


Enhancement of hybrid renewable energy systems control with neural networks applied to weather forecasting: the case of Olvio

P. Chatziagorakis¹ · C. Ziogou⁴ · C. Elmasides^{2,3} · G. Ch. Sirakoulis¹  · I. Karafyllidis¹ · I. Andreadis¹ · N. Georgoulas¹ · D. Giaouris⁴ · A. I. Papadopoulos⁴ · D. Ipsakis⁴ · S. Papadopoulou⁴ · P. Seferlis^{4,5} · F. Stergiopoulos⁴ · S. Voutetakis⁴

Received: 23 January 2015 / Accepted: 31 December 2015
© The Natural Computing Applications Forum 2016

Abstract In this paper, an intelligent forecasting model, a recurrent neural network (RNN) with nonlinear autoregressive architecture, for daily and hourly solar radiation and wind speed prediction is proposed for the enhancement of the power management strategies (PMSs) of hybrid renewable energy systems (HYRES). The presented model (RNN) is applicable to an autonomous HYRES, where its estimations can be used by a central control unit in order to create in real time the proper PMSs for the efficient subsystems' utilization and overall process optimization. For this purpose, a flexible network-based design of the HYRES is used and, moreover, applied to a specific system located on Olvio, near Xanthi, Greece, as part of Systems Sunlight S.A. facilities. The simulation results indicated that RNN is capable of assimilating the given information and delivering some satisfactory future estimation achieving regression coefficient from 0.93 up to 0.99 that can be used to safely calculate the available green energy. Moreover, it has some sufficient for the specific problem computational power, as it can deliver the final results in just a

few seconds. As a result, the RNN framework, trained with local meteorological data, successfully manages to enhance and optimize the PMS based on the provided solar radiation and wind speed prediction and make the specific HYRES suitable for use as a stand-alone remote energy plant.

Keywords Recurrent neural network · Solar radiation · Power management strategy · Hybrid renewable energy system

1 Introduction

The continuous growth of the human population in combination with the decrease in the available fossil fuel supplies has led the researchers to the design and development of generating systems that utilize the renewable energy sources. Nowadays, the worldwide role of renewable energy systems (RES) is very important and multidimensional. The possibilities of renewable energy are endless. The utilization of the various green energy types such as solar, wind, hydroelectric, geothermal, and biomass can lead to sustainable energy solutions. The advantages of a 100 % renewable future energy scenario are numerous. Among them, one of the most important is the environmental pollution abatement, due to the fact that RES produce little or no waste products such as carbon dioxide or other chemical pollutants, thus having minimal impact on the environment. In addition, renewable energy is highly sustainable as it derives from sources that are inexhaustible. Moreover, it can also provide economic benefits to regional areas and remote communities, as most projects are located away from large urban centers and suburbs of the capital cities.

✉ G. Ch. Sirakoulis
gsirak@ee.duth.gr

¹ Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, Greece
² Department of Environmental Engineering, Democritus University of Thrace, Xanthi, Greece
³ Systems Sunlight S.A., Xanthi, Greece
⁴ Chemical Process and Energy Resources Institute, Centre for Research and Technology – Hellas, 57001 Thessaloniki, Thessaloniki, Greece
⁵ Department of Mechanical Engineering, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece

Despite the numerous advantages of RES, they also present some disadvantages. The main disadvantage relates with the amount of energy produced by RES. In particular, it is quite difficult to generate quantities of electricity that are as large as those produced by traditional fossil fuel generators. This may mean that a reduction in the energy consumption is necessitated or simply more energy facilities should be built. It also indicates that the best solution to the energy problem may be the adoption of different green power sources. Another disadvantage of RES is the reliability of supply. Renewable energy often relies on the weather conditions. Hydro-generators need rain to fill dams to supply flowing water, wind turbines need wind to turn the blades, and solar collectors need sunshine to collect heat and make electricity. When these resources are unavailable, it is impossible to make energy out of them. This phenomenon may cause some unpredictable and inconsistent energy production. Finally, the current cost of renewable energy technology is also far in excess of traditional fossil fuel generation [1–3]. To minimize the aforementioned cost and the claimed disadvantages of the RES, better adaptive control and optimization techniques mainly based on efficient intelligent models would be of utter importance for enhancing RES performance.

In recent years, as a response to the continuously growing need for green energy, a new type of renewable energy systems (RES) is becoming all the more popular [1], that is, the *hybrid renewable energy systems* (HYRES). These systems often combine a variety of different renewable technologies with some energy storing units in a single generating plant facility. This combination offers the advantage of exploiting different types of green energy without completely depending on the availability of a single one. Therefore, hybrid systems present a better balance in energy production than the conventional systems, which make use of a single technology and tend to be more inconsistent. The utilization of multiple green energy sources provides to these systems increased efficiency, as well as balance in energy supply due to the fact that each energy source acts as supplement to the others. This is the reason why HYRES are considered as a reliable solution for remote area power generation applications.

However, despite the advantages that the adoption of HYRES may have, there are still some weak spots. Despite the fact that these systems rely on multiple renewable sources, they are still dependent on conventional fuels as long as the green energy is not always available. More specifically, an endless and adequate supply of renewable energy cannot be always guaranteed due to the noncontinuous nature of the green energy, so alternative conventional energy sources are needed to ensure the sufficient and continuous functionality of the understudy system. In

addition, another characteristic that needs further improvement is their efficiency. Although they present a much higher efficiency than the single-technology RES, there is still space for further improvement and optimization [1]. The cooperation between the different discrete systems does not often occur in the most efficient way. For example, storage of the excess energy supply does not always occur in the most effective way, and thus, the system usually depends on conventional fuels. Moreover, without the capability of monitoring and dynamically processing the current local meteorological data, the system tends to present some rather static behavior limiting its efficiency according to supplier's needs.

The great dependency that HYRES efficiency has on both the availability and the values of the critical meteorological variables necessitates their a priori knowledge. Meteorological variables such as solar radiation, air temperature, wind speed, and humidity can affect at a maximum degree the functionality and the efficiency of the corresponding RES. Having this kind of information someone can provide some satisfactory estimation of the total amount of the future renewable energy production for the next hours or, in some cases, even days. In this way, a better management of the HYRES subsystems can be achieved, as well as a much more optimized energy storage and utilization, thus diminishing the need for conventional generators. The optimized management of the various subsystems is the key point toward achieving the best possible green energy utilization and system efficiency.

The goal of the specific study was the design of an intelligent forecasting model based on neural networks (NNs) that will enable the future value estimation for the critical meteorological parameters such as solar radiation and wind speed that greatly affect the efficiency and the overall functionality of the corresponding HYRES. During the previous years, NNs have been proposed as a powerful approach to estimate and predict the solar radiation as well as wind speed in different areas all over the world [2–9]. Taking into account that the solar radiation and wind speed time series as a dynamical system present nonlinear characteristics due to its dependency on meteorological parameters, such as temperature, water vapor, cloud, and water air condition [10], we propose a recurrent neural network (RNN) with nonlinear autoregressive architecture (NAR) as an enhanced forecasting model for solar radiation and wind speed time series. The recurrent network due to its characteristics (internal connections, feedback) and nature (internal memory) inherences more computational power and is better applicable to scenarios where the processing of some arbitrary dynamic input is required. The proposed model will have the ability of assimilating the past meteorological datasets and thus learning the local

behavior of the target parameters. For this reason, it will incorporate the feature of receiving the current meteorological data from locally installed sensing devices. Next, it will provide the corresponding future estimations through the combination of the past knowledge and current feedback. In such a sense, the usage of future meteorological values in combination with provided sensors data of the understudy system will boost its performance in terms of energy storage and availability.

Additionally, the presented model will also be applicable to an autonomous HYRES, where its estimations can be used by a central control unit in order to create in real time the proper power management strategies (PMSs) for the efficient subsystem utilization that can lead to the overall optimization. To do so, a generic network model is also described for the representation of the hybrid power generation systems in this work. Subsequently, the RNN when combined with the presented network model of HYRES serves as a novel framework for a generic approach aiming to facilitate the derivation of various PMSs in a simple and flexible way. As a result, the proposed framework will make the specific HYRES suitable for use as a stand-alone remote energy plant. As a proof of concept, the results of the proposed NN model for solar radiation and wind speed forecasting when applied to an available HYRES system are also presented. It is clear that the proposed RNN after training with meteorological data of the understudy area in our case Olvio of Xanthi in Greece and applied to the proposed HYRES of Systems Sunlight S.A. finally manages to enhance and optimize its PMS based on the provided solar radiation and wind speed prediction. The proposed HYRES system with the applied RNN with nonlinear autoregressive architecture for solar radiation and wind speed forecasting constitutes a fine example of engineering application of theoretical computer science toward the enhancement and optimization of its PMS. Despite the fact that NNs have been already introduced for the prediction of solar radiation and wind speed worldwide [2–9] and several different HYRES have been developed with various characteristics and prominent features as such a complete approach, the introduced system does not meet any other in the relevant literature to the best of our knowledge. In particular, there is no published work concerning the enhancement of energy management in stand-alone hybrid renewable energy systems (HYRES) that comprise photovoltaics, wind generators, electrochemical energy storage systems, and diesel generators, with weather forecasting. Furthermore, with almost no exception, the HYRES presented in the published literature employ very few (usually up to three) pre-specified PMS which hinder the efficient exploitation of renewable energy and storage systems to address the weather variability. The modeling approach proposed for the particular HYRES investigated

here allows the identification and easy implementation of a wide range of operating options in the form of many different PMS, prior to selecting the most appropriate alternative. The combined utilization of the predictive capabilities of the NN model proposed in this work together with the selection of the appropriate PMS provides important operating advantages compared to systems lacking such capabilities. It should be noticed that the existence of a diesel generator, which is utterly necessary for backup reasons, creates environmental problems and operational maintenance costs as well. More specifically, the operation of the diesel generator covers both load requirements and battery charging. In general, battery charging comprises a constant current (CC)–constant voltage (CV) profile. During CV operation, diesel operates in inefficient way due to the fact that its current must be continuously reduced to keep the voltage constant. Moreover, during the CV stage and due to the fact that the battery, wind generator, and photovoltaics are connected in a common DC bus, any available renewable energy must be rejected in order to keep the battery's voltage constant. This stage (CV charging by diesel generator) could totally be avoided if any available renewable energy produced in the near future can be predicted and the corresponding PMS applied.

The rest of the paper is organized as follows. In Sect. 2, the proposed RNN model is described in detail, and several simulation results are presented and compared with real data proving the efficiency of the model for forecasting solar radiation and wind speed. In Sect. 3, the mathematical framework for the generic representation of PMMS for a HYRES is given, while in Sect. 4, the aforementioned flexible PMS representation combined with the model's weather forecasting prediction successfully makes the specific HYRES suitable for use as a stand-alone remote energy plant. Finally, the conclusions are drawn in Sect. 5.

2 The proposed recurrent neural network

Having in mind that the estimation of the future weather conditions constitutes a particularly complex problem due to the nonlinear dynamics of the weather behavior, the computational paradigm of neural networks (NNs) was adopted for the design of the forecasting model. In specific, due to the nature of the problem that involves the estimation of the future values of certain meteorological variables, the adopted type was the *recurrent neural network* (RNN) [11–20]. They constitute some NN type where the unit connections form some directed cycle. This inherent characteristic creates some internal network state that greatly favors the exhibition of some dynamic temporal behavior. Unlike conventional *feed-forward neural networks* (FF-NN), RNNs

can use their internal memory to process arbitrary sequences of inputs. In general, RNNs are mostly preferred when applied to nonlinear problems offering high precision, due to their associative memory, resilient to noise, and input imperfections, as well as temporal dimension. Moreover, RNNs have been highly recommended when time-based problems are considered, where the usage of other nonrecurrent types of NNs is contraindicated [17].

As already mentioned, the RNN architecture differs from the FF-NN one to the point that except for the network inputs, it also takes into consideration its internal state. This internal state can be considered as a trace of the previously presented network inputs that have already been processed. This feature provides to RNNs the ability of learning the temporal sequential dependencies that may occur among the data of a time series. However, it is quite easy to understand the functionality of RNNs through a direct comparison with a simple FF-NN.

It is quite convenient to consider a simple layered architecture that consists of one input, one hidden, and one output layer. Following, Eq. (1) describes the mathematic relations that occur between the data of subsequent levels.

$$y_j(t) = f(\text{net}_j(t)), \text{net}_j(t) = \sum_i^n x_i(t)v_{ji} + \theta_j \quad (1)$$

where y is the layer output, j is the layer number, t is the time, net_j describes the layer state, and f is a differentiable output function. The variable n describes the total number of the network inputs x_i , whereas v_{ji} represents the connection weights and θ_j is a bias value. In case of a FF-NN, the input array x is propagated through the weights V that characterize the connection between the input and the hidden layers. Similarly, the propagation of the input array in a simple RNN is equally affected by the weights of the established connections between the nodes of the two neighboring layers. However, another factor that affects the propagation is an additional recurrent level that sends the previous network state through its own activation function and the corresponding connection weights U as presented by Fig. 1 and described by:

$$y_j(t) = f(\text{net}_j(t)), \text{net}_j(t) = \sum_i^n x_i(t)v_{ji} + \sum_h^m y_h(t-1)u_{jh} + \theta_j \quad (2)$$

where m expresses the total aggregate of the state nodes. The network output is defined in both cases by its own state and the weights W as follows:

$$y_k(t) = g(\text{net}_k(t)), \text{net}_k(t) = \sum_j^m x_j(t)v_{kj} + \theta_k \quad (3)$$

where g is the activation function of the output layer.

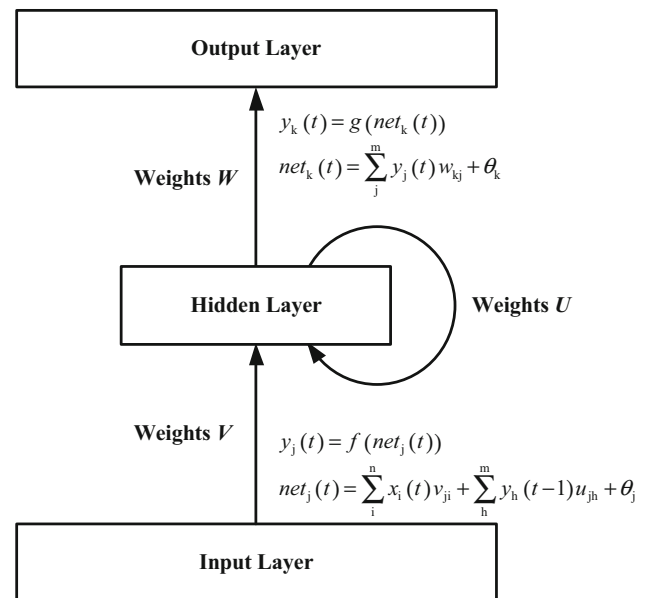


Fig. 1 Simplified recurrent neural network (RNN) structure diagram

The architecture of NN can vary in many ways, such as layers' type, layers' number, number of neurons per layer, neuron activation functions, layer interconnectivity, and number of inputs/outputs. Each combination will provide a network that behaves differently. For this reason, the selection of these parameters must be done very carefully according to the current application requirements. Two examples of different RNN network architecture are demonstrated in Fig. 2. The network of Fig. 2a is a fully interconnected RNN without any discrete input and output layers. Each neuron receives an input from the rest of the network units, as well as from itself through some feedback loop. The specific example refers to the *Elman NN* type which is widely known and used in the literature [19]. On the contrary, the second network in Fig. 2b constitutes a partially interconnected RNN that has some discrete input and output layers. Although there is no neuron self-feedback feature, there is a feedback loop connecting the hidden layer to the input one. This example is known as *Jordan ANN*.

