Evolutionary Engineering Optimization Using Recursive Regional Neural Network and Genetic Algorithm

Jyh-Cheng YU* and Yu-Lung Tseng
Department of Mechanical and Automation Engineering
National Kaohsiung First University of Science and Technology
2, Juoyue Rd., Nantz District, Kaohsiung 811, Taiwan, R.O.C.
*jcyu@ccms.nkfust.edu.tw**

Abstract

This study presents a soft computing based evolutionary optimization for engineering applications with the constraint of sample size. Existing field data or experimental designs are often applied as training samples to establish a simulated network model for the engineering system following by an optimum search. However, possible biased distribution of field data and scarce samples from OA experiments might compromise modeling generality. The proposed methodology defines the Reliable Radius to confine the genetic algorithm search in the hyper-spheres surrounding the training samples for a reliable quasi-optimum. The verification of the optimum is added to the learning samples to retrain the regional network model that evolves intelligently according to the prediction accuracy using a fuzzy inference. Instead of a dense sample distribution to increase global accuracy, the design iteration will provide additional samples in the most probable regions of the optimum, and thus increase sampling efficiency.

1. Introduction

Simulated Evolutionary Optimization [1], such as Genetic Algorithms (GA), is an effective searching tool for optimization even for nonlinear, discontinuous, and non-differentiable problems. By maintain a population search, GA is less likely to get trapped in local optima. However, evolution of populations will require numerous evaluations of design fitness, which limits the direct application to engineering optimization where simulation and experimental costs are expensive. The combinations of a simulated neural network and evolutionary optimization have thus attracted much research attention [2][3]. An artificial neural network (ANN)[4] model could be established from a set of training data to provide estimations for system responses. Instead of a direct interaction with the engineering system, the optimum search is applied to the simulated model to increase the searching quality and reduce the experimental cost.

The size and the distribution of training data are essential to the prediction accuracy of a simulated network model. The training samples are usually existed field data or planned experiments such as orthogonal arrays (OA) [5]. However, possible bias distribution of field data will decrease the prediction accuracy of simulated models. Though OA experiments provide a smaller and even sample distribution, scarce training data might result in the lack of model generality for a complex problem. Also, because ANN is mainly based on fitting methods to learning data, the prediction reliability will be related to the distance between the point of interest and the nearest learning sample [6] and the estimation error of the learning sample [7].

This study proposes a novel optimization methodology based the iterative constrained search using genetic algorithm in the evolutionary regional network model that is defined as the union of the neighboring space of the training sample. The regional model will evolve intelligently based on the fuzzy inference of the prediction accuracy. A numerical example is provided to demonstrate the effectiveness of the proposed scheme.

2. Regional neural network model from limited training samples

2.1 The prediction accuracy of the network model from scarce training samples

* Corresponding author

0-7695-2882-1/07 $25.00 ©2007 IEEE

The prediction accuracy of the network model is closely related to the number of training samples. For a neural network trained from scarce training data of a complex system, experiences tell that the prediction accuracy is getting worse if the predicted design is farther away from the training samples. Nine random learning samples and four testing samples are selected to set up a Back Propagation Network (BPN) model of the Peaks function as Eq.(1) whose contour plot is shown in Figure 1.

$$z = 3(1-x)^2 e^{-x^2-(y+1)^2} - 10(\frac{x}{5} - x^3 - y^5)e^{-x^2-y^2} - \frac{1}{3}e^{-(x+1)^2-y^2} \quad (1)$$

Due to limited number of training samples, significant errors present for the simulated BPN model especially in the regions farther away from the training samples. Based on 180 randomly selected points in the investigating region [-2, +2] of the simulated model, the empirical rules are derived as follows:

(1) The prediction accuracy is worse for a design farther away from the training samples.
(2) The prediction accuracy of an interpolation design is better than an extrapolation designs.

