

Article

Adaptive Neuro-Fuzzy Inference Systems as a Strategy for Predicting and Controlling the Energy Produced from Renewable Sources

Otilia Elena Dragomir, Florin Dragomir, Veronica Stefan * and Eugenia Minca

Received: 3 July 2015 ; Accepted: 9 November 2015 ; Published: 17 November 2015

Academic Editor: Stefan Gößling-Reisemann

Automation, Computer Science and Electrical Engineering Department, Valahia University of Târgoviște, 2 Carol I Bd., Targoviste 130024, Romania; drg_otilia@yahoo.com (O.E.D.); drg_florin@yahoo.com (F.D.); eugenia.minca@gmail.com (E.M.)

* Correspondence: veronica.stefan@ats.com.ro; Tel.: +40-722-312-890; Fax: +40-245-211-727

Abstract: The challenge for our paper consists in controlling the performance of the future state of a microgrid with energy produced from renewable energy sources. The added value of this proposal consists in identifying the most used criteria, related to each modeling step, able to lead us to an optimal neural network forecasting tool. In order to underline the effects of users' decision making on the forecasting performance, in the second part of the article, two Adaptive Neuro-Fuzzy Inference System (ANFIS) models are tested and evaluated. Several scenarios are built by changing: the prediction time horizon (Scenario 1) and the shape of membership functions (Scenario 2).

Keywords: forecasting; neural network; Adaptive Neuro-Fuzzy Inference Systems; renewable energy sources

1. Introduction

For renewable energy source management systems it is necessary to know how to optimally schedule or dispatch the power resources. Among the operations performed for this purpose the forecasting of future supplier or consumer behaviour may be included. The accuracy of short time horizon predictions has a direct economic impact because it enables the energy provider to plan in advance their sources and to take proactive control actions for the power systems. The quality of the forecasting has a significant importance on the economic parameters and the security of the system, for the optimal management of the resources and increasing the business profitability.

The current energy policy of the European Union (EU) considers the security of supply, competitiveness and sustainability as central goals. Studies conducted by us have highlighted the need for an energy management system at the micro to macro level, as existing control strategies are not being always successfully applied.

The problems in this area are numerous. The main interest directions in the field, identified by reviewing the scientific literature are how to increase conversion efficiency [1,2], how to identify new opportunities, complementary to the limitations of existing solutions [3], how to balance the insufficient number of providers to reported market needs [4], how to supply energy to an increasing number of consumers [5], how to provide and promote sustainable buildings and housing projects [6], how to identify new renewable energy sources in order to reduce risks, social and economic costs [7], how to understand the effects associated with the use of RES and the increasing complexity of systems [8], how to identify measures and solutions for ensuring the sustainability of renewable resources [9], how to adjust the impacts of renewable energy for the final beneficiaries, in terms of

cost/benefit ratio [10,11], and how to highlight the importance of being efficiently informed in order to make better decisions about energy options [12].

In this area two challenges have to be addressed: the first one consists of the monitoring, diagnosis and forecasting functioning of microgrids integrating renewable energy sources, in on-grid or off-grid mode and the second to ensure the optimization of these functioning ones, with an integrated, proactive management system based on a multi-objective decision-making scheme.

To support these interests, we consider that integration of the predictive ability is a very important function in the command and control of a power system, especially at the microgrid level, heavily dependent on the consumer and with reduced storage availability. Thus, we access information, cloud stored, for training and testing the prediction tools, based on artificial intelligence techniques, in order create an open and adaptable tool for optimizing microgrid energy management, functioning from an economical point of view, active control of distributed generation, controlled consumption, and loading the storage equipment.

Worldwide there are only a few tools which are able to perform dynamic assessments of electrical networks; none of them take into account the characteristics and the specificities of microgrids integrating RES. The efficiency of using Neural Networks (NN) in the area of energy forecasting was demonstrated in our previous works [13–15].

The challenge for our paper consists in controlling the performance of the future state of a microgrid with energy produced from renewable energy sources. The added value of this proposal consists in identifying the most used criteria, related to each modeling step, able to lead us to an optimal neural network forecasting tool. In order to underline the effects of the users' decision making on the forecasting performance, in the second part of the article, two Adaptive Neuro-Fuzzy Inference System (ANFIS) models are tested and evaluated. Several scenarios are built by changing the prediction time horizon (Scenario 1) and the shape of the membership functions (Scenario 2).

2. Exploration and Assessment of Criteria Used for Choosing a Forecasting Tool

In our previous work [13] the forecasting was studied from the industrial and scientific point of view. This was aimed at proposing a novel classification of forecasting tools and guidelines for choosing a suitable prediction technique, that is, a set of criteria to “assess” the applicability of forecasting tools. In this respect, the steps in the forecasting tool modelling, based on NN, are: data pre-processing, determining the forecasting tool architecture, parametrization and implementation.

The added value of this paper consists in identifying the most used criteria for each modeling step, allowing us to implement an optimal forecasting NN architecture. The approach was difficult because of the high numbers of variables and constraints that condition parameterization of a NN: number of hidden layers, number of neurons from the NNs input, output and hidden layers, testing/validation data sets dimension, the learning algorithm to choose, and the method used to measure the forecasting made by the tool.

2.1. Data Preprocessing

The input selection methods we selected are experimental methods. They are supposed to find the most important input variables, among a large number of them [16]. Some authors [17] assume that the input information can be classified into groups (hierarchical structure). There is no general automatic approach to realize such a grouping, beside from heuristics based on the fusion of physical sensors. Both input selection and hierarchical structure presume that inputs are independent and assign no priority or importance to the selected input variables.

When the analyzed process is complex it is recommended to normalize the input and output real values into the interval between the max and min of the transformation function (usually in $[0, 1]$ or $[-1, 1]$ intervals). The most popular methods are the two following:

$$SV = \frac{(0.9 - 0.1)}{MAX_{VAL} - MIN_{VAL}} (OV - MIN_{VAL}) \quad (1)$$

or:

$$SV = TF_{min} + \frac{(TF_{max} - TF_{min})}{(MAX_{VAL} - MIN_{VAL})} (OV - MIN_{VAL}) \quad (2)$$

where:

- SV : Scaled Value;
- MAX_{VAL} : maximum value of data;
- MIN_{VAL} : minimum value of data;
- TF_{max} : maximum of transformation;
- TF_{min} : minimum of transformation;
- OV : Original Value.

2.2. Determining NN Architecture

There are many techniques for determining the forecasting tool architecture. In this section we will cover some of the general “rules of thumb” that can be used. In nearly all cases some additional experimentation will be required to determine the optimal structure.

