Eng. & Tech. Journal, Vol.30, No.19, 2012

A Neural Network Model to Predict Ultimate Strength of Rectangular Concrete Filled Steel Tube Beam – Columns

Dr. Ahmed Sagban Saadoon
Engineering College, University of Basrah / Basrah
Email:asagban@yahoo.com
Kadhim Zuboon Nasser
Engineering College, University of Basrah / Basrah
Ihsan Qasim Mohamed
Engineering College, University of Basrah / Basrah

Received on:6/6/2011 & Accepted on7/6/2012

ABSTRACT

In this study, a model for predicting the ultimate strength of rectangular concrete filled steel tube (RCFST) beam-columns under eccentric axial loads has been developed using artificial neural networks (ANN). The available experimental results for (111) specimens obtained from open literature were used to build the proposed model. The predicted strengths obtained from the proposed ANN model were compared with the experimental values and with unfactored design strengths predicted using the design procedure specified in the AISC and Eurocode 4 for RCFST beam-columns. Results showed that the predicted values by the proposed ANN model were very close to the experimental values and were more accurate than the AISC and Eurocode 4 values. As a result, ANN provided an efficient alternative method in predicting the ultimate strength of RCFST beam-columns.

Keywords: Beam-Columns, Artificial Neural Networks, Concrete Filled Steel Tube.

نموذج شبكة عصبية لتقدير المقاومة القصوى للجسور-الأعمدة المكوّنة من أنبوب حديدي ذي مقطع مستطيل الشكل مملوء بالخرسانة

الخلاصة

ان الهدف الرئيسي من الدراسة الحالية هو بناء شبكة عصبية صناعية لتقدير مقاومة الجسور-الأعمدة المكوّنة من أنبوب حديدي ذي مقطع مستطيل الشكل مملوء بالخرسانةوالمعرّضة إلى أحمال ضغط لامركزية. وقد استعملت النتائج المختبرية لـ(111) عيّنة (مستخلصة من بحوث سابقة) في بناء الشبكة المقترحة.وقورنت القيم المقدّرة من هذه الشبكة مع القيم المختبرية ومع القيم المحسوبة على ضوء شرط التصميم في الكودين العالميين AISC و Eurocode 4. لقد أظهرت النتائج أنالقيم المقدّرة من الشبكة المقترحة كانت قريبة جداً من القيم المختبرية وكانت أدقّ من القيم المحسوبة حسب مواصفات الكودين المذكورين. وبالتالي فانه من الممكن استخدام الشبكات العصبية في تقدير مقاومة مثل هذا النوع من الجسور-الأعمدة.

INTRODUCTION

Beam-columns are members that are subjected simultaneously to axial forces and bending moments. Thus, their behavior falls somewhere between that of a pure axially loaded column and that of a beam with only moments applied. Also, their behavior must include the effects of the axial loads on the flexural stiffness. This is usually referred to as the second-order elastic analysis. To understand the behavior of beam-columns, it is common practice to look at the response as predicted through an interaction equation between axial loads and moments.

Composite construction is widely used in structural systems to achieve long spans, lower story heights, and provide additional lateral stiffness. Composite construction uses the structural and constructional advantages of both concrete and steel. Concrete has low material costs, good fire resistance, and is easy to place. Steel has high ductility and high strength-to-weight and stiffness-to-weight ratios. There are two basic kinds of composite beams or columns: steel sections encased in concrete (steel-reinforced concrete sections or SRC) and steel sections filled with concrete (concrete filled tubes or CFT). The latter can be either circular (CCFT) or square/rectangular (RCFT) in cross-section. In composite columns additional synergies between concrete and steel are possible: (a) in concrete-filled tubes, the steel increases the strength of the concrete because of its confining effect, the concrete inhibits local buckling of the steel, and the concrete formwork can be omitted; and (b) in encased sections, the concrete delays failure by local buckling and acts as fireproofing while the steel provides substantial residual gravity load-carrying capacity after the concrete fails. The structural behavior of rectangular concrete filled steel tube (RCFST) beam-columns has been investigated through many experimental tests [1-5]. The main objective of these tests was to determine the different parameters that influence the structural behavior of this type of beam-columns.

