A Novel Framework to Design Fuzzy Rule-Based Ensembles Using Diversity Induction and Evolutionary Algorithms-Based Classifier Selection and Fusion

Oscar Cordón[1,2] and Krzysztof Trawiński[1]

[1] European Centre for Soft Computing, Edificio Científico-Tecnológico, planta 3, C. Gonzalo Gutiérrez Quirós s/n, 33600 Mieres (Asturias), Spain
{oscar.cordon,krzysztof.trawinski}@softcomputing.es

[2] Dept. of Computer Science and Artificial Intelligence (DECSAI) and the Research Center on Information and Communication Technologies (CITIC-UGR), University of Granada, 18071 Granada, Spain
ocordon@decsai.ugr.es

Abstract. Fuzzy rule-based systems have shown a high capability of knowledge extraction and representation when modeling complex, non-linear classification problems. However, they suffer from the so-called curse of dimensionality when applied to high dimensional datasets, which consist of a large number of variables and/or examples. Multiclassification systems have shown to be a good approach to deal with this kind of problems. In this contribution, we propose an multiclassification system-based global framework allowing fuzzy rule-based systems to deal with high dimensional datasets avoiding the curse of dimensionality. Having this goal in mind, the proposed framework will incorporate several multi-classification system methodologies as well as evolutionary algorithms to design fuzzy rule-based multiclassification systems. The proposed framework follows a two-stage structure: 1) fuzzy rule-based multiclassification system design from classical and advanced multiclassification system design approaches, and 2) novel designs of evolutionary component classifier combination. By using our methodology, different fuzzy rule-based multiclassification systems can be designed dealing with several aspects such as improvement of the performance in terms of accuracy, and obtaining a good accuracy-complexity trade-off.

1 Introduction

Multiclassification systems (MCSs), also called classifier ensembles, are machine learning tools capable to obtain better performance than a single classifier when dealing with complex classification problems. They are especially useful when the number of dimensions or the size of the data are really large [1]. The most common base classifiers are decision trees [2] and neural networks [3]. More recently, the use of fuzzy classifiers has also been considered [4–6].

I. Rojas, G. Joya, and J. Cabestany (Eds.): IWANN 2013, Part I, LNCS 7902, pp. 36–58, 2013.

On the other hand, fuzzy rule-based classification systems (FRBCSs) have shown a high capability of knowledge extraction and representation when modeling complex, non-linear classification problems. They consider soft boundaries obtained through the use of a collection of fuzzy rules that could be understood by a human being [1, 7]. Interpretability of fuzzy systems is a characteristic that definitely favors this type of models, as it is often a need to understand the behavior of the given model [8, 9].

FRBCSs, however, have one significant drawback. The main difficulty appears when it comes to deal with a problem consisting of a high number of variables and/or examples. In such a case the FRBCS suffers from the so-called *curse of dimensionality* [7]. It occurs due to the exponential increase of the number of rules and the number of antecedents within a rule with the growth of the number of inputs in the FRBCS. This issue also causes a scalability problem in terms of the run time and the memory consumption.

This paper aims to propose an MCS-based global framework allowing FRBCSs to deal with high dimensional datasets avoiding the curse of dimensionality. With this aim, this framework will incorporate several MCS methodologies taken from the machine learning field as well as evolutionary algorithms to design fuzzy rule-based multiclassification systems (FRBMCSs). The proposed framework follows a two-stage structure: 1) component fuzzy classifier design from classical and advanced MCS design approaches, and 2) novel designs of evolutionary component classifier combination. This methodology will allow us to design different FRBMCSs dealing with several aspects such as improvement of the performance in terms of accuracy and obtaining a good accuracy-complexity trade-off.

This manuscript is organized as follows. In the next section, the preliminaries required to understand our work are reviewed. Section 3 briefly presents the proposed framework. Then, Section 4 introduces the proposed FRBMCS design methods, while Section 5 describes evolutionary the classifier combination designs. Each subsection in the latter section will introduce different approaches, referring the author to the corresponding publication, as well as reporting a brief performance analysis considering wide experimentations developed on a large number of UCI datasets. Finally, Section 6 concludes this contribution, suggesting also some future research lines.

2 State of the Art

This section reports a state of the art about MCSs and fuzzy MCSs. We also review FURIA, a novel and good performing fuzzy rule-based classifier, which will be used as the component base classifier. Finally, we briefly describe genetic fuzzy systems, which is a fundamental tool for development of the component fuzzy classifier combination method presented in the current contribution.

2.1 Multiclassification Systems

MCS design is mainly based on two stages [10]: the learning of the component classifiers and the combination mechanism for the individual decisions provided by them into the global MCS output. Since a MCS is the result of the combination of the outputs of a group of individually trained classifiers, the accuracy of the finally derived MCS relies on the performance and the proper integration of these two tasks. The best possible situation for an ensemble is that where the individual classifiers are both accurate and fully complementary, in the sense that they make their errors on different parts of the problem space [3]. Hence, MCSs rely for their effectiveness on the “instability” of the base learning algorithm.

On the one hand, the correct definition of the set of base classifiers is fundamental to the overall performance of MCSs. Different approaches have been thus proposed to succeed on generating diverse component classifiers with uncorrelated errors such as data resampling techniques (mainly, bagging [11] and boosting [12]), specific diversity induction mechanisms (feature selection [2], diversity measures [13], use of different parameterizations of the learning algorithm, use of different learning models, etc.), or combinations between the latter two families, as the well known random forests approach [14].

On the other hand, the research area of combination methods is also very active due to the influential role of this MCS component. It does not only consider the issue of aggregating the results provided by all the initial set of component classifiers derived from the first learning stage to compute the final output (what is usually called *classifier fusion* [15, 16]). It also involves either locally selecting the best single classifier which will be taken into account to provide a decision for each specific input pattern (static or dynamic classifier selection [17]) or globally selecting the subgroup of classifiers which will be considered for every input pattern (overproduce-and-choose strategy [18]). Besides, hybrid strategies between the two groups have also been introduced [1]. In any case, the determination of the optimal size of the ensemble is an important issue for obtaining both the best possible accuracy in the test data set without overfitting it, and a good accuracy-complexity trade-off [19].

2.2 FURIA

Fuzzy Unordered Rules Induction Algorithm (FURIA) [20] is an extension of the state-of-the-art rule learning algorithm called RIPPER [21], considering the derivation of simple and comprehensible fuzzy rule bases, and introducing some new features. FURIA provides three different extensions of RIPPER:

– It takes an advantage of fuzzy rules instead of crisp ones. Fuzzy rules of FURIA are composed of a class C_j and a certainty degree CD_j in the consequent. The final form of a rule is the following:

$$\begin{array}{l}\text{Rule } R_j : \text{If } x_1 \text{ is } A_{j1} \text{ and } \ldots \text{ and } x_n \text{ is } A_{jn} \\ \text{then Class } C_j \text{ with } CD_j;\;\; j = 1, 2, ..., N.\end{array}$$

The certainty degree of a given example x is defined as follows:

$$CD_j = \frac{2\frac{D_T^{C_j}}{D_T} + \sum_{x \in D_T^{C_j}} \mu_r^{C_j}(x)}{2 + \sum_{x \in D_T} \mu_r^{C_j}(x)} \tag{1}$$

where D_T and $D_T^{C_j}$ stands for the training set and a subset of the training set belonging to the class C_j respectively. In this approach, each fuzzy rule makes a vote for its consequent class. The vote strength of the rule is calculated as the product of the firing degree $\mu_r^{C_j}(x)$ and the certainty degree CD_j. Hence, the fuzzy reasoning method used is the so-called voting-based method [22, 23].

- It uses unordered rule sets instead of rule lists. This change omits a bias caused by the default class rule, which is applied whenever there is an uncovered example detected.
- It proposes a novel rule stretching method in order to manage uncovered examples. The unordered rule set introduces one crucial drawback, there might appear a case when a given example is not covered. Then, to deal with such situation, one rule is generalized by removing its antecedents. The information measure is proposed to verify which rule to "stretch".

The interested reader is referred to [20] for a full description of FURIA.

2.3 Related Work on Fuzzy Multiclassification Systems

Focusing on fuzzy MCSs, only a few contributions for bagging fuzzy classifiers have been proposed considering fuzzy neural networks (together with feature selection) [24], neuro-fuzzy systems [4], and fuzzy decision trees [25, 26] as component classifier structures.

Especially worth mentioning is the contribution of Bonissone et al. [25]. This approach hybridizes Breiman's idea of random forests [14] with fuzzy decision trees [27]. Such resulting fuzzy random forest combines characteristics of MCSs with randomness and fuzzy logic in order to obtain a high quality system joining robustness, diversity, and flexibility to not only deal with traditional classification problems but also with imperfect and noisy datasets. The results show that this approach obtains good performance in terms of accuracy for all the latter problem kinds.

