# Design of hetero-hybridized feed-forward neural networks with information granules using evolutionary algorithm

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

[1]Department of Electrical Electronic and Information Engineering, Wonkwang University, 344-2, Shinyong-Dong, Iksan, Chon-Buk, 570-749, South Korea

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

ohsk@suwon.ac.kr

## Abstract

We introduce a new architecture of hetero-hybridized feed-forward neural networks composed of fuzzy set-based polynomial neural networks (FSPNN) and polynomial neural networks (PNN) that are based on a genetically optimized multi-layer perceptron and develop their comprehensive design methodology involving mechanisms of genetic optimization and Information Granulation. The construction of Information Granulation based HFSPNN (IG-HFSPNN) exploits fundamental technologies of Computational Intelligence(CI), namely fuzzy sets, neural networks, and genetic algorithms(GAs) and Information Granulation. The architecture of the resulting genetically optimized Information Granulation based HFSPNN (namely IG-gHFSPNN) results from a synergistic usage of the hybrid system generated by combining new fuzzy set based polynomial neurons (FPNs)-based Fuzzy Neural Networks(FNN) with polynomial neurons (PNs)-based Polynomial Neural Networks(PNN). The design of the conventional genetically optimized HFPNN exploits the extended Group Method of Data Handling(GMDH) with some essential parameters of the network being tuned by using Genetic Algorithms throughout the overall development process. However, the new proposed IG-HFSPNN adopts a new method called as Information Granulation to deal with Information Granules which are included in the real system, and a new type of fuzzy polynomial neuron called as fuzzy set based polynomial neuron. The performance of the IG-gHFPNN is quantified through experimentation.

## 1. Introduction

Recently, a great deal of attention has been directed towards advanced technologies of Computational Intelligence (CI) and their usage towards system modeling. As one of the representative and advanced design approaches comes a family of multi-layer self-organizing neural networks [5,6] such as hybrid fuzzy polynomial neural networks (HFPNN) as well as polynomial neural network (PNN) and fuzzy polynomial neural networks (FPNN) treated as a new category of neuro-fuzzy networks. In this study, we introduce a new genetic design approach. Bearing this new design in mind, we will be referring to these networks as genetically optimized HFPNN ("gHFPNN" for brief). The determination of the optimal values of the parameters available within an individual PN and FPN (viz. the number of input variables, the order of the polynomial, and a collection of preferred nodes) leads to a structurally and parametrically optimized network. As a result, this network becomes more flexible as well as it starts exhibitsing simpler topology in comparison to those conventional HFPNN, FPNN, and PNN as being discussed in the previous research. Moreover the hybrid architecture is much more flexibly designed for a compromise between approximation and generalization, as well as a tradeoff between accuracy and complexity of the overall network. To evaluate the performance of the proposed model, we discuss an experimental study exploiting well-known data being already used in the realm of fuzzy or neurofuzzy modeling [5-8].

## 2. The architecture of HFSPNN

Proceeding with the overall HFPNN architecture, essential design decisions have to be made with regard to the number of input variables, the order of the polynomial, and a collection of the specific subset of input variables. We distinguish between two kinds of layers of the HFPNN architecture (PN-based layer and FPN-based layer of the HFPNN).

### 2.1 fuzzy polynomial neuron (FPN) based layer of HFPNN

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). 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.

$$\begin{aligned} &\text{if } x_p \text{ is } A_k \text{ then } z \text{ is } P_{pk}(x_i, x_j, \boldsymbol{a}_{pk}) \\ &\text{if } x_q \text{ is } B_k \text{ then } z \text{ is } P_{qk}(x_i, x_j, \boldsymbol{a}_{qk}) \end{aligned} \quad (1)$$

where $\boldsymbol{a}_{qk}$ is a vector of the parameters of the conclusion part of the rule while $P(x_i, x_j, \boldsymbol{a})$ denoted the regression polynomial forming the consequence part of the fuzzy rule which uses several type of the high order polynomials besides the constant function forming the simplest version of the consequence.

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) $P_{(l,k)}$.

$$z = \sum_{l=1}^{total\ input}\left(\sum_{k=1}^{total\ rule_l} \mu_{(l,k)} P_{(l,k)}(x_i, x_j, \boldsymbol{a}_{(l,k)}) \Big/ \sum_{k=1}^{total\ rule_l} \mu_{(l,k)}\right) = \sum_{l=1}^{total\ input}\left(\sum_{k=1}^{total\ rule_l} \overline{\mu}_{(l,k)} P_{(l,k)}(x_i, x_j, \boldsymbol{a}_{(l,k)})\right)$$

(2)

### 2.2 The architecture of polynomial neurons (PN) based layer of gHFSPNN

As underlined, the PNN algorithm in the PN based layer of gHFSPNN is based on the GMDH method and utilizes a class of polynomials such as linear, quadratic, modified quadratic, etc. todescribe basic processing realized there. The estimated output $\hat{y}$ reads as

$$\hat{y} = c_0 + \sum_{i=1}^{N} c_i x_i + \sum_{i=1}^{N}\sum_{j=1}^{N} c_{ij} x_i x_j + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} c_{ijk} x_i x_j x_k$$

(3)

## 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={x1,x2,□,xn} related 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 order to construct fuzzy sets on the basis of such clusters, we follow a construct shown in Fig. 1. Here the center point stand for the apex of the membership function of the fuzzy set.