Every input neuron from the input layer of the NN should represent some independent variable that has an influence on the output. If a pattern is presented to the input layer of a NN then the output layer will generate another pattern. The output layer of the NN presents a pattern to the external environment [18]. Whatever a pattern is represented by the output layer this can be directly traced back to the input layer. The number of output neurons is related to the type of work that the NN has to perform. If it is to be used to classify items into groups, then it is often preferable to have one output neuron for each group that the item is to be assigned to. If the NN is to perform noise reduction on a signal then it is likely that the number of input neurons will match the number of output neurons.

One common mistake made by users is to add a large number of variables and neurons, without taking into account the number of parameters to be estimated [19]. The hidden layers are those that don't interact directly with the external environment. The number of hidden layers and the number of neurons in the hidden layers must be considered because these have an important influence on the final output. For many practical problems there's no reason to use more than one hidden layer. The one hidden layer NN can approximate arbitrarily well any continuous mapping from one compact space to another. NNs without hidden layers are only capable of representing linear separable functions or decisions. NN with two hidden layers can represent functions with any kind of shape. Problems that require two hidden layers are rarely encountered and currently there is no theoretical reason to use neural networks with any more than two hidden layers [18].

In [20] the authors proposed as three rules starting points to consider: the number of hidden neurons should be in the range between the size of the input layer and the size of the output layer; the number of hidden neurons should be $2/3$ of the input layer size, plus the size of the output layer; the number of hidden neurons should be less than twice the input layer size.

One additional method that can be used to reduce the number of hidden neurons is called pruning. It involves evaluating the weighted connections between the layers and if the network contains connections with weights equal to zero they can be removed.

2.3. Parametrization

The test samples must be appropriately selected. Since NNs are “data-driven” methods, they typically require large samples in testing. The input selection reduces the testing data dimension. The NN are tested with small subsets that included data from only a few past days selected through statistical measures of similarity. This results in homogeneous samples, but also very small [19].

Usually, the dimension of test data set must be five times greater than the number of updated parameters. The testing process is usually finished as soon as the testing error is reduced to a specified tolerance level (e.g., 10^{-5}). The adequate rate between the number of testing samples and the number of weights in the network has not yet been clearly defined [21,22].

The testing algorithm has to be also established for this task. The most common in use is the back propagation algorithm. The goal of back propagation testing is to converge to a near-optimal solution based on the total squared error calculated. Another algorithm, less used in energy applications, is K-nearest-neighbor. It is a learning algorithm that is simpler than the back propagation algorithm because there is no model to test on the data series. Instead, the data series is searched for situations similar to the current one each time a forecast needs to be made [23,24]. In most of the reviewed papers, testing was stopped after a fixed number of iterations or after the error decreased below some specified tolerance.

2.4. Implementation and Performance Testing

The next stage in modelling is the implementation of the NN, which means the estimation of its parameters. The guidelines proposed in [25] for evaluating effectiveness of implementation were based on the next question: was the NN properly tested so that its performance was the best it could possibly achieve?

The choice of error measures to help comparing forecasting methods has been much discussed, as a consequence of the many competitions that were started in the 1980's [26]. de Gooijer and Hyndman [27] summarized in Table 1 the most used performance indicators.

Table 1. The most used performance indicators.

Commonly Used Forecast Accuracy Measures		
MSE	Mean Squared Error	= $\text{mean}(e_t^2)$
RMSE	Root Mean Squared Error	= \sqrt{MSE}
MAE	Mean Absolute Error	= $\text{mean}(e_t)$
MdAE	Median Absolute Error	= $\text{median}(e_t)$
MAPE	Mean Absolute Percentage Error	= $\text{mean}(p_t)$
MdAPE	Median Absolute Percentage Error	= $\text{median}(p_t)$
MRAE	Mean Relative Absolute Error	= $\text{mean}(r_t)$
MdRAE	Median Relative Absolute Error	= $\text{median}(r_t)$
GMRAE	Geometric Mean Relative Absolute Error	= $\text{gmean}(r_t)$
RelMAE	Relative Mean Absolute Error	= MAE/MAE_b
RelRMSE	Relative Root Mean Squared Error	= $RMSE/RMSE_b$
LMR	Log Mean squared error Ratio	= $\log(\text{RelMSE})$
PB	Percentage better	= $100 * \text{mean}(I\{ r_t < 1\})$
PB(MAE)	Percentage better (MAE)	= $100 * \text{mean}(I\{MAE < MAE_b\})$
PB(MSE)	Percentage better (MSE)	= $100 * \text{mean}(I\{MSE < MSE_b\})$

Error (e) is the difference between real and estimated data; The absolute percentage error is p ; The relative absolute error is r ; Here $I\{u\}=1$ if u is true and 0 otherwise.

Most authors agree on the use of the following as performance indicators: mean squared error (MSE), root mean squared error (RMSE) and mean absolute error (MAE) for training phase

evaluation [27]. Only a few have reported the use of mean absolute percent errors (MAPE) or the standard deviation of the errors [28,29] (Table 1).

However, recent studies and the experience of system operators indicates that the loss function in the load forecasting problem is clearly nonlinear, and that large errors may have disastrous consequences for a utility [30,31]. Because of this, measures based on squared error are sometimes suggested, as they penalize large errors (RMSE was suggested in [32], MAPE in [33]).

Most researchers test their models by examining their errors in samples other than the one used for parameter estimation because the goodness-of-fit statistics are not enough to predict the actual performance of a method [21].

Also, it is generally recognized that error measures should be easy to understand and closely related to the needs of the decision-makers. Some papers have reported that the utilities would rather evaluate forecasting systems by the Absolute Errors Produced, and this suggests that MAE could be useful [19]. The shape of the distribution should be suggested. A few papers included graphs of the cumulative distribution of the errors [34]. Others suggested this distribution by reporting the percentage of errors above some critical values, percentiles [35], the maximum errors [36,37]. In any case, no single error measure could possibly be enough to summarize the efficiency of the forecast.

3. Forecasting Tool Simulation and Performance Evaluation

Considering the bibliographical research presented above, in the next sections we underline the consequences to the forecasting performances of the choices made during the tool modelling.

3.1. Adaptive Neuro-Fuzzy Inference System

Despite all their publicized successes, NN-based load forecasting models have been designed relying on time-consuming, empirical, and suboptimal procedures. The potential for applying NN to short term forecasting depends strongly on the extraction of appropriate input variables. This requirement has often been neglected, and many proposals for building the NN input space still use (linear) correlation analysis. Input selection procedures with the capacity to extract high order statistical information from the input-output data must be employed to fully exploit the NN mapping capability.

