Data Driven Performance Models for Photovoltaic Modules

Joaquim Luís

***Abstract*—With the growth of photovoltaic (PV) global market, and the rising level of penetration of PV within electric grids, the need for an efficient economical and technical integration with the system becomes paramount. Due to the variability of solar irradiance, one of the greatest challenges to successfully integrate PV lies in the ability to accurately predict PV power for different sky conditions. This work studies using Artificial Intelligence (AI) techniques, namely Artificial Neural Networks (ANN) and Adaptive Neuro Fuzzy Inference Systems (ANFIS), to predict PV power in clear and cloudy sky conditions, for forecast resolutions of 1 and 10 minutes, using data collected from three PV modules. The data consists of five days of clean sky conditions and five days of partially cloudy sky. The networks are fed with multiple sets of inputs: global irradiance, global irradiance coupled with the PV cell temperatures, and global irradiance coupled with past output PV power. The networks are trained using four days of the dataset and a prediction is made for one day for each sky condition. To evaluate the model's performance Mean Percentage Absolute Error (MAPE) is used. Both networks presented acceptable results but using both exogenous and endogenous inputs (global irradiance and power) yielded the best results for both networks. ANN outperformed ANFIS for clear sky for 1 minute resolution and cloudy sky for 10 minutes, while ANFIS was better in predicting with 1 minute resolution for cloudy sky and 10 minutes resolution for clear sky.**

***Index Terms*—Adaptive neuro fuzzy system, artificial neural network, forecasting, photovoltaic, PV power.**

I. Introduction

ACCORDING to the latest report of the International Agency Photovoltaic Power Systems Programme (IEA PVPS) [1], 2016 was a record year for global installed PV power, with a registered growth of around 50 %, reaching 76 GW for the first time ever. Because solar irradiance is highly variable and, since most of this installed capacity is connected to the electrical grid, the proper integration of PV plants is crucial for the proper functioning of the electrical system. Despite being prepared for load variation, the electrical grid requires special adaptations when it comes to renewable energy. Electric system operators must deploy a range of strategies in order to accommodate renewable generation while maintaining high levels of reliability. The advancements in forecasting technology, together with better data on historical performance of renewable energy, allows significantly improved forecasting accuracy of renewable energy, which can help system operators for better demand response.

Since that for a solar panel at fixed temperature, the power production is approximatly linearly dependent on global irradiance, it can be expected that forecasting power will be similar to forecasting solar irradiance, so the forecasting techniques are diverse and range from measured weather and PV data, to satellite and sky imagery, to numerical weather prediction (NWP) models. The performance of these techniques varies with the forecast horizon considered. Data based models perform best for very short-term (0 to 6 hours ahead), while NWP models are better suited for forecast horizons beyond 6 hours [2].

For very short-term forecasting, cloud dynamics must be accounted for and, in order to use linear models, global irradiance must be separated into its deterministic and stochastic components. The first is easily modelled, since it depends on the sun position in the sky throughout the day, while the second originates from cloud movement, both at high and low altitudes, and which is highly non-linear. Due to the limitations of linear models, a great deal of attention has been given to non-linear models, particularly to Artificial Intelligence (AI) techniques. These are capable of learning patterns from data without the need for complex mathematical models, and have proven powerful and flexible for forecasting applications [3].

This work aims to study Artificial Neural Networks (ANN) and Adaptive Neuro Fuzzy Inference Systems (ANFIS) for forecasting PV power for a very short-term time horizon. The networks parameters are optimized for the data available, and results for clear and cloudy sky are compared, as well as for time horizons of 1 and 10 minutes. Their performance is evaluated using the Mean Absolute Percentage Error (MAPE).

II. Artificial Neural Networks

ANNs are inspired by the neural structure of the brain, where a network of connected basic units, called neurons, process information by firing synapses. ANNs are a mathematical simplification of such systems, where a simple neuron is composed by an input multiplied by a weight and added a bias, and runs through a non-linear activation function (Figure 1). A feed-forward ANN is the most widely used

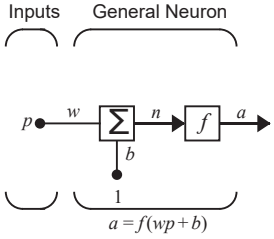

Fig. 1. Simple neuron (from [4].

Joaquim Luís is with Instituto Superior Técnico, Universidade de Lisboa, Portugal (e-mail: joaquim.luis@tecnico.ulisboa.pt).

network architecture, and is comprised of several layers of interconnected neurons, each of which is connected to other neurons in the next layer. Data is presented to the network, processed through one or more hidden layers and through an output layer, where the network response is computed. Layer inputs are processed by multiplying each input by its weight, summing the product, and transformed by a non-linear transfer function to produce an output.

