Non-parametric modelling of a rectangular flexible plate structure

Intan Z. M. Darus*, Ali A. M. Al-Khafaji

Department of System Dynamics & Control, Faculty of Mechanical Engineering, Universiti Teknologi Malaysia (UTM), 81310 Skudai, Johor Bahru, Malaysia

A R T I C L E   I N F O

Article history:
Received 20 October 2010
Received in revised form 15 August 2011
Accepted 1 September 2011
Available online 29 September 2011

Keywords:
Active vibration control
ANFIS
Flexible plate
Neural network
System identification

A B S T R A C T

This research investigates the performance of dynamic modelling using non-parametric techniques for identification of a flexible structure system for development of active vibration control. In this paper, the implementation details are described and the experimental studies conducted in this research are analysed. The input–output data of the system were first acquired through the experimental studies using National Instruments (NI) data acquisition system. A sinusoidal force was applied to excite the flexible plate and the dynamic response of the system was then investigated. Non-parametric modelling of the system were developed using several artificial intelligent methodologies namely Adaptive Elman Neural Networks (ENN), Backpropagation Multi-layer Perceptron Neural Networks (MLPNN) and Adaptive Neuro-Fuzzy Inference System (ANFIS). The performance of all these methodologies were compared and discussed. Finally, validation and verification of the obtained model was conducted using One Step Ahead (OSA) prediction, mean squared error (MSE) and correlation tests. The prediction ability of the model was further observed with unseen data. The results verified that the MLPNN converge to an optimum solution faster and the dynamic model obtained described the flexible plate structure very well. The non-parametric models of the flexible plate structure thus developed and validated will be used as the representation of the transfer function of the system in subsequent investigations for the development of active vibration control strategies for vibration suppression in flexible structures.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

In the past three decades, the use of flexible structure systems has been growing quickly in many engineering applications. The elements for flexible structure such frames, shells, beams and plates are extensively used in a wide range of manufacturing applications and particularly in mechanical, civil, marine, aeronautical, aerospace and other areas of practical attention, for example, flexible manipulators of satellites, solar panels, etc.

Plates with different shapes, boundary conditions at the edges and various complicated effects have often found applications in different structures such as aerospace, machine design, telephone industry, nuclear reactor technology, naval structures and earthquake-resistant structures. Particularly, the dynamic behaviour of flexible, flat, thin, rectangular plates has received huge attention in recent years because of its technical importance (Chakraverty, 2009).

The flexible thin rectangular plates structures are the most commonly used in the industrialised world and in a broad range of engineering applications, for examples, electronic circuit board design, solar panels and bridge decks. The stability of the plate, where it is subjected to loading, would be associated with a range of physical effects that lead to high vibration. The high vibration of flexible structure systems cause noise, fatigue, wear, destruction, human discomfort and reduced system effectiveness. That is why the vibration of flexible structure needs to be controlled. Due to its multiple practical problems and applications, the vibration of the elastic plates has been treated widely from researchers with different boundary conditions, both from theoretical and experimental points of view (Chakraverty, 2009). It is necessary to find an approximate or accurate model of the plate structure to control the vibration of a plate well. Suitable modelling of a dynamic system, for instance a flexible plate, would result in good control (Tavakolpour et al., 2010).

In the initial stages, results were available for some simple cases, namely a limited set of boundary conditions and geometries, in which the analytical solution could be found. With the advent of fast computers and various efficient numerical methods, there has been a big increase in the amount of research done for getting better accuracy in the results. Numerical methods offer reasonable and accepted solution but with complex shapes of plate sometimes lead to inaccuracies and more computing time (Chakraverty, 2009).

To predict the physical system behaviour under different operating conditions or to control it, a model can be created using an approach called system identification.

*Corresponding author. Tel.: +60167616716.
E-mail address: intan@fkm.utm.my (I. Z. M. Darus).
In the present decade, system identification techniques have become potential candidates to many control application. Parametric and non-parametric system identification methods used to find approximate or accurate models of dynamic systems depend on observed inputs and outputs (Mat Darus, 2004). The major aim of system identification is to locate approximate or accurate models of dynamics systems depend on observed inputs and outputs. A number of researchers have applied techniques to solve the problems related to system identification. Several methods have been devised to find out models that describe input output behaviour of a system well (Ismail et al., 2006a).

