

Research Article

Identification of Fuzzy Inference Systems by Means of a Multiobjective Opposition-Based Space Search Algorithm

Wei Huang^{1,2} and Sung-Kwon Oh³

¹ State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China

² School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin 300384, China

³ Department of Electrical Engineering, The University of Suwon, San 2-2, Wau-ri, Bongdam-eup, Gyeonggi-do, Hwaseong-si 445-743, Republic of Korea

Correspondence should be addressed to Sung-Kwon Oh; ohsk@suwon.ac.kr

Received 2 January 2013; Revised 25 February 2013; Accepted 25 February 2013

Academic Editor: Zhuming Bi

Copyright © 2013 W. Huang and S.-K. Oh. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We introduce a new category of fuzzy inference systems with the aid of a multiobjective opposition-based space search algorithm (MOSSA). The proposed MOSSA is essentially a multiobjective space search algorithm improved by using an opposition-based learning that employs a so-called opposite numbers mechanism to speed up the convergence of the optimization algorithm. In the identification of fuzzy inference system, the MOSSA is exploited to carry out the parametric identification of the fuzzy model as well as to realize its structural identification. Experimental results demonstrate the effectiveness of the proposed fuzzy models.

1. Introduction

Fuzzy modeling has been utilized in many fields for engineering, medical engineering, and even social science. Lots of diverse approaches to fuzzy modeling have been proposed in the past decades. Pioneering work by Pedrycz [1], Tong [2], Xu and Zailu [3], Sugeno and Yasukawa [4], Oh and Pedrycz [5], Chung et al. [6], and others [7] has studied different approaches to construct fuzzy models. All these methods reported are based on information granulation and optimization algorithms, and there is a lack of investigations of multiobjective identification of fuzzy models.

There have been a suite of studies focusing on multiobjective when designing fuzzy models. In the design of fuzzy models, two main and conflicting objectives are commonly considered. One is the accuracy and the other is the complexity. In the 1990s, the emphasis of modeling was positioned on the accuracy maximization. As powerful optimization tools in many science fields [8], various approaches such as Genetic Algorithms (GAs) and Particle Swarm Optimization (PSO)

[9, 10] have been proposed to improve the accuracy of fuzzy models. As a result of accuracy maximization, the complexity of the model increases. Some researchers attempted to simultaneously optimize the accuracy and the complexity of the fuzzy models [11, 12]. Since it is impossible to simultaneously optimize these objectives due to the existence of the accuracy-complexity tradeoff, the accuracy maximization and complexity minimization have been often cast in the setting of multiobjective optimization. A number of evolutionary algorithms (EAs) have been developed to solve multiobjective optimization problems such as micro-GA [13] and NSGA-II [14–16]. As a result, multiobjective optimization (MOO) techniques have been applied to the design of fuzzy models exhibiting high accuracy and significant interpretability [17, 18]. Nevertheless, when dealing with the fuzzy models, the previous studies lack an optimization vehicle which considers not only the solution space being explored but also the techniques of MOO.

In our previous study [19], we propose a design of fuzzy models based on multiobjective space search algorithm (MSSA). This work provides some enhancements of the fuzzy

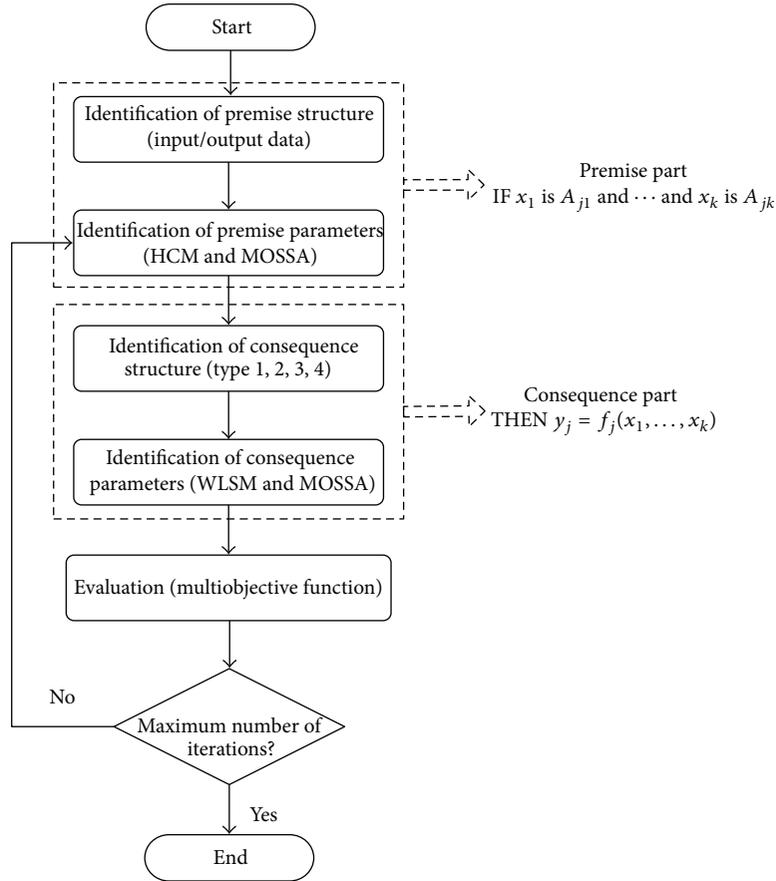


FIGURE 1: An overall scheme of fuzzy rule-based modeling.

modeling. Noticeably, some limitations of this fuzzy model are as follows: (1) the conventional MSSA is essentially stochastic search techniques, and such random mechanism leads to slow convergence speed; and (2) the flexibility and predictive ability are limited due to the only one single type of all polynomials of the consequence part of fuzzy rules.

In this study, we present a multiobjective opposition-based space search algorithm (MOSSA) and introduce a design of fuzzy model by means of the MOSSA and weighted least squares method (WLSM). The resulting fuzzy models address the two constraints mentioned above. First, the proposed MOSSA that is used as a vehicle to maximize the accuracy of the fuzzy inference system could come with more rapid convergence speed in comparison with the conventional MSSA. Second, instead of the ordinary least squares method (LSM), WLSM is used to estimate the coefficients of the consequent polynomials. With the use of WLSM, fuzzy model exhibits different types of polynomials, which can vary from one rule to another.

2. A Design of Fuzzy Inference Systems

Figure 1 depicts an overall scheme of fuzzy rule-based modeling. The identification procedure for fuzzy models is split

into two parts, namely, premise part and consequence part of the fuzzy rules.

