


Neural Modeling of Fuzzy Controllers for Maximum Power Point Tracking in Photovoltaic Energy Systems

JOSE MANUEL LOPEZ-GUEDE ^{1,6} JOSEAN RAMOS-HERNANZ,²
NECMI ALTIN,³ SABAN OZDEMIR,⁴ EROL KURT,³ and GORKA AZKUNE⁵

1.—Faculty of Engineering of Vitoria-Gasteiz, Department of Systems Engineering and Automatic Department, University of the Basque Country (UPV/EHU), C/Nieves Cano 12, 01006 Vitoria-Gasteiz, Spain. 2.—Faculty of Engineering of Vitoria-Gasteiz, Department of Electrical Engineering, University of the Basque Country (UPV/EHU), C/Nieves Cano 12, 01006 Vitoria-Gasteiz, Spain. 3.—Faculty of Technology, Department of Electrical and Electronics Engineering, Gazi University, 06500 Teknikokullar, Ankara, Turkey. 4.—Department of Electrical and Electronics Engineering, Vocational School of Technical Sciences, Gazi University, 06500 Teknikokullar, Ankara, Turkey. 5.—Faculty of Engineering, DeustoTech – Deusto Institute of Technology, University of Deusto, Avda. de las Universidades, 24, 48007 Bilbao, Spain. 6.—e-mail: jm.lopez@ehu.es

One field in which electronic materials have an important role is energy generation, especially within the scope of photovoltaic energy. This paper deals with one of the most relevant enabling technologies within that scope, i.e, the algorithms for maximum power point tracking implemented in the direct current to direct current converters and its modeling through artificial neural networks (ANNs). More specifically, as a proof of concept, we have addressed the problem of modeling a fuzzy logic controller that has shown its performance in previous works, and more specifically the dimensionless duty cycle signal that controls a quadratic boost converter. We achieved a very accurate model since the obtained medium squared error is 3.47×10^{-6} , the maximum error is 16.32×10^{-3} and the regression coefficient R is 0.99992, all for the test dataset. This neural implementation has obvious advantages such as a higher fault tolerance and a simpler implementation, dispensing with all the complex elements needed to run a fuzzy controller (fuzzifier, defuzzifier, inference engine and knowledge base) because, ultimately, ANNs are sums and products.

Key words: Fuzzy logic control, FLC, artificial neural networks, ANN, photovoltaic systems

INTRODUCTION

Recent research has been focused on renewable energy sources and related technologies due to the increasing awareness of environmental issues, energy security and depleting fossil fuels. Electronic materials integrated into photovoltaic (PV) systems which convert solar energy into electrical energy directly are one of the most important types of renewable energy sources with superior advantages

such as not including moving parts, zero fuel cost and low maintenance cost, among others.¹ Although the prices of PV modules have been high, a significant reduction has been obtained with improved manufacturing technologies and increasing production capacity. While the PV module prices were ≥ 1.4 \$/watt in 2008, today they have decreased to ≤ 0.4 \$/watt.² This removes one of the important drawbacks of PV systems. Besides, many researchers have been focused on improving PV cells and module efficiency. In fact, increasing efficiency also decreases the unit cost of the PV cell and module. Anyway, since the PV module generates its maximum power at a single operation point, working at

(Received December 11, 2017; accepted May 23, 2018; published online June 6, 2018)

this maximum power point (MPP) during normal operation is still important. The operation point of a PV system can be controlled by adjusting the output load or output voltage of the PV system and this action is called maximum power point tracking (MPPT). Controlling the output voltage of the PV system via a power electronic converter is the most common way, which, in turn, is usually controlled via a specific control algorithm to drive this action.

Several MPPT algorithms have been proposed such as the constant voltage, the constant current, the look-up table, the pilot cell, the perturb and observe (P&O), the incremental conductance (IC), the ripple correlation method and the sliding mode control (SMC). Besides, artificial intelligence-based methods have also been investigated such as the fuzzy logic controller (FLC) and the artificial neural networks (ANNs) with their superior features not requiring the exact model of the system, robust structures, etc.³⁻⁵

Regarding converters, various direct current to direct current (DC–DC) topologies have been applied to PV systems. The buck, boost and buck-boost converters are the most common topologies. However, the pulsating supply current feature of buck and buck-boost converters limits their usage in PV applications. A boost converter with higher output voltage than input has a superior advantage of drawing current from the PV system in both switching intervals, and thus is very popular in PV applications. The output voltage of the boost converter increases with the increasing duty cycle, but it decreases the stability and increases the control difficulty, and this limits its practical voltage gain value.^{6,7} Therefore, a number of PV modules whose output voltage values are low can be connected in series to fulfill the voltage requirements of common loads. However, the number of series-connected PV modules must be within certain limits in practice due to limitations on PV voltage isolation, efficiency, shadowing effect, etc. Consequently, alternative DC–DC converter topologies have been proposed to provide higher voltage conversion gain. Although the isolated DC–DC converter topology has removed the limitation on the voltage conversion gain, this structure still causes some problems such as increasing cost and complexity.⁸ Alternative non-isolated DC–DC converter topologies have also been proposed to provide high voltage gain such as the coupled inductors, the cascaded boost converters, the voltage lifting topologies and the quadratic boost converters (QBCs).^{6,9-11} QBCs provide quadratic voltage conversion gain with only a single active switch and prevent higher voltage stresses on both active and passive components. Therefore, the QBC has become popular and is used for different DC–DC converter applications such as power factor correction and PV applications.¹²⁻¹⁴ The output voltage of fuel cells or PV modules is usually low, and this low voltage should be increased to supply conventional AC loads or to export the generated energy to the

grid. Therefore, a compact, robust, reliable and high-efficiency converter design with a high voltage conversion ratio is an important requirement for PV and fuel cell-supplied systems. Although various studies have been presented on the control of QBCs, the number of studies on MPPT with QBCs is limited. A much more comprehensive review of the state of art is provided in “[State of the Art](#)” section.

The main objective of this paper is to obtain a model based on ANNs of an FLC suited to implement an MPPT algorithm for a PV system, based on the fact that the FLC has been properly tuned during a previous stage that is not tackled in this paper.⁶ This approach has obvious advantages such as higher fault tolerance and a simpler implementation, dispensing with all the complex elements needed to run a fuzzy controller (fuzzifier, defuzzifier, inference engine and knowledge base) because, ultimately, an ANN consists of sums and products. We have taken the change in PV power (dP_{PV}/dt) and the change in PV voltage (dV_{PV}/dt) as input variables, while the change in the dimensionless duty cycle (cD) signal of the QBC is taken as output one. We achieved the goal since in modeling this last signal, the obtained medium squared error (MSE) accuracy with a test dataset is 3.47×10^{-6} with a maximum error of 16.32×10^{-3} .

The remainder of the paper is structured as follows. “[State of the Art](#)” section gives a comprehensive state of the art on the topics covered by this paper and “[The FLC Based MPPT Algorithm](#)” section provides a deeper insight into the FLC applied to the MPPT task explaining the structure of the FLC that will be modeled using the experimental design described in “[Experimental Design](#)” section, while the obtained results are discussed in “[Experimental Results](#)” section. Finally, “[Conclusions](#)” section summarizes our main conclusions.