Ismail et al. (2006a) have reported identification algorithms of flexible structure using Neural Networks. The research reported a study into the development of system identification methods for dynamic modelling and characterisation of flexible plate structures. The research uses Least Squares and Recursive Least Squares to find linear parametric model of the system. In addition, non-parametric models of the system are developed using Elman Neural Networks (ENN) and Multi-layer Perceptron Neural Networks (MLPNN; Ismail et al., 2006a).

Mohd Hashim et al. (2004) have reported non-linear dynamic modelling of flexible beam structures using Neural Networks. The research investigated the utilisation of neural network (NNs) backpropagation for modelling flexible beam with fixed-free mode. Comparative analysis of the performance of the Recursive Least Squares scheme and intelligent Neural Networks model in characterizing the system was carried out in the frequency and time domains. Simulated results have shown that by Neural Networks the system is modelled better than with the conventional linear modelling method (Mohd Hashim et al., 2004).

Mat Darus et al. (2008) have reported Adaptive Neuro-modelling of a twin rotor system. The research investigated the utilisation of Adaptive Neural Networks (NNs) for dynamic modelling and identification of a highly non-linear TRMS system. An adaptive Elman neuro-model is designed to characterise a twin rotor multi-input multi-output system (TRMS) in vertical motion based on one step-ahead (OSA) prediction. The results obtained, in both frequency and time domains, are compared to the identification using the conventional adaptive technique of Recursive Least Squares (RLS). Simulations indicate the superiority of an adaptive neuro-modelling technique over RLS algorithm in modelling and identification of the TRMS (Mat Darus et al., 2008).

Ismail et al. (2006b) have reported dynamic characterisation of flexible vibrating structures using adaptive neuro-fuzzy inference system (ANFIS). In this research ANFIS was used to develop a model characterizing the vibration of the plate. The input/output data used in this research was obtained from a simulation of a square, flat, flexible plate with all edges clamped using finite difference (FD) algorithm (Ismail et al., 2006b).

Toha et al. (2008) have reported ANFIS modelling of a twin rotor system. An Adaptive Neuro-Fuzzy Inference System (ANFIS) network design is deployed and used for modelling a twin rotor multi-input multi-output system (TRMS). It is demonstrated experimentally that ANFIS can be effectively used for modelling the system with highly accurate results. The accuracy of the modelling results is demonstrated through validation tests including training and test validation and correlation tests (Mat Darus and Tokhi, 2006).

System identification is a broad idiom used to describe algorithms and mathematical tools that build dynamical models from measured data. Over the last two decades system identification has received a lot of attention. System identification methods are widely used as a fundamental requirement in scientific applications and engineering. The practical application domains include the Boolean function generation, symbolic regression and pattern recognition and time-series prediction. The problem of finding an approximate or accurate model for dynamical systems occurs often in engineering applications. System identification is one way to solve this problem (Ismail et al., 2006a).

The major aim of system identification is to find approximate or accurate models of dynamic systems depend on observed inputs and outputs. When a model of the physical system is obtained, it can be used for solving different problems; such as to predict its behaviour under different operating conditions or control the physical system. Numerous researchers have applied techniques to solve the system identification problems. A number of methods have devised to obtain models that best describe input output behaviour of a system.

The reason of this study is to develop a model characterizing vibration of two-dimensional flexible rectangular plate structures using non-parametric identification techniques as soft computing. In this work, a thin rectangular plate with all edges clamped is considered. Prior to this, a dynamic model of the plate structure based on laboratory experiments characterizing the flexible plate structure is developed. Finally, the validity of the obtained model was investigated using correlation tests. The procedure of system identification can be represented as shown in Fig. 1.

2. Experimental setup

From theoretical and experimental points of view, the vibration of the flexible plates has been treated extensively. The vibration of a plate can be excited and detected with a suitable experimental setup. Accurate understanding of the results allows us to achieve useful information. Several researchers made experimental studies with different types of setup and instrumentation to measure the vibration parameters and to control the plate (Shimona et al., 2005; Shimon and Hurmuzlu, 2007; Qiu et al., 2009).

In this investigation, the input–output data of the system were first acquired through the experimental studies using National Instruments (NI) data acquisition system. To provide experimental data, a rectangular plate with dimensions of 1 m × 1.5 m × 0.003 m was investigated. To allow a 0.04 m width clamped boundary on all four sides, the rectangular plate was cut as 1.58 m × 1.08 m. The experimental arrangement developed for this study was established as shown in Fig. 2.