2.1. Identification of Premise Part. The identification completed at the premise level consists of two main steps. First, we select the input variables x_1, x_2, \dots, x_k of the rules. Second, we form fuzzy partitions (by specifying fuzzy sets of well-defined semantics e.g., Low, High, etc.) of the spaces over which these individual variables are defined.

The identification of the premise part is completed in the following manner.

Given is a set of data $\mathbf{U} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_l; \mathbf{y}\}$, where $\mathbf{x}_k = [x_{1k}, \dots, x_{mk}]^T$, $\mathbf{y} = [y_1, \dots, y_m]^T$, where l is the number of variables and m stands for the number of data.

Step 1. Arrange a set of data \mathbf{U} into data set \mathbf{X}_k composed of the corresponding input and output data.

$$\mathbf{X}_k = [\mathbf{x}_k; \mathbf{y}]. \quad (1)$$

Step 2. Run the K -Means to determine the centers (prototypes) \mathbf{v}_{kg} of the data set \mathbf{X}_k .

Step 2.1. Arrange data set \mathbf{X}_k into c -clusters (in essence this is effectively the information granulation process).

Step 2.2. Calculate the centers \mathbf{v}_{kg} of each cluster as follows:

$$\mathbf{v}_{kg} = \{v_{k1}, v_{k2}, \dots, v_{kc}\}. \quad (2)$$

Step 3. Partition the corresponding input space using the prototypes of the clusters \mathbf{v}_{kg} . Associate each cluster with some meaning (semantics), say Small, Large, and so forth.

Step 4. Set the initial apexes of the membership functions using the prototypes \mathbf{v}_{kg} .

2.2. Identification of Consequence Part. The identification of the consequence part of the rules embraces two phases, namely, (1) a selection of the consequence variables of the fuzzy rules (identification of consequence structure) and (2) determination of the parameters of the consequence (identification of consequence parameters).

2.2.1. Identification of Consequence Structure. The identification of the conclusion parts of the rules deals with a selection of their structure (Types 1, 2, 3, and 4) that is followed by the determination of the respective parameters of the local functions occurring there. The consequence part of the rule that is extended form of a typical fuzzy rule in the TSK (Takagi-Sugeno-Kang) fuzzy model has the form

$$\begin{aligned} R^j: & \text{ If } x_1 \text{ is } A_{1c} \text{ and } \dots \text{ and } x_k \text{ is } A_{kc} \\ & \text{ then } y_j - M_j = f_j(x_1, \dots, x_k). \end{aligned} \quad (3)$$

Type 1 (simplified inference). Consider

$$f_j = a_{j0}. \quad (4)$$

Type 2 (linear inference). Consider

$$f_j = a_{j0} + a_{j1}(x_1 - V_{1j}) + \dots + a_{jk}(x_k - V_{kj}). \quad (5)$$

Type 3 (quadratic inference). Consider

$$\begin{aligned} f_j = & a_{j0} + a_{j1}(x_1 - V_{1j}) + \dots + a_{jk}(x_k - V_{kj}) \\ & + a_{j(k+1)}(x_1 - V_{1j})^2 + \dots + a_{j(2k)}(x_k - V_{kj})^2 \\ & + a_{j(2k+1)}(x_1 - V_{1j})(x_2 - V_{2j}) \\ & + \dots + a_{j((k+2)(k+1)/2)}(x_{k-1} - V_{(k-1)j})(x_k - V_{kj}). \end{aligned} \quad (6)$$

Type 4 (modified quadratic inference). Consider

$$\begin{aligned} f_j = & a_{j0} + a_{j1}(x_1 - V_{1j}) + \dots + a_{jk}(x_k - V_{kj}) \\ & + a_{j(k+1)}(x_1 - V_{1j})(x_2 - V_{2j}) \\ & + \dots + a_{j(k(k+1)/2)}(x_{k-1} - V_{(k-1)j})(x_k - V_{kj}), \end{aligned} \quad (7)$$

where R^j is the j th fuzzy rule, x_k represents the input variables, A_{kc} is a membership function of fuzzy sets, a_{jk} is a constant, V_{kj} and M_j are a center value of the input and output data, respectively, and n is the number of fuzzy rules.

The calculation of the numeric output of the model, based on the activation (matching) levels of rules there, relies on the following expression:

$$\begin{aligned} y^* &= \frac{\sum_{j=1}^n w_{ji} y_i}{\sum_{j=1}^n w_{ji}} = \frac{\sum_{j=1}^n w_{ji} (f_j(x_1, \dots, x_k) + M_j)}{\sum_{j=1}^n w_{ji}} \\ &= \sum_{j=1}^n \widehat{w}_{ji} (f_j(x_1, \dots, x_k) + M_j). \end{aligned} \quad (8)$$

Here, y^* is the inferred output value, and w_{ji} is the premise level of matching R^j (activation level).

2.2.2. Identification of Consequence Parameters. The identification of consequence parameters is completed with the aid of WLSM, which is to determine the coefficients of the model through the minimization of the objective function J_L . The main difference between the WLSM and the classical LSM is the weighting scheme, which comes as a part of the WLSM and makes its focused on the corresponding local model as follows:

$$J_L = \sum_{i=1}^n \sum_{k=1}^m w_{ik} (y_k - f_i(\mathbf{x}_k - \mathbf{v}_i))^2. \quad (9)$$

The performance index J_L can be rearranged as

$$\begin{aligned} J_L &= \sum_{i=1}^n (\mathbf{Y} - \mathbf{X}_i \mathbf{a}_i)^T \mathbf{W}_i (\mathbf{Y} - \mathbf{X}_i \mathbf{a}_i) \\ &= \sum_{i=1}^n (\mathbf{W}_i^{1/2} \mathbf{Y} - \mathbf{W}_i^{1/2} \mathbf{X}_i \mathbf{a}_i)^T (\mathbf{W}_i^{1/2} \mathbf{Y} - \mathbf{W}_i^{1/2} \mathbf{X}_i \mathbf{a}_i), \end{aligned} \quad (10)$$

where \mathbf{a}_i is the vector of coefficients of i th consequent polynomial (local model), and \mathbf{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 are read as follows:

$$\begin{aligned} \mathbf{W}_i &= \begin{bmatrix} w_{i1} & 0 & \dots & 0 \\ 0 & w_{i2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w_{im} \end{bmatrix} \in \mathfrak{R}^{m \times m}, \\ \mathbf{X}_i &= \begin{bmatrix} 1 & (x_{11} - v_{i1}) & \dots & (x_{1l} - v_{il}) \\ 1 & (x_{12} - v_{i1}) & \dots & (x_{1l} - v_{il}) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & (x_{1m} - v_{i1}) & \dots & (x_{lm} - v_{il}) \end{bmatrix}, \\ \mathbf{a}_i &= [a_{i0} \ a_{i1} \ \dots \ a_{il}]. \end{aligned} \quad (11)$$

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 subspace. In this sense we can consider the weighting factor matrix to form a discrete version of the fuzzy linguistic representation for the corresponding subspace.

