*636 Journal of Electrical Engineering & Technology Vol. 7, No. 4, pp. 636~645, 2012*
http://dx.doi.org/10.5370/JEET.2012.7.4.636

# Multiobjective Space Search Optimization and Information Granulation in the Design of Fuzzy Radial Basis Function Neural Networks

**Wei Huang*, Sung-Kwun Oh† and Honghao Zhang***

**Abstract** – This study introduces an information granular-based fuzzy radial basis function neural networks (FRBFNN) based on multiobjective optimization and weighted least square (WLS). An improved multiobjective space search algorithm (IMSSA) is proposed to optimize the FRBFNN. In the design of FRBFNN, the premise part of the rules is constructed with the aid of Fuzzy C-Means (FCM) clustering while the consequent part of the fuzzy rules is developed by using four types of polynomials, namely constant, linear, quadratic, and modified quadratic. Information granulation realized with C-Means clustering helps determine the initial values of the apex parameters of the membership function of the fuzzy neural network. To enhance the flexibility of neural network, we use the WLS learning to estimate the coefficients of the polynomials. In comparison with ordinary least square commonly used in the design of fuzzy radial basis function neural networks, WLS could come with a different type of the local model in each rule when dealing with the FRBFNN. Since the performance of the FRBFNN model is directly affected by some parameters such as e.g., the fuzzification coefficient used in the FCM, the number of rules and the orders of the polynomials present in the consequent parts of the rules, we carry out both structural as well as parametric optimization of the network. The proposed IMSSA that aims at the simultaneous minimization of complexity and the maximization of accuracy is exploited here to optimize the parameters of the model. Experimental results illustrate that the proposed neural network leads to better performance in comparison with some existing neurofuzzy models encountered in the literature.

**Keywords**: Fuzzy Radial Basis Function Neural Networks (FRBFNN), Improved Multiobjective Space Search Algorithm (IMSSA), Information Granulation (IG), Weighted Least Squares (WLS)

## 1. Introduction

Fuzzy Radial Basis Function Neural Networks (FRBFNNs) has been utilized in numerous fields for engineering, medical engineering, and social science [1, 2]. They are designed by integrating the principles of Radial Basis Function Neural Networks (RBFNNs) and invoking the mechanisms of information granulation provided by the Fuzzy C-Means (FCM). There are two main and conflicting objectives guiding the design of FRBFNNs. One is the accuracy of the network (to be maximized) and another one is its complexity (to be minimized). The objective of any effective learning method is to develop accurate, simple, and interpretable FRBFNNs. In the 1990s, the emphasis of neurofuzzy modeling was clearly positioned on accuracy maximization. As powerful optimization tools in many science fields [3, 4], various evolutionary algorithms have been proposed to improve the accuracy of models including such as Genetic Algorithms (GAs), Particle Swarm Optimization (PSO), and Genetic Programming (GP) [5-7]. Those methods usually help improve the accuracy, while the complexity of the model increases as a result of the accuracy maximization. Some researchers attempted to simultaneously optimize the accuracy and the complexity of the fuzzy models [8, 9]. Evidently, in such circumstances one has to become aware of the accuracy-complexity tradeoff. Recently, the accuracy maximization and complexity minimization have been often cast in the setting of multi-objective optimization. A number of evolutionary algorithms (EAs) have been developed to solve multi-objective optimization problems such as NSGA-II [10], IC-MOP [11], and so on [12, 13]. These EAs are population-based algorithms, which allow exploring simultaneously different segments of the Pareto front. As a result, multiobjective optimization techniques have been applied to the design of fuzzy models striving for their high accuracy and significant interpretability [14-18]. In our previous study [19], we introduce a FRBFNN based on a multi-objective space search algorithm. This work provides some enhancements of the neural network. Noticeably, some constrains of this neural network are as follows: (1) there is only a single type of all polynomials of

† Corresponding Author: Dep. of Electrical Engineering, The University of Suwon, Korea. (ohsk@suwon.ac.kr)
* School of Computer and Communication Engineering, Tianjin University of Technology, China. (huangwabc@163.com)
Received: June 8, 2011; Accepted: February 17, 2012

the consequence part of fuzzy rules, so the flexibility and predictive ability are limited; and (2) the C-means method is exploited to construct fuzzy rules, yet the information of the dataset is not adequately abstracted for the design of fuzzy rules.

In this paper, we introduce a fuzzy radial basis function neural networks based on information granulation and multi-objective optimization. While this is an extended work of [19], here we use a modified FRBFNN and propose an improved multi-objective space search algorithm (IMSSA) to optimize this FRBFNN. The resulting FRBFNN addresses the two limitations mentioned above. On the one hand, information granulation and resulting information granules themselves become an important design aspect of fuzzy rules. With the use of information granulation, the information of the dataset is efficiently used for the design of fuzzy rules. On the other hand, instead of the ordinary least squares (OLS) that was commonly used in FRBFNN, the weighted least squares (WLS) method is exploited here to estimate the coefficients of the consequent's polynomials. With the use of WLS, FRBFNN exhibits different types of polynomials, which can vary from one rule to another.

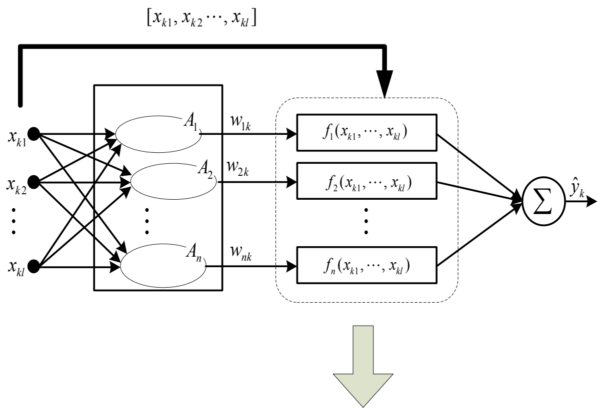
## 2. Architectures and Learning of the FRBFNN

In "conventional" RBFNNs, the output of the RBFNN comes as a weighted sum of the activation levels of the individual RBFs. RBFs realized in the form of some Gaussian function are the activation functions of hidden nodes. Fuzzy clustering may be used to determine the number of RBFs, as well as a position of their centers and the values of the widths [20]. The gradient - method or the least square algorithm is used to realize parametric learning to deal with the conclusions of the rules [21, 22]. In comparison with Fuzzy inference systems (FIS), the number of the RBF units is equal to the number of the fuzzy "if-then" rules present in the FIS. Unlike RBFNNs, in FRBFNNs the FCM is used to form the receptive fields (RBFs), and the RBFs do not assume any explicit functional form (such as Gaussian, ellipsoidal, triangular and others). The FRBFNN has the advantage by providing a through coverage of the entire input space. Given the zero-order polynomial used as a local model, the resulting accuracy of the model becomes limited.

