International Journal of Research Studies in Science, Engineering and Technology
Volume 2, Issue 9, September 2015, PP 54-64
ISSN 2349-4751 (Print) & ISSN 2349-476X (Online)

Comparison between Feedforward Backpropagation and Radial Basis Neural Networks for Optimal Design of Reinforced Concrete Cantilever Beams

Jiin-Po Yeh, Kun-Yu Chen

Department of Civil and Ecological Engineering,
I-Shou University, Kaohsiung 84001, Taiwan

Abstract: *This paper aims to compare the results from the feed forward back propagation networks and radial basis networks for optimal design of reinforced concrete cantilever beams. Genetic algorithms are applied to obtain the optima data for the use of the networks. The inequality and quality constraints used in the algorithm to find the optimal results all comply with the ACI code, considering bending moment, shear force and deflection. The given conditions are the compressive strength of concrete, yield strength of steel, beam span, dead and live loads, and design variables are the effective depth and width of the beam as well as the steel ratio. The objective function is to minimize the total cost of the tension steel, stirrups and concrete. A variety of cantilever beams are designed and the optimal results are randomly divided into three sets: the training set, validation set and test set. The inputs of neural networks are the compressive strength of concrete, yield strength of steel and span, width and effective depth of the beam, as well as vertical loads applied on the beam; the targets of neural networks are the steel ratio and cost of the beam. Numerical results show the performance of feed forward back propagation networks is excellent and better than that of radial basis networks.*

Keywords: *Reinforced concrete beams, Genetic algorithms, neural networks, Regression analysis.*

1. Introduction

The genetic algorithm (GA) is inspired by Charles Darwin's well-known “survival of the fittest” theorem -- Individuals that are more "fit" have better potential for survival, The preliminary structure of genetic algorithms was introduced by Professor John Holland at the University of Michigan, who in 1975 published the ground-breaking book “Adaptation in Natural and Artificial System” [1]. Genetic algorithms have a variety of applications in a wide spectrum of problem areas including structural designs in civil engineering, such as plane frame optimal design [2], optimization of structures [3], multi objective optimization of trusses [4], global optimization of grillages [5], locating the critical slip surface in slope stability analyses [6], dependability assurance in the long-span suspension bridge design [7], continuous reinforced concreted beams [8], etc.

The preliminary theoretical base for contemporary neural networks was independently proposed by Bain [9] and James [10], whose works suggested that interactions among neurons within the brain resulted in both thoughts and body activities. McCullouch and Pitts [11] created a computational model for neural networks based on mathematics and algorithms. They called this model threshold logic that paved the way for neural network research to split into two distinct approaches. One approach focused on biological processes in the brain and the other focused on the application of neural networks to artificial intelligence. McCulloch and Pitts further claimed that neurons with binary inputs and a step-threshold activation function were analogous to first order systems. There were other authors devoted to this field, just to name a few: Hebb [12] revolutionized the perception of artificial neurons; Rosenblatt [13], using the McCulloch-Pitts neuron and the findings of Hebb, developed the first perception model of the neuron still widely accepted nowadays; Hopfield [14] and Hopfield et al. [15] illustrated from the study on the neuronal structure of the common garden slug that artificial neural networks are able to solve non-separable problems by placing a hidden layer between the input and output layers; Rumelhart and McClelland [16] invented the most famous learning algorithm that used a gradient descent technique to propagate error through a network and modify the weights by minimizing the global error, which marks the beginning of the current artificial neural networks.

Due to the strengths of genetic algorithms solving nonlinear optimization problems and superior learning ability of neural networks, this paper combines these two techniques to expedite the optimal design of reinforced concrete cantilever beams and compare the results obtained from feedforward backpropagation networks and radial basis networks.

2. Genetic Algorithms

Natural selection occurs at every life stage of an individual. An individual organism must survive until adulthood before it can reproduce. In many species, adults must compete with each other for mates through sexual selection, and success in this competition determines who will parent the next generation. The longer individuals can survive and the more competitive they are, the more offspring they will reproduce. Inspired by the natural evolution, genetic algorithms simulate the same process: inheritance, selection, crossover and mutation. It is a random search technique that can solve both constrained and unconstrained nonlinear optimization problems, whose constraints can be in the form of equality or inequality with bounds on the variables. Being less susceptible to getting stuck at local optima than gradient search methods is one of its advantages. The evolution usually starts from a population of randomly generated individuals. In each generation, the fitness of every individual in the population is evaluated. Multiple individuals are stochastically selected from the current population based on their fitness, recombined and randomly mutated to form a new population. A small portion of fittest individuals called elites are kept unchanged and passed on to the next generation. The new population is then used in the next iteration of the algorithm. The Matlab Toolbox for Genetic Algorithm [17] is employed in this paper to find the optimal solution.

