Original Article

ANN Model for Deflection Analysis of Superelastic Shape Memory Alloy RC Beams

Y.I. Elbahy

M.E.Sc. Candidate, Department of Civil and Environmental Engineering,

The University of Western Ontario

London, Ontario, Canada, N6A 5B9

M. Nehdi

Professor, Department of Civil and Environmental Engineering,

The University of Western Ontario

London, Ontario, Canada, N6A 5B9

M.A. Youssef

Associate Professor, Department of Civil and Environmental Engineering,

The University of Western Ontario

London, Ontario, Canada, N6A 5B9

Corresponding Author: **M.A. Youssef**

Email: youssef@uwo.ca,

Fax: 519-661-3779,

Phone: 519-661-2111, Ext: 88661

Word Count: (8637)

ABSTRACT

The need for a new model capable of accurately predicting the deflection of shape memory alloy (SMA) reinforced concrete (RC) members is clear from the results obtained in the companion paper. In this paper, artificial neural networks (ANN) are utilized to develop such a model. The objective is developing a design tool for calculating a reduction factor β to be used in the calculation of the effective moment of inertia for SMA RC members. First, a database was developed using the results obtained from the parametric study reported in the companion paper. The main factors affecting the moment of inertia have been considered. The network architecture that results in the optimum performance was selected and trained. After demonstrating the network's ability to predict output data for unfamiliar input data, the network was used to develop a design chart that provides the reduction factor β as a function of the reinforcement ratio and the reinforcement modulus of elasticity. A design example is discussed to show the advantages of using the developed design chart over existing models.

Keywords: shape memory alloy, reinforced concrete, artificial neural networks, load-deflection, moment of inertia.

INTRODUCTION

Shape memory alloys (SMAs) are unique alloys with the ability to undergo large deformations, and return to their undeformed shape upon unloading or heating. Four properties of SMAs have motivated researchers to utilize them in civil engineering: (i) large recoverable strain approaching a value of 10% (Alam et al. 2007), (ii) absorption of large amounts of strain energy under cyclic loading (Dolce and Cardone 2001, Piedboeuf and Gauvin 1998, Grandhi and Wolons 1999), (iii) extraordinary fatigue resistance under repeated large strain cycles (Eggeler et al. 2004, Hornbogen 2004), and (iv) very good durability (Janke et al. 2005). These unique properties have motivated utilizing SMAs as primary reinforcement for reinforced concrete (RC) structures (Elbahy et al. 2008, Saiidi et al. 2007).

The deflection calculation of concrete flexural members depends on the flexural stiffness, which varies along the structural member because of the possible presence of flexural cracks in it. Hence, an effective moment of inertia, I_e which has an average value between the gross moment of inertia, I_g and the cracked moment of inertia, I_{cr} should be used for deflection calculations of cracked members. The ACI 318-05 (2005) and CSA-A23.3-04 (2004) design standards use the equation proposed by Branson (1963) for calculating the effective moment of inertia for steel-reinforced concrete sections, **Eq. [1]**.

$$I_e = \left(\frac{M_{cr}}{M_a}\right)^3 I_g + \left[1 - \left(\frac{M_{cr}}{M_a}\right)^3\right] I_{cr} \qquad \leq I_g \qquad [1]$$

Using different types of reinforcement having different mechanical properties requires assessing the applicability of such an equation. For instance, using **Eq. [1]** for calculating the effective moment of inertia of Fibre-Reinforced Polymer (FRP) RC members can result in a significant overestimation of the member's moment of inertia (Bischoff 2005, Bischoff and Scanlon 2007). This overestimation is attributed to the difference between the modulus of elasticity of steel and that of FRP bars. To overcome this problem, the ACI 440.1R-03 (2003) committee proposed a reduction factor β that is mainly dependent on the value of the modulus of elasticity of the FRP to be multiplied by the gross moment of inertia term in the Branson's equation in the case of FRP reinforcement. The ACI 440.1R-04 (2004) committee modified the equation proposed by ACI 440.1R-03 (2003) for the reduction factor β making it dependent on the reinforcement ratio ρ, rather than the modulus of elasticity E.

To date, there are no available models to calculate the deflection of SMA RC members. Since the modulus of elasticity of FRP is generally close to that of SMA in the pre-yielding zone of the stress-strain model, the ability of the models proposed by the ACI 440.1-R-03 (2003), and ACI 440.1-R-04 (2004) to predict the deflection of SMA RC members was evaluated in a companion paper (Elbahy et al. 2008b). The models gave good predictions for members having high reinforcement ratios ($\rho > 1.0\%$). However, for lightly reinforced members ($\rho < 0.5\%$), the models gave poor predictions. Although the ISIS design manual (2001) gave good predictions for lightly reinforced members ($\rho < 0.5\%$), it was found to significantly overestimate the deflection of members having high reinforcement ratios. Moreover, the equation proposed by ISIS design manual (2001) is based on mathematical derivations of the equations provided by CEP-FIP (1990) rather than modifying Branson's equation. The scope of this paper is modifying Branson's equation by providing and appropriate, more accurate, value for the reduction factor β.

Artificial intelligence (AI) is the study of mental processes through the use of computational models (Charniak and McDermott 1985). It attempts to simulate the human mental faculties through computing. It can also be defined as the science concerned with understanding the intelligence behaviour and how it can be artificially created (Smithers et al. 1990). The use of AI in solving problems and modeling applications in civil engineering has increased over the last decades (El-Chabib and Nehdi 2005, El-Chabib and Nehdi 2006).

This paper focuses on utilizing artificial neural networks (ANNs) to predict the deflection of SMA RC members. A new model is developed to predict the effective moment of inertia of SMA RC members. Similar to ACI 440.1-R-03 (2003) and ACI 440.1-R-04 (2004), the proposed model predicts a suitable, yet more accurate, value for the reduction factor β in case of SMA RC members.

ARTIFICIAL NEURAL NETWORKS

Artificial neural network modeling is inspired by the understanding and abstraction of the biological structure of neurons and the internal operations of the human brain (Haykin 1994). It is a highly non-linear system offering a substantial ability to solve complex computational tasks. For instance, ANN is capable to perform self-organizing, pattern recognition, and functional approximation.

Neural networks are built of neurons (or processing units) that are usually arranged in layers and are often connected to neurons in other layers. The function of these neurons is performing simple computations. Based on the activation level, each processing unit sends signals to other

units in the network. Through propagation, the network learns and adapts to new data examples and stores information about the weights of the connections between neurons. Thus, a neural network has the ability to learn the relationship between a set of inputs and the corresponding outputs. This gives the network the ability to produce appropriate output values when unfamiliar inputs are provided.

The basic features of an artificial neural network are discussed by Rumelhart et al. (1998) and include: (i) processing units, (ii) pattern of connectivity between these processing units, (iii) state of activation for each processing unit, (iv) propagation rules, (v) activation functions for each processing unit, and (vi) rule of learning.

Fig. 1 shows a simplified model for an artificial neuron (processing unit). Generally, each neuron j in layer l receives one or more inputs X_i^{l-1} from neurons in the previous layer (l-1) (Rumelhart et al. 1986). Then, the neuron j performs a simple computation to form a single net input U_j^l given by **Eq. [2]**.

$$U_{ji}^{l} = \sum_{i=1}^{n} W_{ji}^{l} \; X_i^{l-1} + \theta_j^{l} \qquad [2]$$

Where: W_{ji}^l is the connection weight (strength) connecting the neuron j in layer l to neuron i in layer (l-1), n is the number of neurons in layer (l-1), θ_j^l is a threshold value assigned to neuron j in layer l, and X_i^{l-1} is the input coming from neuron i in layer l-1 to neuron j in layer l.