### 2.1 Architecture of FRBFNN Based on IG and WLS

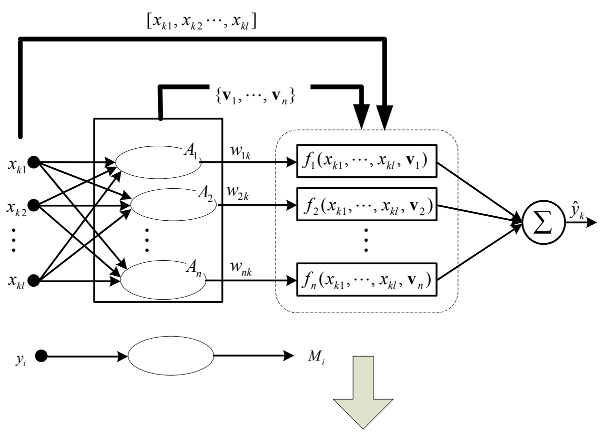
This section we discuss the FRBFNN whose architecture is extended by IG and WLS. In essence, information granules [23-25] are viewed as highly related collections of objects (data points, in particular) drawn together by some criteria of proximity, similarity, or functionality.

Fig. 1 illustrates a comparison of two different types of




$\mathbf{R}^i : IF\ \mathbf{x}_k\ is\ inclued\ in\ cluster\ A_i\ THEN\ y_{ki} = f_i(\mathbf{x}_{k1}, \mathbf{x}_{k2}, ..., \mathbf{x}_{kl})$

(a) Without use of IG




$\mathbf{R}^i : IF\ \mathbf{x}_k\ is\ inclued\ in\ cluster\ A_i\ THEN\ y_{ki} - M_i = f_i(\mathbf{x}_{k1}, \mathbf{x}_{k2}, ..., \mathbf{x}_{kl}, \mathbf{v}_i)$

(b) Use of IG

**Fig. 1.** Comparison of two different types of FRBFNNs

FRBFNN. In the IG-based FRBFNN, information granules (the centers of individual clusters) and activation levels (degree of memberships) are determined by means of the FCM. As shown in Fig. 1, it is clear that the difference between the IG-FRBFNN and our previous FRBFNN [19] arises at the consequent part of fuzzy rules.

The FRBFNN based on IG (see Fig. 1(b)) can be represented in form of "if-then" fuzzy rules

$$\begin{aligned} &\mathbf{R}^i : IF\ \mathbf{x}_k \text{ is inclued in cluster } A_i \\ &THEN\ y_{ki} - M_i = f_i(\mathbf{x}_k, \mathbf{v}_i) \end{aligned} \tag{1}$$

Where $\mathbf{R}^i$ is the $i$th fuzzy rule, $i$=1, ⋯, $n$, $n$ is the number of fuzzy rules (the number of clusters), $f_i(\mathbf{x}_k, \mathbf{v}_i)$ is the consequent polynomial of the $i$th fuzzy rule, i.e., a local model representing input-output relationship of the $i$th sub-space (local area). $w_{ik}$ is the degree of membership (viz. activation level) of the $i$th local model while $\mathbf{v}_i$ =[$v_{i1}$ $v_{i2}$

… $v_{il}]^T$ is the $i$th prototype.

Four types of polynomials, namely a constant type (a zero-order polynomial), linear type (a first-order polynomial), quadratic type (a second-order polynomial), and a modified quadratic type (a modified second-order polynomial) are considered as the type of consequent part of fuzzy rules. One of the four types is selected for each sub-space as the result of the optimization, which will be described later in this study. We stressed that the RBFNN using IG does not suffer from the curse of dimensionality (as all variables are considered en block). As a result, more accurate and compact models with a small number of fuzzy rules by using high-order polynomials may be constructed. Here we admit the consequent polynomials to be in of one of the following forms

Type 1 Zero-order polynomial (constant type):

$$f_j(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) = a_{j0}$$

Type 2 First-order polynomial (linear type):

$$f_j(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) = a_{j0} + a_{j1}(x_1 - V_{j1}) + \cdots + a_{jk}(x_k - V_{jk})$$

Type 3 Second-order polynomial (Quadratic type):

$$\begin{aligned} f_j(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) = {} & a_{j0} + a_{j1}(x_1 - V_{1j}) + \cdots + a_{jk}(x_k - V_{kj}) \\ & + a_{j(k+1)}(x_1 - V_{1j})^2 + a_{j(2k)}(x_k - V_{kj})^2 + \cdots \\ & + a_{j(2k+1)}(x_1 - V_{1j})(x_2 - V_{2j}) + \cdots \\ & + a_{j((k+2)(k+1)/2)}(x_{k-1} - V_{(k-1)j})(x_k - V_{kj}) \end{aligned}$$

Type 4 Modified second-order polynomial (Modified Quadratic type):

$$\begin{aligned} f_j(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) = {} & a_{j0} + a_{j1}(x_1 - V_{1j}) + \cdots + a_{jk}(x_k - V_{kj}) + \cdots \\ & + a_{j(k+1)}(x_1 - V_{1j})(x_2 - V_{2j}) + \cdots + a_{j(k(k+1)/2)}(x_{k-1} - V_{(k-1)j})(x_k - V_{kj}) \end{aligned}$$

The determination of the numeric output of the model, based on the activation levels of the rules, is given in the form

$$\hat{y}_k = \sum_{i=1}^{n} w_{ik} f_i(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) \tag{2}$$

Next, we consider the design of IG-based FRBFNN by means of WLS. A comparison of OLS and WLS in the design of IG-based FRBFNN is shown in Fig. 2. Both OLS and WLS can be used to estimate the coefficients of the consequent part of fuzzy rules. However, OLS can only deal with the same polynomial types of consequence part of fuzzy rules, while WLS can handle the different polynomial types of consequent part of fuzzy rules.

### 2.2 Learning of the consequent part of the FRBFNN

Regarding the learning of the consequent part of the rules, we consider two types in this study. One is the ordinary least squares method (OLS), while the other is a weighted least squares method (WLS).

