Renewable Energy 85 (2016) 631–641

Contents lists available at ScienceDirect

Renewable Energy

journal homepage: www.elsevier.com/locate/renene

An artificial neural network to assess the impact of neighbouring photovoltaic systems in power forecasting in Utrecht, the Netherlands

A.G.R. Vaz [a, b], B. Elsinga [b], W.G.J.H.M. van Sark [b], M.C. Brito [c, *]

[a] Faculdade de Ciências da Universidade de Lisboa, Department of Geographic Engineering, Geophysics and Energy, 1749-016 Lisboa, Portugal
[b] Utrecht University, Copernicus Institute of Sustainable Development, Energy & Resources, Heidelberglaan 2, 3584 CS Utrecht, The Netherlands
[c] Instituto Dom Luiz, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal

ARTICLE INFO

Article history:
Received 13 March 2015
Received in revised form
4 June 2015
Accepted 25 June 2015
Available online 17 July 2015

Keywords:
Photovoltaics
Artificial neural network
NARX model
Time series
Forecasting

ABSTRACT

In order to perform predictions of a photovoltaic (PV) system power production, a neural network architecture system using the Nonlinear Autoregressive with eXogenous inputs (NARX) model is implemented using not only local meteorological data but also measurements of neighbouring PV systems as inputs. Input configurations are compared to assess the effects of the different inputs. The added value of the information of the neighbouring PV systems has demonstrated to further improve the accuracy of predictions for both winter and summer seasons. Additionally, forecasts up to 1 month are tested and compared with a persistence model. Normalized root mean square errors (nRMSE) ranged between 9% and 25%, with the NARX model clearly outperforming the persistence model for forecast horizons greater than 15 min.

1. Introduction

The challenge for electrical grid operators is to continuously synchronize electrical energy demand and supply. As global demand for renewable energy is increasing, the economic and technical issues of photovoltaic (PV) solar power penetrations into the power grid must be addressed. Especially since natural variability of the solar resource, seasonal deviations in production and the high cost of energy storage raises concerns regarding reliability and feasibility of solar power systems. This is due to the fact that solar energy is highly dependent on weather conditions including cloud structure and day/night cycles. Clouds can cause significant ramps in solar insolation and PV output, which may be difficult to handle by the grid operator. Therefore, integration of electricity produced by solar power systems requires accurate solar energy forecasts.

Solar energy forecasts allow grid operators to adapt the load in order to optimize the energy transport, allocate the needed balance energy from other sources if no solar energy is available and plan maintenance activities at the production sites. Accurate solar forecasting methods improve the quality of the energy delivered to the grid and reduce the additional cost associated with weather dependency. The combination of these two factors has been the main motivation behind considerable research activities in solar forecasting.

Linear models, such as Box–Jenkins and autoregressive integrated moving average (ARIMA) type models are regularly used to generate forecasts. They assume linear correlation structures among the time series values and thus no nonlinear patterns can be captured Zhang [30]. Subsequently, Reikard [22], Paoli et al. [25], and Pedro & Coimbra [23] used nonlinear models that show more flexibility in capturing the data underlying characteristics and those nonlinear models outperformed linear models. Moreover, at shorter time interval (less than 1 h), short-term patterns dominate and Artificial Neural Networks (ANN, see Section 3 for definitions and properties) demonstrated good results in solar forecasting Diagne et al. [5].

In Ref. [26] ANN were used to perform one-step ahead forecasting of hourly values of global irradiance and they revealed that those results outperform linear models results. They also compared various models in terms of error and training time and found that the Levenberg–Marquardt algorithm achieved the best performance.

In Ref. [29] a comparative study between different ANN models was conducted to predict insolation 1-day ahead, in which the recurrent neural network outperformed the *feedforward* neural

* Corresponding author.
E-mail address: mcbrito@fc.ul.pt (M.C. Brito).

http://dx.doi.org/10.1016/j.renene.2015.06.061
0960-1481/

network. Additionally, in Ref. [13] the researchers implemented a multilayer neural network for half hour cloudiness forecasting and considered it an important tool for the estimation of cloudiness affecting solar radiation.

ANN forecasting models for hourly solar irradiation for times of up to 6 days ahead were tested in Refs. [14]; concluding that the developed intelligent models outperformed satellite-based models. Moreover, an input selection scheme was used and results revealed that models with slightly larger sets of inputs generally perform better for same-day and 1-day ahead forecasts.

In Ref. [4] forecasting the daily solar radiation with two dynamic artificial neural networks (*Feedforward* Time Delay Neural Network and Nonlinear Autoregressive with eXogenous inputs (NARX)) was proposed. Both models had a satisfactory performance, facilitating energy management of solar systems when storage systems are adopted.

Several studies that compare ANN with other simpler time series techniques (AR, ARIMA, etc.) have been conducted in the past. In Ref. [10] different solar forecasting techniques' performances are assessed based upon a forecasting skill given as

$$s = 1 - \frac{RMSE}{RMSE_p} \tag{1}$$

where RMSEp denotes the root mean square error of the persistence model. Thus, a forecasting skill closer to 1 shows that the model being assessed has significantly improved the accuracy relatively to the persistence model.

Bacher et al. [1] compare an AR and ARX model for short-term and medium-term solar forecasting and conclude that the most important input is the lagged PV values for short-term horizon (2 h) and, for long horizon NWP models become more important. The authors showed a forecasting skill *s* of 0.27 and 0.34 for a horizon of 1 h–6 h for the AR and ARX model, respectively. Paoli et al. [22] presented a forecasting skill *s* of 0.19 and 0.20 for AR and ANN respectively, for one day ahead. The forecasting skill in Ref. [16] for intra-day forecasts was 0.16 and 0.17 for an AR and ANN model, respectively; for day ahead, the forecasting skill was similar varying between 0.18 and 0.20 for the AR and ANN model respectively. Voyant et al. [28] suggests that an ANN model for daily forecasts on 6 months-cloudy period improves the power production prediction by 9% and 1% relatively to the persistence and ARMA model, respectively. Pedro and Coimbra [23] also presented a comparison between ARIMA and ANN to predict 1 h and 2 h average power output; the forecasting skill for the 1 h forecasts were 0.02 and 0.18, and for the 2 h forecasts were 0.10 and 0.11, for the ARIMA and ANN model respectively; thus showing neural networks have greater potential for short-term forecasting.

The present work improves on A.G.R. Vaz [27] and uses an ANN model to capture the short-term (15 min) ramping patterns caused by cloud formations and to forecast a PV system power output up to 1-month ahead. Moreover, using different input combinations, we assess whether or not solar power forecasts can be improved by knowing beforehand the power output of other neighbouring (few km distance) grid-connected PV systems and meteorological information. Additionally, the forecasting accuracy of the ANN is compared to the persistence model. In Section 3, principles of ANNs are briefly discussed, and in Section 4 the used methodology is presented.

2. Clear sky persistence model

The persistence model is a simple forecasting model that requires knowledge of clear sky irradiance. Usually, this simple forecasting tool is very accurate for very short-time horizons and for low irradiance variability. The model has the clear sky conditions persist for the next time-step and meets the definition of Marquez and Coimbra [15] applied to the power production of a PV system,

$$\hat{k}^*(t + \Delta t) = \hat{k}^*(t) = \frac{PVPP(t)_{measured}}{PVPP(t)_{clr}} \tag{2}$$

$$PVPP(t + \Delta t) = \hat{k}^*(t + \Delta t) \times PVPP(t + \Delta t)_{clr} \tag{3}$$

where $\hat{k}^*$ is defined as clear-sky index, t denotes the time instant, $PVPP(t)$ is the measured photovoltaic power production and $PVPP(t)_{clr}$ is the photovoltaic power production for clear sky conditions, calculated according to Ineichen and Perez [9]. Other clear sky models are possible and Gueymard [7] is suggested for thorough analyses of different models.

