PEMFC Modeling Considering Destructive Signal and Design Adaptive Filter

M. khorasani
Department of Electrical Engineering, Shiraz University
Shiraz, Iran
M.khorasani@ieee.org

H. A. Aalami
Department of Electrical Engineering, Imam Hussein University Tehran, Iran
h_aalami@yahoo.com

***Abstract*— Proton exchange membrane fuel cells (PEMFCs) can be used in transportation, power plants and distributed generation due to high power density and fast speed of operation. Dynamic modeling of fuel cells is a primary need for performance assessment studies of real-time and controller design. Some new models of Simulink use artificial intelligence for making graph output model. Neural network is one of these models. Neural network just uses input – output data that has obtained in several experiments and does not need to set all the parameters. In this paper, first, the PEMFC has been simulated using feedforward neural networks. This network has been trained using different algorithms and the results were compared to determine the appropriate algorithm by MADM methods. Then the effect of destructive signal on the neural network is evaluated and the authors are attempted to reduce this effect by developing a suitable adaptive filter.**

***Keywords*— ANN, DG, entropy, filter, MADM, PEMFC, SAW**

I. INTRODUCTION

Today, because of rising price of fossil fuels, new sources of energy such as solar and wind power become more popular. These energies are cheaper and generate less pollution to the environment. Fuel cells are static converters that directly convert chemical energy of fuel into electrical energy, in form of DC. High efficiency, good reliability, low noise and pollution are important characteristics of fuel cells that make these sources suitable for various applications. The important applications of fuel cells are power plant applications, military, transportation, Combined Heat and power production (CHP) and etc. Among the existing fuel cells, proton exchange membrane fuel cells (PEMFCs) show a good potential in military applications and related application to Distributed Generations (DG). PEMFCs are very widely used and reasonable for transportation, plants and backup generation due to high density and fast operating speed. Dynamic modeling of fuel cells is a primary need for performance assessment, controller design and real-time studies. For dynamic modeling of fuel cells, there are three major platforms: 1 - White box modeling: These models are also known to mathematical models [1, 2]. 2 - Gray box modeling: these models were used for setting and diagnosis parameters and mass factors [3]. 3 - Black box modeling [4]. To provide a complete electrochemical model, electrochemical equations are used to model fuel cells for wide range of applications. This modeling approach requires some information that is not available for electrical engineers. A simple method for Simulation is using experimental data which are obtained through various tests. But in this method, the response of model is limited to the input data range. Some new methods of modeling uses artificial intelligence for modeling output graph. One of these methods is using artificial neural networks. Artificial neural network just uses input - output data that has obtained in multiple experiments and does not need to set all the parameters.

Mathematical models are based on mathematical equations, chemical, electrochemical and physical analysis. For such modeling Nernst equation, the activation, resistance and concentration losses and the transfer equations of mass and heat are generally used [1, 2]. Mathematical models are divided to One-dimensional models, two-dimensional and three dimensional models. One-dimensional models are not accurate enough for all kind of studies, especially studies of energy, chemical and etc. [5]. However, models with two and three dimensions have good answers for most of studies, but excessive complexity of these models, requires much memory and time for calculation that led to less usage of this [6,7]. In [8] static model based on artificial neural networks is proposed but the important parameters for modeling and training algorithm are not presented in this reference and effect of destructive signal was not examined. Also the suitable algorithm for modeling does not presented. In this paper, at first the modeling of PEMFC using Feedforward neural network and based on different training algorithms is presented and Outputs and various algorithms performance are compared to each other to determine the appropriate algorithm for simulation of a PEMFC. Then destructive signal is applied to the input of neural network and performance of network is evaluated. At the end the effect of destructive signal on neural network output is reduced by defining an adaptive filter and performance of different algorithms are evaluated.