3. Neural Networks

Two kinds of artificial neural networks are used in this paper: feedforward backpropagation networks and radial basis networks, which are briefly illustrated as follows:

3.1. Feedforward Backpropagation Networks

The neural network used in this paper is a two-layer feedforward backpropagation neural network, as shown in Figure 1, where there are six inputs and two outputs. The transfer function used in the single hidden layer with q neurons is the tan-sigmoid function

$$h_i = f(n_i) = \frac{e^{n_i} - e^{-n_i}}{e^{n_i} + e^{-n_i}} \quad , \quad i = 1,2,3,...,q \tag{1}$$

where $n_i = w_{i,1}R_1 + w_{i,2}R_2 + \ldots + w_{i,6}R_6 + b_i$, $R_1, R_2, \ldots, R_6$ are the inputs, $w_{i,1}, w_{i,2}, \ldots, w_{i,6}$ are the weights connecting the input vector and the ith neuron, and b_i is the bias of the ith neuron. The output layer with two neurons uses the linear transfer function

$$O_i = f(m_i) = m_i \quad , \quad i = 1,\ 2 \tag{2}$$

where $m_i = W_{i,1}h_1 + W_{i,2}h_2 + \ldots\ldots + W_{i,q}h_q + B_i$, $W_{i,1}, W_{i,2}, \ldots, W_{i,q}$ are the weights connecting the neurons of the hidden layer and the ith neuron of the output layer, and B_i is the bias of the ith output neuron

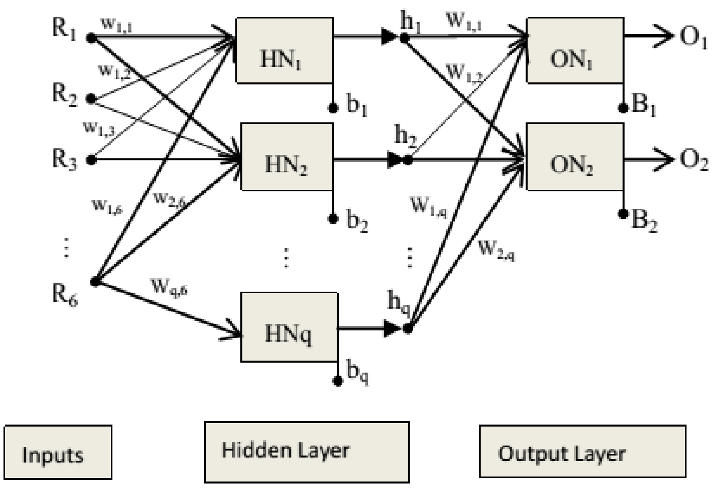

Fig1. *The feedforward backpropagation neural network with two layers*

3.2. Radial Basis Networks

The radial basis network has two layers: the radial basis layer and output linear layer. The transfer function in the radial basis layer is the radial basis function

$$radbas(n) = e^{-n^2} \tag{3}$$

where n= $\|\boldsymbol{w} - \boldsymbol{P}\| \, b$ is the vector distance (Euclidean distance) between the weight vector $\boldsymbol{w}$ and the input vector $\boldsymbol{P}$ multiplied by the bias b. As the distance between $\boldsymbol{w}$ and $\boldsymbol{P}$ decreases, the output increases. Thus the radial basis function acts as a detector that produces 1.0 whenever the input is identical to the weight vector. Each bias in the radial basis layer is set to be 0.8326/*SPREAD*, which causes radial basis function to output 0.5 when $\|\boldsymbol{w} - \boldsymbol{P}\|$ = +/- *SPREAD*. The parameter *SPREAD* needs to be large enough so that several radial basis neurons respond strongly to overlapping region of the input space, but not so large that all the neurons respond in essentially same manner [18]. The *SPREAD* can't be too small either, because it means many neurons are required to fit a smooth function and the network might not generalize well. The transfer function for the output layer is the linear function, which is the same as the feedforward backpropagation network. There are two functions provided in MATLAB to design the radial basis network: *newrb* and *newrbe*. The function *newrb* iteratively creates one radial basis neuron at a time. At each iteration, the input vector rending the network error lower is used to create a radial basis neuron. Neurons are added to the network until the sum-squared error falls below an error goal or maximum number of neurons has been reached. The function *newrbe* can produces a network with zero error on the training vectors. It creates the same number of radial basis neurons as there are input vectors, and each neuron acts as detector for a different input vector. The major disadvantage of *newrbe* is that it produces a network with as many hidden neurons as there are input vectors. For this reason, it does not yield an acceptable solution when many input vectors are required to suitably define a network, as is typically the case. Therefore, the function *newrb* is chosen in this paper

To evaluate the prediction accuracy, this paper performs the regression analysis between the network outputs and targets. The results are expressed by the parameters of the linear regression [19]: the correlation coefficient, the slope of the regression line and y-intercept.