The advantages and the drawbacks of NN led us to choose a nonlinear network (neuro-fuzzy systems-NFs) as reference tool for our short term energy forecasting approach. Thus the case study is performed using an ANFIS network. Neuro-fuzzy systems are a combination of NN and fuzzy sets and represent a powerful tool to model system behavior [38,39]. The NN is used to define the clustering in the solution space, which results in the creation of fuzzy sets [40–42].

A particular neuro-fuzzy system architecture is represented by the Adaptive Neuro-Fuzzy Inference System (ANFIS) [43]. ANFIS is a Sugeno-type fuzzy inference system in which the parameters associated with specific membership functions are computed using either a backpropagation gradient descent algorithm alone or in combination with a least squares method. It has been widely applied to random data sequences with highly irregular dynamics [44], e.g., for forecasting non-periodic short-term stock prices [45]. Figure 1 illustrates the ANFIS architecture for two input parameters, where the nodes of the same layer have similar functions, as described next.

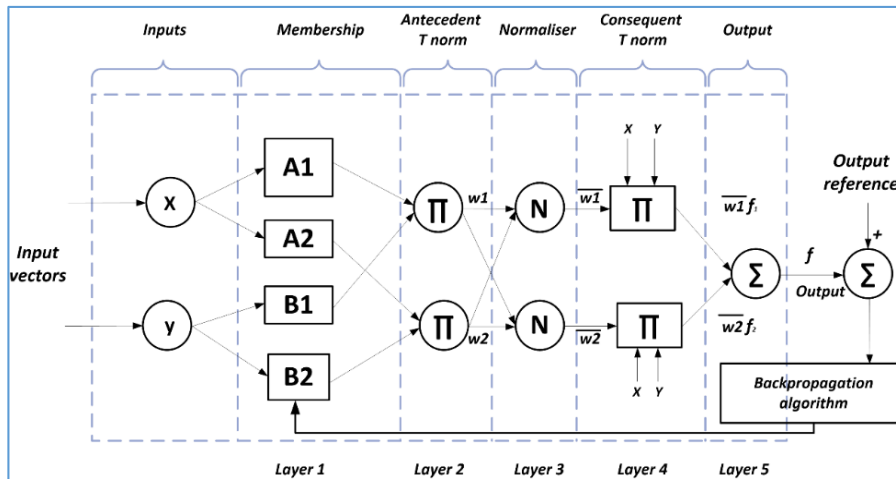


Figure 1. ANFIS architecture.

Here we denote the output of the i th node in layer j as $O_{i,j}(i)$:

Layer 1. Every node i in this layer is an adaptive node with a node function:

$$O_{1,i} = \mu_{A_i}(x), \text{ for } i = 1, 2 \tag{3}$$

or:

$$O_{1,i} = \mu_{B_{i-2}}(y), \text{ for } i = 3, 4 \tag{4}$$

where x (or y) is the input to node i and A_i (or B_{i-2}) is a linguistic label (such as “small” or “large”) associated with this node. In other words, $O_{1,i}$ is the membership grade of a fuzzy set A ($=A_1, A_2, B_1$ or B_2) and it specifies the degree to which the given input x (or y) satisfies the quantifier A .

The membership function for A can be any appropriate parameterized membership function introduced in here, such as the generalized bell function:

$$\mu_A(x) = \frac{1}{1 + \left| \frac{x-c_i}{a_i} \right|^{2b}} \tag{5}$$

where $\{a_i, b_i, c_i\}$ is the parameter set. As the values of these parameters change, the bell-shaped function varies accordingly thus various forms of membership function for fuzzy set A . Parameters in this layer are referred to as premise parameters.

Layer 2. Every node in this layer is a fixed node labelled $O_{i,j}$, whose output is the product of all the incoming signals:

$$O_{2,i} = w_i = \mu_{A_i}(x) \times \mu_{B_i}(y), \text{ } i = 1, 2 \tag{6}$$

Each node output represents the firing strength of a rule. In general, any other T-norm operators that performs fuzzy and can be used as the node function in this layer.

Layer 3. Every node in this layer is a fixed node labelled N . The i th node calculates the ratio of the i th firing strength and the sum of all firing strengths:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \text{ } i = 1, 2 \tag{7}$$

For convenience, outputs of this layer are called *normalized firing strengths*.

Layer 4. Every node i in this layer is an adaptive node with a node function:

$$O_{4,i} = \bar{w}_i \times f_i = \bar{w}_i(p_i x + q_i y + r_i) \tag{8}$$

where \bar{w}_i is a normalized firing strength from layer 3 and $\{p_i, q_i, r_i\}$ is the parameter set of this node. Parameters in this layer are referred to as *consequent parameters*.

Layer 5. The single node in this layer is a fixed node labelled $O_{5,1}$ (overall output), which computes the overall output as the summation of all incoming signals:

$$O_{5,1} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{9}$$

Thus we have constructed an adaptive network that is functionally equivalent to a Sugeno fuzzy model.

The ANFIS learning algorithm. When the premised parameters are fixed, the overall output is a linear combination of the consequent parameters. In symbols, the output f can be written as:

$$O_{5,1} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} f = (\bar{w}_1 \times x) \times c_{11} + (\bar{w}_1 \times y) \times c_{12} + \bar{w}_1 \times c_{10} + (\bar{w}_2 \times x) \times c_{21} + (\bar{w}_2 \times y) \times c_{22} + \bar{w}_2 \times c_{20} \tag{10}$$

which is linear in the consequent parameters $c_{i,j}$ ($i = 1, 2; j = 0, 1, 2$). A hybrid algorithm adjusts the consequent parameters $c_{i,j}$ in a forward pass and the premise parameters $\{a_i, b_i, c_i\}$ in a backward pass [40,41].

In the forward pass the network inputs propagate forward until Layer 4, where the consequent parameters are identified by the least-squares method. In the backward pass, the error signals propagate backwards and the premise parameters are updated by gradient descent.

Because the updating rules for the premise and consequent parameters are decoupled in the hybrid learning rule, a computational speedup may be possible by using variants of the gradient method or other optimization techniques on the premise parameters.

The success of ANFIS is given by aspects like: the designated distributive inferences stored in the rule base, the effective learning algorithm for adapting the system’s parameters or by the own learning ability to fit irregular or non-periodic time series.

On the other hand, used alone in applications to non-periodic short-term forecasting, ANFIS predictions make large residual errors due to high residual variance, consequently degrading prediction accuracy [46,47]. It is very difficult for a non-expert to interpret the fuzzy rules generated by ANFIS because of the form of consequents (linear combination of inputs).