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

$$\mathbf{a}_i = (\mathbf{X}_i^T \mathbf{W}_i \mathbf{X}_i)^{-1} \mathbf{X}_i \mathbf{W}_i \mathbf{Y}. \quad (12)$$

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 Fuzzy Inference Systems

Many optimization problems come with multiple objectives, which not only interact but may be in conflict. MSSA [20] is one of multiobjective optimization algorithms and it has been successfully used in the design of fuzzy models. In this study, we develop a multiobjective optimization algorithm based on an opposition-based space search algorithm as the optimization vehicle of FIS.

3.1. Multiobjective Opposition-Based Space Search Algorithm (MOSSA). We first introduce a single-objective space search algorithm (SSA), an adaptive heuristic optimization algorithm whose search method comes with the analysis of the solution space [21]. Let us recall the space search mechanism to update the current solutions. The role of space search is to generate new solutions from the old ones. The search method is based on the operator of space search, which generates two basic steps: it generates a new subspace (local area) and realizes search in this new space. The search in the new space is realized by randomly generating a new solution (individual) located in this 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 illustrate the operator in detail, we consider the optimization problem:

$$\begin{aligned} & \min (\text{or max}) \quad f(x_1, x_2, \dots, x_n) \\ \text{st:} \quad & x_i \in [l_i, u_i], \quad i = 1, 2, 3, \dots, n. \end{aligned} \quad (13)$$

Here a feasible solution can be represented in the following way (x_1, x_2, \dots, x_n) . Let $X^k = (x_1^k, x_2^k, \dots, x_n^k)$ be the k th solution, $L_i = \min_{j=1}^M x_i^j$, $U_i = \max_{j=1}^M x_i^j$.

We consider two scenarios.

(a) Space search based on M selected solutions: in this case, M solutions are randomly selected from the current population. The role of this operator is to update the current solutions by new solutions approaching

to the optimum. The adjacent space based on M solutions is given in the form

$$\begin{aligned} V_1 = \{ & (x_1, x_2, \dots, x_n) \mid x_i \\ & \in [l_i, u_i] \cap [L_i - \lambda_i(U_i - L_i), \\ & U_i + \lambda_i(U_i - L_i)], \\ & \lambda_i > 0, \quad i = 1, 2, \dots, n\}. \end{aligned} \quad (14)$$

(b) Space search based on the current best solution: in this case, the given solution is the best solution in the current population. The role of this operator is to adjust the best solution by searching the adjacent space as follows:

$$V_2 = \{(x_1, x_2, \dots, x_i, \dots, x_n) \mid x_i \in [Fl_i, Fu_i], F \in [0, 1]\}. \quad (15)$$

Here F is a proportion coefficient being used to adjust the size of adjacent space. In this study, F is set to 1.

To speed up the convergence of the SSA, we develop an opposition-based space search algorithm (OSSA) realized by using a mechanism of opposition-based learning (OBL) [22]. OBL has been shown to be an effective concept to enhance various optimization approaches. Let us recall the basic concept.

Opposition-Based Point [22]. Let $P = (x_1, x_2, \dots, x_D)$ be a point in a D -dimensional space, where $x_1, x_2, \dots, x_D \in R$ and $x_i \in [a_i, b_i]$, for all $i \in \{1, 2, \dots, D\}$. The opposite point $\overset{U}{P} = (\overset{U}{x}_1, \overset{U}{x}_2, \dots, \overset{U}{x}_D)$ is completely defined by its components as follows:

$$\check{x}_i = a_i + b_i - x_i. \quad (16)$$

Opposition-Based Optimization (OBL) [22]. Let $P = (x_1, x_2, \dots, x_D)$ be a point in a D -dimensional space (i.e., a candidate solution). Assume $f(\bullet)$ is a fitness function. According to the definition of the opposite point, we say that $\overset{U}{P} = (\overset{U}{x}_1, \overset{U}{x}_2, \dots, \overset{U}{x}_D)$ is the opposite of site $P = (x_1, x_2, \dots, x_D)$. Now, if $f(\overset{U}{P}) \geq f(P)$, then the point P can be replaced with $\overset{U}{P}$. Hence, the point and its opposite point are evaluated simultaneously in order to continue with the one of the highest fitness.

Based on this concept, we can develop the opposition-based space search operator. Assume that the current solution set (population) is P , where h is the size of solution set, P_o is the size of opposition solution set, and q is the dimension of a solution. The Pseudocode opposition-based space search can be summarized as shown in Pseudocode 1.

With understanding of the OSSA, we can develop an MOSSA. In order to improve convergence to the Pareto front as well as produce a well-distributed Pareto front, the technique of nondominated sort with the aid of the crowding distance [23] is used in the MOSSA. The details are presented in Pseudocode 2. The nondominated sort is realized with the aid of estimation of the crowding distance among solutions

```

(1) Begin
(2) Update the dynamic interval boundaries  $[a_j, b_j]$  in  $P$ .
    /*Generate a new solution set OP by using opposition*/
(3) For ( $i = 1; i \leq P_0; i++$ )
(4)     For ( $j = 1; j \leq q; j++$ )
(5)          $OP_{i,j} = a_j + b_j - P_{i,j}$ ;
(6)         IF  $OP_{i,j}$  out of the box-constraint
(7)              $x_j^{new} = \text{rand}(a_j, b_j)$ ;
(8)         End IF
(9)     End For
(10)    Calculate the fitness value of the  $OP_{i,j}$ 
(11) End For
(12) /* End opposition */
(13) End

```

PSEUDOCODE 1: Pseudocode of opposition-based space search.

```

(1) Initialize a solution set (population);
(2) Evaluate each solution in the solution set  $P$ ;
(3) While (not satisfy terminate condition)
(4)     Space search (case I);
(5)     Sort all solutions in  $S$  based on non-domination;
(6)     Remove the worst solution from the current solution set  $P$ ;
(7)     Space search (case II);
(8)     Sort all solutions in  $S$  based on non-domination;
(9)     Remove the worst solution from the current solution set  $P$ ;
(10)    Complete Opposition-Based Operation, and generate solution set  $OP$ ;
(11)    Sort all solutions in  $\{P, OP\}$  based on non-domination
(12)    Select  $h$  fittest solutions from  $\{P, OP\}$ ;
(13) End While
(14) Output the optimal solution.

