Karaelmas Fen ve Mühendislik Dergisi
Journal home page: http://fbd.beun.edu.tr

Research Article

Estimating Stiffness Modification Factor for the Coupling Beam of Coupled Shear Walls Using a Neural Network Model

Boşluklu Perdelerde Bağ Kirişi Rijitlik Düzeltme Çarpanlarının Yapay Sinir Ağı ile Tahmini

Bulent Akbas[1], Bilge Doran[2*], Sema Alacali[3], Bora Akşar[1]

[1]Gebze Technical University, Department of Earthquake and Structural Engineering, Gebze, Kocaeli, Turkey
[2]Yildiz Technical University, Faculty of Civil Engineering, Department of Civil Engineering, Esenler, Istanbul, Turkey
[3]Yildiz Technical University, Faculty of Civil Engineering, Department of Civil Engineering, Esenler, Istanbul, Turkey

Abstract

Neural Network (NN) have been widely used in modeling engineering problems, for the last two decades as a result of the developments in computer and software technology. The results of NN models, if constructed properly, generally yield a more realistic and accurate predictions. This study is intended to estimate the stiffness modification factor for the coupling beam of coupled shear walls using a neural network model. A multi-layered feed-forward NN model trained with the back-propagation algorithm is developed to model the non-linear relationship between the geometrical and mechanical properties of a coupled shear walls and its stiffness modification factor. The approach adapted in this study was shown to be capable of providing the best accurate estimates of the stiffness modification factor by using the three design parameters.

Keywords: Back propagation algorithm, Coupled shear wall, Equivalent frame, Neural networks

Öz

Son 20 yılda bilgisayar teknolojisi ve bilgisayar yazılımlarındaki gelişmeler nedeniyle, Yapay Sinir Ağları (YSA) mühendislik problemlerinin modellenmesinde sıkça kullanılmaktadır. Doğru bir şekilde kurulması durumunda YSA model sonuçları genellikle daha gerçekçi ve doğruya yakın tahminler vermektedir. Bu çalışma ile boşluklu perdelerin bağ kirişleri için tarif edilmiş olan rijitlik düzeltme çarpanlarının YSA ile tahmini hedeflenmiştir. Boşluklu perdeli sistemlerde geometrik ve mekanik özellikler arasındaki doğrusal olmayan ilişki ile rijitlik düzeltme çarpanlarını modellemek üzere, geri yayılım algoritması ile eğitilen çok katmanlı, ileri beslemeli bir YSA modeli geliştirilmiştir. Çalışmada önerilen yaklaşım ile rijitlik düzeltme çarpanlarının üç tasarım parametresi kullanılarak en iyi şekilde tahmin edilebileceği görülmüştür.

Anahtar Kelimeler: Geri yayılım algoritması, Boşluklu perde, Eşdeğer çerçeve, Yapay sinir ağları

1. Introduction

Neural Network (NN) is a powerful black-box modeling technique which produces output values from a given input set. NNs have been widely used in modeling engineering problems having nonlinear relationships such as concrete strength, cost estimation, structural damage detection, etc., for the last two decades as a result of the developments in computer and software technology (Flood 1989, Cladera and Mari 2004, Saadata et al. 2004, Tehranizadeh and Safi 2004, Williams and Hoit 2004, Amini and Tavassoli 2005, Papadrakakis et al. 2005, Yeung and Smith 2005, Lee et al. 2005, Ashour and Alqedra 2005, Alacali et al. 2006). The results of NN models, if constructed properly, generally yield a more realistic and accurate predictions. NN models are constructed based on simulating the structure and learning activities of the human brain. Garrett et al. (1997) defines a NN model as "a computational mechanism able to acquire, represent, and compute mapping from one multivariate space of information to another, given a set of data representing that mapping".

This study is intended to estimate the stiffness modification factor for the coupling beam of coupled shear walls using a neural network model. In mid and high rise R/C buildings, shear wall and coupled shear wall (CSW) systems are usually

*Corresponding Author: doran@ yildiz.edu.tr

Received / Geliş tarihi : 10.02.2016
Accepted / Kabul tarihi : 28.07.2016

used effectively to provide the required stiffness, strength and ductility (Figure 1). Since lateral loads cause bending with high shear stresses in coupling beams, the structural behavior of a CSW is greatly affected by the behavior of the coupling beams which depends on its geometrical and mechanical properties (Kinh and Tan 1999, Doran 2004).

In modeling of CSW systems, shell or membrane elements can be used. Determining the membrane resultant forces by employing 2D shell elements is somewhat cumbersome. Instead, a simple equivalent 1D frame model can be used for the modeling purposes.

Several modelling techniques are suggested for the evaluation of the elastic behavior of the lateral load resisting structures with CSWs (Syngellakis and Chan 1992, Pala and Ozmen 1995, Kinh and Tan 1999, Doran 2004). In an equivalent 1D frame model, the stiffness of a coupling beam can be defined as the product of the equivalent stiffness (combined bending and shear) of an equivalent frame and the stiffness modification factor. Using a NN model, the engineer can predict the stiffness modification factor of coupling beams, even when the data is insufficient. An optimal solution during the design may also be obtained through the NN model.

A multi-layered feed-forward NN model trained with the back-propagation algorithm is developed to model the non-linear relationship between the geometrical and mechanical properties of a CSW and its stiffness modification factor. For this purpose, first, finite element analyses (FEAs) of CSW systems using 2D shell element and equivalent 1D frame concepts are carried out to evaluate the stiffness modification factor of CSW systems. Second, a NN model is trained and tested to predict the stiffness modification factor and the results are compared with the finite element analyses results.

