2nd International Conference on Autonomous Robots and Agents
December 13-15, 2004 Palmerston North, New Zealand

Nonlinear System Modeling Using GA-based B-spline Membership Fuzzy-Neural Networks

Yih-Guang Leu
Department of Electronic Engineering, Hwa-Hsia Institute of Technology
No.111, Gong Jhuan Rd., Chung Ho City, Taipei, Taiwan, 235, R.O.C.
leuyk@cc.hwh.edu.tw

Abstract

In this paper, we investigate the optimization problem of B-spline membership fuzzy-neural networks. When the B-spline membership fuzzy-neural networks are used for complex nonlinear system modeling, there are some problems, such as how to select the appropriate knot positions, and how to choose control points optimally. These problems are significantly important in achieving good approximation. The unsuitable knot positions and the unsuitable control points often cause the poor performance of B-spline membership fuzzy-neural networks. So far, there is less theory about how both knot points and control points can be chosen optimally. Since the weighting factors, the knot positions, and the control points are considered to be variables, it becomes a highly nonlinear optimization problem. Therefore, we propose a genetic algorithm (GA) to simultaneously optimize these variables. Also, this algorithm can possess the capability of escaping from local minima. For the purpose of illustrating effectiveness of the proposed method, an example of nonlinear systems is simulated.

Keywords: B-spline functions, Fuzzy-neural networks, Nonlinear system modeling, Genetic algorithm

1 Introduction

Since neural networks and fuzzy logic systems are universal approximators [1,2], nonlinear system modeling via these approximators has widely been developed for many practical applications [3,4]. Moreover, many researches [4-7] combining fuzzy logic with neural networks have been developed to improve the efficiency of nonlinear system modeling. The approximators can be expressed as a linear combination of basis functions. An appropriate choice of basis functions is the B-spline. The B-spline function is a piecewise polynomial mapping.

In B-spline fuzzy-neural network structure [3,7], the B-spline membership functions are assumed to be fixed and only control points are adjusted during the learning process. Before the learning process, the designer has to specify the knots of the B-spline membership functions. Because the selection of the appropriated knots of the B-spline membership functions is crucial to obtaining good approximation, it is an important issue for engineering problem. Therefore, the knots must be treated as variables. Then the problem becomes a complex nonlinear and multivariable optimization problem with many local optima. Thus, it is difficult to obtain a global optimum. Recently, some researchers have been trying to use stochastic approaches to solving such problems. For example, simulated annealing [8] and genetic algorithm [9] are stochastic approaches. These algorithms possess the capability of finding the global optimum solutions.

Since B-spline functions possess characteristics of easy local adjusting, simple calculation and implementation, they have widely been used in graphic process [10,11], control [12,13], modeling [3,7,8,14,15], and so on. A deterministic iterative approach to adaptive estimation of parametric deformable contours based on B-spline representations has been developed in [10]. In [12], the actual experiments for a DC motor speed control system have been presented. In [3], the B-spline membership functions have been constructed and applied successfully the fuzzy-neural modeling. Then, the B-spline membership fuzzy -neural networks have been extended to on-line nonlinear control in [13]. When the B-spline network is not approximating a given function, the authors in [14,15] add more knot points uniformly throughout the domain of interest until the approximation is satisfactory. In [8], a novel knot-optimizing B-spline network has been proposed to approximate nonlinear system behavior.

Since traditional optimization methods are difficult to search a global minimum of a multilocal minima nonlinear function, some stochastic approaches have attracted much attention. In [16,17], the authors presented their algorithms mixed by some stochastic approaches in order to increase the efficiency of algorithms. An annealing-genetic algorithm for solving NP-Hard problems has been proposed in [16]. In [17], a stochastic approach mixed by simulated annealing algorithm, genetic algorithm, and chemotaxis algorithm has been presented for solving complex optimization problems..

