ECONTECHMOD. AN INTERNATIONAL QUARTERLY JOURNAL – 2016, Vol.5, No.4, 85-95

BLACK BOX DYNAMIC MODELLING OF PROTON EXCHANGE MEMBRANE FUEL CELLS WITH ARTIFICIAL NEURAL NETWORKS

J. Kapica
Department of Electrical Engineering and Measurement Systems,
University of Life Sciences in Lublin; e-mail: jacek.kapica@up.lublin.pl

Received November 21.2016: accepted December 23.2016

Abstract. The fuel cells are energy sources which can play an important role in transition of the energy sector into broader use of renewable energy. Numerical modelling provides an easy way to investigate properties of the objects modelled. There are various ways to model dynamic behaviour of the PEM fuel cells including methods using artificial neural networks. There are no clear rules of how a neural network should be configured: how many neurons in the hidden layer and which training algorithm should be used. In a time series modelling task additional parameters including sampling frequency, learning data set duration and number of past data points used for training need to be determined. The paper presents results of research on the influence of various model parameters on the PEM fuel cell modelling accuracy.

Key words: PEM fuel cells, neural network model, dynamic behaviour, black box.

INTRODUCTION

The fuel cells (FC) are among the most promising technologies offering transition of the energy sector from the traditional power sources into a new stage. One of the application is energy generation from hydrogen produced by a renewable power source with variable output (mostly wind or photovoltaics). A FC stack supplemented by hydrogen generator creates a power storage unit improving the functionality and usefulness of the renewable energy generation set by load levelling and customer-side peak shaving. In general, storage techniques can be classified into the following four major groups, with FC systems categorized under chemical energy storage technologies [2]:

- electrical energy storage technologies, including: capacitor and supercapacitor storage, superconducting magnetic energy storage,
- mechanical energy storage technologies, including: flywheel energy storage, compressed air energy storage, pumped hydro storage ,
- chemical energy storage technologies, including: battery energy storage, fuel cell systems,
- thermal energy storage technologies, including: aquiferous thermal energy storage, cryogenic energy storage, hot thermal energy storage, pumped heat electrical storage.

Currently there are six types of established fuel cell technologies available on the market [10]:

- Proton Exchange Membrane Fuel Cell (PEMFC),
- Alkaline Fuel Cell (AFC),
- Direct Methanol Fuel Cell (DMFC),
- Phosphoric Acid Fuel Cell (PAFC),
- Molten Carbonate Fuel Cell (MCFC),
- Solid Oxide Fuel Cell (SOFC).

The PEMFCs are considered as having important advantages over other technologies [17]: ability to operate continuously at low temperature and high current densities, long stack life, short start-up time, capability of discontinuous operation, small size and high tolerance to shock and vibration. These features make this technology useful in the creation of electric vehicles, including bicycles, motorcycles, mini-trains, cars and buses[30]. Currently, the main disadvantages are high capital cost and low round trip efficiency of hydrogen fuel cell storage systems, at the level of 40-45 % [3].

Fuel cells, as a relatively new technology, are costly devices which results in high prototyping costs. In order to decrease the expenditure, numerical modelling can be used for general assessment of properties of the system in the preliminary state of the design work. One of the properties of the PEMFCs is scalability, so obtaining a good model of a small (and inexpensive) FC stack allows the creation of a model of a higher power stack, which can be used to test various configurations and layouts of the energy storage or production system.

Another advantage of numerical modelling is that with efficient algorithm and computer system the simulation usually is faster than a real-world experiment.

There can be various approaches to FC modelling. Some authors present models based on the physical and chemical laws that govern inside the cell, thus creating physical or semi-physical models [31, 4, 14, 11]. These models usually provide good accuracy, but they are not convenient to use. The main reason is that they require the cell parameters which are not directly available. Simulating dynamic states is even more challenging as it involves complex phenomena.

The other group is often called “black box”. The name suggests that the model treats the simulated object as a processing unit in which one must find dependency between inputs and outputs without trying to resemble the phenomena taking place in the device. Another name used is data-driven models which suggests that the model is created on the foundation of experiments and measurements conducted on the object being modelled. The methods used include: Uryson-Model [16], constructing equivalent electrical circuit [22], adaptive spline modelling of observation data [25], non-linear autoregressive moving average model with exogenous inputs [7].

The artificial neural networks (ANNs), as data-driven methods, have also been used to model both static and dynamic performance of the PEMFCs [19, 24, 18, 27]. They are mathematical structures inspired by biological neural systems. The most commonly used architecture is a multilayer perceptron which makes use of weights, usually sigmoid activation functions and a hidden layer consisting of varying number of nodes [23]. There is a number of applications of ANNs which can be divided into the following groups [26]: classification (pattern recognition) [29], density estimation, clustering and regression [20], including modelling of time series [9]. In this case the inputs consists of chosen number of past values of the input variables, enabling forecasting future output of a system. This can be done in one of two ways: open loop, when the output is predicted when past measured values of the output are available and closed loop, when only output of the model for the past points of time is available. The first situation takes place in the one step ahead forecasting, the latter in modelling of a system when only the initial state is known or assumed. A major advantage over other modelling techniques is low computational requirements once the network has been trained.

The selection of the hidden layer size is often regarded as "art" as there are no clear rules of how to approach this problem. Some classical textbooks set the "rule of thumb" that the number of hidden neurons should not be greater than the number of inputs [6, 28]. In practice, the preferable network size depends on the particular case of the object modelled. In case of the times series modelling there are additional questions about how long the training period should be, what sampling frequency will provide best results.

Most, if not all of the publications presenting ANN models are not fully "black box" as they take the inner temperature of the cell as one of the input parameters. The approach presented in this paper can be regarded as a true "black box" as it uses ambient temperature as one of the inputs.

