International Journal of Control and Automation
Vol. 6, No. 3, June, 2013

Design of Neuro-Fuzzy Networks by Means of Fuzzy Relation Space-Based Rules

Keon-Jun Park and Yong-Kab Kim*

Dept. of Information and Communication Engineering, Wonkwang University,
460, Iksandae-ro, Iksan, Chonbuk, 570-749 South Korea
bird75@wonkwang.ac.kr, ykim@wonkwang.ac.kr

Abstract

This paper introduce neuro-fuzzy networks by means of fuzzy relation space-based Rules for pattern recognizer. The proposed neuro-fuzzy networks are realized with the aid of the grid partition of the fuzzy relation input space. The partitioned spaces express the fuzzy rules of the networks. The consequence part of the rules is represented by polynomial functions whose coefficients are learned by the back-propagation algorithm. To optimize the parameters of the proposed networks, we consider real-coded genetic algorithms. The proposed networks are evaluated with the use of numerical experimentation for pattern recognizer. Finally, this paper shows that the proposed networks have the good result.

Keywords: *Neuro-Fuzzy Networks (NFNs), Grid Partition, Fuzzy Relation Input Space, Genetic Algorithms (GAs), Pattern Recognizer*

1. Introduction

Neuro-fuzzy networks (NFNs) or fuzzy neural networks(FNNs)[1, 2, 3] have emerged as one of the active areas of research in the human-like reasoning method of fuzzy inference systems and connectionist structure of neural networks. These networks are predominantly designed for the integration of these two fields. Typically, NFNs are represented by fuzzy “if–then” rules, while back propagation (BP) is used to optimize the parameters. Using these characteristics, there are still many approaches to apply to FNNs [4, 5, 6].

The generation of the fuzzy rules and the adjustment of their membership functions of the NFNs were conducted by trial and error and/or on the basis of the operator’s experience. The researchers find it difficult to develop adequate fuzzy rules and membership functions to reflect the essence of the data.

In this paper, we introduce the design of neuro-fuzzy networks with multiple outputs by means of fuzzy relation input space to generate the fuzzy rules for pattern recognition. The premise part of the rules of this network is realized with the aid of the grid partition of the input space. The consequence part of the rules is represented by polynomial functions with multiple outputs for pattern recognition. The coefficients of the polynomial functions are learned by the BP algorithm. We also optimize the parameters of the networks using real-coded genetic algorithms (GAs) [7]. The proposed network is evaluated through numerical experimentation for pattern recognition. Finally, this paper shows that the proposed networks have the good result.

* Corresponding author : ykim@wonkwang.ac.kr

2. Design of the fuzzy relation-based neuro-fuzzy networks

2.1. The structure of the fuzzy relation-based NFNs

The structure of the fuzzy relation-based NFNs (FR-based NFNs) emerges at a junction of fuzzy sets by means of the grid partition of fuzzy relation input space in the premise part and neural networks present in the consequence part of the rules. The structure of the FR-based NFNs is composed of six layers. The overall topology of the networks is illustrated in Figure 1.

