Structural Engineering and Mechanics, Vol. 12, No. 5 (2001) 527-540
DOI: http://dx.doi.org/10.12989/sem.2001.12.5.527

Whole learning algorithm of the neural network for modeling nonlinear and dynamic behavior of RC members*

Kayo Satoh†, Nobuhiro Yoshikawa‡, Yoshiaki Nakano‡ and Won-Jik Yang‡†

Institute of Industrial Science, The University of Tokyo, 4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan

Abstract. A new sort of learning algorithm named whole learning algorithm is proposed to simulate the nonlinear and dynamic behavior of RC members for the estimation of structural integrity. A mathematical technique to solve the multi-objective optimization problem is applied for the learning of the feedforward neural network, which is formulated so as to minimize the Euclidean norm of the error vector defined as the difference between the outputs and the target values for all the learning data sets. The change of the outputs is approximated in the first-order with respect to the amount of weight modification of the network. The governing equation for weight modification to make the error vector null is constituted with the consideration of the approximated outputs for all the learning data sets. The solution is neatly determined by means of the Moore-Penrose generalized inverse after summarization of the governing equation into the linear simultaneous equations with a rectangular matrix of coefficients. The learning efficiency of the proposed algorithm from the viewpoint of computational cost is verified in three types of problems to learn the truth table for exclusive *or*, the stress-strain relationship described by the Ramberg-Osgood model and the nonlinear and dynamic behavior of RC members observed under an earthquake.

Key words: neural network; whole learning algorithm; Moore-Penrose generalized inverse; material nonlinearity; RC members; earthquake response.

1. Introduction

An accurate prediction of the nonlinear and dynamic behavior of structural members is indispensable for the design of a structure which will suffer from crucial events that may violate its integrity during its lifetime, such as earthquakes. A large number of mathematical models have been proposed to model the material nonlinearity of structural members in dynamic problems. In spite of great efforts to derive these mathematical models, their applicability for the analysis of the dynamic behavior of structural members such as reinforced concrete seems questionable, since these models are founded on the basis of the continuum mechanics. In reality, the hypothesis of continuum is not validated to describe the nonlinear and dynamic behavior of structural members just prior to the collapse.

† Technical Assistant
‡ Associate Professor
‡† Graduate Student

*The earlier version of this paper appears in proceedings of ASEM'99, 23-25 August 1999, Seoul, Korea.

A candidate of promising methodology for the description of such nonlinear behavior has been promoted by making use of the feedforward neural network in lieu of the mathematical models (Ghaboussi, Garrett Jr. and Wu 1991, Yamamoto 1992). Its versatility to make up any kind of map from the learning data sets of the input and output without any educated knowledge incites the application of the feedforward neural network in accordance with the development of the computational performance. However, the computational cost to learn the complicated dynamic behavior of real structural members by the feedforward neural network has not been reduced enough for its common usage.

Aiming at the enhancement of the learning efficiency of the feedforward neural network, we propose a new algorithm named "whole learning algorithm" in this study. The learning of the feedforward neural network is categorized as a kind of multi-objective optimization problem to minimize the error functions defined by the difference between the outputs and the target values for all the learning data sets. A legitimate technique to solve this kind of multi-objective optimization problem is given by the minimization of the square-sum of all the error functions. This technique is applied for learning and formulated by means of the Moore-Penrose generalized inverse (Rao and Mitra 1971). The change of the outputs is approximated in the first-order with respect to the change of the weights of the network after the truncation of the higher order terms of the Taylor series expansion in the vicinity of the current value. The governing equation for weight modification is derived so as to make all the approximated outputs equal to their target values, and results in the form of the linear simultaneous equations with a rectangular matrix of coefficients. The contribution of weight modifications for all the outputs in the learning data sets is summarized in a rectangular matrix. The solution of the equation, which corresponds to the solution of least squares, is determined by employing the particular solution obtained by using the Moore-Penrose generalized inverse.

The fundamental characteristic of the proposed whole learning algorithm is revealed to prove its learning efficiency through three problems of a different number of learning data sets to learn the truth table for exclusive *or*, the stress-strain relationship described by the Ramberg-Osgood (1943) model and the nonlinear and dynamic load-displacement relationship of RC members observed under an earthquake.

2. Whole learning algorithm

2.1 Learning of neural network

For the sake of simplicity, we consider the feedforward neural network with a single output unit in this study. The input-output function of each hidden unit is set by the logistic function. On the other hand, both the linear function and the logistic function are employable for the output unit. We introduce an extra input for each unit which always has a value of 1.0. The weight for this extra input is equivalent to the threshold of the opposite sign and is handled by the same manner as ordinary weights for the connection (Rumelhart, Hinton and Williams 1986).

