Artificial Neural Networks Application on PEMFC Predictive Control Systems

Juan José García, *C.S. candidate, Espol*

***Abstract*—Proton Exchange Membrane Fuel Cells (PEMFC) operation fluctuates rapidly in time. To control such fluctuations, an intelligent system is required to both model and predict the PEMFC voltage. This paper explains how to model and train an artificial neural network, to predict the voltage of a PEMFC, 0.1 seconds ahead of time.**

***Index Terms*—Neural networks, Artificial neural networks, Prediction, PEMFC**

I. Introduction

PEMFC voltage follows a nonlinear models that fluctuate rapidly with changes in load current, temperature, hydrogen pressure, oxygen pressure.

In [1], for example, voltage of a 5KW PEMFC fluctuates between 1V and 1.3V during operation, without any perceivable pattern. When a system is required to predict non apparent patterns, neural networks are a solution.

Neural networks are capable of replicating nonlinear models in a less computationally taxing manner; thus making them capable of predicting voltage fluctuations in time to respond properly.

[1,2,3,4] explore the use of neural networks in the operation control of PEMFC, nevertheless no paper was found that considered current and pressure in voltage prediction.

The objective of this research is then: to develop a model capable of predicting PEMFC voltage, considering both current and hydrogen pressure.

II. Related works

ANN have had many applications in the control of PEMFC systems, mostly as: voltage filters, voltage predictors and PEMFC modeling.

[1] and [2] use ANN to model the operation of a PEMFC. The neural network replicates the mathematical model in [4].

[1], [2] and [3] use Neural Network to control the PEMFC operation. In [1] Neural networks are used to filter voltage fluctuations caused by temperature.

In [2] Neural networks are used to control the active power given to the load, considering the pressure of hydrogen. [3] provides a NNARX model to predict output voltage considering the load current.

All of the previous neural networks have been trained using the model in [4].

A. Abbreviations and Acronyms

PEMFC: Proton exchange membrane fuel cell
ANN: Artificial neural network
NNARX: Neural network auto-reggresive model with eXogenous inputs.

III. Mathematical model

PEMFC current is generated through an electrochemical process, this is described by figure 1, indicated in [1].

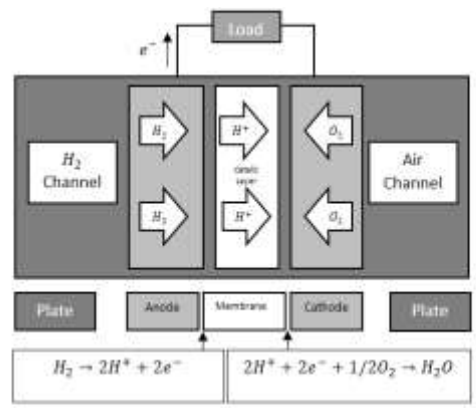

Fig. 1. Diagram of the PEMFC current generation process.

A. Equations

The output voltage of the fuel cell has been modeled by [4], and is the model we will consider in this research.

The output voltage of the fuel cell (V_{Fc}) can be modeled as follows:

$$\mathrm{V_{Fc}} = \mathrm{E_{Nenst}} - \mathrm{V_{act}} - \mathrm{V_{ohmic}} - \mathrm{V_{con}}$$

$$\mathrm{E_{Nenst}} = 1.229 - 0.85 \times 10^{-3}(T - 298.15) + 4.31 \times 10^{-5}T[\ln(P_{H2}) + 1/2\ln(P_{o2})]$$

$$\mathrm{V_{act}} = -[\zeta_1 + \zeta_2 T + \zeta_3 T ln(C_{o2}) + \zeta_4 T ln(i_{Fc})]$$

$$C_{o2} = \frac{P_{o2}}{5.08 \times 10^6 e^{-\frac{498}{T}}}$$

$$\mathrm{V_{ohmic}} = I_{Fc}(R_M + R_c)$$

$$RM = \frac{P_m l}{A}$$

$$P_m = \frac{181.6\left[1 + 0.03\left(\frac{I_{Fc}}{A}\right) + 0.062\left(\frac{T}{303}\right)^2\left(\frac{I_{Fc}}{A}\right)^{2.5}\right]}{\left[\psi - 0.634 - 3\left(\frac{I_{Fc}}{A}\right)\right] \times \exp\left[4.18\left(\frac{T-303}{T}\right)\right]}$$

$$V_{con} = -Bln\left(1 - \frac{J}{J_{max}}\right)$$

$$C_{H2} = 2(C_{O2})$$

IV. Units

Symbol	Unit	Symbol	Unit
$\mathrm{V_{Fc}}$	V	P_{H2}	atm
$\mathrm{E_{Nenst}}$	V	P_{o2}	atm
$\mathrm{V_{ohmic}}$	V	C_{o2}	mol/cm^3
$\mathrm{V_{con}}$	V	i_{Fc}	A
T	K	ζ_1	-
R_M	Ω	ζ_2	-
R_c	Ω	ζ_3	-
A	cm^2	ζ_4	-
ψ	-	l	cm
P_m	$\Omega.cm$	J	A/cm^2
B	v	J_{max}	A/cm^2
C_{H2}	mol/cm^3	-	-

V. Methodology

The development of ANN will consist of three parts: Modeling, feature extraction, and training/testing.

