A novel approach to model magneto-rheological dampers using EHM with a feed-forward neural network

Kittipong Ekkachai[a,*]**, Kanokvate Tungpimolrut**[b]**, Itthisek Nilkhamhang**[a]

[a] Sirindhorn International Institute of Technology, Thammasat University, Thailand
[b] National Electronics and Computer Technology Centre, Thailand

*Corresponding author, e-mail: kittipong.ekkachai@gmail.com

ABSTRACT: This paper proposes a novel method for modelling magneto-rheological (MR) dampers. It uses an elementary hysteresis model (EHM) with a feed-forward neural network (FNN) to capture hysteresis characteristics of an MR damper, and another FNN to determine the current gain. These parts can be trained separately, thus reducing the size of the training dataset. The inputs of the proposed model include velocity, acceleration, and current to estimate the generated damping force. Unlike previous FNN models, this model does not require force sensor inputs. Simulation results show the high performance of the proposed EHM-based FNN when compared to conventional methods such as a recurrent neural network.

KEYWORDS: elementary hysteresis model, system identification

INTRODUCTION

Magneto-rheological (MR) dampers are semi-active control devices that offer several advantages including quick response time, ease of design, very low input power requirement to produce high yield strength, and stable hysteretic behaviour over a broad range of temperatures. They consist of cylinders containing MR fluid with micron-sized, magnetically polarizable particles dispersed in a carrier medium such as mineral or silicone oil[1]. MR dampers can change between liquid and semi-solid characteristics with different damping coefficients depending on the applied strength of a magnetic field. This magnetic field can be controlled by applying a voltage or current. MR dampers are used in a wide range of applications, such as suspension systems in automobiles, seismic protection, and human prosthetics. In automotive applications, MR dampers are installed between the wheel and chassis as part of the suspension system to reduce seat vibration. This provides performance advantages over conventional friction dampers because the damping values can be varied to suit many frequency ranges[2]. Other researchers have also installed MR dampers in buildings and structures to reduce the effects of seismic activities[3]. State-of-the-art prosthetic knee devices employ MR dampers to generate variable knee friction for different phases of the ambulation cycle. This offers amputees the ability to walk more naturally and with greater comfort[4]. These applications show the advantages of MR dampers over conventional

dampers. However, it is a highly nonlinear dynamic system that requires accurate models to develop an efficient controller.

Models of MR dampers fall into two categories: parametric and non-parametric models. Parametric models are based on mechanical elements represented by springs, viscosity, and friction. The most well known parametric model is the modified Bouc-Wen model that consists of 14 parameters determined from curve fitting of experimental data[1]. Modified Bouc-Wen gave accurate results but the limitation of this model is the linear dependency between damping force and applied voltage. To remove this restriction, a third-order polynomial equation can be used to represent the nonlinear relationship between force and applied current. The third-order polynomial modified Bouc-Wen[5] increases the total number of parameters to 19. Other parametric models such as modified Lugre[6] and hyperbolic tangent models[7] were proposed as simpler alternatives to the modified Bouc-Wen. These parametric modelling methods require assumptions about the structure of the mechanical model, and accuracy can decrease if the initial assumptions about model structure are flawed, or if the proper constraints are not applied to the parameters[8]. Another type of MR damper model employs non-parametric approaches such as a feed-forward neural network (FNN)[9], recurrent neural network (RNN)[10], neuro-fuzzy[11] and black-block model[8]. Non-parametric models generally require more experimental data for training than parametric models.

In this paper, a non-parametric FNN using an elementary hysteresis model (EHM) is proposed. EHM is used to represent hysteresis characteristics of the MR damper. Conventional FNN can estimate damping force accurately but requires force information from a previous state[9]. To do this, force sensors must be installed in the system, otherwise the damping force cannot be predicted accurately because it can only approximate one-to-one or multiple-to-one mapping, whereas hysteresis requires multi-valued mapping. RNN is an alternative model that does not require force sensors[10]. However, it is complex and requires more time to train. The EHM-based FNN is proposed to solve these problems. The model consists of a hysteresis model and a gain function. The hysteresis model is constructed as an EHM-based FNN[12, 13] whose inputs contain system excitation variables. The gain function employs another FNN that is used to determine the force gain caused by electromagnetic effects, which is chosen as the applied current to the MR damper. The model does not require external force information unlike the previous FNN model. The EHM-based FNN also uses a simple training process because both parts of the model can be trained separately. The proposed model is compared to previous FNN and RNN models through simulations. The results show that the accuracy of the proposed model outperforms the RNN model for various types of datasets. Details of the EHM-based FNN implementation are shown in the following sections.

