Engineering Journal of the University of Qatar, Vol. 13, 2000, pp.

PRELIMINARY DESIGN OF REINFORCED CONCRETE BEAMS USING NEURAL NETWORKS

Ahmed B. Senouci
Civil Engineering Department
University of Qatar
P. O. Box 2713
Doha, Qatar

ABSTRACT

This paper presents a backpropagation neural network model for the preliminary design of rectangular concrete beams. The model, which is developed based on the strength design procedure of the American Concrete Institute (ACI), minimizes the beam total cost including the costs of concrete, steel, and shuttering. The backpropagation neural network was successful in accurately capturing the nonlinear characteristics of the strength design procedure. The network adequately learned a set of 375 examples during the training phase. A case study, where a set of 960 new cases were considered, was used to validate the network and to demonstrate the system's generalization and fault-tolerance properties. The network showed good generalization properties since it was able to predict the correct beam depth and steel area with a fair accuracy.

INTRODUCTION

The preliminary design represents the first step in the analysis/design of a structural system. With the results of the preliminary design, the members are proportioned, and the resulting dimensions are compared with those previously assumed . If necessary, the assumed section properties are modified, and the analysis/design is repeated. Since the procedure may become lengthly and laborious, it is advantageous to make the best possible estimate of member sizes, in the hope of reducing the number of analysis/design cycles. Structural design is a creative process in which the experience and knowledge of the designer are combined. Less experienced structural designers may go through many trials before a suitable solution is achieved. The development of a computerized tool to assist less-experienced structural designers in their preliminary design is definetely useful and needed.

Neural networks are computational models that have been successfull in performing complicated pattern recognition and nonlinear mapping tasks over a broad spectrum of applications in a number of fields such as engineering, finance, science, and medecine. A number of researchers have reported the use of neural networks in the area of Structural Engineering. Liu and Gan (1991) developed a neural network for the preliminary design of space-grid structures. Hajela and Berke (1991) developed a Hopfield network for the optimum design of trusses. Park and Adeli (1995) developed a neural dynamics model for the design optimization of steel structures.

Recently, Senouci and Abdul-Salam (1998) developed a backpropagation neural network for the depth prediction of reinforced concrete beams. This paper, which extends the work done by Senouci and Abdul-Salam, presents the development of a neural network system for the preliminary design of rectangular concrete beams. The system minimizes the beam total cost including the costs of concrete, steel, and shuttering. It was validated by data and criteria provided by the American Concrete Institute (ACI-318 1995). The generalization properties of the proposed system were also demonstrated using a case study.

NEURAL NETWORK FOR PRELIMINARY CONCRETE BEAM DESIGN

The total structural design process is not well suited for computer applications because it requires the use of human intelligence, past experience/knowledge, and intuition. The rule-based expert system approach, which represents an obvious choice for the computerization of the structural design process, has unfortunately failed to program the total design process (Adeli and Balasubramanyam 1988). The main drawback of the rule-based expert system approach is that it lacks the learning capability for a structural design application and is unable to generalize the situation on its own to apply the given solution to an entirely new situation. In view of these difficulties, it becomes very difficult for a design program to incorporate the needed intuition and use of past experience, which is essential in the case of a preliminary design.

Neural networks can learn through designs created by experts. The expert's knowledge, intuition, and past experience can be simulated by a neural network trained to learn the many possibilities of beam design. In other words, the neural network can be trained to learn the different design alternatives used locally. Subsequently, they are able to apply this knowledge to solve a new problem. It may also be noted that neural networks allow massive parallel processing. Therefore, they can generate a good design faster than a mathematical optimizer.

A backpropagation neural network is developed for the preliminary design of rectangular concrete beams. Backpropagation neural networks have a proven ability to model nonlinear relationships such as encountered in the design of rectangular concrete beams.

Detailed discussion on backpropagation neural networks can be found elsewhere (Simpson 1991, Hertz et. Al. 1991, Masters 1993).

NEURAL NETWORK DEVELOPMENT

Optimal Design Model

The first step in the development of a neural network system is to obtain good training and testing examples. In this paper, the training and testing examples were obtained by generating minimum cost designs of singly-reinforced rectangular beams.

