ASIAN JOURNAL OF CIVIL ENGINEERING (BUILDING AND HOUSING) VOL. 2, NO. 1, (2001)
PAGES 11-29

PREDICTION OF MOMENT-ROTATION CHARACTERISTIC FOR SADDLE-LIKE CONNECTIONS USING FEM AND BP NEURAL NETWORKS

A. Kaveh[a] *, D. Fazel-Dehkordi[b] and H. Servati[a]
[a] Iran University of Science and Technology, Narmak, Tehran-16, Iran
[b]Mazandaran University of Science and Technology, Babol, Iran

ABSTRACT

Finite element models are suggested for the analysis of a special type of semi-rigid connections of frame structures, generally known as saddle-like connections. The performance of these models is verified through the available experimental results.

Non-linear finite element analysis is performed for a total number of 138 connections, and M-ϕ diagrams are obtained. Each diagram is represented by a multi-linear relationship through seven points.

A neural network is trained for the prediction of M-ϕ diagrams for saddle-like connections. The results obtained by the trained backpropagation neural network are found to be quite satisfactory compared to those of the finite element analysis. The computational time for such a prediction is of the order of one second compared to approximately 12 hours for a complete non-linear finite element analysis.

Keywords: neural network, backpropagation, saddle-like connection, steel structures, semi-rigid joint, finite element, moment-rotation.

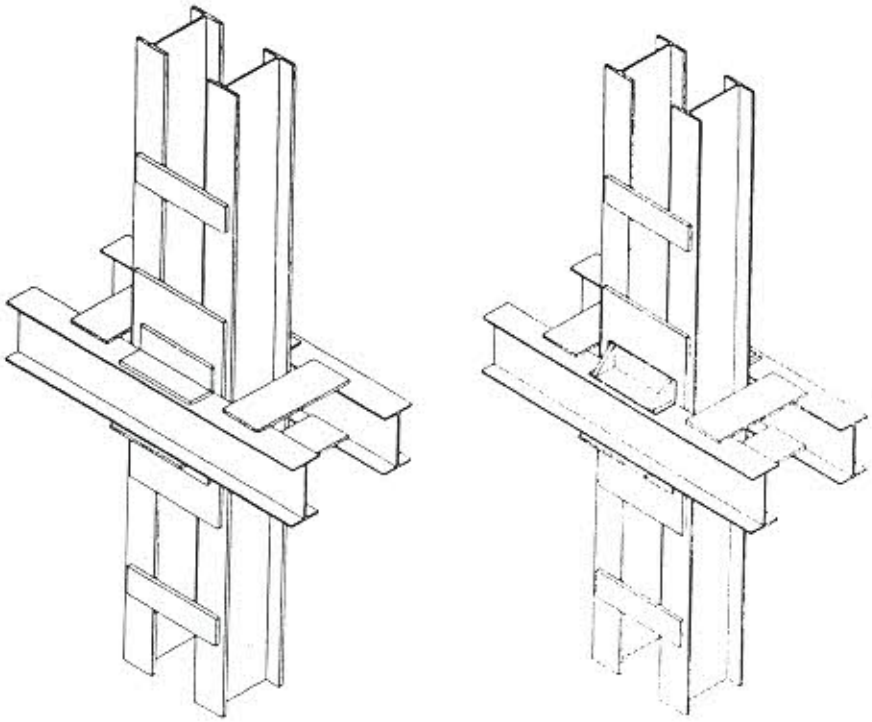
1. INTRODUCTION

Steel frames for buildings are generally designed on the basis that beam-to-column connections being either pinned or rigid. The actual behaviors of the connections are usually somewhere between these two extremes. These connections are known as *semi-rigid joints*. This behavior can be represented by a moment-rotation diagram, relating the moment M transmitted by the connection to the relative rotation ϕ between the beam end and the connected column.

There is a special connection known as *saddle-like* beam-to-column connection that is extremely easy for construction. This simplicity makes the designers to use them extensively for low-rise buildings, e.g. in Iran 90% of steel structures are made of saddle-like connections. In this type of connection first seat angles (stiffened or non-stiffened) are welded to the flat faces of the columns at story level in flat position. Then the columns are raised to vertical position and a pair of parallel and continuous beams are positioned on the seats. The final stage consists

* E-mail address of the corresponding author: alikaveh@iust.ac.ir

of putting the top angles and completing the welds. Two typical saddle-like connections with and without stiffeners are shown in Figure 1.



(a) Connection without stiffener (b) Connection with stiffener

Figure 1 Typical saddle-like connections

Some of the advantages of the saddle-like connections are summarized in the following:

- Using two beams of low depth in place of a single beam of high depth.
- Taking the advantages of continuous beam properties such as low positive bending moment compared to simply supported span, and reduction of the negative bending moments compared to rigidly supported span.
- Reduction of the deflections.
- Simplicity and economy of the erection process.

In this article finite element models are suggested for the analysis of saddle-like connections. The performance of these models is verified through five available experimental results. Non-linear finite element analysis is performed for a total number of 143 connections and M-ϕ diagrams are obtained. Each diagram is represented by a multi-linear relationship through seven points. A backpropagation (BP) neural network is trained for the prediction of M-ϕ diagrams using 820 pairs for training and 128 pairs for testing. The computational time for such a prediction is less than one second compared to approximately 12 hours for a complete non-linear FE analysis.

2. MOMENT–ROTATION CHARACTERISTIC OF A CONNECTION

According to AISC, connections are divided into three categories "simple", "rigid" and "semi-rigid" connections, with respect to their potential to transfer end moments of beams to the columns. Simple connections do not transfer the end moments and allow the beam to rotate freely with respect to the column. In practice, if the change of angle between "beam-to-column" is equal to about 80% of simple beam end rotation, the connection is considered as a simple connection.

