A Multiobjective Evolutionary Approach to Concurrently Learn Rule and Data Bases of Linguistic Fuzzy-Rule-Based Systems

Rafael Alcalá, Pietro Ducange, Francisco Herrera, Beatrice Lazzerini, *Member, IEEE*, and Francesco Marcelloni, *Member, IEEE*

***Abstract*—In this paper, we propose the use of a multiobjective evolutionary approach to generate a set of linguistic fuzzy-rule-based systems with different tradeoffs between accuracy and interpretability in regression problems. Accuracy and interpretability are measured in terms of approximation error and rule base (RB) complexity, respectively. The proposed approach is based on concurrently learning RBs and parameters of the membership functions of the associated linguistic labels. To manage the size of the search space, we have integrated the linguistic two-tuple representation model, which allows the symbolic translation of a label by only considering one parameter, with an efficient modification of the well-known (2 + 2) Pareto Archived Evolution Strategy (PAES). We tested our approach on nine real-world datasets of different sizes and with different numbers of variables. Besides the (2 + 2)PAES, we have also used the well-known nondominated sorting genetic algorithm (NSGA-II) and an accuracy-driven single-objective evolutionary algorithm (EA). We employed these optimization techniques both to concurrently learn rules and parameters and to learn only rules. We compared the different approaches by applying a nonparametric statistical test for pairwise comparisons, thus taking into consideration three representative points from the obtained Pareto fronts in the case of the multiobjective EAs. Finally, a data-complexity measure, which is typically used in pattern recognition to evaluate the data density in terms of average number of patterns per variable, has been introduced to characterize regression problems. Results confirm the effectiveness of our approach, particularly for (possibly high-dimensional) datasets with high values of the complexity metric.**

***Index Terms*—Accuracy–interpretability tradeoff, fuzzy rule-based systems (FRBSs), linguistic two-tuple representation, multiobjective evolutionary algorithms (EAs).**

I. Introduction

THE AUTOMATIC identification of fuzzy systems from data samples for carrying out specific tasks can be considered as an optimization or search process. Currently, evolutionary algorithms (EAs), particularly genetic algorithms (GAs) [1], [2], are considered to be among the most effective and widely used global search techniques. As a result, EAs have been extensively applied to identify fuzzy systems, mainly with the objective of increasing accuracy, thus leading to the so-called genetic fuzzy systems [3]–[5].

The problem of finding the right tradeoff between interpretability and accuracy, in spite of the original nature of fuzzy logic, has given rise to a growing interest in methods that take both aspects into account [6]. Ideally, both criteria should be satisfied to a high degree; since they are in conflict, however, this is not generally possible. As this problem is of a multiobjective nature, the use of multiobjective EAs (MOEAs) [7], [8] or multiobjective machine-learning techniques [9], [10] to obtain a set of solutions with different tradeoffs between both kinds of criteria represents a new and interesting approach.

Although there are some works in the literature that use both standard and specific MOEAs to improve the difficult tradeoff between interpretability and accuracy of Mamdani fuzzy-rule-based systems (FRBSs) [11]–[20], most of them are applied to classification problems. In these works, authors try to obtain sets of nondominated solutions corresponding to different tradeoffs by selecting (postprocessing) or learning the set of rules that best represents the example data. None of these approaches, however, consider the learning or tuning of membership function (MF) parameters, which involves a more complex search space. It is only recently that two similar MOEAs have been proposed to perform a tuning of the MFs while evolving the antecedents of a previously obtained rule base (RB) in classification problems [21], [22]. The authors first create a decision tree through the classical C.4.5 algorithm and transform this decision tree into a fuzzy-rule-based classifier (FRBC). Thus, relevant variables are selected and an initial partition of the input space is performed. Then, they randomly modify some parameters of this FRBC to generate the individuals of the initial population of the NSGA-II MOEA [23]. Thus, in practice, they start from an initial knowledge base (KB) and apply an MOEA to tune this KB rather than to learn it.

With regard to regression, only a few works can be found in the literature. An efficient, modified version of the well-known (2+2) Pareto Archived Evolution Strategy (PAES) [25] and appropriate genetic operators have been proposed in [24] to generate a set of Mamdani FRBSs with different tradeoffs between accuracy and complexity. Though this approach does not consider database (DB) learning, it allows the derivation of a large set of nondominated RBs. Recently, two approaches have also

Manuscript received July 29, 2008; revised December 8, 2008, February 23, 2009, and March 30, 2009; accepted April 1, 2009. First published May 15, 2009; current version published October 8, 2009. This work was supported by the Spanish Ministry of Education and Science under Grant TIN2005-08386-C05-01 and Grant TIN2008-06681-C06-01.

R. Alcalá and F. Herrera are with the Department of Computer Science and Artificial Intelligence, University of Granada, Granada 18071, Spain (e-mail: alcala@decsai.ugr.es; herrera@decsai.ugr.es).

P. Ducange, B. Lazzerini, and F. Marcelloni are with the Dipartimento di Ingegneria dell'Informazione: Elettronica, Informatica, Telecomunicazioni, University of Pisa, Pisa 56122, Italy (e-mail: p.ducange@iet.unipi.it; b.lazzerini@iet.unipi.it; f.marcelloni@iet.unipi.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TFUZZ.2009.2023113

taken MF parameter learning into account. The authors in [26] and [27] have applied classic tuning of the MF parameters (three points per MF), together with a rule selection. This approach focuses the search on the most accurate part of the Pareto front, thus obtaining a short Pareto with very fine solutions in terms of accuracy. An MOEA has been applied in [28] to adapt MF parameters to specific contexts so as to maximize accuracy and the value of a purposely defined interpretability index. These algorithms were usually tested on two or three datasets, which is the common practice in the regression community, since the curse of dimensionality is more difficult to address in continuous output domains, and further, few datasets are usually available.

In this paper, we propose a technique to perform both the RB identification and the DB learning in the same multiobjective framework for regression problems. In this way, the whole KB of a set of nondominated FRBSs is generated at the same time by taking the existing dependencies between the RB and the DB into account. Learning both parts concurrently represents a way to improve the accuracy of FRBSs; on the other hand, the search space becomes extremely complex and must be handled by state-of-the-art EAs.

In order to reduce the search space, the linguistic two-tuple representation [29], [30], together with the RB identification approach [24] based on an effective modification of the well-known (2+2)PAES [25], has been exploited. The two-tuple representation allows the lateral displacement of a label by only considering one parameter instead of the classic three-point representation used for triangular fuzzy sets. As shown in [29], this helps obtain more accurate solutions when large search spaces are considered. Further, the two-tuple representation preserves the original shapes of the MFs. Finally, since the lateral displacements are restricted to a short interval, the interpretability is maintained at a good level.

Unlike our approach, the previous works that focused on parametric adaptation of a DB have proposed techniques to postprocess a predefined KB. These techniques are very useful for the improvement of the accuracy of the KB but cannot generate it from scratch. Consequently, they could be considered as complementary to our approach. For example, those proposed in [26] and [27] could be applied as a second stage to the FRBSs generated by the algorithm proposed in this paper (from lateral displacements to classic three points in a reduced search space). Obviously, we would lose part of the interpretability provided by the linguistic two-tuple representation.

We tested our approach on nine real-world problems with a number of variables ranging from 2 to 26 and a number of samples ranging from 495 to 15 000. We used the well-known NSGA-II and a single-objective GA (SOGA) driven only by accuracy as comparisons. To assess the results obtained by the different algorithms, we have applied a nonparametric statistical test [31]–[34] for pairwise comparisons, considering, for the MOEAs, three representative points of the obtained Pareto fronts. Finally, we have proposed a categorization of the datasets based on a simple data-complexity measure imported from the pattern-recognition literature [35], [36]. This measure, which is denoted by T2, is defined as the average number of patterns per variable. Results confirm a positive synergy between the learning of rules and the two-tuple-based technique, particularly in (possibly high-dimensional) datasets characterized by high values of T2.

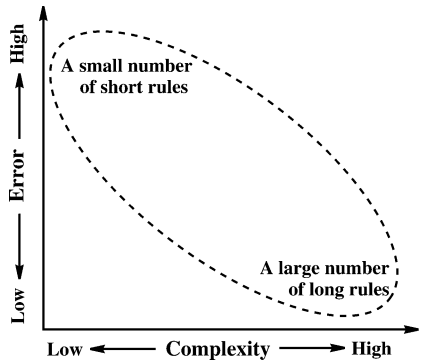

Fig. 1. Tradeoff between the error and the complexity of rule sets.

This paper is arranged as follows. The next section briefly analyzes the state of the art in the use of MOEAs to generate Mamdani FRBSs with different tradeoffs between accuracy and interpretability. Section III explains how we codify the linguistic RB and presents the linguistic two-tuple representation. Section IV describes the proposed MOEA for learning KBs. Section V shows and discusses the results obtained on nine real-world problems. Section VI gives some conclusions. Finally, Appendix I describes the Wilcoxon signed-rank test, and Appendix II shows the results obtained in two representative points of the Pareto fronts.

II. Use of MOEAs for Generating FRBSs With Different Accuracy–Interpretability Tradeoffs

MOEAs generate a family of equally valid solutions, where each solution tends to satisfy a specific criterion to a higher extent than another. For this reason, MOEAs have also been applied to generate a set of FRBSs with different tradeoffs between complexity and accuracy. Here, each solution in the Pareto front represents a different tradeoff (see Fig. 1).

The most prolific research activity in the application of MOEAs to Mamdani FRBS generation has been performed by Ishibuchi's group. Earlier works [13] were devoted to using simple first-generation MOEAs to perform a rule selection on an initial set of classification rules involving "*don't care*" conditions and considering two different objectives: classification accuracy and number of rules. Then, a third objective was included to minimize the length of the rules by rule selection [15] or rule learning [15]. The authors in [18] apply a more effective MOEA, i.e., the multiobjective genetic local search (MOGLS) [37], for rule selection with the same three objectives. Finally, two algorithms based on an MOEA of the second generation (NSGA-II) have been proposed for rule selection [20] and rule learning [19], considering the same concepts. We can also find related papers from other researchers. For instance, in [11], Cordon *et al.* use a classical MOEA (MOGA) to jointly perform feature selection and fuzzy set granularity learning with only two objectives.

It should be highlighted that all the mentioned methods were applied to classification problems for rule selection or rule learning, without learning or tuning the MF parameters that were

initially fixed. Moreover, the used MOEAs are slight modifications of MOEAs proposed for general use (MOGA, NSGA-II, etc.) or are based on them. Note that these MOEAs might need an adaptation to improve their search capability in order to obtain good results also for the accuracy objective. For example, one possibility could be the use of similarity measures as in [20] (favoring the crossover of similar solutions). Recently, two similar MOEAs (based on NSGA-II) were proposed to postprocess an initial FRBS in classification problems [21], [22]. Both algorithms perform a tuning of the MFs while evolving the premises of the initial RB.

On the other hand, there are few works in the framework of fuzzy modeling for regression problems. The authors in [17] show how a simple MOEA can be applied to a three-objective optimization problem to obtain Mamdani FRBSs. An adaptation of the efficient (2 + 2)PAES [25] has been applied in [24] for the identification of Mamdani FRBSs for regression problems by considering the minimization of two objectives (the system error and the number of variables involved in the antecedents of the rules). These approaches do not consider learning or tuning of MF parameters. A new method was recently proposed in [26] and deeply discussed in [27] to perform rule selection and MF parameter tuning of Mamdani FRBSs by forcing mating restrictions to concentrate the search around the most promising solutions, thus allowing exploration at the beginning and favoring the exploitation of the most accurate solutions at the end (SPEA2$_{\mathrm{Acc}}$). This approach was proposed as a postprocessing technique and can be applied to improve previously obtained KBs. Further, in [28], another MOEA has been adopted to perform context adaptation (adaptation of the MFs by using scaling functions) as a postprocessing algorithm applied to an initial KB. This algorithm considers the system error and an interpretability index as objectives.

Some applications of MOEAs have also been discussed in the literature to improve the difficult tradeoff between accuracy and interpretability of Takagi–Sugeno models [42]. Accuracy, interpretability, and compactness have been considered as objectives in [38]–[41] to obtain interpretable and very accurate Takagi–Sugeno models.

Table I summarizes the different methods, which, to the best of our knowledge, have been proposed in the literature. In this paper, we propose a new method to concurrently learn the RB and the DB (MF parameter learning) of Mamdani FRBSs. In Table I, we also summarize the main characteristics of this new approach.