In this paper, we investigate the optimization problem of B-spline membership fuzzy-neural networks. Making the problem-optimized structure consists of two parameter search problems. The first is to obtain the weighting factors of fuzzy-neural networks. The second is to construct B-spline functions, including the knot positions, and a set of control points. Since the weighting factors, the knot positions, and the control points are considered to be variables, it becomes a highly nonlinear optimization problem. Thus, the objective is to propose a stochastic optimization algorithm to simultaneously optimize these variables. Also, this algorithm can possess the capability of escaping from local minima. Because genetic algorithms have been theoretically and empirically proven to provide the efficient search for many highly nonlinear problems, they offer a good chance of success. Here, the genetic algorithm is used as the stochastic optimization algorithm..

2 Fuzzy-Neural Networks

The basic configuration of fuzzy logic systems consists of some fuzzy IF-THEN rules and a fuzzy inference engine. The fuzzy inference engine uses the fuzzy IF-THEN rules to perform a mapping from an input linguistic vector $\mathbf{x} = [x_1 x_2 \cdots x_n]^T \in \Re^n$ to an output linguistic variable $y \in \Re$. The ith fuzzy IF-THEN rule is written as

If x_1 is A_1^i and ... and x_n is A_n^i

Then y is B^i

where $A_1^i, A_2^i, \ldots, A_n^i$ and B^i are fuzzy sets. Let h be the number of the fuzzy IF-THEN rules. By using product inference, center-average and singleton fuzzifier, the output of the fuzzy logic system can be expressed as

$$y(\mathbf{x}) = \frac{\sum_{i=1}^{h} w^i \left(\prod_{j=1}^{n} \mu_{A_j^i}(x_j) \right)}{\sum_{i=1}^{h} \left(\prod_{j=1}^{n} \mu_{A_j^i}(x_j) \right)} \qquad (1)$$

where $\mu_{A_j^i}(x_j)$ is the membership function value of the fuzzy variable x_j, h is the number of the total IF-THEN rules. The weighting factors w^p, $p = 1,2,\ldots,h$ are adjustable parameters. Fig. 1 shows the configuration of fuzzy-neural networks. The system has a total of four layers. Nodes at layer I are input nodes that represent input linguistic variables. Nodes at layer II are term nodes which act as B-spline membership functions to represent the terms of the respective linguistic variables. Each node at layer III is a fuzzy rule. Layer IV is the output layer.

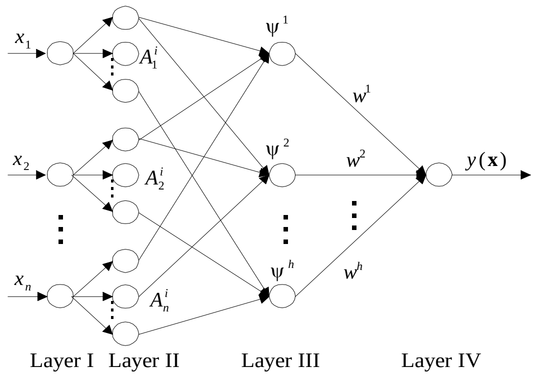

Fig. 1. The configuration of fuzzy-neural networks

2.1 B-spline membership functions

A B-spline function is a piecewise polynomial. Let $T = \{t_0, t_1, \ldots, t_{r+\alpha}\}$ be the knot vector, where the t_k are calls knots with $t_0 \le t_1 \le \cdots \le t_{r+\alpha}$. The kth B-spline basis function of order α, denoted by $N_{k,\alpha}$, is defined as

$$N_{k,1} = \begin{cases} 1 & \text{if } t_k \le t < t_{k+1} \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

and

$$N_{k,\alpha}(t) = \left(\frac{t - t_k}{t_{k+\alpha-1} - t_k}\right) N_{k,\alpha-1}(t) + \left(\frac{t_{k+\alpha} - t}{t_{k+\alpha} - t_{k+1}}\right) N_{k+1,\alpha-1}(t) \qquad (3)$$

For $r+1$ control points, the B-spline function $s(t)$ is defined as

$$s(t) = \sum_{k=0}^{r} c_k N_{k,\alpha}(t) \,. \qquad (4)$$

According to [3], the B-spline membership function $\mu_A(x_j)$ is defined as

$$\mu_A(x_j) = \sum_{k=0}^{r} c_k N_{k,\alpha}(x_j) \qquad (5)$$

where x_j is the input data and A is a fuzzy set. Fig. 2 shows illustration of B-spline membership function. The B-spline membership function has 7 control points, 9 knot points, and order 2.