The selected architecture of RNN for the current study was the *nonlinear autoregressive neural network* (NAR). This NN has been extensively used for statistical forecasting modeling of time series [21–24]. The specific model constitutes dynamic RNN that includes feedback connections between layers. NAR is based on the *linear autoregressive model* (AR), which is known for its effectiveness in modeling time series [24]. The equation that defines NAR functionality is:

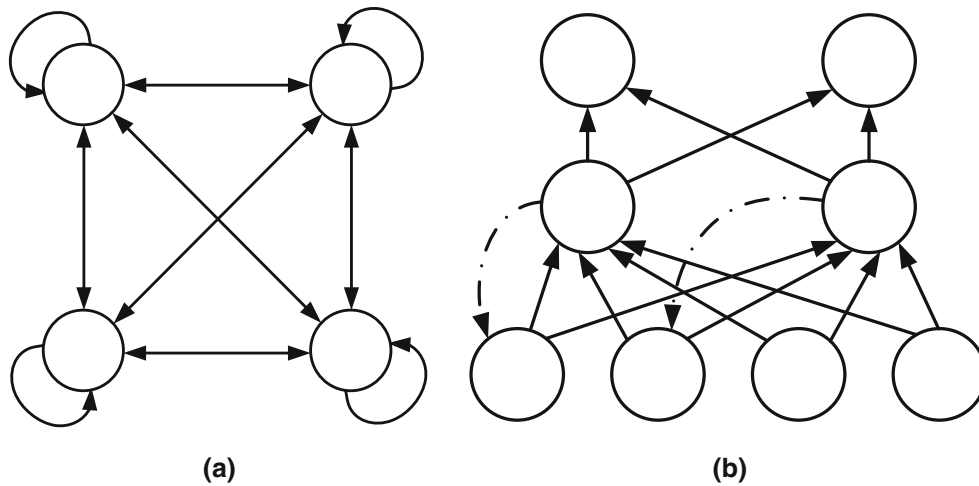


Fig. 2 Examples of different recurrent neural network (RNN) architectures: **a** fully interconnected network, **b** partially interconnected network of feed-forward architecture

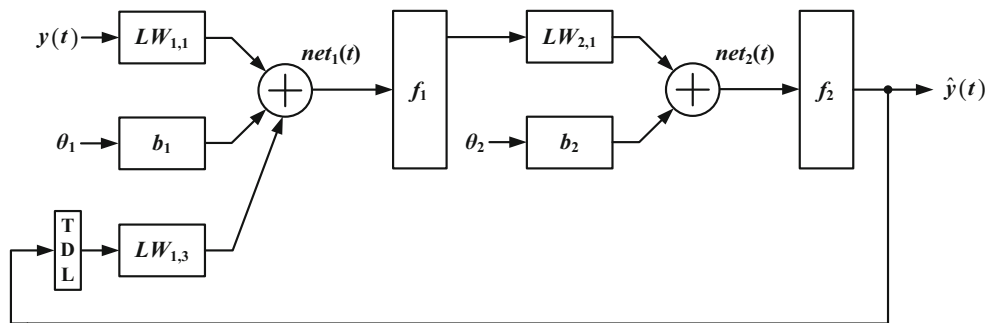


Fig. 3 Autoregressive model (AR) architecture

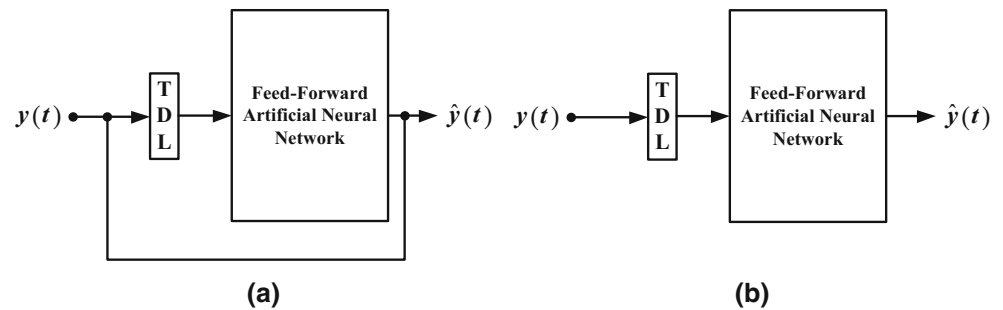
$$y(t) = f(y(t - 1), y(t - 2), y(t - 3), \dots, y(t - k)) \quad (4)$$

where $y(t)$ is the model output that depends on the k temporally previous values of the output signal. This is also represented in Fig. 3, which shows the block diagram of a two-layer AR model with feed-forward architecture. The purpose of the specific model was the estimation of function f . The input of the model can be a multidimensional array, while each layer has an additional bias input b_1 and b_2 for quicker convergence of the NN. Furthermore, the connection between two layers is characterized by the corresponding layer weight $LW_{i,j}$, where i refers to current layer and j refers to previous layer. Finally, the *time-delayed line* (TDL) expresses the time delay that is inflicted upon the output feedback data that are sent back to the input through the feedback loop. This feature enables the estimation of the temporal dependencies that may occur between the input and the output. This property is of great significance when trying to model systems that are described by time series.

In general, there are two different architecture options for NAR. Both these architectures include a time delay line

that was described above. However, the first one includes a feedback loop that sends the data from the output directly back to the input and is presented by Fig. 4a. Due to this output–input feedback connection, the specific architecture is regarded as parallel. The second option excludes this parallelism property and makes use of a more straightforward logic, as it is completely serial. As presented in Fig. 4b, the serial NAR architecture lacks any feedback property. The main difference between these two options relates to the training procedure. The accuracy of the training is higher in the second case, because through the serial feed-forward architecture, the network is fed only with real data. On the contrary, the parallel NAR combines both feedback and real data. This often has a negative effect at the network training accuracy as long as the output data are already processed. Another advantage of the serial model over the parallel one refers to its simplicity. The serial architecture produces more responsive models that are easier to implement and train faster. This is considered as a very significant feature when the NAR model is meant to be used in real-time applications.

Fig. 4 Basic nonlinear autoregressive network (NAR) architectures: **a** parallel with output feedback, **b** serial without feedback



For the needs of the current study, the proposed NAR model was initially trained through the adoption of the serial architecture. The basic structure of the model has three distinct layers: the input, the hidden layer, and the output. The utilization of a single hidden layer was decided upon the fact that in the literature, there are a lot of NN examples where such architecture provides enough computational power for confronting problems of similar complexity [2–8]. Moreover, in order to decide which NAR architecture and network size is the most suitable, a series of different tests were realized that confirm the suitability of the aforementioned structure and resulted to network efficiency. Making use of a small but representative dataset, a variety of networks was tested including NARs with multiple hidden layers and variable neurons numbers per layer.

2.1 Nonlinear autoregressive artificial neural networks

The proposed NAR model was designed and simulated through the use of the *Neural Time Series Tool* that constitutes part of the *Neural Network Toolbox* library of *MATLAB* software. The design of the NAR model has been done in accordance with the network architecture of Fig. 3. The main structure of the model is similar to the one of a multilayer FF-ANN. In specific, it consists of three discrete layers: the input, the hidden layer, and the output. As mentioned before, a single hidden layer was used, as it is considered to deliver some adequate computational power and network efficiency in combination with some good performance [5, 7]. The NAR model as designed with *Neural Time Series Tool* is presented in Fig. 5. The specific architecture does not include any output feedback property so as to deliver the best possible training results. This open-loop network was used for training purposes, as long as its output is not sent back to the input. In this way, every presented training pattern belongs to the training dataset and the NAR model assimilates only the real training data.

As presented in Fig. 5, each block has a specific number on its bottom side. This value represents the total number of neurons per layer. Both input and output have one neuron, whereas the hidden layer consists of 10 artificial neurons that are fully interconnected. However, due to the abstractive nature of the diagram, the connections between the neurons are not visible. The network input and output are described by the same variable $y(t)$. This happens because the designed NAR model outputs the same physical quantity that it receives on its input. The goal of the network training is to assimilate any existing temporal relations that may occur between the subsequent data points of the training time series. For this reason, no exogenous variable is involved into the training procedure. However, a time delay element is added to the hidden layer in order to create the necessary temporal difference between the input and the output data. The exact magnitude of this applied delayed is depicted in Fig. 5 and is equal to 72 samples. In this way, a time window is created pointing out the specified number of previous inputs that will be used by the network to estimate the next value of the given time series. The size of the applied delay and thus the time window can be altered according to the requirements of the current application and the time resolution of the target time series. Additionally, the activation function used for the hidden layer is the hyperbolic tangent, whereas a simple linear function was adopted for the output layer. Moreover, the synaptic weights and the bias values are represented by W and b blocks in respect. All the above settings were the results of a series of “trial and error” regarding the NAR training. These settings were found to be the optimal combination that provided the best results without requiring an excessive amount of system resources.

Following the network design, the available real meteorological data were used for its training. In accordance with the experimental setup that will be introduced in Sect. 3, the training time series were collected from the location of Olvio with coordinates 41.0249 N and 24.7885 E near the city of Xanthi in Greece. These datasets include solar radiation and wind speed measurements for the time period of the last 2 years, having a time resolution of

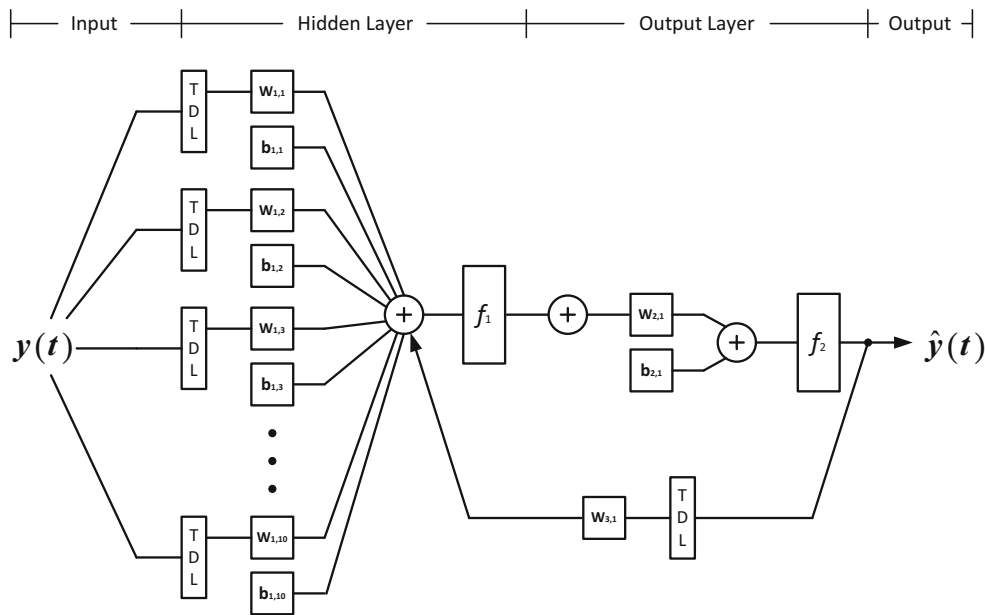


Fig. 5 Schematics of the proposed nonlinear autoregressive network (NAR). Each block has a specific number on its bottom side. This value represents the increased number of neurons per layer. Both input and output have one neuron, whereas the hidden layer consists

of 10 artificial neurons that are fully interconnected. The network input and output are described by the same variable $y(t)$. Moreover, the synaptic weights and the bias values are represented by W and b blocks in respect

5 min. From these datasets can be easily noticed that every month has its own characteristics, present some nonlinear behavior, and can potentially present great differences from 1 day to the other. Especially in case of solar radiation, the differences between months are considered normal due to the different season conditions. Furthermore, some intense fluctuations that take place during the same month often are caused by the occurrence of some extreme weather phenomena. The time resolution of the measurements is equal to 1 sample per 5 min. This means that 288 samples are available per day. However, for the needs of the specific study, three additional datasets with lower resolution were created and tested, with 96, 24, and 4 samples/day. The generic form of the tested data vectors for the corresponding samples is given by the formula: $x = [x_1, x_2, x_3, \dots, x_n]$ where index n takes values [4, 24, 96, 288], respectively, and the data sample $x_i \in [0, 1300]$ W/m² in case of solar radiation or $x_i \in [0, 13]$ m/s in case of wind speed, respectively. Through the realization of different training scenarios for the four datasets, it was proved that 24 samples/day provided the best results in terms of accuracy and computational speed, without lacking any significant information comparing to the initial dataset. A possible explanation in regard to this fact is the fluctuation of the data is not so abrupt while their correlation was high especially in case of solar radiation where the resulting matching between the actual data and the predicted was even better. On the other hand, the possibility of overfitting and thus the NN to memorize the training examples

resulting to false generalization on new examples was proven small, after testing many various initial conditions with different datasets that resulted in most cases in sufficient and robust NN performance. All these are discussed in more detail across the section. By adopting the 24 samples/day dataset with an input delay line of 72 samples, the proposed NAR model receives the measurements of the three last days as an input to estimate the future value. It should be noticed that the proposed ANN uses the last 72 samples (1 sample = 1 h) to estimate the next sample as output. Moreover, during every iteration, the synaptic weights of ANN are being dynamically updated in order to optimize its output and produce the next sample. For the production of full 24-h estimation, 24 iterations are needed, and in correspondence, the synaptic weights of the ANN are being updated 24 times. Please notice that for these new ANNs, the architecture and the resulting complexity of the network are not meant to alter during every iteration.

The data were imported to MATLAB and formatted as a “struct” variable. The specific data structure is a quite common and efficient way of manipulating time series in Neural Network Toolbox. In order to make the data suitable for insertion into the NAR model, the command “preparats” was firstly used. This function simplifies the normally complex and error-prone task of reformatting input and target time series. It automatically shifts input and target time series as many steps as are needed to fill the initial input and layer delay states. In case that the network has open-loop feedback, then it copies feedback targets into

the inputs as needed to define the open-loop inputs. Each time a new network is designed, with different numbers of delays or feedback settings, it can be called to reformat input and target data accordingly.

Additionally, before feeding the designed model with the training dataset, two optimization techniques were used in order to improve the learning efficiency. The first one processes the values of the given dataset and then removes the any rows with constant values. This means that any measurement values that remain unchanged for some time period will be discarded from the final training set. For example, solar radiation during night time always equals to zero, thus disrupting the consistency of the training patterns. In practice, this can deteriorate the training accuracy because it imports zero values. After the adoption of this technique, the final training dataset contains only the necessary measurements that describe the variation in solar radiation during the sunlight hours. The above procedure was implemented by using the “*removeconstantrows*” function of the Neural Network Toolbox. While the first technique was mostly applied to solar radiation measurement, the second technique is used to process the training data by mapping row minimum and maximum values to [0, 1]. This helps in simplifying the synaptic 73–96 updating process as long as the target value variation is significantly limited maintaining at the same time all the necessary information. This normalization also prevents uncommonly high training values from affecting the overall NAR training accuracy. It was realized through the “*mapmin-max*” command.

Following the learning dataset was divided into the necessary subsets: training, validation, and testing. Usually, this separation procedure is done in a random way. However, this would ruin the essence of the time series itself. For this reason, the training dataset was divided using the “*divideblock*” function. This function targets into three sets using blocks of indices without ruining the sequence properly. In this way, the designed NAR can assimilate the temporal dependencies that characterize the given meteorological measurements. Another significant factor that may affect the overall training efficiency is the definition of the subsets ratios. The adopted ratios were equal to 75, 15, and 10 % for the training, validation, and testing subsets in respect. It is very important that training subset includes the largest portion of the available dataset in order to achieve the best possible training results and network learning.

Next, the *Levenberg–Marquardt* back-propagation algorithm was adopted for the training of the proposed NAR model. The specific technique is widely used for the NN training as it is considered as one of the most efficient solutions. During the learning procedure, it implements the error back-propagation method, whereas it updates the

synaptic weights and bias values according to Levenberg–Marquardt optimization [25]. It can be used to train any network as long as its weight, net input, and transfer functions have derivative functions. There is a great variety of different approaches, such as *Bayesian Regulation Back-propagation*, *Gradient Descent Back-propagation*, and *BFGS Quasi-Newton Back-propagation* [26]. Bearing that in mind, all the above learning algorithms were tested in training the proposed model. However, deliver the best training results in combination with the quickest convergence were achieved through Levenberg–Marquardt algorithm.

The success of the network training does not depend on the available dataset and the learning algorithm [27]. Defining the basic training parameters and the ending conditions of the learning procedure is also a significant task. The values of these parameters, as well as the validity of the ending conditions, are checked during every training epoch, and in case that some of them is satisfied, the network training comes to a complete halt. For this reason, the definition of these parameters and conditions is very important for the achievement of the desired training results. In specific, these parameters include the training performance function, the maximum number of training epochs, maximum allowed training time, the desirable training performance, the minimum training performance gradient, and the maximum numbers of validation fails. Each of them plays a different role in defining the progress of the network training procedure.