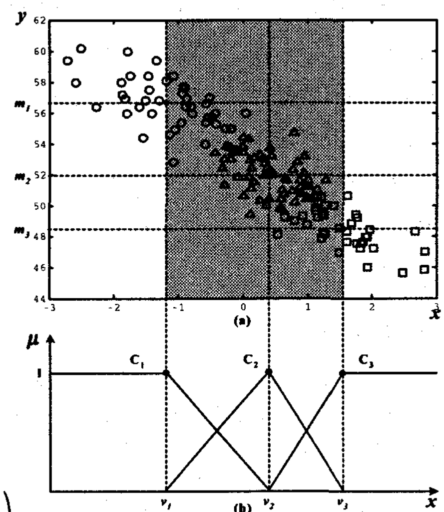



Fig. 1 Forming membership functions with the use of information granules

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.

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

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

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 $\text{cluster}_{pk}$, $v^i_{pk}$ is the center point related to the *i-th* input variable on $\text{cluster}_{pk}$ 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

Universe set $U=\{\{x_{11}, x_{12}, \cdots, x_{1m}, y_1\}, \{x_{21}, x_{22}, \cdots, x_{2m}, y_2\}, , \{x_{n1}, x_{n2}, \cdots, x_{nm}, y_n\}\}$

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

**Step 3)** classify the universe set U into *l* clusters such as $c_{i1}$, $c_{i2}$, , $c_{il}$ (subsets) by using HCM according to the reference data pair $[\mathbf{x}_i;Y]$. Where $c_{ij}$ means the *j-th* cluster (subset) according to the reference data pair $[\mathbf{x}_i;Y]$.

**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

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

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)

$$center\ points = [\ v^1_{ij} v^2_{ij} \cdots v^m_{ij}\ ] \quad (5)$$

## 4. The algorithm and design procedure of genetically optimized HFPNN (gHFPNN)

The framework of the design procedure of the HFSPNN based on genetically optimized multi-layer perceptron architecture comprises the following steps.

**[Step 1]** *Determine system's input variables.*

**[Step 2]** *Form a training and testing data.*

**[Step 3]** *Decide initial information for constructing the gHFSPNN structure.*

**[Step 4]** *Decide a structure of the PN and FSPN based layer of gHFSPNN using genetic design.*

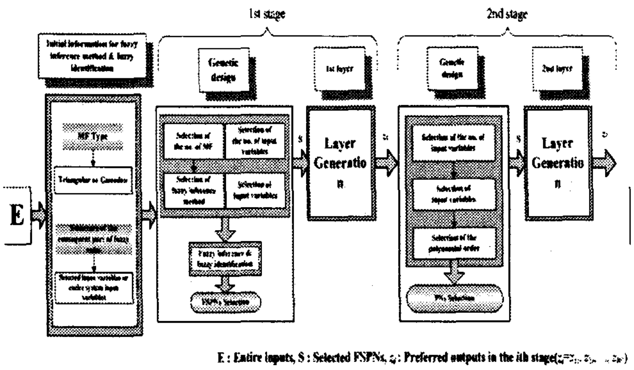



Fig 2. Overall scheme of the genetically-driven structural optimization process of gHFSPNN

**[Step 5]** *Estimate the coefficient parameters of the polynomial in the selected node (PN or FSPN).*

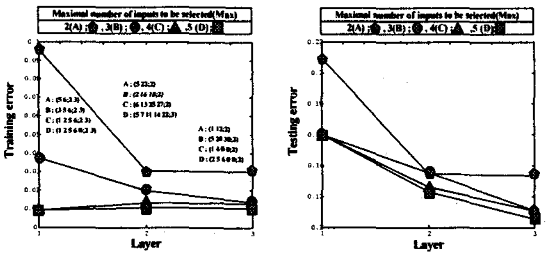
**[Step 6]** *Select nodes (PNs or FSPNs) with the bestpredictive capability and construct their corresponding layer.*