STATE OF THE ART

This section gives the state of the art of the different and specific topics covered in the paper. “[Photovoltaic Elements Modeling](#)” section describes the PV elements modeling, “[The Quadratic Boost Converter \(QBC\)](#)” section discusses the need for converters, their latest designs and the internal structure of the QBCs, while the state of the art of the algorithms to modify their internal parameters for getting the maximum power is presented in “[Maximum Power Point Tracking \(MPPT\) Algorithms](#)” section. Finally, a short description of ANNs and some cutting-edge works within the scope of the modeling task that this work faces is given in “[Artificial Neural Networks \(ANN\)](#)” section.

Photovoltaic Elements Modeling

There are many studies on modeling PV cells. The PV cell can be modeled with a current source modeling the generated photocurrent related to

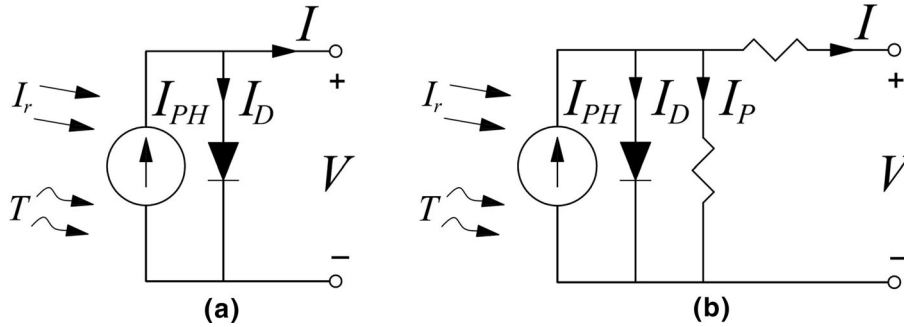


Fig. 1. Single-diode model of the PV cell. (a) Ideal and (b) practical PV cell models.

the solar irradiation and temperature levels. In the literature, researchers usually define the PV cell with a single-diode model or two-diode model. Since it has only five parameters and provides good accuracy, the single-diode model is the most common modeling. This model should also contain an anti-parallel diode which models effects of the temperature variation via the diode current as shown in Fig. 1a. According to this figure, Eq. 1 can be written:

$$I = I_{ph} - I_d \tag{1}$$

where I_{ph} represents the generated photocurrent and I_d is the diode current related to the saturation given as Eq. 2:

$$I_d = I_0 \left[e^{\left(\frac{qV}{AN_S k T_C} \right)} - 1 \right] \tag{2}$$

where I_0 is the reverse saturation current of the diode, q is the electron charge ($1.602 \times 10^{-19} \text{C}$), V is the output voltage, A is the ideality factor, N_S is the number of serial connected PV cells, k is the Boltzmann constant ($1.381 \times 10^{-23} \text{ J/K}$) and T_C is the actual cell temperature in Kelvin. However, this model does not include the internal losses. The series resistance (R_S) and the parallel resistance (R_P), which are impossible to neglect, should be added to the model to make it more practical as depicted in Fig. 1b. When these resistors are considered the diode current (I_d) and the output current of the PV cell (I) can be written as given in Eqs. 3 and 4:

$$I_d = I_0 \left[e^{\left(\frac{qV + IR_S}{AN_S k T_C} \right)} - 1 \right] \tag{3}$$

$$I = I_{ph} - I_d - I_P \tag{4}$$

Finally, the output current of the PV module, which contains N_S series-connected cells is given by Eq. 5:

$$I = I_{ph} - I_0 \left[e^{\left(\frac{qV + IR_S}{AN_S k T_C} \right)} - 1 \right] - \frac{V + I \cdot R_S}{R_P} \tag{5}$$

As seen from Eq. 5, the PV model has a nonlinear current–voltage (I–V) characteristic. Efficiency values of commercial PV panels are around 9–24% depending on their technology and materials.¹⁵ Because of the nonlinear characteristics of the PV module, the efficiency value given in datasheets is only applicable at a certain current and voltage values for every solar irradiation and temperature condition. Therefore, the output current and voltage of the PV system should be continuously controlled to obtain maximum energy conversion efficiency at different operating conditions. Otherwise, maximum available power cannot be reached by the PV array. This control action is called MPPT implemented through converters (“The Quadratic Boost Converter (QBC)” section) using specific algorithms (“Maximum Power Point Tracking (MPPT) Algorithms” section).

More complete references about the state of the art on modeling different photovoltaic elements can be found in Refs. 16–18.

The Quadratic Boost Converter (QBC)

DC–DC converters are switching systems that control the average value of the voltage (or current) at the output (load) varying the switching times between the input (DC source) and the output, allowing adjustment of the uncontrolled voltage supplied by the photovoltaic modules to a regulated DC voltage at its output. Step-up, step-down or step-up/down converters can be designed. The buck, boost and buck-boost converters are the most common non-isolated DC–DC converter topologies.

Although buck, boost and buck-boost converter topologies can be used in PV systems, boost converters have a superior advantage of drawing current from the PV system in both switching intervals and are commonly used in PV systems to

track the MPP. The boost converter steps up its input DC voltage. Although, its output voltage increases with the increasing duty cycle, this decreases the stability and increases the control difficulty. Therefore, voltage conversion gain of the boost converter is limited. The practical voltage conversion gain of the conventional boost converter is recommended to be selected at a maximum of four.^{6,7} Although the isolated DC–DC converter topologies provide greater voltage conversion gain values, these topologies have some drawbacks such as increasing cost, complexity and size.⁸ The conventional boost converter combined with a switched capacitor has been proposed to extend the voltage conversion gain. In this topology, voltage conversion gain is related to the number of capacitors used in the circuit. However, the voltage regulation action decreases the efficiency of the converter significantly, and thus an additional converter is required to combine the high voltage conversion gain and voltage regulation features with high efficiency.¹⁹ In addition, the power switches suffer from a high charge current of the capacitors. The DC–DC multilevel boost converter topology, which also combines the boost converter and switched capacitor, is proposed to remove this additional converter requirement. In this topology, a number of capacitors are charged with the same voltage, which is controlled by a conventional boost converter.²⁰ This solves the voltage regulation problem; however, total load current flows through the output capacitors, limiting its usage.⁶

Another technique used to obtain high voltage conversion gain is the use of coupled inductors.⁹ However, the leakage inductor energy of the coupled inductors causes voltage spikes that increase the switching losses and decrease the efficiency.¹¹ Although active and passive clamp circuits are designed to recycle the leakage inductor energy, these additional clamp circuits increase the cost and complexity of the system.⁶ Using two cascaded boost converters to obtain high voltage conversion gain is also possible; however, this topology doubles the number of required voltage controllers and active switches.

The QBC schematic diagram given in Fig. 2 is similar to two cascaded boost converters and provides the same voltage conversion ratio by using only one active switch and one voltage controller. The output voltage is given as a quadratic function of the duty cycle of the switching signal.¹⁰ Since the QBC has only one active switch, additional active switch and driver circuit requirements are removed and a more reliable and efficient converter design is obtained. Therefore, the QBC has become popular and used in several DC–DC converter applications.^{12–14}

In the literature, there are a number of studies on quadratic converters which can be designed as a buck or boost converter.^{7,20,21} Besides, alternative control schemes have also been proposed for these

types of converters in recent studies.^{9,19,20} Actually, the quadratic converter can be implemented by cascaded connection of two conventional converters with elimination of the capacitor of the first (supply side) converter. Thus, an equal conversion ratio with a cascaded converter is obtained with a reduced number of components. This yields lower cost and more compact designs. However, the efficiency of this converter is still lower than the conventional one.²¹ It is well known that a higher duty ratio which provides higher voltage gain decreases the conventional converter efficiency dramatically. This also increases voltage stress on the switches and electromagnetic interference.^{10,11,13} Therefore, it can be noted that the QBC has superior performance for high step-up applications.

