Mechanical behaviour of a fuel cell stack under vibrating conditions linked to aircraft applications part II: Three-dimensional modelling

V Rouss, D Candusso, W Charon

• V. Rouss, D. Candusso, W. Charon (novembre 2008). *Mechanical behaviour of a fuel cell stack under vibrating solicitations linked to aircraft applications. Part II: Three-dimensional modelling*. Int. J. Hydrogen Energy. Vol. 33, n°21, pp. 6281-6288. Ed. Elsevier.

Mechanical behaviour of a fuel cell stack under vibrating conditions linked to aircraft applications Part II: Three-dimensional modelling

Vicky Rouss [a], Denis Candusso [c], Willy Charon [a]

[a] M3M, University of Technology Belfort - Montbéliard
[c] INRETS, The French National Institute for Transport and Safety Research

FCLAB, Rue Thierry Mieg, F 90010 BELFORT Cedex, FRANCE

Corresponding author:
Tel: 0033 3 84 58 31 36
Fax: 0033 3 84 58 31 46
Email addresses: vicky.rouss-01@utbm.fr (V. Rouss)

Other email adresses:
willy.charon@utbm.fr
denis.candusso@inrets.fr

Abstract
The implementation of fuel cells (FC) in transportation systems such as airplanes requires better understanding of their mechanical behaviour in vibrating environment. To this end, a FC stack was tested on a vibrating platform for all three orthogonal axes. The experimental procedure is described in the first part of the paper.
This second part of the paper demonstrates how the experimental data collected can be used to create a three-dimensional, multi-input and multi-output model based on the Artificial Neural Network (ANN) approach. Indeed FCs are nonlinear mechanical systems, difficult to be physically modelled. The ANN methodology which depends strictly on raw data is a particularly interesting alternative solution to model FCs, for example, for monitoring purpose. The ANN model is described along with the training, pruning and validation stages. The results are exposed and commented.

Keywords: PEM Fuel Cell; Vibration; Neural network; Modelling; Nonlinear; Complex mechanical system

1. Introduction

In order to implement Fuel Cells (FC) in transportation systems and in particular in aircrafts, the mechanical behaviour of FCs and the effects of mechanical loads on their

structures have to be accurately determined and mastered. A behavioural representation of the investigated FC stack can also be very helpful to reach this aim.
In the literature, a large number of FC dynamic models are proposed. These models can be divided into analytical models and experimental models.
Analytical models are based on electrochemical, thermodynamic, fluid dynamic relationships. The descriptions are made using phenomenological equations such as the Butler-Volmer equation for the cell voltage or the Stefan-Maxwell equation for the gas-phase transport. These models are often very detailed and complex. The usual drawbacks are the development duration and the difficulty of the validation due to the high number of parameters that have to be estimated. Analytical models provide the foundation for exploring mechanisms and understanding physical phenomena in the PEM fuel cells. In spite of advances in modelling, several of these physical methods are not accurate enough because of the complex nature of the FC. These models require a good knowledge of the physicochemical process parameters. In most cases, these parameters are difficult to determine for an operating FC system [1].
The experimental methods exploit raw data and create a "black box" model which requires no preliminary knowledge about the system [2]. Several techniques can be implemented to create a black box model but an artificial neural network (ANN) model provides useful and reasonably accurate input-output relations because of its excellent multidimensional mapping capability. In the literature, many experimental models are proposed: an adaptive fuzzy identification model based on input-output sampled data [3], a dynamic recurrent neural network model for FC dynamic operating modes [4], a Q-Newton neural network model which estimates the voltage considering different operating conditions and delivered current [5], a Multi-Layer Perceptron (MLP) model that inputs pressures and temperatures at the stack and outputs voltage of the FC [6], a neural network model of the polymer electrolyte membrane fuel cell performance [7], a hybrid neural network model consisting of an ANN component and a physical component that takes into consideration the effect of Pt loading [8], a back-propagation feed-forward network and a radial basis function network to predict the cell voltage and to study the effect of Pt loading [9].
All of these studies were conceived to predict the FC electrical, thermal performances (cell voltage or cell power density vs. load current or current density) or to determine the optimal operating conditions for PEM fuel cells. In any case, they do not treat the mechanical aspect of the FC. However, ANNs can be used to model mechanical systems. From the existing models in the literature, we quote: an ANN classifier for condition monitoring of rotating mechanical systems [10], a radial basis function neural network model for hydraulic turbine generating unit system [11], an ANN with back-propagation algorithm for modelling a mechanical cooling system with variable cooling capacity [12], a back-propagation neural network that uses gradient descent learning algorithm to predict mechanical properties of hot rolled series carbon steel bars [13], an hybrid neural network based friction component model for powertrain dynamic analysis and controller design [14], Elman neural network for modelling non-lubrication mechanical systems that present a highly nonlinear flexible mechanical motion [15], a time-delay neural network to predict vertical wheel/rail force using vertical rail profile as input [16] and a genetic algorithm and a neural network model of micro sliding on cantilever quartz beam [17].
In an earlier study, a multi layer perceptron (MLP) neural network combined with a time regression input vector was proposed to model the mechanical behaviour of a PEM

fuel cell system in one axis direction [18]. In this paper, this approach is developed in order to create a three-dimensional model that enables to predict the FC mechanical behaviour in the three axes of excitations: X, Y and Z. In comparison with the solution that consists in using three different models for the three axis directions, a unique model is mandatory for predicting the coupling between spatial behaviours and is furthermore very useful both for reducing computation time and for providing an easier overview of the complete FC behaviour in the three axes referential. In fact, input and output correlations between different axes lead to reduce the number of the global neural network neurons. The specific experimental set-up and the collected results described in the first part are used to feed the model with data and to validate it as well.
The second part of the paper is organised as follows. Section 2 explains the neural network modelling approach. Section 3 illustrates and analyses the results. The paper is concluded with section 4.