The net input U_j^l is then modified using an activation function f to generate an output value Y_j^l (**Eq. [3]**).

$$Y_j^l = f(U_j^l) \quad [3]$$

Where: f is a nonlinear activation function assigned to each neuron in the network.

A commonly used activation (transfer) function is the continuous nonlinear sigmoid function. The advantage of this function lies in its ability to meet the differentiability requirements needed in the back propagation algorithm. The activation function can be represented by **Eq. [4]**.

$$f_j^l = \frac{1}{1 + e^{-(u_j^l - \theta_j^l)}} \quad [4]$$

The features explained by Rumelhart et al. (1986) can vary from one network to another. As a result, different types of networks can be obtained. These include the Hopfield network (Hopfield 1982), Bolzmann machines (Ackaly et al. 1985), the Kohonen network (Kohonen 1982), and the multi-layer feed-forward back-propagation neural networks (Rumelhart et al. 1986). Among these different types of networks, the multi-layer feed-forward back-propagation network is the most commonly used in engineering applications. The concept and methodology of ANNs are discussed in more detail by Haykin (1994).

FEED-FORWARD BACK-PROPAGATION NEURAL NETWORKS

Feed-forward back-propagation (FFBP) or multi-layer perceptron (MLP) networks are the most widely used neural networks in engineering applications. They have the ability to perform non-linear transformations for functional approximation problems, recognize logic functions, and subdivide the pattern space for classification. FFBP or MLP networks have multiple layers and each layer may contain perceptrons or processing units. The perceptron is similar to the artificial neuron and was first introduced by Rosenblatt (1958).

A multi-layer FFPB neural network structure consists of an input layer, an output layer, and a number of hidden layers. **Fig. 2** shows a typical architecture of the multi-layer FFPB neural networks. Some researchers do not consider the input layer as an integral layer in the network architecture. However, it is agreed that the processing units in the input layer do not perform any computations. They only serve as a link between the input vector and processing units in the first hidden layer. Each layer may contain several processing units (neurons). The neurons in one layer are fully or partially connected to the neurons in the subsequent layer with different weights. No backward connections exist between neurons and no connections between neurons in the same layer are allowed in FFBP neural networks.

Based on the number of input and output parameters, the number of neurons in the input and output layers can be determined. However, there are no rules to decide on the optimum number of hidden layers or the optimum number of processing units (neurons) in the hidden layers. More research is still needed in this area. However, it is known that the number of hidden layers and neurons is generally dependent on the complexity of the problem. Although there are a few

recommendations in the literature to determine the suitable number of hidden neurons, the numbers of hidden layers and corresponding neurons is usually obtained by trial and error and usually depends on the experience of the user.

The performance of a trained FFPB neural network depends on the final weights (strengths) of connections between the processing units (neurons) in the different layers. Choosing a number of neurons smaller than the optimum number results in a smaller number of connections and their associated weights. Thus, the ability of the network to implement non-linear transformations for functional approximation problems is reduced. Choosing a number of neurons higher than the optimum number will result in a much higher number of connections and their associated weights. This will slow down the training process and reduce the ability of the network to generalize.

Generally, there are three important steps that should be carefully considered and addressed while constructing an effective ANN model (El-Chabib et al. 2003): (i) Database selection, (ii) Network architecture selection, and (iii) Network training and validation.

DATABASE SELECTION

The selection of the database to train a neural network is very critical. It should contain the necessary information to teach the network the relationships between the inputs and outputs. Two important principles must be considered: (i) the database should contain complete information about the relationships between the inputs and outputs, and (ii) the training data should be large enough and continuous for the training process.

The results from the parametric study conducted in the companion paper (Elbahy et al. (2008b)) were used to construct the used database. The main parameters in the parametric study were the basic factors affecting the effective moment of inertia I_e: cross-section dimensions (h and b), reinforcement ratio ρ, modulus of elasticity of the reinforcement E, concrete compressive strength $f_c^{'}$, and the applied load level M_a. The parametric study was carried out for simply supported beams loaded with two point loads at third span. The deflection values were calculated using the moment-area method based on the results obtained from moment-curvature analysis and the corresponding β values were then obtained.

The database was compiled in a patterned format. Each pattern consists of an input vector containing the main six input parameters, and an output vector containing the corresponding β value for this input vector. The input parameters are the cross-section height to width ratio h/b, beam slenderness ratio L/h, the ratio between the actual reinforcement ratio and the balanced reinforcement ratio ρ_s/ρ_b, reinforcement modulus of elasticity E_{SMA}, concrete compressive strength $f^{'}_c$, and the ratio of the applied moment to the cracking moment M_a/M_{cr}.

The database comprised more than 1050 patterns (900 training patterns and 150 cross-validation patterns) so that it provides the network with sufficient data to capture the relationships between each input vector and the corresponding output value. An additional 150 patterns within the same range of the training data were used to test the network's ability to predict the output when unfamiliar input data is presented. The database contains sections with reinforcement ratios as low as 0.35%, and modulus of elasticity of reinforcing bars as low as 20 GPa. The load range started from the cracking load and extended till the load causing yielding of the reinforcement. Error! Reference source not found. shows the ranges, mean values and standard deviations of all

input and output variables in the final database. **Fig. 3** shows a graphical representation of the selected range of reinforcement ratios versus the corresponding β values.

SELECTION OF NETWORK ARCHITECTURE

As mentioned earlier, feed-forward back-propagation neural networks have been most suited and commonly used for engineering applications. Therefore, it was used in this study to predict the reduction factor β. Since there are no rules to determine the architecture of a neural network that would result in optimum performance, a trial and error approach was adopted. Many network architectures were tested. Some of these architectures had one, two, or three hidden layers. It should be noted that it is possible to obtain different network architectures that result in satisfactory performance. A network architecture that consists of an input layer, an output layer, and one hidden layers was found to offer best (optimum) performance in the present study, **Fig. 4**.

The input layer consists of six processing units (neurons) that represent the parameters affecting the reduction factor β. The output layer contains one processing unit which represents the reduction factor β. The hidden layer consists of 10 neurons. A sigmoid function was used as the activation (transfer) function for all of the processing units in the hidden and output layers.

The learning algorithm used in this study is the Levenberg-Marquardt algorithm (Demuth and Beal 1998). The main advantage of this learning algorithm is simplifying the learning process and reducing the time required for training. The Levenberg-Marquardt algorithm propagates the error computed at the output layer back to the network. However, this propagation is based on the

Jacobian matrix J that contains the first derivatives of the network errors with respect to weights and biases (Nehdi et al. 2001). One iteration of the Levenberg-Marquardt algorithm is given by **Eq. [5]**.

$$W_{k+1} = W_k - \left[J^T J + \mu I\right]^{-1} J^T e \qquad [5]$$

Where: W_k is a vector of current weights and biases, μ is the learning rate, J is the Jacobian matrix, J^T is the transpose of the Jacobian matrix, I is the identity matrix, and e is a vector of the network errors.