The purpose of the research presented in this paper is to create a simple ANN model driven entirely on data external to the PEM FC, determine which hidden layer size - training set duration - sampling frequency - training function - performance function combinations give better results than other. Also, it was interesting to check whether it is necessary to use temperature and hydrogen pressure as inputs in order to achieve satisfactory modelling accuracy.

ARTIFICIAL NEURAL NETWORKS AND TIME SERIES MODELLING

A feedforward neural network with one input, one hidden and one output layer can be regarded as one of the simplest ANN configurations (Fig. 1). The input layer consists of I nodes, with each node representing one input variable $x_1 \dots x_I$. The main structure within the network is the hidden layer made of N neurons.

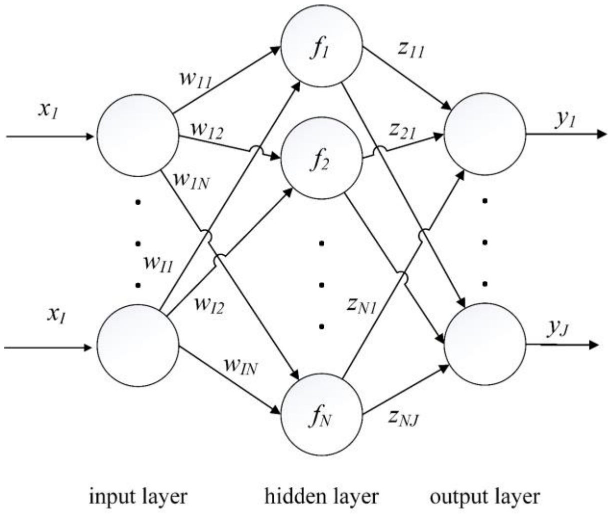

Fig. 1. Structure of a feedforward ANN with one hidden layer

Each of the hidden neurons performs an operation adding products of input variables x, the corresponding weights w and biases b with the use of an activation function g [12]:

$$f_n = g\left[\sum_{i=1}^{I}(w_i + b_n)\right], \quad i = 1,2,...,I\ , \quad n = 1,2,...,N\ . \quad (1)$$

Usually g is a log-sigmoid function defined as:

$$g(x) = \frac{1}{1+e^{-x}}\ , \quad (2)$$

or a tan-sigmoid function [8]:

$$g(x) = \frac{2}{1+e^{-2x}} - 1\ . \quad (3)$$

The output layer performs calculations according to equation:

$$y_j = \sum_{n=1}^{N} f_n z_{nj}, \quad j = 1,2,\dots,J, n = 1,2,\dots,N\ . \quad (4)$$

The major step in creation of an ANN model of a given object is training of the network. In this process the information is stored in the weights and biases. It involves minimising a cost function C by finding near-optimal values for the network weights. This is done by feeding the inputs with an example data and calculating partial derivatives of C with respect to all the individual weights within the network [5] with required outputs known from the example data. The cost function can have various forms, one of them is a mean squared error (6).

Time series is a sequence of vectors *X(t)*, with *t* denoting successive points in time, where *t*=1, 2, ... , *n*. *X* is commonly obtained from measurements, in which one of the stages is sampling the continuous signal in order to obtain discrete, usually uniformly spaced in time, data points.

The most common task where ANNs are used is forecasting the *X* values of one or more time steps ahead. This can be achieved by creating a network with input vector *I* consisting of stacked *k* past values of *X*. Assuming that *X* is formed of *m* scalars:

$$I = \begin{bmatrix} X(1,t-k), X(1,t-k+1),..., \\ X(1,t-1), X(2,t-k), X(2;t-k+1),..., X(2,t-1),..., \\ X(m,t-k), X(m,t-k+1),..., X(m,t-1) \end{bmatrix}. \quad (5)$$

Vector *I* is used to model the value of *X* at time point *t*. In addition to the past values of *X*, other variables can be used in creation of *I*. Examples of applications of ANNs to time series modelling include forecasting in the area of finances [1], weather [15], electricity consumption [13], hydrology [21] and other.

EXPERIMENTAL SET-UP AND METHODS USED

Fig. 2 shows the experimental set-up. It consists of a PEM fuel cell, metal hydride hydrogen storage tank and electronic programmable load. In order to measure the parameters current, voltage, temperature and pressure sensors were used. The whole experiment was controlled and the data collected by a computer program developed by the author. The temperature, current and voltage were measured via an analogue to digital converter and the pressure through the RS232 interface. The electronic load was also controlled using the RS232 interface. In order to control the ambient temperature, the fuel cell was placed in a climate chamber. The fuel cell used was a 12 W PEM fuel cell stack, type H-12 manufactured by Horizon. It is cooled by air with an integrated fan. The compact construction makes it suitable for small projects. It is a simple, self-humidified air-fed stack.

The first step was to collect data from an experiment, during which the ambient temperature was changed from 10 to 40 ºC. The hydrogen pressure was changing only as a result of the tank discharge and varied between 50 and 56 kPa. During the experiment, the fuel cell was put under load of various current levels changed in a step-like and triangle manner in the range of 0 to 2 A. The data were collected over a period of 28 minutes with the rate of 400 samples per second, which resulted in the raw set of almost 70 000 data points used later to produce input and output sets for the neural network training and testing.

The network was configured as feedforward with backpropagation training algorithm. In order to test the accuracy of the modelling various configurations of the training algorithm and input data set were used which are summarized in Table 1. The two training function algorithms were chosen for the sake of low training computation time as compared to other. In this way 1440 combinations of the parameters were obtained. The following stoppage criteria were defined: the maximum of 400 training epochs or 6 maximum validation failures, whichever comes first.