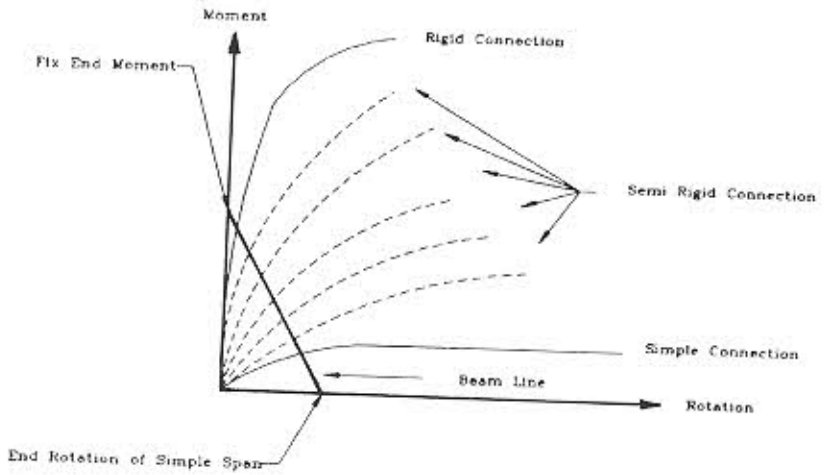


Figure 2 Moment–rotation characteristic of different types of connections

Rigid connections, however, provide integrity between beams and columns; i.e. they allow full transference of end moments by keeping the original angle between beams and columns. A semi rigid connection has behavior between simple and rigid connections. Moment–rotation characteristics (M–ϕ diagram) of simple and rigid connections are shown in Figure 2.

3. FINITE ELEMENT MODEL OF A CONNECTION

Many parameters are involved in the behavior of a saddle-like connection and only those having the most dominant effects in the shape of the M-ϕ diagram of the connection are considered in this study. The use of saddle-like connections is normally limited to 5 to 6 story buildings. Thus the sizes of the double beams are limited to those between IPE12 and IPE24 sections.

For simplicity of welding, the size of the lower angles should be chosen larger than the width of flange of the beam. The size of the top angle should be less than the width of the beam, and for both cases the distance between web of the beam and edge of the angles should not exceed

2cm. This imposes limitations on the size of the top and bottom angles. Therefore, the size of the bottom angle (N_b) and the top angle (N_u) are confined as follows:

$$b_b + D < N_b < b_b + D + 2$$
$$b_b - D - 2 < N_u < b_b - D$$

where b_b is the width of the beam and D is the dimension of the weld.

The lengths of the angles are nearly the same as the center-to-center distance of the columns. Thus, the length for angles is taken as:

$$L_b = L_u = b_c + d_c,$$

where b_c is the width of the web of the column and d_c is the free distance between the two sections of the column.

For finite element models, the length of the columns is taken as L=3 meters. The maximum dimension of the welds is chosen as the largest thickness of the angles and the minimum dimension is taken as the least thickness of the angles.

In order to design the model of a connection, the columns are cut at story levels, and beams are cut at a distance of 50cm from the connection, as shown in Figure 3. The bending moment M affecting the beams at their centers are applied as compressive and tensile forces at the two flanges of the beam, i.e.

$$M = \tfrac{1}{2} d_b \sigma A .$$

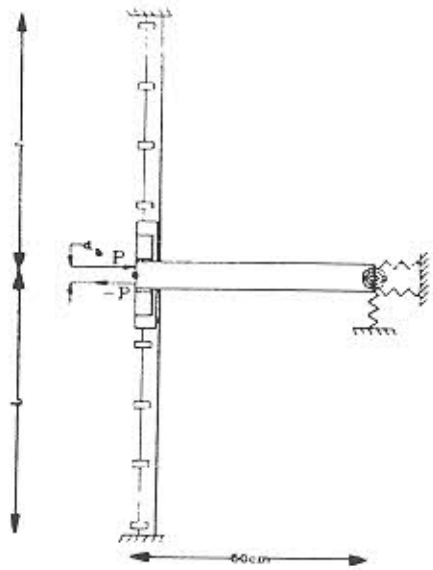


Figure 3 the model of a saddle-like connection

where d_b is the center-to-center distance of tensile and compressive forces, σ is the applied stress, and A is the cross-sectional area of the beam. Selecting an arbitrary stress level, in the process of the analysis, secures the plastic limit to be attained.

An 8-node solid element with three degrees of freedom at each node is chosen for the finite element analysis and the three displacement components at the ends of columns are considered to be zero. The axial deformations of the beams are also neglected due to high axial stiffness of the beams. Anti-symmetry is used and a hinge is considered at center point of the beam with its displacements at x and z directories taken as zero.

A total number of 138 connections are selected and studied. The free distance between two columns, d_c, is taken as 15cm when two profiles are not connected, and as 0 when connected. The ratio "ratmt" of the thickness of the selected flanges to the maximum thickness of the flange, which can be selected from the table of profiles considering the width of the beam, is denoted symbolically by 1. For minimum thickness of the flange, this ratio is designated by 0. The inclusion and the exclusion of stiffener are indicated by 1 and 0, respectively. Some typical data are shown in *Table I*.