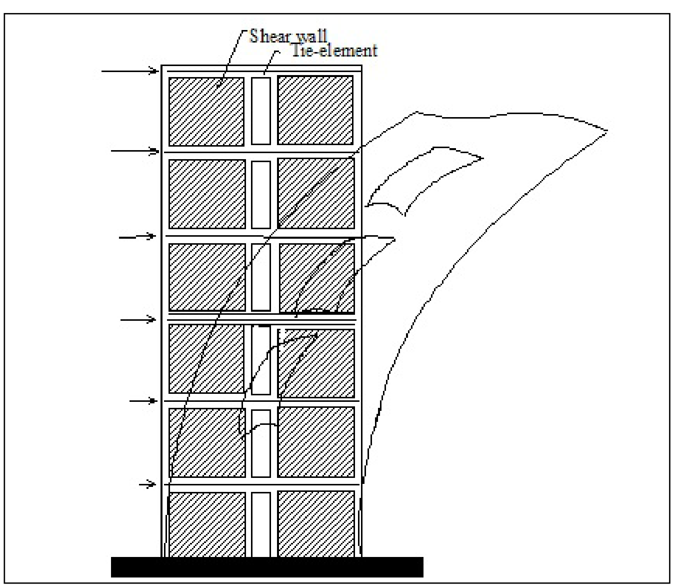


Figure 1. A typical CSW.

2. Estimating the Stiffness Modification Factor of a CSW System

The actual stiffness of the coupling beams is defined as (Figure 2):

$$m_{i\theta_i} = \frac{M_i}{\theta_i} \tag{1}$$

where M_i and θ_i are the bending moment and the rotation for section (i), respectively (Figure 2). On the other hand, equivalent stiffness (combined bending and shear) of an equivalent frame (Figure 2b) can be written as (Pala and Ozmen 1995):

$$\bar{m}_{i\theta_i} = \frac{6EI}{L}\frac{L^2}{(L^2+3.9d^2)} \tag{2}$$

where EI is the bending stiffness. The actual stiffness of the coupling beams (ij member) in Equation (1) can also be defined using the equivalent stiffness of an equivalent frame as (Figure 2b):

$$m_{i\theta_i} = \eta \bar{m}_{i\theta_i} \tag{3}$$

where η is called as stiffness modification factor.

The stiffness of the coupling beams of a CSW is greatly influenced by the openings. A simple function for η can be used as (Doran 2004):

$$\eta = a_0\left(\frac{h}{l}\right)^{a1}\left(\frac{b}{l}\right)^{a2}\left(\frac{d}{l}\right)^{a3} \tag{4}$$

where h is the storey height, b is the length of the wall, d is the gross height of the beam, l is the length of coupling beam. Besides, the ratios of h/l, b/l, d/l describe the openings between the two shear walls and a_0, a_1, a_2, a_3 are constants. The actual stiffness of coupling beams in Equation (1) can be obtained through detailed FEAs using 2D shell elements. Table 1 presents the results of the analysis of 54 CSW systems having geometrical parameters of h = 3m, L = 6m, b = 3.0, 3.2, 3.4, 3.6, 3.8, 4.0, 4.2, 4.4, 4.6 m and d = 0.20, 0.40, 0.60, 0.80, 1.00, 1.20*m* (Figure 2a) by SAP-2000 (Doran 2004).

The results are designated as η^{FEA} in Table 1. Through a nonlinear regression analysis (Wadsworth, H.M, 1997), Equation (4) takes the form of (correlation coefficient=0.949):