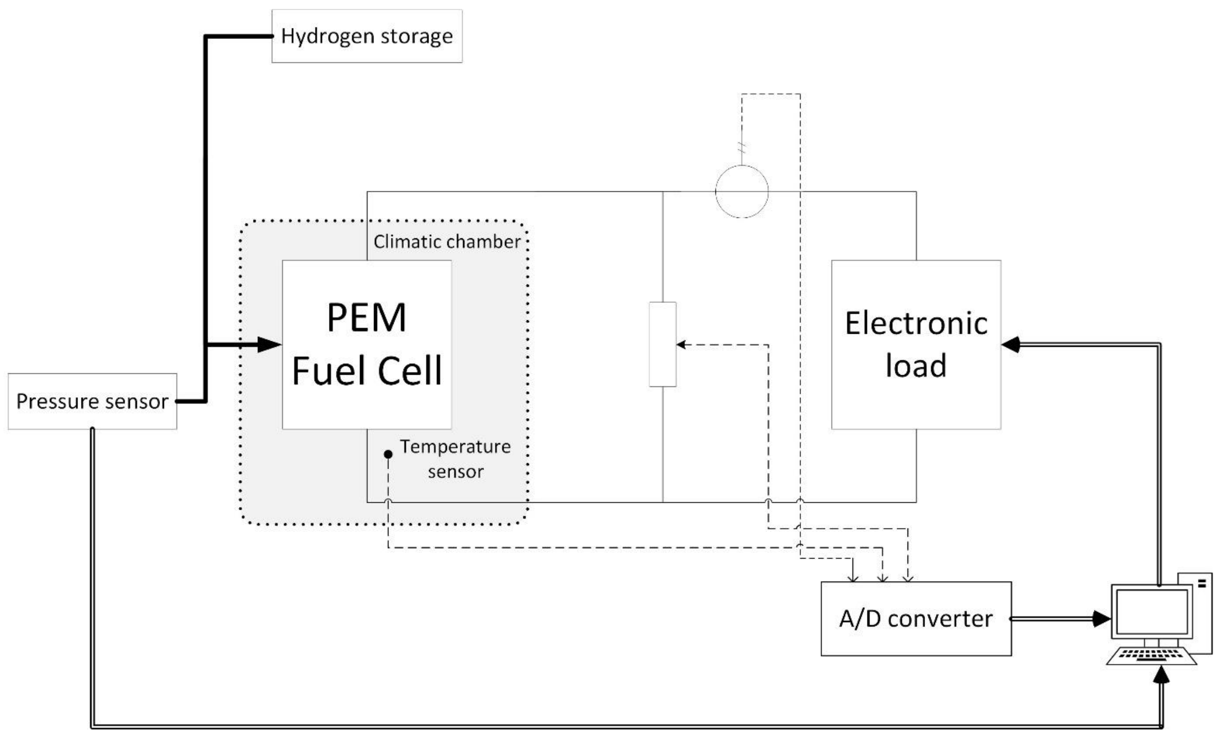

Fig. 2. Diagram of the experimental set-up

Table 1. Values of the parameters changed for modelling accuracy testing

Parameter	Values
Resampling	1, 2, 3
Number of delays	1, 2, 3, 4
Duration of the training and testing set	50, 100, 150, 200, 250 seconds
Training function	Levenberg-Marquardt (trainlm) and Bayesian regulation backpropagation (trainbr),
Performance function	Mean squared normalized (mse), Sum squared error (sse)
Number of neurons in the hidden layer	2, 5, 8, 11, 14, 17, 20, 23

Resampling procedure relates to collecting every i-th data point and deleting other points, thereby reducing the effective sampling frequency. Delay parameter specifies how many past data points are fed into the neural network as input.

Neural network model was implemented in Matlab. A script has been created to handle the parameter variation, training and testing of the network. The procedure involved dividing the original raw set into 30 pieces evenly distributed in time. As the temperature was changing in time, the subsets were also distributed according to temperature. Half of the sets were used for training (with subdivisions into training, testing and validation). The other half was then used to test the modelling accuracy. This was done in order to prevent the effect of over-fitting influencing the results - in our case the learning set and set used for testing and error estimation are completely different. The tests were performed in a closed loop configuration, which means that the past voltage data points were not taken from measurements but as a result of modelling from the previous time points, as it would be in a practical modelling task, where the previous voltage values are not known but have to be calculated. Fig. 3 shows the data flow in the model.

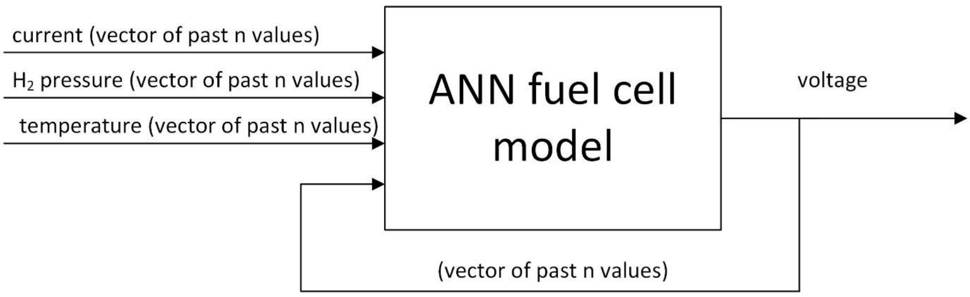

Fig. 3. Data flow in the model

Additionally, three other tests have been performed: modelling without using temperature as input, without pressure and without both of these parameters. This was done to test how the network will perform when limited data is available.

The outcome of the experiments was a 6-dimensional array containing mean square error (MSE) calculated as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(X_i - Y_i\right)^2 . \tag{6}$$

where: X_i is the modelled voltage (output of the ANN) and Y_i is the measured voltage at point i.

RESULTS AND DISCUSSION

In order to evaluate the influence of various factors on the modelling accuracy the results were analysed in two ways. The first method used was to divide the results into two groups: one under a certain MSE level, the other above that point. The MSE threshold values were calculated according to the Matlab quantile function, which returns quantile value for a given probability. Table 2 shows MSE threshold value corresponding to the given quantile threshold probability level, minimum and maximum MSE values. Results are illustrated on Fig. 4 and 5. The 'dot' markers represent configurations for which the MSE was above threshold and the 'x' markers - under the threshold. Therefore, network configurations with better modelling quality are marked with an 'x'.

