

## Hybrid neural network model for the design of beam subjected to bending and shear

H SUDARSANA RAO and B RAMESH BABU

Department of Civil Engineering, J.N.T.U College of Engineering,  
Anantapur 515 002  
e-mail:hanchatel123@yahoo.co.in

MS received 25 September 2006; revised 8 March 2007

**Abstract.** There is no direct method for design of beams. In general the dimensions of the beam and reinforcement are initially assumed and then the interaction formula is used to verify the suitability of chosen dimensions. This approach necessitates few trials for coming up with an economical and safe design. This paper demonstrates the applicability of Artificial Neural Networks (ANN) and Genetic Algorithms (GA) for the design of beams subjected to moment and shear. A hybrid neural network model which combines the features of feed forward neural networks and genetic algorithms has been developed for the design of beam subjected to moment and shear. The network has been trained with design data obtained from design experts in the field. The hybrid neural network model learned the design of beam in just 1000 training cycles. After successful learning, the model predicted the depth of the beam, area of steel, spacing of stirrups required for new problems with accuracy satisfying all design constraints. The various stages involved in the development of a genetic algorithm based neural network model are addressed at length in this paper.

**Keywords.** Design of beam; hybridization; genetic algorithms; neural networks.

### 1. Introduction

Structural engineering involves understanding of material behaviour, laws of mechanics, intuition, past experience or expertise and analysis techniques. The modern computer can bring speed, efficiency and accuracy in analysis of structures. But to computerize the areas such as conceptual design, modelling of natural phenomenon and material behaviour, damage assessment, etc., is extremely difficult as it requires human expertise. Structural design is an iterative process. The initial design is the first step in design process. Though the various aspects of structural design are controlled by many codes and regulations, the structural engineer has to exercise caution and use his judgement in addition to calculations in the interpretation of the various provisions of the I.S 456–2000 code to obtain an efficient and economic design. After the design process the designer makes an overall guess about the possible optimum solution consistent with designer's experience, knowledge, constraints, and requirements. The analysis of the structure is then carried out using initial design. Based on the results of the analysis

a re-design of the structure is carried out if any of the constraints is not satisfied. The efficiency of the design process depends heavily on initial guess. A good initial design reduces the number of subsequent analysis–design cycles. This phase is extremely difficult to computerize as it needs human intuition. In recent years efforts have been made to computerize the initial design process using artificial neural networks as they can learn from available designs during training process. Artificial neural network is a new technology emerged from approximate simulation of human brain and has been successfully applied in many fields of engineering. Neural networks and genetic algorithm demonstrate powerful problem solving ability. They are based on quite simple principles but take advantage of their mathematical nature in terms of non-linear iteration. Neural networks with Back Propagation (BP) learning showed results by searching for various kinds of functions. However, the choice of basic parameters (Network topology, learning rate, initial weights) often already determine the success of the training process. However, there are no clear rules how to set these parameters. Yet these parameters determine the efficiency of training. On the other hand, genetic algorithms are global search methods, that are based on principles like selection, cross over, and mutation. By combining genetic algorithm with neural networks, considerable reduction in network parameters can be achieved. Thus, hybridization of neural networks with genetic algorithms considerably improves their efficiency. More details about the principles of neural networks and genetic algorithms can be found in many text books (Rajasekharan & Vijayalakshmi Pai 2003, and Davis 1991). Lot of research has taken place on applications of artificial neural networks in structural engineering. Artificial neural networks have been used for predicting compressive strength of concrete (Ni Hong-Guang & Wang 2000), ultimate shear strength of reinforced concrete deep beams (Sanad & Saka 2001), shear design of reinforced cement concrete beams (Cladera & Mari 2004), design of fibre reinforced concrete beams (Hadi 2002). A preliminary structural design for cable stayed bridges has also been developed (Namhee Kim *et al* 2002) and many researchers have successfully applied neural networks in diversified problems (Ghaboussi & Joghathie 1995; Mukherjee & Deshpande 1995; Mishra & Akhil Upadhyay 2004; Patodi & Purani 1998; Patodi & Satodia 1999). In most of these works the neural networks have been trained by using traditional back propagation algorithm (Rumelhart *et al* 1986). In this approach the connection weights of neural networks are initially set to some random values. These values are then modified automatically according to the learning algorithm during the process of learning. This type of learning requires huge number of training cycles and also requires higher network configuration. It is reported that these networks trained by back propagation algorithm may get trapped in a local minima (Rajasekaran & Vijayalakshmi Pai, 2003). To alleviate this problem the present paper proposes to use genetic algorithm in conjunction with back propagation neural networks. Genetic algorithms have been successfully used in the field of structural engineering. (Jenkins 1991 & 1992; Le Riche & Haftka 1993; Leite & Topping 1988; Rajeev & Krishnamurthy 1992; Smith & Tate 1993; Topping & Leite 1998; Topping & Bahreininejad 1997; Ramanjaneyulu & Lakshmi 2003).

