Available online at www.sciencedirect.com
ScienceDirect
Journal of Power Sources 175 (2008) 1–17
JOURNAL OF POWER SOURCES
www.elsevier.com/locate/jpowsour

Review

Multi-input and multi-output neural model of the mechanical nonlinear behaviour of a PEM fuel cell system

Vicky Rouss *, Willy Charon

M3M, Université de Technologie de Belfort Montbéliard, Rue de Château, 90010 Belfort, France

Received 11 July 2007; accepted 5 September 2007
Available online 12 September 2007

Abstract

A fuel cell system model is necessary to prepare and analyse vibration tests. However, in the literature, the mechanical aspect of the fuel cell systems is neglected. In this paper, a neural network modelling approach for the mechanical nonlinear behaviour of a proton exchange membrane (PEM) fuel cell system is proposed. An experimental set is designed for this purpose: a fuel cell system in operation is subjected to random and swept-sine excitations on a vibrating platform in three axes directions. Its mechanical response is measured with three-dimensional accelerometers. The raw experimental data are exploited to create a multi-input and multi-output (MIMO) model using a multi-layer perceptron neural network combined with a time regression input vector. The model is trained and tested. Results from the analysis show good prediction accuracy. This approach is promising because it can be extended to further complex applications. In the future, the mechanical fuel cell system controller will be implemented on a real-time system that provides an environment to analyse the performance and optimize mechanical parameters design of the PEM fuel system and its auxiliaries.

Keywords: Vibration; Neural network; Nonlinear; Mechanical system; Fuel cell modelling

Contents

* Corresponding author. Tel.: +33 3 84 58 31 36; fax: +33 3 84 58 31 46.
E-mail addresses: vicky.rouss-01@utbm.fr (V. Rouss), willy.charon@utbm.fr (W. Charon).

0378-7753/$ – see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.jpowsour.2007.09.008

Nomenclature

$f^{(i)}$	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_{sj}	output j time regressor
q_{ij}	Lipschitz coefficient
$\hat{r}_{\varepsilon\varepsilon}(\tau)$	autocorrelation 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 minimization 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
$\sigma^2_P(t)$	variance of the prediction error
$\varphi(t)$	time regression vector

1. Introduction

Because of a worldwide increase in air pollution and power demand, an efficient and clean generation of electrical energy has become a necessity. With low emissions and very high-conversion efficiencies, fuel cells are the best candidate to become the primary source of power in the future [1]. Therefore worldwide attention has been focused on their development.

A fuel cell is a device that can directly convert chemical to electric and thermal energy. Among different kinds of fuel cells, the proton exchange membrane fuel cell (PEM) has the advantage of a low-operational temperature (20–100 °C), high-power density and light weight. PEM has gained a lot of attention and is considered as the most promising fuel cell technology in the future and a potential alternative power source [2]. It fits at best the requirements of the transportation systems. However in order to implement fuel cells in transportation systems, we need to master the mechanical behaviour of fuel cell systems and the influences of mechanical loads on their structure. Durability versus mechanical loads is also a matter of concern.

Therefore, experimental investigations under mechanical loads are necessary to acquire further knowledge. In order to optimally pilot and control the tests, highly efficient models with good performance in static and dynamic operational conditions are required.

Before enumerating these models, it is useful to distinguish the two terms: 'dynamic' and 'mechanical' in the fuel cell field. The term 'dynamic' is related to the fuel cell dynamical physicochemical aspect as transient behaviour while 'mechanical' indicates its mechanical vibrating aspect as eigen modes and frequencies.

In the literature, a large number of steady-state models are available, which mainly focus on designing the PEM and choosing the operating point [3]. Baschuk and Xianguo [4] have developed a model considering all three reasons of polarization in a unified fundamental approach. Voss et al. [5] suggested a technique for water removal from the PEM. Nguyen and White [6] developed a model to investigate the effectiveness of three humidification designs. Fuller and Newman [7] simulated a PEMFC operation in conditions of moist gas in the electrodes. Bernardi and Verbrugge [8] formulated a simplified one-dimensional model for liquid water transport in porous electrodes. Wang et al. [9] studied gas–liquid two-phase flow and transport in air operated PEMFC as well as direct methanol fuel cell. Djilali and Berning [10] presented an overview of the role of various transport phenomena in fuel cell operation and some of the physical and computational modelling challenges. Du and Shi [11] exposed a computational fluid dynamic (CFD) model for a PEM by taking into account the catalyst layer with agglomerate structures.

There are relatively few dynamic models proposed in the published works that have studied transient behaviour [12] and fuel cell control: a quasi-three-dimensional dynamic model for physical, chemical and electrochemical processes in PEM [13], a model to predict responses of the electric load on the cell

by emphasising the temperature response on the dynamic load [14], a model predictive control-based strategy [15], an exact linearization approach to create a dynamic model [16], a study of a fuel cell stack dynamic response including the nonlinearities [17], a model of a fuel cell hybrid system connected to a nonlinear load power [18].

These fuel cell static and 'dynamic' models provide a fundamental foundation for exploring mechanisms and understanding physical phenomena in the PEM. In spite of advances in modelling, several of these physical methods are not accurate enough because of the complex nonlinear nature of the fuel cell. Usually, modelling is done with models based on the knowledge of physicochemical phenomena. These models require a good knowledge of the process parameters. In most cases, these parameters are difficult to determine for an operating fuel cell system [19]. One way to solve this problem is to exploit raw experimental data by creating a "black box" model which requires no preliminary knowledge about the system [20]. Several techniques can be implemented to create a black box model but a well-designed artificial neural network (ANN) model provides useful and reasonably accurate input–output relations because of its excellent multi-dimensional mapping capability. It can predict the desired output variables faster and more accurately than a physical model. For this reason, ANN has been extensively employed in various areas of science and technology.

All applications of neural networks fuel cell models proposed in publications concern the 'dynamic' aspect such as: an adaptive fuzzy identification model based on input–output sampled data [21], a dynamic recurrent neural network model for fuel cell dynamic operating modes [22], a Q-Newton neural network model which estimates the voltage considering different operating conditions and delivered current [23], a multi-layer perceptron (MLP) model that inputs pressures and temperatures at the stack and outputs voltage of the fuel cell [24,25], a hybrid neural network model consisting of an ANN component and a physical component that takes into consideration the effect of Pt loading [26], 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 [27]. All of these studies were conceived to predict the cell voltage or cell power density (mA cm^{-2}) or to determine the operating temperature of PEM.

However, a model that predicts the nonlinear mechanical response of a fuel cell system submitted to vibrations during a duration of time has not yet appeared in published works. In this paper, a method based on a MIMO multi-layer perceptron (MLP) neural network combined with a time regression input vector is proposed [28]. A specified experimental set has been designed and a fuel cell system, in operation, vibrated according to the three axes.

The paper is organised as follows. Section 2 provides a general description of the test bench which includes the vibrating platform, the fuel cell system and the acquisition system. Section 3 explains the neural network modelling approach. Section 4 illustrates simulation results for different excitations; they are compared to experimental findings and discussed. Finally, Section 5 concludes the paper and outlines the practical implementation of the black box model.

Fig. 1. The vibrating platform.

2. The test bench

The experimental setup is composed of a vibrating platform, a PEM fuel cell system to be modelled and an acquisition system.

2.1. The vibrating platform

The vibration tests were realized by using the vibrating platform represented in Fig. 1.

The platform is composed of two tables: one for horizontal solicitations and the other for vertical excitations. The first table vibrates in *X* and *Y* axes directions, the second one in the *Z*-axis direction. The platform has the following properties:

- frequency sweeping: 6–3000 Hz,
- maximum displacement of the vibrating table (peak to peak): 50 mm,
- maximum velocity: 1.8 m s^{-1},
- maximum force: 35.6 kN.

Using specific software, it is thus possible to control excitations as swept sine, shocks and random vibrations. The vibration platform also has the capacity to reproduce any kind of signals.

2.2. The PEM fuel cell system

The PEM fuel cell system is considered as a mechanical system to be modelled. The aim is to represent its nonlinear mechanical behaviour. The system is screwed to the vibrating table (see Fig. 2).

The control accelerometer is directly fixed on the vibrating table and 3 three-dimensional measurement accelerometers are distributed on the fuel cell system (see Fig. 3).

Fig. 2. The mechanical system fixed on the vibrating table.

2.3. The acquisition system

The installation is equipped with an acquisition system. However, raw data, necessary to feed the neural model, cannot be accessed. For this reason, a USB interface with a new acquisition program is implemented.

Another advantage of the new program is the method of choosing the sampling frequency. In fact, frequency bandwidth for the realized tests varies from 6 to 1250 Hz. If data are collected with a constant sample rate, its value should be around 12,500 Hz. Therefore, with excitations like swept sine, a very large data vector is obtained. That will lead to increase the training set dimension and eventually the model estimation time. The processing of such a vector with common devices, even with powerful PCs is time consuming.

To solve this problem, the new acquisition program allows collection of data with a variable sampling frequency in real time. This permits to cover all the frequency bandwidth with a reasonable number of points.