MAGNETO-RHEOLOGICAL DAMPER

A MR damper typically is a cylinder that contains MR fluid whose characteristics change depending on an applied magnetic field. The current (or voltage) controller input is varied to regulate its damping force. Due to the highly nonlinear dynamic properties of the MR damper, system identification is required for accurate control. The behaviour of the MR damper depends on three important variables: (1) excitation such as displacement, velocity and acceleration, (2) applied current and (3) damping force. The relationship between damping force and velocity shows hysteresis loops whose shapes vary according to the applied current. Many models have been created to describe its behaviour. The most commonly used model is the modified Bouc-Wen proposed by Spencer et al[1]. However, the relationship between force and current in the modified Bouc-Wen model is linear, unlike the actual behaviour of the MR damper. Therefore, a third-order polynomial equation is used to adapt the modified Bouc-Wen to represent the nonlinear dependency between applied current and force[5], which is
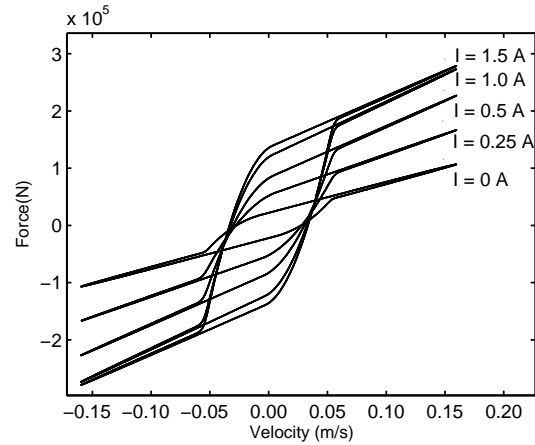

Fig. 1 Hysteresis at different current values generated by the third-order polynomial modified Bouc-Wen model.

governed by the following seven equations:

$$F = c_1 \dot{y} + k_1(x - x_0) \tag{1}$$

$$\dot{y} = \frac{1}{(c_0 + c_1)}[\alpha z + c_0 \dot{x} + k_0(x - y)] \tag{2}$$

$$w = \dot{x} - \dot{y} \tag{3}$$

$$\dot{z} = \gamma |w||z|z|^{(n-1)} - \beta w|z|^n + Aw \tag{4}$$

$$\alpha = \alpha_a + \alpha_b I + \alpha_c I^2 + \alpha_d I^3 \tag{5}$$

$$c_1 = c_{1a} + c_{1b}I + c_{1c}I^2 + c_{1d}I^3 \tag{6}$$

$$c_0 = c_{0a} + c_{0b}I + c_{0c}I^2 + c_{0d}I^3 \tag{7}$$

where F, x, $\dot{x}$ and I are force, displacement, velocity, and applied current, respectively. The 19 physical parameters can be identified from experimental data by using optimization techniques. Parameters of a large-scale 20 t MR damper[5] are given as $A = 2679.0$ m^{-1}, $\beta = 647.46$ m^{-1}, $\gamma = 647.46$ m^{-1}, $k_1 = 617.31$ N/m, $k_0 = 137\,810$ N/m, $n = 10$, $x_0 = 0.18$ m, $\alpha_a = 15\,114$, $\alpha_b = 168\,326$, $\alpha_c = -87\,071$, $\alpha_d = 16\,566$, $c_{1a} = -2\,791\,630$, $c_{1b} = 48\,788\,640$, $c_{1c} = 5\,334\,183$, $c_{1d} = -9\,363\,108$, $c_{0a} = 457\,741$, $c_{0b} = 1\,641\,376$, $c_{0c} = -1\,545\,407$ and $c_{0d} = 437\,097$. The third-order polynomial modified Bouc-Wen generates hysteresis as shown in Fig. 1 when excited at 2 Hz and varying the current from 0–1.5 A. The model is better able to simulate the nonlinear relationship between force and applied current than the modified Bouc-Wen. All validation and training datasets in the simulations are generated from a third-order polynomial modified Bouc-Wen model.

