NMEEF-SD: Non-dominated Multiobjective Evolutionary Algorithm for Extracting Fuzzy Rules in Subgroup Discovery

Cristóbal José Carmona, Pedro González, María José del Jesus, and Francisco Herrera, *Member, IEEE*

***Abstract*—A non-dominated multiobjective evolutionary algorithm for extracting fuzzy rules in subgroup discovery (NMEEF-SD) is described and analyzed in this paper. This algorithm, which is based on the hybridization between fuzzy logic and genetic algorithms, deals with subgroup-discovery problems in order to extract novel and interpretable fuzzy rules of interest, and the evolutionary fuzzy system NMEEF-SD is based on the well-known Non-dominated Sorting Genetic Algorithm II (NSGA-II) model but is oriented toward the subgroup-discovery task using specific operators to promote the extraction of interpretable and high-quality subgroup-discovery rules. The proposal includes different mechanisms to improve diversity in the population and permits the use of different combinations of quality measures in the evolutionary process. An elaborate experimental study, which was reinforced by the use of nonparametric tests, was performed to verify the validity of the proposal, and the proposal was compared with other subgroup discovery methods. The results show that NMEEF-SD obtains the best results among several algorithms studied.**

***Index Terms*—Descriptive rule induction, fuzzy rules, genetic fuzzy system, multiobjective evolutionary algorithm, subgroup discovery.**

I. Introduction

DATA mining displays supervised as well as nonsupervised learning approaches. Generally, supervised learning methods have a predictive nature, while nonsupervised ones have a descriptive nature. Currently, several techniques are located halfway between descriptive and predictive data mining, such as subgroup discovery (SD) [1], contrast-set mining [2], and emerging-pattern mining [3], which has aroused the interest of researchers. These techniques are known as "Supervised Descriptive Rule Induction" [4] because they combine the features of both types of induction and their main objective is to extract from the data descriptive knowledge that concerns a property of interest.

Manuscript received August 20, 2009; revised December 19, 2009 and April 22, 2010; accepted June 25, 2010. Date of publication July 19, 2010; date of current version September 29, 2010. This work was supported in part by the Spanish Ministry of Education, Social Policy, and Sports under Project TIN-2008-06681-C06-01 and Project TIN-2008-06681-C06-02 and in part by the Andalusian Research Plan under Project TIC-3928.

C. J. Carmona, P. González, and M. J. del Jesus are with the Department of Computer Science, University of Jaen, 23071 Jaen, Spain (e-mail: ccarmona@ujaen.es; pglez@ujaen.es; mjjesus@ujaen.es).

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

Digital Object Identifier 10.1109/TFUZZ.2010.2060200

This paper focuses on SD, which is a form of supervised inductive learning of subgroup descriptions in which, given a set of data and having a property of interest to the user, the algorithm attempts to locate subgroups that are "most interesting" for the user. SD has the objective to discover properties of interest to subgroups and, in the process, obtain simple rules, with high generality, accuracy, and interest. Nowadays, SD is under application to problems in a variety of fields such as medicine [5], [6], marketing [7], and e-learning [8].

In recent years, new algorithms for SD have been developed using soft-computing techniques such as fuzzy rules [9] and genetic algorithms (GAs) [10]. The conjunction of these techniques is called genetic fuzzy systems (GFSs) [11], [12], which has triggered considerable attention in the computational intelligence community. Several useful tools have been provided for the SD task; see, for instance, the *KEEL Data Mining tool* [13].

The induction of rules describing subgroups can be considered a multiobjective problem, since there are different quality measures which can be used for the evaluation of an SD rule. Therefore, multiobjective evolutionary algorithms (MOEAs) are adapted to solve problems in which different objectives must be optimized [14], [15]. In particular, Non-dominated Sorting Genetic Algorithm II (NSGA-II) [16] is a high-quality exponent of this type of algorithm that is widely used in GFSs [17].

This paper describes a proposal based on the NSGA-II approach for the induction of fuzzy rules that describe subgroups: the non-dominated multiobjective evolutionary algorithm for extracting fuzzy rules in subgroup discovery (NMEEF-SD). As a novelty, this algorithm permits the selection of different combinations of quality measures as objectives in the evolutionary process and introduces an operator to promote diversity in the process.

In order to verify the validity of the model presented, an elaborate experimental study of SD was performed for the evolutionary SD algorithms NMEEF-SD, Subgroup Discovery Iterative Genetic Algorithm (SDIGA) [7], and Multi-objective Evolutionary Subgroup DIscovery Fuzzy rules (MESDIF) [18], as well as the classical SD algorithms CN2-SD [1] and Apriori-SD [19]. These studies were reinforced by the use of nonparametric tests for comparison and show good results in the quality measures analysis and in the interpretable analysis obtained by NMEEF-SD. Furthermore, an analysis of scalability and time complexity is performed among NMEEF-SD, CN2-SD, and Apriori-SD.

The paper is organized as follows: Section II provides a short description of SD, the quality measures used, and a presentation of the GFS for SD. Section III describes the proposed

NMEEF-SD algorithm. Section IV discusses the tests conducted on the datasets for the compared algorithms. Section IV-A shows the experimental framework. Section IV-B contains the study with different combinations of quality measures for NMEEF-SD. Section IV-C includes a study of the evolutionary algorithms for SD. Section IV-D shows the behavior of the NMEEF-SD algorithm versus classical algorithms for SD. Section IV-E presents an analysis of time complexity and scalability. Section IV-F displays different rules obtained. Finally, some concluding remarks are outlined in Section V.

II. Subgroup Discovery

The SD task was initially formulated by Klösgen [20] and Wrobel [21] and is defined as follows: "Given a population of individuals and a property of those individuals we are interested in, find population subgroups that are statistically 'most interesting,' i.e. are as large as possible and have the most unusual statistical characteristics with respect to the property of interest."

The objective of any SD algorithm is to discover properties of interest of subgroups by obtaining simple rules (i.e., with an understandable structure), which are highly significant and have high support (i.e., covering many of the instances of the objective property).

An induced subgroup can be represented as

$$R : \mathrm{Cond} \rightarrow \mathrm{Class} \tag{1}$$

where the property of interest for SD is the class value Class that appears in the consequent part of the rule, and the antecedent part Cond is a conjunction of features (attribute-value pairs) selected from the features describing the training instances [22].

Different classical proposals on SD algorithms can be found: EXPLORA [20] was the first approach for SD; Multi-relational Discovery Of Subgroups (MIDOS) [21], which applies the EXPLORA approach to multirelational databases; Apriori-SD [19], which was developed by adapting to SD the classification-rule-learning algorithm Apriori-C [23], which is a modification of the original Apriori association-rule-learning algorithm [24]; CN2-SD [1], which is based on the CN2 classification-rule algorithm [25]; Relational Subgroup Discovery (RSD) [26], which is an upgrade of the CN2-SD algorithm which enables relational SD; or SD-MAP [27], which is an exhaustive SD algorithm based upon the FP-growth algorithm [28] for mining association.

A. Quality Measures for Subgroup Discovery

The quality measures used in an SD algorithm are the key factors for obtaining accurate, simple, and interpretable rules. Different studies cater to objective quality measures applied to the descriptive induction process [22], [29], but it is difficult to reach an agreement on their use. The most used quality measures in the SD literature, which, therefore, are considered in this study, are the following:

- *Number of rules (n_r)*: a complexity measure computed as the number of induced rules;
- *Number of variables (n_v)*: the number of variables of the antecedent. The number of variables for a set of rules is computed as the average of the variables for each rule of that set.
- *Support of a rule*: the frequency of correctly classified examples covered by the rule [1]. It can be computed as

$$\begin{aligned}\mathrm{Sup}(R_i) &= \mathrm{Sup}(\mathrm{Cond}_i \rightarrow \mathrm{Class}_j) \\ &= \frac{n(\mathrm{Class}_j \cdot \mathrm{Cond}_i)}{n_s}\end{aligned} \tag{2}$$

where a rule R_i $n(\mathrm{Class}_j \cdot \mathrm{Cond}_i)$ is the number of examples that satisfy the conditions and belong to the value for the target variable, and n_s is the number of examples. Another way to measure support is by considering the *Support based on examples of the class*, which is defined as the degree of coverage that the rule offers to examples of that class [7]

$$\begin{aligned}\mathrm{Sup}_c(R_i) &= \mathrm{Sup}_c(\mathrm{Cond}_i \rightarrow \mathrm{Class}_j) \\ &= \frac{n(\mathrm{Class}_j \cdot \mathrm{Cond}_i)}{n(\mathrm{Class}_j)}\end{aligned} \tag{3}$$

where $n(\mathrm{Class}_j)$ is the number of examples of the class. In this paper, we use this expression for the support measure. The support for a set of rules is computed as

$$\mathrm{SUP}_c = \frac{1}{n_r} \cdot \sum_{i=1}^{n_r} \mathrm{Sup}_c(R_i). \tag{4}$$

- *Confidence of a rule*: a standard measure that determines the relative frequency of examples that satisfy the complete rule among those that satisfy only the antecedent. It can be computed with different expressions, for example [30]

$$\begin{aligned}\mathrm{Cnf}(R_i) &= \mathrm{Cnf}(\mathrm{Cond}_i \rightarrow \mathrm{Class}_j) \\ &= \frac{n(\mathrm{Class}_j \cdot \mathrm{Cond}_i)}{n(\mathrm{Cond}_i)}\end{aligned} \tag{5}$$

where $n(\mathrm{Cond}_i)$ is the number of examples that verify the condition.

