2005년도 대한전기학회 하계학술대회 논문집 2005. 7. 18 - 20

Evolutionary Optimized Fuzzy Set-based Polynomial Neural Networks Based on Classified Information Granules

Sung-Kwun Oh[1], Seok-Beom Roh[2], and Tae-Chon Ahn[2]

[1]Department of Electrical Engineering, The University of Suwon, South Korea

[2]Department of Electrical Electronic and Information Engineering, Wonkwang University

Abstract - In this paper, we introduce a new structure of fuzzy-neural networks Fuzzy Set-based Polynomial Neural Networks (FSPNN). The two underlying design mechanisms of such networks involve genetic optimization and information granulation. The resulting constructs are Fuzzy Polynomial Neural Networks (FPNN) with fuzzy set-based polynomial neurons (FSPNs) regarded as their generic processing elements. First, we introduce a comprehensive design methodology (viz. a genetic optimization using Genetic Algorithms) to determine the optimal structure of the FSPNNs. This methodology hinges on the extended Group Method of Data Handling (GMDH) and fuzzy set-based rules. It concerns FSPNN-related parameters such as the number of input variables, the order of the polynomial, the number of membership functions, and a collection of a specific subset of input variables realized through the mechanism of genetic optimization. Second, the fuzzy rules used in the networks exploit the notion of information granules defined over systems variables and formed through the process of information granulation. This granulation is realized with the aid of the hard C- Means clustering (HCM). The performance of the network is quantified through experimentation in which we use a number of modeling benchmarks already experimented with in the realm of fuzzy or neurofuzzy modeling.

1. Introduction

A lot of researchers on system modeling have been interested in the multitude of challenging and conflicting objectives such as compactness, approximation ability, generalization capability and so on which they wish to satisfy. As one of the representative advanced design approaches to build models with substantial approximation capabilities comes a family of self-organizing networks with fuzzy polynomial neuron (FPN)[6]. The design procedure of the FPNNs exhibits some tendency to produce overly complex networks as well as comes with a repetitive computation load caused by the trial and error method being a part of the development process. In this paper, in considering the above problems coming with the conventional FPNN [6], we introduce a new structure of fuzzy rules as well as a new genetic design approach. The new structure of fuzzy rules based on the fuzzy set-based approach changes the viewpoint of input space division. In other hand, from a point of view of a new understanding of fuzzy rules, information granules seem to melt into the fuzzy rules respectively. The determination of the optimal values of the parameters available within an individual FSPN leads to a structurally and parametrically optimized network through the genetic approach.

2. The architecture of FSPNN

The FSPN encapsulates a family of nonlinear "if-then" rules. When put together, FSPNs results in a self-organizing Fuzzy Set-based Polynomial Neural Networks (FSPNN). As visualizedin Fig. 1, the FSPN consists of two basic functional modules. The first one, labeled by F, is a collection of fuzzy sets (here denoted by $\{A_k\}$ and $\{B_k\}$) that form an interface between the input numeric variables and the processing part realized by the neuron. The second module (denoted here by **P**) refers to the function based nonlinear (polynomial) processing that involves some input variables This nonlinear processing involves some input variables (x_i and x_j), which are capable of being the input variables (Here, x_p and x_q), or entire system input variables. Each rule reads in the form

$$\begin{aligned} &if\ x_p\ is\ \mathbf{A}_k\ then\ z\ is\ P_{pk}(x_i, x_j, \mathbf{a}_{pk}) \\ &if\ x_q\ is\ \mathbf{A}_k\ then\ z\ is\ P_{qk}(x_i, x_j, \mathbf{a}_{qk}) \end{aligned} \tag{1}$$

where $\mathbf{a}_{qk}$ is a vector of the parameters of the conclusion part of the rule while $P(x_i, x_j, \mathbf{a})$denoted the regression polynomial forming the consequence part of the fuzzy rule. The activation levels of the rules contribute to the output of the FSPN being computed as a weighted average of the individual condition parts (functional transformations) P_K (note that the index of the rule, namely "K" is a shorthand notation for the two indexes of fuzzy sets used in the rule (1), that is $K = (l, k)$).