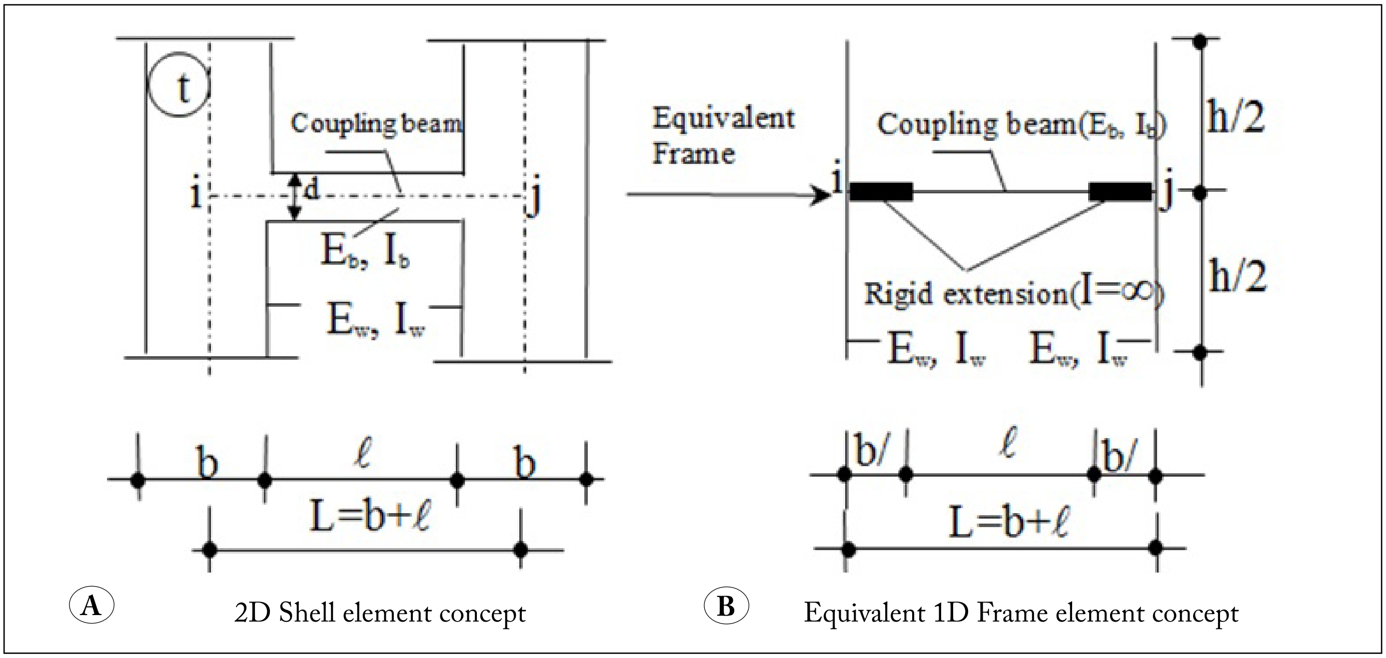


Figure 2A, B. CSW and equivalent frame.

$$\eta = 1.9210\left(\frac{h}{l}\right)^{0.0282}\left(\frac{b}{l}\right)^{1.6824}\left(\frac{d}{l}\right)^{-0.5860} \quad (5)$$

3. Neural Network Design

Neural networks (NNs) are models for generalization and used in establishing a reliable relationship among the various parameters in engineering problems. They can handle highly non-linear problems easily. NN models do not require complex mathematical formulations and gather knowledge by learning from examples. A NN model consists of three simple components: transfer function, network architecture and learning law. Some of the major advantages of a NN are (Rafiq et al. 2001): (1) NNs learn and generalize from examples and experience to produce meaningful solutions to the problems even in cases where the input data contains error or is incomplete; (2) NNs are able to adapt solutions over time and to compensate for changing circumstances; (3) NNs can evaluate theoretical, experimental, or empirical data based on good and reliable past experience or a combination of these.

A typical three-layer feed-forward NN with n input nodes, m hidden nodes and one output node is shown in Figure 3.

The input nodes represent the data presented to the NN, whereas the output nodes produce the NN output. The main function of the hidden layer (Figure 3) is to serve as the interface to extract and to remember the useful features and the sub features from the input patterns to predict the outcome of the network (Rafiq et al. 2001, Gunaydin and Dogan 2004).

In a typical NN, a group of processing elements (PEs) (called neurons) is linked together in an attempt to construct a relation in an input/output set of learning patterns. A PE is an information-processing unit with three basic components: (1) a set of synapses; (2) an adder; (3) an activation function (Haykin 1994).

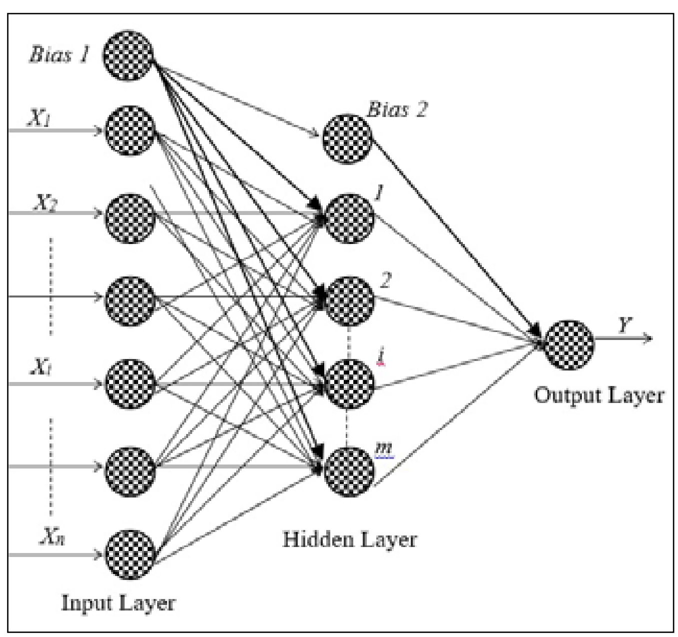


Figure 3. A typical neural network model.

Table 1. The stiffness modification factors.