At first, the performance function is one of the most significant training parameters. It defines the way that the training performance is calculated during every epoch. For the needs of this study, the “mean squared normalized error” (MSE) performance function was adopted [18]. It measures the network performance according to the mean of squared errors. In other words, it calculates the produced error that is the distance between the NAR actual outputs and the desired target values. Next, the maximum duration of the NAR training was specified in terms of epochs and time. The maximum number of epochs was equal to 10^3 , as specified by default in MATLAB, whereas the maximum time period was defined to be 60 min. The long duration of the training procedure is often a sign that either the designed network cannot fully assimilate the given information, or the rest of the training parameters were not selected very carefully. Thus, these conditions are meant to stop the training in case that it requires a lot of epochs/time. The selection of the specific values depends both on the target system and the designed network complexity. In any case, they both should be set quite high in order to avoid prematurely stopping the network training that may result in some unsatisfactory results.

Following, one of the most significant constraints is the desired training goal in terms of training performance. Once this goal has been reached, the network training comes to a halt. For the needs of the current model, a training goal of 10^{-3} was defined. Additionally, another significant parameter is the gradient of the performance function output. Its magnitude indicates how much is the training performance altered from one epoch to another. It is always a good idea to define a minimum value for this gradient, because after reaching a certain training performance value usually a great amount of time is required to further improve. Most of the times, this improvement does not compensate for the extra training time and system sources. To avoid such phenomena, the minimum gradient limit was defined as equal to 10^{-5} . Finally, the maximum number of validation fails was equal to 6. The occurrence of successive validation fails often means that the designed model has reached the maximum possible assimilation level for the given training dataset and network architecture. This constraint constitutes the most usual stopping condition of the network training.

To sum up, choosing the right settings for the above training parameters is considered as a significant step toward the achievement of the successful network training. Due to the fact that they can affect at a great degree the outcome of the training procedure, they should be chosen wisely. Unfortunately, similarly to the problem of network design, here also there is some lack of methodologies that could provide some necessary guidance. Every problem has its own requirements, and it is quite difficult to find always the right setting combinations for these parameters. For example, defining some strict conditions for the training interruption could cause some premature training stopping that might result into some insufficiently trained network. On the contrary, adjusting these settings to be quite loose can result to the “*overfitting*” problem, where the network learning is trapped into local minima excluding some part of the available information. Both these scenarios end to some poorly trained networks with low efficiency.

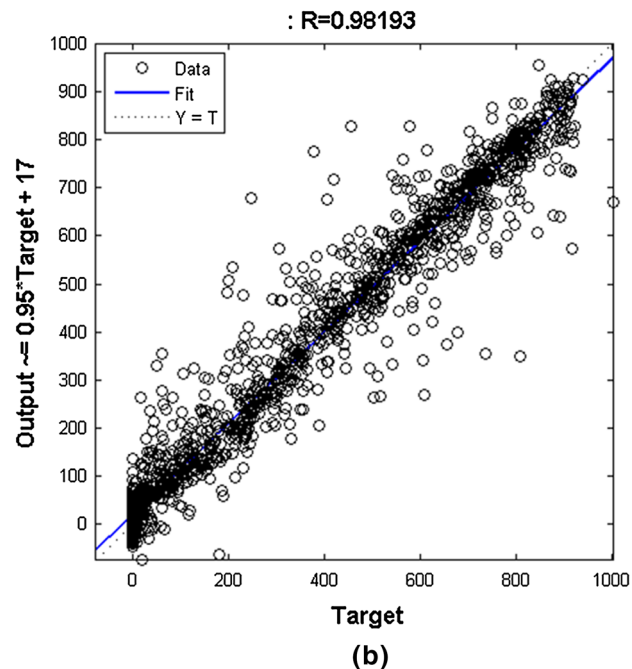
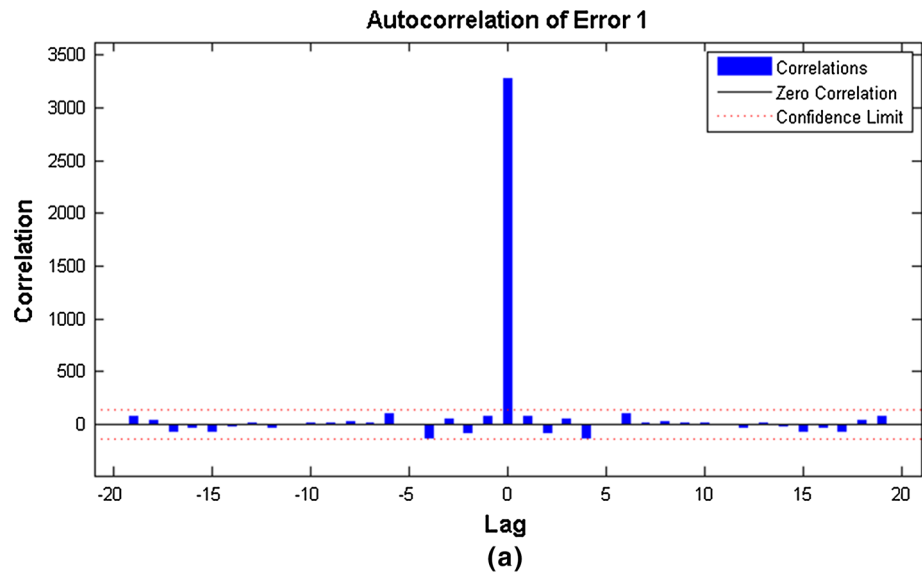
Finally, the training results were quite satisfactory bearing in mind the complexity of the target system. In specific, the final MSE was equal to 70 W/m^2 that is considered quite decent regarding the actual solar radiation datasets values that may vary up to 1300 W/m^2 . In Fig. 6a, the error autocorrelation diagram is depicted. It can be remarked that there is no significant correlation between subsequent errors, whereas there is no value exceeding the confidence limit. Moreover, the linear regression diagram between the NAR estimation and the target values is presented in Fig. 6b, and the achieved regression coefficient was equal to $R = 0.98193$.

After the end of the NAR training procedure, during which the serial network architecture was adopted, there was a slight change in its structure. In specific, a feedback connection was created between the NAR output and its input. This conversion was previously explained in Fig. 4a, b. The serial model is always used for the network training instead of the parallel, due to the fact that it presents better learning results. On the other hand, the already trained parallel architecture can provide some multistep future estimation. At first, the network is supplied with the 72 samples that fill the time-delayed line. After the NAR model produces its first output, this value is redirected through the feedback connection back to the input in order to supplement the corresponding dataset. The oldest sample is discarded, and the time delay window slides one value further toward the future, thus including the first estimation. During every step, the time delay window is populated with the last 72 samples. Through this iterative procedure, a multistep estimation can be realized. The exact iteration number defines the magnitude of prediction horizon. In practice, the future prediction horizon was defined to be equal to 24 samples. In other words, the proposed NAR model receives the measurement samples of the last 3 days in order to deliver the estimation for the corresponding values of the next day.

Following, the trained NAR model was initially tested on real solar radiation data from the location Olvio (41.0249 N, 24.7885 E), Xanthi, Greece, for the creation of next-day estimations. The input datasets are real random data that were acquired from the same location as the training set, but were completely excluded from the network training procedure. The corresponding testing results can be found in Fig. 7c, d. The actual solar radiation values are represented by the blue line, whereas the predicted values by the red line. Samples 1–72 constitute the model input, and samples 73–96 are the produced future estimations. For this reason, the most significant part of the diagram is the comparison between the two representations regarding the actual and the predicted values during these last 24 samples. However, for demonstrative reasons, the rest of the samples were included.

In the view of the foregoing in Fig. 8a–c, some more detailed forecasting results considered longer time estimations based on the proposed NAR model output for the location Olvio (41.0249 N, 24.7885 E), Xanthi, Greece, are shown. Once again, data are referred to the solar radiation, while the number of samples is now 648 for Fig. 8a, b and 672 for Fig. 8c. Nevertheless, the corresponding time for each one of them is 60 min, and the time resolution is 27 dates for Fig. 8a, b and 29 for Fig. 8c. It is reasonable for someone to assume that the resolution and the estimation is much more difficult compared to the previous cases, but once again the results show the quantitative and qualitative

Fig. 6 **a** Error autocorrelation diagram. **b** Diagram of linear regression between NAR output and target values



agreement of the simulation results when compared to the real data.

Finally in Figs. 9, 10, 11, and 12, we present detailed results during testing of ANN for different months one per season, for readability reasons, of year 2010. For example, in Fig. 9a, there is the solar radiation estimation as output of the NAR model when compared with the real solar data measurements for January. In correspondence, the resulting power of understudy HYRES is shown in Fig. 9b. In Fig. 9c, d there are similar diagrams now referring to daily prediction, while for readability reasons the input data are not presented. It should be clarified that the specified day is one selected randomly as a proof of concept from the January data. Finally, in Fig. 9e, f, the NAR model results for wind

speed and the resulting power of the understudy HYRES are presented. Once again, in the last two Fig. 9g, h, the produced results of the NAR model and comparison with real data for one randomly selected January day are shown. The rest of Figs. 10, 11, and 12 represent the exact same results for one month selected randomly of each season of 2010. As a result, the presented Figs. 9, 10, 11, and 12 correspond to four different seasons of one full-year data exploiting the efficacy and the robustness of the under test RNN to predict the solar radiation and provide monthly and daily estimations of the solar radiation and wind speed, respectively, and the corresponding PV and wind turbine outputs. Moreover, the presented results refer to randomly selected days and months of the aforementioned seasons, thus indicating the

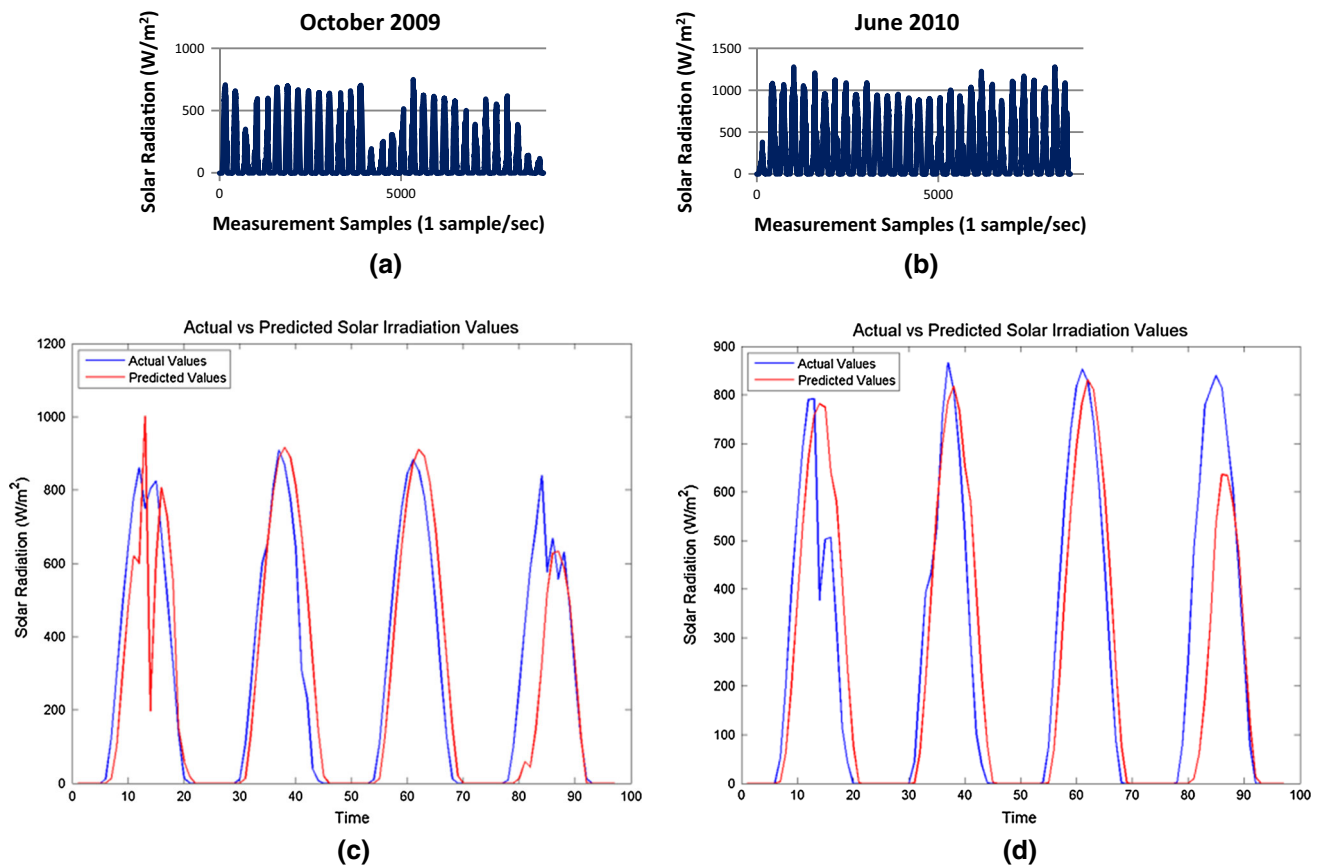


Fig. 7 Real solar radiation measurement samples for: **a** October and **b** June, and **c**, **d** comparison between the produced forecasting results (*red line*) and corresponding real solar radiation measurements on the horizontal level (*blue line*) for two different time periods (color figure online)

adaptability of the proposed RNN under quite different case studies in terms of solar radiation and wind speed.

The above simulation results can help us reach some conclusions about the effectiveness of the proposed NAR and the accuracy of its produced future estimations. In specific, from the results of Fig. 11a–c, it can be seen that the samples that constitute the network input present some unexpected fluctuations that may be connected to extreme meteorological conditions. Moreover, it should be further noticed that although time series of solar radiation and wind speed, as well as the corresponding climate phenomena, is different, we are targeting in producing a robust NAR model which could be possibly applied to predict both, although this is by default a hard to go for weather forecasting application. However, the actual NAR model estimation does not seem to differ much from the real solar radiation data, and in almost all the cases, the same applies for the wind speed. This happens due to the fact that the network input includes the data from the past 3 days. So, in case that something unusual happens, this dataset still contain some information depicting the general meteorological conditions of the current time period. In addition, it can be easily observed from the whole set of the simulation results that the proposed NAR model succeeds in

delivering the periodicity of the target phenomenon. The accuracy of the estimations also appears to be quite satisfactory in most cases. However, in the previous figure, there are some spots where the estimation does not seem to be quite accurate. Bearing in mind both the complexity and the dynamic character of the target system, it can be quite easily realized that these error can be considered as normal. Moreover, the wind speed tends to present some extremely non-linear behavior in comparison with other meteorological measures. For this reason, as already mentioned, some level of inaccuracy should be expected at some extent. Nevertheless, one of the primary goals of the specific case study as shown in Sect. 4 is to enable a cost-effective estimation solution coupled with efficient energy management strategy for the forthcoming energy supply of the presented hybrid system. Taking that into consideration, the generated estimation is based on training on historicized local data, enabling the prediction of the future energy supply based on local conditions. Still in cases that the provided wind estimation seems to be less accurate than usual, the system is not greatly affected due to the fact that it strongly relies to more than one energy sources. Furthermore, it should be noted that due to the reduction in the training dataset and the proper sizing of the