In the present paper a hybrid neural network which combines the features of genetic algorithms and back propagation networks is presented as an improved approach for handling structural design problems. The applicability of genetic algorithms for modelling the structural design process of beam has been explored. As an example, a genetic algorithm based neural network model has been developed for the design of beam subjected to bending and shear. Genetic algorithm based neural network model has been developed to satisfy all the codal requirements. (I.S. 456–2000). The neural network learns the design process in an adaptive fashion through the training examples presented to it. The training examples have been obtained by posing different beam problems to various design experts and structural engineers.

The weights for the neural network have been obtained by using a genetic algorithm. This alleviates large number of training cycles required for training through back propagation algorithm and also reduces the configuration considerably. The developed neural network model has been validated by presenting new design problems after successful training. The hybrid neural network model successfully predicted the depth of beam, reinforcement and spacing of stirrups for new beam problems.

## 2. Development of hybrid neural network model

Development of a hybrid neural network model for the design of beam subjected to the moment and shear involves various stages which are addressed in the following sections.

### 2.1 Generation of exemplar patterns

The objective of this work is to develop a hybrid neural network i.e genetic algorithm based neural network model for the design of beams subjected to moment and shear. This requires a comprehensive set of examples that cover various parameters influencing the design of beams. All the training examples should invariably satisfy I.S. 456–2000 (2000) code provisions. For the present work, all the training examples have been developed by presenting different beam problems to various design experts. The experts were asked to provide designs satisfying codal provisions. The design variables considered are the moment, shear, breadth, grade of concrete and grade of steel. The example designs have been obtained for different moments and shear ranging from 30 kN-m to 125 kN-m and 30 kN to 120 kN respectively.  $M_{20}$  and  $M_{25}$  grades of concretes have been considered. Reinforcement steel of three different grades viz. Fe 250, 415 and 500 have been considered. For each set, the depth of beam, reinforcement required and spacing of vertical stirrups are obtained. For the present problem, a total of one hundred and twenty examples have been obtained from different experts such that these examples cover all the possible combinations of design variables considered. Out of these, one hundred examples have been used for training and twenty examples are used for validation. A part of the training set data is presented in tables 1 and 2. Table 1 presents the part of the training set for input vector and table 2 presents the part of the training set for output vector.

**Table 1.** Part of the training set (input vectors).

S. No.	Moment (kN-m)	Shear (kN)	Grade of concrete (N/mm <sup>2</sup> )	Grade of steel (N/mm <sup>2</sup> )	Breadth (mm)
1	30	70	20	250	250
2	40	80	25	415	300
3	50	80	20	415	350
4	60	100	20	250	300
5	100	120	25	415	300
6	70	90	20	250	300
7	75	55	25	500	300
8	90	60	20	500	300
9	80	35	25	250	320
10	45	50	25	250	300

**Table 2.** Part of the training set (out put vector).

S. No.	Depth (mm)	Area of steel (mm <sup>2</sup> )	Spacing of stirrups (mm)
1	220	878.8	75
2	215	698.47	95
3	250	768.92	150
4	280	1370.92	75
5	330	1110.39	110
6	300	1476.38	100
7	300	792.54	210
8	360	769.896	255
9	280	1827.90	95
10	220	1318.19	100

## 2.2 Selection of input and output

In the present work, it is required to develop a model for the design of beam subjected to bending and shear. This means, the model should be able to predict the values of depth, area of reinforcement, and spacing of stirrups for given moment, shear, breadth of section, grade of steel and concrete. Accordingly the input to the network is chosen as follows:

• Moment (M) • shear (V) • Grade of concrete ( $f_{ck}$ ) • Grade of steel ( $f_y$ ) • Breadth of beam (B).