Some advanced GFS-based contributions should also be remarked. On the one hand, an FRBCS ensemble design technique is proposed in [28] considering some niching genetic algorithm (GA) [29] based feature selection methods to generate the diverse component classifiers, and another GA for classifier fusion by learning the combination weights. On the other hand, another interval and fuzzy rule-based ensemble design method using a single- and multiobjective genetic selection process is introduced in [30, 31]. In this case, the coding scheme allows an initial set of either interval or fuzzy rules, considering the use of different features in their antecedents, to be distributed among different component classifiers trying to make them as diverse as possible by means of two accuracy and one entropy measures. Besides, the same authors presented a previous proposal in [32], where an evolutionary multiobjective (EMO) algorithm generated a Pareto set of FRBCSs with different accuracy-complexity trade-offs to be combined into an ensemble.

2.4 Genetic Fuzzy Systems

Fuzzy systems, which are based on fuzzy logic, became popular in the research community, since they have ability to deal with complex, non-linear problems being too difficult for the classical methods [33]. Besides, its capability of knowledge extraction and representation allowed them to become human-comprehensible to some extent (more than classical black-box models) [8, 9].

The lack of the automatic extraction of fuzzy systems have attracted the attention of the computational intelligence community to incorporate learning capabilities to these kinds of systems. In consequence, a hybridization of fuzzy systems and GAs has become one of the most popular approaches in this field [34–37]. In general, genetic fuzzy systems (GFSs) are fuzzy systems enhanced by a learning procedure coming from evolutionary computation, i.e. considering any evolutionary algorithm (EA).

Fuzzy rule-based systems (FRBSs), which are based on fuzzy "IF-THEN" rules, constitute one of the most important areas of fuzzy logic applications. Designing FRBSs might be seen as a search problem in a solution space of different candidate models by encoding the model into the chromosome, as GAs are well known optimization algorithms capable of searching among large spaces with the aim of finding optimal (usually nearly optimal) solutions.

The generic coding of GAs provides them with a large flexibility to define which parameters/components of FRBS are to be designed [36]. For example, the simplest case would be a parameter optimization of the fuzzy membership functions. The complete rule base can also be learned. This capability allowed the field of GFSs to grow over two decades and to still be one of the most important topics in computational intelligence.

In the current contribution, we will relay on the GFS paradigm to define some of the proposed FRBMCS designs.

3 Proposal of the Framework

The main objective of this paper is to enable FRBCSs to deal with high dimensional datasets by means of different MCS approaches. Thus, we sketched a global framework containing several FRBMCSs designs. This framework is composed of two stages (see Fig. 1). The first one, called "component fuzzy classifier design from classical ML approaches", includes the use of FURIA to derive the component classifiers considering the classical MCS design approaches such as:

– *Static* approaches. From this family we incorporate classical MCS approaches to obtain accurate FRBMCSs such as bagging, feature selection, and the combination of bagging and feature selection. Thanks to the intrinsic parallelism of bagging they will also be time efficient.
– *Dynamic* approaches. From this family we employ the combination of bagging and random oracles (ROs) [38, 39], since ROs induce an additional diversity to the base classifiers, the accuracy of the final FRBMCSs is thus improved.

In [19], a study to determine the size of a parallel ensemble (e.g. bagging) by estimating the minimum number of classifiers that are required to obtain stable aggregate predictions was shown. The conclusion drawn was that the optimal ensemble size is very sensitive to the particular classification problem considered. Thus, the second stage of our framework, called "Evolutionary component classifier combination", is related to post-processing of the generated ensemble by means of EAs to perform component classifier combination. All the approaches used consider classifier selection and some of them also combine it with classifier fusion.

Of course, the second stage follows the approaches from the first stage. This is indicated by a red arrow in the figure, showing exactly which approach is used for the FRBMCS design (Stage 1) together with its corresponding evolutionary post-processing (Stage 2). A dashed red arrow points out a proposal that was not developed and is left for the future works.

The second stage includes the following evolutionary component classifier selection designs:

– *Classifier Selection.* Within this family, we opted for a EMO overproduce-and-choose strategy (OCS) [18] (also known as test-and-select methodology [40]) strategy, using the state-of-the-art NSGA-II algorithm [41], in order to obtain a good accuracy-complexity trade-off.
– *Classifier Selection and Fusion.* As a combination method joining both families, classifier selection and classifier fusion, we proposed the use of a GFS, which allows us to benefit from the key advantage of fuzzy systems, i.e., their interpretability.

4 Component Fuzzy Classifier Design Methods

4.1 Static Approaches: Bagging, Feature Selection, and Bagging with Feature Selection

In [42, 43] it was shown that a combination between bagging and feature selection composed a general design procedure usually leading to good MCS designs, regardless the classifier structure considered. Hence, we decided to follow that approach by integrating FURIA into a framework of that kind. Our aim was to combine the diversity induced by the MCS design methods and the robustness of the FURIA method in order to derive good performance FURIA-based FRBMCSs for high dimensional problems [44]. We also tried a combination of FURIA with bagging and feature selection separately in order to analyze which is the best setting for the design of FURIA-based FRBMCSs.

We considered three different types of feature selection algorithms: random subspace [2], mutual information-based feature selection (MIFS) [45], and the random-greedy feature selection based on MIFS and the GRASP approach [46].

The term *bagging* is an acronym of bootstrap aggregation and refers to the first successful method to generate MCSs proposed in the literature [11].

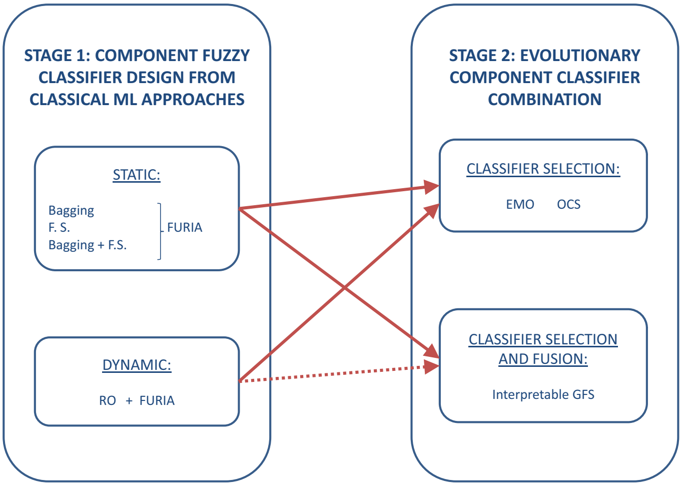


Fig. 1. The proposed framework is composed of several FRBMCSs design methodologies embedded into two stages: 1) FRBMCS design from classical ML approaches and 2) evolutionary component classifier combination

This approach was originally designed for decision tree-based classifiers, however it can be applied to any type of model for classification and regression problems. Bagging is based on bootstrap and consists of reducing the variance of the classification by averaging many classifiers that have been individually tuned to random samples that follow the sample distribution of the training set. The final output of the model is the most frequent value, called voting, of the learners considered. Bagging is more effective when dealing with unstable classifiers (the so-called "weak learners"), what means a small change in the training set can cause a significant change in the final model. In addition, it is recommended when the given dataset is composed of small amount of examples. Furthermore, bagging enables a parallel and independent learning of the learners in the ensemble.

Random subspace is a method in which a subset of features is randomly selected from the original dataset. Alternatively, the greedy Battiti's MIFS method is based on a forward greedy search using the mutual information measure [47], with regard to the class. This method orders a given set S of features by the information they bring to classify the output class considering the already selected features. The mutual information $I(C, F)$ for a given feature F is defined as:

$$I(C, F) = \sum_{c,f} P(c, f) \log \frac{P(c, f)}{P(c)P(f)} \tag{2}$$

where $P(c)$, $P(f)$ and $P(c, f)$ are respectively the values of the density function for the class, the feature variables, and the joint probability density. In the MIFS method, a first feature f is selected as the one that maximizes $I(C, f)$, and then the features f that maximize $Q(f) = I(C, f) - \beta \sum_{s \in S} I(f, s)$ are sequentially chosen until S reaches the desired size. β is a coefficient to reduce the influence of the information brought by the already selected features.

The random-greedy variant is an approach where the feature subset is generated by iteratively adding features randomly chosen from a restricted candidate list (RCL) composed of the best τ percent features according to the Q measure at each selection step. Parameter τ is used to control the amount of randomness injected in the MIFS selection. With $\tau = 0$, we get the original MIFS method, while with $\tau = 1$, we get the random subspace method.