3.2. Simulation Conditions and Strategy

ANFIS is applied on a database obtain from an experimental photovoltaic amphitheater of minimum dimensions (0.4 kV/10 kW), located in the east-center region of Romania, more precisely in the city of Targoviste [14]. The used data base has 296 data points $\{(y(t), u(t), t = 1, \dots, 296)\}$.

For ANFIS the input selection method is based on the assumption that the ANFIS model with the smallest RMSE [34] after one epoch of training has a greater potential of achieving a lower RMSE when given more epochs of training.

This assumption is not absolutely true, but it is heuristically reasonable. ANFIS can usually generate satisfactory results right after the first epoch of training, that is, only after the first application of the least-squares method [16]. For example, in case of a problem with 10 entries, for the identification of the three most influential entries, should be built firstly $C_3^{10}=120$ ANFIS models.

Everyone should be “learned” once and a mix of inputs with the smallest error learning will be selected. The learning time of 120 models involves less computations that learning to 120 times of a simple ANFIS model. For 10 inputs the Sugeno inference system generates $2^{10}= 1024$ rules and $(10 + 1) \times 1024$ parameters.

The too costly computation time determined us to build two groups for the selection of variables: a first set for historic outputs $\{y(t - 1), y(t - 2), y(t - 3)\}$ and another set for $\{u(t) u(t - 1) u(t - 2) u(t - 3) u(t - 4) u(t - 5)\}$. Thus, only $6 \times 4 = 24$ ANFIS models are tested and analyzed.

The difference between the electricity produced and consumed from renewable energy sources is considered as output ($y(t)$) to be forecast by ANFIS and the exterior temperature as input ($u(t)$). The inputs and outputs values aren't normalised.

The estimation procedure is carried out as two ANFIS Models, presented in Figure 2, having the following features:

- Model 1 represents an ANFIS structure with two inputs $\{y(t - a), u(t - b)\}$
- Model 2 is an ANFIS structure with three inputs: $\{y(t - a), u(t - b), u(t - c)\}$, $a = 1, \dots, 4$; $b, c = 1, \dots, 6$

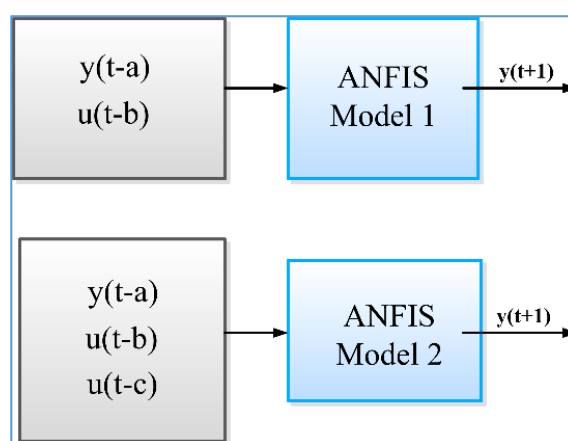


Figure 2. ANFIS models used for simulation.

Being a particular architecture of neuro-fuzzy network, the user can't change the number of ANFIS' layers but we can investigate the influence of the membership functions shapes over the output.

The learning algorithm tested is back propagation, the most used. In our future works we intend to also identify the influence of the learning method. Related to the dimension of training and testing data sets, we split our data considering the recommendation "test data set must be five times greater than the number of update parameters".

The investigation uses as performance metrics: error, MAPE, RMSE and MAE. In any case, these error measures cannot replace the need to calculate the error distribution, the shape of the probability density function (PDF) being the most suggestive for the evaluation of the forecasting model robustness and stability.

4. Simulations Results

The challenge for our paper consists in controlling the performance of the future state of the system. Thus, considering the conditions and the strategy described above, we have performed some simulations. In order to underline the effects of the users' decisions over the forecasting performance, several scenarios have been built by changing the prediction time horizon (Scenario 1) and the effect of membership function type and number changing (Scenario 2). The testing parameters in each case remain the same: number of epochs = 20, $ss = 0.01$, $ss_dec_rate = 0.5$, $ss_inc_rate = 1.5$, training/test data set = 145/145.

4.1. Scenario 1

This scenario identifies the performance of the ANFIS models, in relation with the time horizon. In this respect, each ANFIS model is trained and tested for a short-term horizon, with a time horizon equal with 20 steps ahead ($t + 20$), mid-term horizon ($t + 40$) and long- term horizon ($t + 100$).

The performance of ANFIS models is measured using the error metric (prediction error) and mean absolute percentage error (MAPE).

Figures 3 and 4 represent comparatively the performance of “Model 1” (Figure 3a,b) and of “Model 2” (Figure 4a,b) for a time horizon equal to 20 and 40. We can see that the ANFIS structure with three inputs, having additional information on the input, is better than “Model 1”, especially in the short time horizon. For the medium and long term cases, the errors became bigger and bigger and these affect the forecasting in terms of accuracy and confidence.

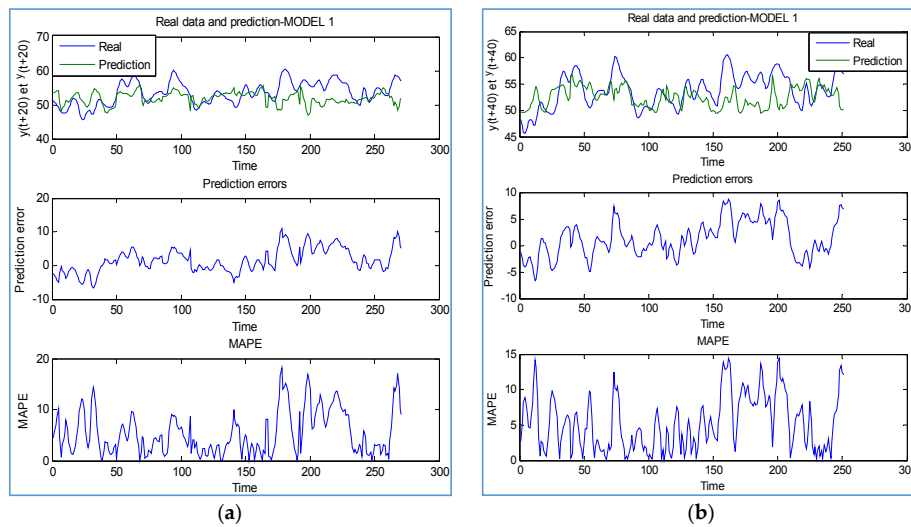


Figure 3. The performance of ANFIS for “Model 1”, for a time horizon equal with: (a) 20 steps ahead and (b) 40 steps ahead.