Artificial Neural Networks (ANN) is a branch of artificial intelligence. Networks are modeled after the human brain consisting of brain cells and connections. As in the human brain, these networks are capable of learning from examples. The technology has been used successfully in pattern recognition problems. It can deal with new situations and incomplete information. Neural networks learn by adjusting their connection weights. Most networks are based on supervised learning algorithms in which pairs of input and desired output are shown to them during a training session. For the last two decades, different modeling methods based on ANN have become popular and have been used by many researchers for a variety of civil engineering applications [6-10].

The aim of this study is to propose a model using ANN to predict the ultimate strength of RCFST beam-columns under eccentric axial loads (Figure (1)).

ARTIFICIAL NEURAL NETWORKS (ANN)

ANN is a system that mimics the human brain and therefore a great deal of the terminology is borrowed from neuroscience. The most basic element of the human brain is a specific type of cell, which provides us with the abilities to remember, think and apply previous experience to our every action. These cells are known as neurons, each of these neurons can connect with up to 200,000 other neurons. One type of network sees the nodes as artificial neurons. These are called artificial neural networks (ANNs). An artificial neuron is a computational model inspired in the natural neurons. Natural neurons receive signals through synapses located on the dendrites or membrane of the neuron. When the signals received are strong enough (surpass a certain threshold), the neuron is activated and emits a signal through the axon. This signal might be sent to another synapse, and might activate other neurons [11].

The complexity of real neurons is highly abstracted when modeling artificial neurons. These basically consist of inputs (like synapses), which are multiplied by weights (strength of the respective signals), and then computed by a mathematical function which determines the activation of the neuron. Another function (which may be the identity) computes the output of the artificial neuron (sometimes in dependence of a certain threshold). ANNs combine artificial neurons in order to process information. Figure (2) shows an artificial neuron model with input, sum function, sigmoid activation function and output. The input to a neuron from another neuron is obtained by multiplying the output of the connected neuron by the synaptic strength (weight) of the connection between them. The higher a weight of an artificial neuron is, the stronger the input which is multiplied by it will be. The weighted sums of the input components $(net)_j$ are calculated by using the following equation:

$$(net)_j = \sum_{i=1}^{n} W_{ij} Y_i + B, \qquad (1)$$

where $(net)_j$ is the weighted sum of the jth neuron for the input received from the preceding layer with n neurons, W_{ij} is the weight between the jth neuron in the preceding layer, Y_i is the output of the ith neuron in the preceding layer. The quantity B is called the bias and is used to model the threshold. The output signal of the neuron, denoted by Y_j in Fig. (2), is related to the network input $(net)_j$ via a transformation function called the activation function. The most common activation functions are sigmoid and Gaussian function. The output of the jth neuron Y_j is calculated by using Eq. (2) with a sigmoid function as follows:

$$Y_j = f(net)_j = \frac{1}{1+e^{-a(net)_j}} \qquad (2)$$

where α is a constant used to control the slope of the semi-linear region. Depending on the weights, the computation of the neurons will be different. By adjusting the weights of an artificial neuron, the output is obtained for specific inputs. But when an ANN has hundreds or thousands of neurons, it would be quite complicated to find by hand all the necessary weights. But an algorithm can be found which can adjust the weights of the ANN in order to obtain the desired output from the network. This process of adjusting the weights is called learning or training

A trained network presents some distinct advantages over the numerical computing. It provides a rapid mapping of a given input into the desired output quantities. The other important advantage of neural networks is either correct or nearly correct responses to the incomplete tasks, their extraction of information from noisy or poor data and their production of generalized results. This makes neural networks a very powerful tool to solve many civil engineering problems, particularly in the problems which data maybe complex or in insufficient amount [6].

The number of types of ANNs and their uses is very high. The differences in them might be the functions, the accepted values, the topology, the learning algorithms, etc. There are several important neural network models such as Hopfield net, Hamming net, Single-layer preceptor and Multi-layer network. The most important and powerful nets are multi-layer networks [12]. Multi-layer feed-forward networks are the most popular and most widely used models in many practical applications. In a feed-forward NN, the artificial neurons are arranged in layers, and all the neurons in each layer have connections to all the neurons in the next layer. However, there is no connection between neurons of the same layer or the neurons which are not in successive layers. In general, the feed-forward NN consists of one input layer, one or two hidden layers and one output layer of neurons. The input layer receives input information and passes it onto the neurons of the hidden layer(s), which in turn pass the information to the output layer. The output from the output layer is the prediction of the net for the corresponding input supplied at the input nodes. Each neuron in the network behaves in the same way as discussed in Eqs. (1) and (2). There is no reliable method for deciding the number of neural units required for a particular problem. This is decided based on experience and a few trials are required to determine the best configuration of the network. In this study, the multi-layer feed-forward type of neural networks, as shown in Fig. (3), is considered.

