

An Improved Multi-Objective Genetic Algorithm for Tuning Linguistic Fuzzy Systems*

M. J. Gacto, R. Alcalá and F. Herrera

University of Granada, Dept. Computer Science and A.I., E-18071 Granada, Spain
mjgacto@ugr.es, {alcala, herrera}@decsai.ugr.es

Abstract

This work proposes the use of Multi-Objective Evolutionary Algorithms to obtain Fuzzy Rule-Based Systems with good accuracy-interpretability trade-off. To do this, we present a new post-processing method that performs rule selection and membership function tuning by focusing in the Pareto zone containing the most accurate solutions but with the least number of possible rules. This method is based on the well-known SPEA2 algorithm, applying an intelligent crossover operator, considering some modifications to concentrate the search in the desired Pareto zone and including an incest prevention mechanism in order to obtain more global optima. The results show that improving the trade-off between exploration and exploitation in the search process enhances the SPEA2 algorithm performance.

1 Introduction

A widely-used approach to improve the accuracy of linguistic Fuzzy Rule-Based Systems (FRBSs) is the *tuning* of Membership Functions (MFs) [2], which refines a previous definition of the data base once the rule base has been obtained. Although tuning usually

improves the system performance, sometimes a large number of rules is used to reach an acceptable degree of accuracy. To avoid this problem, some works [2] consider the selection of rules together with the tuning of MFs by only considering accuracy criteria. In this way, rules are extracted only if it is possible to maintain or even improve the system accuracy. A very interesting conclusion from [2] is that both techniques can present a positive synergy in most of the cases (similar or more accurate models could be obtained by reducing the number of rules) when they are combined within the same process.

Since this problem presents a multi-objective nature the use of Multi-Objective Evolutionary Algorithms (MOEAs)[4] to obtain a set of solutions with different degrees of accuracy and number of rules by using both characteristics as objectives can represent an interesting way to work. In fact, MOEAs have been recently applied to improve the difficult trade-off between interpretability and accuracy of linguistic FRBSs, by obtaining Mamdani type models not only accurate but also interpretable [3, 6, 10, 11]. However, these works do not consider learning or tuning [2] of the MFs, i.e., they are engaged to Linguistic Fuzzy Modeling (LFM) with improved interpretability which is closer to the interpretability than the accuracy (see Figure 1).

Indeed, to directly apply the most recognized MOEAs for general use in order to perform together tuning and rule selection could present some important problems [1]. The main problem is that it is practically impossible to ob-

*Supported by the Spanish Project TIN-2005-08386-C05-01.

tain the complete optimal Pareto due to the large search space and due to a faster tuning of the simplest solutions before exploring more promising rule configurations. However, by taking into account that non-dominated solutions with a small number of rules and high errors are not interesting since they have not the desired trade-off between accuracy and interpretability, we could focus the search only in the Pareto zone with the most accurate solutions trying to obtain the least possible number of rules.

In this way, a new method was recently proposed in [1], which by modifying the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [13] progressively concentrates the search in the most promising solutions, allowing exploration at first and favoring the exploitation of the most accurate solutions at the end (the Accuracy-Oriented SPEA2, SPEA2_{Acc}). The main objective was to get almost one improved solution with respect to the classic single objective algorithm (a solution that could dominate the one obtained by such algorithm in terms of the system error and the number of rules).

This work proposes a new MOEA (called Exploration-Exploitation based SPEA2, SPEA2_{E/E}) to improve even more the desired accuracy/interpretability balance (see Figure 1) by better handling the search space involved in the tuning of MFs and at the same time obtaining more compact models. This algorithm is based on the well known SPEA2 [13] (the same fitness computation and selection mechanism are used) and incorporates the main ideas of SPEA2_{Acc} proposed in [1] for guiding the search towards the desired Pareto zone. However, in order to improve the search ability of these algorithms the proposed algorithm includes a mechanism for incest prevention based on the concepts of CHC [8] (maintaining population diversity) and a more intelligent crossover operator. This favors a better trade-off between exploration and exploitation in the more complex search space of the real parameters and the derivation of more useful and simple offspring. The results obtained show that

improving the trade-off between exploration and exploitation in the search enhances the SPEA2 performance and outperforms the results obtained by SPEA2_{Acc}.