A. Modeling

The PEMFC output voltage is modeled using Matlab/Simulink. Figure 2 describes the block diagram.

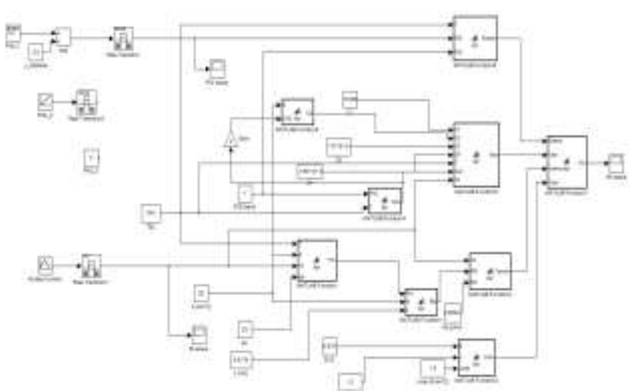

Fig. 2. Block diagram for PEMFC voltage.

B. Feature Extraction

The feature vector of our ANN consists of: Load current (i_{Fc}), Hydrogen pressure (P_{H2}) and Fuel cell voltage ($\mathrm{V_{Fc}}$). Using the feature extraction of [3] as guideline, the model was evaluated with a I_{Fc} given by figure 3, and a P_{H2} given by figure 4. Figure 5 reflects the PEMFC voltage with the given input.

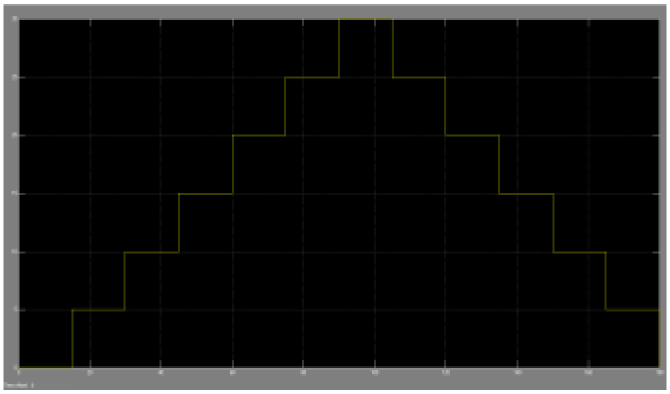

Fig. 3. Load current changing every 15 seconds

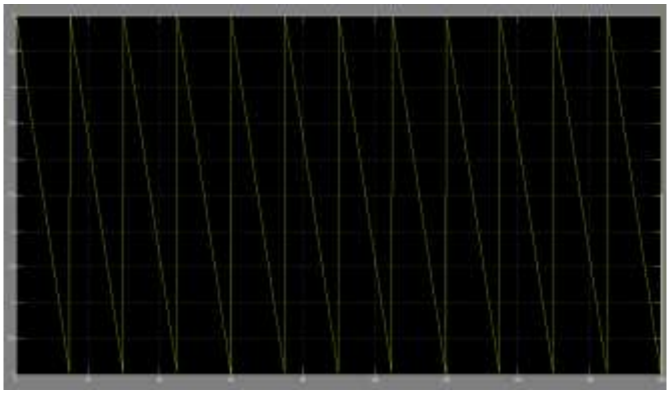

Fig. 4. Hydrogen pressure linearly changing over a period of 15 seconds.

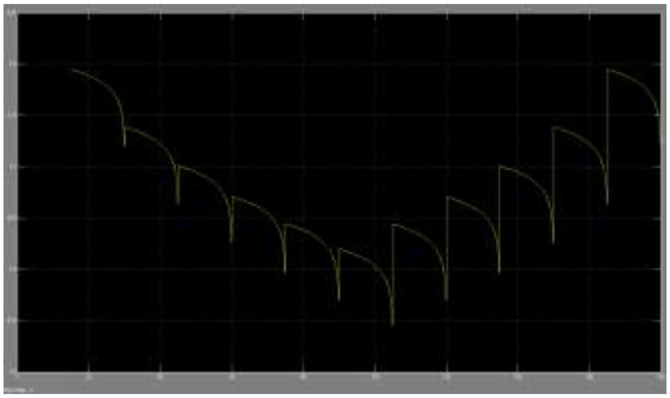

Fig. 5. Output voltage of the PEMFC model.