2. Modelling Approach

The neural network approach for system modelling consists of several steps, as shown in Fig. 1. During the tests, data are collected. Then, the neural network structure is chosen: Hopfield, Elman networks, multi-layer perceptron (MLP) or radial basis networks. This stage also includes the selection of inputs and outputs, the choice of the number of layers and of the number of neurons in each neural network layer. Afterwards, the data collected during the tests are divided into three different sets: the first and second sets are used to estimate the model through training and pruning (estimate the model) and the third set is implemented to validate the model.

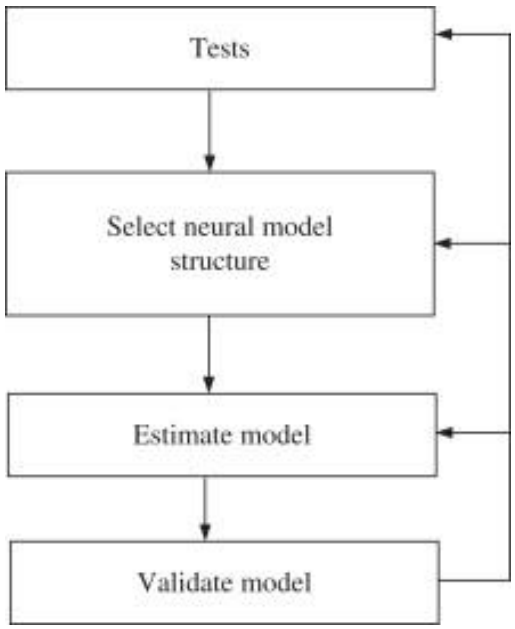

Fig. 1 - Steps of the modelling approach.

2.1. Tests

Before capturing data needed to set up the neural network, it is important to carry out tests to assess the general behaviour of the system, e.g the resonance frequencies and

nonlinearity degree [19]. As explained in the first part of the paper, the mechanical loads applied to the stack were derived from airplane requirements using a reference test curve, which corresponds to the maximum acceleration envelope for the vibration tests. The sinusoidal excitations of the test procedure are applied to each axis separately.

2.2. Selection of the neural model structure

The second step is the selection of the neural network structure. A multi-layer structure of feed-forward MLP is chosen here because of its capability of representing non-linear behaviours. It has one hidden layer, even if the number of layers may be gradually increased as greater flexibility is needed for more complex systems.
The network contains five outputs for predicting the measurements of the three-dimensional (3D) accelerometer on the top surface of a bipolar plate and of the two one dimensional (1D) accelerometers on the external left and right sides of the same bipolar plate.
The number of inputs has to be determined. The only actual input is the acceleration of the vibrating table. However, as shown see fig. 2, the neural network is fed with values of acceleration and measured outputs from previous time instants. This is known as time regression (NNARXM), which is essential in modelling non-linear behaviours and is a dominant factor in the computation time. It can be determined using the Lipschitz approach [20].

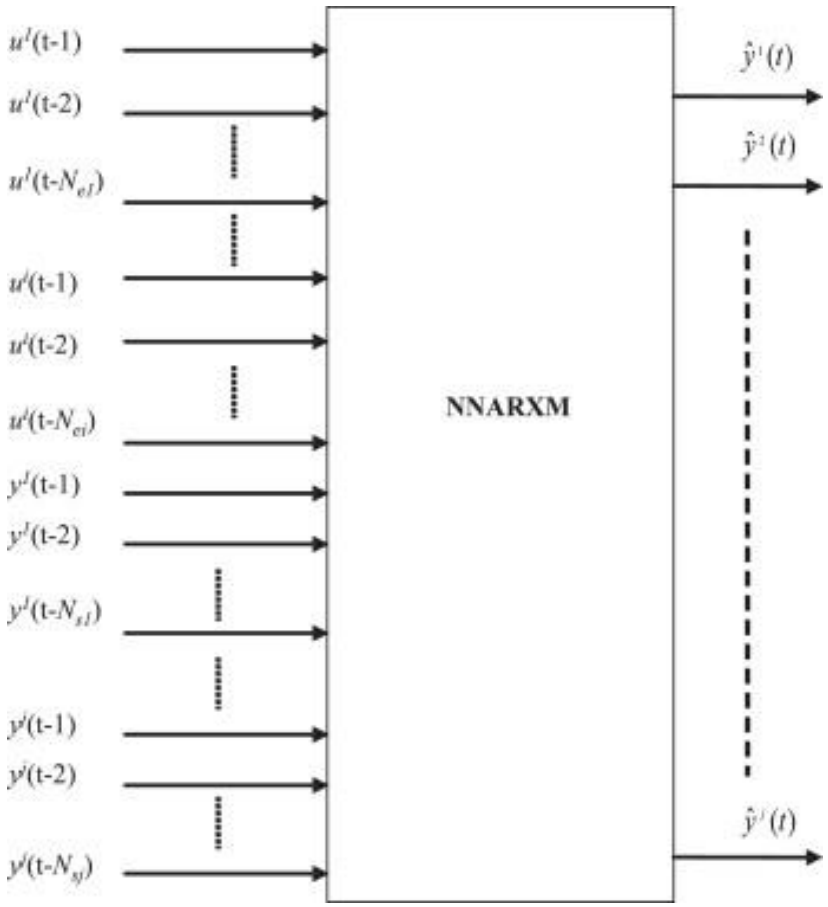

Fig. 2 - A neural network with NNARXM regression vector.

Now, a brief presentation of the procedure attached to this method is given.
Define the training set:
$Z_N = \{[U^1(t), Y^1(t)], t=1, ...N\}$ (1)
composed of:

- $U^1(t)$, a matrix of (4,*N*) dimensions computed by:
  $U^1(t) = [u^1(t); u^2(t) ; u^3(t); u^4(t)]$ (2)
  where $u^1(t)$; $u^2(t)$ *and* $u^3(t)$ are the imposed acceleration in the X, Y and Z axes direction, $u^4(t)$ is the sampling frequency corresponding to each point and N is the number of points used in the training set.
- $Y^1(t)$, a matrix of (5,*N*) dimension computed by:
  $Y^1(t) = [y^1(t); y^2(t); ...; y^5(t)]$ (3)
  where $y^1(t)$, $y^2(t)$, $y^3(t)$, $y^4(t)$ and $y^5(t)$ correspond to the five channels of the three measurement accelerometers (one 3D and two 1D) of the bipolar plate. The measurements of these channels are compared with the prediction of the neural network five outputs.