Multi-layer feed-forward NNs are trained with supervised learning rules. The ANN which learns using the backpropagation algorithm is one of the most common models used in ANNs. In ANNs that used backpropogation, also known as Error Backpropagation or Generalized Delta Rule, as a learning algorithm training is implemented by adjusting the weights according to the error (the distance between the target and the actual output vector) in the output layer. The learning error for *r*th example is calculated by the following performance function usually called the mean-square error:

$$E_r = \frac{1}{2}\sum_j (T_j - Y_j)^2 , \qquad (3)$$

where T_j is the target output at neuron j and Y_j is the output predicted at neuron j. As presented in Eqs. (1) and (2) the output Y_j is a function of synaptic strength and outputs of the previous layer. In the back-propagation phase, the error between the network output and the desired output values is calculated using the so called generalized delta rule, and weights between neurons are updated from the output layer to the input layer. These operations are repeated for each example and for all the neurons until a satisfactory convergence is achieved for all the examples present in the training set. The training process is successfully completed, when the iterative process has converged. The connection weights are captured from the trained network, in order to use them in the recall phase. There are several different backpropagation training algorithms. They have a variety of different computation and storage requirements and no one algorithm is best suited to all locations. The resilient backpropagation (RPROP) algorithm is used in this study. The commonly used back-propagation neural networks are trained by feeding a data of associated with input and target variables. The main objective of training the neural network is to update the connection weights to reduce the errors between the actual output values and target output values to a satisfactory level. This process is carried out through the minimization of the defined error function using update method. Also, the determination of number of hidden layers, number of hidden nodes, transfer functions, and normalization of data are considered to get the best performance of model. At the end of the training phase, the neural network represents a model that should be able to predict the target value given for the input pattern. After the errors are minimized, the associated trained weights of the model are tested with a separated set of testing data that was not used in training

ANN FOR BEAM-COLUMNS

An ANN model was developed to predict the ultimate strength of RCFST beam-columns under eccentric axial loads. The backpropagation algorithm and construction of the proposed ANN model was carried out in the neural network toolbox of MATLAB version 7.0 (R14).

Selection of Training and Testing Data

The experimental results of (111) test RCFST beam-columns that are used to build the proposed ANN model are obtained from a database developed by Kim [13]. The data used to build the ANN model should be divided into two subsets: training data and validating or testing data. The testing data contains approximately 20% from total database. The training phase is needed to produce a NN that is both stable and convergent. Therefore, selection of what data to use for training a network is one of most important steps in building a NN model. The total number of (111) test beam-columns were utilized. The training data contained (88) samples and the testing data comprised of (23) samples. ANNs interpolate data very well. Therefore, patterns chosen for training set must cover upper and lower boundaries and a sufficient number of samples representing particular features over the entire training domain [14].

Input and Output Layer

The nodes in the input layer and output layer are usually determined by the nature of the problem. In this study the parameters which may be introduced as the components of the input vector consist of yield stress of steel tube (f_y), cylinder concrete compressive strength (f'_c), width of composite cross-section perpendicular to plane of bending (h), width of composite cross-section parallel to plane of bending (b), thickness of steel tube (t), laterally unbraced length of member (L), and eccentricity of applied load (e). The output vector is the ultimate axial load (P). Table (1) summarizes the ranges of each different variable.

Proposed ANN Model

For selecting the best configuration of the networks, there are no special guidelines, and trial and error approach should be employed that takes into consideration the best network performance, average error and the best network performance for the testing data. A multi-layer feed-forward NN with a resilient back-propagation algorithm was employed in the present study. The NN architecture developed has seven neurons in the input layer and one neurons in the output layer as demonstrated in Fig. (4). Two hidden layers were used in the architecture of multi-layer feed-forward NN due to its minimum absolute percentage error values for training and testing sets. In the first hidden layer six and in the second hidden layer five neurons were determined. The transfer (activation) functions used are hyperbolic tangent (tansig) function in first hidden layer and linear (purelin) function in both second hidden and output layer.

