Artificial Intelligence Review (2020) 53:3177–3199
https://doi.org/10.1007/s10462-019-09759-8

An integrating genetic algorithm and modified Newton method for tracking control and vibration suppression

Chuen-Jyh Chen[1]

Published online: 5 September 2019

Abstract
With the trend toward taller and more flexible building structures, a mass-damper shaking table system has been considered as means for vibration suppression to external excitation and disturbances in recent years. However, there are few researches on the control of nonlinear structure using active mass damper under earthquake excitation, especially for high-rise building. This study presents a model combining the advantages of adaptive genetic algorithm and modified Newton method is developed for system identification and vibration suppression of a building structure with an active mass damper. The genetic algorithm with adaptive reproduction, crossover, and mutation operators is to search for initial weight and bias of the neural network, while the modified Newton method, similar to BFGS algorithm, is to increase network training performance. Experimental results show that the controller performance is strongly influenced by the accuracy of system identification. The controller is also shown to be robust to variations in system parameters.

Keywords Genetic algorithm · Modified Newton method · Tracking control · Vibration control

1 Introduction

With the trend toward taller building structures, undesirable vibration levels can arise from large environmental loads such as strong wind and earthquakes. State-of-the-art active control techniques for building structures were reported by Soong (1988) and Yoshida (1994), in which some active control devices have been implemented. The tuned mass damper devices were seen in stacks, observatories, bridges, communication towers, and machinery foundations (Catalogue of Mitsubishi Heavy Industries, LTD. 2019, Tokyo, Japan). Extraordinary loading episodes can occur and cause casualties and structural damage. Development of mass-damper systems, such as tuned mass-dampers and active mass-dampers (AMD), has been introduced into high-rise buildings to reduce vibration. Vibration suppression of struc-

✉ Chuen-Jyh Chen
chuenjyh@mail.cjcu.edu.tw

[1] Department of Aviation and Maritime Transportation Management, Chang Jung Christian University, Tainan, Taiwan, ROC

ture systems subject to external excitation and disturbances must be intensively studied (Li et al. 2010; Liu et al. 2011; Lazar et al. 2013; Wang et al. 2018). Recent research has been focusing on the applications of active mass damper (AMD) -an auxiliary mass with sensor, actuator and controller- to vibration suppression of a building structure (Chen 2012; Song et al. 2011). The progress in peripheral technologies, such as computation speed, low friction of linear bearing, and high horsepower servo motor have made active control feasible in structure systems (Korkmaz 2011).

Conventionally, the proportional–integral–derivative (PID) controller dominates servo system applications; however, it is not sufficiently robust to accommodate external disturbances, parameter variations, and structural perturbations. Many alternate controllers have been developed to replace the PID controller, including model reference adaptive control (Ohnishi et al. 1986), the field acceleration method (Zhong et al. 1991), fuzzy logic (Tseng and Hwang 1993), sliding mode control (Utkin 1993), neural/fuzzy control (Goode and Chow 1995a, b), input–output linearization (Marino et al. 1993; Grcar et al. 1996), and internal model control (Tzou 1996). Nevertheless, the PID controllers remain the mainstream controller for AC motors because of their relatively simple implementation. Within AC motors, the self-turning controller, in particular, may be redesigned to track differing reference inputs and structural vibration suppression.

Unmodeled nonlinear friction has been known to have an adverse influence on controller performance; therefore, a model-based approach is often employed to reduce friction effects. The critical issue in use of the model-based approach is the development of a precise friction model. Johnson and Lorenz (1992) experimentally estimated the parameters within their friction model, while Yang (1993) suggested proportional–integral–derivative (PID) control to eliminate friction effects through online model parameter estimation. Haessing and Friendland (1991) and Wit et al. (1995) have also proposed alternative friction modeling approaches. The effectiveness of these model-based studies has been found to heavily rely upon the model accuracy. Because of the inherent nonlinearity of friction, an accurate model is difficult, if not impossible, to construct.

The artificial neural network technique, with its high learning and nonlinear mapping abilities, has been successfully applied to many system identification and control problems (Sastry et al. 1994; Levin and Narendra 1996; Omatu et al. 1997). Amini et al. (1997) indicated that the structural dynamics of an object subjected to earthquake loading can be modeled by neural network. Han et al. (2001) introduced a neural network model, with recurrent architecture, for earthquake response. Yu and Li (2001) applied the neural networks to identify a nonlinear system, while Yang et al. (2006) and Chen and Yang (2014) applied the neural network models for the experimental identification of system dynamics. Yang et al. (2007) have also developed a self-organized neuro-fuzzy design methodology for modeling. Chen et al. (2009a, b, 2013, 2014) proposed a self-organized neuro-fuzzy model for system identification, in which their five-layer network adaptively adjusts the membership functions and dynamically optimizes the fuzzy rules. Learning-based control methodology via the neural networks has recently emerged as an alternative to adaptive control (Jagannathan 2001). However, there are few researches on the control of nonlinear structure using neural network for tracking control and vibration suppression under shaking table conditions, especially for high-rise building. In this work, a model combining the advantages of adaptive genetic algorithm and modified Newton method, similar to BFGS algorithm, is developed for effective training in feedforward neural networks. The benchmark tests show that the model is superior to many conventional ones: steepest descent, steepest descent with adaptive learning rate, conjugate gradient, and Newton-based methods and is suitable to small network in engineering applications. In the experimental verification, a modified Newton method with genetic

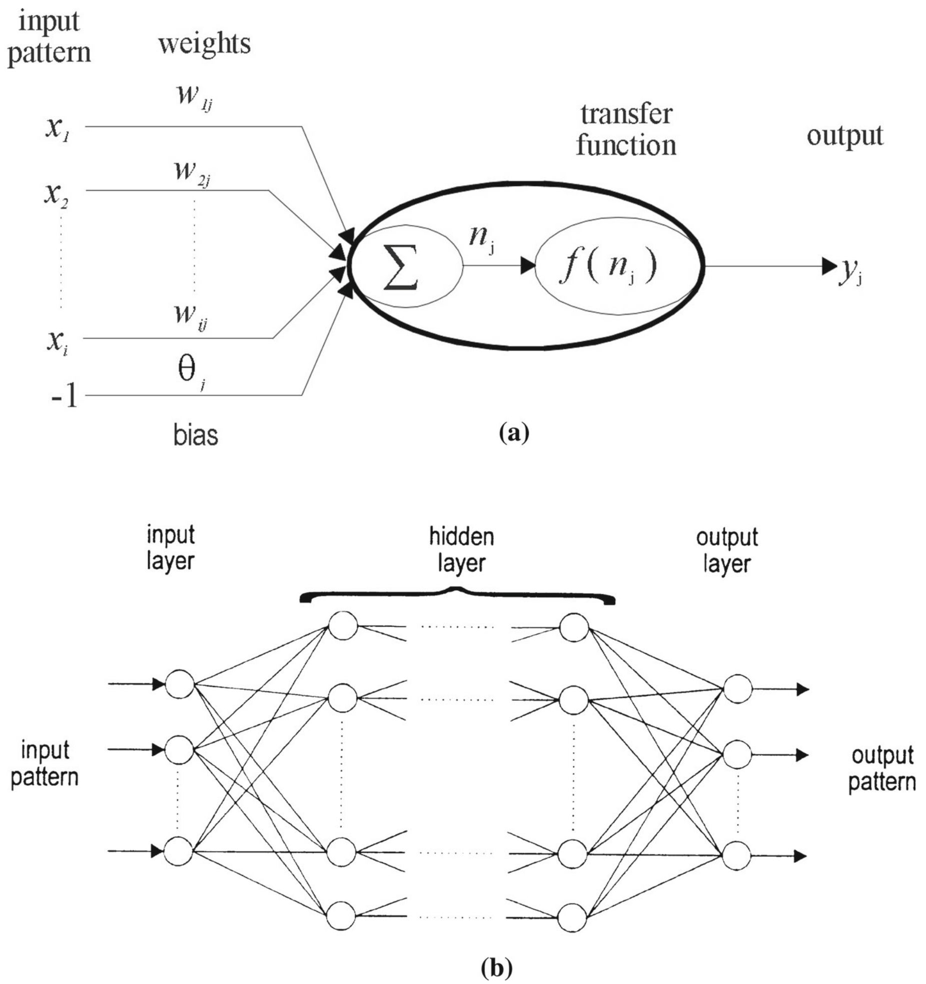


Fig. 1 **a** The structure of an artificial neuron. **b** The schematic diagram of a feedforward neural network

algorithm for neural network training is applied to system identification, tracking control, and vibration suppression of a shaking table system.