```

PSEUDOCODE 2: Pseudocode of multiobjective opposition-based space search algorithm (MOSSA).

TABLE 1: List of parameters of the MOSSA.

MOSSA parameters	Values in this study
Solution set (Populations)	200
Number of solutions for search space (case I)	8
Number of generations for structural optimization	$m = 150$
Number of generations of parametric optimization in each generation of structural optimization	$k = 5$
Number of generations for parametric optimization	$n = 1000$

in the current solution set. The termination condition of the MOSSA is such that all the solutions in the current population have the same fitness or terminate after a certain fixed number of generations.

3.2. Arrangement of Solutions in the MOSSA. The standard gradient-based optimization techniques might not be effective in the context of rule-based systems given their non-linear character (in particular the form of the membership functions) and modularity of the systems. This suggests us

to explore other optimization techniques. Figure 2 depicts the arrangement of solutions present in the MOSSA-based method. The first part supporting structural optimization is separated from the second part used for parametric optimization. The size of the solutions for structural optimization of the fuzzy model is determined according to the number of all input variables of the system. The size of the solutions for parametric optimization depends on structurally optimized fuzzy inference system. When running the optimization method, we use an improved tuning method.

Figure 3 illustrates the comparison of “conventional” tuning and the proposed tuning. In the “conventional” tuning, the structural and the parametric optimization are carried out sequentially. First, the structural optimization is completed with m number of generations and then we proceed with the parametric phase with n number of generations, where m and n are given numbers. The structural optimization of the fuzzy model is carried out assuming that the apexes of the membership functions are kept fixed. The fixed apexes of the membership functions are taken as the center values produced by the C -Means algorithm, while the parametric optimization is applied to the fuzzy model derived through the structural optimization. In a nutshell, from the viewpoint

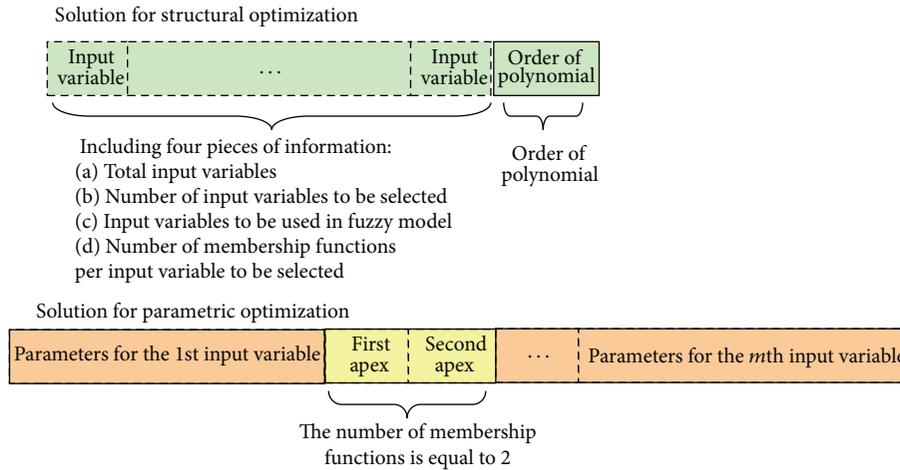


FIGURE 2: Arrangement of solutions using the MOSSA.

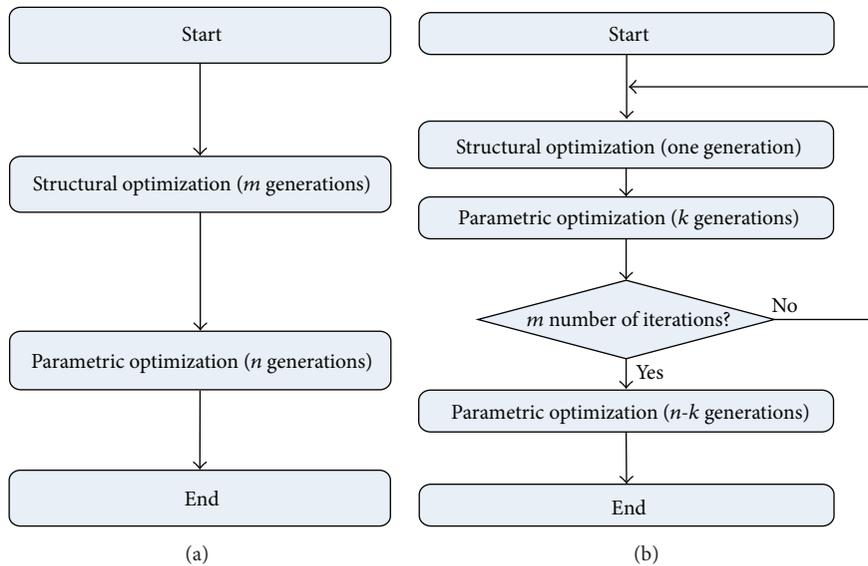


FIGURE 3: Comparison of “conventional” tuning and the proposed tuning: (a) overall flowchart of “conventional” tuning; and (b) overall flowchart of the proposed tuning.

of structure identification, only one fixed parameter set, which is the assigned apexes of membership functions obtained by *C*-Means clustering, is considered to carry out the overall structural optimization of fuzzy model. From the viewpoint of parameter identification, only one structurally optimized model that is obtained during the structure identification is considered to be involved in the overall parametric optimization. In order to construct the optimized fuzzy model, the range of search space for the structural as well as the parametric optimization is clearly restricted in the sequential tuning method. To alleviate this problem, we present a MOSSA-based improved tuning method. In this approach, we realize the structural with the aid of *k* number of generations of parametric optimization, where *k* is a given number. The several generations of parametric optimization will help to determine the optimal structure of the fuzzy model.

3.3. Objective Functions of the FIS. Three objective functions are used to evaluate the accuracy and the complexity of an FIS. Those are performance indexes, entropy of partition, and the total number of the coefficients of the polynomials to be estimated, respectively. Once the input variables of the premise part have been specified, the optimal consequence parameters that minimize the assumed performance index can be determined.