3. Modelling approach

The neural network approach for system modelling consists of several steps, as shown in Fig. 4. 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 selecting inputs and outputs, the number of layers and 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 and the third one is employed to validate the model.

3.1. Tests

Collecting results from several dynamic load tests is the first step of the procedure. Before capturing data needed to setup 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 [29].

The system is excited in the three axes directions separately. Therefore, there are three models to show the system's mechanical behaviour in each of the three directions of excitations. The procedures for developing these models are similar. This is the reason why only the *Z*-axis model is exposed in the following.

During the *Z*-axis direction tests, the system is excited with two types of vibrations:

- swept sine with a frequency bandwidth from 6 to 1250 Hz and acceleration amplitude from 0.1 to 1 g,
- random vibration with frequency bandwidth from 6 to 1250 Hz, a root mean square (rms) value of 0.35 g and a maximum value of 0.95 g. In fact, the random vibration used here is related to a road profile directly measured on the vehicle where the fuel cell system is destined to be installed.

A unique neural network is created in order to model the system according to these two types of excitation in the *Z*-axis direction.

3.2. Selection of the neural model structure

The second step deals with finding the neural network structure. A multi-layer structure of feed-forward MLP is chosen here because of its capability in nonlinear modelling. 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 nine outputs (see Fig. 7) which corresponds to the prediction of nine measuring channels for the three three-dimensional accelerometers. They are here compared with the measurements of these accelerometers.

A point should be clarified: the excitation signal is in the *Z* axis direction but the measured accelerations are in the *X*, *Y* and *Z* axes directions.

The number of inputs has to be determined. Indeed, the only actual input is the acceleration subjected to the vibrating table. However, the neural network is fed with values of acceleration and measured outputs from the previous time instants. This is known as time regression (NNARXM), which is essential in modelling the nonlinear behaviour and is a dominant factor in the calculation time (see Fig. 5). It can be derived by using Lipschitz approach [30]. In the next paragraph, a brief presentation of this method is given.

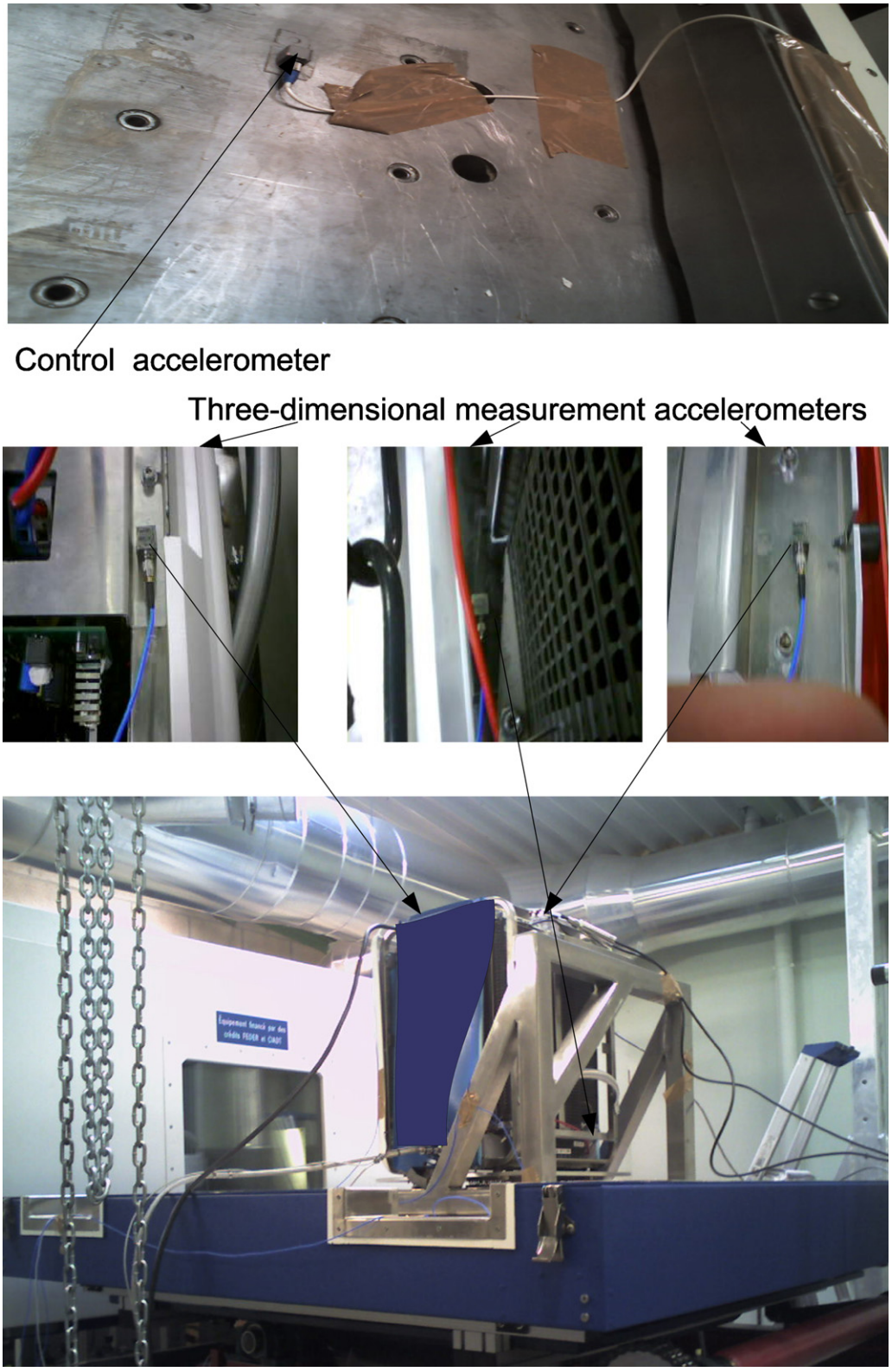

Fig. 3. The accelerometers.

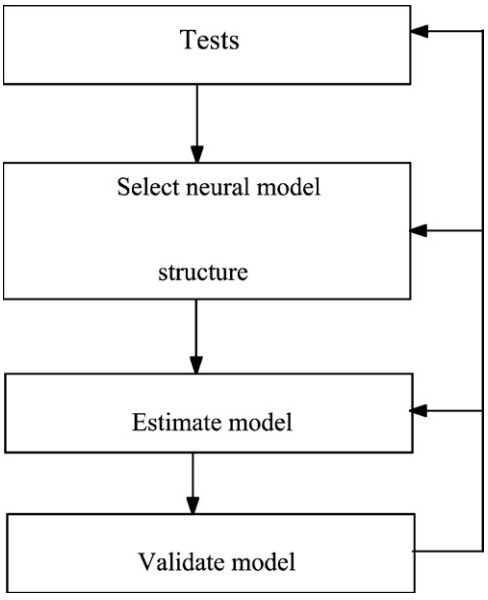

Fig. 4. Steps of the modelling approach.

Define the training set:

$$Z_N = \{[U^1(t), Y^1(t)], \quad t = 1, \ldots, N\} \tag{1}$$

which is composed of an (2,*N*) imposed acceleration matrix:

$$U^1(t) = [u^1(t); u^2(t)] \tag{2}$$

and an (9,*N*) output matrix $Y^1(t)$ corresponding to the measured outputs:

$$Y^1(t) = [y^1(t); y^2(t); \ldots; y^9(t)] \tag{3}$$

where N is the total number of points in the training set.

The neural network input is the time regression vector $\varphi(t)$, which is calculated as follows:

$$\begin{aligned}\varphi(t) &= [\varphi_1, \varphi_2, \varphi_3, \ldots, \varphi_z] \\ &= [u^1(t-1)\, u^1(t-2) \ldots u^1(t-N_{e1})\, u^2(t-1)\, u^2 \\ &\quad \times (t-2) \ldots u^2(t-N_{e2})\, y^1(t-1)\, y^1(t-2) \ldots y^1 \\ &\quad \times (t-N_{s1})\, y^2(t-1)\, y^2(t-2) \ldots y^2(t-N_{s2}) \ldots y^9 \\ &\quad \times (t-1)\, y^9(t-2) \ldots y^9(t-N_{s9})]\end{aligned} \tag{4}$$

where N_{ei} and N_{sj} are, respectively, the number of input (i) and output (j) time regressors, i varies from 1 to 2 and j varies from 1 to 9 and $u^1(t)$ and $u^2(t)$ are, respectively, the values of the control accelerometer signal and the sampling frequency corresponding to each point.