**Fuzzy Rules:**

$\mathbf{R}^i$ : *IF* $\mathbf{x}_k$ *is inclued in cluster* $A_i$ *THEN* $y_{ki} - M_i = f_i(\mathbf{x}_{k1}, \mathbf{x}_{k2}, \ldots, \mathbf{x}_{kl}, \mathbf{v}_i)$

**OLS: Same type, different coefficients**

**e.g.**

$$y_1 - M_1 = a_{10} + a_{11}(x_1 - v_{11}) + a_{12}(x_2 - v_{12})$$

$$y_2 - M_2 = a_{20} + a_{21}(x_1 - v_{21}) + a_{22}(x_2 - v_{22})$$

$$y_3 - M_3 = a_{30} + a_{31}(x_1 - v_{31}) + a_{32}(x_2 - v_{32})$$

$$y_4 - M_4 = a_{40} + a_{41}(x_1 - v_{41}) + a_{42}(x_2 - v_{42})$$

(a) Use of OLS

**Fuzzy Rules:**

$\mathbf{R}^i$ : *IF* $\mathbf{x}_k$ *is inclued in cluster* $A_i$ *THEN* $y_{ki} - M_i = f_i(\mathbf{x}_{k1}, \mathbf{x}_{k2}, \ldots, \mathbf{x}_{kl}, \mathbf{v}_i)$

**WLS: different type, different coefficients**

**e.g.**

$$y_1 - M_1 = a_{10}$$

$$y_2 - M_2 = a_{20} + a_{21}(x_1 - v_{21}) + a_{22}(x_2 - v_{22})$$

$$y_3 - M_3 = a_{30} + a_{31}(x_1 - v_{31}) + a_{32}(x_2 - v_{32}) + a_{33}(x_1 - v_{31})^2 + a_{34}(x_2 - v_{32})^2$$

$$y_4 - M_4 = a_{40} + a_{41}(x_1 - v_{41}) + a_{42}(x_2 - v_{42})$$

(b) Use of WLS

**Fig. 2.** Comparison of OLS and WLS in the design of IG-based FRBFNNs:

#### *2.2.1 OLS Learning*

OLS is a well-known global learning algorithm that minimizes an overall squared error JG between output of the model and the experimental data.

$$J_G = \sum_{k=1}^{m} \left( y_k - \sum_{i=1}^{n} w_{ik} f_i(\mathbf{x}_k - \mathbf{v}_i) \right)^2 \tag{3}$$

where $w_{ik}$ is the normalized firing (activation) level of the $i$-th rule, $\mathbf{v}_i$ is the prototype (centroid), $\mathbf{x}_k$ and $y_k$ are input data and output data, respectively.

The performance index $J_G$ can be expressed in a concise form as follows

$$J_G = (\mathbf{Y} - \mathbf{Xa})^T (\mathbf{Y} - \mathbf{Xa}) \tag{4}$$

where $\mathbf{a}$ is the vector of coefficients of the polynomial, $\mathbf{Y}$ is the output vector of real data, $\mathbf{X}$ is matrix which rearranges input data, information granules (centers of each cluster) and activation level. In case all consequent polynomials are linear (first-order polynomials), $\mathbf{X}$ and $\mathbf{a}$ can be expressed as follows

$$\mathbf{X} = \begin{bmatrix} 1 & w_{11}(x_{11}-v_{11}) & \cdots & w_{11}(x_{l1}-v_{1l}) & \cdots & \cdots & 1 & w_{n1}(x_{11}-v_{n1}) & \cdots & w_{n1}(x_{l1}-v_{nl}) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & w_{1k}(x_{12}-v_{11}) & \cdots & w_{1k}(x_{l1}-v_{1l}) & \cdots & \cdots & 1 & w_{nk}(x_{12}-v_{n1}) & \cdots & w_{nk}(x_{l1}-v_{nl}) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & w_{1m}(x_{1m}-v_{11}) & \cdots & w_{1m}(x_{lm}-v_{1l}) & \cdots & \cdots & 1 & w_{nm}(x_{1m}-v_{n1}) & \cdots & w_{nm}(x_{lm}-v_{nl}) \end{bmatrix}$$

$$\mathbf{a} = [a_{10} \quad a_{11} \quad \cdots \quad a_{1l} \quad \cdots \quad \cdots \quad a_{n0} \quad a_{n1} \quad \cdots \quad a_{nl}]^T$$

The optimal values of the coefficients of the consequent are determined in the form

$$\mathbf{a} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \tag{5}$$

### 2.2.2 WLS Learning

The WLS method is to determine the coefficients of the model through the minimization of the objective function $J_L$. The main difference between the WLS and OLS is the weighting scheme, which comes as a part of the WLS and makes its focused on the corresponding local model.

$$J_L = \sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}(y_k - f_i(\mathbf{x}_k - \mathbf{v}_i))^2 \tag{6}$$

The performance index $J_L$ can be rearranged as

$$\begin{aligned} J_L &= \sum_{i=1}^{n}(\mathbf{Y} - \mathbf{X}_i\mathbf{a}_i)^T \mathbf{W}_i (\mathbf{Y} - \mathbf{X}_i\mathbf{a}_i) \\ &= \sum_{i=1}^{n}\left(\mathbf{W}_i^{1/2}\mathbf{Y} - \mathbf{W}_i^{1/2}\mathbf{X}_i\mathbf{a}_i\right)^T \left(\mathbf{W}_i^{1/2}\mathbf{Y} - \mathbf{W}_i^{1/2}\mathbf{X}_i\mathbf{a}_i\right) \end{aligned} \tag{7}$$

where $\mathbf{a}_i$ is the vector of coefficients of $i$th consequent polynomial (local model), $W_i$ is the diagonal matrix (weighting factor matrix) which involves the activation levels. $\mathbf{X}_i$ is a matrix which includes input data shifted by the locations of the information granules (more specifically, centers of clusters). In case the consequent polynomial is Type 2 (linear or a first-order polynomial), $\mathbf{X}_i$ and $\mathbf{a}_i$ read as follows

$$\mathbf{W}_i = \begin{bmatrix} w_{i1} & 0 & \cdots & 0 \\ 0 & w_{i2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_{im} \end{bmatrix} \in \Re^{m\times m},$$

$$\mathbf{X}_i = \begin{bmatrix} 1 & (x_{11}-v_{i1}) & \cdots & (x_{l1}-v_{il}) \\ 1 & (x_{12}-v_{i1}) & \cdots & (x_{l1}-v_{il}) \\ 1 & \vdots & \ddots & \vdots \\ 1 & (x_{1m}-v_{i1}) & \cdots & (x_{lm}-v_{il}) \end{bmatrix}, \quad \mathbf{a}_i = [a_{i0}\ a_{i1} \cdots a_{il}]$$

For the local learning algorithm, the objective function is defined as a linear combination of the squared error being the difference between the data and the corresponding output of each fuzzy rule, based on a weighting factor matrix. The weighting factor matrix, $\mathbf{W}_i$, captures the activation levels of input data to $i$th sub-space. In this sense we can consider the weighting factor matrix to form a discrete version of the fuzzy linguistic representation for the corresponding sub-space.