The learning of the feedforward neural network is classified in the multi-objective optimization problem to minimize the error functions f_n defined by Eq. (1) for all the learning data sets with respect to the weights associated with the connection between the units,

$$f_n = | T_n - O_n | \qquad (n = 1, \cdots, N) \tag{1}$$

where O_n denotes the output from the network for the n-th set of learning data, T_n the corresponding target value and N the total number of learning data sets.

In the conventional learning algorithm such as back-propagation algorithm (Rumelhart, Hinton and Williams 1986), individual error function f_n is minimized successively in line with the conventional optimization technique such as the steepest descent method. Although the final goal of the learning is the minimization of all the error functions $f_1, \cdots, f_N$, the conventional learning algorithms have been applied under the anticipation that all the functions are minimized by the successive minimizations of the individual error functions. The successive minimizations without any consideration about the effect of weight modification for other error functions cause the low efficiency of learning and the loss of the learning stability, so that a technique to compensate these disabilities should be introduced in terms of the learning rate and the momentum parameter (Rumelhart, Hinton and Williams 1986, Plaut, Nowlan and Hinton 1986). Consequently, we suppose that the learning algorithm to minimize the Euclidean norm of the error vector $\{ f \}$, which consists of the error functions $f_1, \cdots, f_N$, can increase the learning efficiency and name it "whole learning algorithm" in the sense of the minimization of the whole error functions.

2.2 Formulation

In the proposed whole learning algorithm, the value of the j-th weight w_j is changed from the current value $\bar{w}_j$ with small increment Δw_j. The governing equation of the amount of weight modification Δw_j is derived in line with the optimization technique based on the approximation with derivatives. After the Taylor series expansion and the truncation of the higher order terms, the change of the output is linearly approximated with respect to Δw_j, then the error function f_n is approximated as,

$$f_n = \left| T_n - \left(O_n + \sum_{j=1}^{J} \frac{\partial O_n}{\partial w_j} \Delta w_j \right) \right| \qquad (n=1, \cdots, N) \tag{2}$$

where J denotes the total number of weights. All the learning data sets are taken into account to derive N approximated error functions, and we constitute the governing equation to determine the amount of weight modification in the matrix form of Eq. (3).

$$\begin{Bmatrix} f_1 \\ \vdots \\ f_N \end{Bmatrix} = \begin{Bmatrix} T_1 - O_1 \\ \vdots \\ T_N - O_N \end{Bmatrix} - \begin{bmatrix} \frac{\partial O_1}{\partial w_1} & \cdots & \frac{\partial O_1}{\partial w_J} \\ \vdots & \ddots & \vdots \\ \frac{\partial O_N}{\partial w_1} & \cdots & \frac{\partial O_N}{\partial w_J} \end{bmatrix} \begin{Bmatrix} \Delta w_1 \\ \vdots \\ \Delta w_J \end{Bmatrix} = \{0\} \tag{3}$$

For the sake of simplicity, we rewrite Eq. (3) as,

$$\{f\} = \{b\} - [A]\{\Delta w\} = \{0\} \tag{4}$$

where $[A]$ is a rectangular matrix of coefficients in the dimension of N rows by J columns, $\{b\}$ a

constant vector of N components and $\{\Delta w\}$ an unknown variable vector. In case that the network consists of K output units, $N \times K$ outputs are linearly approximated as Eq. (2) to constitute the governing equation with the coefficient matrix of $N \times K$ rows.

We determine the solution of an unknown variable vector $\{\Delta w\}$, which minimizes the Euclidean norm of $\{f\}$, by using the Moore-Penrose generalized inverse $[A]^-$ as Eq. (5) (Rao and Mitra 1971).

$$\{\Delta w\} = [A]^-\{b\} \tag{5}$$

The condition of solution existence given by Eq. (6) is satisfied, the solution of least squares by Eq. (5) makes the error vector $\{f\}$ of Eq. (4) null vector,

$$\left([A]\,[A]^- - [I] \right)\{b\} = \{0\} \tag{6}$$

where $[I]$ denotes the identity matrix of proper size. Even in the case that the condition of solution existence is not satisfied, Eq. (5) gives the approximate solution of least squares, that minimizes the Euclidean norm of the error vector $\{f\}$ of Eq. (4).

The Moore-Penrose generalized inverse is calculated by the method of singular value decomposition based on the eigenvalue analysis of $[A]^T[A]$ or $[A][A]^T$ (Hangai and Kawaguchi 1991). Suprefix T denotes the matrix transpose hereafter. We need to calculate the eigenpairs of the smaller matrix of $[A]^T[A]$ or $[A][A]^T$. The row and column size of $[A]$ are equal to the number of learning data sets and weights, respectively. In general, the number of weights is much less than that of learning data sets in practical applications and does not increase so much as. Therefore the computational time does not increase in proportion to the number of learning data sets.