2.2 The sampling distance

The Sampling Distance, r_{ij}, is proposed as a neighboring index between a predictive design, P_i, and the sample S_j, which is defined as the mean Euclid distance:

$$r_{ij} = \left[\frac{1}{n} \sum_{k=1}^{n} \left(P_{ik} - S_{jk} \right)^2 \right]^{0.5} \quad (2)$$

where n represents the number of variables. To prevent the scaling problem, continuous variables x_k are first normalized to z_k to transform all the dimensional entries of the training samples into the space of [-1, +1]. For discrete variables, the factorial values are assigned equally spaced between -1 and +1.

2.3 The reliable regions of a simulate network model

As the predictive errors of the training samples will be controlled to an acceptable level, the neighboring space of the training samples is likely as reliable as the training samples. This study proposes the Reliable Interpolating Radius (*RIR*) and the Reliable Extrapolating Radius (*RER*) to define the reliable regions of a simulated network model as shown in Figure 1. If the minimum Sampling Distance of a predictive design is not larger than the reliable radii, the prediction is assumed reliable. The determination of the reliable radii will depend on the model complexity and prediction accuracy, and is a key issue in the following study.

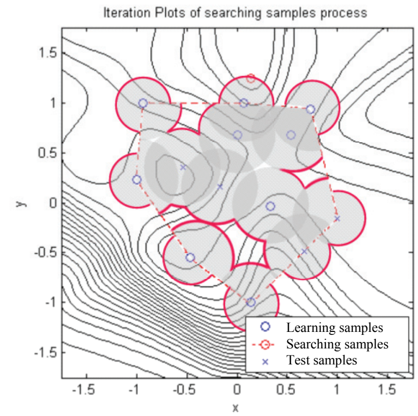

Figure 1. The reliable regions (*RIR* = 0.3, *RER* = 0.2) of a simulated network model

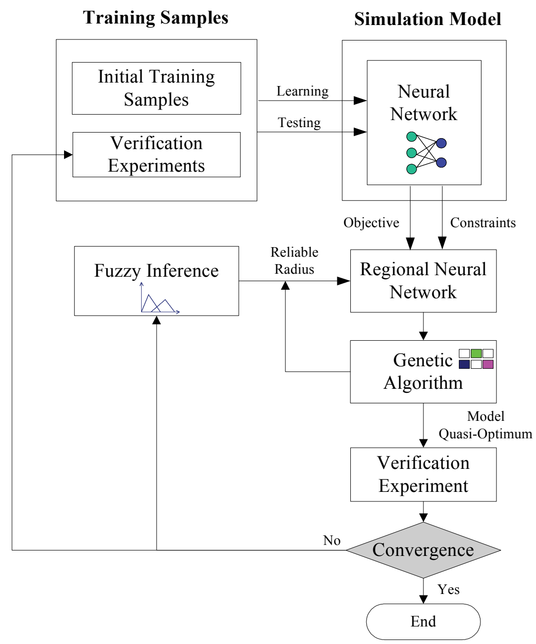

Figure 2. The optimization flowchart of ERNGA.

3. Evolutionary regional neural network with genetic algorithm

The searching flowchart of the proposed optimization scheme, Evolutionary Regional Neural network with Genetic Algorithm, (ERNGA), is shown in Figure 2. A simulated network model is established from the initial training data. The proposed scheme confines the GA search to the reliable regions of the network model for a quasi-optimum. The reliable regions are the union of the hyper-spheres defined by the *RIR* and the *RER* surrounding the training data.

The verification of the quasi-optimum provided by the GA search is introduced to retrain the neural network and to adjust the Reliable Radii using a fuzzy inference. The reliable regions of the simulated network will then continue to evolve from the accumulation of training samples and self-learning mechanism of the Reliable Radii. The optimum search iterates until the convergence of optimum.