The cost per unit length of a beam is given by the following equation (Chakrabarty 1992):

$$\text{Cost} = C_1 A_s + C_2 b d + C_3 d + C_4 b \qquad (1)$$

where Cost = cost per unit length of the beam (\$/m), C_1 = unit cost of tensile reinforcing steel bars (\$/kg), C_2 = unit cost of concrete (\$/m^3), C_3 = unit cost of concrete formwork along the vertical surfaces (\$/m^2), C_4 = unit cost of concrete formwork along the bottom horizontal surface (\$/m^2), A_s = area of tensile steel reinforcement (m^2), d = depth of the beam (m), and b = width of the beam (m). Table 1 summarizes the values for C_1, C_2, C_3, and C_4 obtained from local contractors and suppliers.

The variables (A_s), (d), and (b) affect the beam cost and strength. In order to generate optimum designs, different values for the beam width (b) were selected. The variables (d) and (A_s) were, then, determined so that the total cost of the beam is minimum and its strength is adequate.

Table 1. Values of C_1, C_2, C_3, and C_4 (State of Qatar)

Parameter	Unit	Value (Q.R.) (1 US$=3.65 Q.R.)
C1	kg	1.32
C2	m^3	226.00
C3	m^2	5.50
C4	m^2	5.50

The computer program BeamDesign, whose flowchart is shown in Figure 1, was written in FORTRAN to generate minimum cost designs for rectangular concrete beams (Fintel 1985 and McCormac 1993).

Network Architecture

In developing a neural network for the preliminary design of rectangular concrete beams, a backpropagation neural network with one hidden layer was used. The neural network, which has been chosen for the present research, has 19 input neurons and two output neurons. Table 2 summarizes the input and output components of the neural network. Figure 2 shows the selected backpropagation neural network.

The input components take binary values (i.e., either 0 or 1). The set of components 1-5 represents whether the beam span length is very low, low, medium, high, or very high; the set of components 6-10 represents whether the beam load intensity is very low, low, medium, high, or very high; the set of components 11-15 represents whether the beam width is equal to 200, 250, 300, 350, or 400 mm; the set of components 16-17 represents whether the concrete compressive strength is equal to 20 or 30 MPa; the set of components 18-19 represents whether the reinforcing steel yield strength is equal to 300 or 400 MPa. The components in each set are not mutually independent. In other words, if a component in a set takes the value of 1.0, all the other components in the set will take the value of 0. Table 3 summarizes the value ranges of the network input attributes.