The coefficients of the consequent polynomial of the $i$th fuzzy rule can be determined in a usual manner, namely

$$\mathbf{a}_i = (\mathbf{X}_i^T\mathbf{W}_i\mathbf{X}_i)^{-1}\mathbf{X}_i\mathbf{W}_i\mathbf{Y} \tag{8}$$

Notice that the coefficients of the consequent polynomial of each fuzzy rule are computed independently using a certain subset of the training data. These computations can be implemented in parallel and in this case the overall computing load becomes unaffected by the total number of the rules.

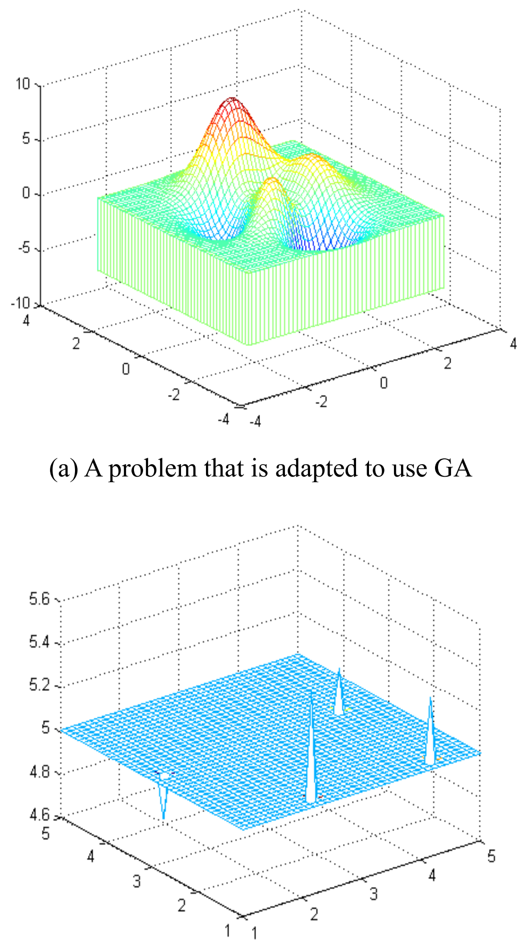
## 3. Multiobjective Optimization of the FRBFNN

The objective of learning method is to develop the FRBFNN satisfying the criteria of accuracy as well as simplicity (complexity). The complexity of the model can be expressed through the rate of interaction between the local models, an issue being closely related with the so-called overlapping criterion [26]. As far as the structure of the FRBFNN is concerned, there are two components to be considered (optimized), i.e., the number of fuzzy rules, and the order of polynomial in each consequent part. With regard to parameter identification, there is an adjustable fuzzification coefficient used in the FCM. These three components affect the performance of the FRBFNN and have to be optimized. Hence, the goal of multiobjective optimization of the FRBFNN is to determine a value of the fuzzification coefficient, the number of fuzzy rules and the type of polynomial being used in each conclusion part of the rules by simultaneously maximizing the accuracy and minimizing the complexity of the model. Here we consider the structural as well parametric optimization realized within the framework of an IMSSA. We start with the introduction of IMSSA and then discuss the arrangement and interpretation of the solutions and multiobjective functions used for the optimization of the FRBFNN.

### 3.1 Improved multi-objective space search algorithm

Generally, multiobjective optimization generates a Pareto front, which is the set of non-dominated solutions. A solution is said to be non-dominated if it is impossible to improve one objective of the solution without worsening at least one other objectives which is presented in the problem. In this section, we develop an improved multiobjective space search algorithm (IMSSA).

We first introduce the single-objective space search algorithm (SSA), an adaptive heuristic optimization algorithm whose search method comes with the analysis of the solution space [27, 28]. The idea of the SSA comes from the analysis of the solution space. In fact, a precondition should be satisfied when evolutionary algorithm can find the optimal solution. The precondition is that, in most of local areas, a point (solution) and the other points located in the point's adjacent space have the similar values of the objective function (fitness values). In other words, in most of local areas, a solution with better fitness is closer to the optimal solution. Fig. 3 depicts the comparison of two different optimization problems using the genetic algorithm (GA).



(a) A problem that is adapted to use GA

(b) A problem that is not adapted to use GA

**Fig. 3.** Comparison of two types of optimization problems

The search method of SSA is based on the operator of space search, which generates two basic steps: generate new subspace (local area) and search the new space. Regarding the generation of the new space, we consider two cases: (a) space search based on M selected solutions (denoted here as Case I), and (b) space search based on the current best solution (Case II).

To improve the performance of SSA, we developed a new spacer search operator of Case I with the aid of an orthogonal approach [29]. The salient feature of this new operator is to incorporate an experimental design method called orthogonal design into the space search operator of SSA. As a result, it can search the solution space more efficiently and it is well suited for parallel implementation.

The steps that generate new solutions using this space search operator are as follows:

Step1. Record the M selected solutions for space search in a set $P_m$

Step2. Generate a new solution set $P_{m1}$, where a new solution of the set $a_{ij}$ is generated by using the orthogonal approach.

Step2.1. Set $J = \log_Q((Q-1)*n+1)$, where $Q$ is the factor of orthogonal experiments and we fix it as 5.

Step2.2. Set $a_{ij} = ((i-1)/Q^{J-k}) \bmod Q$, where $k = 2,3,\ldots,J, j = \dfrac{Q^{k-1}-1}{Q-1}+1, i = 1,2,\ldots,Q^J$.