3. Artificial neural networks

3.1. Definitions and properties

In its most general form, an ANN is a machine that models a task or function of interest, performing useful computation through a process of *learning*. In fact, the artificial neural network derives its computing power through its massively parallel distributed structure and its ability to learn and generalize, which means finding reasonable outputs whenever inputs are not encountered during training (learning) [8].

The ANN consists of simple processing units, the neuron, and directed, weighted connections between those neurons. The inputs channels have an associated weight, such that the incoming information x_i is multiplied by a corresponding weight w_i. The network input is the result of the so-called propagation function. Here, the strength of a connection between two neurons i and j is a connecting weight w_{ij}. Experimental knowledge, acquired by the network through a learning process, is stored by massively interconnecting these units (synaptic weights). These connecting weights can be inhibitory or excitatory and by being connected with the neurons, data are transferred.

The output is a function of the particular activation function chosen and a possible bias. The latter is similar to a weight, albeit it has a constant input of 1. This bias term is used by the neuron to generate an output signal in the absence of input signals. Fig. 1 illustrates the nonlinear model of a neuron [8].

The transfer function or activation function controls the amplitude of the output of the neuron and is based on the neuron reactions to the input values and depends on the level of activity of the neurons (activation state). Essentially, neurons are activated

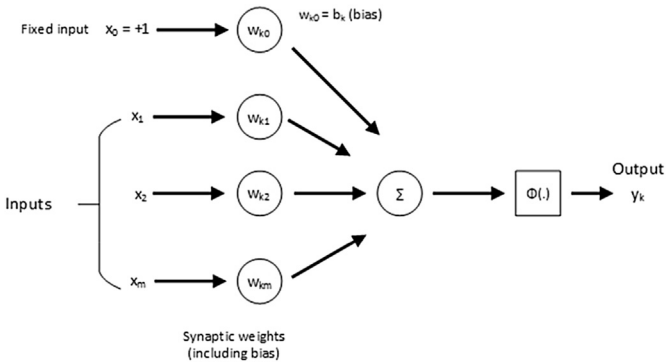

Fig. 1. Nonlinear model of a neuron (Redrawn from Ref. [8]).

when the network input exceeds the uniquely maximum gradient assigned value of the activation function, known as threshold. The activation function is dependent of the previous activation state of the neuron and the external input. It is also referred as a squashing function because limits the permissible amplitude range of the output signal to some finite value. Although theoretically any differential function can be used as an activation function, the identity and sigmoid functions are the most used. In fact, an important feature of the ANN theory is the need for differentiability.

Typically, the normalized amplitude range of the output of a neuron is written as the closed unit interval [0, +1] or alternatively [−1, +1]. The sigmoid function, whose graph is s-shaped, is one of the most common forms of activation used in the construction of ANNs. It exhibits a balance between linear and nonlinear behaviour. The logistic function assumes a continuous range of values from 0 to +1, and it is easily differentiated.

The hyperbolic tangent function *tanh* is also, often, used because of the simplicity in finding its derivatives. In this case, the activation function assumes an antisymmetric form with respect to the origin, ranging from −1 to +1. This function is applied in the hidden layer of the network and takes the input with any value between plus and minus infinity to generate an output in the range between −1 and +1.

The identity or linear function $f(x) = x$, where the inputs and outputs range from minus infinity to plus infinity, is a flow-through mapping of the networks' potential to its outputs.

3.1.1. Feedforward neural networks

In the simplest form of a layered network, an input layer of source nodes projects onto an output layered network, but not vice versa. This network is designated as a *feedforward* type. *Feedforward* neural network is a nonlinear function of its inputs, which is the composition of the functions of its neurons. The network consists of set of sensory units that constitute the input layer and an output layer of computation nodes. The information runs through the connected neurons only in the forward direction or layer-by-layer basis, from inputs to outputs; such network is called a single-layer network, with the designation referring to the output layer of neurons.

3.1.2. Multilayer neural networks

Most neural networks applications require the use of multilayer networks where each output is a nonlinear function of the nonlinear functions computed by the hidden neurons. The neurons are grouped in input layer, hidden processing layers and output layer. This function of hidden neurons intervenes between the external input and the network output in some useful manner. Haylin [8] mentions that the hidden neurons in the hidden layer enable the network to extract higher-order statistics and to learn complex tasks by extracting progressively more meaningful features from the input patterns.

For prediction purposes, data is presented to multilayer networks as a sliding window over the time series observations. According to Bramer [3] the task of the multilayer network is to model the underlying generator of the data during training, so that a valid forecast is made when the trained neural network is subsequently presented with a new input vector value. When examples of the observation data are trained, the networks can learn the characteristic features “hidden” in the examples of the collected data and even generalize the knowledge learnt. In fact, the network exhibits high degrees of connectivity, determined by the synapses of the network. A change in the connectivity of the network requires a change in the population of synaptic or their weights. Fig. 2 displays the architecture of a multilayer *feedforward* neural network with one hidden layer.

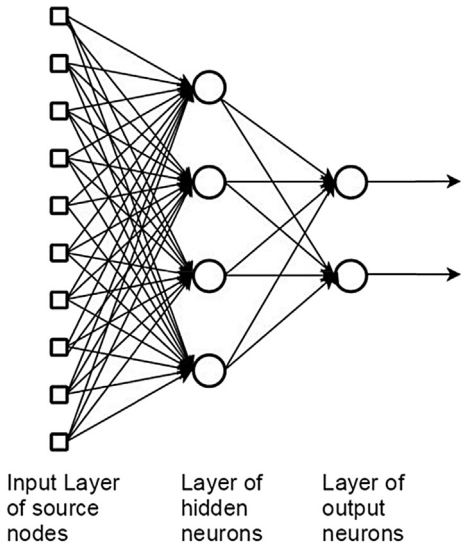

Fig. 2. *Feedforward* neural network with one hidden layer, one input layer and one output layer (Redrawn from Ref. [8]).

3.1.3. Dynamic driven recurrent networks

Most dynamical systems involve an autonomous part and a part governed by external force that usually is difficult to identify or noisy. Forecasting deals with dynamic models whose inputs and outputs are related through differential equations, or, for discrete-time systems, by recurrent equations.

Therefore, this work uses recurrent networks as input–output mapping networks. A recurrent network distinguishes itself from a *feedforward* neural network in that it has at least one feedback loop. The presence of feedback loops has a profound impact on the learning capability of the network and on its performance. Furthermore, the feedback loops involve the use of unit-delay elements (z^{-1}), which result in nonlinear dynamical behaviour.

Basically, an external input is applied and the recurrent network has a temporary response, being considered as dynamically driven recurrent network. This characteristic enables recurrent networks to acquire state representations, which are fundamental for applications such as nonlinear predictions and modelling.

3.1.4. NARX model

The input–output recurrent model is illustrated in Fig. 3, with a design that follows the typical multilayer perceptron, which are neurons with adjustable synaptic weights and bias. Typically, this model has supervised learning, involving modification of the synaptic weights of the neural network by applying a set of labelled training samples. To a unique input signal there is a corresponding desired response. After being presented with an example picked from the set, the synaptic weights of the network are modified to minimize the differences between the desired response and the actual value response of the network produced by the input signal. A tapped-delay-line (TDL) memory of q elements is applied to the model inputs. A delay line tap extracts a signal output from somewhere within the delay line and usually sums with other taps to form an output signal. Moreover, via another TDL memory with q units, the single output is also fed back to the input. Thus, the contents from both TDL memories are fed to the input layer of the multilayer perceptron.