TABLE I
USE OF MOEAS FOR GENERATING FRBSS WITH DIFFERENT ACCURACY–INTERPRETABILITY TRADEOFFS

Year	Ref.	Problem	MOEA/Gen.	#Objs.	RS	FS	RL	LP
MAMDANI LINGUISTIC MODELS								
1995/7/8	[12]–[14]	Class.	MOGA/1^{st}	2	$\surd$	$\surd$	-	-
2001	[11]	Class.	MOGA/1^{st}	2	-	$\surd$	$\surd$	-
2001	[15], [16]	Class.	MOGA/1^{st}	3	$\surd$	$\surd$	$\surd$	-
2004	[18]	Class.	MOGLS/1^{st}	3	$\surd$	$\surd$	-	-
2005	[20]	Class.	NSGA-II*/2^{nd}	3	$\surd$	$\surd$	-	-
2007	[19]	Class.	NSGA-II*/2^{nd}	3	-	$\surd$	$\surd$	-
2008	[21]	Class.	NSGA-II*/2^{nd}	3	-	$\surd$	$\surd$	$\surd$
2008	[22]	Class.	NSGA-II*/2^{nd}	3	-	$\surd$	$\surd$	$\surd$
2003	[17]	Regr.	MOGA/1^{st}	3	$\surd$	$\surd$	$\surd$	-
2007	[24]	Regr.	PAES*/2^{nd}	2	-	$\surd$	$\surd$	-
2007/2009	[26], [27]	Regr.	SPEA2*/2^{nd}	2	$\surd$	-	-	$\surd$
2009	[28]	Regr.	NSGA-II*/2^{nd}	2	-	-	-	$\surd$
-	Proposed here	Regr.	PAES*/2^{nd}	2	-	$\surd$	$\surd$	$\surd$
TAKAGI-SUGENO MODELS								
2001	[38], [39]	Regr.	Specific/1^{st}	3	-	-	$\surd$	$\surd$
2005	[40]	Regr.	MOGA*/1^{st}	5	$\surd$	$\surd$	$\surd$	$\surd$
2005	[41]	Regr.	NSGA-II*/2^{nd}	5	$\surd$	$\surd$	$\surd$	$\surd$

RS: rule selection; FS: feature selection; RL: rule learning; LP: learning/tuning of parameters; class./regr.: classification/regression; $1^{st}/2^{nd}$: first/second generation of MOEAs; * based on that algorithm.

III. PRELIMINARIES: RB AND DB REPRESENTATION

This section discusses how we represent the RB and introduces the linguistic two-tuple representation used to learn MF parameters.

A. Linguistic Fuzzy RB Representation

Let $X = \{X_1, \ldots, X_F\}$ be the set of input variables and $X_{(F+1)}$ be the output variable. Let U_f be the universe of the fth variable. Let $P_f = \{A_{f,1}, \ldots, A_{f,T_f}\}$ be a uniform fuzzy partition with T_f linguistic terms (labels) on the fth variable.

The mth rule ($m = 1, \ldots, M$) of a Mamdani fuzzy system can be expressed as

$$R^m: \text{ If } X_1 \text{ is } A_{1,j_1^m} \text{ and} \cdots \text{and } X_F \text{ is } A_{F,j_F^m}$$
$$\text{Then, , } X_{(F+1)} \text{ is } A_{(F+1),j_{(F+1)}^m}$$

where $j_f^m \in [1, T_f]$ identifies the index of the label (among the T_f linguistic terms of partition P_f), which has been selected for X_f in rule R^m. A linguistic RB can be completely described by the following matrix $J \in \Re^{M \times (F+1)}$:

$$J = \begin{bmatrix} j_1^1 & \cdots & j_F^1 & j_{(F+1)}^1 \\ \cdots & \cdots & \cdots & \cdots \\ j_1^m & \cdots & j_F^m & j_{(F+1)}^m \\ \cdots & \cdots & \cdots & \cdots \\ j_1^M & \cdots & j_F^M & j_{(F+1)}^M \end{bmatrix}$$

where the generic element (m, f) indicates that fuzzy set A_{f,j_f^m} has been selected for variable X_f in rule R^m. As an example, let us consider a two-input–single-output system with three labels per variable. Let $P_1 = \{A_{1,1}, A_{1,2}, A_{1,3}\}$, $P_2 = \{A_{2,1}, A_{2,2}, A_{2,3}\}$, and $P_3 = \{A_{3,1}, A_{3,2}, A_{3,3}\}$ be the uniform fuzzy partitions of the two input variables and the output variable, respectively. Then, the following Mamdani system:

$$R^1: \text{If } X_1 \text{ is } A_{1,2} \text{ and } X_2 \text{ is } A_{2,1} \text{ Then, } X_3 \text{ is } A_{3,1}$$
$$R^2: \text{If } X_1 \text{ is } A_{1,2} \text{ and } X_2 \text{ is } A_{2,2} \text{ Then, } X_3 \text{ is } A_{3,1}$$
$$R^3: \text{If } X_1 \text{ is } A_{1,1} \text{ and } X_2 \text{ is } A_{2,1} \text{ Then, } X_3 \text{ is } A_{3,2}$$

is represented by the matrix

$$J = \begin{bmatrix} 2 & 1 & 1 \\ 2 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}.$$

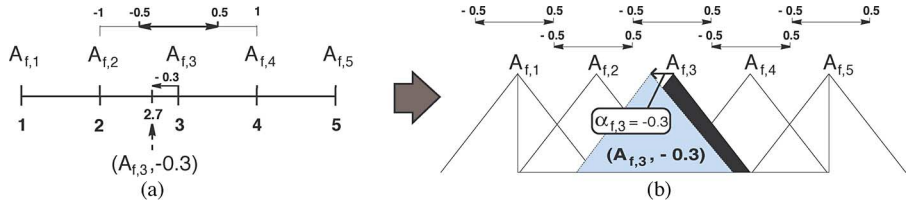

Fig. 2. Symbolic translation of a label and lateral displacement of the associated MF. (a) Simbolic Translation of a label. (b) Lateral Displacement of a Membership function.

In matrix J, the antecedent of each rule involves all input variables. On the other hand, some input variables could be irrelevant or even misleading in practical applications. Thus, it would be desirable to retain only those input variables that actually contribute to the characterization of the system. To this end, we add a new fuzzy set, denoted by $A_{f,0}$, $f = 1, \ldots, F$ for each of the F input partitions P_f: $A_{f,0}$ is characterized by an MF equal to 1 on the overall universe. This means that the condition X_f *is* $A_{f,0}$, denoted as *don't care condition* in the literature, which does not affect the computation of the activation degree. In other words, for the specific rule, the variable X_f is not taken into account.

B. Linguistic Two-Tuple Representation

A new model of the tuning of FRBSs was proposed in [29] considering the linguistic two-tuple representation scheme introduced in [30], which allows the lateral displacement of the support of a label and maintains the interpretability at a good level. This proposal introduces a new model for rule representation based on the concept of symbolic translation [30]. The symbolic translation of a label is a number in $[-0.5, 0.5]$. This interval defines the variation domain of a label when it is moving between its two adjacent lateral labels [see Fig. 2(a)]. Let us consider a generic linguistic fuzzy partition $P_f = \{A_{f,1}, \ldots, A_{f,T_f}\}$. Formally, we represent the symbolic translation of a label $A_{f,j}$ in P_f by means of the two-tuple notation

$$(A_{f,j}, \alpha_{f,j}), \qquad A_{f,j} \in P_f, \qquad \alpha_{f,j} \in [-0.5, 0.5).$$

The symbolic translation of a label involves the lateral variation of its associated MF. This proposal decreases the complexity of the tuning problem, since the three parameters considered per label are reduced to only one translation parameter. Fig. 2 shows the symbolic translation of a label represented by the two-tuple $(A_{f,3}, -0.3)$, together with the associated lateral variation.

In [29], two different rule representation approaches have been proposed: a global approach and a local approach. The global approach tries to obtain more interpretable models, while the local approach tries to obtain more accurate ones. In our case, learning is applied at the level of linguistic partitions (global approach). By considering this approach, the label $A_{f,j}$ is translated with the same $\alpha_{f,j}$ value in all the rules where it is considered, i.e., a global collection of two-tuple is used in all the fuzzy rules. Note that from the parameters $\alpha_{f,j}$ applied to each label, we could obtain the equivalent triangular MFs. Thus, an FRBS based on linguistic two-tuple can be represented as a classic Mamdani FRBS [43], [44]. See [29] for more details on this approach. In this way, from the point of view of interpretability, we have the following.

1) The original shapes of the MFs are maintained (in our case triangular and symmetrical) by laterally changing the location of their supports.
2) The lateral variation of the MFs is restricted to a short interval, thus ensuring overlapping among adjacent labels to some degree but preventing their vertex points from crossing.

C. Positive Synergy Between Rule Learning and the Two-Tuple Representation

Rule learning and MF parameter learning can be used in a synergic way to obtain FRBSs that are not only accurate but also simple. There are several reasons for this positive synergy. Let us presuppose the use of rule learning before parameter learning, as is usual in the literature. In this case, the existence of predefined rules, whose generation process has been optimized for a specific predefined DB, might impose strict constraints on fuzzy partitions, thus reducing the ability of the MF parameter learning to obtain precise linguistic models. Furthermore, the interpretability itself could be affected since the MFs used in some rules could not have the shape and location suited to represent their meaning. On the contrary, learning rules concurrently with MF parameters allows the appropriate adaptation of the RB by changing, removing, or adding rules, in order to facilitate choosing the most appropriate parameters for the problem at hand. In this way, system accuracy can be significantly improved, and system interpretability can be maintained at an acceptable level. This is increasingly true as the problem complexity increases because a larger number of variables usually requires a larger number of rules. Thus, in real complex problems, most of the effort is typically devoted to increasing the performance of some wrong rules rather than to improving the performance of the overall FRBS by performing a complex MF parameter-learning process.

On the other hand, learning the RB and the DB concurrently can make the search space so large that suboptimal models are generated [45]. In this context, the use of the two-tuple representation can help reduce the search space by allowing proper convergence to better global solutions that take the existing dependencies between rules and MF parameters into account. In the following section, we propose to adapt an efficient version of the $(2 + 2)$PAES [25] to learn rules and MF parameters concurrently, thus exploiting the symbolic translation of the MFs. This allows us to achieve the following objectives:

1) to increase the linguistic model accuracy by concurrently searching the best combinations of rule sets and MF definitions;
2) to obtain interpretable linguistic models since the two-tuple representation approach maintains the original shape of MFs, and the lateral displacements are restricted to a short interval. Further, linguistic two-tuple could be interpreted as a change in the linguistic meaning of the labels, as indicated in [29];
3) to favor the combined action of the RB and DB learning strategies by reducing the search space due to the two-tuple representation [29] (only one parameter per label).

IV. Multiobjective Approach for Learning KBs

To obtain a set of KBs with different tradeoffs between accuracy and interpretability, we look for both the J matrix, which represents linguistic RBs, and the displacement parameters of the corresponding MFs in the associated DB. We exploit a double coding scheme (coding of rules and coding of parameters) and prove the effectiveness of the approach by using a modified version of the (2 + 2)PAES [25], which is denoted as *modified* PAES [24]. We call this new approach as PAES_{KB}. Unlike classical (2 + 2)PAES, which only uses mutation to generate new candidate solutions, *modified* PAES exploits the one-point crossover and two appropriately defined mutation operators. The authors experimentally verified that, with these kinds of problems (linguistic modeling), crossover helps to create an approximation of the Pareto front where solutions are uniformly distributed. The crossover and the mutation operators were proposed in [24] to evolve linguistic RBs and have to be combined here with appropriate operators for the parameter-coding part. To assess the goodness of PAES_{KB}, we have also implemented the well-known NSGA-II [23], considering the same objectives as PAES_{KB} and an SOGA driven only by accuracy. All of them use the same chromosome coding scheme, i.e., real coding for the DB and integer coding for the RB, and the same mutation and crossover operators with the same probabilities. The algorithms differ because SOGA and NSGA-II are population-based MOEAs, whereas PAES is a steady-state algorithm. SOGA and NSGA-II use the same initial population. Further, SOGA and NSGA-II use the binary tournament selection, whereas PAES randomly extracts two parents from the archive. In the following sections, we describe the common coding scheme, the initial population generation, the objectives to be optimized, the mating operators, and, finally, the two MOEAs and the SOGA.

A. Coding Scheme and Initial Population

Fig. 3 shows the chromosome representation. The chromosome is composed of two parts: C_{RB} and C_{DB}, which codify the RB and the DB, respectively.

1) C_{RB}: The C_{RB} part is a sequence of integer genes j_f^m ($j_f^m \in [0, T_f]$ for the input variables and $j_{(F+1)}^m \in [1, T_{(F+1)}]$ for the output variable), which identify the indices of the labels selected for variable X_f in rule R^m.
2) C_{DB}: The C_{DB} part is a sequence of real genes $\alpha_{f,j}$, with $f \in [1, F+1]$ and $j \in [1, T_f]$, where $\alpha_{f,j}$ is the translation parameter associated with the label $A_{f,j}$ of partition P_f.

