The Sixteenth International Symposium on Artificial Life and Robotics 2011 (AROB 16th '11),
B-Con Plaza, Beppu,Oita, Japan, January 27-29, 2011

Evaluation of an optimal design method for multilayer perceptron by using the Design of Experiments

E. Inohira and H. Yokoi

Kyushu Institute of Technology, 2-4 Hibikino, Kitakyushu, Fukuoka, Japan
(Tel: 81-93-695-6050; Fax: 81-93-695-6050)
({inohira,yokoi}@life.kyutech.ac.jp)

Abstract: We evaluate performance of an optimal design method for multilayer perceptron (MLP) by using the Design of Experiments (DOE). In our previous work, we have proposed the optimal design method for MLPs in order to determine optimal values of such parameters as number of neurons in hidden layers and learning rates. In this paper, we evaluate performance of the proposed design method through a comparison with a genetic algorithm (GA) based design method. We target at optimal design of MLPs with six layers. Moreover, we evaluate the proposed designed method in terms of calculation amount of optimization. Through the above-mentioned evaluation and analysis, we aim at improving of the proposed design method in order to obtain the optimal MLP with less effort.

Keywords: Multilayer perceptron, Neural network, Optimal design, Design of Experiments, Genetic algorithm

I. INTRODUCTION

A multilayer perceptron (MLP) can approximate an arbitrary nonlinear mapping at an arbitrary accuracy [1]. Accuracy of a trained MLP depends on two factors. The first factor is learning algorithm. Connection weights and biases of the neurons are adjusted according to the learning algorithm. Typical learning algorithm is error back-propagation (EBP) algorithm [2]. There are many learning algorithms other than EBP algorithm. It is clear that accuracy of a trained MLP depends on it learning algorithm. However, the learning algorithm is the unique factor. Another factor is design of the MLP. Before training, number of layers, numbers of neurons in hidden layers and training conditions such as learning rates are determined. Trial-and-error, brute-force approaches, network construction and pruning are used as conventional design methods. It is difficult to apply these methods to MLPs with many layers because their design parameter space becomes huge. Another problem is that the approximation accuracy of MLPs with the same design parameters has variation due to use of random number to initial values of connection weights. We need a design method with statistical analysis for MLP.

In our previous work [3,4], we proposed design method using the Design of Experiments (DOE) [5], which features efficient experiments with an orthogonal array and quantitative analysis with analysis of variance (ANOVA). We demonstrated that optimal design of five-layer MLPs could be obtained using our design method. However, we have a problem of evaluation of the proposed design method. The problem is quantitative comparison between the proposed design method and other design methods. It is clear that the proposed designed method is better than trail-and-error, brute-force approaches. We focused on a genetic algorithm based design method, which is a nonlinear optimization technique and expected to be as efficient method as the proposed method.

In this paper, we evaluate performance of the proposed design method through a comparison with a genetic algorithm based design method. We target at optimal design of MLPs with six layers. When we deal with few design parameters, a difference between the proposed design method and other method is small. Our previous work implied that accuracy of MLPs with more layers would become high for the same training data. Therefore, we should focus on MLPs with six layers. Moreover, we evaluate the proposed designed method in terms of calculation amount of optimization. We use various types of training data because performance of a trained MLP depends on training data. We refer to UCI machine learning repository [6] for the evaluation. Through the above-mentioned evaluation and analysis, we aim at improving of the proposed design method in order to obtain the optimal MLP with less effort.