II. PEMFC

Fuel cells with polymer electrolyte are one of the five different kinds of fuel cells that are attended for plant and distributed generation uses due to high power density (1400 watt.lit^{-1}) and are considered as an alternative for internal combustion engines. An electrolyte is a substance that is decomposed to positive and negative ions in the presence of water and the obtained soluble is electrically conductive. But the electrolyte that is used in PEMFC is a membrane organic matter with ion-exchange properties that made of sulfonic acid polymer or polymer impregnated with fluorine. In fact, in this cell type electrolyte is a kind of plastic that composed of polymer molecules and is named membrane. Polymer electrolyte membrane according to the characteristics of electrolytes is an unusual electrolyte. Because in presence of water that is absorbed quickly by the membrane, Negative ions kept in the electrolyte structure and only free positive ions move along the membrane for transferring positive charge. For this reason polymer membrane is also called proton exchange membrane. Hydrogen ion transportation across the membrane from anode to cathode forms polymer fuel cells basis. Polymer electrolytes are different, but Nafion TM is used in most of them. The thickness of this kind of membrane is 50 to 175 microns. Since the membrane is made of an organic material, the proton exchanger polymer is not able to conduct electrons. This feature makes the membrane non-conductive electrically. Since electrons cannot move through membrane, an external connector is used in order to move them from one side of cell to another. The membrane has High ionic conductivity; very low gas permeability and chemical and thermal stability in cell function temperature. Also in terms of mechanical properties is solid and is not sensitive to moisture (Fig.1).

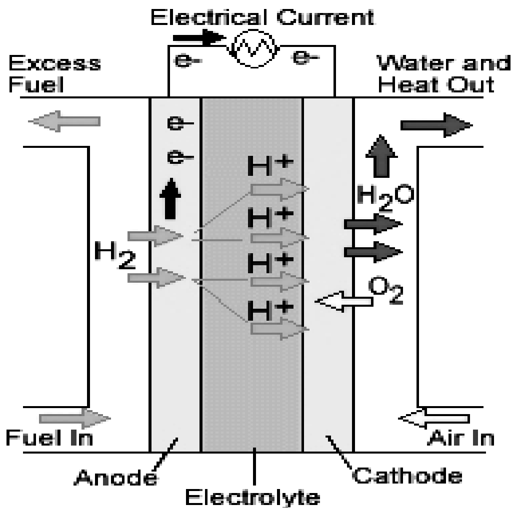

Figure 1. PEMFC Structure

Since polymer fuel cells have low function temperature, catalyst is used in their construction. The best catalysts are platinum and other neutral metals, which a very minor amount of them is deposited on porous carbon. The presence of Porosity in electrolyte makes penetration of reactants so easy. The basic point for producing electricity current or flow of electrons by fuel cell is wide dispersion of the catalyst on the electrode surface. Both carbon and platinum conduct electrons, so the produced electrons can leave electrode easily [9].

III. ARTIFICAL NEURAL NETWORK (ANN)

A neural network (NN), in the case of artificial neurons called artificial neural network (ANN) or simulated neural network (SNN), is an interconnected group of natural or artificial neurons that uses a mathematical or computational model for information processing based on a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network. In more practical terms neural networks are non-linear statistical data modeling or decision making tools. They can be used to model complex relationships between inputs and outputs or to find patterns in data. In these networks, design of data structures is done by help of programming knowledge and can act as neurons. It is said nodes to this data structure. Then a network will be created between the nodes and this network trains with a training algorithm. In this neural network or memory, nodes have two modes; active mode (clear or 1) or disable (Off or 0) and each edge (synapses or connections between nodes) has a weight. Edges with positive weight, activate next inactive nodes and edges have negative weight, disabled or inhibited the next node (if was activated) [10].

A. ANN Structure

Components of a neural network are layers and weights. Network behavior also depends on the relationship between components. In general, there are three kinds of neurons layers in neural networks:

- *Input layer*: accepts raw data that is fed to the network.
- *Hidden layers*: the inputs and the related weights between input and hidden layers determine the performance of hidden layers. Actually what specifies hidden layers activation time is weights between input units and hidden layers.
- *Output layer*: the performance of output units depends on hidden unit activities and weights between hidden and output units.