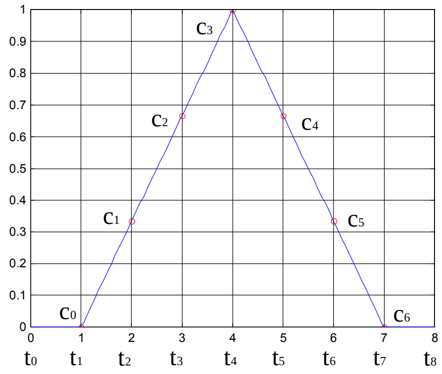

Fig. 2. Illustration of the B-spline membership functions

3 The proposed genetic algorithm

In order to solve the optimized B-spline membership fuzzy-neural networks, we assume the *i*th rule and *j*th B-spline membership function $\mu_{A_j^i}(x_j)$ has r+1 control points. Then, the adjusted variables include the weighting factors w^p, $p=1,2,\ldots,h$, the control points c_{kj}, $k=0,1,\ldots,r$, $j=1,2,\ldots,n$, and the knot points t_{qj}, $q=0,1,\ldots,r+\alpha$, $j=1,2,\ldots,n$. Note that because the equality $t_{qj}=t_{(q+1)j}$ is allowed, the redundant knots will be avoided. Hence, the objective of the search algorithm is to minimize the error function $\mathrm{E}(\mathbf{w},\mathbf{c},\mathbf{t})$, where $\mathbf{c}=\{c_{kj} \mid k=0,1,\ldots r, j=1,2,\ldots,n\}$, $\mathbf{t}=\{t_{qj} \mid q=0,1,\ldots r+\alpha, j=1,2,\ldots,n\}$, and $\mathbf{w}=\{w^p \mid p=1,2,\ldots,h\}$. The error function is define as

$$\mathrm{E}^2(\mathbf{w},\mathbf{c},\mathbf{t})=\frac{1}{m}\sum_{l=1}^{m}(y_l-y_l^*)^2 \qquad (6)$$

where m is the number of the training data pairs, and y_l and y_l^* represent the outputs and the desired outputs respectively. Define a chromosome as $\mathbf{z}^l=[\mathbf{w}^T\,\mathbf{c}^T\,\mathbf{t}^T]=[z_1^l z_2^l \cdots z_{\kappa-1}^l z_k^l]$, where a set of the weighting factors $\mathbf{w}$ range from within the interval $D_1=[w_{\min},w_{\max}]\subseteq R$, a set of the control points $\mathbf{c}$ range from within the interval $D_2=[c-\eta,c+\eta]\subseteq R$, $\eta\geq 0$ and a set of the knot points $\mathbf{t}$ range from within the interval $D_3=[t-\varepsilon,t+\varepsilon]\subseteq R$, $\varepsilon\geq 0$. The t and c are the initial values of the knot points and the control points. Besides, define a fitness function as

$$\text{fitness}=\frac{1}{1+\mathrm{E}^2(\mathbf{w},\mathbf{c},\mathbf{t})} \qquad (7)$$

The proposed genetic algorithm performed shown as Fig. 3. The detail description is as follows:

```
Genetic_Algorithm()
{
Initialize the Population_of_Chromosomes;
Calculate the Fitness_Function;
While (not terminate-condition)
{
Perform Selection with Sorting;
Perform Crossover According to the Sorted Population;
Perfom Mutation According to the Sorted Population ;
Calculate the Fitness_Function;
}
```

Fig. 3. The proposed genetic algorithm.

The initialization procedure begins with the initialization of the chromosomes in the population. Each chromosome is coded as an adjustable vector with floating point type components. During the initialization step, the initial values of chromosomes are randomly created in some intervals. During the selection process, the population is first sorted by ranking the fitness of chromosomes. In particular, the first chromosome of the sorted population has the highest fitness value (or smallest error). Then, based on the sorted population, the selection process retains a best individual in the current generation unchanged for the next generation. After the selection process, the crossover procedure selects randomly subparts from two parent chromosomes and creates a new offspring chromosome. Here, the two parent chromosomes are selected according to the sorted population. In particular, a pair of crossover chromosomes is first selected if they have better fitness values. During the mutation process, certain components in some randomly selected chromosomes may be randomly replaced by new components. Moreover, some chromosomes with worse fitness values may be completely replaced, and the probability of the complete replacement is based on the sorted population.