$$\begin{aligned} z &= \sum_{l=1}^{m}\sum_{k=1}^{l}\mu_{(l,k)}P_{(l,k)}(x_i, x_j, \mathbf{a}_{(i,k)}) / \sum_{k=1}^{l}\mu_{(l,k)} \\ &= \sum_{l=1}^{m}\sum_{k=1}^{l}\tilde{\mu}_{(l,k)}P_{(l,k)}(x_i, x_j, \mathbf{a}_{(i,k)}) \end{aligned} \tag{2}$$

When developing an FSPN, we use genetic algorithms to produce the optimized network. This is realizedby selecting such parameters as the number of input variables, the order of polynomial, and choosing a specific subset of input variables. Based on the genetically optimized number of the nodes (input variables) and the polynomial order, refer to Table 1, we construct the optimized self-organizing network architectures of the FSPNNs.

Table 1. Different forms of the regression polynomials forming the consequence part of the fuzzy rules.

Order of the polynomial \ No. of inputs	1	2	3
0 (Type 1)	Constant	Constant	Constant
1 (Type 2)	linear	Bilinear	Trilinear
2 (Type 3)	Quadratic	Biquadratic-1	Triquadratic-1
2 (Type 4)		Biquadratic-2	Triquadratic-2

1: Basic type, 2: Modified type

3. Information Granulation through Hard C-Means clustering algorithm

Information granules are defined informally as linked collections of objects (data points, in particular) drawn together by the criteria of indistinguishability, similarity or functionality. Granulation of information is a procedure to extract meaningful concepts from numeric data and an inherent activity of human being carried out with intend of better understanding of the problem. We granulate information into some classes with the aid of Hard C-means clustering algorithm, which deals with the conventional crisp sets.

We assume that given a set of data $X=\{x_1,x_2,\cdot,x_n\}$ r elated to a certain application, there are some clusters revealed by the HCM. Each cluster is represented by its center and all elements, which belong to it. In ord er to construct fuzzy sets on the basis of such cluster s, we follow a construct shown in Fig. 1. Here the ce nter point stand for the apex of the membership funct ion of the fuzzy set.

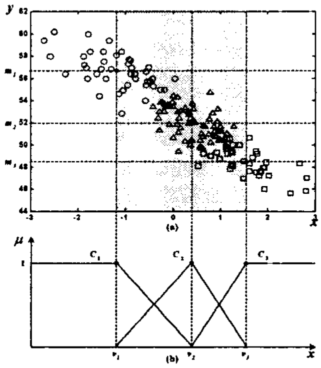

Fig. 1 Forming membership functions with the use of information granules

Where, C is the cluster, v means the center point of input variable x and m denotes the center point of output variable y. In Fig. 2, denotes the element belonging to 1st cluster, is the element allocated to the 2nd cluster, and to 3rd cluster. Each membership function in the premise part of the rule is assigned to be complementary with neighboring ones in the form being shown in the above figure. Let us consider building the consequent part of fuzzy rule. We can think of each cluster as a sub-model composing the overall system. The fuzzy rules of Information Granulation-based FSPN are as followings.

if x_p is A^*_k then $z-m_{pk} = P_{pk}((x_i-v^i_{pk}),(x_j- v^j_{pk}),a_{pk})$

if x_q is B^*_k then $z-m_{qk} = P_{qk}((x_i-v^i_{qk}),(x_j- v^j_{qk}),a_{qk})$ (3)

Where, A^*_k and B^*_k mean the fuzzy set, the apex of which is defined as the center point of information granule (cluster) and m_{pk} is the center point related to the output variable on cluster_{pk}, v^i_{pk} is the center point related to the i-th input variable on cluster_{pk} and a_{qk} is a vector of the parameters of the conclusion part of the rule while $P((x_i-v_i),(x_j- v_j),a)$ denoted the regression polynomial forming the consequence part of the fuzzy rule which uses several types of high-order polynomials (linear, quadratic, and modified quadratic) besides the constant function forming the simplest version of the consequence; refer to Table 1. If we are given m inputs and one output system and the consequent part of fuzzy rules is linear, the overall procedure of modification of the generic fuzzy rules is as followings.

The given inputs are $\mathbf{X}=[\mathbf{x}_1\ \mathbf{x}_2 \cdots \mathbf{x}_m]$ related to a certain application, where $\mathbf{x}_k=[x_{k1} \cdots x_{kn}]^T$, n is the number of data and m is the number of variables and the output is $\mathbf{Y}=[y_1\ y_2 \cdots y_n]^T$.