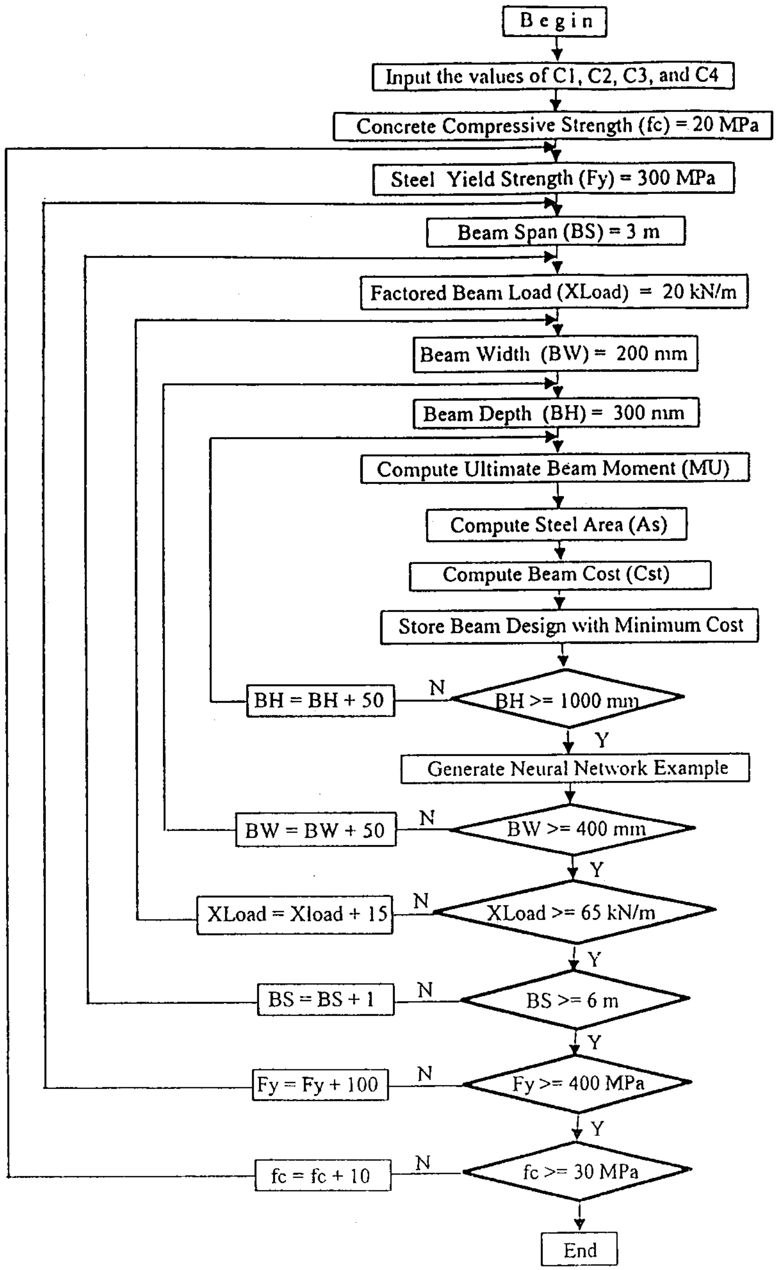

Fig. 1. Program beamdesign flowchart

Table 3. Value Range of User-Provided Information

Design Variable	Description	Value Range
Span	Very Short	Span <= 3m
Span	Short	3 < Span <= 4
Span	Medium	4 < Span <= 5
Span	Long	5 < Span <= 6
Span	Very Long	Span >= 6
Load	Very Low	Load <= 20 kN/m
Load	Low	20 < Load <= 35
Load	Medium	35 < Load <=50
Load	High	50 < Load <= 65
Load	Very High	Load >= 65
Beam Width	Width = 200 mm	
Beam Width	Width = 250 mm	
Beam Width	Width = 300 mm	
Beam Width	Width = 350 mm	
Beam Width	Width = 400 mm	
Concrete Strength	f'_c = 20 Mpa	
Concrete Strength	f'_c = 30 Mpa	
Steel Yield Strength	F_y = 300 Mpa	
Steel Yield Strength	F_y = 400 Mpa	

As an alternative scheme, a continuously valued component could be used for representing each set of factors. Though this scheme reduces the network size and the amount of computation, it is not preferred because: 1) a neural network learns faster with binary input, 2) a binary input helps in clustering the data right from the input level, and 3) the size of the network is not of concern since the prediction of the network is extremely fast. A training example describes the beam characteristics for preliminary design with 19 factors (though all of them are not independent). The pattern associator forms an arbitrary mapping that exists between the inputs and the outputs. Thus, the implicit interfactor dependencies are also learned by the network during the training process.

In contrast, the output components, which represent the beam depth and steel area, take continuous values between 0 and 1. The number of hidden neurons is chosen to be 19 on a trial-and-error basis. It is desirable to have as few hidden. neurons as possible, but too few hidden neurons, the network will not converge during the training process. We started with 5 hidden neurons and increased each time the network did not converge to a desired level.

Network Input and Output Data

The program BeamDesign was used to generate the training and testing examples. Table 4 summarizes the values of the design variables used to generate the example set. Each example consists of 19 input components and two output components. The beam depth and steel area have to be scaled to values in the range 0-1. This is necessary because the sigmoid transfer function modulates the output values between 0 and 1.

Table 4. Design Variable Values for Training and Testing Examples

Design Variables	Values
Span	3.0, 4.0, 5.0, 6.0, 7.0 m
Load	20, 35, 50, 65, 80 kN/m
Beam Width	200, 250, 300, 350, 400 mm
Concrete Strength	20, 30 Mpa
Steel Yield Strength	300, 400 Mpa

Network Training and Testing

The training and testing of the neural network were performed using the neural network simulator, Brainmaker (Brainmaker 1993). The network was trained with 375 examples. The optimum number of hidden neurons for the network was determined on a trial-and-error basis. Figures 3 through 6 show the training convergence of the network with 5, 10, 15, and 19 hidden neurons, respectively. The figures show that the network with 19 hidden neurons yielded the lowest average training results. It should also be noted that increasing the number of hidden neurons above 19 did not improve the network training convergence.