The matrix $[A]$ does not become full rank in many cases, and in that case we must carefully go through the identification of non-zero eigenvalues. It seems legitimate to give the criterion of non-zero eigenvalue in terms of the ratio to the maximum eigenvalue for the handling of computational results. We set the ratio as 10^{-7} through the preliminary numerical experiments to learn the truth tables for the logical sum, exclusive *or* and three-bit parity problems.

Pal *et al*. have also proposed a learning algorithm to utilize the generalized inverse in the similar manner with the proposed whole learning algorithm (Pal, Kayaba, Morishita and Hagiwara 1994). However, they determine the amount of weight modification by successive changes of the learning data sets and constitute the governing equation in the form of a linear equation with the coefficient matrix of a single row so as to eliminate the laborious procedure to identify the rank of the coefficient matrix in a practical application. Hence, they obligatorily employ the method of successive weight modifications by successive changes in the learning data sets.

The problem of the approximation error shadows the proposed whole learning algorithm based on the first-order derivations of the output, as is always the case with the optimization technique based on the derivations. We expect that this kind of error will be compensated by the iterations of weight modifications in the manner of Eq. (7) with the coefficient ε of small value,

$$w_j = \overline{w}_j + \varepsilon \Delta w_j \tag{7}$$

where ε corresponds with the learning rate which is commonly used in the conventional back-propagation algorithm.

3. Numerical examples

The computational efficiency of the proposed whole learning algorithm is verified through the learning of three problems of different number of learning data sets. The smallest problem is to learn the truth table for exclusive *or*, which has been widely employed in the benchmark test of the learning algorithm and necessitates three layered neural network (Rumelhart, McClelland and the PDP Research Group 1986, Hertz, Krogh and Palmer 1991). The intermediate one is to learn the nonlinear stress-strain relationship under uni-axial loading described by the Ramberg-Osgood model, which is one of the simplest mathematical models to deal with the material nonlinearity based on continuum mechanics. Finally, the applicability of the proposed whole learning algorithm is examined in the problem to learn the nonlinear and dynamic behavior of real RC members subjected to an earthquake. The robustness of the proposed whole learning algorithm in the learning efficiency against the increase of learning data sets is proven by these numerical examples in comparison with the conventional back-propagation algorithm and relatively modern algorithm proposed by Pal *et al.* (1994).

We employ a three layered neural network with an adequate number of units in these numerical examples. The initial values of the weights are set by a random number generator with uniform distribution function. We use two measures for the comparison of the efficiency of the learning algorithm by time for learning and number of learnings. The number of learnings is counted by the number of usages of all the learning data sets.

3.1 Truth table of exclusive or

The truth table for exclusive *or* is listed in Table 1 where 0.0 and 1.0 indicate false and true, respectively. The total number of learning data sets is four in this example. As is always the case for the learning, the inputs are normalized so as to make the minimum and maximum value equal to −1.0 and 1.0, and the output is equal to 0.0 and 1.0, respectively. The network consists of two input units, two hidden units and a single output unit. The input-output function is set by the logistic function for the hidden units and by the linear function for the output unit. The coefficient matrix $[A]$ of Eq. (4) becomes the rectangular matrix of 4 rows by 9 columns. One learning consists of one weight modification by Eq. (7) for the proposed whole learning algorithm, whereas four weight modifications for other two learning algorithms. The criterion to stop the learning is given in terms of the Euclidean norm of the error vector $\{f\}$. We stop the learning when the norm becomes less than 0.02, that is, when the average of the square error per data set becomes less than 10^{-4}. In case that the norm is not reduced less than the value of 0.02 after ten thousands learnings, we finish the learning and regard the trial as failure.

Table 1 Truth table for exclusive *or*

Input 1	Input 2	Output
0.0	0.0	0.0
0.0	1.0	1.0
1.0	0.0	1.0
1.0	1.0	0.0

Table 2 Learning results for exclusive *or*

Algorithm	Number of successful trials	Average learning number	Average learning time [msec]
Whole learning	5	8	68.0
Back-propagation	9	158	43.3
Proposed by Pal *et al*.	9	340	51.3

Table 3 Outputs for exclusive *or* by a trained network

Input 1	Input 2	Target value	Whole learning	Back-propagation	Proposed by Pal *et al.*
0.0	0.0	0.0	−0.0014	0.0065	0.0087
0.0	1.0	1.0	0.9897	0.9898	0.9907
1.0	0.0	1.0	0.9943	0.9912	0.9909
1.0	1.0	0.0	−0.0011	0.0130	0.0092
Norm of error vector			0.0119	0.0198	0.0182

We try ten trials of different initial weights which are randomly generated in accordance with the uniform distribution between −1.5 and 1.5. The learning rate ε of Eq. (7) is set equal to 0.5 for the proposed whole learning algorithm, whereas the learning rate and the momentum parameter are 0.1 and 0.5, respectively, for the conventional back-propagation algorithm. These values of the parameter yield the most effective learning for each algorithm in the preliminary numerical experiments.