Step 1) build the universe set

Step 2) build m reference data pairs composed of $[\mathbf{x}_1;\mathrm{Y}]$, $[\mathbf{x}_2;\mathrm{Y}]$, and $[\mathbf{x}_m;\mathrm{Y}]$.

Step 3) classify the universe set U into l clusters by using HCM according to the reference data pair $[\mathbf{x}_i;\mathrm{Y}]$. Where c_{ij} means the j-th cluster (subset) according to the reference data pair $[\mathbf{x}_i;\mathrm{Y}]$.

Step 4) construct the premise part of the fuzzy rules related to the i-th input variable (x_i) using the directly obtained center points from HCM.

Step 5) construct the consequent part of the fuzzy rules related to the i-th input variable (x_i). On this step, we need the center points related to all input variables. We should obtain the other center points through the indirect method as followings.

Sub-step1) make a matrix as equation (4) according to the clustered subsets

$$A^i_j=\begin{vmatrix} x_{21} & x_{22} & \cdots & x_{2m} & y_2 \\ x_{51} & x_{52} & \cdots & x_{5m} & y_5 \\ \vdots & \vdots & & \vdots & \vdots \\ x_{k1} & x_{k2} & \cdots & x_{km} & y_k \\ \vdots & \vdots & & \vdots & \vdots \end{vmatrix} \tag{4}$$

Where, $\{x_{k1}, x_{k2}, , x_{km}, y_k\} \in c_{ij}$ and A_{ij} means the membership matrix of j-th subset related to the i-th input variable.

Sub-step2) take an arithmetic mean of each column on A_{ij}. The mean of each column is the additional center point of subset cij. The arithmetic means of column is equation (5)

$$centerpoints=[\ v^1_{ij}\ v^2_{ij} \cdots v^m_{ij}\ m_{ij}] \tag{5}$$

4. Genetic Optimization of FPNN

GAs are aimed at the global exploration of a solution space. The main features of genetic algorithms concern individuals viewed as strings, population-based optimization and stochastic search mechanism (selection and crossover). GAs use serial method of binary type, roulette-wheel as the selection operator, one-point crossover, and an invert operation in the mutation operator [2]. In this study, for the optimization of the FPNN model, GA uses the serial method of binary type, roulette-wheel used in the selection process, one-point crossover in the crossover operation, and a binary inversion operation in the mutation operator. To retain the best individual and carry it over to the next generation, we use elitist strategy. The overall genetically-driven structural optimization process of FPNN is shown in Fig. 2.

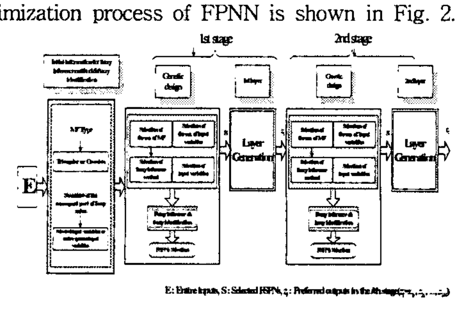

Fig. 2. Overall genetically-driven structural optimization process of FPNN

The framework of the design procedure of the genetically optimized FSPNN comprises the following steps

[Step 1] Determine systems input variables

[Step 2] Form training and testing data

[Step 3] specify initial design parameters

[Step 4] Decide FSPN structure using genetic design

[Step 5] Carry out fuzzy-set based fuzzy inference a nd coefficient parameters estimation for fuzzy identific ation in the selected node (FSPN)

[Step 7] Check the termination criterion

[Step 8] Determine new input variables for the next l ayer

5. Experimental studies

We consider a nonlinear static system with two inputs, x_1, x_2 and a single output that assumes the following form

$$y = (1 + x_1^{-2} + x_2^{-1.5})^2, \quad 1 \le x_1, x_2 \le 5 \qquad (6)$$

This nonlinear static equation is widely used to evaluate modeling performance of the fuzzy modeling. This system represents the nonlinear characteristic as shown in Fig. 3. Using (6), 50 input-output data are generated: the inputs are generated randomly and the corresponding output is then computed through the above relationship. We consider the MSE to serve as a performance index

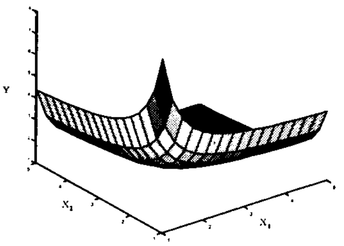

Fig. 3. Input-output relation of the nonlinear static system used in the experiment

Fig. 4 summarizes the detailed results related to IG_gFSPNN with Type T* at all layers. As shown in Fig. 19, the performance in the 2nd layer of the genetically optimized network gets radically much better in both cases (triangular and Gaussian MFs).