(1)	(2)	(3)	(4)	(5)	(6)	(1)	(2)	(3)	(4)	(5)	(6)
d (m)	b (m)	$\frac{h}{l}$	$\frac{b}{l}$	$\frac{d}{l}$	η^{FEA}	d (m)	b (m)	$\frac{h}{l}$	$\frac{b}{l}$	$\frac{d}{l}$	η^{FEA}
	3.00	1	1	0.0667	7.1773		3.00	1	1	0.2667	4.6417
	3.20	1.0714	1.1429	0.0714	8.7477		3.20	1.0714	1.1429	0.2857	5.415
	3.40	1.1538	1.3077	0.0769	10.81		3.40	1.1538	1.3077	0.3077	6.3633
	3.60	1.25	1.5	0.0833	13.57		3.60	1.25	1.5	0.3333	7.5333
0.20	3.80	1.3636	1.7273	0.0909	17.35	0.80	3.80	1.3636	1.7273	0.3636	8.9733
	4.00	1.5	2	0.1	22.6633		4.00	1.5	2	0.4	10.79
	4.20	1.1667	2.3333	0.1111	30.3667		4.20	1.1667	2.3333	0.4444	13.0513
	4.40	1.875	2.75	0.125	41.9767		4.40	1.875	2.75	0.5	15.89
	4.60	2.1429	3.2857	0.1429	60.1833		4.60	2.1429	3.2857	0.5714	19.4433
	3.00	1	1	0.1333	6.0683		3.00	1	1	0.3333	4.2067
	3.20	1.0714	1.1429	0.1429	7.29		3.20	1.0714	1.1429	0.3571	4.8317
	3.40	1.1538	1.3077	0.1538	8.861		3.40	1.1538	1.3077	0.3846	5.581
	3.60	1.25	1.5	0.1667	10.907		3.60	1.25	1.5	0.4167	6.48
0.40	3.80	1.3636	1.7273	0.1818	13.6217	1.00	3.80	1.3636	1.7273	0.4545	7.559
	4.00	1.5	2	0.2	17.29		4.00	1.5	2	0.5	8.855
	4.20	1.1667	2.3333	0.2222	22.3567		4.20	1.1667	2.3333	0.5555	10.4083
	4.40	1.875	2.75	0.25	29.51		4.40	1.875	2.75	0.625	12.2567
	4.60	2.1429	3.2857	0.2857	39.8833		4.60	2.1429	3.2857	0.7143	14.4367
	3.00	1	1	0.2	5.2483		3.00	1	1	0.4	3.9
	3.20	1.0714	1.1429	0.2143	6.2117		3.20	1.0714	1.1429	0.4286	4.41
	3.40	1.1538	1.3077	0.2308	7.4233		3.40	1.1538	1.3077	0.4615	5.0053
	3.60	1.25	1.5	0.25	8.9567		3.60	1.25	1.5	0.5	5.7
0.60	3.80	1.3636	1.7273	0.2727	10.9267	1.20	3.80	1.3636	1.7273	0.5455	6.5117
	4.00	1.5	2	0.3	13.4793		4.00	1.5	2	0.6	7.45
	4.20	1.1667	2.3333	0.3333	16.8417		4.20	1.1667	2.3333	0.6667	8.528
	4.40	1.875	2.75	0.375	21.3167		4.40	1.875	2.75	0.75	9.7567
	4.60	2.1429	3.2857	0.4286	27.2667		4.60	2.1429	3.2857	0.8571	11.13

A PE may be described by computing the sum of their weighted inputs, subtracting its threshold from the sum, and transferring these results by a function as follows (Haykin 1994):

$$u_i = \phi\left(\sum_{j=1}^{n} w_{ij} x_j - \theta_i\right) \tag{6}$$

where u_i represents the output of a PE, w_{ij} represents the synaptic weights associated with PE i, x_j represents the input signal, θ_i represents the threshold value of the PE, and ϕ (.) presents the transformation function which can be linear of non-linear. The transformation function is used for limiting the amplitude of the output of a PE. Any change in the synaptic weights changes the input-output behavior of the NN (Haykin 1994). The PEs work independent of each other through weighted connections that form the power of the influence between the PEs. All PEs operate in parallel and are connected to the other PEs in the next layer. A transformation function defines the output of a PE in terms of the activity level at its input. A linear transformation function's output will be equal to its input. The most common form of transformation function used in the construction of NN is (Neuro Solutions 2003):

$$f(x_i) = \tanh(\beta x_i) \quad (7)$$

where β is used for controlling the slope of the function. Equation (7) is a hyperbolic tangent function and generates output values between -1 and 1. There is no simple rule to determine the number of PEs required and is generally determined by trial-and-error. It highly depends on the problem and the number and the characteristics of training pattern. It is recommended to carry out a parametric study by changing the number of PE in the hidden layer in order to test the stability of the network (Neuro Solutions 2003). For most of the practical engineering problems, a single hidden layer with an optimum number of PEs is generally sufficient (Rafiq et al. 2001).

There are basically two classes of NN algorithm: (a) supervised learning, (b) unsupervised learning. Supervised learning NN algorithm, i.e. back propagation NNs, requires the training data to have been previously specified in different classes so that a subsequent test sample may be assigned to the most appropriate class. The major disadvantages of the back-propagation neural network (BPNN) are that they train slowly and require many training data. However, BPNN is the most commonly used NN for the analysis of structural and civil engineering problems due to its versatile and robust technique and are capable of solving predictive problems (Neuro Solutions 2003). Unsupervised learning, on the other hand, requires no a priori information, because it mainly organizes the data into clusters that effectively define the various classes or similarities that exist within the data set (Yeung and Smith 2005). In this study, supervised learning algorithms with static back-propagation neural network are selected.

The NN model in this study was constructed in three phases: the modeling, the training, and the testing phases. The data preparation and the adaptation of the learning law for the training were performed during the training phase.

3.1. Modeling Neural Network

The accuracy of the NN model is highly affected by the selection of the input variables significantly. Different results can be obtained for different parameters. The learning process slows down if there are too many input and output parameters, while too few training sets provides insufficient information. In this study, the three design parameters for the input layer; $X_1 = h/l$, $X_2 = b/l$, $X_3 = d/l$; were selected to evaluate the stiffness modification factor ($Y_1 = \eta^{FEA}$) as shown in Table 1 and Table 2. The parameters are h/l, b/l, d/l indicate the effect of openings between the two shear walls. The ranges of data for the selected variables which are practically used in design process can also be seen in Table 1, Columns 3, 4, and 5. All the design variables were input to the NN as given in Table 2.

A total of 54 cases were used for constructing the NN model. Table 2 shows the organized form of input and output parameters in Table 1. The stiffness modification factor ($Y_1 = \eta^{FEA}$) and the input variables from the 54 cases were divided into two sets. The first 44 cases in Table 2 was put aside for the training of the NN, and the last 10 cases (20% of the data) was used for testing the performance of the trained network (cases below the dotted line in Table 2). The testing data was selected at random order between the maximums and minimums. Before the modeling of the NN, the whole data set (training and testing) was first normalized. Normalization of the data set was carried out through normalization coefficients: amplitude (α) and offset (π).