The QBC circuit given in Fig. 2 can be easily analyzed. If the controlled switch S_1 is turned on (ON state), then diodes D_1 and D_3 pass to the OFF state, and supply current flows through L_1 and D_2 . In this condition, inductor L_1 gathers energy from the supply and inductor L_2 gathers energy from capacitor C_1 . Simultaneously, the load is supplied by the output capacitor C_2 . Then, switch S_1 turns off (OFF state). In this condition, diode states are completely contrary; D_1 and D_3 are in the ON state, and D_2 is in the OFF state. At the same time, inductors L_1 and L_2 charge capacitors C_1 and C_2 . In addition, inductors supply the load energy demand. The converter conversion ratio equation can be obtained from the differential Eqs. 6–9 obtained according to the control signal u .¹⁴

$$\frac{di_{L_1}}{dt} = \frac{v_{PV}}{L_1} - \frac{v_{C_1}}{L_1}(1-u) \quad (6)$$

$$\frac{di_{L_2}}{dt} = \frac{v_{C_1}}{L_2} - \frac{v_{C_2}}{L_2}(1-u) \quad (7)$$

$$\frac{di_{vC_1}}{dt} = \frac{i_{L_2}}{C_1} - \frac{i_{L_1}}{C_1}(1-u) \quad (8)$$

$$\frac{dv_{c_2}}{dt} = -\frac{v_{C_2}}{R_L C_2} + \frac{i_{L_2}}{C_2}(1-u) - \frac{i_0}{C_2} \quad (9)$$

where u is the control signal whose value is “1” when S_1 is turned on (ON state) and “0” when S_1 is turned off (OFF state). Here, i_0 is the load current. In steady-state conditions, all the derivative terms

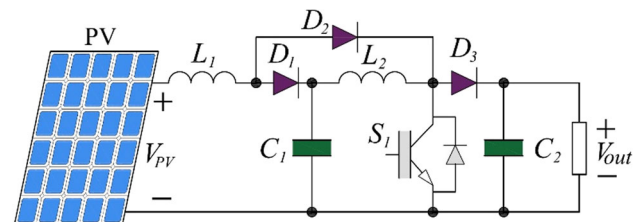


Fig. 2. The PV-supplied quadratic boost converter.

are equal to zero and Eq. 10 is obtained by substituting the control signal D instead of u in Eqs. 6–9:

$$\frac{v_{C_1}}{v_{PV}} = \frac{v_{C_2}}{v_{C_1}} = \frac{1}{1-D} \tag{10}$$

Finally, the conversion ratio of the converter ($M(D)$) can be obtained as given by Eq. 11:

$$M(D) = \left(\frac{v_{C_1}}{v_{PV}}\right) \left(\frac{v_{C_2}}{v_{C_1}}\right) = \frac{1}{(1-D)^2} \tag{11}$$

As can be seen from Eq. 11, the conversion ratio of the QBC is an exponential expression that provides a high conversion rate even with a lower duty cycle. In addition, using Eqs. 6–9, the state equations of the QBC can be described in state-space form by Eqs. 12 and 13:

$$\begin{bmatrix} \dot{i}_{L_1} \\ \dot{v}_{c_1} \\ \dot{i}_{L_2} \\ \dot{v}_{C_2} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{(1-u)}{L_1} & 0 & 0 \\ \frac{(1-u)}{C_1} & 0 & -\frac{1}{C_1} & 0 \\ 0 & \frac{1}{L_2} & 0 & -\frac{(1-u)}{L_2} \\ 0 & 0 & -\frac{(1-u)}{C_2} & -\frac{1}{RC_2} \end{bmatrix} \begin{bmatrix} i_{L_1} \\ v_{c_1} \\ i_{L_2} \\ v_{C_2} \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \\ 0 \\ 0 \end{bmatrix} v_{PV} \tag{12}$$

$$v_{out} = [0 \ 0 \ 0 \ 1] \begin{bmatrix} i_1 \\ v_{c_1} \\ i_2 \\ v_{C_2} \end{bmatrix} \tag{13}$$

where R is the load resistance.

Maximum Power Point Tracking (MPPT) Algorithms

The output voltage and current of the PV module has nonlinear characteristics and varies with environmental factors such as radiation, temperature, pollution and shadowing. So, the amount of energy produced by the PV module changes depending on these environmental conditions. Therefore, the PV output parameters should be continuously monitored and the operation point of the system must be continuously controlled to extract the maximum possible power from the PV module or the PV system. The MPPT algorithms are used for this aim,^{5,22} and since they have a major effect on the efficiency and the economical benefit of the PV system, many different MPPT algorithms have been proposed.^{5,22,23} These algorithms are used to obtain the maximum power of a PV element by modifying

the internal structure of the converter placed between the PV module and the load. These algorithms can be grouped into two categories as direct and indirect methods. Indirect methods such as the constant voltage, the constant current, the look-up table and pilot cell methods are simple in nature, providing high tracking speed and can be implemented easily. However, these methods use some predefined measurements, equations and/or characteristics to estimate the MPP, and some variable characteristics of the PV modules depend on age, pollution and dirt on the module surface which can decrease their accuracy and limit their application areas. Besides, these methods are module-dependent and must be revised for each different module model.

The direct methods use the online measurements of some PV module characteristics and track the MPP of the system. Therefore, they are module-independent and track the real MPP independently from above-mentioned effects. However, they usually have more complicated structures. The P&O and the IC algorithms are the most common direct MPPT methods. In the P&O algorithm, the output voltage of the PV system is perturbed in one direction and the variation of the output power is checked to determine the next variation. Since the algorithm is based on the perturbation of the operation point, this algorithm causes an oscillation around the operation point (in steady state, around the MPP). Although that can be reduced by using a smaller step size, this also decreases the speed of the MPPT action. The IC method checks the variation of the PV output power versus PV output voltage. The sign of this variation gives the place of the operation point on power versus voltage (P-V) curve of the PV system, being zero at the MPP. This method causes a power oscillation around the MPP too, but the amplitude of these oscillations are lower than in the case of the P&O method. Comparable to the P&O algorithm, using small step sizes can reduce the oscillations, but this also reduces the tracking speed. Therefore, some updates have been proposed to obtain adaptive step size depending on the operation points: larger step sizes when the system is far from the MPP and smaller step size when the system moves closer to the MPP. However, these methods usually use a coefficient that directly affects the speed and oscillation amplitudes, and determining this coefficient is a challenge. A number of algorithms such as ripple correlation control, current sweep algorithm and sliding mode control algorithm have been proposed.^{4,22–24} These methods have some advantages and disadvantages in terms of tracking speed, complexity, number of required measurements, etc.