Step2.3. Calculate the basic column of M solutions by using the following expression:

$$a_j + (s-1)*(Q-1)+t = (a_s * t + a_j) \bmod Q$$

where $k = 2,3,\ldots,J, j = \dfrac{Q^{k-1}-1}{Q-1}+1, s = 1,2,\ldots,j-1,$ and $t = 1,2,\ldots,Q-1$.

Step3. Selected M solutions with best fitness from $P_m \cup P_{m1}$.

With understanding of the improved SSA, we can develop an IMSSA. As so far, several techniques have been incorporated into multi-objective optimization algorithms in order to improve convergence to the Pareto front as well as produce a well-distributed Pareto front. These techniques include elitism, diversity operators, mutation operators, and constraint handling. The technique of non-dominate sort with the aid of the crowding distance [11] is used in the IMSSA. The details are presented in Table 1. The operator of the search space is realized by means of the following two basic steps: we generate a new subspace (local area) and realize search therein. The non-dominated sort is realized with the aid of estimation of the crowding distance among solutions in the current solution set. The termination condition of the IMSSA is such that all the solutions in the current population have the same fitness, or terminate after a certain fixed number of generations.

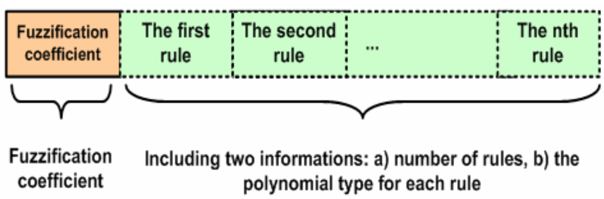
**Table 1.** The flow of computing of the IMSSA

| |
|---|
| *BEGIN* |
| *Initialize the solution set S (population)* |
| *Evaluate each solution in the solution set S* |
| *Sort all current solutions in S with the aid of non-domination strategy* |
| *While { the termination conditions are not met }* |
| *Select the solutions from S* |
| *Search solution space (case I)* |
| *Search solution space (case II)* |
| *Sort all current solutions in S with the aid of non-domination strategy* |
| *End while* |
| *Report the optimal solutions* |
| *End* |

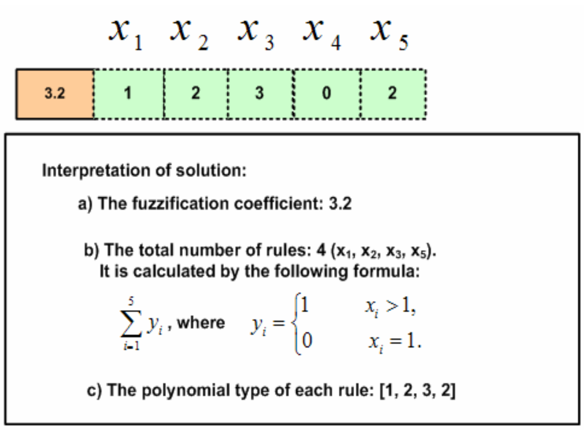
### 3.2 An Arrangement and Interpretation of Solutions

In the IMSSA, a solution is represented as a vector comprising the fuzzification coefficient, the number of fuzzy rules and the type of polynomial standing in the consequent part for each fuzzy rule as illustrated in Fig. 4, The length of the solution vector corresponds to the maximal number of fuzzy rules to be considered in the optimization.

Fig. 4(b) offers an interpretation of the content of the particle in case the upper bound of search space of the fuzzy rule is set to 6. As the number of rules, and the orders of the polynomials have to be integer number, we round off these values standing in the particle to the nearest integer. The fuzzification coefficient is 3.2, while the number of the rule is 4. The first local model is of constant type while the other three local models are linear and quadratic.




(a)




(b)

**Fig. 4.** Solution composition of IMSSA and its interpretation: (a) Arrangement of the solution for the IMSSA; (b) Example of interpretation of solution

### 3.3 Objective Functions of the IG-FRBFNN

Three objective functions are used to evaluate the accuracy, the complexity and the interpretability of an IG-FRBFNN. These three objective functions are mean squared error (MSE) or the root mean squared error (RMSE), entropy of partition and the total number of the coefficients of the polynomials to be estimated, respectively.

The MSE and RMSE serving as the accuracy criterion of the IG-RBFNN are given as

$$PI(or\ E\_PI) = \begin{cases} \sqrt{\dfrac{1}{m}\sum_{i=1}^{m}(y_i - y_i^*)^2}, & (RMSE) \\ \dfrac{1}{m}\sum_{i=1}^{m}(y_i - y_i^*)^2. & (MSE) \end{cases} \tag{9}$$

To evaluate the structural complexity of the model, we consider the entropy of the partition [15]. The entropy of partition reflects a degree of overlap between the regions of the fuzzy sets. Considering all samples of the training dataset, the entropy of the partition reads as

$$H = -\sum_{j=1}^{n}\sum_{k=1}^{m} w_{ik}\log(w_{ik}) \tag{10}$$

As a measure of simplicity, we consider the overall number of coefficients of the local models, which is computed as

$$N = \sum_{j=1}^{n} C_i ,$$

$$C_i = \begin{cases} 1 & \text{if type of local model is constant} \\ 1+l & \text{if type of local model is linear form} \\ 1+l+(l^2-l)/2+l & \text{if type of local model is quadratic form} \\ 1+l+(l^2-l)/2 & \text{if type of local model is modified quadratic form} \end{cases} \tag{11}$$

where, $C_i$ is the number of coefficients of the $i$th polynomial and $l$ stands for the number of input variables.

In a nutshell, we find the Pareto optimal sets and Pareto front by minimizing {E, H, N} by means of the IMSSA. This leads to easily interpretable, simple, and accurate fuzzy models.

## 4. Experimental Studies

In this study, we carried out two cases of multiobjective optimization. In the first case, we used the OLS method to estimate the coefficients of the polynomial. In the second scenario, the WLS method is used. Table 2 summarizes the list of parameters and boundaries of the search spaces of solutions used by the IMSSA.