The α and π were computed based on the minimum and maximum values found among all of the data set. The normalization coefficients for each input PE i (three in this study) were obtained using the following formula (Neuro Solutions 2003):

$$\alpha(i) = [\upsilon - \lambda] / [\max(i) - \min(i)] \quad (8)$$

$$\pi(i) = \upsilon - \alpha(i) x \max(i) \quad (9)$$

where max(i) and min(i) represent the maximum and minimum values found within the input PE i, and υ and λ were taken as (1) and (-1), respectively. Then, the data set was normalized as follows (Neuro Solutions 2003):

$$\rho(i) = \alpha(i) x \mathrm{Data}(i) + \pi(i) \quad (10)$$

At the end of the training, the NN data were denormalized using Equation (10) as follows:

$$\mathrm{Data}(i) = [\rho(i) - \pi(i)] / \alpha(i) \quad (11)$$

3.2. The Training Phase

Standard BPNN algorithm for the training of the network was employed in this study. A commercial NN software (Neuro Solutions 2003) was also used to implement

Table 2. Cases used in the NN model and comparisons

Case No.	(X_1) $\frac{h}{l}$	(X_2) $\frac{b}{l}$	(X_3) $\frac{d}{l}$	(Y_1) η^{FEA}	η^{NN}	η (Equation 5)	$\frac{\eta^{FEA}}{\eta}$	$\frac{\eta^{FEA}}{\eta^{NN}}$
1	1	1	0.0667	7.1773	7.2799	9.3884	0.764	0.986
2	1.0714	1.1429	0.0714	8.7477	8.7535	11.3161	0.773	0.999
3	1.1538	1.3077	0.0769	10.8100	10.7685	13.6188	0.794	1.004
4	1.25	1.5	0.0833	13.5700	13.5408	16.4066	0.827	1.002
5	1.3636	1.7273	0.0909	17.3500	17.3561	19.8132	0.876	1.000
6	1.5	2	0.1	22.6633	22.6564	24.0406	0.943	1.000
7	1.1667	2.3333	0.1111	30.3667	30.2951	29.0871	1.044	1.002
8	1.875	2.75	0.125	41.9767	42.8122	36.2719	1.157	0.980
9	2.1429	3.2857	0.1429	60.1833	57.4470	45.4138	1.325	1.048
10	1	1	0.1333	6.0683	6.1557	6.1954	0.979	0.986
11	1.0714	1.1429	0.1429	7.2900	7.2845	7.4573	0.978	1.001
12	1.1538	1.3077	0.1538	8.8610	8.8125	8.9742	0.987	1.006
13	1.25	1.5	0.1667	10.9070	10.8709	10.8031	1.010	1.003
14	1.3636	1.7273	0.1818	13.6217	13.6225	13.0461	1.044	1.000
15	1.5	2	0.2	17.2900	17.2291	15.8250	1.093	1.004
16	1.1667	2.3333	0.2222	22.3567	22.4006	19.1495	1.167	0.998
17	1.875	2.75	0.25	29.5100	28.8412	23.8667	1.236	1.023
18	2.1429	3.2857	0.2857	39.8833	41.4327	29.8852	1.335	0.963
19	1	1	0.2	5.2483	5.3092	4.9154	1.068	0.989
20	1.0714	1.1429	0.2143	6.2117	6.1911	5.9183	1.050	1.003
21	1.1538	1.3077	0.2308	7.4233	7.3697	7.1194	1.043	1.007
22	1.25	1.5	0.25	8.9567	8.9421	8.5733	1.045	1.002
23	1.3636	1.7273	0.2727	10.9267	11.0067	10.3523	1.055	0.993
24	1.5	2	0.3	13.4793	13.6369	12.5574	1.073	0.988
25	1.1667	2.3333	0.3333	16.8417	16.9136	15.1953	1.108	0.996
26	1.875	2.75	0.375	21.3167	20.8576	18.9385	1.126	1.022
27	2.1429	3.2857	0.4286	27.2667	26.6631	23.7127	1.150	1.023
28	1	1	0.2667	4.6417	4.6665	4.1712	1.113	0.995
29	1.0714	1.1429	0.2857	5.4150	5.3622	5.0229	1.078	1.010
30	1.1667	2.3333	0.5555	10.4083	10.2612	11.3544	0.917	1.014
31	1.875	2.75	0.625	12.2567	12.5238	14.1514	0.866	0.979
32	2.1429	3.2857	0.7143	14.4367	14.6553	17.7193	0.815	0.985
33	1.1538	1.3077	0.3077	6.3633	6.2788	6.0423	1.053	1.013
34	1.25	1.5	0.3333	7.5333	7.4844	7.2762	1.035	1.007
35	1.3636	1.7273	0.3636	8.9733	9.0437	8.7855	1.021	0.992
36	1.5	2	0.4	10.7900	11.0004	10.6560	1.013	0.981
37	1.1667	2.3333	0.4444	13.0513	13.0868	12.8950	1.012	0.997
38	1.875	2.75	0.5	15.8900	16.0888	16.0720	0.989	0.988
39	2.1429	3.2857	0.5714	19.4433	19.3161	20.1250	0.966	1.007
40	1	1	0.3333	4.2067	4.1819	3.6731	1.145	1.006
41	1.875	2.75	0.75	9.7567	9.7341	12.7536	0.765	1.002
42	2.1429	3.2857	0.8571	11.1300	11.0020	15.9697	0.697	1.012
43	1.0714	1.1429	0.3571	4.8317	4.7362	4.4228	1.092	1.020
44	1.1538	1.3077	0.3846	5.5810	5.4528	5.3203	1.049	1.024
1	1.25	1.5	0.4167	6.4800	6.3734	6.4059	1.012	1.017
2	1.3636	1.7273	0.4545	7.5590	7.5365	7.7355	0.977	1.003
3	1	1	0.4	3.9000	3.8223	3.3101	1.178	1.020
4	1.0714	1.1429	0.4286	4.4100	4.2730	3.9855	1.107	1.032
5	1.5	2	0.5	8.8550	8.9574	9.3832	0.944	0.989
6	1.1538	1.3077	0.4615	5.0053	4.8426	4.7949	1.044	1.034
7	1.25	1.5	0.5	5.7000	5.5518	5.7735	0.987	1.027
8	1.3636	1.7273	0.5455	6.5117	6.4129	6.9707	0.934	1.015
9	1.5	2	0.6	7.4500	7.4195	8.4564	0.881	1.004
10	1.1667	2.3333	0.6667	8.5280	8.1977	10.2320	0.833	1.040
				Mean			1.011	1.004
				Standard Deviation			0.135	0.016
				Coefficient of variation			0.133	0.016