The confidence for a set of rules is computed as

$$\mathrm{CNF} = \frac{1}{n_r} \cdot \sum_{i=1}^{n_r} \mathrm{Cnf}(R_i). \tag{6}$$

- *Significance of a rule*: the significance of a finding, if measured by the likelihood ratio of a rule [20]

$$\begin{aligned}\mathrm{Sig}(R_i) &= \mathrm{Sig}(\mathrm{Cond}_i \rightarrow \mathrm{Class}_j) \\ &= 2 \cdot \sum_{k=1}^{n_c} n(\mathrm{Class}_k \cdot \mathrm{Cond}_i) \\ &\quad \cdot \log \frac{n(\mathrm{Class}_k \cdot \mathrm{Cond}_i)}{n(\mathrm{Class}_k) \cdot p(\mathrm{Cond}_i)}\end{aligned} \tag{7}$$

where $p(\mathrm{Cond}_i)$, which is computed as $n(\mathrm{Cond}_i)/n_s$, is used as a normalized factor, and n_c is the number of classes. It must be noted that, although each rule stands for a specific class value, the significance measures the novelty in the distribution impartially for all the class values.

The significance for a set of rules is computed as

$$\mathrm{SIG} = \frac{1}{n_r} \cdot \sum_{i=1}^{n_r} \mathrm{Sig}(R_i). \tag{8}$$

- *Unusualness of a rule*: the weighted relative accuracy of a rule [31]

$$\begin{aligned} &\mathrm{WRAcc}(R_i) \\ &= \mathrm{WRAcc}(\mathrm{Cond}_i \rightarrow \mathrm{Class}_j) \\ &= \frac{n(\mathrm{Cond}_i)}{n_s} \left(\frac{n(\mathrm{Class}_j \cdot \mathrm{Cond}_i)}{n(\mathrm{Cond}_i)} - \frac{n(\mathrm{Class}_j)}{n_s} \right). \end{aligned} \tag{9}$$

The weighted relative accuracy of a rule can be described as the balance between the coverage of the rule $p(\mathrm{Cond}_i)$ and its accuracy gain $p(\mathrm{Class}_j.\mathrm{Cond}_i) - p(\mathrm{Class}_j)$.

The unusualness for a set of rules is computed as

$$\mathrm{WRAcc} = \frac{1}{n_r} \cdot \sum_{i=1}^{n_r} \mathrm{WRAcc}(R_i). \tag{10}$$

B. GFSs for Subgroup Discovery

A GFS is basically a fuzzy system augmented by a learning process based on evolutionary computation, which includes GAs, genetic programming, and evolutionary strategies, among other evolutionary algorithms [32]. Fuzzy systems are one of the most important areas for the application of the fuzzy set theory [33], [34]. Usually, these kinds of systems consider a model structure in the form of fuzzy rules. They are called fuzzy-rule-based systems (FRBSs), which have demonstrated their ability with respect to different problems like control problems, modeling, classification, or data mining in a large number of applications. The pioneering works in application of FRBSs to these types of problems can be found in [35]–[38], respectively. FRBSs provide us a comprehensible representation of the extracted knowledge and, moreover, a suitable tool for processing the continuous variables.

The first step in designing a GFS is to decide which parts of the fuzzy system are subjected to optimization for coding the problem solution into chromosomes. GFS approaches can be divided into two processes: tuning and learning. In SD, learning processes must be used due to the fact that interpretability is a crucial issue, and with tuning processes, it could be decreased. Many of the approaches for automatic learning focus on the extraction of descriptive rules for data mining; for instance, in [39]–[41], some of the most cited papers based on the extraction of association rules can be found, and in [42]–[48], some of the latest approaches presented for association fuzzy rules can be observed, or an approach for SD can be found in [7].

Within the task of learning rules in an FRBS, two approaches are used in order to encode the individuals of the population [11]: the "*Chromosome = Set of rules*" approach, also called the Pittsburgh approach, in which each individual represents a set of rules; and the "*Chromosome = Rule*" approach, in which each individual codifies a single rule, and the whole rule set is provided by combining several individuals in the population. Within this approach, there are three generic proposals: Michigan [49], iterative rule learning (IRL) [50], and the "cooperative–competitive" approach [51].

Currently, GFSs are being applied in real-world applications to solve complex problems. The research in this area is growing and a number of open problems and future directions can be found in [52]–[55] and in the review in [12].

To our knowledge, there are few previous evolutionary proposals in literature for extracting fuzzy rules for SD.

- SDIGA [7] is a GA which follows the iterative rule-learning approach in which each chromosome represents a rule, but the GA solution is the best individual, obtained and the global solution is formed by the best individuals obtained when the algorithm is run multiple times. SDIGA considers linguistic fuzzy rules and uses support, confidence, and interest as quality measures.
- MESDIF [56] is a multiobjective evolutionary algorithm for SD based on the SPEA2 approach [57], which applies the concepts of elitism in rule selection (using a secondary or elite population) and optimal solution search in the Pareto front. In order to preserve diversity at a phenotypic level, the algorithm uses a niching technique that considers the proximity in values of the objectives and an additional objective based on novelty to promote rules that give information on examples not described by other rules of the population. It considers linguistic fuzzy rules and uses support and confidence as quality measures.

III. Non-dominated Multiobjective Evolutionary Algorithm for Extracting Fuzzy Rules in Subgroup Discovery

NMEEF-SD is a GFS whose objective is to extract descriptive fuzzy and/or crisp rules for the SD task, depending on the type of variables present in the problem.

There are several quality measures currently utilized for SD, but as mentioned previously, there is no consensus on which are the most suitable. The idea of this proposal is to include some quality measures in order to obtain rules with suitable values not only in the selected quality measures but in the others as well. The best way to obtain solutions with a good compromise between the quality measures for SD is through an MOEA approach. In this sense, NMEEF-SD has a multiobjective approach based on NSGA-II [16], which is a computationally fast MOEA based on a non-dominated sorting approach, and on the use of elitism. The proposed algorithm is oriented toward SD and uses specific operators to promote the extraction of simple, interpretable, and high-quality SD rules. The proposal permits a number of quality measures to be used both for the selection and the evaluation of rules within the evolutionary process.

As the general objective of NMEEF-SD is to obtain a set of rules, which should be general and accurate, the algorithm includes components which enhance these characteristics. In particular, diversity is enhanced in the population by the use of a new operator to perform a re-initialization based on coverage, in addition to a niching technique (the crowding distance in the selection operator). To promote generalization, as well

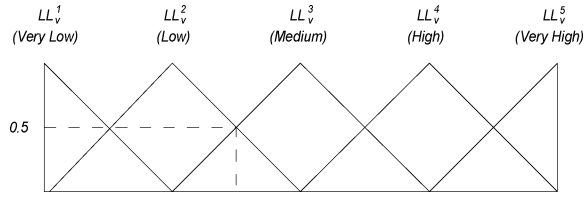


Fig. 1. Example of fuzzy partition for a continuous variable with five labels.

as the objectives considered in the evolutionary approach, the algorithm includes operators of biased initialization and biased mutation. Finally, to ensure accuracy in addition to the objectives, NMEEF-SD returns as its final solution those rules which reach a predetermined confidence threshold.

NMEEF-SD has advantages over other existing algorithms for SD, which are confirmed by experiments (shown in Section IV-A) on a large set of databases. With respect to the classical (not evolutionary) algorithms developed thus far, NMEEF-SD permits the use of numerical features without the need for a previous discretization; it also uses fuzzy rules, which contribute to the interpretability of the extracted rules since it uses a knowledge representation close to the expert. With respect to the other evolutionary algorithms, NMEEF-SD offers more flexibility through the ability to select different quality measures and a powerful multiobjective evolutionary model based on the well-known NSGA-II algorithm.

Next, the structure and representation of the rules used in NMEEF-SD are depicted in Section III-A. Afterward, the quality measures considered as objectives in the algorithm are explained in Section III-B. Furthermore, the operation of the algorithm and its components are outlined in Section III-C. Finally, in Section III-D, a comparison between the evolutionary fuzzy algorithms for SD can be observed.

A. Rule Structure and Representation in NMEEF-SD

With respect to the rule structure, NMEEF-SD uses fuzzy logic to represent the continuous variables by means of linguistic variables that allow us in data mining processes to use numerical features without the need to increase the interpretability of the extracted knowledge by their discretization. The continuous variable are considered linguistic, and the fuzzy sets corresponding to the linguistic labels can be specified by the user or defined by means of a uniform partition if the expert knowledge is not available. In this paper, uniform partitions with triangular membership functions are used, as shown in Fig. 1, for a variable m with five linguistic labels: $X_m : \{LL_m^1, LL_m^2, \ldots, LL_m^5\}$.