Table 2. Description of the conditions used in the illustration of the accuracy (Fig. 4 and 5)

Figure	Pressure used as input?	Temperature used as input?	Quantile threshold	MSE threshold (V)	Minimum value of MSE (V)	Maximum value of MSE (V)
Fig. 4a	Yes	Yes	0.003	0.0046	0.0043	143
Fig. 5a	Yes	Yes	0.02	0.0053	0.0043	143
Fig. 4b	Yes	No	0.003	0.0148	0.0137	186
Fig. 5b	No	Yes	0.003	0.0056	0.0053	81
Fig. 4c	No	Yes	0.02	0.006	0.0053	81
Fig. 5c	No	Yes	-	0.0053	0.0053	81

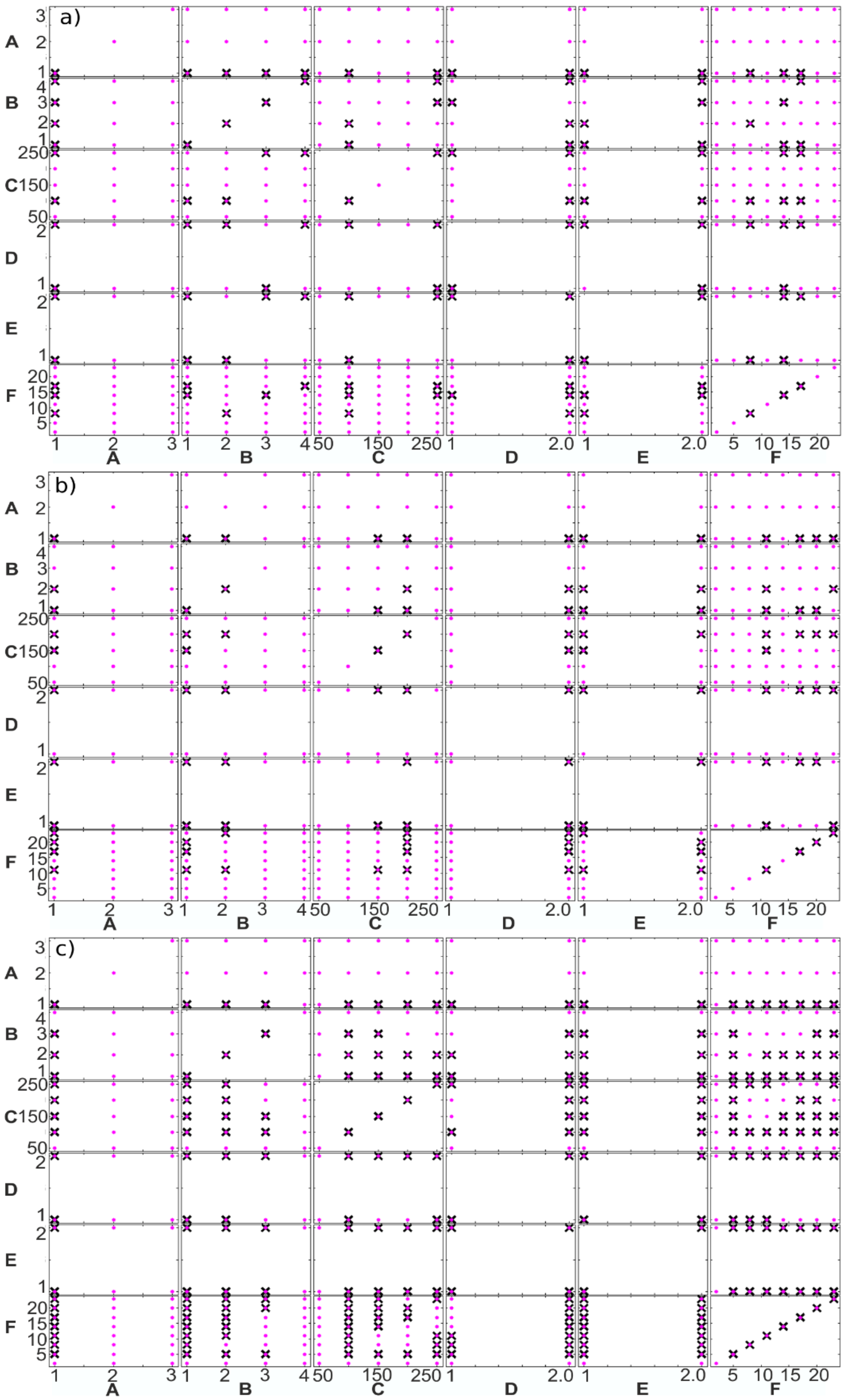

Fig. 4. Accuracy of modelling. The 'x' markers show simulation and learning properties for which MSE is below the threshold. Description of modelling parameters can be found in Table 2. The Levenberg-Marquardt (trainlm) training function is marked as 1 and Bayesian regulation backpropagation (trainbr) - as 2. Mean squared normalized (mse) performance estimator is marked as 1, Sum squared error (sse) - as 2. Axes description: A - Resampling factor, B – Number of delays, C – Training time (s), D- Training function, E – Performance estimator, F – Number of neurons

