AIAA JOURNAL
Vol. 34, No. 2, February 1996

Neural Networks with Modified Backpropagation Learning Applied to Structural Optimization

Srinivas Kodiyalam* and Ram Gurumoorthy†
General Electric Corporate Research and Development Center, Schenectady, New York 12301

A multilayer feedforward neural network with a modification to the standard backpropagation training of neural nets is considered. Information on the gradients of target outputs with respect to network inputs is used in this modification. This modified neural network is then applied for approximating design requirements of aerospace composite components, such as an aircraft engine guide vane and a satellite reflector assembly. This neural network functional approximator is then used with an optimization algorithm for determining the optimal parameters for component design requirements. The modified neural network provides for faster convergence of learning process and requires a smaller number of hidden layer nodes for convergence to similar training error levels.

I. Introduction

THE application of neural networks as universal approximators has been the focus of tremendous amount of activity in the past few years.[1–6] Neural networks have been used for approximating and modeling both memoryless systems and dynamical systems with memory. This paper does not go into the various neural net architectures and training schemes, and the interested reader could refer to standard texts on this subject matter.[1,2] This paper considers static neural networks (systems without memory) formed by interconnections without loops, called feedforward networks.[2] In addition, this work considers a backpropagation learning scheme with adaptation of learning rate. There have been many additions to, or adaptations of, the basic backpropagation scheme, such as adaptive learning rate, learning with momentum, and the like,[1,2] for improving convergence of the learning scheme, and these are not addressed in this paper.

This paper explores a modification of the backpropagation learning scheme using information on the gradient of the target outputs with respect to the inputs of the network. The performance of the modified learning scheme is compared with that of the standard backpropagation learning scheme.[2] A feedforward network with a single hidden layer trained with this modified learning scheme is applied in structural optimization problems. Figure 1 shows a feedforward net with a single hidden layer. The restriction to a single hidden layer is not a limitation of the modified learning scheme, and the new training scheme can be used on networks with multiple hidden layers also.

The use of neural networks for structural response approximation has been reported by several authors.[4–6] The primary reason for investigating such properties is to assess the applicability of trained neural networks for quick and accurate approximation of response functions during structural design optimization. Structural optimization, utilizing either gradient-based numerical search or genetic search algorithms, requires repeated function (response) evaluations during the search process. A single finite element analysis of large-scale structures requires a significant amount of computational time and resource and hence the need for robust approximations to minimize the number of detailed finite element analyses during optimization. Also, by training the network on a wide domain, the converged solution from optimization can be more global. In this paper, a feedforward network, trained with the modified backpropagation scheme, is used for function approximation during the structural optimization process. Comparisons of this new method's performance with respect to a neural network trained using the standard backpropagation scheme and with respect to a Taylor series approximation scheme are presented.

II. Modification to the Backpropagation Learning Scheme: Single Neuron

To illustrate the philosophy of the proposed modification to the backpropagation learning scheme, a net with a single neuron is considered. Figure 2 shows a single tansigmoid neuron net.

The output of the net is given by

$$o = f(wu) \tag{1}$$

where

$$f(x) = \frac{2}{1 + e^{-2x}} - 1$$

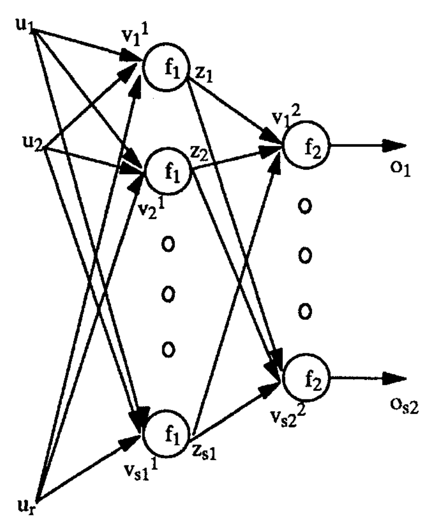

Fig. 1 Feedforward neural net with single hidden layer.

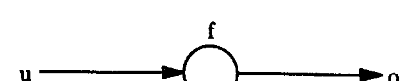

Fig. 2 Neural net with single neuron.