There are single-layer and multi-layer networks, single-layer organization in which all units are connected to just one layer is more popular and has more computing potential than multi-layer organization. In multilayer organization units are numbered by layers (rather than pursuing national numbers). Both layers of a network are associated by weights. There are different kinds of connection in neural network:

- *Feedforward*: Most links are classified in this group, in which signals flows in one direction. There is no feedback loop from input to output. Output of Each layer has no effect on the same layer.
- *Backward*: there is a feedback which transfers data from above layer nodes to bottom layer nodes.
- *Accessories*: output of nodes in each layer, is used as nodes input of the same layer [11].

B. Feedforward Neural Networks

Feedforward networks often have one or more hidden layers of sigmoid neurons and they use a linear final layer. Presence of several layers of neurons with a nonlinear transfer function will allow the network to learn linear and non-linear relationship between input and output. Linear output layer allow to the network that have the desired output in any area.

C. Feedforward Neural Networks Training

The important feature of any neural network is its ability to learn from environment and enhance its performance. A neural network can find a better understanding of the environment by repeating training. . In other words after each iteration, knowledge of network from environment can be improved. The process training requires some examples of expected behavior of the network that include input and network object. In this article this pair is current-voltage pair that is expected to be obtained at the fuel cell output. During training process, weights and bias of Feedforward network must be set to minimum network performance. In the Following training algorithms of Feedforward networks are examined. All these functions use performance function gradient to adjust the weight and bias. The training algorithm can be divided into two general categories:

1) Heuristic techniques: are used in first category. Exploration in these algorithms is based on standard descent performance analysis algorithm. These methods include: variable learning rate, resilient backpropagation, Gradient descent with momentum.

2) The second category uses standard numerical optimization techniques. These techniques include the following three categories:

- Conjugate gradient technique including: Fletcher-Reeves, Polak-Ribiere, Gradient Descent, Powell-Beale, Scaled conjugate.
- Quasi-Newton: the two algorithms one step secant, BFGS Quasi-Newton using this technique to work.
- The method of Levenberg-Marquardt and Bayesian Regulation [11].

In the following a fuel cell simulation is performed using these algorithms and the results are examined to determine the appropriate algorithm to simulate a fuel cell.

IV. SIMULATION

In this section, the neural networks performance and training algorithms in a PEMFC simulation are examined. For training the neural network the voltage and current corresponding to the experimental test are used. In this test temperature and pressure are kept constant. While the current drawn from the fuel cell increases from zero to 1.2 ampere, output voltage is measured. For simulation all conditions are the same for all networks and so we can compare results.

A. Algorithms Performance Evaluation

Using SCC, MSE, ME indices, errors of training algorithms in simulation are evaluated. These indices are defined as follows.

$$ME = Max\left|V_{act,i} - V_{pre,i}\right| \quad i = 1,2,\ldots,T \tag{1}$$

$$MSE = \frac{1}{T}\sum_{i=1}^{T}(V_{act,i} - V_{Pre,i})^2 \tag{2}$$

$$SCC = \frac{SS_{err}}{SS_{tot}} \tag{3}$$

$$SS_{err} = \sum_{i=1}^{T}(V_{act,i} - V_{Pre,i})^2 \tag{4}$$

$$SS_{tot} = \sum_{i=1}^{T}(V_{act,i} - \bar{V}_{act})^2 \tag{5}$$

$$\bar{V}_{act} = \frac{1}{T}\sum_{i=1}^{T}V_{act,i} \tag{6}$$

In the above equations $V_{act,i}$ is real value of output voltage in the i-th point and $V_{pre,i}$ is the model output and T determines the number of points of test data. MSE, ME and SCC that are closer to zero determine better ability to predict output voltage. In addition to these parameters time of training is proposed as an indicator of performance evaluation and shorter training time is more desired.