We consider two performance indexes, that is, the standard root mean squared error (RMSE) and mean squared error (MSE) as follows:

$$PI \text{ (or E_PI)} = \begin{cases} \sqrt{\frac{1}{m} \sum_{i=1}^m (y_i - y_i^*)^2}, & \text{(RMSE),} \\ \frac{1}{m} \sum_{i=1}^m (y_i - y_i^*)^2, & \text{(MSE),} \end{cases} \quad (17)$$

TABLE 2: Optimal solutions based on training data and testing data (NO_x).

(a) Based on the LSM								
Number of solutions	Selected input variables	Number of MFs	Type	PI	E.PI	Objective values		
						E	H	N
1	1,4, 5	2, 2, 2	3	0.042	0.104	0.073	8	80
2	1, 4, 5	2, 2, 2	4	0.087	0.197	0.142	8	56
3	1, 4, 5	2, 2, 2	2	0.371	0.412	0.392	8	32
4	1, 4	2, 2	3	1.016	0.810	0.913	4	24
5	1, 2, 4, 5	2, 2, 2, 2	1	2.823	6.997	4.910	16	16
6	4, 5	2, 2	4	5.525	8.527	7.026	4	16
7	4, 5	2, 2	2	6.029	10.41	8.217	4	12
8	2, 4, 5	2, 2, 2	1	6.290	13.74	10.02	8	8
9	4, 5	2, 2	1	13.60	20.23	16.92	4	4

(b) Based on the WLSM								
Number of solutions	Selected input variables	Number of MFs	Type	PI	E.PI	Objective values		
						E	H	N
1	1, 2, 3, 4, 5	2, 2, 2, 2, 2	4	0.00085	0.00042	0.00064	32	512
2	1, 2, 4, 5	2, 2, 2, 2	4	0.022	0.036	0.029	16	176
3	1, 4, 5	2, 2, 2	4	0.096	0.14	0.12	8	56
4	1, 3, 5	2, 2, 2	2	3.11	2.07	2.59	8	32
5	4, 5	2, 2	4	5.69	6.80	6.24	4	16
6	4, 5	2, 2	2	7.55	9.20	8.38	4	12
7	2, 4, 5	2, 2, 2	1	39.45	32.55	36.01	8	8
8	2, 5	2, 2	1	57.86	58.86	58.36	4	4

where y^* is the output of the fuzzy model, m is the total number of data, and i is the data index.

The accuracy criterion E includes both the training data and testing data and comes as a convex combination of the two components as follows:

$$E = \theta \times PI + (1 - \theta) \times E.PI. \tag{18}$$

Here, PI and E.PI (V.PI) denote the performance index for the training data and testing data (Validation data), respectively. θ is a weighting factor that allows us to strike a sound balance between the performance of the model for the training and testing data. Depending upon the values of the weighting factor, several specific cases of the objective function are worth distinguishing.

- (i) If $\theta = 1$, then the model is optimized based on the training data. No testing data is taken into consideration.
- (ii) If $\theta = 0.5$, then both the training and testing data are taken into account. Moreover it is assumed that they exhibit the same impact on the performance of the model.
- (iii) The case $\theta = \alpha$ where $\alpha \in [0, 1]$ embraces both the cases stated above. The choice of α establishes a certain tradeoff between the approximation and generalization aspects of the fuzzy model.

As a measure for evaluating the structure complexity of a model we consider the following partition criterion:

$$H = \prod_{i=1}^n F_i, \tag{19}$$

where n is the total number of selected input variables, and F_i is the number of membership functions for the i th corresponding input variable.

As a simplicity criterion we consider the consequence part of the local models, which is computed as

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

$$C_i = \begin{cases} 1 & \text{if type of local model is constant} \\ 1 + l & \text{if type of local model is linear form} \\ 1 + l + \frac{(l^2 - l)}{2} + l & \text{if type of local model is quadratic form} \\ 1 + l + \frac{(l^2 - l)}{2} & \text{if type of local model is modified quadratic form,} \end{cases} \tag{20}$$

TABLE 3: Optimal solutions based on training data, validation data, and testing data (NO_x).

(a) Based on LSM									
Number of solutions	Selected input variables	Number of MFs	Type	PI	V_PI	E_PI	Objective values		
							E	H	N
1	1, 4, 5	2, 2, 2	3	0.014	0.044	7.046	0.029	8	80
2	1, 4, 5	2, 2, 2	4	0.053	0.062	146.31	0.057	8	56
3	1, 4, 5	2, 2, 2	2	0.091	0.118	117.79	0.104	8	32
4	1, 4	2, 2	3	1.149	0.447	3.503	0.798	4	24
5	4, 5	2, 2	4	5.269	5.409	16.40	5.339	4	16
6	4, 5	2, 2	2	5.985	5.789	21.94	5.887	4	12
7	1, 4, 5	2, 2, 2	1	9.097	8.655	15.80	8.876	8	8
8	4, 5	2, 2	1	14.10	18.95	20.64	16.52	4	4

(b) Based on WLSM									
Number of solutions	Selected input variables	Number of MFs	Type	PI	V_PI	E_PI	Objective values		
							E	H	N
1	1, 2, 3, 4, 5	2, 2, 2, 2, 2	4	0.00068	0.00062	0.0000074	0.00065	32	512
2	1, 3, 4, 5	2, 2, 2, 2	4	0.0086	0.0068	0.0804	0.00765	16	176
3	1, 4, 5	2, 2, 2, 2, 2	4	0.097	0.122	0.135	0.109	8	56
4	1, 3, 5	2, 2, 2, 2	2	2.55	0.596	2.949	1.572	8	32
5	1, 4	2, 2	4	5.88	1.456	3.419	3.667	4	16
6	4, 5	2, 2	2	7.51	5.255	12.39	6.383	4	12
7	2, 4, 5	2, 2, 2	1	39.53	38.74	19.03	3.914	8	8
8	4, 5	2, 2	1	52.44	40.92	39.90	4.668	4	4

TABLE 4: Comparative analysis of selected models (NO_x).

Model		PI (MSE)	V_PI (MSE)	E_PI (MSE)	Number of rules	
Regression model		17.68		19.23		
Hybrid FS-FNNs [20]		2.806		5.164	30	
Hybrid FR-FNNs [24]		0.080		0.190	32	
Multi-FNN [25]		0.720		2.205	30	
Hybrid rule-based FNNs [26]		3.725		5.291	30	
Choi's model [27]		0.012		0.067	18	
Our model	LSM	Two split dataset	0.042	0.104	8	
		Three split dataset	0.014	0.044	7.046	8
	WLSM	Two split dataset	0.022		0.036	16
		Three split dataset	0.009	0.007	0.080	16

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 MOSSA. This leads to easily interpretable, simple, and accurate fuzzy models.