A set of 125 examples was used to test the network during training. The average testing errors for the network with 5, 10, 15, and 19 hidden neurons were 0.0331, 0.0315, 0.0253, and 0.0180, respectively. Nineteen hidden neurons were selected for the network because the average network training and testing errors were the smallest (0.0164 and 0.0180).

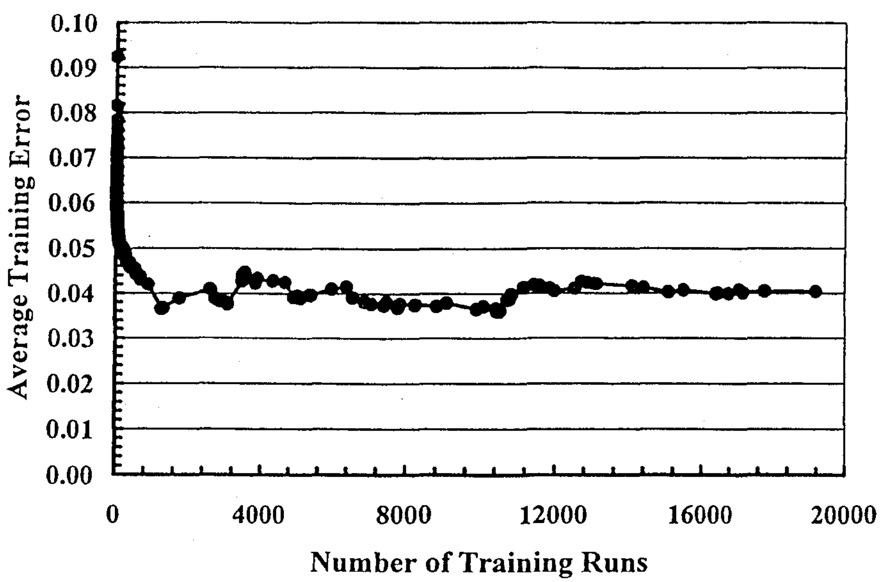

Fig. 3. Network training convergence (5 hidden neurons)

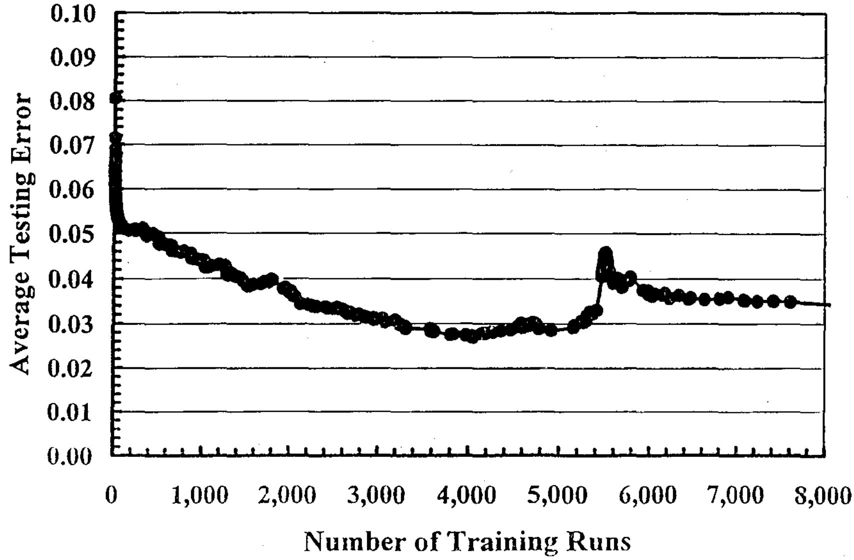

Fig. 4. Network training convergence (10 hidden neurons)