NETWORK TRAINING AND VALIDATION

Training a FFBP neural network is basically teaching it the embedded relationships between the inputs and the corresponding outputs. However, the learning process is usually complex. It depends on several undefined parameters. During the training process, the FFBP neural network searches for the optimum connections' weights (strengths) between the processing units to predict accurate values of the outputs.

The training process can be performed in either a supervised or unsupervised manner. In a supervised training, the network is provided with patterns of data which contain the input data and the corresponding output values. Thus, the network is told what to learn. In an unsupervised training, there are no target outputs. The network is provided only with the input data. Therefore, the network must modify its weights and biases based on the input data only by categorizing the

input patterns into a finite number of classes. Once the network is successfully trained using either a supervised or unsupervised manner, it should not only successfully predict the output of the training data, but it should also be able to accurately predict the output of unfamiliar sets of input data located within the range of the training data (Nehdi et al. 2001).

The training process of a FFPB neural network is usually done in two stages: (i) Feed–forward, and (ii) Back–propagation. During the feed-forward stage, the data flow from the input processing units to predict the desired network output, β. Then, the obtained output is compared to a predefined output. If the difference is greater than a predefined tolerance, the error is propagated backward from the output layer to modify the network connections weights (back-propagation stage). The back propagation of the error is performed in the present model based on the Levenberg-Marquardt algorithm as explained earlier. **Fig. 5** shows a flow chart of the training process.

The training process of the selected network to predict the reduction factor β is an iterative process, which includes feeding the selected network with data as pairs (input/output). The input data includes (h/b, L/h, ρ_s/ρ_b, E, f'_c, M_a/M_{cr}), while the output includes the target value of β. Based on these pairs of data, the network modifies its weights based on the Levenberg-Marquardt algorithm. The network receives the first input vector, carries out the appropriate computations and activation through the processing units in the hidden layers, and produces an output value for β. The network compares its output to the corresponding target provided in the training pair (the predefined output value for this input vector). The difference between the network output and target value is then calculated and stored. After the first training pattern is completed, the network is provided with a second training pair, and so on until the network goes through all data

available for training. This completes the first epoch. After each epoch, the mean squared of all errors is calculated and stored. Then, the network back-propagates the error using the assigned learning algorithm to adjust the weights and biases for all processing units in the network (Nehdi et al. 2001).

The training process continues till either the network converges, the mean square error (MSE) of the cross-validation data set increases, or the maximum number of epochs provided for early stopping is reached. Maximum generalization of the network is achieved when the MSE of the cross-validation data set is minimum. Training the network beyond this limit will result in overtraining (i.e. memorization of the training data). While this reduces the MSE of the training data set, the ability of the network to correctly predict the output of unfamiliar data generally decreases. Figure 6 **Fig. 6**

shows a general representation for the change in the MSE value of the training and cross-validations data sets during the training process.

RESULTS AND DISCUSSION

The acceptance/rejection of an ANN model for predicting the reduction factor β for the deflection calculations of SMA RC members depends on its ability to generalize its predictions to new input patterns not previously used in network training. Good generalization mainly depends on the final set up of weights and biases, and the success of the training process. In other words, before testing the model's ability to generalize, the success of the training process must first be evaluated. In addition, the model response to training patterns should be evaluated.

Performance of ANN using training data

The selected network architecture (**Fig. 4**) was trained to predict the reduction factor β that can be used to predict the deflection of SMA RC members. As there was no clear trend in neural predictions for either over- or underestimating the reduction factor, the reliability of its predictions was evaluated using the average absolute error (AAE) given by **Eq. [6]**.

$$AAE = \frac{1}{n}\sum_{i=1}^{n}\frac{|\beta_{\text{Analytical}} - \beta_{\text{Predicted}}|}{\beta_{\text{Analytical}}} \qquad [6]$$

Where: n is the number of the data patterns.

After completing the training process, the performance of the network was first checked using the input patterns used in the training process. The response is plotted in **Fig. 7-(a)** and **Fig**. 7-**(b)**. The model predicted values were plotted versus the values obtained from the parametric study. As shown in the figures, the network has successfully learned the relationships between the input and corresponding output values. The network performance was satisfactory and an AAE of 3.47% was calculated for the training data, while an AEE of 3.44% was obtained for the cross-validation data.

Performance of ANN using testing data

As explained earlier, the acceptance/rejection of a trained neural network is determined by its ability to generalize its predictions beyond the training data. The network model should be able to successfully predict the reduction factor β when presented with new input data that is unfamiliar to the network (not used in training). The model was presented with an input vector consisting of 150 input patterns that are new to the network and no knowledge of the exact values of β was provided. **Fig.** 7-**(c)** shows that the model successfully predicted the β values corresponding to the testing data with an AAE of 4.50%, indicating that the model predictions are appropriate.

SENSITIVITY ANALYSIS

The ANN model thus developed showed superior performance and demonstrated its ability to predict the reduction factor β. This section examines the ability of this model to capture the effects of the individual input parameters on the desired output value β. The strategy adopted to examine the sensitivity of the developed model to the input parameters consists of randomly

selecting one database pattern from the training data, and subsequently creating other database patterns by changing the parameter of interest, while keeping all other input parameters unchanged. The levels of the parameters of interest in the created database records are not similar to those of the training data to avoid the possibility that the ANN had memorized the output of such data.

The sensitivity of the proposed model to *h/b*, *L/h*, and f_c' values was evaluated. Two input records that have a reinforcement ratio of 0.40% and 0.62% were used in the analysis. Values of 0.25% and 0.35% were assigned to the ρ_s/ρ_b input parameter, while a value of 40 GPa was assigned to the *E* input parameter. The load level was kept constant with M_a/M_{cr} = 1.20. The *h/b* input parameter was varied from 0.85 to 2.60 with a step of 0.05. A step of 0.1 was used to increase the *L/h* input parameter from 8.00 to 15.00. The f_c' input parameter was varied within the normal strength concrete range, 20 MPa to 55 MPa. **Figures 8-(a) to 8-(c)** indicate a minor sensitivity of the proposed model to the *h/b*, *L/h*, and f_c' input parameters, respectively.

The effect of the ρ_s/ρ_b and *E* input parameters on the predicted value of β was studied by constructing two separate databases which have constant input parameters while varying the ρ_s/ρ_b and *E* input parameters, respectively. The *h/b*, *L/h*, f_c', and M_a/M_{cr} input parameters were chosen as 2.00, 10.00, 40 MPa, and 1.20, respectively. The ρ_s/ρ_b input parameter was varied between 0.18% and 1.70%. A step of 0.05 was used. Values between 20 GPa and 65 GPa were assigned to the *E* input parameter. As shown in **Fig. 8-(d)** and **Fig**. 8-**(e)**, a significant increase in the β value was observed with the increase in the ρ_s/ρ_b and *E* input parameters.

PROPOSED MODEL

From the sensitivity analysis carried out in the previous section, it is clear that the main factors affecting the reduction factor β are the section reinforcement ratio, and the reinforcement modulus of elasticity. Thus, a new database was developed with constant parameters b/h, L/h, f'_c, ρ_s/ρ_b varying between 0.17 and 1.0, and modulus of elasticity values of 20 GPa, 25 GPa, 30 GPa, 35 GPa, 40 GPa, and 50 GPa. The response of the model was summarized in **Fig. 9**. Knowing the section reinforcement ratio and the reinforcement modulus of elasticity values, the value of the reduction factor β can be obtained from this figure, which can offer a simplified and useful design tool.