FURIA-based FRBMCSs are designed as follows. A normalized dataset is split into two parts, a training set and a test set. The training set is submitted to an instance selection and a feature selection procedures in order to provide individual training sets (the so-called *bags*) to train FURIA classifiers. Let us emphasize that FURIA already incorporates an internal feature selection algorithm, being one of the features inherently owned from the RIPPER algorithm.

An exhaustive study was developed comparing all the variants proposed. We selected 21 datasets from the UCI machine learning repository [48] with different characteristics concerning the number of examples, features, and classes. For validation we used Dietterichs 5×2-fold cross-validation (5×2-cv) [49]. Three different feature subsets of different sizes (Small "S", Medium "M", and Large "L") were tested for the FURIA-based fuzzy MCSs using the three different feature selection algorithms. A small number of component fuzzy classifiers (up to 10) was considered in this study. Finally, the best choices of FURIA-based FRBMCSs were compared to two state-of-the-art MCS algorithms such as bagging decision trees and random forests, as well as with the use of the same methodology combined with a different fuzzy classifier generation method, Ishibuchi-based fuzzy MCS [7].

We show Table 4 presenting this final comparison, as the most representative results we have obtained. It consists of 5×2-cv training and test error values. For each algorithm, we only show the best obtained result in terms of accuracy for each dataset and highlight the best values in boldface. Random subspace and random-greedy feature selection are denoted as "R" and "RG", respectively.

The main conclusions obtained in [44] are as follows:

- A MCS framework based on a quick and accurate fuzzy classification rule learning algorithm, namely FURIA, can be competitive if not better than two state-of-the-art machine learning classifier ensembles such as random forests and C4.5 decision tree [50] MCSs generated from bagging [51].
- The proposed FURIA-based FRBMCSs are *accurate* and can be directly applied to high dimensional datasets, high in terms of large number of attributes, number of instances, and/or number of classes, thanks to the fact we use FURIA as a component classifier learning method.
- FURIA-based FRBMCSs with bagging clearly outperform FURIA-based FRBMCSs with feature selection and FURIA-based FRBMCSs with bagging and feature selection. Thus, it is the recommended MCSs combination method.

Table 1. A comparison of the best choice for different approaches for FURIA-based fuzzy MCSs against the best choice of bagging C4.5 MCSs, random forests, and Ishibuchi-based fuzzy MCSs

FURIA-based MCSs																					
	aba	bre	gla	hea	ion	let	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	0.753	**0.037**	0.313	**0.178**	0.134	0.091	0.136	**0.628**	**0.028**	**0.015**	0.136	**0.235**	0.105	0.035	**0.198**	0.061	**0.036**	0.276	**0.156**	**0.036**	**0.408**
feat sel.	G	R	-	-	RG	-	-	RG	R	R	R	RG	-	-	R	-	-	-	-	RG	-
feat.	L	L	-	-	S	-	-	L	L	L	L	L	-	-	L	-	-	-	-	M	-
sub. size																					
nr of cl.	10	10	7	7	7	10	7	10	10	10	10	10	10	10	10	10	10	10	10	10	10
C4.5 ensembles with bagging																					
	aba	bre	gla	hea	ion	let	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	0.772	0.043	0.306	0.194	0.149	0.103	**0.134**	0.697	0.030	0.028	0.131	0.253	0.112	0.042	0.247	0.067	0.051	0.289	0.193	0.097	0.415
nr of cl.	10	7	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
random forests																					
	aba	bre	gla	hea	ion	let	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	0.777	0.041	**0.282**	0.211	0.140	**0.080**	**0.134**	0.695	0.031	0.016	**0.119**	0.264	**0.104**	**0.034**	0.239	**0.060**	0.040	**0.269**	0.185	0.048	0.438
nr of cl.	7	7	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
Ishibuchi-based fuzzy MCSs																					
	aba	bre	gla	hea	ion	let	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	**0.751**	0.056	0.379	0.213	**0.129**	0.420	0.202	0.629	0.075	0.062	0.208	0.238	0.175	0.166	0.245	0.223	0.256	0.398	0.181	0.056	0.482
nr of cl.	3	7	7	10	7	10	7	3	7	10	3	7	7	10	0	10	7	3	7	10	7
feat. sel.	R	R	G	R	RG	RG	R	R	RG	R	G	G	RG	RG	RG	G	RG	RG	RG	G	G

The interested reader is referred to [44] for a deeper explanation of the presented approach.

4.2 Dynamic Approach: Bagging with Random Oracles

This section introduces the use of random oracles (ROs) [38, 39] within the bagging MCS framework to derive FURIA-based FRBMCSs. Our idea is that, thanks to the additional diversity introduced by ROs into the base classifiers, the obtained FRBMCSs are able to achieve an outstanding performance in terms of accuracy [52].

An RO is a structured classifier, also defined as a "mini-ensemble", encapsulating the base classifier of the MCS. It is composed of two subclassifiers and an oracle that decides which one to use in each case. Basically, the oracle is a random function whose objective is to randomly split the dataset into two subsets by dividing the feature space into two regions. Each of the two generated regions (together with the corresponding data subset) is assigned to one classifier. Any shape for the decision surface of the function can be applied as far as it divides the training set into two subsets at random.

Let us emphasize that during the classification phase, the oracle commits an internal dynamic classifier selection, that is to say it decides which subclassifier makes the final decision for the given example to be further used at the ensemble level (classifier fusion). Thus, this MCS method belongs to the *dynamic* family [17, 53].

The RO approach owns several interesting features, making it quite unique among the existing MCS solutions:

– It is a generic approach composing a framework in which ROs embed only the base classifier. Thus, it allows a design choice at two different levels: i) any MCS strategy can be applied; ii) any classifier learning algorithm can be used. Apart from that, it can be used as the MCSs generation method on its own.

- It induces an additional diversity through the randomness coming from the nature of ROs. Generating a set of diverse base classifiers was shown to be fundamental for the MCSs overall performance [3, 54]. Let us emphasize that ROs are applied separately to each of the base classifiers and no training of the oracle is recommended, as it will strongly diminish the desired diversity.
- It embeds the two most common and complementary MCS combination methods, i.e. *classifier fusion* and *(dynamic) classifier selection*.
- A wide study has been carried out over several MCS generation approaches [38, 39] in order to analyse the influence of ROs on these methods. C4.5 [50] (in [38]) and Naïve Bayes [55] (in [39]) were the base classifiers used. All the MCS approaches took an advantage of the ROs, outperforming the original MCSs in terms of accuracy. Especially, the highest accuracy improvement was obtained by random subspace and bagging according to [38].

In particular, we considered two versions of ROs: random linear oracle (RLO) [38, 39] and random spherical oracle (RSO) [39]. The former uses a randomly generated hyperplane to divide the feature space, while the latter does so using a hypersphere.

We selected 29 datasets with different characteristics concerning a high number of examples, features, and classes from the UCI machine learning [48] and KEEL [56] repositories. For validation, 5×2-cv was used. We studied the performance of both RO-based bagging FRBMCSs in comparison with bagging FRBMCSs considering both accuracy and complexity. Then, the best performing FRBMCSs were compared against state-of-the-art RO-based bagging MCSs. By doing so, we wanted to show that RO-based bagging FRBMCSs are competitive against the state-of-the-art RO-based bagging MCSs using C4.5 [38, 39] and Naïve Bayes [39] as the base classifiers, when dealing with high dimensional datasets, thanks to the use of the FURIA algorithm. Finally, we presented some kappa-error diagrams [57] to graphically illustrate the relationship between the diversity and the individual accuracy of the base classifiers among FRBMCSs.

For an illustrative purpose, we include Table 2 in the current contribution, reporting the test results achieved by RSO-based bagging FRBMCSs and RSO-based bagging MCS using C4.5 and NB over the 29 selected datasets.

We highlight the main conclusions drawn from the study developed in [52] as follows:

- Both RO-based bagging FRBMCSs show significant differences in comparison to bagging FRBMCSs considering accuracy, as well as complexity in terms of overall average number of rules. This happens due to the additional diversity induced by the ROs, which was clearly seen in the Kappa-error diagrams [57].
- RSO-based bagging FRBMCSs not only outperform classical RSO-based bagging MCSs using C4.5 and NB, but they also show a lower complexity in comparison to RSO-based bagging MCSs using C4.5. FURIA again turned out to be robust and accurate algorithm, belonging to the fuzzy rule-based classifier family, which obtains an outstanding performance in combination with classical MCS techniques.