Another important architecture frequently used in time series forecasting is the Non Linear AutoRegressive with eXogenous inputs (NARX) network. This is a feedback network, meaning that its output is fed back as input. The NARX model is defined by the equation:

$$y(t) = f(y(t-1), y(t-2), \cdots, y(t-n_y), \\ u(t-1), u(t-2), \cdots, u(t-n_u)), \quad (1)$$

where y is the network output, u is the input and n_y and n_u are the time lags.

An ANN can perform a variety of tasks, depending entirely on the data with which is fed. The ANN is capable of learning patterns in data using known inputs/targets pairs to adjust the weights between neurons using a backpropagation algorithm. The backpropagation algorithm aims to minimize a cost function, that measures the network error. The inputs are processed through the network, which is initialized with random initial weights, and generates an output. The error is computed and backpropagated through the network, updating the weights and bias in the direction of the steepest gradient descent. However this method can become trapped in local minima. The Marquardt-Levenberg algorithm [5] is an improvement to the backpropagation algorithm and consists in modifying the Newton method by solving

$$\Delta \mathbf{x} = \left[J^T(\mathbf{x})J(\mathbf{x}) + \mu \mathbf{I}\right]^{-1} J^T(\mathbf{x})\mathbf{e}(\mathbf{x}) \quad (2)$$

where $V(\mathbf{x})$ is the cost function, $J(\mathbf{x})$ the Jacobian, μ the step size and $\mathbf{I}$ the identity matrix, avoiding the computational expensive task of computing the Hessian matrix.

III. Adaptive Neuro Fuzzy Inference Systems

Adaptive neuro fuzzy inference systems [6] combine fuzzy inference systems (FIS) with ANN. FIS employ logic rules based on natural language to capture the qualitative aspects of human knowledge and its reasoning process. These rules are expressions of the form *IF* A *THEN* B, where A and B are linguistic labels of fuzzy sets [7], characterized by appropriate membership functions. For example, a simple fact can be described as

If $pressure$ *is high, then* $volume$ *is small*

where $pressure$ and $volume$ are linguistic variables and $high$ and $small$ linguistic labels that are characterized by membership functions.

Takagi and Sugeno [8] proposed another form for fuzzy if-then rules, where fuzzy sets are only used in the premise part, . For example, the consequent part of

If $velocity$ *is high, then* $force = k \times (velocity)^2$

is described by an nonfuzzy linear equation of the input variable.

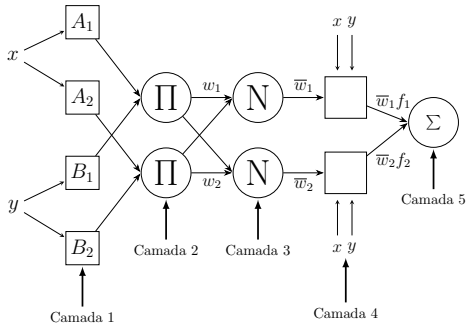

Fig. 2. ANFIS architecture

The ANFIS architecture is shown in Figure 2. This system has two inputs, x and y, and two Takagi-Sugeno rules, R_1 and R_2 of the form

R_1: If x is A_1 and y is B_1, then $f_1 = p_1 x + q_1 x + r_1$,

R_2: If x is A_2 and y is B_2, then $f_2 = p_2 x + q_2 x + r_2$.

The network output is given by

$$output = \sum_i^N \overline{w_i} f_i = \frac{\sum w_i f_i}{\sum w_i} \quad (3)$$

where w_i and f_i are the i-th rule's weight and the consequent transfer function, respectively.

The parameters to be adjusted in an ANFIS network are the rule's weights and the consequent linear equation's parameters. ANFIS training uses a hybrid approach, where backpropagation learning algorithm is used to determine the rule's weights, and the least squares method to compute the linear equation parameters.

Two of the essential tasks for the construction of a FIS are: structure learning and parameter learning. The first is associated with the selection of rules and membership functions that characterize the linguistic variables, while the second comprises the adjustment of the parameters of membership functions and the linear transfer functions of Takagi-Sugeno rules. The selection of proper rules and membership functions is often done in a subjective manner, based on the knowledge and experience of some specialist. There are, however, ways to automatically generate rules and membership functions based on data. Data clustering algorithms can be used to find relationships in the data, and group together similar data points around a center point, denominated by cluster center, that can be used as fuzzy rules. As for parameter learning, the learning capability of an ANN can fine tune the randomly generated system parameters.