To acquire the perfect conditions of clamped boundaries, four steel bars with rectangular cross sections were used to clamp the plate edges giving 40 mm as the thickness of the clamped
boundaries. The opposing faces and the inside edges of each bar were milled flat and straight. The plate with the clamped boundaries was then fixed firmly to a main frame. To tie up the frame pieces jointly, bolts of type M9 were used, with shoulder bolts being used to ensure precise alignment of the opposing frame sections upon final assembly. The test structure was excited by a sinusoidal force generated by a magnetic shaker and applied at the excitation point as shown in Fig. 3. To sense the plate response at the desired detection and observation points, a piezo-beam type accelerometer (Kistler-8636C5) with sensitivity of 1004 mV/g was used as shown in Fig. 4. For direct connection of the piezo-type accelerometers, the acceleration signal was acquired through a National Instruments (NI) compact-data acquisition unit as shown in Fig. 5, which is equipped with NI-9234 module (with 24-bit resolution) as shown in Fig. 6. The necessary signal conditioning circuits such as anti-aliasing filter have been built-in into the data acquisition system. The acquired signal was then analysed using Intel Core TM Duo Processor and LabVIEW software.
This research used the NARX model. Mathematically the model is represented in the NARX form (Mat Darus, 2004; Lennart, 1999). It is considered as additive term at the output, the model can be identified as a suitable learning algorithm. If the model is acceptable, which can be constructed through non-parametric methods with values of the system input vector, output vector and noise. 

$y = f(u(t-1), \ldots, u(t-n_u), y(t-1), \ldots, y(t-n_y), e(t-1), \ldots, e(t-n_e))$  \hspace{1cm} (1)

where $\dot{y}$ represents the output vector determined by the past values of the system input vector, output vector and noise. $n_u$, $n_y$ and $n_e$ represent model orders. $f$ represent the system mapping, which can be constructed through non-parametric methods with a suitable learning algorithm. If the model is acceptable to identify the system without noise term incorporated or the noise is considered as additive term at the output, the model can be represented in the NARX form (Mat Darus, 2004; Lennart, 1999). This research used the NARX model. Mathematically the model in Eq. (1) can be written in discrete form as in Eq. (2):

$y = f(u(k-1), \ldots, u(k-n_u), y(k-1), \ldots, y(k-n_y)) + e(k)$  \hspace{1cm} (2)

### 4. Non-parametric identification

For non-parametric estimation, most popular members of non-parametric models such as fuzzy logic (FL) and neural networks (NNs) are normally utilised. Neural networks have a variety of attractive characteristics such as generalisation ability, distributed representation and computation, adaptability massive parallelism and inherent contextual information processing. NNs used widely in a range of identification and control applications owing to the proficient nature of their working principles and other attractive features. Amongst the diverse types of NNs, the Radial-Basis Function (RBF), Multi-Layered Perceptron Neural Network (MLPNN), and Elman recurrent Neural Network (ENN) are usually used in identification and control of dynamic systems. The most attractive applications propose a suitable grouping of two methods, NN and FL resulting in a hybrid system, ANFIS, which both operate on linguistic descriptions of the variables and the numeric values through a parallel and fault tolerant architecture (Mat Darus, 2004).

### 4.1. Artificial Neural Networks

An Artificial Neural Network is composed of a massive amount of extremely interconnected processing elements called neurons working as one to solve specific problems. It is an information processing concept which is stimulated by the way biological nervous systems, for instance the brain. Artificial Neural Networks, like people, learn by example. An Artificial Neural Network is constructed for a specific application, for instance data classification or pattern recognition, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons (Aleksander and Morton, 1990).

#### 4.1.1. Structure of Neural Networks (NNs)

In Neural Networks, nodes or computational models are joined through weights that are modified during use to improve performance. The key idea is to obtain best performance through large interconnection of simple computational elements. The simple node provides a linear combination of $N$ weights $w_1, w_2, \ldots, w_N$ and $N$ inputs $x_1, x_2, \ldots, x_N$ and passes the result through a nonlinearity $\Phi$, as shown in Fig. 7. Models of NNs are specified by the net topology, node characteristics and training or learning rules. NNs are specified by:

1. **Node**: normally a sigmoid function.
2. **Layer**: a set of nodes at the same hierarchical level.
3. **Connection**: constant weights or weights as a linear dynamical system, feedforward or recurrent.
4. **Architecture**: an arrangement of interconnected neurons.
5. **Mode of operation**: analogue or digital.