this training method. The NN model in this study was created using an input layer of three-interconnected PEs corresponding to the three-input parameters, and one PE corresponding to an output layer selected as the target. Several trials during the testing phase led to the selection of one hidden layer. The hyperbolic tangent function was used as the transformation function in this study.

In BPNN algorithm, training or learning data are introduced into the NN with a series of examples of associated input and target output values. First, the input data is processed through the input layer to hidden layer until it reaches the output in a forward direction; then, the output is compared to the given output. The error between the NN output and target output is processed back through the network (also called backward pass) adjusting the individual weights (Ashour and Alqedra 2005). During this learning process, a gradual reduction of error between the model output and the target output occurs and the error is minimized so as to minimize the sum of squared errors (Haykin 1994). The mean square error (MSE) is defined as (Gunaydin and Dogan 2004):

$$MSE = \frac{\sqrt{\sum_{i=1}^{n}(x_i - X(i))^2}}{n} \tag{12}$$

where n is the number of samples to be evaluated in the training phase, (n=44 for this study), x_i is the model output related to the sample i (i=1, 2, ...,44), and X(i) is the target output, i.e. the estimated stiffness modification factor. MSE is considered to be a good indicator of how successful the training run was. The error was measured for each run of the epoch number selected and the results are shown in Figure 4.

Epoch number defines the training of all cases in a training set. Figure 4 is a typical learning curve and indicates a reduction in the MSE from 0.01 to 0.0002. Training should be stopped when the MSE remains unchanged for a given number of epochs. If it is not stopped, the NN starts memorizing the training values and will not be able to make predictions when an unknown example is introduced to the NN. For supervised learning control, the maximum number of epochs should be specified showing the number of iterations over the training set. In this study, an epoch number of 5000 was found to be adequate for the final training process in a series of more than 100 runs for each NN model.

3.3. The Testing Phase

In a NN model, testing set is used for testing the performance of the network. The testing is performed with the best weights obtained during the training, which remain unchanged during the testing. The trained weighting factors of the NN are verified by the testing data to test the accuracy of the predictions of the trained NN model. The NNs performance in this study was measured by using the stiffness modification factor percentage error (PE_{SR}) formula as follows:

$$PE_{SR} = \frac{x(i) - X(i)}{X(i)} x100\% \tag{13}$$

To evaluate the entire NNs overall performance, weighted error (WE) was defined as follows (Hegazy and Ayed 1998):

$$WE(\%) = 0.5\,(\text{Average } PE_{SR} \text{ for Training Set}) + 0.5\,(\text{Average } PE_{SR} \text{ for Testing Set}) \tag{14}$$

PE_{SR} for the testing set in the NN model is given in Figure 5.

Average PE_{SR} for the 10 testing cases was calculated as 1.98%, while it was 1.06% for the training set (44 cases).

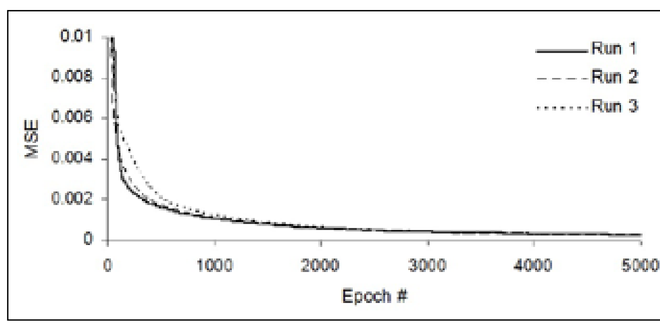


Figure 4. Learning curve.

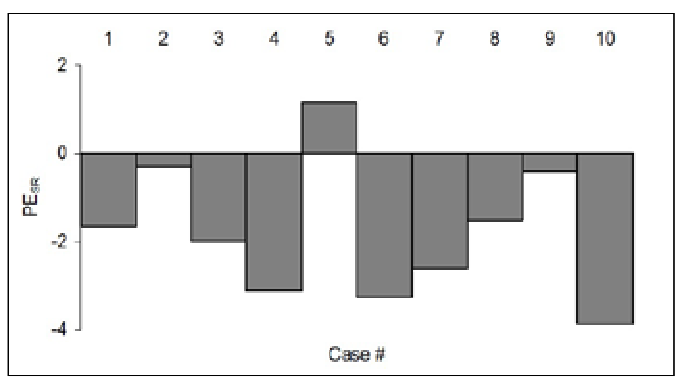


Figure 5. Error on the estimated stiffness modification factor of the coupling beam of CSW vs. theoretical stiffness modification factor of the coupling beam of CSW for the ten testing samples.