In Fig. 3, $u(n)$ denotes the present value of the model input and $y(n + 1)$ corresponds to the value of the model output. Hence, the present and past values of the input, which are exogenous inputs generated from outside the network, and delayed values of the

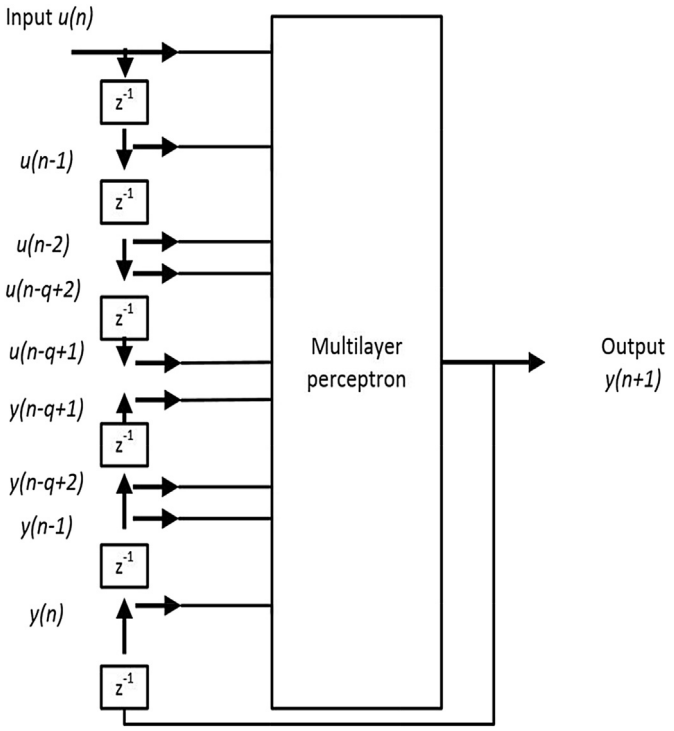

Fig. 3. Nonlinear autoregressive with exogenous inputs (NARX) mode (Redrawn from Ref. [8]).

output, on which the model output is regressed, are the data window of the signal vector applied to the input layer.

This recurrent network described above is also referred as nonlinear autoregressive with exogenous inputs (NARX) model.

Equation (4) demonstrates the dynamic behaviour of the NARX model, where F is a nonlinear function of its arguments. The two delay line memories in the model are generally different, albeit they can have the same q size.

$$y(n+1) = F(y(n), \ldots, y(n-q+1), u(n), \ldots, u(n-q+1)) \quad (4)$$

3.1.5. Training

The training of the network is repeated for many examples in the set until the network reaches a steady state where there are no further important changes in the synaptic weights. However, *overfitting* may occur. This is caused when the validation error increases while training error decreases progressively. To avoid it [21]; solved the training termination problem based on stopping criteria by developing an automated stopping principle using a predetermined number of training steps. Consequently, when the training stopping achieves that stage, the network reaches the maximum *generalization*. This action is known as *early stopping*. Additionally, in Ref. [24] the method of early stopping with cross-validation is suggested. This method proposes the division of collected data into a training set and a test set, and for further partitioning of the training set into the estimation set and validation set.

Thus, before activating the network, the time series were divided into three different sets: training, testing and validation sets. The range selected for the training, testing, and validation sets was 70%, 15%, and 15% Beale et al. [2], respectively, of the time series used as inputs. This division intends to avoid the risk of using a testing set characterized by a certain type of trend.

During the training phase the true output is available and, therefore, is used to reduce the associated errors (series–parallel architecture, see Fig. 4b). With this mechanism the ANN adopts the *feedforward* architecture with more accurate inputs. After the training phase the architecture changes and the output is fed back to the input of the *feedforward* neural network (Parallel architecture, see Fig. 4a) to perform multistep predictions, which is part of the standard NARX architecture.

4. Methodology

4.1. Data collection

In the present work, we selected variables that are directly or indirectly influenced by cloud movements. For that reason, the fundamental inputs selected were the time series of meteorological data (solar radiation and ambient temperature) and data of five geographically separated PV systems from different households in the city of Utrecht, The Netherlands, collected from the 1st of August of 2012 to the 31st of July of 2013. Usually, the weather in the summer in Utrecht can be characterized as pleasantly warm, with more than 200 h of sunshine in June. On the other hand, the winter becomes chilly with the average high temperature dropping to 4° Celsius, with less than 40 h of sunshine in December and January.

The meteorological data was collected from the Royal Netherlands Meteorological Institute (KNMI) website [12] and the performance data regarding the PV systems was collected by the PV-Group of the Copernicus Institute of Sustainable Development, Utrecht University, using meters with Wh pulse outputs. KNMI employs a Kipp&Zonen CM11 [11] pyranometer to measure global horizontal irradiance. Fig. 5 illustrates the geographic distribution of the PV systems (red (in the web version) circles) and the location where the Utrecht meteorological data is collected (blue (in the web version) circle at De Bilt). The PV systems were designated as Centre, West, East, North and South PV systems; technical data are shown in Tables 1 and 2.

Figs. 6 and 7 illustrate the raw PV power output of 5 different systems and meteorological data from the 1st of August of 2012 until the 31st of July of 2013.

4.2. Data preprocessing

The raw energy data collected once every minute from the PV systems was smoothed using a moving average and it was used to determine the PV power with 15 min steps.

Conversely, ambient temperature (in degrees Celsius) and solar irradiance (W/m^2) data from the meteorological station had originally 1 h time steps and interpolation was required to temporally combine at higher frequencies the PV systems output and solar irradiance ground measurements. As stated in Refs. [17]; sun position and solar zenith angle are non-linear functions of time, and, thus, solar irradiance is not a linear process and direct linear interpolation is incorrect. As an alternative, clear sky index interpolation $\hat{k}^*(t2)$ is used to interpolate meteorological data to 15 min time steps [The persistence model will result in $\mathrm{GHI}(t2) = \hat{k}^*(t2)\, GHI_{clr}(t2)$].

All of the missing observations (Table 1) were considered as undefined numbers and were ignored by the ANN. Additionally, all observations between 9.00 pm and 6.30 am of each day (night time) were removed in order to reduce the complexity and simulations' running time.

To complete the data preprocessing, all time series were normalized between 0 and 1 because each dataset had different magnitudes.

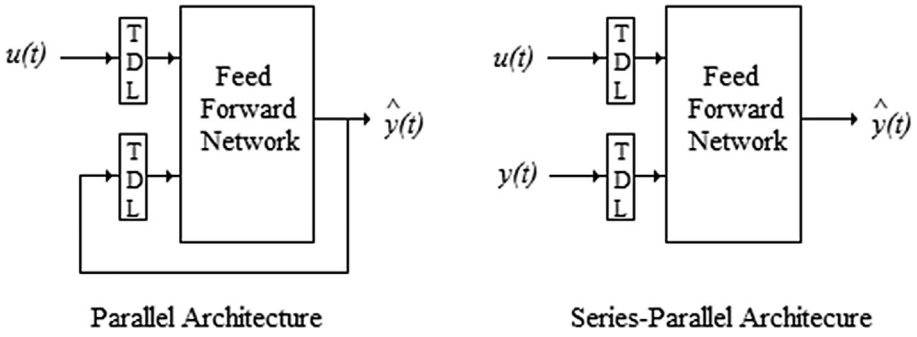

Fig. 4. NARX network architecture variations (Redrawn from Ref. [2]).

