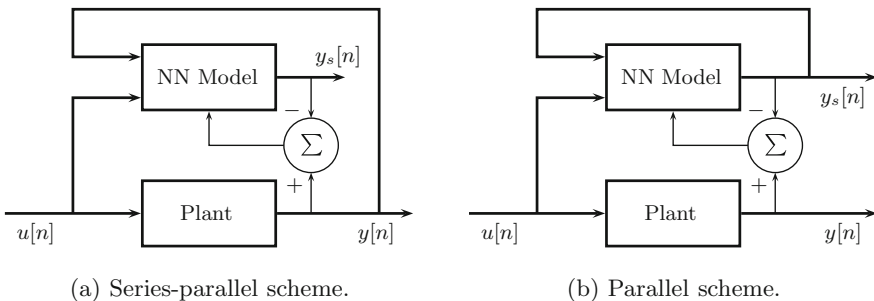


Fig. 7. Series-parallel and parallel identification scheme.

6 Implementation and Results

Implementing the process of identification, together with microturbine control are made in MATLAB[®] (version 7). The plant model used can be seen in Fig. 8; it can be noted that the input corresponds to the flux of fuel and the output to the angular velocity. A random signal is employed to obtain the data used in plant identification which allows to describe the different behaviors present in the system output (velocity). Figure 9 permits to see the neural controller architecture using MATLAB[®].

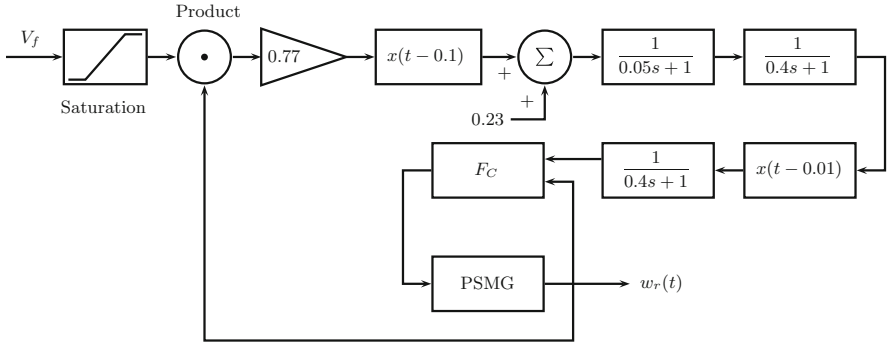


Fig. 8. Microturbine model used.

The general expressions for neural networks used for identification are given by f_p and for control by f_c functions. Taken p as the output delays, q as the input delays, m as reference delays, then the equations for plant model $y_s[n + 1]$ and the control signal $u[n + 1]$ are:

$$y_s[n + 1] = f_p(y[n], y[n - 1], \dots, y[n - p + 1], u[n], u[n - 1], \dots, u[n - q + 1]) \tag{14}$$

$$u[n + 1] = f_c(y[n], y[n - 1], \dots, y[n - p + 1], r[n], r[n - 1], \dots, r[n - m + 1], u[n - 1], \dots, u[n - q + 1]) \tag{15}$$

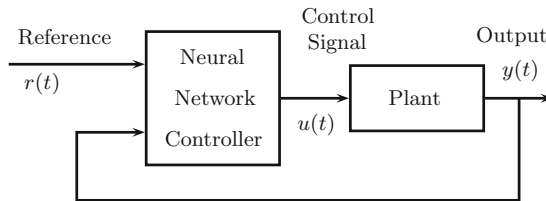


Fig. 9. Neural control system scheme.