In NMEEF-SD, a fuzzy rule describing a subgroup is represented as

$$R : \text{If } X_1 \text{ is } \mathrm{LL}_1^2 \text{ and } X_7 \text{ is } \mathrm{LL}_7^1 \text{ then Class}_j \tag{11}$$

considering the following points.

- $\{X_m / m = 1, \ldots, n_v\}$ is a set of features used to describe the subgroups, where n_v is the number of features. These variables can be categorical or numerical.
- $\{\mathrm{Class}_j / j = 1, \ldots, n_c\}$ is a set of values for the target variable, where n_c is the number of values.
- $\{E^k = (e_1^k, e_2^k, \ldots, e_{n_v}^k) / k = 1, \ldots, n_s\}$ is a set of examples, where Class_j is the value of the target variable for the example E^k (i.e., the class for this example), and n_s is the number of examples for the descriptive induction process.

The following assumptions are important to understand the fuzzy quality measures that can be used by NMEEF-SD.

1) An example E^k verifies the APC of a rule R_i if

$$\mathrm{APC}(E^k, R_i) = T(\mu_{LL_1^1}(e_1^k), \ldots, \mu_{LL_{n_v}^{l_{n_v}}}(e_{n_v}^k)) > 0 \tag{12}$$

where $Antecedent\,Part\,Compatibility\,(APC)$ is the degree of compatibility between an example and the antecedent part of a fuzzy rule, i.e., the degree of membership for the example to the fuzzy subspace delimited by the antecedent part of the rule, where we have the following.

- $\mathrm{LL}_{n_v}^{l_{n_v}}$ is the linguistic label number l_{n_v} of the variable n_v.
- $\mu_{\mathrm{LL}_{n_v}^{l_{n_v}}}(e_{n_v}^k)$ is the degree of membership for the value of the feature n_v for the example E^k to the fuzzy set corresponding to the linguistic label l_{n_v} for this variable (n_v).
- T is the $t-$norm selected to represent the meaning of the AND operator (the fuzzy intersection), which, in our case, is the minimum $t-$norm.

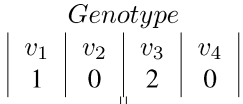
2) An example E^k is covered by a rule R_i if

$$\mathrm{APC}(E^k, R_i) > 0 \quad \text{AND} \quad E^k \in \mathrm{Class}_j. \tag{13}$$

This means that an example is covered by a rule if the example has a degree of membership higher than 0 to the fuzzy input subspace delimited by the antecedent part of the fuzzy rule, and the value indicated in the consequent part of the rule agrees with the value of the target feature for the example. For the categorical variables, the degree of membership is 0 or 1.

On the other hand, in NMEEF-SD, each candidate solution is coded according to the "*Chromosome = Rule*" approach, where only the antecedent is represented in the chromosome, and the consequent is prefixed to one of the possible values of the target feature in the evolution. Therefore, the algorithm must be executed as many times as the number of different values the target variable contains.

NMEEF-SD uses an integer representation model with as many genes as variables contained in the original dataset without considering the target variable. The set of possible values for the categorical features is that indicated by the problem, and for numerical variables, it is the set of linguistic terms determined heuristically or with expert information. In Fig. 2, we can observe a representation for a rule with continuous and discrete variables for the value of target variable Positive.



IF $(v_1 = Low)$ AND $(v_3 = 14)$ THEN $(Class = Positive)$

Fig. 2. Representation of a fuzzy rule with continuous and categorical variables in NMEEF-SD.

B. Quality Measures Considered as Objectives

The goal of any multiobjective optimization algorithm is to find the decision vectors, which correspond to objective vectors and cannot be improved in a dimension without degrading one-another, which is called optimal Pareto front [15]. In SD, maximization of the quality measures is required, not only for the measures selected as objectives in the algorithm but for the other measures used in SD as well.

In the NMEEF-SD algorithm, the quality measures considered as objectives in the evolutionary process can be selected. This permits us to study the combinations of quality measures that provide better results for the problem to solve.

Following are the quality measures and their characteristics that can be selected in this proposal.

- Support (3) is used to quantify the quality of individual rules according to the individual patterns of interest covered. It is a measure with precision and generality characteristics.
- Fuzzy confidence shows the precision of the subgroups and is defined as [7]:

$$\mathrm{FC}nf(R_i) = \frac{\sum_{E^k \in E / E^k \in Class_j} \mathrm{APC}(E^k, R_i)}{\sum_{E^k \in E} \mathrm{APC}(E^k, R_i)}. \tag{14}$$

- Unusualness (9) attempts to obtain a tradeoff between generality, interest, and precision in the results.

C. Evolutionary Model

NMEEF-SD is an MOEA based on the NSGA-II approach [16] oriented toward the extraction of a set of rules for the SD task. Therefore, some changes are performed with respect to the NSGA-II approach.

- The algorithm can use different combinations of quality measures both for the evaluation and the selection of the rules.
- A biased initialization operator is employed.
- A biased mutation operator is used.
- An operator called re-initialization based on coverage is used to prevent the algorithm falling into a local maximum.
- The algorithm only returns the non-dominated solutions of the Pareto, which are obtained at the end of the evolutionary process and reach a predetermined confidence threshold.

The algorithm begins with an initial population (P_t) of a predetermined size and produces an offspring population (Q_t) with the same size. These two populations are joined to form a new population (R_t), where the fast non-dominated sort is applied, and the algorithm forms different fronts in the following way: "The first front (F_1) comprises the non-dominated individuals of this population, which is the Pareto front; the second front (F_2) comprises the individuals dominated by one individual; the third front (F_3) comprises the individuals dominated by two, and so on."

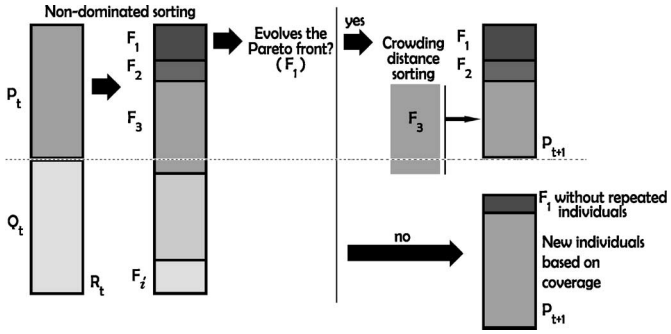


Fig. 3. Operation diagram of the algorithm NMEEF-SD.

The operation scheme of the algorithm NMEEF-SD can be observed in Fig. 3.

A new operator has been added in order to obtain the population of the next generation (P_{t+1}), where the algorithm first checks the Pareto front, as can be observed in Fig. 3. If the Pareto front covers no new example during a period of the evolutionary process, a re-initialization based on coverage is carried out. Otherwise, the algorithm obtains the population for the next generation (P_{t+1}) introducing, in order, the first complete fronts of R_t which fit the size of the population. When the number of individuals of the next front to introduce (F_i) exceeds the number of individuals to introduce in the population P_{t+1}, the first individuals of front F_i are introduced, where the individuals in the front are ordered from highest to lowest crowding distance values.

At the end of the evolutionary process, the algorithm returns the Pareto front (i.e., the front F_1) as the set of optimal solutions. The individuals obtained are those that reach a predetermined fuzzy confidence threshold.

The operation scheme of the algorithm NMEEF-SD can be observed in Fig. 4.

The main aspects considered in the algorithm are presented next.

- *Biased initialization:* The objective is to obtain an initial population with general individuals which cover a high number of examples. This operator generates some of the individuals of the population using only a maximum percentage of variables for each rule. In this way, this initialization operator generates 75% of the individuals of the population with only a maximum of 25% of the variables in each rule, while the rest of the individuals (25%) are generated randomly. This operator allows the algorithm to begin with a set of rules with high generality because most of the generated individuals are rules with a low number of variables.

InitialiseCounters ($t \leftarrow 0$ and $i \leftarrow 1$)
Generate P_t with BiasedInitialisation
while NumberEvaluations is not reached **do**
  GeneratePopulation the offspring Q_t through the GeneticOperators applied in P_t
  Join the parent population P_t and offspring population Q_t in a combined population R_t
  Generate all non-dominated fronts $F = (F_1, F_2, \ldots, F_i)$ of R_t
  if the Pareto front evolves **then**
    $N \longleftarrow Size(P_t)$
    while $N \geq$ NumberIndividuals (F_i) **do**
      Include F_i front in the parent population (P_{t+1})
      $N \longleftarrow N-$NumberIndividuals(F_i)
      $i \longleftarrow i+1$
    end while
    DescendingSort(F_i) using crowding distance operator
    Introduces N individuals of F_i front in the parent population P_{t+1}
  else
    Apply Re-initialisationBasedOnCoverage
  end if
  UpdateCounters ($t \longleftarrow t+1$ and $i \longleftarrow 1$)
end while
Return the individuals of the Pareto front which reach a fuzzy confidence threshold

Fig. 4. Operation scheme of NMEEF-SD algorithm.