DESIGN EXAMPLE

A design example has been worked out for a 7.0 m span simply supported beam. The beam is reinforced with SMA that has a yield stress of 401 MPa. Normal-strength concrete with a compressive strength of 35 MPa is used. As shown in **Fig. 10**, the beam has a cross-section height and width of 700 mm, and 300 mm, respectively. In addition to its own weight, the beam supports two point loads at third span. Service moment levels of 1.2 M_{cr}, 1.5 M_{cr}, 2.0 M_{cr}, and the moment causing a stress of 0.6 $f_{y\text{-}SMA}$ in the SMA were used.

As explained above, the cross-section reinforcement ratio and the reinforcement modulus of elasticity greatly affect the deflection of SMA RC members. Thus, in this example, three different reinforcement ratios, namely 0.35%, 0.70%, and 0.90%, were selected. For each reinforcement ratio, modulus of elasticity values of 20 GPa, 25 GPa, 30 GPa, 35 GPa, 40 GPa, and 50 GPa were

assigned. The deflection analysis was performed first using the moment-area method. The deflection analysis was subsequently performed using Branson (1963), ACI 440.1R-03 (2003), ACI 440.1R-04 (2004), and the proposed β values. The deflection values obtained from the moment-area method were plotted versus the different models predictions as shown in **Fig. 11**. Superior performance for the ANN model can be observed when compared to that of the other models.

CONCLUSIONS

This paper presents a non-traditional approach for predicting the deflection of SMA RC members. An artificial neural network model was developed to predict the reduction factor β to be used in the calculation of the effective moment of inertia I_e. First, a database was developed using the results obtained from the parametric study conducted by Elbahy et al. (2008b). The database includes the main factors affecting the effective moment of inertia, I_e: cross-section height to width ratio, h/b, beam slenderness ratio, L/h, the ratio between the used reinforcement ratio to the balanced reinforcement ratio, ρ_s/ρ_b, reinforcement modulus of elasticity E, concrete compressive strength, $f_c^{'}$, and the applied load level as a ratio between the applied moment and cracking moment, M_a/M_{cr}. The network architecture that results in the optimum performance was subsequently selected based on a trial and error approach.

The trained ANN model showed superior performance not only in predicting the training data output, but also in predicting the output for testing data unfamiliar to the model. A sensitivity analysis was carried out to study the effects of the individual input parameters on the predicted β

values. While h/b, L/h, and $f_c^{'}$ did not have a significant effect on the predicted β values, ρ_s/ρ_b and E were found to greatly affect β.

A new database was developed using a range of ρ_s/ρ_b and different E values and their effect on β were explored in detail. The results thus obtained were presented in a chart format that can be easily used by designers to estimate the deflection of SMA RC members. Knowing the cross-section reinforcement ratio and the reinforcement modulus of elasticity, the reduction factor β can be obtained from the chart. To illustrate the feasibility of using the developed chart to predict the deflection of SMA RC members, a deign example was discussed. The developed chart showed superior performance to that of existing design tools. It is argued that the developed chart can be used to predict β for RC structures with ρ_s/ρ_b of 0.2 to 1.0 given that the material of the reinforcing bars has a modulus of elasticity ranging between 20 GPa and 50 GPa.

REFERENCES

ACI 318-05 2005. Building code requirements for structural concrete and commentary. American Concrete Institute. Farmington Hills, Michigan.

ACI 440.1R-03 2003. Guide for the design and construction of concrete reinforced with FRP bars. American Concrete Institute, Farmington Hills, Michigan, 42 p.

ACI 440.1R-04 2004. Guide for the design and construction of concrete reinforced with FRP bars, proposed revision. American Concrete Institute, Farmington Hills, Michigan, 35 p.

Ackley, D.H., Hinton, G.E., and Sejnowski, T.J. 1985. A learning algorithm for boltzmann machines. Cognitive Science, 9(1): 147-169.

Alam, M.S., Youssef, M.A., and Nehdi, M. 2007. Utilizing shape memory alloys to enhance the performance and safety of civil infrastructure: a review. Canadian Journal of Civil Engineering, 34(9): 1075-1086.

Bischoff, P.H. 2005. Reevaluation of deflection prediction for concrete beams reinforced with steel and fiber reinforced polymer bars. ASCE Journal of Structural Engineering, 131(5): 752-762.

Bischoff, P.H., and Scanlon, A. 2007. Effective moment of inertia for calculating deflections of concrete members containing steel reinforcement and fiber-reinforced polymer reinforcement. ACI Structural Journal, 104(1): 2007, 68-75.

Branson, D.E. 1963. Instantaneous and time-dependent deflections of simple and continuous reinforced beams. HPR Rep. No. 7, Part 1, Alabama Highway Department, Bureau of Public Roads, Montgomery, 78 p.

CEB-FIP 1990. Model Code for Concrete Structures. Comité Euro-International du Béton, Thomas Telford, London, UK.

Charniak, E., and McDermott D. 1985. Introduction to artificial intelligence. Addisoon-Wesley Publishing Company, Reading Massachusetts, USA, 701 p.

CSA A23.3-04 2004. Design of concrete structures. Canadian Standards Association, Rexdale, Ontario, Canada, 240 p.

Demuth, H., and Beal, M. 1998. Neural network tool box for use with MATLAB version 3. The Math Works Inc., 5.20-5.58.

Dolce, M., and Cardone, D. 2001. Mechanical behaviour of shape memory alloys for seismic application 1: martensite and austenite NiTi bars subjected to torsion. International Journal of Mechanical Sciences, 43, 2631–56.

Eggeler, G., Hornbogen, E., Yawny, A., Heckmann, A. and Wagner, M. 2004. Structural and functional fatigue of NiTi shape memory alloys. Materials Science and Engineering, A 378 (1-2): 24-33.

Elbahy, Y.I., Youssef, M.A., and Nehdi, M. 2008a. Flexural behaviour of concrete members reinforced with shape memory alloys. 2nd Canadian Conference on Effective Design of Structures, McMaster University, Hamilton, Ontario, Canada, 477-486.

Elbahy, Y.I., Youssef, M.A., and Nehdi, M. 2008b. Deflection of shape memory alloy reinforced concrete beams: assessment of existing models. Submitted to the Canadian Journal of Civil Engineering.

El-Chabib, H., and Nehdi, M. 2006. Effect of mixture design parameters on segregation of self-consolidating Concrete. ACI Materials Journal, 103(5): 374-383.

El-Chabib, H., and Nehdi, M. 2005. Neural network modeling of properties of cement-based materials demystified. Advances in Cement Research, 17(3): 91-201.

El-Chabib, H., Nehdi, M., and Sonebi, M. 2003. Artificial intelligence model for flowable concrete mixtures used in underwater construction and repair. ACI Materials Journal, 100(2): 165-173.

Grandhi, F., and Wolons, D. 1999. Characterization of the pseudoelastic damping behavior of shape memory alloy wires using complex modulus. Smart Materials and Structures, 8: 49–56.

Haykin, S. 1994. Neural networks: a comprehensive foundation. Macmillan, New York, 842 p.

Hopfield, J.J. 1982. Neural networks and physical systems with emergent collective computational abilities. Proceedings, National Academy of Science, USA, 79(8): 2554-2558.