Thus, the WE was found to be 1.52%. Sensitivity analysis provides valuable information about how each network input affects the network output. This analysis allows the option of removing the insignificant channels from the network by reducing the size of the network.

This reduces the complexity and the training time of the NN. Since the network training is disabled during the sensitivity analysis, the network weights are not affected. The inputs to the network are shifted slightly and the corresponding change in the output is shown as a percentage summing to 100% in total (Neuro Solutions 2003). In this study, the most effective parameter was found to be b/ℓ and d/ℓ while h/ℓ was observed to be the least significant design parameter (Figure 6).

3.4. Comments on Results

Data from 10 cases were used for testing purposes in this study out of a total 54 cases. The results showed 98.48% of average accuracy with a MSE of 0.0002. These figures were considered to be quite good. Even though, h/ℓ had no significant effect on estimating the stiffness modification factor, the small attributes provided by this design parameter may have enhanced the NNs prediction capability, i.e. the more the number of design parameters, the higher the accuracy. The results were also tested through another NN application and similar results were obtained (Hegazy and Ayed 1998). The WE was found to be 1.24% with an average PE_{SR} of 1.56% and 0.93% for training and testing cases, respectively, using Hegazy and Ayed's method (1998).

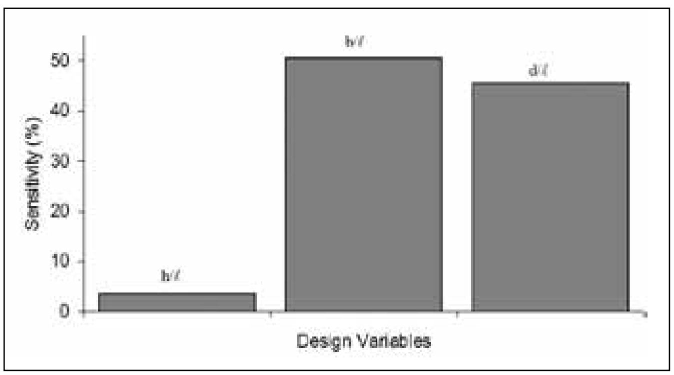
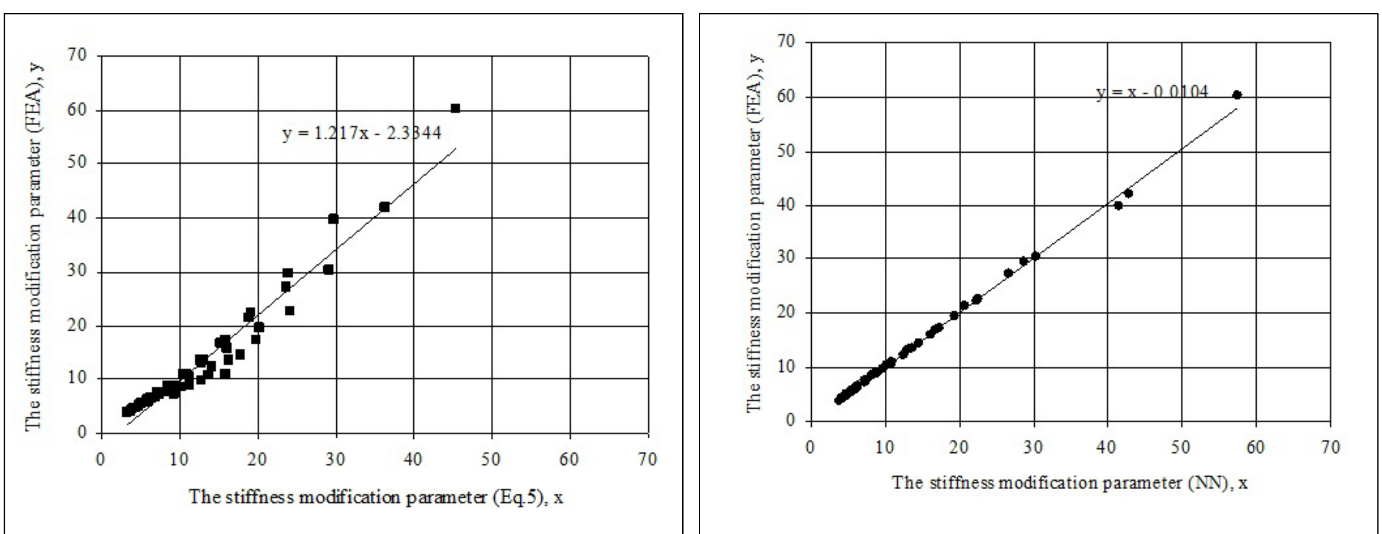


Figure 6. Sensitivity analysis.

The discrepancies between the finite element and analytical values obtained from equivalent frame method and NN algorithm are further investigated by means of statistical analysis. Table 2 lists the complete set of the values of the stiffness modification factor for finite element solutions by using 2D shell element concept (η^{FEA}), NN solution (η^{NN}), and analytical values of the stiffness modification factors obtained from Equation (5) (η).

Mean values, standard deviations and coefficients of variation of the ratio of the finite element to analytical values, are given in Table 2. The standard deviation is accepted as a convenient measure of any dispersion in the statistical approach. However, exclusively on the basis of the standard deviation, any decision about whether the dispersion is large or small cannot be reached. Therefore, coefficient of variation is often a preferred and convenient



a. analytical solutions

b. NN solutions

Figure 7. Scatter diagrams.

non-dimensional measure of dispersion or variability. For ratio η^{FEA} / η, the mean, standard deviation and coefficient of variation are 1.011, 0.135 and 0.133, respectively, as shown in Table 2. Also for ratio η^{FEA} / η^{NN}, the mean, standard deviation and coefficient of variation are 1.004, 0.016 and 0.016, respectively. These results imply that the NN model constructed in this study provides better results than obtained from Equation (5).