- *Genetic operators:* These operators allow us to generate the offspring population (Q_t) departing from the population P_t. The population Q_t is obtained using the standard operators' tournament selection [58] and multipoint crossover [59] and a biased mutation operator presented in [7].
- *Re-initialization based on coverage:* The last step for each generation is to generate the population for the next generation (P_{t+1}). Before carrying out this step, a verification is performed on the Pareto to see whether or not it evolves. We consider that the Pareto evolves if it covers at least one example of the dataset not covered by the Pareto of the previous generation. If the Pareto does not evolve for more than 5% of the evolutionary process (quantified through the number of evaluations) re-initialization is performed through the following steps.
  a) The nonrepeated individuals of the Pareto front are directly replicated in the population of the next generation (P_{t+1}).
  b) This population is completed with individuals generated through initialization based on coverage, as explained next. This operator generates individuals covering examples of the dataset up to the moment.

  To avoid the generation of very specific rules, this operator generates individuals with a maximum of 90% of the variables taking part in the rule. With this biased initialization of the individuals, we achieve new rules, which probably cover more examples since they tend to be more general. This operator allows us to improve the diversity in the evolutionary process. It also prevents the algorithm from entering a local maximum. The improvement of diversity achieved by this operator has been demonstrated by statistical tests of the results obtained by the algorithm, which are available on the Web site containing additional material on this work (http://simidat.ujaen.es/NMEEF-SD). The use of this new operator, together with the crowding distance operator, aims to obtain a good balance between convergence and diversity.
- *Stop condition:* The evolutionary process ends when the maximum number of evaluations is reached. The algorithm returns only the rules in the Pareto which reach a predefined confidence (14) threshold.

D. NMEEF-SD Versus the Remaining Evolutionary Fuzzy Algorithms for Subgroup Discovery

Table I shows a comparison of the features of the evolutionary algorithms for SD: SDIGA, MESDIF, and NMEEF-SD.

In this table, it can be seen that, with respect to SDIGA, NMEEF-SD uses a multiobjective approach more suited to the optimization of several objectives in the evolutionary process; with respect to MESDIF, NMEEF-SD uses a powerful multiobjective evolutionary model based on the well-known NSGA-II algorithm. NMEEF-SD also offers, with respect to SDIGA and MESDIF, more flexibility because of its ability to select different quality measures as objectives in the evolutionary process.

In SDIGA, the generality of the rules is enhanced by the application of the local search, while in MESDIF and NMEEF-SD, the generality is promoted by means of a biased initialization of the population.

Diversity is promoted in MESDIF using a niches technique (density as defined in SPEA2) and an objective in the evolutionary process (the original support) and, in NMEEF-SD, by means of a niches technique (the crowding distance as defined in NSGA-II) and re-initialization based on coverage operator.

IV. Experimental Study

In this experimental study, the aim was to analyze which combinations of quality measures used in the evolutionary process of NMEEF-SD offer better results and to compare the performance of the algorithm with other SD algorithms (both evolutionary and nonevolutionary). Therefore, we first studied the behavior of the NMEEF-SD algorithm with respect to the use of different combinations of quality measures within the evolutionary process.

The best combination was then compared with other evolutionary and classical SD algorithms. The experimentation was undertaken with real datasets from UCI repository [60]. The properties of these datasets are presented in Table II: number of variables (n_v), number of discrete variables (n_{vD}), number of continuous variables (n_{vC}), number of classes of the dataset (n_c), and number of examples (n_s).

For this purpose, the experiments performed are described in the following sections:

TABLE I
COMPARISON OF FEATURES OF THE EVOLUTIONARY ALGORITHMS FOR SUBGROUP DISCOVERY

Feature	*SDIGA*	*MESDIF*	*NMEEF − SD*
Evolutionary model	IRL Mono-objective with weights	MOEA based on SPEA2 model	MOEA based on NSGA-II model
Knowledge representation	Canonical and DNF rules	Canonical and DNF rules	Canonical and DNF rules
Individual Coding	Chromosome=Rule	Chromosome=Rule	Chromosome=Rule
Quality measures used as objectives	Fuzzy Confidence and Support	Fuzzy Confidence, Support and Original Support	Selected by the user among: among: Support, Unusualness and Fuzzy Confidence
Selection	Steady step GA	Tournament selection	Tournament selection
Operators	Biased mutation, Two point crossover and Local search	Biased mutation, Two point crossover	Biased mutation, Two point crossover Re-initialisation based on coverage
Elitism	No	Yes	Yes

TABLE II
PROPERTIES OF THE DATASETS USED FROM THE UCI REPOSITORY

Name	n_v	n_{vD}	n_{vC}	n_c	n_s
Appendicitis	7	0	7	2	106
Australian	14	8	6	2	690
Balance	4	0	4	3	625
Breast-w	9	9	0	2	699
Bridges	7	4	3	2	102
Bupa	6	0	6	2	345
Car	6	6	0	4	1728
Chess	36	36	0	2	3196
Cleveland	13	0	13	5	303
Dermatology	33	33	0	6	366
Diabetes	8	0	8	2	768
Echo	6	1	5	2	131
German	20	13	7	2	1000
Glass	9	0	9	6	214
Haberman	3	0	3	2	306
Hayesroth	4	4	0	3	132
Heart	13	6	7	2	270
Hepatitis	19	13	6	2	155
Hypothyroid	25	18	7	2	3163
Ionosphere	34	0	34	2	351
Iris	4	0	4	3	150
Led	7	0	7	10	500
Lymp	18	18	0	4	148
Marketing	13	13	0	10	8993
Mushrooms	22	22	0	2	8124
Nursery	8	8	0	5	12960
Tic-tac-toe	9	9	0	2	958
Vehicle	18	0	18	4	846
Vote	16	16	0	2	435
Wine	13	0	13	3	178

TABLE III
PARAMETER SPECIFICATION FOR THE EVOLUTIONARY ALGORITHMS EMPLOYED IN THE EXPERIMENTATION

Algorithm	Parameters
NMEEF-SD	Executions=5 Linguistic labels=(3 and 5) Minimum confidence=(0.6, 0.7, 0.8 and 0.9) Population size=50 Maximum evaluations=10000 Crossover probability=0.60 Mutation probability=0.10
MESDIF	Executions=5 Linguistic labels=(3 and 5) Elite population=(3, 4, 5 and 10) Population size=100 Maximum evaluations=10000 Crossover probability=0.60 Mutation probability=0.01
SDIGA	Executions=5 Linguistic labels=(3 and 5) Minimum confidence=(0.6, 0.7, 0.8 and 0.9) Population size=100 Maximum evaluations=10000 Crossover probability=0.60 Mutation probability=0.01

- In Section IV-A, the experimental framework for the evolutionary and classical algorithms is presented, and the statistical tests used for the comparisons are introduced.
- In Section IV-B, an analysis of NMEEF-SD algorithm with different quality measures as objectives in the evolutionary process is performed using three different approaches, and then, the best combination of quality measures for NMEEF-SD is selected.
- In Section IV-C, a study comparing NMEEF-SD with the evolutionary algorithms for SD (SDIGA and MESDIF) is presented. For this comparison, the best result for each algorithm was selected for each dataset with different levels of granularity.
- In Section IV-D, the results of NMEEF-SD algorithm are compared with the best results of the classical algorithms for SD (CN2-SD and Apriori-SD).
- In Section IV-E, time complexity and scalability studies for NMEEF-SD, CN2-SD, and Apriori-SD are performed.
- Finally, in Section IV-F, rules extracted by NMEEF-SD, CN2-SD, and Apriori-SD are shown.

Statistical analyses were used in order to find significant differences among the results obtained by the methods studied following the recommendations performed in [61].

A. Experimental Framework

The experimentation for the evolutionary algorithms was carried out with the parameters shown in Table III over a tenfold cross validation for each dataset. These parameters were established from various experimental studies which allowed us to determine the values that performed better in the algorithms.

With respect to the experimentation for the classical SD algorithms, a previous discretization of the datasets with continuous variables was needed, as these algorithms are not able to conduct continuous variables. The discretization process used is the Fayyad discretize [62], i.e., the one used in the papers describing CN2-SD [63] and Apriori-SD [19], and in scalability studies with Apriori-SD [64]. In addition, we used the multiplicative weights 0.3, 0.5, 0.9, and the additive weight for CN2-SD,

while we used different minimum confidence values, 0.6, 0.7, 0.8, and 0.9 for Apriori-SD.

The results shown for the experiments are the average of the results obtained for each dataset for the different executions, i.e., 50 for evolutionary algorithms (five per group of cross validation, because these algorithms are nondeterministic) and ten for classical ones (one per group). The quality measures shown in the result tables are the average results for the rule sets: number of rules (n_r), number of variables (n_v), significance (SIG), unusualness (WRAcc), and support (SUP_c). Furthermore, for the evolutionary algorithms that obtain fuzzy rule, the fuzzy confidence (FCNF) is used, while for the classical algorithms which extract crisp rules, standard confidence (CNF) is used.