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:

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

The approach consists in choosing different delay couples N_{ei} and N_{sj}. For each couple, the Lipschitz coefficients are computed

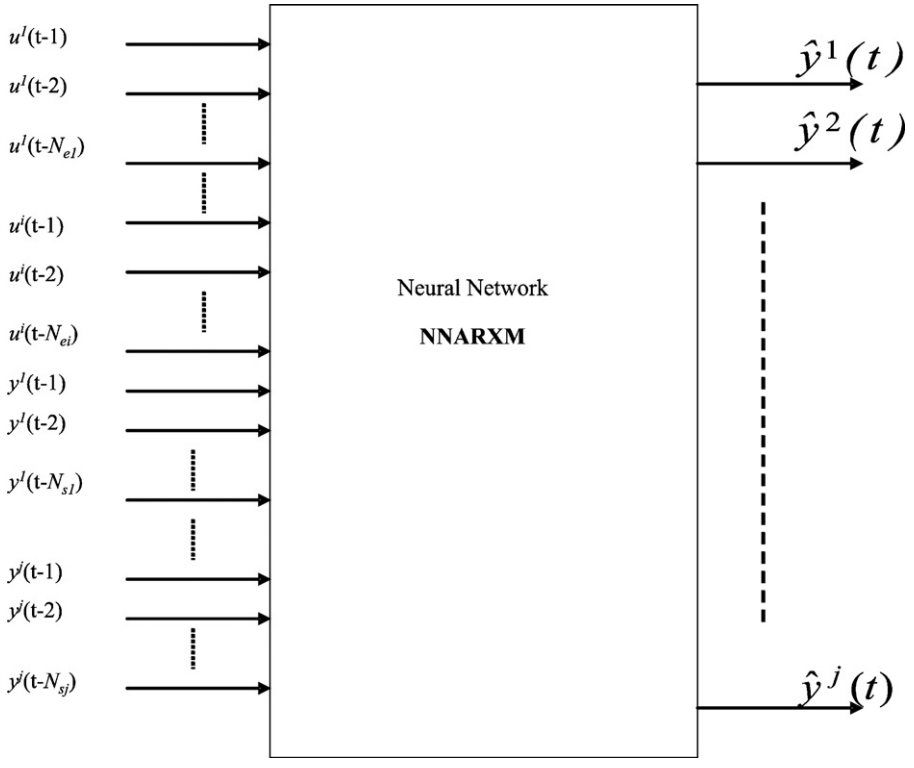

Fig. 5. A neural network with NNARXM regression vector.

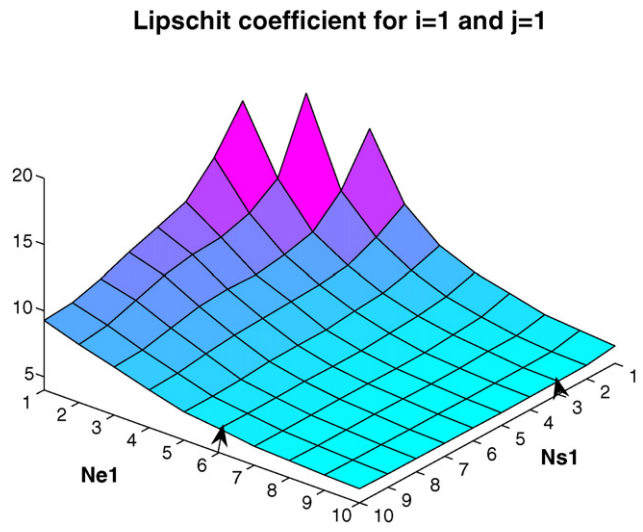

Fig. 6. Lipschitz coefficient.

for all input $u^i(t)$ and output $y^j(t)$ combinations. The greatest p_{ij} coefficients are selected to calculate the criterion:

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

where $n = N_{ei} + N_{sj}$ 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.

The results of the calculations for $i = 1$ and $j = 1$ are shown in Fig. 6. At first, the curve decreases and then stabilizes for a precise value of N_{ei} and N_{sj}. In this study, N_{e1} has a value of 6 and N_{s1} a value of 3.

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

$$z = N_{e1} + N_{e2} + N_{s1} + \cdots + N_{s9} \tag{7}$$

The results of the calculations are

$$N_{e1} = 6, \quad N_{e2} = 1, \quad N_{s1} = 3, \quad N_{s2} = 3, \quad N_{s3} = 5,$$
$$N_{s4} = 3, \quad N_{s5} = 3, \quad N_{s6} = 5, \quad N_{s7} = 3,$$
$$N_{s8} = 3 \text{ and } N_{s9} = 5$$

That means z equals 40 ($z = 6 + 1 + 3 + 3 + 5 + 3 + 3 + 5 + 3 + 3 + 5 = 40$).

One hidden layer containing 18 neurons is initially chosen (Fig. 7). The experience showed that the number of neurons is large, but it will be optimized later using specific algorithms. The nonlinear activation functions [31] for the hidden neurons are a hyperbolic tangent type $f(x) = \tan h(x)$ 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. Finally, the model is validated. In order to do this, data collected during the tests are divided into three different sets, one for each of these tasks.

3.3. Estimate the model

This section is composed of two parts: training [32] and pruning [33] discussed, respectively.

3.3.1. Training

The first set of the collected data $Z_N(U^1, Y^1)$ is used in the training. 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 2080 points ($N = 2080{:}1520$ coming from random tests and 560 from swept-sine tests) 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:

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

where N is the number of points of the training set (TS), $Y(t) = Y^1(t) = [y^1(t); y^2(t); \ldots; y^9(t)]$ is the vector of real measured outputs in the TS, $\hat{Y}(t) = [\hat{y}^1(t|\theta); \hat{y}^2(t|\theta); \ldots; \hat{y}^9(t|\theta)]$ is the vector of the predicted outputs, and θ 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 ($40(z) \times 18$(initial number of neurons) $= 720$), and the 18 bias values for the neurons in the hidden layer.
- W^2, containing the weights between the hidden layer and the outputs (18(initial number of neurons) $\times$ 9(number of outputs) $= 162$) and the 9 bias for the outputs neurons.

Thus, $\theta = [W^2 W^1]$ is a vector containing: $720 + 18 + 162 + 9 = 909$ weights.

This approach based on the minimization of V_N is called the *prediction error method* (PEM) [34]. Among the different minimization methods which use the PEM criterion, the MLP network combined with NNARXM regression vector uses the 'Levenberg–Marquardt' method [35]. 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 $\theta^{(i)}$:

$$\theta^{(i+1)} = \underset{\theta}{\operatorname{argmin}}(L^{(i)}(\theta)) \text{ in the neighbourhood } |\theta - \theta^{(i)}| \leq \delta^{(i)} \tag{9}$$

The next iteration values are computed by the formula:

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

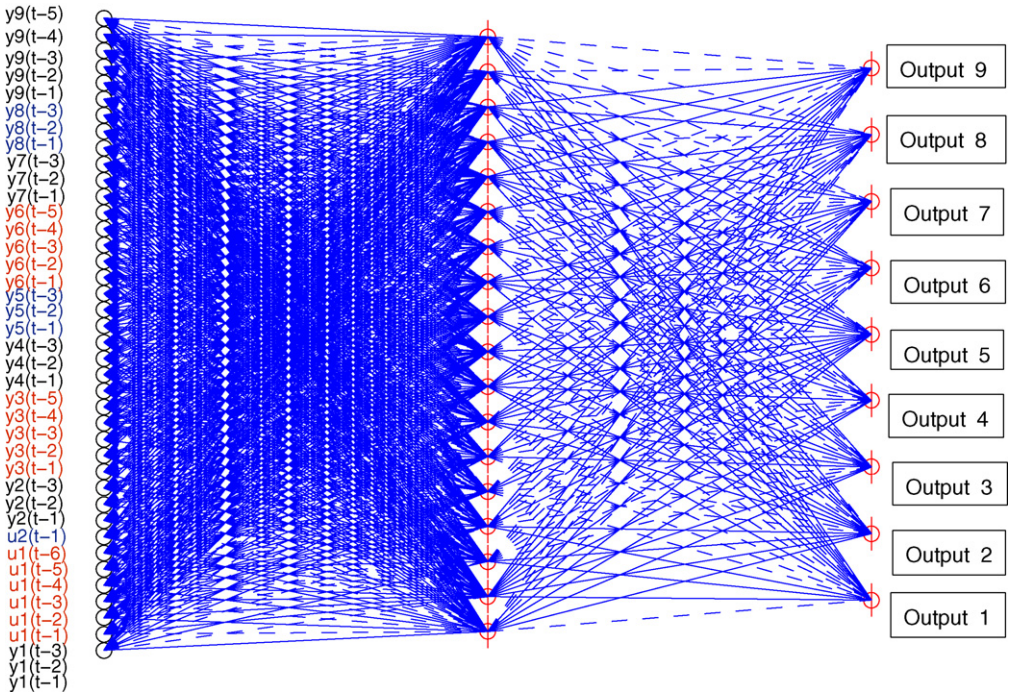

Fig. 7. The initial neural network.