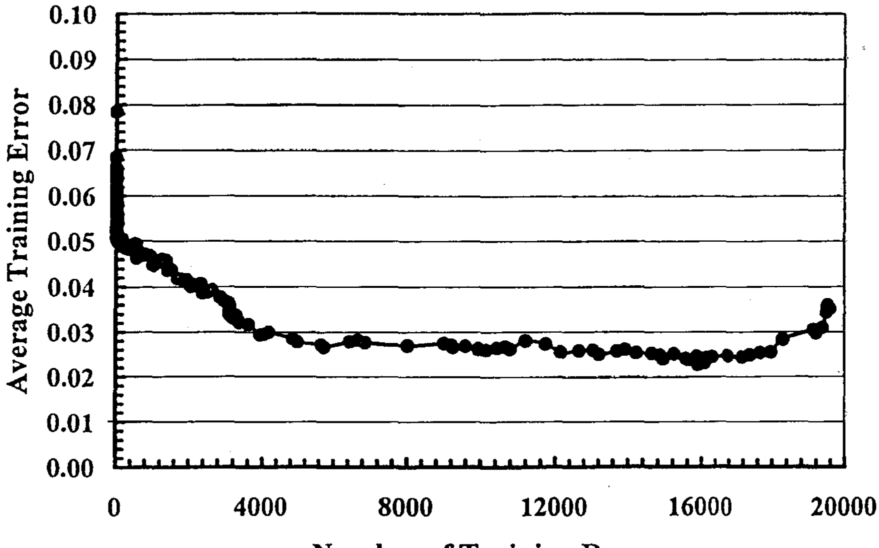

Fig. 5. Network training convergence (15 hidden neurons)

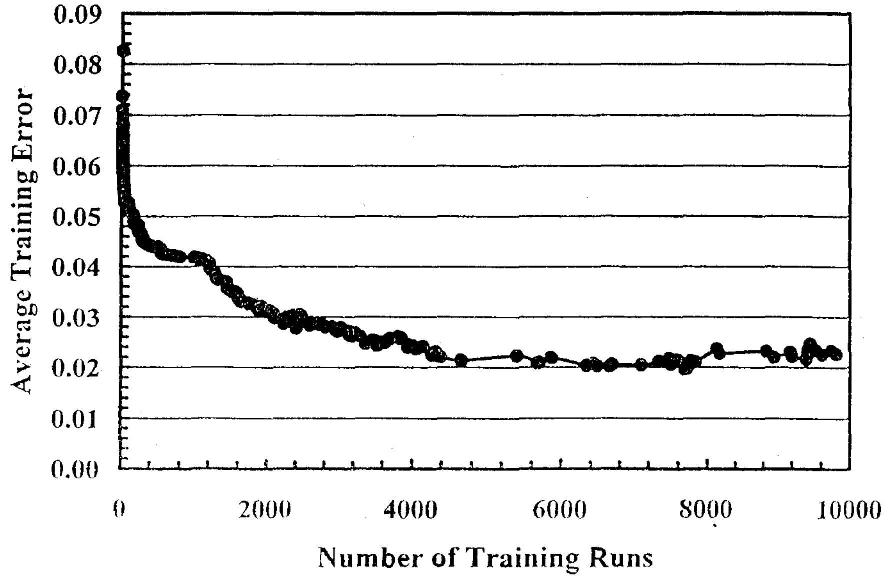

Fig. 6. Network training convergence (19 hidden neurons)

GENERALIZATION PROPERTIES

A trained network should be capable of generalizing the governing rules to accurately determine an output from new (not previously introduced) inputs. In order to verify the generalization properties of the developed system and to evaluate its performance, 960 new examples were used. The beam design values of the variables used to generate the case study examples cover a reasonable domain of reinforced-concrete beam spans and lengths, concrete compressive strength, reinforcing steel yield strength, and beam width (Table 5). The selected beam span lengths and loads, reinforcing steel yield strength, and concrete compressive were different from the ones used to generate the training and testing examples.

Table 5. Design Variable Values for Case Study Examples

Design Variables	Values
Span	3.5, 4.5, 5.5, 6.5 m
Load	70, 90, 100, 110 kN/m
Beam Width	200, 250, 300, 350, 400 mm
Concrete Strength	20, 25, 30, 35 MPa
Steel Yield Strength	300, 350, 400 MPa

The beam depth and steel area obtained using the optimizer (i.e., program BeamDesign) were compared with those predicted by the network. Table 6 show a sample of the case study results. The table shows the values of the beam depths and steel areas obtained using the optimizer and the network, respectively. The last two columns of the table represent the percentage error in the beam depth and steel area predictions of the network. The values predicted by the network are not in exact agreement with the values predicted by the optimizer. Table 7 summarizes the mean errors and standard deviations of the case study results. Figure 7 shows the histogram of the percent error in the beam depth and steel area predictions of the network.