2 Neural network and genetic algorithm

The artificial neural network is a data processing system composed of numerous interconnected artificial neurons to mimic the biological neural network. An artificial neuron (processing element) is shown in Fig. 1a. The relationship between the input and output can be represented by $n_j \equiv \sum_i w_{ij}x_i - \theta_j$ and $y_j = f(n_j)$ where n_j is the linear combination output; w_{ij} is the synaptic weight (or simply weight) imitating the biological synapse strength between the jth neuron and the ith input; x_i is the ith input of the processing element (neuron); θ_j is the bias representing the threshold of the transfer function; y_j is the output of the jth neuron and $f(\circ)$ is the transfer function (also called the activation function) for converting the weighted summation input n_j to the output y_j. The connective weight of each connection represents the relative strength between two artificial neurons. A feedforward network may have several hidden layers as shown in Fig. 1b. The hidden layers provide the interaction of the neurons that represent the system's internal structure. The goal of learning (training) processes is to adjust the connection weights in the layers for minimizing the error function by repeatedly training. The error function E_p is defined as

$$E_p = \frac{1}{2}\sum_{i=1}^{n_o}(t_i^p - y_i^p)^2,\ \ p = 1,\ 2,\ \ldots,\ n_p \tag{1}$$

where n_o is the number of neurons in the output layer; t_i^p and y_i^p are the desired (target) and actual output of the ith output neuron for the pth output sample, respectively; and n_p is the number of training samples. A neural network contains many processing neurons such that a higher degree of robustness and ault tolerance is possible.

Genetic algorithm is an optimization algorithm based on the mechanism of natural selection and genetics. It has been demonstrated to be an effective global optimization tool. Genetic algorithm evolves a population of individuals, also called strings or chromosomes. Each individual $v_i (i = 1, \ldots, s)$ of the population V represents a trial solution of the problem. Chromosomes are usually represented by string of variables, and each element of which is called a gene. Every gene controls the inheritance of one or several characters. The value of a gene is called allelic value in [0, 1]. The algorithm can be described by reproduction, crossover, and mutation.

Reproduction is a process by which the most highly rated chromosomes in the current generation are reproduced in the new generation. Selection is often in the roulette wheel process to calculate the fitness value for each chromosome $q_i = \sum_{j=1}^{i} p_j$, $i = 1, \ldots, s$, where s is population size, and to find the total fitness of the population

$$f = \sum_{i=1}^{s} e(v_i), \tag{2}$$

where $e(v_i)$ is fitness function. The probability of a selection for each chromosome is

$$p_i = \frac{1}{f} e(v_i). \tag{3}$$

and the cumulative probability is

$$q_i = \sum_{j=1}^{i} p_j. \tag{4}$$

For a random (floating) number a within (0, 1), if $a < q_1$, then select the first chromosome v_1, otherwise select the ith chromosome v_i $(2 \leqq i \leqq s)$ such that $q_{i-1} < a \leqq q_i$.

Crossover operator provides a mechanism for chromosomes to mix and match. The operators are selected the arithmetical crossover method

$$\begin{cases} \boldsymbol{x}'_1 = a\boldsymbol{x}_1 + (1-a)\boldsymbol{x}_2 \\ \boldsymbol{x}'_2 = a\boldsymbol{x}_2 + (1-a)\boldsymbol{x}_1 \end{cases} \tag{5}$$

where $\boldsymbol{x_1}, \boldsymbol{x_2}$ are parents, $\boldsymbol{x}'_1, \boldsymbol{x}'_2$ are offsprings, and a is a random value in $[0, \ldots, 1]$.

Mutation operator is a random alteration of some gene value in a chromosome. The operators are selected the non-uniform mutation method. For a parent $\boldsymbol{x}$, if the element x_k is selected for this mutation, the result is

$$\boldsymbol{x}' = [x_1, \ldots, x_k', \ldots, x_q],$$

where

$$x_k' = \begin{cases} x_k + \Delta(t, u_k - x_k), & \text{if a random digit is 0} \\ x_k - \Delta(t, x_k - l_k), & \text{if a random digit is 1}, \ x_k \in [l_k, u_k] \end{cases} \tag{6}$$

The function $\Delta(t, y)$ returns a value in the range $[0, y]$ such that the probability of $\Delta(t, y)$ is close to zero as the generation number t increases. The function $\Delta(t, y)$ can be expressed as

$$\Delta(t, y) = ya\left(1 - \frac{t}{T}\right)^b, \tag{7}$$

where a is a random number within [0, 1], T is the maximal generation number, and b is a system parameter of the degree of non-uniformity.

There remain few guidelines in providing their initial values, though they are known critical to the training effectiveness of a neural network. The initial weight and bias are usually generated randomly. Genetic algorithm performs a multi-directional search by maintaining a population of potential solutions and it can find a near optimal solution. Although the algorithm usually requires a long training time, it can be used in the early stages of training to search for good starting point and detect the region where the global minimum is likely to be found. A two-stage model for training a feedforward neural network is proposed. The first stage is to find better initial weight and bias by using floating genetic algorithm. The second stage is to train the feedforward neural network by using modified Newton method with the specific initial weight and bias. This algorithm includes the advantages of genetic algorithm in finding better initial weight and bias and modified Newton method in accelerating the convergence of the network training.

3 Accelerated convergence by modified Newton algorithm

BPN is to utilize the gradient steepest descent algorithm to minimize the error between the desired and actual outputs in an iterative manner. The back-propagation algorithm is a first-order approximation of the steepest descent technique in the sense that it depends on the gradient of the instantaneous error surface in the weight space. The update rule is thus plagued by the drawbacks of converging slowly and being stuck easily in local minimum of the error function. To accelerate the convergence of the back-propagation algorithm, many studies have presented many different training algorithms. The modified Newton method similar to BFGS algorithm includes the advantages of the steepest descent algorithm and Newton's algorithm, and they require less memory storage and computational load than the Newton's algorithm and the Levenberg–Marquardt algorithm in numerical analysis. Start with the initial weights, biases and an initial estimate of the inverse of the Hessian matrix, denoted as $\boldsymbol{B}$, of the error function as in Eq. (1). The update rule can be written by

$$\Delta w_{ij}(k+1) = -\eta \boldsymbol{B}(k) \nabla E(w_{ij}(k)) \tag{8}$$

where E is the error function, η is network learning rate, w_{ij} is the connection weight, ∇E is the gradient of E, and B is an approximate inverse of the Hessian matrix of E. The inverse of the Hessian matrix is update by

$$B(k+1) = B(k) + \left(1 + \frac{q^T(k)B(k)q(k)}{p^T(k)q(k)}\right)\frac{p(k)p^T(k)}{p^T(k)q(k)} - \frac{p(k)q^T(k)B(k) + B(k)q(k)p^T(k)}{p^T(k)q(k)}$$

where

$$p(k) = w_{ij}(k+1) - w_{ij}(k), \tag{9}$$

$$q(k) = \nabla E(w_{ij}(k+1)) - \nabla E(w_{ij}(k)). \tag{10}$$

or if $\left\| p^T(k)q(k) \right\| \leq \varepsilon$, where ε is a positive constant, then $\boldsymbol{B}(k+1)$ is the identity matrix. The modified Newton method is essentially by substituting the inverse Hessian matrix with an approximation from weight and gradient changes in iteration process. It combines the stability of gradient descent with the speed of Newton-based method.

4 System identification using modified Newton method with genetic algorithm

Two examples are provided to illustrate the performance of the modified Newton method with genetic algorithm. The first example is the application of the modified Newton method with genetic algorithm in the identification of a differential equation system. Consider the plant is governed by the difference equation (Narendra and Parthasarathy 1990)

$$y(k+1) = F[\,y(k), y(k-1), y(k-2), u(k), u(k-1)] \tag{11}$$

where the function F has the form

$$F[x_1, x_2, x_3, x_4, x_5] = \frac{x_1 x_2 x_3 x_5 (x_3 - 1) + x_4}{1 + x_3^2 + x_2^2}. \tag{12}$$

A $[5 \times 5 \times 1]$ feedforward neural network model, representing 5 neurons in the input layer, 5 in the hidden layer, and 1 in the output layer, is constructed to simulate the plant represented by the differential equation. The input to the plant and network model is assumed to be an independent and identically distributed random signal that is uniformly ranged in the interval $[-1, 1]$. The identification procedure consists of two phases: the network training phase and the validation phase. The first is to establish the appropriate connection weights of the network by a form of supervised learning from a set of network input and desired network output. At each training step, all input are passed forward through the network to yield the network output which are then compared with the desired output. If the error is small enough, then the connection weights are fixed. Conversely, the error is passed backwards through the network, and the back-propagation algorithm with the adaptive learning rate and momentum techniques by batch learning is applied to adjust the connection weights. Note that batch learning rather than recursive learning is adopted here due to the fact that the latter tends to terminate at local minima of the parameters, hence leading to a higher total error than the former. Although recursive learning is adaptive, however, batch learning method is relatively more efficient and stable. During the validation phase, the network is given many other sets of input in addition to those used for training. The network will produce the output $\hat{y}(k)$ very close to the actual system output $y(k)$ if the training is successful. From this point of view, the