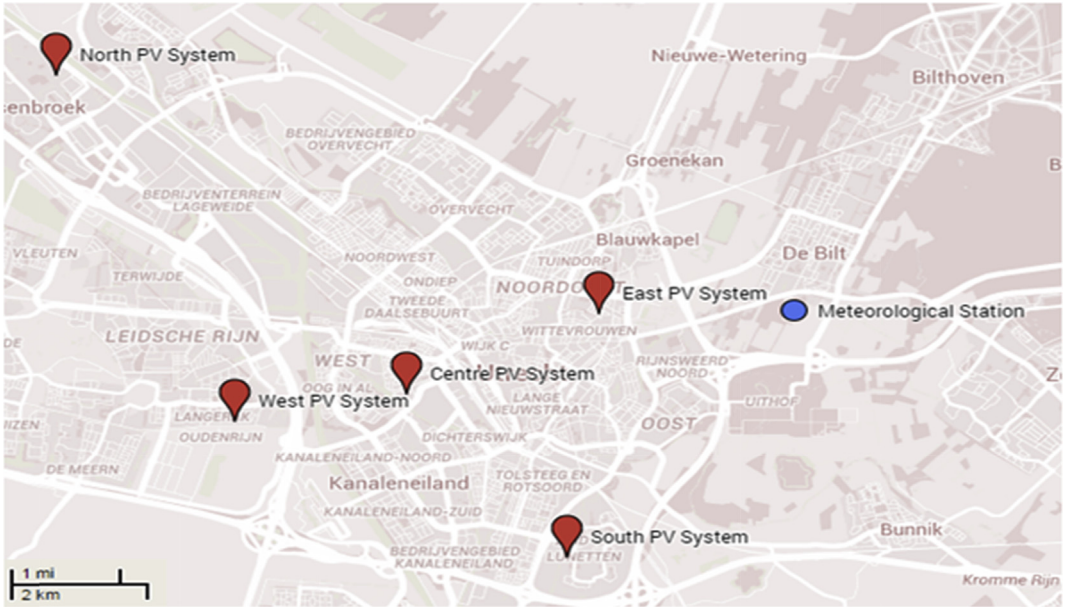

Fig. 5. Map of Utrecht illustrating the distribution of the PV systems and the meteorological station (Google Maps, 2014).

Table 1
Technical information of PV systems.

PV system	Max power installed (Watt)	Date of installation	Distance to centre PV (km)	Distance to meteorological station (km)	Missing data points in the raw dataset (total of 525600 points)
Centre	500	1 Nov 2003	–	5.5	5194
West	720	29 Nov 2002	2.5	8	951
East	1650	1 Mar 2002	5.5	2.6	3879
North	880	1 Jan 2004	7.4	11	460
South	570	1 Nov 2001	3.5	5.3	1451

4.3. Artificial neural network paradigms

Selecting the number of hidden neurons involves a heuristic approach. After testing a range of hidden neurons, it was found that the difference in the final results was negligible, and, therefore, the MATLAB Neural Network Toolbox's default value of 10 hidden neurons was used. Moreover, the selection of the number of the tapped-delay-lines (TDLs) is to a certain degree similar to the hidden neuron selection process. Having the hidden neurons selected, the influence of feedback delays in the neural network was also tested. Similarly, the variation in the network's final results was insignificant and, thus, 2 TDL were used throughout this work.

The transfer functions selected for the hidden layer and the output layer are the hyperbolic tangent and linear function, respectively. Therefore, to be consistent with the transfer function being used, the input data was scaled between −1 and +1. Yet, the data was scaled back to the original dimensions (0–1) after the

Table 2
Location of the PV systems in Utrecht.

PV system	Latitude	Longitude	Altitude (m)
Centre	52° 5′14.99″ N	5° 5′ 53.26″ E	5
West	52° 4′ 59.88″ N	5° 3′ 46.79″ E	1
East	52° 3′ 43.56″ N	5° 7′ 50.55″ E	3
North	52° 8′ 12.48″ N	5° 1′ 35.29″ E	0
South	52° 5′ 59.63″ N	5° 8′ 13.88″ E	5

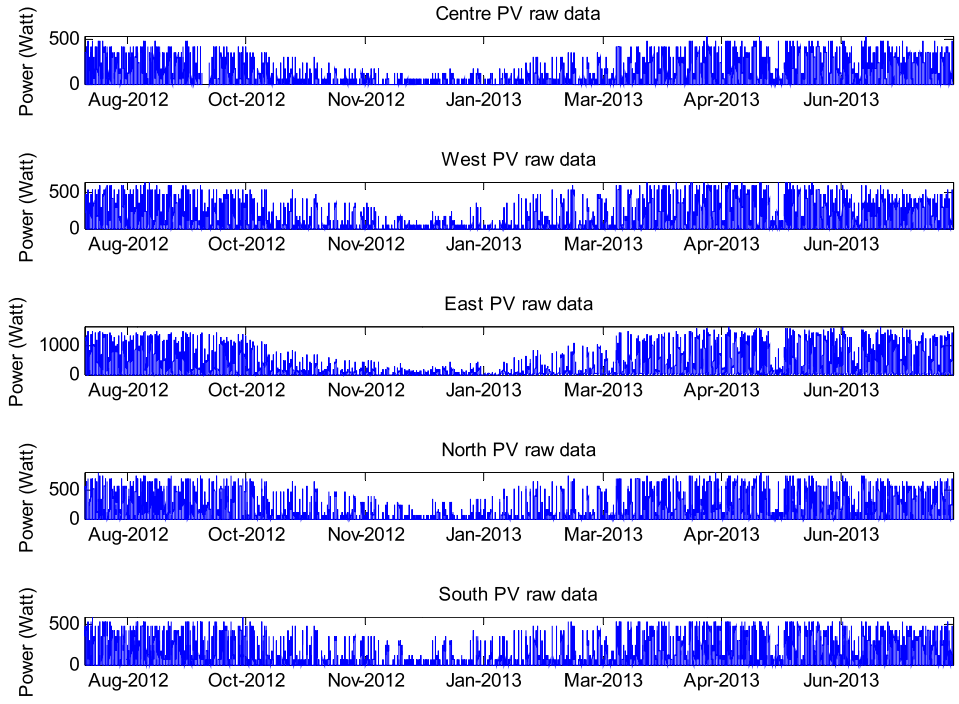

Fig. 6. Time series generated by the PV system from August of 2012 to July of 2013, on a 1 min basis.

network being processed.

The data from the Centre PV system was used as target series and the other PV and meteorological data were used as exogenous inputs. Many different exogenous inputs combinations were performed to understand the impact on the target data series.

Every time a network was trained a different solution was achieved given the different initial weights and bias values. Thus, to ensure good accuracy, each architecture was simulated eleven times and the median of the eleven simulations was calculated; the results larger than 15% of the median are disregarded to eliminate outliers. Furthermore, the mean value of the remaining results was calculated and assumed as the final value of the performance of that specific network architecture.