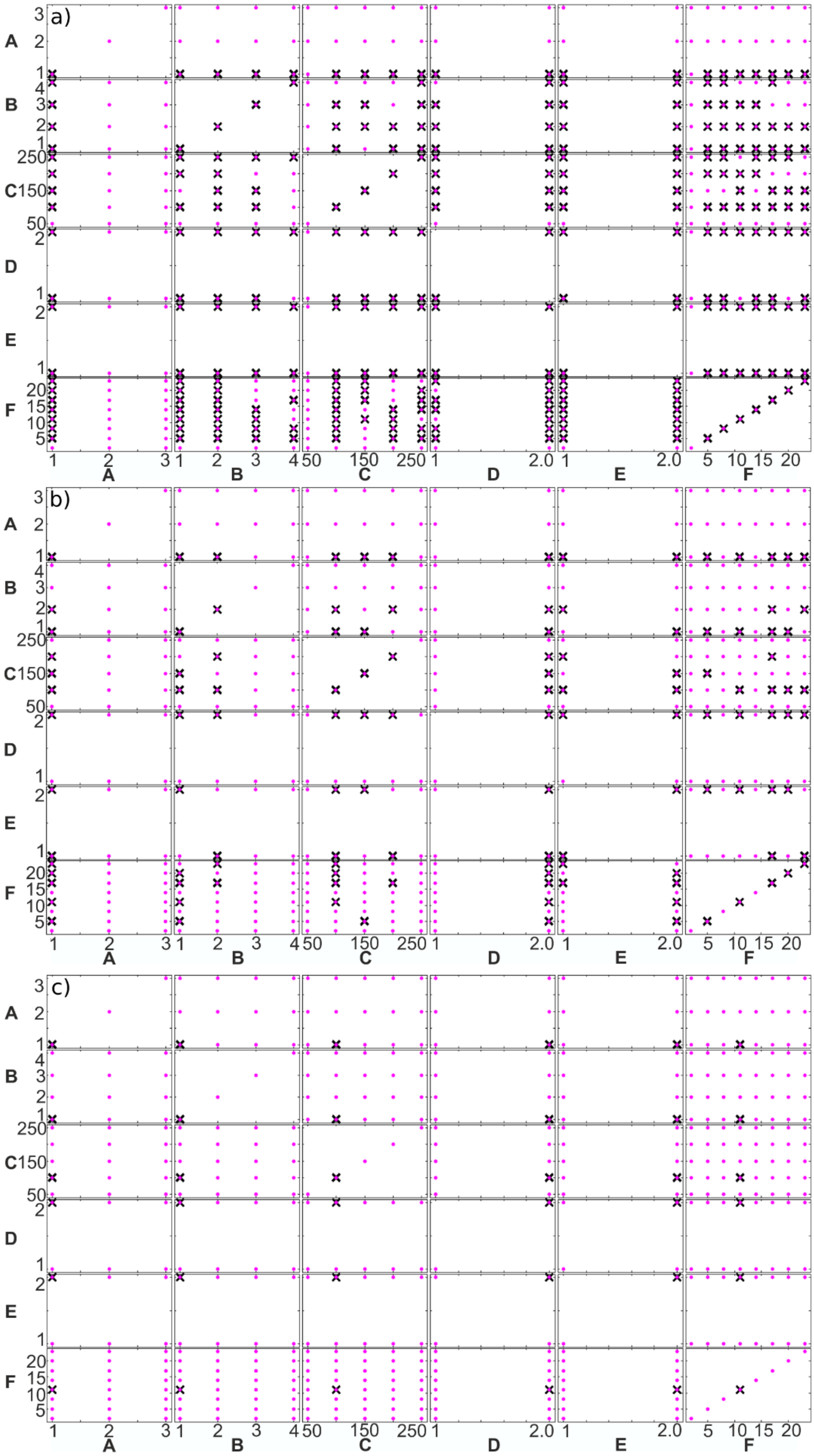

Fig. 5. Accuracy of modelling. Description as in Fig. 4

The table shows that in a case when temperature was not considered, the minimum MSE is much higher than in other cases. Moreover, when both the temperature and pressure were neglected, the accuracy was even worse and therefore not presented in the paper. In the model where only the pressure was ignored the error was slightly higher compared to the case with both temperature and pressure included. This is caused by a low variation of the pressure value during the experiment. All the results show that higher sampling frequency offers the best accuracy of the model. Usually, taking an input vector of just one past value gives best results, except the case shown on Fig. 4a, where longer delays offer good accuracy when combined with longer training period duration. Also increasing the training time period over 100 seconds does not improve results, except the case when the temperature is not taken into account, where at least 150 seconds are needed for better outcome. The Bayesian regulation back-propagation training function usually offers better results. Sum of squared errors serves slightly better as a performance estimator, mean squared error performs in a similar way. Increasing the number of the neurons in the hidden layer over 5 usually improves accuracy. The network can well model the current voltage value with data from just one, past point in time.

In order to achieve similar accuracy in the model without pressure input as in the model with both pressure and temperature (MSE threshold 0.0053 V) the following parameters were needed (Fig. 5c): resampling factor - 1, number of delays - 1, training time - 100 s, training function - trainbr, performance estimator - MSE, number of neurons - 12.

The second way to compare the results was to order them by the MSE, starting with ones with best accuracy (Table 3 and 4 present some of best configurations). Also here it can be seen that shorter example sets in most cases offer best results. Bayesian regulation backpropagation performs better as a training function. It can be seen clearly looking at the tables where this function is placed in most of the cases. To achieve similar results as in the first row of Table 3 with Levenberg-Marquardt backpropagation a vector of 3 past and an example set of 250 s duration is needed and still the MSE is slightly higher. In the case presented in Table 4 with Levenberg-Marquardt backpropagation a network with more neurons in the hidden layer were needed for a similar accuracy as in the first row. Tables 3 and 4 contain a column with time that was needed to train the network. This parameter is highly dependent on the hardware of the computer used so the numbers can be only used to compare various cases. Generally network with more inputs (longer delays) and more nodes in the hidden layer need more time to train, however this is not a strict rule - for example first two rows in both of the tables show that despite larger network size the training time was shorter. The computer used for training was a Windows 7 64 bit - based system with Intel(R) Core(TM) i5-3340M CPU @ 2.70GHz processor with 4 GB of RAM.