Table 2. A comparison of RSO-based bagging MCSs using FURIA, C4.5, and NB in terms of accuracy

Dataset	FURIA Test err.	C4.5 Test err.	NB Test err.
abalone	**0.7472**	0.7696	0.7624
bioassay_688red	**0.0090**	**0.0090**	0.0153
coil2000	**0.0601**	0.0616	0.1820
gas_sensor	**0.0081**	0.0094	0.3003
isolet	**0.0727**	0.0813	0.1253
letter	0.0760	**0.0658**	0.2926
magic	0.1304	**0.1268**	0.2366
marketing	**0.6690**	0.6745	0.6875
mfeat_fac	**0.0461**	0.0501	0.0655
mfeat_fou	**0.1924**	0.1948	0.2205
mfeat_kar	0.0737	0.0867	**0.0597**
mfeat_zer	**0.2220**	0.2294	0.2473
musk2	0.0321	**0.0283**	0.1121
optdigits	**0.0289**	0.0297	0.0717
pblocks	0.0341	**0.0330**	0.0705
pendigits	**0.0136**	0.0161	0.0861
ring_norm	0.0326	0.0397	**0.0202**
sat	0.1007	**0.0967**	0.1731
segment	**0.0296**	0.0326	0.1198
sensor_read_24	**0.0231**	0.0232	0.3703
shuttle	**0.0009**	**0.0009**	0.0157
spambase	**0.0640**	0.0658	0.1777
steel_faults	0.2379	**0.2286**	0.3429
texture	**0.0280**	0.0351	0.1426
thyroid	0.0218	**0.0215**	0.0393
two_norm	0.0288	0.0327	**0.0222**
waveform	**0.1482**	0.1698	0.1672
waveform1	**0.1459**	0.1654	0.1541
wquality_white	0.3825	**0.3737**	0.5216
Avg.	**0.1312**	0.1357	0.2068
Std. Dev.	0.1819	0.1856	0.1892

5 Evolutionary Component Classifier Combination

5.1 Evolutionary Multiobjective Overproduce-and-Choose Static Classifier Selection

In this section, we describe our proposal of an EMO method defining an OCS strategy for the component classifier selection [58]. Our goal is to obtain a good accuracy-complexity trade-off in the FURIA-based FRBMCSs when dealing with high dimensional problems. That is, we aim to obtain FRBMCSs with a low number of base classifiers, which jointly keep a good accuracy. Thus, we have selected the state-of-the-art NSGA-II EMO algorithm [41] in order to generate good quality Pareto set approximations.

NSGA-II is based on a Pareto dominance depth approach, where the population is divided into several fronts and the depth of each front shows to which front an individual belongs to. A pseudo-dominance rank being assigned to each individual, which is equal to the front number, is the metric used for the selection of an individual.

We have used a standard binary coding in such a way that a binary digit/gene is assigned to each classifier. When the variable takes value 1, it means that the current component classifier belongs to the final ensemble, while when the variable is equal to 0, that classifier is discarded. This approach provides a low operation cost, which leads to a high speed of the algorithm.

Five different biobjective fitness functions combining the three existing kinds of optimization criteria (accuracy, complexity, and diversity) are proposed in order to study the best setting. We use the following measures: the training error (accuracy), the number of classifiers (complexity), and the difficulty measure θ and the double fault δ (diversity). Table 3 presents the five combinations proposed.

Table 3. The five fitness function proposed

1st obj.	2nd obj.
TE	Complx
TE	θ
TE	δ
θ	Complx
δ	Complx

The initial fuzzy classifier ensembles are based on applying a bagging approach with the FURIA method as described in Section 4.1. Each FRBMCS so generated is composed of 50 weak learners.

We carried out an experiment comparing all five biobjective fitness functions. We have selected 20 datasets from the UCI machine learning repository with different characteristics concerning the number of examples, features, and classes. To compare the Pareto front approximations of the global learning objectives (i.e. MCS test accuracy and complexity) we considered two of the usual kinds of multiobjective metrics, namely hypervolume ratio (HVR) [59] and C-measure [60], respectively. We also analyzed single solutions extracted from the obtained Pareto front approximations.

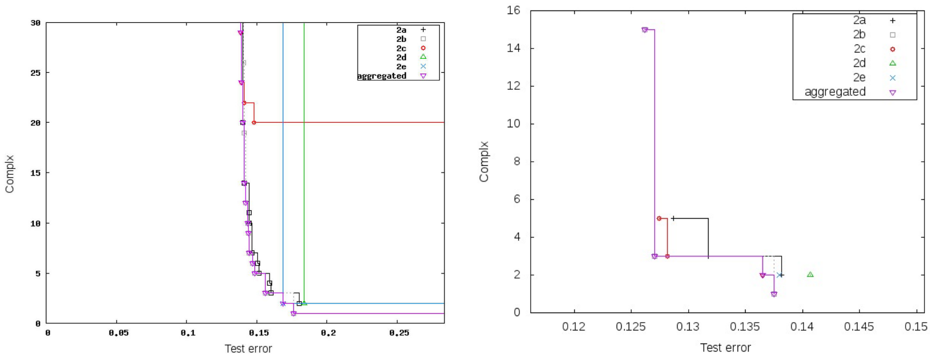
In Table 4, we show a representative comparison for this study. FURIA-based fuzzy MCSs are comprised by 7 or 10 classifiers, the small ensemble sizes providing the best results in our previous contribution [44] (see Section 4.1), and with 50 classifiers, the initial structure of the EMO-selected fuzzy MCSs. We also compare them with two state-of-the-art algorithms, random forests [14] and bagging C4.5 MCSs [50], comprised by 7 or 10 classifiers [44]. Besides, for illustration purposes, the aggregated Pareto fronts are represented graphically for the magic and waveform datasets in Figure 2, which allows an easy visual comparison of the performance of the different EMO OCS-based FRBMCSs variants.

The main conclusions drawn from the study developed are as follows [58]:

– Comparing Pareto Fronts using the HVR metric, the fitness function composed of training error (accuracy) and variance (diversity) clearly reported the best performance, while combining variance (diversity) with the number of classifiers (complexity) and double fault (diversity) with the number of classifiers (complexity) turned out to be deceptive combinations. To make a fair comparison, the reference Pareto Fronts, that is to say those based on test error and the number of classifiers, were considered.
– NSGA-II bagging FURIA-based FRBMCSs turned out to be competitive with the static bagging FURIA-based FRBMCSs and classical MCSs such as random forests and bagging C4.5 decision trees in terms of accuracy.

Table 4. A comparison of the NSGA-II FURIA-based fuzzy MCSs against static FURIA-based MCS

NSGA-II combined with FURIA-based MCSs.																				
	aba	bre	gla	hea	ion	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	**0.741**	**0.037**	0.283	**0.170**	**0.126**	**0.132**	**0.625**	**0.027**	**0.014**	**0.125**	**0.231**	**0.101**	**0.027**	**0.188**	**0.056**	**0.028**	**0.255**	**0.146**	**0.018**	**0.396**
fit. func.	2b	2b	2c	2b	2c	2a	2b	2c	2c	2c	2e	2b	2c	2e	2b	2c	2b	2c	2c	2b
# cl.	18.6	2.7	5.5	2	18.7	5.6	26	4.8	21.8	9	2	14.6	17.6	2	6.8	23.2	7.5	18.7	18.7	7.1
FURIA-based MCSs algorithms Small ensemble sizes.																				
	aba	bre	gla	hea	ion	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	0.753	**0.037**	0.313	0.178	0.134	0.136	0.628	0.028	0.015	0.136	0.235	0.105	0.035	0.198	0.061	0.036	0.276	0.156	0.036	0.408
# cl.	10	10	7	7	7	7	10	10	10	10	10	10	10	10	10	10	10	10	10	10
FURIA-based MCSs algorithms. Ensemble size 50.																				
	aba	bre	gla	hea	ion	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	0.748	0.041	0.287	0.182	0.145	0.135	0.630	0.028	0.016	0.135	0.241	0.102	0.034	0.226	0.059	0.031	0.275	0.149	0.035	0.400
C4.5 ensembles with bagging. Small ensemble sizes.																				
	aba	bre	gla	hea	ion	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	0.772	0.043	0.306	0.194	0.149	0.134	0.697	0.03	0.028	0.131	0.253	0.112	0.042	0.247	0.067	0.051	0.289	0.193	0.097	0.415
# cl.	10	7	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
Random forests. Small ensemble sizes.																				
	aba	bre	gla	hea	ion	mag	opt	pbl	pen	pho	pim	sat	seg	son	spa	tex	veh	wav	win	yea
test err.	0.777	0.041	**0.282**	0.211	0.14	0.134	0.695	0.031	0.016	0.119	0.264	0.104	0.034	0.239	0.06	0.04	0.269	0.185	0.048	0.438
# cl.	7	7	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10



(a) (b)

Fig. 2. The Pareto front approximations obtained for two datasets using the five fitness functions: (a) waveform and (b) magic. Objective 1 stands for test error and objective 2 for complexity. The pseudo-optimal Pareto front is also drawn for reference.