In this work, the Levenberg–Marquardt algorithm was the algorithm used for every training process and the number of epochs is set to a maximum of 1000 (see Ref. [20] for a thorough description of the algorithm). Moreover, the number of training interactions is defined automatically as the early stopping principle

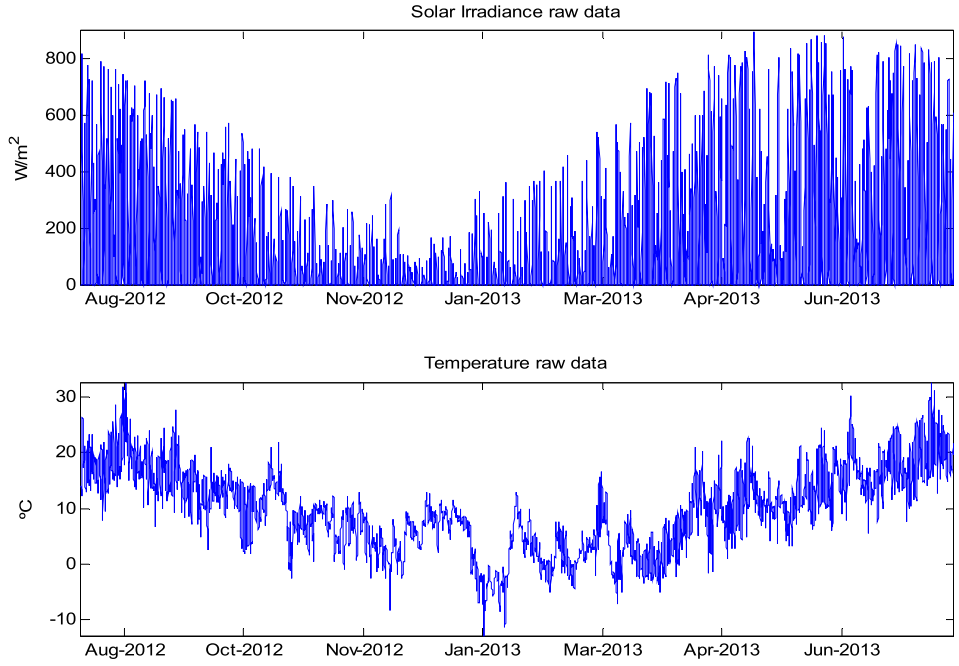

Fig. 7. Meteorological data from August of 2012 to July of 2013, on a 1 h basis.

was applied.

4.4. Optimization

The following different scenarios were applied to help selecting the best network (with the lowest forecasting error) architecture and to learn about the impact of using variations of the exogenous variables.

- Case 1 – NARX network with data of 4 PV systems as exogenous inputs.
- Case 2 – NARX network with data of 2 PV systems as exogenous inputs.
  - Case 2.1 – West and East PV systems
  - Case 2.2 – North and South PV systems
- Case 3 – NARX network with Meteorological data as exogenous inputs
- Case 4 – NARX network with data of 4 PV systems and Meteorological data as exogenous inputs
- Case 5 – Multistep ahead forecasting

The first 4 scenarios were used to predict values for the last month of the dataset (for the summer and winter season) on a one step forecast basis, i.e. to forecast the PV output at $t + 15$ min we use the exogenous inputs' time series until t. These cases were applied for the entire dataset (1 year) and for the dataset divided in two sections. This allowed the determination of the most relevant data and selection of the best configuration amongst the previous scenarios.

On the other hand, Case 5 intends to study the step ahead NARX neural network forecasting performances, which means, trying to forecast the PV output at $t + 15$ min using the input time series until $t - dt$, where "dt" varies between 15 min, 30 min, 1 h, 24 h, 4 days, 7 days, 15 days, 20 days, and 1 month.

4.5. Prediction accuracy evaluation

To quantify the quality of a prediction, let us assume the time series target value $y(t)$ and the predicted value $y_f(t)$ for a series of length n. The normalize root mean square error (nRMSE) is representative of the size of a "typical" error and tends to exaggerate large errors because squaring gives more weight to very large errors.

$$RMSE = \frac{1}{y_{max} - y_{min}} \sqrt{\frac{\sum_{i=1}^{n} \left(y(t) - y_f(t)\right)^2}{n}} \tag{5}$$

where y_{max} and y_{min} are the maximum and minimum observed values [4].

The coefficient of variation (CV) was also calculated, which is determined by the ratio between the standard deviation (σ) and the mean (M) of the nRMSE trials. The lower the CV value, the smaller the deviations between the multiple trials. Consequently, this may suggest a good model fit and gives a rather good idea of how the ANN trials differ between themselves.

$$\text{CV } (\%) = \frac{\sqrt{\frac{\sum \left(x_i - \frac{\sum x}{N}\right)^2}{N}}}{\frac{\sum x}{N}} \times 100 = \frac{\sigma}{M} \times 100 \tag{6}$$

where $x_i - \sum$ calculates the difference between a nRMSE trial and the mean nRMSE value of all trials; N is the number of trials.

5. Results

Table 3 displays the different situations for which the networks for case 1 to 4 were used. Neural network predictions were performed for different seasons (winter and summer) by dividing the original dataset in two different subsets. In the "Winter" dataset, data points from August 2012 to December 2012 (5 months) were used to train the network, and then January of 2013 was forecasted. The "Summer" dataset required training the network from February 2013 to June 2013, and later July 2013 was predicted. Finally, given the fact the original dataset is from August 2012 to July 2013, it was only possible to perform forecasts for July 2013 using the entire dataset. This is the reason for designating "1-year–summer" for this situation (see Table 3). The first 11 months were used for training the network and after that the last month (July 2013) was forecasted.

Table 4 and Fig. 8 show results of the different network configurations in each case when performing multistep predictions. "t" denotes the training and modelling of the NARX network, whereas "f" denotes the process of forecasting the last month of the time series.

Winter forecasts showed lower nRMSEf values than summer forecasts, which may be a consequence of the easy predictability of low variability depicted in overcast days. On the other hand, significant variability was visible during July of 2013 (summer forecasts), and consequently, the nRMSEf is likely to increase.

Although nRMSEf's magnitude varied in the three different situations presented, one may detect that the behaviour is similar for the different exogenous inputs' cases proposed, i.e., a tendency was identified regarding which exogenous inputs perform better within the network. For the three different situations the best performance was obtained by combining all exogenous inputs available. Similarly, in case 2.2 the nRMSEf values are worse than case 2.1 regardless of being summer or winter. Even though the nRMSEf accuracy may differ for different time series lengths, results indicate that the type of correlation between exogenous inputs and output variables is the key factor.

Case 2.2 presented the highest nRMSEf results (0.079 for the Winter situation, 0.121 for the Summer situation, and 0.140 for the 1year-summer situation), probably due to the fact that the distance between the North PV system and South PV system is higher than, for instance, between the West PV system and the East PV system. Consequently, the interactions between the North and South PV systems appear to have less impact in the final output than other cases.

Although both NARX neural networks in Case 2.1 and Case 2.2 use data of 2 PV systems as exogenous inputs, the West and East PV systems combination (Case 2.1) clearly outperforms the North and South PV systems input combination, with a 20% relative reduction of the nRMSEf value for the winter situation, 13% relative reduction for the summer situation and a 17% relative reduction for the 1year-summer situation. This may be due to a closer distance between the West and East PV systems with the target system (notably within the decorrelation length of $\approx$5 km at 15 min interval (see Ref. [6])), and, thus, a more relevant correlation between the time series. Indeed, Still, the NARX neural network in Case 1 can improve the prediction accuracy when combining the 4 PV systems: the 0.059 nRMSEf for winter, 0.099 nRMSEf for summer, and 0.014 nRMSEf for the 1-year summer situation, represent, respectively, a 6%, 6%, and 2% accuracy improvement comparing to Case 2.1.