4 Simulation results

An example is illustrated to show the effects of the B-spline membership fuzzy-neural network to approximate nonlinear systems using the proposed genetic algorithm. Each input of the fuzzy-neural network has 5 B-spline membership functions. All of the B-spline membership functions are with order α=2 and each B-spline membership function has 7 control points and 9 knot points. Consider a nonlinear system governed by the difference equation

$$y(k) = 0.3y(k-1) + u(k)(1-0.1y^2(k-1)) + u^3(k)$$

where $u(k) = 0.6\sin(2\pi k/25) + 0.4\sin(2\pi k/10)$. 49 training data pairs are given. It is assumed that the number of the chromosome is 20 and $y(-1)=0$. The initial parameters w^p are random values in the intervals $D_1 = [w_{\min}, w_{\max}] = [-5,5]$, and The initial parameters c_{kj} and t_{qj} of the B-spline membership functions for $u(k)$ or $y(k-1)$ are values according to knot points and control points shown in Fig. 4. Fig. 5 and Fig. 6 show the B-spline membership functions of the fuzzy-neural network for $y(k-1)$ and $u(k)$ after 400 iterations, respectively. The error curve of the proposed method after 400 iterations is shown in Fig. 7. As demonstrated in Figs. 7, the proposed genetic algorithm successfully approximates the nonlinear system.

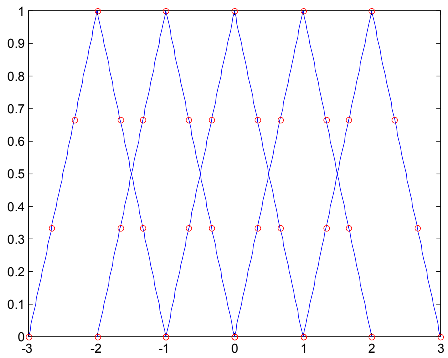

Fig. 4. B-spline membership functions of the fuzzy-neural network for $y(k-1)$ or $u(k)$ before iterations.

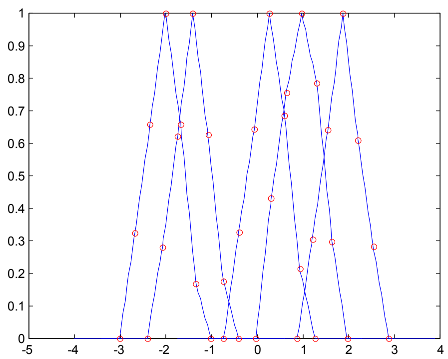

Fig. 5. B-spline membership functions of the fuzzy-neural network for $y(k-1)$ after 400 iterations.

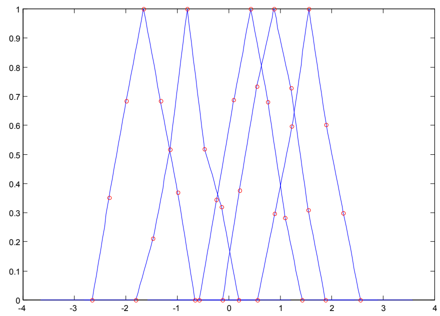

Fig. 6. B-spline membership functions of the fuzzy-neural network for $u(k)$ after 400 iterations.

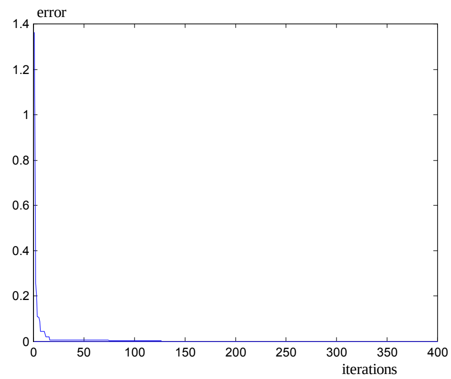

Fig. 7. Error curve of the fuzzy-neural network with respect to iterations.