The neural network input is the time regression vector $\varphi(t)$, which is calculated as follows:
$\varphi(t)=[\varphi_1, \varphi_2, \varphi_3, \ldots, \varphi_z]=[u^1(t\text{-}1)\ u^1(t\text{-}2)\ldots u^1(t\text{-}N_{e1})\ldots\ u^4(t\text{-}1)\ u^4(t\text{-}2)\ldots u^4(t\text{-}N_{e4})\ y^1(t\text{-}1)\ y^1(t\text{-}2)\ \ldots\ y^1(t\text{-}N_{s1})\ y^2(t\text{-}1) y^2(t\text{-}2)\ldots y^2(t\text{-}Ns_2)\ldots y^5(t\text{-}1) y^5(t\text{-}2)\ldots y^5(t\text{-}Ns_5)]$ (4)
where:

- N_{ei} and N_{sj} are, respectively, the number of input (*i*) and output (*j*) time regressors, *i* varies from 1 to 4 and *j* varies from 1 to 5,
- *z* is the total number of regressors in time.

The values of time regressors depend on the degree of nonlinearity computed by means of the Lipschitz coefficient. For each different input $u^i(t)$ and output $y^j(t)$, the Lipschitz coefficient is calculated with the expression:

$$q_{ij} = \left| \frac{y^j(t_m) - y^j(t_k)}{\phi_i(t_m) - \phi_i(t_k)} \right| = \left| \frac{\partial y^j}{\partial \phi_i} \right|, (t_m \neq t_k) \quad (5)$$

The approach consists in choosing different delay couples N_{ei} and N_{sj}. For each couple, the Lipschitz coefficients are computed for all input $u^i(t)$ and output $y^j(t)$ combinations. The greatest p_{ij} coefficients are selected to calculate the criterion:

$$\bar{q}_{ij}^{\,n} = \left(\prod_{k=1}^{p_{ij}} \sqrt{n} q_{ij}^{\,n}(k)\right)^{1/p_{ij}} \quad (6)$$

where *n* equals $N_{ei} + N_{sj}$ for each combination. If *n*, the number of delays, is too small, the Lipschitz coefficient tends to infinity. From a certain value of *n*, the Lipschitz coefficient decreases slightly.
After the calculation of the N_{ei} and N_{sj} for all the inputs $u^i(t)$ and the outputs $y^j(t)$, *z*, the total number of time regressors is calculated by:
$z=N_{e1}+\ldots N_{e4}+N_{s1}+\ldots+N_{s5}$ (7)
The results of the calculations are:
N_{e1}=2, N_{e2}=2, N_{e3}=2, N_{e4}=1, N_{s1}=2, N_{s2}=2, N_{s3}=4, N_{s4}=2 and N_{s5}=2.
One hidden layer containing seven neurons is initially chosen. The number of neurons may be large, but it will be optimized later using specific algorithms. The non-linear activation functions [21] for the hidden neurons are a hyperbolic tangent type $f(x) = \tanh(x)$ with values between -1 and 1. For the output neuron, a linear activation function $f(x) = x$ is selected. This choice results from various tests carried out with

different activation functions on each layer.
After the choice of the neural network structure, it is trained according to a minimization criterion and then optimized by a pruning technique.

2.3. Estimate the model

This section is composed of two parts: training and pruning discussed respectively.

2.3.1 Training

The first set of the collected data Z_N (U^1, Y^1) is used for training the network. It is important that Z_N includes information concerning the global behaviour of the system like all the amplitude levels and frequencies of interest. If Z_N shows redundancy for an operating range and a lack of information for another operating range, the model accurately predicts in the first case but with difficulties in the second. During the study, a vector composed of 2000 points (N = 2000) is chosen. All the values of this vector are normalized between -1 and 1.
The model parameters (weights) are determined during the training. They constitute the model that will be able to give the best prediction of the real outputs of the system. The model that satisfies the minimal value of the following criterion is chosen:

$$V_N = \frac{1}{2N}\sum_{t=1}^{N}\left\{[Y(t) - \hat{Y}(t|\theta)]^T [Y(t) - \hat{Y}(t|\theta)]\right\} \quad (8)$$

where:

- $Y(t) = Y^1(t) = [y^1(t); y^2(t); ...; y^5(t)]$ is the vector of measured outputs in the training set,
- $\hat{Y}(t) = [\hat{y}^1(t|\theta); \hat{y}^2(t|\theta); ...; \hat{y}^5(t|\theta)]$ is the vector of the predicted outputs,
- $\theta = [W^2\ W^1]$, is the vector of weights to be defined.

In fact, there are two types of weights:

- W^1, containing the weights between the inputs and the hidden layer and the bias values for the neurons in the hidden layer (W^1=140),
- W^2, containing the weights between the hidden layer and the outputs and the bias for the outputs neurons (W^2=40).

This approach based on the minimization of V_N is called the *prediction error method* [22]. Among the different minimization methods which use this criterion, the MLP network combined with NNARXM regression vector uses the 'Levenberg-Marquardt' method [23]. It minimizes an approximation of the criterion V_N called $L^{(i)}(\theta)$ in a neighbourhood which is a sphere of radius $\delta^{(i)}$ centred around the current iteration $\theta^{(i)}$.

$$\theta^{(i+1)} = \arg\min_{\theta}(L^{(i)}(\theta)) \text{ in the neighbourhood } \left|\theta - \theta^{(i)}\right| \le \delta^{(i)} \quad (9)$$

The next iteration values are computed by the formula:

$$\theta^{(i+1)} = \theta^{(i)} + f^{(i)} \quad (10)$$

where $f^{(i)}$ is the search direction given by:

$$f^{(i)} = -[R(\theta^{(i)}) + \lambda^{(i)} I]^{-1} G(\theta^{(i)}) \quad (11)$$

where:

- $\lambda^{(i)}$ is a parameter that varies between 0 and infinity, whose determination is realized with the method of Fletcher [24].