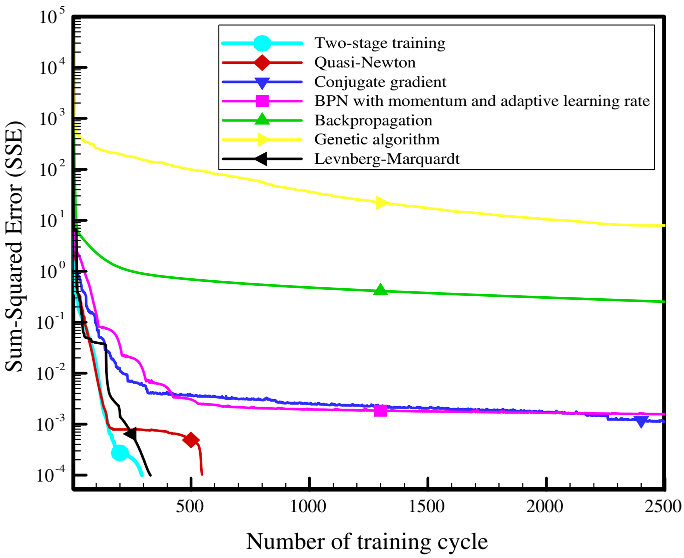


Fig. 2 The variations of sum-squared error of the differential Eq. (12) by using different algorithms

neural network can therefore be regarded as just another form of functional representation. For a single output system, define the sum-squared error function as the performance index

$$E = \sum_r (\hat{y}_r(k) - y_r(k))^2 \tag{13}$$

where the index r ranges over all training samples.

Figure 2 shows the results for sum-squared error during training obtained by using the quasi-Newton algorithm, back-propagation algorithm with momentum term and adaptive learning rate (Yang and Lee 1998), conjugation gradient algorithm (Jeng et al. 1997), back-propagation algorithm (Yang and Lee 1997), and two-stage training algorithm. The two-stage training algorithm is applied as follows: First, the floating genetic algorithm is applied with the population size $s = 500$ and 100 training cycles to obtain the initial weight and bias. Then, the modified Newton method is applied to train the feedforward neural network. The training pattern consists of 500 input/output pairs, and the training continues until the sum-squared error is less than 0.0001. Figure 2 shows that the result of the two-stage training algorithm is better than that of other algorithms. Compared with other algorithms, the two-stage training algorithm provides better initial weights and bias. After the training process, the validation phase is carried out by using the following input,

$$\begin{aligned} u(k) &= 0.6\sin(\pi\, k/20), && \mathrm{k} < 100 \\ &= 0.6, && 100 \le \mathrm{k} < 150 \\ &= -0.6, && 150 \le \mathrm{k} < 200 \\ &= 0.6\sin(\pi\, k/15), && 200 \le \mathrm{k} < 300. \end{aligned} \tag{14}$$

The outputs of the plant and the network model are shown in Fig. 3, which also shows that neural network identification is highly accurate.

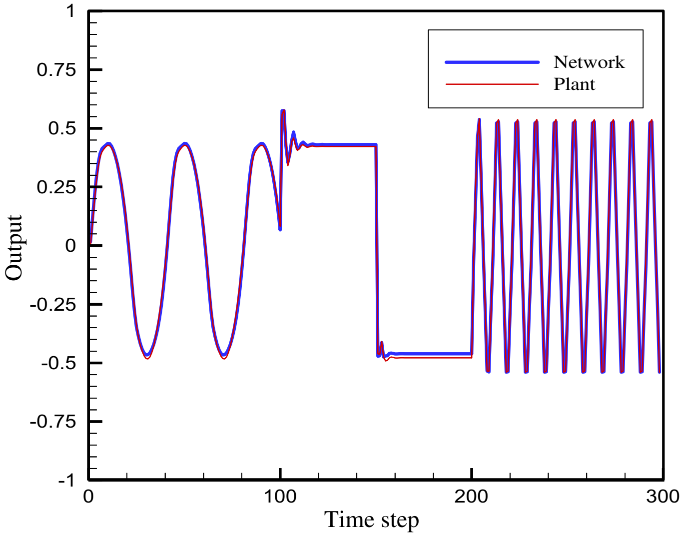


Fig. 3 Plant output and the network model output of the differential equation

The other benchmark test involves a nonlinear, damped Duffing oscillator, and the equation of motion is

$$\ddot{y} + 2\dot{y} + 400y + 200y^3 = 500u(t) \tag{15}$$

where y is the displacement of the hard spring and $u(t)$ is the external applied force. By transforming the continuous system into discrete-time form with the sampling interval Δt, the equation of motion can be written as

$$y(k+1) = F[y(k), \dot{y}(k), u(k), \Delta t] \tag{16}$$

A $[3 \times 10 \times 2]$ feedforward neural network representing 3 input nodes, 10 neurons in a hidden layer, and 2 output nodes are employed to identify the Duffing system. The input is a random signal in the range of [− 2, 2] with an effective sampling frequency of 100 Hz. The velocity is scaled by a factor of 30 to avoid neuron disparities. The training pattern consists of 700 input/output pairs, and the training proceeds until the sum-squared error is less than 0.01. Figure 4a shows the variation in the sum-squared error of the 8 algorithms during training under the same neural network parameters. The modified Newton method is more efficient than algorithms and requires 250 training iterations. By contrast, the quasi-Newton, Levenberg–Marquardt, and Gauss–Newton algorithms require 456, 478, and 319 training iterations, respectively. Moreover, the other training algorithms do not converge even after 500 iterations. The comparison of the time responses of the system output and neural network model during the validation phase is presented in Fig. 4b, which shows that neural network identification is highly accurate such that both lines coincide. The neural network model retains its excellent performance.

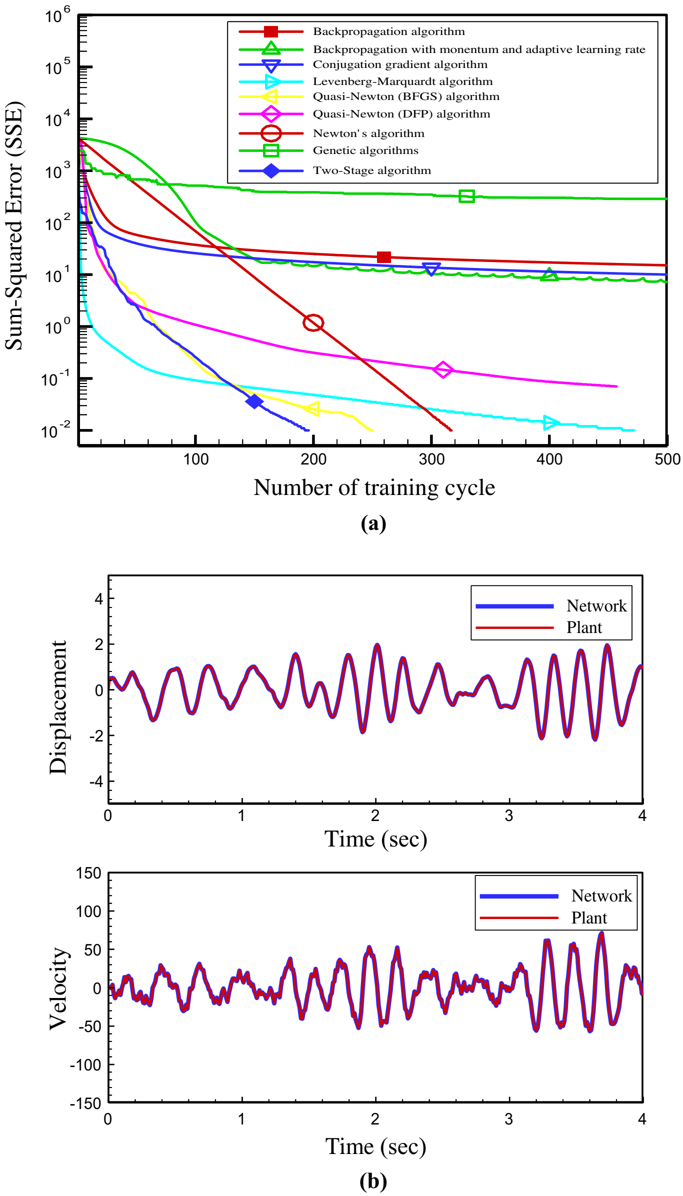


Fig. 4 **a** The variations of sum-squared error and **b** the plant output and the modified Newton model with genetic algorithm output in Duffing oscillator

5 Experimental verification for tracking control and vibration suppression in a shaking table system

In the experimental verification, the neural network with modified Newton method and genetic algorithm controller is applied in the positional tracking and vibration suppression of a