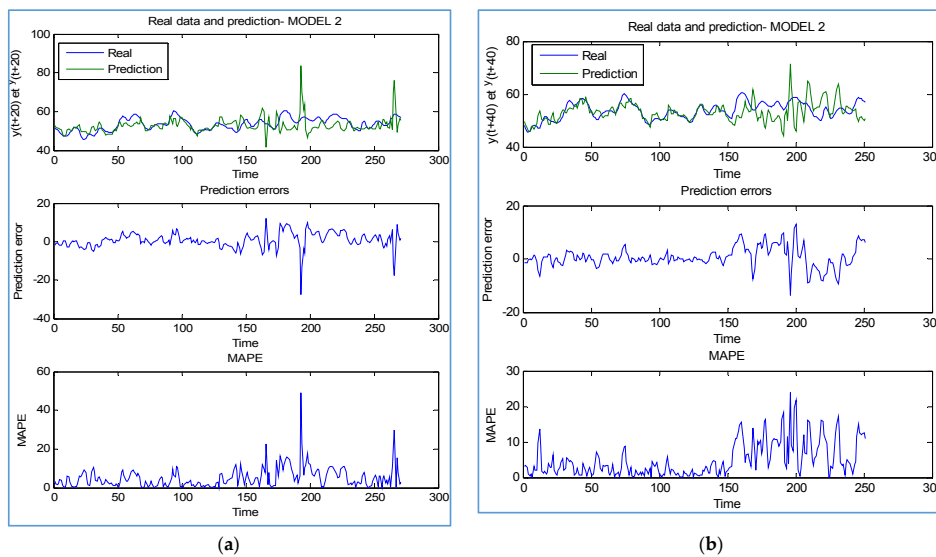


Figure 4. The performance of ANFIS for “Model 2”, for a time horizon equal with: (a) 20 steps ahead and (b) 40 steps ahead.

The performance metrics RMSE and MAE, measured on the test set, for “Model 1” and “Model 2” (Table 2, Figure 5a,b) show that these models don’t achieve a satisfactory compromise between short-term accuracy and long-term stability.

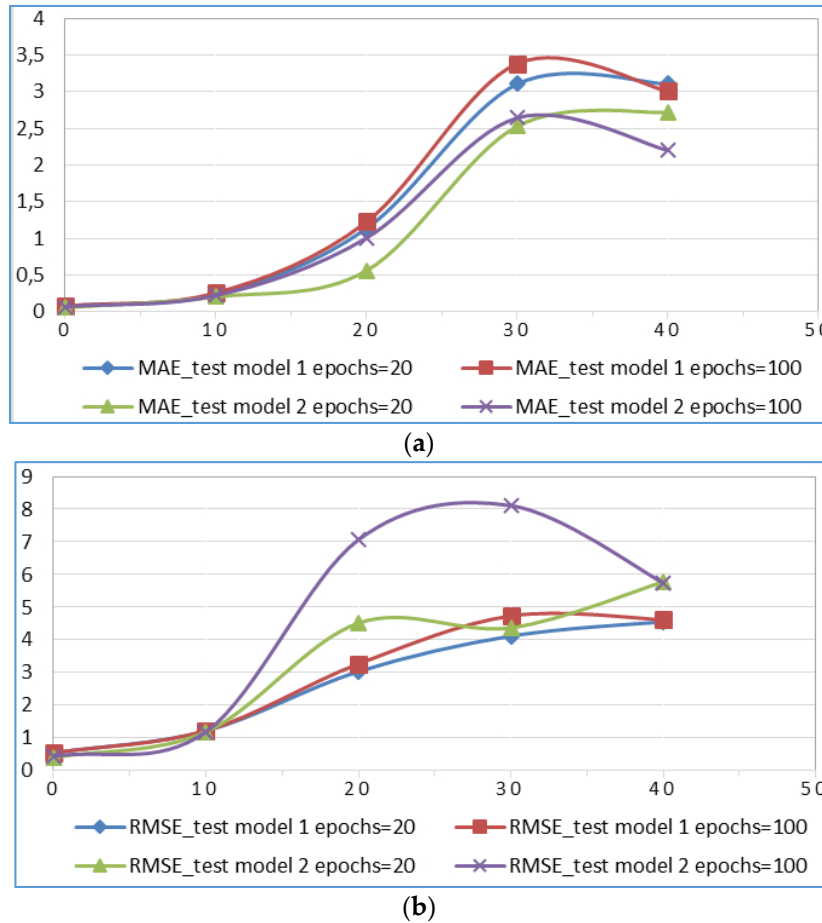


Figure 5. (a) MAE for Model 1 and Model 2 on different prediction of test data set; (b) RMSE for Model 1 and Model 2 on different prediction of test data set.

Table 2. RMSE and MAE values computed in case of Scenario 1.

t + pred	ANFIS	RMSE_Test		MAE_Test	
		Epochs = 20	Epochs = 100	Epochs = 20	Epochs = 100
1	model 1	0,5532	0,5514	0,0809	0,0816
1	model 2	0,4083	0,4432	0,0597	0,0708
4	model 1	1,2228	1,23	0,2384	0,2591
4	model 2	1,1955	1,1959	0,2042	0,2239
10	model 1	3,0388	3,2713	1,1357	1,226
10	model 2	4,5201	7,0661	0,5617	1,0066
20	model 1	4,1209	4,7385	3,1039	3,382
20	model 2	4,374	8,1093	2,5285	2,6426
40	model 1	4,5597	4,622	3,1039	3,0073
40	model 2	5,7804	5,7492	2,7198	2,2013

In any case, error measures are only intended as summaries of the error distribution. This distribution is usually expected to be normal white noise in a forecasting problem, but it will probably not be so in a complex problem like load forecasting. No single error measure could possibly be

enough to summarize it. The shape of the distribution should be suggested. Keeping the total error low, therefore, means keeping the model simple.

In this respect, we have computed the probability density function (PDF) for the selected models, for short time ($t + 20$), medium time ($t + 40$) and long time ($t + 100$) prediction horizon (Figure 6a,b).