- the gradient $G(\theta^{(i)}) = \left.\frac{dL^{(i)}(\theta)}{d\theta}\right|_{\theta=\theta^{(i)}} = \frac{1}{N}\sum_{t=1}^{n}\left\{[\psi(t,\theta^{(i)})]^T[Y(t) - \hat{Y}(t|\theta^{(i)})]\right\}$ (12)

  where:

  $$\psi(t,\theta^{(i)}) = \frac{d\hat{Y}(t|\theta^{(i)})}{d\theta} \quad (13)$$

- the hessian $R(\theta^{(i)}) = \left.\frac{d^2L^{(i)}(\theta)}{d\theta^2}\right|_{\theta=\theta^{(i)}} = \frac{1}{N}\sum_{t=1}^{n}\left\{[\psi(t,\theta^{(i)})]^T[\psi(t,\theta^{(i)})]\right\}$ (14)

2.3.2 Pruning

The principle of pruning is to initially start out with relatively large network architecture and then successively prune the network branches (weights) of one at a time until the optimal architecture is found. Stopping criterions other than the quadratic error V_N used in the training section are explored. Thus, the final prediction error *FPE* and the test error V_T are introduced in this section [25]. Both criteria provide information about the ability of the model to reliably predict outputs for unknown entries, i.e. the second set of data $Z_T(U^2, Y^2)$ which is not used in the training section.

- *Final prediction error*

The final prediction error is expressed in the formula:

$$FPE = \hat{V}_M = \frac{1}{2}\sigma_e^2(1 + \frac{p}{N}) \quad (15)$$

where p is the number of weights and σ_e^2 is the noise variance estimated as follows:

$$\hat{\sigma}_e^2 = 2\frac{N}{N+p}V_N(\hat{\theta}, Z^N) \quad (16)$$

- *Test error*

Test error V_T computation is based on a second data set $Z_T(U^2, Y^2)$ which is completely different from the first data set $Z_N(U^1, Y^1)$ used for training. During the study, a vector composed of 1500 points (T = 1500) is chosen. A value of V_T close to V_N means that the model obtained after the training is accurate. In fact, the test error is composed of two types of error:

- The bias error which appears when the optimised model is not found in the set of candidate models defined in the neural structure selection section. It happens when there are not enough neurons to model the system. This is called undertraining.
- The variance error which is due to an excessive number of neurons in the network. This increases the number of local minima and the variance of estimated weights. The model identifies not only the system but also the noise present in the vector Z_N. This case is called overtraining.

The opposing effects of these two types of errors lead to the fact that the error of test decreases at the beginning of the pruning then it increases. The curve of V_T passes by a minimum which is representative of the best model.

- *Architecture of the connections*

Another important criterion is the choice of the best neural architecture. In fact, the network should not be entirely connected. The principle is to initially start out with relatively large network architecture, train the network with 'Levenberg-Marquardt' method, evaluate V_T and FPE, calculate $\delta V^{(j)}$, $\delta V_M^{(j)}$ et $p^{(j)}$ as follows:

$$\delta V^{(i)} = V_N(\hat{\theta}^{(j)}, Z^N) - V_N(\hat{\theta}, Z^N) \quad (17)$$

where $\hat{\theta}^{(j)}$ is the reduced vector of weights,

$$\delta V_M^{(i)} = V_M(\hat{\theta}^{(j)}, Z^N) - V_M(\hat{\theta}, Z^N) \quad (18)$$

where V_M is the *FPE* error, $p^{(j)}$ is the number of remaining weights at the j^{th} iteration and j is the number of iteration.

At each iteration, the smallest value of $\delta V_M^{(j)}$ is chosen and the corresponding weight is eliminated. The remaining weights and neurons are determined by eliminating the hidden neurones that are no longer connected to any input. The procedure is repeated until the best architecture is found [26]. It corresponds to the one which yields a minimal *FPE*. Fig. 3 shows how pruning affects the initial network with seven neurons in the hidden layer. Only six neurons and 36 weights are necessary.