For analysis of different training algorithms based on these four parameters, and selecting the best algorithm we have to Multi Attribute Decision Making (MADM) methods. Actually we want to consider four criteria; (ME, MSE, SCC, Time) and twelve alternatives and show the best algorithm by MADM method. There are different methods for solving a MADM problem. In this paper Entropy Ranking Technique (ERT) and Simple Additive Weighting (SAW) has been used. A brief review on ERT method is discussed in the following.

B. Entropy Method

Entropy is a criterion in information theory that explains the uncertainly in a discrete distribution function (P_i). This uncertainly could be formulated as follows [12]:

$$E = -K\sum_{i=1}^{m}[P_i - LnP_i] \qquad 0 \le E \le 1 \tag{7}$$

In which: K is a positive constant.

The entropy technique could be used to evaluate the criteria in a MADM model. Suppose a decision matrix, D, as follows:

$$D = \begin{array}{c} \scriptstyle alternative\,\#1 \\ \\ \scriptstyle alternative\,\#m \end{array} \begin{bmatrix} \chi_{11} & \cdots\cdots\cdots & \chi_{1n} \\ \cdots & \cdots\cdots\cdots & \cdots \\ \chi_{m1} & \cdots\cdots\cdots & \chi_{mn} \end{bmatrix}$$

where:

χ_{ij} = Performance of i-th alternative, regarding j-th criterion. Each element of decision matrix could be normalized as:

$$P_{ij} = \frac{\chi_{ij}}{\sum_{i=1}^{m}\chi_{ij}} \tag{8}$$

Then E_j could be calculated as:

$$E_j = -K\sum_{i=1}^{m}[P_{ij} - LnP_{ij}] \qquad \forall j \tag{9}$$

in which $k = (Ln\ m)^{-1}$.

Now deviation degree (dj) according to j-th criterion is:

$$d_j = 1 - E_j \tag{10}$$

Finally, the weigh for each criterion, W_j, will be calculated as:

$$W_j = \frac{d_j}{\sum_{j=1}^{n} d_j} \tag{11}$$

In this study we have 12 alternatives and 4 criteria. By using above equations W_j has been calculated for different criteria. Now, Simple Additive Weighting (SAW) according to i-th alternative (algorithm) is as (12). Lower amount of SAW shows better performance of algorithm.

$$\mathrm{SAW}_i = \sum W_j \times \chi_{ij} \tag{12}$$

Table 1 provides results of PEMFC simulation with different neural network training algorithms for Feedforward network.
As it is clear, from point of training time gradient based methods have acceptable performance and the Polak-Ribeire and Fletcher-Reeves methods have shortest training time. Among all methods based on exploration techniques, the Resilient backpropagation algorithm has the best performance from both training time and accuracy of simulation. In quasi-newton based methods, BFGS quasi-Newton and One step secant`s performance are not acceptable. Levenberg-Marquardt and Bayesian Regulation methods, both in terms of accuracy have acceptable performance but the Levenberg-Marquardt method need less time to train and is more reasonable method. Based on SAW parameter the ranking of different algorithms are determined and shows that Polak-Ribiere has the best performance and one step secant has the worst performance.

C. *PEMFC Simulation in presence of Destructive Signal*

In some cases we need to train networks online with data obtained from experiments and predict the behavior of fuel cells in electrical network simultaneously. The problem occurs when we make mistake in transmitting data signal between measurement instruments and PC which, we use this data to train networks. Actually presence of noise and destructive signal has effect on network performance. In this situation the result of simulation is inappropriate. Therefore it is necessary to examine effect of destructive signal on fuel cell output. For assessing the performance of various neural network models, random destructive signal added to input data. This data are random variable. According to the previous section, the five training algorithms that had the best performance and function are selected and their performance evaluated in presence of destructive signal. These algorithms are: Polak-Ribiere, Fletcher-Reeves, Levenberg-Marquardt, Bayesian Regulation and Resilient Backpropagation. Neural network output in presence of this signal have been shown in Figures 2 to 6. In this figures the blue curve shows experimental data output, the red curve shows neural network output regardless of destructive signal and the green curve shows the network output with regarding destructive.
As it is clear, destructive signal has impact on ANN performance. Although the effect of this signal on the output, is various in different algorithms, but is high in all of them and thus the results are not appropriate. Therefore, a method for improving network performance in the presence of destructive signal is necessary. The next section describes an example of such method.