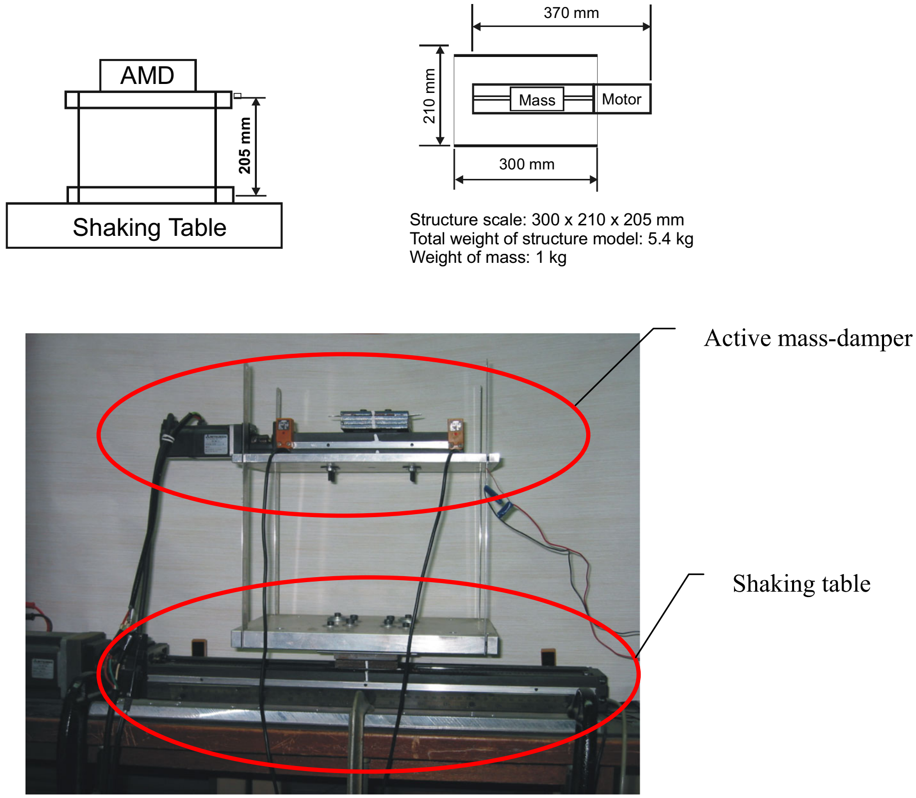


Fig. 5 Experimental model of the building structure combined with an active mass-damper system

Table 1 Specification of AC servo motor

Item	Servo motor	
	HC-MF13	HC-MF73
Rated output (kW)	0.10	0.75
Rated torque (N m)	0.32	2.40
Maximum torque (N m)	0.95	7.20
Rated speed (rpm)	3000	
Maximum speed (rpm)	4500	
Permissible instantaneous speed (rpm)	5175	
Inertia moment (kg cm^2)	0.03	0.60
Weight (kg)	0.53	3.00

shaking table system. An one-story shear-building structure model of $300 \times 210 \times 205$ mm, weights 5.4 kg combined with an active mass damper (AMD) mounted on the top floor is manufactured as shown in Fig. 5. A low-frequency accelerometer (KISTLER 8628B5) attached at the top floor acquires the acceleration. The measurement of accelerometer is then integrated to yield the velocity signal by the integrator (G-TECH 350133) which is then integrated and filtered to displacement signal. One linear wire potentiometer (HPS-

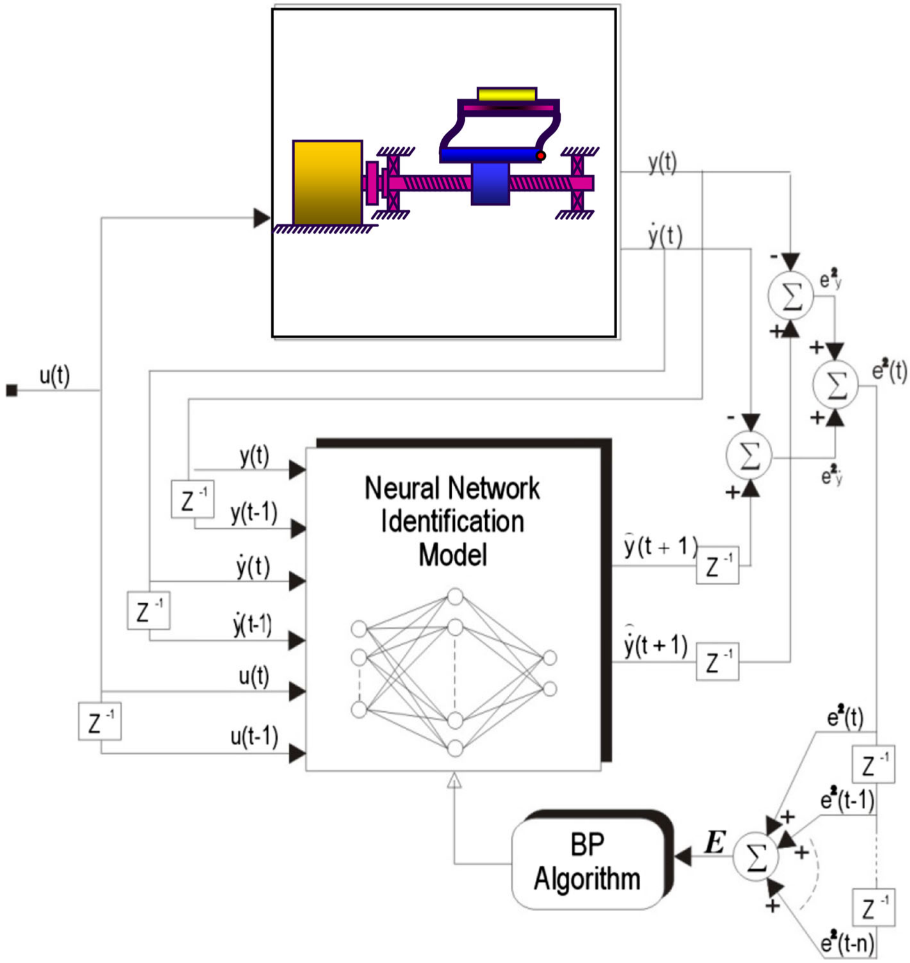


Fig. 6 The schematic of the shaking table identification (Narendra and Parthasarathy 1990) test

M-10-5K-F) fixed at the bottom floor through a analog I/O board (National Instruments PCI-6052E) is used to measure the displacement of shaking table. The analog I/O board (National Instruments PCI-6052E) is employed to generate the control signal to one of the AC servomotor (MITSUBISHI HC-MF73) for positional trackiing of the shaking table. The specification of the AC servomotor: the rated output is 0.75 kw, the rated torque is 2.4 N m, the maximum torque is 7.2 N m, the rated speed is 3000 rpm, the maximum speed is 4500 rpm, the permissible instantaneous speed is 5175 rpm, the inertia moment J is 0.6 kg cm^2 and the total weight is 3 kg. The data acquisition and control system consists of a multi-input/output board (National Instruments PCI-6052E) and the LAB-VIEW software. Table 1 lists the specification of the AC servo motors. For safety concerns, the following devices are used to ensure the reliability of the shaking table and avoid any dangerous malfunction on the building: a mechanical brake is combined with a limit switch to stop the auxiliary mass of the shaking table system when it overruns a certain position, and an automatic switching function which cuts off the motor current in case of abnormal situation or power failure.

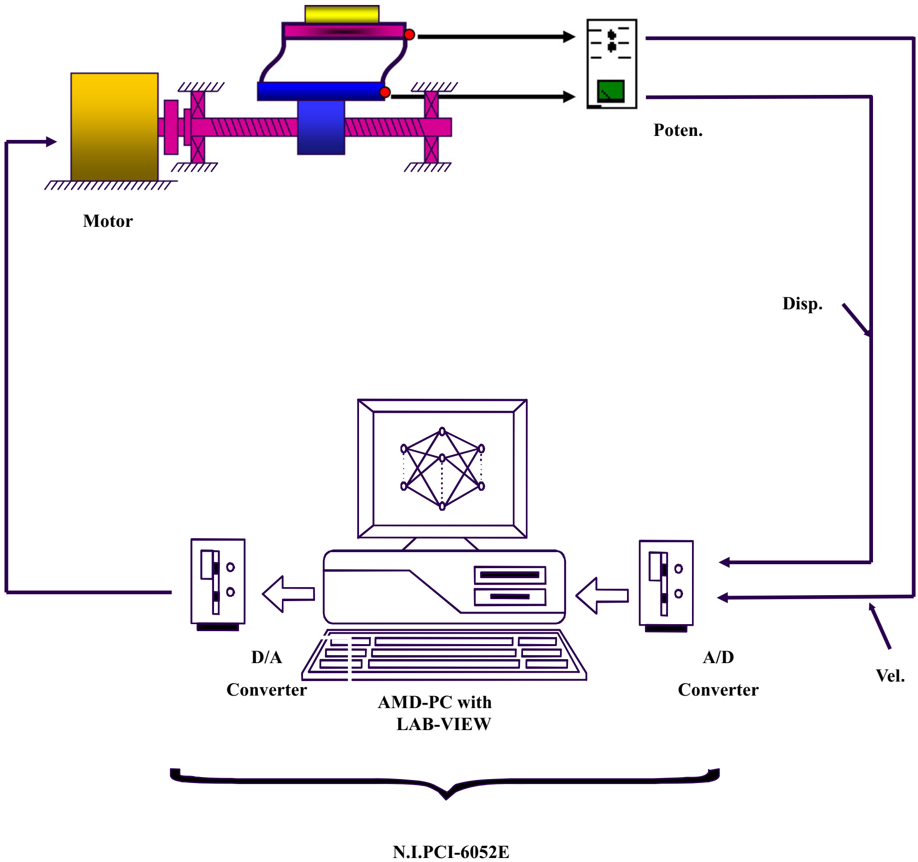


Fig. 7 The schematic diagram of the positional tracking control and vibration suppression experiment

The data acquisition is based on LAB-VIEW6.1 (National Instruments) and a PCI-6052E multifunction I/O board.