To produce the set of rules subtractive clustering is employed [9]. This algorithm uses data points as candidates for cluster centres. The greater density of data points within some defined radius surrounding a candidate point, the greater its potential to be chosen as a cluster centre, where a cluster is a group of similar data points. The potential of a data point x_i is defined as

$$P_i = \sum_{j=1}^{n} e^{-\alpha \|x_i - x_j\|^2}, \quad (4)$$

where

$$\alpha = \frac{2}{r_a^2}$$

and r_a is a positive constant that defines the radius. After the potential of all point is computed, the candidate with the greatest potential is chosen as the first cluster centre. The potential of all other points is updated, according to

$$P_i \Leftarrow P_i - P_1^* e^{-\beta \|x_i - x_1^*\|^2} \tag{5}$$

where

$$\beta = \frac{4}{r_b^2}$$

is a positive constant and defines the radius inside which the potential of other data points is greatly reduced, so that the centre clusters are evenly spread across the data. This process is repeated until the potential reaches some threshold or a given number of clusters is attained. Each cluster centre x_i^* is considered as a fuzzy rule that describes the system behaviour.

IV. Data

The data collected consists of global irradiance measurements, G (W/m^2), module cell temperature, T_c (°C), and DC output power, P (W). This data is for three modules, codenamed O1, N1 and E1, where the O1 and E1 modules are 220 W_p and the N1 is 250 W_p. There's 10 days worth of data, with 5 days in winter time, from 1–5 of January and the other 5 days in summer time, from 1–5 of August. All measurements have minute to minute resolution. From figures 3 and 4 one can observe that the days of January are clear sky days, while the days of August present some cloudiness, especially on day 3. For that reason, the data will be divided according to the sky conditions, clear and cloudy, where four days will be the training set and one day will be the testing set, with a forecasting window of 7 hours. The power forecast for clear sky corresponds to the 4th of January, and the other days of that month will be used for training. Because it's winter, the power reaches significant values only around 10 h, so for clear sky the forecasting window will be from 10 h to 17 h. As for cloudy sky, the 3rd of August is an overcast day therefore this will be the day to forecast. Since it's summer, the sun rises earlier and by 9 h the PV modules already have considerable power output, so the forecast window will be from 9 h to 16 h.

The error measured used to evaluate the results will be the Mean Absolute Percentage Error (MAPE):

$$APE_p = \left| \frac{P_{p_{\text{measured}}} - P_{p_{\text{predicted}}}}{P_{p_{\text{measured}}}} \right| \times 100\,\% \tag{6}$$

$$MAPE = \frac{1}{P} \sum_{p=1}^{P} APE_p \,. \tag{7}$$

In order to determine the proper time lags to apply to the input variables, a partial autocorrelation function analysis was performed. The results showed strong correlation for lags 0 and 1 for all variables, thus the time delay to apply is 1. This means that for a forecast at time t, the corresponding time series input value is at $t-1$.

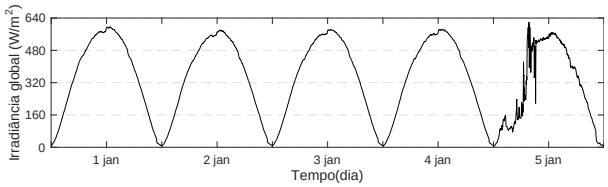

Fig. 3. Global irradiance for 1–5 of January.

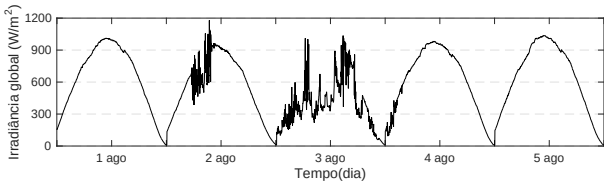

Fig. 4. Global irradiance for 1–5 of August.

V. ANFIS parameters

Data was normalized to the range $[-1, 1]$ using *minmax* normalization,

$$y = \frac{(y_{\max} - y_{\min}) \times (x - x_{\min})}{x_{\max} - x_{\min}} + y_{\min} \,. \tag{8}$$

and randomly divided into training and validating sets: 85 % for the first and 15 % for the second. A Takagi-Sugeno FIS was generated using Matlab's *genfis2* function, that uses subtractive clustering for the construction of premises fuzzy rules and membership functions. The radius for subtractive clustering was determined through trial and error, and it was verified that the default value of 0.55 was appropriate. Input membership functions are Gauss functions, derived from the clustering method, and output membership functions are linear, since it's a Takagi-Sugeno type FIS. The network was trained to tune its parameters using Matlab's *anfis* function, with 300 epochs for training. Subtractive clustering generated two rules for cloudy sky with input G, and three rules for all other combinations.