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

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

where

- $\lambda^{(i)}$ is a parameter that varies between 0 and infinity, whose determination is explained in Fig. 8,
- the gradient:

$$G(\theta^{(i)}) = \left.\frac{\mathrm{d}L^{(i)}(\theta)}{\mathrm{d}\theta}\right|_{\theta=\theta^{(i)}} = \frac{1}{N}\sum_{t=1}^{n}\{[\psi(t,\theta^{(i)})]^T[y(t) - \hat{y}(t|\theta^{(i)})]\} \tag{12}$$

where

$$\psi(t,\theta^{(i)}) = \frac{\mathrm{d}\hat{y}(t|\theta^{(i)})}{\mathrm{d}\theta} \tag{13}$$

- the hessian:

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

To determine $\lambda^{(i)}$, there are two different methods: direct and indirect methods. The indirect method of Fletcher [36], presented in Fig. 8, is used here. The approach consists in computing the ratio:

$$r^{(i)} = \frac{V_N(\theta^{(i)}) - V_N(\theta^{(i+1)})}{V_N(\theta^{(i)}) - L^{(i)}(\theta^{(i+1)})} = \frac{(V_N(\theta^{(i)}) - V_N(\theta^{(i)} + f^{(i)})}{(V_N(\theta^{(i)}) - L^{(i)}(\theta^{(i)} + f^{(i)})} \tag{15}$$

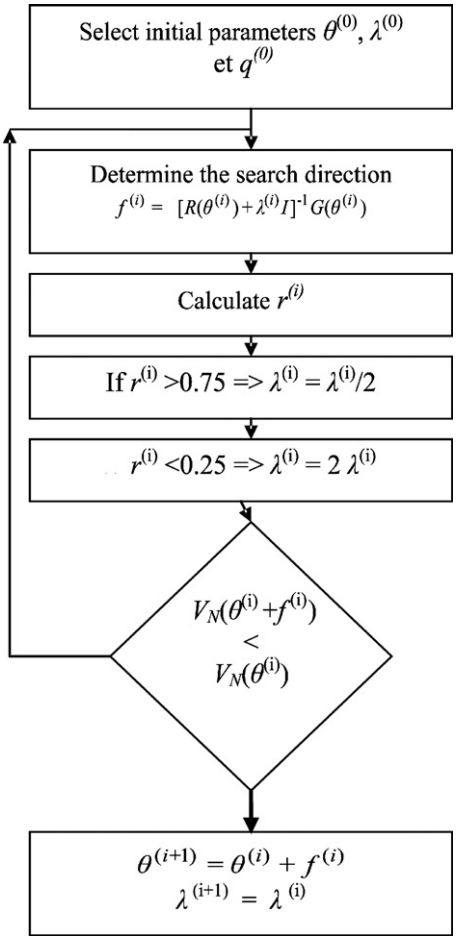

Fig. 8. Determination of $\lambda^{(i)}$.

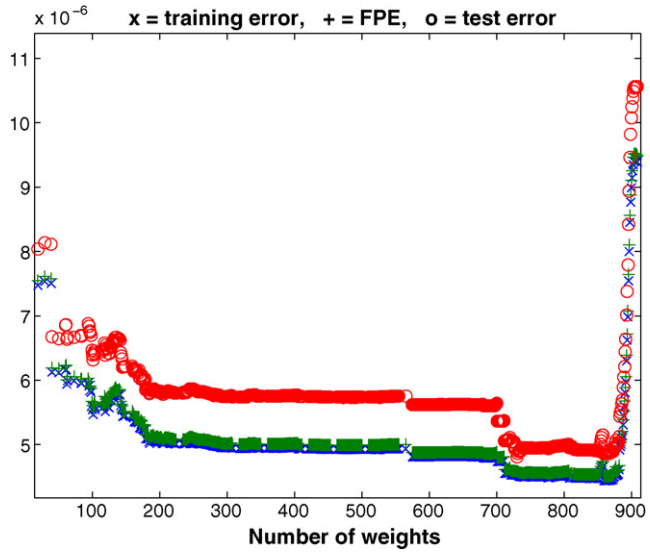

Fig. 9. Variation of the test error.

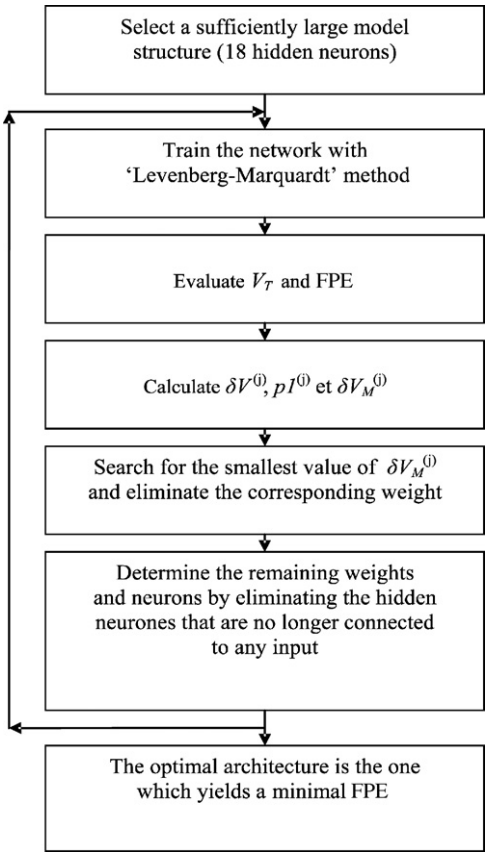

Fig. 10. Pruning.

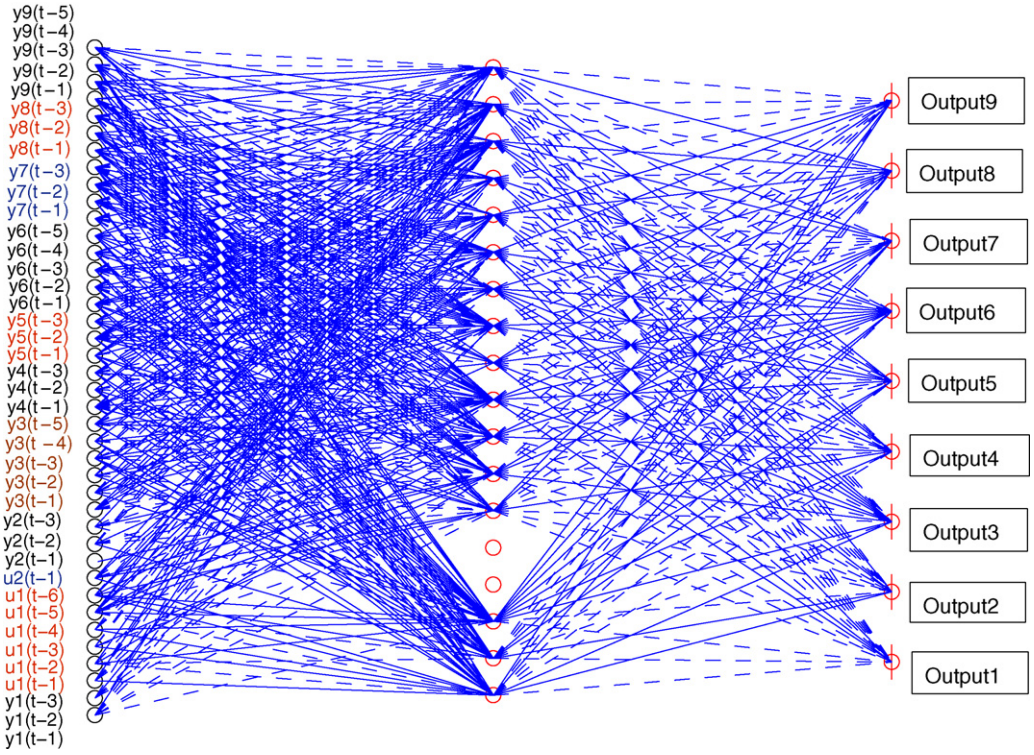

Fig. 11. The final neural network after pruning.

where

$$L^{(i)}(\theta^{(i)} + f) = V_N(\theta^{(i)}) + f^T G(\theta^{(i)}) + \tfrac{1}{2} f^T R(\theta^{(i)}) f \qquad (16)$$

3.3.2. 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 [37]. 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