Received Feb. 27, 1995; presented as Paper 95-1370 at the AIAA/ASME/ASCE/AHS/ASC 36th Structures, Structural Dynamics, and Materials Conference, New Orleans, LA, April 10–12, 1995; revision received Oct. 2, 1995; accepted for publication Oct. 15, 1995.

*Staff Engineer, Engineering Mechanics Laboratory, Building K1/2A25, P.O. Box 8; currently Senior Staff Scientist, Structures Laboratory, Research and Development Division, Lockheed Martin, Palo Alto, CA 94304-1191. Senior Member AIAA.

†Staff Engineer, Control Systems and Electronics Technology Laboratory, Building K1/4C20, P.O. Box 8.

A scalar energy function P is defined as follows:

$$P = 0.5(T - o)^2 = 0.5(e)^2 \quad (2)$$

where T is the target value of the output. The objective of the training scheme is to make the difference between the target output and the actual output to be zero. This can be achieved by maintaining the gradient of the energy function to be negative:

$$\mathrm{d}P = (T - o)\left[\frac{\partial e}{\partial w}\,\mathrm{d}w + \frac{\partial e}{\partial u}\,\mathrm{d}u\right] \quad (3)$$

$$= (T - o)\left\{-(1 - o^2)u\,\mathrm{d}w + \left[\frac{\partial T}{\partial u} - (1 - o^2)w\right]\mathrm{d}u\right\} \quad (4)$$

To make $\mathrm{d}P$ negative, that is, to have the training scheme reduce the difference between the target output and the actual output, the weight update is defined as

$$\mathrm{d}w = (T - o)(1 - o^2)u + \left[-w + \frac{1}{(1 - o^2)}\frac{\partial T}{\partial u}\right]\frac{\mathrm{d}u}{u} \quad (5)$$

The term inside the square bracket in the weight update law is the modification proposed to the standard backpropagation.

III. Modification to the Backpropagation Learning Scheme: Two-Layer Feedforward Network

The modification to the backpropagation learning scheme has been derived for a two-layer feedforward network with one hidden layer (refer to Fig. 1). The derivation is as follows:

$$o = f_2(v^2); \qquad v^2 = w_2 z + b_2; \qquad f_2 = \text{linear} \quad (6)$$

$$z = f_1(v^1); \qquad v^1 = w_1 u + b_1; \qquad f_1 = \text{tansigmoid} \quad (7)$$

where o is of dimension $(s2 \times 1)$, v^2 is $(s2 \times 1)$, z is $(s1 \times 1)$, v^1 is $(s1 \times 1)$, and u is $(r \times 1)$. The scalar energy function is defined earlier as

$$P = 0.5(T - o)^2 \quad (8)$$

The gradient $\mathrm{d}P$ is given by

$$\mathrm{d}P = \frac{\partial P}{\partial w_1}\mathrm{d}w_1 + \frac{\partial P}{\partial b_1}\mathrm{d}b_1 + \frac{\partial P}{\partial w_2}\mathrm{d}w_2 + \frac{\partial P}{\partial b_2}\mathrm{d}b_2 + \frac{\partial P}{\partial u}\mathrm{d}u \quad (9)$$

where

$$\frac{\partial P}{\partial w_1} = -(1 - z \cdot z)\left(w_2^T e\right)u^T \quad (10)$$

$$\frac{\partial P}{\partial b_1} = -(1 - z \cdot z)\left(w_2^T e\right) \quad (11)$$

$$\frac{\partial P}{\partial w_2} = -(e\,z^T) \quad (12)$$

$$\frac{\partial P}{\partial b_2} = -(e) \quad (13)$$

$$\frac{\partial P}{\partial u} = -w_1^T(1 - z \cdot z)\left(w_2^T e\right) + \frac{\partial T}{\partial u}e \quad (14)$$

To get the network output o to converge to the target outputs T, the weight and bias updates are chosen to be as follows:

$$\mathrm{d}w_2 = -\frac{\partial P}{\partial w_2}; \qquad \mathrm{d}b_2 = -\frac{\partial P}{\partial b_2}; \qquad \mathrm{d}b_1 = -\frac{\partial P}{\partial b_1}$$
$$\mathrm{d}w_1 = -\frac{\partial P}{\partial w_1} - \left[\left(\frac{\partial P}{\partial w_1}\right)^{-1}\left(\frac{\partial P}{\partial u}\right)_{\text{diag}}\right] \quad (15)$$

where $(\partial P/\partial u)_{\text{diag}}$ is a matrix with each element of the vector $(\partial P/\partial u)$ made the corresponding diagonal element of the matrix. The term in the square bracket of Eq. (15) is the modification to the training scheme.