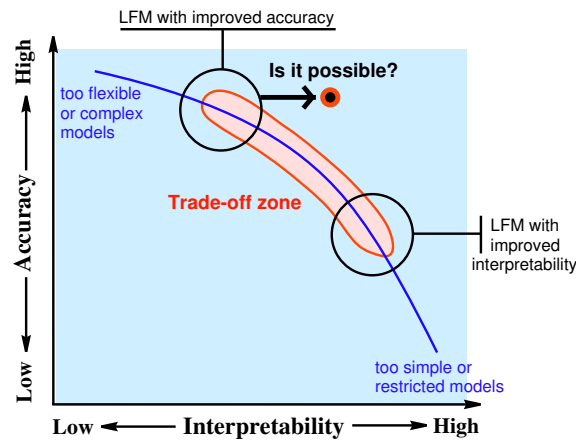


Figure 1: Interpretability-Accuracy Trade-off in LFM

In the next section, we propose the new SPEA2_{E/E} describing its main characteristics and the genetic operators considered. Section 3 shows an experimental study of the proposed method in a real-world problem. Finally, Section 4 points out some conclusions.

2 Using Incest Prevention to Evolve Accuracy-Oriented Pareto Sets: SPEA2_{E/E} Algorithm

This section proposes SPEA2_{E/E} to get solutions with higher accuracy and the least possible number of rules by performing rule selection together with a tuning of the MF parameters. In the next subsections, the main components of this algorithm are described and then the specific characteristics and its main steps are presented.

2.1 Coding Scheme and Initial Gene Pool

A double coding scheme for both *rule selection* (C_S) and *tuning* (C_T) is used:

$$C^p = C_S^p C_T^p$$

In the $C_S^p = (c_{S1}, \dots, c_{Sm})$ part, the coding scheme consists of binary-coded strings with

size m (with m being the number of initial rules). Depending on whether a rule is selected or not, values ‘1’ or ‘0’ are respectively assigned to the corresponding gene. In the C_T part, a real coding is considered, being m^i the number of labels of each of the n variables comprising the data base,

$$\begin{aligned} C_i &= (a_1^i, b_1^i, c_1^i, \dots, a_{m^i}^i, b_{m^i}^i, c_{m^i}^i), \\ i &= 1, \dots, n, \\ C_T^p &= C_1 C_2 \dots C_n. \end{aligned}$$

The initial population is obtained with all individuals having all genes with value ‘1’ in the C_S part. And in the C_T part the initial data base is included as first individual. The remaining individuals are generated at random within the corresponding variation intervals. Such intervals are calculated from the initial data base. For each MF, $C_i^j = (a^j, b^j, c^j)$, the variation intervals are calculated in the following way:

$$\begin{aligned} [I_{a^j}^l, I_{a^j}^r] &= [a^j - (b^j - a^j)/2, a^j + (b^j - a^j)/2] \\ [I_{b^j}^l, I_{b^j}^r] &= [b^j - (b^j - a^j)/2, b^j + (c^j - b^j)/2] \\ [I_{c^j}^l, I_{c^j}^r] &= [c^j - (c^j - b^j)/2, c^j + (c^j - b^j)/2] \end{aligned}$$

Besides, we have to highlight that the way to create the solutions of the initial population for the part of rule selection is a very important factor. In order to get solutions with a high accuracy, we should start with solutions selecting all the possible rules, which by means of the mutation at the beginning and then by means of the crossover favors a progressive extraction of bad rules (those that can not be improved).

2.2 Objectives

Two objectives are minimized for this problem: the number of rules (interpretability) and the Mean Squared Error (accuracy),

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

with $|E|$ being the data set size, $F(x^l)$ being the output obtained from the FRBS decoded from such chromosome when the l -th example is considered and y^l being the known desired

output. The fuzzy inference system considered to obtain $F(x^l)$ is the *center of gravity weighted by the matching* strategy as defuzzification operator and the *minimum t-norm* as implication and conjunctive operators.

2.3 Crossover and Mutation

In this subsection, we propose an intelligent crossover based on our experience in this concrete problem that is able to adequately profit from the corresponding parents. To obtain each offspring the following steps are applied:

1. BLX-0.5 [9] crossover is applied to obtain the C_T part of the offspring.
2. Once the real parameters are obtained determining a whole data base, for each gene in the C_S part the corresponding rule is independently extracted from each individual involved in the crossover (offspring and parents 1 and 2). In this way, the same rule is obtained three times with different MFs (those concerning these three individuals).
3. Euclidean normalized distances are computed between offspring and each parent by only considering the center points (vertex) of the MFs involved in the extracted rules. The differences between each pair of centers are normalized their respective variation interval amplitudes.
4. The nearest parent is the one that determines if this rule is selected or not for the offspring by directly copying its value in C_S for the corresponding gene.
5. This process is repeated until all the C_S values are assigned for the offspring.