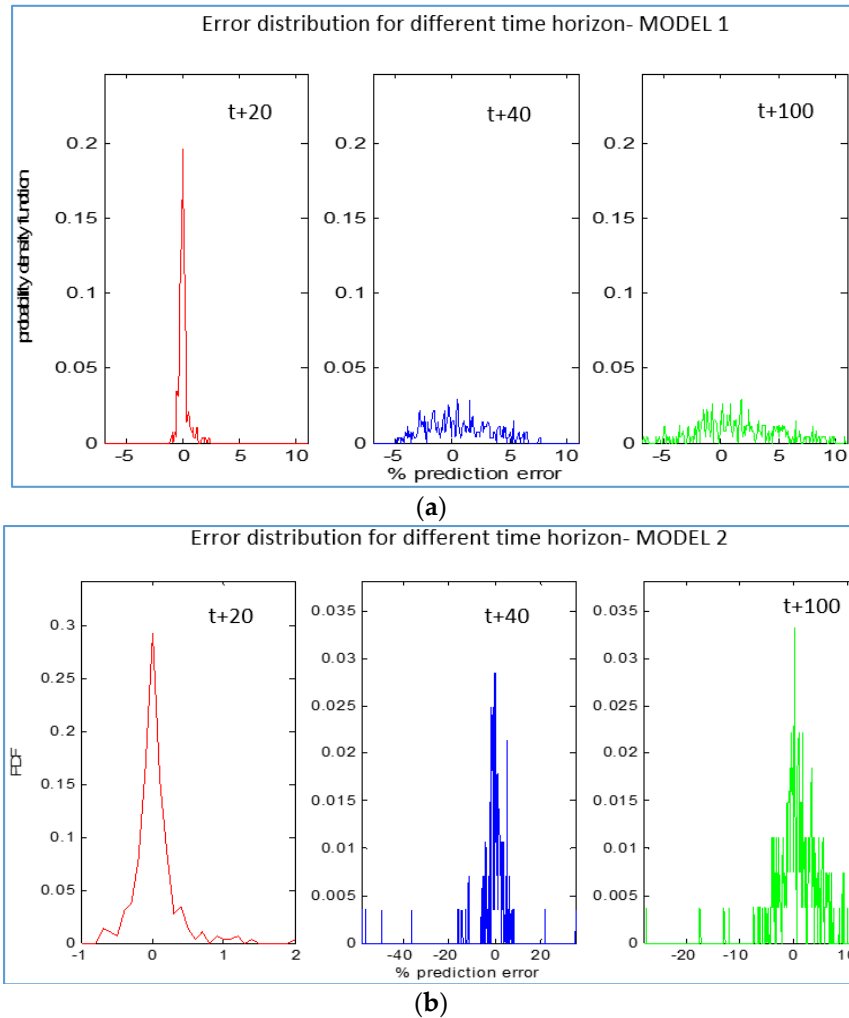


Figure 6. Probability density function for twenty, forty and a hundred steps ahead forecasting, for (a) “Model 1” (b) “Model 2”.

In the first case, in the medium and long term the error dispersion increases with the forecasting horizon. In our view this isn't the expected tendency. In consequence, we have trained and tested the second model. The growth trend of the error, observed in the medium term in case of “Model 1” is reduced. This indicate a “stabilization” of the forecasting model with the prediction horizon. Thus, we can observe that the additional information brought in the input of ANFIS contributes to the reduction of the errors and the improvement of the forecasting. In the future, in this direction, we intend to link ANFIS “in series” so one “forecast” can learn from the other ones.

4.2. Scenario 2

The second simulation scenario identifies in terms of forecasting performance (MAE and RMSE) the effect of membership function type and number changing. The models have been configured having two, or three membership functions (MFs), respectively, for each input and Gaussian and Gaussian Bell MFs. The computational results are presented in Tables 3 and 4.

Table 3. MAE and RMSE values computed in Scenario 2, for 2 MFs associated with the ANFIS inputs.

t + pred	ANFIS		MAE_Test		RMSE_Test	
	Type	No	Model 1	Model 2	Model 1	Model 2
1	gbellmf	2	0,0718	0,0579	0,4186	0,5325
4	gbellmf	2	0,2665	0,2863	1,1395	1,228
10	gbellmf	2	1,0757	−915	10,515	3,2758
20	gbellmf	2	3,3012	1,6894	5,3083	4,8206
40	gbellmf	2	3,0616	1,3997	5,5443	4,7185
1	gaussmf	2	0,0718	0,0579	0,4186	0,5325
4	gaussmf	2	0,2665	0,2863	1,1395	1,228
10	gaussmf	2	1,0757	−915	10,515	3,2758
20	gaussmf	2	3,3012	1,6894	5,3083	4,8206
40	gaussmf	2	3,0616	1,3997	5,5443	4,7185

Table 4. MAE and RMSE values computed in Scenario 2, for 3 MFs associated with the ANFIS inputs.

t + pred	ANFIS		MAE_Test		RMSE_Test	
	Type	No	Model 1	Model 2	Model 1	Model 2
1	gbellmf	3	0,0718	0,0598	0,5739	0,5325
4	gbellmf	3	0,2665	0,4808	1,2949	1,228
10	gbellmf	3	1,0757	−0,915	10,515	3,2758
20	gbellmf	3	3,0232	1,6526	5,4733	4,4743
40	gbellmf	3	3,2591	1,3997	5,5739	5,1016
1	gaussmf	3	0,0718	0,0598	0,5739	0,5325
4	gaussmf	3	0,2665	0,4808	1,2949	1,228
10	gaussmf	3	1,0757	−0,915	10,515	3,2758
20	gaussmf	3	3,0232	1,6526	5,4733	4,4743
40	gaussmf	3	3,2591	1,3997	5,5739	5,1016

Changing the type of MFs, for each model case, does not improve the forecasting performance. The error magnitude measured using MAE and RMSE metrics remains the same. This could be explained as a consequence of the small number of MFs used in the test phase. A bigger number of assigned MFs will refine the partitioning, but will also increase the computational time with no relevant influence over the forecasting precision. We have compared the effects of MF number and type changing for the two models in Figures 7 and 8.

The MAE and RMSE put in relation target and network’s output on testing data set and give an overview of generalization and memorization abilities of ANFIS. As expected, the error values in the case of Model 1 are higher.