The results of the learning computed by UltraSPARC170 are shown in Table 2, where the number of successful trials and the average number and time of learnings for the successful trials are listed. It takes a longer time for the learning by the proposed whole learning algorithm, though the average learning number is lower than the other two learning algorithms. In this example of small number of learning data sets, most of the computational time is consumed for the eigenvalue analysis. This is the reason for the longer time of the proposed whole learning algorithm. We should also take the low number of successful trials into account. The fact implies the fragility of the proposed whole learning algorithm against the inadequate initial weights, especially for this example of small numbers of learning data sets.

The quality of the learning is ensured by the outputs from the trained network. The outputs and the Euclidean norm of the error vector are shown in Table 3. These outputs are given by the network of the shortest learning time among ten trials. It shows that the proposed whole learning algorithm performs accurately enough in comparison with the other two learning algorithms.

3.2 Ramberg-Osgood model

We employ two hysteresis loops of 200 and 300 MPa in stress amplitude with a yield stress of 245 MPa and a yield strain of 0.12 % for the learning as shown in Fig. 1. Each loop is discretized into 51 points by the equivalent stress intervals to make 102 learning data sets. A single set of learning data for the input consists of the maximum stress and strain, the latest peak stress and strain; that is, if stress and strain is at the point of the latest unloading, and current strain, then the output is set as current stress (Yamamoto 1992). The inputs are normalized categorywise so as to

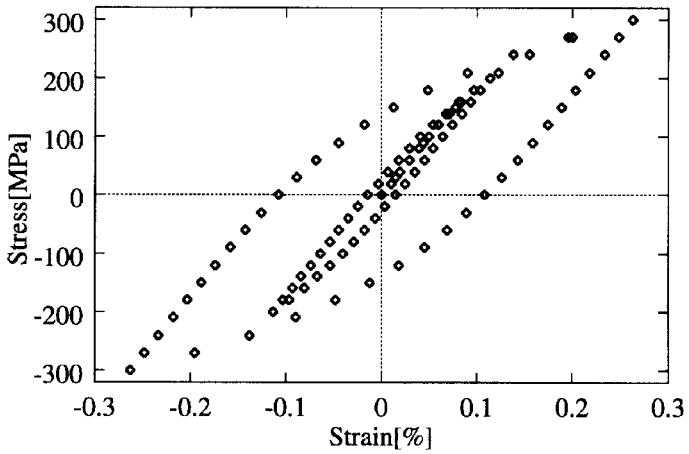

Fig. 1 Learning data for the Ramberg-Osgood model

make the minimum and maximum value equal to −1.0 and 1.0, and the output is equal to 0.0 and 1.0, respectively.

The network consists of five input units, seven hidden units and a single output unit. The input-output function is set by the logistic function for the hidden units and by the linear function for the output unit. The coefficient matrix [*A*] of Eq. (4) becomes a rectangular matrix of 102 rows by 50 columns.

The criterion for the successful trial is given in terms of the Euclidean norm of the error vector $\{f\}$, as is the case of exclusive *or*. The value for the successful trial is set equal to 0.032 so as to make the average of the square error per data set less than 10^{-5}. The upper limit of the learning number for a successful trial is set as ten thousands. The uniform distribution of initial weights is confined from −0.3 to 0.3 for this example.

The results of the learning, that is, the number of successes out of ten trials and the average number and time of learnings for the successful trials are listed on Table 4 for three learning algorithms. The results by changing the learning rate ε from 0.1 to 1.0 are given for the proposed whole learning algorithm, whereas the result of the fastest learning is realized by the learning rate of 0.1 and the momentum parameter of 0.9 for the conventional back-propagation algorithm. We use UltraSPARC170 for these learning.

As is anticipated, the number of successful trials is enhanced, but the learning time is prolonged

Table 4 Learning results for the Ramberg-Osgood model

Algorithm	Number of successful trials	Average learning number	Average learning time [sec]
Whole learning ($\varepsilon = 1.0$)	1	16	7.01
Whole learning ($\varepsilon = 0.5$)	8	13	6.28
Whole learning ($\varepsilon = 0.1$)	10	54	26.9
Back-propagation	10	1578	20.5
Proposed by Pal *et al.*	10	5170	80.6

by employing the smaller value of the learning rate for the proposed whole learning algorithm. The error of the first-order approximation of Eq. (2) is too large to neglect in this kind of network based on the logistic function. The expected large error of the approximation is ensured by the abrupt increase of the norm of the error vector $\| f \|$ during the successful trial by employing the learning rate ε of 1.0, as is shown in Fig. 2 by a solid line. The learning instability illustrated by the large change of the norm in the figure is due to the error of the approximation. The decrease of the norm becomes smoother and the learning is stabilized in accordance with the decrease of the learning rate as shown in Fig. 2 by the broken and dotted lines for the learning by $\varepsilon = 0.5$ and 0.1, respectively. The results in Fig. 2 are given by the trained networks of the same initial weights with those for the only successful case of $\varepsilon = 1.0$. The consistency of the comparison is kept by this manner of the same initial weights in this section.