ELEMENTARY HYSTERESIS MODEL

The elementary hysteresis model (EHM) was proposed by Tong et al[12] and improved by Ma et al[13]

to construct a one-to-one mapping between inputs and outputs of a neural network. When an input extremum (local minimum or local maximum point) occurs, the output will move along a regular curve, such as a monotone conic path. This produces a branch of major or minor loop. By continuous transformation, an arbitrary number of minor loops can be obtained so that an EHM can be constructed. The constructed model is used as a bridge between the input signals and the hysteresis output.

Calculation of EHM

The EHM signal is calculated using a parabolic function and defined as:

$$f(x_i(t))$$
$$= \begin{cases} f(x_e) + a[x_i(t) - x_e]^2; x_i(t+1) > x_i(t) \\ f(x_e) - a[x_e - x_i(t)]^2; x_i(t+1) < x_i(t) \end{cases}$$
(8)

where x_i is the current input, x_e is the input at the extremum adjacent to the current input, $f(x_i)$ is the output of the current point and $f(x_e)$ is the output of the point x_e, a is a constant parameter of the parabolic equation. Consider the input profile shown in Fig. 2(a). The resulting EHM loop is given in Fig. 2(b), where the major loop traverses via point a to f and returns to a. While EHM output travels in this major loop, if the input signal reaches an extremum point, a minor loop will be generated. Considering Fig. 2(a), the initial extremum point is set to a as the input signal starts to increase from point a to point b. The EHM output in Fig. 2(b) therefore generates a parabolic curve from point a to point b. After that, the input signal decreases from point b to point c, and the extremum point is changed to point b, and EHM switches to a decreasing function. The next calculation process will go on this manner until the input signal traverses back to point a. In some cases, the EHM function may generate minor loops inside other minor loops, as seen in loop g-h-i- j-g. When the input signal traverses back to its outer loop, the current extremum point is set as the extremum point of that outer loop.

EHM-based neural network hysteresis model

The EHM function can be combined with a neural network to approximate hysteresis types of behaviour. FNN with structure 2-n-1 was used successfully to approximate backlash-based hysteresis simulation model in a previous work[12]. The two inputs of their FNN consist of input signal and EHM signal that is
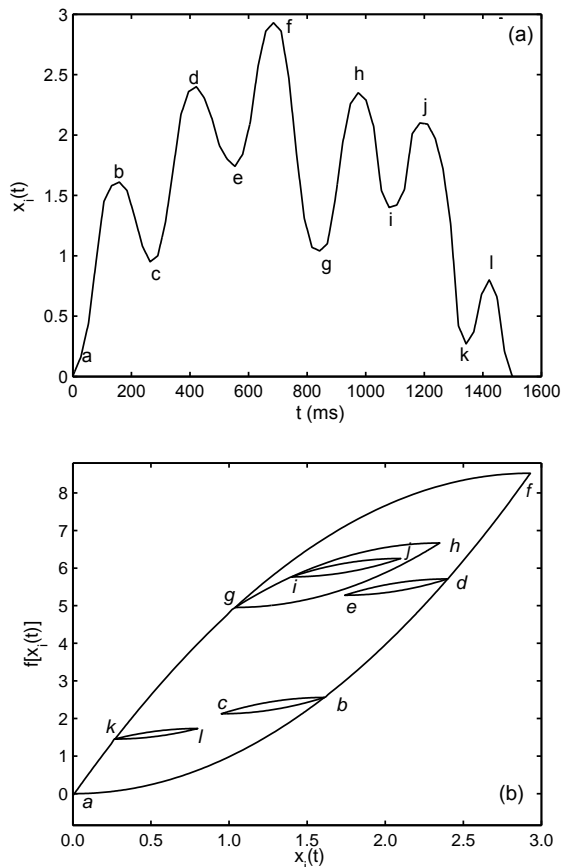

Fig. 2 (a) Example of an input signal and (b) the generated EHM curve.

generated from the input signal. MR damper hysteresis is similar to backlash, but the width of the loop also depends on excitation frequency[14]. Change of excitation frequency can be measured from acceleration. The proposed EHM-based neural network model for approximating MR damper hysteresis is shown in Fig. 3, with a 3-n-1 structure. The three inputs consist of velocity ($\dot{x}$), acceleration ($\ddot{x}$) and the EHM signal. The hidden layer includes a sufficient number of nodes (n) to approximate hysteresis and uses the tangent sigmoid function. The output layer uses the linear transfer function to produce an estimated force (F_h).