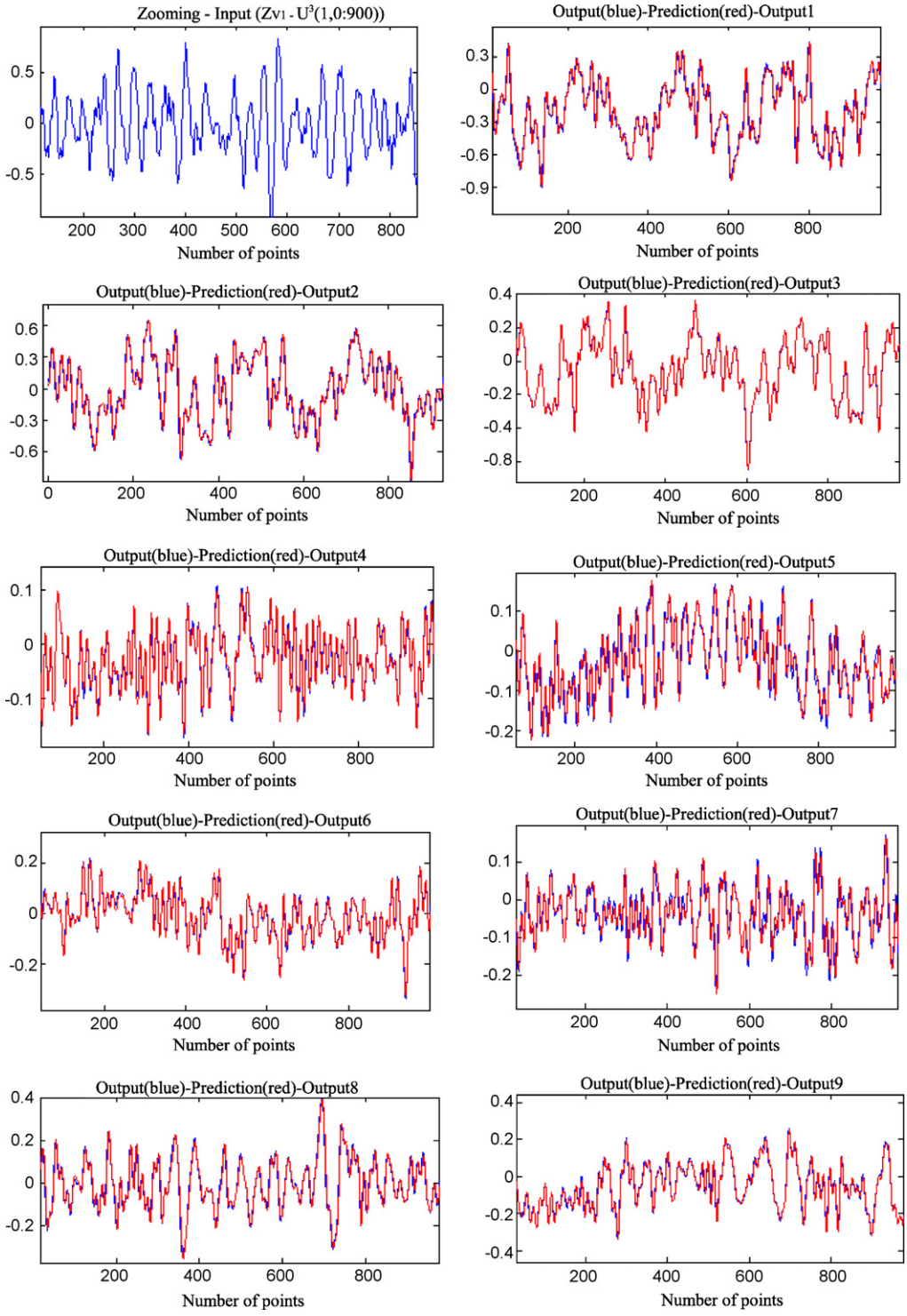

Fig. 12. Visualization of the prediction—random tests. (For interpretation of the references to color in this figure citation, the reader is referred to the web version of the article.)

training section. The number of points of Z_T is here equal to 2000.

3.3.2.1. Final prediction error. The final prediction error is expressed in the formula:

$$\mathrm{FPE} = \hat{V}_M = \frac{1}{2}\sigma_e^2\left(1 + \frac{p}{N}\right) \tag{17}$$

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

$$\hat{\sigma}_e^2 = 2\frac{N}{N+p}V_N(\hat{\theta}, Z^N) \tag{18}$$

The FEP reaches its maximum value $\sigma_e^2/2$ when N tends towards infinity. However, as the vector of training is finite ($N = 2080$ in this study), the error is always higher than $\sigma_e^2/2$.

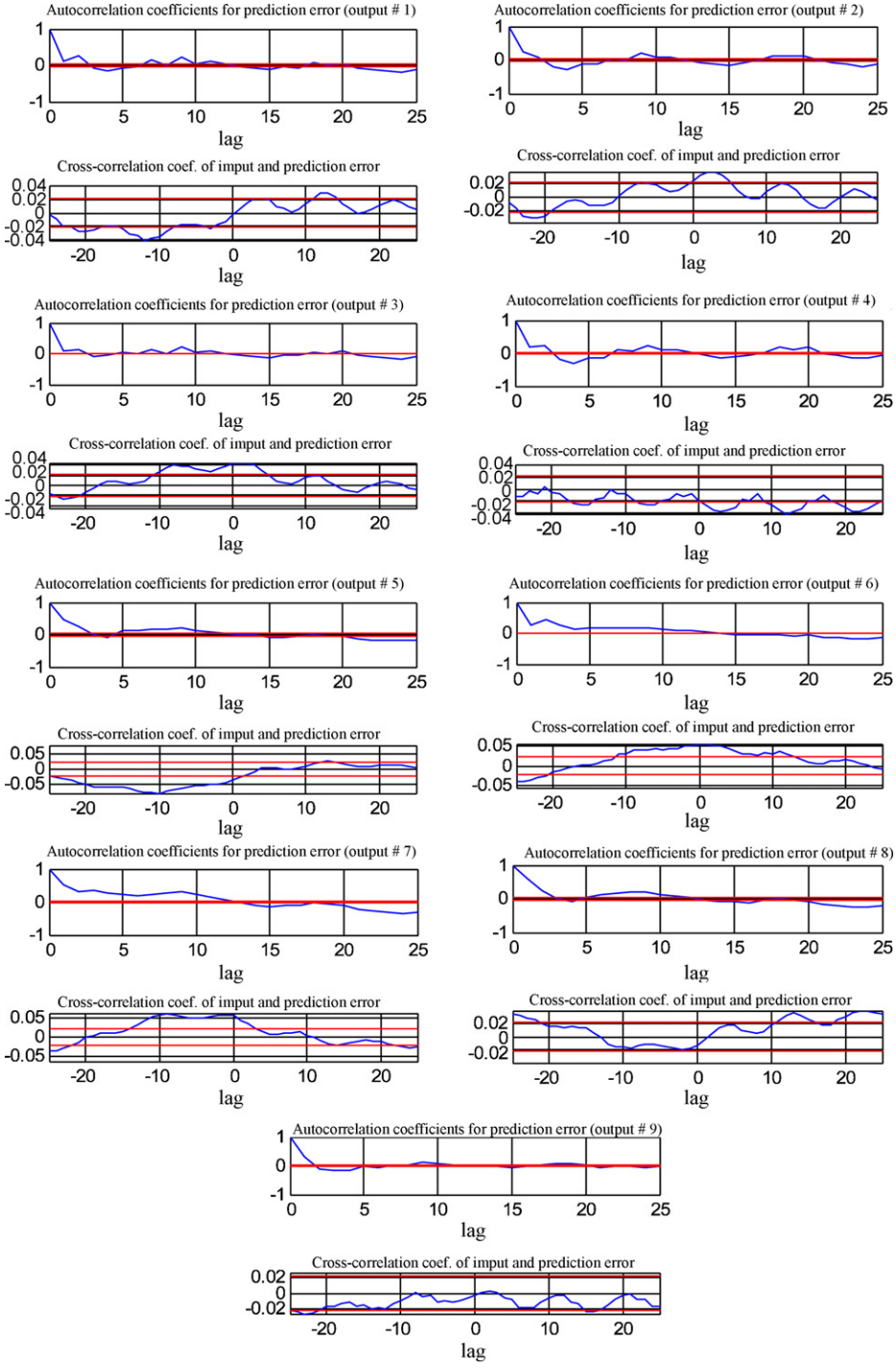

Fig. 13. Correlations for random tests. (For interpretation of the references to color in this figure citation, the reader is referred to the web version of the article.)

3.3.2.2. 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. 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 optimized model is not found in the set of candidate models defined in the neural structure selection (Section 3.2). 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.

Fig. 9 shows the effects of both errors. The curve of V_T decreases, then increases. The curve's minimum corresponds to the best model.

3.3.2.3. 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, and then prune the weights of one at a time until the best architecture is found [38]. Fig. 10 explains this procedure.