4. Constraints of Cantilever Beams

A number of singly reinforced cantilever beams with uniformly distributed dead load w= 1.2w_D +1.6w_L are optimally designed by the genetic algorithm, as shown in Figure 2(a), where w_D *i*s the dead load and w_L is the lived applied on the beam, based on which the neural network is then trained and tested. The constraints required to design the beam are formulated according to the ultimate-strength design of the ACI Building Code Requirements for Structural Concrete and Commentary [20], considering the moment, shear force and deflection. The equality and inequality constraints for the genetic algorithm are discussed as follows, where force and length are measured in the units of kgf (=9.81N) and cm, respectively.

4.1. The Strength Requirement For Flexure

The moment diagram is shown in Figure 2(b). The strength requirement for flexure takes the form of

$$M_u \le \phi_m M_n \tag{4}$$

where $M_u = wL^2/2$ is the factored bending moment and w =1.2 w_D +1.6 w_L is the factored uniformly distributed load applied to the cantilever beam, ϕ_m is the strength reduction factor for flexure, which is

$$\phi_m = 0.65 + 0.25 \times \frac{\varepsilon_t - \varepsilon_y}{0.005 - \varepsilon_y} \le 0.9 \tag{5}$$

for tied sections, where ε_t is the tensile strain in the steel and ε_y is the yielding strain of the tensile reinforcement. The nominal resisting moment is

$$M_n = \rho b d^2 f_y \left(1 - \frac{\rho f_y}{2(0.85) f_c'}\right) \tag{6}$$

where f_y is the yield strength of the tension reinforcement, f_c' is the compressive strength of concrete, b and d are the width and effective depth of the beam, respectively, and ρ is the tension reinforcement ratio. To have reasonable assurance of ductile mode of failure, ACI code requires the steel strain to be at least 0.004; therefore, the reinforcement ratio ρ has the upper limit as

$$\rho \le \rho_{\max} = 0.85\beta_1 \frac{3f_c'}{7f_y} \tag{7}$$

where β_1 is the value of the concrete stress block depth factor. The code also stipulates the minimum steel requirement as

$$\rho \ge \rho_{\min} = \max(\frac{14}{f_y},\ \frac{0.8\sqrt{f_c'}}{f_y}) \tag{8}$$

so that the reinforced concrete element does not behave as a plain concrete section. Once ρ is decided, the total amount of tension steel can be found, which is $\rho bd\ell$. In order to have enough space to arrange the longitudinal steel bars and stirrups, the width of the beam is assumed to be at least 20 cm.

4.2. The Strength of Shear Reinforcement

The shear diagram is shown in Figure 2(c). Assuming that vertical stirrups are used, the strength of shear reinforcement

$$V_s = \frac{A_v f_y d}{s} \tag{9}$$

where s is the shear reinforcement spacing. If the nominal shear resistance $V_c = 0.53\sqrt{f_c'}bd$ is less than the nominal vertical shear force $V_u / \phi_s = V_n$, the shear reinforcement has to carry the difference in the two values, but the strength of shear reinforcement cannot be more than $2.12\sqrt{f_c'}bd$; hence

$$V_s = V_n - V_c = \frac{V_u}{\phi_s} - V_c \le 2.12\sqrt{f_c'}bd \tag{10}$$

where V_u is the factored shear force and $\phi_s = 0.75$ is the strength reduction factor for shear. A minimum shear reinforcement $A_v = 0.2\sqrt{f_c'}\frac{bs}{f_y}$ or $A_v = 3.5\frac{bs}{f_y}$, whichever is larger, must be provided to prevent brittle failure, if the factored shear force V_u exceeds one-half the shear strength $\phi_s V_c$.

4.3. Shear Reinforcement Spacing

According to the ACI code, the critical section for determining the closest stirrup spacing may be taken at a distance d from the face of support. It also stipulates that the maximum stirrup spacing s is $d/2$ but no to exceed 60 cm, $\frac{A_v f_y}{0.2\sqrt{f_c'}b}$ or $\frac{A_v f_y}{3.5b}$ for $V_s = \frac{V_u}{\phi_s} - V_c \le 2V_c$ (i.e., $V_u \le 3\phi_s V_c$), and $d/4$ but not to exceed 30 cm, $\frac{A_v f_y}{0.2\sqrt{f_c'}b}$ or $\frac{A_v f_y}{3.5b}$ for $V_s = \frac{V_u}{\phi_s} - V_c \ge 2V_c$ (i.e., $V_u \ge 3\phi_s V_c$). Since the spacing of stirrups cannot be varied continuously, they must change by jumps. Therefore, the span of the beam is divided into four regions: I, II, III and IV, as shown in Figure 3, where $3\phi_s V_c$ is assumed to be less than or equal to (wL-wd) and no stirrups are needed in Region IV. The first stirrup is place at 5 cm from the support. Once the minimum spacing of stirrups in each region is found, the total number and amount of stirrups required in the beam can be obtained.