From Fig. 7, the equation $y = \Phi(\sum x_i w_i + w_0)$ is a mathematical explanation of a neuron where the input vector is given by $x = [x_1, x_2, \ldots, x_N]^T$ while $w = [w_1, w_2, \ldots, w_N, w_0]^T$ represents the weight vector of a neuron.

#### 4.1.2. Multi-layer Perceptron Neural Networks (MLPNN)

MLPNN is considered as possibly the most frequently used member of the neural network family. The major reason for this is its capability to model simple functional relationships in addition to very complex one. This has been demonstrated throughout a large amount of practical applications including system identification, speech and natural language processing, prediction and control, function approximation and pattern recognition. An MLPNN is able to representing the Boolean functions and forming arbitrary decision boundaries (Demuth et al., 1992).

A MLPNN needs a set of data, to be presented as inputs to the input node element layer. The outputs from this layer are fed to

![Fig. 6. NI-9234 module.](image)

![Fig. 7. Connectives within a node (Mat Darus, 2004).](image)
the first hidden layer as weighted inputs, and subsequently the outputs from the first layer are fed, as weighted inputs, to the second hidden layer. This process continues until the output layer is reached. In these networks it is assumed that the network can be made up of any number of layers with reasonable number of neurons in each layer, based on the nature of the particular application. A generalised structure of MLP with its basic function is shown in Fig. 7.

The input layer is formed from one layer of nodes and then the second layer of nodes forms the output layer, with a number of intermediate or hidden layers existing between them. In detail, the input layer is the layer to which the input data is supplied and intermediate or hidden layers existing between them. All other intermediate layers are called hidden layers. Usually, one, two or even no hidden layers are employed. Fig. 8 depicts other intermediate layers are called hidden layers. Usually, one, two or even no hidden layers are employed. Fig. 8 depicts $m$ inputs and $m$ outputs, it is not necessary for these values to be equal. The layers are completely interconnected, that means each neuron is connected to every neuron in the previous and succeeding layers. However, the neurons in the same layer are not connected to each other. A neuron performs two functions; combining and activation. Several type of activation function such as sigmoid, piecewise linear, threshold, Gaussian and tan-sigmoid are used for activation.

The backpropagation (BP) algorithm, which is the most commonly adopted MLP learning algorithm, is a gradient descent algorithm. The design of the BP learning algorithm for the MLPNN is a landmark in the development history of neural networks. Actually, the powerful properties of neural networks have been well recognised after the introduction of BP learning algorithm. The backpropagation algorithm was created by generalising the Widrow and Hoff learning rule to multiple layer networks and non-linear differentiable transfer functions. The network is trained, using input vectors and the corresponding target or output vectors, until it can associate input vectors with specific output vectors, approximate a function or classify input vectors in an appropriate way as defined by the user. Standard backpropagation is a gradient descent algorithm, as is the Widrow and Hoff learning rule, in which minimising the sum squared error between the actual output and desired output. Through the BP learning algorithm the mean squared error of the network is minimised by continually adjusting the weights and biases in the direction of the steepest descent with respect to the error. This is known as gradient descent procedure. Any function with a finite number of discontinuities can be approximated using Networks with biases, with at least one sigmoid neuron layer, and a linear output neuron layer. The best number of neurons and hidden layers to be selected in NN can be obtained by a simple trial and error or by optimisation technique. Correctly trained BP networks tend to give reasonable answers when presented with inputs that they have never seen. Typically, a new input leads to an output similar to the correct output for input vectors used in training that are similar to the new input being presented. This generalisation property makes it possible to train a network on a representative set of input target pairs and get good results without training the network on all possible input output pairs (Shaheed and Tokhi, 2001). Rules of the backpropagation algorithm for the connection weights between hidden and output layers are clearly described in Shaheed and Tokhi (2001).

This research studies the utilisation of backpropagation MLPNN for modelling a single-input single-output flexible plate system. Fig. 9 shows the diagrammatic representation of the neural network algorithm.

### 4.2. Elman Neural Network (ENN)

Recurrent Neural Networks (RNNs) are the classes of Neural Networks which contain cycles or feedback connections. An RNN can take arbitrary topology as any node in the network may be linked with any other node (including itself), while the set of topologies of feedforward networks is fairly constrained. The recurrent network developed by Elman has a simple architecture; this network has been proved to be effective for modelling linear systems not higher than the first order (Mandic and Chambers, 2001). Elman networks are two-layer backpropagation networks with the addition of a feedback connection from the output of the hidden layer to its input. This feedback path allows Elman networks to recognise, to learn and to generate temporal patterns, as well as spatial patterns.