In the experimental verification, the neural network with modified Newton method and genetic algorithm controller is applied in the positional tracking and vibration suppression of a shaking table system (Fig. 5). The system state of displacement and velocity together with the actuator input voltage at the sampling rate of 10 Hz are used as the input of the network model. Given that a continuous system has an infinite degree-of-freedom, the number of steps of the preceding information fed as the network input depends mainly on the accuracy of the identification model and system dynamics. In this study, the previous step of system state and force, $[y(k-1), \dot{y}(k-1), u(k-1)]$, are used as network input to predict the present system state $[y(k), \dot{y}(k)]$. The schematic of the shaking table identification (Narendra and Parthasarathy 1990) test is shown in Fig. 6. Positional tracking control experiment with the sensor and the actuator is shown in Fig. 7. The signal passes through the sensors (HPS-M-10-5K-F & KISTLER 8628B5) and then estimates the displacement $y(k)$ and velocity $\dot{y}(k)$. Since the output voltage of the interface board is limited to ± 10 V, a voltage amplifier (Krohn–Hite 7500) is needed to drive the actuator in the range for effective control performance.

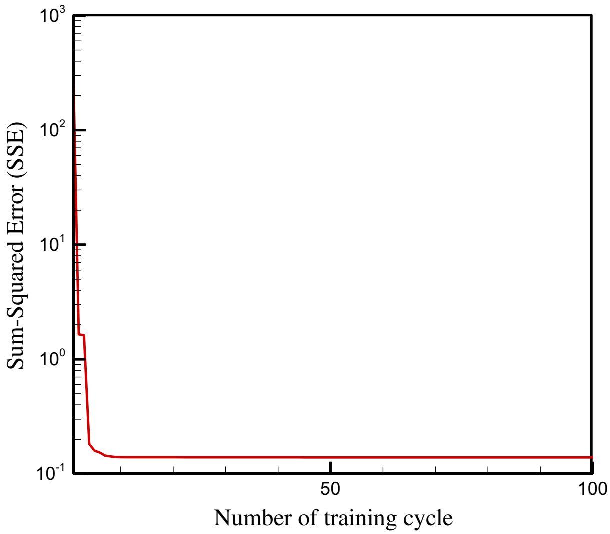


Fig. 8 The variations of sum-squared error during training phase in the [3–3–2] neural network with modified Newton method and genetic algorithm

The system displacement and velocity state variables together with the actuator input voltage at the sampling rate of 10 Hz are input into the network model, which is sufficient for the first natural frequency at approximately 1.9 Hz. The excitation force used to generate the training sets is an identically distributed (*iid*) random signal that is uniformly ranged in the [−1, 1] voltage range. The training pattern consists of 500 input/output pairs. A concise [3–3–2] three-layer network with the modified Newton algorithm is used to model the input/output transfer function. In this function, the number of hidden neurons is selected to be adequately small to restrict the error function within a small percentage. Increasing the numbers of hidden nodes does not necessarily provide good results because it reduces the generalizability of the trained network while increasing training time and complexity. The sum-squared error variation during training is shown in Fig. 8. The command responses of the system and identification model are compared in Figs. 9 and 10. Neural network identification is highly accurate such that both lines coincide. The modified Newton method effectively identifies the highly nonlinear system. This [3–3–2] BPN will be used as the identification model when training the neural network with modified Newton controller in the next verification test.

The basic structure of a neural network with the modified Newton method and genetic algorithm controller design is shown in Fig. 11. This neural network is similar to the indirect model-following control system. In this neural network, a feedforward neural network architecture is first constructed to represent system dynamics. Then, the neural network is established on the basis of the identification model. All that required for controller design is a neural network with the two-stage model of the system plant and a reference model for the desired system performance. The neural network with the two-stage model controller can be self-organized on the basis of output error between the plant and reference model. The learning algorithm then adjusts the weight and bias of the neural network with the modified Newton method and genetic algorithm controller to meet the design requirement.

The neural network with modified Newton method controller's training update rule is defined as

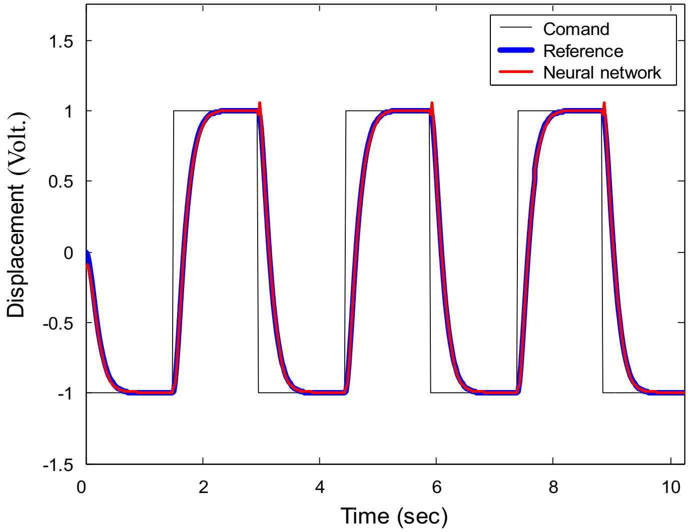


Fig. 9 The experimental results of system identification for the shaking table under square excitation

$$(\Delta w_{ij})_p = \eta \delta_i o_j. \tag{17}$$

where the delta δ_i of each layer is back propagated from the neural network model output layer to the previous layer(s) through the error function. During training process, the weight and bias of neural network identification model does not change, only by updating those in the neural network with modified Newton controller. The neural network model communicates delta δ_i subsequently to the output layer. δ_i is defined by

$$\begin{cases} \delta_{co} = \delta_{mi}[(n_{mi} - n_{co}) : n_{mi}] \text{ for the hidden layer} \\ \delta_{ch} = \sum_{k=1}^{n_{co}} (\delta_k w_{ki}) f^{'}(net_i) \text{ for the hidden layer} \end{cases} \tag{18}$$

where δ_{mi} and n_{mi} is from the input layer of neural network identification model, n_{co} is the output layer neurons of neural controller, and $f'(net_i)$ is the derivation of the transfer function. In this study, the neural network with modified Newton controller training is conducted off-line; i.e., it is trained after the successful neural network identification of the plant, for the following reasons: (1) hardware limitations of real-time computation, (2) the converging too slowly of neural network training process, and (3) the inherent short transient of the structural dynamics.

The governing equation of the structure is assumed to be unknown. However, the controller design of any unknown plant requires the mathematical model a priori. System identification is therefore an important step in the design of neural network controllers. The desired performance of the closed loop system with respect to a command input is specified in the reference model in the form of a linear differential equation.

The neural network with the modified Newton method and genetic algorithm is also applied in the tracking control experiment of a shaking table and the vibration suppression of a building structure. Because a continuous structure system has infinite degree-of-freedom, the number of steps of preceding information being fed as network input depends mainly on the accuracy of the identification model and system dynamics. In this study, the previous two steps of system state and force, $[y(k-1), \dot{y}(k-1), u(k-1)]$, are used as network input