For reasons of brevity, the paper only includes the results obtained by the statistical tests. For more details, please visit the Web site (http://simidat.ujaen.es/NMEEF-SD) associated with this work. It contains the partitioned datasets used in the present analysis so that any interested researcher can use them to include his own results and extend the present comparison. The tables with all the results are also available so that they can be used as a reference for future comparisons using the same dataset partitions.

To perform the statistical analysis in order to find significant differences between the algorithms, we used nonparametric tests, following the recommendations made in [65], providing a set of simple, safe, and robust methods for statistical comparisons. We employed different approaches for multiple comparison: the Iman and Davenport test [66] and the Holm method [67]. Detailed information related to these statistical tests is available in [68] and [69] and on the website http://sci2s.ugr.es/sicidm/.

In all the experiments, we have used a level of significance of $\alpha = 0.05$. For all of them, the results of the different statistical tests applied are shown in two tables: one for the Iman–Davenport test and another for the Holm tests.

1) Iman–Davenport test shows whether there are significant differences between the results of different algorithms. The tables of results for this test show the name of the statistic $Method$, the $Test\ value$ for the method, the $Distribution\ value$ it employs, and the $p - Value$. The $Distribution\ value$ for Iman–Davenport is employed through an F-distribution, for example, for 30 datasets and three algorithms, we employ an F-distribution with two and 58 degrees of freedom. Furthermore, the highest of the two values compared ($Test\ value$ and $Distribution\,value$) are marked in bold, and as the smallest corresponds to the value given by the statistic, it informs of the rejection of the null hypothesis of equality of means. In this case, there are significant differences among the results in all the datasets.
2) The Holm test is applied when significant differences among the algorithms are found in the Iman–Davenport test. To be able to apply the Holm test, it is first necessary to obtain the Friedman ranking, through the computation of the average values of the different algorithms for all datasets. The algorithm with the best result in this ranking is considered the control algorithm (Alg_{Cont}), which controls the Holm test.

TABLE IV
RESULTS OF IMAN–DAVENPORT TEST IN THE DIFFERENT QUALITY MEASURES STUDIED FOR THE COMPARISON OF THE COMBINATIONS OF QUALITY MEASURES AS OBJECTIVES FOR NMEEF-SD ALGORITHM

$Quality_{mea}$	$Test\ value$	$Distribution\ value$	$p - Value$
SIG	3.155	**25.290**	1.0E-6
WRAcc	3.155	**70.239**	1.0E-12
SUP_c	3.155	**61.940**	1.0E-11
CNF	**3.155**	2.080	0.134

In the result tables for the Holm test, the algorithms are shown in descending order of z. Thus, by using the normal distribution, we can obtain the corresponding $p - Value$ associated with each comparison and this can be compared with the associated α/i in the same row of the table to show whether the associated hypothesis of equal behavior is rejected in favor of the best ranking algorithm or not.

3) $p - Value$ is the value which denotes the point where there are significant differences in the comparison. As we use a level of confidence of $\alpha = 0.05$, all $p - Value$s lower than this value show us significant differences in the comparison. The lower the $p - Value$, the more significant the result.

B. Quality Measures Analysis

The optimization of quality measures is a fundamental objective for an SD algorithm. With the use of the multiobjective approach in NMEEF-SD algorithm, it is possible to study different combinations of quality measures as objectives in the evolutionary process and then compare the results obtained in order to choose the best one.

To the best of our knowledge, there are no current studies on determining the most suitable quality measures for an SD algorithm. In this study, several combinations of quality measures in NMEEF-SD were considered. A combination of quality measures can be considered the best if it obtains the best results, not only in the quality measures used in the evolutionary process but in the other measures as well.

The following combinations were studied:

1) support (3) and unusualness (9), labeled with the abbreviation SU;
2) support and confidence (14), labeled with the abbreviation SC;
3) support, confidence, and unusualness, labeled with the abbreviation SCU.

The complete results table can be found on the website associated with this paper (http://simidat.ujaen.es/NMEEF-SD).

The statistical studies which support this work offer accuracy, conciseness, and clarity, and they were performed for the quality measures of significance, unusualness, support, and confidence. These measures were studied independently through the different nonparametric tests mentioned above. The results of these tests show the existence or not of significant differences between the algorithms for each measure. The level of confidence used was $\alpha = 0.05$ in all the experiments.

In Table IV, the results of the Iman–Davenport test for each quality measure can be observed, where the results obtained

TABLE V
RESULTS OF HOLM TEST FOR THE DIFFERENT QUALITY MEASURES ANALYZED FOR THE COMPARISON OF THE DIFFERENT COMBINATIONS STUDIED FOR NMEEF-SD

Quality measure	$Alg_{Control}$	i	algorithm	z	p	α/i	Hypothesis
SIG	SU	2	SC	5.2285	1.7086E-7	0.025	**Rejected**
		1	SCU	3.2920	9.9464E-4	0.05	**Rejected**
WRAcc	SU	2	SC	6.3258	2.5180E-10	0.025	**Rejected**
		1	SCU	4.5184	6.2284E-6	0.05	**Rejected**
SUP_c	SU	2	SC	6.0676	1.2977E-9	0.025	**Rejected**
		1	SCU	4.7766	1.7821E-6	0.25	**Rejected**

TABLE VI
RESULTS OF IMAN–DAVENPORT TEST FOR THE QUALITY MEASURES STUDIED FOR THE COMPARISON OF THE DIFFERENT EVOLUTIONARY ALGORITHMS FOR SUBGROUP DISCOVERY

$Quality_{mea}$	$Test\ value$	$Distribution\ value$	$p-Value$
SIG	3.155	**9.213**	3.4E-4
WRAcc	3.155	**31.839**	1.0E-7
SUP_c	3.155	**12.894**	4.0E-5
CNF	3.155	**27.862**	1.0E-7

show significant differences among the different combinations in significance, unusualness, and support. Therefore, it is necessary to apply the Holm test in order to detail these differences.

The results for the Holm test can be observed in Table V, highlighting the fact that the combination of the quality measures SU obtains the best results with significant differences with respect to the other combinations of quality measures. Therefore, the best combination of quality measures for the evolutionary approach NMEEF-SD with respect to the dataset studied is the use of SU. The use of support in the evolutionary process aims to obtain generality in subgroups, while unusualness provides a relative accuracy measure with respect to the coverage of the rule and the accuracy gain. These measures allow us to obtain a good balance between precision, interest, and coverage.

In the following sections, the version of the NMEEF-SD algorithm using the combination of SU was considered.

C. Comparison of the Existing Evolutionary Algorithms for Subgroup Discovery

In this study, a comparison between NMEEF-SD and the other evolutionary algorithms for SD, SDIGA, and MESDIF can be observed. Complete tables of results are available on the website associated with this paper (http://simidat.ujaen.es/NMEEF-SD).

The results of the statistical study can be observed in Tables VI and VII.

Table VI shows that the Iman–Davenport test finds significant differences in all the quality measures, and therefore, it is necessary to apply the Holm test to entail these differences.

The results of the Holm test for each quality measure are shown in Table VII, where the NMEEF-SD algorithm obtained significant differences with respect to the others. Therefore, NMEEF-SD is the evolutionary algorithm which obtains the best results for the datasets studied.

These results show that NMEEF-SD obtains significantly better results than the other evolutionary algorithms for SD, not only for the quality measures used in the evolutionary process but for the rest of quality measures studied as well, obtaining a good balance between them.

D. Comparison of NMEEF-SD and the Classical-Subgroup Discovery Algorithms

In this section, the main objective was to search for significant differences between the algorithms NMEEF-SD and the classical CN2-SD and Apriori-SD. For this purpose, a subset of 20 datasets from the datasets of Table II was used. This is because the classical algorithms do not work properly with high-dimensional datasets. This subset comprises datasets Appendicitis, Australian, Balance, Breast-w, Bridges, Bupa, Car, Cleveland, Diabetes, Echo, German, Haberman, Hayes-Roth, Heart, Hepatitis, Iris, Led, Tic-tac-toe, Vote, and Wine. Complete tables of results are available on the website associated with this paper (http://simidat.ujaen.es/NMEEF-SD).

Once more, we first applied the Iman–Davenport test to the results of the three algorithms compared, whose results are shown in Table VIII for the quality measures (Qua_{mea}) studied. This table shows the following.

1) In significance (SIG), it can be observed that there are no significant differences between the algorithms because the null hypothesis was not rejected.
2) In unusualness (WRAcc), support (SUP_c), and confidence (CNF), there are significant differences between the algorithms, because the null hypothesis was rejected.

Therefore, it was necessary to apply the Holm test for unusualness, support, and confidence in order to find which algorithm had significant differences with respect to the others and achieves greater detail and accuracy. The results for the Holm test for these measures can be observed in Table IX.

The analysis of the results in Table IX shows the following.