Hornbogen, E. 2004. Review: thermo-mechanical fatigue of shape memory alloys. Journal of Material Science, 39(2): 385-399.

ISIS Canada 2001. Reinforcing concrete structures with fibre reinforced polymers, Design Manual 3, Winnipeg, Manitoba.

Janke, L., Czaderski, C., Motavalli, M., and Ruth, J. 2005. Applications of shape memory alloys in civil engineering structures - overview, limits and new ideas. Materials and Structures, 338(279): 578-592.

Kohonen, T. 1982. Self-organized formation of topologically correct feature maps. Biological Cybernetics, 43: 59-69.

Nehdi, M., El-Chabib, H., and El-Naggar, M. 2001. Predicting the performance of self-compacting concrete mixtures using artificial neural networks. ACI Materials Journal, 98: 394-401.

Piedboeuf, M.C., and Gauvin, R. 1998. Damping behaviour of shape memory alloys: strain amplitude, frequency and temperature effects. Journal of Sound and Vibration, 214(5): 885–901.

Rosenblatt, F. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. Psychological Review, 65(6): 386-408.

Rumelhart, D.E., Hinton, G.E., and William, R.J. 1986. Learning internal representation by error propagation. Parallel Distributed Processing, Foundation, MIT Press, Cambridge, Mass, 1: 318-362.

Saiidi, M.S., Sadrossadat-Zadeh, M., Ayoub, C., and Itani, A. 2007. Pilot study of behavior of concrete beams reinforced with shape memory alloys. ASCE, Journal of Materials in Civil Engineering, 19(6): 454-461.

Smithers, T., Conkie, A., Doheny, J., Logan, B., Millington, K., and Tang, M. X. 1990. Design as intelligent behavior: an AI in design research. Journal of Artificial Intelligence in Engineering, 5(2): 78-109.

LIST OF SYMBOLS

AAE	Average absolute error.
AGE	Average algebraic error.
AI	Artificial intelligence.
ANN	Artificial Neural Network.
b	Cross-section width.
E	Modulus of elasticity of the reinforcement.
e	Vector of the network errors.
E_{FRP}	Modulus of elasticity of FRP.
E_s	Modulus of elasticity of steel.
f	Neuron nonlinear activation function.
f_c'	Concrete compressive strength.
FFBP	Feed-forward back-propagation.
$f_{y\text{-}SMA}$	SMA yielding stress.
h	Cross-section height.
I_{cr}	Moment of inertia for the cracked section.
I_e	Effective moment of inertia.
I_g	Moment of inertia for the un-cracked section.
J	Jacobian matrix.
J_T	Transpose of the Jacobian matrix.
L	Beam span.
M_a	Applied moment.
M_{cr}	Cracking moment.
MLP	Multi-layer perceptron.
MSE	Mean square error.
n	Number of neurons in layer (l-1).
U_j^l	Neuron net input.
W_{ji}^l	Connection weight (strength) that connects the neuron j in layer l to neuron i in layer (l-1).
W_k	Vector of current weights and biases.
X_i^{l-1}	Input coming from neuron i in layer l-1 to neuron j in layer l.
Y_j^l	Neuron output value.
α	Bond dependent factor.
β	Reduction factor for the calculation of the effective moment of inertia.
θ_j^l	Threshold value assigned to neuron j in layer l.
μ	Learning rate
ρ	Reinforcement ratio.
ρ_b	The reinforcement ratio of the balanced section.
ρ_{FRP}	The reinforcement ratio of the FRP RC section.

LIST OF TABLES

Table 1: Range, average and standard deviation of measured input and output variables

Variable	Training data			Testing data		
	Range	Average	Standard deviation	Range	Average	Standard deviation
h/b	0.83 to 2.67	1.97	0.27	0.83 to 2.67	1.98	0.28
L/h	7.50 to 15.00	10.01	0.97	7.50 to 15.00	9.96	0.97
ρ_s/ρ_b	0.17 to 1.74	0.62	0.34	0.17 to 1.74	0.62	0.35
E_{SMA} (GPa)	20 to 65	42200	13052	20 to 65	41454	13168
f'_c (MPa)	20 to 55	40.36	4.79	20 to 55	40.45	4.40
M_a/M_{cr}	1.00 to 7.42	2.84	1.36	1.04 to 6.49	2.75	1.22
β	0.01 to 2.13	0.52	0.47	0.01 to 2.13	0.52	0.42

LIST OF FIGURE CAPTIONS

Fig. 1: Simplified model of artificial neuron

Fig. 2: General multi-layer FFBP neural network

Fig. 3: Graphical representation of the selected reinforcement ratios and the corresponding β values

Fig. 4: Schematic diagram for the selected network architecture

Fig. 5: Flow chart of the training process

Fig. 6: General representation for the change in th MSE value of the training and cross-validation data sets during the traning process.

Fig. 7: ANN model response in predicting the training, cross-validation, and testing data output

Fig. 8: Sensitivity of the ANN model

Fig. 9: Developed chart for predicting the reduction factor β

Fig. 10: Elevation and cross-section details of the beams used in the design example

