Soft-computing techniques in FE model updating

Bartosz Miller*
Department of Structural Mechanics, Rzeszow University of Technology
ul. W. Pola 2, 35-959 Rzeszow, Poland
e-mail: bartosz.miller@prz.edu.pl

Abstract

The paper presents the application of Artificial Neural Networks (ANNs) for finite element (FE) models updating. The investigated structures are beams and frames, their models are being updated by ANNs with input vectors composed of dynamic characteristics of structures, measured on laboratory models. The ANNs (multi layer feed-forward networks and Bayesian neural networks) are trained on numerical data disturbed by an artificial noise, the calibration of the FE models is performed by trained ANNs with measurement data on input, the updated models are verified by the comparison of their results with laboratory measurements.

Keywords: artificial neural networks, updating, dynamics, vibrations, identification

1. Introduction

1.1. Problem description

The nowadays progress as well in in the development of new computational tools as in the construction of new, powerful computer hardware makes it possible to build very complicated dynamic models. Unfortunately, even those very complicated models are sometimes not able to handle the present-day requirements. In some particular cases it is necessary to employ models which are able to simulate or reproduce the behaviour of real structures with very high accuracy. In order to minimize the differences between the results of experimental measurements and numerical simulations some changes could be introduced to the model. The introduction of those changes is called *updating* [2].

This paper presents the updating of computational models on the basis of limited number of measurement data. As a tool for the model updating different types of Artificial Neural Networks (ANN) are applied. The comparison of the results of numerical simulations and experimental measurements is assessed herein by Root Mean Square Error (RMSE^n), Maximal Relative Error (MRE^n) and/or Average Relative Error (ARE^n), all calculated for the first n eigenfrequencies.

1.2. Standard artificial neural networks

The type of standard ANN adopted in this study is called Multi Layer Perceptron (MLP) [3]. The calibration of the MLP to the present purpose ("training" or "learning" procedure) is based here on P vector pairs called "patterns" and consisting of vector of selected FE model parameters $\mathbf{t}$ and corresponding eigenfrequencies vector $\mathbf{x} = \mathrm{FEM}(\mathbf{t})$, obtained by means of FEM and, as in the paper, disturbed by an artificial noise mimicking measurements errors. The computation of the optimal vector of network weights $\mathbf{w}*$ can be formulated as follows:

$$\mathbf{w}* = \arg\min_w \{\Omega(\mathbf{w})\}\ , \tag{1}$$

where:

$$\Omega(\mathbf{w}) = \frac{1}{2P}\sum_{j=1}^{P}\|\mathbf{t}_j - \mathbf{y}_j\| = \frac{1}{2P}\sum_{j=1}^{P}\|\mathbf{t}_j - \mathrm{MLP}(\mathbf{x}_j,\mathbf{w})\| \tag{2}$$

and $\mathbf{y}_j = \mathrm{MLP}(\mathbf{x}_j, \mathbf{w})$ is the approximation of vector $\mathbf{t}_j$ given by MLP.

1.3. Bayesian neural networks

Besides the traditional MLP also very promising, new type of neural networks, namely Bayesian Neural Networks (BNN) [1], are utilized in this paper. BNNs are probabilistic networks based on *the Bayes' theorem*:

$$p(\mathbf{w}|\mathbf{t},\alpha,\beta) = \frac{p(\mathbf{t}|\mathbf{w},\beta)p(\mathbf{w}|\alpha)}{p(\mathbf{t}|\alpha,\beta)}, \tag{3}$$

where the denominator in Eq. 3, called *evidence* (or *marginal likelihood*), is defined as the following integral:

$$E = p(\mathbf{t}|\alpha,\beta) = \int\limits_{\mathcal{R}^W} p(\mathbf{t}|\mathbf{w},\beta)p(\mathbf{w}|\alpha)d\mathbf{w}. \tag{4}$$

The maximum of the logarithm of evidence with respect to the vector $\mathbf{w}$ is evaluated

$$\max_{\mathbf{w}}|\ln E = \max_{\mathbf{w}}|\ln p(\mathbf{t}|\alpha,\beta) \tag{5}$$

and a new criterion called *Maximum Marginal Likelihood (MML)* is formulated, see references in [6]. This criterion can be applied for design of network architecture instead of MLP minimization by the mean square error paradigm [5, 6].

In BNNs the overfitting phenomenon is controlled by a regularization term $\frac{\alpha}{2}\sum_{i=1}^{W}\mathbf{w}_i^2$ introduced into the network cost function:

$$F(\mathbf{w}) = \frac{\beta}{2}\sum_{j=1}^{M}\{\mathbf{t}_j - \mathbf{y}_j)\}^2 + \frac{\alpha}{2}\sum_{i=1}^{W}\mathbf{w}_i^2\,. \tag{6}$$

In so-called *Semi*-Bayesian Neural Networks (SBNNs) the *hyperparameters* α and β may be computed during network learning in the iterative way, while in *True*-Bayesian Neural Networks (TBNNs) they have to be *marginalized* (the integrals over those parameters have to be computed). In the paper both types of BNNs are utilized.