– NSGA-II combined with FURIA-based FRBMCSs is a good approach to obtain high quality, well performing ensembles with a good accuracy-complexity trade-off, when dealing with high dimensional datasets.

5.2 Joint Classifier Selection and Fusion via an Interpretable Genetic Fuzzy System

The aim of the current section is to present a fuzzy linguistic rule-based classification system playing the role of MCS combination method (a FRBCS-CM) [61]. Our design fulfills several requirements, namely: i) showing a human-understandable structure; ii) being able to deal with high dimensional problems avoiding the curse of dimensionality; iii) having the chance to be automatically learned from training data; and iv) being able to perform both classifier fusion and selection in order to derive low complexity fuzzy classifier ensembles with a good accuracy-complexity trade-off [1].

Using the novel FRBCS-CM together with a fuzzy classifier ensemble, we have the additional advantage of handling a two-level hierarchical structure composed of the individual classifiers in the first level and the FRBCS-CM in the second. These kinds of hierarchical structures [62–65] are well known in the area as they allow fuzzy systems to properly deal with high-dimensional problems while maintaining their descriptive power, especially when considering the single-winner rule fuzzy reasoning method in the component fuzzy classifiers as done in our case.

One step further, using it in combination with a bagging fuzzy classifier ensemble strategy as done in this proposal, we can also benefit from some collateral advantages for the overall design of the FRBMCS: a) the simplicity of the implicit parallelism of bagging, which allows for an easy parallel implementation; and b) the problem partitioning due to the internal feature selection at the component classifier level and the classifier selection capability of the fuzzy linguistic combination method, resulting in a tractable dimension for learning fuzzy rules for each individual classifier and for achieving a compact fuzzy classifier ensemble. These characteristics make the fuzzy ensemble using the FRBCS-CM specially able to deal with the curse of dimensionality.

Our approach might thus be assigned to the stacking (or stacked generalization) group [66], which after bagging and boosting is probably the most popular approach in the literature. Its basis lay in the definition of the meta-learner, playing a role of (advanced) MCS combination method, giving a hierarchical structure of the ensemble. Its task is to gain knowledge of whether training data have been properly learned and to be able to correct badly trained base classifiers. The FRBCS-CM proposed acts as the meta-learner, by discarding the rule subsets in the base fuzzy classifiers providing incorrect decisions at individual class level and promoting the ones leading to a correct classification.

Moreover, fuzzy classification rules with a class and a certainty degree in the consequent used in FRBCS-CM allows the user to get an understandable insight to the MCS. This means that this approach allows interpretability (to some extent) of such complicated system.

The proposed FRBCS-CM is built under the GFS approach (in particular, being an interpretable GFS). A specific GA, which uses a sparse matrix to codify features and linguistic terms in the antecedent parts of the rules and a fitness function based on three accuracy components performs both classifier fusion and classifier selection at class level. The complexity of the final ensemble, defined by the number of terms in the sparse matrix different than zero ("nonzero value"), which is a designed parameter provided by the user.

To evaluate the performance of the FRBCS-CM in the ensembles generated, 20 popular datasets from the UCI machine learning repository have been selected with a number of features varying from a small value (i.e., 5) to a large one

[1] We should remind that the proposed combination method can be applied to any multiclassification system with the only restriction that the component classifiers must additionally provide certainty degrees associated to each class in the dataset.

(i.e., 64), while the number of examples scales from 208 to 19 020. In order to compare the accuracy of the considered classifiers, we used 5×2-cv. This study was carried in a three-fold manner. Firstly, we compared bagging FRBMCSs combined with our interpretable GFS performing classifier selection and fusion over bagging FRBMCSs with the full ensemble using standard majority voting (MV). Secondly, we compared the novel interpretable GFS with state-of-the-art crisp and fuzzy multiclassification combination methods, as well as with a hybrid method based on GA considering both classifier selection and classifier fusion [67]. Finally, we showed some interpretability aspects of the proposed fuzzy linguistic combination method.

For the comparison, apart from the standard MV, we select average (AVG) [1] and decision templates (DT) [68] based on Euclidean distance, as crisp and fuzzy fusion methods respectively, being the best methods of each group according to Kuncheva [69]. Since the proposed FRBCS-CM includes classifier selection and classifier fusion, we also apply classifier selection with the mentioned classifier fusion methods in order to make a fair comparison. To select classifiers we will use two standard greedy approaches, Greedy Forward Selection (FS) and Greedy Backward Selection (BS) [70], which will use the abovementioned classifier fusion methods (these methods are also used to guide the search of the greedy algorithms). The hybrid method based on GA proposed in [67] (GA-Dimililer) embeds both classifier selection and classifier fusion, thus we directly apply it without any modifications.

For illustrative purpose, Tables 5 and 6 present a comparison between FRBCS-CM (interpretable GFS) and the other MCS combination methods in terms of accuracy and complexity, respectively. Table 5 shows the test error obtained for MV (operating on the full original ensemble), FRBCS-CM (nonzero values: 10%, 25%, 50%, 75%, and 90%), Greedy FS with MV, AVG, and DT, Greedy BS with MV, AVG, and DT, and GA-Dimililer. Then, Table 6 reports the total number of rules in the ensembles considering the same approaches. The comparison was conducted with respect to the complexity of the obtained FRBMCSs. For example, FRBCS-CM with nonzero values 10% and 25% were compared to Greedy FS with MV, AVG, and DT.

The experiments conducted in this study allowed us to obtain the following conclusions [61]:

- Bagging FRBMCSs combined with the interpretable GFS obtain good results in comparison with bagging FRBMCSs with the full ensemble using standard MV. Apart from obtaining good performance in terms of accuracy, it is also very competitive in terms of complexity reduction, after the selection of the component classifiers. We notice that, the final results highly depends on the parameter defining the complexity of the FRBCS-CM, which leads to different accuracy-complexity trade-offs.
- Our approach turned out to be competitive with the algorithms compared in terms of accuracy, while showing low complexity of the FRBMCSs obtained. Notice that, we aimed to propose a MCS combination method providing a good accuracy-complexity trade-off.

Table 5. Accuracy of the fuzzy MCSs, FRBCS-CM, and the other MCS combination methods in terms of test error

Dataset	fuzzy MCSs	FRBCS-CM 10%	25%	50%	75%	90%	Greedy FS MV	AVG	DT	Greedy BS MV	AVG	DT	GA Dimil.
Low dim.:													
abalone	0.7458	0.7581	0.7537	0.7493	0.7470	0.7461	0.7524	0.7582	0.7610	0.7484	0.7524	0.7511	0.7494
breast	0.0409	0.0472	0.0469	0.0452	0.0438	0.0432	0.0455	0.0418	0.0398	0.0412	0.0386	0.0372	0.0409
glass	0.2822	0.3159	0.2879	0.2832	0.2692	0.2710	0.2981	0.3271	0.3000	0.2832	0.2720	0.2776	0.3131
heart	0.1822	0.1785	0.1733	0.1719	0.1696	0.1696	0.1859	0.2015	0.1874	0.1778	0.1770	0.1674	0.1726
magic	0.1346	0.1340	0.1314	0.1309	0.1302	0.1300	0.1329	0.1328	0.1323	0.1338	0.1326	0.1298	0.1336
pblocks	0.0288	0.0285	0.0265	0.0271	0.0268	0.0261	0.0282	0.0302	0.0296	0.0286	0.0269	0.0263	0.0402
phoneme	0.1332	0.1277	0.1252	0.1261	0.1256	0.1264	0.1260	0.1232	0.1258	0.1291	0.1271	0.1248	0.1301
pima	0.2385	0.2492	0.2484	0.2411	0.2432	0.2424	0.2503	0.2516	0.2596	0.2385	0.2375	0.2414	0.2398
wine	0.0393	0.0461	0.0382	0.0303	0.0404	0.0393	0.0629	0.0551	0.0607	0.0393	0.0371	0.0360	0.0348
yeast	0.4008	0.4155	0.4054	0.3985	0.4034	0.4013	0.4116	0.4142	0.4189	0.4011	0.3978	0.4018	0.4116
Avg. Low	0.2227	0.2301	0.2237	0.2204	0.2199	0.2196	0.2294	0.2336	0.2315	0.2221	0.2199	0.2193	0.2266
High dim.:													
ionosphere	0.1459	0.1527	0.1413	0.1458	0.1430	0.1430	0.1584	0.1532	0.1646	0.1476	0.1430	0.1413	0.1464
optdigits	0.0329	0.0337	0.0327	0.0327	0.0318	0.0313	0.0367	0.0352	0.0351	0.0329	0.0284	0.0279	0.0721
pendigits	0.0156	0.0174	0.0152	0.0140	0.0140	0.0138	0.0171	0.0150	0.0162	0.0156	0.0129	0.0126	0.0160
sat	0.1021	0.1067	0.1027	0.0997	0.0986	0.1005	0.1044	0.1010	0.1005	0.1022	0.0967	0.0971	0.1040
segment	0.0336	0.0334	0.0319	0.0304	0.0316	0.0302	0.0318	0.0326	0.0336	0.0330	0.0309	0.0306	0.0345
sonar	0.2269	0.2404	0.2183	0.2077	0.2077	0.2058	0.2163	0.2337	0.2452	0.2260	0.2183	0.2163	0.2231
spambase	0.0587	0.0569	0.0559	0.0555	0.0539	0.0546	0.0576	0.0573	0.0574	0.0579	0.0554	0.0549	0.0574
texture	0.0307	0.0343	0.0312	0.0304	0.0291	0.0285	0.0343	0.0330	0.0336	0.0308	0.0268	0.0270	0.0325
vehicle	0.2726	0.2773	0.2664	0.2690	0.2664	0.2674	0.2671	0.2690	0.2693	0.2723	0.2641	0.2600	0.2721
waveform	0.1492	0.1554	0.1490	0.1503	0.1489	0.1479	0.1508	0.1535	0.1533	0.1498	0.1468	0.1472	0.1532
Avg. High	0.1068	0.1108	0.1045	0.1036	0.1025	0.1023	0.1075	0.1084	0.1109	0.1068	0.1023	0.1015	0.1111
Avg. All	0.1647	0.1704	0.1641	0.1620	0.1612	0.1609	0.1684	0.1710	0.1712	0.1644	0.1611	0.1604	0.1689