Besides, popular artificial intelligence-based methods such as FLCs, ANNs, genetic algorithms and particle swarm intelligence have also been investigated for MPPT applications.^{3,4,22,23} The FLC has significant advantages such as removing

the system model requirement and lower dependency of the mathematical model and system parameters, thus providing high-performance MPPT action,^{1,5,6} as will be discussed in “The FLC Based MPPT Algorithm” section. The FLC-based MPPT algorithm has become popular in recent years because it provides variable step size, and thus improves the tracking speed and accuracy and decreases the power oscillations.⁶ ANNs are also another technique used to track the MPP of the PV system due to their ability to learn complex behaviors from experimental data and their capacity to generalize results.

Artificial Neural Networks (ANNs)

ANNs²⁵ are used, among other tasks, to model dynamic systems²⁶ which could include nonlinearities and complex behaviors. There are a number of different types of ANNs, each one having some characteristics that make them well-suited for facing specific problems. All types of ANNs have a number of outstanding characteristics to tackle the modeling task that is being faced in this paper. The first property is that they work in a parallel fashion, in such a way that they have real-time calculation capacity. The second one is that they have proven learning capabilities because they can learn complex models if they are properly designed (appropriate inputs and outputs) and the most suitable training algorithm is used. Finally, they have generalization capabilities; i.e., if the training process has been properly driven, the prediction given by the outputs of the ANNs in new situations will be adequate. A more complete and recent state of the art dealing with the modeling of PV elements through ANNs can be found in Refs. 17, 18, 27, and 28.

THE FLC-BASED MPPT ALGORITHM

FLCs are one of the most successful applications of the fuzzy sets. They use linguistic variables as human beings do rather than numerical variables and have the advantages of working with imprecise inputs, not requiring an accurate mathematical model of the system, and handling nonlinearities.

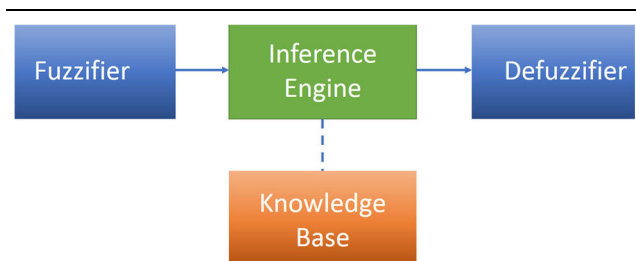


Fig. 3. Structure of a generic fuzzy logic controller.

The FLC theory allows designers to use non-precise or ill-defined concepts. The nonlinear and adaptive nature of FLC provides robust performance even under parameter variations and external disturbances; therefore, it provides superior MPPT performance even under rapidly changing atmospheric conditions.

The general structure of the FLC is shown in Fig. 3. It is composed of a fuzzifier, a knowledge base, an inference engine and a defuzzifier. Defining the input and output variables is one of the most important steps in the FLC design. In this work, an FLC with two inputs and one output is designed as an MPPT algorithm. The change in PV power (dP_{PV}/dt) and the change in PV voltage (dV_{PV}/dt) are used as inputs of the FLC, while the output variable is defined as the change in duty cycle (cD) of the QBC. An integrator is used at the output of the FLC to obtain the duty cycle value of the QBC. Seven triangular membership functions are used for both input and output variables of the FLC, as shown in Fig. 4. As we can see, linguistic labels such as positive large (PL), positive medium (PM), positive small (PS), zero (Z), negative small (NS), negative medium (NM) and negative large (NL) are used for these input and output membership functions. The required linguistic definitions about the input and output variables and rule base are stored in the knowledge base. The rule base consists of some fuzzy rules expressed in the IF-THEN fuzzy conditional statements and maps the relations between inputs and outputs. The rule base of the FLC is determined as given in Table I, including 49 rules that are set based upon the knowledge and working of the system.

The inference engine acts like the human decision process and generates the fuzzy output depending on the knowledge of the control rules and the linguistic variable definitions. The active rules are determined by using fuzzified input variables and the rule base, and then these active rules are evaluated using fuzzy reasoning methods. In this work, the commonly used min-max inference method is used to generate the fuzzy control command. Finally, the defuzzifier converts the fuzzy control command to a real one. This action is called defuzzification and it is performed using the center of gravity method given by Eq. 14:

$$X^* = \frac{\sum_{i=1}^n \mu(x_i) \cdot x_i}{\sum_{i=1}^n \mu(x_i)} \quad (14)$$

where X^* is the defuzzified value, x_i is the sample element, $\mu(x_i)$ is the membership function and n is the number of elements in the sample.

EXPERIMENTAL DESIGN

This section describes the experimental design carried out to obtain a neural model of a specific

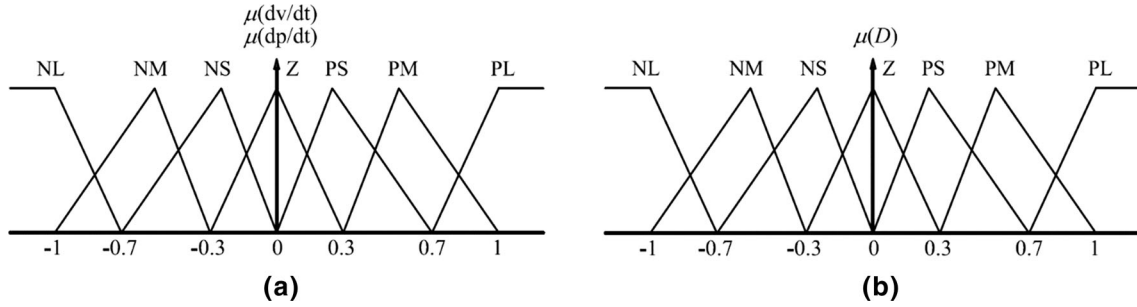


Fig. 4. (a) Membership functions of input variables. (b) Membership functions of output variable.

Table I. Rule base of the FLC

dV/dt	NL	NM	NS	Z	PS	PM	PL
Change in PV power (dP/dt)							
NL	NL	NL	NM	Z	PM	PL	PL
NM	NL	NM	NM	Z	PM	PM	PL
NS	NL	NM	NS	Z	PS	PM	PL
Z	Z	Z	Z	Z	Z	Z	Z
PS	PS	PS	PS	Z	NS	NM	NL
PM	PM	PM	PS	Z	NS	NM	NL
PL	PL	PL	PM	Z	NM	NL	NL

PL positive large; *PM* positive medium; *PS* positive small; *Z* zero; *NS* negative small; *NM* negative medium; *NL* negative large.

Table II. More relevant statistical descriptors of the gathered data

\overline{cD}	σ_{cD}	$ \overline{cD} $	$\min(cD)$	$\min(cD)$	$\max(cD)$	$\max(cD)$
- 0.105	0.150	0.115	- 0.907	5.01×10^{-19}	0.165	0.907

FLC, including the process of data gathering and dataset configuration in “[Data Gathering and Dataset Configuration](#)” section, the internal and external structure of such a model in “[Model Structure](#)” section and the experimental setup used to find out the most appropriate ANN in “[Experimental Setup](#)” section.