Neural network output in presence of this signal have been shown in Figures 2 to 6. In this figures the blue curve shows experimental data output, the red curve shows neural network output regardless of destructive signal and the green curve shows the network output with regarding destructive.
As it is clear, destructive signal has impact on ANN performance. Although the effect of this signal on the output, is various in different algorithms, but is high in all of them and thus the results are not appropriate. Therefore, a method for improving network performance in the presence of destructive signal is necessary. The next section describes an example of such method.

TABLE I. RESULT OF SIMULATION

		Criteria / Alternatives	Time	ME	MSE	SCC	SWA	Ranking
Heuristic techniques		variable learning rate	13.8851	0.0498	5.6353e-004	0.0420	6.7139	11
		Resilient backpropagation	7.1360	0.0197	0.0197	0.0079	3.4524	5
		Gradient descent with momentum	7.0230	0.2266	0.0203	0.5070	3.4972	7
Numerical optimization techniques	Conjugate Gradient	Gradient Descent	7.1306	0.1082	0.0093	0.3078	3.5020	8
		Fletcher-Reeves	1.4437	0.0701	8.4179e-004	0.0646	0.7157	2
		Polak-Ribiere	1.4159	0.0764	9.7584e-004	0.0726	0.7043	1
		Powell-Beale	7.1360	0.0497	4.7816e-004	0.0391	3.4563	6
		Scaled conjugate gradient	10.0581	0.0177	8.9637e-005	0.0080	4.8576	10
	Quasi-Newton	BFGS	7.5714	0.0170	6.5235e-005	0.0061	3.6572	9
		One step secant	19.2420	0.0669	7.7992e-004	0.0616	9.3043	12
		Levenberg-Marquardt	2.1779	0.0168	6.0893e-005	0.0057	1.0542	3
		Bayesian Regulation	4.2975	0.0165	6.1232e-005	0.0057	2.0771	4
Weigh of each criterion (Wj)			0.1421	0.2385	0.1368	0.4826	-	-

D. Reducing the effect of destructive signal on neural network output using adaptive filter

As it was discussed in the previous section Simulation of a fuel cell in the presence of destructive signal shows bad effect on the output and it is necessary to lessen this effect. For this aim an adaptive filter is designed for Feedforward neural network. This filter uses real data of voltage and current of fuel cells to predict interference of destructive signals on these data, and then subtracts predictive interference value from data that is used in neural network training; Hereby reduces the effect of destructive signal on the network. Now, to evaluate the effect of filter on the output, the previous neural network, is implemented with the filter and the results are given in the figures 7 to 11. In this figures the blue curve shows experimental data output and red curves shows the neural network output with adaptive filter.

Comparing the new simulation results (Figures 7 to 11) with the previous ones (Figures 2 to 6) indicates that with all training algorithms, adaptive filter performance is quite good and reduces the effect of destructive signal on ANN performance.

V. CONCLUSION

As it was stated, in order to evaluate the performance of a PEMFC, smart method (ANN) can be used. Feedforward network with several layers of neurons and a nonlinear transfer function will allow the network to learn linear and non-linear relationship between input and output, so it is an appropriate way for modeling PEMFC. There are several algorithms for training a neural network. In this paper different training algorithms are compared with training time and accuracy by MADM methods. Polak-Ribiere has the best performance and then Fletcher-Reeves, Levenberg-Marquardt, Bayesian-Regulation and Resilient-Backpropagation have shown tolerable performance. Furthermore, by applying a destructive signal to the neural network input for the five training algorithms that had the best performance it was perceived that destructive signal has bad effect on the neural network output. In order to overcome this problem and reduce the effect of destructive signal, an appropriate adaptive filter for ANN was proposed. Simulation results show the positive impact of the proposed adaptive filter on all five networks, and a great reduction in effect of destructive signal on the network output.