For multiple hidden layer networks, only the weights corresponding to the first hidden layer would be affected by this modification.

IV. Normalization of Network Variables

While presenting the input/output pairs to the neural network, they were normalized between 0 and 1:

$$u = \frac{u_o - u_o^{\min}}{u_o^{\max} - u_o^{\min}} \quad (16)$$

$$T = \frac{T_o - T_o^{\min}}{T_o^{\max} - T_o^{\min}} \quad (17)$$

where u_o = original inputs, u = normalized inputs, T_o = original outputs, and T = normalized outputs. With the normalized inputs and outputs, the gradients now change to

$$\frac{\partial T}{\partial u} = \frac{\partial T}{\partial T_o}\frac{\partial T_o}{\partial u_o}\frac{\partial u_o}{\partial u} \quad (18)$$

where

$$\frac{\partial T}{\partial T_o} = \frac{1}{T_o^{\max} - T_o^{\min}}$$

$$\frac{\partial u_o}{\partial u} = u_o^{\max} - u_o^{\min}$$

$$\frac{\partial T}{\partial u} = \left(\frac{u_o^{\max} - u_o^{\min}}{T_o^{\max} - T_o^{\min}}\right)\frac{\partial T_o}{\partial u_o}$$

Note that though the original inputs and outputs have been normalized, with this normalization the outputs of the neural network may still be out of the [0, 1] range as we have a linear neuron at the output layer. This is especially so when we extrapolate out of the input range.

V. Generation of Training Data and Choice of Initial Weights

The generation of training data is very important in determining the quality of approximation as well as the globality of the approximation. In this paper, the inputs were selected to be uniformly spaced over the data range. Also, the number of data sets used for training were limited, and to this end the data sets were restricted to some quadrants and did not cover the entire space.

The gradient information was used in selecting the quadrants and also in selecting the number of points for each input. In the future, more sophisticated data selection techniques based on techniques like the Taguchi method[7] will be tried.

The Nguyen and Widrow random generator[8] is used for the choice of initial weights. The weights are initialized such that the linear region of tan-sigmoid neurons lies near the region where inputs are likely to occur. The same initial weights are used while training with the standard and modified backpropagation algorithms for comparison of performances.

VI. Structural Optimization Architecture

The neural network approximation (NNA) based structural optimization methodology is shown in Fig. 3. This methodology is similar to the well-established Taylor series approximation (TSA) based structural optimization.[9] Both the finite element analysis and design sensitivity analysis are used for training the neural network. The network training is performed off line, and the trained network is then used with an optimization algorithm for solving the structural design optimization problem. The trained network is used to closely approximate the true values of objective and constraint functions during the numerical search process and thereby to minimize the number of detailed finite element analysis.

The structural optimization problem is posed as a constrained optimization problem of the following form:

To find the set of design variables X that minimizes or maximizes $F(X)$ subject to $g_j(X) \leq 0$, $j = 1$, number of inequality constraints, and $x_i^l \leq x_i \leq x_i^u$, bounds on design variables.

The method of feasible directions, programmed in the Automated Design Synthesis numerical optimization code,[10] is used to solve the constrained optimization problem.

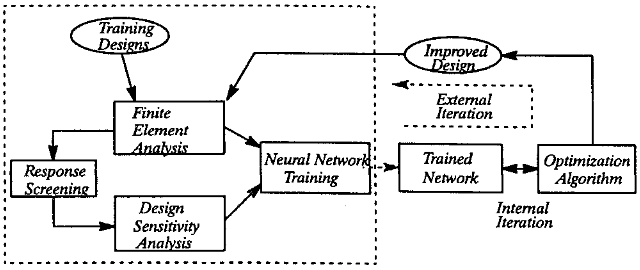

Fig. 3 Structural optimization program architecture.