We should consider the trade-off between the learning stability and the learning time to set the

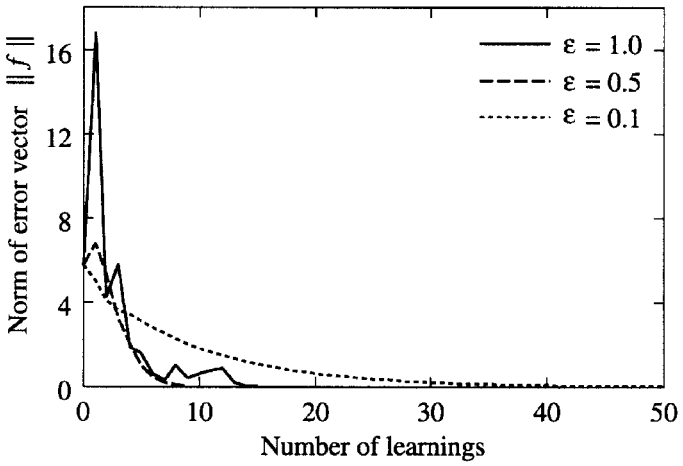

Fig. 2 Decrease of Euclidean norm of the error vector by the whole learning algorithm

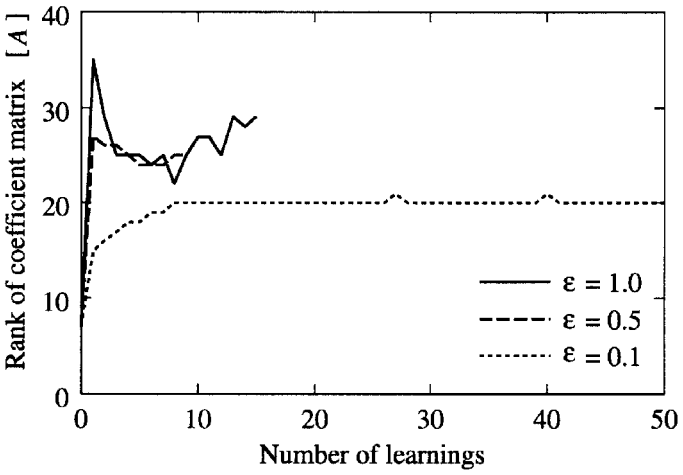

Fig. 3 Change of rank of the coefficient matrix by the whole learning algorithm

value of the learning rate for the application of the whole learning algorithm. The adequate value seems equal to 0.5 even though the average learning time for $\varepsilon = 0.1$ is not too large in comparison with the time by the conventional back-propagation algorithm. In the conventional learning algorithms, the learning efficiency is improved by employing the adaptive learning rate method, in which the learning rate is adjusted depending on the state of the learning (Weir 1991, Magoulas, Vrahatis and Androulakis 1999). The adaptive learning rate seems applicable to the proposed whole learning algorithm by changing the value of ε so as to enhance the efficiency of the learning. Stable but very slow learning is performed by the algorithm proposed by Pal *et al.* in this example, since it follows the conventional method of successive changes of the learning data sets. The algorithm proposed by them seems ineffective for this kind of problem with a large number of learning data sets.

The change of the rank of the coefficient matrix [*A*] is shown in Fig. 3. The solid line indicates the change for the successful trial by employing $\varepsilon = 1.0$, and the broken and dotted lines for the learning with the same initial weights by $\varepsilon = 0.5$ and 0.1, respectively. We can conclude that the employed method to calculate the Moore-Penrose generalized inverse is robust against the meager condition of the coefficient matrix, since all the learnings go on with the coefficient matrix of depressed rank. It seems that the larger number of rank corresponds to the higher efficiency of the learning in the proposed whole learning algorithm.

The trained networks by means of the proposed whole learning algorithm are examined in a new problem to predict the hysteresis loop of 250 MPa in stress amplitude, which is different from the hysteresis loops to generate the learning data sets. The results of the prediction by means of the trained networks, that is, the outputs from the trained networks are shown in Fig. 4 with the mathematical solution, that is, the correct answer by the Ramberg-Osgood model. The results of the prediction by the conventional learning algorithms are also shown in Fig. 5. The outputs from the trained networks by the proposed whole learning algorithm correspond with the mathematical solution as well as those of the conventional learning algorithms except in the case of $\varepsilon = 1.0$. This result also implies that the learning with the learning rate of 1.0 is not recommendable.