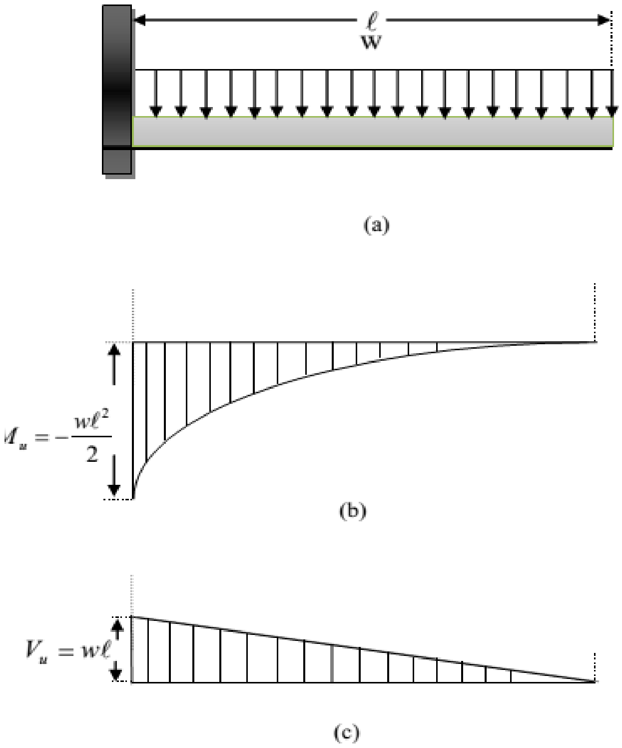

Fig2. *The cantilever beam. (a) Uniformly distributed load w; (b) moment diagram and (c) shear diagram.*

4.4. Deflection

Serviceability of a structure is determined by its deflection, cracking, extent of corrosion of its reinforcement and surface deterioration of its concrete. This paper only deals with deflection. The maximum instantaneous deflection in an elastic cantilever beam caused by dead load plus live load can be expressed as

$$\Delta_{iDL} = \frac{(w_D + w_L)L^4}{8E_c I_e} \tag{11}$$

If there is only dead load applied to the elastic beam, the maximum instantaneous deflection

$$\Delta_{iD} = \frac{(w_D)L^4}{8E_c I_e} \tag{12}$$

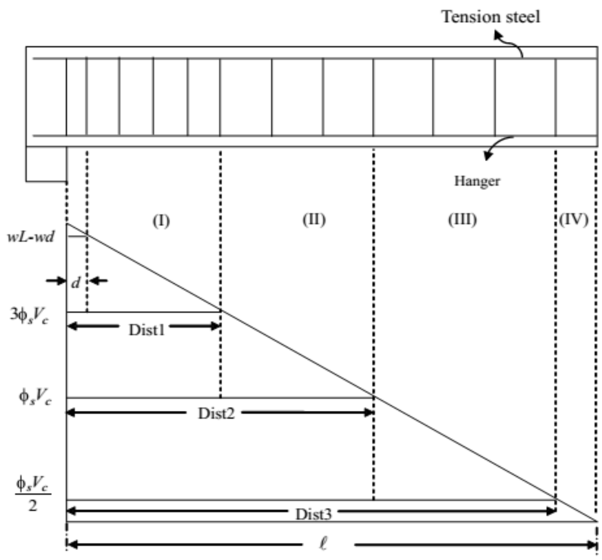

Fig3. *Four regions of the cantilever beam to arrange the web reinforcement.*

and the instantaneous deflection due to live load can then be obtained by subtracting Eq. (12) from Eq. (11), that is,

$$\Delta_{iL}=\Delta_{iDL}-\Delta_{iD} \tag{13}$$

In Eqs. (11) and (12), E_c is the modulus of elasticity of concrete and I_e is the effective moment of inertia, a smooth transition between the moment of inertia I_{cr} of the cracked section and the moment of inertia I_g of the gross uncracked concrete section. The effective moment of inertia is defined as

$$I_e=\left(\frac{M_{cr}}{M_a}\right)^3 I_g+\left[1-\left(\frac{M_{cr}}{M_a}\right)^3\right]I_{cr}\le I_g \tag{14}$$

where M_a is the moment at the fixed end and

$$M_{cr}=\frac{I_g f_r}{h/2} \tag{15}$$

is the cracking moment, where f_r is the modulus of concrete rupture strength and h is the depth of the beam. Suppose that the cantilever beam will support or be attached to nonstructural elements likely to be damaged by large deflections. The ACI code stipulates that sum of long-term deflection due to the sustained dead load plus immediate deflection due to the live load has to be less than $L/480$, that is,

$$\Delta_{sum}=\lambda\Delta_{iD}+\Delta_{iL}\le\frac{L}{480} \tag{16}$$

where λ is a multiplying factor considering long-term loading and shrinkage. According to the ACI code,

$$\lambda=\frac{T}{1+50\rho'} \tag{17}$$

where ρ' is the compression reinforcement ratio and T is a time-dependent factor that is taken as 2.0 for loading time duration of 5 years or more. Because ρ' is zero for a singly reinforced beam and 5 years or more are considered, the multiplying factor $\lambda=2$.