Fig. 11: Design example: moment-area method versus different models deflections

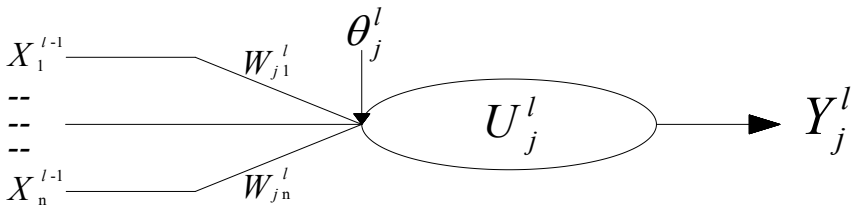


Fig. 1

Fig. 2 Elbahy, Nehdi, and Youssef

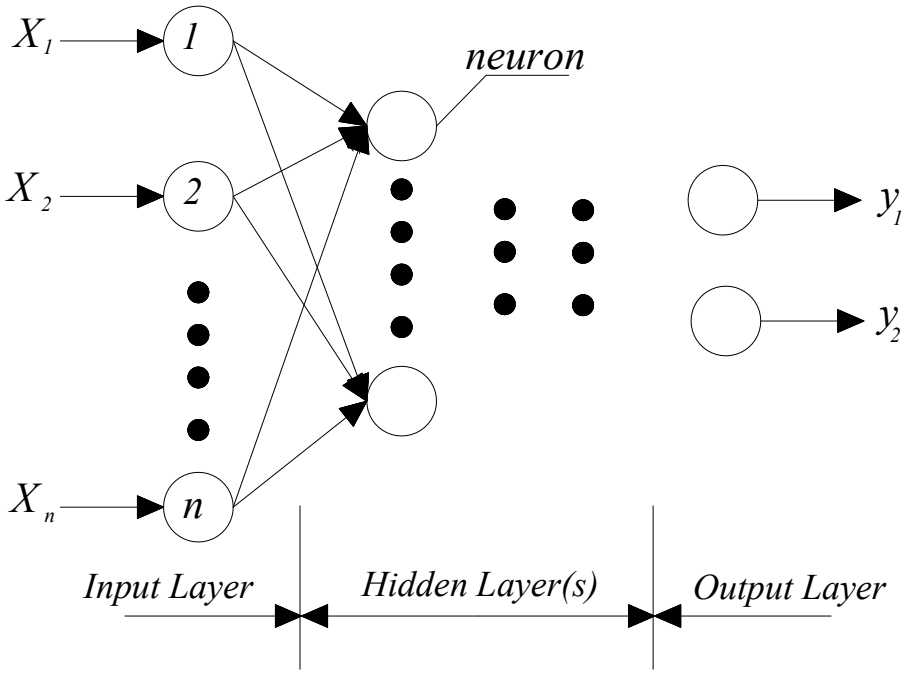


Fig. 2

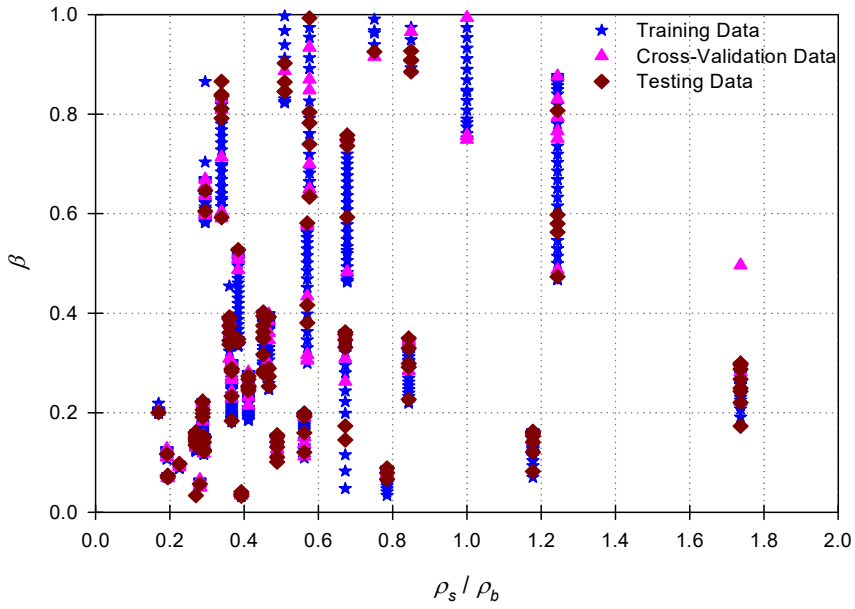

Training Data
Cross-Validation Data
Testing Data
β
ρs / ρb
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1.8
2.0

Fig. 3

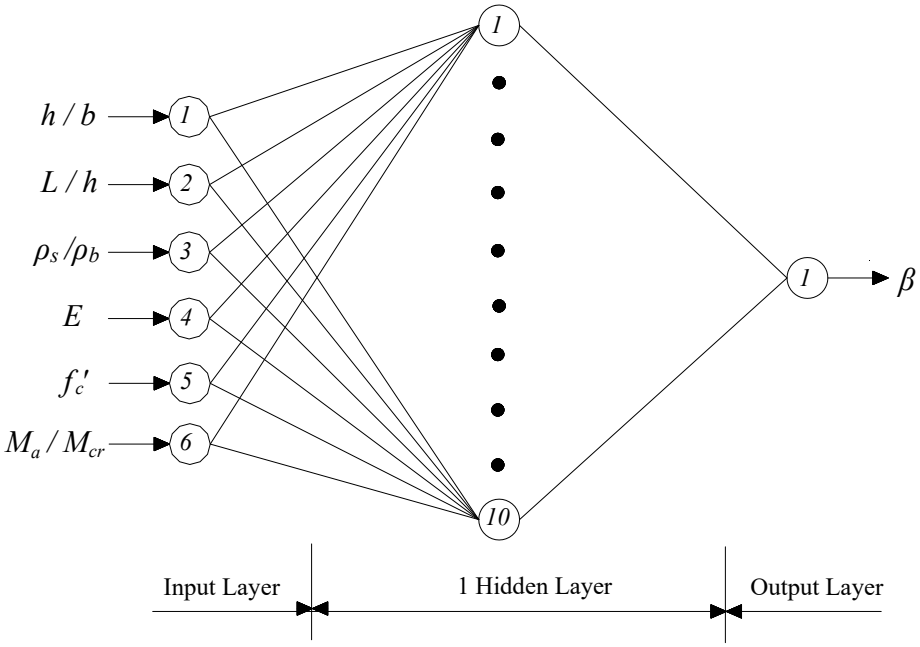

h / b
L / h
ρs /ρb
E
f'c
Ma / Mcr
1
2
3
4
5
6
1
10
1
β
Input Layer
1 Hidden Layer
Output Layer