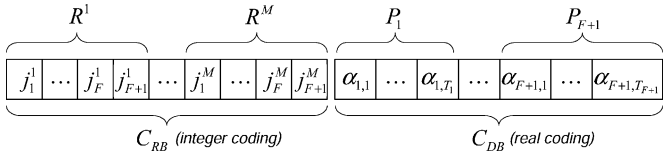

Fig. 3. Chromosome representation.

In real applications, we are not interested in generating an approximation of the overall Pareto optimal front; too-complex solutions with a large number of rules are not typically significant. Thus, to speed up the accuracy computation, we limit the search space by imposing a fixed maximum number $M_{\max}$ of rules that can compose the system. Thus, the maximum length of the chromosome is fixed. Also, the minimum length is fixed to $M_{\min}$, where $M_{\min}$ is the minimum number of rules that a model can comprise.

The initial population is obtained with all individuals generated at random within the corresponding variation intervals. For each individual, a random number is generated in $[M_{\min}, M_{\max}]$, determining the number of rules that are going to be generated for this individual. Then, in order to generate each rule, the same process described in Section IV-C for the first mutation operator is considered. Once the integer part of the chromosome has been generated, the real part is obtained by randomly extracting real values in $[-0.5, 0.5]$.

B. Objectives

Every chromosome is associated with a 2-D objective vector, with each element expressing the fulfillment degree of a different objective. The first element of the vector measures the complexity as the number of genes, corresponding to the antecedents, which differ from 0, i.e., the sum of the input variables actually used in each of the M rules. The second element corresponds to the mean squared error (MSE), which is defined as

$$\text{MSE} = \frac{1}{2|E|}\sum_{l=1}^{|E|}(F(x^l) - y^l)^2$$

where $|E|$ is the dataset size, $F(x^l)$ is the output of the FRBS when the lth example is an input, and y^l is the known desired output. The fuzzy inference system considered to obtain $F(x^l)$ uses the *center of gravity weighted by the matching* strategy as a defuzzification operator and the *minimum t-norm* as implication and conjunctive operators.

C. Genetic Operators and Their Application

This section describes the specific genetic operators employed to concurrently evolve RB and DB. For offspring generation, we adopt a hybrid crossover operator acting in both chromosome parts and three different mutation operators, with the first two acting in the RB part and the third in the DB part.

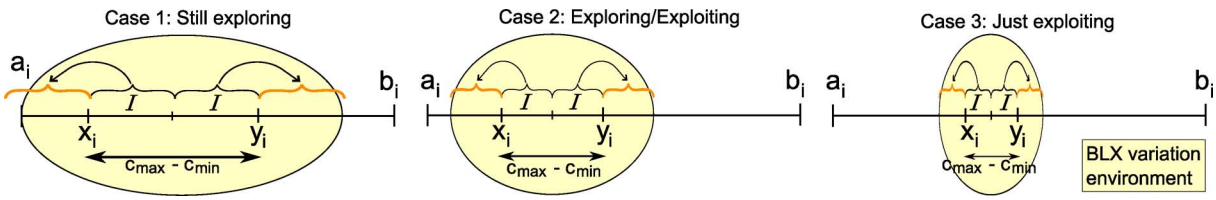

Fig. 4. Different cases/stages of the application of the BLX-α crossover operator, with $\alpha = 0.5$.

1) Crossover Operator: With regard to the crossover operator, we apply the classical one-point crossover for the $C_{\rm RB}$ part and the BLX-α operator [46], [47], with $\alpha = 0.5$, for the $C_{\rm DB}$ part. Let C^1 and C^2 be the two current solutions.

1) The one-point crossover operator cuts the chromosomes $C^1_{\rm RB}$ and $C^2_{\rm RB}$ at some chosen common gene and swaps the resulting subchromosomes. The common gene is chosen randomly in $[M_{\min}, \rho_{\min}]$, where $M_{\min}$ is the minimum number of rules, which must be present in an RB, and $\rho_{\min}$ is the minimum number of rules in $C^1_{\rm RB}$ and $C^2_{\rm RB}$.
2) The BLX-α operator is applied twice considering $C^1_{\rm DB}$ and $C^2_{\rm DB}$ in order to obtain the $C_{\rm DB}$ parts of both offsprings. Let us assume that $C^1_{\rm DB} = (x_1, \ldots, x_g)$ and $C^2_{\rm DB} = (y_1, \ldots, y_g)$, $(x_i, y_i \in [a_i, b_i) = [-0.5, 0.5) \subset \Re, i = 1, \ldots, g)$ are the two real-coded chromosomes that are going to be crossed. Using the BLX-α crossover, one descendant $Z = (z_1, \ldots, z_g)$ is obtained, where z_i is randomly (uniformly) generated within the interval $[l_i, u_i]$, with $l_i = \max\{a_i, c_{\min} - I\}$, $u_i = \min\{b_i, c_{\max} + I\}$, $c_{\min} = \min\{x_i, y_i\}$, $c_{\max} = \max\{x_i, y_i\}$, and $I = (c_{\max} - c_{\min})\alpha$. Fig. 4 shows how the BLX-α operator works at different stages of the evolution process.

2) Mutation Operators: Three different mutation operators are applied (the first two to the $C_{\rm RB}$ part and the other to the $C_{\rm DB}$ part).

1) The first mutation operator adds γ rules to the RB, where γ is randomly chosen in $[1, \gamma_{\max}]$. The upper bound $\gamma_{\max}$ is fixed by the user (we used $\gamma_{\max} = 5$ in the experiments). If $\gamma + M > M_{\max}$, then $\gamma = M_{\max} - M$. For each rule R^m added to the chromosome, we generate a random number $t \in [1, F]$, which indicates the number of input variables used in the antecedent of the rule. Then, we generate t natural random numbers between 1 and F to determine the input variables that compose the antecedent part of the rule. Finally, for each selected input variable X_f, we generate a random natural number j^m_f between 1 and T_f, which determines the linguistic label A_{f,j^m_f} to be used in the corresponding input variable of rule R^m. To select the consequent linguistic term, a random number between 1 and $T_{(F+1)}$ is generated.
2) The second mutation operator randomly changes δ elements of matrix J. The number is randomly generated in $[1, \delta_{\max}]$, where $\delta_{\max}$ is fixed by the user (we used $\delta_{\max} = 5$ in the experiments). For each element to be modified, a number is randomly generated in $[0, T_f]$, where f is the input variable corresponding to the selected matrix element.
3) The third mutation operator simply changes a gene value at random in the $C_{\rm DB}$ part.

The crossover operator helps the algorithm perform a good exploitation of the solutions generated so far. The first mutation operator (the one that adds rules) is useful to generate solutions of high complexity, and the second and the third operators (the ones that change randomly some genes in $C_{\rm RB}$ and $C_{\rm DB}$, respectively) allow a good exploitation of the search space. Thus, it follows that all the operators are necessary. In addition, we performed several simulations in order to determine the effectiveness of all the operators and to find a good setup for the associated probabilities. For example, we removed the one-point crossover and observed that the algorithm did not explore satisfactorily the search space, thus converging toward solutions concentrated in the "low complexity–low accuracy" region. Regarding the mutation operators, we tried to remove the different operators and execute the algorithm. We verified that, with the removal of the first operator (the one that adds rules), the Pareto front approximation was shorter and did not have complex solutions. Further, we observed that when the second operator (the one that changes randomly some gene in $C_{\rm RB}$) was removed, the solutions of the final Pareto front approximations presented worse tradeoffs.

The probability of applying the crossover operator is $P_{\rm cross} = 0.5$. As regards mutation, we manage separately the RB and the DB parts. For the RB part, when the application of the crossover operator is selected, the mutation is applied with probability $Pm_{\rm RB} = 0.01$; otherwise, the mutation is always applied with $Pm_{\rm RB} = 1$. When the application of the RB mutation is selected, the first mutation operator is applied with probability $Pm_{Add} = 0.55$, and the second mutation operator is applied when the first is not applied. The third mutation operator is always applied with probability $Pm_{\rm DB} = 0.2$. Pseudocode describing the application scheme of the different operators is shown in Fig. 5. These values of probabilities have been determined after an accurate experimentation aimed at balancing exploitation of the current solutions and exploration of the search space.

D. Modified PAES Evolutionary Model

The execution of the modified PAES starts with two randomly generated current solutions C^1 and C^2. At each iteration, the application of crossover and mutation operators produces two new solutions S^1 and S^2 from the current solutions C^1 and C^2. Unlike classical (2 + 2)PAES, which maintains C^1 and C^2 as current solutions as long as they are not replaced by solutions with particular characteristics, we randomly extract,

```
[p1, p2] = parent selection(archive/population); // PAES/NSGA-II/SOGA)
                                                 // parent selection strategy
if (rand() < Pcross)
      [s1, s2] = crossover(p1,p2);
      PmKB = 0.01;
else
      s1 = p1;
      s2 = p2;
      PmKB = 1;
endif
Loop i=1,2
      if (rand() < PmKB)
            if (rand <PmAdd)
                  s1 = add_rule();
            else
                  s1 = modify_rule_base();
            endif
      endif
      if (rand() < PmDB)
            s1 = mutate_DB();
      endif
endLoop
```

Fig. 5. Genetic operators application scheme.

at each iteration, the current solutions. If the archive contains a unique solution, C^1 and C^2 correspond to this unique solution. It was verified in [24] that the random extraction of the current solutions from the archive allows the extension of the set of nondominated solutions contained in the archive in order to obtain a better approximation of the Pareto front. With regard to the candidate solution acceptance, S^1 and S^2 are added to the archive only if they are not dominated by any solution contained in the archive; possible solutions in the archive dominated by S^1 and S^2 are removed. Typically, the size of the archive is fixed at the beginning of the execution of PAES. In this case, when the archive is full and a new solution S has to be added to the archive, if S dominates no solution in the archive, then S is inserted into the archive and the solution (possibly S itself), which belongs to the region with the highest crowding degree, is removed. If the region contains more than one solution, then the solution to be removed is randomly chosen. The algorithm ends when a specified number of evaluations are reached.

E. NSGA-II Evolutionary Model

NSGA-II is a population-based MOEA, which uses an *ad hoc* density-estimation metric and a nondominance rank assignment [23]. NSGA-II starts from an initial random population P_0 of $N_{\rm pop}$ individuals sorted according to nondominance. Each individual is associated with a rank equal to its nondominance level (1 for the best level, 2 for the next best level, and so on). More precisely, first, the nondominated individuals are found and associated with rank 1; then, in order to find the individuals with rank 2, i.e., those in the next nondominated front, the individuals of the first front are temporarily discarded, and so on. At each iteration t, an offspring population Q_t of size $N_{\rm pop}$ is generated by selecting mating individuals through the binary tournament selection and by applying crossover and mutation operators. Then, parent population P_t and offspring population Q_t are combined to generate a new population $P_{\rm ext} = P_t \cup Q_t$. A rank based on the nondominance level is assigned to each individual in $P_{\rm ext}$. Using these ranks, $P_{\rm ext}$ is split into different nondominated fronts, one for each different rank. Within each front, a specific crowding measure, which represents the sum of the distances from each individual to its closest individual along each objective, is used to define an ordering among individuals. The new parent population P_{t+1} is generated deleting from $P_{\rm ext}$ the worst $N_{\rm pop}$ individuals (considering first the ordering among fronts and then among individuals). The algorithm ends when a specified number of evaluations are reached.

TABLE II
DATASETS CONSIDERED FOR THE EXPERIMENTAL STUDY

Datasets	Name	Variables	Patterns
Electrical Length	ELE1	2	495
Electrical Maintenance	ELE2	4	1056
Abalone	ABA	8	4177
Weather Izmir	IZ	10	1461
Weather Ankara	WA	10	1609
Treasury	TR	15	1049
Mortgage	MO	15	1049
Computer Activity	CA	21	8192
Pole Telecommunication	PT	26	15000

Available at http://sci2s.ugr.es/kee1/datasets.php

F. Single-Objective Evolutionary Model: SOGA

SOGA starts from an initial random population P_0 of $N_{\rm pop}$ individuals. At each iteration t, the MSE of each individual is evaluated, and individuals are selected for reproduction by using the binary tournament selection. The crossover and the mutation operators explained in Section IV-C are applied to the selected individuals with the same probabilities to generate the offspring population Q_t of $N_{\rm pop}$ individuals. The new population P_{t+1} coincides with Q_t, except for the worst 5%, which is replaced with the best 5% individuals of P_t. The algorithm terminates when a specified number of evaluations are reached.