RESULTS AND DISCUSSION

The performance of the proposed ANN is tested by the regression analysis between the output of this network (predicted values) P(ANN) and the corresponding targets (experimental values) P(exp) for both training and testing data as shown in Figs. (5) and (6), respectively. The coefficient of correlation (R^2) is a measure of how well the variation in the output is explained by the targets and is given by the following equation:

$$R^2 = \frac{(n\sum T_i Y_i - \sum T_i \sum Y_i)^2}{(n\sum T_i^2 - (\sum T_i)^2)(n\sum Y_i^2 - (\sum Y_i)^2)} \qquad (4)$$

where T is the target value, Y is the output value, and n is the pattern. If this number is equal to 1, then there is a perfect correlation between targets and output. In these figures, the coefficient of correlation $R^2 = 0.98$ for both training and testing data. This value indicates an excellent agreement between the predicted values and the experimental values.

Table (2) presents the actual and predicted values for testing data. As seen from this table, the values obtained are very close to the experimental results. The average value of ratios of actual to predicted ultimate loads is 1.0. This result demonstrates that ANN can be successfully applied to establish accurate and reliable prediction models.

Based on these results, the proposed ANN architecture (7-6-5-1) with activation functions (tansig, purelin, purelin) with (RPROP) is used for this study. Table (3) shows the properties of this network.

COMPARATIVE STUDY

The predicted strengths, of the RCFST beam-columns in Table (2), obtained from the proposed ANN model are compared with unfactored design strengths predicted using the design procedure specified in the American Institute of Steel Construction (AISC) [15] and the Eurocode 4 [16] for RCFST beam-columns as calculated by Kim [13]. The predicted strengths of the proposed ANN model P(ANN) are compared with the design strengths calculated using AISC specifications P(AISC) and the design strengths calculated using Eurocode 4 specifications P(Euro) as shown in Table (4). In the present study, two norms were used to comparative evaluation of the performance of the proposed ANN and the calculated design strengths using AISC and Eurocode 4 specifications. These norms are the absolute fraction of variance (*VAF*) and the mean absolute percentage error (*MAPE*) between predicted and experimental results and they are given, respectively, as:

$$VAF = 1 - \frac{\sum (T_i - Y_i)^2}{\sum Y_i^2} \qquad (5)$$

$$MAPE = \frac{1}{n}\left[\frac{\Sigma\left|T_i - Y_i\right|}{\Sigma T_i} \times 100\right] \qquad (6)$$

where T is the target value, Y is the output value, and n is the pattern. The calculated indices are given in Table (4). If VAF is 1 and $MAPE$ is 0, then the model will be excellent.

It can be seen, from Table (4), that the statistical values of VAF and $MAPE$ are 0.994, 8.9% for ANN, 0.808, 81.2% for AISC, and 0.767, 94.3% for Eurocode 4, respectively. These values indicate that the proposed ANN model can predict more accurate results than AISC and Eurocode 4 methods.

In Fig. (7), the predicted strengths P(ANN) and the design strengths P(AISC) and P(Euro) are plotted against the experimental strengths. As shown in this figure, the coefficient of correlation R^2 = 0.98, 0.88 and 0.84 for ANN, AISC, and Eurocode 4, respectively. These values clearly show that the proposed ANN performs much better than the AISC and Eurocode 4 methods and that ANN provided an efficient alternative method in predicting the ultimate strength of RCFST beam-columns.

CONCLUSIONS

It has been shown that artificial neural networks can effectively be used to predict the ultimate strengths of RCFST beam-columns subjected to eccentric axial loads. It has also been shown that the network designed predicted the outputs with acceptable accuracy. It should be noted that once the network was trained, the time required to output results for a given set of inputs was instantaneous. This indicates the potential of neural networks for solving time-consuming problems. Furthermore, artificial neural networks directly use the experimental results in training, there is no need to make any assumptions on material parameters particularly in problems that have more than one existing calculation method, or the one based on only empirical approximations. For selecting the best configuration of the networks, there are no special guidelines, and trial and error approach should be employed that takes into consideration the best network performance, average error and the best network performance for the testing data. A multi-layer feed-forward NN with a resilient back-propagation algorithm was used in the developed ANN model. In this model two hidden layers were selected. In the first hidden layer 6 neurons and in the second hidden layer 5 neurons were determined. This model was trained with input and output data. Using only the input data in trained model the ultimate strengths of RCFST beam-columns were found. The ultimate strength values predicted were very close to the experimental results.