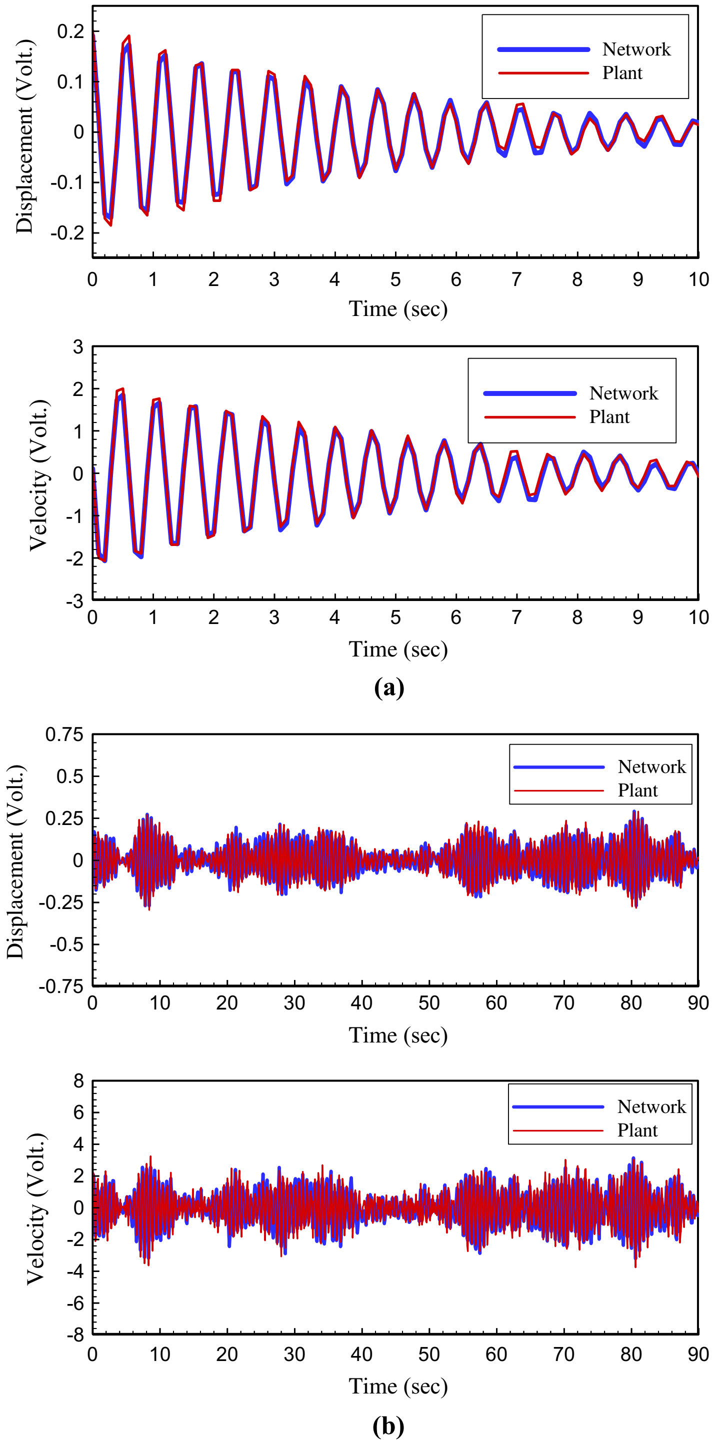


Fig. 10 Model validation of **a** the impulsive displacement **b** the sweep-sine excitation by [3–3–2] BPN

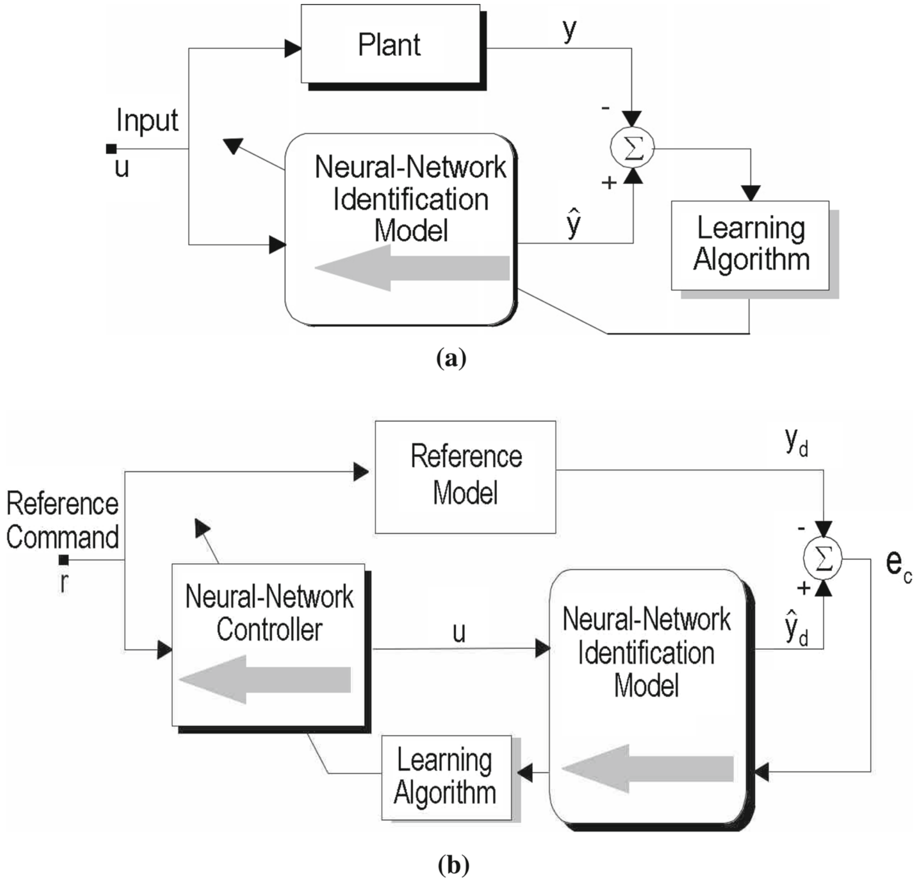


Fig. 11 The control structure of indirect model-following control system **a** system identification and **b** controller training

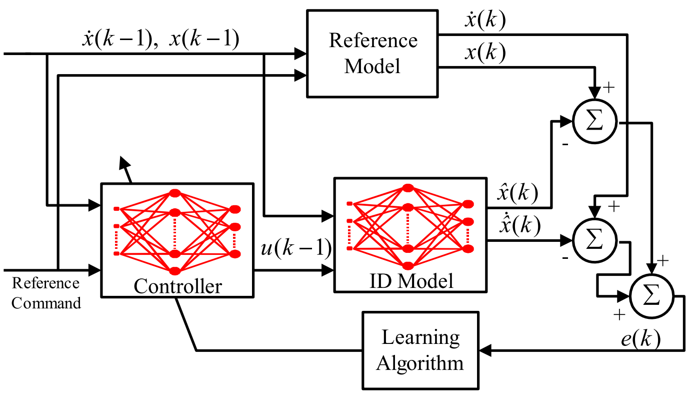


Fig. 12 The schematic diagram of the two-stage model controller design process

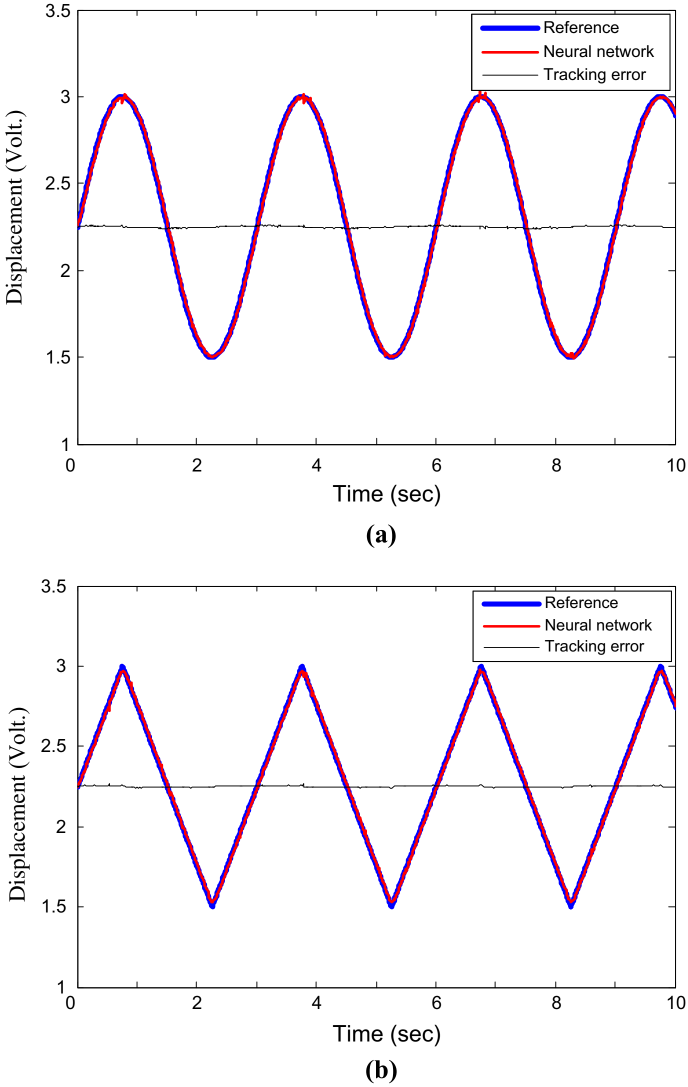


Fig. 13 Position tracking control under the periodic reference input

to predict the present system state [$y(k)$], respectively. The schematic of the neural network controller design is similar to that shown in Fig. 12. Experimental observation reveals that the first vibration mode of the shaking table dominates over other modes. Thus, the reference model is chosen as a second-order system:

$$\ddot{x}_m(t) + 5.0\dot{x}_m(t) + 142.5x_m(t) = 142.5u(t) \tag{19}$$

wherein the damping coefficient of the model is selected to be higher than that of the uncontrolled plant and is limited by the undamped natural frequency near 1.9 Hz. Note that a considerably damped reference model can always be selected and the neural controller will