**Table 2.** List of the parameters of IMSSA

| IMSSA parameters | |
|---|---|
| Number of generations | 150 |
| Number of solutions | 200 |
| Number of solutions (case I) | 8 |
| Boundary of search space of decision variables | |
| Fuzzification coefficient | 1.01 ~ 5 |
| Number of rules | 2~20 |
| Order of polynomials | 1~4 (Type) |

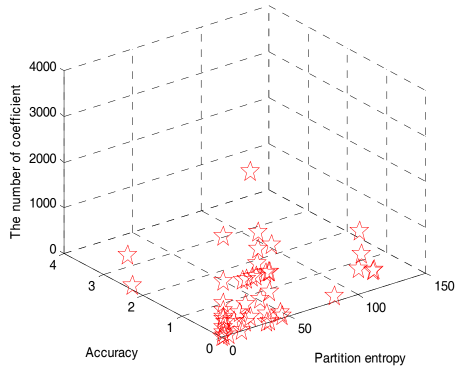
### 4.1 Gas furnace process

The first well-known dataset is time series data of a gas furnace utilized by Box and Jenkins. The time series data, which consists of 296 input-output pairs resulting from the gas furnace process has been intensively studied in the previous literature [30-34]. The delayed terms of methane gas flow rate $u(t)$ and carbon dioxide density $y(t)$ are used as six input variables with vector formats such as $[u(t-3), u(t-2), u(t-1), y(t-3), y(t-2), y(t-1)]$. $y(t)$ is used as output variable. The gas furnace process is partitioned into two parts. The first 50% of data set is used for the construction of the fuzzy model. The remaining 50% data set, the testing data set, is used to quantify the predictive quality of the model. The performance index is specified as the MSE given by (9).

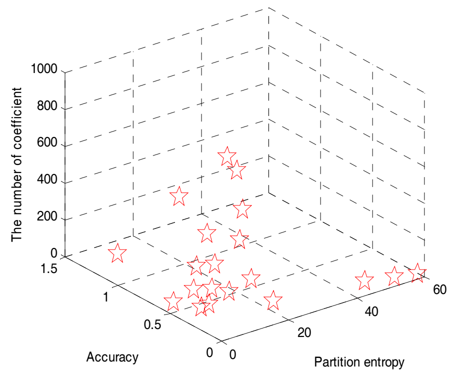
Fig. 5 illustrates the Pareto fronts generated by means of the IMSSA when using WLS and OLS. Generally, as it could have been expected, by increasing the total number of coefficients, the accuracy of IG-FRBFNN becomes better. The total number of coefficients relates directly to the interpretability of the model.

The IMSSA just provides the Pareto optimal sets, which are non-dominated solutions. Selecting the optimal solution within the Pareto optimal sets is a different from the IMSSA. If we place more emphasis on the accuracy than the simplicity (complexity) of the model, then we define the particle having the best accuracy within the Pareto optimal set to be the best solution.

The performance of the proposed model is compared with the performance of some other models available in the literature; refer to Table 3. Here PI stands for the performance index of accuracy for training data and E_PI means accuracy for the testing data. Local models of other models have same type of fuzzy rule such as constant or linear form. The proposed model can have different types of local models. In this comparison, the proposed model,




(a) Use of OLS




(b) Use of WLS

**Fig. 5.** Pareto front produced by IMSSA

having a small number of rules, shows better accuracy, while the model leads to disadvantage of having a large number of coefficients of local model in case of selecting the quadratic form.

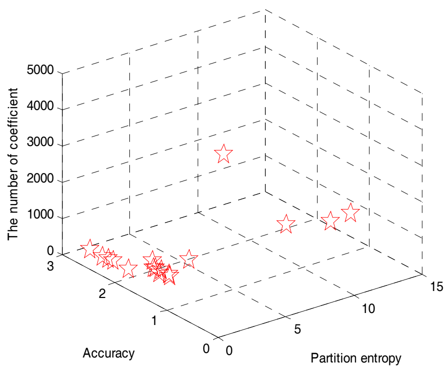
**Table 3**. Results of selected models (Gas)

| Model | | Pit | PI | E_PI | No.of rules |
|---|---|---|---|---|---|
| Pedrycz's model [30] | | 0.776 | | | 20 |
| Tong's model [31] | | 0.469 | | | 19 |
| Xu's model [32] | | 0.328 | | | 25 |
| Oh et al.'s Model [33] | Simplified | | 0.024 | 0.328 | 4 |
| | Linear | | 0.022 | 0.326 | 4 |
| | | | 0.021 | 0.364 | 6 |
| HCM+GA [34] | Simplified | | 0.035 | 0.289 | 4 |
| | | | 0.022 | 0.333 | 6 |
| | Linear | | 0.026 | 0.272 | 4 |
| | | | 0.020 | 0.264 | 6 |
| Our model | Sequential tuning | | 0.017 | 0.266 | 4 |
| | Simultaneous tuning | | 0.015 | 0.260 | 6 |

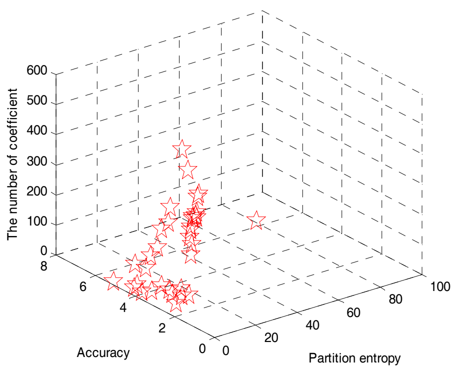
### 4.2 Automobile Miles Per Gallon (MPG) Data

We consider the automobile MPG data (ftp://ics.uci.edu/pub/machine-learning-databased/auto-mpg) with the output being the automobile's fuel consumption expressed in miles per gallon. The data set includes 392 input-output pairs (after removing incomplete instances) where the input space involves 8 input variables. To come up with a quantitative evaluation of the fuzzy model, we use the standard RMSE performance index as the one described by (9).

The automobile MPG data is partitioned into two separate parts. The first 235 data pairs are used as the training data set for IG-FRBFNN while the remaining 157 pairs are the testing data set for assessing the predictive performance. Fig. 6 illustrates Pareto fronts generated by means of the IMSSA in case of using WLS and OLS. Table 4 offers the values of the performance indexes of the IG-FRBFNN having the best accuracy within Pareto optimal set.