For control and identification process, different neural networks are used taking 2, 3, and 4 delays for the input as well as to output feedback. The neural networks used have two hidden layers, the first layer is set using 2, 3, and 5 neurons, and the second layer one neuron. For plant identification is used the function “TRAINLM” that implements the Levenberg-Marquardt Backpropagation, and for controller training is used “TRAINBFGC” corresponding to the BFGS (Broyden-Fletcher-Goldfarb-Shanno) quasi-Newton Backpropagation method. The additional configurations of neural controller are:

- Hidden layers: 2
- Activation functions layer 1: sigmoid
- Activation function layer 2: linear
- Training epoch for identification: 300
- Training epoch for controller: 50
- Number of training data for identification: 6000
- Number of training data for controller: 1000

Each configuration is respectively executed 20 times to identification and control training; thus, obtaining the values of the Mean Squared Error (MSE) as shown in Table 1a for identification, and Table 1b for controller training. These tables present for each configuration the minimum, maximum, average and standard deviation (STD) values. To perform the controller training is used the configuration of 2 delays and 5 neurons obtained from the identification process (neural-model). Part of data used for identification can be seen in Fig. 10a; this figure also displays the results of the identification process made with the neural network.

Table 1. Results for identification and controller training.

(a) MSE training process values obtained for identification.

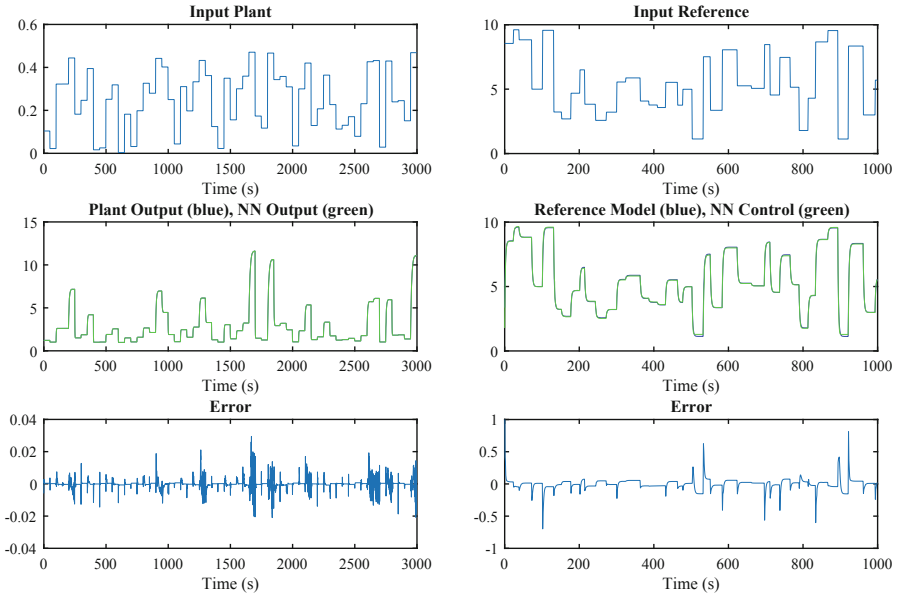
Delays: 1			
Value	Neurons: 2	Neurons: 3	Neurons: 5
Min	1.49E-04	1.36E-04	1.17E-04
Max	3.72E-03	3.68E-03	7.95E-04
Mean	1.54E-03	7.86E-04	1.79E-04
STD	1.66E-03	1.04E-03	1.50E-04
Delays: 2			
Value	Neurons: 2	Neurons: 3	Neurons: 5
Min	2.33E-05	1.38E-05	1.15E-05
Max	2.44E-03	2.30E-03	1.11E-03
Mean	1.50E-03	5.58E-04	1.89E-04
STD	9.24E-04	8.42E-04	3.66E-04
Delays: 3			
Value	Neurons: 2	Neurons: 3	Neurons: 5
Min	2.17E-05	1.86E-05	1.14E-05
Max	1.68E-03	1.59E-03	1.39E-03
Mean	1.11E-03	5.44E-04	2.78E-04
STD	5.74E-04	5.97E-04	4.26E-04

(b) MSE values obtained for controller training.