1) In unusualness (WRAcc), the hypothesis was rejected for all the algorithms, and the associated hypothesis of the same behavior was rejected in favor of the best ranking algorithm, leading us to conclude that NMEEF-SD has significant differences in all the algorithms in this quality measure.
2) In support (SUP_c), the results show that the hypothesis was rejected in favor of the control algorithm; therefore,

TABLE VII
RESULTS OF HOLM TEST FOR THE DIFFERENT QUALITY MEASURES ANALYZED FOR THE DIFFERENT EVOLUTIONARY ALGORITHMS FOR SUBGROUP DISCOVERY

Quality measure	$Alg_{Control}$	i	algorithm	z	p	α/i	Hypothesis
SIG	NMEEF-SD	2	MESDIF	3.3565	7.8912E-4	0.025	**Rejected**
		1	SDIGA	3.2274	0.0012	0.05	**Rejected**
WRAcc	NMEEF-SD	2	MESDIF	5.0348	4.7815E-7	0.025	**Rejected**
		1	SDIGA	4.6475	3.3585E-5	0.05	**Rejected**
SUP_c	NMEEF-SD	2	SDIGA	4.0020	6.2787E-5	0.025	**Rejected**
		1	MESDIF	3.3565	7.8911E-4	0.25	**Rejected**
CNF	NMEEF-SD	2	SDIGA	5.0348	4.7815E-7	0.025	**Rejected**
		1	MESDIF	4.2602	2.0416E-5	0.25	**Rejected**

TABLE VIII
RESULTS OF IMAN–DAVENPORT TEST FOR THE QUALITY MEASURES STUDIED FOR THE COMPARISON BETWEEN NMEEF-SD, CN2-SD, AND APRIORI-SD

Qua_{mea}	$Test\ value$	$Distribution\ value$	$p-Value$
SIG	**3.2448**	0.7916	0.460
WRAcc	3.2448	**7.8788**	1.3E-3
SUP_c	3.2448	**13.9004**	3.0E-5
CNF	3.2448	**7.0497**	2.4E-3

NMEEF-SD has significant differences with respect to the rest of the algorithms.

3) In confidence (CNF), it can be observed that NMEEF-SD is the control algorithm and obtains significant differences with respect to CN2-SD but not with respect to Apriori-SD.

In SD, it is considered that an algorithm shows good behavior if it obtains good results with respect to the different quality measures, considering a good relationship between support and confidence, as well and, furthermore, obtaining simple, general, and accurate subgroups. In this sense, NMEEF-SD is the algorithm which obtains the best results in this experimentation. Specifically, this algorithm obtains the best results in support, unusualness, and confidence, while in significance, the algorithms studied obtain similar results.

E. Time Complexity and Scalability Analysis

Another important aspect involved in comparison of different algorithms is the cost in time and the scalability of each algorithm. In order to estimate the cost in time of the algorithms studied, an empirical study was performed that considered the execution time of the different algorithms in the training sets. In Table X, the average time in seconds for the experiments performed for each algorithm in the different datasets can be observed, where the best values are highlighted. These experiments were carried out using a computer with an Intel Core 2 Duo with a microprocessor of 2.4 GHz and 4 GB of RAM.

The results show that NMEEF-SD obtains the best average results in the total experimentation with a value lower than 5 s. This is because the complexity time for NMEEF-SD is related to the number of classes and instances of the dataset, while in CN2-SD and Apriori-SD, the time is related to the number of variables and instances of the dataset.

In order to further study the scalability of the algorithms CN2-SD, Apriori-SD, and NMEEF-SD, the dataset Nursery was used. This dataset was resized in multiples of ten, and for every set, the execution time for each algorithm was measured. In Fig. 5, the complete results with respect to the number of instances and seconds can be observed, where this axis is shown in logarithmic scale. In this figure, it can be observed that, in this dataset, NMEEF-SD obtains good results with respect to the other algorithms with a large time difference, where the growth of the distribution function of time for NMEEF-SD is lower than the function of the classical algorithms studied.

F. Rules Obtained for NMEEF-SD, CN2-SD, and Apriori-SD in Different Datasets

The number of rules and variables obtained by the SD algorithms offers information about the interpretability of the subgroup obtained for the algorithms in SD. In Table XI, the average of the number of rules and variables for NMEEF-SD, CN2-SD, and Apriori-SD is shown. In this table, it can be observed that the algorithm with the best interpretability is NMEEF-SD because it obtains few rules with few variables.

Some rules from the dataset Diabetes obtained by the algorithms NMEEF-SD, CN2-SD, and Apriori-SD can be observed in Table XII. Diabetes is a real-world dataset with 768 examples and eight features ($preg$, $plas$, $pres$, $skin$, $insu$, $mass$, $pedi$, and age), where all them are continuous variables. Furthermore, this dataset has two values for class variable: $Positive$ and $Negative$.

In Table XII, all the subgroups obtained by NMEEF-SD for a group of cross validation are shown, while for CN2-SD and Apriori-SD, only one rule of each class is shown, because the number of rules obtained for these classical algorithms is too big. The rules were obtained for NMEEF-SD using three linguistic labels for the continuous variables, so $LL_1 = Low$, $LL_2 = Medium$, and $LL_3 = High$, while classical algorithms were discretized by the Fayyad method.

For NMEEF-SD, the values of the quality measures of significance, unusualness, support, and confidence are shown for each rule. For CN2-SD, the percentage of samples covered the class by the rule and the percentage of samples of the other classes are shown, and for Apriori-SD, the number of samples for the different classes are shown.

TABLE IX
HOLM TEST TABLE FOR THE QUALITY MEASURES WITH SIGNIFICANT DIFFERENCES BETWEEN THE ALGORITHMS

Quality measure	Alg_{Cont}	i	$Algorithm$	z	p	α/i	$Hypothesis$
WRAcc	NMEEF-SD	2	CN2-SD	3.0041	0.0026	0.025	**Rejected**
		1	Apriori-SD	2.9251	0.0034	0.05	**Rejected**
SUP_c	NMEEF-SD	2	Apriori-SD	4.1109	3.9401E-5	0.025	**Rejected**
		1	CN2-SD	2.0554	0.0398	0.05	**Rejected**
CNF	NMEEF-SD	2	CN2-SD	2.9251	0.0034	0.025	**Rejected**
		1	Apriori-SD	0.1581	0.8743	0.05	Accepted

TABLE X
TIMES IN SECONDS FOR THE ALGORITHMS STUDIED IN THE DIFFERENT DATASETS

Dataset	NMEEF-SD	CN2-SD	Apriori-SD
Appendicitis	**1.90**	3.70	2.72
Australian	**7.32**	107.92	214.13
Balance	4.24	2.22	**1.63**
Breast	3.27	48.74	**2.19**
Bridges	2.57	**1.51**	2.64
Bupa	2.93	**1.41**	1.94
Car	12.50	88.15	**3.03**
Cleveland	**6.89**	12.14	375.92
Diabetes	4.73	54.94	**2.51**
Echo	2.08	**1.17**	1.94
German	**8.12**	264.65	361,481.36
Haberman	2.25	2.20	**1.41**
Hayesroth	2.77	1.66	**1.35**
Heart	**2.82**	9.63	251.91
Hepatitis	**3.82**	6.02	135,074.87
Iris	**1.29**	2.08	1.30
Led	13.87	34.18	**3.10**
Tic-tac-toe	4.14	65.80	**3.31**
Vote	**3.27**	25.27	3,815.07
Wine	3.56	**1.36**	121.04
Average	**4.72**	36.74	25,068.14

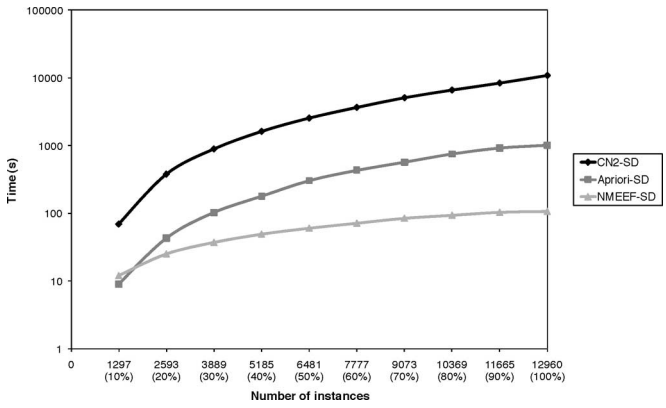


Fig. 5. Scalability study for CN2-SD, Apriori-SD, and NMEEF-SD in the dataset Nursery.