(a) Use of OLS




(b) Use of WLS

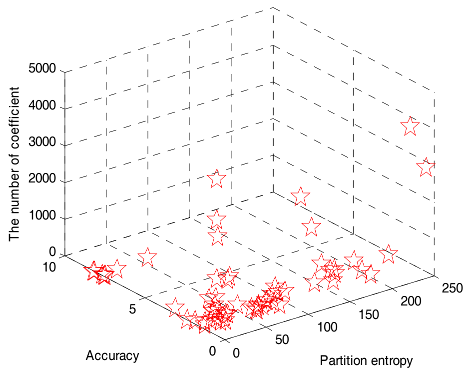
**Fig. 6.** Pareto front produced by IMSSA

**Table 4.** Results of comparative analysis (MPG)

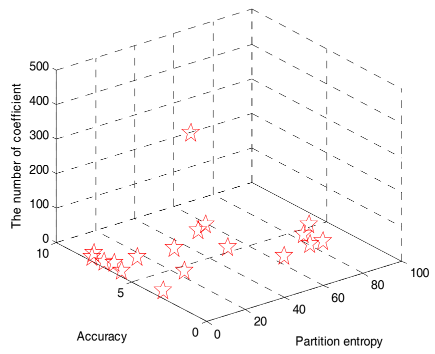
| Model | | No. of rules | Order of Polynomial | PI | E_PI |
|---|---|---|---|---|---|
| Linguistic model [5] | | 36 | Constant | 2.86 | 3.24 |
| RBFNN [6] | | 36 | Constant | 3.24 | 3.62 |
| Functional RBFNN[6] | | 33 | Constant | 2.41 | 2.82 |
| Our model | OLS | 2 | Modified quadratic | 2.67 | 2.90 |
| | WLS | 4 | Different types | 2.82 | 2.54 |

### 4.3 Boston housing data

Last we experiment with the Boston housing data set. This data set concerns a description of real estate in the Boston area where houses are characterized by features such as crime rate, size of lots, number of rooms, age of houses, etc. and their median price. The dataset consists of 506 14-dimensional data. The performance index is defined as the RMSE as given by (9).




(a) Use of OLS




(b) Use of WLS

**Fig. 7.** Pareto front produced by IMSSA

We consider the Boston Housing data set, which is split into two separate parts. The construction of the fuzzy model is completed for 253 data points being regarded as a training set. The rest of the data set is retained for testing purposes. Fig. 7 shows Pareto fronts generated by means of the IMSSA in case of using WLS and OLS. Table 5 shows the results of comparative analysis of the proposed model when being contrasted with other models.

**Table 5.** Results of comparative analysis (Housing)

| Model | | PI | E_PI | No. of rules |
|---|---|---|---|---|
| RBFNN [6] | | 6.63 | 7.14 | 20 |
| SVR [35] | | 1.17 | 5.84 | |
| FNN [36] | | 3.76 | 4.08 | 21 |
| NN | | 3.27 | 5.14 | 24 |
| FPNN | | 3.51 | 16.93 | 16 |
| Our model | OLS | 3.41 | 3.71 | 2 |
| | WLS | 2.02 | 3.40 | 4 |

## 5. Conclusions

In this study, we have introduced an information granulation-based FRBFNNs by means of IMSSA and WLS. The proposed IMSSA was exploited as a multi-objective optimization vehicle to carry out the structural as well as parametric optimization of the FRBFNN. The OLS and the WLS learning to estimate the coefficients of the consequent parts of the information granulation-based FRBFNN are presented and compared. From the results of the IMSSA, we have shown that there exists the accuracy-interpretability tradeoff in an identification of the FRBFNN.

## Acknowledgements

This work was supported by National Research Foundation of Korea Grant funded by the Korean Government (NRF-2010-D00065 & NRF-2009-0074928) and supported partially by the GRRC program of Gyeonggi province [GRRC SUWON 2011-B2, Center for U-city Security & Surveillance Technology].

## References

[1] S. Mitra, J. Basak, "FRBF: A Fuzzy Radial Basis Function Network," *Neural Comput. Applic.,* vol. 10, pp. 244-252, 2001.

[2] F. Behloul, B.P.F. Lelieveldt, A.Boudraa, and J.H.C. Reiber, "Optimal design of radial basis function neural networks for fuzzy-rule extraction in high dimensional data," *Pattern Recognition*, vol. 35, pp. 659-675, 2002.

[3] W. Huang, L. Ding, "Project-scheduling problem with random time-dependent activity duration times," *IEEE Transactions on Engineering Management,* vol. 58, no. 2, pp. 377–387, 2011.

[4] F.J. Lin, L.T. Teng, J.W. Lin, S.Y. Chen, "Recurrent Functional-Link-Based Fuzzy-Neural-Network-Controlled Induction-Generator System Using Improved Particle Swarm Optimization," *IEEE Trans. Indust. Elect*., vol. 56, no. 5, pp. 1557-1577, 2009.

[5] W. Pedrycz, K.C Kwak, "Linguistic models as a framework of user-centric system modeling," *IEEE Trans. Syst., man cybern. –PART A : Systems and humans*, vol. 36, no. 4, pp. 727–745, 2006.

[6] W. Pedrycz, H.S. Park, S.K. Oh, "A granular-oriented development of functional radial basis function neural networks," *Neurocomputing*, vol. 72, pp. 420–435, 2008.

[7] D.R. Marylyn, A.M. Alison, S.B. William, S.C. Christopher, H.M. Jason, "Genetic programming neural networks: A powerful bioinformatics tool for human genetics," *Applied Soft Computing*, vol. 7, Issue 4, pp. 471-479, 2007.

[8] Y. Jin, "Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 2, pp. 212–221, 2000.

[9] M. Setnes, H. Roubos, "GA-based modeling and classification: complexity and performance," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 5, pp. 509–522, 2000.

[10] K. Deb, A. Pratab, S. Agrawal, T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.,* vol. 6, pp. 182-197, 2002.

[11] G. Avigad, A. Moshaiov, "Interactive Evolutionary Multiobjective Search and Optimization of Set-Based Concepts," *IEEE Trans. Syst., Man cybern.-Part B,* vol. 38, nol. 2, pp. 381-403, 2008.

[12] B.Zafer, "Adaptive genetic algorithms applied to dynamic multiobjective problems," *Applied Soft Computing*, vol. 7, Issue 3, pp. 791-799, 2007.

[13] S. Chuan, Y. Zhenyu, S. Zhongzhi, Z. Lei, "A fast multi-objective evolutionary algorithm based on a tree structure," *Applied Soft Computing,* vol. 10, Issue 2, 468-480, 2010.

[14] K.T. Parveen, B. Sanghamitra, K.P. Sankar, "Multi-Objective Particel Swarm Optimization with time variant inertia and acceleration coefficients," *Information Sciences*, vol. 177, no. 22, pp. 5033-5049, 2007.