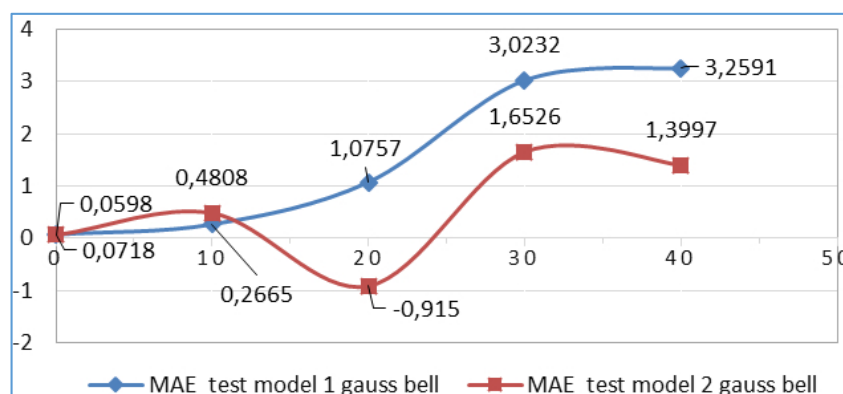


Figure 7. MAE for Gaussian Bell of MFs typologies.

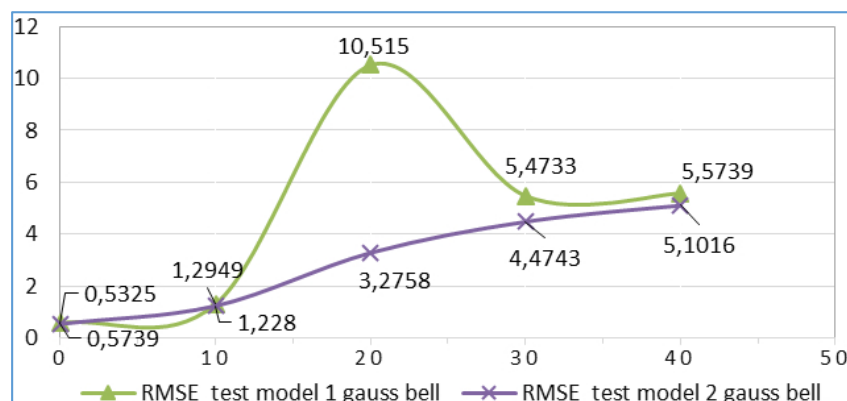


Figure 8. RMSE for Gaussian Bell of MFs typologies.

In the future, in this direction, we intend to extend the computations by using different types of MFs in the same interaction. For example, in the case of three MFs, the middle one will be triangular shaped and the other one, a Gaussian shape.

5. Conclusions

The paper presented here has as challenge controlling the performance of the future state of a microgrid with energy produced from renewable energy sources. The novelty aspect of our proposal consists in identifying the most used criteria, related to each modeling step, able to lead us to an optimal neural network forecasting tool. The approach was difficult because of the high numbers of variables and constraints that condition parameterization of a neuro-fuzzy network.

On the other hand, considering the aspects identified in the bibliographical research, we have proposed two case studies based on ANFIS models, underlining the consequences of the choices made by users during the tool modelling over the forecasting performances. Several scenarios are built by changing the prediction time horizon and the shape of the membership function in this respect. The work is still in progress and the developments are at present extended to the design and implementation of an intelligent informatics platform for forecasting and controlling energy generation from renewable energy resources and load in a distributed power system.

Acknowledgments: This work was supported by the Romanian National Authority for Scientific Research, CNDE-UEFISCDI, project code PN-II-PT-PCCA-2011-3.2-1616, “Intelligent Decision Support and Control System for Low Voltage Grids with Distributed Power Generation from Renewable Energy Sources” (INDESEN), Contract nr. 42/2012.

Author Contributions: O.E.D. conceived the idea of research, provide guidance and supervision and wrote the paper; F.D., V.S. and E.M. implemented the research and performed the analysis. All authors have contributed significantly to this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Boyle, G. *Renewable Energy: Power for a Sustainable Future*; Oxford University Press: Oxford, UK, 2012.
- Sorensen, B. *Renewable Energy, Fourth Edition: Physics, Engineering, Environmental Impacts, Economics & Planning*; Academic Press: Waltham, MA, USA, 2010.
- Masters, G. *Renewable and Efficient Electric Power Systems*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2004.
- Kohl, H. The Development. In *Renewable Energy*; Wengenmayr, R., Buhrke, T., Eds.; Wiley-VCH: Weinheim, Germany, 2008; pp. 4–14.
- Da Rosa, A.V. *Fundamentals of Renewable Energy Processes*; Academic Press: Oxford, UK, 2012.
- Kemp, W. *The Renewable Energy Handbook, Revised Edition: The Updated Comprehensive Guide to Renewable Energy and Independent Living*; Aztext Press: Tamworth, ON, Canada, 2009.