Table 6. Complexity of the fuzzy MCSs, FRBCS-CM, and the other MCS combination methods in terms of the number of rules

Dataset	fuzzy MCSs	FRBCS-CM 10%	25%	50%	75%	90%	Greedy FS MV	AVG	DT	Greedy BS MV	AVG	DT	GA Dimil.
Low dim.:													
abalone	3990.9	398.2	995.7	1996.9	2983.6	3578.4	1211.0	1047.6	1037.7	2711.3	3306.9	3398.7	2391.9
breast	435.2	46.1	110.9	217.0	326.2	391	33.0	25.7	24.1	415.9	426.6	427.4	221.1
glass	590.3	57.4	140.6	289.9	434.4	528	88.7	43.6	54.7	560.5	576.4	577.5	173.8
heart	466.0	49.4	120.3	235.3	352.6	421	48.9	35.7	33.4	444.6	455.7	454.7	221.1
magic	3882.1	421.0	968.3	1965.6	2969.9	3475.8	528.2	424.6	417.3	2247.8	3203.6	3319	2123.6
pblocks	1329.4	131.2	328.9	628.1	967.8	1182.2	248.2	108.9	106.1	1259	1288	1297.3	314.1
phoneme	2197.3	241.7	587.8	1132.5	1679.0	2000	493.2	381.1	339.4	1442.8	2046	2049.4	996.9
pima	1050.9	110.9	260.7	530.1	782.4	946	239.3	149.4	118.1	957	1025	1027.7	530
wine	231.4	23.7	57.9	116.4	172.7	208	9.1	6.8	6.2	222.4	226.9	226.9	71.2
yeast	2449.0	260.8	630.9	1198.4	1825.1	2198.4	511.5	389.5	434.9	1901.3	2296.7	2291.9	902.4
Avg. Low	1662.3	174.0	420.2	831.0	1249.4	1492.8	341.1	261.3	257.2	1216.3	1485.2	1507.1	794.6
High dim.:													
ionosphere	367.7	37.8	95.4	211.0	279.8	334	27.0	22.2	24.4	353.3	361.2	360.6	190.3
optdigits	3584.6	359.2	893.5	1787.7	2678.8	3227.2	652.7	428.7	423.7	3398.5	3513.8	3513.1	661.5
pendigits	4395.3	448.8	1098.1	2208.7	3299.9	3964.3	892.1	569.8	470.8	4167.2	4306.4	4307.5	1874.6
sat	4207.2	427.2	1046.9	2107.2	3128.1	3762.8	1214.0	728.7	800.6	3575.2	4006.8	4055	1431.9
segment	1175.3	130.1	290.9	593.4	876.9	1051.4	165.6	109.2	86.7	1100.5	1151.3	1151.4	414.2
sonar	319.3	32.4	80.4	162.0	240.0	288	24.4	22.9	19.8	306.4	312.1	311.9	158.8
spambase	2220.9	229.0	557.2	1115.5	1661.7	2002.6	340.7	286.1	292.8	2135.5	2152.4	2139.8	1026
texture	2912.2	300.1	716.6	1458.8	2175.0	2610.9	433.6	333.8	352.5	2759.8	2852.9	2852.8	1240.4
vehicle	1415.3	154.3	380.4	735.3	1075.3	1283	364.1	173.3	193.4	1304.7	1387.6	1380	425.7
waveform	3484.3	354.0	861.5	1749.8	2601.2	3137.6	1355.9	753.1	727.1	3125.9	3408.3	3381.1	828.9
Avg. High	2408.2	247.3	602.1	1212.9	1801.7	2166.1	547.0	342.8	339.2	2222.7	2345.3	2345.3	825.2
Avg. All	2035.2	210.7	511.1	1022.0	1525.5	1829.4	444.1	302.0	298.2	1719.5	1915.2	1926.2	809.9

- This proposal allows the user to estimate the reduction of the complexity of the final MCS *a priori* by selecting the appropriate non zero parameter value. This high flexibility, an *a priori* choice of how simple will the MCS obtained be, constitutes an advantage over the compared approaches.
- We showed that the proposed fuzzy linguistic combination method provides a good degree of interpretability to the MCS, making the combination method operation mode more transparent for the user. Furthermore, when combined with a FRBMCS, the whole system takes a pure hierarchical structure based on fuzzy classification rules structure (in the sense that the weak learners constitute individual FRBCSs becoming the input to the FRBCS-based combination method). The type of rules with a class and a certainty degree in the consequent used in FRBCS-CM allows the user to get an understandable insight to the MCS, thus allowing interpretability of such complicated system to some extent.

5.3 Evolutionary Multiobjective Overproduce-and-Choose Dynamic Classifier Selection

This section presents an OCS strategy for the classifier selection of our *dynamic* FRBMCSs, the RSO-based bagging FRBMCSs (see Section 4.2). On the one hand, the aim is again to refine the accuracy-complexity trade-off in the RSO-based bagging FRBMCSs when dealing with high dimensional classification problems. On the other hand, an interesting objective is to study whether the additional diversity induced by RSOs is beneficial for the EMO OCS-based FRBMCSs. Thus, we have again chosen the state-of-the-art NSGA-II EMO algorithm in order to generate good quality Pareto set approximations.

In this study [52], we take one step further and use a three-objective fitness function combining the three existing kinds of optimization criteria: accuracy, complexity, and diversity. We use the following measures: the training error (accuracy), the total number of fuzzy rules in the ensemble (complexity), and the difficulty measure θ (diversity). Notice that, in order to make a fair comparison, we consider the final complexity in terms of the total number of rules instead of the total number of classifiers, since RSO-based classifiers produce twice as much classifiers and usually they are less complex than a standard base classifier.

RSO offers a tremendous advantage over a standard component classifier, because each classifier can be independently selected within each pair component. Because of that, our classifier selection is done at the level of the component classifiers and not at the whole pair of classifiers. A specific coding scheme, which permits that none, one, or both FURIA fuzzy subclassifiers can be selected, is introduced. We also develop a reparation operator, whose objective is to correct the unfeasible solutions.

We compared the proposed NSGA-II for RSO-based bagging FRBMCSs classifier selection with the standard NSGA-II using two different approaches from the first stage. Table 7 summarizes the three EMO OCS-based FRBMCSs approaches.

Table 7. The three EMO approaches used for the classifier selection

abbreviation	base classifier	MCS methodology	OCS strategy
2a	FURIA	bagging	standard NSGA-II
2b	RSO (2×FURIA+oracle)	bagging+RSO	standard NSGA-II
2c	RSO (2×FURIA+oracle)	bagging+RSO	proposed NSGA-II

We conducted exhaustive experiments considering 29 datasets with different characteristics concerning a high number of examples, features, and classes from the UCI [48] machine learning and KEEL [56] repositories. For validation we used 5×2-cv. To compare the Pareto front approximations of the global learning objectives (i.e. MCS test accuracy and complexity) we considered the most common multiobjective metric, HVR [59]. We also analyzed single solutions extracted from the obtained Pareto front approximations. We compared the three EMO variants in order to check whether the additional diversity induced by the RSO is beneficial to the performance of the final FRBMCS selected by the NSGA-II.