Data Gathering and Dataset Configuration

The creation of the FLC model we are facing is based on data that describes the behavior of such FLC (dP_{PV}/dt , dV_{PV}/dt and cD). Since the FLC has been implemented in Matlab/Simulink, we have gathered information from that environment. The initial data capture was done using a 100-kHz sampling frequency; so, with a sampling time of $10 \mu s$, in 0.5 s, we recorded 50,000 samples. As

there were many points, we postprocessed this data carrying out a resampling process with a ratio of 1:5, in such a way that at the end, we obtained 10,000 samples, which is equivalent to an initial sampling frequency of 20 kHz and a sampling time of $20 \mu s$. Once we had gathered the data, we proceeded to do a brief statistical analysis to discover the basic statistical descriptors of the dataset, which are shown in Table II.

After the gathering process, it is mandatory to properly configure the datasets to carry out the training and test processes. The configuration consists of normalizing and/or adding noise to the input and output patterns of the dataset, obtaining in this case 10 different datasets:

- The normalization process of the input/output attributes of the datasets translates them into

the range $[-1, 1]$. So, this transformation leads to two different versions of each dataset; i.e, a raw un-normalized and a normalized one.

- The noise addition process of the input/output attributes of the datasets obtains a number of dataset versions adding noise to each attribute based on uniformly distributed pseudo-random numbers in the range $r \in [-1, 1]$, weighted by a parameter $\text{noisew} \in [0, 100]$, as specified by the actualization rule of Eq. 15. This modification allows additive or subtractive noise modulated by the noisew parameter. In the experiments carried out in this paper, the values $\text{noisew} \in \{0, 1, 2, 5, 10\}$ have been used (note that $\text{noisew} = 0$ means that the original value has not changed).

$$\text{attribute} \leftarrow \text{attribute} \{1 + [\text{noisew} \cdot (2r - 1)]\} \quad (15)$$

Model Structure

Regarding the external interface of the neuronal model with the overall system in which it will be integrated, it must mimic the structure of the FLC that it will model and which was previously explained in “[The FLC Based MPPT Algorithm](#)” section. So, it will have the following input/output specifications:

- Inputs: the change in PV power (dP_{PV}/dt) and the change in PV voltage (dV_{PV}/dt)
- Outputs: the change in duty cycle (cD) of the QBC

With regard to its internal structure, the ANN will have a feedforward structure with only one hidden layer. Given the previous input and output requirements, the input layer will have two neurons and the output layer will have only one.

for each dataset (combination of normalization and noise)

for each nodes size h of the hidden layer

for each feasible activation function f

for each different initialization i

Create an ANN with:

- two inputs
- one output
- h nodes in the hidden layer
- f as activation function

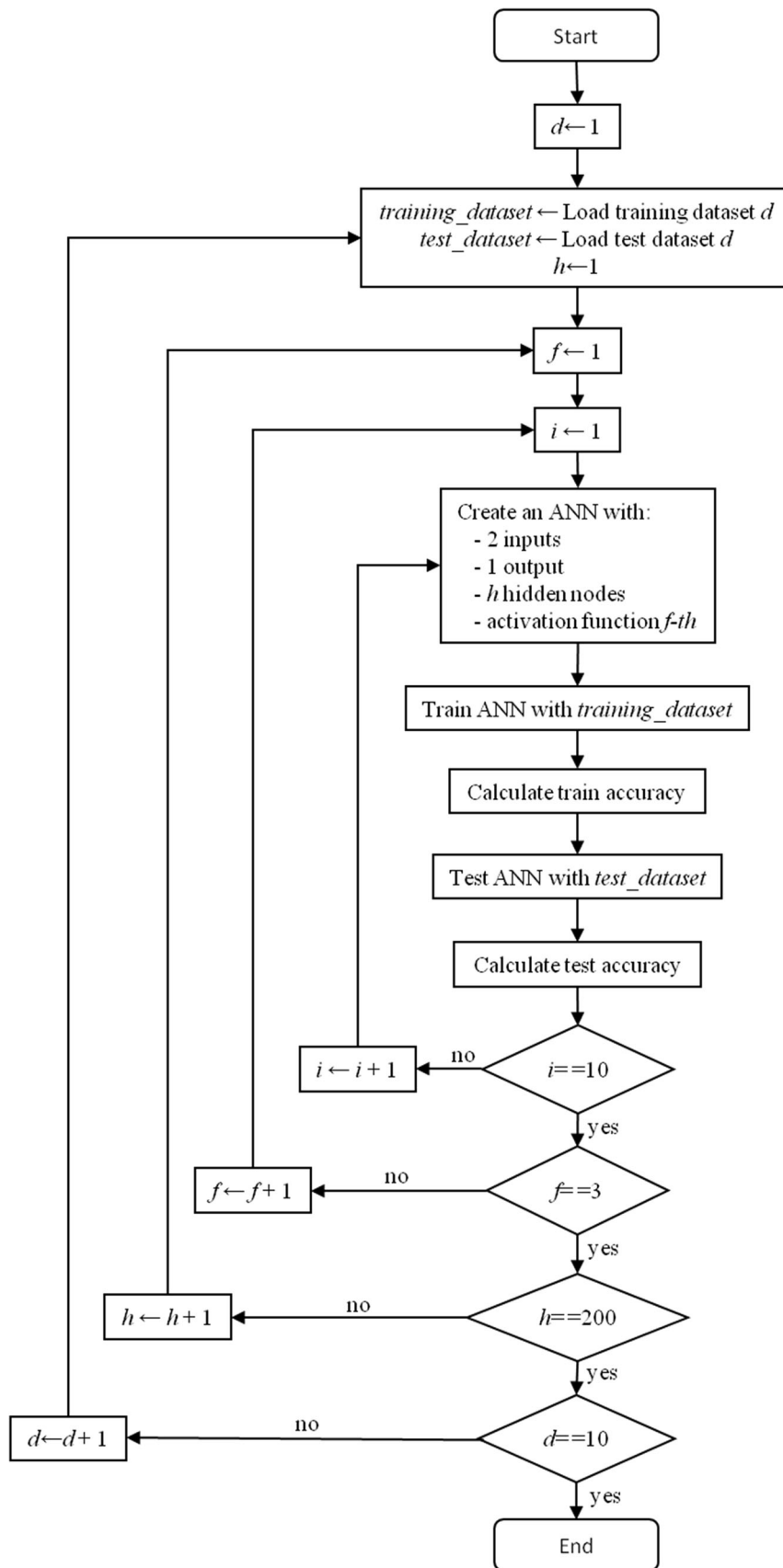
Train the $i - th$ ANN using the train dataset,

Obtain accuracy values of the ANN

Test the $i - th$ ANN using the test dataset

Obtain accuracy values of the ANN

Algorithm 1: FLC model learning procedure (Pseudocode form)



Algorithm 2: FLC model learning procedure (Flowchart form)

Table III. Performance results of the best ANN for each combination of normalization and noise possibility, merging all available combinations of hidden nodes and activation functions

Norm	Noise (%)	Acc. (10^{-6})	Error _{max} (10^{-3})	$\overline{\text{error}}$ (10^{-3})	σ_{error} (10^{-3})	Nodes	Funct.
No	0	3.47	16.32	1.29	1.93	75	tansig
	1	6.12	33.50	1.54	2.48	75	tansig
	2	10.30	38.30	2.07	3.21	49	tansig
	5	32.10	33.31	3.93	5.68	18	tansig
	10	79.50	72.50	5.68	8.91	45	tansig
Yes	0	3.47	16.32	1.29	1.93	75	tansig
	1	28.70	31.64	3.54	5.36	63	tansig
	2	71.70	41.51	5.54	8.47	100	tansig
	5	213.85	71.71	10.12	14.64	26	logsig
	10	430.18	88.87	15.26	20.78	100	logsig