Actually, the Elman neural network comprises four layers, namely the input layer, hidden layer, output layer and context layer that can store internal states. So, it belongs to special type of feedforward neural network with additional memory neurons and local feedback (Ismail et al., 2006a). Fig. 10 shows the architecture of an Elman neural network.

![Fig. 9. Diagrammatic representation of the neural network modelling algorithm.](image)

![Fig. 10. Structure of the Elman Neural Networks model.](image)
Symbol $z^{-1}$ represents unit delay. It can be seen from Fig. 10 that in Elman Neural Networks Model, besides the input layer, hidden layer, and output layer, also exists a context layer. Each two adjacent layers are adjusted by connection weights. The input and output layers interact with the outside environment, while the hidden and context layers do not. The input layer is only buffer layer that passes the signals without changing them. The output layer is linear and it sums the signal fed to it. The hidden layer can have linear or non-linear activation functions. The context layer is used only to memorise the previous activations of the hidden layer and can be considered to function as one-step delay. Generally, it can be considered as a special type of feedforward neural network with additional memory neurons and local feedback. The distinct self-connections of the context nodes in Elman network make it sensitive to the history of input data, which is essentially useful in modelling dynamic system (Pham and Liu, 1995).

The research reported in (Ismail et al., 2006a) investigated the utilisation of backpropagation Elman Neural Networks for modelling a SISO flexible, flat, plate system. The diagrammatic representation of an Elman neural network for system identification is similar to MLPNN algorithms as shown in Fig. 9.

4.3. Adaptive Neuro-Fuzzy Inference System (ANFIS)

Neural networks and fuzzy inference systems are the most popular members of the non-parametric methods. Fuzzy logic based mechanisms employ the verbal power whereas neural networks provide the mathematical power of the brain. The largely attractive applications offer a suitable combination of these two methods resulting in a hybrid system. The hybrid system operates on both linguistic descriptions of the variables and the numeric values through a parallel and fault tolerant architecture (Mat Darus, 2004).

The term ANFIS, created by Jang, 1993 represents Adaptive-Network-Based Fuzzy Inference System. It is classified as a hybrid neuro-fuzzy model, constructed by combination of a fuzzy system and a neural network into a uniform architecture (Nauck, 1999). ANFIS can integrate human expertise in addition to adapt itself through repeated learning. This architecture has verified a high performance in many applications (Toha et al., 2008).

The network-type structure of ANFIS is similar to that of a neural network; it can be used to interpret the input/output map. ANFIS maps inputs through input membership functions and associated parameters, and then through output membership functions and associated parameters to outputs.

4.3.1. ANFIS architecture

ANFIS architecture uses Sugeno-type fuzzy system or Takagi and Sugeno’s IF-THEN rules with appropriate membership functions implant into adaptive networks. Fig. 11 presents type-3 fuzzy reasoning where Takagi and Sugeno’s if-then rules are used (Ismail et al., 2006b). Fig. 12 depicts the equivalent type-3 fuzzy reasoning ANFIS architecture for system with two inputs $x$ and $y$ (Jang, 1993). The rules for ANFIS structure are described in details in (Mat Darus, 2004; Ismail et al., 2006b; Mat Darus and Tokhi, 2006).

Neural-adaptive learning methods present a technique for the fuzzy modelling process to learn information about a data set, so as to compute the membership function parameters that best allow the associated fuzzy inference system to follow the given input/output data. This learning method works likewise to that of neural networks. To use ANFIS for identification problem, the following steps are needed:

1. A Sugeno FIS appropriate for identification problem is design.
2. The FIS, with given actual input identification data is optimised.
3. Training and testing matrices composed of inputs and the desired identification corresponding to those inputs are set up.
4. The ANFIS algorithm is run on the training data.
5. The results are tested using the testing data.

During the identification using ANFIS structure, input/output data set are used to constructs a Fuzzy Inference System (FIS). The membership function parameters in FIS are adjusted using either a combination of a backpropagation algorithm and a Least Squares type of method, or backpropagation algorithm alone. This allows the fuzzy systems to learn from the data given.