V. EXPERIMENTS

To evaluate the proposed approach, we have used nine real-world problems with different complexities (different numbers of variables and available data). Table II summarizes the main characteristics of the nine datasets and shows the link to the KEEL project Web page [48] from which they can be downloaded. This section is organized as follows.

1) First, we describe the experimental setup in Section V-A.
2) Second, we compare the different MOEAs in terms of the most accurate solutions in Section V-B.
3) Third, we discuss the relevance of data complexity in affecting the performance of the proposed approach in Section V-C.
4) Fourth, we compare the $\rm PAES_{KB}$ with the two SOGAs in Section V-D.
5) Fifth, in Section V-E, for each dataset and for each MOEA, we plot on the accuracy–complexity plane the centroids of the first (the most accurate), median, and last (the least accurate) solutions obtained on the training and test sets in the different trials of the algorithms. We denote these centroids as FIRST, MEDIAN, and LAST. These plots provide a glimpse of the trend of the Pareto fronts. We also show the centroids of the solutions generated by the two SOGAs in the different trials. For completeness, we also show some representative Pareto fronts achieved by the different MOEAs.

TABLE III
METHODS CONSIDERED FOR COMPARISON

Method	Ref.	Description
Algorithms without DB learning		
SOGA$_{RB}$	[24]	RB Learning by SOGA
NSGA-II$_{RB}$	[24]	RB Learning by NSGA-II
PAES$_{RB}$	[24]	RB Learning by PAES†
Algorithms with DB learning		
SOGA$_{KB}$	—	RB+DB Learning by SOGA
NSGA-II$_{KB}$	—	RB+DB Learning by NSGA-II
PAES$_{KB}$	—	RB+DB Learning by PAES†

† modified version of PAES.

6) Finally, we show the computational costs of the different algorithms in Section V-F.

A. Experimental Setup

We applied the algorithms described in the previous sections, namely, the modified PAES, NSGA-II, and SOGA, with the MSE as the unique objective. In order to evaluate the advantages of concurrently learning DB and RB, we applied the algorithms both with and without DB learning (in practice, we consider chromosomes composed of both C_{RB} and C_{DB} parts in the former case and composed of only the C_{RB} part in the latter case). Table III summarizes the algorithms used in the experiments.

We initially consider uniform fuzzy partitions of triangular symmetrical MFs. We used five MFs for all the datasets, except for Abalone and PT. For Abalone, we decided to use three fuzzy sets for all the variables, as one of the variables has only three values. On the other hand, we verified that the accuracy of the models is quite similar by considering three or five linguistic labels. With regard to PT, since the number of variables is quite high (26 inputs), we decided to consider only three linguistic labels in order to limit the number of parameters. The algorithm parameters are set as follows: the maximum $M_{\max}$ and the minimum $M_{\min}$ numbers of rules to 30 and 5, the maximum number of evaluations to 300 000, and the population or archive size to 64.

In all the experiments, we adopted a *fivefold cross-validation model*, i.e., we randomly split the dataset into five folds, with each containing 20% of the patterns of the dataset, and used four folds for training and one for testing.[1] For each of the five partitions, we executed six trials of the algorithms. For each dataset, we therefore consider the average results of 30 runs. In the case of methods with multiobjective approach (all but SOGA), for each dataset and for each trial, we generate the approximated Pareto front. Then, we focus on three representative points: the first (the most accurate), the median, and the last (the least accurate) points. For each dataset, we compute the mean values and the standard deviations over the 30 trials of the MSEs on the training and test sets and the number of rules and number of variables actually used in the FRBSs. For the SOGAs, we compute the same mean values and standard deviations over the 30 solutions obtained for each dataset. In order to assess whether significant differences exist among the results, we adopt statistical analysis [32]–[34] and, in particular, nonparametric tests, according to the recommendations made in [31], where a set of simple, safe, and robust nonparametric tests for statistical comparisons of classifiers have been introduced. In particular, we use the Wilcoxon signed-rank test [49], [50] for pairwise comparison of the different algorithms. A detailed description of this test is presented in Appendix I. To perform the test, we use a level of confidence $\alpha = 0.1$. This statistical test is based on computing the differences on two sample means (typically, mean test errors obtained by a pair of different algorithms on different datasets). In the classification framework, these differences are well defined since these errors are in the same domain. In the regression framework, to make the differences comparable, we propose to adopt a normalized difference DIFF, which is defined as

$$\mathrm{DIFF} = \frac{\mathrm{MSE}_{Ts}(\mathrm{Other}) - \mathrm{MSE}_{Ts}(\mathrm{PAES}_{\mathrm{KB}})}{\mathrm{MSE}_{Ts}(\mathrm{Other})}. \quad (1)$$

This difference expresses the improvement percentage of the reference algorithm ($\mathrm{PAES}_{\mathrm{KB}}$ in our case) on the other one.

B. Comparison Among the Different MOEAs in Terms of the Most Accurate Solution

Table IV shows the average results corresponding to the most accurate solutions for the nine datasets and the four MOEAs. In the table, R, C, V, MSE_{Tr}, MSE_{Ts}, DIFF, and SD, denote, respectively, the number of rules, the complexity, the number of variables actually used in the solution, the MSE on the training set, the MSE on the test set, the improvement percentage of the reference algorithm ($\mathrm{PAES}_{\mathrm{KB}}$), and the standard deviation.

We observe that both $\mathrm{PAES}_{\mathrm{KB}}$ and $\mathrm{NSGA\text{-}II}_{\mathrm{KB}}$ always outperform their counterparts without DB learning on the training set, i.e., $\mathrm{PAES}_{\mathrm{RB}}$ and $\mathrm{NSGA\text{-}II}_{\mathrm{RB}}$, respectively. This proves the effectiveness of the synergy between rule and parameter learning. We also note that $\mathrm{PAES}_{\mathrm{KB}}$ outperforms all the MOEAs on the training set and on six out of nine datasets on the test set. In these three datasets, however, only $\mathrm{PAES}_{\mathrm{RB}}$ outperforms $\mathrm{PAES}_{\mathrm{KB}}$. This confirms that the modified PAES is one of the best performing MOEAs for FRBS generation. To assess whether we can conclude that statistically $\mathrm{PAES}_{\mathrm{KB}}$ outperforms $\mathrm{PAES}_{\mathrm{RB}}$ in terms of MSEs, we apply the Wilcoxon signed-rank test to the results achieved by these algorithms.

Table V shows the results of the application of the Wilcoxon test on the test set. Here, R^+ and R^-, respectively, denote the sum of the ranks corresponding to $\mathrm{PAES}_{\mathrm{KB}}$ and $\mathrm{PAES}_{\mathrm{RB}}$, respectively. The null hypothesis associated with the Wilcoxon signed-rank test is accepted ($p > \alpha$). Thus, we cannot conclude that the results achieved by $\mathrm{PAES}_{\mathrm{KB}}$ and $\mathrm{PAES}_{\mathrm{RB}}$ are statistically different on the test set. On the other hand, since $\mathrm{PAES}_{\mathrm{KB}}$ always outperforms $\mathrm{PAES}_{\mathrm{RB}}$ on the training set, the null hypothesis is always rejected on the training set.

By analyzing Table IV, we can observe that $\mathrm{PAES}_{\mathrm{RB}}$ outperforms $\mathrm{PAES}_{\mathrm{KB}}$ on the datasets that are characterized by a low number of patterns in comparison with the number of variables. Since $\mathrm{PAES}_{\mathrm{KB}}$ has to manage a higher number of parameters

[1]The corresponding data partitions (fivefold) for these datasets are available at the KEEL project Web page (see [48]: http://sci2s.ugr.es/keel/datasets.php)

TABLE IV
AVERAGE RESULTS ACHIEVED BY THE DIFFERENT MOEAS IN CORRESPONDENCE TO THE FIRST POINT

Data set	Measure	NSGA-II$_{RB}$						NSGA-II$_{KB}$						PAES$_{RB}$						PAES$_{KB}$				
NAME (V;SIZE)		R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$
ELE1 (2;495)	Mean	29	45	2	156952	205182	0.36	29	46	2	157613	202923	0.19	29	47	2	156775	210162	0.08	27	46	2	**145995**	**194028**
	SD				4767	37396					5072	34661					4303	45841					3855	24745
ELE2 (4;1056)	Mean	29	63	4	17115	19834	0.05	29	66	4	13136	15587	0.04	30	62	4	15453	17135	0.26	30	65	4	**11043**	**12606**
	SD				4282	4995					3378	4805					3881	4233					2771	3105
ABA (8;4177)	Mean	24	84	8	2.52	2.63	0.06	19	68	8	2.47	5.53	0.55	29	98	8	2.55	2.64	0.06	29	107	8	**2.32**	**2.48**
	SD				0.10	0.23					0.19	7.58					0.11	0.27					0.08	0.18
IZ (9;1462)	Mean	23	82	8	2.05	2.38	0.37	22	80	8	1.64	1.91	0.22	25	81	9	1.62	1.89	0.21	25	91	9	**1.30**	**1.49**
	SD				0.71	0.96					0.34	0.48					0.43	0.54					0.27	0.26
WA (9;1609)	Mean	27	93	8	3.46	4.16	0.06	19	70	8	2.57	16.84	0.77	28	93	9	2.70	**3.50**	-0.12	28	103	9	**1.64**	3.92
	SD				1.05	1.83					0.87	25.52					1.27	1.90					0.40	9.27
TR (15;1049)	Mean	7	23	11	0.16	0.22	0.36	8	30	11	0.12	0.18	0.20	9	28	11	**0.08**	**0.12**	-0.18	11	40	12	**0.08**	0.14
	SD				0.08	0.24					0.06	0.14					0.03	0.09					0.04	0.15
MO (15;1049)	Mean	7	23	11	0.11	0.16	0.41	10	41	13	0.09	0.13	0.29	9	34	12	0.06	**0.08**	-0.21	12	49	14	**0.05**	0.09
	SD				0.06	0.16					0.04	0.12					0.03	0.05					0.02	0.10
CA (21;8192)	Mean	8	14	7	26.43	27.20	0.51	13	22	8	17.07	20.96	0.36	7	16	10	18.21	19.67	0.32	10	30	12	**11.99**	**13.43**
	SD				13.24	12.52					7.00	9.02					3.97	5.16					2.99	4.66
PT (26;15000)	Mean	12	39	15	446	447	0.80	13	53	16	201	304	0.70	17	58	17	370	370	0.76	14	53	19	**87**	**89**
	SD				92	91					190	407					77	80					26	25

TABLE V
WILCOXON TEST TO COMPARE PAES$_{\mathrm{RB}}$ AND PAES$_{\mathrm{KB}}$ WITH RESPECT TO THE FIRST SOLUTION

Comparison	R^+	R^-	Hypothesis ($\alpha = 0.1$)	p-value
PAES$_{RB}$ vs. PAES$_{KB}$	13	32	Accepted	0.260

R$^+$ corresponds to PAES$_{\mathrm{RB}}$ and R$^-$ to PAES$_{\mathrm{KB}}$.

than PAES$_{\mathrm{RB}}$ (the difference is equal to the number of variables multiplied by the number of labels), PAES$_{\mathrm{KB}}$ is more prone to the overfitting problem. It is likely that the number of patterns for the three datasets is not too large to allow a good generalization. In the next section, we will introduce a very simple measure of data complexity and will show how, based on this measure, we can decide whether applying the learning of parameters, together with the rule learning, is convenient or not. Finally, we observe that the application of the MOEAs, especially in datasets characterized by a high number of variables, produces a variable selection.

C. On the Use of a Data Complexity Measure for a Characterization of the Proposed Method

The analysis of data complexity is an emergent topic in the framework of pattern recognition; it aims to characterize the datasets in order to assess whether specific learning techniques [35], [36] can perform satisfactorily with them. Indeed, it is well known that various aspects of data complexity affect the behavior of the classifiers in different ways. Determination of these aspects, however, is not an easy task. Over the part few years, different research has been devoted to discussing and proposing several measures useful for this characterization [36].

To the best of our knowledge, data complexity has not been sufficiently analyzed in regression. Indeed, most metrics have been proposed based on the existence of classes (for instance, fraction of points on class boundary and error rate of linear classifiers extracted by linear programming). Nevertheless, some of these metrics can be used directly in regression. For instance, when dealing with the *curse of dimensionality* problem, a simple index, which is denoted by T2, is typically used [35], [36] to compare the number of patterns to the number of variables. T2 is defined as the average number of patterns per variable. T2 can obviously be directly applied to datasets used in regression problems. In this section, we propose to use T2 to characterize the problems we are considering in order to derive indications on the applicability of the proposed technique.