Table 1 Typical data for connections

sample No.	size of the beam	size of the column	d_c (cm)	ratmt	stiffener
1	IPE12	IPE12	0.0	0	0
5	IPE 12	IPE 12	0.0	0	1
10	IPE 12	IPE 18	0.0	1	0
15	IPE 12	IPE 18	0.0	1	1
20	IPE 12	IPE 24	15.0	0	0
25	IPE 14	IPE 12	0.0	0	0
30	IPE 14	IPE 12	15.0	0	1
35	IPE 14	IPE 18	15.0	1	0
40	IPE 14	IPE 18	15.0	1	1
45	IPE 14	IPE 24	0.0	0	1
50	IPE 16	IPE 12	0.0	1	0
55	IPE 16	IPE 12	0.0	1	1
60	IPE 16	IPE 18	15.0	0	0
65	IPE 16	IPE 24	0.0	0	0
70	IPE 16	IPE 24	15.0	0	1
75	IPE 18	IPE 12	15.0	1	0
80	IPE 18	IPE 12	15.0	1	1
85	IPE 18	IPE 18	0.0	0	1
90	IPE 18	IPE 24	0.0	1	0
95	IPE 18	IPE 24	0.0	1	1
100	IPE 24	IPE 12	15.0	0	0
105	IPE 24	IPE 18	0.0	0	0
110	IPE 24	IPE 18	15.0	0	1
115	IPE 24	IPE 24	15.0	1	0
120	IPE 24	IPE 24	15.0	1	1

In order to generate the data for a saddle-like connection for its finite element analysis using the software NISA II, reference [1], a program is developed to employ the Display III of NISA II, to create finite element meshes for the models and apply the boundary conditions. The elements being used consist of 8-node solids and 6-node solids, both types having three degrees of freedom per node.

Modulus of elasticity E is taken as constant in x, y and z directions, i.e. $E_x = E_y = E_z = 2.1\times10^4$ kN/cm^2 and $\nu_{xy} = \nu_{xz} = \nu_{yz} = 0.3$. The stress-strain diagram is taken as bi-linear elastic linear hardening with different slopes. The plastic parameters are calculated as:

$$\beta = 0.5 \quad \text{hardening} = \frac{E_t \times E}{E_t - E} = 486\,\text{kN/cm}^2 \text{ and } \quad f_y = 24\text{kN/cm}^2,$$

where E is the slope of the elastic range and $E_t = \beta E$ is the slope of the strain hardening range corresponding to the slope of 2.5% in the strain hardening diagram. For comparison between the produced stress with the yield stress, Von-Mises criteria is employed. Modified Newton-Raphson is used for the non-linear analysis.

The magnitude of the oscillation for the displacements, energy and applied force P is taken as 0.001, i.e. for convergence

$$\frac{\delta_{i+1} - \delta_i}{\delta_i} < 0.001$$

is used, where δ is the displacement, energy or force. The starting time is taken as 0 and the end is considered as 1, and 20 equal steps are used for non-linear analysis.

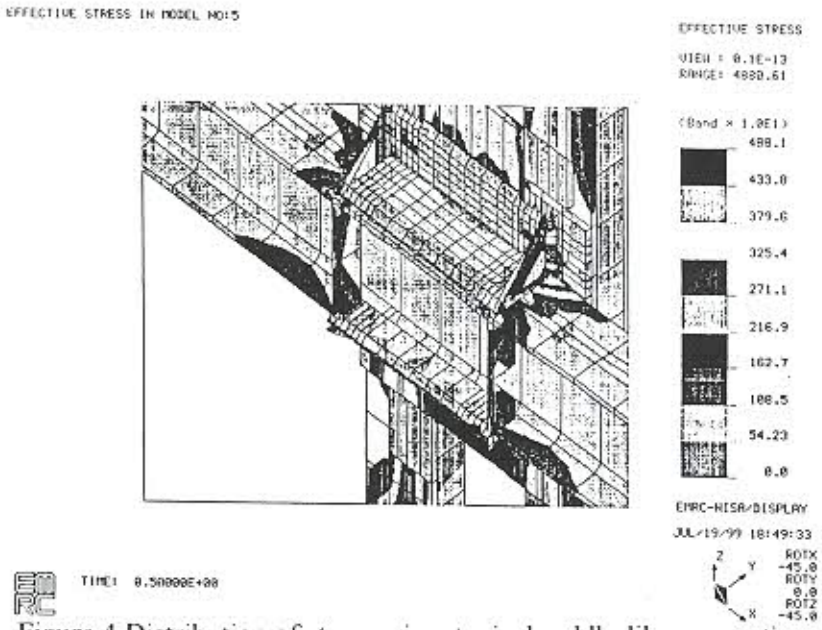
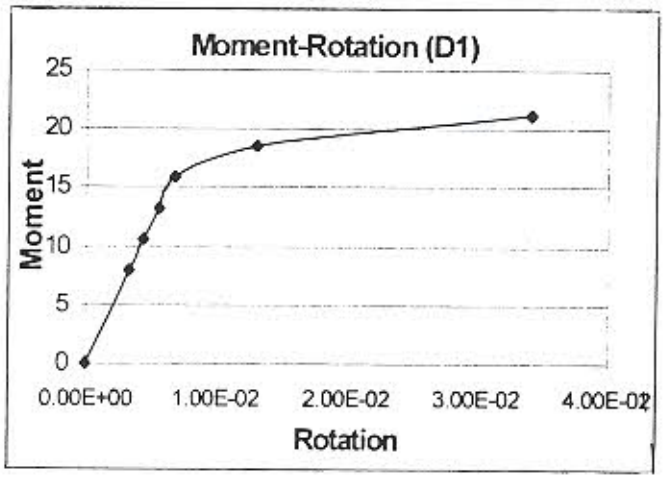
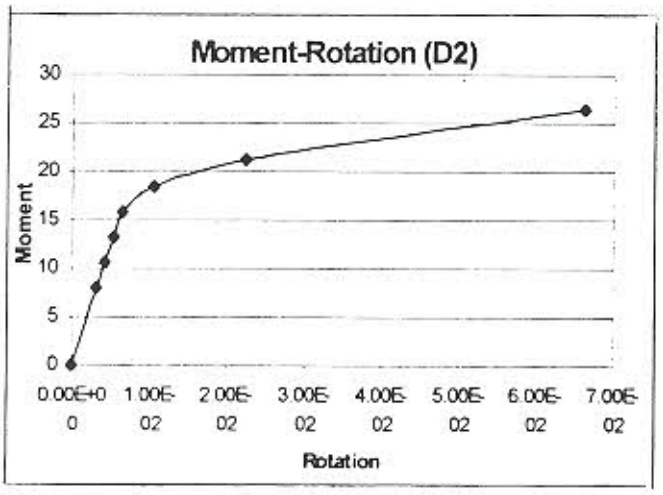