4. Experimental Studies

This section reports on comprehensive numeric studies illustrating the design of the fuzzy model and quantifying its performance. We use three well-known data sets. Each data set is divided into two parts of the same size. PI denotes the performance index for the training data, V_PI represents the

validation data, and E_PI stands for the testing data. In all considerations, the weighting factor θ was set to 0.5.

The parameters of the MOSSA are summarized in Table 1 (the choice of these specific values of the parameters is a result of intensive experimentations; as a matter of fact, those values are in line with those reported in the literature).

4.1. NO_x Emission Process Data of Gas Turbine Power Plant.

NO_x emission process is modeled using the data of gas turbine power plants. A NO_x emission process of a GE gas turbine power plant located in Virginia, USA, is chosen in this experiment. The input variables include AT (ambient temperature a site), CS (compressor speed), LPTS (low pressure turbine speed), CDP (compressor discharge pressure), and

TABLE 5: Optimal solutions based on training data and testing data (MPG).

(a) Based on the LSM

Number of solutions	Selected input variables	Number of MFs	Type	PI	E_PI	Objective values		
						E	H	N
1	4, 5, 6	2, 2, 2	2	2.535	2.372	2.454	8	32
2	4, 6	2, 2	2	2.792	2.718	2.755	4	12
3	4, 5, 6	2, 2, 2	1	2.893	2.585	2.739	8	8
4	4, 6	2, 2	1	2.982	2.728	2.855	4	4

(b) Based on the WLSM

Number of solutions	Selected input variables	Number of MFs	Type	PI	E_PI	Objective values		
						E	H	N
1	1, 2, 3, 5, 6, 7	2, 2, 2, 2, 2	4	0.646	0.245	0.445	64	1408
2	3, 4, 5, 6, 7	2, 2, 2, 2, 2	4	1.351	0.683	1.017	32	512
3	2, 5, 6, 7	2, 2, 2, 2	4	1.551	0.940	1.246	16	176
4	1, 4, 5, 6	2, 2, 2, 2	2	2.266	1.689	1.977	16	80
5	2, 5, 6	2, 2, 2	4	2.396	1.658	2.027	8	56
6	4, 5, 6	2, 2, 2	2	2.636	2.132	2.384	8	32
7	4, 6	2, 2	4	2.849	2.399	2.624	4	16
8	4, 6	2, 2	2	2.889	2.457	2.673	4	12
9	2, 3, 6	2, 2, 2	1	3.228	2.668	2.948	8	8
10	4, 6	2, 2	1	3.355	2.982	3.168	4	4

TABLE 6: Optimal solutions based on training data, validation data, and testing data (MPG).

(a) Based on the LSM

Number of solutions	Selected input variables	Number of MFs	Type	PI	V_PI	E_PI	Objective values		
							E	H	N
1	2, 3, 6	2, 2, 2	4	2.284	2.632	3.174	2.458	8	56
2	3, 4, 6	2, 2, 2	2	2.450	2.757	2.536	2.603	8	32
3	3, 4, 5, 6	2, 2, 2, 2	1	2.679	2.704	2.292	2.692	16	16
4	4, 6	2, 2	2	2.781	2.819	2.592	2.800	4	12
5	3, 4, 6	2, 2, 2	1	2.772	2.895	2.290	2.834	8	8
6	4, 6	2, 2	1	3.037	2.792	2.773	2.914	4	4

(b) Based on the WLSM

Number of solutions	Selected input variables	Number of MFs	Type	PI	V_PI	E_PI	Objective values		
							E	H	N
1	2, 3, 4, 5, 6, 7	2, 2, 2, 2, 2, 2	4	0.451	0.083	0.096	0.267	64	1408
2	2, 3, 5, 6, 7	2, 2, 2, 2, 2	4	0.637	0.088	0.199	0.362	32	512
3	1, 2, 4, 5, 6, 7	2, 2, 2, 2, 2, 2	2	1.589	1.044	0.897	1.316	64	448
4	3, 5, 6, 7	2, 2, 2, 2	4	1.802	0.842	0.978	1.322	16	176
5	1, 4, 5, 6	2, 2, 2, 2	2	2.206	1.658	1.719	1.932	16	80
6	2, 5, 6	2, 2, 2	4	2.368	1.618	1.282	1.993	8	56
7	3, 4, 6	2, 2, 2	2	2.548	2.284	1.708	2.416	8	32
8	4, 6	2, 2, 2	4	2.876	2.532	2.026	2.704	4	16
9	4, 6	2, 2	2	2.913	2.634	2.350	2.773	4	12
10	3, 4, 6	2, 2, 2	1	2.974	2.887	2.763	2.930	8	8
11	4, 6	2, 2	1	3.423	3.036	3.174	3.230	4	4

TET (turbine exhaust temperature). The output variable is NO_x . The performance index is MSE defined by (17).

First, the NO_x emission process is split into two separate data sets. The first 50% of data set (consisting of 130 pairs)

is used in the design of the fuzzy model. The remaining 50% data set (consisting of 130 pairs) helps quantify the predictive quality of the model. Table 2 summarizes the performance values of optimal solutions (individuals) objective functions

TABLE 7: Comparative analysis of the selected models (MPG).

Model		PI (RMSE)	V_PI (RMSE)	E_PI (RMSE)	Number of rules
RBFNN [28]		3.24		3.62	36
Linguistic model [10]		2.86		3.24	36
Functional RBFNN [28]		2.41		2.82	33
LSM	Two split dataset	2.54		2.37	8
	Three split dataset	2.45	2.76	2.54	8
Our model	Two split dataset	2.40		1.66	8
	WLSM	1.36		0.68	32
	Three split dataset	2.37	1.62	1.28	8
		0.64	0.09	0.20	32

TABLE 8: Optimal solutions based on training data and testing data (Housing).