TABLE VI
DATASETS SORTED BY THE T2 DATA COMPLEXITY MEASURE

Data set	MSE$_{Ts}$(PAES$_{RB}$)	MSE$_{Ts}$(PAES$_{KB}$)	DIFF	**T2**
TR (15;1049)	0.12	0.14	-0.18	69.93
MO (15;1049)	0.08	0.09	-0.21	69.93
IZ (9;1462)	1.89	1.49	0.21	162.33
WA (9;1609)	3.50	3.92	-0.12	178.78
ELE1 (2;495)	210162	194028	0.08	247.50
ELE2 (4;1056)	17135	12606	0.26	264.00
CA (21;8192)	19.67	13.43	0.32	390.10
ABA (8;4177)	2.64	2.48	0.06	522.13
PT (26;15000)	370	89	0.76	576.92

TABLE VII
WILCOXON TEST TO COMPARE PAES$_{\mathrm{RB}}$ AND PAES$_{\mathrm{KB}}$ WITH RESPECT TO THE MOST ACCURATE SOLUTIONS BY ONLY CONSIDERING THE SEVEN DATASETS WITH LARGEST T2 VALUES

Comparison	R^+	R^-	Hypothesis ($\alpha = 0.1$)	p-value
PAES$_{RB}$ vs. PAES$_{KB}$	3	25	Rejected for PAES$_{KB}$	0.063

We computed T2 for each dataset. Table VI shows the datasets sorted by increasing T2 values. In the table, we also present the mean MSEs of the most accurate solutions obtained on test sets by PAES$_{\mathrm{RB}}$ and PAES$_{\mathrm{KB}}$ and the normalized difference DIFF defined in Section V-A. We can observe that two out of the three datasets on which PAES$_{\mathrm{RB}}$ outperforms PAES$_{\mathrm{KB}}$ are characterized by the lowest value of T2. Consequently, we have applied the Wilcoxon test to the seven datasets with the highest T2 values. Table VII shows the results of the application of the Wilcoxon test. We note that the null hypothesis is rejected ($p < \alpha$). Thus, we can conclude that PAES$_{\mathrm{KB}}$ outperforms PAES$_{\mathrm{RB}}$ on the test set.

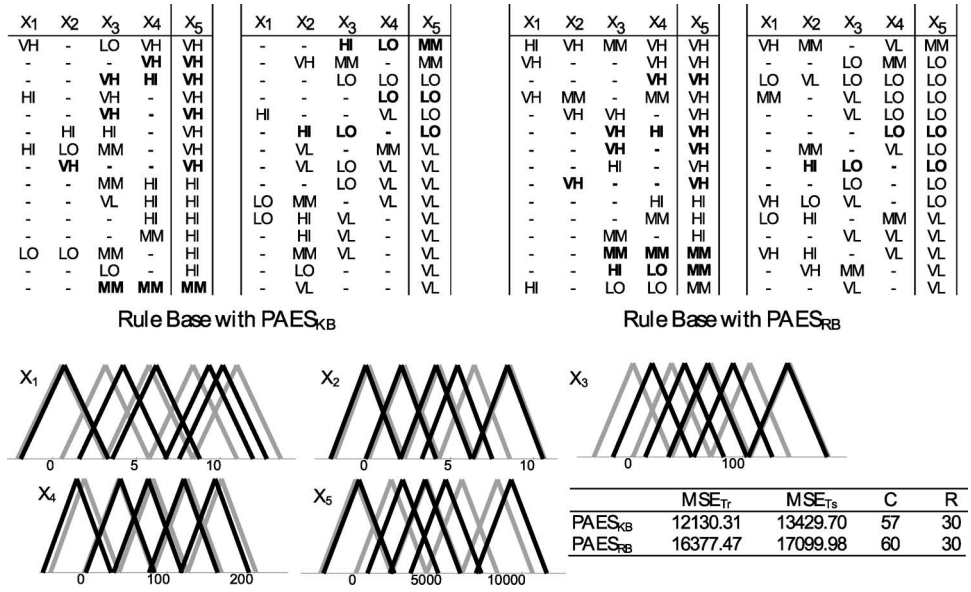

Fig. 6. Example of KBs generated by PAES$_{\mathrm{KB}}$ and PAES$_{\mathrm{RB}}$ on the same data partition and seed (DBs with and without MF parameter learning are represented in black and gray, respectively). Rules that appear in both the KBs are in bold.

TABLE VIII
AVERAGE RESULTS OF THE TWO SOGAS COMPARED WITH THE MOST ACCURATE SOLUTIONS GENERATED BY PAES$_{\mathrm{KB}}$

Data set	SOGA$_{RB}$						SOGA$_{KB}$						PAES$_{KB}$				
NAME (V;SIZE)	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$
ELE1 (2;495)	30	52	2	167346	209198	0.56	30	51	2	171689	211035	0.34	27	46	2	**145995**	**194028**
ELE2 (4;1056)	30	83	4	24340	28633	0.07	30	88	4	16502	19112	0.08	28	46	4	**11043**	**12606**
ABA (8;4177)	30	157	8	2.77	2.87	0.14	30	168	8	2.45	2.60	0.05	29	107	8	**2.32**	**2.48**
IZ (9;1462)	30	150	9	4.17	4.81	0.69	30	160	9	3.65	4.89	0.69	25	91	9	**1.30**	**1.49**
WA (9;1609)	30	170	9	4.46	4.96	0.21	30	169	9	5.25	6.44	0.39	28	103	9	**1.64**	**3.92**
TR (15;1049)	30	257	15	0.34	0.47	0.70	30	298	15	0.33	1.37	0.89	11	40	12	**0.08**	**0.14**
MO (15;1049)	30	259	15	0.38	0.64	0.86	30	287	15	0.28	0.66	0.86	12	49	14	**0.05**	**0.09**
CA (21;8192)	30	502	21	21.47	24.65	0.46	30	497	21	19.32	21.65	0.38	10	30	12	**11.99**	**13.43**
PT (26;15000)	28	228	12	1286	1299	0.90	30	225	15	892	893	0.93	14	53	19	**87**	**89**

From these results, we can deduce that, for datasets characterized by low values of T2, concurrently learning data and RBs has a high probability of generating FRBSs with overfitting problems. Thus, for these datasets, we conclude that PAES$_{\mathrm{RB}}$ is preferable to PAES$_{\mathrm{KB}}$ since the number of parameters managed by PAES$_{\mathrm{RB}}$ is smaller than the one managed by PAES$_{\mathrm{KB}}$. On the other hand, when the value of T2 increases, the probability of generating FRBSs with overfitting problems decreases, and PAES$_{\mathrm{KB}}$ outperforms PAES$_{\mathrm{RB}}$.

In Fig. 6, we show the RBs and DBs for one of the most accurate solutions generated by PAES$_{\mathrm{KB}}$ and PAES$_{\mathrm{RB}}$ on the ELE2 dataset. The two solutions are extracted from the same trial (same data partition and seed). The FRBS generated by PAES$_{\mathrm{KB}}$ achieves MSEs on the training and test sets equal to 12 130 and 13 430, respectively, with a complexity of 57 and a number of rules equal to 30. The FRBS generated by PAES$_{\mathrm{RB}}$ achieves MSEs on the training and test sets equal to 16 377 and 17 100, respectively, with a complexity of 60 and a number of rules equal to 30. The difference between the accuracies is, therefore, very large, although the difference between the complexities is quite low. In Fig. 6, we can, however, observe that the interpretability of the fuzzy partitions in terms of coverage, order, and distinguishability is maintained. As a result, due to the features of the two-tuple representation, the improvement in accuracy is achieved without considerable detriment to interpretability. With regard to the RBs, we note that the rules generated by PAES$_{\mathrm{KB}}$ are different from the rules generated by PAES$_{\mathrm{RB}}$, except for eight rules. Indeed, when learning concurrently MF parameters and rules, the rules are adapted to the new linguistic meanings determined by the evolving MFs.

D. Comparing PAES$_{\mathrm{KB}}$ With SOGA

In this section, we aim to compare the most accurate solutions generated by PAES$_{\mathrm{KB}}$ with the solutions provided by the single-objective approaches (with and without DB learning). Table VIII shows the average results on the nine datasets generated by SOGA$_{\mathrm{RB}}$, SOGA$_{\mathrm{KB}}$, and PAES$_{\mathrm{KB}}$, respectively. We observe that PAES$_{\mathrm{KB}}$ outperforms the two single-objective approaches on both the training and test sets. In this case, the Wilcoxon test rejects the null hypothesis with $p = 0$. For the sake of readability, we do not show the standard deviations associated with the mean MSEs. The values of the standard deviations for the two single-objective approaches are, on average,

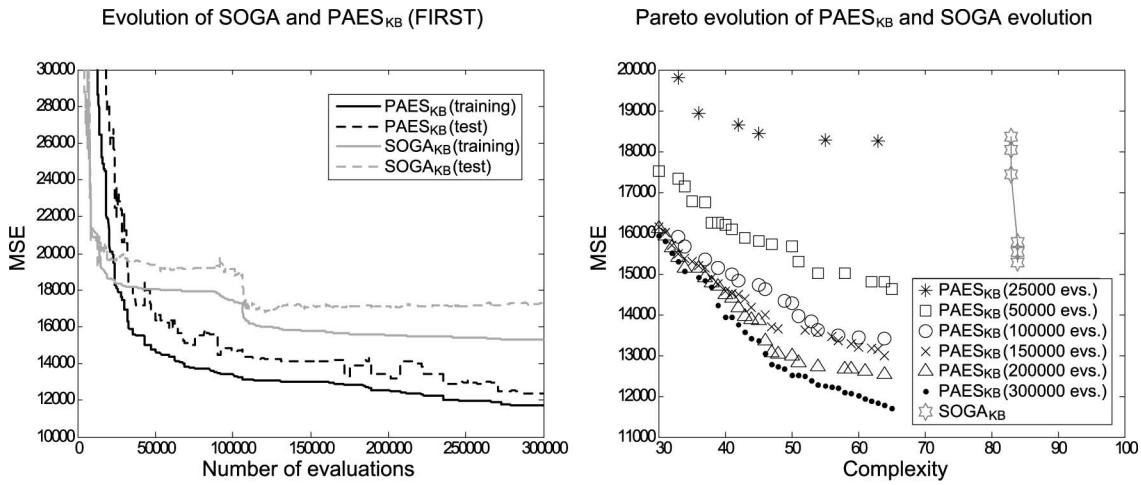

Fig. 7. Example of evolution of SOGA and PAES$_{\mathrm{KB}}$ on the ELE2 dataset. (a) Evolution of the solution of SOGA and the most accurate solution of the Pareto fronts. (b) Evolution of the solution of SOGA and the Pareto fronts.

a little higher than the ones associated with the most accurate solutions generated by the MOEAs. This allows us to conclude that MOEAs achieve the most accurate solutions with a higher stability. With regard to the complexity, we can observe that the number of rules of the FRBSs generated by the two SOGAs is always equal to the maximum possible value (30), except for the PT dataset, in which SOGA$_{\mathrm{RB}}$ generates FRBSs with 28 rules on average. On the other hand, PAES$_{\mathrm{KB}}$ manages to generate solutions with a lower number of rules and a lower complexity level. This trend can be further appreciated when dealing with high-dimensional datasets. The better performance of the multiobjective approaches, in particular of PAES$_{\mathrm{KB}}$, is probably due to the fact that they are able to maintain the population diversity. Indeed, MOEAs are able to perform a more accurate exploration of the search space that allows PAES$_{\mathrm{KB}}$ to avoid getting stuck at local optima, thus obtaining the best results for both MSE and complexity objectives at the same time.

In order to show how both SOGA$_{\mathrm{KB}}$ and PAES$_{\mathrm{KB}}$ converge, Fig. 7 presents the evolution in one representative run on ELE2. The two plots represent the most accurate solution in the Pareto front from PAES$_{\mathrm{KB}}$ (left plot) and the evolution of the complete Pareto front from PAES$_{\mathrm{KB}}$ (right plot). In this figure, we can observe how SOGA$_{\mathrm{KB}}$ gets stuck in the zone with more complex rules, thus causing premature convergence.

The ELE2 dataset has been extensively used in the literature as a benchmark for different FRBS identification approaches. For example, in [51] (where this dataset was introduced), an FRBS, which was generated by applying an SOGA for selecting rules and tuning MF parameters of the KB of a Mamdani linguistic model identified by the Wang–Mendel approach, was applied to ELE2. The MSEs obtained on the training and test sets were, respectively, 19 679 and 22 591 with an FRBS composed of 63 rules (these results refer to only one run performed on a data partition, which corresponds to the first partition used in the our experiments). By comparing these results with the ones in Table VIII, we note that concurrently learning RBs and DBs in a single-objective evolutionary approach compares favorably with the SOGA approach proposed in [51]. Further, we can conclude that the multiobjective approaches have to be preferred in order to obtain FRBSs with good tradeoffs between accuracy and complexity.