TABLE XI
AVERAGE NUMBER OF RULES AND VARIABLES FOR NMEEF-SD, CN2-SD, AND APRIORI-SD IN THE COMPLETE EXPERIMENTATION

$Algorithm$	n_r	n_v
NMEEF-SD	3.30±2.18	2.53±0.50
CN2-SD	15.16±9.64	2.44±1.11
Apriori-SD	10.24±7.68	1.75±0.49

TABLE XII
EXAMPLES OF RULES AND THE RESULTS FOR THE QUALITY MEASURES OF THE ALGORITHMS NMEEF-SD, CN2-SD, AND APRIORI-SD FOR DIABETES DATASETS

NMEEF-SD				
$Rule$	$SIGN$	$WRAcc$	SUP_c	$FCNF$
IF (plass=LL_2 AND pres=LL_2 AND age=LL_1) THEN Negative	8.736	0.078	0.800	0.774
IF (plass=LL_2) THEN Negative	1.235	0.052	0.980	0.700
IF (plass=LL_2 AND pres=LL_2 AND pedi=LL_1) AND insu=LL_1 THEN A15=0	6.117	0.051	0.740	0.764
IF (plass=LL_2 AND age=LL_1) THEN Negative	5.597	0.087	0.880	0.769
IF (plass=LL_2 AND insu=LL_1) THEN Negative	4.793	0.083	0.940	0.711
IF (plass=LL_2 AND pres=LL_2 AND insu=LL_1) THEN Negative	7.391	0.069	0.820	0.732
IF (plass=LL_2 AND pres=LL_2) THEN Negative	6.058	0.081	0.860	0.728
IF (plass=LL_2 AND insu=LL_1 AND pedi=LL_1) THEN Negative	5.057	0.066	0.840	0.747
IF (plass=LL_2) THEN Positive	6.467	0.054	0.630	0.595

CN2-SD		
$Rule$	C	$\overline{C}$
IF (age≠1 AND plas≤1 AND preg≠1) THEN Negative	0.926	0.074
IF (mass=1 AND insu≠1 AND plas≠2) THEN Positive	0.539	0.461

Apriori-SD		
$Rule$	C_1	C_2
IF (pedi=0) THEN Negative	326	132
IF (preg=1 AND mass=1) THEN Positive	43	79

V. CONCLUSION

In this paper, a multiobjective algorithm for extracting fuzzy rules in SD has been proposed. The main objective of this GFS is to extract a set of simple and descriptive rules for SD with proper quality values by taking into consideration different quality measures usually used in descriptive and predictive data mining.

NMEEF-SD follows a multiobjective approach of NSGA-II but is oriented toward SD, and it uses specific operators to promote the extraction of simple, interpretable, and high-quality SD rules. The algorithm can use different combinations of quality measures for both the evaluation and the selection of the rules. It also includes components to enhance diversity, generality, and accuracy among the rules. In particular, diversity is enhanced in the population by using an operator of re-initialization based on coverage and a niching technique (the crowding distance in the selection operator) used in the NSGA-II-based models. To enhance generalization, the algorithm includes operators of biased initialization and biased mutation, as well as the objectives considered in the evolutionary approach, and to assure accuracy, in addition to the objectives, NMEEF-SD returns as its final

solution those rules of the Pareto front, which reach a certain fuzzy confidence threshold.

An elaborate experimental study has been performed for the NMEEF-SD and other subgroup discovery methods such as CN2-SD, Apriori-SD, MESDIF, and SDIGA supported by the use of nonparametric tests for comparison. This study shows that the algorithm obtains the best results using the quality measures of support and unusualness as objectives of the evolutionary process. In addition, good results were obtained for NMEEF-SD in comparison with other existing algorithms, where NMEEF-SD obtains significant differences with respect to the other algorithms in most of the quality measures. Furthermore, a study of time complexity and scalability has been presented, showing the superiority of NMEEF-SD in the use of different types of datasets.

All in all, NMEEF-SD is a robust algorithm which obtains better results using unusualness and support to guide the genetic search, obtaining simple, accurate, and interpretable subgroup descriptions which achieve competitive results not only for the quality measures used in the evolutionary process but for the other quality measures considered for SD as well.

REFERENCES

[1] N. Lavrac, B. Kavsek, P. A. Flach, and L. Todorovski, "Subgroup Discovery with CN2-SD," *J. Mach. Learn. Res.*, vol. 5, pp. 153–188, 2004.

[2] S. Bay and M. Pazzani, "Detecting group differences: Mining contrast sets," *Data Mining Knowl. Discov.*, vol. 5, pp. 213–246, 2001.

[3] G. Dong, J. Li, and L. Wong, "The use of emerging patterns in the analysis of gene expression profiles for the diagnosis and understanding of diseases," in *New Generation Data Mining Applications*, New York: Wiley, 2005, pp. 331–354.

[4] P. Kralj, N. Lavrac, and G. I. Webb, "Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining," *J. Mach. Learn. Res.*, vol. 10, pp. 377–403, 2009.

[5] P. Kralj, N. Lavrac, D. Gamberger, and A. Krstacic, "CSM-SD: Methodology for contrast set mining through subgroup discovery," *J. Biomed. Inf.*, vol. 42, no. 1, pp. 113–122, 2009.

[6] B. Nannings, R. J. Bosnian, and A. Abu-Hanna, "A subgroup discovery approach for scrutinizing blood glucose management guidelines by the identification of hyperglycemia determinants in ICU patients," *Methods Inf. Med.*, vol. 47, no. 6, pp. 480–488, 2009.

[7] M. J. del Jesus, P. González, F. Herrera, and M. Mesonero, "Evolutionary fuzzy rule induction process for subgroup Discovery: A case study in marketing," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 4, pp. 578–592, Aug. 2007.

[8] C. Romero, P. González, S. Ventura, M. J. del Jesus, and F. Herrera, "Evolutionary algorithm for subgroup discovery in e-learning: A practical application using Moodle data," *Expert Syst. Appl.*, vol. 36, pp. 1632–1644, 2009.

[9] E. Hüllermeier, "Fuzzy methods in machine learning and data mining: Status and prospects," *Fuzzy Sets Syst.*, vol. 156, no. 3, pp. 387–406, 2005.

[10] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA/White Plains, NY: Addison-Wesley/Longman, 1989.

[11] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena, *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. Singapore: World Scientific, 2001.

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

[13] 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 for data mining problems," *Soft Comput.*, vol. 13, no. 3, pp. 307–318, 2009.

[14] C. A. Coello, D. A. V. Veldzhuizen, and G. B. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd ed. Norwell, MA: Kluwer, 2007.

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

[16] K. Deb, A. Pratap, S. Agrawal, 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.

[17] H. Isibuchi, "Multiobjective genetic fuzzy systems: Review and future research directions," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2007, pp. 913–918.

[18] F. J. Berlanga, M. J. del Jesus, P. González, F. Herrera, and M. Mesonero, "Multiobjective evolutionary induction of subgroup discovery fuzzy rules: A case study in marketing," in *Proc. 6th Ind. Conf. Data Mining*, ser. LNCS, vol. 4065. New York, Springer-Verlag, 2006, pp. 337–349.

[19] B. Kavsek and N. Lavrac, "APRIORI-SD: Adapting association rule learning to subgroup discovery," *Appl. Artif. Intell.*, vol. 20, pp. 543–583, 2006.

[20] W. Kloesgen, "Explora: A multipattern and multistrategy discovery assistant," in *Advances in Knowledge Discovery and Data Mining*. Menlo Park, CA: Amer. Assoc. Artif. Intell., 1996, pp. 249–271.

[21] S. Wrobel, "An algorithm for multi-relational discovery of subgroups," in *Proc. 1st Eur. Symp. Principles of Data Mining and Knowl. Discov.*, ser. LNAI, vol. 1263. New York: Springer-Verlag, 1997, pp. 78–87.

[22] D. Gamberger and N. Lavrac, "Expert-guided subgroup Discovery: Methodology and application," *J. Artif. Intell. Res.*, vol. 17, pp. 501–527, 2002.

[23] V. Jovanoski and N. Lavrac, "Classification rule learning with APRIORI-C," in *Proc. 10th Portuguese Conf. Artif. Intell. Prog. Artif. Intell., Knowl. Extraction, Multi-agent Syst., Logic Program. Constraint Solving*, ser. LNCS, vol. 2258. New York: Springer-Verlag, 2001, pp. 44–51.

[24] R. Agrawal, T. Imieliski, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, New York: ACM, 1993, pp. 207–216.

[25] P. Clark and T. Niblett, "The CN2 induction algorithm," *Mach. Learn.*, vol. 3, pp. 261–283, 1989.

[26] F. Zelezny and N. Lavrac, "Propositionalization-based relational subgroup discovery with RSD," *Mach. Learn.*, vol. 62, pp. 33–63, 2006.

[27] M. Atzmueller and F. Puppe, "SD-Map—A fast algorithm for exhaustive subgroup discovery," in *Proc. 17th Eur. Conf. Machine Learn. and 10th Eur. Conf. Principles Pract. Knowl. Discov. Databases*, ser. LNCS, vol. 4213. New York: Springer-Verlag, 2006, pp. 6–17.

[28] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, New York: ACM Press, 2000, pp. 1–12.

[29] P. N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. London, U.K.: Pearson, 2006.

[30] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. Verkamo, "Fast discovery of association rules," in *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds. Menlo Park, CA, AAAI, 1996, pp. 307–328.

[31] N. Lavrac, P. A. Flach, and B. Zupan, "Rule evaluation measures: A unifying view," in *Proc. 9th Int. Workshop Inductive Logic Program.*, ser. LNCS, vol. 1634. New York: Springer-Verlag, 1999, pp. 174–185.

[32] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computation*. New York: Springer-Verlag, 2003.

[33] L. A. Zadeh, "Information control," *Fuzzy Sets*, vol. 8, pp. 338–353, 1965.