C. Training/Testing

To train the neural network, all the possible pressures for the given load currents are considered. The upper and lower bounds of possible hydrogen pressures are given by [2]. The values of the constants in the model are given by [1].

The training sample will consist of the feature vector that is within the first 115 seconds; the test sample will be the feature vector among the last 65 seconds. A total of 1650 measurements were taken, separated by a 0.1 second time frame. The predicted voltage was compared with the real voltage obtained 0.1 seconds after. This means that the ANN will train itself to predict the voltage 0.1 seconds ahead of the input.

The R programming language was used to train and evaluate the ANN. The package 'nnet' was used to create an ANN with two inputs (i_{Fc}, P_{H2}) and one output (V_{Fc}); 200 neurons in one hidden layer; and a training sample size of 1000 samples.

The function 'predict' was used to evaluate the ANN with 650 test samples. The error yielded a median of 0.012 % and a mean of 0.077%. The maximum error observed is 12.08% at the prediction of sudden changes in load current; the ANN predicts a voltage approximately 0.12 [V] less than the modeled voltage at the instant of change.

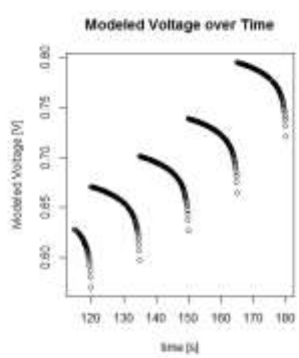

Fig. 7. Real voltage over time

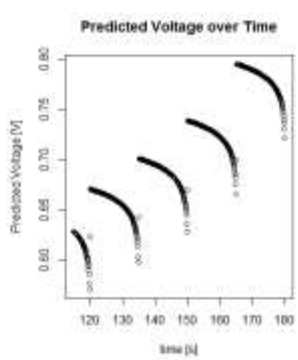

Fig. 8. Predicted voltage over time

Figure 7 and Figure 8 are voltage graphs. The former refers to the modeled voltage, the latter to the predicted voltage. It is apparent that both are very similar.

D. Statistical Analysis

A T-Student statistical test was performed with the predicted voltage and the real voltage. Considering a 95% confidence interval, a t value of 0.146, and a p value of 0.8842.
We can conclude that the means of both datasets are equal in 88.42% of the cases.

Normal distribution and homogeneous variances were assumed.

VI. Conclusion

ANN are a promising solution to control the output voltage of a PEMFC. This is because of the 0.012 % error between predicted and real voltage, and how fast input data can be processed to predict the voltage.

This model is also capable of learning further voltage patterns, so the PEMFC will be capable to adapt to different environments, should the voltage predicted is inaccurate.

VII. Future work

This research aims to lay a foundation when training ANNs to predict the output voltage of a PEMFC. With that in mind, the next step would be to include both temperature and relative oxygen pressure in the feature vector of the ANN.

If the readers intends to do so, I strongly encourage that the ANN uses the weights provided here as starting weights. This way, adding the variables mentioned won't hinder convergence time when training.
To obtain the data for both the temperature and the oxygen pressure it is recommended that they are measured from a built PEMFC.

Finally, analyzing the instruction count of a simple matrix multiplication algorithm, and a 16MHz Arduino clock speed; a 204 neuron ANN will predict voltage in less than 0.8 mili seconds (by evaluating the equation: $Execution\ time = Instruction\ count * CPI * \frac{1}{Clock\ rate}$). This means that data obtained should be separated by a larger timeframe. This is important when designing a system that controls the hydrogen pressure valve.

Acknowledgment

To Ricardo Macias for helping model the PEMFC in Matlab, and to Karen Guarco for deducing the equation of hydrogen concentration.

REFERENCES

[1] M. Sedighizadeh, M. Rezaei, and V. Najmi, “A Predictive control based on Neural Network for Proton Exchange Membrane Fuel Cell,” World Acad. Sci. Eng. Technol., vol. 5, pp. 395–399, 2011.

[2] K. Mammar and A. Chaker, “Neural Network-Based Modeling of PEM fuel cell and Controller Synthesis of a stand-alone system for residential application,” Int. J. Comput. Sci. Issues IJCSI, vol. 9, pp. 244–253, 2012.

[3] S.-J. Cheng, T.-J. Chang, K.-H. Tan, and S.-L. Kuo, “Nonlinear Modeling of the PEMFC Based On NNARX Approach,” World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inf. Eng., vol. 9, no. 5, pp. 1124–1128, 2015.

[4] J. Kim, “Modeling of proton exchange membrane fuel cell performance with anempirical equation,” J. Electrochem. Soc., no. 142, pp. 2670–2674, 1995.

[5] Michael A. Nielsen, “Neural Networks and Deep Learning”, Determination Press, 2015