VI. ANN parameters

For ANN forecasting two architectures were used. A time delay network, that is fed exogenous inputs, which was used for forecasting with G and G, T_c; and a NARX network, for forecasting with G, P. Data was normalized using *minmax* normalization using (8) and divided in 70 % training, 15 % validation and 15 % testing.

An ANN architecture is mainly defined by the number of hidden layers, the number of neurons in the hidden layers and the transfer functions. For the number of hidden layers, it's well established that an ANN with only one layer is capable of fitting any function with an arbitrary level of accuracy, as long as it has sufficient hidden neurons [10].

As for the selection of hidden neurons, a trial and error approach was used, as there's no standard method and it's highly dependent on the data set. In order to optimize the number of hidden neurons, the MAPE of forecasts was evaluated for different numbers of hidden neurons, ranging from 1 to 20. For each number of hidden neurons tested, 10 forecast were performed and the average taken. It was concluded that

the MAPE does not change significantly with the number of hidden neurons so, for simplicity, only one hidden neuron is used.

The transfer function for the hidden layer is the hyperbolic tangent (*tansig*) and for the output layer the linear function *purelin*. The ANN is trained using the Marquardt-Levenberg algorithm. Table I summarises the ANN architecture used. Given that an ANN initial conditions are random and produce different results each time, a new network is initialized and trained for 20 runs for all sky conditions, input variables and modules. The best result is chosen for each different case and analysed in detail.

TABLE I
ANN ARCHITECTURE.

Parameter	Values
No. layers	3 (input,hidden,output)
No. hidden neurons	1
Time lag	1
Transfer functions	tansig, purelin
Learning algorithm	Marquardt-Levenberg

VII. RESULTS

A forecast was made for clear and cloudy sky, using as inputs three sets of variables: global irradiance G, global irradiance and PV cell temperature, G, T_c, and finally, global irradiance and PV power, G, P. The forecast is for a time horizon of 7 h with 1 minute and 10 minutes resolution, where each point is the 10 minute average of measured power.

Overall the best results were obtained using as input G, P for both networks. Results for different modules are similar, so all curves and MAPEs shown are for the module with the lowest MAPE, to facilitate analysis. Forecasts for clear sky show a lower MAPE than for cloudy sky. Results for 1 minute resolution are more accurate than for 10 minutes, except for the ANN with G, P as input, where MAPE decreased.

All results were obtained using MATLAB® Release 2015a and the Neural Network and Fuzzy Logic Toolboxes.

A. Clear sky

1) 1 minute resolution: Results for clear sky are shown in Table II. ANN performed better with G and G, P as inputs while ANFIS slightly outperformed ANN using G, T_c. The coupling of G with T_c and P decreased the error, as opposed to using G alone. Considering average values for all modules and inputs, the inclusion of T_c reduced MAPE by 22 % for ANFIS, and 12 % for ANN, and the inclusion of P resulted in an even greater reduction, of 59 % and 63 % for ANFIS and ANN, respectively.

Predicted power for ANFIS and ANN is depicted on figures 5 and 6, as well as the deviation from measured values, respectively. The predicted curve follows the real curve very closely. Looking at the error, a tendency of overestimation between 10 h and 13 h and underestimation after 13 h.

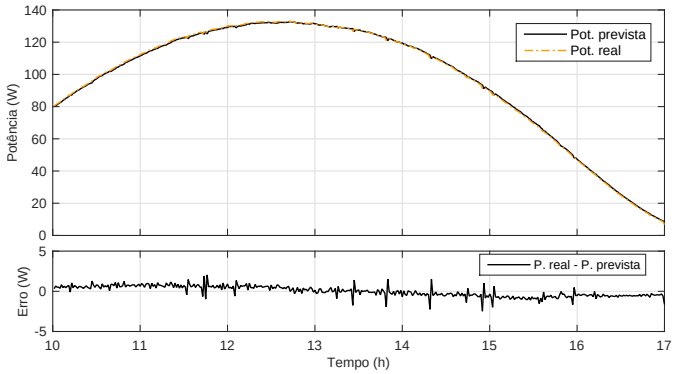

Fig. 5. Predicted power with ANFIS for clear sky with input G, P with 1 minute resolution.