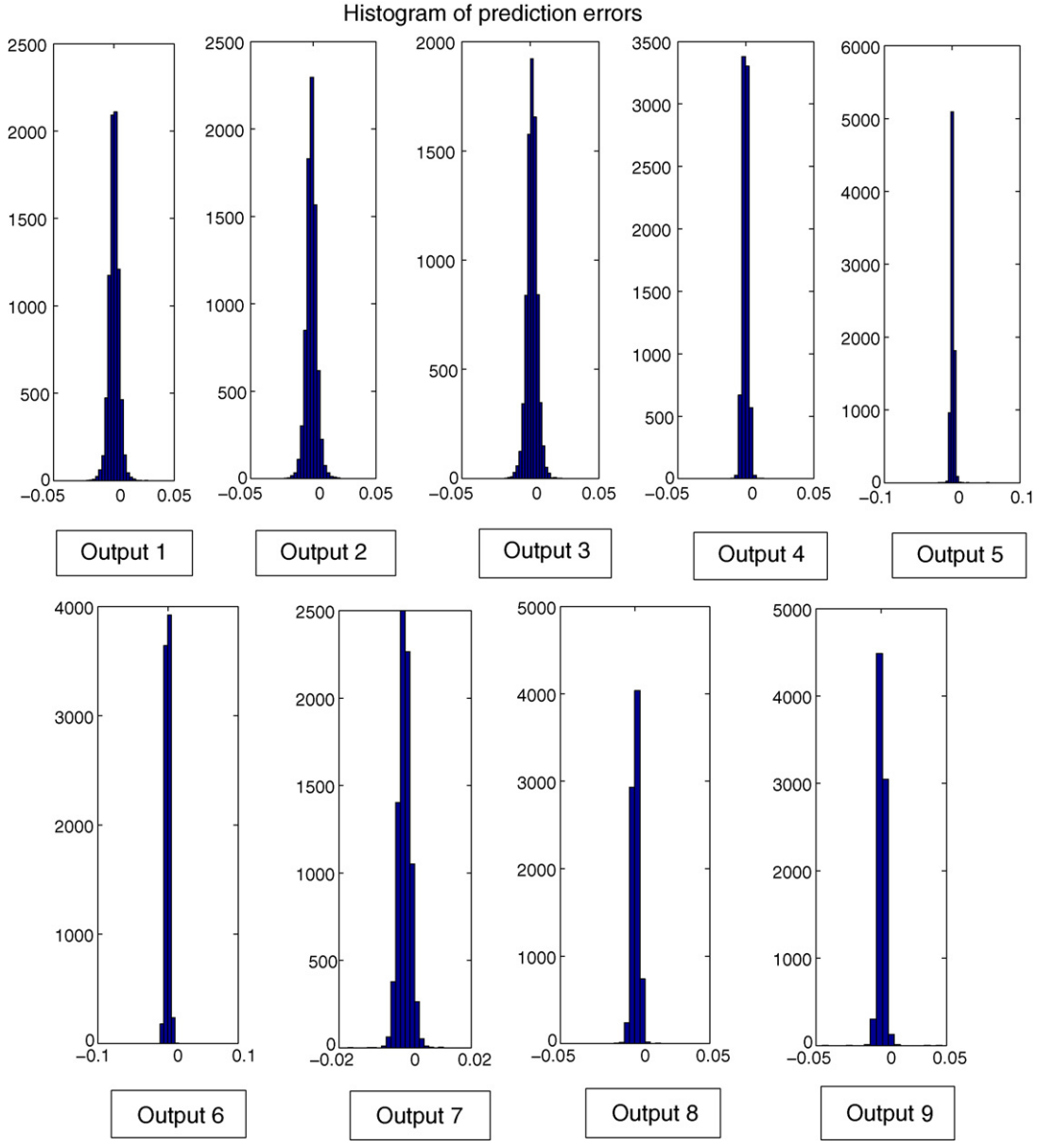

Fig. 14. Histograms for random tests. (For interpretation of the references to color in this figure citation, the reader is referred to the web version of the article.)

Listed below is an explanation of some symbols mentioned in the above figure:

$$\delta V^{(i)} = V_N(\hat{\theta}^{(j)}, Z^N) - V_N(\hat{\theta}, Z^N) \quad (19)$$

where $\hat{\theta}^{(j)}$ is the reduced vector of weights [39]:

$$\delta V_M^{(i)} = V_M(\hat{\theta}^{(j)}, Z^N) - V_M(\hat{\theta}, Z^N) \quad (20)$$

where V_M is the FPE error and $p1^{(j)}$ is the number of remaining weights at the jth iteration.

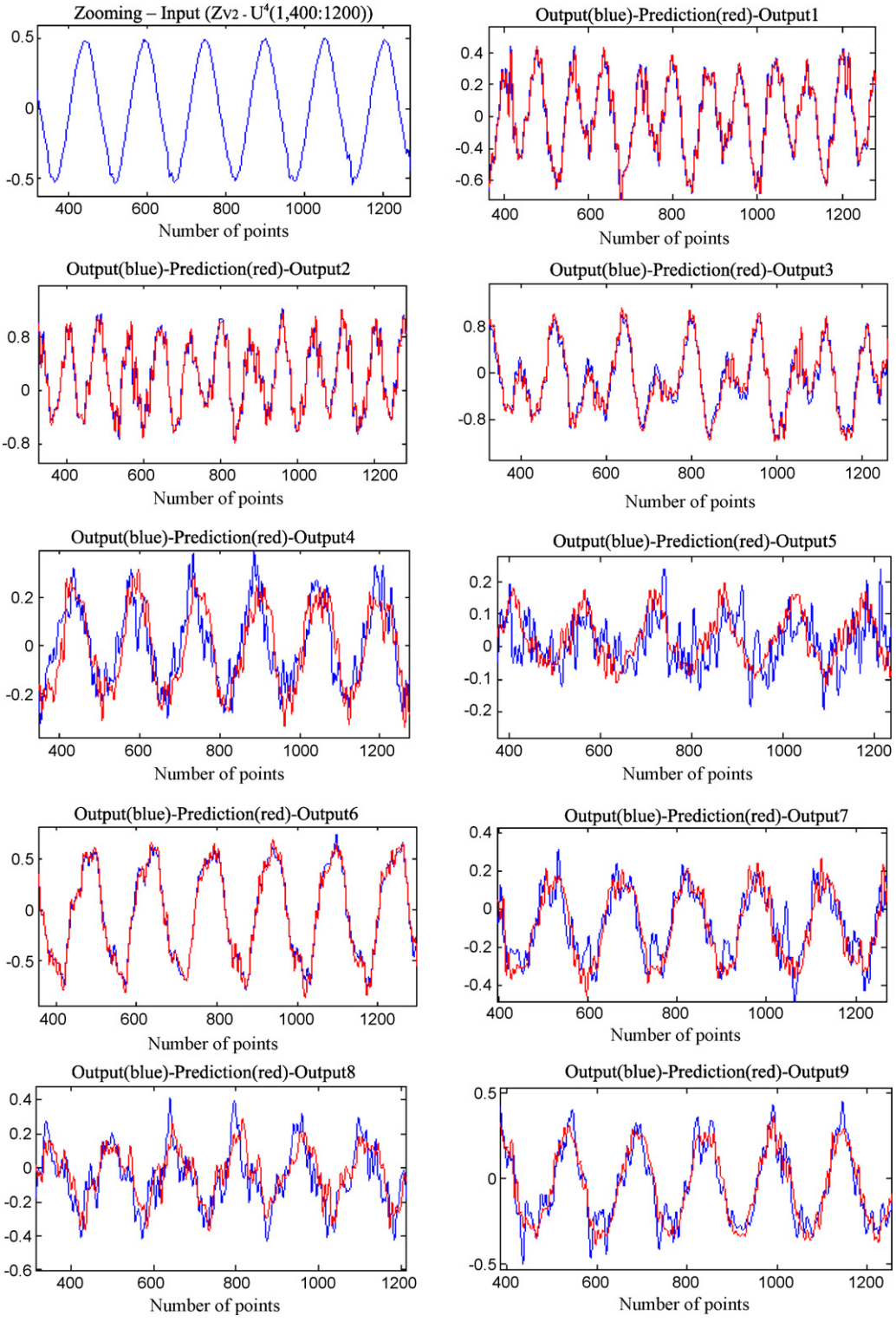

Fig. 15. Visualization of the prediction—sine tests. (For interpretation of the references to color in this figure citation, the reader is referred to the web version of the article.)

Fig. 11 shows how pruning affects the initial network with 18 neurons in the hidden layer. Only 16 neurons and 728 weights are necessary.

3.4. Validate the model

Model validation is performed with a third data set $Z_V(U^3, Y^3)$ different from Z_N and Z_T. The validation approach is based on three analyses: correlations between the predictions and the measurements, visualization and reliability of the predictions.