In ANFIS, backpropagation learning is used to learn the parameters related to membership functions and least mean square estimation to determine the consequent parameters. In the learning procedure, every step includes two parts. The input sets are propagated, and the optimal consequent parameters are estimated by an iterative least mean square procedure. Through

![Fig. 11. Type-3 fuzzy reasoning (Ismail et al., 2006b).](image)

![Fig. 12. Structure of type-3 ANFIS (Jang, 1993).](image)
the training set, the premise parameters are assumed fixed for the current cycle. The set is propagated again, and in this epoch, backpropagation is used to adjust the premise parameters while the consequent parameters remain fixed (Mandic and Chambers, 2001).

Through the learning process, the parameters associated with the membership functions will be changed. The parameters’ adjustment is facilitated by a gradient vector, which provides a measure of how well the fuzzy inference system is modelling the input/output data for a given set of parameters. After obtaining the gradient vector, any of several optimisation routines could be applied to adjust parameters that will reduce some error measure (usually defined by the sum of the squared differences between actual and desired response; Sutton and Craven, 1998; Minghui et al., 2009).

In this research the utilisation of Neuro-fuzzy model is investigated using ANFIS structure to model the flexible, flat, plate system. The diagrammatic representation of an ANFIS network for system identification is similar to MLPNN algorithms as shown in Fig. 9.

5. Model validation

After obtaining the model of the system, it is necessary to validate whether the model is sufficient to represent the system or not. The procedures that considered for sensing the sufficiency of a fitted model are called Model validity tests. The principle of model validation is:

- Compare model simulation or prediction with real data in time domain.
- Compare estimated model’s frequency response and spectral analysis result in frequency domain.
- Perform statistical test on prediction errors.

Numerous validation tests are existing in the literature, some of which are mean squared error, correlation error, model predicted output and one step-ahead prediction (Mat Darus, 2004). In this research one step-ahead prediction, mean squared error and correlation test are used to validate the model.

5.1. One step-ahead prediction

The One-Step-Ahead (OSA) prediction of the system output is a familiar measure of predictive accuracy used in system identification and control. This is expressed as

\[ \hat{y} = f(u(t), u(t-1), \ldots, u(t-n_u), y(t-1), \ldots, y(t-n_y)) \]  

(3)

where \( u \) and \( y \) are the inputs and outputs respectively. \( f(\cdot) \) is a non-linear function. The error or prediction is given as shown in Eq. (4).

\[ e(t) = y(t) - \hat{y}(t) \]  

(4)

If the model is biased, \( \hat{y} \) will be a quite good prediction of \( y(t) \) over the estimation set because the model was estimated by minimising the prediction errors (Tokhi and Veres, 2002).

5.2. Mean Squared Error (MSE)

MSE is one of the most common methods of validations. The MSE is different between the real output \( y(n) \) of the system and the predicted output \( \hat{y}(n) \) produced from the input to the system and the optimised parameters as shown in Eq. (5).

\[ mse = \frac{1}{N} \sum_{t}^{N} (y(t) - \hat{y}(t))^2 \]  

(5)

5.3. Correlation test

Correlation test is a statistical test that shows the degrees of the relationship between two variables. There are two types of correlation test:

1. Autocorrelation test is representing as a vector.
2. Cross correlation test is representing as matrix.

Correlation test is a more convincing method of model validation. It is the usual statistical method to validating identified non-linear models. It has been shown that a suitable prediction through different data sets is produced only if the model is unbiased. The prediction error sequence \( e(t) \) should be uncorrelated with all linear and non-linear combinations of past inputs and outputs (unbiased) when the model structure and the estimated parameters are correct. This will hold if and only if the following conditions are satisfied (Billings and Voon, 1986):

\[ \phi_{uu}(\tau) = E[e(t-\tau)e(t)] = \delta(t) \]  

\[ \phi_{ue}(\tau) = E[u(t-\tau)e(t)] = 0, \quad \tau \neq 0 \]  

\[ \phi_{ee}(\tau) = E[e(t-\tau)e(t)] = 0, \quad \tau \geq 0 \]  

\[ \phi_{ee}(\tau) = E[e(t)e(t-1-\tau)u(t-1-\tau)] = 0 \]  

(6)

where \( \phi_{uu}(\tau) \) indicates the cross-correlation function between \( u(t) \) and \( e(t) \), \( \phi_{ee}(\tau) = e(t+1)u(t+1), \delta(t) \) is an impulse function.

Actually, the correlation will never be precisely zero for all lags but the model is considered as satisfactory if the correlation tests lay within 95% confidence limits, defined as \( 1.96/\sqrt{N} \), where \( N \) is the data length. Autocorrelation of the error also will never be an ideal delta function but will be considered as sufficient if the autocorrelation plot enters the 95% confidence limits before lag one.