Table IV. Performance results of the best ANN structure (defined by the number of hidden nodes and activation function) for each combination of normalization and noise possibility

Norm	Noise (%)	Acc. (10^{-6})	σ_{Acc} (10^{-3})	Error _{max} (10^{-3})	σ_{errormax} (10^{-3})	Nodes	Funct.
No	0	33.30	12.20	306.50	300.15	167	logsig
	1	43.60	7.44	302.86	292.24	91	logsig
	2	51.20	5.94	309.66	327.50	91	logsig
	5	68.80	7.00	315.94	322.68	170	logsig
	10	126.18	31.00	315.87	310.70	167	logsig
Yes	0	38.50	67.60	317.68	310.68	195	logsig
	1	51.50	12.60	315.10	306.49	150	tansig
	2	97.00	13.30	314.89	306.46	126	logsig
	5	259.62	7.52	312.39	304.19	123	tansig
	10	498.14	20.50	313.06	302.17	141	logsig

Experimental Setup

This subsection describes some considerations about the model learning process. The algorithm that has driven the process is unique, and it has been represented in its pseudo-code form (Algorithm 1) and in its flowchart form (Algorithm 2):

- We have partitioned each one of the 10 modified datasets into train, validation and test datasets using interleaved indices, distributing 60% of the samples for training, 20% for validation and the last 20% for testing.
- Regarding the hidden layer size, we have trained ANNs with a different number of nodes, trying all natural numbers $h \in [1, 200]$.
- The activation function of the output layer is linear, but different activation functions $f \in \{\text{tan-sigmoid}, \text{log-sigmoid}, \text{linear}\}$ have been tried in the hidden layer.
- We have executed five different training trials for each resulting combination of number of nodes in the hidden layer and activation function. For each one of these trials, we have measured different performance indexes in order

to study the behavior of each individual ANN and the behavior of each generic ANN structure.

- Finally, regarding the training algorithm that has been used, we have chosen the Levenberg–Marquardt algorithm for training due to its speed capacity. All the input vectors are presented once per iteration in a batch.

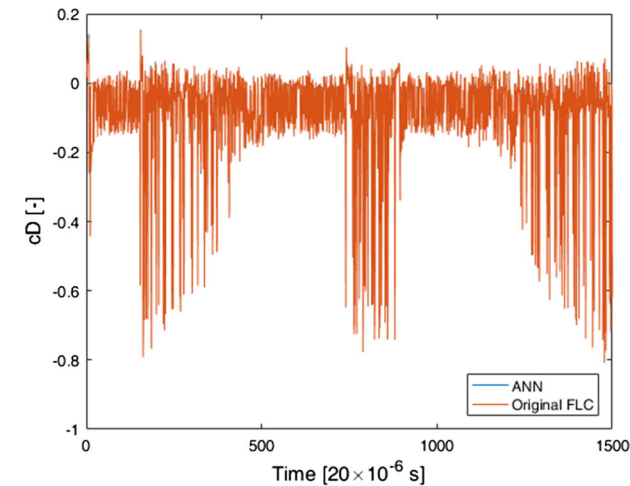
EXPERIMENTAL RESULTS

In this section, we discuss the experimental results obtained for the modelization task that we are facing, following the experimental design described in the previous section.

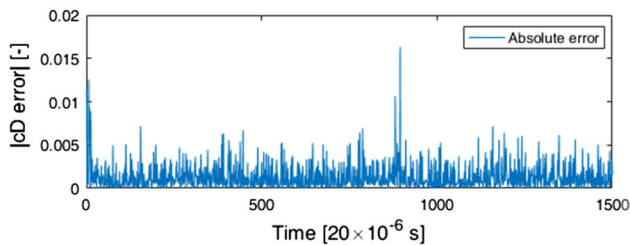
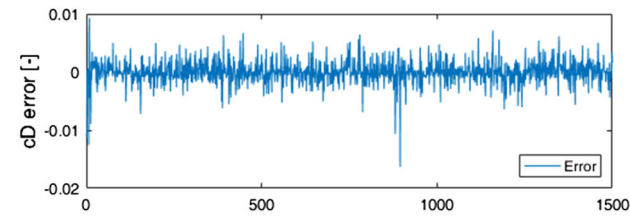
In Table III, we show the best ANN for each combination regarding the normalization and noise addition possibilities to the training patterns, according to the MSE accuracy criteria. It is important to recall that, as we explained in the description of the experimental design, five different trials were carried out for each of these combinations, merging them with activation functions and hidden nodes. The information about each ANN describes:

Table V. Characterization and performance of the best ANN for the test patterns

Norm	No
Noise	0%
Nodes	75
Funct.	tansig
Epoch	786
$Acc_{train}^{(10^{-6})}$	2.86
$Acc_{val}^{(10^{-6})}$	3.94
$Acc_{test}^{(10^{-6})}$	3.47
$error_{test_{max}}^{(10^{-3})}$	16.32
$\overline{error}_{test}^{(10^{-3})}$	1.29
$\sigma_{error_{test}}^{(10^{-3})}$	1.93

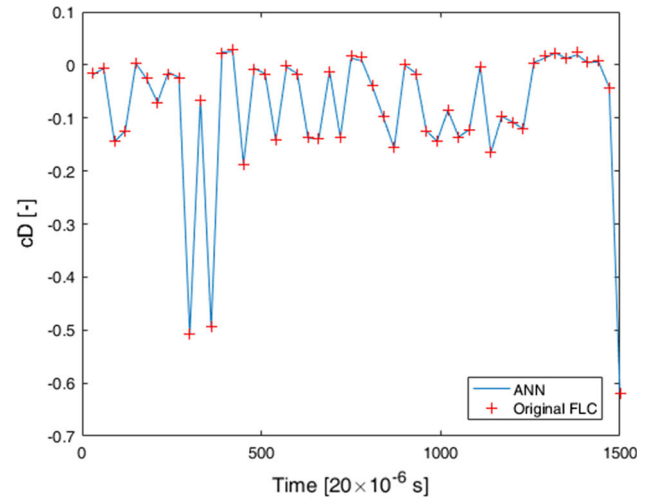


(a)

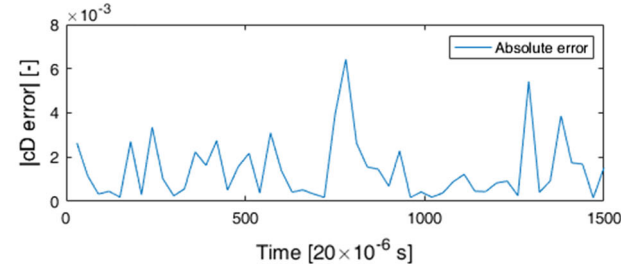
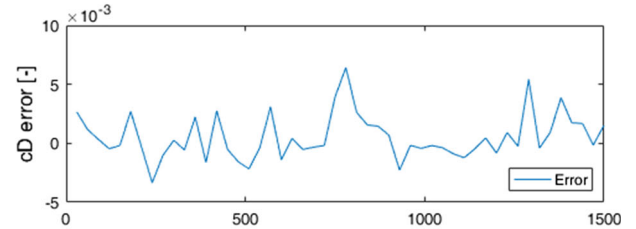


(b)

Fig. 5. (a) Original FLC signal and the output of the ANN for the test dataset of 1500 samples. (b) Error and absolute error of case (a). The X axis scale is 20×10^{-6} s.