In the comparative study, the predicted strengths obtained from the proposed ANN model were compared with current design provision for RCFST beam-columns (AISC and Eurocode 4). It was noticed that the proposed ANN model can predict more accurate results than AISC and Eurocode 4 specifications. Thus, ANN provided an efficient alternative method in predicting the ultimate strength of RCFST beam-columns.

ANN gives better results if the number of the training data is large. However, collecting this data for RCFST beam-columns is difficult because little experimental study exists in the open literature. Therefore, the present ANN model can further be improved using more experimental dataset.

REFERENCES

[1]. Hardika, M. S. and Gardner, N. J., 2004, "Behavior of Concrete-filled Hollow Structural Section Beam Columns to Seismic Shear Displacements", ACI Structural Journal, 101(1), 39-46.

[2]. Han, L. and Yao, G., 2000, "Influence of Concrete Compaction on the Strength of Concrete-filled Steel RHS Columns", Journal of Constructional Steel Research, 59(6), 751-767.

[3]. Seo, S. Y. and Chung, J. A., 2002, "An Experimental Study on Strength of Slender Square Tube Columns Filled with High Strength Concrete", International Journal of Steel Structures, Korean Society of Steel Construction, 14(4), 471-479.

[4]. Seo, S. Y., Tsuda, K., and Nakamura, A., 2002, "A Study on Strength of Steel Square Tubular Columns Filled with High Strength Concrete under Biaxial Eccentric load", International Journal of Steel Structures, Korean Society of Steel Construction, 14(5), 567-576.

[5]. Uy, B., 2001, "Strength of Short Concrete-filled High Strength Steel Box Columns", Journal of Constructional Steel Research, 57(2), 113-134.

[6]. Sanad, A. and Saka, M.P., 2001, "Prediction of Ultimate Shear Strength of Reinforced Concrete Deep Beams Using Neural Networks", Journal of Structural Engineering, 127(7), pp. 818-828.

[7]. Lima, L.R.O., Vellasco, P.C.G., Andrade, S.A.l., and Silva, J.G.S., 2005, "Neural Networks Assessment of Beam-to-column Joints", Journal of the Brazilian Society of Mechanical Sciences and Engineering, 27(3), pp.314-324.

[8]. Seleemah, A.A., 2005, "A Neural Network Model for Predicting Maximum Shear Capacity of Concrete Beams Without Transverse Reinforcement", Canadian Journal of Civil Engineering, 32(4), pp. 644-657.

[9]. Kim, J.I., Kim, D.K., Feng, M.Q., and Yazdani, F., 2004, "Application of Neural Networks for Estimation of Concrete Strength", Journal of Materials in Civil Engineering, 16(3), pp. 257-264.

[10]. Oreta, W.C. and Kawashima, K., 2003, "Neural Network Modeling of Confined Compressive Strength and Strain of Circular Concrete Columns", Journal of Structural Engineering, 129(4), pp. 554-561.

[11]. Hajela, P. and Berke, L., 1991, "Neurobiological Computational Models in Structural Analysis and Design", Computers and Structures, 4(41), 657-667.

[12] Chassiakos, A.G. and Masri, S F., 1991, "Identification of Internal Faces of Structural Systems Using Feed Forward Multi-layer Networks", Computing Systems Engineering, 1(5), 125-134.

[13] Kim, D.K., 2005, "A Database for Composite Columns", M.Sc thesis, School of Civil and Environmental Engineering, Georgia Institute of Technology.

[14]. Rafiq, M.Y., Bugmann, G., and Easterbrook, D.J., 2001, "Neural Network Design for Engineering Applications", Journal of Computers and Structures, 79(17), pp. 1451-1552.

[15]. AISC, 2005, "Specification for Structural Steel Buildings", American Institute of Steel Construction, Chicago, IL.