[34] L. A. Zadeh, "The concept of a linguistic variable and its applications to approximate reasoning. Parts I, II, III," *Inf. Sci.*, vol. 8–9, pp. 199–249, 301–357, 43–80, 1975.

[35] R. Palm, H. Hellendoorn, and D. Driankov, *Model Based Fuzzy Control*. New York: Springer-Verlag, 1997.

[36] W. Pedrycz, *Fuzzy Modelling: Paradigms and Practices*. Norwell, MA: Kluwer, 1996.

[37] L. Kuncheva, *Fuzzy Classifier Design*. New York: Springer-Verlag, 2000.

[38] H. Ishibuchi, T. Nakashima, and M. Nii, *Classification and Modeling With Linguistic Information Granules: Advanced Approaches to Linguistic Data Mining*. New York: Springer-Verlag, 2004.

[39] O. Cordón and F. Herrera, "A three-stage evolutionary process for learning descriptive and approximative fuzzy logic controller knowledge bases from examples," *Int. J. Approx. Reason.*, vol. 17, pp. 369–407, 1997.

[40] H. Ishibuchi, T. Nakashima, and T. Murata, "Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 29, no. 5, pp. 601–618, Oct. 1999.

[41] M. Setnes and J. A. Roubos, "GA-fuzzy modeling and classification: Complexity and performance," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 5, pp. 509–522, Oct. 2000.

[42] T. P. Hong, C. H. Chen, Y. L. Wu, and Y. C. Lee, "A GA-based fuzzy mining approach to achieve a trade-off between number of rules and suitability of membership fuctions," *Soft Comput.*, vol. 10, no. 11, pp. 1091–1101, 2006.
[43] M. Kaya, "Multi-objective genetic algorithm based approaches for mining optimized fuzzy association rules," *Soft Comput.*, vol. 10, no. 7, pp. 578–586, 2006.
[44] C. H. Tsang, S. Kwong, and H. Wang, "Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection," *Pattern Recognit.*, vol. 40, no. 9, pp. 2373–2391, 2007.
[45] R. Alhajj and M. Kaya, "Multi-objective genetic algorithms based automated clustering for fuzzy association rules mining," *J. Intell. Inf. Syst.*, vol. 31, no. 3, pp. 243–264, 2008.
[46] C. H. Chen, T. P. Hong, V. S. Tseng, and C. S. Lee, "A genetic-fuzzy mining approach for items with multiple minimum supports," Special Issue on genetic fuzzy systems: Recent developments and future directions. *Soft Comput.—Fusion Found., Method. Appl.*, vol. 13, no. 5, pp. 521–533, 2009.
[47] C. H. Chen, T. P. Hong, and V. S. Tseng, "An improved approach to find membership functions and multiple minimum supports in fuzzy data mining," *Expert Syst. Appl.*, vol. 36, no. 6, pp. 10 016–10024, 2009.
[48] J. Alcalá-Fdez, R. Alcalá, M. J. Gacto, and F. Herrera, "Learning the membership function contexts for mining fuzzy association rules by using genetic algorithms," *Fuzzy Sets Syst.*, vol. 160, no. 7, pp. 905–921, 2009.
[49] J. Casillas, B. Carse, and L. Bull, "Fuzzy-XCS: A Michigan genetic fuzzy system," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 4, pp. 536–550, Aug. 2007.
[50] O. Cordón, M. J. del Jesus, F. Herrera, and M. Lozano, "MOGUL: A methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach," *Int. J. Intell. Syst.*, vol. 14, pp. 1123–1153, 1999.
[51] M. L. Wong and K. S. Leung, *Data Mining Using Grammar Based Genetic Programming and Applications*. Norwell, MA: Kluwer, 2000.
[52] J. Casillas and B. Carse, "Special issue on genetic fuzzy systems: Recent developments and future directions," *Soft Comput.*, vol. 13, no. 5, pp. 417–418, 2009.
[53] O. Cordón, R. Alcalá, J. Alcalá-Fdez, and I. Rojas, "Editorial: Special Issue on genetic fuzzy systems: What's next?" *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 4, pp. 533–535, Aug. 2007.
[54] B. Carse and A. G. Pipe, "Special issue on genetic fuzzy systems," *Int. J. Intell. Syst.*, vol. 22, no. 9, pp. 905–907, 2007.
[55] J. Casillas, M. J. del Jesus, F. Herrera, R. Pérez, and P. Villar, "Special issue on genetic fuzzy systems and the interpretability–accuracy trade-off," *Int. J. Approx. Reason.*, vol. 44, no. 1, pp. 1–3, 2007.
[56] M. J. del Jesus, P. González, and F. Herrera, "Multiobjective genetic algorithm for extracting subgroup Discovery fuzzy rules," in *Proc. IEEE Symp. Comput. Intell. Multicriteria Decision Making*, Piscataway, NJ: IEEE Press, 2007, pp. 50–57.
[57] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Proc. Int. Congr. Evol. Methods Design Optim. Control Appl. Ind. Problems*, 2002, pp. 95–100.
[58] B. L. Miller and D. E. Goldberg, "Genetic algorithms, tournament selection, and the effects of noise," *Complex Syst.*, vol. 9, pp. 193–212, 1995.
[59] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. Michigan Press, 1975.
[60] A. Asuncion and D. J. Newman. (2007). "UCI machine learning repository," [Online]. Available: http://www.ics.uci.edu/∼mlearn/MLRepository.html
[61] 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. 2677–2694, 2008.
[62] U. M. Fayyad and K. B. Irani, "Multi-interval discretization of continuous-valued attributes for classification learning," in *Proc. 13th Int. Joint Conf. Artif. Intell.*, 1993, pp. 1022–1029.
[63] N. Lavrac, P. Flach, B. Kavsek, and L. Todorovski., "Rule induction for subgroup Discovery with CN2-SD," in *Proc. 2nd Int. Workshop Integr. Collab. Aspects Data Mining, Decis. Support Meta-Lear.*, 2002, pp. 77–87.
[64] J. R. Cano, S. García, and F. Herrera, "Subgroup Discovery in large size data sets preprocessed using stratified instance selection for increasing the presence of minority classes," *Pattern Recognit. Lett.*, vol. 29, pp. 2156–2164, 2008.
[65] J. Demsar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, 2006.
[66] R. L. Iman and J. M. Davenport, "Approximations of the critical region of the Friedman statistic," *Commun. Stat., A—Theory Methods*, vol. 9, pp. 571–595, 1980.
[67] S. Holm, "A simple sequentially rejective multiple test procedure," *Scand. J. Stat.*, vol. 6, pp. 65–70, 1979.
[68] S. García, A. Fernández, J. Luengo, and F. Herrera, "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.
[69] 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*, vol. 15, pp. 617–644, 2009.

Cristóbal José Carmona received the M.Sc. degree in computer science from the University of Jaen, Jaen, Spain, in 2006.

He is a Research Cooperator with the Department of Computer Science, University of Jaen, where he is with the Intelligent Systems and Data Mining Research Group. His current research interests include subgroup discovery, contrast set mining, genetic fuzzy systems, genetic algorithms, and data mining.

Pedro González received the M.Sc. and Ph.D. degrees in computer science from the University of Granada, Granada, Spain, in 1991 and 2008, respectively.

Currently, he is an Associate Professor with the Department of Computer Science, University of Jaen, Jaen, Spain. His research interests include fuzzy rule-based systems, machine learning, genetic fuzzy systems, subgroup discovery, and data mining.

María José del Jesus received the M.Sc. and Ph.D. degrees in computer science from the University of Granada, Granada, Spain, in 1994 and 1999, respectively.

She is an Associate Professor with the Department of Computer Science, University of Jaen, Jaen, Spain. Her research interests include fuzzy rule-based systems, fuzzy and linguistic modeling, fuzzy classification, subgroup discovery, feature selection, genetic fuzzy systems, evolutionary radial basis neural networks, knowledge extraction based on evolutionary algorithms, and data mining.

Francisco Herrera (M'10) received the M.Sc. degree in mathematics in 1988 and the Ph.D. degree in mathematics in 1991, both from the University of Granada, Granada, Spain.

He is currently a Professor with the Department of Computer Science and Artificial Intelligence, University of Granada. He is the author or coauthor of more than 150 papers in international journals. He has co-edited five international books and 20 special issues in international journals on different soft computing topics. He is a coauthor of *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases* (World Scientific, 2001). His current research interests include computing with words and decision making, bibliometrics, 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 Associate Editor of the IEEE TRANSACTIONS ON FUZZY SYSTEMS, *Information Sciences*, *Mathware and Soft Computing*, *Advances in Fuzzy Systems*, *Advances in Computational Sciences and Technology*, and the *International Journal of Applied Metaheuristics Computing*. He currently serves as an Area Editor of *Soft Computing* (area of genetic algorithms and genetic fuzzy systems) and is a member of several journal editorial boards, including *Fuzzy Sets and Systems*, *Applied Intelligence*, *Knowledge and Information Systems*, *Information Fusion*, *Evolutionary Intelligence*, the *International Journal of Hybrid Intelligent Systems*, and *Memetic Computation*.