3.4.1. 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

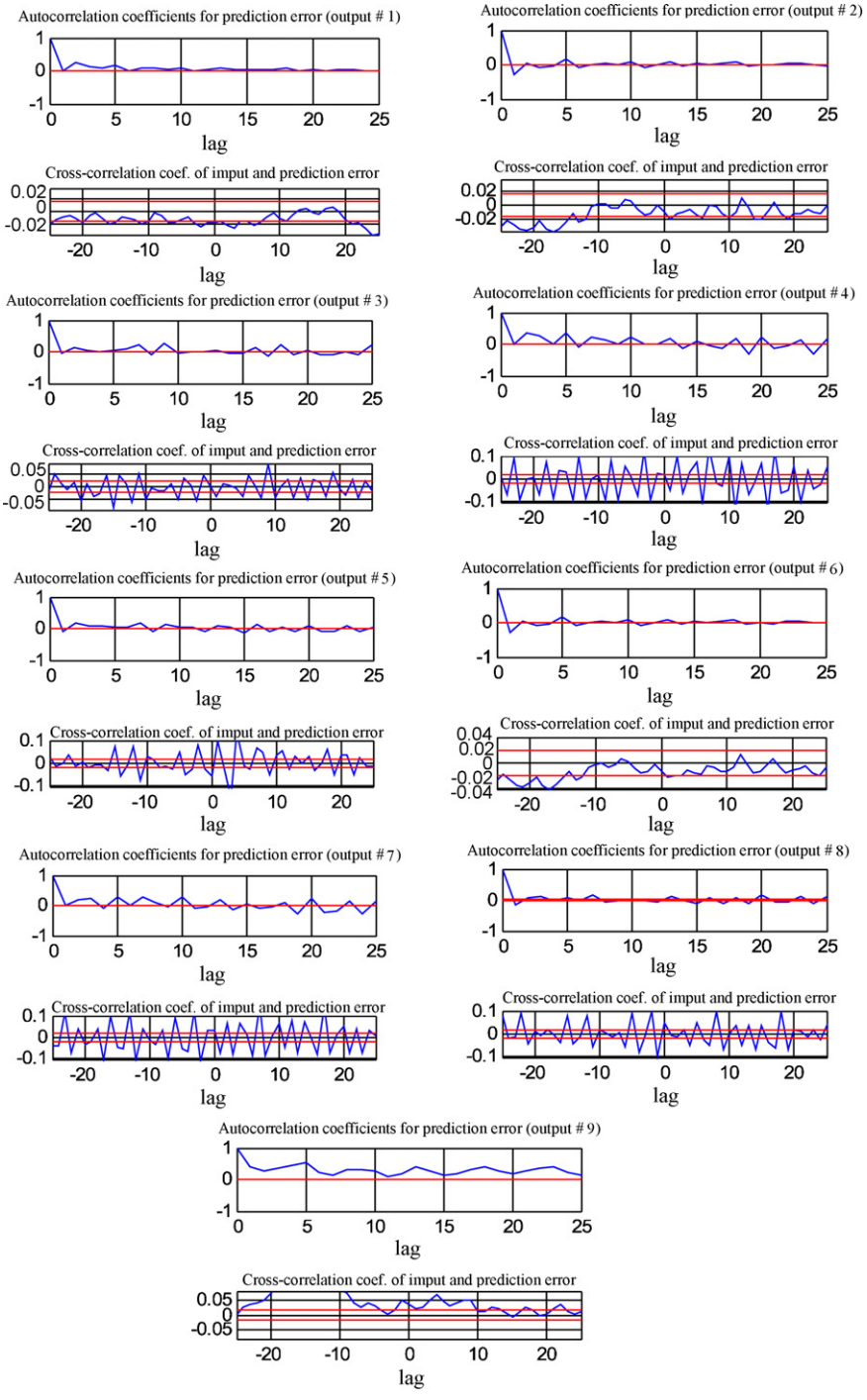

Fig. 16. Correlations for sine tests. (For interpretation of the references to color in this figure citation, the reader is referred to the web version of the article.)

error ε and the cross-correlations $\hat{r}_{U\varepsilon}(\tau)$ between the inputs and the prediction error [40]. The correlations results for this study are shown in Figs. 13 and 16:

$$\hat{r}_{\varepsilon\varepsilon}(\tau) = \frac{\sum_{t=1}^{N-\tau}(\varepsilon(t,\hat{\theta})-\bar{\varepsilon})(\varepsilon(t-\tau,\hat{\theta})-\bar{\varepsilon})}{\sum_{t=1}^{N}(\varepsilon(t,\hat{\theta})-\bar{\varepsilon})^2} = \begin{cases} 1 & \text{if } \tau = 0 \\ 0 & \text{if } \tau \neq 0 \end{cases} \tag{21}$$

$$\hat{r}_{U\varepsilon}(\tau) = \frac{\sum_{t=1}^{N-\tau}(\varepsilon(t-\tau,\hat{\theta})-\bar{\varepsilon})}{\sqrt{\sum_{t=1}^{N}(U(t)-\bar{U})^2\sum_{t=1}^{N}(\varepsilon(t,\hat{\theta})-\bar{\varepsilon})^2}} = 0, \quad \forall\tau \tag{22}$$

3.4.2. Visualization 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 Figs. 12 and 15).

3.4.3. Prediction reliability

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

- Estimation of the prediction error variance compared to the regression vector $\varphi(t)$:

$$\sigma_{\mathrm{P}}^2(t) = E\{\varepsilon^2(t,\hat{\theta})|\phi(t)\} \tag{23}$$

- Estimation of the interval of prediction confidence [41], see Figs. 14 and 17, by using the estimated variance and assuming that the prediction error has a Gaussian distribution:

$$Y(t) \in [\hat{Y}(t|\hat{\theta}) - \sigma_{\mathrm{p}}; \hat{Y}(t|\hat{\theta}) + \sigma_{\mathrm{p}}] \tag{24}$$

4. Results and discussions

For all the figures of this section, the measurement $Y(t)$ is in blue and the prediction $\hat{Y}(t)$ is in red. The validation is realised for two different inputs: random and swept-sine tests vector.

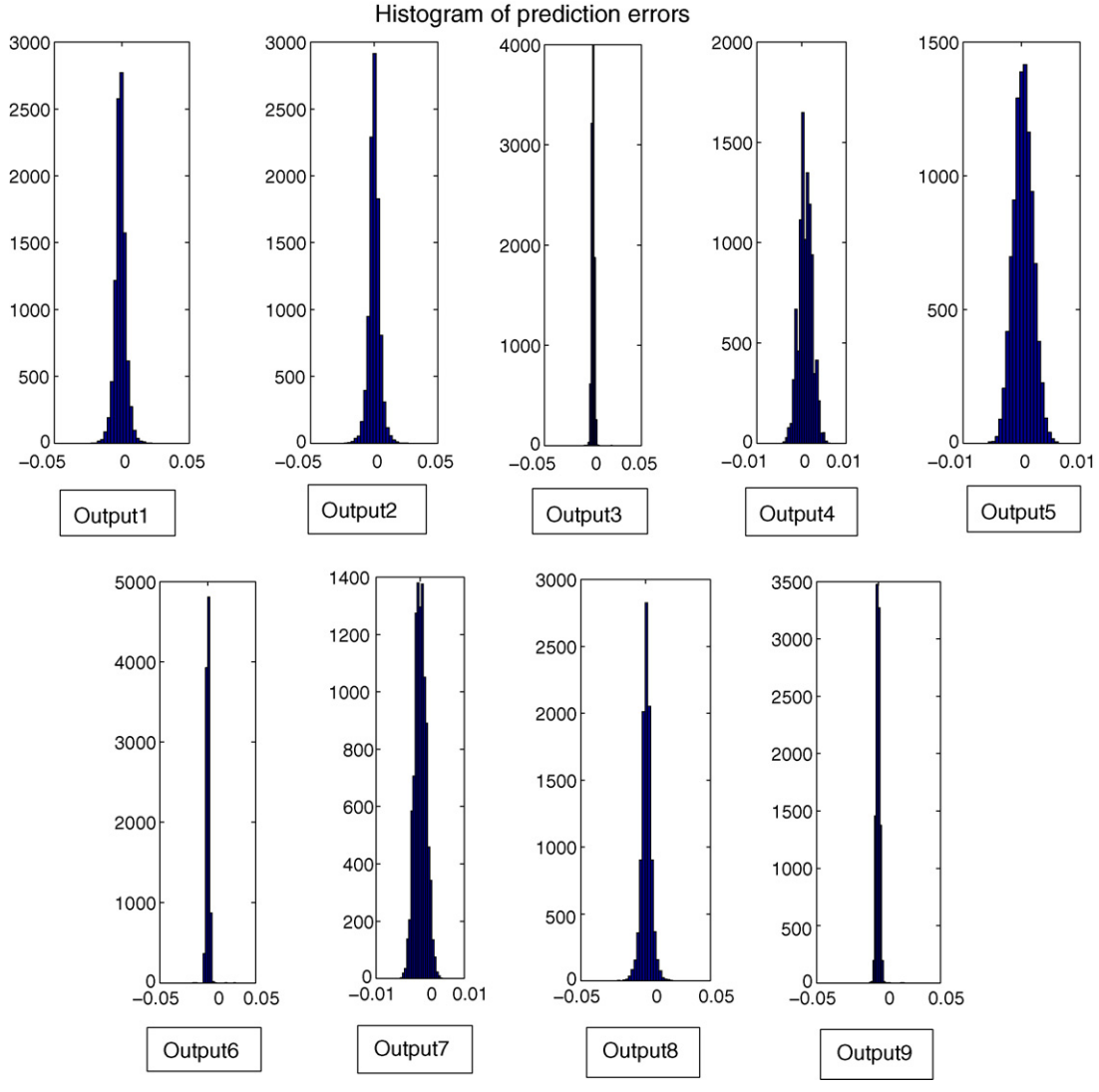

Fig. 17. Histograms for sine tests. (For interpretation of the references to color in this figure citation, the reader is referred to the web version of the article.)

4.1. Random tests

The 9000 samples in $Z_{V1}(U^3, Y^3)$ are issued from random tests. The results are shown in Figs. 12–14.

The comparison between the measured outputs and the one-step ahead predictions shows how the model describes the system dynamic behaviour. Fig. 12 shows the visualization of the nine outputs predictions. It is important to indicate that the prediction is not the same for the nine outputs. There are two reasons for this difference:

- The first reason is related to the fact that outputs 3, 6 and 9 measure the accelerations in the Z axis direction according to the three measurement accelerometers positions. Outputs 1, 4 and 7 measure the X axis direction and outputs 2, 5 and 8 measure the Y axis directions. Therefore, it is expected that the prediction of the outputs which measure the Z axis accelerations (direction of the excitation), is better because the correlations between the excitation and these outputs have higher values than the other outputs.
- The second reason is that the dynamic response of the fuel cell system is not the same in every part of the system. The measurement accelerometers have different positions. Therefore, the amplitudes of the measured responses vary with the accelerometers positions. For example, the outputs 1 (axis X), 2 (axis Y) and 3 (axis Z) for the first three-dimensional accelerometer measure higher acceleration values than the two other accelerometers.