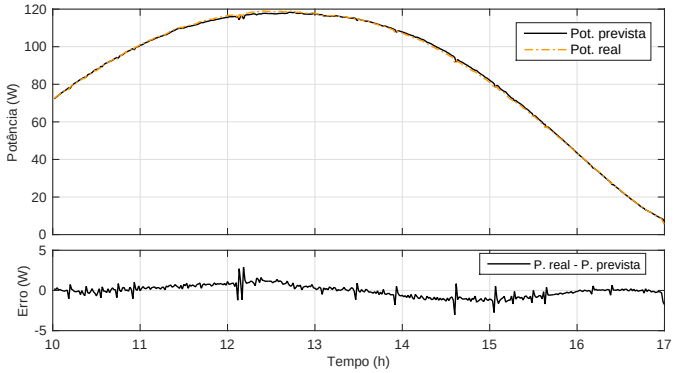

Fig. 6. Predicted power with ANN for clear sky with input G, P with 1 minute resolution.

2) 10 minute resolution: Predicting with a forecast resolution of 10 minutes yielded an increased error, as shown in Table III. Unlike the case before, ANFIS produced the best results. Again, including more input variables resulted in smaller error for both networks. Observing the predicted power curve for ANFIS, in Figure 7, it shows a bigger deviation from the measured power. Like before, there's a tendency to underestimate in the morning period and to overestimate in

TABLE II
BEST MAPE FOR CLEAR SKY (IN %).

Input	ANFIS	ANN
G	2.44	1.99
G, T_c	1.65	1.70
G, P	0.91	0.83

TABLE III
BEST MAPE FOR CLEAR SKY (IN %).

Input	ANFIS	ANN
G	8.94	8.01
G, T_c	4.69	5.80
G, P	3.07	4.90

the afternoon, around after 14 h. For ANN, the contrary is

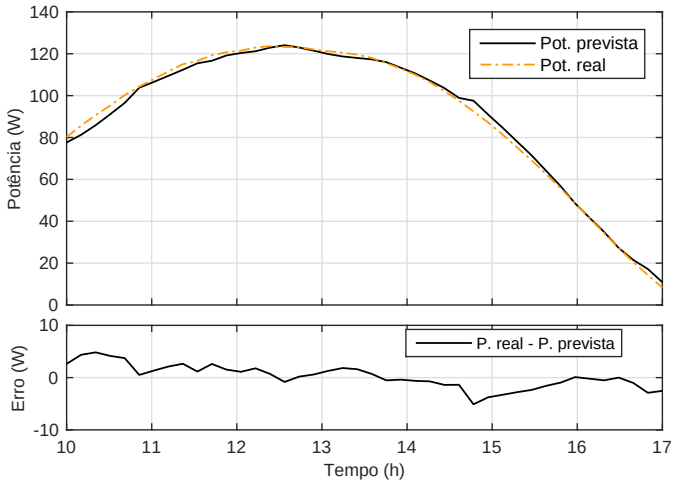

Fig. 7. Predicted power with ANFIS for clear sky with input G, P with 10 minute resolution.

seen, as the network shows a tendency to overestimate in the morning, to around 12 h and to underestimate after that.

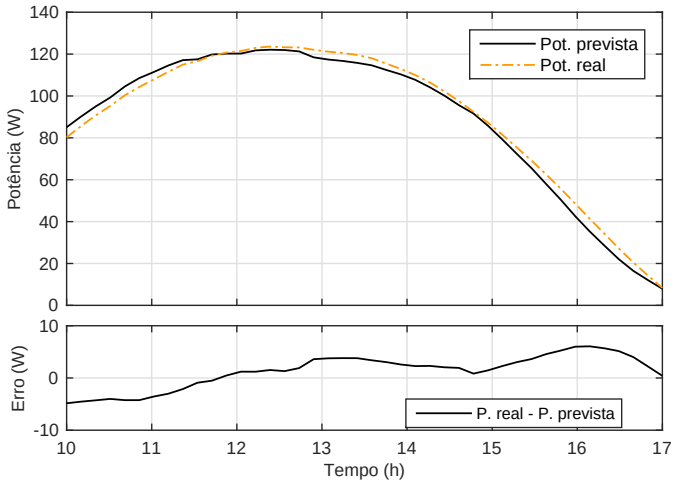

Fig. 8. Predicted power with ANN for clear sky with input G, P with 10 minute resolution.

B. Cloudy sky

1) 1 minute resolution: Table IV shows the results for ANFIS and ANN for cloudy sky. There's a considerable rise of the error comparing to the clear sky condition. ANN displays the same behaviour when T_c and P are included as inputs, where MAPE decreases. For ANFIS, however, there's a rise in error when T_c is included, compared to using just G. Apart from the result for G, T_c, ANFIS has achieved better results than ANN.

In figures 9 and 10, the curves for both ANFIS and ANN are quite similar, as well as the their errors. Errors are greater between 11 h and 12 h, and between around 15h30 and 16 h, where power variability is more prominent.