Figure 4 Distribution of stresses in a typical saddle-like connection

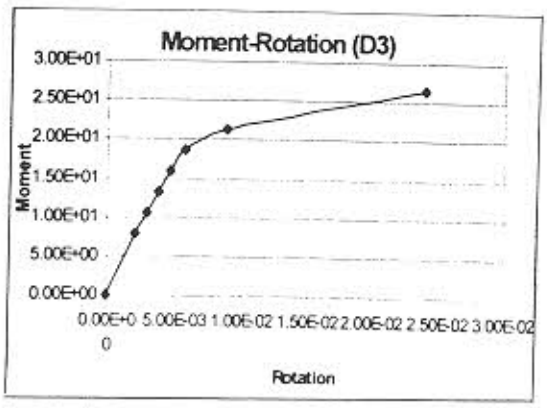
Stress analysis on a connection is performed and the stresses for a typical connection are illustrated in Figure 4. The selected models contain 1400-1500 finite elements having 3000 to 3200 merged nodes. Typical M-ϕ diagrams for six samples, idealized by 7 segments, are shown in Figure 5. The computational time for a complete non-linear finite element analysis has been approximately 12 hours on a PC.



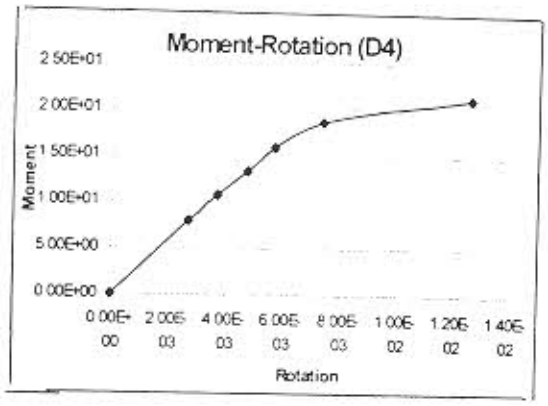
(a)



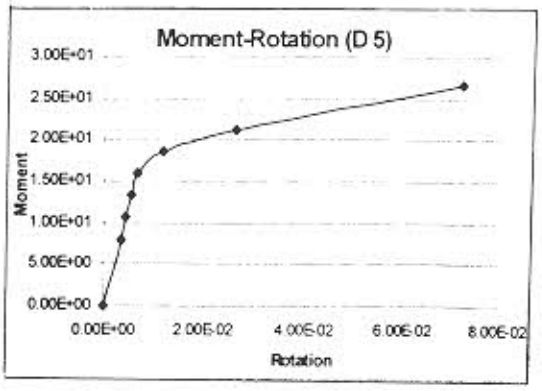
(b)



(c)



(d)



(e)

Figure 5 M-ϕ diagrams for typical saddle-like connections

4. EXPERIMENTAL RESULTS

For each experimental sample, two IPE14 sections are used as a beam and two IP16 sections covered by PL8 plates are employed as columns. The top and bottom angles are taken as L6 and L8, respectively. Two additional PL8 plates are welded to the beams in the neighbourhood of the joint to provide a box-type behavior, Figure 6.

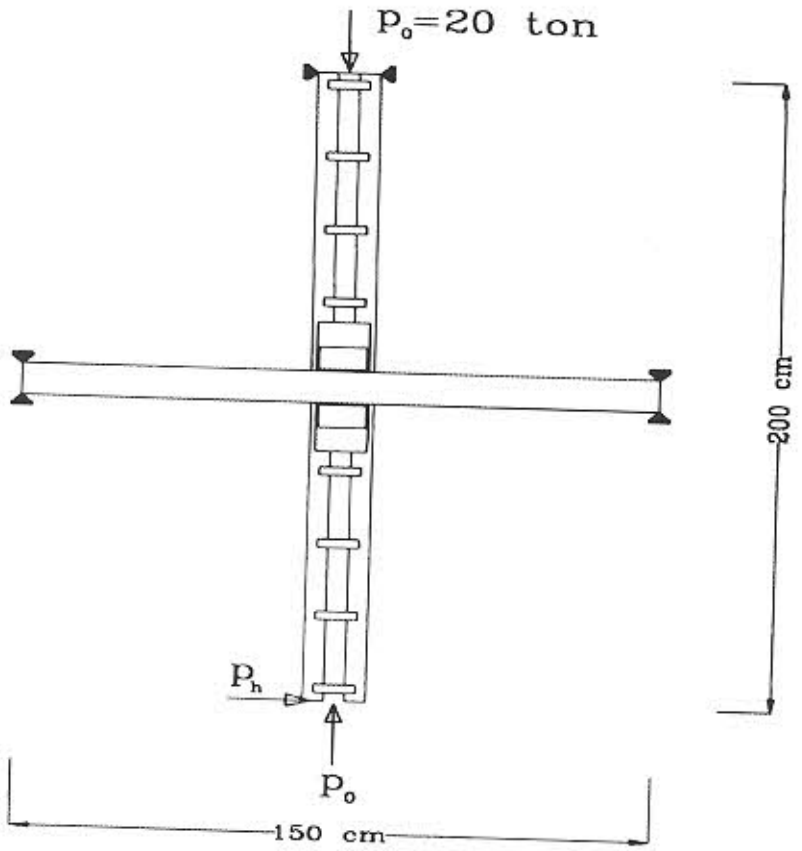
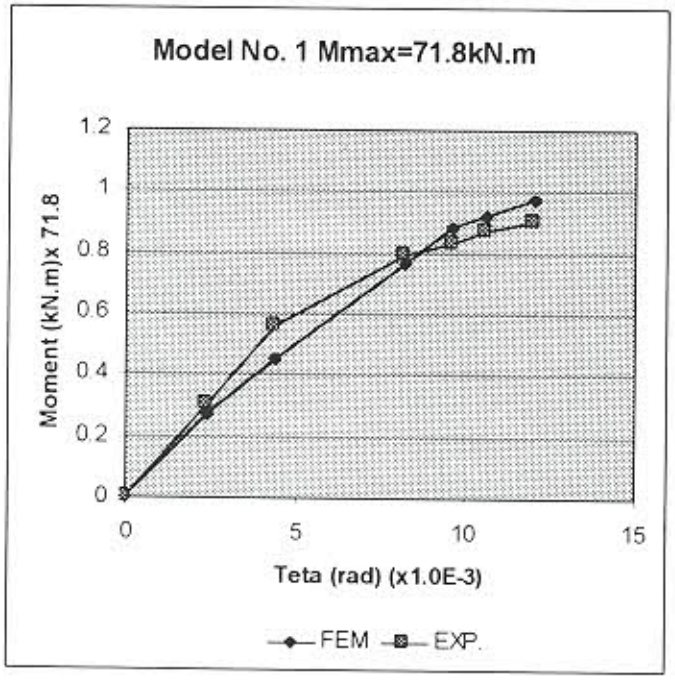