Fig. 6 and 7 present time and regression plots of the best performing models from Table 3 and 4. The model output matches the measurement curve well, as can be seen in both cases. In the model without hydrogen pressure considered, the output of the model differs from measurements more significantly especially during fast transitions and floats around the target curve.

The regression plots show very good output approximation by the network. The overwhelming majority of points lay on the $Y=T$ line and the correlation coefficient is well over 0.99 in all of the cases, with small number of outliers.

Table 3. Modelling parameters offering best accuracy - model including both temperature and pressure

Resampling	Delays	Duration of the example set (s)	Training function	Performance estimator	Number of neurons	MSE (V)	Time needed to train (s)
1	1	100	trainbr	sse	14	0.0043	75
1	1	100	trainbr	sse	17	0.0044	55
1	2	100	trainbr	mse	8	0.0045	71
1	3	250	trainlm	sse	14	0.0046	45
1	1	100	trainbr	mse	14	0.0046	99
1	4	250	trainbr	sse	17	0.0046	356
1	2	150	trainbr	sse	20	0.0046	171

Table 4. Modelling parameters offering best accuracy - model including temperature, without pressure

Resampling	Delays	Duration of the example set (s)	Training function	Performance estimator	Number of neurons	MSE (V)	Time needed to train (s)
1	1	100	trainbr	sse	11	0.0053	64
1	1	100	trainlm	sse	20	0.0054	58
1	1	100	trainbr	sse	17	0.0054	29
1	2	200	trainbr	mse	17	0.0055	145
1	1	150	trainbr	sse	5	0.0056	130
1	2	100	trainbr	mse	23	0.0056	38
1	1	250	trainbr	sse	8	0.0056	95
1	1	100	trainbr	mse	8	0.0056	51

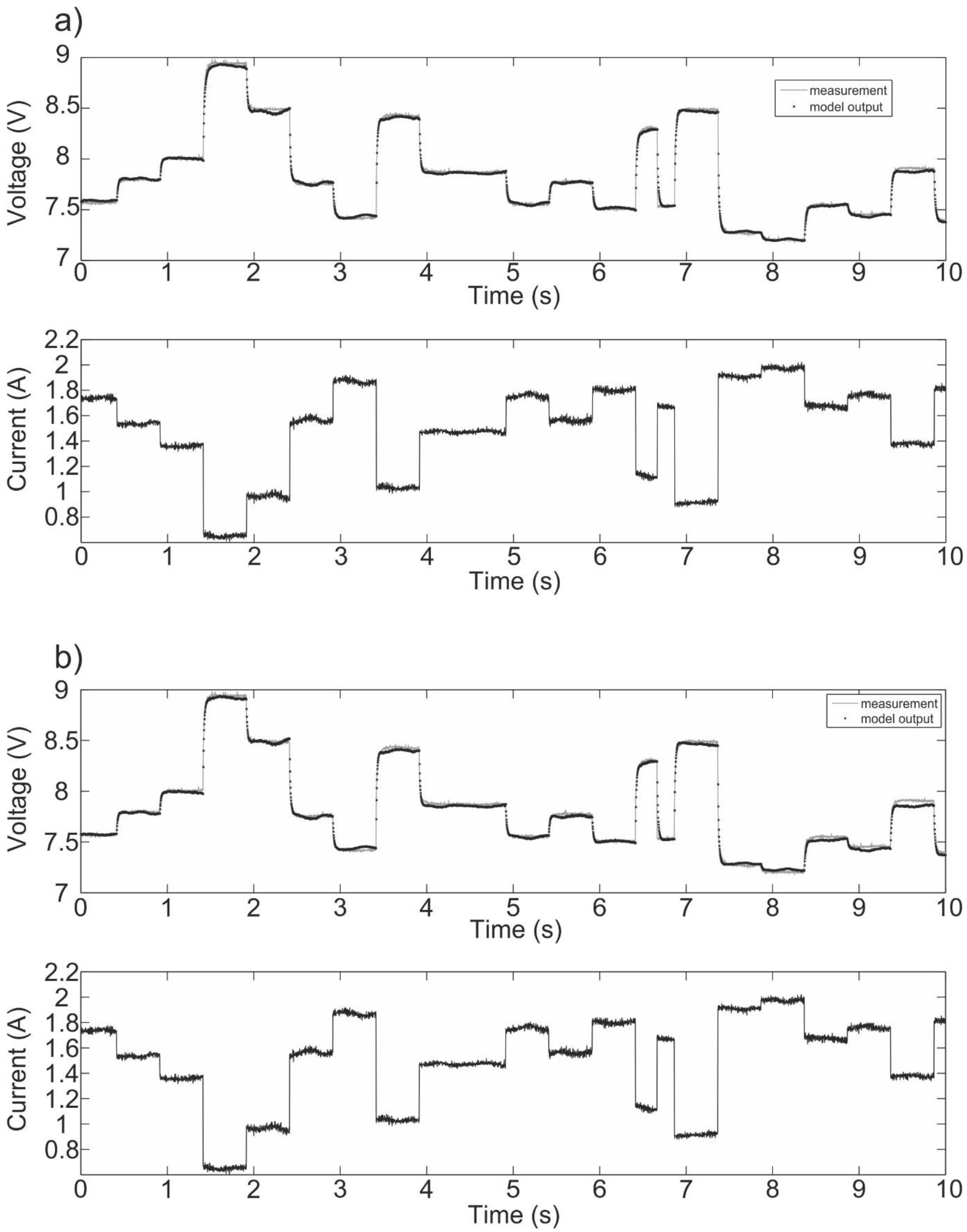

Fig. 6. Time plots for the model configuration providing best accuracy with pressure used as input (a) and without temperature (b)