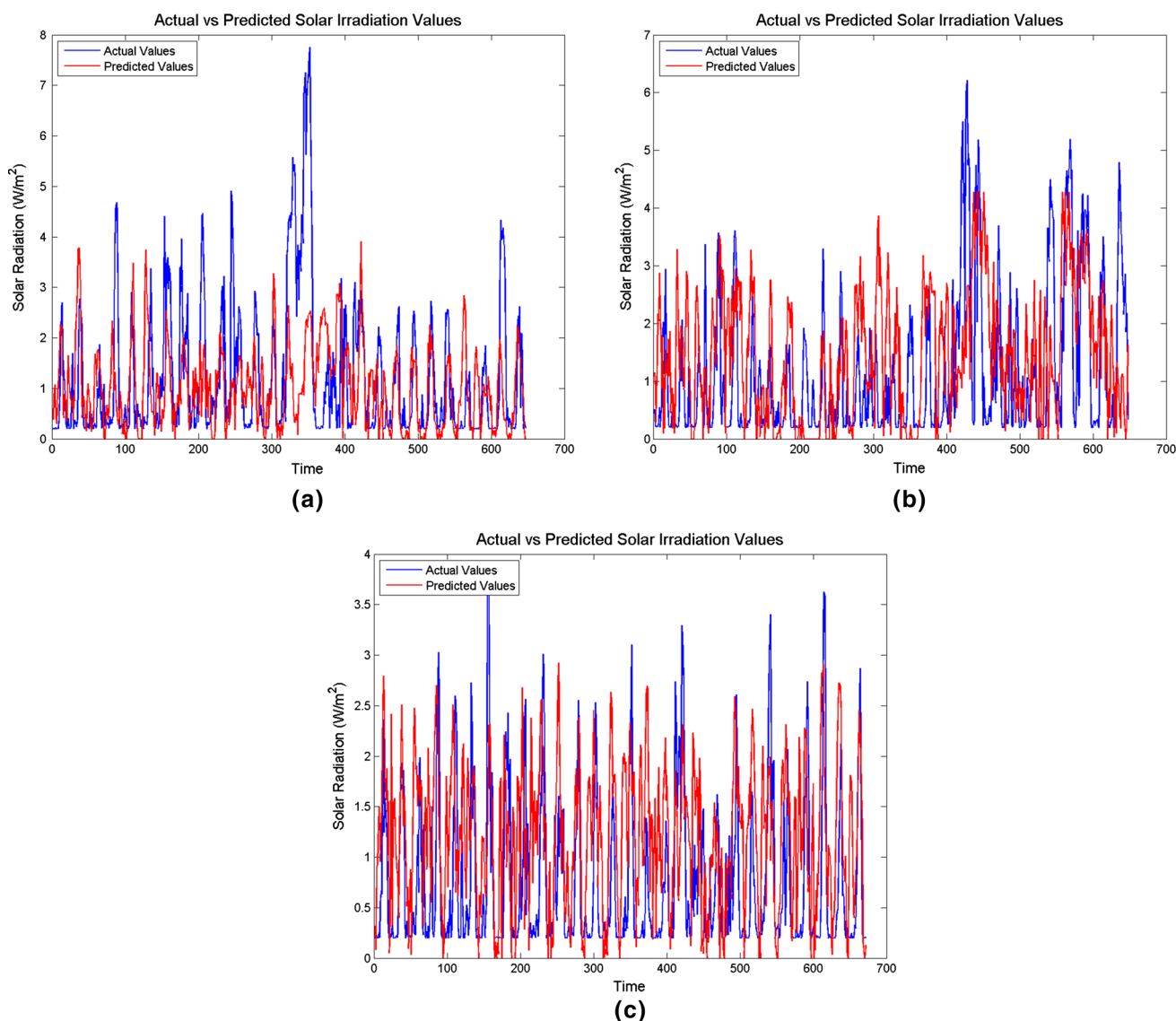


Fig. 8 Real solar radiation measurements on the horizontal level (*blue line*) compared with produced forecasting results (*red line*) for three randomly selected different months (color figure online)

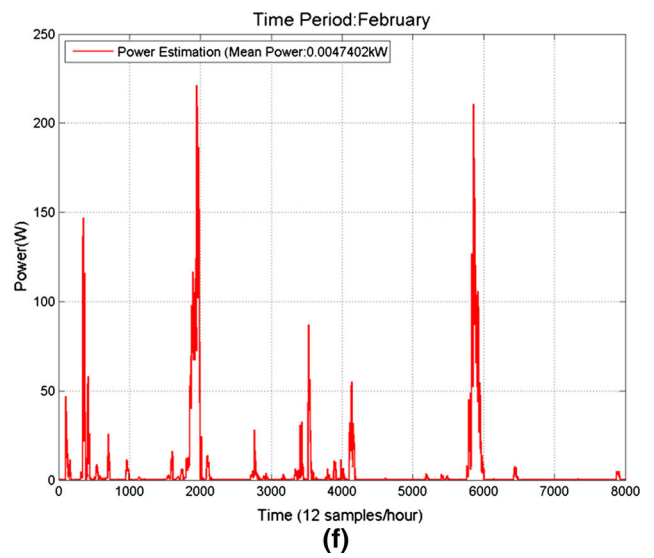
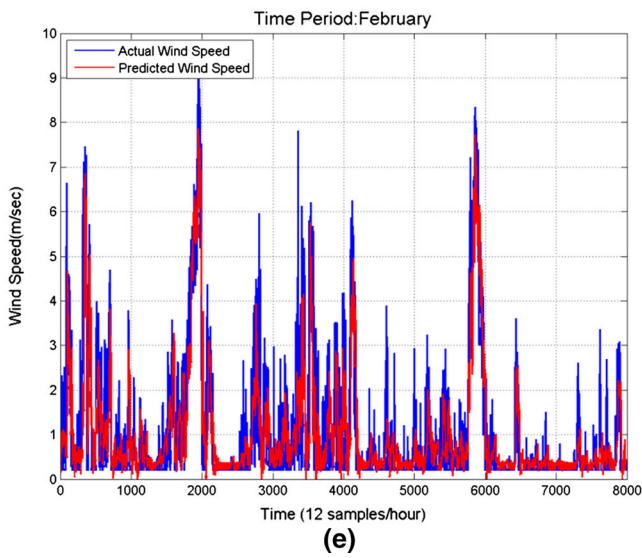
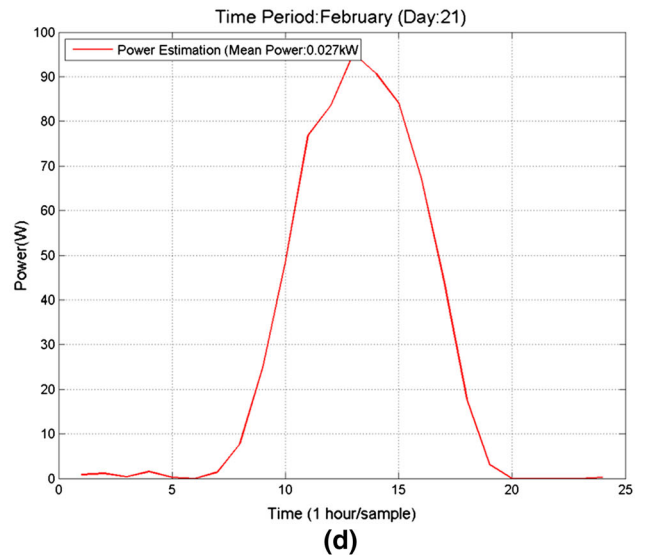
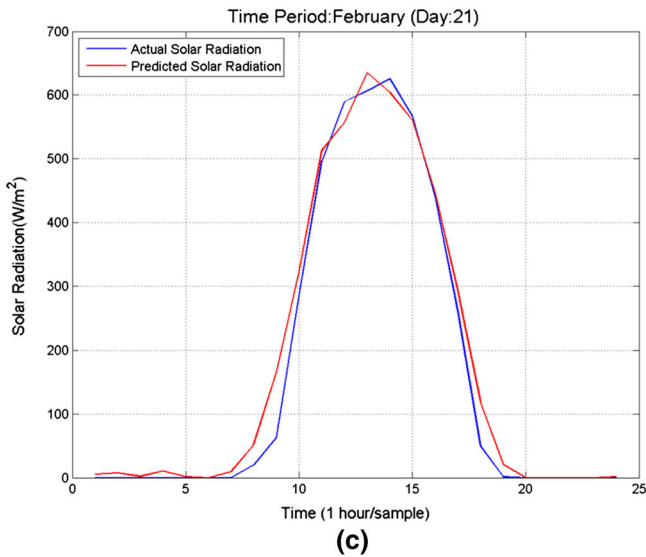
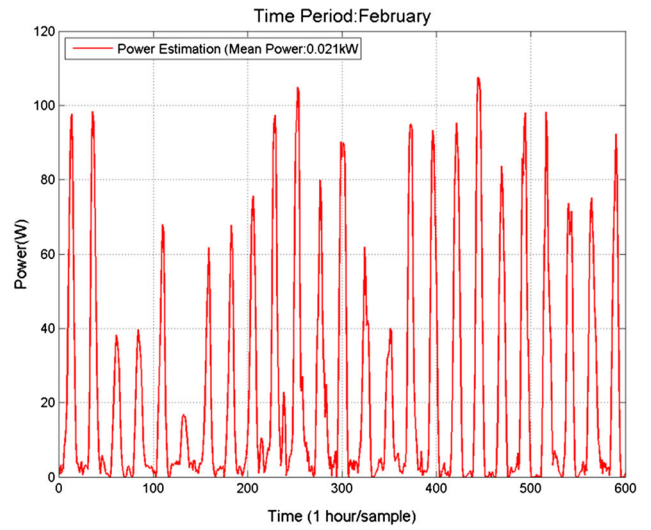
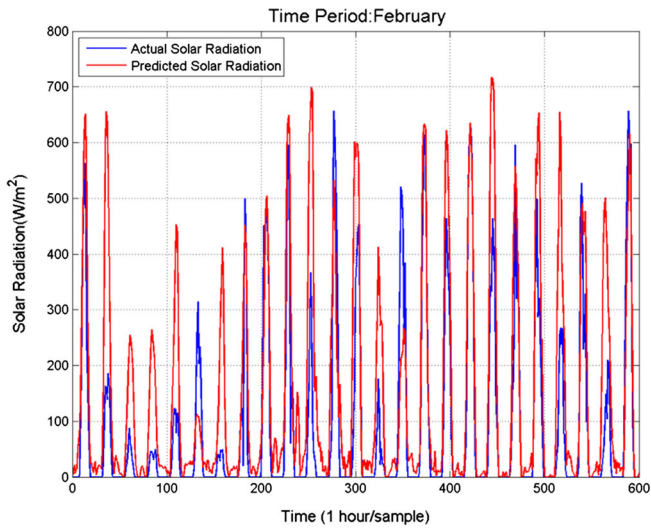
network, the proposed NAR has an especially fluid and swift functionality. It only requires some seconds to produce the corresponding future estimations. This feature is considered as very important in real-time applications, such as the on-the-fly HYRES optimization.

3 Efficient representation of energy management strategies

In order to effectively utilize the available renewable energy, a well-described PMS is necessary that will take into consideration the possible interconnection and energy distribution possibilities. The appropriate power distribution that changes dynamically due to the

Fig. 9 Data and modeling results for February: **a** monthly estimation (1 h/sample) of the active and predicted solar radiation (W/m^2), **b** monthly (1 h/sample) power estimation (W) of the PV output provided the mean power estimation, **c** daily estimation for 24 h of the solar radiation (W/m^2), **d** daily for 24 h of power estimation (W) of the PV output provided the mean power estimation, **e** monthly estimation (12 samples/h) of wind speed (m/s), **f** monthly (12 samples/h) power estimation (W) of the wind turbine provided the mean power estimation, **g** daily estimation (12 samples/h) of wind speed (m/s), **h** daily (12 samples/h) power estimation (W) of the wind turbine provided the mean power estimation

influence of the weather conditions is an important aspect for the effective operation of a HYRES system. In Fig. 13, a conceptual representation of the proposed scheme is provided that shows how the RNN and the PMS are connected. In particular, as shown in Fig. 14,



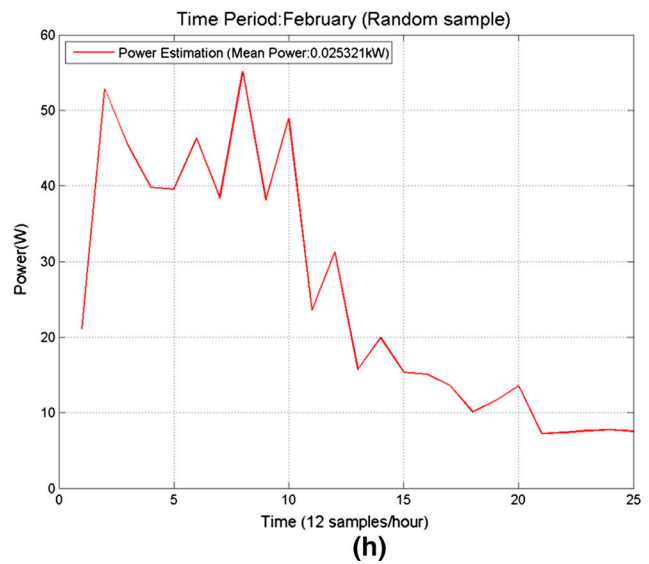
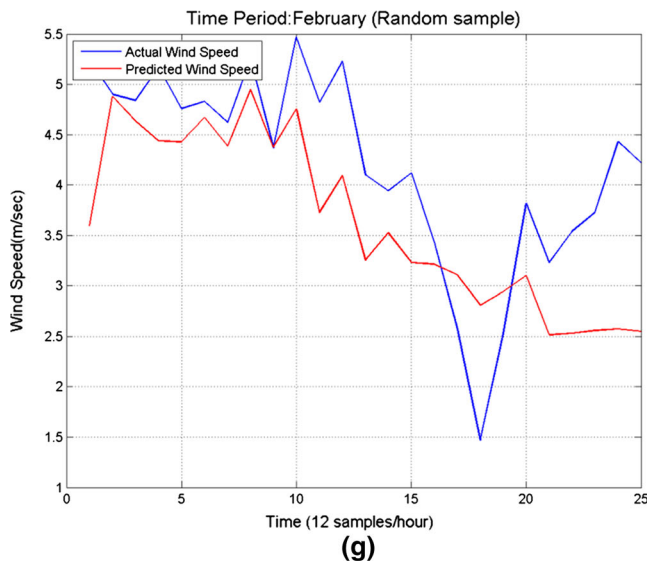


Fig. 9 continued

the RNN may provide a vector of weather forecasts (e.g., solar irradiation or wind speed data points) used for the resulting vector of PV and WG power forecasts. The data obtained are used by the PMS to affect the parameters that determine the appropriate instant for activation/deactivation of different subsystems. This is implemented through binary parameters and logical propositions used within the PMS to regulate the connection between the nodes of the graph emulating the interactions among different HYRES subsystems. For example, in the case of weather data, if the output of the RNN forecasts very limited or no solar irradiation, the logical rules comprising the PMS are first checked to investigate the operating status of the subsystems (e.g., availability of energy in battery or hydrogen in tanks and activation of fuel cell in the previous instant). Once the decision is taken regarding the activation of the appropriate subsystem (e.g., diesel generator or fuel cell), a binary parameter is used to request its connection with the battery in order to ensure that the requested load demand is satisfied. In the case of using the forecast of RNN to decide about power availability, this is also directly funneled to the PMS and associated with the appropriate binary parameter. For example, the option of directly using the corresponding power forecasts may be considered for the case of the diesel generator which is very important for the operation of the system. Figure 12 shows how the forecasted (expected) power availability or deficit from the PV and WG is directly associated with the binary variable and therefore the PMS regulating the activation of the diesel generator (i.e., connection with the battery). Therefore, in this section, we will review the representation of PMSs as described in [28] where the

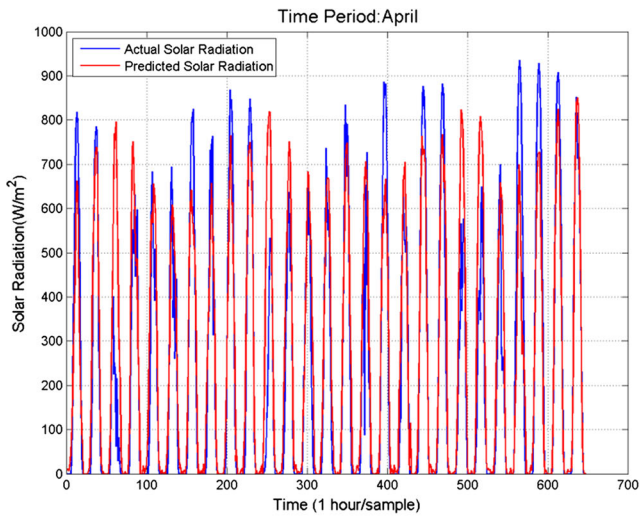
Fig. 10 Data and modeling results for April: **a** monthly estimation (1 h/sample) of the active and predicted solar radiation (W/m^2), **b** monthly (1 h/sample) power estimation (W) of the PV output provided the mean power estimation, **c** daily estimation for 24 h of the solar radiation (W/m^2), **d** daily for 24 h of power estimation (W) of the PV output provided the mean power estimation, **e** monthly estimation (12 samples/h) of wind speed (m/s), **f** monthly (12 samples/h) power estimation (W) of the wind turbine provided the mean power estimation, **g** daily estimation (12 samples/h) of wind speed (m/s), **h** daily (12 samples/h) power estimation (W) of the wind turbine provided the mean power estimation

microgrid was seen as a graph, and the flow of power and hydrogen within was described through flow sheets. More specifically, each device in the microgrid is seen as a node of a graph and its connection as an edge in Fig. 14 (it should be noticed that the full names of the abbreviated corresponding parameters and subsystems as found in Fig. 14 are provided in full detail in Table 2 for readability reasons). Table 1 presents the specifics of the selected systems that resembles the actual configuration of the HYRES located at Olvio, Xanthi.