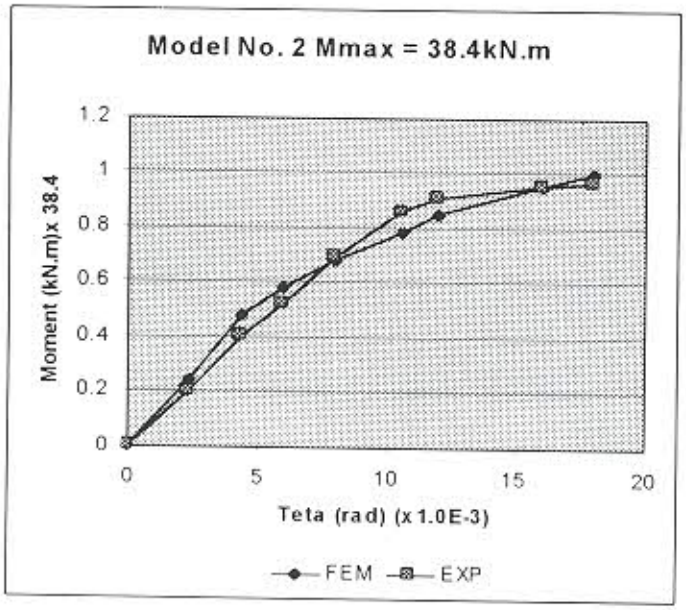
Figure 6 A typical experimental specimen

5. COMPARISON BETWEEN THE EXPERIMENTAL AND THEORETICAL RESULTS

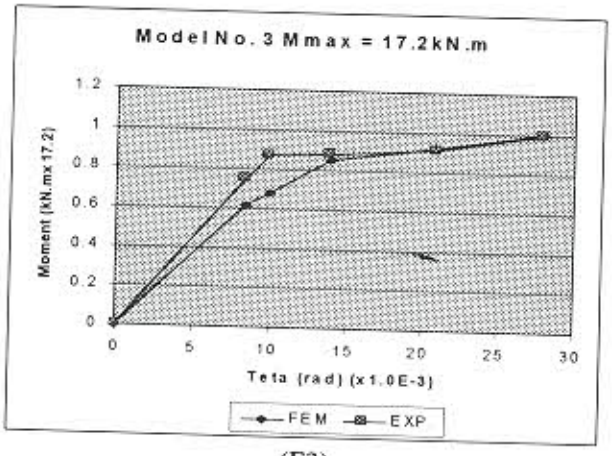
The specimens presented in *Table II* and the results obtained from the finite element analysis are compared in Figure 7. A satisfactory correlation can be observed for the M-ϕ diagrams, verifying the suitability of the finite element analysis being performed.


Model No. 1 Mmax=71.8kN.m
Moment (kN.m)x 71.8
1.2
1
0.8
0.6
0.4
0.2
0
0
5
10
15
Teta (rad) (x1.0E-3)
FEM
EXP.

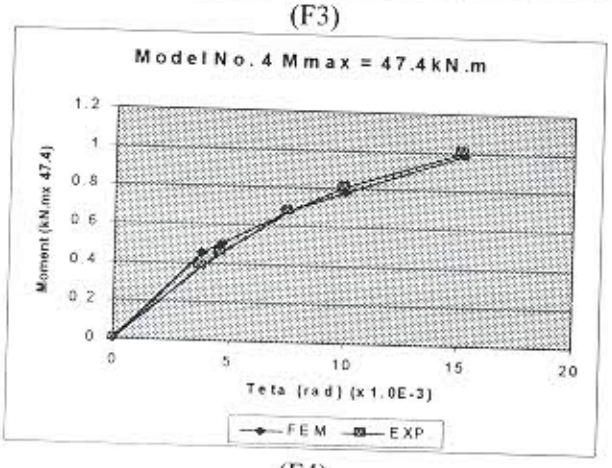
(F1)


Model No. 2 Mmax = 38.4kN.m
Moment (kN.m)x 38.4
1.2
1
0.8
0.6
0.4
0.2
0
0
5
10
15
20
Teta (rad) (x1.0E-3)
FEM
EXP

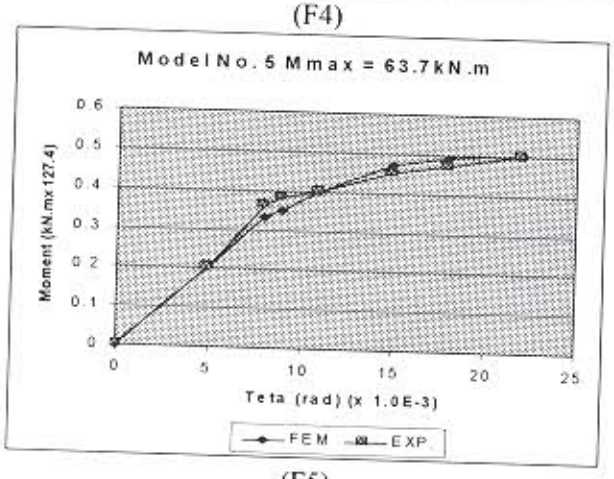
(F2)