Delays: 1			
Value	Neurons: 2	Neurons: 3	Neurons: 5
Min	2.65E-02	1.20E-02	3.21E+00
Max	7.63E+01	2.28E+01	2.07E+02
Mean	1.52E+01	6.25E+00	3.60E+01
STD	2.56E+01	7.85E+00	5.59E+01
Delays: 2			
Value	Neurons: 2	Neurons: 3	Neurons: 5
Min	4.92E-03	4.30E-02	1.25E-01
Max	2.28E+01	9.17E+01	2.28E+02
Mean	1.24E+01	1.50E+01	5.32E+01
STD	9.67E+00	2.29E+01	7.23E+01
Delays: 3			
Value	Neurons: 2	Neurons: 3	Neurons: 5
Min	6.35E-03	4.85E-02	1.18E-02
Max	2.20E+01	2.28E+01	1.34E+02
Mean	6.60E+00	9.12E+00	2.69E+01
STD	7.83E+00	8.41E+00	3.22E+01

Figure 10b shows the control system response after the training, as an example, it is taken a configuration using 1 delay and 3 neurons. This process includes the identified model (with the neural network) to train the controller which is implemented with another neural network.

Finally, the results of the simulation of the neural controller with the plant model can be seen in Fig. 11; in this case, the microturbine model is used for simulation.



(a) Example of identification process result. (b) Example of controller training process result.

Fig. 10. Graphical results for identification and controller training. (Color figure online)

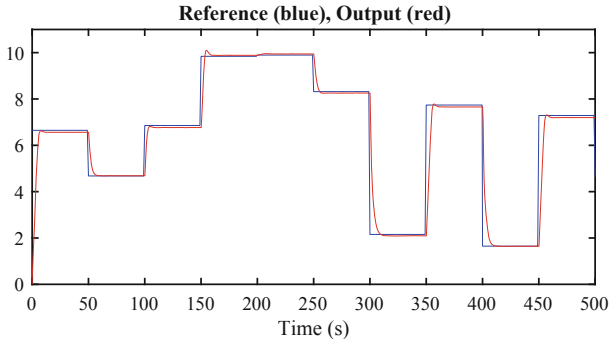


Fig. 11. Simulation for microturbine control. (Color figure online)

These simulations allow to see the system behavior when it is performing the controller training and also the controller response operating with the microturbine model.

7 Conclusions

In this work, the identification and control of a microturbine were achieved using a neural network scheme. Observations show that the performance of the

system is linked to the initial configuration of the neural network, which is made randomly.

The traditional Backpropagation algorithm was used to identify the plant, while the Dynamic Backpropagation algorithm was used for the optimization of the controller.

Taking the most suitable configuration obtained from the system, it is observable that the system can achieve the control for different speed values.

Plant parameter variations can be considered in a further work in such a way that an adaptive process may be implemented, as well as a neuro-fuzzy system in which random parameter initializations be needless in the neural system.

References

1. Dugan, R., McDermott, T., Ball, G.: Planning for distributed generation. *IEEE Ind. Appl. Mag.* **7**, 80–88 (2001)
2. Tong, J.P., Yu, T.: Nonlinear PID control design for improving stability of micro-turbine systems. In: *Electric Utility Deregulation and Restructuring and Power Technologies*, pp. 2515–2518 (2008)
3. Jiayi, H., Chuanwen, J., Rong, X.: A review on distributed energy resources and MicroGrid. *Renew. Sustain Energy Rev.* **12**, 2472–2483 (2008)
4. Khorshidi, A., Zolfaghari, M., Hejazi, M.A.: Dynamic modeling and simulation of microturbine generating system for stability analysis in microgrid networks. *Int. J. Basic Sci. Appl. Res.* **3**(9), 663–670 (2014)
5. Yu, T., Tong, J.P.: Auto disturbance rejection control of microturbine system. In: *IEEE Power Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century* (2008)
6. Li, Z., Wang, D.H., Xue, Y.L.: Research on ways of modeling of microturbine (part 1): analysis of dynamic characteristic. *Chin. J. Power Eng.* **25**, 13–17 (2005)
7. Lasseter R.: Dynamic models for micro-turbines and fuel cells. In: *Power Engineering Society Summer Meeting*, pp. 761–766 (2001)
8. Rowen, W.J.: Simplified mathematical representation of heavy duty gas turbines. *ASME J. Eng. Power* **105**, 865–869 (1983)
9. Nguyen, H., Prasad, N., Walker, C., Walker, E.: *A First Course in Fuzzy and Neural Control*. Chapman & Hall/CRC, Boca Raton (2003)
10. Jurado, F.: Modeling micro-turbines using Hammerstein models. *Int. J. Energy Res.* **29**, 841–85 (2005)
11. Guda, S., Wang, C., Nehrir, M.: Modeling of microturbine power generation systems. *Electr. Power Compon. Syst.* **34**(9), 1027–1041 (2006)
12. Wanik, M., Erlich, I.: Dynamic simulation of microturbine distributed generators integrated with multi-machines power system network. In: *2nd IEEE International Conference on Power and Energy, PECon 2008* (2008)
13. Chen, J., Wu, H., Sun, M., Jiang, W., Cai, L., Guo, C.: Modeling and simulation of directly driven wind turbine with permanent magnet synchronous generator. In: *IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia)* (2012)
14. Rolan, A., Luna, A., Gerardo, V., Aguilar, D., Azevedo, G.: Modelling of a variable speed wind turbine with a permanent magnet synchronous generator. In: *IEEE International Symposium on Industrial Electronics (ISIE)*, pp. 734–739, 5–8 July 2009