(a)



(b)

Fig. 6. (a) Original FLC signal and the output of the ANN for the test dataset of 1500 samples with a subsampling ratio of 1:30. (b) Error and absolute error of case (a). The X axis scale is 20×10^{-6} s.

- *Norm* : Yes if the dataset used for training has been normalized, No otherwise
- *Noise* : is the percentage of noise added to the training patterns (0% means no noise addition)
- $Acc^{(10^{-6})}$: is the accuracy (MSE) obtained for the patterns of the test dataset, in 10^{-6}
- $error_{max}^{(10^{-3})}$: is the maximum error obtained for the patterns of the test dataset, in 10^{-3}
- $\overline{error}^{(10^{-3})}$: is the mean of the errors obtained for the patterns of the test dataset, in 10^{-3}
- $\sigma_{error}^{(10^{-3})}$: is the standard deviation of the errors obtained for the patterns of the test dataset, in 10^{-3}
- *Nodes* : is the number of nodes of the hidden layer

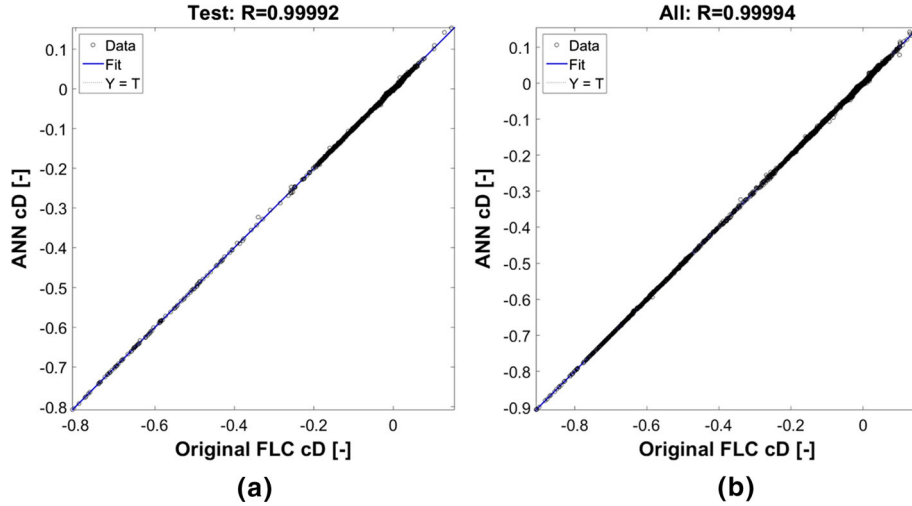


Fig. 7. Regression coefficients for the (a) test dataset and (b) the entire dataset.

- *Funct.* : is the activation function of the hidden layer (the output layer is always kept as linear)

In this table, we can see that in the modelization problem that we are facing, carrying out normalization and noise addition processes does not improve the results, obtaining worse results according to the accuracy criteria. It is also clear that adding more hidden nodes does not improve the results and that the tan-sigmoid activation function has obtained the best results in 80% of cases.

Given that Table III shows only the best ANN for each combination regarding the normalization and noise addition possibilities, it is convenient to analyze if those results are circumstantial or structural. So, Table IV provides a summary of results regarding the ANN structure that best fits the modeling problem under each one of these combinations of normalization and noise addition, taking into account the five different initializations done for each ANN structure (combination of nodes and activation function of the hidden layer). The data provided in Table IV is similar to data in Table III with the following additions:

- $\overline{Acc}^{(10^{-6})}$: is the mean of the accuracy (MSE) obtained for the test dataset patterns for the five initializations of that structure, in 10^{-6}
- $\sigma_{Acc}^{(10^{-6})}$: is the standard deviation of the accuracy (MSE) obtained for the test dataset patterns for the five initializations of that structure, in 10^{-6}
- $\overline{error}_{max}^{(10^{-3})}$: is the mean of the maximum errors obtained for the test dataset for the five initializations of that structure, in 10^{-3}
- $\sigma_{error_{max}}^{(10^{-3})}$: is the standard deviation of the maximum errors obtained for the test dataset

- patterns for the five initializations of that structure, in 10^{-3}

After analyzing Table IV we can see that the results are clearly worse than those of Table III, which is absolutely logical since these are mean values for a structure. The results show that for this modeling problem, it is better not to normalize patterns because, for the same noise addition, the results are worse with the normalized version. Also, it is clear that it is not convenient to add noise because the accuracy decreases as the added noise percentage increases. The first relevant difference is that the best structure for each case is much larger than the best individual ANN for that case indicating that there are relevant differences among the trials for a given structure, while the second one is that the log-sigmoid is the best activation function for 80% of cases and the tan-sigmoid function is the best for 20% of cases, just the opposite than for the individual ANNs. So, we can conclude that, in general, the best ANNs compiled in Table III are the fruit of good randomized initializations but do not correspond to fixed structures.

Once the best ANN has been detected through the analysis of Table III and it is clear that there is not any structural tendency, we analyze its performance in a deeper way. Table V gives further insight on the structure and performance of the best ANN for this modeling problem, showing the following data:

- *Epoch*: the epoch number at which the training process stopped
- $Acc_{train}^{(10^{-6})}$: is the obtained accuracy (MSE) for the training patterns, in 10^{-6}
- $Acc_{val}^{(10^{-6})}$: is the obtained accuracy (MSE) for the validation patterns, in 10^{-6}

- $\text{Acc}_{\text{test}}^{(10^{-6})}$: is the obtained accuracy (MSE) for the test patterns, in 10^{-6}
- $\text{error}_{\text{test}_{\text{max}}}^{(10^{-3})}$: is the maximum obtained error for the test patterns, in 10^{-3}
- $\overline{\text{error}_{\text{test}}}^{(10^{-3})}$: is the mean of the obtained errors for the test patterns, in 10^{-3}
- $\sigma_{\text{error}_{\text{test}}}^{(10^{-3})}$: is the standard deviation of obtained errors for the test patterns, in 10^{-3}

Given the statistical data of the working dataset shown in Table II, we can conclude that the selected ANN has obtained a very good model approximation for the FLC controller, with an MSE of 3.47×10^{-6} and a maximum error of 16.32×10^{-3} for test data. In order to obtain a more intuitive view of the performance of this ANN, Fig. 5 shows its behavior during the 1500 samples of the test dataset. More specifically, Fig. 5a shows the almost perfect overlapping between the cD signals of the original FLC and the output of the ANN (in such a way that one can see only the signal drawn in second place), while Fig. 5b shows the error and the absolute error between them. With the purpose of offering a better understanding of the fitting, in Fig. 6, we have done a subsampling with a ratio of 1:30 inside the 1500 samples of the test dataset to gain a clearer view, in such a way that Fig. 6a shows a subset of the points of Figs. 5a and 6b a subset of Fig. 5b.