To give a brief view to the results obtained, Table 8 shows the average and standard deviation values for the four different solutions selected from each Pareto front approximation in the 29 problems. Besides, the aggregated Pareto fronts for the bioassay 688red dataset are represented graphically in Figure 3, which allows an easy visual comparison of the performance of the different EMO OCS-based FRBMCSs variants.

Table 8. A comparison of the averaged performance of the four single solutions selected from the obtained Pareto sets

		Best train			Best complx			Best trade-off			Best test		
	Card.	Tra	Tst	Cmpl	Tra	Tst	Cmpl	Tra	Tst	Cmpl	Tra	Tst	Cmpl
avg. 2a	40.1	0.0512	0.1321	1175	0.0920	0.1628	159	0.0673	0.1367	338	0.0543	0.1298	966
2b	40.3	0.0441	0.1315	1281	0.0920	0.1679	188	0.0612	0.1368	405	0.0480	0.1288	1078
2c	50.0	0.0442	0.1332	931	0.1516	0.2206	104	0.0745	0.1494	270	0.0469	0.1304	853
dev. 2a	43.1	0.1403	0.1829	2180	0.1643	0.1922	166	0.1514	0.1831	533	0.1449	0.1811	1897
2b	42.5	0.1231	0.1826	2164	0.1579	0.1921	188	0.1380	0.1827	574	0.1293	0.1808	2000
2c	32.4	0.1218	0.1841	1497	0.1454	0.1858	109	0.1417	0.1842	427	0.1246	0.1825	1434

From the wide study carried out we concluded that [52]:

– According to the HVR metric, the variant considering the RSO-based bagging FRBMCSs with the NSGA-II method proposed (2c) clearly outperformed the other approaches, mainly due to the low complexity of the final FRBMCSs. To make a fair comparison, the reference Pareto Fronts (based on test error and the number of classifiers) were considered.
– When selecting the best individual FRBMCS design according to the test error, the proposed approach is not significantly worst than the other variants in terms of accuracy, however it obtains a much lower complexity. On the other hand, the best individual FRBMCS design considering the complexity criterion is obtained by our approach, since it provides a solution with the lowest number of rules.

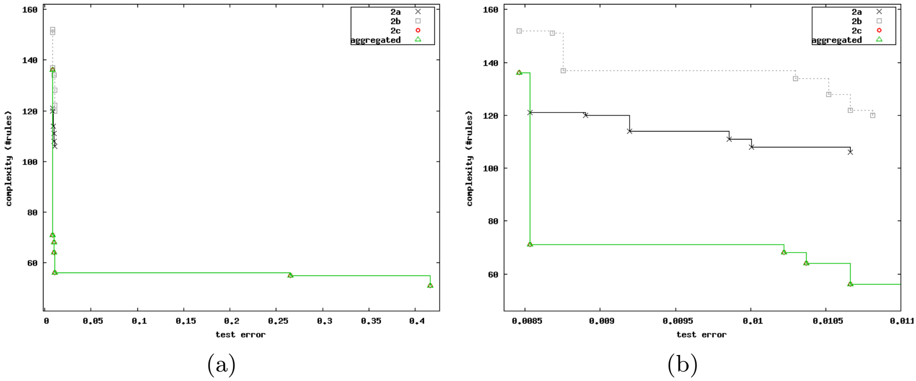


Fig. 3. The Pareto front approximations obtained from the three EMO approaches for three datasets: (a) bioassay_688red, (b) bioassay_688red (zoom). Objective 1 stands for test error and objective 2 for complexity in terms of number of rules. The pseudo-optimal Pareto front is also drawn for reference.

– In general, the additional diversity induced by the RSO have a positive influence on the final FRBMCSs selected by NSGA-II resulting in a strong reduction of complexity, while maintaining a similar accuracy. Thus, the diversity is beneficial for this kind of designs.

6 Conclusions and Future Work

We have proposed a global framework for FRBCS design in order to allow them dealing with high dimensional datasets. Our proposal is composed of different methods for component fuzzy classifier derivation, which consider several MCS methodologies, as well as evolutionary algorithms for classifier selection and fusion. We carried out exhaustive experiments for each component FRBMCS design. The results obtained have shown that we have reached the global goal. Besides, we obtained several sub-goals within the approaches proposed such as improvement of the performance in terms of accuracy and accuracy-complexity trade-off.

The promising results obtained lead to several research lines as future works. Combining bagging RO-based FRBMCSs with interpretable GFS for joint classifier selection and fusion is a future step to take into account. Besides, we will consider a combination of an EMO algorithm with interpretable GFS. Finally, we would like to apply the FRBMCS framework proposed to the real-world applications, consisting of complex and high dimensional classification problems. For instance, a topology-based WiFi indoor localization problem was already solved by one of our FRBMCS designs in [71].

Acknowledgements. This work was supported by the Spanish Ministerio de Economía y Competitividad under Project TIN2012-38525-C02-01, including funding from the European Regional Development Fund.

References

1. Kuncheva, L.I.: Combining Pattern Classifiers: Methods and Algorithms. Wiley (2004)
2. Ho, T.: The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(8), 832–844 (1998)
3. Optiz, D., Maclin, R.: Popular ensemble methods: An empirical study. Journal of Artificial Intelligence Research 11, 169–198 (1999)
4. Canul-Reich, J., Shoemaker, L., Hall, L.O.: Ensembles of fuzzy classifiers. In: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), London, pp. 1–6 (2007)
5. Pedrycz, W., Kwak, K.C.: Boosting of granular models. Fuzzy Sets and Systems 157(22), 2934–2953 (2006)
6. Cordón, O., Quirin, A.: Comparing two genetic overproduce-and-choose strategies for fuzzy rule-based multiclassification systems generated by bagging and mutual information-based feature selection. International Journal of Hybrid Intelligent Systems 7(1), 45–64 (2010)
7. Ishibuchi, H., Nakashima, T., Nii, M.: Classification and Modeling With Linguistic Information Granules. Springer (2005)
8. Casillas, J., Cordon, O., Herrera, F., Magdalena, L.: Interpretability Issues in Fuzzy Modeling. Springer, Heidelberg (2003)
9. Alonso, J.M., Magdalena, L., González-Rodríguez, G.: Looking for a good fuzzy system interpretability index: An experimental approach. International Journal of Approximate Reasoning 51, 115–134 (2009)
10. Dasarathy, B.V., Sheela, B.V.: A composite classifier system design: Concepts and methodology. Proceedings of IEEE 67(5), 708–713 (1979)
11. Breiman, L.: Bagging predictors. Machine Learning 24(2), 123–140 (1996)
12. Schapire, R.: The strength of weak learnability. Machine Learning 5(2), 197–227 (1990)
13. Tsymbal, A., Pechenizkiy, M., Cunningham, P.: Diversity in search strategies for ensemble feature selection. Information Fusion 6(1), 83–98 (2005)
14. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
15. Xu, L., Krzyzak, A., Suen, C.Y.: Methods of combining multiple classifiers and their application to handwriting recognition. IEEE Transactions on Systems, Man, and Cybernetics 22(3), 418–435 (1992)
16. Woods, K., Kegelmeyer, W.P., Bowyer, K.: Combination of multiple classifiers using local accuracy estimates. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(4), 405–410 (1997)
17. Giacinto, G., Roli, F.: Dynamic classifier selection based on multiple classifier behaviour. Pattern Recognition 34(9), 1879–1881 (2001)
18. Partridge, D., Yates, W.B.: Engineering multiversion neural-net systems. Neural Computation 8(4), 869–893 (1996)
19. Hernández-Lobato, D., Martínez-Muñoz, G., Suárez, A.: How large should ensembles of classi ers be? Pattern Recognition 46(5), 1323–1336 (2013)
20. Hühn, J.C., Hüllermeier, E.: FURIA: an algorithm for unordered fuzzy rule induction. Data Mining and Knowledge Discovery 19(3), 293–319 (2009)