(a) Based on the LSM

Number of solutions	Selected input variables	Number of MFs	Type	PI	E_PI	Objective values		
						E	H	N
1	6, 9, 10, 11, 13	2, 2, 2, 2, 2	2	3.148	3.306	3.227	32	192
2	6, 7, 10, 11, 13	2, 2, 2, 2, 2	1	3.459	3.637	3.548	32	32
3	1, 5, 6	2, 2, 2	4	4.275	4.171	4.223	8	56
4	3, 6, 13	2, 2, 2	1	4.114	3.907	4.008	8	8
5	2, 3	2, 2	3	7.087	7.593	7.340	4	24
6	3, 11	2, 2	1	7.300	7.496	7.398	4	4

(b) Based on the WLSM

Number of solutions	Selected input variables	Number of MFs	Type	PI	E_PI	Objective values		
						E	H	N
1	3, 6, 7, 8, 11, 13	2, 2, 2, 2, 2, 2	4	1.208	0.077	0.643	64	1408
2	1, 5, 6, 8, 13	2, 2, 2, 2, 2	4	1.319	0.726	1.022	32	512
3	6, 7, 10, 11, 12, 13	2, 2, 2, 2, 2, 2	2	2.134	1.410	1.772	64	448
4	1, 6, 9, 11, 13	2, 2, 2, 2, 2	2	2.776	2.105	2.440	32	192
5	1, 6, 13	2, 2, 2	4	2.872	2.272	2.572	8	56
6	6, 7, 10, 12, 13	2, 2, 2, 2, 2	1	3.939	3.687	3.813	32	32
7	2, 6, 10	2, 2, 2	2	5.533	3.893	4.713	8	32
8	6, 11, 13	2, 2, 2	1	4.776	3.970	4.373	8	8
9	3, 13	2, 2	1	5.824	5.961	5.892	4	4

(E, H, N) of the FIS. Generally, as could have been expected, by increasing the number of coefficients or rules, the accuracy of FIS becomes better.

Second, the NO_x emission process data is divided into three parts. 130 pairs out of 260 pairs of input-output data are used for training; 78 pairs out of 260 pairs of input-output data are utilized for validation; the remaining part (consisting of 52 pairs) serves as a testing set. The results of the proposed model are shown in Table 3.

Table 4 illustrates the results of comparative analysis of the proposed model when being contrasted with other models. The selected values of the performance indexes of the FIS are included in Tables 2 and 3, respectively. We note that the proposed model outperforms several previous fuzzy models known in the literature.

4.2. Automobile Miles Per Gallon (MPG) Data. We consider automobile MPG data (<http://archive.ics.uci.edu/ml/datasets/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 (17).

The automobile MPG data is partitioned into two separate parts. The first 235 data pairs are used as the training data set for FIS while the remaining 157 pairs are the testing data set for assessing the predictive performance. Table 5 summarizes the performance values of solution (individual) objective functions (E, H, N) of the FIS.

TABLE 9: Optimal solutions based on training data, validation data, and testing data (Housing).

(a) Based on the LSM									
Number of solutions	Selected input variables	Number of MFs	Type	PI	V_PI	E_PI	Objective values		
							E	H	N
1	1, 6, 11, 13	2, 2, 2, 2	1	3.529	2.888	5.093	3.209	16	16
2	1, 6, 13	2, 2, 2	1	3.680	3.312	5.713	3.496	8	8
3	6, 13	2, 2	1	4.662	3.299	4.859	3.980	4	4

(b) Based on the WLSM									
Number of solutions	Selected input variables	Number of MFs	Type	PI	V_PI	E_PI	Objective values		
							E	H	N
1	3, 6, 8, 11, 12, 13	2, 2, 2, 2, 2, 2	4	0.850	0.031	0.163	0.441	64	1408
2	5, 6, 8, 11, 13	2, 2, 2, 2, 2	4	1.087	0.448	0.510	0.768	32	512
3	6, 8, 11, 13	2, 2, 2, 2	4	1.652	1.210	1.738	1.431	16	176
4	1, 6, 13	2, 2, 2	4	2.841	2.321	2.246	2.581	8	56
5	3, 6, 7, 9, 13	2, 2, 2, 2, 2	1	3.965	3.300	4.678	3.633	32	32
6	4, 6, 13	2, 2, 2	1	5.042	3.487	4.623	4.264	8	8
7	10, 13	2, 2	1	5.720	5.347	7.020	5.533	4	4

TABLE 10: Comparative analysis of selected models (Boston Housing).

Model		PI (RMSE)	V_PI (RMSE)	E_PI (RMSE)	Number of rules	
FIS [27]		3.13		3.17	12	
RBFNN [28]		6.63		7.14	20	
SVR [29]		1.17		5.84		
MARS [30]		0.97		5.96		
NN		4.19		4.26	12	
		3.27		5.14	24	
FNN [31]		3.76		4.08	21	
FPNN		3.51		16.93	16	
Our model	LSM	Two split dataset		3.64	32	
		Three split dataset	2.89	5.09	16	
	WLSM	Two split dataset	2.87		2.27	8
		Three split dataset	1.65	1.21	1.73	16

Next, we divide the automobile MPG data into three separate parts. The first one (consisting of 196 pairs) is used for training. The second (consisting of 118 pairs) part of the series is utilized for validation. The remaining part (consisting of 78 pairs) serves as a testing set. The values of the performance index are presented in Table 6.

The identification error of the proposed model is compared with the performance of some other model; refer to Table 7. The selected values of the performance indexes of the FIS are marked in Tables 5 and 6, respectively. The performance of the proposed model is better in the sense of its approximation and prediction abilities.

4.3. *Boston Housing Data.* Here we experiment with the Boston housing data set [29]. 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, and their median price. The

dataset consists of 506 14-dimensional data. The performance index is defined as the RMSE as given by (17).

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 (i.e., 253 data points) is retained for testing purposes. The values of the performance index are summarized in Table 8.

Next, we move on to the Boston Housing data set, which is partitioned into three separate parts. 253 pairs out of 506 pairs of input-output data are used for training; 152 pairs out of 506 pairs of input-output data are utilized for validation; the remaining part (consisting of 101 pairs) serves as a testing set. Table 9 summarizes the results obtained for the optimized structure and performance index for optimized parameters by the MOSSA.

Table 10 supports a comparative analysis considering some existing models. The selected values of the performance

indexes of the FIS are marked in Tables 8 and 9, respectively. It is evident that the proposed model compares favorably both in terms of accuracy and prediction capabilities.

5. Concluding Remarks

This paper contributes to the research area of the hybrid optimization of fuzzy inference systems in the following two important aspects: (1) we proposed a multiobjective opposition-based space search algorithm; (2) we introduced the identification of fuzzy inference systems based on the MOSSA and WLSM. Numerical experiments using three well-known datasets show that the model constructed with the aid of the MOSSA exhibits better performance in comparison with the fuzzy model reported in the literature.