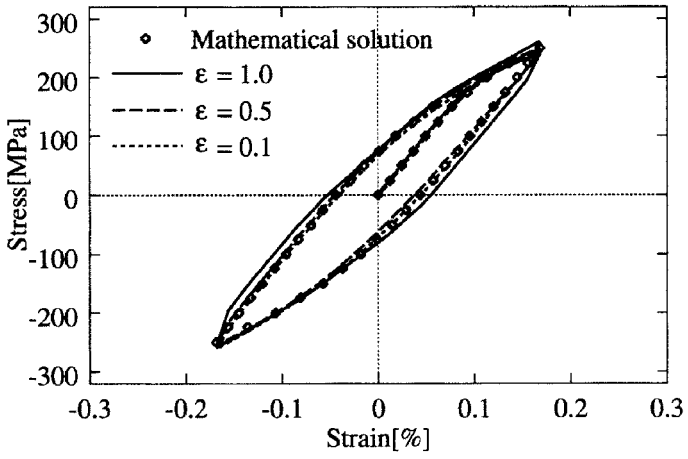

Fig. 4 Prediction results by trained networks with the whole learning algorithm

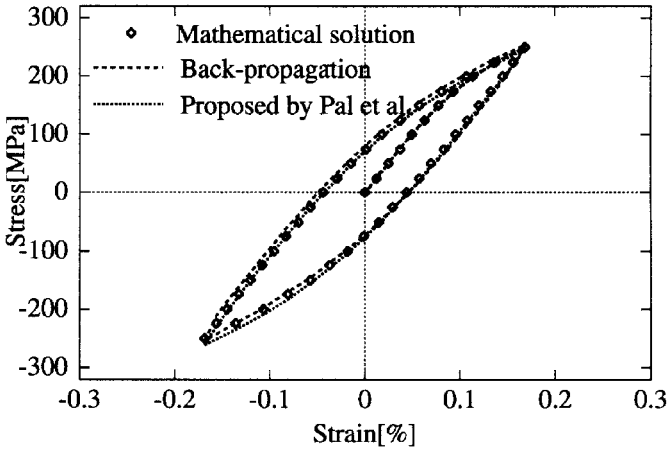

Fig. 5 Prediction results by trained networks with the conventional learning algorithms

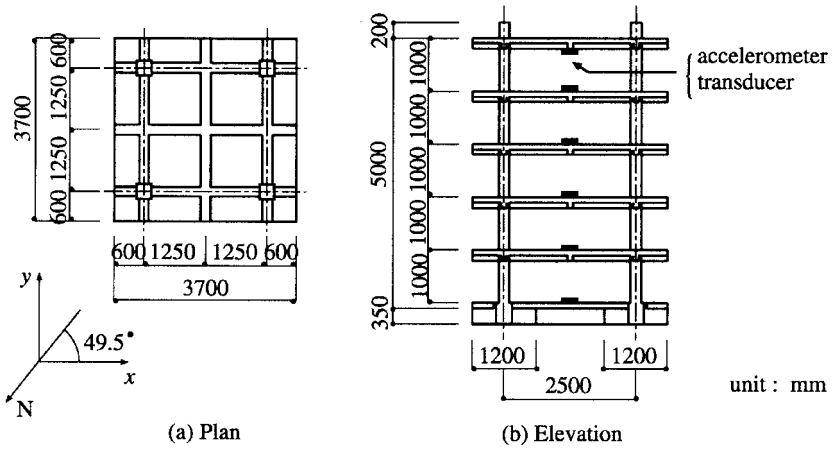

Fig. 6 Plan and elevation of the weak model structure

3.3 Nonlinear and dynamic behavior of RC members

The proposed whole learning algorithm, after investigation in the mathematical problems, is applied to a realistic one to describe the nonlinear and dynamic behavior of real RC members for the purpose of the structural reliability analysis. The behavior of the RC members is so far beyond the description by continuum mechanics that its modeling based on the feedforward neural network seems promising.

We employ the experimental data of an earthquake response of RC frame weak model structure observed at the Chiba Experimental Field of the Institute of Industrial Science, The University of Tokyo (Okada, Kumazawa and Nishida 1988). The experimental structure model is a quarter size of five-story building as shown in Fig. 6. It is designed to cause yield hinges at beam ends. The design base shear is reduced to about 50% of the real practice in Japan so that the structure is able to take damage even in a moderate earthquake. Accelerometers and transducers are placed at the center of

all the floor slabs, and the accelerations of the slabs and the inter-story displacements in the x and y directions are recorded. The dynamic behavior of the structure is represented by the relationship between the inter-story displacement and the inertia force applied to the corresponding story. The inter-story displacement means the relative displacement between the ceiling and the floor. Assuming the masses of all the floor slabs are uniform, we substitute the average of the observed accelerations from the second floor to the roof for the normalized inertia force applied to the first story.