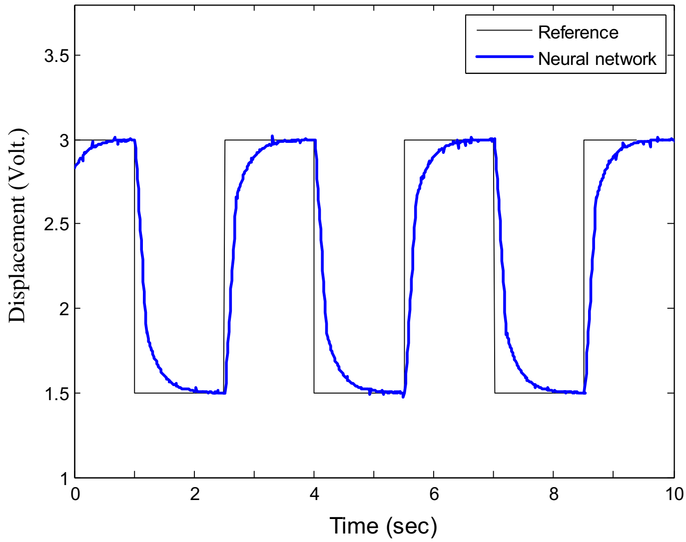


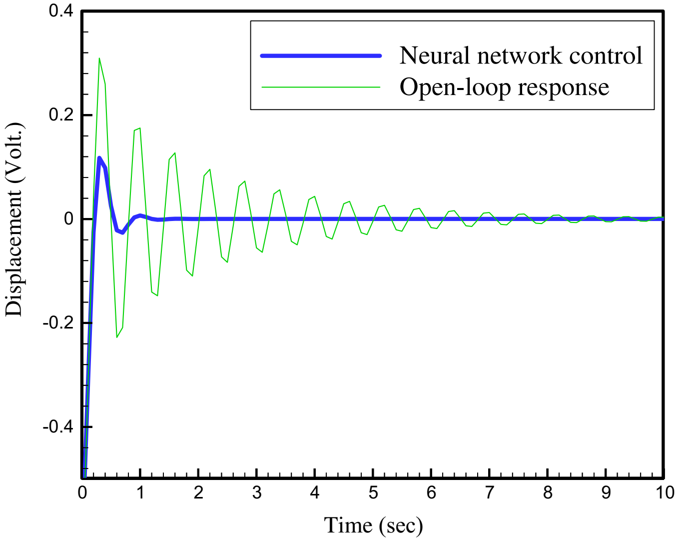
Fig. 14 A tracking control experiment of the shaking table by using the neural network with modified Newton method and genetic algorithm controller

remain effective. However, the control input may be saturated. The damping ratio is limited by the capability limit of the AC servo motor.

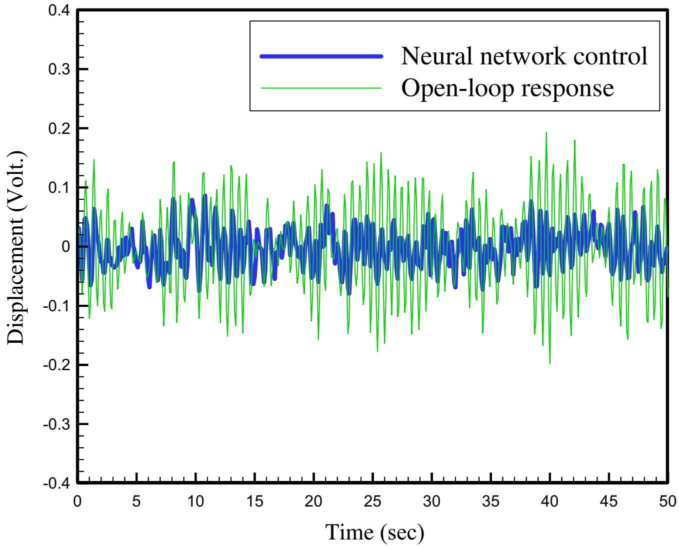
The previous section provided a discussion of the shaking table test. A [3–3–1] neural network with modified Newton method and genetic algorithm is used to train the controller with the data sampling rate of 10 Hz. Thus, the first mode can be controlled effectively without exciting the other modes that may complicate the control experiment. The transfer function of the hidden layer neurons is a hyperbolic tangent function. The neural controller has three input nodes for receiving system displacement, velocity, and command input and has one output node for the control input signal. The desired output set is then calculated through the fourth-order Runge–Kutta method from Eq. (1) every 0.1 s with the input set as the initial condition. A total of 1000 training sets that is uniformly distributed in the possible tracking range defined above is generated and collected in the initial phase for controller training.

Some experiments are employed to illustrate the effectiveness of the two-stage model controller. Although numerous works have focused on the tracking control and vibration suppression of structural systems, the application of neural networks in the positional tracking and vibration suppression of shaking tables remains limited. The schematic of the positional tracking experiment is shown in Fig. 7. A low-frequency accelerometer (KISTLER 8628B5) attached at the top floor acquires the acceleration. The measurement of accelerometer is then integrated to yield the velocity signal by the integrator (G-TECH 350133) which is then integrated and filtered to displacement signal. One linear wire potentiometer (HPS-M-10-5K-F) fixed at the bottom floor through a analog I/O board (National Instruments PCI-6052E) is used to measure the displacement of shaking table. The data acquisition and control system consists of LAB-VIEW6.1 (National Instruments) and a PCI-6052E multifunction I/O board.

Figure 13a, b show the reference sigmoidal track and the response trajectory with the two-stage model under the nominal condition. The shaking table position is controlled effectively. The corresponding performances of positional control under the neural network with modified



(a)

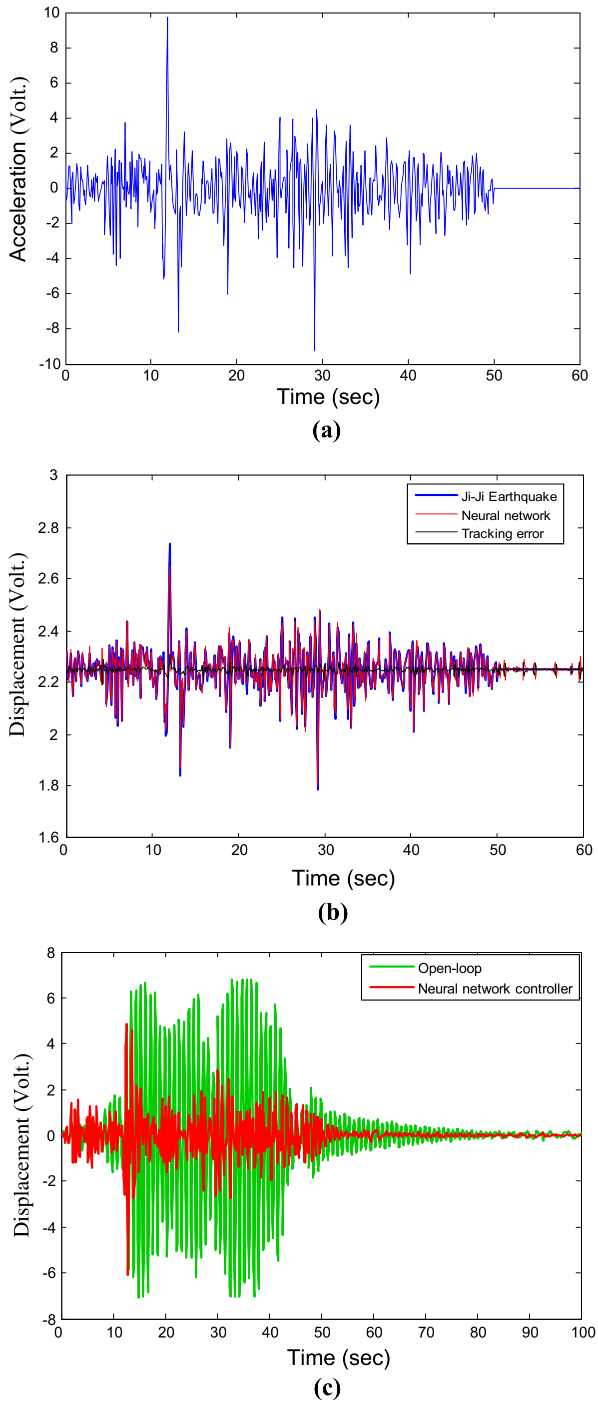


(b)

Fig. 15 A time-domain comparison of the open- and the closed-loop building structure responses under **a** initial displacement and **b** resonance excitation

Newton method and genetic algorithm control are shown in Fig. 14. Figure 15 illustrates the response of the vibration suppression of a building structure under initial displacement and resonance excitation. To further validate the effectiveness of the neural network with modified Newton method and genetic algorithm controller, positional tracking under Ji–Ji Earthquake, Taiwan on Sep. 21, 1999 of Richer scale 7.1 is conducted. Figure 16a shows the time history of ground acceleration record in which the maximum value is 862 gal (1 gal =

Fig. 16 a Time history of ground acceleration record of Ji–Ji Earthquake, Taiwan (Sept. 21, 1999). **b**, **c** A time-domain comparison of the neural network with modified Newton method and genetic algorithm controller under Ji–Ji Earthquake, Taiwan (Sept. 21, 1999)



1 cm/s^2). Because the experimental model can not sustain such earthquake magnitude, the maximum acceleration is reduced to about 20%, 172 gal in order to simulate the earthquake signal. Figure 16b indicates the displacement vibration amplitude is seen that there is very small difference between these two traces and Fig. 16c shows the vibration response of the building structure vibration suppression under Ji–Ji Earthquake can reduce largely by using the neural network with modified Newton method and genetic algorithm controller. The above experimental results show that satisfactory performance can be achieved by using the neural network with modified Newton method and genetic algorithm controller.