REFERENCES

[1] M.H. Nehrir, and C. Wang, “Modeling and control of fuel cells distributed generation applications”, John Wiley & Sons, 2009.

[2] R. T. Jagaduri, and G.Radman, “Modeling and control of distributed generation systems including PEM fuel cell and gas turbine”, Electr. Power Syst. Res., vol. 77, pp. 83–92, 2007.

[3] H. Schichlein, A.C. Müller, M. Voigts, A. Krügel, and E. Ivers-Tiffée, “Deconvolution of electrochemical impedance spectra for the identification of electrode reaction mechanisms in solid oxide fuel cells”, Applied Electrochemistry, vol. 32, pp. 875-882, 2002.

[4] X. Kong, W. Yeau, and A.M Khambadkone, “ANN modeling of nonlinear subsystem of a PEMFC stack for dynamic and steady state operation”, 32nd annual Conf. on IEEE Industrial Electronics, 2006.

[5] G. Hu, J. Fan, S. Chen, Y. Liu, and K. Cen, “Three-dimensional numerical analysis of proton exchange membrane fuel cells (PEMFCs) with conventional and interdigitated flow fields”, Power Sources, vol. 136, pp. 1–9, 2004.

[6] N. Akhtar, S. P.Decent, D. Loghin, and K. Kendall, “A three-dimensional numerical model of a single-chamber solid oxide fuel cell”, Hydrogen Energy, vol. 34, pp.8645–8663, 2009.

[7] J. Li, G.Y. Cao, X.J. Zhu, and H.Y. Tu, “Two-dimensional dynamic simulation of a direct internal reforming solid oxide fuel cell”, Power Sources, vol. 171, pp. 585–600, 2007.

[8] Z.-D. Zhong, X.-J. Zhu, and G.-Y. Cao, “Modeling a PEMFC by a support vector machine”, Power Sources, vol. 160, pp. 293–298, 2006.

[9] Gyu-Yeong Choe, Jong-Soo Kim, Hyun-Soo Kang, Byoung-Kuk Lee,Won-Yong Lee, “Proton Exchange Membrane Fuel Cell (PEMFC) Modeling for High Efficiency Fuel Cell Balance of Plant(BOP)”, Proceeding of International Conference on Electrical Machines and Systems, Oct. 8~11, Seoul, Korea, 2007.

[10] J. Hertz, A. Krogh, and R.G. Palmer, “Introduction to the theory of Neural Computation”, Addison-Wesley, Reading, Mass, 1991.

[11] Seyed Mostafa Kia, “Neural Network in MATLAB,” Kia Rayaneh propagation, 2000.

[12] R. Madlener, S. Stagl, “Sustainability Guided Promotion Of Renewable Electricity Generation”, Elsevier, 2005.