(F3)



(F4)



(F5)

Figure 7 Comparison between the experimental and theoretical results

Table 2 Details of the specimens

specimen number	Length of the angles cm	stiffener	maximum bending capacity
F1	20	Top and bottom plates and two angle stiffeners	71.8
F2	20	Top and bottom plates and two angle stiffeners	38.4
F3	10	Top and bottom plates and two angle stiffeners	17.2
F4	20	Stiffeners for the two sides of the beam's web	47.4
F5	20	Stiffeners for the two sides of the beam's web, and two stiffeners	63.7

6. NEURAL NETWORKS, BACKPROPAGATION ALGORITHM

Recently neural networks are extensively employed in structural mechanics [2-9]. Anderson et al applied artificial neural networks to predict the minor axis steel connections [10].

In the following, some basic concepts of neural networks is summarized, however, the interested reader may refer to any textbook on this subject, e.g. reference [11].

6.1 Basic concepts

Artificial neural networks (ANNs) try to mimic the functions of the brain through mathematical and computational models. ANNs are structured from a set of artificial neurons or processing units that can be arranged on a set of layers. A processing element of an ANN is a simple device that approximates the function of a biological neuron. The topology of ANNs may be in the form of feedforward or recurrent. Learning may be supervised or unsupervised depending on the topolgy. Feedforward neural networks are usually trained using supervised training procedures.

In a processing element, the input signals are translated to an output signal by an appropriate transfer function. Weights are given to input signals according to their importance. These weights w_{st} are the strength of connection between processing elements. After a processing element receives all of its input signals, through its input path, it computes the net sum of input according to:

$$net_{pt} = \sum_{s} w_{st} o_{ps}, \tag{1}$$

where w_{st} is the connecting weight from the source "s" layer to the target "t" layer and o_{ps} is the output produced from pattern p as result of the input $o_{p(s-1)}$.

The net sum is applied to an activation function. The most commonly used activation function is the sigmoid function defined as

$$o_{pt} = \frac{1}{1 + e^{-\frac{(net_{pt} + bias)}{\theta}}} \tag{2}$$

where bias serves as a fictitious weight of a unit value which always fires (o_{pt} =1). The effect of a positive bias is to shift the activation function to the right along the horizontal axis, and θ changes the shape of the sigmoid [6-9]. This function is bounded within the range (0,1) and has a continuous first derivative. The output of a unit is obtained by passing the value of the net-sum through the activation function as

$$o_{pt} = f_t(net_{pt}). \tag{3}$$

Other functions are also used as activation functions. The most suitable one that is employed in this paper is Tangent Hyperbolic function.

6.2 The backpropagation algorithm

The backpropagation learning method was discovered by Parker [13]. This method was popularized and developed into a workable process by Rumelhart et al [14] and generalized by Werbos [15]. Numerous applications have been found for backpropagation since 1986.

The method of training adopted in this paper is the backpropagation. This approach is one of the most successfully and widely used algorithm among artificial neural networks. In backpropagation, learning is carried out when a set of training patterns is propagated through a network consisting of an input layer, one or more hidden layers and an output layer. Each layer has its corresponding units and weight connections.

The learning rule associated with the backpropagation method is known as *generalized delta rule* reference [15]. According to this rule, for changing the weights in the network, for a given pattern p, the following relation may be used:

$$\Delta w_{st}(n) = \eta \delta_{pt} o_{ps} + \alpha w_{st}(n-1), \tag{4}$$

where n is the input-output presentation number and η is the *learning rate* and α is the *momentum*. The momentum, α, is a fraction which is multiplied by the previous weight change and added to the current weight change; α should be positive and less than unity. The two terms are not independent, and should be chosen and modified as a pair. In general, dynamic adjustment of both parameters can accelerate convergence, however, values of either η or α that are too large, may lead to instability in the training process [11-12].

At the output layer, the computed outputs are subtracted from the desired (target) output and the squared sum of the output difference is a measure of the error. This measure indicates the level at which the network has learned the input-output data, and may also be used to determine the gradient of the learning procedure.

The sum of the squares of the error for a single pattern at the output units is measured by:

$$E_p = \frac{1}{N_{output}} \sum_{outputs}^{network} (t_p - o_p)^2 , \tag{5}$$

where E_p is the measure of the error of pattern p, t_p is the target output, o_p is the observed output at the output nodes, and N_{output} is the number of the output units.

The total sum of the error after, all patterns have been processed by the network, is given by:

$$E = \frac{1}{PN_{output}} \sum_{P} \sum_{outputs}^{network} (t_p - o_p)^2 \tag{6}$$

where P is the total number of patterns in the training set.
The delta signal for the output layer is defined by:

$$E = \frac{1}{PN_{output}} \sum_{P} \sum_{outputs}^{network} (t_p - o_p)^2 , \tag{7}$$

and for the hidden layer:

$$\delta_{pt} = f_t'(net_{pt}) \sum_k \delta_{pk} w_{kt} . \tag{8}$$

The derivative of the activation function that is used in Eq. (7) is given by:

$$f_t' = \frac{\partial o_{pt}}{\partial net_{pt}} = o_{pt}(1 - o_{pt}) , \tag{9}$$

where k represents an upper layer unit (the output layer is the uppermost and the input layer is the lowermost layer).