5. NUMERICAL RESULTS

The width b and effective depth d of the cantilever beam and tension reinforcement ratio ρ are the three variables for genetic algorithms. The fitness function is the total cost in New Taiwan Dollars of the tension reinforcement, stirrups and concrete. The inequality and equality constraints to formulate the optimization problem are built according to the discussion in Section 4. Based on the often-used materials and customs in Taiwan, this paper selects three kinds of yield strength f_y of the tension reinforcement: 2800 kgf/cm^2, 3500 kgf/cm^2 and 4200 kgf/cm^2, three kinds of compressive strength f'_c of the concrete: 210 kgf/cm, 280 kgf/cm^2 and 350 kgf/cm^2, three kinds of span L: 2 m, 3 m and 4 m and four kinds of dead load w_d: 2100 kgf/m, 2300 kgf/m, 2500 kgf/m and 2700 kgf/m. For simplicity, fix the live load at 1800 kgf/m. Hence, there are 108 combinations of beams to be designed. The prices for steel and concrete in Taiwan are NT$ 19.5/kgf and NT$ 1800/m^3, respectively. No. 3 vertical closed stirrups are used for all regions of the cantilever beam.

5.1. Genetic Algorithms

To run the genetic algorithm of MATLAB, some parameters need to be selected. Here are the values used in this paper: The population size is set to be 20, crossover rate 0.8, and elite number 2. Furthermore, all the individuals are encoded as real numbers; “Rank” is used as the scaling function that scales the fitness values based on the rank of each individual; “Roulette” is the selection function to choose parents for the next generation; The crossover function applies the “Two-Point Strategy” to form a new child for the next generation; The “Adaptive Feasible Function” is chosen as the mutation function to make small random changes in the individuals and ensure that linear constraints and bounds are satisfied. Taken as examples, some of the optimal results are listed in Table 1, where *dist1* represents the range of region I, *dist2-dist1* the range of region II, and *dist3-dist2* the range of region III, as indicated in Figure 3, and *spacing* represents the stirrup spacing in each region.

5.2. Feedforward Backpropagation Networks

For the purpose of training and testing the neural networks, the optimal results of the 108 beams are divided into three sets: training set (68 data), validation set (20 data) and test set (20 data). The input vector of the neural network consists of six elements: f_y, f'_c, w_d, L, b, and d, and the targets are the tension reinforcement ratio ρ and minimum price C. In light of past experience, six neurons are used in the hidden layer, which is the same number of elements in the input vector. The training process of the network with 6 neurons in the hidden layer is shown in Figure 4. The training stops at epoch 90 and the performance function is minimized to be 1.4229×10^{-4}. After the neural network is trained, the test data are then substituted into the network to do simulation. Figures 5 and 6 show the scatter plots of network outputs and targets of the 20 test data for the steel ratio and minimum cost C, respectively. Regression analysis of the network outputs and desired outputs (targets) are also carried out to evaluate the network accuracy. Table 2 shows the results, where the correlation coefficients between the network outputs and targets for the steel ratio and minimum cost are as high as 0.99971 and 0.99992, respectively, and the slope of the regression line and y-intercept are also close to one and zero, respectively, which suggests the excellent performance of the network. If more neurons are used in the hidden layer, the accuracy does not improve significantly or even sometimes becomes worse, as shown in Table 2.

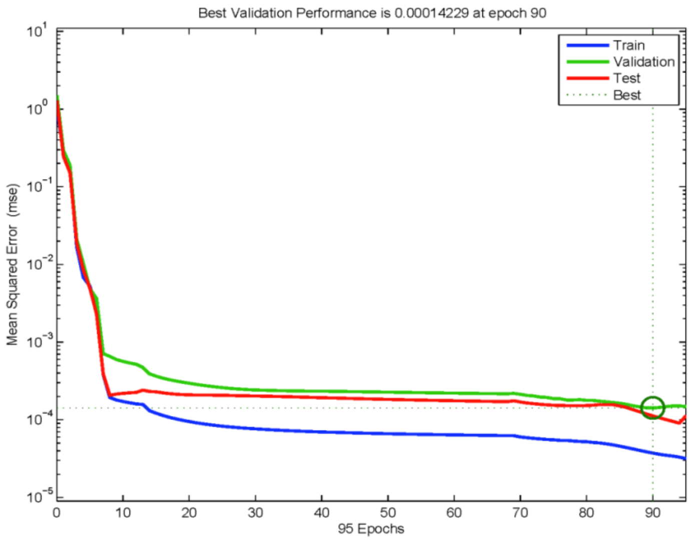

Fig4. *The training process of the feedforward backpropagation network with 6 neurons in the hidden layer.*