[16]. Eurocode 4, 1994, DD ENV 1994-1-1, "Design of Composite Steel and Concrete Structures, Part 1.1: General Rules and Rules for Buildings (with U.K National Application Document)", British Standards Institution, London.

Table (1) Range of input parameters.

Parameters	**Range**
Yield stress of steel tube (f_y) (MPa)	255-835
Cylinder concrete compressive strength (f'_c) (MPa)	23.1-103
Width of composite cross-section perpendicular to plane of bending (*h*) (mm)	76.2-324
Width of composite cross-section parallel to plane of bending (*b*) (mm)	76.2-324
Thickness of steel tube (*t*) (mm)	2.13-10
Laterally unbraced length of member (*L*) (mm)	250-3750
Eccentricity of applied load (*e*) (mm)	4.6-465.6

Table (2) Actual (experimental) and predicted values for testing data.

Column designation	f_y **(MPa)**	f'_c **(MPa)**	***h*** **(mm)**	***b*** **(mm)**	***t*** **(mm)**	***L*** **(mm)**	***e*** **(mm)**	***P(exp)*** **(kN)**	***P(ANN)*** **(kN)**	***P(exp)/ P(ANN)***
-	485	44.8	127	127	4.8	914.	61.7	667.5	672.1	0.993
-	330	23.4	101.	101.	2.13	914.	13.5	373.8	424.6	0.880
-	330	23.4	101.	101.	2.13	914.	144.	89.4	100.0	0.894
-	330	28.8	101.	101.	3.18	914.	60.7	301.7	221.5	1.362
-	330	28.8	101.	101.	3.18	914.	183.	129.1	185.3	0.697
-	325	41.4	76.2	76.2	3.33	812.	25.4	216.7	301.3	0.719
-	255	35	150	150	6.5	3050	64	513.1	424.1	1.210
4	440	96	120	120	5	3195	20	830.4	626.2	1.326
9	380	103	120	120	8	3195	20	1000.4	821.6	1.218
11	375	93	120	120	8	3195	20	1030.6	998.1	1.033
14	380	80	120	120	8	3195	20	1610.9	940.5	1.106
25	395	92	120	120	8	3195	20	960.3	963.2	0.997
ER4-C-4-	260	41	216	216	4.37	647.	10	1037.7	1079.	0.962
ER4-C-8-	260	80.1	216	216	4.37	647.	10	1456.0	1712.	0.850
ER6-C-2-	615	25.4	210	210	6.35	630	6	2425.7	2475.	0.980
ER6-C-4-	615	41	210	210	6.35	630	10	2108.4	2086.	1.011
ER6-C-8-	615	80.1	210	210	6.35	630	6	3410.5	3492.	0.977
ER8-C-4-	835	40.4	174	174	6.48	365	9.4	1864.6	1946.	0.958
H-4-2	340	23.1	195	130	2.54	780	14	740.0	805.9	0.918
H-5-2	340	23.1	195	130	2.54	780	31	514.0	550.9	0.933
H-7-1	340	23.1	195	130	2.54	2340	14	525.1	631.9	0.831
HSS4	750	30	110	110	5	3000	30	1281.2	1130.	1.134
C12-1	450	96	125	125	3.2	1500	20.6	1129.9	1110.	1.018
Average										1.000

Table (3) Values of parameters used in the proposed ANN model.

Number of input layer neurons	Number of hidden layer	Number of first hidden layer neurons	Number of second hidden layer neurons	Number of output layer neurons	Error after learning	Learnin g cycle
7	2	6	5	1	0.0034	20000

Table (4) Comparison between actual (experimental) and predicted values.