MODEL STRUCTURE

Proposed MR damper model with EHM-based neural network hysteresis

The MR damper is a nonlinear dynamic system consisting of two kinds of inputs: (1) excitation and (2) applied current that generates a magnetic field. Some recent research[8,14] divide MR damper model into two
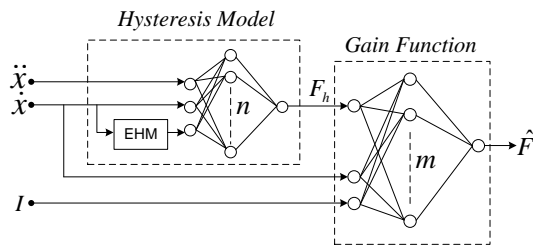
Fig. 3 The proposed model structure.

functions because the effect of applied current on the width of hysteresis loops is relatively small. A typical MR damper model is shown as follows:

$$F = f(I, x, \dot{x}, \ddot{x}) \qquad (9)$$

where F is the damping force, I is the applied current, x, $\dot{x}$ and $\ddot{x}$ are displacement, velocity and acceleration, respectively. The proposed MR damper model separates the damping force into two parts:

$$F_h = f_h(\dot{x}, \ddot{x}) \qquad (10)$$
$$F = f_I(F_h, I, \dot{x}) \qquad (11)$$

where f_h is the hysteresis function produced by excitation of the MR damper, f_I is the nonlinear gain function that depends on applied current and velocity. Each part of the proposed model is developed using FNN as shown in Fig. 3 and the details are described in following section.

The hysteresis model

The hysteresis model is created by EHM-based neural network described in the previous section. In the training process, the dataset must include all possible ranges of excitation that may be applied to the MR damper during actual operation. It is important to construct the boundary (major loop) of EHM. For example, in Fig. 2(b) the boundary of EHM is constructed between point a and point f. In the validation process, the EHM output of validation datasets must stay inside this boundary. Moreover, the starting point of EHM signal is important for calculating the next EHM output. If it starts from the wrong position then the estimation result will not be accurate. The method to determine EHM signal at the starting point requires information of actual force at the first extremum point. Optimization techniques can then be employed to determine the EHM signal at the first extremum point.

The gain function

The gain function (f_I) is the nonlinear function that scales the output of the hysteresis model (F_h) depending on the applied current. The maximum and minimum force of each hysteresis shape depends on applied current value as shown in Fig. 1. However, the gain between two hysteresis shapes is not equal at all velocity and is considered to be a nonlinear function. The proposed model uses another FNN to construct a gain function for estimating the total force. Inputs of the gain function consist of the applied current, velocity and F_h generated from the hysteresis model.

IMPLEMENTATION OF THE PROPOSED MODEL

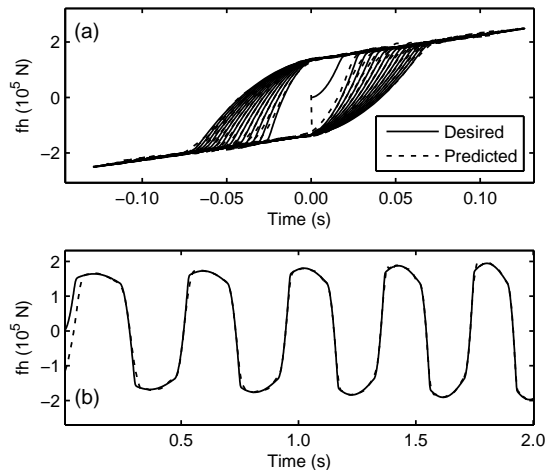**Implementation of the hysteresis model**

The constant value (a) in eq.(8) and the number of hidden nodes are determined in this section. In simulations, the value of a is varied from $a = 0.5$–2 without demonstrating significant differences, so $a = 1$ is arbitrarily set. Another question is how to find the best number of nodes. The proposed hysteresis model selects one hidden layer and varies the number of nodes from 5–65 nodes. The node number estimation process runs 5 times with different initial weight and random data division. To select the best number of hidden nodes, the lowest number that can adequately represent the training set will be selected to avoid generalization errors. A suitable node count (n) is determined by simulation as 45.