6 Conclusion

In this paper, system modeling of nonlinear dynamic system is conducted by the identification using neural networks with modified Newton method and genetic algorithm. A [3–3–2] two-stage model has been employed to identify successfully the dynamics of a shaking table and a building structure. It is should be noted that no assumption is required during modeling and training phase. Accuracy of the neural network with modified Newton method and genetic algorithm identification is validated by above two cases.

A neural network with modified Newton method and genetic algorithm controller based on indirect linear model-following control (LMFC) has been presented. A three-layer [3–3–1] two-stage model controller is developed for the tracking control of a shaking table and the building structure vibration suppression by using an AMD system. The neural controller with displacement feedback from potentiometer measurements can then minimize the positional error and suppress the vibration by using modified Newton method and genetic algorithm. Experimental verifications show that the peak-to-peak amplitude of the structure response under Ji–Ji Earthquake (Sept. 21, 1999) excitation, respectively. The experimental results show that the neural network with modified Newton method and genetic algorithm controller is effective in tracking control and vibration suppression applications.

Acknowledgements The author is grateful to the reviewers and AE for their exceptional efforts in enhancing the style and clarity of this paper.

References

Amini F, Chen HM, Qi GZ, Yang JCS (1997) Generalized neural network based model for structural dynamic identification, analytical and experimental studies. In: Proceedings of the intelligent information systems, pp 138–142

Catalogue of Mitsubishi Heavy Industries, LTD (2019) Tokyo, Japan. https://www.mhi-ms.com/products/steelstructures/vibrationcontrol/classification.html

Chen CJ (2012) Structural vibration suppression by using neural classifier with genetic algorithm. Int J Mach Learn Cybern 3(3):215–221

Chen CJ, Yang SM (2014) Application neural network controller and active mass damper in structural vibration suppression. J Intell Fuzzy Syst 27(6):2835–2845

Chen CJ, Yang SM, Wung ZC (2009a) A neuro-fuzzy system and its application to vibration suppression. J Aeronaut Astronaut Aviat 41(3):195–202

Chen CJ, Yang SM, Wang ZC (2009b) System identification by neuro-fuzzy model with Sugeno and Mamdani fuzzy rules. J Aeronaut Astronaut Aviat 41(4):263–270

Chen CJ, Yang SM, Chen CY (2013) Development of a rule selection mechanism by using neuro-fuzzy methodology for structural vibration suppression. J Intell Fuzzy Syst 25(4):881–892

Chen CJ, Yang SM, Lin SG (2014) Development of a self-organized neuro-fuzzy model by using genetic algorithm for system identification. J Aeronaut Astronaut Aviat 46(4):281–290

Goode PV, Chow MY (1995a) Using a neural/fuzzy system to extract heuristic knowledge of incipient faults in induction motors—part I: methodology. IEEE Trans Ind Electron 42:131–138
Goode PV, Chow MY (1995b) Using a neural/fuzzy system to extract heuristic knowledge of incipient faults in induction motors—part II: application. IEEE Trans Ind Electron 42:139–146
Grcar B, Cafuta P, Znidaric M, Gausch F (1996) Nonlinear control of synchronous servo drive. IEEE Trans Control Syst Technol 4:177–184
Haessig DA, Friedland B (1991) On the modeling and simulation of friction. Trans ASME J Dyn Syst Meas Control 113(5):354–362
Han M, Han G, Jiang X, Lian Z (2001) Study of dynamic response of dams with neural network. IEEE Int Conf Syst Man Cybern 1:134–139
Jagannathan S (2001) Control of a class of nonlinear discrete-time systems using multilayer neural networks. IEEE Trans Neural Netw 12(5):1113–1120
Jeng CA, Yang SM, Lin JN (1997) Multi-mode control of structures by using neural networks with Marquardt algorithms. Int J Intell Mater Syst Struct 8(12):1035–1043
Johnson GT, Lorenz RD (1992) Experimental identification of friction and it's compensation in precise, position controlled mechanisms. IEEE Trans Ind Appl 28:1392–1398
Korkmaz S (2011) A review of active structural control: challenges for engineering informatics. Comput Struct 89:2113–2132
Lazar IF, Neild SA, Wagg DJ (2013) Using an inerter-based device for structural vibration suppression. Earthq Eng Struct Dyn. https://doi.org/10.1002/eqe.2390
Levin AU, Narendra KS (1996) Control of nonlinear dynamical systems using neural networks—part II: observability, identification, and control. IEEE Trans Neural Netw 7:30–42
Li L, Song G, Ou J (2010) Nonlinear structural vibration suppression using dynamic neural network observer and adaptive fuzzy sliding mode control. J Vib Control 16(10):1503–1526
Liu J, Li L, Dai J, Ou J (2011) A single-degree-of-freedom system using a time-delayed control algorithm: principle and experimental study. J Vib Control 17(12):1840–1850
Marino R, Peresada S, Valigi P (1993) Adaptive input–output linearizing control of induction motors. IEEE Trans Autom Control 38:208–221
Narendra KS, Parthasarathy K (1990) Identification and control of dynamical systems using neural networks. IEEE Trans Neural Netw 1(1):4–27
Ohnishi K, Ueda Y, Miyachi K (1986) Model reference adaptive system against rotor resistance variation in induction motor drive. IEEE Trans Ind Electron 4:217–223
Omatu S, Khalid M, Yusof R (1997) Neuro-control and its applications. Springer, Berlin
Sastry PS, Santharam G, Unnikrishnan KP (1994) Memory neuron networks for identification and control of dynamical systems. IEEE Trans Neural Netw 5:306–319
Song L, Li G, Ou J (2011) Hybrid active mass damper (AMD) vibration suppression of nonlinear high-rise structure using fuzzy logic control algorithm under earthquake excitations. Struct Control Health Monit 18(6):698–709
Soong TT (1988) Active structural control in civil engineering. Eng Struct 10:78–84
Tseng HC, Hwang VH (1993) Servo controller tuning with fuzzy logic. IEEE Trans Control Syst Technol 1:262–269
Tzou YY (1996) DSP-based robust control of an AC induction servo drive for motion control. IEEE Trans Control Syst Technol 4:614–626
Utkin VI (1993) Sliding mode control design principles and applications to electric drives. IEEE Trans Ind Electron 40:23–36
Wang L, Wang X, Yang Y, Li Y, Chang X (2018) Active force control of structure-borne sound based on robust optimization subjected to an irregular cavity with uncertainties. Aerosp Sci Technol 73:318–331
Wit CD, Olsson H, Astrom KJ, Lischinsky P (1995) A new model for control of systems with friction. IEEE Trans Autom Control 40:419–425
Yang YP (1993) Adaptive velocity control of DC motor with coulomb friction identification. Trans ASME J Dyn Syst Meas Control 115(2):95–102
Yang SM, Lee GS (1997) Vibration control of smart structure by using neural networks. Trans ASME J Dyn Syst Meas Control 119(1):34–39
Yang SM, Lee GS (1998) System identification of smart structures using neural networks. Int J Intell Mater Syst Struct 8(10):883–890
Yang SM, Chen CJ, Huang WL (2006) Structural vibration suppression by a neural-network controller with a mass-damper actuator. J Vib Control 12(5):495–508
Yang SM, Chen CJ, Chang YY, Tung YZ (2007) Development of a self-organized neuro-fuzzy model for system identification. J Vib Acoust ASME 129:507–513

Yoshida, K. (1994). Frontiers of active motion and vibration control. In: Proceedings of the 2nd international conference on motion and vibration control, pp 1–10

Yu W, Li X (2001) Some new results on system identification with dynamic neural networks. IEEE Trans Neural Netw 12(2):412–417

Zhong HY, Behera AK, Rashid MH (1991) 8096 microcontroller based field acceleration method control for induction motor with new digital PWM inverter technique. In: Conference Rec. IEEE-IAS annual meeting, 1662–1668