First, we intend to describe the dynamic behavior of the first story by a three layered neural network. We employ 1,000 sets of accelerations and inter-story displacements in the x direction recorded at time intervals of 1/200 sec from the beginning of the observation under the Chiba-ken Toho-oki Earthquake on 17 December 1987 (Okada, Kumazawa and Nishida 1988). A single set of learning data for the input consists of the maximum and minimum values of the inter-story displacement and normalized inertia force, the increment of the inter-story displacement at the current time interval and the current value of the inter-story displacement. The output is set as the current value of the normalized inertia force. All the data used for the learning is normalized categorywise so as to make the minimum and maximum values equal to –0.5 and 0.5, respectively.

The network consists of six input units, ten hidden units and a single output unit. The input-output function is set by the logistic function for the hidden units and by the linear function for the output unit. The coefficient matrix [A] becomes a rectangular matrix of 1,000 rows by 81 columns. The criterion to stop the learning is given in terms of the maximum absolute value of the error between the output and the target value defined by Eq. (1). The learning stops when the maximum error becomes less than 0.085. The learning rate is set as 1.0 for the proposed whole learning algorithm in this example. We also try to train the network to simulate the behavior by means of the conventional back-propagation algorithm for the emphasis on the learning efficiency of the proposed whole learning algorithm. The learning rate and the momentum parameter are set equal to 0.1 and 0.9, respectively. We employ these values of parameters which give rise to the most efficient learnings for the both algorithms in the preliminary numerical experiments.

The completion of the learning for the first story is verified by the hysteresis loops of inertia force

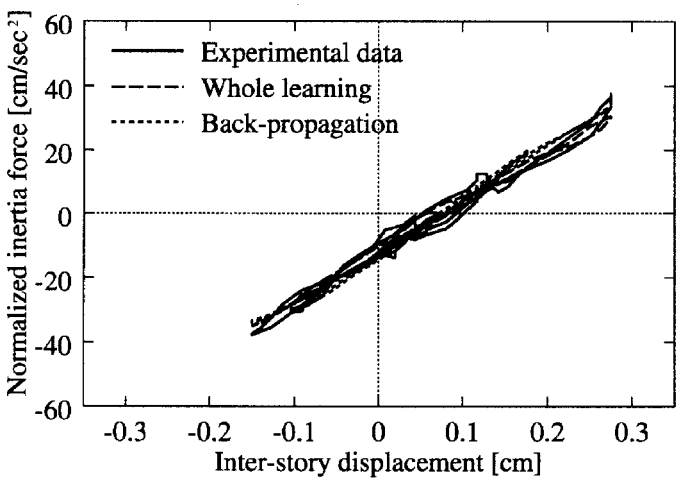

Fig. 7 Learning results for the first story

and inter-story displacement shown in Fig. 7 for both the proposed whole learning algorithm and the conventional back-propagation algorithm. The hysteresis loops from four to five seconds, which correspond from the 900th learning data set to the 1,000th one, are shown. If these learnings are performed by the Pentium 2/300 then it takes 51 seconds by 29 learnings by the proposed whole learning algorithm and 7 minutes and 56 seconds by the conventional back-propagation algorithm. The number of the learning data sets is enhanced to roughly ten times of that of the previous ones for the Ramberg-Osgood model. The increase of the learning data sets increases the computational time as much as 23 times of the previous one for the conventional back-propagation algorithm, whereas the increase of the computational time is 7.3 times for the proposed whole learning algorithm. The small increase of the computational time by the proposed whole learning algorithm results from the inventive method of weight modification based on the Moore-Penrose generalized inverse to take account of all the learning data sets. The computational time of the learning by the proposed whole learning algorithm is governed by the eigenvalue analysis of $[A]^T [A]$ or $[A] [A]^T$. The number of learning data sets is much greater than the number of weights in general, so that the magnitude of the eigenvalue analysis is limited by the number of weights regardless of the increase of the learning data sets. This is the reason why the computational time never increases in proportion to the number of learning data sets.

The final objective of this study is the prediction of the dynamic behavior of RC members by means of the trained network. We roughly look on one story of the structure as one structural member in this numerical example to learn the dynamic behavior of RC members. So the trained network in this example does not hold enough generality to predict the behavior of other structures. The prediction is not precise enough even in the case of the second story of the same structure under the identical earthquake, as is demonstrated in Fig. 8 from four to five seconds. We can recognize the remarkable difference of the stiffnesses, that is, the slopes of the hysteresis loops between the real record and the prediction. This kind of stiffness seems difficult to identify prior to the experience of an earthquake, and should be treated with uncertainty. The investigation for the contents of a learning data set is to be implemented so as to complement this kind of uncertainty in future studies.