VII. Design Examples

Two aerospace composite components, an aircraft engine guide and a satellite reflector assembly, are used to demonstrate the function approximation capability of the modified neural network, and the network approximation is further used with an optimization algorithm for component design optimization.

A. Satellite Reflector

The dual surface, parabolic reflector structure, shown in Fig. 4, is to be designed for minimum thermally induced surface distortions, so as to improve the reflector pointing accuracy. The reflector consists of a front and rear dish, intercostal rings, and a backing structure. The rear dish and backing structure are made of composite facesheets and nonmetallic honeycomb core, whereas the intercostal rings and front reflector dish are made of composite facesheets and nonmetallic honeycomb core. The finite element model consists of over 1900 triangular and quadrilateral plate elements and a total of 6570 degrees of freedom. The design load conditions include quasistatic transient loads, acoustic pressure loads, and on-orbit thermal cases, including, worst case hot and worst case cold conditions. Thermal analyses are performed to determine the maximum and minimum temperature predictions and gradients corresponding to these cases and to provide inputs for thermal distortion and stress analyses.

The design optimization problem is formulated to minimize the structural weight of the reflector. Design constraints are imposed on thermally induced surface distortions at the center of the front reflector dish in the deployed configuration, reflector frequency in stowed configuration, ply strength, and sandwich construction local buckling failure margins of safety. A total of seven design variables are used, including the composite sandwich facesheet and honeycomb core thicknesses corresponding to the front reflector dish (t_f^F, t_c^F) and rear reflector dish (t_c^R), intercostal ring (t_f^I, t_c^I), and backing structure (t_f^B, t_c^B).

A feedforward neural net with a single hidden layer is used. Two cases, first with 10 hidden layer nodes and second with eight hidden layer nodes, are investigated to evaluate the performance of this neural network approximation. The seven design variables are treated as the network inputs, whereas the four structural parameters (weight, fundamental frequency, thermal distortion, and sandwich buckling strength minimum margin of safety) are used as the network outputs. A total of 45 input-output training pairs are used for training the network. The network convergence criterion is defined based on the sum squared error of the approximation being less than a preset value.

The optimization results are provided in Table 1. For the TSA-based optimization, a first-order, hybrid approximation based on the sign of the derivatives is used for frequency, distortion, and sandwich local buckling, whereas a linear approximation is used for structural weight. An upper and lower bound of ±50% of the current design variable values is used for each iteration. This is a fairly large move limit for optimization with TSA, and the approximate responses were sufficiently accurate to overcome any constraint violations quickly. A total of six iterations is required for convergence of the TSA-based optimization. The initial neural network approximation and the final converged approximation of both the networks—one with standard backpropagation training and the other with modified backpropagation training—of output 1 (thermal distortion) are presented in Fig. 5. As can be seen, both networks have converged on a similar approximation, which is almost piecewise linear. This ties in with the results in Table 1 that both networks' performances are similar.

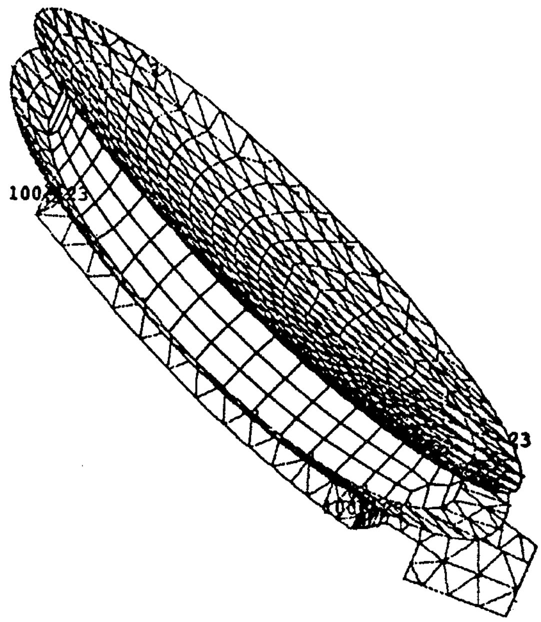

Fig. 4 Satellite reflector.