5 Conclusions

In this paper, since the selection of the weighting factors, the knot positions, and the control points of the B-spline membership fuzzy-neural networks is crucial to obtaining good approximation for complex nonlinear systems, we develop a genetic algorithm with an efficient search strategy to optimize these variables and escape from local minima. For the purpose of illustrating effectiveness of the proposed method, an example with high input dimensions has been simulated.

6 Acknowledgements

This work was supported partially by the National Science Council, Taiwan, under Grant NSC 92-2213-E-146-002.

7 References

[1] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, no.2, pp.359-366,1989.

[2] L.X. Wang and J.M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least squares learning," *IEEE Trans.*

Neural Networks, vol. 3, no. 5, pp.807-814, 1992.

[3] C.H. Wang, W.Y. Wang, T.T. Lee, and P.S. Tseng, "Fuzzy B-spline membership function (BMF) and its applications in fuzzy-neural control," *IEEE Trans. Syst. Man, Cyber*., vol. 25, no. 5, pp.841-851, 1995.

[4] L.X. Wang, *Adaptive fuzzy systems and control: design and stability analysis*, Englewood Cliffs,NJ: Prentice-Hall, 1994.

[5] S. Horikawa, T. Furuhashi, and Y. Uchikawa, "On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm," *IEEE Trans. Neural Networks*, vol. 3, no.5, September 1992.

[6] C. T. Lin and C. S. George Lee, "Neural-network-based fuzzy logic control and decision system," *IEEE Trans. Computer*, vol. 40, no. 12, pp.1320-1336, December 1991.

[7] W.Y. Wang, T.T. Lee, and C.L. Liu, "Function approximation using fuzzy-neural networks with robust learning algorithm," *IEEE Trans. Syst. Man, Cyber. Part B*, vol. 27, no. 4, pp.740-747, 1997.

[8] K.F. C. Yiu, S. Wang, K. L. Teo, and A. C. Tsoi, "Nonlinear System Modeling via Knot-Optimizing B-Spline Networks," *IEEE Trans. NEURAL NETWORKS*, vol. 12, no. 5, pp.1013-1022, 2001.

[9] J. N. Aaral, K. Tumer, and J. Ghosh, "Designing genetic algorithms for the state assignment problem," *IEEE Trans. Syst., Man, Cybern*., vol. 25, pp. 687–694, 1995.

[10] M. Figueiredo, J. Leitão, and A. K. Jain, "Unsupervised contour representation and estimation using B-splines and a minimum description," *IEEE Transactions on Image Processing*, vol. 9, no. 6, pp.1075-1087, 2000.

[11] P. Saint-Marc, H. Rom, and G. Medioni, "B-spline contour representation and symmetry detection," *IEEE Trans. Pattern Anal. Machine In-tell*., vol. 15, pp. 1191–1197, Nov. 1993.

[12] S. Cong and G. Li, "The design of adaptive B-spline fuzzy neural network controller," *Computer Engineering and application*, 35(9), pp.66-68, 1999

[13] Y.G. Leu, T.T. Lee, and W.Y. Wang, " On-line tuning of fuzzy-neural network for adaptive control of nonlinear dynamical systems," *IEEE Trans. Syst. Man, Cyber.* part B: Cybernetics, vol. 27, no. 6, pp.1034-1043, December 1997.

[14] T. Kavli, "ASMOD—An algorithm for adaptive spline modeling of observation data," *Int. J. Contr*., vol. 58, no. 4, pp. 947–967, 1993.

[15] K. Hlavachova and M. Verleysen, "Placing spline knots in neural networks using splines as activation functions," *Neurocomput*., vol. 17, pp.159–166, 1997.

[16] F.-T. Lin *et al*., "Applying the genetic approach to simulated annealing in solving some NP-hard problems," *IEEE Trans. Syst., Man, Cybern*., vol. 23, pp. 1752–1767, Nov./Dec. 1993.

[17] B. Li and W. Jiang, "A novel stochastic optimization algorithm," *IEEE Trans. Syst. Man, Cyber. Part B*, vol. 30, no. 1, pp.193-198, 2000.