That the input vector selected for this model is

$$IP = \{M, V, f_{ck}, f_y, B\}$$

The designer would like to know the depth of beam, area of the reinforcement and spacing of stirrups for any given problem.

Accordingly, the output vector for the neural network model is selected as

$$OP = \{D, A_{st}, S_v\}$$

The input and output parameters have been normalized in the range (0, +1) using suitable normalization or scaling factors. The details of the scaling factors for input parameter are presented in the table 3 and output parameter are presented in table 4.

**Table 3.** Scaling factors for input parameters.

Input parameter	Min. value	Max. value	Normalization factor
M (moment) kN-m	30	125	200
V (shear) kN.	30	120	200
B (Breadth) mm	250	350	500
$f_{ck}$ (Grade of concrete) N/mm <sup>2</sup>	20	25	50
$f_y$ (Grade of steel) N/mm <sup>2</sup>	250	500	1000

**Table 4.** Scaling factors for output parameters.

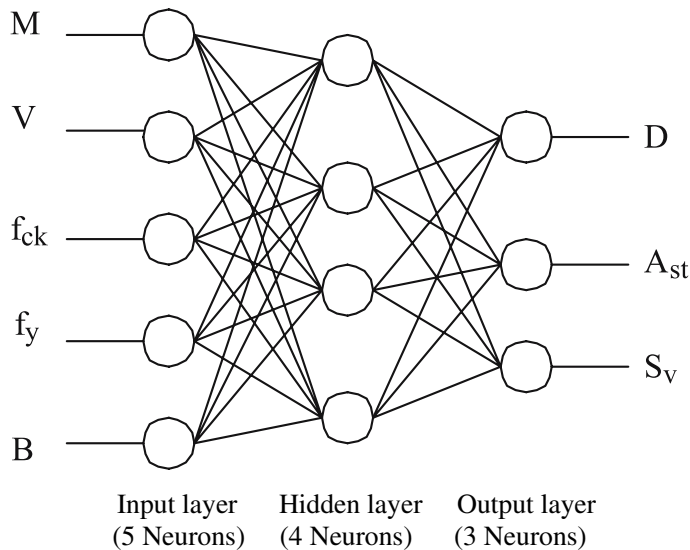
Output parameter	Min. value	Max. value	Normalization factor
$D$ (Depth of beam) mm	215	410	500
$A_{st}$ (Area of steel) N/mm <sup>2</sup>	566.1	2038.82	3000
$S_v$ (Spacing of stirrups) mm	75	280	500

2.3 *Selecting a suitable network configuration*

As mentioned earlier, the network configuration is defined in terms of the number, size, nodal properties, etc. of the input/output vectors and the intermediate hidden layers, once the input and output vectors are decided to cater the present investigation requirements, the task of selecting a suitable configuration has been taken up. There is no direct method to select number of nodes in hidden layers. Generally a trial and error method is adopted for arriving at the network configuration. After doing a few trials, it is observed that the network with 4 neurons in one hidden layer is behaving well. Accordingly a configuration of (5-4-3) has been selected for this network model. The architecture is depicted in figure 1.

2.4 *Training of the network*

Conventionally, a BPN determines its weights based on a gradient search technique and hence runs the risk of encountering local-minima. GA on the other hand is found to be good at finding ‘acceptably good’ solutions. The idea to hybridize the two networks has been successful to enhance the speed of training. (Rajasekaran & Vijayalakshmi Pai 2003). In the present work, the weights for the BPN have been obtained by using GA. Genetic algorithms which use a direct analogy of natural behaviour work with a population of individual strings, each representing a possible solution to the problem considered. Each individual string is assigned