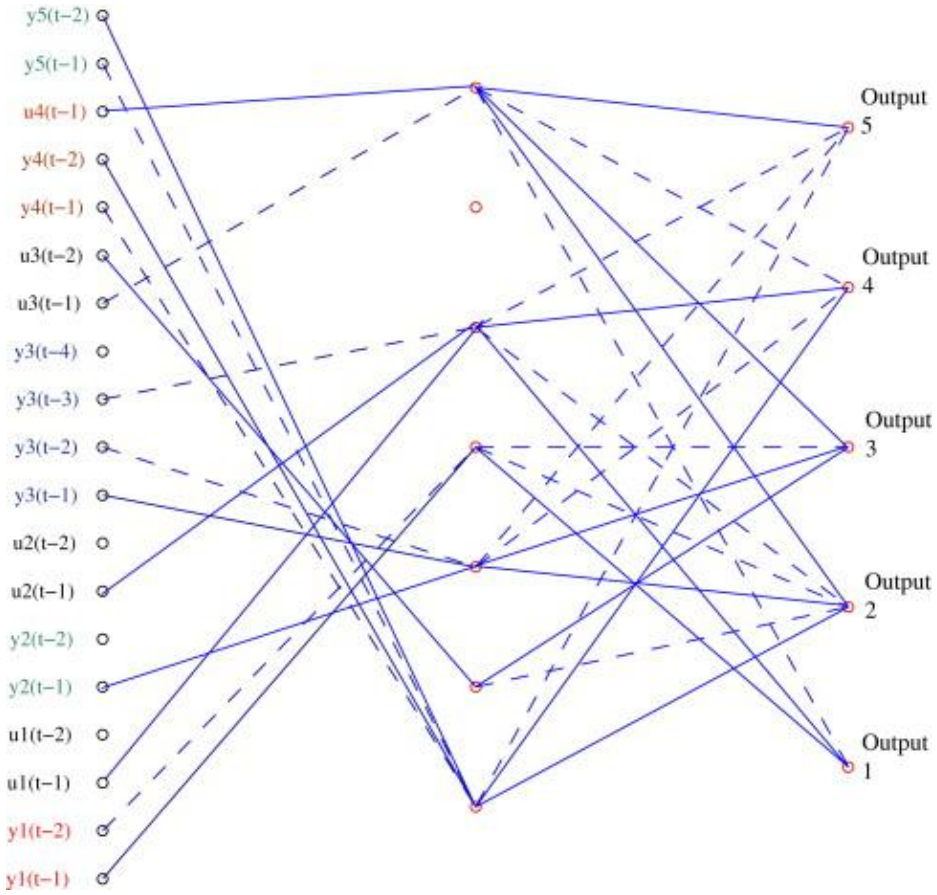

Fig. 3 - The final neural network after pruning.

2.4. Model Validation

Model validation is performed with a third data set $Z_V(U^3, Y^3)$ independent of Z_N and Z_T.

The validation approach is based on three analyses: visualization of the prediction, correlations between the predictions and the measurements and reliability of the predictions.

2.4.1. Visualisation of the prediction

The graphic representation, that contains respectively the measured outputs and the predictions calculated by the model, gives an idea of the accuracy of the predictions according to different modes (see Fig. 4).

2.4.2. Correlations

If the whole information concerning the dynamics of the system is introduced into the model, the prediction error $\varepsilon = Y(t) - \hat{Y}(t)$ is independent from the particular set of data used for validation. To prove this independence, two important functions are calculated: the autocorrelation $\hat{r}_{\varepsilon\varepsilon}(\tau)$ of the prediction error ε and the cross-correlations $\hat{r}_{U\varepsilon}(\tau)$ between the inputs and the prediction error [27]. The correlations results for this study are shown in Fig. 5.

$$\hat{r}_{\varepsilon\varepsilon}(\tau) = \frac{\sum_{t=1}^{N-\tau} (\varepsilon(t,\hat{\theta}) - \bar{\varepsilon})(\varepsilon(t-\tau,\hat{\theta}) - \bar{\varepsilon})}{\sum_{t=1}^{N} (\varepsilon(t,\hat{\theta}) - \bar{\varepsilon})^2} = \begin{cases} 1 if \tau = 0 \\ 0 if \tau \neq 0 \end{cases} \quad (19)$$

$$\hat{r}_{U\varepsilon}(\tau) = \frac{\sum_{t=1}^{N-\tau} (\varepsilon(t-\tau,\hat{\theta}) - \bar{\varepsilon})}{\sqrt{\sum_{t=1}^{N} (U(t) - \bar{U})^2 \sum_{t=1}^{N} (\varepsilon(t,\hat{\theta}) - \bar{\varepsilon})^2}} = 0, \forall\ \tau \quad (20)$$

2.4.3. Prediction reliability

The prediction reliability for a given input is computed as follows:

- Estimation of the prediction error variance compared to the regression vector *φ(t)*:
  $$\sigma_P^2(t) = E\left\{\varepsilon^2(t,\hat{\theta}) \middle| \varphi(t)\right\} \quad (21)$$
- Estimation of the interval of prediction confidence [28], see Fig. 6, by using the estimated variance and assuming that the prediction error has a Gaussian distribution:
  $$Y(t) \in [\hat{Y}(t|\hat{\theta}) - \sigma_p; \hat{Y}(t|\hat{\theta}) + \sigma_p] \quad (22)$$

3. Results and discussions

The NN model is validated using the third set of data $Z_V(U^3, Y^3)$ which contains 1500 points. The first 500 points of the set Z_V correspond to an excitation in the Z axis direction, the points from 501 to 1000 correspond to an excitation in the Y axis direction and finally points from 1001 to 1500 correspond to an excitation in the X axis direction. The comparison between the measured outputs and the one-step ahead

predictions shows how the model describes the system dynamic behaviour. Fig. 4 shows the visualisation of the first output prediction which corresponds to the X axis measuring channel of the three dimensional accelerometer on the top surface of the bipolar plate.