**[Step 7]** *Check the termination criterion.*

**[Step 8]** *Determine new input variables for the next layer.*

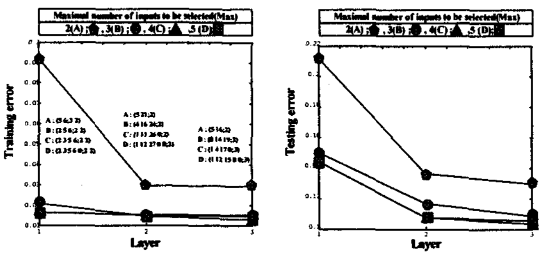
## 5. Simulation studies

We illustrate the performance of the network and elaborate on its development by experimenting with data coming from the gas furnace process. The time series data (296 input-output pairs) resulting from the gas furnace process has been intensively studied in the previous literature [12, 13, 17-34]. The delayed terms of methane gas flow rate, $u(t)$ and carbon dioxide density, $y(t)$ are used as system input variables such as $u(t-3)$, $u(t-2)$, $u(t-1)$, $y(t-3)$, $y(t-2)$, and $y(t-1)$. The output variable is $y(t)$. The first part of the dataset (consisting of 148 pairs) was used for training. The remaining part of the series serves as a testing set. Fig. 3 depicts the performance index of each layer of IG-gHFSPNN with Type T according to the increase of maximal number of inputs to be selected.




(a-1) Training error (a-2) Testing error

(a) Triangular MF




(b-1) Training error (b-2) Testing error

(b) Gaussian-like MF

Fig. 3. Performance index of IG_gFSPNN (with Type T) with respect to the increase of number of layers

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

| Model | | | | Performance Index | | |
|---|---|---|---|---|---|---|
| | | | | PI | $PI_s$ | $EPI_s$ |
| sugeno and yasukawa'smodel | | | | 0.190 | | |
| Oh and Pedrycz's model | | | | 0.123 | 0.020 | 0.271 |
| Kim et al.' model | | | | | 0.034 | 0.244 |
| Proposed IG-gFSPNN | TypeII I (I=6) | Triangular | 3rd layer(Max=3) | | 0.008 | 0.110 |
| | | Gaussian | 3rd layer(Max=3) | | 0.008 | 0.099 |

### 6. Conclusion

In this study, the GA-based design procedure of Information Granulation Hybrid Fuzzy Set based Polynomial Neural Networks (IG-gHFSPNN) along with its architectural considerations has been investigated. Through the consecutive generation of a layer through a growth process (iteration) of the IG-gHFSPNN, the depth (layer size) and width (node size of each layer) of the network could be flexibly selected based on a diversity of local characteristics of these preferred FSPNs and PNs (such as the number of input variables, the order of the consequent polynomial of rules/the polynomial order, a collection of specific subset of input variables, and the number of membership functions) available within HFSPNN. The design methodology comes as a hybrid structural optimization (based on GMDH method and genetic optimization) and parametric learning being viewed as two fundamental phases of the design process. The comprehensive experimental study involving well-known datasets quantify a superb performance of the network in comparison to the existing fuzzy and neuro-fuzzy models. Most importantly, through the proposed framework of genetic optimization we canefficiently search for the optimal network architecture (structurally and parametrically optimized network) and this becomes crucial in improving the performance of the resulting model.

This work has been supported by KESRI(R-2004-B-133-01), which is funded by MOCIE(Ministry of commerce, Industry and Energy).

## 7. Reference

[1] V. Cherkassky, D. Gehring, and F. Mulier, Comparison of adaptive methods for function estimation from samples , *IEEE Trans. Neural Networks*, Vol. 7, pp. 969 984, July 1996.Electrical Engineering, Vol. 29, Issue 6, pp. 703-725, 2003.

[2] J. A. Dickerson and B. Kosko, Fuzzy function approximation with ellipsoidal rules , *IEEE Trans. Syst., Man, Cybernetics*, Part B, Vol. 26, pp.542 560, Aug. 1996.

[3] V. Sommer, P. Tobias, D. Kohl, H. Sundgren, and L. Lundstrom, Neural networks and abductive networks for chemical sensor signals: A case comparison , *Sensors and Actuators*, B. 28, pp. 217-222, 1995.

[4] S. Kleinsteuber and N. Sepehri, A polynomial network modeling approach to a class of large-scale hydraulic systems , *Computers Elect. Eng.*, 22, pp. 151-168, 1996.

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

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

[7] M. Sugeno and T. Yasukawa, A Fuzzy-Logic-Based Approach to Qualitative Modeling , *IEEE Trans. Fuzzy Systems*, Vol. 1, No. 1, pp. 7-31, 1993.

[8] E.-T. Kim, et al, A new approach to fuzzy modeling , *IEEE Trans. Fuzzy Systems*, Vol. 5, No. 3, pp. 328-337, 1997.