[15] S. Masato, G. Mitsuo, "Fuzzy multiple objective optimal system design by hybrid genetic algorithm," *Applied Soft Computing*, vol. 2, Issue 3, pp. 189-196, 2003

[16] M. Delgado, M.P. Ceullar, M.C. Pegalajar, "Multiobjective Hybrid Optimization and Training of Recurrent Neural Networks," *IEEE Trans. Syst., Man cybern. –Part B,* vol. 38, nol. 2, pp. 381-403, 2008.

[17] C. Marco, L. Beatrice, M. Francesco, "On reducing computational overhead in multi-objective genetic Takagi-Sugeno fuzzy systems," *Applied Soft Computing*, vol. 11, Issue 1, pp. 675-688, 2011.

[18] Z.Yong, W. Xiao-bei, X. Zong-yi, H. Wei-li, "On generating interpretable and precise fuzzy systems based on Pareto multi-objective cooperative co-evolutionary algorithm," *Applied Soft Computing*, vol. 11, Issue 1, pp. 1284-1294, 2011.

[19] W. Huang, L. Ding, S.K. Oh, "Design of fuzzy radial basis function neural networks with the aid of multi-objective optimization based on simultaneous tuning," *Eighth International Symposium on Neural Networks*, LNCS 6677, pp. 264-273, 2011.

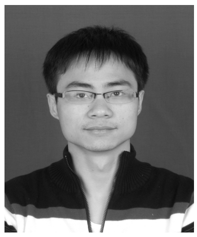
[20] A. Staiano, R. Tagliaferri, W. Pedrycz, "Improving RBF networks performance in regression tasks by means of a supervised fuzzy clustering," *Neurocomputing*, vol. 69, pp. 1570-1581, 2006.

[21] X. Hong, S. Chen, "A New RBF Neural Network with Boundary Value Constraints," *IEEE Trans. Syst.,*

*Man Cybern. –PART B*, vol. 39, nol. 1, pp. 298-303, 2009.
[22] C.M. Huang, F.L. Wang, "An RBF Network with OLS and EPSO Algorithms for Real-Time Power Dispatch," *IEEE Trans. Power Syst.,* vol. 22, no. 1, pp.96-104, 2007.
[23] F. Hoffmann, "Combining boosting and evolutionary algorithms for learning of fuzzy classification rules", *Fuzzy Sets and Syst.,*vol. 141, pp. 47-58, 2004.
[24] W. Pedrycz, K.C. Kwak, "Boosting of granular models," *Fuzzy Sets and Syst.,* vol. 157, pp. 2934-2953, 2006.
[25] W. Pedrycz, P. Rai,, "Collaborative clustering with the use of fuzzy C-Means and its quantification", *Fuzzy Sets and Syst.,* vol. 159, pp. 2399-2427, 2008.
[26] R.R. Yager, D.P. Filev, "Unified Structure and Parameter Identification," *IEEE Trans. Syst. Man Cybern*., vol. 23, no. 4, pp. 1198-1205, 1993.
[27] W. Huang, L. Ding, S.K. Oh, C.W. Jeong, S.C. Joo, "Identification of Fuzzy Inference System Based on Information Granulation," *KSII Transactions on Internet and Information Systems*, vol. 4, no. 4, pp. 575-594, 2010.
[28] W. Huang, S.K. Oh, L. Ding, H.K. Kim, S.C. Joo, "Identification of Fuzzy Inference Systems Using a Multi-objective Space Search Algorithm and Information Granulation," *Journal of Electrical Engineering & Technology*, vol. 6, no. 6, pp. 853-866, 2011.
[29] Q. Zhang, Y.W. Leung, "An orthogonal genetic algorithm for multimedia multicast routing," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 1, pp. 53-62, 1999.
[30] W. Pedrycz, "An identification algorithm in fuzzy relational system," *Fuzzy Sets Syst.,* vol. 13, pp. 153-167, 1984.
[31] R. M. Tong, "The evaluation of fuzzy models derived from experimental data," *Fuzzy Sets Syst.,* vol. 13, pp 1-12, 1980.
[32] C. W. Xu., Y. Zailu, "Fuzzy model identification self-learning for dynamic system" *IEEE Trans. Syst., Man cybern.,* vol. 17, no 4, pp, 683-689, 1987.
[33] S. K. Oh, W. Pedrycz, "Identification of Fuzzy Systems by means of an Auto-Tuning Algorithm and Its Application to Nonlinear Systems," *Fuzzy Sets and Syst.,* vol. 115, no 2, pp, 205-230, 2000.
[34] B. J. Park, W. Pedrycz, S. K. Oh, "Identification of Fuzzy Models with the Aid of Evolutionary Data Granulation," *IEE Proc.-Control Theory and Applications,* vol. 148, pp, 406-418, 2001.
[35] H.D. Chris, J.C. Burges, L. Kaufman, A. Smola, V. Vapnik, "Support vector regression machines," *Adv. Neural Inform. Process Syst.* 9 (1997) 155-161.
[36] A.F. Gobi, W. Pedrycz, "Fuzzy modeling through logic optimization", *Int. J. Approx. Reason.,* vol. 45, pp. 488-510, 2007.

**Wei Huang** He received the MS degree at school of Information Engineering from East China Institute of Technology, China, in 2006, and Ph.D. degree at State Key Laboratory of Software Engineering, Wuhan University, China, in 2011. He is currently a lecturer with the School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin, China. His research interests include evolutionary computation, operations research, fuzzy system, fuzzy-neural networks, and software reliability.

**Sung-Kwun Oh** He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Yonsei University, Seoul, Korea, in 1981, 1983, and 1993, respectively. During 1983-1989, he was a Senior Researcher of R&D Lab. of Lucky-Goldstar Industrial Systems Co., Ltd. From 1996 to 1997, he was a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada. He is currently a Professor with the Department of Electrical Engineering, University of Suwon, Suwon, South Korea. His research interests include fuzzy system, fuzzy-neural networks, automation systems, advanced computational intelligence, and intelligent control. He currently serves as an Associate Editor of the *KIEE Transactions on Systems and Control*, *International Journal of Fuzzy Logic* and *Intelligent Systems of the KFIS*, and *International Journal of Control, Automation, and Systems of* the ICASE, South Korea.



**Honghao Zhang** He received the MS degree at School of Information Science and Engineering from Ningbo University, China, in 2010. He is currently a teacher with the School of Tianjin University of Technology, Tianjin, China. His research interests include network security, trusted networks and next generation network.