6. Implementation and results

For experimental purposes, a rectangular, thin, flat aluminium plate with all edges clamped (C–C–C–C) boundary condition is considered. Table 1 shows the properties of the plate.

For the purpose of the development of active vibration control (AVC), which will be conducted for the future work, a new arrangement of the experimental rig is shown in Fig. 13. A setup strategic location for shaker (for input signal) at point (X) is identified appropriately on the plate. The location of sensor (for detection signal) at point (Y) and the location of sensor (for observation signal) at point (Z) are chosen so that it is far enough from the nodal lines defined by the first five natural frequencies of the plate.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length (a) (m)</td>
<td>1.58</td>
</tr>
<tr>
<td>Width (b) (m)</td>
<td>1.08</td>
</tr>
<tr>
<td>Thickness (h) (m)</td>
<td>0.003</td>
</tr>
<tr>
<td>Density (p) (Kg m(^{-3}))</td>
<td>2690</td>
</tr>
<tr>
<td>Modulus of elasticity (E) (N m(^{-2}))</td>
<td>6.831010</td>
</tr>
<tr>
<td>Poisson ratio (ν)</td>
<td>0.34</td>
</tr>
</tbody>
</table>
These arrangements are as such as to ensure the best performance of AVC for vibration reduction at the observation point (Z), will be achieved. Nevertheless, the best AVC can only be achieved

Fig. 13. Schematic description of the plate with all edged clamped boundary conditions for development of AVC (Tavakolpour et al., 2010).

Fig. 14. Lateral deflection detected at $x=0.75a$, and $y=0.75b$ point $H$ (detection point).

Fig. 15. Lateral deflection detected at $x=0.75a$, and $y=0.25b$ point $H$ (observation point).

Fig. 16. The actual and MLPNN predicted output.

Fig. 17. Error between actual and MLPNN predicted output.

Fig. 18. Mean-squared error vs. number of training passes.
if the model of the system is described accurately. Therefore, an experimental studies were carried out to obtained the input/output data (between detection and observation points) and these data will later used for system identification to obtain the best model describing the model of the plate. A sinusoidal input force \((F)\) with an amplitude of \([19 \text{ Hz}, 5 \text{ V}]\) at time instance of 4 s was applied to the excitation point \((X)\) located at \(x=0.25a, \ y=0.25b\). The experimental lateral deflection of the plate at detection point \((Y)\) located at \(x=0.75a, \ y=0.75b\), in time domain response is plotted in Fig. 14. The experimental lateral deflection of the plate at observation point \((Z)\) located at \(x=0.75a, \ y=0.25b\), in time domain response is plotted in Fig. 15.

6.1. Multi-layer Perceptron Neural Network (MLPNN) modelling

A coding has been produce base on the Multi-layer Perceptron using MATLAB environment. An MLPNN with 2 hidden layers, with 6 tansigmoid neurons in first hidden layer, 6 tansigmoid neurons in second hidden layer, and one output layer with linear neuron. Since there was not a priori knowledge about a suitable order of the model for the flexible plate system, the structure realisation was performed by a trial-and-error method. So, the deflection model was experimented with different orders. The data set, comprising 4000 data points, was separated into two sets of 3000 and 1000 data points. The model was trained using the first set and the model was

Fig. 19. Correlation tests of MLPNN. (a) Auto–correlation of error, (b) correlation of the input and the error, (c) correlation of the square of input and the error, (d) correlation of square of the input square of the error and (e) correlation of multiplication of the input by the error and the error.
validated with the whole 4000 points including the 1000 points that had not been used in the training process. Both output and estimated outputs are plotted in the Fig. 16. The error between actual and predicted MLPNN output are plotted in the Fig. 17 and the mean-squared error vs. number of training passes in Fig. 18. The best result was achieved with an order 20, which means, $n_u = n_y = 10$ for 4000 data length was trained to characterize the plate. The models reached a sum-squared error level of 0.00017103 with 150 training passes for MLPNN modelling.

The correlation tests were carried out to determine the effectiveness of the (MLP) BP-based model. Fig. 19 shows the results of the correlation tests. The results were also found to be within 95% confidence level thus confirmed the accuracy of the results.