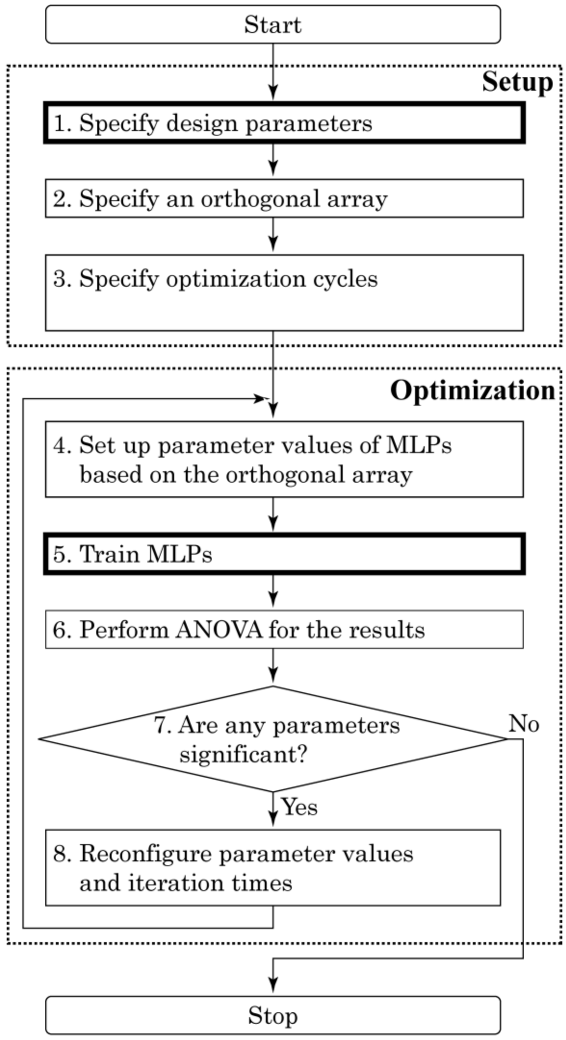

Fig.1. A flow chart of DOE-based optimal design method

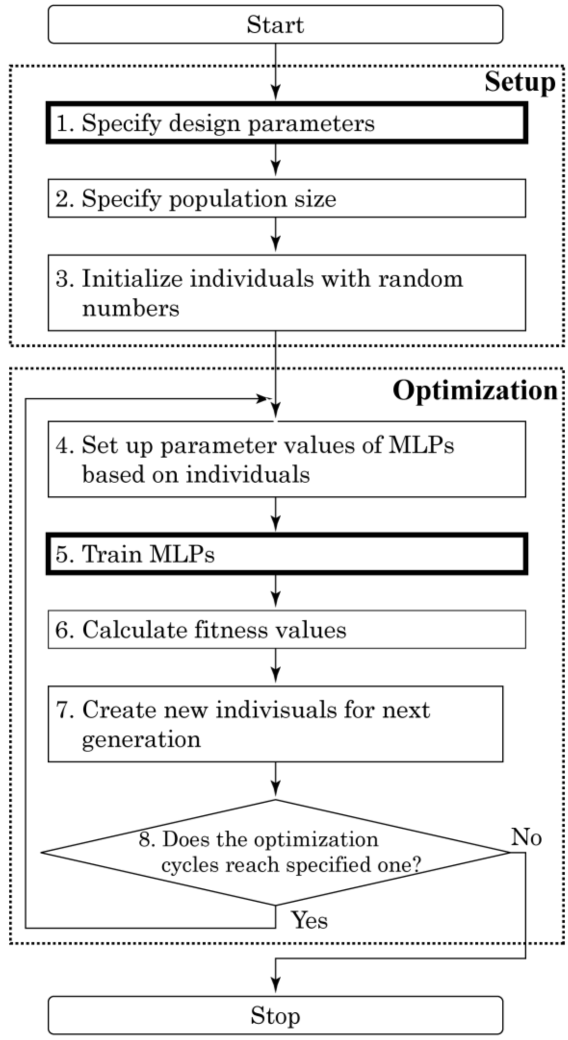

Fig.2. A flow chart of GA-based optimal design method

II. OPTIMAL DESIGN OF MLP

1. DOE-based optimal design method

We have proposed an optimal design method using DOE in our previous work [3,4]. Our basic idea is that DOE is applied to an optimal design problem of MLP. We show a flow chart of our proposed method using DOE in Fig.1. Detailed explanation has been described in [4]. Training of MLPs corresponds to experiments in DOE. For example, we used the number of hidden nodes, learning rates, momentum coefficients and range parameters of initial connection weights as the design parameters in [4]. And we used squared sum of training errors after specific learning cycles as the performance index. The used training dataset is common in all experiments.

We implemented our optimal design process on MATLAB programs except training of MLP. In [4], Step 8 needed manual operation. On the other hand, in [5] and this paper, Step 8 is programmed with predefined rules.

2. GA-based optimal design method

We have proposed the optimal design method using DOE and evaluated it on three-layer MLPs and five-layer MLPs. We have demonstrated our method and not compared with other methods. We focused on genetic algorithm (GA). GA-based approaches [7,8] have already been proposed. However, our proposed method and GA-based approaches cannot be compared directly because selected design parameters are different.