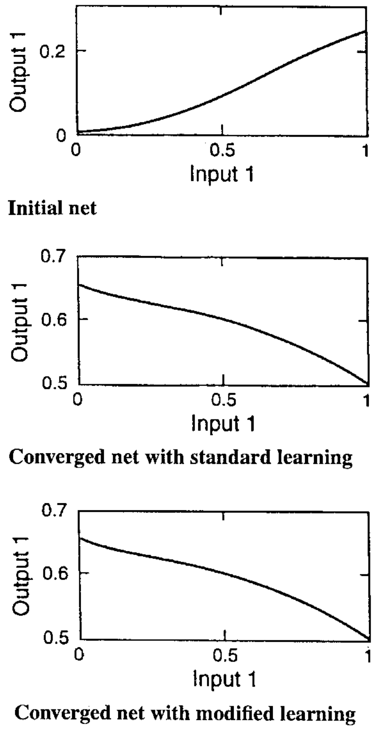

Fig. 5 Satellite reflector—initial and final approximations of networks.

B. Aircraft Engine Guide Vane

The aircraft engine component, shown in Fig. 6, is a composite guide vane with a viton cap and a viton ring at each of its two ends. Viton is soft, rubbery material that is used for vibration damping. The design problem addressed here is to determine the composite ply orientations of the guide vane that would produce a stiffer structure. The stiffness requirements are specified in terms of frequency requirements, and these are primarily imposed to prevent the viton cap and ring from degrading quickly and thereby eliminating the need to frequently replace them. Producibility requirements of the composite ply layup are also an integral part of the guide vane design.

Table 1 Optimization results for satellite reflector

Data	Baseline	TSA optimization	NNA-based optimization: 10 hidden layer nodes, Standard	NNA-based optimization: 10 hidden layer nodes, Modified	NNA-based optimization: 7 hidden layer nodes, Standard	NNA-based optimization: 7 hidden layer nodes, Modified
t_f^{Front}	0.064	0.053	0.051	0.052	0.056	0.053
t_c^{Front}	1.0	1.0	1.0	1.0	1.0	1.0
t_c^{Rear}	1.0	1.0	1.0	1.0	1.0	1.0
$t_f^{\mathrm{Back.Structure}}$	0.064	0.045	0.042	0.042	0.049	0.044
$t_c^{\mathrm{Back.Structure}}$	1.0	1.0	1.0	1.0	1.0	1.0
$t_f^{\mathrm{Int.Ring}}$	0.08	0.08	0.08	0.08	0.073	0.08
$t_c^{\mathrm{Int.Ring}}$	1.0	1.0	1.0	1.0	1.0	1.0
Structure weight	1.0	0.95	0.94	0.95	0.95	0.95
Frequency	1.025	1.0	1.0	1.0	1.0	1.0
Thermal distortion	1.0	1.0	1.0	0.98	1.0	0.99
Minimum sandwich buckling margin of safety	0.663	0.187	0.009	0.012	0.348	0.156
No. of data sets for NN training			42	42	45	45
No. of epochs for NN training convergence			5335	5304	19,694	19,428