Furthermore, in Case 3 where meteorological data was used as exogenous inputs, the NARX network had the least accurate performance in the winter case (nRMSEf 0.083), but in the 1year-summer situation had the most accurate value (nRMSEf 0.106) relatively to the other cases in the same 1year-situation.

Table 3
Analysis information of time series after applying 15 min time intervals.

Designation	Range	Total data points	Training	Forecasting (last month of the time series)
Winter	1/08/2012–1/01/2013	10672	Aug 12 – Dec 12 8874 points (5 months)	January 2013 1798 points (1 month)
Summer	1/02/2013–31/07/2013	10498	Feb 13 – Jun 13 8700 points (5 months)	July 2013 1798 points (1 month)
1 year – Summer	1/08/2012–31/07/2013	21170	Aug 12 – Jun 13 19372 (11 months)	July 2013 1798 points (1 month)

Table 4
NARX network exogenous inputs, results of RSME and CV "t" (training) and "f" (forecasting) for the different cases and seasons. The output variable is fixed – PV Centre. In bold best performances (lowest errors).

NARX network with 10 hidden neurons & 2 TDL			Exogenous inputs				
			4 PV systems (N.S.E.W)	2 PV systems – west and east	2 PV systems – north and south	Meteorological	4 PV Systems (N.S.E.W) and meteorological
			Case 1	Case 2.1	Case 2.2	Case 3	Case 4
5 months training & 1 month forecasting	Winter forecasting	nRMSEt	0.007	0.006	0.013	0.012	0.007
		nRMSEf	0.059	0.063	0.079	0.083	**0.058**
		CV_{nRMSEt}	35%	41%	16%	46%	26%
		CV_{nRMSEf}	19%	27%	8%	29%	14%
	Summer forecasting	nRMSEt	0.019	0.022	0.029	0.025	0.013
		nRMSEf	0.099	0.105	0.121	0.109	**0.085**
		CV_{nRMSEt}	11%	14%	17%	51%	12%
		CV_{nRMSEf}	6%	7%	8%	25%	13%
11 months training & 1 month forecasting	1 year – Summer forecasting	nRMSEt	0.051	0.052	0.077	0.045	0.029
		nRMSEf	0.114	0.116	0.140	0.106	**0.092**
		CV_{nRMSEt}	6%	13%	9%	7%	11%
		CV_{nRMSEf}	3%	7%	5%	4%	11%

Adding the meteorological data and the 4 PV systems as exogenous inputs (Case 4) further improved the NARX neural network prediction accuracy with the nRMSEf decreasing by another 2% in the winter situation, 14% in the summer situation, and 19% in the 1year-summer situation. Thus, combining all the information available revealed to be the best method. The following section used the Case 4 NARX network architecture in the 1year-summer situation to illustrate graphically the training and forecasting. The 1 year-summer situation was also selected in case 5 due to the fact that it has more data points and, although the nRMSEf are higher than in the case where subsets of the original dataset are used, CVf shows in Table 4 that the multiple trials were more consistent than in the other cases.

Figs. 9–11 display the one-step predictions using the best network configuration achieved (Case 4). This is merely an example of the expected output values using a NARX neural network with 10 hidden neurons, 2 TDL and 6 exogenous inputs. Fig. 10 shows forecasts of the Centre PV system output (in green) for July 2013, using 11months (Fig. 9) for training, testing and validation. The network forecasting results can successfully approximate the expected targets and the intra-hour ramping is well captured. Moreover, the early and late variations of the day are effectively captured.

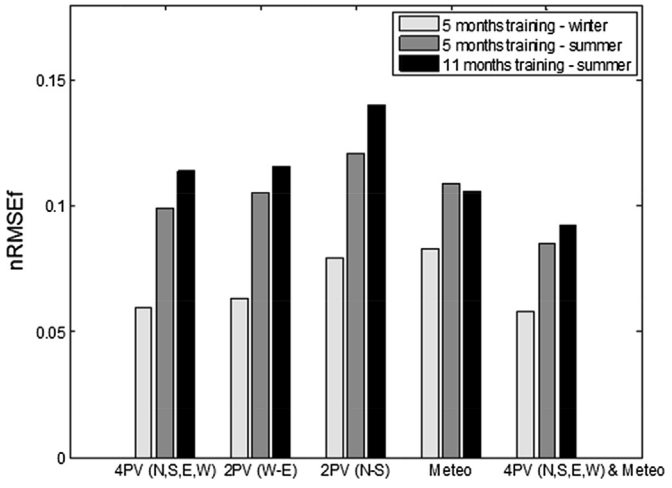

Fig. 8. nRMSEf ("f" denotes forecasting) results for different cases. The *x* label illustrates the exogenous inputs used in the NARX Network – Case 1 (4 PV systems (N, S, E, W)); Case 2.1 (2 PV systems (W, E)); Case 2.2 (2 PV systems (N, S)); Case 3 (Meteorological); Case 4 (4 PV systems (N, S, E, W) & Meteorological).

5.1. Case 5 – multistep forecasting

As mentioned above, in the previous cases the past steps (TDL) of the exogenous inputs and outputs were used to perform several one-step predictions until achieving one month of forecasts. In multiple step forecasting, and as an example, a 1 h forecast for a 15-min resolution using a NARX neural network with 2 TDL consisted in considering the $t - 15$ min and $t - 30$ min values to predict the $t - 1$ h value (instead of the $t + 30$ min and the t + 45 min values).

Fig. 12 shows the comparison between the NARX model and the persistence model. The purpose of this comparison is to highlight the time scales most favourable for the application of the NARX model.

The global minimum errors of both models occurred with a single step ahead predictions (15 min), albeit the NARX model (0.092 nRMSEf) was outperformed by the persistence model (nRMSEf 0.091) by a marginal fraction. However, the persistence model was surpassed by the NARX model beyond the 15 min step forecasts, which is expected considering the very good ability of the NARX model to capture the underlying time series' patterns.

After ($\boldsymbol{t - dt}$), the NARX model's performance improves, for the July 2013 forecasts, relatively to the persistence model. The NARX model's forecasts up to 1 h ahead, which in this study is a 4 step forecast due to the fact that the time interval is 15-min, resulted in a 0.103 nRMSEf, whereas for the persistence model resulted in a 0.16 nRMSEf. Both methods' performances decrease with time, but when $dt = 15$ *min*, a local minimum (nRMSEf 0.19 for the NARX model and 0.25 for the persistence model) is achieved. After $dt = 24$ *h* the curves' slope becomes more gradual.