E. Comparing the Pareto Fronts

Since we perform 30 trials with different training and test partitions, it would not be readable to show all the Pareto fronts. Thus, to have a glimpse of the trends of the Pareto fronts in the complexity–accuracy plane, we plot the FIRST, MEDIAN, and LAST points for each MOEA and for each dataset in Fig. 8. We recall that the FIRST, MEDIAN, and LAST points denote the centroids of the first, median, and last points, respectively, of the 30 Pareto fronts obtained by applying the fivefold cross-validation described in Section V-B. We also show the centroids of the solutions generated by the two SOGAs. In Appendix, for the sake of completeness, we show the average results of the different MOEAs for the MEDIAN and LAST points.

The analysis of Fig. 8 shows that the approximations of the Pareto fronts achieved by PAES$_{\mathrm{KB}}$ are, in general, below the approximations of the Pareto fronts obtained by the other MOEAs (except for the data sets for which we have already discussed some flaw in the previous sections). To compare in detail the different MOEAs with respect to the MEDIAN and LAST points, in Table XII, we also show the results of the application of the Wilcoxon test for these points between PAES$_{\mathrm{KB}}$ and PAES$_{\mathrm{RB}}$ (we do not show the application of the Wilcoxon test between PAES$_{\mathrm{KB}}$ and the other MOEAs because it always rejects the null hypothesis). With regard to the MEDIAN point, we observe the same behavior as for the FIRST point, i.e., PAES$_{\mathrm{KB}}$ outperforms in terms of accuracy all the other MOEAs, both on the training set and, except for PAES$_{\mathrm{RB}}$, on the test set. On the test set, statistical differences between PAES$_{\mathrm{RB}}$ and PAES$_{\mathrm{KB}}$ do not exist when considering all the datasets. On the contrary, PAES$_{\mathrm{KB}}$ statistically outperforms PAES$_{\mathrm{RB}}$ when applying the Wilcoxon test to the seven datasets with the highest T2 values

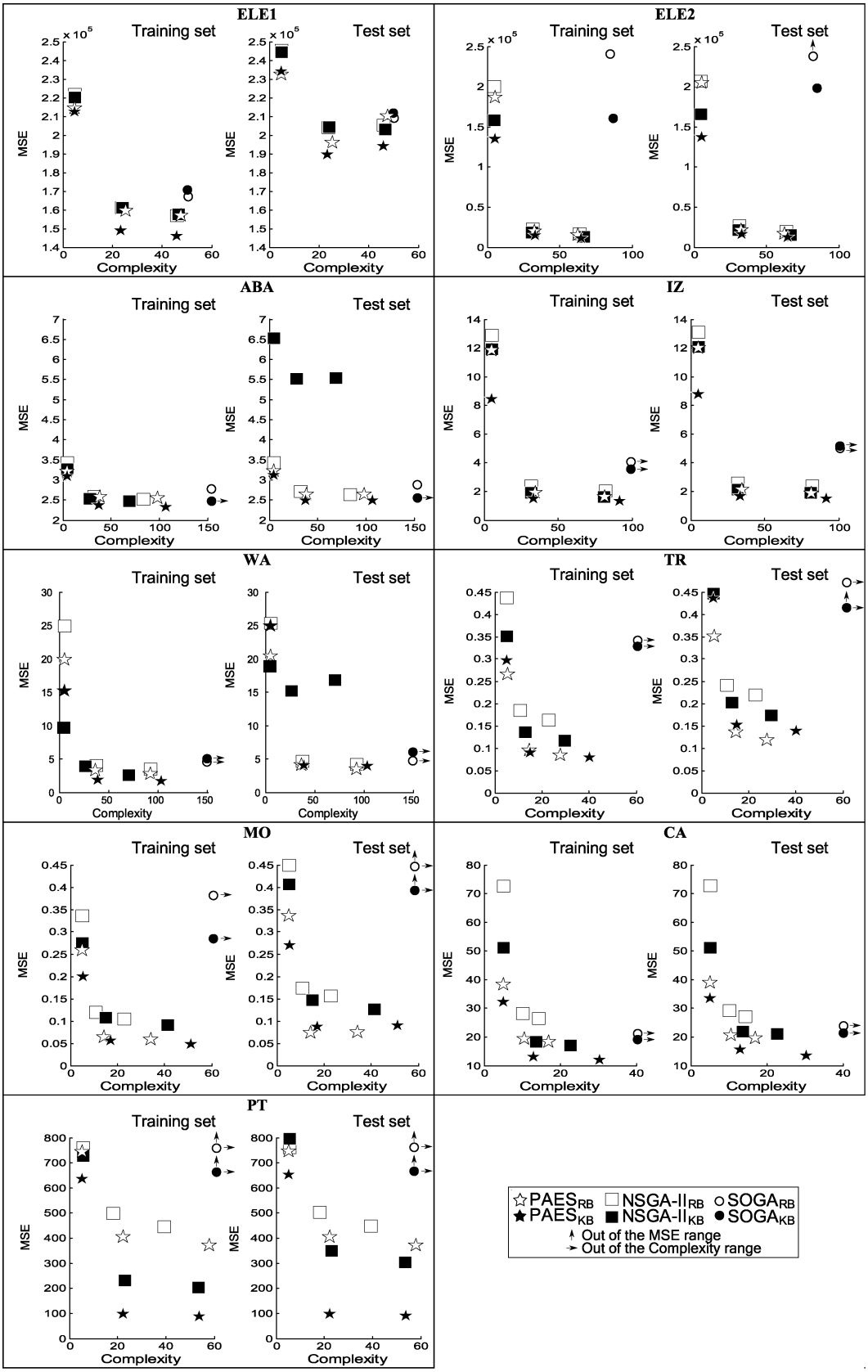

Fig. 8. Plots of the average solutions obtained by the two SOGAs and of the FIRST, MEDIAN, and LAST points obtained by the four MOEAs.

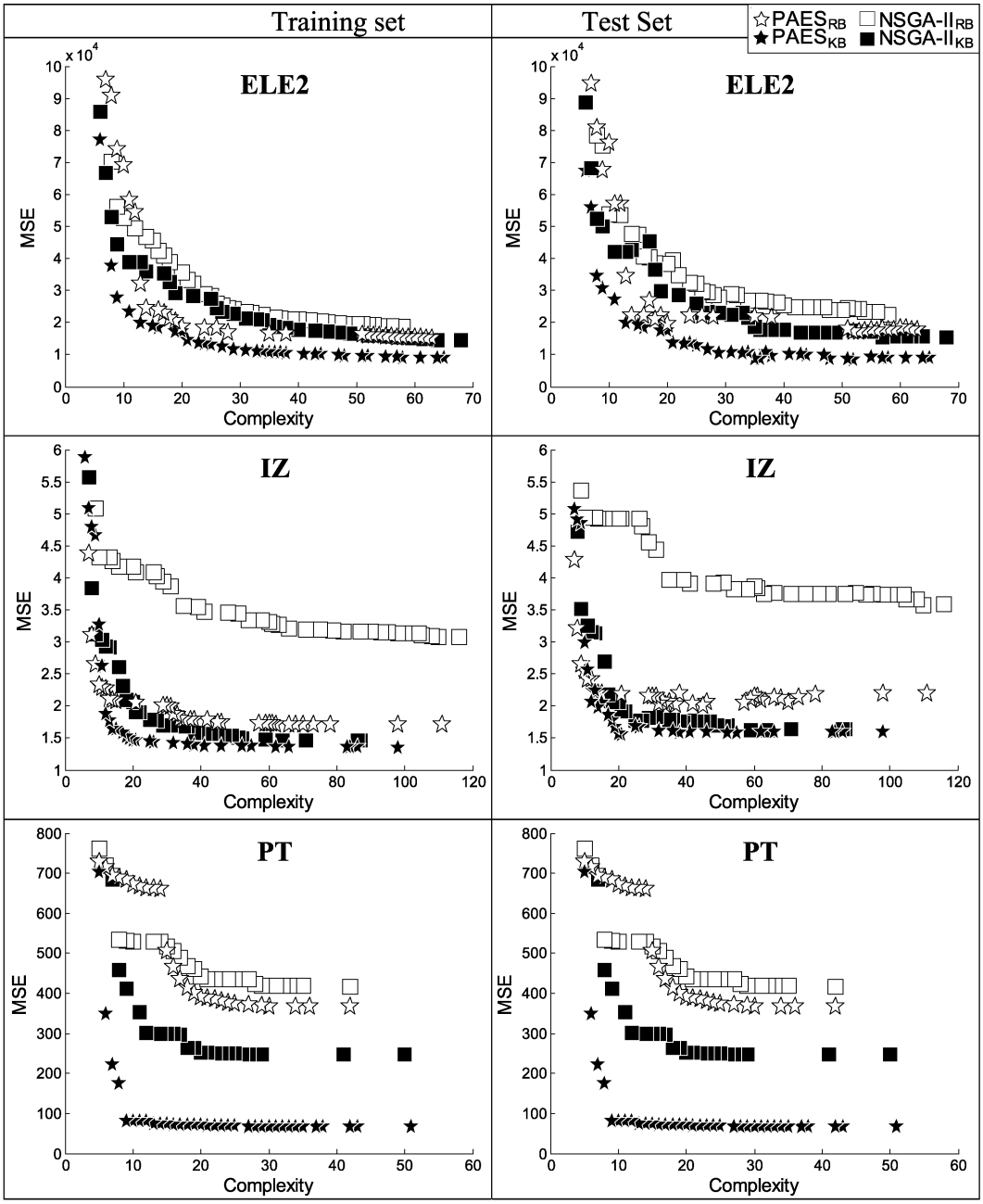

Fig. 9. Examples of the complete Pareto fronts generated by the four MOEAs in the ELE2, IZ, and PT datasets.

(see Table XII). With regard to the LAST point, by considering the results of the Wilcoxon tests, we can say that there exist no statistical differences between PAES_{KB} and PAES_{RB}, not even when we use only seven datasets. On the other hand, we observe that PAES_{KB} outperforms the other MOEAs approaches on seven datasets on the training set and on six datasets on the test set. In particular, PAES_{KB} outperforms PAES_{RB} on CA and PT, which are the datasets with the highest numbers of variables. Thus, even though we are not supported by the results of the statistical test in the LAST point for all datasets, we can still affirm that PAES_{KB} has to be preferred when dealing with datasets with high values of T2. In order to show the actual behavior of the approximated Pareto fronts provided by each MOEA, in Fig. 9, we show some representative Pareto fronts (the results of a single trial) on three datasets both on training and test sets.

F. Computational Costs

In the previous sections, we have compared the different algorithms in terms of the complexity and accuracy of the generated solutions. Actually, it is also interesting to evaluate the computational costs of the algorithms. To provide an idea of the computational cost of the different algorithms, we show the average execution time for each dataset in Table IX. We used a cluster with eight nodes, each one with 8 GB RAM and an Intel Core 2 Quad Q9300 with 2.5 GHz (a total of 32 concurrent

TABLE IX
AVERAGE COMPUTATIONAL COSTS IN MINUTES

Datasets	PAES_{RB}	PAES_{KB}	NSGA-II_{RB}	NSGA-II_{KB}	SOGA_{RB}	SOGA_{KB}
ELE1	4	4	4	4	10	11
ELE2	27	27	25	26	33	36
ABA	126	131	122	129	145	148
IZ	51	55	52	55	71	82
WA	62	69	58	61	92	101
TR	59	61	55	69	134	153
MO	57	64	52	66	121	146
CA	621	634	509	558	1392	1423
PT	907	921	852	881	1591	1672

identical CPUs). As expected, the execution times increase with the increase in the number of points and the number of variables. Further, the execution times required by the MOEAs are comparable with each other. On the other hand, the execution times needed by the SOGAs are longer than the ones required by the MOEAs. These different computational costs between MOEAs and SOGAs are due to the different complexities of the FRBSs generated by the two types of algorithms. Indeed, during the evolution, the complexity of the solutions generated by the MOEAs decreases, thus making the computation of their fitness function faster. By contrast, during the evolution, the solutions generated by the SOGAs always lie in the high-complexity zone, and therefore, the time required for the evaluation of their fitness does not decrease. Since the computational cost of the algorithms is mainly due to the fitness evaluation, this explains the different execution times needed by the MOEAs and the SOGAs. Since PAES_{KB} and PAES_{RB} are comparable from the computational cost point of view, the choice between these two algorithms depends on the type of dataset, as explained in the previous sections.