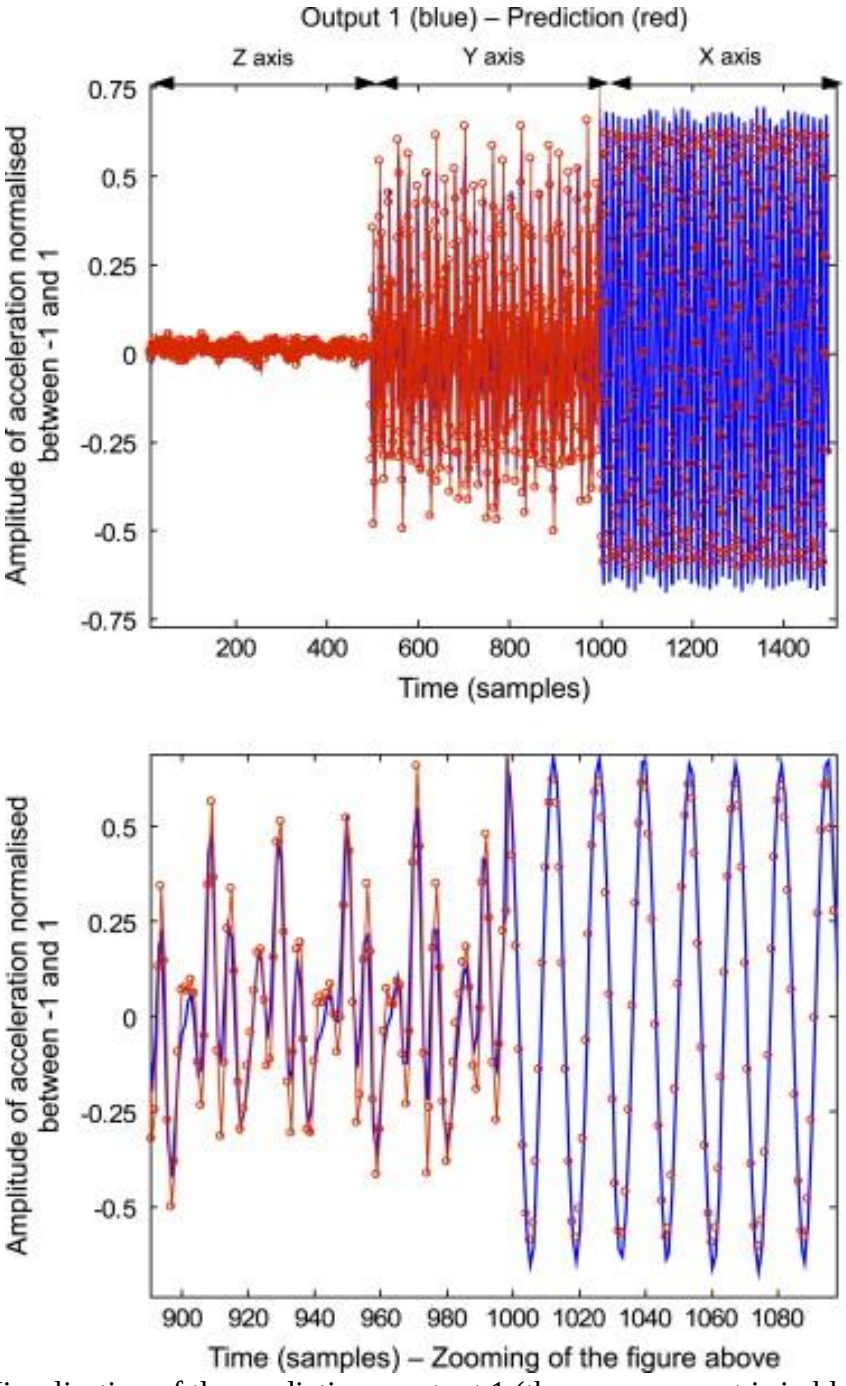

Fig. 4 - Visualisation of the prediction – output 1 (the measurement is in blue and the prediction is in red).

According to Fig. 4, output 1 is insensible to the excitation in the Z axis direction (points from 1 to 500). On the other hand, it is influenced by the excitation in the Y axis direction (points from 501 to 1000). This means that if the FC is vibrated in the Y axis

direction, an important signal, which corresponds to the dynamic response of the bipolar plate, is measured on the X axis direction. It can be deducted that the fixation of the FC bipolar plates are not rigid, its response in X and Y axis are much correlated.
This leads to the conclusion that, in order to have robust PEM fuel cells, a mechanical study for the conception of the bipolar plate fixation is necessary to be done with the aim of realizing a system with less correlation between excitement and measurements on different axes.
Fig. 4 shows also that prediction is close to the measured values. However, visual inspection is not enough. For this reason, correlation results and the histogram of the prediction error ε for output 1 are shown in Fig. 5 and Fig. 6.

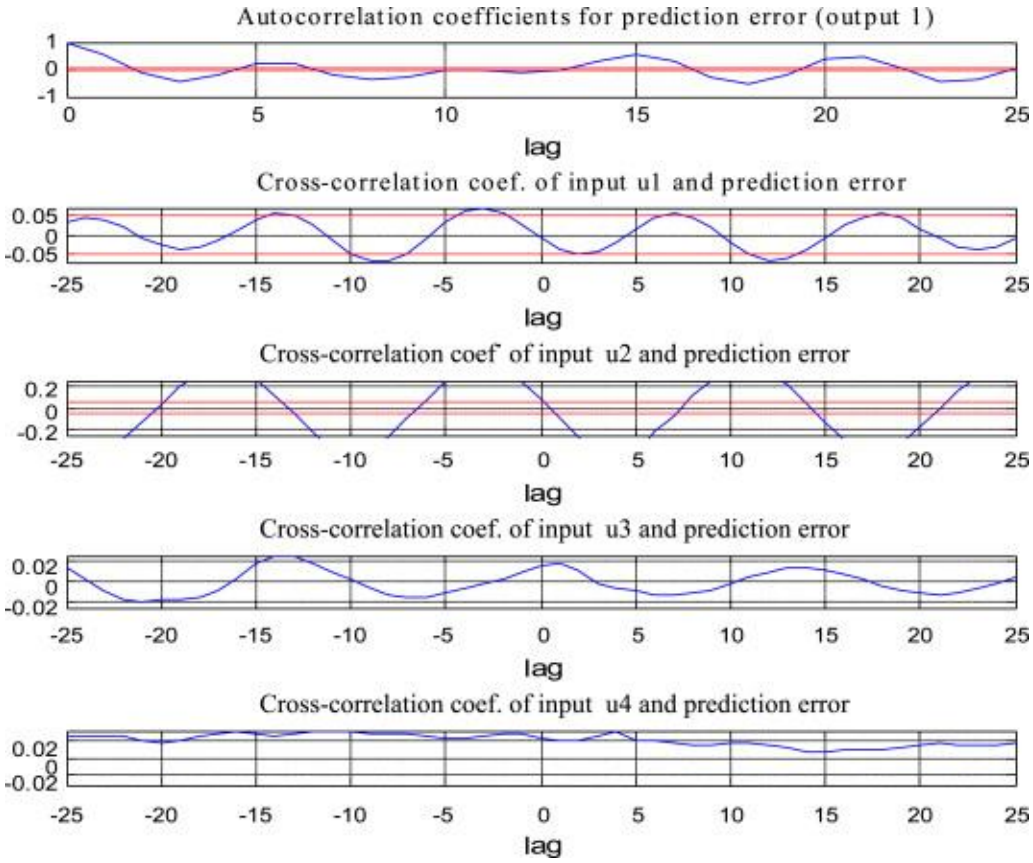

Fig. 5 - Correlations - output 1.