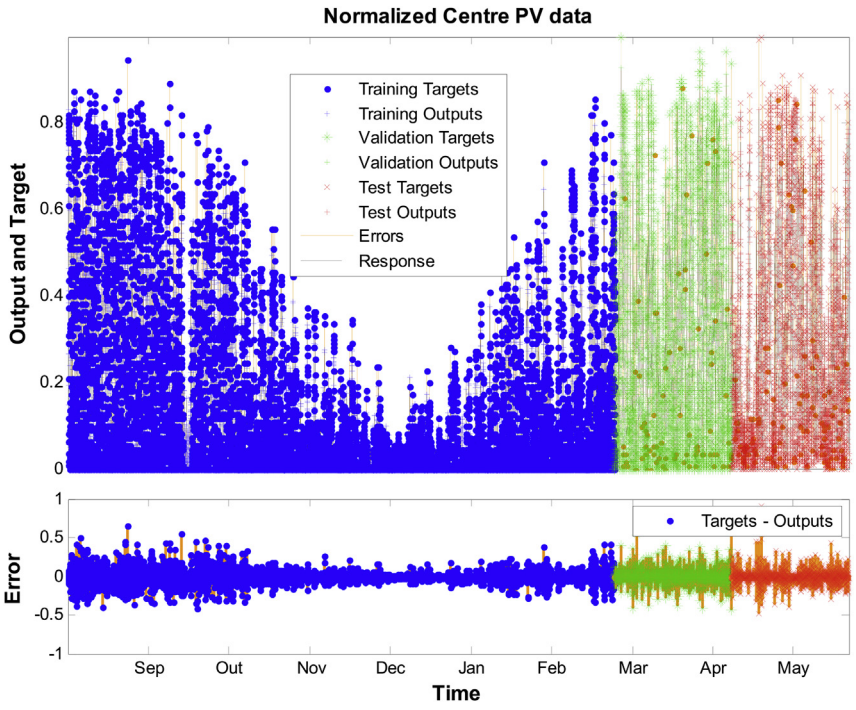

Fig. 9. Example of training (in blue), validation (in green) and test (in red) from 1st of August 2012 to 30th of June 2013 for the Centre PV system, using a NARX network with 4 PV systems and meteorological data as exogenous inputs. The y-axis shows normalized output [0, 1] and the time interval is 15 min. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The difference between the accuracy of the NARX neural network performance for a 1 month forecast horizon and the 15-min forecasting horizon is very significant, being the nRMSEf value at $dt = 24$ h, 2 times greater than the nRMSEf at $dt = 1$ month for the persistence model and NARX model. Indeed, our NARX model appears to be more appropriate for short-time horizons, given the fact that for longer horizons the NARX model's nRMSE is identical to the nRMSE of the persistence model.

In our work, the forecasting skill s (introduced in Refs. [10]), increases for intra-hour forecasts, reaching 0.36 in the 1 h

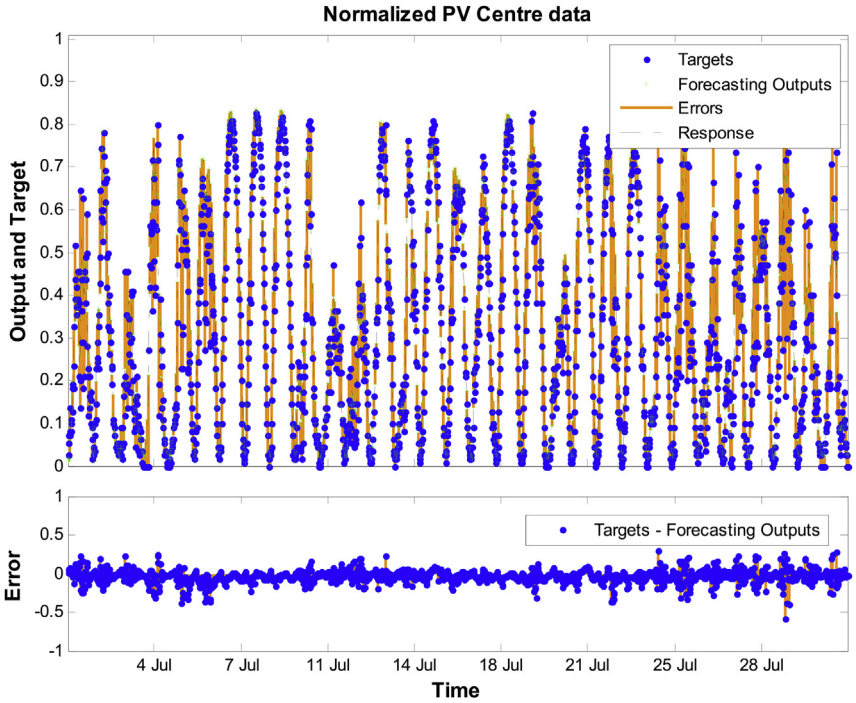

Fig. 10. Closer look of the example of the network forecasts (green) and the expected target (blue) for the Centre PV system, for July 2013. The y-axis shows normalized output [0, 1] and the time interval is 15 min. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

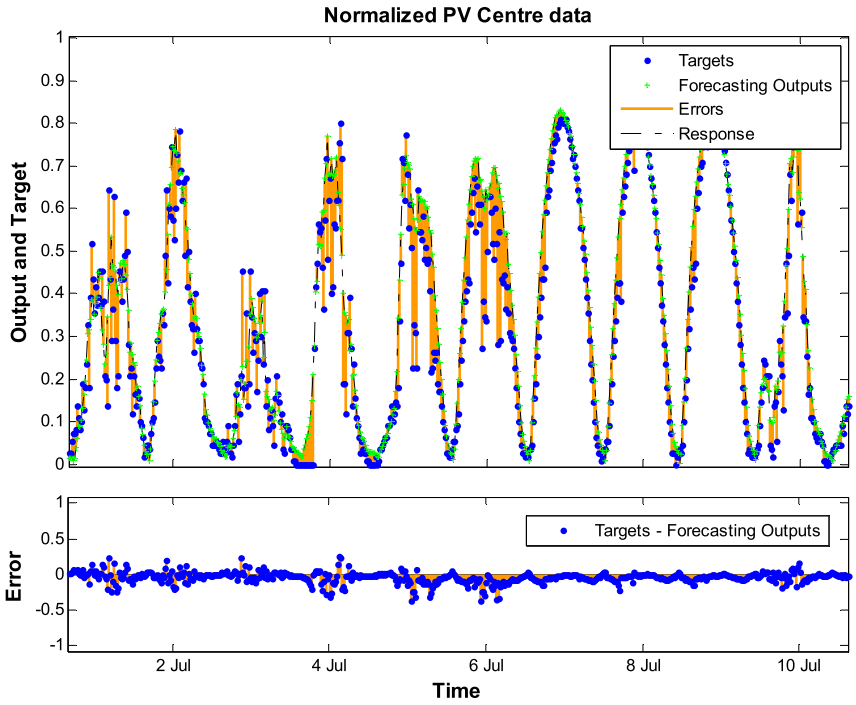

Fig. 11. Closer look of the first 10 days of the example of the network forecasts (green) and the expected target (blue) for the Centre PV system, for July 2013. The y-axis shows normalized output [0, 1] and the time interval is 15 min. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

forecasting; however, after the 1 h forecasting, *s* decreases sharply and has values close to 0 for the 20 days and 1 month forecasts. One may verify that our results are similar to the results found in the literature (see Section 1), particularly for short/medium term predictions.

6. Conclusion

A NARX model for solar power forecasting was developed. Several input combinations allowed the determination of the relevant parameters to the forecast performance of a PV system. For the different input combinations, three situations were applied, to be exact, one with the entire dataset, and two others as the result of splitting the original dataset. This allowed us to analyse the winter and summer seasons.

In general, the NARX model for the winter season had lower nRMSE (root mean square error) values, albeit this was more related to the fact that overcast days are rather easy to predict and low variability was associated with the predicted month. NARX model's summer forecasts showed accurate nRMSE values, and while using the original dataset (1 year in a 15 min interval) the results were more consistent during the many performed trials.

It was shown that considering more information of neighbouring distributed PV systems can enhance the forecasts accuracy. Moreover, adding meteorological information to the network using the information of the 4 PV systems has proved to further improve the accuracy. Yet, the network that solely used meteorological data as exogenous inputs also showed accurate forecasts.