3.1 The fuzzy inference for the reliable radii

The selections of *RIR* and *RER* will depend on the model generality. The verification of the regional optimum will become a feedback mechanism to adjust the Reliable Radii based on the following fuzzy concepts. If the verification result is close to the model prediction, increase the Reliable Radii to expand the reliable regions; otherwise, decrease the Reliable Radii. This section proposes a heuristics based fuzzy inference to intelligently evolve the Regional Neural Network Model as follows:

R1: If Extrapolating design and *MEI* is [Small] then [Slightly Increase] *RER* and [Increase] *RIR*.
R2: If Extrapolating design and *MEI* is [Medium] then [Maintain] *RER* and [Slightly Increase] *RIR*.
R3: If Extrapolating design and *MEI* is [Large] then [Slightly decrease] *RER* and [Maintain] *RIR*.
R4: If Interpolating design and *MEI* is [Small] then [Maintain] *RER* and [Slightly increase] *RIR*.
R5: If Interpolating design and *MEI* is [Medium] then [Maintain] *RER* and [Slightly decrease] *RIR*.
R6: If Interpolating design and *MEI* is [Large] then [Slightly decrease] *RER* and [Decrease] *RIR*.

Three linguistic levels are defined to describe the condition variable of prediction error: Large, Medium, and Small based on the Modeling Error Index, (*MEI*) in Eq.(3). Five action levels are defined to describe the Adjustment Factor (*AF*) for the Reliable Radii: Increase, Slightly Increase, Maintain, Slightly Decrease, and Decrease.

$$MEI = \frac{|Y_j - T_j|}{RMSE_{Test}} \quad (3)$$

where Y_j is the model prediction, T_j is the verification result of the quasi-optimum at iteration *j*, and $RMSE_{Test}$ is the root mean squared error of the testing samples.

The adjustment factors from the fuzzy inference are used to modify the reliable radii as shown in Eq. (4) and (5) for the next iteration, and dynamically adjust the reliable regions.

$$RIR_{i+1} = AF_i \times RIR_i \quad (4)$$
$$RER_{i+1} = AF_e \times RER_i \quad (5)$$

Therefore, the searched result will be less sensitive to the imperfection of the trained model. If the verification result is good, the reliable regions will expand in the next iteration to investigate more possible regions; otherwise, the reliable region will retract for a more conservative search.

4. Numerical examples

4.1 Optimum search of the peaks function

L9 OA is selected for the learning samples, and L4 OA is selected for the testing samples in the initial investigating ranges of $x, y \in [-2,+2]$ for the peaks example. Figure 3 show that the iterations converge smoothly to the theoretical optimum although the accuracy of the initial NN model is poor due to a bad distribution in the flat region for initial samples. As the addition of the learning samples from the searched optima and the self-learning mechanism of the reliable regions, the accuracy of the simulated model improves, especially in the most probable regions of the global optimum as shown in Figure 4. ERNGA will congregate additional samples in the most probable regions of the design optimum without wasting costly experimental resources in unlikely regions.

If the constraint of the reliable regions is relieved, GA will assume global accuracy, and search for the design with the best fitness based on the model prediction in the investigating range. The prediction accuracy of the optimum is poor because of the lack of generality for the network model. The iteration is very unstable and shows no convergence tendency at the 45th iteration as shown in Figure 5. The search result is very sensitive to the global accuracy. Also, unlike ERNGA, additional samples from the iteration of conventional NN and GA may scatter all over, and are thus less efficient.

4.1.1 Comparison of Results

ERNGA converges to the optimum with relative error of 0.7% at 24th iteration. AS to the conventional NN and GA, even though the best result of the iteration is happened to be close to the global optimum with the relative error of 1.6%, the iteration is very unstable and shows no convergence tendency at the 45^{th} iteration. A continuous discrepancy presents between the predicted optimum and the verified results for the conventional NN and GA iteration due to the over-confidence on the global accuracy of the simulated models. On the other hand, the fuzzy inference of reliable radii in ERNGA constrains the

GA search and provides a reliable quasi-optimum. The evolutionary regional network model expands intelligently to the most probable regions of global optimum.

5. Conclusions

The integration of a simulated neural network from sampling data following by an optimum search using genetic algorithm has shown a promising tool in engineering applications. However, if the generality of prediction accuracy is compromised due to limited number or possible biased distribution of training samples, the iteration process may be unstable and inefficiency. Certainly, a well-trained network will have better prediction accuracy and thus reduce the number of iteration in the optimum search, but the optimization of the simulated model is no guarantee in engineering applications. ERNGA reduces the sensitivity of the searched optimum to the trained generality of the network model, and provides additional samples in the most probable regions to increase sampling efficiency, which is particularly important in engineering applications.