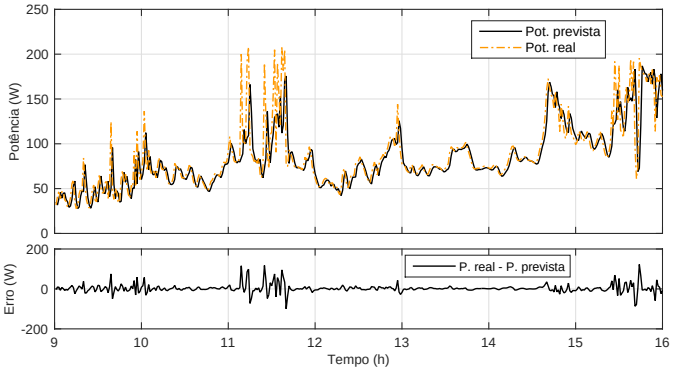

Fig. 9. Predicted power with ANFIS for cloudy sky with input G, P with 1 minute resolution.

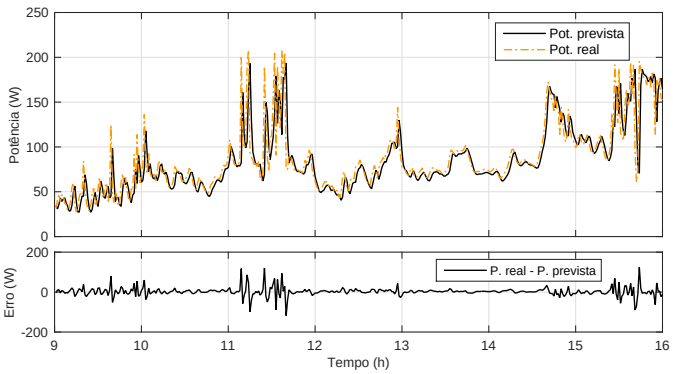

Fig. 10. Predicted power with ANN for cloudy sky with input G, P with 1 minute resolution.

2) 10 minute resolution: As shown in Table V, results for ANFIS and ANN using G and G, T_c have little difference between them. The addition of T_c to G resulted in a higher MAPE, whereas adding P improved results, especially for ANN, that greatly exceeds ANFIS. ANN's MAPE improved by 68 % with G, P in regard to G alone, while ANFIS registered an improvement of about 11 %.

TABLE IV
BEST MAPE FOR CLOUDY SKY (IN %).

Input	ANFIS	ANN
G	19.60	19.80
G, T_c	20.22	19.48
G, P	13.08	14.06

TABLE V
BEST MAPE FOR CLOUDY SKY WITH 10 MINUTE RESOLUTION (IN %).

Input	ANFIS	ANN
G	20.81	19.93
G, T_c	21.37	20.01
G, P	18.56	6.34

ANFIS predicted power curve, in Figure 11, presents a significant off set from the measured power, with the biggest

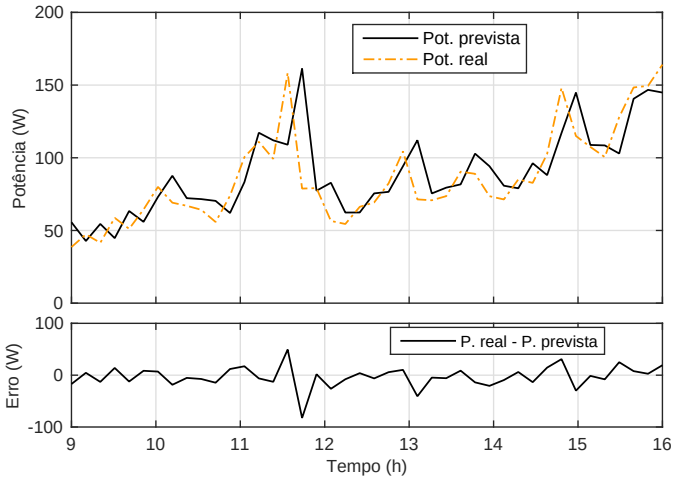

Fig. 11. Predicted power with ANFIS for cloudy sky with input G, P with 10 minute resolution.

displacement taking place between 11 h and 12 h. On the other hand, ANN predicted power follows the measured curve very closely, as indicative by its low MAPE.