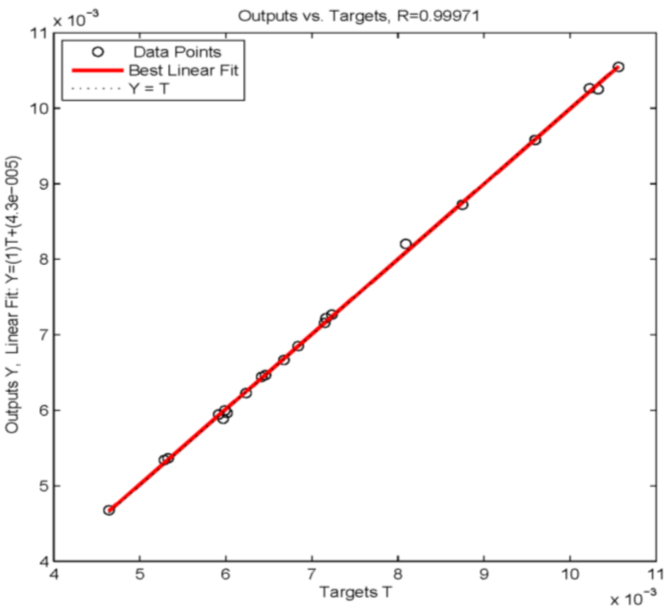

Fig5. *The scatter plot of the network outputs and targets of the 20 test data for the tension reinforcement ratio ρ with 6neurons in the hidden layer.*

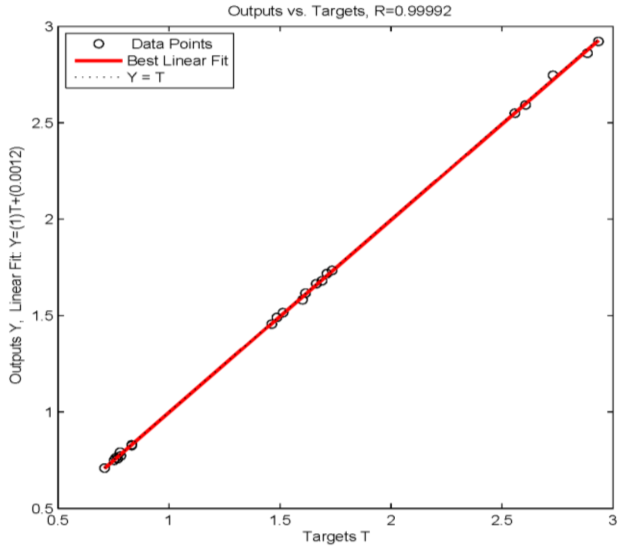

Fig6. *The scatter plot of the network outputs and targets of 20 test data for the minimum cost C (10^3NT$) with 6 neurons in the hidden layer.*

Table1. *Some optimal design results of the reinforced concrete cantilever beams.*

f_y (kgf/cm^2)	f'_c (kgf/cm^2)	L (m)	w_d (kgf/m)	*dist*1/spacing (cm/cm)	(*dist*2-*dist*1)/*spacing* (cm/cm)	(*dist*3-*dist*2)/*spacing* (cm/cm)
2800	210	3	2100	74.0/18.5	(139.5-74.0)/37.0	(219.8-139.5)/37.0
2800	280	4	2300	96.0/24.0	(172.2-96.0)/48.0	(286.1-172.2)/48.0
3500	280	2	2100	54.9/13.7	(62.3-54.9)/27.4	(131.2-62.3)/27.4
3500	350	4	2100	94.4/23.6	(131.1-94.4)/47.2	(265.6-131.1)/47.2
4200	210	3	2700	90.7/22.7	(127.8-90.7)/45.4	(213.9-127.8)/45.4
4200	350	3	2500	79.8/20.0	(97.8-79.8)/39.9	(198.9-97.8/39.9

b (cm)	d (cm)	ρ	C (10^3NT$)
20.3	74.0	0.009	1.675
20.1	96.0	0.010	2.884
20.4	54.9	0.006	0.748
20.7	94.4	0.008	2.617
20.2	90.7	0.005	1.601
20.0	79.8	0.006	1.489