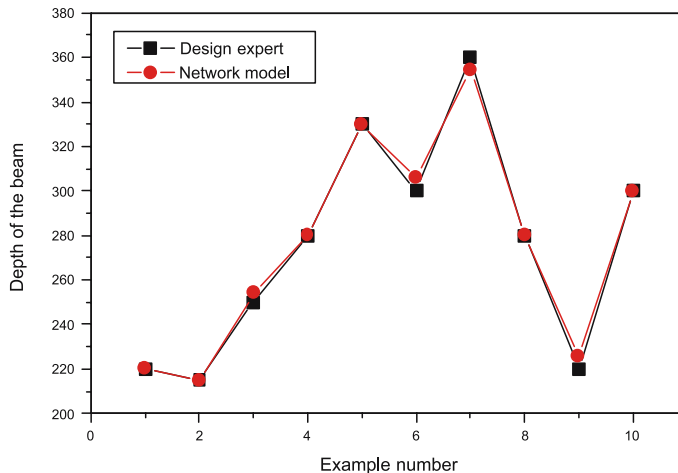


**Figure 1.** Configuration of GA/BPN model.

**Table 5.** Learning progress of the hybrid network.

S. No.	No. of training cycles	R.M.S error
1	100	0.024171
2	200	0.023880
3	500	0.018844
4	1000	0.016556

a fitness value which is an assessment of how good a solution is to a problem. The high-fit individuals participate in 'reproduction' by cross-breeding with other individuals in the population. This yields new individual strings as offspring which share some features with each parent. The least-fit individuals are kept out from reproduction and so they 'die out'. A whole new population of possible solutions to the problem is generated by selecting the high-fit individuals from the current generation. This new generation contains characteristics which are better than their ancestors. The parameters which represent a potential solution to the problem, *genes*, are joined together to form a string of values referred to as a *chromosome*. A decimal coding system has been adopted for coding the *chromosomes* in the present work. The network configuration chosen for the present work is 5 – 4 – 3. Therefore, the number of weights (*genes*) that are to be determined are  $5 \times 4 + 4 \times 3 = 32$ . With each *gene* being a real number, and taking the *gene* length as 5, the string representing the *chromosomes* of weights will have a length of  $32 \times 5 = 160$ . This string represents the weight matrices of the input-hidden layer-output layers. An initial population of *chromosomes* is randomly generated. Weights from each *chromosome* have been extracted then using the procedure suggested by Rajasekharan & Vijayalakshmi Pai (2003). The fitness function has been devised using FITGEN algorithm (Rajasekharan & Vijayalakshmi Pai 2003). A constant learning rate of 0.6 and a momentum factor of 0.9 have been adopted during the training. Satisfactory training has been obtained after just 1000 training cycles. The progress of the learning of the network is presented in table 5.

**Figure 2.** Learning for depth of the beam.

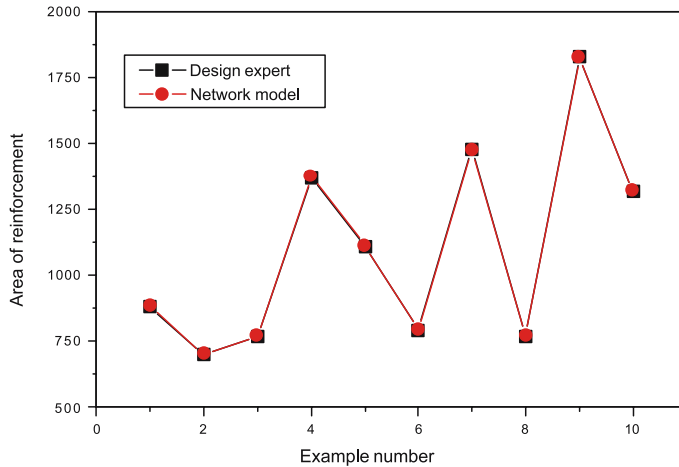


Figure 3. Learning for area of reinforcement.

The learning of the network model is presented in figures 2–4. These figures are presented for 10 data sets only. However, the authors have verified all the hundred data sets used for training and found that the network has learned the beam design problem satisfactorily. From the figures 2–4, it can be observed that the hybrid neural network model is able to predict the depth, reinforcement, spacing of stirrups correctly for the problems in the training set.

2.5 Validation of the network model

Validation of the network is to test the network for the parameters that are not used in training of the network. The network was asked to predict depth, reinforcement, spacing of stirrups for ten new problems which are not included in the training set. It can be seen that from figures 5–7, the values predicted by hybrid model for new sets match satisfactorily with results