Despite this difference, for all outputs, predictions are 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 all the outputs are shown in Figs. 13 and 14.

As displayed in Fig. 13, the correlation coefficients remain around 1S.D. The autocorrelation functions of all the outputs tend to be zero and the cross-correlation functions vary in a band ranging from -0.05 to 0.05, close to zero. The result means that the prediction error is independent of the control input. So, all the information about the dynamics of the system was incorporated into the model. Another proof is given by the shapes of the histograms shown in Fig. 14. The majority of the ε values are around zero.

4.2. Swept-sine tests

The 5000 samples of $Z_{V2}(U^4, Y^4)$ are issued from swept-sine tests. The results are available in Figs. 15–17.

A good visual prediction is noticed in this section (Fig. 15). The autocorrelation functions are shown in Fig. 16. They converge towards zero and stay within a band around zero. The cross-correlation functions vary from -0.1 to 0.1. The histograms are shown in Fig. 17. They are symmetrical around zero and have the majority of values around ε equal zero.

For the two types of excitations (random and swept sine), predictions are within close range when compared to measurements. However, the accuracy of the predictions is not the same. The results for the swept sine are less good than the random case but are still largely acceptable. It is related to the fact that the training vector is composed from 1520 points coming from random tests for just 560 points coming from swept-sine tests. And for the same kind of excitation, the model predicts better in the Z axis direction (outputs 3, 6 and 9) than the other axes.

5. Conclusion

In this paper, a neural approach, using a multi-layer perceptron combined with 'NNARXM' time regression input vector, is proposed to model the mechanical nonlinear behaviour of a PEM fuel cell system. An experimental setup is designed for this purpose. The mechanical system is vibrated in the three axes direction X–Z with swept sine and random vibrations. The goal is to create models for all three axes. Only the Z axis model is discussed here because of the methodology similarity in all three cases. A MIMO model is trained and validated with data collected from the experimental setup. The obtained results are accurate. This approach is promising because it may be extended to more complex cases. Future work will deal with practical implementations of the black box model that involves:

- The creation of a unique model for all three axes. Such a model, which predicts the mechanical behaviour according to the three axes directions, is very useful to reduce calculations time. In fact, inputs and outputs correlations between different axes permit the reduction of the number of the global neural network neurons. A second advantage is that a global model may be used as a fuel cell system controller. It may be implemented in a real-time system in order to provide an environment to analyse the performance and optimize mechanical parameters design of the PEM fuel system and its auxiliaries. For example, durability tests versus mechanical loads are for sure the most important tests to be carried out. The black box model allows simulating the expected response and then building indicators that are compared in real time with the actual response during tests. A discrepancy warns against a fundamental change in the system as the failure of a component.
- Creating a neural modal that can predict the best accelerometers positions is a major milestone in test preparations. Thus for the preliminary phase, it is very interesting to clamp on the tested mechanical structure the greatest number of available sensors. Indeed, as the knowledge about the system is a priori rather weak, it is difficult to know where to clamp sensors in order to get the most sensitive mechanical response to loads. A Fourier analysis of each sensor response can drive the choice of the sensors number and positions to keep for the main tests. A more interesting method is the possibility to create a neural model that can predict the best sensor positions, eventually not used within the preliminary tests.

References

[1] Y.M. Ferng, Y.C. Tzang, B.S. Pei, C.C Sun, A. Su, Int. J. Hydrogen Energy 29 (2004) 381–391.

[2] R.F. Mann, J.C. Amphlett, M.A.I. Hooper, H.M. Jensen, B.A. Peppley, J. Power Sources 86 (2002) 173–180.
[3] R.N. Methekar, V. Prasad, R.D. Gudi, J. Power Sources 165 (2007) 152–170.
[4] J.J. Baschuk, Li. Xianguo, J. Power Sources 86 (2000) 181–196.
[5] H.H. Voss, D.P. Wilkinson, P.G. Pickup, M.C. Johnson, V. Basura, Electrochem. Acta 40 (1995) 321–328.
[6] T.V. Nguyen, R.E. White, J. Electrochem. Soc. 140 (1993) 2178–2186.
[7] T.F. Fuller, J. Newman, J. Electrochem. Soc. 140 (1993) 1218–1225.
[8] D.M. Bernardi, M.W. Verbrugge, J. Electrochem. Soc. 139 (1992) 2477–2491.
[9] Z.H. Wang, C.Y. Wang, K.S. Chen, J. Power Sources 94 (2001) 40–50.
[10] N. Djilali, T. Berning, Computational Modelling and Simulation of Proton-Exchange Membrane Fuel Cells, ASME Pressure Vessels Piping Div Publ PVP, Vancouver, Canada, 2002, pp. 271–275.
[11] C.Y. Du, P.F. Shi, J. Harbin Inst. Technol. 38 (2006) 1511–1514.
[12] R.P. Pathapati, X. Xue, J. Tang, Renew. Energy 30 (2005) 1–22.
[13] F. Mueller, J. Brouwer, S. Kang, H. Kim, K. Min, J. Power Sources 163 (2007) 814–829.
[14] Y. Shan, S.Y. Choe, J. Power Sources 145 (2005) 30–39.
[15] J. Golbert, D. Lewin, J. Power Sources 135 (2005) 135–151.
[16] W. Na, B. Gou, B. Diong, Proceeding IEEE Industry Applications Conference, 2005, pp. 2937–2943.
[17] S. Yerramalla, A. Davari, A. Feliachi, T. Biswas, J. Power Sources 124 (2003) 104–113.
[18] A. Nasiri, V.S. Rimmalapudi, A. Emadi, D.J. Chmielewski, S. Al-Hallaj, Conference Proceedings IPEMC 4th International Power Electronics and Motion Control Conference, 2004, pp. 491–496.
[19] D.E. Adams, R.J. Randall, Proceedings of the 23rd International Conference on Noise and Vibration Engineering ISMA, Leuven, Belguim, 1998, pp. 517–529.
[20] M. Nørgaad, Neural Network Based System Identification Toolbox, Technical University of Denmark, Lyngby, 2000.
[21] D. Wei, G. Cao, X. Zhu, J. Shanghai Jiaotong Univ. 38 (2004) 1581–1586.
[22] S. Jemei, D. Hissel, M.C. Péra, J.M. Kauffmann, Industrial Electronics-IEEE International Symposium, 2004, pp. 471–476.
[23] M. Hatti, M. Tioursi, W. Nouibat, Industrial Informatics IEEE International Conference, 2006, pp. 1352–1357.
[24] R. Caponetto, L. Fortuna, A. Rizzo, Emerging Technologies and Factory Automation 10th IEEE Conference, 2005, pp. 1–6.
[25] S.O.T. Ogaji, R. Singh, P. Pilidis, M. Diacakis, J. Power Sources 154 (2006) 192–197.
[26] S. Ou, L.E.K. Achenie, J. Power Sources 140 (2005) 319–330.
[27] S. Ou, L.E.K. Achenie, J. Fuel Cell Sci. Technol. 2 (2005) 226–233.
[28] M. Nørgaad, O. Ravn, N.K. Poulsen, L.K. Hansen, Neural Networks for Modelling and control of Dynamic Systems, Springer-Verlag, 2000.
[29] R. Haber, in: H.A. Barker, P.C. Young (Eds.), Proceeding IFAC Symposium on Identification and System Parameter Estimation, York, UK, 1985, pp. 409–414.
[30] X. He, H. Asada, Proceeding of the American Control Conference, San Francisco, CA, 1993, pp. 2520–2523.
[31] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice-Hall, 1998.
[32] M. Salgado, G. Goodwin, R. Middleton, Int. J. Control 47 (1988) 477–491.
[33] B. Hassibi, D.G. Stork, in: S.J. Hanson, J.D. Cowan, C.L. Giles (Eds.), Advances in Neural Information Processing Systems 5, Proceedings of the 1992 Conference, San Mateo, 1993.
[34] L. Ljung, System Identification—Theory for the User, 2nd ed., Prentice-Hall, Upper Saddle River, 1999.
[35] D. Marquardt, SIAM J. Appl. Math. 11 (1963) 164–168.
[36] R. Fletcher, Practical Methods of Optimisation, 2nd ed., Wiley and Sons, 1987.
[37] J. Larsen, L. Hansen, Proceeding of the IEEE Workshop on Neural networks for Signal Proceeding IV, Piscataway, NJ, 1994.
[38] Y. Le Cun, J. Denker, S. Solla, in: D.S. Touretzky (Ed.), Neural Information Processing Systems, Denver, 1990.
[39] M.W. Pedersen, L.K. Hansen, in: G. Tesauro, D. Touretzky, T. Leen (Eds.), Advances in Neural Information Processing systems, The MIT Press, 1995.
[40] L.K. Hansen, J. Larsen, Adv. Comput. Math. 5 (1996) 269–280.
[41] O. Sørensen, Eur. J. Control 2 (1996) 36–43.