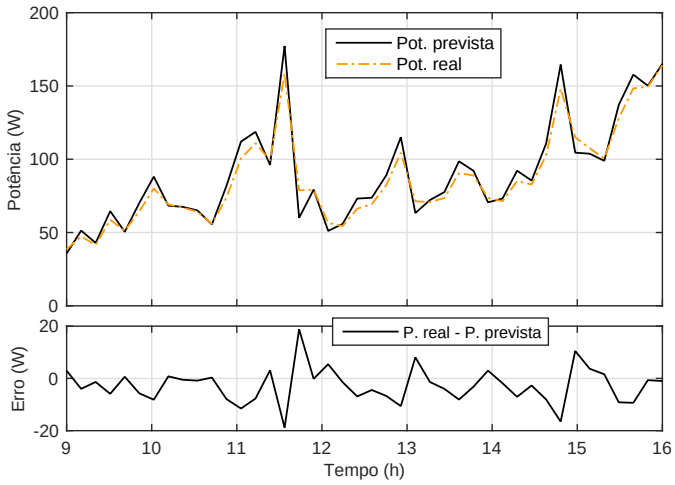

Fig. 12. Predicted power with ANN for cloudy sky with input G, P with 10 minute resolution.

VIII. APE CURVES

Figures 13 and 14 illustrate the APE for a forecast day. APE progression throughout the day is similar for both networks, so just one is depicted for each case. For the clear sky day, APE remains relatively low during most of the day, and starts to ramp up around 16h30. The clear sky day is the 3rd of January, in winter time. Due to the sunset and the corresponding drop of solar irradiance, the power drops and because APE is scaled to the measured power, it means that a low power value results in a high APE.

For the cloudy day (Figure 14), in summer time, the sun sets later and no ramping occurs at the end of the day. For 1 minute resolution (Figure 14a) the largest APE values take place between 9 h and 10 h, 11 h and 12 h and finally 15 h and 16 h, where the largest variations of power happen. For

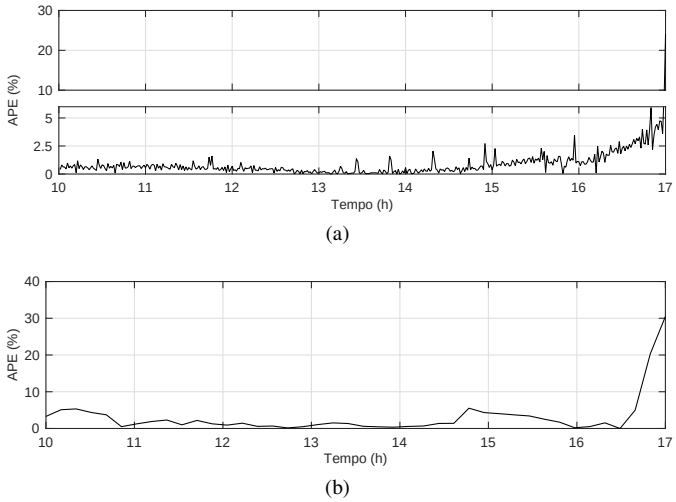

Fig. 13. APE for clear sky. (a) 1 minute resolution, (b) 10 minute resolution.

10 minute resolution (Figure 14b), the averages for 10 minute intervals are used and as result larger variations of power are smoothed.

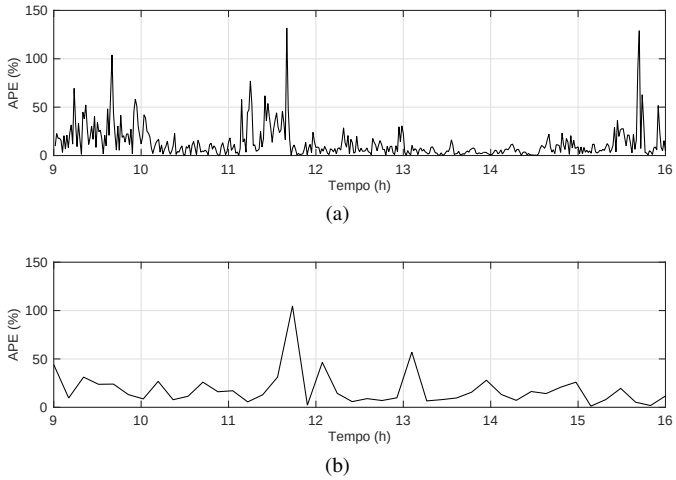

Fig. 14. APE for cloudy sky. (a) 1 minute resolution, (b) 10 minute resolution.

IX. CONCLUSIONS

This work studied two types of AI networks for the prediction of PV power for a forecast horizon of 7 h for both a clear and cloudy sky days, using data collected from three PV modules. These networks used 4 days of data for each sky condition and three different sets of inputs, G, G, T_c and G, P, in order to determine what combination of variables could better account for the strong non linearity caused by cloudy sky condition. PACF analysis concluded that a 1 minute ahead prediction was the most adequate for the data available. A prediction using the mean value for 10 minutes of measurements and that uses the 10 minutes before as input, was also performed for comparison purposes.