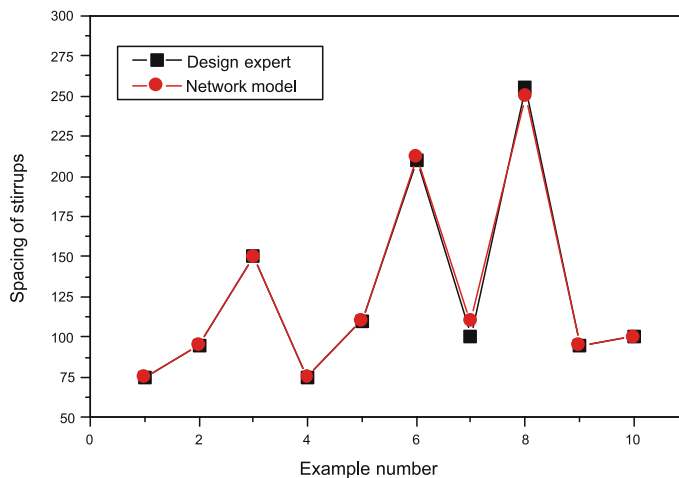
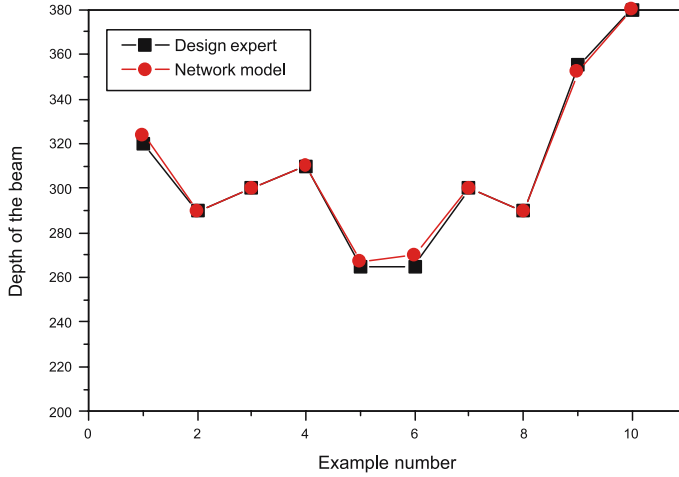


Figure 4. Learning for spacing of stirrups.

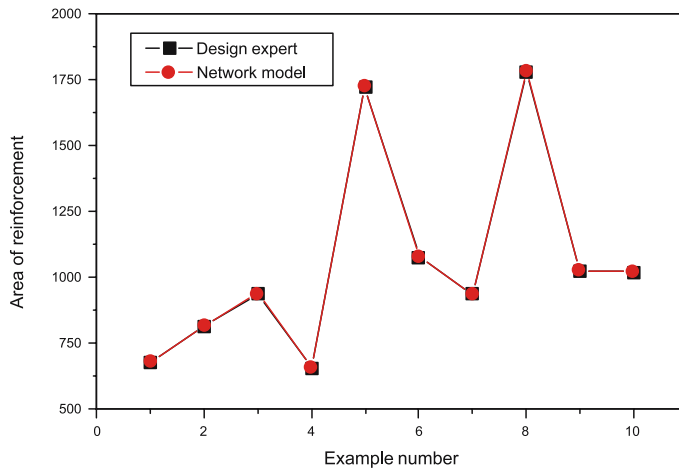


**Figure 5.** Validation for depth of the beam.

of design experts. Hence, the results of GA based model can be directly used for the design of beams.

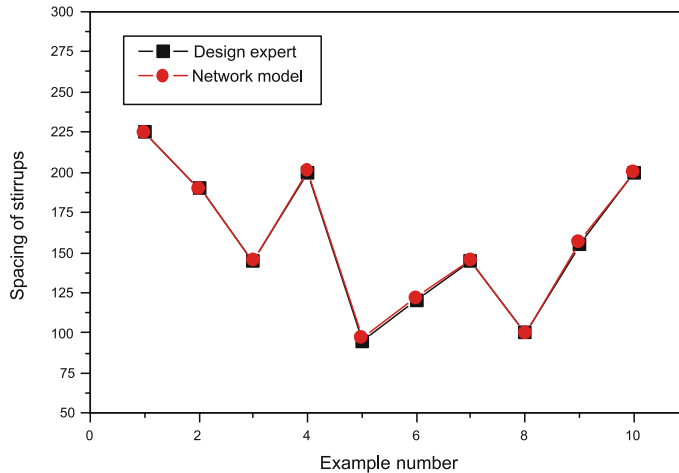
### 3. Conclusions

In this paper, the application of genetic algorithm based neural network model for the design of beam subjected to bending and shear has been demonstrated. The hybrid network model has been trained using one hundred examples obtained from different design experts. The training examples are so chosen that they will cover all the design variables involved in the problem. The weights for the network have been obtained using a genetic algorithm. The network could learn the beam design problem with just 1000 training cycles. After successful



**Figure 6.** Validation for area of reinforcement.





**Figure 7.** Validation for spacing of stirrups.

training, the neural network model is able to predict depth, area of reinforcement, spacing of stirrups satisfactorily for new beam problems. Thus, it is concluded that the developed neural network model can provide a safe design for the design of beams subjected to bending and shear.