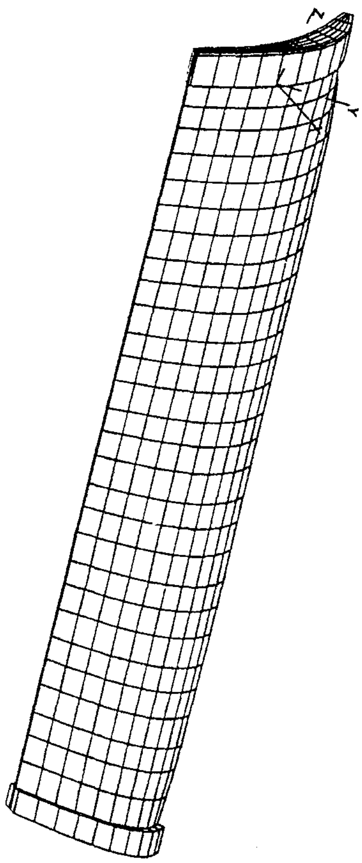

Fig. 6 **Aircraft engine outlet guide vane.**

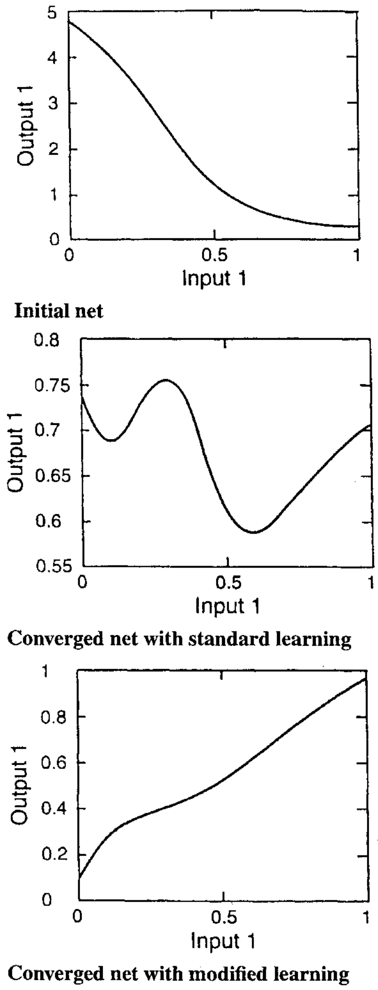

Fig. 7 Aircraft engine guide vane—initial and final approximation of networks.

The guide vane is modeled using 990 eight-noded isoparametric finite elements. The composite ply layup is symmetric and is given by

$$\left\{[\theta_1,\theta_2,\theta_1,\theta_3,\theta_4,\theta_3,\theta_1],[\theta_2,\theta_1,\theta_1,\theta_4,\theta_1,\theta_1]_n\right\}_{\mathrm{sym}}$$

The ply layup consists of the two generation sets with four unique ply angles (θ_1–θ_4). Some of the producibility requirements on the composite ply orientations include that 1) the plies be layered as three-ply packs, 2) the first and third ply orientation angle in each pack be the same (θ_1, θ_2, θ_1), and 3) all plies be of constant thickness of approximately 5 mil.

The design requirements on the guide vane frequencies include the following: mode 1 (first flex frequency): $\omega_{1F} > 1.0$ Hz, mode 6 (third flex frequency): $4.0 < \omega_{3F} < 6.7$ Hz, mode 7 (third torsion frequency): $4.0 < \omega_{3T} < 6.7$ Hz, and mode 14 (two stripe frequency): $\omega_{2S} > 9.8$ Hz.

A finite element normal modes analysis is performed to compute the guide vane natural frequencies. The frequencies are normalized

Table 2 Optimization results for aircraft engine guide vane

			NNA-based optimization			
			10 hidden layer nodes		7 hidden layer nodes	
Data	Baseline	TSA optimization	Standard	Modified	Standard	Modified
θ_1, deg	0	0	−5	−5	−5	−5
θ_2	+45	+45	+30	+30	+30	+30
θ_3	+90	+90	+90	+90	+90	+85
θ_4	−45	−25	−25	−15	−25	−20
ω_{1F}, Hz	1.0	1.03	1.058	1.062	1.058	1.059
ω_{3F}	5.7	5.56	5.52	5.39	5.52	5.41
ω_{3T}	6.0	5.88	5.92	5.87	5.92	5.88
ω_{2S}	9.9	9.88	9.96	9.98	9.96	9.97
No. of data sets for NN training			29	29	29	29
No. of epochs for NN training convergence			233,822	244,658	445,598	237,641
NN error goal			0.05	0.05	0.06	0.06

with respect to first flex frequency corresponding to the baseline layup.

A feedforward neural net with a single hidden layer is used. Two cases, first with 10 hidden layer nodes and second with eight hidden layer nodes, are investigated to evaluate the performance of the neural network approximation. The four unique ply orientation angles are treated as the network input, whereas the four frequencies of interest make up the outputs. A total of 29 input-output training pairs are used for training the network. The network convergence criterion is defined based on the sum squared error of the approximation being less than a preset value.