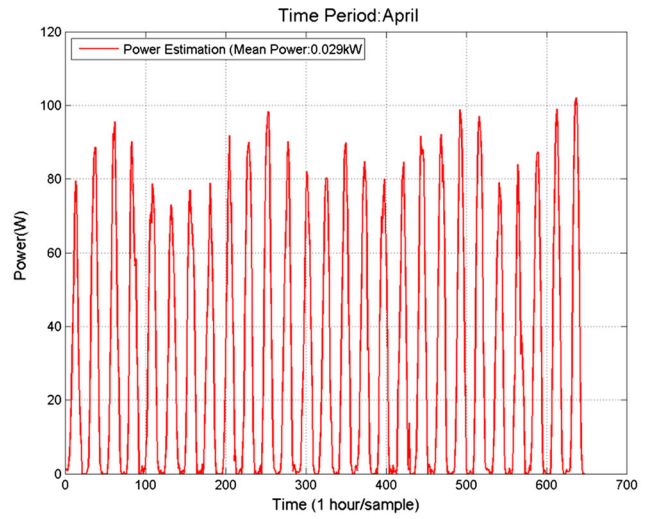
In our system, the flows between the nodes can be in various states like electrical energy (POW) or hydrogen in high pressure (H2P), and hence, the input to each node for each state j is given by:

$$F_n^{In,j}(t) = SF_n^j(t) + \sum_{l=1}^N \varepsilon_{l \rightarrow n}(t) F_{l \rightarrow n}^{Out,j}(t) \quad (5)$$

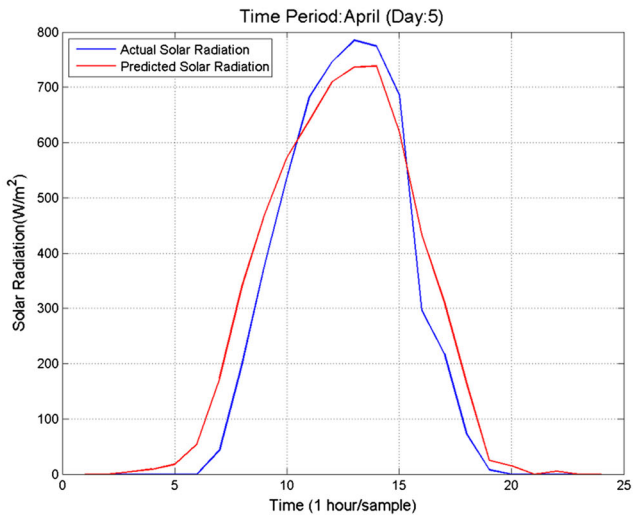
where $F_n^{In,j}(t)$ is the input to node n at the instant t , $SF_n^j(t)$ are external inputs, $F_{l \rightarrow n}^{Out,j}$ are the outputs of the other nodes, $\varepsilon_{l \rightarrow n}(t)$ are binary variables that determine the connection of a specific edge, and N is the number of nodes in the



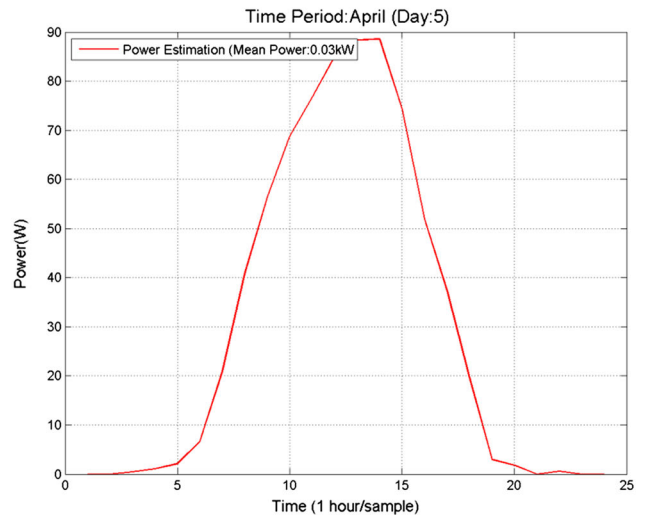
(a)



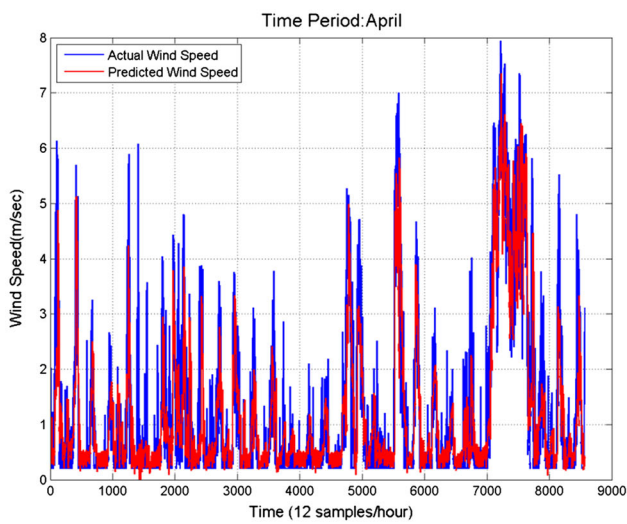
(b)



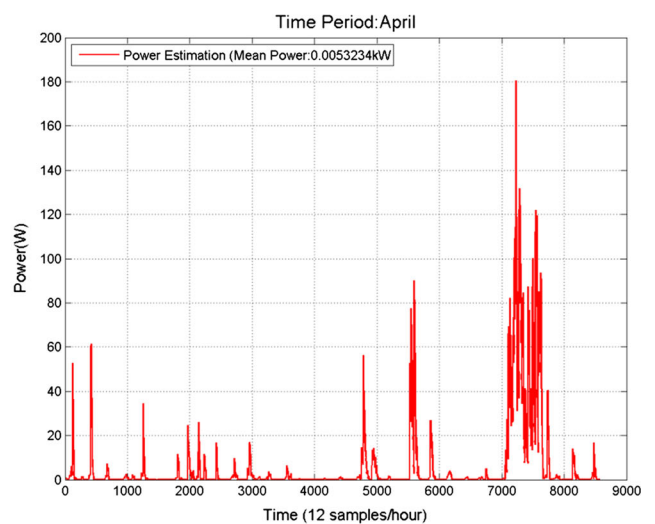
(c)



(d)



(e)



(f)

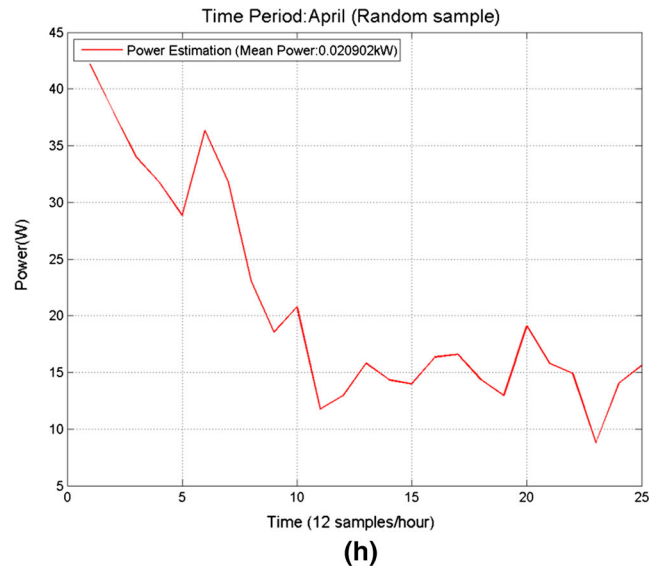
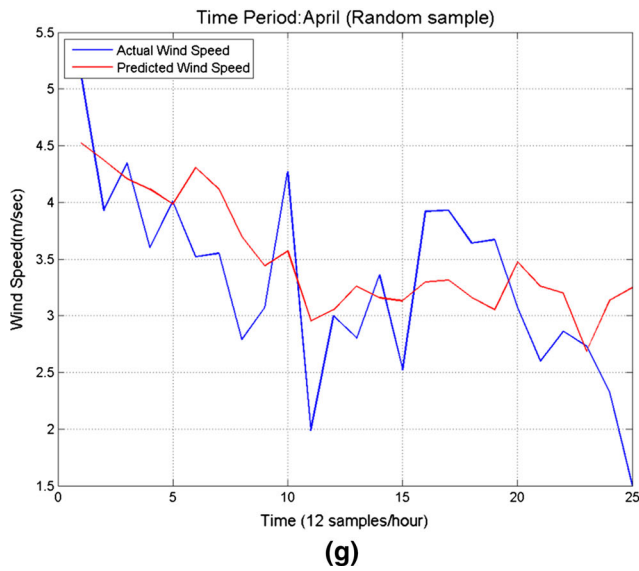


Fig. 10 continued

graph. For example, in the case of the battery, Eq. (5) can be written as:

$$\begin{aligned}
 F_{BAT}^{In.POW}(t) &= \sum_{l=1}^N \left(\varepsilon_{l \rightarrow BAT}(t) F_{l \rightarrow BAT}^{Out.POW}(t) \right) \\
 &= \varepsilon_{FC \rightarrow BAT}(t) F_{FC \rightarrow BAT}^{Out.POW}(t) + \varepsilon_{RES \rightarrow BAT}(t) F_{RES \rightarrow BAT}^{Out.POW}(t) \\
 &\quad + \varepsilon_{DSL \rightarrow BAT}(t) F_{DSL \rightarrow BAT}^{Out.POW}(t)
 \end{aligned}
 \tag{6}$$

where $F_{RES \rightarrow BAT}^{Out.POW} = F_{PV \rightarrow BAT}^{Out.POW} + F_{WG \rightarrow BAT}^{Out.POW}$.

The binary variables that determine the connection can be defined as

$$\varepsilon_{l \rightarrow n}(t) = L(\varepsilon_{l \rightarrow n}^{Avl}(t), \varepsilon_{l \rightarrow n}^{Req}(t), \varepsilon_{l \rightarrow n}^{Gen}(t))
 \tag{7}$$

where L is a logical operator (like AND, OR, ...) and $\varepsilon_{l \rightarrow n}^{Avl}(t)$, $\varepsilon_{l \rightarrow n}^{Req}(t)$, $\varepsilon_{l \rightarrow n}^{Gen}(t)$ are three binary variables that determine the availability, the requirement, and other general conditions necessary to activate the connection l to n . In general, the activation of a connection (from node l to n) depends on logical propositions c_i that can be described by binary variables ρ_i . For example, for the activation of the FC in order to supply power to the battery, we have $c_{FC \rightarrow BAT}$. There is a requirement for energy to be delivered to the battery which in terms of the ρ variables can be written as:

$$\rho_{FC \rightarrow BAT}^{SOC(t)} = \left[SOC(t) < Str_{FC \rightarrow BAT}^{SOC(t)} \right]
 \tag{8}$$

where the numerical variable $Str_{FC \rightarrow BAT}^{SOC(t)}$ defines the lack of available energy in the battery, and SOC is the state of charge. In case there is a hysteresis zone (as it is usually the case in such systems), then Eq. (8) can be written as:

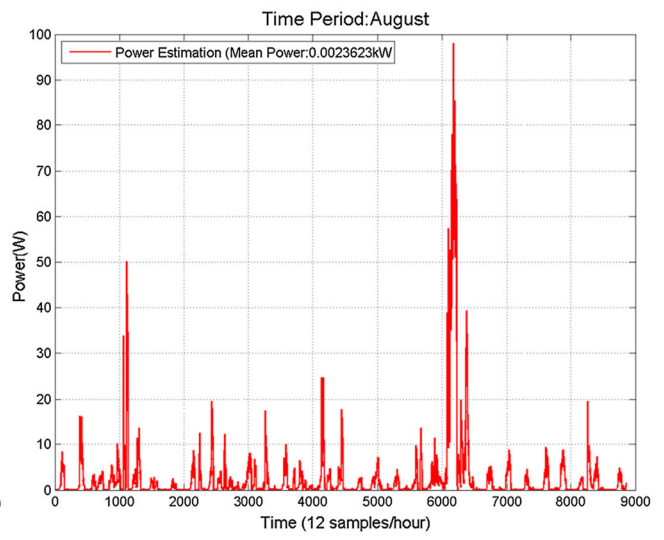
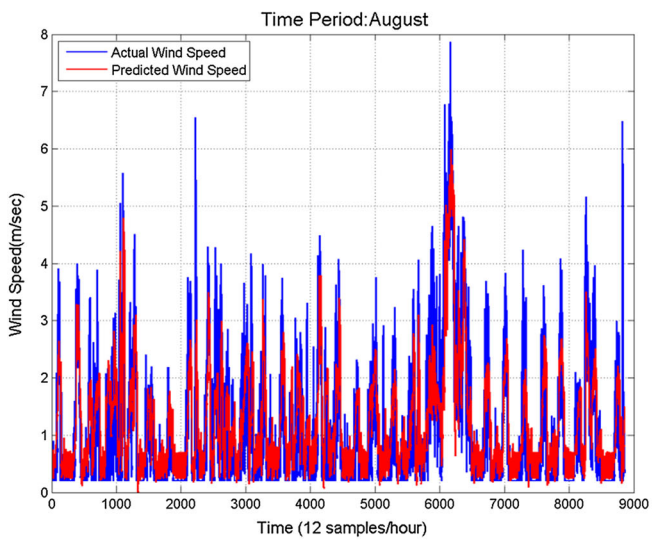
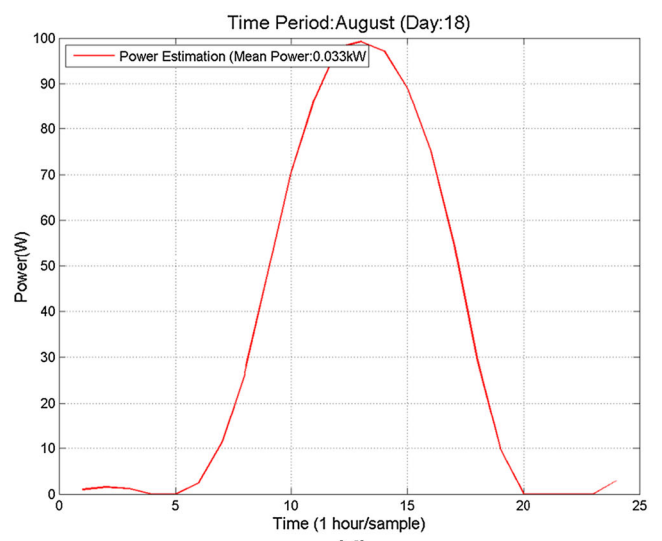
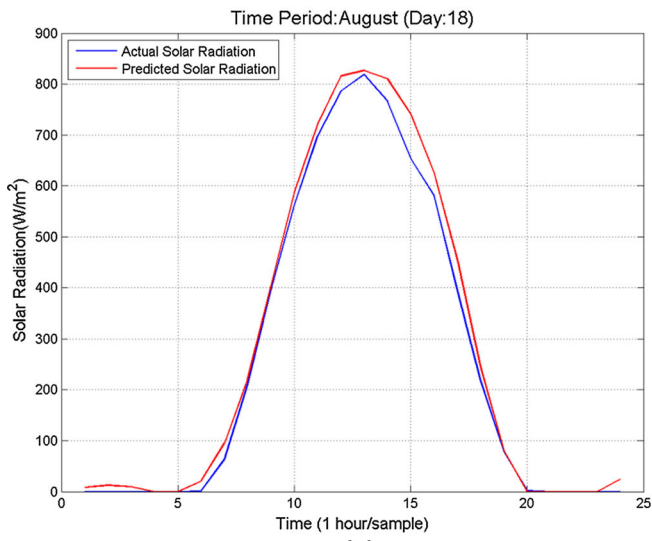
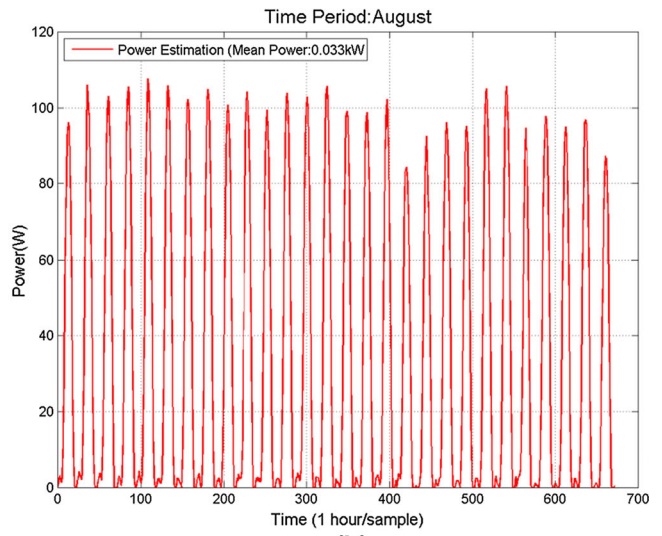
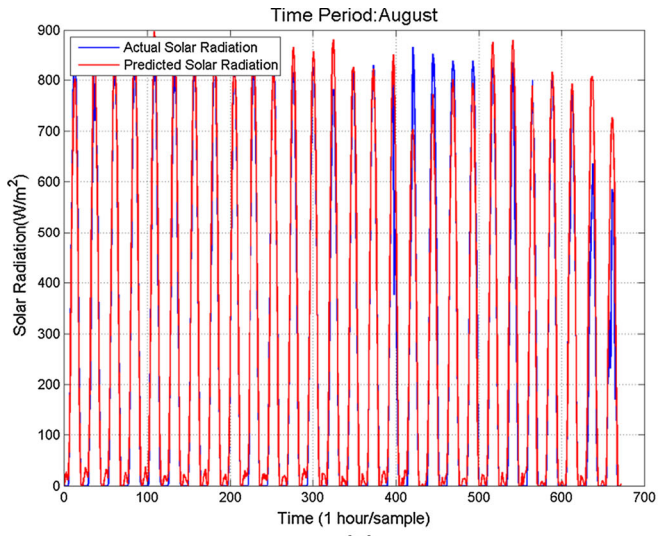
Fig. 11 Data and modeling results for August: **a** monthly estimation (1 h/sample) of the active and predicted solar radiation (W/m^2), **b** monthly (1 h/sample) power estimation (W) of the PV output provided the mean power estimation, **c** daily estimation for 24 h of the solar radiation (W/m^2), **d** daily for 24 h of power estimation (W) of the PV output provided the mean power estimation, **e** monthly estimation (12 samples/h) of wind speed (m/s), **f** monthly (12 samples/h) power estimation (W) of the wind turbine provided the mean power estimation, **g** daily estimation (12 samples/h) of wind speed (m/s), **h** daily (12 samples/h) power estimation (W) of the wind turbine provided the mean power estimation

$$\begin{aligned}
 \rho_{FC \rightarrow BAT}^{SOC(t)} &= \left[SOC(t) < Lo_{FC \rightarrow BAT}^{SOC(t)} \right] \\
 &\quad \vee \left[\left[Str_{FC \rightarrow BAT}^{SOC(t)} < SOC(t) < Stp_{FC \rightarrow BAT}^{SOC(t)} \right] \right. \\
 &\quad \left. \wedge [\varepsilon_{FC \rightarrow BAT}(t^-) = 1] \right]
 \end{aligned}
 \tag{9}$$

where $Stp_{FC \rightarrow BAT}^{SOC(t)}$ is the upper limit of the hysteresis zone, and t^- is the previous observation instant. Using this approach, it is possible to systematically represent any PMS for a microgrid.

4 Combination of the flexible PMS representation and weather forecast

In order to test the efficiency of the proposed model, an already implemented system HYRES was taken under consideration. As a result, the meteorological measurements used in this study are real data collected from the location of Olvio near the city of Xanthi in Greece, where



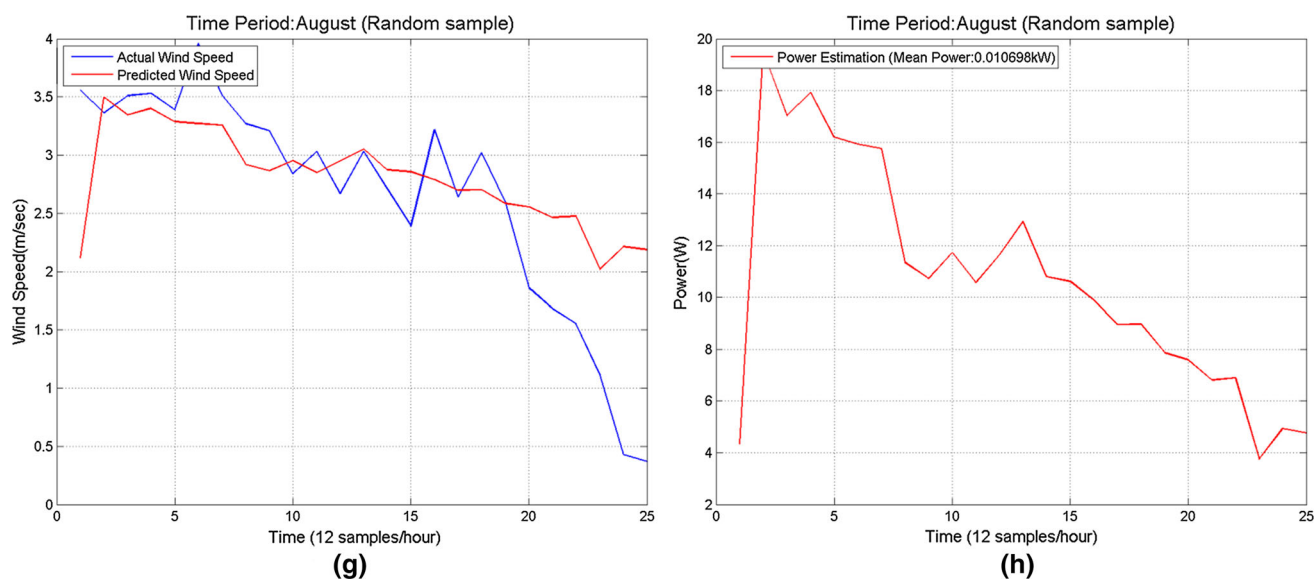


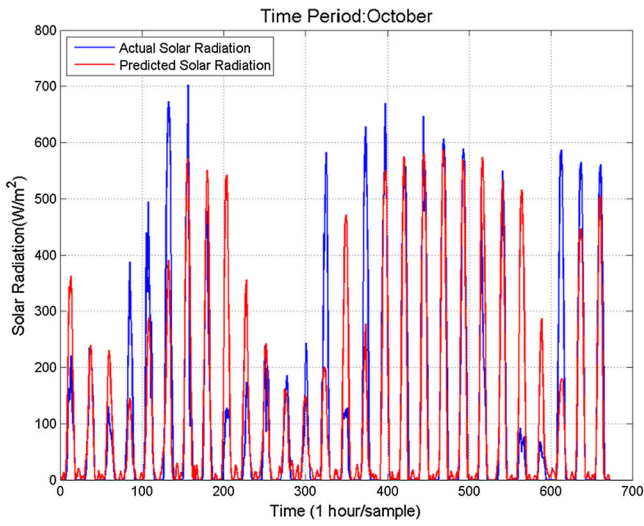
Fig. 11 continued

the main compound of Systems Sunlight S.A. is located [29]. In Fig. 15, there is a block diagram describing the general architecture and components of the available HYRES system. At the first level, there is a 8.5 kW photovoltaic array (PV Array), a 3 kW wind generator, a polymer electrolyte membrane (PEM) fuel cell, and a 48 volt, 3000 Ah battery array. These subsystems are all directly connected to the DC bus and indirectly to the AC bus through a DC/AC converter. In addition, there is a diesel generator and a PEM Electrolyzer directly connected to the AC bus. The specific system combines different renewable technologies, has some storing capabilities and also includes a conventional energy generator as a backup unit. The current HYRES system supplies a part of Systems Sunlight facilities with electricity, without the interference of any other power plant.

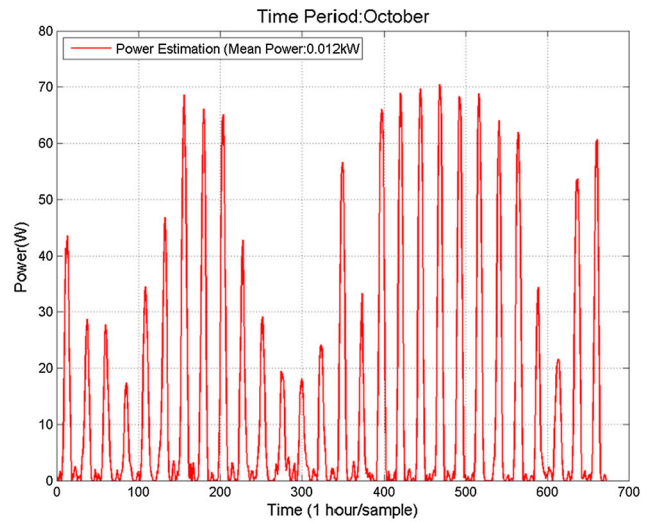
In the following, we use the solar radiation and wind speed prediction RNN presented in Sect. 2, and we combine it with the PMS representation of Sect. 3. As it has been said, when the second PMS is utilized, the FC is not allowed to operate during the summer months even in the case where the SOC is low (during the night) and there is available hydrogen. The main argument for that is that there will be intense solar irradiation after a few hours and hence energy will be produced that not only will charge the battery but most likely it will also produce hydrogen that can be stored in the FT. While in most cases this approach will reduce the usage of the FC without over depleting the battery or force the activation of the DSL, in some cases it is possible to cause many problems. This is happening if during the next 24 h, there is a rather low (comparing to the other days) solar irradiation, and hence, the DSL is forced

to be activated. This issue becomes more serious in case that there are multiple successive days with low solar irradiation. This problem can be overcome if using the weather forecast method explained before. This knowledge can be used to enable the selective usage of the FC (i.e., operate under the first PMS) during these periods of time. As a case study in Fig. 16a, we see the power produced by the PVs in the system for 4 days during August. We see that in August 3, the maximum power is above 12 kW while in August 5 less than 8 kW. More specifically, the total energy produced during these 4 days is approximately 92kWh, 41kWh, 48kWh, and 67kWh, respectively. In this case study, the load was fixed at 3 kW, and hence, each day has a requirement of 72 kWh. The FC was operated at 2kW and the DSL at 3 kW (in order to protect the battery).

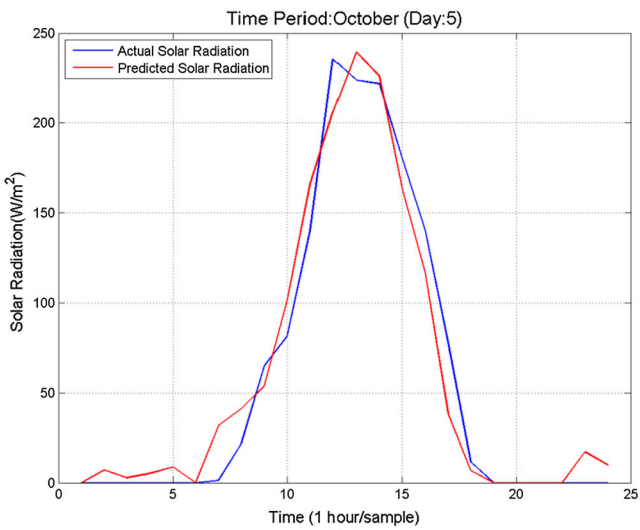
In Fig. 16b, we see the charge state of the battery when the two PMS were used. In the first case (solid trace), the PMS did not allow the activation of the FC, and while this did not cause any problems during day 1 (August 3) at the end of the second day, the SOC dropped below 0.2 and the DSL was activated. In the second case where the weather



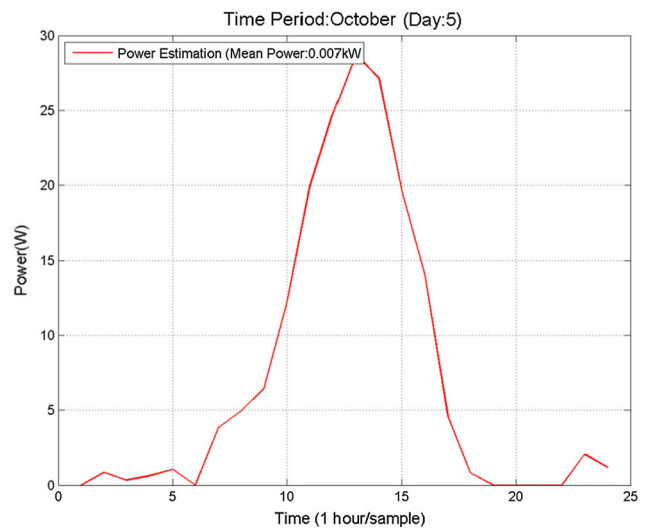
(a)



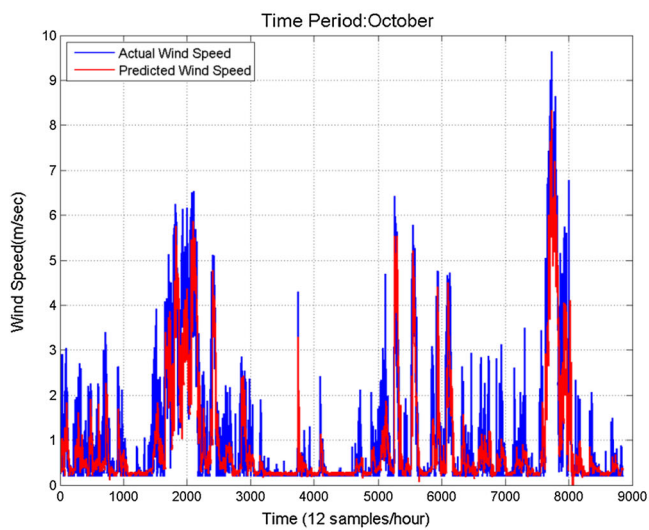
(b)



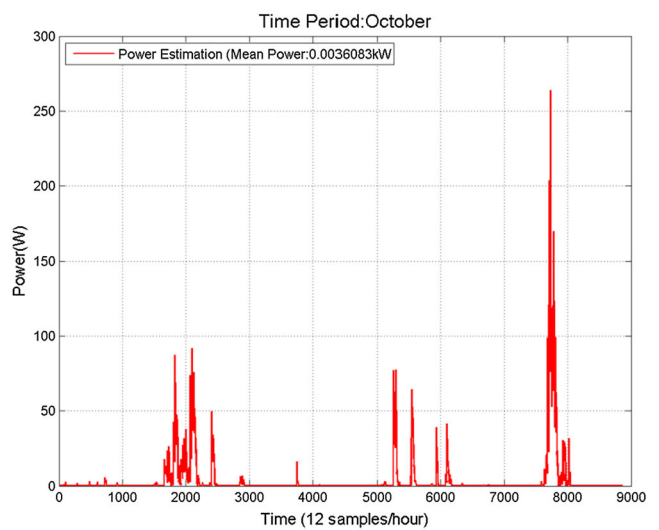
(c)