The change to the weights are defined by Eq. (4) and the new weights are computed as:

$$w_{st}(n+1) = w_{st}(n) + \Delta w_{st}(n) . \tag{10}$$

The errors are usually high at the beginning of the learning process. This necessitates bigger changes in weights in the early stages of the learning process. Using the backpropagation procedure, the network calculates delta signals for the output layer and hidden layers, using Eqs

(7) and (8), respectively. These deltas are employed to compute the changes for all the weight values according to Eq. (9) and the subsequent weight values using Eq. (4).

7. SOFTWARE FOR TRAINING

There are many software available for training a neural network [16-17]. In this paper Neuralworks of reference [16] are employed. Neuralworks program is found more flexible and provides additional useful information. In the process of training by BP, both sigmoid and tangent hyperbolic available functions are employed. Neuralworks is capable of mapping the input data to an arbitrary range. This program reads the input data and the information at each column is mapped into the considered domain, employing the maximum and minimum values in that column. The data mapped is then used in the training.

8. TRAINING AND TESTING OF THE NETWORKS

This network consists of six inputs comprising of the beam size, column size, d_c, ratmt, existence or non-existence of the stiffener, and the bending moment. The output consists of a single node representing the rotation of the connection. The number of training pairs has been 820, and for testing 128 pairs is used. Different topologies and activation functions are used, the details of which are illustrated in Table 3.

Table 3: Neural networks trained with different topologies

Name of the net	Num. of layers	No. of nodes in the hidden layer			Activation function	Mapping boundary for		correlation coefficient between actual and desired outputs		R	
		1	2	3		Input data	Output data	training set	testing set	training set	testing set
HK1	4	7	4	0	S	(0.1,1)	(0,1)	0.9890	0.9907	0.0245	0.0274
HK2	4	7	4	0	T	(-1,1)	(-1,1)	0.9962	0.9798	0.0290	0.0770
HK3	4	6	5	0	S	(0.1,1)	(0,1)	0.9867	0.9919	0.0269	0.0264
HK4	4	6	5	0	S	(0.05,1)	(0,1)	0.9864	0.9922	0.0273	0.0281
HK5	4	6	5	0	T	(-1,1)	(-1,1)	0.9959	0.9713	0.0303	0.0952
HK6	4	7	4	0	S	(0.1,1)	(0,1)	0.9825	0.9869	0.0311	0.0342
HK7	4	6	6	0	S	(0.1,1)	(0,1)	0.9872	0.9860	0.0265	0.0322
HK8	4	7	3	0	S	(0.1,1)	(0,1)	0.9858	0.9867	0.0277	0.0328
HK9	5	5	4	3	S	(0.1,1)	(0,1)	0.9859	0.9865	0.0278	0.0314
HK10	5	6	4	2	S	(0.1,1)	(0,1)	0.9882	0.9876	0.0253	0.0323
HK11	5	5	4	2	S	(0.1,1)	(0,1)	0.9866	0.9833	0.0270	0.0378

In Table 3 the followings information are used:

Number of nodes in input layer : 6	Number of pairs for training : 820
Number of nodes in output layer : 1	Number of pairs for testing : 128
Employed software: Neuralworks	T : Tangent Hyperbolic
Learning rule : Delta-Bar-Delta	S : Sigmoid

During the training the following points are observed:

1. The best topology for the net contains two hidden layers with 7 nodes in the first hidden layer and 4 nodes in the second hidden layer, and the layers use sigmoid as the activation function, Figure 8.
2. For most of the nets tangent hyperbolic function leads to better results then employing sigmoid function, since tangent hyperbolic function uses wider range for mapping compared to sigmoid. However, in this problem comparison of HK1, HK2, HK4 with HK5, reveals the reverse, i.e. the learning and the performance of tangent hyperbolic function is not any better than sigmoid. This is due to the non-uniform distribution of the input data between its maximum and minimum values. Using tangent hyperbolic function, the difference between the mapped input values is increased the accuracy of the net for interpolation by means of which the net is trained is decreased. It should also be mentioned that the RMS depends also on the range being used for mapping the output of the net. Wider the range becomes, the magnitude of the RMS will be larger. For this reason, in this problem considering the fact that the RMS for HK5 is larger than HK3, one can not conclude that the learning rate for HK3 is better, Figure 9. The magnitudes of the correlation coefficients illustrate clearly this fact.

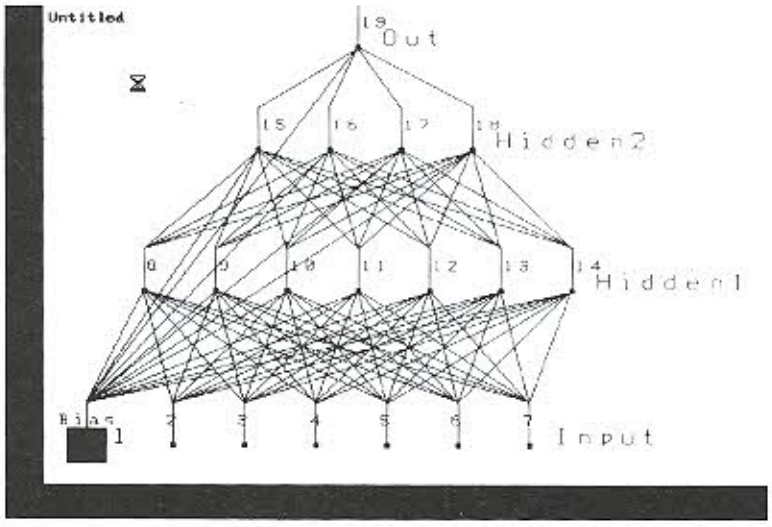


Figure 8 Topology of the selected neural network

3. Increasing the number of hidden layers does not necessarily improve the learning and performance of a net. Unreasonable increment of the number of hidden layers, increases the complexity of the net and can reduce the capability of the net in responding to the new input data. As an example HK3 and HK10 can be compared. HK3 has two hidden layers with 73 weights, while HK10 contains 3 hidden layers with the same number of weights. These nets have all the other characteristics the same, and have different topology. The results are different. HK10 is contained in a better learning group than HK3, however, in relation with responding to the input, its performance is weaker. Therefore one can conclude that increasing the number of hidden layers may increase the learning may improve, however, the capability of the net to new input may decrease.