6. Acknowledgments

The authors would like to thank the National Science Council of the Republic of China for financially supporting this research under Contract No. NSC91-2212-E-155-007.

7. References

[1]. D.B. Fogel, "A Introduction to Simulated Evolutionary Optimization", *IEEE transductions on Neural Networks*, vol. 5, No. 1, 1994.

[2]. H. Oktem, T. Erzurumlu, and F. Erzincanli, "Prediction of minimum surface roughness in end milling mold parts using neural network and genetic algorithm," *Materials and Design*, v 27, 735–744, 2006.

[3]. S. Sette, L. Boullart, and L. V. Langenhove, "Optimising a Production Process by a Neural Network/Genetic Algorithm Approach," *Engineering Applications in Artificial Intelligence*. Vol. 9, No. 6, 681-589, 1996.

[4]. J. S. Jang, C. T. Sun and E. Mizutani, *Neuro-Fuzzy and Soft Computing: a computational approach to learning and machine intelligence*, Prentice-Hall, 1997.

[5]. G. Taguchi, "Performance Analysis Design." *Int. Journal of Production Research*, 16 (1978) 521-530.

[6]. J. Yu, X. Chen, T.R. Hung and F. Thibault, "Optimization of Extrusion Blow Molding Processes Using Soft Computing and Taguchi Method", *Journal of Intelligent Manufacturing,* 15, (2004), 625-634.

[7]. T. Hanai, N. Iwata, T. Furuhashi, H. Honda and T. Kobayashi, "Proposal of Reliability Index in Search for Reliable Solution of Reverse Calculation Based on Fuzzy Neural Network Modeling", *Journal of Chemical Engineering of Japan*, Vol. 37, No. 4, 523-530, 2004.

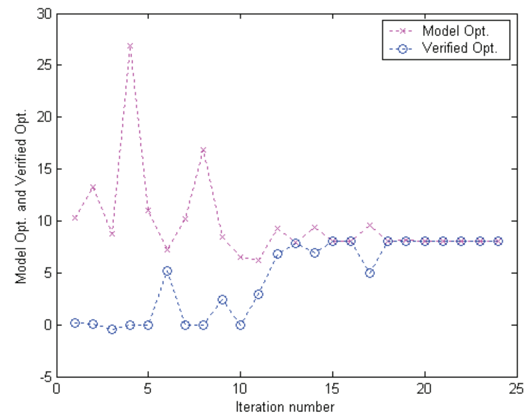

Figure 3. ERNGA iteration for the Peaks problem

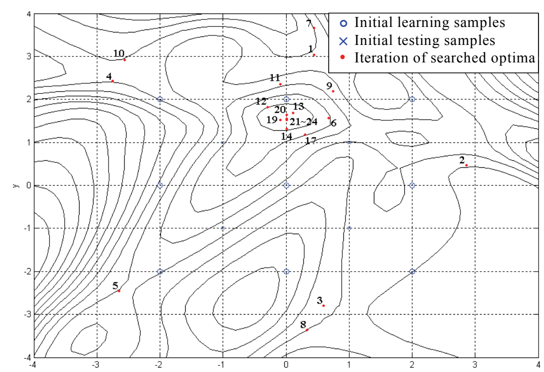

Figure 4. The contour plot of the simulated model and the distribution of the searched optima using ERNGA

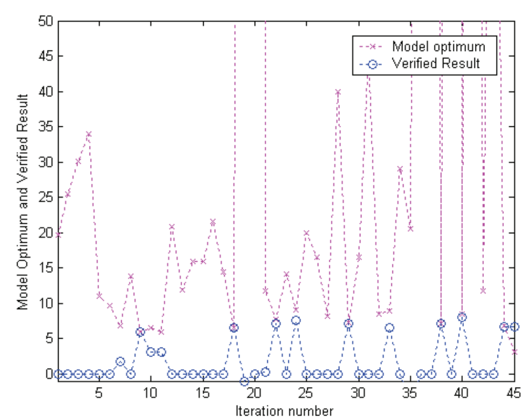

Figure 5. Conventional NN and GA iteration using for the Peaks problem