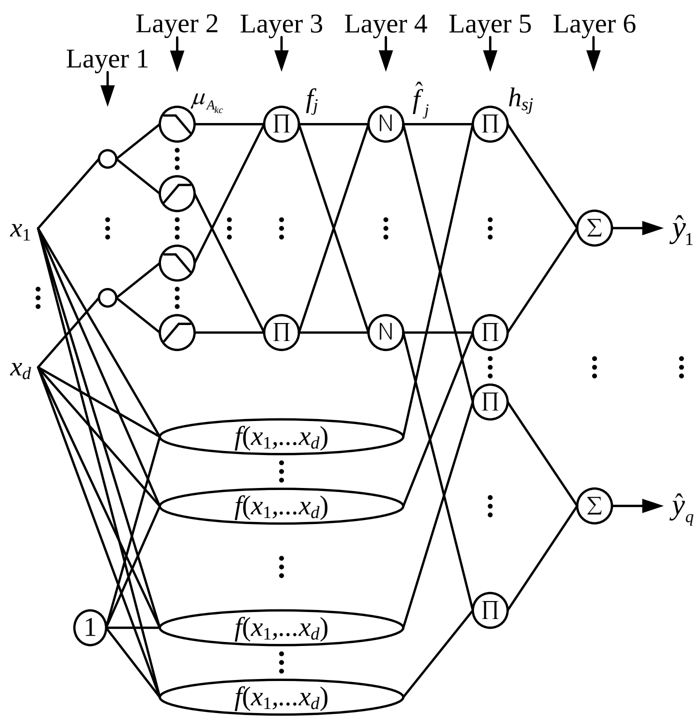


Figure 1. The structure of FR-based NFNs

The proposed FR-based NFNs are implied by the fuzzy grid partition of input spaces. In this sense, each rule can be viewed as a certain rule of the following format.

$$R^j : If \ \ x_1 \ is \ A_{1c} and \cdots and \ x_d \ is \ A_{dc} \ \ Then \ \ y_{sj} = f(x_1, \ldots, x_d) \tag{1}$$

As far as inference schemes are concerned, we distinguish these cases:
Type 1 (Simplified Inference):

$$f = w_{j0}^s \tag{2}$$

Type 2 (Linear Inference):

$$f = w_{j0}^s + \sum_{k=1}^{d} w_{jk}^s x_k \tag{3}$$

Type 3 (Modified Quadratic Inference):

$$f = w_{j0}^{s} + \sum_{k=1}^{d} w_{jk}^{s} x_k + \sum_{k=1}^{d} \sum_{i=k+1}^{d} w_{jz}^{s} x_k x_i \,. \tag{4}$$

To be more specific, R^j is the j-th fuzzy rule, while A_{kc} denotes j-th membership function. w's are consequent parameters of the rule.

The functionality of each layer is described as follows.

[Layer 1] The nodes in this layer transfer the inputs.

[Layer 2] The nodes here are used to calculate the membership degrees for triangular membership functions.

$$\mu_{kc}(x_k) = triangle(x; a, b, c) = \begin{cases} 0, & x \le a \\ \dfrac{x-a}{b-a}, & a \le x < b \\ \dfrac{c-x}{c-b}, & b \le x < c \\ 0, & c \le x \end{cases} \tag{5}$$

[Layer 3] The nodes in this layer compute the firing strength for each rule.

$$f_j = \prod_{k=1}^{d} \mu_{kc}(x_k) \tag{6}$$

[Layer 4] The nodes in this layer normalize the membership degrees

$$\hat{f}_j = \frac{f_j}{\sum_{j=1}^{n} f_j} \tag{7}$$

[Layer 5] The nodes in this layer realize a certain inference process.

$$h_{sj} = \sum_{j=1}^{n} \hat{f}_j y_j \tag{8}$$

[Layer 6] The nodes in this layer compute the outputs.

$$\hat{y}_s = \sum_{j=1}^{n} h_{sj} \tag{9}$$

2.2 The learning algorithm

The parametric learning of the networks is realized by adjusting connections of the neurons and as such it could be realized by running a standard back-propagation (BP) algorithm. The performance index is based on the Euclidean distance. As far as learning is concerned, the connections are adjusted in a standard fashion,

$$w_{j0}^{s}(p+1) = w_{j0}^{s}(p) + \Delta w_{j0}^{s} \tag{10}$$

where this update formula follows the gradient descent method.

Quite commonly to accelerate convergence, a momentum coefficient α is being added to the learning expression. The complete equations are as follows.

$$\Delta w_{j0}^{s} = \eta\left(y_{ps} - \hat{y}_{ps}\right)\hat{f}_j + \alpha\left(w_{j0}^{s}(p) - w_{j0}^{s}(p-1)\right) \tag{11}$$

$$\Delta w_{jk}^{s} = \eta\left(y_{ps} - \hat{y}_{ps}\right)\hat{f}_j x_k + \alpha\left(w_{jk}^{s}(p) - w_{jk}^{s}(p-1)\right) \tag{12}$$

$$\Delta w_{jz}^{s} = \eta\left(y_{ps} - \hat{y}_{ps}\right)\hat{f}_j x_k x_i + \alpha\left(w_{jz}^{s}(p) - w_{jz}^{s}(p-1)\right) \tag{13}$$

3. Genetic optimization

It has been demonstrated that genetic algorithms (GAs) [7] are useful global population-based optimizers. GAs are shown to support robust search in complex search spaces. Given their stochastic character, such methods are less likely to get trapped in local minima (which becomes quite a common problem in case of gradient-descent techniques).