As displayed in Fig. 5, the correlation coefficients remain around one standard deviation. The autocorrelation function tends to be zero which means that the prediction error is independent of the control input. The cross-correlation functions vary in a band ranging from -0.05 to 0.05 for u^1 (excitation in X direction), from -0.2 to 0.2 for u^2 (excitation in Y direction), from -0.02 to 0.02 for u^3 (excitation in Z direction) and from -0.02 to 0.02 for u^4 (sampling frequency), close to zero.
The result of the cross-correlation is less good for u^2 (excitation in Y axis direction), this is overdue to the correlations between the excitation in Y direction (u^2) and the measurement in X direction (output 1). The same remark was given in the visualisation of the prediction.

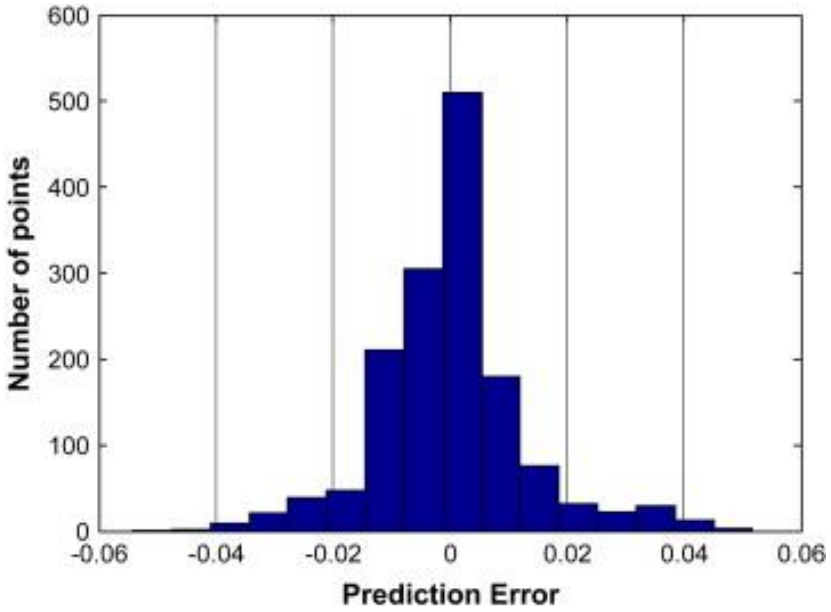

Fig. 6 - Histogram - output 1.

The histogram is shown in Fig. 6. Its shape is rather symmetric around zero and it has the majority of values around ε equal zero. This is a proof that the dynamics of the system were incorporated into the model.
The results for all outputs are acceptable and close to the measurements. However, it is important to indicate that the prediction is not the same for all axes. For example, for an output that measures the X axis (output 1), as detailed above, a better prediction is obtained for X, Z than Y axis respectively.
Also, the prediction is not the same for all outputs. This is related to the fact that the dynamic response of the FC is not the same in all parts of the FC. The measurement accelerometers have different positions. Therefore, the amplitudes of the measured responses vary with the accelerometer position. Only the results of output 1 are shown here as an example.

4. Conclusion and outlook

A neural approach, using a multi-layer perceptron combined with time regression input vector, is proposed to realize a three-dimensional model of the mechanical non-linear behaviour of a PEM fuel cell. An experimental setup is designed for this purpose. The mechanical system is independently vibrated in its three orthogonal axes X, Y and Z with swept sine derived from airplane requirements. The goal is to create a single spatial model. This is necessary to take account of the interferences between X, Y and Z behaviours. It is furthermore very useful to reduce calculations time instead of creating a model for each axis. The global model is trained and validated with data collected from the experimental setup. The obtained results are accurate.
The FC neural network model proposed can be used for monitoring purpose and especially to detect abnormalities in the mechanical behaviour of similar FC stacks placed under vibrating conditions. In this case, the anomaly patterns will be detected by considering the deviations between the measurements of the accelerometers and the corresponding values predicted by the neural network outputs. Such an approach which combines experiments with ANNs can be an efficient diagnosis tool that enables the

detection of structural damages inside the FC assembly and the moments when the failures happen.
A future work could be the implementation of the NN model as a controller. The model may then be used in a real time system in order to provide an environment allowing the analysis of FC mechanical performances and the optimisation of FC design parameters.

Acknowledgments

This work was done in the framework of the CELINA project. The EU is gratefully acknowledged for its financial support.