### 6.2. Elman Neural Network (ENN) modelling

A coding has been produce based on the Elman Neural Networks within MATLAB environment. An Elman Neural Network with 2 hidden layers, with 6 tansigmoid neurons in first hidden layer, 6 tansigmoid neurons in second hidden layer, and one output layer with linear neuron. Since there was no prior knowledge about a suitable order of the model for the flexible plate system, the structure realisation was performed by a heuristic method. The deflection model was tested with different orders. The data set, comprising 4000 data points, was divided into two sets of 3000 and 1000 data points. The model was trained using the first set and the model was validated with the whole 4000 data, including the first 1000 data, which had not been used in the training process. Fig. 20 shows the result of the actual and Elman predicted output. Fig. 21 shows the error between actual and predicted Elman output and the mean-squared error vs. number of training passes in Fig. 22. The best result was achieved with an order 20, which means, $n_u = n_y = 10$ for 4000 data length was trained to characterize the plate. The models reached a sum-squared error level of 0.002 with 150 training passes for Elman Neural Network modelling.

The correlation tests were carried out to determine the effectiveness of the ENN-based model. Fig. 23 shows the results of the correlation tests. The results were also found to be within 95% confidence level thus confirmed the accuracy of the results.

### 6.3. Adaptive Neuro-Fuzzy Inference System (ANFIS) modelling

A coding has been produce based on the ANFIS Networks using MATLAB environment. Since there was not a priori knowledge about a suitable order of the model for the flexible plate system, the structure realisation was performed by a trial-and-error method. So, the deflection model was experimented with different orders. The data set, comprising 4000 data points, was divided into two sets of 3000 and 1000 data points. The model was trained with The first set and the model was validated with the whole 4000 points, including the 1000 points that had not been used in the training process. The best result was achieved with an order 4, which means, $n_u = n_y = 2$ for 4000 data length was trained to characterize the plate. The models reached a sum-squared error level of 0.00039781 with 150 training passes for ANFIS network modelling. The model has 16 rules for model order $n = 4$ and two Gaussian membership functions. Gaussian membership functions with product inference rule are applied at the fuzzification level. The fuzzifier outputs the firing strengths for each rule. At the defuzzification level, the first order sugeno model is utilised. Fig. 24 shows the result of the actual and ANFIS predicted output, Fig. 25 shows error between actual and predicted ANFIS output.

The correlation tests were carried out to determine the effectiveness of the ANFIS-based model. Fig. 26 shows the results of the correlation tests. The results were also found to be within 95% confidence level thus confirmed the accuracy of the results.
Fig. 23. Correlation tests of ENN. (a) Auto-correlation of error, (b) correlation of the input and the error, (c) correlation of the square of input and the error, (d) correlation of square of the input square of the error and (e) correlation of multiplication of the input by the error and the error.

Fig. 24. Actual and ANFIS predicted output.

Fig. 25. Error between actual and predicted output.
7. Comparative assessment

Comparative performance of non-parametric modelling methods in terms of the mean-squared of error is summarised in Table 2. It follows from detection and observation mapping as presented earlier in this study, non-parametric models have performed very well. Validations through test procedures and correlation tests have also been performed with the MLPNN, ENN and ANFIS based models. It is observed from the validation tests that different modelling methods considered in this study have performed sufficiently well. Comparing the mean-squared errors in Table 2, it is noted that for non-parametric identification technique, MLPNN have performed better than ENN and ANFIS in characterising the flexible plate structure.

<table>
<thead>
<tr>
<th>Method</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLPNN</td>
<td>0.00017103</td>
</tr>
<tr>
<td>ENN</td>
<td>0.002</td>
</tr>
<tr>
<td>ANFIS</td>
<td>0.00039781</td>
</tr>
</tbody>
</table>

8. Discussion

Results of various modelling methods have been validated with a range of tests including input/output mapping, mean-squared
error and correlation tests. It is observed that all the modelling methods have performed very well in approximating the system response.

A comparative assessment of the performance of non-parametric approaches in modelling a flexible plate structure has been carried out. It is demonstrated that MLPNNs and ANFIS perform better than ENN in modelling and identification of a flexible plate structure. Thus, the system data can closely be predicted with a very small prediction error with suitable choice of the input data structure.

The non-parametric models of the flexible plate structure thus developed and validated will be used as the transfer function of the system in subsequent investigations for the development of active vibration control strategies for vibration suppression in flexible structures.

Acknowledgement

The authors would like to express their gratitude to the Universiti Teknologi Malaysia (UTM) for their continuous support in the research work. This research was fully supported by the UTM Research University Grants (GUP) using vote no. 00H11.

References