Because it is proposed to use the neural network for the preliminary design, the network results can serve as a good starting values. Once the preliminary design model is finalized, the remaining part in a design program can well be handled using the conventional design programs to fine-tune the preliminary design. The present network was able to map the complicated functional relation between the input and the output parameters, which give the desired near-optimal values.

CONCLUSIONS

In this study, a neural network system for the preliminary design of rectangular concrete beams was developed. The study showed that the neural network system performed well in the design of concrete beams. The following summarizes the findings of the study.

The presented neural network offered a systematic procedure that accurately predicted the minimum cost beam design. The proposed network can be used by structural engineers to speed up the design process.

The network was able to adequately learn from the training examples and was able to capture the characteristics of the beam design problem.

The network showed a good generalization capability, and was able to predict the beam depth and steel areas for 960 new beam designs with an average absolute percent error of 4.1 and 12.3, respectively.

Similar neural networks can be developed for different sets of data related to other design environment.

Table 6. Case study result sample

Beam Span (m)	Beam Factored Load (kN/m)	Beam Width (mm)	Concrete Compressive Strength (MPa)	Steel Yield Strength (MPa)	Optimum Design Beam Depth (mm)	Optimum Design Steel Area (mm^2)	Neural Network Beam Depth (mm)	Neural Network Steel Area (mm^2)	Beam Depth Error (%)	Steel Area Error (%)
3.5	90	200	20	300	800	753	849	829	6.2	10.1
3.5	90	250	20	300	750	810	806	873	7.5	7.8
3.5	90	350	20	300	600	1050	665	1114	10.8	6.1
3.5	100	400	20	300	600	1166	646	1094	7.6	6.2
4.5	45	200	20	300	750	691	859	767	14.5	10.9
4.5	90	300	20	300	850	1194	908	1305	6.8	9.3
4.5	90	350	20	300	800	1281	863	1347	7.8	5.1
6.5	45	350	20	300	850	1349	930	1549	9.4	14.8
6.5	45	400	20	300	800	1456	916	1640	14.5	12.7
7.5	25	350	20	300	750	1237	819	1307	9.2	5.7
7.5	25	400	20	300	700	1354	769	1449	9.9	7.0
3.5	100	250	20	350	750	772	806	873	7.5	13.1
3.5	100	400	20	350	600	1000	646	1094	7.6	9.4
3.5	90	200	20	400	800	565	842	626	5.2	10.8
3.5	90	350	20	400	600	787	660	834	9.9	5.9
3.5	90	400	20	400	600	788	646	828	7.7	5.0
3.5	100	250	20	400	750	675	799	621	6.6	7.9
3.5	100	400	20	400	600	875	646	828	7.7	5.4
4.5	45	200	20	400	750	518	860	567	14.6	9.4
4.5	45	350	20	400	600	680	678	777	13	14.3
4.5	90	300	20	400	850	895	907	955	6.7	6.7
3.5	90	200	25	300	800	746	849	829	6.2	11.2
3.5	90	350	25	300	600	1038	665	1114	10.8	7.4
3.5	90	400	25	300	600	1041	646	1094	7.6	5.1
3.5	100	400	25	300	600	1154	646	1094	7.6	5.2
4.5	45	200	25	300	750	684	859	767	14.5	12.1
4.5	90	300	25	300	850	1182	908	1305	6.8	10.4
4.5	90	350	25	300	800	1269	863	1347	7.8	6.1
6.5	45	400	25	300	800	1442	916	1640	14.5	13.8
7.5	25	250	25	300	850	1030	931	1125	9.5	9.2
7.5	25	300	25	300	800	1123	875	1186	9.4	5.6
7.5	25	350	25	300	750	1225	819	1307	9.2	6.7
7.5	25	400	25	300	700	1340	769	1449	9.9	8.1
3.5	100	250	25	350	750	764	806	873	7.5	14.3
3.5	100	400	25	350	600	989	646	1094	7.6	10.6
3.5	90	200	25	400	800	559	842	626	5.2	11.9
3.5	90	350	25	400	600	779	660	834	9.9	7.0

Table 7. Case Study Error Analysis

Network Number	Beam Depth Mean Error (%)	Beam Depth Standard Deviation (%)	Steel Area Mean Error (%)	Steel Area Standard Deviation (%)
1	4.1	5.7	12.3	9.2
2	4.2	5.6	12.7	9.4