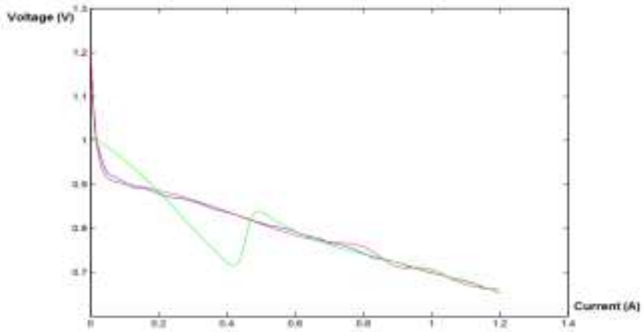

Figure 2. ANN output with Resilient backpropogation training algorithm

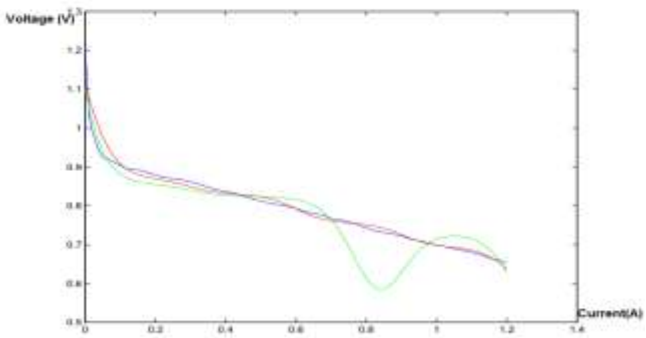

Figure 3. ANN output with Fletcher-Reeves training algorithm

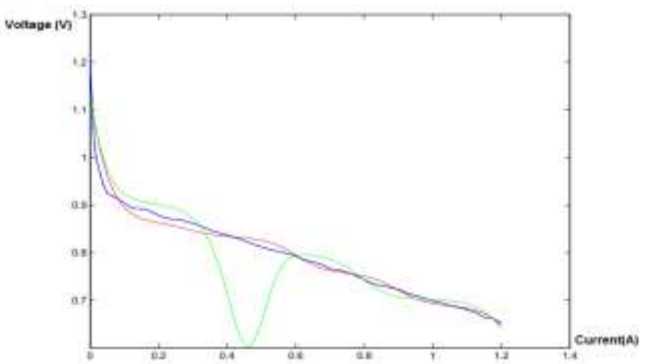

Figure 4. ANN output with Polak-Ribiere training algorithm

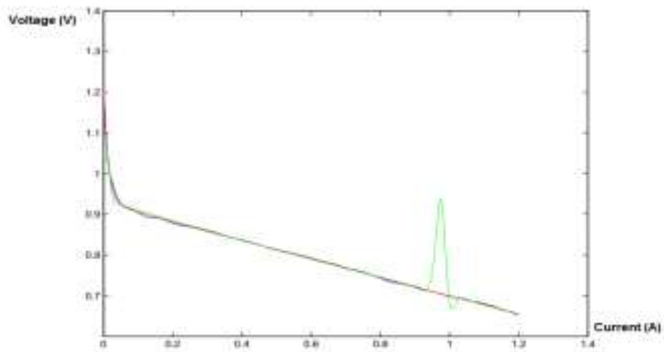

Figure 5. ANN output with Levenberg-Marquardt training algorithm

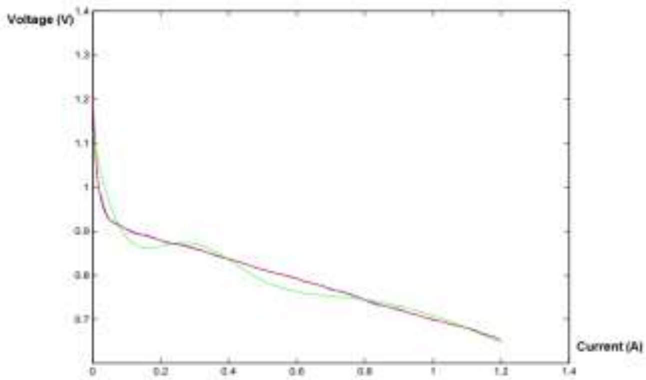

Figure 6. ANN output with Bayesian regularization training algorithm

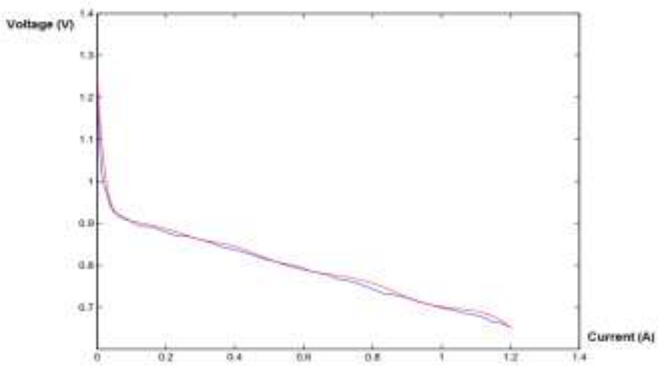

Figure 7. ANN output with Resilient backpropogation training algorithm and applying adaptive filter