## References

- Cladera A, Mari A R 2004 Shear design procedure for reinforced normal and high strength concrete beams using artificial neural networks. Part I: Beams without stirrups. *Eng. Struct.* 26: 927–936
- Cladera A, Mari A R 2004 Shear design procedure for reinforced normal and high strength concrete beams using artificial neural networks. Part II: Beams with stirrups. *Eng. Struct.* 26: 917–926
- Code of practice for plain and reinforced concrete IS 456–2000
- Davis L 1991 *Hand book of genetic algorithms*, (New York: Van Nostrand Reinhold)
- Ghaboussi J, Joghathie 1995 Active control of structures using networks. *J. Struct. Eng. ASCE* 121(4): 555–567
- Hadi N 2002 Neural networks applications in concrete structures. *Compute & Struct.* 81: 373–381
- Jenkins W 1991 Towards structural optimization via the genetic algorithm. *Compute & Struct.* 40(5): 1321–1327
- Jenkins W M 1992 Plane frame optimum design environment based on genetic algorithm. *J. Struct. Eng.* 118(11): 3103–3112
- Le Riche Rodolphe, Haftka Raphael T 1993 Optimization of laminate stacking sequence for buckling load maximization by genetic algorithm. *AIAA Journal* 31(5): 951–956
- Leite J P B, Topping B H V 1998 Improved genetic operators for structural engineering optimization. *Adv. Eng. software* 29(7–9): 529–562
- Mishra A K, Akhil Upadhyay 2004 Column design using ANN *Indian Concrete Institute Journal* 17–19
- Mukharjee A, Deshpande 1995 Application of artificial neural networks in structural design expert systems. *Compute & Struct.* 54(3): 367–375
- Namhee Kim Hong, Sung-Pil Chang, Seung-Chul Lee 2002 Development of Ann-based preliminary structural design systems for cable-stayed bridges. *Adv. Eng. software* 33: 85–96
- Ni Hong-Guang, Wang Ji-Zong 2000 Prediction of compressive strength of concrete by neural networks. *Cem. Concr. Res.* 30: 1245–1250

- Patodi S C, Satodia S M 1999 Application of neural networks in predicting behaviour of FRC deep beams. *J. Civil Eng. Today, ASCE-IS* 8(4): 6–14
- Patodi S C, Purani V S 1998 Modelling flexural behaviour of steel fibre reinforced concrete beams using neural networks. *J. New Build. Mater & Construct. World* 4(5): 28–35
- Rajasekharan S, Vijayalakshmi Pai G A 2003 Neural networks, Fuzzy logic and genetic algorithms, (New Delhi: Prentice Hall)
- Rajeev S, Krishnamoorthy C S 1992 Discrete optimization of structures using genetic algorithms. *J. Struct. Eng.-ASCE* 118(5): 1233–1249
- Ramanjaneyulu K, Lakshmi Ch 2003 Genetic algorithm based optimal design of continuous prestressed concrete bridge box girders. *J. Struct. Eng.* 30(1): 33–41
- Rumelhart D E, Hinton G E, Williams R J 1986 *Learning internal representations by error propagation. In Parallel distributed processing: Explorations in microstructure of cognition.* (eds) D E Rumelhart, J L McClelland, Foundations (Mass: MIT Press Cambridge) 1: 318
- Sanad A, Saka M P 2001 Prediction of ultimate strength of reinforced concrete deep beams by neural networks. *ASCE J. Struct. Eng.* 127(7): 818–828
- Smith A E, Tate D M 1993 Genetic optimizing using a penalty function. 5<sup>th</sup> *International conference on genetic algorithms*, July, San Mateo 1993
- Topping B H V, Bahreininejad A 1997 *Neural Computing for Structural mechanics*, (Edinburgh, UK: Saxe-Coburg Publications)
- Topping B H V, de Barros Leite J P 1998 Parallel genetic models for structural optimization. *Int. J. Eng. Optimization* 31(1): 65–99