VI. CONCLUDING REMARKS

In this paper, we have proposed PAES_{KB}, which is an evolutionary multiobjective algorithm developed to generate a set of FRBSs with different tradeoffs between accuracy and complexity, which is measured as the number of rule antecedent conditions. The proposed method is able to learn RBs together with the MF parameters of the associated linguistic labels in the corresponding DB. This allows us to take the existing dependencies into account, but involves a huge search space. In order to better handle the huge search space, we have exploited the positive synergy existing between an interesting RB identification mechanism and the linguistic two-tuple representation model for the learning of MF parameters.

We have shown that a multiobjective framework allows us to obtain FRBSs characterized by better tradeoffs between complexity and accuracy than the ones provided by considering only accuracy as the unique objective. In addition, we have shown that the modified version of the (2 + 2)PAES used in our approach outperforms the well-known NSGA-II when concurrently learning rules and MF parameters.

Finally, we have proposed a categorization of the datasets based on the simple data complexity measure T2, which is defined as the average number of patterns per variable. Results confirm a positive synergy between the learning of rules and the two-tuple-based technique, particularly in (possibly high-dimensional) datasets characterized by high values of T2.

APPENDIX I
WILCOXON SIGNED-RANK TEST

The Wilcoxon signed-rank test is a pairwise test that aims to detect significant differences between two sample means; it is analogous to the paired *t*-test in nonparametric statistical procedures. If these means refer to the outputs of two algorithms, then the test practically assesses the reciprocal behavior of the two algorithms [49], [50]. Let d_i be the difference between the performance scores of the two algorithms on the *i*th out of N_{ds} datasets. The differences are ranked according to their absolute values; average ranks are assigned in case of ties. Let R^+ be the sum of ranks for the datasets on which the first algorithm outperformed the second, and R^- the sum of ranks for the contrary outcome. Ranks of $d_i = 0$ are split evenly among the sums; if there is an odd number of them, then one is ignored:

$$R^+ = \sum_{d_i > 0} \text{rank}(d_i) + \frac{1}{2} \sum_{d_i = 0} \text{rank}(d_i)$$

$$R^- = \sum_{d_i < 0} \text{rank}(d_i) + \frac{1}{2} \sum_{d_i = 0} \text{rank}(d_i).$$

Let T be the smaller of the sums, i.e., $T = \min(R^+, R^-)$. If T is less than, or equal to, the value of the distribution of Wilcoxon for N_{ds} degrees of freedom [52, Tab. B.12], the null hypothesis of equality of means is rejected.

The Wilcoxon signed-rank test is more sensible than the *t*-test. It assumes commensurability of differences, but only qualitatively: Greater differences still count for more, which is probably desired, but the absolute magnitudes are ignored. From the statistical point of view, the test is safer since it does not assume normal distributions. Also, the outliers (exceptionally good/bad performances on a few datasets) have less of an effect on the Wilcoxon test than on the *t*-test. The Wilcoxon test assumes continuous differences d_i; therefore, they should not be rounded to one or two decimals, since this would decrease the test power due to a high number of ties.

When the assumptions of the paired *t*-test are met, the Wilcoxon signed-rank test is less powerful than the paired *t*-test. On the other hand, when the assumptions are violated, the Wilcoxon test can be even more powerful than the *t*-test. This allows us to apply it to the means obtained by the algorithms in each dataset, without any assumption about the distribution of the obtained results.

APPENDIX II
ADDITIONAL TABLES OF RESULTS

This Appendix presents the additional tables of results, i.e., Tables X–XII.

TABLE X
AVERAGE RESULTS ACHIEVED BY THE DIFFERENT MOEAS IN CORRESPONDENCE TO THE MEDIAN POINT

Data set	Measure	NSGA-II$_{RB}$						NSGA-II$_{KB}$						PAES$_{RB}$						PAES$_{KB}$				
NAME (V;SIZE)		R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$
ELE1 (2;495)	Mean	17	23	2	160988	204150	0.40	16	24	2	161089	204341	0.24	17	25	2	159498	195961	0.25	16	23	2	**148845**	**189497**
	SD				5387	38116					5313	36052					4752	36796					4554	22472
ELE2 (4;1056)	Mean	18	31	4	23264	26844	0.07	17	30	3	18284	21373	0.07	19	32	4	19590	21542	0.03	19	33	4	**14714**	**16233**
	SD				5057	6437					4686	6415					4850	5561					4220	4063
ABA (8;4177)	Mean	14	32	7	2.60	2.72	0.09	12	28	7	2.54	5.52	0.55	16	38	8	2.59	2.63	0.06	15	37	8	**2.37**	**2.49**
	SD				0.11	0.26					0.18	7.52					0.11	0.27					0.08	0.17
IZ (9;1462)	Mean	14	32	8	2.40	2.59	0.37	14	32	8	1.89	2.11	0.22	16	34	8.20	1.90	2.10	0.22	15	33	8.31	**1.48**	**1.64**
	SD				0.92	0.98					0.40	0.49					0.66	0.76					0.34	0.34
WA (9;1609)	Mean	17	38	8	3.98	4.60	0.12	12	26	7	3.92	15.16	0.73	17	36	8	3.33	4.11	0.02	16	39	8	**1.88**	**4.04**
	SD				1.24	2.34					4.67	21.96					1.61	2.32					0.49	9.22
TR (15;1049)	Mean	6	11	7	0.18	0.25	0.36	7	13	8	0.13	0.20	0.23	7	14	8	**0.09**	**0.03**	-0.13	8	15	8	**0.09**	0.16
	SD				0.09	0.24					0.08	0.15					0.03	0.12					0.04	0.16
MO (15;1049)	Mean	6	11	7	0.12	0.17	0.63	7	15	9	0.11	0.15	0.57	7	14	8	0.07	**0.07**	-0.18	8	17	9	**0.06**	0.09
	SD				0.07	0.17					0.05	0.13					0.03	0.04					0.03	0.06
CA (21;8192)	Mean	6	10	6	28.14	29.17	0.54	8	13	7	18.29	21.90	0.39	6	10	6	19.37	20.55	0.24	8	13	8	**13.11**	**15.64**
	SD				13.75	13.39					6.77	8.99					4.08	5.03					3.60	6.70
PT (26;15000)	Mean	8	18	12	498	503	0.81	8	23	13	230	349	0.72	11	22	13	404	404	0.76	8	22	14	**97**	**98**
	SD				92	90					192	433					82	86					28	26

TABLE XI
AVERAGE RESULTS ACHIEVED BY THE DIFFERENT MOEAS IN CORRESPONDENCE TO THE LAST POINT

Data set	Measure	NSGA-II$_{RB}$						NSGA-II$_{KB}$						PAES$_{RB}$						PAES$_{KB}$				
NAME (V;SIZE)		R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$	DIFF	R	C	V	MSE$_{Tr}$	MSE$_{Ts}$
ELE1 (2;495)	Mean	5	5	2	221761	245409	0.34	5	5	2	220192	244249	0.17	5	5	2	214215	**232268**	0.33	5	5	2	**212337**	234179
	SD				21543	33519					22524	31260					12291	37800					24264	41825
ELE2 (4;1056)	Mean	5	5	3	199845	206854	0.05	5	5	3	157998	165764	0.04	5	5	3	186424	204952	-0.01	5	5	2	**135026**	**136880**
	SD				43175	61702					66655	60736					59677	76559					65428	58123
ABA (8;4177)	Mean	5	5	5	3.44	3.43	0.09	5	5	4	3.26	6.53	0.52	5	5	3	3.20	3.23	0.03	5	5	3	**3.09**	**3.13**
	SD				0.31	0.31					0.17	8.12					0.24	0.31					0.41	0.48
IZ (9;1462)	Mean	5	5	3	12.90	13.10	0.33	5	5	3	11.91	12.08	0.27	5	5	3	11.83	12.00	0.27	5	5	3	**8.42**	**8.77**
	SD				7.00	7.40					10.98	11.16					6.17	6.56					4.95	5.20
WA (9;1609)	Mean	5	5	3	23.15	25.06	0.01	5	5	3	**9.69**	**18.89**	-0.32	5	5	3	19.87	20.45	-0.22	5	5	3	15.15	24.88
	SD				10.23	10.64					7.03	20.11					9.75	9.98					14.62	44.73
TR (15;1049)	Mean	5	5	3	0.43	0.55	0.23	5	5	4	0.34	0.45	0.07	5	5	3	**0.27**	**0.36**	-0.18	5	5	3	0.28	0.42
	SD				0.32	0.51					0.23	0.38					0.13	0.37					0.21	0.38
MO (15;1049)	Mean	5	5	4	0.34	0.45	0.40	5	5	4	0.28	0.42	0.35	5	5	4	0.26	0.34	0.20	5	5	4	**0.20**	**0.27**
	SD				0.24	0.48					0.16	0.50					0.26	0.42					0.18	0.27
CA (21;8192)	Mean	5	5	4	72.63	72.77	0.54	5	5	4	51.01	51.07	0.35	5	5	4	38.27	38.97	0.14	5	5	4	**32.25**	**33.44**
	SD				82.85	83.50					37.84	34.38					13.50	13.59					16.99	17.30
PT (26;15000)	Mean	5	5	4	760	760	0.14	5	5	5	727	795	0.18	5	5	4	743	745	0.13	5	5	4	**635**	**651**
	SD				73	74					196	274					77	78					177	208

TABLE XII
WILCOXON TEST TO COMPARE PAES$_{\mathrm{RB}}$ AND PAES$_{\mathrm{KB}}$ IN MEDIAN AND LAST POINTS

Comparison	R^+	R^-	Hypothesis ($\alpha = 0.1$)	p-value
PAES$_{RB}$ vs. PAES$_{KB}$ (MEDIAN)	9	36	Accepted	0.110
PAES$_{RB}$ vs. PAES$_{KB}$ (LAST)	13	32	Accepted	0.260
By only considering the seven data sets with the highest T2 values				
PAES$_{RB}$ vs. PAES$_{KB}$ (MEDIAN)	0	28	Rejected for PAES$_{KB}$	0.018
PAES$_{RB}$ vs. PAES$_{KB}$ (LAST)	6	22	Accepted	0.176

REFERENCES

[1] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. New York: Addison-Wesley, 1989.

[2] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. of Michigan Press, 1975 (2nd ed., Cambridge, MA: MIT Press, 1992).

[3] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena, *Genetic Fuzzy Systems. Evolutionary Tuning and Learning of Fuzzy Knowledge Base* (Advances in Fuzzy Systems—Applications and Theory), vol. 19. Singapore: World Scientific, 2001.

[4] O. Cordón, F. Gomide, F. Herrera, F. Hoffmann, and L. Magdalena, "Ten years of genetic fuzzy systems: Current framework and new trends," *Fuzzy Sets Syst.*, vol. 41, no. 1, pp. 5–31, 2004.

[5] F. Herrera, "Genetic fuzzy systems: Taxonomy, current research trends and prospects," *Evol. Intell.*, vol. 1, pp. 27–46, 2008.

[6] J. Casillas, O. Cordón, F. Herrera, and L. Magdalena, Eds., *Interpretability Issues in Fuzzy Modeling*. New York: Springer-Verlag, 2003.

[7] C. A. Coello, D. A. V. Veldhuizen, and G. B. Lamont, Eds., *Evolutionary Algorithms for Solving Multi-Objective Problems*. Norwell, MA: Kluwer, 2002.

[8] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. New York: Wiley, 2001.

[9] Y. Jin Ed., *Multi-Objective Machine Learning*. Berlin, Germany: Springer-Verlag, 2006.

[10] Y. Jin and B. Sendhoff, "Pareto-based multi-objective machine learning: An overview and case studies," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 38, no. 3, pp. 397–415, May 2008.

[11] O. Cordon, F. Herrera, M. J. del Jesus, and P. Villar, "A multiobjective genetic algorithm for feature selection and granularity learning in fuzzy-rule based classification systems," in *Proc. 9th IFSA World Congr. XX NAFIPS Int. Conf.*, Vancouver, BC, Canada, 2001, pp. 1253–1258.

[12] H. Ishibuchi, T. Murata, and I. B. Turksen, "Selecting linguistic classification rules by two-objective genetic algorithms," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Vancouver, BC, Canada, 1995, pp. 1410–1415.

[13] H. Ishibuchi, T. Murata, and I. B. Türksen, "Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems," *Fuzzy Sets Syst.*, vol. 89, no. 2, pp. 135–150, 1997.