Besides, the scatter diagrams for statistical evaluation are shown in Figure 7. The correlation between random variables η and η^{FEA} is measured by the correlation coefficient ρ.

Based on the values of η and η^{FEA} in Table 2, the value of the correlation coefficient is estimated to be 0.974. Similarly, the value of the correlation coefficient is estimated to be 0.999 based on the values of η^{NN} and η^{FEA} in Table 2. Two values of ρ indicate that there are very high correlations between the random variables. It shows that this relationships are likely linear. After this assumption, the least-squares regression line for ($X = \eta$, $Y = \eta^{FEA}$) is given by $E(Y/x) = 1.217x - 2.3344$ (Figure 7a). On the other hand, the least-squares regression line for ($X = \eta^{NN}$, $Y = \eta^{FEA}$) is given by $E(Y/x) = x - 0.0104$ (Figure 7b). As can be seen in Figure 7b, the scatter diagram of the stiffness modification factors, η^{NN} and η^{FEA}, yields closer results.

4. Conclusions

In this study, the NN model was employed to develop and test the stiffness modification factor predictions for coupling beams of CSW systems. The data of 44 cases were used to train the NN. The testing of the NN was done by the data of 10 testing cases. The approach adapted in this study was shown to be capable of providing the best accurate estimates of the stiffness modification factor by using the three design parameters. The results are quite promising for further research of expanded data sets. The stiffness modification factor for coupling beams of CSW systems can be predicted by setting up some random variations in the design parameters.

5. References

Alacalı, SN., Doran, B., Akbas, B. 2006. Predicting Lateral Confinement Coefficient for R/C Rectangular Columns By a Neural Network Model. *EAEE, First European Conference on Earthquake Engineering and Seismology*, Geneva, Switzerland, 3-8 September, Paper Number: 1159.

Amini, F., Tavassoli, MR. 2005. Optimal Active Structural Control Force, Number and Placement of Controllers. *Eng. Struct.*, 27: 1306-1316.

Ashour, A.F., Alqedra, MA. 2005. Concrete breakout strength of single anchors in tension using neural networks. *Adv Eng Softw.*, 36: 87–97.

Cladera, A., Mari, AB. 2004. Shear design procedure for reinforced normal and high-strength concrete beams using artificial neural networks. Part I: beams without stirrups. *Eng. Struct.*, 26: 917–926.

Doran, B. 2004. Elastic-Plastic analysis of R/C coupled shear walls: the equivalent stiffness ratio of the tie elements. *Ind. Inst. of Sci.,IISC,* 83: 87-94.

Flood, I. 1989. A Neural Network Approach to the Sequencing of Construction Tasks. *IAARC, Proceedings of the Sixth International Symposium on Automation and Robotics in Construction*, Construction Industry Institute, Austin, Texas, 204-11.

Garret, JH., Gunaratham, DJ., Ivezic, N. 1997. In: Kartam, N., Flood, I., editors. Artificial neural networks for civil engineers: fundamentals and applications. *In: Kartam N., I.Flood (Ed.), ASCE*, 1-17.

Gunaydin, HM., Dogan, SZ. 2004. A neural network approach for early cost estimation of structural systems of buildings. *Int J Proj Manag.*, 22: 595-602.

Haykin, S. 1994. *Neural Networks: A comprehensive Foundation*, Macmillan College Publishing Company, Inc. NJ.

Hegazy, T., Ayed, A. 1998. Neural network model for parametric cost estimation of highway projects. *J Civ Eng Manag, ASCE,* 124(3): 210-218.

Kinh, HH., Tan, TMH. 1999. An efficient analysis of continuum shear wall models. *Can. J. Civ. Eng.*, 26(4): 425-433.

Lee, JJ., Lee, WL., Yi, JH., Yun, CB., Jung, HY. 2005. Neural networks-based damage detection for bridges considering errors in baseline finite element models. *J. Sound Vib,* 280: 555–578.

Neuro Solutions. 2003. Neurodimension, Inc., Version 4.24.

Pala, S., Özmen, G. 1995. Effective Stiffness of Coupling Beams in Structural Walls. *Comput Struct.*, 54(5): 925-931.

Papadrakakis, M., Lagaros, ND., Plevris, V. 2005. Design optimization of steel structures considering uncertainties. *Eng. Struct.*, 27: 1408–1418.

Rafiq, MY., Bugmann, G., Easterbrook, DJ. 2001. Neural network design for engineering applications. *Comput Struct., 79*: 1541-52.

Saadata, S., Nooria, MN., Bucknera, GD., Furukawab, T., Suzukic, Y. 2004. Structural health monitoring and damage detection using an intelligent parameter varying (IPV) technique. *Int. J. Non-linear Mech.,* 39: 1687 – 1697.

Syngellakis, S., Chan, AKL. 1992. Free vibrations of coupled walls by transfer matrices and finite element modelling of joints. *Comput Struct.,* 44(5):1239-1247.

Tehranizadeh, M., Safi, M. 2004. Application of artificial intelligence for construction of design spectra. *Eng. Struct.,* 26: 707–720.

Wadsworth, HM. 1997. *Handbook of Statistical Methods for Engineers and Scientists*. McGrawHill Inc, NewYork.

Williams, ME., Hoit, MI. 2004. Bridge pier live load analysis using neural networks. *Adv. Eng. Softw.*, 35: 645–652.

Yeung, WT., Smith, JW. 2005. Damage detection in bridges using neural networks for pattern recognition of vibration signatures. *Eng. Struct.*, 27: 685–698.