15. Patil, K., Mehta, B.: Modeling and control of variable speed wind turbine with permanent magnet synchronous generator. In: International Conference on Advances in Green Energy (ICAGE), pp. 17–18 (2014)
16. Espitia, H., Díaz, G.: Identification of a permanent magnet synchronous generator using neural networks. In: Workshop on Engineering Applications - International Congress on Engineering (WEA) (2015)
17. Espitia, H., Díaz, G., Díaz, S.: Control of a permanent magnet synchronous generator using a neuro-fuzzy system. In: Figueroa-García, J.C., López-Santana, E.R., Rodríguez-Molano, J.I. (eds.) WEA 2018. CCIS, vol. 915, pp. 89–101. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00350-0_8
18. Torres, N., Hernandez, C., Pedraza, L.: Redes neuronales y predicción de tráfico. *Revista Tecnura* **15**(29), 90–97 (2011)
19. Martínez, F., Gómez, D., Castiblanco, M.: Optimization of a neural architecture for the direct control of a Boost converter. *Revista Tecnura* **16**(32), 41–49 (2012)
20. Pérez, F., Fernández, H.: Las redes neuronales y la evaluación del riesgo de crédito. *Revista Ingenierías Universidad de Medellín* **6**(10), 77–91 (2007)
21. Arrieta, J., Torres, J., Velásquez, H.: Predicciones de modelos econométricos y redes neuronales: el caso de la acción de SURAMINV. *Semestre Económico* **12**(25), 95–109 (2009)
22. Heaton, J.: Introduction to Neural Networks with Java, pp. 41–49. Heaton Research Inc., St. Louis (2008)
23. Hagan, M., De Jesús, O., Schultz, R.: Training recurrent networks for filtering and control. In: Medsker, L.R., Jain, L.C. (eds.) *Recurrent Neural Networks: Design and Applications*, pp. 325–354. CRC Press, Boca Raton (2000)
24. De Jesús, O., Hagan, M.T.: Backpropagation algorithms for a broad class of dynamic networks. *IEEE Trans. Neural Netw.* **18**(1), 14–27 (2007)
25. Garrido, S.: Identificación, estimación y control de sistemas no-lineales mediante RGO. Tesis para optar al título de Doctor, Departamento de Ingeniería de Sistemas y Automática, Universidad Carlos III, Madrid, España (1999)
26. Singh, M., Singh, I., Verma, A.: Identification on non linear series-parallel model using neural network. *MIT Int. J. Electr. Instrum. Eng.* **3**(1), 21–23 (2013)
27. Li, Y., Zhao, X., Jiao, L.: A nonlinear system identification approach based on neurofuzzy networks. In: *Proceedings of ICSP* (2000)