In order to compare our proposed method and a GA-based approach, we prepared GA-based optimization programs using Global Optimization Toolbox on MATLAB. We show a flow chart of optimal design method using GA in Fig.2. We programmed GA-based optimization programs as similar to our proposed method using DOE as possible. Step 1 to specify design parameters and Step 5 to train MLPs are common in both optimization programs. The remaining part of the optimization depends on the optimization algorithms, that is, DOE or GA. Our GA-based optimization program can deal with the same design parameters and training data set as the DOE-based program.

We used a simple GA-based optimization algorithm. An individual consists of design parameter values in real numbers. A value of the used fitness function is a training error of a MLP. The default crossover function of the Global Optimization Toolbox is used.

III. Experiments

1. Training dataset

We used concrete compressive strength data [9,10] in UCI machine learning repository as the training dataset. The training dataset has 8 input variables and 1 output variable. Therefore, input and output of a MLP are determined by specifying the training dataset. The number of instances is 1030. Ranges of the variables are largely different in this dataset. We normalized each variable to be range -1 from +1 for input and -0.99 and +0.99 for output. We used a sigmoid function ranging from -1 to +1. When an input of the sigmoid function is infinity, its output becomes +1. This takes very long learning time. To avoid this situation, we set range of output variable to be from -0.99 to +0.99.

2. Targeted MLP and design parameters

We used six-layer MLPs for the target system. Their design parameters are as follows: the number of nodes in hidden layers (4 parameters) and learning rate in each layer (5 parameters). Range of the number of nodes in hidden layers is from 1 to 50 in order to limit the parameter space. Range of the learning rate in each layer is from 0 to 1. On the optimization program, the design parameters are normalized in range from 0 to 1. When MLPs is set up before training, the design parameter values are restored. In DOE, we used an orthogonal array with 27 combinations and 3 levels.

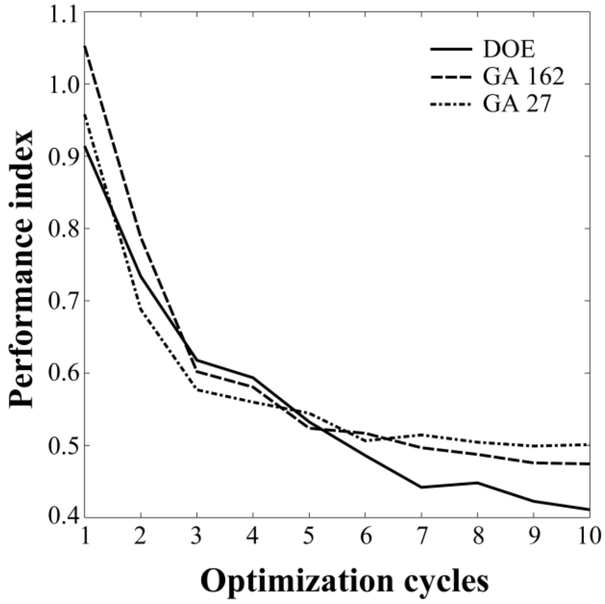

Fig.3. A comparison between DOE-based method and GA-based method in median of performance indices

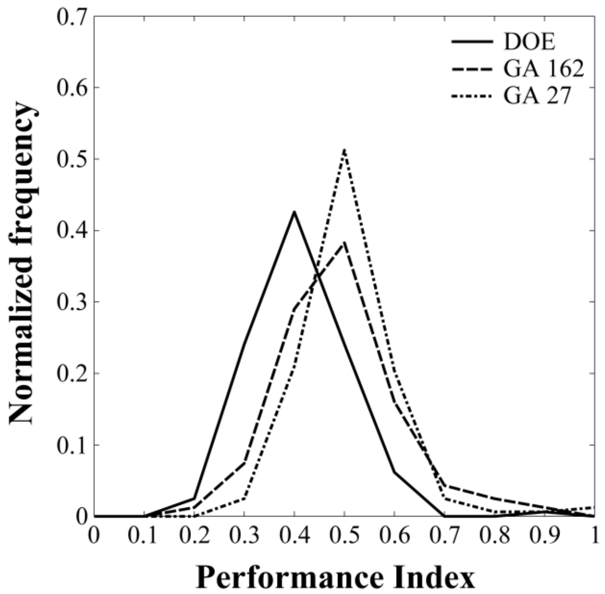

Fig.4. A comparison between the two methods in distribution of performance indices at the tenth cycle

Therefore, 27N MLPs are trained each optimization cycle. Here N denotes trial iteration counts and is adjusted automatically in the optimization. In GA, number of individuals is set to 27 or the maximum number of 27N because of comparison with the DOE-based optimization. This means that calculation amount is the same in both methods. When number of individuals is 27, training of MLPs is performed N times under different initial connection weights. In this case, value of fitness function is average of those results.