Table2. *Regression analysis of the network outputs and targets of the test data for different neurons in the hidden layer.*

No. of Neurons in the Hidden Layer	Targets (or Outputs)	Slope of the linear regression.	Y-intercept of the linear regression	Correlation coefficient
6	Steel Ratio ρ	0.995088	0.000043	0.999711
	Minimum Cost *C*	0.996955	0.001150	0.999922
12	Steel Ratio ρ	1.005153	-0.000028	0.999637
	Minimum Cost *C*	0.998978	0.001651	0.999924
18	Steel Ratio ρ	0.985153	0.000164	0.994179
	Minimum Cost *C*	1.002919	0.004005	0.999719
24	Steel Ratio ρ	0.958795	0.000293	0.944512
	Minimum Cost *C*	0.978368	0.065326	0.995196

5.3. Radial Basis Networks

Because the radial basis network does not include the validation data during the training process, only training and testing data are considered. For comparison, these two sets of data are identical to those of feedforward backpropagation with 6 neurons in the hidden layer. The mean square error between the network outputs and targets is set to be 0.00014229, which is the same as the feedforward backpropagation. The value of *SPREAD* is gradually increased from 0.1when the network is being trained. By observing the regression analysis of the network outputs and targets for each *SPREAD*, the network with *SPREAD*=3 is found to have best performance; therefore, *SPREAD*=3 is selected for the radial basis layer. After the training is complete, the test data are then substituted into the network. Figures 7 and 8 show the scatter plots of network outputs and targets of the 20 test data for the steel ratio ρ and the minimum *C*, respectively. Regression analysis of the network outputs and desired outputs (targets) are also carried out to determine the network accuracy. The results reveal that the

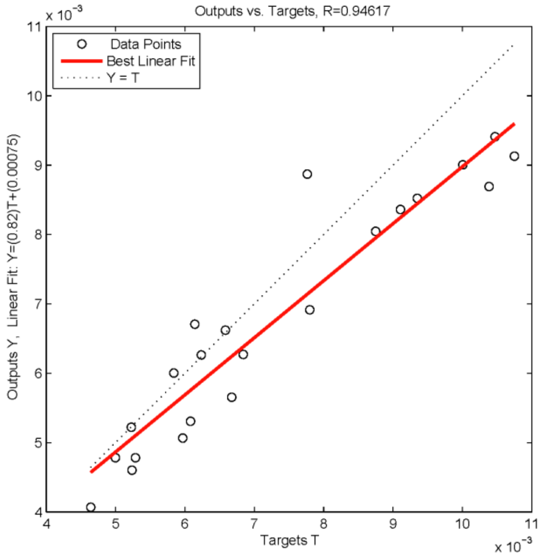

Fig7. *The scatter plot of the network outputs and targets of the 20 test data for the tension reinforcement ratio ρ of the radial basis network*

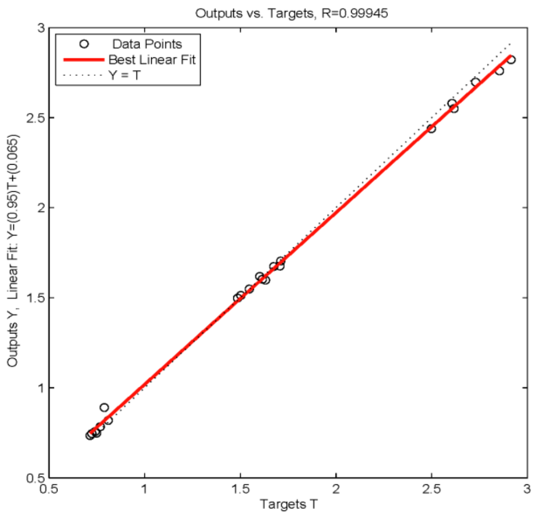

Fig8. *The scatter plot of the network outputs and targets of the 20 test data for the minimum cost C (10^3NT$) of the radial basis network.*

correlation coefficients between the network outputs and targets for the steel ratio and minimum cost are 0.94617 and 0.99945, respectively, and the slope of the regression line and y-intercept are not simultaneously close to one and zero, respectively.

6. Conclusions

Taking the optimal design of cantilever beams as example, the performances of feedforward backpropagation networks and radial basis networks are compared. Genetic algorithms are first used to obtain the optimal results, which serve as the training, validation and test data of the neural networks. Using 6 neurons in the hidden layer of the feedforward backpropagation network is enough for the network to have very high accuracy, with correlation coefficients of the steel ratio and cost reaching higher than 0.999, the slope of the regression line close to 1 and y-intercept of the regression close to zero. More neurons in the hidden layer are not necessarily beneficial to the network. The trained networks can quickly design cantilever beams with high accuracy once the required inputs are entered. As to the radial basis networks, although the correlation coefficients are good, the slope and y-intercept of the regression line are not simultaneously close to one and zero, respectively, which renders them worse than feedforwad backpropagation networks. The validation data not included in the radial basis networks to monitor the training process is probably the major reason to have poor performance.