Let us briefly recall the essence of these operators. Reproduction is a process in which the mating pool for the next generation is chosen. Individual strings are copied into the mating pool according to the values of their fitness functions. Crossover usually proceeds in two steps. First, members from the mating pool are mated at random. Secondly, each pair of strings undergoes crossover as follows; a position *l* along the string is selected uniformly at random from the interval [1, *l*-1], where *l* is the length of the string. Swapping all characters between the positions *k* and *l* creates two new strings. Mutation is a random alteration of the value of a string position. In real coding, mutation is defined as an alternation at a random value in special boundary. Usually mutation occurs with a small probability. Those operators, combined with the proper definition of the fitness function, constitute the main body of the genetic optimization.

In order to optimize the parameters of the FR-based NFNs, we determined the apexes of the membership functions of each input variable, the learning rate, and the momentum coefficient as the parameters. Each chromosome is coded using real numbers. This type of coding is helpful from the point of view of the effectiveness of the overall search process. Figure 2 visualizes an arrangement of the content of the chromosome to be used in genetic optimization.

MF apexes of input variable x_1 (3)			• • •	MF apexes of input variable x_d (3)			learning rate (1)	momentum coefficient (1)
Small_1	Middle_1	Big_1		Small_d	Middle_d	Big_d		

Figure 2. Data structure of chromosomes

4. Experimental Studies

In this paper, we use the Iris dataset [8]. The Iris dataset is a collection of 150 Iris flowers of 3 kinds, with four attributes, leaf and petal width and length in cm. Three classes are the setosa, versicolor, and virginica.

For the evaluation of the performance of the network, the random sub-sampling method was applied. The random sub-sampling was performed with 5 data splits of the data set. Each split was randomly selected from the training examples and the test examples with the ratio of 7:3.

The classification ratio (*CR*) is defined as the average of the separate estimates *Ep*.

$$E_p = \frac{No.\,of\ \ classification}{No.\,of\ \ test\ examples} \times 100 \tag{14}$$

$$CR = \frac{1}{K}\sum_{i=1}^{K} E_p \tag{15}$$

Another performance index (*PI*) is based on the Mean Squared Error (MSE).

$$PI = \frac{1}{K}\sum_{i=1}^{K} MSE \tag{16}$$

We experimented with the proposed networks using the parameters outlined in Table 1 with the weight factor $\theta = 0.5$.

Table 1. Initial parameters

	Parameter	Value
GAs	Generation	100
	Population size	50
	Crossover rate	0.65
	Mutation rate	0.1
NFNs	No. of MFs	2, 3
	Learning rate	$0.0 \leq \eta \leq 0.01$
	Moment coefficient	$0.0 \leq \alpha \leq 0.001$

Table 2 summarizes the performance for FR-based NFNs before optimization and Table 3 shows the performance for FR-based NFNs using genetic optimization. From these tables we know that the optimized FR-based NFNs is better than before optimization. From Table 3 we select the network that has three MFs for each input variable and linear inference engine. This network exhibits CR=99.24±0.43, PI=0.017±0.00 for training datasets and CR=99.56±0.99, PI=0.014±0.01 for testing datasets.

Table 4 shows the confusion matrix for the training and testing data set. From Table 4, the results show some misclassification. In the training data, versicolor is classified as virginica with 1.14±1.56 error ratio and virginica is classified as versicolor with 1.14±1.56 error ratio. In the testing data, versicolor is classified as virginica with 1.33±2.98 error ratio.