The hysteresis model is constructed with an FNN that is structured as a 3-45-1 network. The inputs of the network consist of velocity, acceleration and the EHM signal. The simulation is designed by using dataset generated from a large-scale MR damper which has parameters defined previously. Velocity is varied from -0.16 to 0.16 m/s and the frequency of piston movement between 1 and 5 Hz. The training dataset is a sinusoidal signal with different amplitude and frequency in 80 s range as in Table 1 and the applied current is fixed at 1.5 A. The training dataset is divided randomly into three groups consisting of training data, validation data and testing data with ratio 60:20:20, respectively. The Levenberg-Marquardt algorithm[15] is used as the training algorithm and mean square error is used as performance function. Maximum training iterations in the training process is 3000 and the process is stopped when the results of validation checks do not improve significantly within 6 iterations. In this paper, the training process was conducted 10 times and the weight of the hysteresis model was obtained from the best result. The average training time of the hysteresis model is approximately 21 min.

After the hysteresis model was trained, a 5-s sinusoidal signal that varies the amplitude from 0–

Table 1 Sinusoidal velocity signals used in training dataset of the hysteresis model.

Duration (s)	Amplitude (m/s)	Frequency (Hz)
10	$0.016t$	1
10	$-0.016t+0.16$	1
10	$0.016t$	2
10	$-0.016t + 0.16$	2
10	$0.016t$	3
10	$-0.016t + 0.16$	3
5	$0.032t$	4
5	$-0.032t + 0.16$	4
5	$0.032t$	5
5	$-0.032\,t + 0.16$	5

Fig. 4 The result of the hysteresis model is shown in (a) velocity domain and (b) time domain from 0–2 s.

0.16 m/s and frequency from 1–5 Hz is used as a validation dataset. The desired F_h is equal to forces generated by a large-scale MR damper with applied current at 1.5 A. Performance of the hysteresis model is evaluated from normalized root-mean-square error (NRMSE) between desired and predicted F_h values. The result of validation data is shown in Fig. 4 with an NRMSE of 0.92% when compared to the full range between maximum and minimum generated forces. It shows that the hysteresis model can predict F_h accurately except before the velocity reaches the first extremum point because of the limitation of EHM.

Implementation of the gain function

Another FNN is used to implement the gain function. It is applied to estimate damping force from F_h that is generated by the hysteresis model. The simulated

MR damper is operated at an applied current between 0 and 1.5 A and the hysteresis model is trained at 1.5 A. The gain function will scale down the value of F_h to be the estimated force at other applied current values. From simulation results, the gain FNN does not require many hidden nodes, with only 5 nodes being sufficient to estimate the gain ratio. Thus the structure of gain FNN is a 3-5-1 network with tangent sigmoid transfer function in the hidden layer and linear transfer function in the output layer.

The training dataset used to train the gain FNN consists of data shown in Fig. 1. This includes the damping force generated from two cycles of sinusoidal velocity at amplitude 0.16 m/s and frequency 2 Hz, and applied current at 0, 0.1, 0.25, 0.5, 0.75, 1, 1.25, and 1.5 A. The force generated at 1.5 A is used as F_h in the training process. All parameters and environment in the training process of the gain function is the same as the hysteresis model. Average time of the training process is 1.74 min.

The validation datasets consist of hysteresis loops generated at 0 A and 1.1 A. The results show that NRMSE of the predicted force compared to the desired force at 0 A is 0.42%, and at 1.1 A is 0.13%. The predicted force at 1.1 A is more accurate than at 0 A, because it is closer to the base current of 1.5 A. However, choosing the largest operating current as a base current gives an advantage, because it only scales down the output from the hysteresis model and does not amplify any errors.

VALIDATION OF THE PROPOSED MODEL AND COMPARISON

Validation of the proposed model

After the hysteresis model and the gain function are trained separately, the next step is to combine both parts together. From the previous section, the proposed model selects the number of hidden nodes in the hysteresis model (n) at 45 and the number of hidden nodes in the gain function (m) at 5. This section validates the model and compares it with conventional FNN and RNN that have been developed from previous work[9, 10] using two datasets.