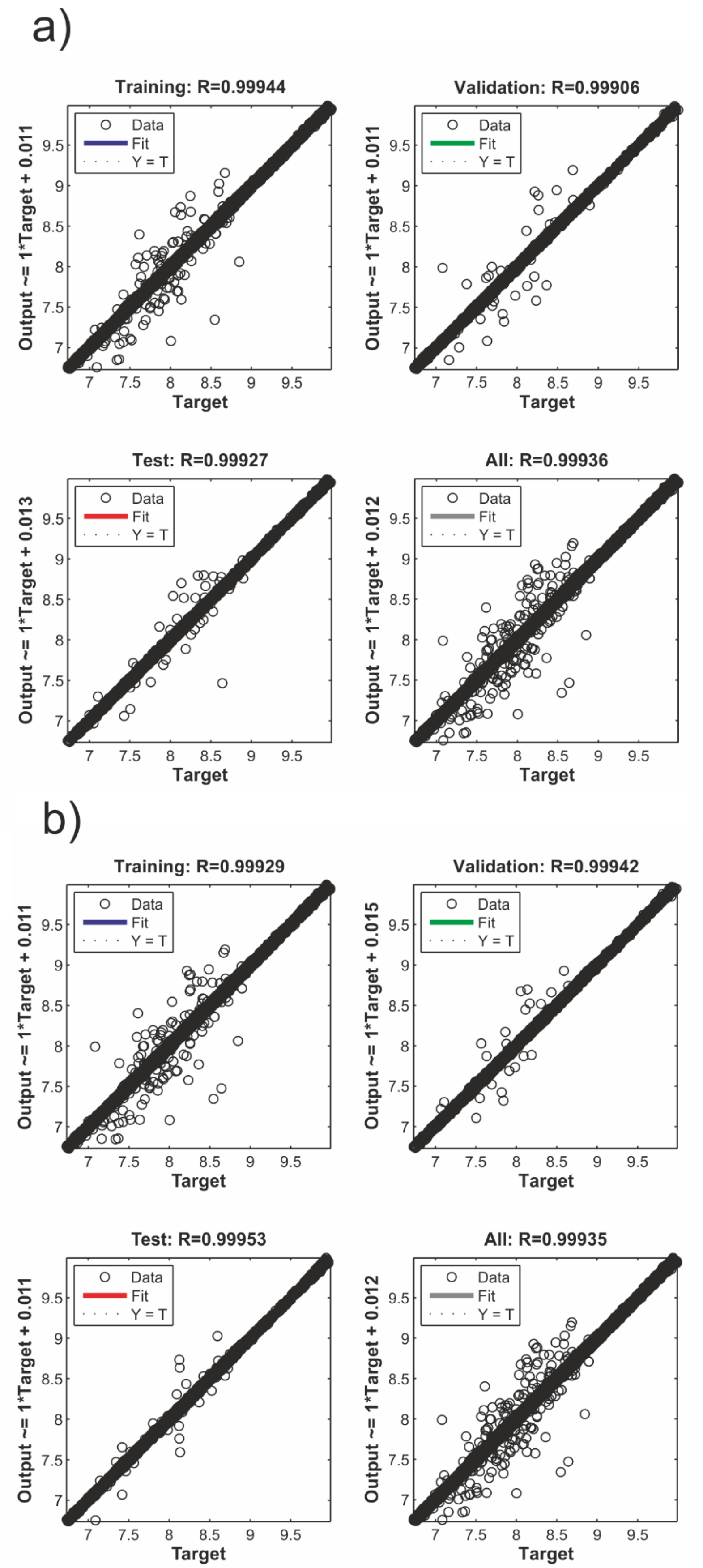

Fig. 7. Regression plots for the model configuration providing best accuracy with pressure used as input (a) and without temperature (b)

CONCLUSIONS

A model of a 12 W PEM fuel cell employing artificial neural networks has been created. The input variables of the model were: electrical current, hydrogen pressure and outside temperature. The output of the model was stack voltage. The model accuracy was tested in various configurations in which the following parameters were changed: sampling frequency, learning dataset duration, training function, performance estimation function, number of neurons, number of delays (past data points used). Altogether 1440 combinations were tested.

The results prove that with proper selection of the parameters it is possible to achieve good simulation accuracy of up to MSE=0.0043 V in modelling of the dynamic state. It has been shown that the high sampling frequency (400 S/s in the analysed case) and one delay point provide best accuracy. There is no serious difference whether MSE or SSE is used as error estimator. Bayesian regulation backpropagation usually gives better training results than Levenberg-Marquardt when used as a training algorithm. In most cases an example data set of a duration of 100 s was enough to provide best results. There is no evidence that the “rule of thumb” (number of neurons in the hidden layer does not need to be higher than number of inputs) mentioned in the Introduction is useful in designing neural network.

It has also been tested if it is necessary to take temperature and hydrogen pressure as inputs of the model. In the analysed case the variation of the hydrogen pressure was relatively small (50 to 56 kPa) and it was possible to achieve good results (up to MSE=0.0053 V) with this variable ignored. Temperature was varied in the range of 10 to 40 °C. Omitting this variable resulted in a considerable decrease of the model accuracy - best results with MSE= 0.0137 V.

The investigation presented in this paper may give guidance to a researcher seeking a simple model, which can be created without detailed physical data of the FC being modelled when sample data can be obtained from an easy to perform experiment. The model can be used to model dynamic behaviour of a PEM fuel cell as a part of energy management system.

REFERENCES

1. **Adhikari R., Agrawal R., 2014.** A combination of artificial neural network and random walk models for financial time series forecasting. Neural Computing and Applications. 24(6). 1441-1449.

2. **Akinyele D., Rayudu R., 2014.** Review of energy storage technologies for sustainable power networks. Sustainable Energy Technologies and Assessments. 8. 74-91.

3. **Andrews J., Mohammadi S., 2014.** Towards a 'proton flow battery': Investigation of a reversible {PEM} fuel cell with integrated metal-hydride hydrogen storage. International Journal of Hydrogen Energy. 39(4). 1740-1751.