REFERENCES

[1] J. H. Holland, Adaptation in Natural and Artificial Systems, The University of Michigan Press, Ann Arbor, Michigan, 1975.

[2] W. M. Jenkins, “Plane Frame Optimum Design Environment Based on Genetic Algorithm,” ASCE Journal of Structural Engineering,Vol. 118, No. 11, pp. 3103-3113, 1992..

[3] S. Rajeev and C. S. Krishnamoorthy, “Discrete Optimization of Structures Using Genetic Algorithms,” ASCE Journal of Structural Engineering, Vol. 118, No. 5, pp. 1233-1250, 1992.

[4] C. A. Coello and A. D. Christiansen, “Multiobjective Optimization of Trusses Using Genetic Algorithms,” Computers and Structures, Vol. 75, pp. 647-660, 2000.

[5] R. Belevičius and D. Šešok, Global optimization of grillages using genetic algorithms, Mechanika. Nr., Vol. 6, No. 74, pp. 38-44, 2008.

[6] Y.-C. Li, Y.-M. Chen, T. L. T. Zhan, D.-S. Ling and P. J. Cleall, An efficient approach for locating the critical slip surface in slope stability analyses using a real-coded genetic algorithm. Canadian Geotechnical Journal, Vol. 47, No. 7, pp. 806–820, 2010.

[7] L. Sgambi, K. Gkoumas and F. Bontempi, “Genetic Algorithms for the Dependability Assurance in the Design of a Long-Span Suspension Bridge,” Computer-Aided Civil and Infrastructure Engineering, Vol. 27, No. 9, pp. 655-675, 2012.

[8] J-P Yeh and R-P Yang, “ Optimal Design of Continuous Reinforced Concrete Beams Using Neural Networks,” Transactions on Machine Learning and Artificial Intelligence, Vol. 3, No. 4, pp. 1-15, 2015.

[9] A. Bain, Mind and Body: The Theories of Their Relation. D. Appleton and Company New York, 1873.

[10] W. James, the Principles of Psychology, H. Holt and Company, New York, 1890.

[11] W. S. McCulloch and W. Pitts, “A Logical Calculus of Ideas Immanent in Nervous Activity,” Bulletin of Mathematical Biophysics, Vol. 5, pp. 115-133, 1943.

[12] D. D. Hebb, The Organization Behavior, Wiley, New York, 1949.

[13] F. Rosenblatt, “The Perception: A Probabilistic Model for Information Storage and Organization in the Brain,” Psychological Review, Vol. 65, No. 6, pp. 386-408, 1958.

[14] J. J. Hopfield, “Neural Networks and Physical Systems with Emergent Collective Computational Abilities,” Proceedings of the National Academy of Sciences of the United States of America, Vol. 79, No. 8, pp. 2554-2558, 1982.

[15] J. J. Hopfield, D. I. Feinstein and R. G. Palmers, “Unlearning Has a Stabilizing Effect in Collective Memories,” Nature, Vol. 304, pp. 158-159, 1983.

[16] D. E Rumelhart and J. L McClelland, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1 (Foundations), The MIT Press, Cambridge, Massachusetts, 1986.

[17] The MathWorks, Global Optimization Toolbox 3 User’s Guide, The MathWorks, Inc., Natick, Massachusetts, 2010.

[18] H., Demuth, M. Beale and M. Hagan, Neural network toolbox 6: user’s guide, The MathWorks, Inc., Natick, MA, 2009.

[19] D. D. Wackerly, W. Mendenhall and R. L. Scheaffer, Mathematical Statistics with Applications, Cengage Learning, Boston, Massachusetts, 2008.

[20] ACI committee 318, Building Code Requirements for Structural Concrete (ACI 318-08) and Commentary (ACI 318R-08), American Concrete Institute, Farmington Hills, Michigan, 2008.

AUTHORS' BIOGRAPHY

Jiin-Po Yeh received his M.S. and Ph.D. degrees in Civil Engineering and Engineering Mechanics from Columbia University, New York, USA, in 1986 and 1990, respectively. He is currently a professor in the Department of Civil and Ecological Engineering at I-Shou University, Taiwan. He has been working withI-Shou University since 1990, whose research interests focus on chaotic dynamics, structural optimization and neural networks.

Kun-Yu Chen was a student of the first author and just received his M.S. degree in Civil and Ecological Engineering from I-Shou University, Taiwan. He currently works at Chung Hsin Electric & Machinery Mfg. Corp., Ltd., Taiwan, as an engineer.