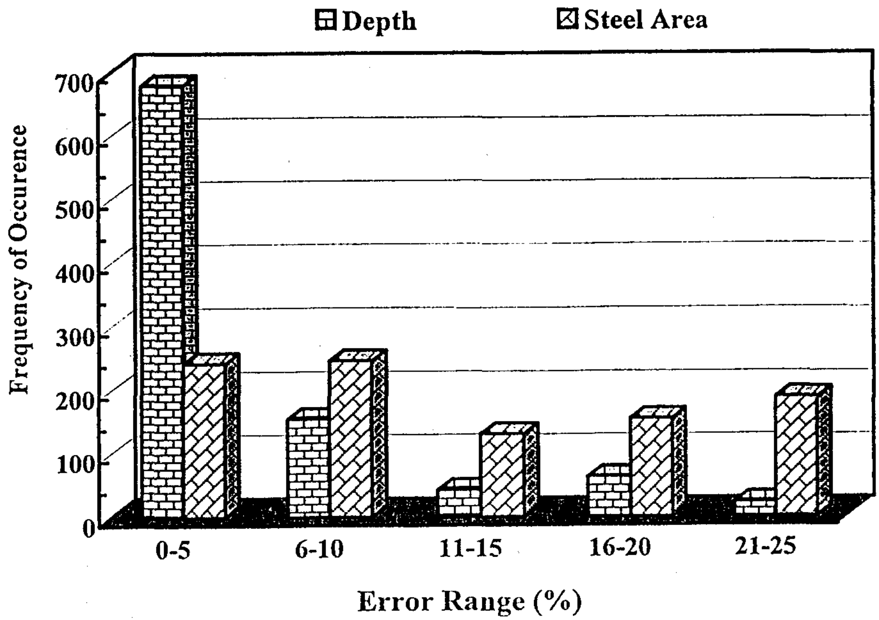

Fig. 7. Testing result error histogram

REFERENCES

1. **ACI-318 (1995).** Building code requirements for reinforced concrete. American Concrete Institute, Detroit, USA.

2. **Adeli, H. and Balasubramanyam, K., V. (1988).** " A knowledge-based system for design of bridge truss," Journal of Computing in Civil Engineering, American Society of Civil Engineers, Vol. 2, No. 1, pp. 1-20.

3. **Brainmaker (1993).** Neural Network Simulator. California Scientific Software, Nevada City, CA 95959, USA.

4. **Chakrabarty, B. K. (1992).** " A model for optimal design of reinforced concrete beam," Journal of Structural Engineering, American Society of Civil Engineers, Vol. 118, No. 11, pp. 3238-3242.

5. **Hajela, P. and Berke, L. (1992).** "Neural networks in structural analysis and design: An overview." Computing Systems in Engineering, 3 (1-4), pp. 525-538.

6. **Hertz, J., Krogh, A., and Palmer R. G. (1991).** Introduction to the theory of neural computation. Addison-Wesley Publishing Company, California 94605, USA.

7. **Fintel, M. (1985).** Handbook of Concrete Engineering. 2nd Edition, Van Nostrand Reinhold Company, New York, New York 10020, USA.

8. **Liu, X. and Gan, M. (1991).** "A preliminary design expert system (SPRED-1) based on neural networks." Artificial Intelligence in Design, Butterfield-Heinman Ltd., Oxford, England, pp. 785-800.

9. **McCormac, J. C. (1993).** Design of Reinforced Concrete. Harper-Collins College Publishers, New York, NY 10022.

10. **Masters, T. (1993).** Practical neural network recipies in C++ . Academic Press, San Diego, California 92101-4311, USA.

11. **Park, H. S. and Adeli, H. (1995).** "A neural dynamics model for structural optimization-Application to plastic design of structures." Computers and Structures, Vol. 57, No. 3, pp. 391-399.

12. **Simpson, P. K. (1991).** Artificial Neural Systems. Pergamon Press, New York, NY, USA.

13. **Senouci, A. B. and Abdul-Salam M. A. (1998).** "Prediction of Reinforced Concrete Beam Depth Using Neural Networks." Engineering Journal of the University of Qatar, Vol. 11, pp. 117-132.