Table 2. Performance of the FR-based NFNs

No. of MFs	Inference (Type)	CR		PI	
		Training	Testing	Training	Testing
2	1	95.43±1.04	97.78±2.72	0.043±0.00	0.041±0.01
	2	97.52±1.44	96.00±2.90	0.035±0.00	0.036±0.01
	3	96.95±1.24	97.78±0.00	0.034±0.00	0.038±0.01
3	1	97.71±0.85	95.56±3.51	0.023±0.00	0.027±0.01
	2	97.52±0.85	96.00±2.90	0.019±0.00	0.025±0.01
	3	97.33±0.43	96.89±1.99	0.020±0.00	0.026±0.01

Table 3. Performance of the optimized FR-based NFNs

No. of MFs	Inference (Type)	CR		PI	
		Training	Testing	Training	Testing
2	1	98.10±0.95	98.67±1.22	0.031±0.01	0.028±0.01
	2	99.43±0.85	98.67±1.22	0.021±0.00	0.022±0.01
	3	99.43±0.52	98.22±1.86	0.022±0.00	0.021±0.00
3	1	99.43±0.52	97.78±2.22	0.018±0.00	0.025±0.01
	2	99.24±0.43	99.56±0.99	0.017±0.00	0.014±0.01
	3	99.43±0.52	98.67±1.22	0.014±0.00	0.017±0.01

Table 4. Performance evaluation by confusion matrix

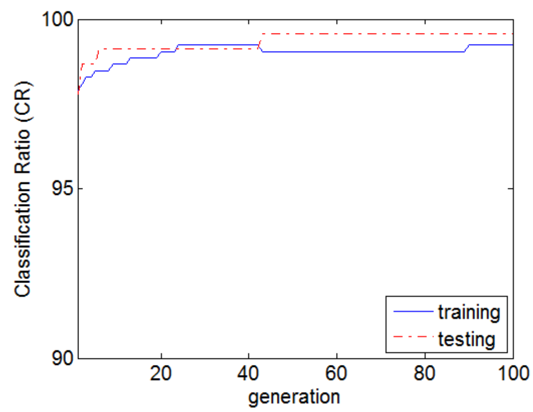
(a) Training data

	Setosa	Versicolor	Virginica
Setosa	100.00±0.0	0.00±0.0	0.00±0.0
Versicolor	0.00±0.0	98.86±1.56	1.14±1.56
Virginica	0.00±0.0	1.14±1.56	98.86±1.56

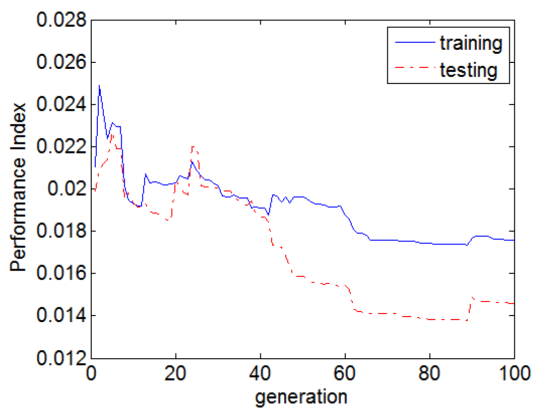
(b) Testing data

	Setosa	Versicolor	Virginica
Setosa	100.00±0.0	0.00±0.0	0.00±0.0
Versicolor	0.00±0.0	98.67±2.98	1.33±2.98
Virginica	0.00±0.0	0.00±0.0	100.00±0.0

Figure 3 presents the optimization procedure for the CR and PI in the selected network obtained in successive generations of the genetic optimization. These figures depict the average values using the random sub-sampling.



(a) CR



(b) PI

Figure 3. Optimization process for the selected network

The performance of the proposed network is compared with the performance of some other models reported in the literature; refer to Table 5. The comparison shows that the proposed network outperforms several previous developed models.

Table 5. Comparison of performance with previous models

Model	Classification Ratio (%)
NEFCLASS [9]	96.0
C4.5 [10]	94.0
FID3.1 [11]	96.0
HNFB [12]	98.67
HNFQ [13]	98.67
HNFB-1 [14]	98.67
Our model	99.56

5. Conclusions

This paper introduced neuro-fuzzy networks by means of fuzzy relation space-based rules for pattern recognition and discussed optimal design using real-coded genetic algorithms. The proposed neuro-fuzzy networks are realized with the aid of the grid partition of the fuzzy relation input space to generate the fuzzy relation space-based rules. From this method, we have designed neuro-fuzzy networks. And genetic algorithms were also used for parametric optimization of the proposed networks. From the result in the previous section, we were able to design good networks and to achieve a balance between the approximation and generalization abilities of the resulting network using CR and PI. The proposed networks may encounter difficulties in case of high-dimension problem and theses dimensionality issues need to be tackled (e.g., through exploiting various partition strategies).