(d)



(e)



(f)

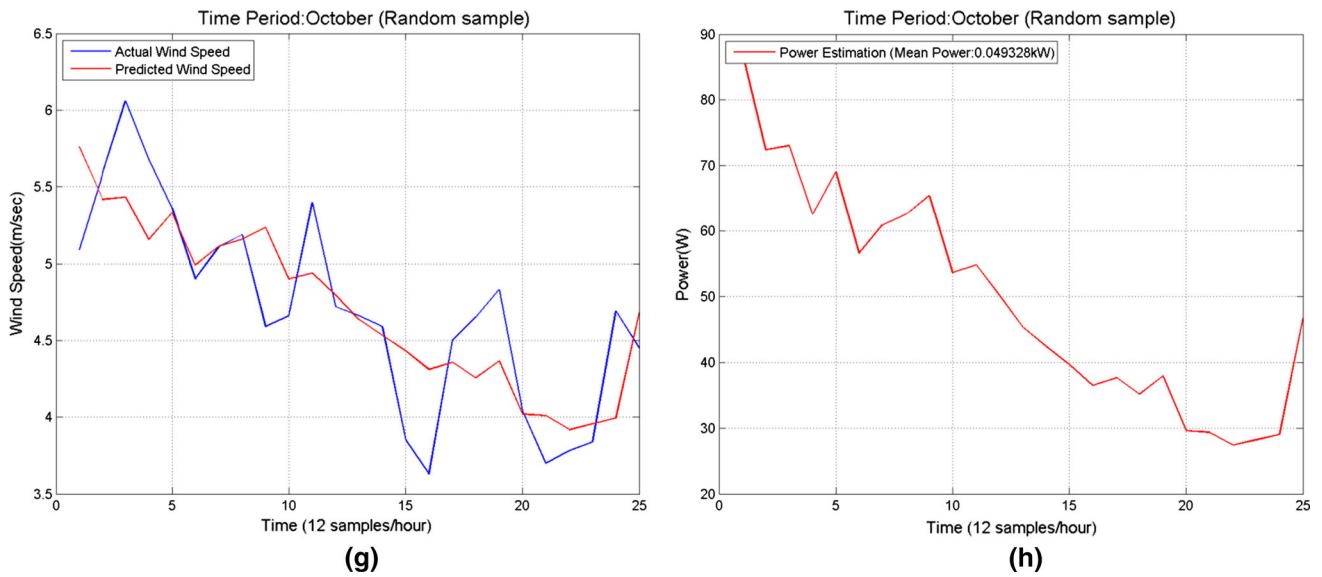


Fig. 12 continued

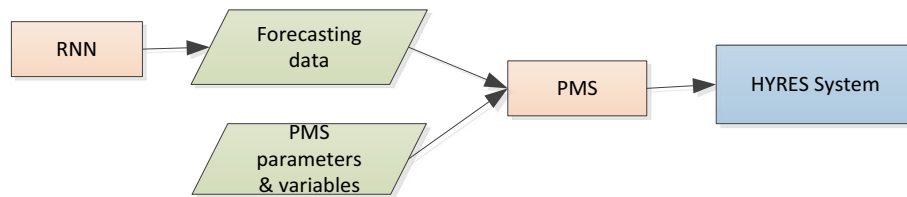


Fig. 13 An overview of the proposed approach presenting the interaction between RNN, PMS, and HYRES

Fig. 14 Network diagram of the stand-alone microgrid [28]. The parameters of the system are given in Table 1

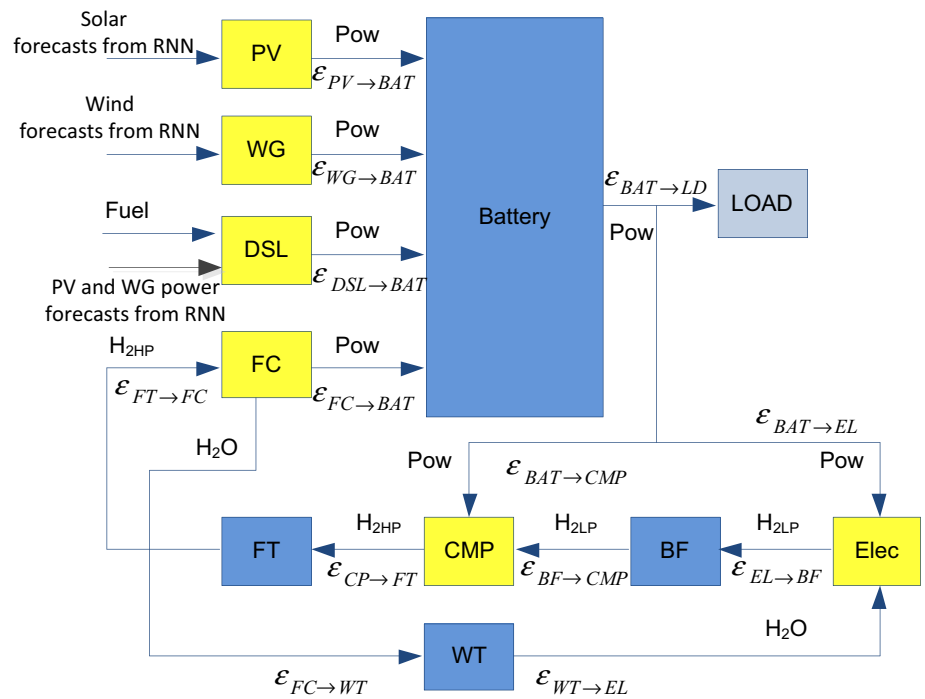


Table 1 Microgrid parameters

PV (66.64 W rated power)	217
WG (1 kW rated power)	3
BAT	3000 Ah
EL	5000 W
BF	8 bar, $\sim 1 \text{ m}^3$
FT	20 bar, $\sim 220 \text{ m}^3$

Table 2 Terminology of the stand-alone microgrid presented in Fig. 14

Pow	Power
WG	Wind generators
PV	Photovoltaic panels
DSL	Diesel generator
FC	Fuel cell
FT	Final hydrogen storage tank (high pressure)
BF	Buffer hydrogen storage tank (low pressure)
CMP	Compressor
Elec or EL	Electrolyzer
WT	Water tank
H _{2HP}	Hydrogen at high pressure
H _{2LP}	Hydrogen at low pressure
H _{2O}	Water
LD	Load
BAT	Battery
$\varepsilon_{l \rightarrow n}(t)$	Binary variable determining the connection between two subsystems l and n (e.g., electrolyzer with buffer tank)

prediction is used, based on the fact that the solar irradiation in August 4–5 may not be high enough, the system changed the PMS and activated the FC. As a result, the SOC was prevented from dropping below 0.26.

After careful consideration, an appropriate PMS was selected based on overall system-related criteria considering the provided input from the RNN for the solar radiation and wind speed. The decision variables during the operation of a stand-alone power system are the power level from the PV system and the wind generators and the voltage of the accumulators. The most effective PMS was the one that the lead-acid accumulator along with the RES can support the electrolyzer operation in cases of excess of power [30]. Below a minimum limit for the SOC of the accumulator, the fuel cell will meet the system's energy demands in cases of shortage of energy. The simulation time period is for 1 month, and the solar radiation and wind speed data have been calculated on a 10-min average value. The power from the PV array and the wind generators due to the solar radiation profile in a particular day is presented in Figs. 17 and 18, respectively.

Part of the power produced was to be supplied to the load unit with a constant power of 1 kW. Figure 18

presents the net amount of power, as the difference between the power produced from the RES and the power needed for the load. The negative values indicate that current from the accumulator had to be drawn in order to cover the need while positive values implied the excess power that could be stored to the accumulator or supplied to the electrolyzer. A very important parameter that needed to be studied in detail is the state of charge (SOC) of the battery as it influences the operation of the accumulator, of the electrolyzer, and of the fuel cell. The upper SOC limit (it was assumed to be 91 %) sets the upper bound that charging needs to be stopped in order to start the operation of the electrolysis as long as there is sufficient surplus energy to be provided from the RES to the electrolyzer. The lower SOC limit (it was assumed to be 84 %) sets the lower bound that the fuel cell needs to provide the necessary power to charge the accumulator, as long as there is not surplus energy according to Fig. 19.

As was mentioned before, the electrolysis took place only when sufficient surplus energy existed from the RES, and the state of charge of the accumulator (Fig. 20) was above the upper limit. In that case, the hydrogen was produced from the excess energy. Figure 21 represents the hydrogen production and consumption during the various stages of operation in a one-month period. Hydrogen is stored in compressed cylinders for future use.

As it can be seen from Fig. 22, there was a constant rise in the pressure when the electrolyzer was operating, and after it was stopped, there was a constant pressure inside the bottle until it started again. When the fuel cell was working, hydrogen was removed from the bottle and the pressure decreased while this operation took place. It is highlighted that the pressure at the cylinders did not fall down a lower limit and it did not overcome the upper limit. Table 3 shows the total operation time for each subsystem. As can be seen, there was a period of a total time of 8 h during the month where the amount of energy was beyond the one that the electrolyzer could use. Therefore, this extra amount can be used elsewhere to cover other needs. Furthermore, the hydrogen stored at the end of the month was approximately 17.2 m^3 . This amount of hydrogen provides autonomy of the fuel cell of 16.5 h.

In this study, the simulated results for the various subsystems of the stand-alone power system during a one-month period were presented. The application scenarios that took place revealed that there was sufficient amount of stored hydrogen at the end of month. The fuel cell autonomy for the stored hydrogen was calculated at approximately 16.5 h, which is equivalent to 33 kWh/month. Moreover, there were short periods during the month when there was surplus energy. Part of it was provided to the electrolyzer and the rest (if any)

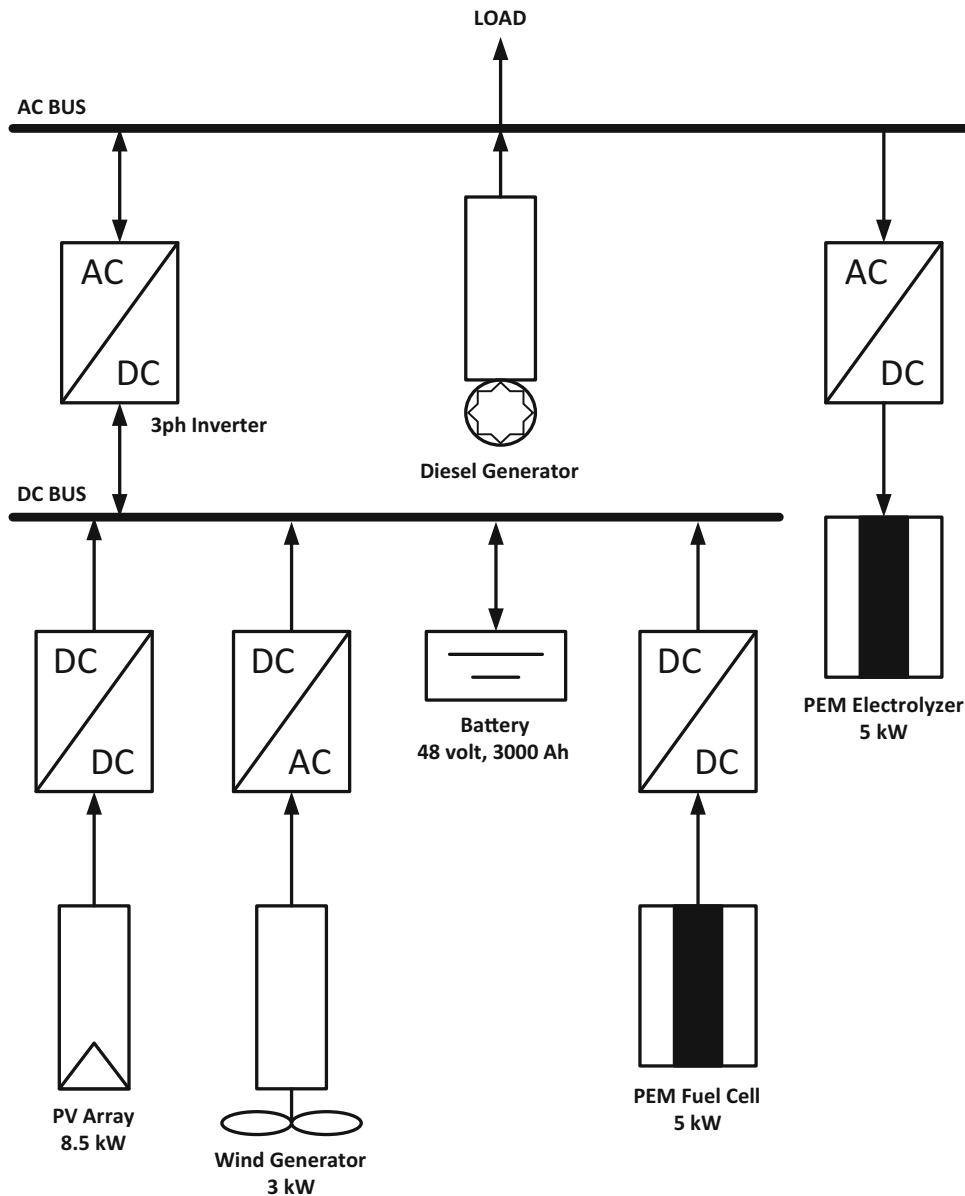


Fig. 15 Schematic of the available hybrid renewable energy systems (HYRES) that is installed in the location of Olvio, Xanthi, Greece, and is used for the needs of the current research

could be used to cover secondary electrical needs of the system. The hydrogen compressor is more likely to be provided with the extra energy.

One important outcome of the study was that several charges and discharges took place during the month. This situation can be hazardous for the accumulator as it reduces its lifetime. The total cycles during the simulation were calculated at 42 which is considered a bit high for an accumulator of this type. Thus, in order to protect the accumulator and increase its lifetime, it is vital to introduce a backup diesel generator (DG) which would provide the system with the necessary power when there is lack of energy. Another critical comment is that the state of charge

of the accumulator though it is a parameter that provides important information about its condition cannot be measured easily online. Therefore, a more applicable parameter to be used is the voltage of the accumulator where there would also exist a lower and an upper limit to control the operation of the accumulator.