The first dataset is a 2.5-s time span dataset. It consists of sinusoidal velocity at 4 Hz with amplitude 0.16 m/s while the applied current is a ramp function that increases from 0.1 A to 1.35 A. Dataset 1 is shown in Fig. 5(a) and tests the performance of the gain function at various current levels. The errors of the predicted force are calculated after the first extremum point as 1.95% when compared to the full range between maximum and minimum generated forces.
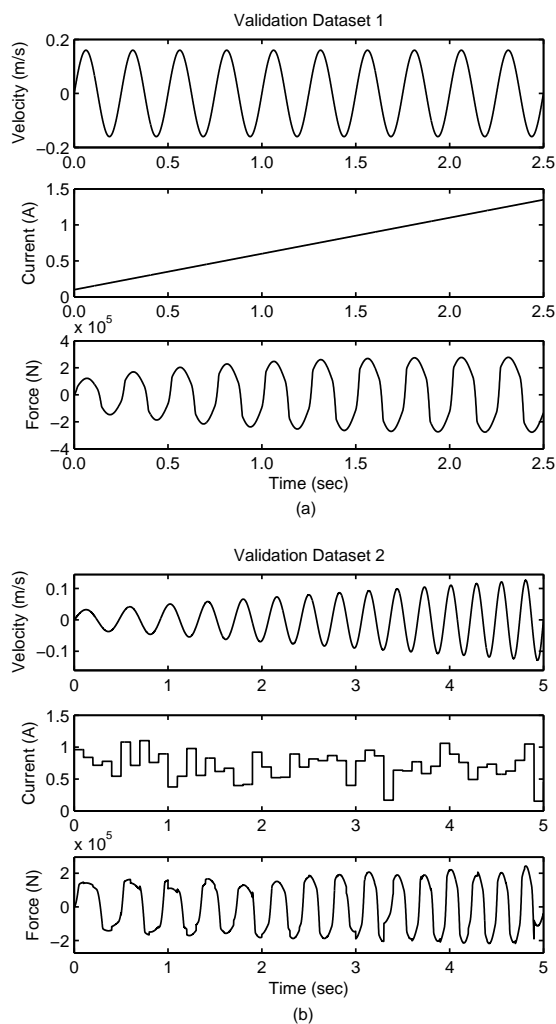
Fig. 5 (a) Validation dataset 1 and (b) dataset 2.

The second dataset is shown in Fig. 5(b). It is a 5-s time span dataset consisting of a chirp velocity of 2–4 Hz with ramp amplitude varying from 0–0.12 m/s, and the applied current is a random value between 0 and 1.5 A that changes every 0.1 s. This dataset verifies the performance of the hysteresis model by using a wide range of velocity frequency and amplitude. NRMSE of predicted force calculated after the first extremum is 0.89% when compared to the full range between maximum and minimum generated forces.

Previous feed-forward neural network setup

The previous FNN developed by Chang et al[9] is used for comparison with the proposed model. The structure of the previous FNN model is shown in Fig. 6(a) and consists of 6 inputs including two displacements, two currents and two force information from previous

states. The model has a hidden layer with 12 nodes using the tangent sigmoid transfer function, and an output layer using the linear transfer function. This model uses actual force from the previous states as inputs, thus requiring that a force sensor be installed for online measurement. The training dataset uses the same velocity and acceleration profiles as the dataset in Table 1 and duplicates these profiles 5 times with different applied currents such as 0, 0.25, 0.5, 1, and 1.5 A. The total dataset length covers a 400-s time span; which is larger than the training data of the hysteresis model. The training process and environment is the same as the process to train the hysteresis model.