3. Training algorithm and performance index

We used a back-propagation algorithm to training MLPs. In this paper, the learning rates are prepared every layer. This approach has an effect to reduce the training error as shown in our previous work [4]. We used common logarithm of squared sum of error for all instances as the performance index. Small performance index is better because small error is good. We also evaluated distribution of performance indices at specific optimization cycles instead of best performance index. The reason is that the best performance index has large variance. We used median of the performance indices as representative value.

IV. EXPERIMENTAL RESULTS

We show experimental results in Figs. 3 and 4. Fig.3 shows a comparison between DOE-based optimization method and GA-based method in median of performance indices at each optimization cycle. In both methods, median of performance indices decreases when optimization proceeds. After the sixth cycle, the DOE-base method is better than the other. Fig.4 shows distributions of performance indices of the two methods at the tenth cycle. The distribution in the DOE-based method is on the left of the distribution in the GA-based method. This means a set of combinations of parameter values in the DOE-based method is better than the GA-based method. In other words, probability of getting the best performance index in the DOE-based methods is higher than the GA-based method.

V. DISCUSSION

We showed that the DOE-based method is better than the simple GA-based method in optimal design of MLP. The problem of the simple GA-based method is to ignore variance of performance indices. In Fig.4, the result of GA 27, which means number of individuals is 27, is worst. The reason is that mean of *N* times results is used in GA 27. Calculation amount is wasted by using mean because variance is ignored. GA 162 is worse than the DOE-based method under the same calculation amount. The reason is that the GA-based method has no statistical analysis. In the GA-based method, children of a good individual are not always better because performance indices have variance. On the other hand, the DOE-based method can adjust parameter values properly without misleading due to ANOVA.

In this paper, a kind of MLP and training dataset was used. It is necessary to apply the optimal design methods to many types of problems in order to clarify their limitation

VI. CONCLUSION

We evaluated performance of an optimal design method for MLP by using DOE through a comparison with a GA-based design method. We demonstrated optimal design of MLP with six layers by using DOE-base method is better than GA-based method under the same calculation amount.

In the future work, our proposed optimal design method will be applied to other types of problem such as classification and control in order to investigate applicable scope of our method.

REFERENCES

[1] Tikk D, Koczy T and Gedeon TD (2003), A sur vey on universal approximation and its limits in sof t computing techniques, Intl J of Approximate Reas oning 185-202

[2] Rumelhart D, Hinton G and Williams R (1986), Parallel Distributed Processing, MIT Press.

[3] Inohira E and Yokoi H (2007), An optimal design method for artificial neural networks by using the design of experiments, J of Advanced Computational Intelligence and Informatics. 11(6):593-599.

[4] Inohira E and Yokoi H (2010), Development of an optimal design method for multilayer perceptrons by using the design of experiments (in Japanese), Proc of the 2010 IEICE general conference. D-2-1.

[5] Dean A and Voss D (1999), Design and analysis of experiments.

[6] Frank A and Asuncion A (2010), UCI Machine Learning Repository [http://archive.ics.uci.edu/ml], Irvine, CA: University of California, School of Information and Computer Science.

[7] Castullo PA, Merelo JJ, Prieto A, Rivas V and Romero G (2000), G-Prop: global optimization of multilayer perceptrons using Gas, Neurocomputing, 35:149-163

[8] Leung FHF, Lam HK, Ling SH and Tam PKS (2003), Tuning of the structure and parameters of a neural network using an improved genetic algorithm, IEEE Trans Neural Networks 4(1):79-88

[9] Yeh IC (2007), Modeling slump flow of concrete using second-order regressions and artificial neural networks, Cement and Concrete Composites 29(6):474-480.

[10] Yeh IC (1998), Modeling of strength of high performance concrete using artificial neural networks, Cement and Concrete Research 28(12):1797-1808