21. Cohen, W.W.: Fast effective rule induction. In: Proceedings of the Twelfth International Conference on Machine Learning, pp. 115–123. Morgan Kaufmann (1995)
22. Ishibuchi, H., Nakashima, T., Morisawa, T.: Voting in fuzzy rule-based systems for pattern classification problems. Fuzzy Sets and Systems 103(2), 223–238 (1999)
23. Cordón, O., del Jesus, M.J., Herrera, F.: A proposal on reasoning methods in fuzzy rule-based classification systems. International Journal of Approximate Reasoning 20, 21–45 (1999)
24. Takahashi, H., Honda, H.: Lymphoma prognostication from expression profiling using a combination method of boosting and projective adaptive resonance theory. Journal of Chemical Engineering of Japan 39(7), 767–771 (2006)
25. Bonissone, P.P., Cadenas, J.M., Garrido, M.C., Díaz-Valladares, R.A.: A fuzzy random forest. International Journal of Approximate Reasoning 51(7), 729–747 (2010)
26. Marsala, C.: Data mining with ensembles of fuzzy decision trees. In: IEEE Symposium on Computational Intelligence and Data Mining, Nashville, USA, pp. 348–354 (2009)
27. Janikow, C.Z.: Fuzzy decision trees: issues and methods. IEEE Transactions on Systems, Man, and Cybernetics, Part B 28(1), 1–14 (1998)
28. Aguilera, J.J., Chica, M., del Jesus, M.J., Herrera, F.: Niching genetic feature selection algorithms applied to the design of fuzzy rule based classification systems. In: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), London, pp. 1794–1799 (2007)
29. Goldberg, D.E.: Genetic Algorithms in Search Optimization and Machine Learning. Addison-Wesley (1989)
30. Nojima, Y., Ishibuchi, H.: Designing fuzzy ensemble classifiers by evolutionary multiobjective optimization with an entropy-based diversity criterion. In: International Conference on Hybrid Intelligent Systems and Conference on Neuro-Computing and Evolving Intelligence, CD-ROM, 4 pages (2006)
31. Nojima, Y., Ishibuchi, H.: Genetic rule selection with a multi-classifier coding scheme for ensemble classifier design. International Journal of Hybrid Intelligent Systems 4(3), 157–169 (2007)
32. Ishibuchi, H., Nojima, Y.: Evolutionary multiobjective optimization for the design of fuzzy rule-based ensemble classifiers. International Journal of Hybrid Intelligent Systems 3(3), 129–145 (2006)
33. Yager, R.R., Filev, D.P.: Essentials of fuzzy modeling and control. Wiley-Interscience, New York (1994)
34. Cordón, O., Herrera, F., Hoffmann, F., Magdalena, L.: Genetic Fuzzy Systems. Evolutionary Tuning and Learning of Fuzzy Knowledge Bases. World Scientific (2001)
35. Cordón, O., Gomide, F., Herrera, F., Hoffmann, F., Magdalena, L.: Ten years of genetic fuzzy systems: Current framework and new trends. Fuzzy Sets and Systems 141(1), 5–31 (2004)
36. Herrera, F.: Genetic fuzzy systems: taxonomy, current research trends and prospects. Evolutionary Intelligence 1, 27–46 (2008)
37. Cordón, O.: A historical review of evolutionary learning methods for mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. International Journal of Approximate Reasoning 52(6), 894–913 (2011)
38. Kuncheva, L.I., Rodríguez, J.J.: Classifier ensembles with a random linear oracle. IEEE Transactions on Knowledge and Data Engineering 19(4), 500–508 (2007)

39. Rodríguez, J.J., Kuncheva, L.I.: Naïve bayes ensembles with a random oracle. In: Haindl, M., Kittler, J., Roli, F. (eds.) MCS 2007. LNCS, vol. 4472, pp. 450–458. Springer, Heidelberg (2007)
40. Sharkey, A.J.C., Sharkey, N.E.: The *test and select* approach to ensemble combination. In: International Workshop on Multiclassifier Systems, Cagliari, pp. 30–44 (2000)
41. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6, 182–197 (2002)
42. Panov, P., Džeroski, S.: Combining bagging and random subspaces to create better ensembles. In: Berthold, M., Shawe-Taylor, J., Lavrač, N. (eds.) IDA 2007. LNCS, vol. 4723, pp. 118–129. Springer, Heidelberg (2007)
43. Stefanowski, J.: An experimental study of methods combining multiple classifiers - diversified both by feature selection and bootstrap sampling. In: Atanassov, K.T., Kacprzyk, J., Krawczak, M., Szmidt, E. (eds.) Issues in the Representation and Processing of Uncertain and Imprecise Information, pp. 337–354. Akademicka Oficyna Wydawnicza EXIT, Warsaw (2005)
44. Trawiński, K., Cordón, O., Quirin, A.: On designing fuzzy rule-based multiclassification systems by combining furia with bagging and feature selection. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 19(4), 589–633 (2011)
45. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. IEEE Transactions on Neural Networks 5(4), 537–550 (1994)
46. Feo, T.A., Resende, M.G.C.: Greedy randomized adaptive search procedures. Journal of Global Optimization 6, 109–133 (1995)
47. Shannon, C.E., Weaver, W.: The Mathematical Theory of Communication. University of Illlinois Press (1949)
48. Blake, C.L., Merz, C.J.: UCI repository of machine learning databases (1998), `http://archive.ics.uci.edu/ml`
49. Dietterich, T.G.: Approximate statistical test for comparing supervised classification learning algorithms. Neural Computation 10(7), 1895–1923 (1998)
50. Quinlan, J.R.: C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco (1993)
51. Dietterich, T.G.: An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine Learning 40(2), 139–157 (2000)
52. Trawiński, K., Cordón, O., Sánchez, L., Quirin, A.: Multiobjective genetic classifier selection for random oracles fuzzy rule-based multiclassifiers: How benefical is the additional diversity? Technical Report AFE 2012-17, European Centre for Soft Computing, Mieres, Spain (2012)
53. Dos Santos, E.M., Sabourin, R., Maupin, P.: A dynamic overproduce-and-choose strategy for the selection of classifier ensembles. Pattern Recognition 41(10), 2993–3009 (2008)
54. Kuncheva, L.I., Whitaker, C.J.: Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine Learning 51(2), 181–207 (2003)
55. Domingos, P., Pazzani, M.J.: On the optimality of the simple bayesian classifier under zero-one loss. Machine Learning 29(2-3), 103–130 (1997)

56. Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S.: Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. Journal of Multiple-Valued Logic and Soft Computing 17(2-3), 255–287 (2011)
57. Margineantu, D.D., Dietterich, T.G.: Pruning adaptive boosting. In: Proceedings of the Fourteenth International Conference on Machine Learning, ICML 19897, pp. 211–218. Morgan Kaufmann Publishers Inc., San Francisco (1997)
58. Trawiński, K., Quirin, A., Cordón, O.: A study on the use of multi-objective genetic algorithms for classifier selection in furia-based fuzzy multiclassifers. International Journal of Computational Intelligence Systems 5(2), 231–253 (2012)
59. Coello, C.A., Lamont, G.B., van Veldhuizen, D.A.: Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd edn. Springer (2007)
60. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation 3, 257–271 (1999)
61. Trawiński, K., Cordón, O., Sánchez, L., Quirin, A.: A genetic fuzzy linguistic combination method for fuzzy rule-based multiclassifiers. IEEE Transactions on Fuzzy Systems (in press, 2013), doi:10.1109/TFUZZ.2012.2236844.
62. Torra, V.: A review of the construction of hierarchical fuzzy systems. International Journal of Intelligent Systems 17(5), 531–543 (2002)
63. Gegov, A.E., Frank, P.M.: Hierarchical fuzzy control of multivariable systems. Fuzzy Sets and Systems 72, 299–310 (1995)
64. Yager, R.R.: On the construction of hierarchical fuzzy systems model. IEEE Transactions on Systems, Man, and Cybernetics - Part B 28(1), 55–66 (1998)
65. Cordón, O., Herrera, F., Zwir, I.: A hierarchical knowledge-based environment for linguistic modeling: Models and iterative methodology. Fuzzy Sets and Systems 138(2), 307–341 (2003)
66. Wolpert, D.: Stacked generalization. Neural Networks 5(2), 241–259 (1992)
67. Dimililer, N., Varoglu, E., Altincay, H.: Classifier subset selection for biomedical named entity recognition. Applied Intelligence 31, 267–282 (2009)
68. Kuncheva, L.I., Bezdek, J.C., Duin, R.P.W.: Decision templates for multiple classifier fusion: An experimental comparison. Pattern Recognition 34(2), 299–314 (2001)
69. Kuncheva, L.I.: "Fuzzy" versus "nonfuzzy" in combining classifiers designed by boosting. IEEE Transactions on Fuzzy Systems 11(6), 729–741 (2003)
70. Ruta, D., Gabrys, B.: Classifier selection for majority voting. Information Fusion 6(1), 63–81 (2005)
71. Trawiński, K., Alonso, J.M., Hernández, N.: A multiclassifier approach for topology-based wifi indoor localization. Soft Computing (in press, 2013)