Recurrent neural network setup

RNN is another method that can be used to model MR dampers. The architecture of RNN has one or more output feedback loop to estimate the next output. It is a powerful network that is widely used in many applications. The RNN model of MR damper which was proposed by Metered et al[10] has two kinds of external inputs; such as previously applied currents and past displacements, and uses the estimated force from prior iterations as an internal input. Simulation results indicated that the network using only displacement cannot provide accurate results, so the RNN is modified in this paper by replacing the displacement input with velocity and acceleration. The modified RNN is shown in Fig. 6(b) and consists of 15 input signals including previous current, velocity, acceleration and the estimated force, two hidden layers with 18 nodes each that uses the tangent sigmoid transfer function, and an output layer using the linear transfer function. In this paper, RNN is developed by using a nonlinear autoregressive network with exogenous inputs (NARX). The training dataset is the same as data used in the training process of the previous FNN. Since the training dataset is very large, the model cannot be trained using conventional NARX method. Thus the network is constructed in a series-parallel architecture and trained using the actual force. After the network is completely trained, it is converted back to an NARX model. The training process uses the same parameter and environment as the training process of the hysteresis model, and is repeated 15 times. The best result is selected as the RNN model.
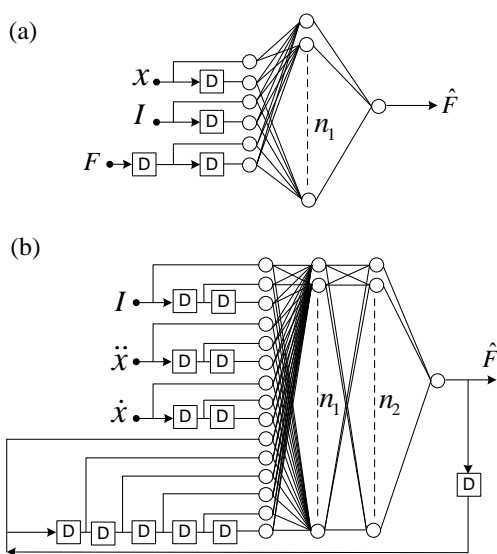
Result and comparison

The results of the previous FNN model, RNN model and proposed model are shown in Table 2, where NRMSE values of these models are calculated after the first extremum point. Since the previous FNN

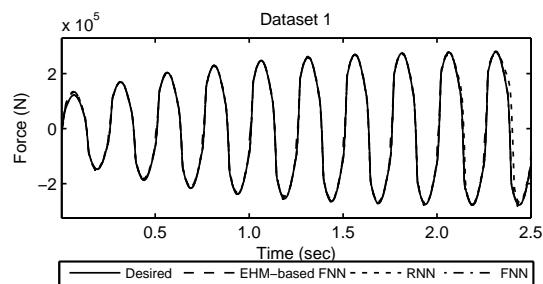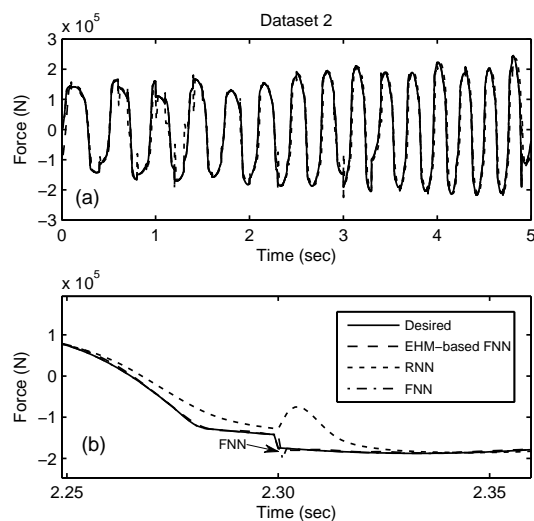**Table 2** Average training time and validation results.

Methods	Average Training Time[a]	Normalized Root-Mean-Square Error[b]	
		Dataset 1	Dataset 2
FNN	26.46 min	0.091%	0.063%
RNN	7 h 27.48 min	4.43%	8.07%
EHM-based FNN	21.23 min + 1.74 min	1.95%	0.89%
	(the hysteresis model + the gain function)		

[a] Training processes conducted using MATLAB running on Windows 7 with CPU Intel i7-3.4Hz RAM 8 GB.
[b] Error is calculated by root-mean-square error divided by a max-min range of its dataset.

Fig. 6 (a) The previous FNN and (b) RNN architecture (D represents delay by once step).

Fig. 7 Result of dataset 1.