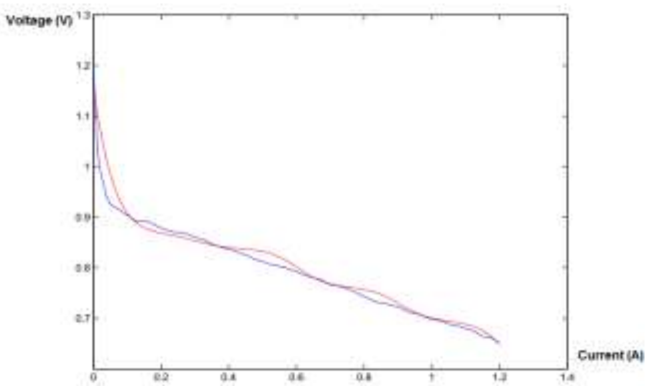

Figure 8. ANN output with Fletcher-Reeves training algorithm and applying adaptive filter

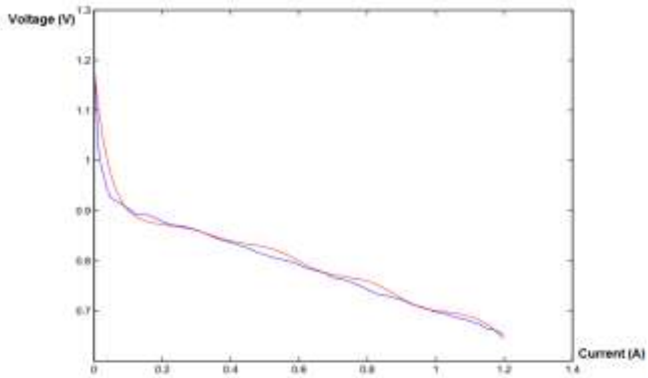

Figure 9. ANN output with Polak-Ribiere training algorithm and applying adaptive filter

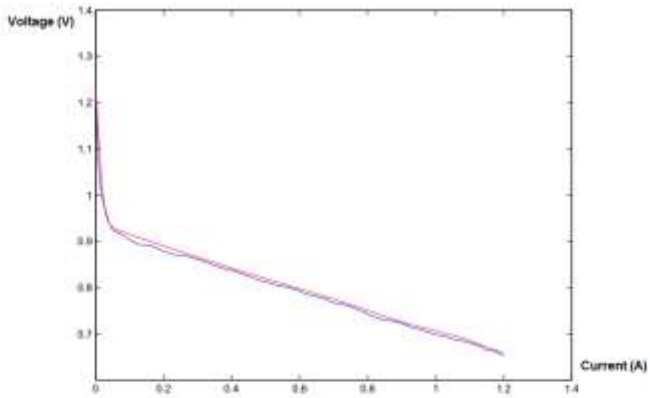

Figure 10. ANN output with Levenberg-Marquardt training algorithm and applying adaptive filter

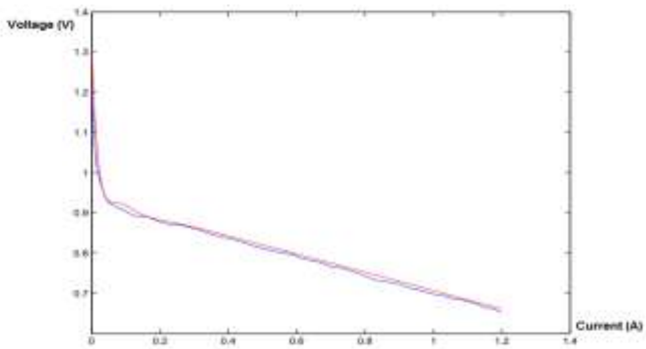

Figure 11. ANN output with Bayesian regularization training algorithm and applying adaptive filter