For ANFIS architecture, the selection of rules and membership function parameters were determined in an automated fashion, using the subtractive clustering algorithm.

ANN required the determination of the number of hidden layers and hidden neurons. Given that one hidden layer alone is sufficient for most problems, this was the number chosen for the ANN architecture. The number of hidden neurons was determined by trial and error, testing the network for a changing number of hidden neurons, ranging from 1 to 20. Results were very similar regardless of this number, so the simplest solution was adopted, the one with only a hidden neuron.

Forecasting a clear sky day reveals better results compared to a cloudy day. MAPE didn't exceed 9 % in the worst case for the first condition, averaging around 3.7 % (for both networks and considering all variables and time resolutions), while for a cloudy day MAPE averaged around 18 % . ANN displayed better results than ANFIS overall, although results were very similar.

The addition of input variables improved results. Using PV cell temperature T_c improved clear sky forecasts but had little effect for cloudy sky. For forecasts with 1 minute resolution inclusion of T_c showed MAPE improvements of 22 % for ANFIS and 12 % and for ANN. For 10 minute resolution the improvements were greater, of 48 % for ANFIS and 32 % for ANN.

Pairing G with power P exhibited the most significant improvements over G alone. For forecast with 1 minute resolution, both networks improved by roughly 60 %. Forecasts with 10 minute resolution improved ANFIS by 63 % and ANN by 39 %.

For clear sky, a greater forecast resolution (1 minute) significantly enhanced results, regardless of the network or input variables used. For cloudy sky however, the two networks showed different behaviours for different time resolutions and input variables. On the one hand, for G and G, T_c, forecast resolution had little impact on the results for both networks. On the other hand, decreasing forecast resolution with G, P as input had contrasting results. Predicting 10 minute averages with ANFIS using G, P, increased MAPE by 42 % (considering best MAPE), whereas the same forecast resolution produced a reduction of 55 % for ANN, clearly outperforming the former. The fact that ANN requires only one neuron indicates that the relation between PV power and the variables used as inputs is not very complex. Given the small size of the data set, the complexity of the problem is further decreased, resulting in a very low MAPE. In general, ANFIS performed better for clear sky, while ANN for cloudy sky, despite minor differences.

The APE evolution throughout the clear sky day reveals ramping values nearing the end of the day. Clear sky data is for 5 days in January, in winter time, when the sun sets earlier. Because APE scales the error to actual values, small values lead to larger errors, meaning that the APE is not a good metric for time periods where irradiance is low. A forecasting period between 10h and 16h30 would be appropriate for a winter day at the panels geographic latitude.

The major challenge faced was the small size of the data set, which ultimately dictates the network capacity, despite optimizations attempts. Errors from system modelling are of two types: interpolation and extrapolation. The first happens when the model can't successfully fit a function between two knowns data points, and the second when input data is outside of training data range. The latter are impossible to avoid, since a model has no way of knowing how a system behaves for data that's unavailable for training. However both can be mitigated by a larger data set, where more examples help the model learn system particularities. This limitation is notable for cloudy sky data, since that, of the 5 days available, two exhibit cloudy weather, and only one is overcast.

REFERENCES

[1] G. Masson and I. Kaizuka, "Trends in Photovoltaic Applications," International Energy Agency Photovoltaic Power Systems Programme, Tech. Rep., 2017.

[2] S. Pelland, J. Remund, J. Kleissl, T. Oozeki, and K. D. Brabandere, "Photovoltaic and Solar Forecasting: State of the Art," *International Energy Agency: Photovoltaic Power Systems Programme, Report IEA PVPS T14*, pp. 1–40, 2013.

[3] A. Mellit, "Artificial Intelligence technique for modelling and forecasting of solar radiation data: a review," *International Journal of Artificial Intelligence and Soft Computing*, vol. 1, no. 1, p. 52, 2008.

[4] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural Network Design*. Martin Hagan, 1995.

[5] M. T. Hagan and M. B. Menhaj, "Training Feedforward Networks with the Marquardt Algorithm," *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994.

[6] J. S. R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[7] L. A. Zadeh, "Fuzzy Logic," *Information and Control*, vol. 8, pp. 338–353, 1965.

[8] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-15, no. 1, pp. 116–132, 1985.

[9] S. L. Chiu, "Fuzzy model identification based on cluster estimation," *Journal of Intelligent and Fuzzy Systems*, vol. 2, no. 3, pp. 267–278, 1994.

[10] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359–366, 1989.