For forecasts up to 15-min ahead, the persistence and the NARX method showed similar performances. However, the NARX model clearly outperformed the persistence model beyond 15-min forecasts, which indicates that considering recent past observations is a better method to perform predictions. Global minimum and local minimum were achieved for 15 min forecasts (9% nRMSE for both the NARX and persistence model) and 24 h (19% and 20% nRMSE value for the NARX and persistence model respectively), for the input and output data considered. Thus, we can also determine that for the PV system output and for the predicted month, a current value of the PV system shows very good ability to recognize patterns of the 24 h previous observation. The NARX model is highly effective in multistep predicting, even for many points.

Neural networks associated with other models, such as the total sky imagery or satellite based models, or more PV systems [6], may well be a fundamental instrument to improve intra-day forecasts accuracy. The intra-day horizon is currently of smaller economic value than the day-ahead forecasts; however, substantial markets opportunities will probable occur when solar PV penetration is increased and accuracy of intra-day forecasts is further improved.

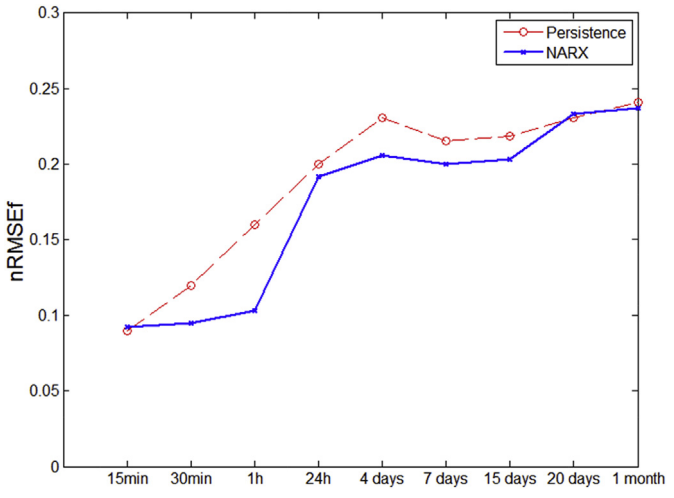

Fig. 12. Comparison of nRMSEf ("f" denotes forecasting) results for July 2013 between persistence model and NARX model [training the network until $t+45$ *min*], with *dt* varying between 15 min and 1 month.

References

[1] P. Bacher, H. Madse, H.A. Nielsen, Online short-term solar power forecasting, Sol. Energy 83 (2009) 1772–1783.
[2] M. Beale, M. Hagan, H. Demuth, Neural Networks Toolbox – User's Guide, Mathworks, Massachusetts, 2013.
[3] M. Bramer, Artificial Intelligence in Theory and Practice, Springer, Santiago, Chile, 2006.
[4] A. Di Piazza, M.C. Di Piazza, G. Vitale, Solar radiation estimate and forecasting by neural networks-based approach, in: XIII Spanish–Portuguese Conference on Electrical Engineering, 2013.
[5] M. Diagne, M. David, P. Lauret, J. Boland, N. Schmutz, Review of solar irradiance forecasting methods and a proposition for small-scale insular grids, Renew. Sustain. Energy Rev. (2013) 65–76.
[6] B. Elsinga, W. van Sark, Spatial power fluctuation correlations in urban rooftop photovoltaic systems, Prog, Photovolt. Res. Appl (2014), http://dx.doi.org/10.1002/pip.2539.
[7] C.A. Gueymard, Clear-sky irradiance predictions for solar resource mapping and large-scale applications: improved validation methodology and detailed performance analysis of 18 broadband radiative models, Sol. Energy (2012) 2145–2169.
[8] S. Haylin, Neural Networks: a Comprehensive Foundation, Prentice Hall International Inc, Hamilton, Ontario, Canada, 1999.
[9] P. Ineichen, R. Perez, A new airmass independent formulation for the Linke turbidity coefficient, Sol. Energy (2002) 151–157.
[10] R.H. Inman, H.T.C. Pedro, C.F.M. Coimbra, Solar forecasting methods for renewable energy integration, Prog. Energy Combust. Sci. 39 (2013) 535–576.
[11] Kipp, Zonen, CMP 11 Pyranometer, 2014. Retrieved from, http://www.kippzonen.com/Product/13/CMP-11-Pyranometer#.VI3YniusV31.
[12] KNMI, Royal Netherlands Meteorological Institute – Climate Data and Advice, 2014. Retrieved from Metadata KNMI stations – De Bilt, http://www.knmi.nl/klimatologie/metadata/debilt.html.
[13] V.H. Mantzari, D.H. Mantzaris, Solar radiation: cloudiness forecasting using a soft computing approach, Artif. Intell. Res. 2 (1) (2013).
[14] R. Marquez, C.F.M. Coimbra, Forecasting of global and direct solar irradiance using stochastic learning methods, ground experiments and the NWS database, Sol. Energy 85 (2011) 746–756.
[15] R. Marquez, C.F.M. Coimbra, Comparison of clear-sky models for evaluating solar forecasting skill, Proc. World Renew. Energy Forum 2012 (2012) 4443–4449.
[16] L. Martín, L.F. Zarzalejo, J. Polo, A. Navarro, R. Marchante, M. Cony, Prediction of global irradiance based on time series analysis: application to solar thermal power plants energy production planning, Sol. Energy 84 (2010) 1772–1781.
[17] P. Mathiesen, J. Kleissl, Evaluation of numerical weather prediction for intra-day solar forecasting in the continental United States, Sol. Energy 85 (2011) 967–977.
[20] J. Nocedal, S.J. Wright, Numerical Optimization, Springer, New York, 1999.
[21] A.K. Palit, D. Popovic, Computational Intelligence in Time Series Forecasting: Theory and Engineering Applications, Springer, 2005.
[22] C. Paoli, C. Voyant, M. Muselli, M.L. Nivet, Forecasting of preprocessed daily solar radiation time series, Sol. Energy 84 (2010) 2146–2160.
[23] H.T.C. Pedro, C.F.M. Coimbra, Assessment of forecasting techniques for solar power production with no exogenous inputs, Sol. Energy 86 (2012) 2017–2028.
[24] L. Prechelt, Automatic Early Stopping Using Cross Validation: Quantifying the Criteria, Universität Karlsruhe, Karlsruhe, Germany: Neural Network, 1998.
[25] G. Reikard, Predicting solar radiation at high resolution: a comparison of time series forecasts, Sol. Energy 83 (2009) 342–349.
[26] A. Sfetsos, A.H. Coonick, Univariate and multivariate forecasting of hourly solar radiation with artificial intelligence techniques, Sol. Energy 68 (2000) 169–178.
[27] A.G.R. Vaz, Photovoltaic Forecasting with Artificial Neural Networks, University of Lisbon, 2014 (MSc thesis) Retrieved from, http://repositorio.ul.pt/bitstream/10451/11405/1/ulfc107351_tm_Andre_Vaz.pdf.
[28] C. Voyant, M. Muselli, C. Paoli, M. Nivet, Optimization of an artificial neural network dedicated to the multivariate forecasting of daily global radiation, Energy 36 (2010) 348–359.
[29] A. Yona, T. Senjyu, A.Y. Saber, T. Funabashi, Application of neural network to one-day-ahead 24 hours generating power forecasting for photovoltaic system, *Intell. Syst. Appl. Power Syst.* (2007), http://dx.doi.org/10.1109/ISAP.2007.4441657.
[30] G.P. Zhang, Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing 50 (2003) 159–175.