Column designation	***P(exp)*** **(kN)**	***P(ANN)*** **(kN)**	***P(AISC)*** **(kN)**	***P(Eurocode)*** **(kN)**
-	667.5	672.1	717.7	695.3
-	373.8	424.6	342.9	342.9
-	89.4	100.0	100.4	70.4
-	301.7	221.5	245.3	223.5
-	129.1	185.3	111.3	78.7
-	216.7	301.3	249.1	246.3
-	513.1	424.1	649.5	557.7
4	830.4	626.2	912.5	703.7
9	1000.4	821.6	1087.4	862.4
11	1030.6	998.1	1051.6	844.8
14	1610.9	940.5	1061.6	839.0
25	960.3	963.2	1067.0	857.4
ER4-C-4-10	1037.7	1079.0	2207.9	2413.3
ER4-C-8-10	1456.0	1712.5	3466.7	3935.1
ER6-C-2-6	2425.7	2475.9	3731.8	3850.3
ER6-C-4-10	2108.4	2086.0	4054.6	4216.8
ER6-C-8-6	3410.5	3492.0	5413.5	5880.2
ER8-C-4-04	1864.6	1946.5	3884.6	4053.5
H-4-2	740.0	805.9	822.2	870.6
H-5-2	514.0	550.9	659.0	685.3
H-7-1	525.1	631.9	729.3	729.3
HSS4	1281.2	1130.2	653.7	469.3
C12-1	1129.9	1110.4	1202.0	1202.0
VAF		0.994	0.808	0.767
MAPE		0.089	0.812	0.943

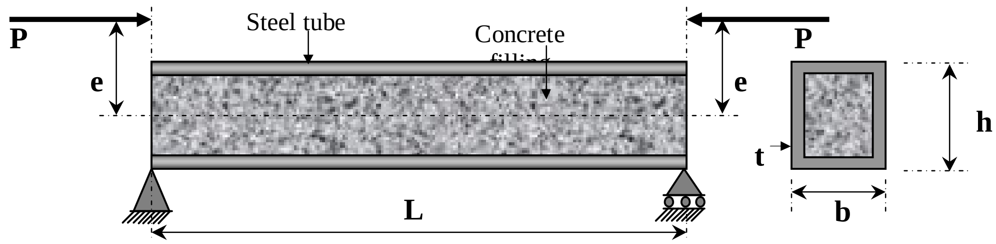

Figure (1) RCFST beam-column under eccentric axial loads.

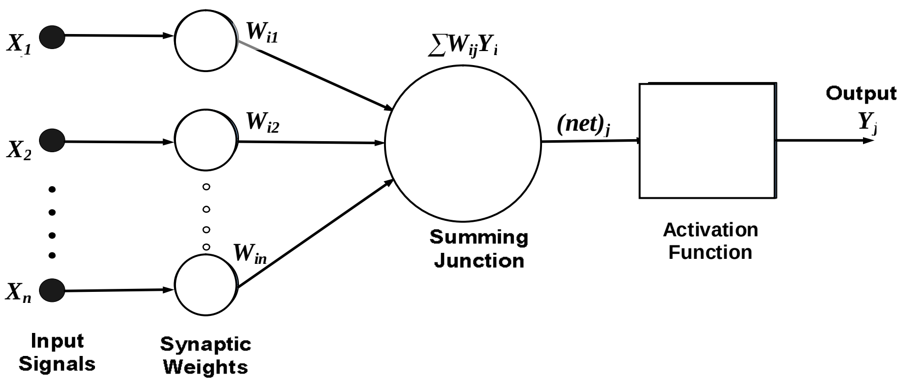

Figure (2) A simple neuron model.

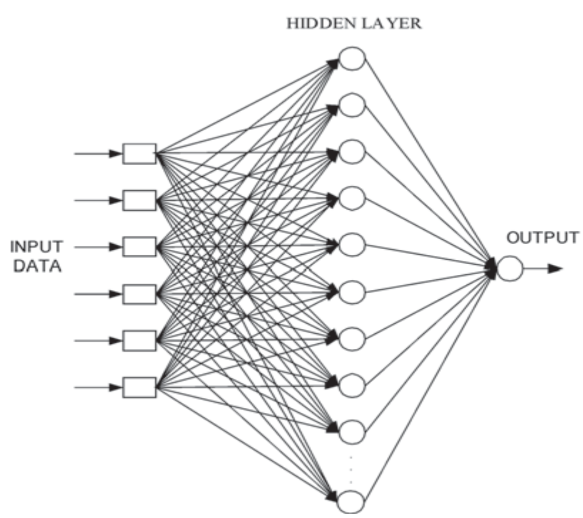

Figure (3) Multi-layer feed-forward Neural Network with a single hidden layer.