7. MacKay, D. *Sustainable Energy—Without the Hot Air*; UIT Cambridge Ltd.: Cambridge, UK, 2009.
8. Kaltschmitt, M.S. *Renewable Energy: Technology, Economics and Environment*; Springer: Berlin, Germany, 2010.
9. Bryce, R. *Power Hungry: The Myths of “Green” Energy and the Real Fuels of the Future*; Public Affairs: New York, NY, USA, 2011.
10. Chiras, D. *The Homeowner’s Guide to Renewable Energy: Achieving Energy Independence through Solar, Wind, Biomass, and Hydropower*; New Society Publishing: Gabriola Island, BC, Canada, 2011.
11. Boyle, G. *Renewable Electricity and the Grid*; Routledge: London, UK, 2007.
12. Rapiet, R. *Power Plays: Energy Options in the Age of Peak Oil*; Apress: New York, NY, USA, 2012.
13. Dragomir, O.; Dragomir, F.; Minca, E. An application oriented guideline for choosing a prognostic tool. In Proceedings of the AIP Conference Proceedings: 2nd Mediterranean Conference on Intelligent Systems and Automation, Zarzis, Tunisia, 23–25 March 2009; Volume 1107, pp. 257–262.
14. Dragomir, O.; Dragomir, F.; Gouriveau, R.; Minca, E. Medium term load forecasting using ANFIS predictor. In Proceedings of the 18th IEEE Mediterranean Conference on Control and Automation, Marrakech, Morocco, 23–25 June 2010; pp. 551–556.
15. Dragomir, O.; Dragomir, F.; Minca, E. Forecasting of renewable energy load with radial basis function (RBF) neural networks. In Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics, Noordwijkerhout, The Netherlands, 28–31 July 2011.
16. Jang, J.-S.R.; Sun, C.-T.; Mizutan, E. *Neuro-Fuzzy and Soft Computing*; Prentice Hall: New Jersey, USA, 1997.
17. Lacrose, A.; Tilti, A. Fusion and hierarchy can help fuzzy logic controller designer. In Proceedings of the IEEE International Conference on Fuzzy Systems, Barcelona, Spain, 1–5 July 1997.
18. Heaton, J. Introduction to neural networks with java, Heaton Research, Inc. Available online: <http://www.heatonresearch.com> (accessed on 21 May 2015).
19. Hippert, H.S. Neural networks for short-term load forecasting: A review and evaluation. *IEEE Trans. Power Syst.* **2001**, *16*, 44–55. [[CrossRef](#)]
20. Hong, Y.-Y.; Wei, Y.-H.; Chang, Y.-R.; Lee, Y.-D.; Liu, P.-W. Fault detection and location by static switches in microgrids using wavelet transform and adaptive network-based fuzzy inference system. *Energies* **2014**, *7*, 2658–2675. [[CrossRef](#)]
21. Maqsood, I.; Khan, M.R.; Abraham, A. Intelligent weather monitoring systems using connectionist neural models. *Parallel Sci. Comput.* **2002**, *10*, 157–178.
22. Hernández, L.; Baladrón, C.; Aguiar, J.M.; Calavia, L.; Carro, B.; Sánchez-Esguevillas, A.; Pérez, F.; Lloret, J. Artificial neural network for short-term load forecasting in distribution systems. *Energies* **2014**, *7*, 1576–1598. [[CrossRef](#)]
23. Elsholberg, A.; Simonovic, S.P.; Panu, U.S. Estimation of missing streamflow data using principles of chaos theory. *J. Hydrol.* **2002**, *255*, 123–133.
24. Cerjan, M.; Matijas, M.; Delimar, M. Dynamic hybrid model for short-term electricity price forecasting. *Energies* **2014**, *7*, 3304–3318. [[CrossRef](#)]
25. Babuska, R.; Jager, R.; Verbruggen, H.B. Interpolation issues in Sugeno-Takagi Reasoning. In Proceedings of the 3rd IEEE Conference on Fuzzy Systems, IEEE World Congress on Computational Intelligence, Orlando, FL, USA, 26–29 June 1994; pp. 859–863.
26. Fildes, R. The evaluation of extrapolative forecasting methods. *Int. J. Forecast.* **1992**, *8*, 81–98. [[CrossRef](#)]
27. De Gooijer, J.G.; Hyndman, R.J. 25 years of time series forecasting. *Int. J. Forecast.* **2006**, *22*, 443–473. [[CrossRef](#)]
28. Alfuhaid, A.S.; EL-Sayed, M.A.; Mahmoud, M.S. Cascaded artificial neural networks for short-term load forecasting. *IEEE Trans. Power Syst.* **1997**, *12*, 524–529. [[CrossRef](#)]
29. Chow, T.W.S.; Leung, C.T. Neural network based short-term load forecasting using weather compensation. *IEEE Trans. Power Syst.* **1996**, *11*, 1736–1742. [[CrossRef](#)]
30. Hobbs, B.F.; Jitprapaikulsarn, S.; Konda, S.; Chankong, V.; Loparo, K.A.; Maratukulam, D.J. Analysis of the value for unit commitment of improved load forecasting. *IEEE Trans. Power Syst.* **1999**, *14*, 1342–1348. [[CrossRef](#)]
31. Karady, G.G.; Farner, G.R. Economic impact analysis of load forecasting. *IEEE Trans. Power Syst.* **1997**, *12*, 1388–1392.
32. Armstrong, J.S.; Collopy, F. Error measures for generalizing about forecasting methods: Empirical comparisons. *Int. J. Forecast.* **1992**, *8*, 69–80. [[CrossRef](#)]

33. Armstrong, J.S.; Fildes, R. Correspondence on the selection of error measures for comparisons among forecasting methods. *J. Forecast.* **1995**, *14*, 67–71. [[CrossRef](#)]
34. Mohammed, O.; Park, D.; Merchant, R.; Dinh, T.; Tong, C.; Azeem, A.; Farah, J.; Drake, C. Practical experiences with an adaptive neural network short-term load forecasting system. *IEEE Trans. Power Syst.* **1995**, *10*, 254–265. [[CrossRef](#)]
35. Papalexopoulos, A.D.; Hao, S.; Peng, T.M. An implementation of a neural network based load forecasting model for the EMS. *IEEE Trans. Power Syst.* **1994**, *9*, 1956–1962. [[CrossRef](#)]
36. Peng, T.M.; Hubele, N.F.; Karady, G.G. Advancement in the application of neural networks for short-term load forecasting. *IEEE Trans. Power Syst.* **1992**, *7*, 250–257. [[CrossRef](#)]
37. Norouzi, A.; Hamed, M.; Adineh, V.R. Strength modeling and optimizing ultrasonic welded parts of ABS- PMMA using artificial intelligence methods. *Int. J. Adv. Manuf. Technol.* **2012**, *61*, 135–147. [[CrossRef](#)]
38. Collotta, M.; Messineo, A.; Nicolosi, G.; Giovanni, P.A. Dynamic Fuzzy Controller to Meet Thermal Comfort by Using Neural Network Forecasted Parameters as the Input. *Energies* **2014**, *7*, 4727–4756. [[CrossRef](#)]
39. Moon, J.W.; Chang, J.D.; Kim, S. Determining adaptability performance of artificial neural network-based thermal control logics for envelope conditions in residential buildings. *Energies* **2013**, *6*, 3548–3570. [[CrossRef](#)]
40. Jang, J.S.R.; Suni, C.T.; Mizutani, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*; Prentice Hall: New York, NY, USA, 1997.
41. Kusiak, A.; Wei, X. Prediction of methane production in wastewater treatment facility: A data-mining approach. *Ann. Oper. Res.* **2014**, *216*, 71–81. [[CrossRef](#)]
42. Iranmanesh, H.; Abdollahzade, M.; Miranian, A. Mid-term energy demand forecasting by hybrid neuro-fuzzy models. *Energies* **2012**, *5*, 1–21. [[CrossRef](#)]
43. Jang, J.S.R. ANFIS: Adaptive network based fuzzy inference systems. *IEEE Trans. Syst. Manuf. Cybern.* **1993**, *23*, 665–685. [[CrossRef](#)]
44. Wang, J.S. An efficient recurrent neuro-fuzzy system for identification and control of dynamic systems. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Washington, DC, USA, 5–8 October 2003.
45. Chiang, L.H.; Russel, E.; Braatz, R. *Fault Detection and Diagnosis in Industrial Systems*; Springer Verlag: London, UK, 2001.
46. Gourierou, C. *ARCH Models and Financial Applications*; Springer: New York, NY, USA, 1997.
47. Chan, K.-Y.; Gu, J.-C. Modeling of turbine cycles using a neuro-fuzzy based approach to predict turbine-generator output for nuclear power plants. *Energies* **2012**, *5*, 101–118. [[CrossRef](#)]



© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).