Fig. 4

Fig. 5 Elbahy, Nehdi, and Youssef

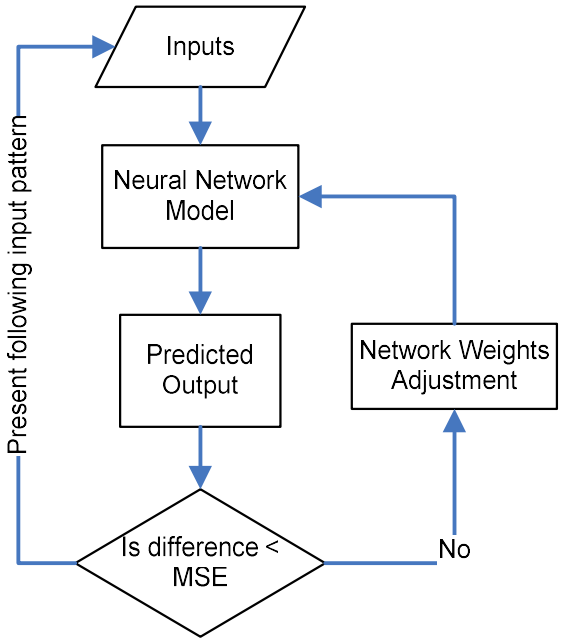


Fig. 5

Fig. 6 Elbahy, Nehdi, and Youssef

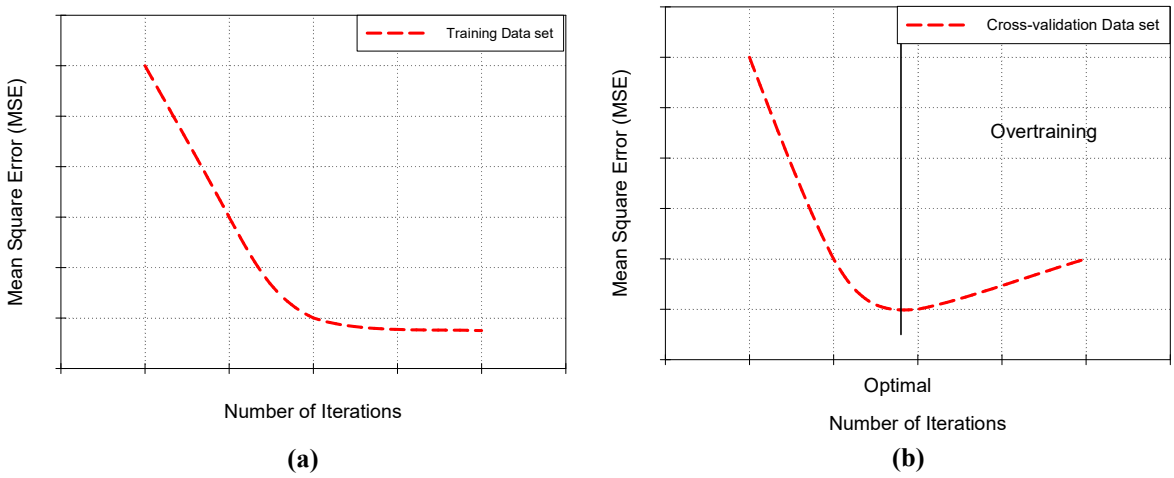
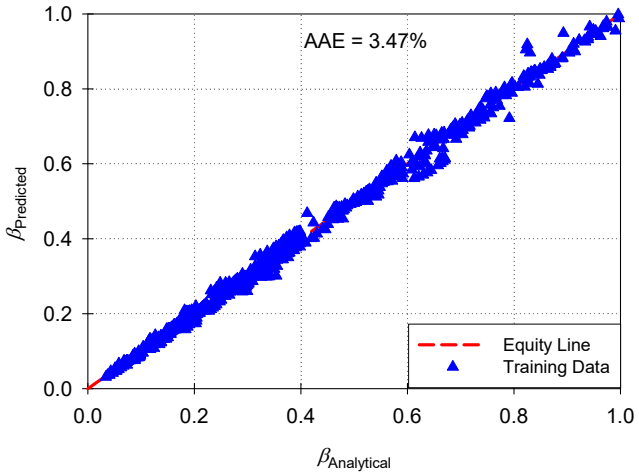
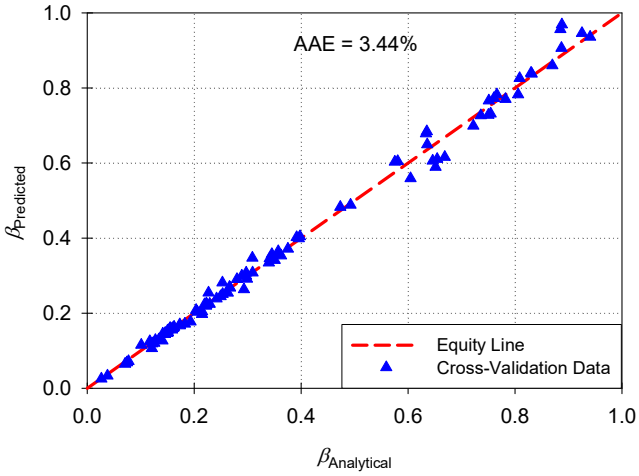


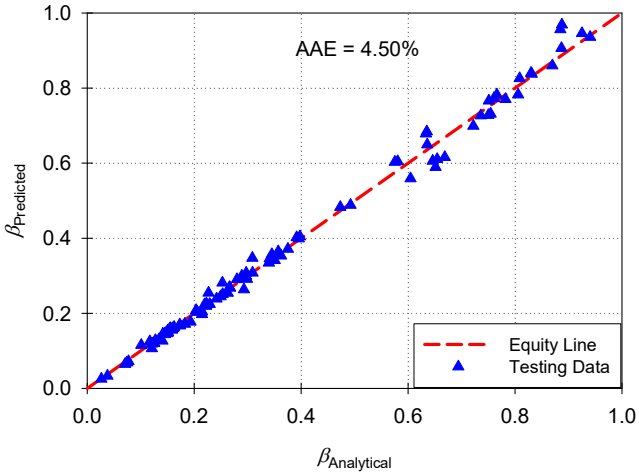
Fig. 6


AAE = 3.47%
Equity Line
Training Data
$\beta_{Predicted}$
$\beta_{Analytical}$
0.0
0.2
0.4
0.6
0.8
1.0

(a)


AAE = 3.44%
Equity Line
Cross-Validation Data
$\beta_{Predicted}$
$\beta_{Analytical}$
0.0
0.2
0.4
0.6
0.8
1.0

(b)


AAE = 4.50%
Equity Line
Testing Data
$\beta_{Predicted}$
$\beta_{Analytical}$
0.0
0.2
0.4
0.6
0.8
1.0

(c)