Figure 9 RMS of the trained net

The results of testing are provided in Fig. 10, where the deflections from exact analysis and the rotations predicted by the trained neural net "HK6" are shown. Figure 11 illustrates the difference between the exact and predicted deflections.

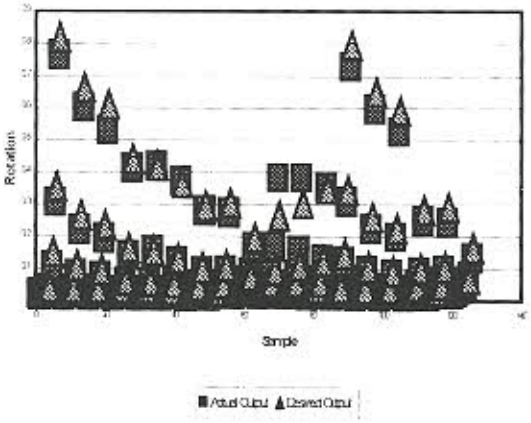


Figure 10 Test results

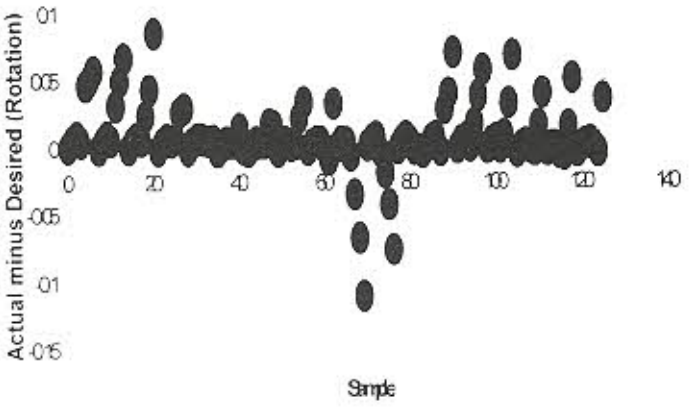


Figure 11 Difference between the exact and predicted rotations

9. DISCUSSION AND CONCLUDING REMARKS

An efficient neural network is trained for the prediction of the moment-rotation diagram for saddle-like connections. The prediction of M-ϕ curve for a typical saddle-like joint takes less than a Second compared to 12 hours of computational time required for a complete non-linear analysis of the same connection.

The neural net trained in this article can efficiently be incorporated in a non-linear analysis of frames comprising of saddle-like connections with different M-ϕ diagrams. The application can also be extended to dynamic analysis of frames with semi-rigid joints. Similar nets can be trained for joints with beams and columns formed of sections other than IPE.

Acknowledgment: The authors are grateful to Dr. A. Mazroie of Building and Housing Research Center of Iran for providing the experimental results for five samples.

REFERENCES

1. NISA II, *Engineering Mechanics Research Corporation*, Troy, Michigan, USA.
2. Berke, L. and Hajela, P. Applications of artificial neural nets in structural mechanics, *Journal of Structural Optimization*, **4**(1992) 90-98.
3. Stavroulakis, G.E. and Abdalla, K.M. A symmetric neural network classifier in mechanics, An application in semi-rigid joints. International Journal Engineering Analysis and Design, **1**(1994)270-292.
4. Jenkins, W.M. Neural network-based approximations for structural analysis,

Developments in Neural Networks and Evolutionary Computing for Civil and Structural Engineering, Edit. B.H.V. Topping, *Civil-Comp Press*, Edinburgh, 1995.

5. Jenkins, W.M. An introduction to neural computing for the structural engineering, *The Structural Engineer*, **75**(1997) 38-41.
6. Abdalla, K.M. and Stavroulakis, G. A backpropagation neural model for semi-rigid connections, Microcomp. Civil Eng., **10**(1995) 77-87.
7. Kaveh, A. and Iranmanesh, A. Comparative study of backpropagation and improved counterpropagation neural nets in structural analysis and optimization, *International Journal of Space Structures*, **13**(1998) 177-185.
8. Iranmanesh A. and Kaveh, A. Structural optimization by gradient-based neural networks, *International Journal for Numerical Methods in Engineering*, **46** (1999)297-311.
9. Topping, B.H.V. and Bahreininejad, A. *Neural Computing for Structural Mechanics*, Saxe Coburg, UK, 1997.
10. D. Anderson, E.L. Hines, S.J. Arthur and E.L. Eiap, Application of artificial neural networks to prediction of minor axis steel connections, *Computers and Structures*, **63** (1997)685-692.
11. Patterson, Dan W. *Artificial Neural Networks, Theory and Applications*, Prentice Hall, Singapore, 1996
12. Topping, B.H.V. and Bahreininejad, A. Neural networks for learning and optimization, in *Emergent Computing Methods in Engineering Design*, Edit. D.E. Grierson and P. Hajela, NATO ASZ Series, Springer, 1996.
13. Parker, D.B. Learning logic, TR-47, Center for Computational Research in Economics and Management Science, MIT, Cambridge, M.A. 1985.
14. Rumelhart, D.E. Hinton, G.E. and Williams, R.J. Learning representations by backpropagation errors, *Nature (London)*, **323**,(1986)533-536.
15. Werbos, P.J. Generalization of backpropagation with application to a recurrent gas market model, *Neural Networks*, **1**(1988)339-356.
16. NeuralWare, *Using NeuralWorks*, NeuralWare, USA, 1993.
17. Baffes, Paul T. *NET User's Guide*, Software Technology Branch of NASA's Johnson Space Center, 1991.