4. **Sharifi A., Rowshanzamir S., Eikani M., 2010.** Modelling and simulation of the steady-state and dynamic behaviour of a {PEM} fuel cell. Energy. 35(4). 1633-1646. Demand Response Resources: the {US} and International Experience.

5. **Bengio Y., Goodfellow I., Courville A., 2015.** Deep learning. Book in preparation for MIT Press.

6. **Berry M., Linoff G., 1997.** Data Mining Techniques: For Marketing, Sales, and Customer Support. John Wiley & Sons, Inc. New York, NY, USA.

7. **Cheng S., Liu J., 2015.** Nonlinear modelling and identification of proton exchange membrane fuel cell (pemfc). International Journal of Hydrogen Energy. 40(30). 9452-9461.

8. **Chowdhury S., Das Saha P., 2013.** Artificial neural network modelling of adsorption of methylene blue by naoh-modified rice husk in a fixed-bed column system. Environmental Science and Pollution Research. 20(2).1050-1058.

9. **Cui Y., Shi J., Wang Z., 2015.** Complex rotation quantum dynamic neural networks using complex quantum neuron: Applications to time series prediction. Neural Networks 71. 11-26.

10. **Elmer T., Worall M., Wu S., Riffat S., 2015.** Fuel cell technology for domestic built environment applications: State of-the-art review. Renewable and Sustainable Energy Reviews. 42. 913-931.

11. **Guarnieri M., Alotto P., Moro F., 2015.** Modeling the performance of hydrogen-oxygen unitized regenerative proton exchange membrane fuel cells for energy storage. Journal of Power Sources. 297. 23-32.

12. **Han H., Qiao J., 2013.** A structure optimisation algorithm for feedforward neural network construction. Neurocomputing. 99. 347-357.

13. **Kaytez F., Taplamacioglu M., Cam E., Hardalac F., 2015.** Forecasting electricity consumption: A comparison of regression analysis, neural networks and least squares support vector machines. International Journal of Electrical Power & Energy Systems. 67. 431-438.

14. **Latha K., Vidhya S., Umamaheswari B., Rajalakshmi N., Dhathathreyan K., 2013.** Tuning of {PEM} fuel cell model parameters forediction of steady state and dynamic performance under various operating conditions. International Journal of Hydrogen Energy. 38(5). 2370-2386.

15. **Marohasy J., Abbot J., 2015.** Assessing the quality of eight different maximum temperature time series as inputs when using artificial neural networks to forecast monthly rainfall at cape Otway, Australia. Atmospheric Research. 166. 141-149.

16. **Meiler M., Schmid O., Schudy M., Hofer E. 2008.** Dynamic fuel cell stack model for real-time simulation based on system identification. Journal of Power Sources. 176(2). 523-528. Selected Papers presented at the10th {ULM} ElectroChemical Days.

17. **Moreno N., Molina M., Gervasio D., Robles J., 2015.** Approaches to polymer electrolyte membrane fuel cells (pemfcs) and their cost. Renewable and Sustainable Energy Reviews. 52. 897-906.

18. **Napoli G., Ferraro M., Sergi F., Brunaccini G., Antonucci V., 2013.** Data driven models for a {PEM} fuel cell stack performance prediction. International Journal of Hydrogen Energy. 38(26). 11628-11638.

19. **Ou S., Achenie L., 2005.** A hybrid neural network model for {PEM} fuel cells. Journal of Power Sources. 140(2). 319–330.

20. **Pei J., Mai E., Wright J., Masri S., 2013.** Mapping some basic functions and operations to multilayer feedforward neural networks for modelling nonlinear dynamical systems and beyond. Nonlinear Dynamics. 71(1-2). 371-399.

21. **Piotrowski A., Napiorkowski M., Napiorkowski J., Osuch M., 2015.** Comparing various artificial neural network types for water temperature prediction in rivers. Journal of Hydrology. 529, Part 1. 302-315.

22. **Raga C., Barrado A., Lazaro A., Fernandez C., Valdivia V., Quesada I., Gauchia L., 2014.** Black-box model, identification technique and frequency analysis for pem fuel cell with overshooted transient response. IEEE Transactions on Power Electronics. 29 (10). 5334-5346.

23. **Rocha M., Cortez P., Neves J., 2005.** Evolution of neural networks for classification and regression. Neurocomputing. 70(16-18). 2809-2816, 2007. Neural Network Applications in Electrical Engineering. Selected papers from the 3-rd International Work-Conference on Artificial Neural Networks (IWANN 2005).

24. **Saengrung A., Abtahi A., Zilouchian A,. 2007.** Neural network model for a commercial {PEM} fuel cell system. Journal of Power Sources. 172(2). 749-759.

25. **Schrems A., Pichler K., Steinmaurer G., Wahlmuller E., 2008.** Data based modelling for pem fuel cell monitoring - a test bench application, part i. In Control Applications. CCA 2008. IEEE International Conference. 262-267.

26. **Sodhi S., Chandra P., 2014.** Bi-modal derivative activation function for sigmoidal feedforward networks. Neurocomputing. 143. 182-196.

27. **Sreemathy S., Karthik M., 2015.** Modelling and performance analysis of neural network based fuel cell driven electric traction system. In Innovations in Information, Embedded and Communication Systems (ICIIECS). 2015 International Conference. 1-6.

28. **Swingler K., 1996.** Applying neural networks: A practical guide. London: Academic Press.

29. **van der Velde F., 2015.** Computation and dissipative dynamical systems in neural networks for classification. Pattern Recognition Letters. 64. 44-52.

30. **Wang F., Gao C., Li S., 2014.** Impacts of power management on a {PEMFC} electric vehicle. International Journal of Hydrogen Energy. 39(30). 17336-17346.

31. **Xue X., Cheng K., Sutanto D., 2006.** Unified mathematical modelling of steady-state and dynamic voltage-current characteristics for {PEM} fuel cells. Electrochimica Acta. 52(3). 1135-1144.