Fig. 7

Fig. 8 Elbahy, Nehdi, and Youssef

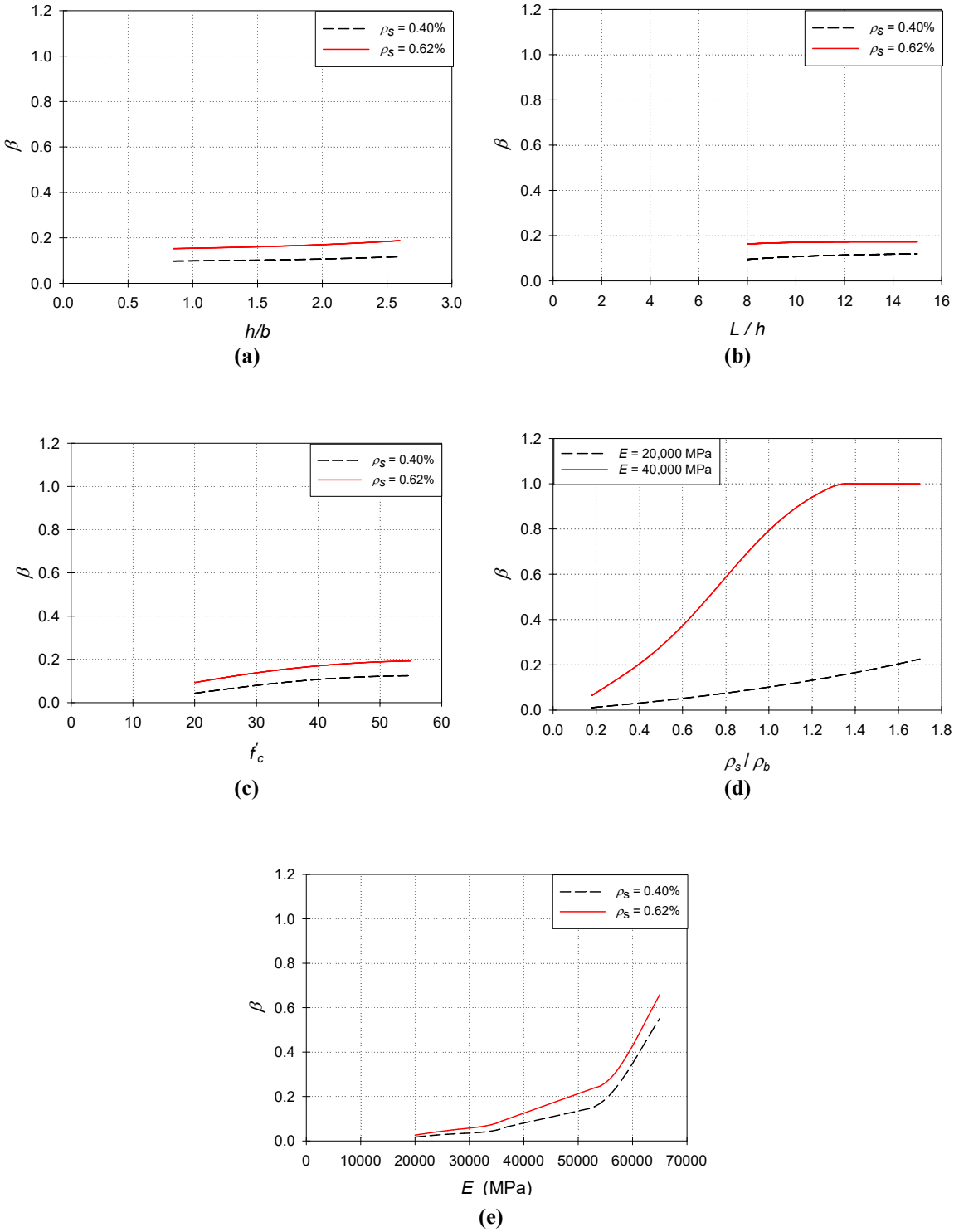


Fig. 8

Fig. 9 Elbahy, Nehdi, and Youssef

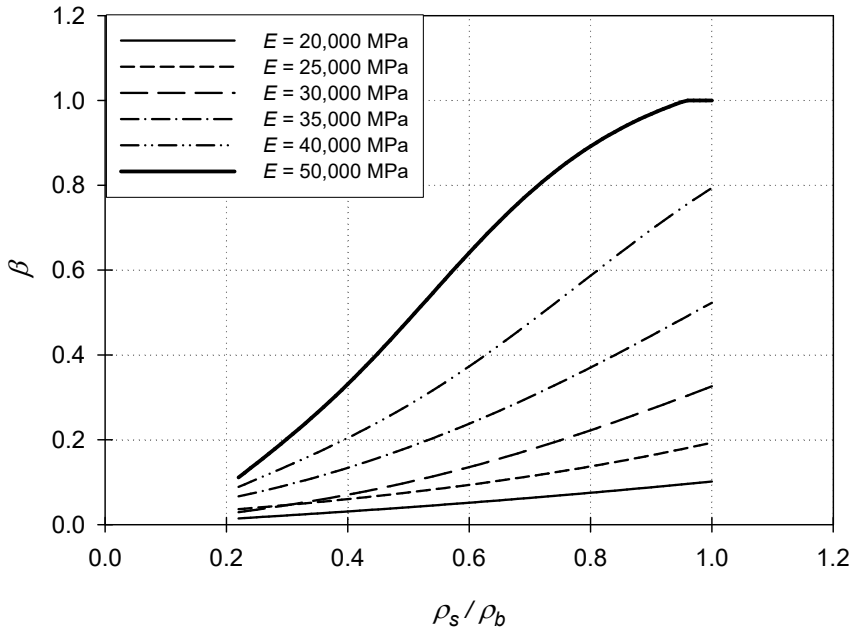
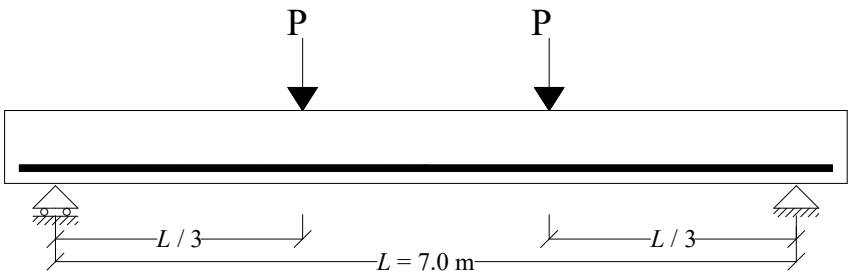
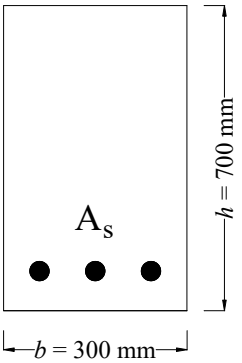


Fig. 9

Fig. 10 Elbahy, Nehdi, and Youssef

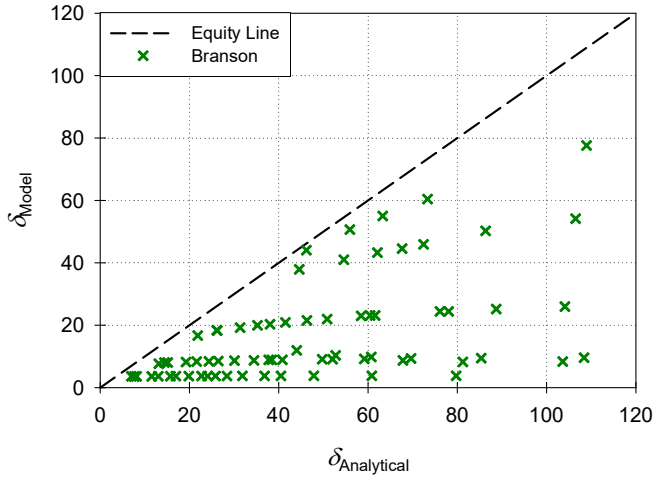


Elevation

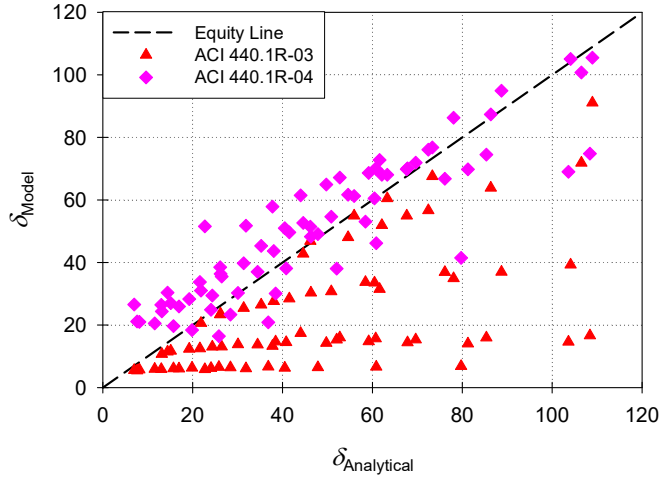


Cross-section

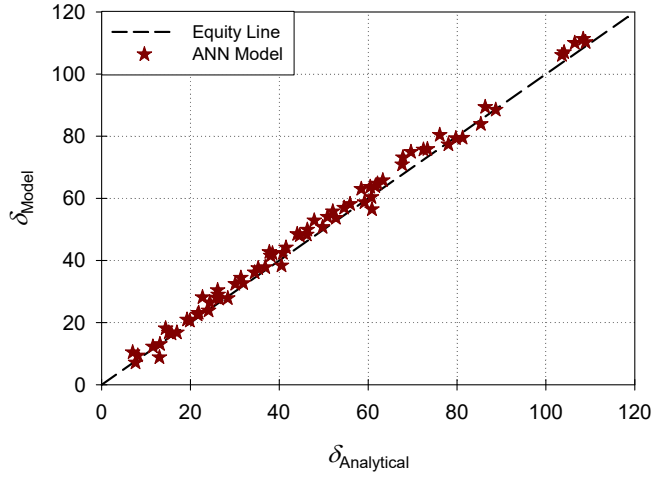
Fig. 10


Equity Line
Branson
δ_{Model}
$\delta_{Analytical}$
0
20
40
60
80
100
120

(a)


Equity Line
ACI 440.1R-03
ACI 440.1R-04
δ_{Model}
$\delta_{Analytical}$
0
20
40
60
80
100
120

(b)


Equity Line
ANN Model
δ_{Model}
$\delta_{Analytical}$
0
20
40
60
80
100
120

(c)

Fig. 11