The results are tabulated in Table 2. A first-order hybrid approximation is used with the TSA-based optimization. Since the design responses (frequencies) are highly nonlinear with respect to the ply orientations, a small move limit (10%) is used with the TSA-based optimization. A total of seven iterations is required for convergence of the TSA-based optimization. The initial and final approximations of network output 1 (fundamental frequency) by both the networks are presented in Fig. 7. As can be seen, the network trained with the standard backpropagation scheme has overfit the curve, whereas the network trained with the modified backpropagation scheme has approximated with a lower order fit. This ties with the lesser number of epochs taken while training, by the net trained with the modified backpropagation scheme. However, both network approximations perform better than the TSA-based optimization.

VIII. Summary

This paper investigates a modification of the standard backpropagation training of a feedforward neural network and the application of neural networks for approximating structural responses for use with design optimization.

The modified backpropagation training scheme is compared with the standard backpropagation training in two examples, a satellite reflector example and an aircraft engine guide vane component. As seen from the results, the modified training scheme requires a smaller or similar number of training epochs to achieve same error levels. Also, the approximations achieved with the modified training scheme are similar and sometimes improved over the standard training scheme, as can be seen from the improved optimal designs (especially in the guide vane example). The modified neural network provides for faster convergence of learning process and requires a smaller number of hidden layer nodes for convergence to similar training error levels, compared with the standard network. This is especially true for the aircraft engine guide vane problem where the design responses (frequencies) are nonlinear functions of the design variables (ply layup parameters), and the modified neural network converges in half the number of epochs as compared with the standard network.

The trained network is then used for structural optimization, and the results are compared with those from a Taylor series approximation based structural optimization. Based on the two examples, the neural network approximation provides for improved optimal designs compared with the Taylor series approximation. This is achieved at the cost of and because of the increased number of finite element analyses used for training the neural network. The increased number of analyses has made the neural network a more global approximator, thereby improving performance. It is here, precisely, that the modified learning scheme could help in reducing the number of inputs required for learning the similar approximation attained in the standard backpropagation scheme. It is under further investigation.

References

[1]Kosko, B., *Neural Networks for Signal Processing*, Prentice–Hall, Englewood Cliffs, NJ, 1992.

[2]Zurada, J. M., *Introduction to Artificial Neural Systems*, West Publishing Co., St. Paul, MN, 1992.

[3]Hunt, K. J., Sbarbaro, D., Zbikowski, R., and Gawthrop, P. J., "Neural Networks for Control Systems: A Survey," *Automatica*, Vol. 28, No. 4, 1992, pp. 1083–1122.

[4]Hajela, P., and Berke, L., "Neurobiological Computational Models in Structural Analysis and Design," *Computers and Structures*, Vol. 41, No. 4, 1991, pp. 657–667.

[5]Szewczyk, Z., and Hajela, P., "Feature-Sensitive Neural Networks in Structural Response Estimation," *Proceedings of the Annual Conference of Neural Networks in Engineering—ANNIE '92*, St. Louis, MO, 1992, pp. 85–90.

[6]Carpenter, W., and Barthelemy, J., "A Comparison of Polynomial Approximations and Artificial Neural Nets as Response Surfaces, *Proceedings of the 33rd Structures, Structural Dynamics, and Materials Conference*, AIAA, Washington, DC, 1992, pp. 2474–2482 (AIAA Paper 92-2247).

[7]Taguchi, G., and Konishi, S., *Taguchi Methods: Orthogonal Arrays and Linear Graphs—Tools for Quality Engineering*, American Supplier Inst., Dearborn, MI, 1987.

[8]Nguyen, D., and Widrow, B., "Improving the Learning Speed of Two Layer Neural Networks by Choosing Initial Values of the Adaptive Weights," *International Joint Conference of Neural Networks*, Vol. 3, July 1990, pp. 21–26.

[9]Miura, H., Vanderplaats, G. N., and Kodiyalam, S., "Experiences in Large Scale Structural Design Optimization," *Applications of Supercomputers in Engineering: Fluid Flow and Stress Analysis Applications*, Elsevier, Amsterdam, 1989, pp. 251–268.

[10]Vanderplaats, G. N., *ADS User's Manual*, VMA Engineering, Colorado Springs, CO, 1988.