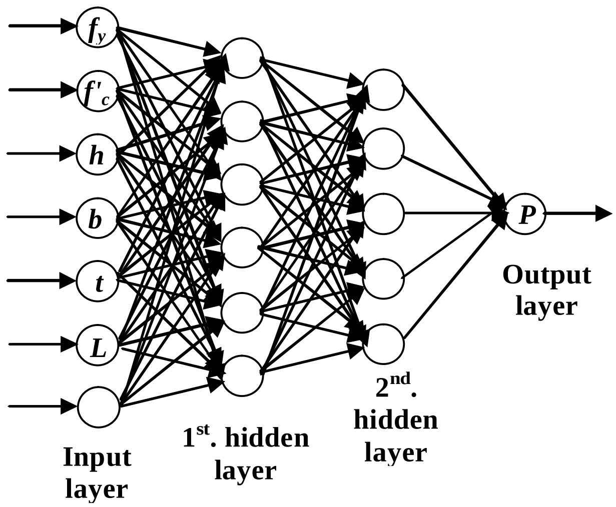

Figure (4) Architecture of proposed ANN.

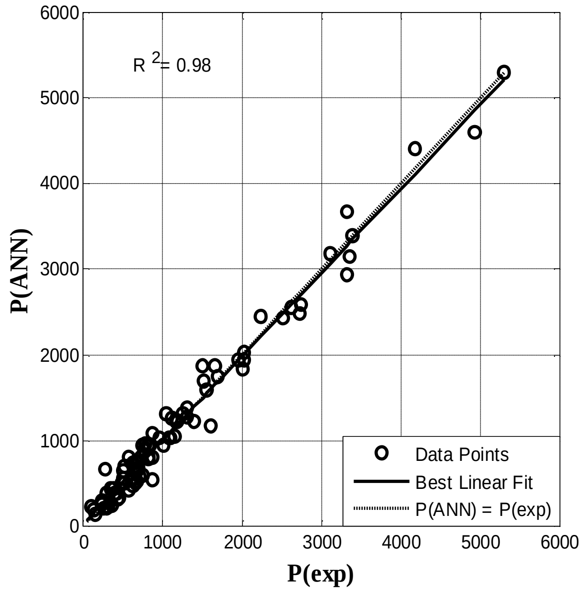

Figure (5) Regression analysis between predicted and actual values for training data.

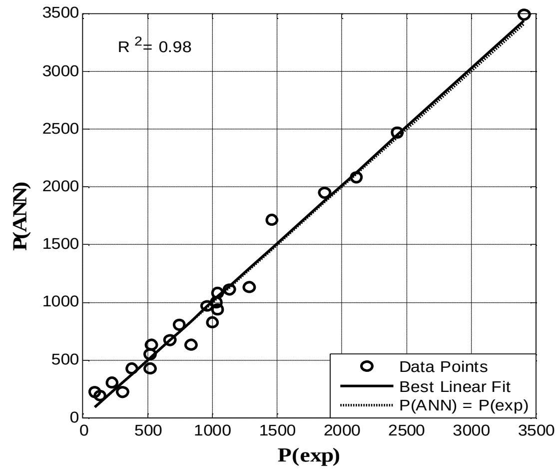

Figure (6) Regression analysis between predicted and actual values for testing data.

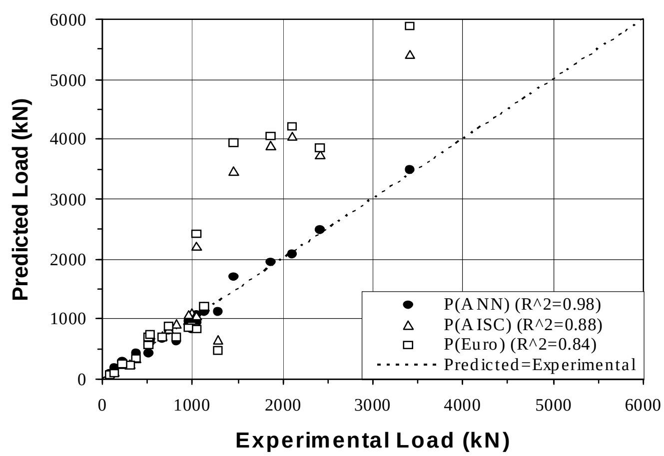

Figure (7) Regression analysis between predicted and actual values.