Fig. 8 (a) Result of dataset 2 and (b) expanded result from 2.25–2.36 s.

model uses a force sensor to provide past force information, the results are significantly better than the other two methods. Therefore, it can be used as a benchmark for comparison purposes. The advantage of the proposed model can clearly be seen when compared with the RNN model, since both methods do not require a force sensor to be installed. In simulation, the proposed model estimates force better than the RNN model for both datasets. Fig. 7 and Fig. 8(a) show the results of dataset 1 and dataset 2, respectively. In dataset 1, the proposed model is slightly better than the RNN model, but the RNN error grows larger as the current increases. In dataset 2, the error of the proposed model is smaller than the RNN, especially when the applied current is changed from one value to another. For example, in Fig. 8(b), when the applied current changes from 0.53 A to 0.9 A at time 2.3 s, the RNN result contains higher error. RNN

responds slower than the proposed model because it uses predicted force from previous iterations. When a change in the applied current occurs, more time is needed to adapt the network.

CONCLUSIONS

This paper proposes an MR damper model that consists of the hysteresis model and the gain function. The proposed model estimates force less accurately

compared to the previous FNN model, but provides an advantage in that it does not require a force sensor to measure external force information from previous iterations. Hysteresis prediction is dependent on memory of previous states to generate the next state. The RNN model and the proposed model have different mechanisms to solve this problem. The RNN model uses predicted force from previous iteration as feedback to the system. In contrast, the proposed model uses EHM with a parabolic function to capture hysteresis characteristic. Simulation results show that the proposed model can predict the damping force more accurately when compared to RNN. Secondly, the proposed model allows for an easier training process compared to the previous FNN and RNN because the model is divided into two parts: the hysteresis model and the gain function. These two parts can be trained separately, so reducing the size of the required training dataset.

REFERENCES

1. Spencer BF, Dyke SJ, Sain MK, Carlson JD (1997) Phenomenological model of a magnetorheological damper. *J Eng Mech* **123**, 230–8.

2. Savaresi SM, Poussot-Vassal C, Spelta C, Sename O, Dugard L (2010) *Semi-Active Suspension Control Design for Vehicles*, 1st edn, Butterworth-Heineman, Elsevier, pp 15–39.

3. Bharti SD, Dumne SM, Shrimali MK (2010) Seismic response analysis of adjacent buildings connected with MR dampers. *Eng Struct* **32**, 2122–33.

4. Johansson JL, Sherrill DM, Riley PO, Bonato P, Herr H (2005) A clinical comparison of variable-damping and mechanically passive prosthetic knee devices. *Am J Phys Med* **84**, 563–75.

5. Yang G, Spencer BF, Carlson JD, Sain MK (2002) Large-scale MR fluid dampers: modeling and dynamic performance considerations. *Eng Struct* **24**, 309–23.

6. Sakai C, Ohmori H, Sano A (2003) Modeling of MR damper with hysteresis for adaptive vibration control. *Decis Contr* **4**, 3840–5.

7. Kwok NM, Ha QP, Nguyen TH, Li J, Samali B (2006) A novel hysteretic model for magnetorheological fluid dampers and parameter identification using particle swarm optimization. *Sensor Actuator Phys* **132**, 441–51.

8. Truong DQ, Ahn KK (2011) Nonlinear black-box models and force-sensorless damping control for damping systems using magneto-rheological fluid dampers. *Sensor Actuator Phys* **167**, 556–73.

9. Chang CC, Roschke P (1998) Neural network modeling of a magnetorheological damper. *J Intell Mater Syst Struct* **9**, 755–64.

10. Metered H, Bonello P, Oyadiji S (2010) The experimental identification of magnetorheological dampers and evaluation of their controllers. *Mech Syst Signal Process* **24**, 976–94.

11. Schurter KC, Roschke PN (2000) Fuzzy modeling of a magnetorheological damper using ANFIS. *Fuzzy Syst Conf* **1**, 122–7.

12. Tong Z, Tan Y, Zeng X (2005) Modeling hysteresis using hybrid method of continuous transformation and neural networks. *Sensor Actuator Phys* **119**, 254–62.

13. Ma L, Tan Y, Chu Y (2008) Improved EHM-based NN hysteresis model. *Sensor Actuator Phys* **141**, 6–12.

14. Wang DH, Liao WH (2011) Magnetorheological fluid dampers: a review of parametric modeling. *Smart Mater Struct* **20**, 023001.

15. Hagan MT, Menhaj MB (1994) Training Feedforward Networks with the Marquardt Algorithm. *IEEE Trans Neural Network* **5**, 989–93.