References

[1] T. Yamakawa, "A Neo Fuzzy Neuron and Its Application to System Identification and Prediction of the System Behavior", Proceeding of the 2nd International Conference on Fuzzy logic & Neural Networks, (**1992**) July 17-22; Lizuka, Japan.

[2] J. J. Buckley and Y. Hayashi, "Fuzzy neural networks: A survey", Fuzzy Sets Syst., vol. 66, (**1994**).

[3] J. S. R. Jang, E. Mizutani and C. T. Sun, "Neuro-Fuzzy and Soft Computing", A Computational Approach to Learning and Machine Intelligence, Prentice Hall, NJ, **(1997)**.

[4] M. Beldjehem, "A Unified Granular Fuzzy-Neuro Framework for Predicting and Understanding Software Quality", International Journal of Software Engineering and Its Applications, vol. 4, no. 4, (**2010**).

[5] M. R. Homaeinezhad, E. Tavakkoli1 and A. Ghaffari, "Discrete Wavelet-based Fuzzy Network Architecture for ECG Rhythm-Type Recognition: Feature Extraction and Clustering-Oriented Tuning of Fuzzy Inference System", International Journal of Signal Processing, Image Processing and Pattern Recognition, vol. 4, no. 3, (**2011**).

[6] K. J. Park, D. Y. Lee and J. P. Lee, "Design of FCM-based fuzzy neural networks and its optimization for pattern recognition", Communications in Computer an Information Science, vol. 261 CCIS, (**2011**).

[7] D. E. Golderg, "Genetic Algorithm in search, Optimization & Machine Learning", Addison Wesley, (**1989**).

[8] UCI Machine Learning Repository: Data Sets, http://archive.ics.uci.edu/ml/datasets/Iris.

[9] F. Klawonn, D. Nauck and R. Kruse, "Generating rules from data by fuzzy and neuro-fuzzy methods", in Proc. Fuzzy-Neuro-Systeme, (**1995**).

[10] J. R. Quinlan, "Improved use of continuous attributes in C4.5", J. Artif. Intell. Res., vol. 4, (**1996**).

[11] C. Z. Janikow and M. Faifer, "Fuzzy partitioning with FID3.1", in Proc. IEEE 18th Int. Conf. North American Fuzzy Information Processing Soc., (**1999**).

[12] F. J. de Souza, M. M. B. R. Vellasco and M. A. C. Pacheco, "Load forecasting with the hierarchical neuro-fuzzy binary space partitioning model", Int. J. Comput. Syst. Signals, vol. 3, no. 2, (**2002**).

[13] F. J. de Souza, M. M. B. R. Vellasco and M. A. C. Pacheco, "Hierarchical neuro-fuzzy quadtree models", Fuzzy Sets Syst., vol. 130, no. 2, (**2002**).

[14] L. B. Gonçalves, M. M. B. R. Vellasco, M. A. C. Pacheco and F. J. de Souza, "Inverted hierarchical neuro-fuzzy BSP system: A novel neuro-fuzzy model for pattern classification and rule extraction in databases", IEEE Transactions on Systems, Man & Cybernetics. Part C, vol. 36, no. 2, (**2005**).

Authors

Keon-Jun Park

He received the B.S degree in Electrical & Electronics Engineering from the Wonkwang University, Iksan, South Korea, in 2003, and the M.S degree in Control and Instrumentation Engineering in 2005. He received the Ph. D degree in Electrical Engineering from University of Suwon, Suwon, South Korea, in 2010. He is currently a research professor with Information & Communication Engineering, Wonkwang University, Korea. His research interests include computational intelligence, intelligence modeling and control, and pattern recognition.

Yong-Kab Kim

He received the B.S degree in Electronics Engineering from Ajou University and the M.S.E degree in Electrical & Computer Engineering from University of Alabama in Huntsville. He received his Ph.D degree in Electrical & Computer Engineering in North Carolina State University. He is currently in Professor at School of Information and Communication Engineering, Wonkwang University, Korea. His research interests are Remote Sensing for Visible Communication, Optical Fiber Sensing, Intelligent Control, Advanced Computational Intelligence.