Finally, we assess the learning of the selected ANN through the regression coefficient R of our approach for two cases. The first one is shown in Fig. 7a where a value $R = 0.99992$ has been obtained for the test dataset, while a value $R = 0.99994$ for all the dataset is shown in Fig. 7b. Taking into account that a regression coefficient $R = 1$ means a perfect fitting between the original signal and the model, we can conclude that the obtained model is extremely accurate.

CONCLUSIONS

In this paper, we focused on one relevant application of the electronic materials, i.e., on PV energy generation and on how the PV elements implemented with these materials need some components (a converter and a control) to obtain their maximum power. More specifically, we reviewed the state of the art on PV elements modeling, controllers as QBC and MPPT algorithms, all of them as enabling technologies to obtain the maximum available energy from electronic materials. We also recalled a specific FLC design and posed how to model it using an ANN, with an experimental design that was used successfully in such a way that an accurate model of the specific FLC was obtained since the MSE is 3.47×10^{-6} with a maximum error 16.32×10^{-3} for test data and a regression coefficient $R = 0.99992$ for the test dataset. This leads to much simpler implementations of converters based

on MPPT algorithms with fuzzy controllers and to an enhanced simplicity of the overall system for the same power production.

ACKNOWLEDGEMENTS

The authors would like to thank to Alicia Lopez-Guede and Daniel Bray for their support with the English translation of the paper. The authors are also grateful to the European Union Ministry of Turkey, National Agency of Turkey for the support of this project under the Project Code: 2015-1-TR01-KA203-021342 entitled Innovative European Studies on Renewable Energy Systems.

REFERENCES

1. T. Mukai, M. Tomasella, A. Parlikad, N. Abe, and Y. Ueda, *IEEE Trans. Sustain. Energy* **5**(4), 1176 (2014).
2. EnergyTrend. PV spot price. EnergyTrend, a Business Division of TrendForce Corp (2017). <http://pv.energytrend.com/pricequotes.html>. Access 19 May 2018.
3. A. Anurag, S. Bal, S. Sourav, and M. Nanda, *Int. J. Sustain. Energy* **35**(5), 478 (2016).
4. A. Khateb, N. Rahim, J. Selvaraj, and M. Uddin, *IEEE Trans. Ind. Appl.* **50**(4), 2349 (2014).
5. N. Altin, *Adv. Electr. Comput. Eng.* **13**(3), 65 (2013).
6. S. Ozdemir, N. Altin, and I. Sefa, *Int. J. Hydrog. Energy* (2017). <https://doi.org/10.1016/j.ijhydene.2017.02.191>.
7. Y. Alcazar, D. de Souza Oliveira, F. Tofol, and R. Torricobascope, *IEEE Trans. Ind. Electron.* **60**(10), 4438 (2013).
8. O. Lopez-Santos, L. Martinez-Salamero, G. Garcia, H. Valderrama-Blavi, and D. Zambrano-Prada, *IEEE Trans. Power Electron.* **32**(3), 2253 (2017).
9. B. Axelrod, Y. Berkovich, and A. Ioinovici, in *IEEE 32nd Annual Conference on Proceeding of Industrial Electronics, IECON*, vol. 1, p. 2366 (2006).
10. J. Leyva-Ramos, M. Ortiz-Lopez, L. Diaz-Saldierna, and J. Morales-Saldana, *IET Power Electron.* **2**, 605 (2009).
11. Y. Hsieh, J. Chen, T. Liang, and L. Yang, *IET Power Electron.* **5**(1), 11 (2012).
12. T. Yan, J. Xu, Z. Dong, L. Shu, and P. Yang, in *2013 International Conference on Communications, Circuits and Systems (ICCCAS)*, Chengdu, China, pp. 402–406 (2013). <https://doi.org/10.1109/ICCCAS.2013.6765367>.
13. N. Altin and E. Ozturk, in *ECAI, 2016—International Conference—8th Edition Electronics, Computers and Artificial Intelligence 30 June–02 July, 2016* (Ploiesti, Romania, 2016).
14. R. Kadri, J.-P. Gaubert, G. Champenois, and M. Mostefai, in *Proceedings of the 19th International Conference on Electrical Machines (ICEM '10)*, Rome, Italy (2010).
15. M. Green, Y. Hishikawa, W. Warta, E. Dunlop, D. Levi, J. Hohl-Ebinger, and A. Ho-Baillie, *Prog. Photovolt.* **25**(7), 668 (2017).
16. J. Ramos-Hernanz, J. Lopez-Guede, I. Zamora-Belver, P. Eguia-Lopez, E. Zulueta, O. Barambones, and F. Oterino-Echavarrri, *Int. J. Tech. Phys. Probl. Eng. (IJTPE)* **6**(4), 37 (2014).
17. J.M. Lopez-Guede, J.A. Ramos-Hernanz, E. Zulueta, U. Fernandez-Gamiz, and F. Oterino, *Int. J. Hydrog. Energy* **41**(29), 12672 (2016). <https://doi.org/10.1016/j.ijhydene.2016.04.175> (Special Issue on 3rd European Conference on Renewable Energy Systems (ECRES 2015), 7-10 October 2015, Kemer, Antalya, Turkey).
18. J.M. Lopez-Guede, J.A. Ramos-Hernanz, E. Zulueta, U. Fernandez-Gamiz, and G. Azkune, *Int. J. Hydrog. Energy* **42**(28), 18103 (2017). <https://doi.org/10.1016/j.ijhydene.2017.02.062>. <http://www.sciencedirect.com/science/article/pii/S0360319917305372> (Special Issue on The 4th European Conference on Renewable Energy Systems (ECRES 2016), 28–31 August 2016, Istanbul, Turkey).

19. O. Abutbul, A. Gherlitz, Y. Berkovich, and A. Ioinovici, *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.* **50**(8), 1098 (2003).
20. J. Rosas-Caro, J. Ramirez, F. Peng, and A. Valderrabano, *ET Power Electron.* **3**(1), 129 (2010).
21. D. Wijeratne and G. Moschopoulos, *IEEE Trans. Circuits Syst. I Reg. Pap.* **59**(2), 426 (2012).
22. V. Salas, E. Olias, A. Barrado, and A. Lazaro, *Solar Energy Mater. Sol. Cells* **90**(11), 1555 (2006).
23. T. Eswam and P. Chapman, *IEEE Trans. Energy Convers.* **22**(2), 439 (2007).
24. N. Altin and S. Ozdemir, *Energy Convers. Manag.* **69**, 17 (2013).
25. B. Widrow and M. Lehr, *Proc. IEEE* **78**(9), 1415 (1990). <https://doi.org/10.1109/5.58323>.
26. K. Narendra and K. Parthasarathy, *IEEE Trans. Neural Netw.* **1**(1), 4 (1990). <https://doi.org/10.1109/72.80202>.
27. J.M. Lopez-Guede, J.A. Ramos-Hernanz, E. Zulueta Guerrero, and U. Fernandez-Gamiz, in *The Proceedings of Third European Conference on Renewable Energy Systems-ECRES 2016*, ed. by E. Kurt, p. 885 (2016).
28. J.M. Lopez-Guede, J.A. Ramos-Hernanz, E. Zulueta Guerrero, and U. Fernandez-Gamiz, in *The Proceedings of Third European Conference on Renewable Energy Systems-ECRES 2015*, ed. by E. Kurt (2015).