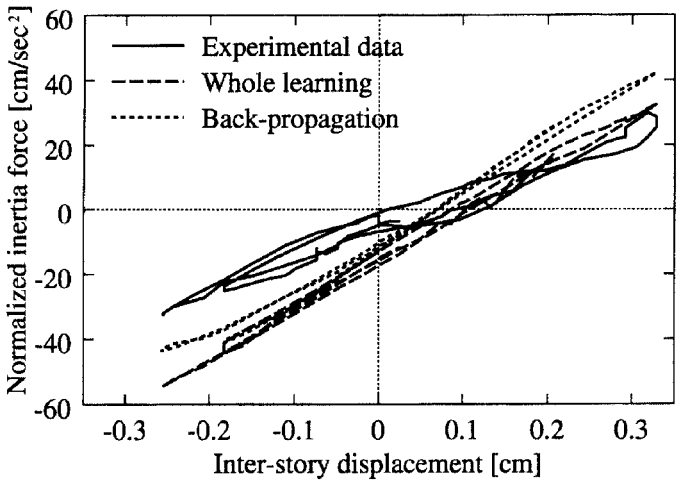

Fig. 8 Prediction results for the second story

4. Conclusions

The whole learning algorithm is newly proposed for the feedforward neural network in accordance with the multi-objective optimization technique formulated in terms of the Moore-Penrose generalized inverse. The governing equation to determine the amount of weight modification, which corresponds to the unknown variable for the optimization problem, is derived so as to make the approximated outputs equal to the target values for all the learning data sets. The approximation is in the first-order with respect to the change of the weights as is often the case with the optimization technique. The equation is summarized in the matrix form with a rectangular matrix of coefficients and solved by means of the Moore-Penrose generalized inverse to determine the solution of least squares, giving rise to the minimization of the square-sum of the error functions.

The fundamental property of the proposed algorithm is investigated through three problems different in the number of learning data sets with the proof of the learning efficiency by the proposed algorithm. The solution of the equation by the proposed algorithm is very sensitive to the approximation error of the outputs in the case of the small number of learning data sets. So the learning efficiency of the proposed algorithm is not remarkable to learn the truth table for exclusive *or*. However, the increase of the computational time caused by the enlargement of the number of learning data sets is smaller by the proposed algorithm than by the conventional algorithms. More improvement in computational time by the proposed algorithm is accounted for in the case of the greater number of learning data sets, since the computational time is governed by the analysis of the eigenvalue problem in the dimension of the fixed number of weights. The learning efficiency for the greater number of learning data sets is proven in problems to learn the material nonlinearity described by the Ramberg-Osgood model and the real dynamic behavior of RC members under an earthquake.

References

Ghaboussi, J., Garrett, Jr., J.H., and Wu, X. (1991), "Knowledge-based modeling of material behavior with neural networks", *J. Eng. Mech.*, ASCE, **117**(1), 132-153.

Hangai, Y., and Kawaguchi, K. (1991), *Shape Analysis* (in Japanese), Baifukan, Tokyo.

Hertz, J., Krogh, A., and Palmer, R.G. (1991), *Introduction to the Theory of Neural Computation*, Addison-Wesley Publishing Company, Redwood City.

Magoulas, G.D., Vrahatis, M.N., and Androulakis, G.S. (1999), "Improving the convergence of the backpropagation algorithm using learning rate adaptation"', *Neural Computation*, **11**(7), 1769-1796.

Okada, T., Kumazawa, F., and Nishida, T. (1988), "Earthquake response of reinforced concrete weak-model structures due to December 17, 1987 earthquake", Institute of Industrial Science, The University of Tokyo, Bull. ERS, (21), 67-78.

Pal, C., Kayaba, N., Morishita, S., and Hagiwara, I. (1994), "New learning method of neural network by pseudo-inverse technique (in Japanese)", *Trans. JSME C*, **60**(573), 1699-1704.

Plaut, D., Nowlan, S., and Hinton G.E. (1986), "Experiments on learning by back propagation", Technical Report CMU-CS-86-126, Department of Computer Science, Carnegie Mellon University, Pittsburgh.

Ramberg, W., and Osgood, W.R. (1943), "Description of stress-strain curves by three parameters", NACA Technical Note, **902**, 1-22.

Rao, C.R., and Mitra, S.K. (1971), *Generalized Inverse of Matrices and Its Applications*, John Wiley & Sons, Inc., New York.

Rumelhart, D.E., Hinton, G.E., and Williams, R.J. (1986), "Learning representations by back-propagating errors", *Nature*, **323**(9), 533-536.

Rumelhart, D.E., McClelland, J.L., and the PDP Research Group (1986), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations*, MIT Press, Cambridge.

Weir, M.K. (1991), "A method for self-determination of adaptive learning rates in back propagation", *Neural Networks*, **4**(3), 371-379.

Yamamoto, K. (1992), "Modeling of hysteretic behavior with neural networks and its application to non-linear dynamic analysis (in Japanese)", *J. Struct. Eng.*, JSCE, **38**, 85-94.