*Financial support by the Polish Ministry of Science and High Education, grant No. **N N506 432636**, is gratefully acknowledged

Table 1: The updated FE model parameters and eigenfrequencies prediction accuracy

	k_1	k_2	k_3	H	Δf_1	Δf_2	Δf_3	Δf_4	...	Δf_7	$RMSE^7$	MRE^7	ARE^7
		[Nm/rad]		[cm]			[%]				[Hz]	[%]	[%]
MLP	1004.3	1003.7	2497.1	4.3	0.0	0.0	0.2	4.1	...	-4.9	8.36	4.94	1.70
SBNN	1919.1	2964.3	2606.1	2.9	-0.1	-0.0	0.1	4.9	...	-4.6	7.94	4.93	1.75
1-O MLP	1582.4	3405.8	2668.3	2.6	1.0	-0.1	0.0	6.0	...	-4.4	7.80	6.04	1.99
TBNN	1923.6	2790.3	2846.5	2.9	-0.3	-0.3	-0.4	5.0	...	-4.7	8.15	4.98	1.93

1.4. *The neural network updating procedure*

The paper presents the updating of some computational models of engineering structures. The updating procedure consists of the following steps: (1) the preparation of learning and testing patterns, each of them is composed of the numerically computed (by means of FEM) response of the structure obtained for the particular values of model parameters which are being updated (input vector $\mathbf{x}$) and the parameters themselves (target vector $\mathbf{t}$), (2) disturbance of the input vectors $\mathbf{x}$ by the artificial, random noise in order to mimic the measurements errors, (3) training of the ANN to predict the values of model parameters using the response of the structure as input information, (4) the application of experimental structure response to ANN in order to predict updated values of model parameters, (5) the testing of the updated model by the comparison of responses calculated using the updated model and measured on a real structure.

2. The FE model updating examples

The investigated examples are beams and one- and two-storey frames. This paper presents the updating of the two-storey frame FE model only, all the previous examples are described in [4].

The investigated structure is two-storey frame made from aluminium alloy. The height of the frame is $40cm$, the width is $46.9cm$, beam and columns are of a rectangular cross-section $2.6cm$ by $0.6cm$. The columns are bolted to steel plates mounted in the ground The laboratory model is presented in Fig. 1a, the investigated FE model is shown in Fig. 1b.

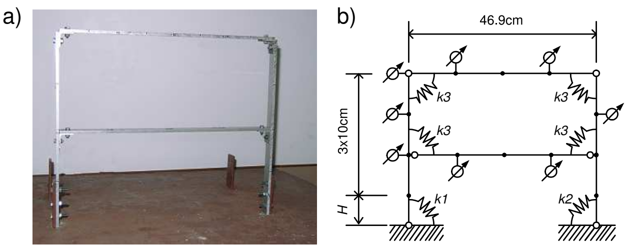

Figure 1: Portal frame: a) laboratory model, b) FE model

All the eigenfrequencies obtained from FE simulations are disturbed by the artificial, random noise with mean $\mu = 0.0$ and standard deviation $\sigma = 0.1Hz$ (what corresponds with the measurements accuracy possible to obtain using the equipment at the disposal).

The input vector is composed of model eigenfrequencies: $\mathbf{x} = \{f_1, f_2, \ldots, f_n\}$, $n = 1, 2, \ldots, 7$, and it had n elements. The output vector consists of FE model parameters being updated: $\mathbf{y} = \{k_1, k_2, k_3, H\}$ (see Fig. 1b). The architecture of MLP applied is $n - h - 4$, where $h = 2, 3, \ldots, 30$. In each case MLP is trained on numerical patterns to update the FE model parameters and after training the parameters are identified using measured eigenfrequencies as MLP. The ARE^7 error of the first seven eigenfrequencies is calculated (even for $n < 7$, when less than seven eigenfrequencies are involved during the updating procedure) and the analysis of its mean versus different n clearly shows, that in order to minimize the error of prediction of the first seven eigenfrequencies the updating should be preformed with only the first three eigenfrequencies involved ($n = 3$). When the number of eigenfrequencies involved is higher the model adjusts better to higher eigenfrequencies (mainly the fourth and the seventh) but the overall updating accuracy decreases.

Table 1 presents the results from various types ANN updating parameters of FE model. The network applied are MLP, SBNN trained using Scaled Conjugate Gradients (SCG) algorithm (both with four element output vector), and, finally, MLP and TBNN with one output (four independent networks, each with one output).

3. Final remarks

The paper presents the application of ANN, both standard MLP and probabilistic BNN, in the updating of dynamic models of engineering structures. The ANN procedure is able to deal at the same time with the data obtained both from numerical simulations and experimental measurements. Future works should include wider examination of the influence of the artificial noise on the results obtained, the inclusion of the eigenforms in both the ANN input vector and the estimation of the accuracy of obtained results.

References

[1] Bishop C.M., *Pattern Recognition and Machine Learning*, Heidelberg: Springer, 2006.

[2] Friswell M.I., Mottershead J.E., *Finite Element Model Updating in Structural Dynamics*, Kluwer Academic Publishers, Dordrecht, 1996.

[3] Haykin S., *Neural Networks: a Comprehensive Foundation*, Prentice Hall International, Inc., 2^{nd} ed., 1999.

[4] Miller B., Ziemiański L., FE model updating using eigenfrequencies and eigenforms, in K Dems, D Gawin, M Lefik, Z Wieckowski, ed., *CMM-2007 Computer Methods in Mechanics, Short Papers*, 271–272, 2007.

[5] Waszczyszyn Z., Ziemiański L., Neural Networks in Mechanics of Structures and Materials — New Results and Prospects of Applications, *Computers&Structures*, 79(2001):2261–2276, 2001.

[6] Waszczyszyn Z., Słoński M., Maximum of marginal likelihood criterion instead of cross-validation for designing, In: Rutkowski L. et al., ed. *Artificial Intelligence and Soft Computing ICAISC2008*, Berlin-Heidelberg-New York: Springer, 2008.