5 Conclusions and discussion

In this paper, a RNN for solar radiation and wind speed prediction is presented for the enhancement of the PMSs of HYRES. The presented RNN with NAR architecture can

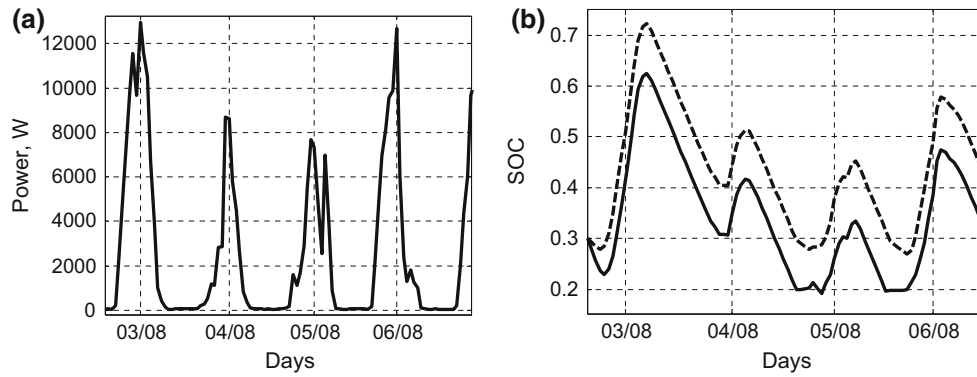


Fig. 16 a Power produced by the PVs. b SOC response under the two PMSs

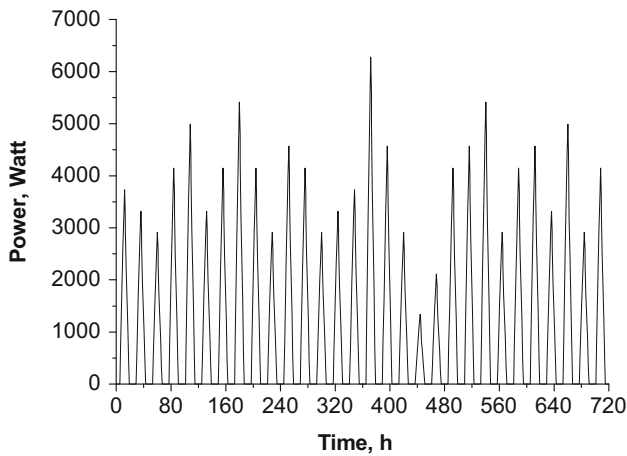


Fig. 17 Output power from photovoltaic system

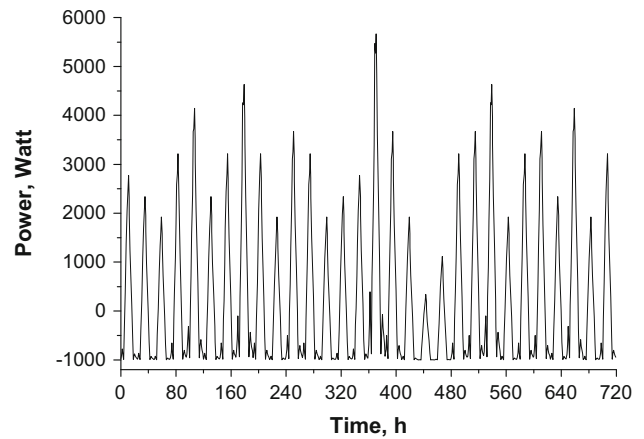


Fig. 19 Excess or lack of power during the month

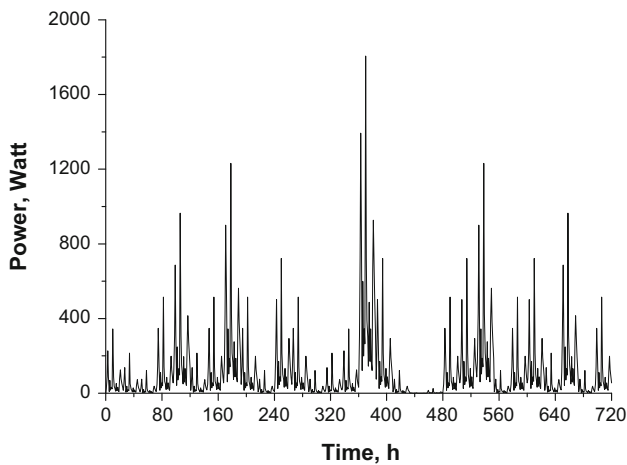


Fig. 18 Output power from the wind generators

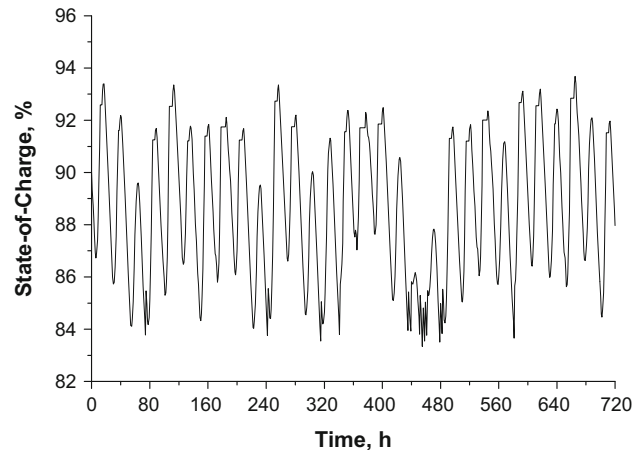


Fig. 20 State of charge of the accumulator during the month

offer both daily and hourly predictions concerning solar irradiation and wind speed forecasting.

The accuracy and the effectiveness of the proposed NAR model indicates that its estimations could be used for

the direct calculation of the exact amount of the future produced green energy by the target HYRES. Through a simple mathematical model, the produced forecasting could be imported to calibrate the current HYRES mathematical model for the calculation of the future production.

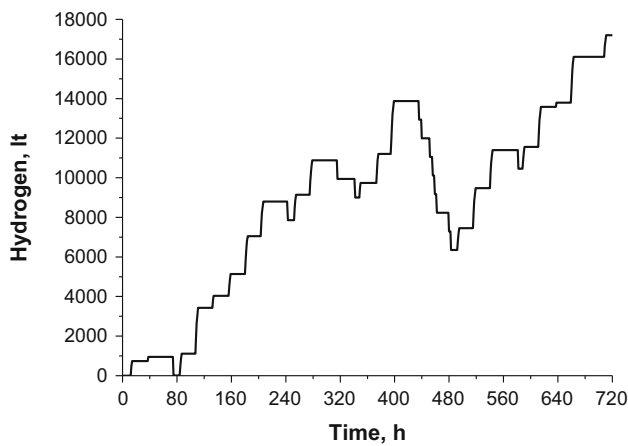


Fig. 21 Hydrogen production and consumption during the month

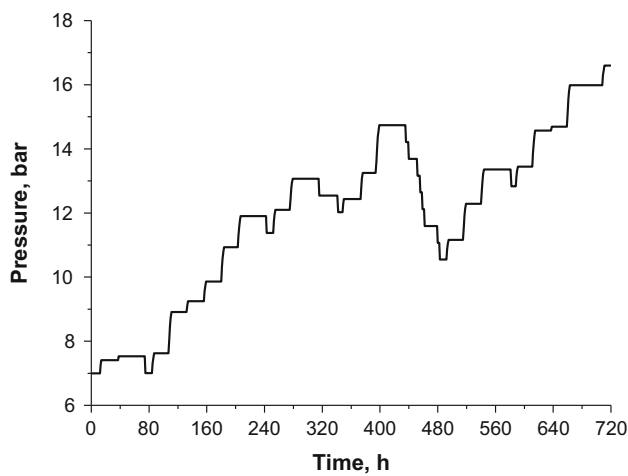


Fig. 22 Bottle pressure during the inlet and outlet of hydrogen

Finally, solar radiation and wind speed data were demonstrated so far. Of course, each of the examined case has its own requirements, but the average problem complexity still remains relatively the same.

The goal of the current research was to design and propose an intelligent model for the estimation of the future meteorological parameter values that are considered critical for the efficiency of HYRES. The intense dependence that these systems have on parameters such as solar radiation, air temperature, and wind speed makes

their forecasting quite a significant task. This a priori knowledge of the future values can be utilized for the overall function and efficiency optimization of these systems through the creation and application of the corresponding strategies. The most recent and advanced analytical weather forecasting models offer far more superior accuracy than the proposed NAR model. However, they necessitate the use of series of expensive sensing devices and some great amount of information. This fact makes them quite difficult to adopt in cases when some fully automated function is required. On the contrary, the proposed NAR model has the capability of estimating the needed meteorological parameters, requiring only the data from a local sensing device. In addition, it can easily adapt to the current conditions by learning and assimilating the available past data and can thereafter function completely autonomously.

The presented NAR model was calibrated on real meteorological data collected from the location of Olvio, Xanthi, in Greece. The simulation results indicated that it is capable of assimilating the given information and delivering some satisfactory future estimation that can be used to safely calculate the available green energy. Moreover, it has some sufficient for the specific problem computational power, as it can deliver the final results in just a few seconds. All the above were the result of the proper network architecture and sizing.

Additionally, the presented model has been also applied to the aforementioned autonomous HYRES, where its estimations were been used by a central control unit in order to create in real time the proper power management strategies (PMSs) for the efficient subsystems utilization that can lead to the overall optimization. For doing so, a generic network model was also described for the representation of the hybrid power generation systems taking into consideration in this work. Subsequently, the RNN when combined with the presented network model of HYRES serves as a novel framework for a generic approach aiming to facilitate the derivation of various PMSs in a simple and flexible way. As a result, the proposed framework will make the specific HYRES suitable for use as a stand-alone remote energy plant. As a proof of concept, the results of the proposed NN model for solar radiation and wind speed forecasting when applied to an available HYRES system were also presented. It is clear

Table 3 Results of the integrated system

	Operation time (h)	Hydrogen (l)
Charge of the accumulator	248	–
Discharge of the accumulator	398	–
PEM electrolyzer	66	29,430
PEM fuel cell	13	12,220
Excess power for other electrical needs	8	–

that the proposed RNN after training with meteorological data of the understudy area in our case Olvio of Xanthi in Greece and applied to the proposed HYRES of Systems Sunlight S.A. finally manages to enhance and optimize its PMS based on the provided solar radiation and wind speed prediction.

Despite the satisfactory results, it should be once again remarked that the designed NAR model cannot compete with the advanced weather forecasting tools also because the described ANN model and its performance are not generic but many more described periodically. A quite obvious disadvantage refers to its inability to estimate any sudden and extreme value changes that may be caused by random and unexpected phenomena. Nevertheless, it offers a cheap, sufficient, and easy-to-implement-and-apply forecasting solution suitable for use in HYRES.

Moreover, it should also be noted that most of the available studies in the literature [6, 8, 11] that adopt the NN paradigm for the estimation of meteorological parameters do not produce an actual future estimation. They usually make use of a series of different variables in order to statically estimate the target parameter for the given time and thus produce an artificial time series that refer to the past. This resembles more to the function approximation approach than the forecasting one. However, when dealing with some actual temporal estimations, the complexity of the problem becomes much greater. Furthermore, the lack of additional input data makes the specific task even more demanding.

In addition, another characteristic of the proposed NAR model that should be taken under consideration refers to its ability to learn and adapt to the data of the target system. Even when there is no available a sufficient amount of data in order to fully describe the local system behavior, the designed NAR can adapt over time to the current conditions. Whenever a certain amount of information is gathered, the specific model can be retrained in order to assimilate the newly acquired datasets. This characteristic makes it suitable for use in remote areas, where previous weather data are usually scarce.

Finally, as a future work, some additional NN architectures could be studied. Furthermore, the adoption of hybrid artificial intelligence systems could be also considered. Additionally, another significant task is the hardware implementation of the designed NAR and its integration to the available HYRES. In specific, the hardware implementation of the proposer NAR in a Field Programmable Gate Array (FPGA) device could be proven advantageous in terms of computational power, computational speedup, and parallelization. Taking advantage of the inherent parallelism of the proposed hardware chip, the outcome will be beneficial for the computational performance of the designed NAR as also for the proposed power management strategy. Consequently, these implementations could be considered as

modules of a generic system that will work autonomously providing all the requested functionality in a real-time fashion. In view of the foregoing and as a preliminary approach and proof of concept, the proposed RNN has been already implemented with some relaxations in a 16-bit PIC microprocessor with prominent features that helps us to speed up the learning and forecasting process of the designed RNN. Nevertheless, more work is requested to proceed with the final hardware implementation, and this should be considered part of a future work.

Acknowledgments This work is co-financed by National Strategic Reference Framework (NSRF) 2007–2013 of Greece and the European Union Research Program “SYNERGASIA” (SUPERMICRO – 09ΣΥΝ-32-594).

References

1. Zakeri B, Syri S (2015) Electrical energy storage systems: a comparative life cycle cost analysis. *Renew Sustain Energy Rev* 42:569–596
2. Garcia P, Torreglosa JP, Fernandez LM, Jurado F (2013) Optimal energy management system for standalone wind turbine/photovoltaic/hydrogen/battery hybrid system with supervisory control based on fuzzy logic. *Int J Hydrogen Energy* 38:14146–14158
3. Zhang X, Chan SH, Ho HK, Tan S-C, Li M, Li G, Li J, Feng Z (2015) Towards a smart energy network: the roles of fuel/electrolysis cells and technological perspectives. *Int J Hydrogen Energy* 40:6866–6919
4. Deshmukha MK, Deshmukh SS (2008) Modeling of hybrid renewable energy systems. *Renew Sustain Energy Rev* 12(1):235–249
5. Alam S, Kaushik SC, Garg SN (2006) Computation of beam solar radiation at normal incidence using artificial neural network. *Renew Energy* 31(10):1483–1491
6. Mubiru J, Banda EJKB (2008) Estimation of monthly average daily global solar irradiation using artificial neural networks. *Sol Energy* 82(2):181–187
7. Rehman S, Mohandes M (2008) Artificial neural network estimation of global solar radiation using air temperature and relative humidity. *Energy Policy* 36(2):571–576
8. Ghanbarzadeh A, Noghrehabadi R, Assareh E, Behrang MA (2009) Solar radiation forecasting using meteorological data. In: 7th IEEE international conference on industrial informatics (INDIN 2009), UK
9. Benganem M, Mellit A (2010) Radial basis function network-based prediction of global solar radiation data: application for sizing of a stand-alone photovoltaic system at Al-Madinah, Saudi Arabia. *Energy* 35:3751–3762
10. Paoli C, Voyant C, Muselli M, Nivet ML (2010) Forecasting of preprocessed daily solar radiation time series using neural networks. *Sol Energy* 84(12):2146–2160
11. AbdulAzeez MA (2011) Artificial neural network estimation of global solar radiation using meteorological parameters in Gusau, Nigeria. *Arch Appl Sci Res* 3(2):586–595
12. Mellit A, Kalogirou SA, Hontoria L, Shaari S (2009) Artificial intelligence techniques for sizing photovoltaic systems: a review. *Renew Sustain Energy Rev* 13(2):406–419
13. Zeng Z, Yang H, Zhao R, Meng J (2013) Nonlinear characteristics of observed solar radiation data. *Sol Energy* 87:204–218
14. Zhang N, Behera, PK (2012) Solar radiation prediction based on recurrent neural networks trained by Levenberg–Marquardt

- backpropagation learning algorithm. In: Innovative smart grid technologies (ISGT), 2012 IEEE PES, pp 1–7
15. Cao Q, Ewing BT, Thompson MA (2012) Forecasting wind speed with recurrent neural networks. *Eur J Oper Res* 221(1):148–154
 16. Grossberg S (1988) Nonlinear neural networks: principles, mechanisms, and architectures. *Neural Netw* 1:17–61
 17. Chang F-J, Chen P-A, Lu Y-R, Huang E, Chang K-Y (2014) Real-time multi-step-ahead water level forecasting by recurrent neural networks for urban flood control. *J Hydrol* 517:836–846
 18. Anderson JA (1995) Introduction to neural networks. MIT Press, Cambridge
 19. Elman J (1990) Finding structure in time. *Cogn Sci* 14:179–211
 20. Pearlmutter BA (1995) Gradient calculations for dynamic recurrent neural networks: a survey. *IEEE Trans Neural Netw* 6(5):1212–1228
 21. Hwang SY, Basawa IV (1994) Large sample inference based on multiple observations from nonlinear autoregressive processes. *Stoch Process Appl* 49(1):127–140
 22. Kapetanios G (2006) Nonlinear autoregressive models and long memory. *Econ Lett* 91(3):360–368
 23. Taskaya-Temizel T, Casey M (2005) A comparative study of autoregressive neural network hybrids. *Neural Netw* 18(5–6):781–789
 24. Guo WW, Xue H (2014) Crop yield forecasting using artificial neural networks: a comparison between spatial and temporal models. *Math Prob Eng* 857865:7
 25. Kohonen T (1989) Self-organization and associative memory. Springer, Berlin
 26. Haykin S (1998) Neural networks: a comprehensive foundation. Prentice Hall, Englewood Cliffs
 27. Anderson JA, Rosenfield E (1989) Neurocomputing: foundations of research. MIT Press, Cambridge
 28. Giaouris D, Papadopoulos AI, Ziogou C, Ipsakis D, Voutetakis S, Papadopoulou S, Seferlis P, Stergiopoulos F, Elmasides C (2013) Performance investigation of a hybrid renewable power generation and storage system using systemic power management models. *Energy* 61:621–635
 29. Chatziagorakis P, Elmasides C, Sirakoulis GCh et al (2014) Application of neural networks solar radiation prediction for hybrid renewable energy systems. In: Mladenov V et al (eds) EANN 2014, CCIS, vol 459. Sofia, Bulgaria, pp 133–144
 30. Ipsakis D, Voutetakis S, Seferlis P, Stergiopoulos F, Elmasides C (2009) Power management strategies on a stand-alone power system using renewable energy sources and hydrogen storage. *Int J Hydrogen Energy* 34:7081–7095