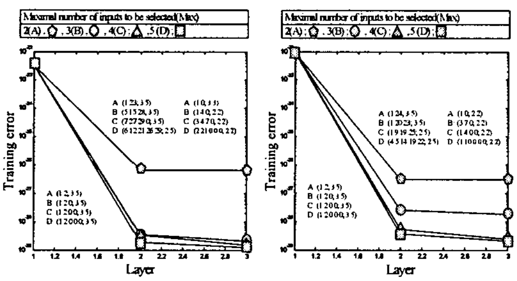

Fig. 4. Performance index of the network shown with respect to the increase of maximal number of inputs t o be selected (in case of using Type T*)

Table 2 covers a comparative analysis including several previous fuzzy and neuro-fuzzy models.

Table 2. Comparative analysis of the performance of the network; considered are models reported in the literature

Model				Performance Index
				PI
Sugeno and Yasukawa's model				0.0790
Gomez-Skarmeta et al.'s model				0.0700
Kim et al.'s model				0.0089
Proposed IG-gFSPNN	Type T	Triangular	2nd layer(Max=4)	1.47e-29
		Gaussian	2nd layer(Max=4)	1.07e-29

6. Conclusion

In this study, we have developed the new structure and formed the semantics of fuzzy rules driven to information granulation along with its architectural considerations. The entire system is divided into some sub-systems that are classified according to the data characteristics named information granules. Each information granule is a sound representative of the related subsystems. A new fuzzy rule with information granule describes a sub-system as a stand-alone system. A fuzzy system with some new fuzzy rules depicts the whole system as a combination of some stand-alone components. The comprehensive experimental studies involving well-known data sets quantify a superb performance of the network in comparison to the existing fuzzy and neuro-fuzzy models.

감사의 글

본 연구는 산업자원부의 지원에 의하여 기초전력연구원 (R-2004-B-133-01) 주관으로 수행된 과제임.

[참 고 문 헌]

[1]S.-K. Oh and W. Pedrycz, Self-organizing Polynomial N eural Networks Based on PNs or FPNs : Analysis and D esign , *Fuzzy Sets and Systems*, 2003(in press).

[2]Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs* , Springer-Verlag, Berlin Heidelberg, 1996.

[3]D. Jong, K. A., *Are Genetic Algorithms Function Optim izers?* , Parallel Problem Solving from Nature 2, Manner, R. and Manderick, B. eds., North-Holland, Amsterdam.

[4]S.-K. Oh and W. Pedrycz, Fuzzy Polynomial Neuron-Ba sed Self-Organizing Neural Networks , *Int. J. of General Systems*, Vol. 32, No. 3, pp. 237-250, May, 2003.

[5]L. X. Wang, J. M. Mendel, Generating fuzzy rules from n umerical data with applications, *IEEE Trans. Systems, Ma n, Cybern.*, Vol. 22, No. 6, pp. 1414-1427, 1992.

[6]J. S. R. Jang, ANFIS: Adaptive-Network-Based Fuzzy I nference System , *IEEE Trans. System, Man, and Cyber n.*, Vol. 23, No. 3, pp. 665-685, 1993.

[7]L. P. Maguire, B. Roche, T. M. McGinnity, L. J. McDaid, Predicting a chaotic time series using a fuzzy neural net work , *Information Sciences*, Vol. 112, pp. 125-136, 1998.

[8]S.-K. Oh, W. Pedrycz and T.-C. Ahn, Self-organizing n eural networks with fuzzy polynomial neurons , *Applied Soft Computing*, Vol. 2, Issue 1F, pp. 1-10, Aug. 2002.

[9]L. A. Zadeh, Toward a theory of fuzzy information gran ulation and its centrality in human reasoning and fuzzy lo gic , *Fuzzy Sets Syst.* vol. 90, pp. 111-117, 1997.