[14] H. Ishibuchi, T. Murata, and M. Gen, "Performance evaluation of fuzzy rule-based classification systems obtained by multi-objective genetic algorithms," *Comput. Ind. Eng.*, vol. 35, no. 3/4, pp. 575–578, 1998.

[15] H. Ishibuchi, T. Nakashima, and T. Murata, "Three-objective genetics-based machine learning for linguistic rule extraction," *Inf. Sci.*, vol. 136, pp. 109–133, 2001.

[16] H. Ishibuchi, T. Nakashima, and T. Murata, "Multiobjective optimization in linguistic rule extraction from numerical data," in *Proc. 1st Int. Conf. Evol. Multi-Criterion Optim.* (Lecture Notes in Computer Science), Zurich, Switzerland, 2001, vol. 1993, pp. 588–602.

[17] H. Ishibuchi and T. Yamamoto, "Interpretability issues in fuzzy genetics-based machine learning for linguistic modelling," in *Modelling With Words: Learning, Fusion, and Reasoning Within a Formal Liguistic Representation Framework* (Lecture Notes in Computer Science), vol. 2873, J. Lawry, J. G. Shanahan, and A. L. Ralescu, Eds. Berlin, Germany: Springer-Verlag, 2003, pp. 209–228.

[18] H. Ishibuchi and T. Yamamoto, "Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining," *Fuzzy Sets Syst.*, vol. 141, no. 1, pp. 59–88, 2004.

[19] H. Ishibuchi and Y. Nojima, "Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning," *Int. J. Approx. Reason.*, vol. 44, no. 1, pp. 4–31, 2007.

[20] K. Narukawa, Y. Nojima, and H. Ishibuchi, "Modification of evolutionary multiobjective optimization algorithms for multiobjective design of fuzzy rule-based classification systems," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Reno, NV, 2005, pp. 809–814.

[21] P. Pulkkinen and H. Koivisto, "Fuzzy classifier identification using decision tree and multiobjective evolutionary algorithms," *Int. J. Approx. Reason.*, vol. 48, no. 2, pp. 526–543, 2008.

[22] P. Pulkkinen, J. Hytönen, and H. Koivisto, "Developing a bioaerosol detector using hybrid genetic fuzzy systems," *Eng. Appl. Artif. Intell.*, vol. 21, no. 8, pp. 1330–1346, 2008.

[23] K. Deb, S. Agrawal, A. Pratab, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[24] M. Cococcioni, P. Ducange, B. Lazzerini, and F. Marcelloni, "A Pareto-based multi-objective evolutionary approach to the identification of Mamdani fuzzy systems," *Soft Comput.*, vol. 11, pp. 1013–1031, 2007.

[25] J. D. Knowles and D. W. Corne, "Approximating the non dominated front using the Pareto archived evolution strategy," *Evol. Comput.*, vol. 8, no. 2, pp. 149–172, 2000.

[26] R. Alcalá, M. J. Gacto, F. Herrera, and J. Alcalá-Fdez, "A multi-objective genetic algorithm for tuning and rule selection to obtain accurate and compact linguistic fuzzy rule-based systems," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 15, no. 5, pp. 539–557, 2007.

[27] M. J. Gacto, R. Alcalá, and F. Herrera, "Adaptation and application of multi-objective evolutionary algorithms for rule reduction and parameter tuning of fuzzy rule-based systems," *Soft Comput.*, vol. 13, no. 5, pp. 419–436, 2009.

[28] A. Botta, B. Lazzerini, F. Marcelloni, and D. C. Stefanescu, "Context adaptation of fuzzy systems through a multi-objective evolutionary approach based on a novel interpretability index," *Soft Comput.*, vol. 13, no. 5, pp. 437–449, 2009.

[29] R. Alcalá, J. Alcalá-Fdez, and F. Herrera, "A proposal for the genetic lateral tuning of linguistic fuzzy systems and its interaction with rule selection," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 4, pp. 616–635, Aug. 2007.

[30] F. Herrera and L. Martínez, "A 2-tuple fuzzy linguistic representation model for computing with words," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 6, pp. 746–752, Dec. 2000.

[31] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, 2006.

[32] S. García and F. Herrera, "An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2596, 2008.

[33] S. García, A. Fernández, J. Luengo, and F. Herrera, "A study of statistical techniques and performance measures for genetics-based machine learning: Accuracy and interpretability," *Soft Comput.*, vol. 13, no. 10, pp. 959–977, 2009.

[34] S. García, D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization," *J. Heuristics*, to be published. DOI: 10.1007/s10732-008-9080-4.

[35] T. K. Ho and M. Basu, "Complexity measures of supervised classification problems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 289–300, Mar. 2002.

[36] M. Basu and T. K. Ho Eds., *Data Complexity in Pattern Recognitio* (Advanced Information and Knowledge Processing). vol. XVI, Berlin, Germany: Springer-Verlag, 2006.

[37] H. Ishibuchi and T. Murata, "Multi-objective genetic local search algorithm," in *Proc. 3rd IEEE Int. Conf. Evol. Comput.*, 1996, pp. 119–124.

[38] F. Jimenez, A. F. Gomez-Skarmeta, H. Roubos, and R. Babuska, "Accurate, transparent, and compact fuzzy models for function approximation and dynamic modeling through multi-objective evolutionary optimization," in *Proc. 1st Int. Conf. Evol. Multi-Criterion Optim.* (Lecture Notes in Computer Science), Zurich, Switzerland, 2001, vol. 1993, pp. 653–667.

[39] F. Jimenez, A. F. Gomez-Skarmeta, H. Roubos, and R. Babuska, "A multi-objective evolutionary algorithm for fuzzy modelling," in *Proc. 9th IFSA World Congr. 20th NAFIPS Int. Conf.*, Vancouver, BC, Canada, 2001, pp. 1222–1228.

[40] H. L. Wang, S. Kwong, Y. C. Jin, W. Wei, and K. F. Man, "Multi-objective hierarchical genetic algorithm for interpretable fuzzy rule-based knowledge extraction," *Fuzzy Sets Syst.*, vol. 149, no. 1, pp. 149–186, 2005.

[41] H. L. Wang, S. Kwong, Y. C. Jin, W. Wei, and K. F. Man, "Agent based evolutionary approach for interpretable rule-based knowledge extraction," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 35, no. 2, pp. 143–155, May 2005.

[42] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Feb. 1985.

[43] E. H. Mamdani, "Application of fuzzy algorithms for control of simple dynamic plant," *Proc. IEEE*, vol. 121, no. 12, pp. 1585–1588, Dec. 1974.

[44] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *Int. J. Man-Mach. Stud.*, vol. 7, pp. 1–13, 1975.

[45] R. Alcalá, J. Alcalá-Fdez, J. Casillas, O. Cordón, and F. Herrera, "Hybrid learning models to get the interpretability-accuracy trade off in fuzzy modeling," *Soft Comput.*, vol. 10, no. 9, pp. 717–734, 2006.

[46] L. J. Eshelman and J. D. Schaffer, "Real-coded genetic algorithms and interval-schemata," *Found. Genetic Algorithms*, vol. 2, pp. 187–202, 1993.

[47] F. Herrera, M. Lozano, and A. M. Sánchez, "A taxonomy for the crossover operator for real-coded genetic algorithms: An experimental study," *Int. J. Intell. Syst.*, vol. 18, pp. 309–338, 2003.

[48] J. Alcalá-Fdez, L. Sánchez, S. García, M. del Jesus, S. Ventura, J. Garrell, J. Otero, C. Romero, J. Bacardit, V. Rivas, J. Fernández, and F. Herrera, "KEEL: A software tool to assess evolutionary algorithms to data mining problems," *Soft Comput.*, vol. 13, no. 3, pp. 307–318, 2009.

[49] D. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*. London, U.K./Boca Raton, FL: Chapman & Hall/CRC, 2003.

[50] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics*, vol. 1, pp. 80–83, 1945.

[51] O. Cordón, F. Herrera, and L. Sánchez, "Solving electrical distribution problems using hybrid evolutionary data analysis techniques," *Appl. Intell.*, vol. 10, pp. 5–24, 1999.

[52] J. Zar, *Biostatistical Analysis*. Upper Saddle River, NJ: Prentice–Hall, 1999.

Rafael Alcalá received the M.Sc. and Ph.D. degrees in computer science from the University of Granada, Granada, Spain, in 1998 and 2003, respectively.

From 1998 to 2003, he was with the Department of Computer Science, University of Jaén, Spain. He is currently an Assistant Professor with the Department of Computer Science and Artificial Intelligence, University of Granada, where he is also a Member of the Soft Computing and Intelligent Information Systems Research Group. He has authored or coauthored more than 60 papers in international journals, book chapters, and conferences. He has worked on several research projects supported by the Spanish government and the European Union. He is currently a member of the Editorial/Reviewer Boards of the journals *International Journal of Computational Intelligence Research*, *Journal of Advanced Research in Fuzzy and Uncertain Systems*, and *Journal of Universal Computer Science*. His current research interests include multiobjective genetic algorithms and genetic fuzzy systems, particularly the learning/tuning of fuzzy systems for modeling and control with a good tradeoff between accuracy and interpretability, as well as fuzzy association rules.

Dr. Alcalá coedited the IEEE TRANSACTIONS ON FUZZY SYSTEMS Special Issue entitled "Genetic Fuzzy Systems: What's Next."

Pietro Ducange received the M.Sc. degree in computer systems engineering and the Ph.D. degree in information engineering from the University of Pisa, Pisa, Italy, in 2005 and 2009, respectively.

He is currently a Research Fellow with the Department of Information Engineering, University of Pisa. His current research interests include multiobjective evolutionary algorithms, genetic fuzzy systems, and neural network design with applications to regression problems and pattern recognition.

Francisco Herrera received the M.Sc. and Ph.D. degrees in mathematics from the University of Granada, Granada, Spain, in 1988 and 1991, respectively.

He is currently a Professor with the Department of Computer Science and Artificial Intelligence, University of Granada. He has authored or coauthored more than 140 papers in international journals. He is the coauthor of the book *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases* (World Scientific, 2001). He has coedited four international books and 20 special issues in international journals on different soft computing topics. He is an Associate Editor of the journals *Mathware and Soft Computing*, *Advances in Fuzzy Systems*, *Advances in Computational Sciences and Technology*, and the *International Journal of Applied Metaheuristic Computing*. He is currently an Area Editor of the journal *Soft Computing* (area of genetic algorithms and genetic fuzzy systems). He is a member of the Editorial Boards of the journals *Fuzzy Sets and Systems*, *Applied Intelligence*, *Knowledge and Information Systems*, *Information Fusion*, *Evolutionary Intelligence*, the *International Journal of Hybrid Intelligent Systems*, *Memetic Computation*, the *International Journal of Computational Intelligence Research*, *The Open Cybernetics and Systemics Journal*, *Recent Patents on Computer Science*, the *Journal of Advanced Research in Fuzzy and Uncertain Systems*, and the *International Journal of Information Technology and Intelligent Computing*. His current research interests include computing with words and decision making, data mining, data preparation, instance selection, fuzzy-rule-based systems, genetic fuzzy systems, knowledge extraction based on evolutionary algorithms, memetic algorithms, and genetic algorithms.

Prof. Herrera is an as Associate Editor of the journal IEEE TRANSACTIONS ON FUZZY SYSTEMS.

Beatrice Lazzerini (M'98) received the Laurea degree in computer science from the University of Pisa, Pisa, Italy, and the Laurea degree in computer engineering from the University of Florence, Florence, Italy.

She is a Full Professor with the Faculty of Engineering, University of Pisa, Pisa, Italy. She teaches intelligent systems and decision support intelligent systems. She is the President of the Specialized Laurea degree in "computer engineering for enterprise management." Her current research interests include the area of computational intelligence and its applications to engineering. She has coauthored seven books and has authored or coauthored more than 140 papers in international journals and conferences. She is the coeditor of two books. She has been involved in several national and international research projects.

Francesco Marcelloni (M'06) received the Laurea degree in electronics engineering and the Ph.D. degree in computer engineering from the University of Pisa, Pisa, Italy, in 1991 and 1996, respectively.

From 1994 to 1995 and from June to December 1997, he was a Visiting Researcher with the University of Twente, Enschede, The Netherlands, where he joined the Twente Research and Education on Software Engineering project. He is currently an Associate Professor with the Faculty of Engineering, University of Pisa, where he cofounded the Computational Intelligence Group and has founded and currently heads the Mobile Value-Added Services Centre. His current research interests include multiobjective genetic algorithms, genetic fuzzy systems, fuzzy clustering algorithms, pattern recognition, neural networks, mobile information systems, and data compression and aggregation in wireless sensor networks. He has authored or coauthored more than 120 papers in international journals, book chapters, a book, and conference presentations.