Acknowledgments

This work was supported by the Open Foundation of State Key Laboratory of Virtual Reality and Technology and Systems, Beihang University, China (Grant no. BUAA-VR-12KF-13), by Educational Commission of Tianjin City, China (Grant no. 20120703), and also supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2012-003568). This work was partly supported by the GRRC program of Gyeonggi province (GRRC Suwon 2012-B2, Center for U-city Security & Surveillance Technology).

References

- [1] W. Pedrycz, "An identification algorithm in fuzzy relational systems," *Fuzzy Sets and Systems*, vol. 13, no. 2, pp. 153–167, 1984.
- [2] R. M. Tong, "The evaluation of fuzzy models derived from experimental data," *Fuzzy Sets and Systems*, vol. 4, no. 1, pp. 1–12, 1980.
- [3] C. W. Xu and Y. Zailu, "Fuzzy model identification self-learning for dynamic system," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 17, no. 4, pp. 683–689, 1987.
- [4] M. Sugeno and T. Yasukawa, "Linguistic modeling based on numerical data," in *Proceedings of the International Fuzzy Systems Association Congress (IFSA '91)*, Computer, Management & System Science, pp. 264–267, Brussels, Belgium, 1991.
- [5] S. Oh and W. Pedrycz, "Identification of fuzzy systems by means of an auto-tuning algorithm and its application to nonlinear systems," *Fuzzy Sets and Systems*, vol. 115, no. 2, pp. 205–230, 2000.
- [6] Y. Chung, W. Pedrycz, and E. T. Kim, "A new two-phase approach to fuzzy modeling for nonlinear function approximation," *IEICE Transactions on Information and Systems*, vol. E89-D, no. 9, pp. 2473–2483, 2006.
- [7] R. Alcalá, J. Alcalá-Fdez, F. Herrera, and J. Otero, "Genetic learning of accurate and compact fuzzy rule based systems based on the 2-tuples linguistic representation," *International Journal of Approximate Reasoning*, vol. 44, no. 1, pp. 45–64, 2007.
- [8] W. Huang and 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.
- [9] A. Fernández, M. J. del Jesus, and F. Herrera, "Hierarchical fuzzy rule based classification systems with genetic rule selection for imbalanced data-sets," *International Journal of Approximate Reasoning*, vol. 50, no. 3, pp. 561–577, 2009.
- [10] M. P. Ana, S. Luciano, and C. Ines, "Diagnosis of dyslexia with low quality data with genetic fuzzy systems," *International Journal of Approximate Reasoning*, vol. 51, no. 8, pp. 993–1009, 2010.
- [11] Y. Jin, "Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 2, pp. 212–221, 2000.
- [12] M. Setnes and H. Roubos, "GA-based modeling and classification: complexity and performance," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 5, pp. 509–522, 2000.
- [13] C. Coello and G. Pulido, "Multiobjective optimization using a micro-genetic algorithm," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '01)*, pp. 274–282, 2001.
- [14] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [15] G. Avigad and A. Moshaiov, "Interactive evolutionary multi-objective search and optimization of set-based concepts," *IEEE Transactions on Systems, Man and Cybernetics B*, vol. 38, no. 2, pp. 381–403, 2008.
- [16] G. G. Yen and W. F. Leong, "Dynamic multiple swarms in multiobjective particle swarm optimization," *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 39, no. 4, pp. 890–911, 2009.
- [17] L. J. Herrera, H. Pomares, I. Rojas, O. Valenzuela, and A. Prieto, "TaSe, a Taylor series-based fuzzy system model that combines interpretability and accuracy," *Fuzzy Sets and Systems*, vol. 153, no. 3, pp. 403–427, 2005.
- [18] M. Delgado, M. P. Ceullar, and M. C. Pegalajar, "Multiobjective hybrid optimization and training of recurrent neural networks," *IEEE Transactions on Systems, Man and Cybernetics B*, vol. 38, no. 2, pp. 381–403, 2008.
- [19] W. Huang, L. Ding, S. K. Oh, L. Ding, H. K. Kim, and 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.
- [20] S.-K. Oh, W. Pedrycz, and H.-S. Park, "Hybrid identification in fuzzy-neural networks," *Fuzzy Sets and Systems*, vol. 138, no. 2, pp. 399–426, 2003.
- [21] W. Huang, L. Ding, S. K. Oh, C. W. Jeong, and 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.
- [22] R. S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 64–79, 2008.
- [23] I. Karahan and M. Koksalan, "A territory defining multi-objective evolutionary algorithms and preference incorporation," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 4, pp. 636–664, 2012.
- [24] H. S. Park and S. K. Oh, "Fuzzy relation-based fuzzy neural networks using a hybrid identification algorithm," *International Journal of Control, Automation and Systems*, vol. 1, no. 2, pp. 289–300, 2003.

- [25] H. S. Park and S. K. Oh, "Multi-FNN identification based on HCM clustering and evolutionary fuzzy granulation," *International Journal of Control, Automation and Systems*, vol. 1, no. 2, pp. 194–202, 2003.
- [26] S. K. Oh, W. Pedrycz, and H. S. Prak, "Implicit rule-based fuzzy-neural networks using the identification algorithm of hybrid scheme based on information granulation," *Advanced Engineering Informatics*, vol. 16, no. 4, pp. 247–263, 2002.
- [27] J. N. Choi, S. K. Oh, and W. Pedrycz, "Identification of fuzzy relation models using hierarchical fair competition-based parallel genetic algorithms and information granulation," *Applied Mathematical Modelling*, vol. 33, no. 6, pp. 2791–2807, 2009.
- [28] W. Pedrycz, H. S. Park, and S. K. Oh, "A granular-oriented development of functional radial basis function neural networks," *Neurocomputing*, vol. 72, no. 1–3, pp. 420–435, 2008.
- [29] H. D. Chris, J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Advances in Neural Information Processing Systems*, vol. 9, pp. 155–161, 1997.
- [30] J. H. Friedman, "Multivariate adaptive regression splines," *The Annals of Statistics*, vol. 19, no. 1, pp. 1–67, 1991.
- [31] A. F. Gobi and W. Pedrycz, "Fuzzy modelling through logic optimization," *International Journal of Approximate Reasoning*, vol. 45, no. 3, pp. 488–510, 2007.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