References

[1] Adams DE, Randall RJ. Survey of nonlinear detection and identification techniques for experimental vibrations. In: Proceedings of the 23rd International Conference on Noise and Vibration Engineering ISMA, Leuven, Belguim, 1998. p. 517-529.
[2] Nørgaad M, Ravn O, Poulsen NK, Hansen LK. Neural Networks for Modelling and control of Dynamic Systems. Springer-Verlag; 2000.
[3] Wei D, Cao G, Zhu X. Modeling and novel adaptive fuzzy neural network control of proton exchange membrane fuel cell (PEMFC). J. Shanghai Jiaotong University 2004;38:1581-1586.
[4] Jemei S, Hissel D, Péra MC, Kauffmann JM. Dynamical Recurrent Neural Network towards modelling of on-board Fuel Cell power supply. In: Industrial Electronics-IEEE International Symposium, 2004. p. 471-476.
[5] Hatti M, Tioursi M, Nouibat W. A Q-Newton method neural network model for PEM fuel cells. In: Industrial Informatics IEEE International Conference, Corte-Ajaccio, France, 2006. p. 1352-1357.
[6] Caponetto R, Fortuna L, Rizzo A. Neural network modelling of fuel cell systems for vehicles. In: EFTA, Emerging Technologies and Factory Automation, Catania, Italy, 2005. p. 1-6.
[7] Lee WY, Park GG, Yang TH, Yoon YG, Kim CS. Empirical modeling of polymer electrolyte membrane fuel cell performance using artificial neural networks. International Journal of Hydrogen Energy 2004;29:961-966.
[8] Ou S, Achenie LEK. A hybrid neural network model for PEM fuel cells. Journal of Power Sources 2005;140:319-330.
[9] Ou S, Achenie LEK. Artificial neural network modelling of PEM fuel cells. Journal of Fuel Cell Science and Technology 2005;2:226-233.
[10] Saxena A, Saad A. Evolving an artificial neural network classifier for condition monitoring of rotating mechanical systems. Applied Soft Computing 2007;7:441-454.
[11] Xiao Z, Wang S, Zeng H, Yuan X. Identifying of Hydraulic Turbine Generating Unit Model Based on Neural Networks. In: Proceedings of the sixth International Conference on Intelligent Systems Design and Applications ISDA, Jinan, China, 2006. p. 113-117.
[12] Yilmaz S, Atik K. Modeling of a mechanical cooling system with a variable cooling capacity by using artifical neural network. Applied Thermal Engineering 2007;27:2308-2313.
[13] Ozerdem MS, Kolukisa S. Artificial neural network approach to predict mechanical properties of hot rolled, nonresulfurized, AISI 10xx series carbon steel bars. Journal of

Materials Processing Technology 2008;199:437-439.
[14] Ciao M, Wang KW, Fujii Y, Tobler WE. Development of a friction component model for automotive powertrain system analysis and shift controller design based on parallel-modulated neural networks. Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME 2005;127:382-405.
[15] Linchong Y, Guangchen B, Junting J, Lui L. Prediction on nonlinear dynamic responds of non-lubrication mechanism system considered friction and flexibleness based on neural networks. Run Hua Yu Mi Feng/Lubrication Engineering 2006;4:45-47.
[16] Zolock J, Greif R. A methodology for the modelling of rail vehicles from time series measurements using time-delay neural networks. In: Proceedings of the ASME rail transportation division fall conference, Chicago, IL, 2007. p. 95-104.
[17] Nouria H, Foltete E, Hirsinger L, Ballandras S. Experimental evidence and Modeling of microsliding on cantilever quartz beam. In: Proceedings of IEEE Ultrasonics Symposium, New York, NY, 2007. p. 195-199.
[18] Rouss V, Charon W. Multi-input and multi-output neural model of the mechanical nonlinear behaviour of a MEM fuel cell system. Journal of Power Sources 2007;175:1-17.
[19] Haber R. Nonlinearity tests for dynamic process. In: Barker HA, Young PC, editors. Proc. IFAC Symp. on Identification and System Parameter Estimation. York, UK, 1985. p. 409-414.
[20] He X, Asada H. A new method for identifying orders of input-output models for nonlinear dynamic systems. In: Proc. of the American Control Conference. San Francisco, California, 1993. p. 2520-2523.
[21] Haykin S. Neural Networks: a Comprehensive Foundation. Prentice Hall; 1998.
[22] Ljung L. System Identification – Theory for the User, Upper Saddle River. Prentice-Hall; 1999.
[23] Marquardt D. An Algorithm for Least-Squares Estimation of Nonlinear parameters. SIAM Journal Appl. Mathematics 1963;11:164-168.
[24] Fletcher R. Practical Methods of Optimisation. Second ed., Wiley and Sons; 1987.
[25] Larsen J, Hansen L. Generalization Performance of Regularized Neural Network Models. In: Proceeding of the IEEE Workshop on Neural networks for Signal Proc. IV. Piscataway, New Jersey, 1994.
[26] Le Cun Y, Denker J, Solla S, Optimal Brain Damage. In: Touretzky DS, editors. Neural Information Processing Systems, Denver, 1990.
[27] Hansen LK, Larsen J. Linear Unlearning for Cross-Validation. Advances in Computational Mathematics 1996;5:269-280.
[28] Sørensen O. Non-linear pole-placement control with a neural network. European J. Control 1996;2:36-43.

Nomenclature

Alphabets

DO-160D	Environmental conditions and test procedures for airborne equipment
$f^{(i)}$	Search direction of the 'Levenberg-Marquardt' method
FPE	Final prediction error
$G^{(i)}(\theta)$	The gradient of $L^{(i)}(\theta)$
i	Number of inputs

j	Number of outputs
$L^{(i)}(\theta)$	Minimization criterion of the 'Levenberg-Marquardt' method
n	Number of delays
N	Number of points in the training set
N_{ei}	Input i time regressor
N_s	Output j time regressor
q_{ij}	Lipschitz coefficient
$\hat{r}_{\varepsilon\varepsilon}(\tau)$	Auto-correlation of the prediction error
$\hat{r}_{U\varepsilon}(\tau)$	Cross-correlation between the inputs and the prediction error
$R^{(i)}(\theta)$	The Hessian of $L^{(i)}(\theta)$
T	Number of points in the pruning set
$u^1(t)$	Excitation signal vector
$u^2(t)$	Sampling frequency vector
V	Number of points in the validation set
V_N	Levenberg-Marquardt minimisation criterion or training error
V_T	Test error
W^1	Vector of weights between inputs and the hidden layer
W^2	Vector of weights between the hidden layer and outputs
$Y(t)$	Measured outputs
$\hat{Y}(t)$	Predicted outputs
z	Total number of time regressors
Z_N	Training data set
Z_T	Pruning data set
Z_V	Validation data set

Greek symbols

ε	Prediction error
θ	Vector of weights
$\varphi(t)$	Time regression vector
$\sigma_P^2(t)$	Variance of the prediction error