Four offspring are obtained repeating this process four times (after considering mutation, only the two most accurate are taken as descendant). By applying this operator, exploration is performed in the C_T part and the C_S part is directly obtained based on the previous knowledge each parent has about the use or not of a specific configuration of MFs for each rule. This avoid to recover a bad rule that

was discarded for a concrete configuration of MFs, or allow to recover a good rule that is still considered for a concrete configuration of MFs, increasing the probability of succeed in the selection or elimination of a rule for each concrete configuration of MFs. Since a better exploration is performed for the C_S part, the mutation operator does not need to add rules. In this way, once an offspring is generated the mutation operator changes a gene value at random in the C_T part and directly sets to zero a gene selected at random in the C_S part (one gene is considered in each part) with probability P_m .

Applying these operators two problems are solved. Firstly, crossing individuals with very different rule configurations is more productive. And secondly, this way to work favors rule extraction since mutation is only engaged to remove unnecessary rules.

2.4 Main Characteristics and Steps of SPEA2_{E/E}

In order to focus the search on the desired Pareto zone, high accuracy with least possible number of rules, and to obtain a more global optimum we propose several mechanisms that give more selective pressure to those solutions that have a high accuracy and favor a better exploration for the tuning process:

- A restarting operator is applied by only maintaining the most accurate individual as a part of the new population (external population is forced to be empty) and obtaining the remaining individuals with the same rule configuration and tuning parameters generated at random within the corresponding variation intervals. In this way, we concentrate the search only in the desired Pareto zone (similar solutions in a zone with high accuracy) and get away from local optima or specific configurations in the C_T part.
- This algorithm includes an incest prevention mechanism based on the concepts of CHC [8] in order to avoid premature convergence in the C_T part (real cod-

ing), that is the main responsible of accuracy improvements and represents a more complicated search space than the C_S part (binary coding). In CHC, only those parents whose hamming distance divided by 4 is higher than a threshold are crossed. Since we consider a real coding scheme (only C_T parts are considered), we have to transform each gene considering a Gray Code with a fixed number of bits per gene ($BGene$) determined by the system expert. In this way, the threshold value is initialized as:

$$L = (\#C_T * BGene)/4,$$

where $\#C_T$ is the number of genes in the C_T part of the chromosome.

At each generation of the algorithm, the threshold value is decremented by one allowing to cross closer solutions.

- CHC performs restarting when L is 0 (moment in which similar solutions have been crossed to locally exploit the information in the parents). However, the transformation to Gray Code is a discretization and we detected all crossovers were allowed with still high values of L , obtaining when L is 0 very specific parameters that make difficult next restarts to get away from local optima.

Since each problem presents different values of L we propose to detect when all crossovers are allowed in order to perform restart. In fact, to avoid local convergence we apply first restart if 50 percent of crossovers are detected at any generation. This value is updated each time restarting is performed as $\%Required = (100 + \%Required)/2$. Moreover, the most accurate solution should be improved before each restarting. On the other hand, to preserve a well formed Pareto at the end, it is determined that restarting is not applied in the last evaluations. The number of evaluations without restart can be estimated as the number of evaluations needed to apply the first restart multiplied by 4. Additionally, restart is

disabled if it was never applied and the mid of the total number of evaluations are used. See Steps 5 and 7 in Figure 2.

- In each stage of the algorithm (between restarting points), the number of solutions in the external population (\bar{P}_{t+1}) considered to form the mating pool is progressively reduced, by focusing only on those with the best accuracy. To do that, the solutions are sorted from the best to the worst (considering accuracy as sorting criterion) and the number of solutions considered for selection is reduced progressively from 100% at the beginning to 50% at the end of each stage by taking into account the value of L .

Based on the descriptions of SPEA2 in [13], the main steps of SPEA2_{E/E} algorithm are finally presented in Figure 2.

3 Experiments

To evaluate the usefulness of the method proposed, we have considered a real-world problem [5] with 4 input variables that consists of estimating the maintenance costs of medium voltage lines in a town. SPEA2_{E/E} has been compared with several single objective based methods, with SPEA2, SPEA2_{Acc} and with the paradigm of MOEAs, NSGA-II [7] and a version of NSGA-II analyzed in [1] considering the same modifications of SPEA2_{Acc}. Methods considered for the experiments are briefly described in Table 1. WM [12] method is considered to obtain the initial rule base to be tuned. T and S methods perform the tuning of parameters and rule selection respectively. TS indicates tuning together with rule selection in the same algorithm. All of them consider the accuracy of the model as the sole objective. MOEAs considered perform rule selection from a given fuzzy rule set together with a tuning of the MFs considering two objectives, system error and number of rules.

The initial linguistic partitions are comprised by *five linguistic terms* with equally distributed triangular shape MFs. The values of the input parameters considered by S, T and

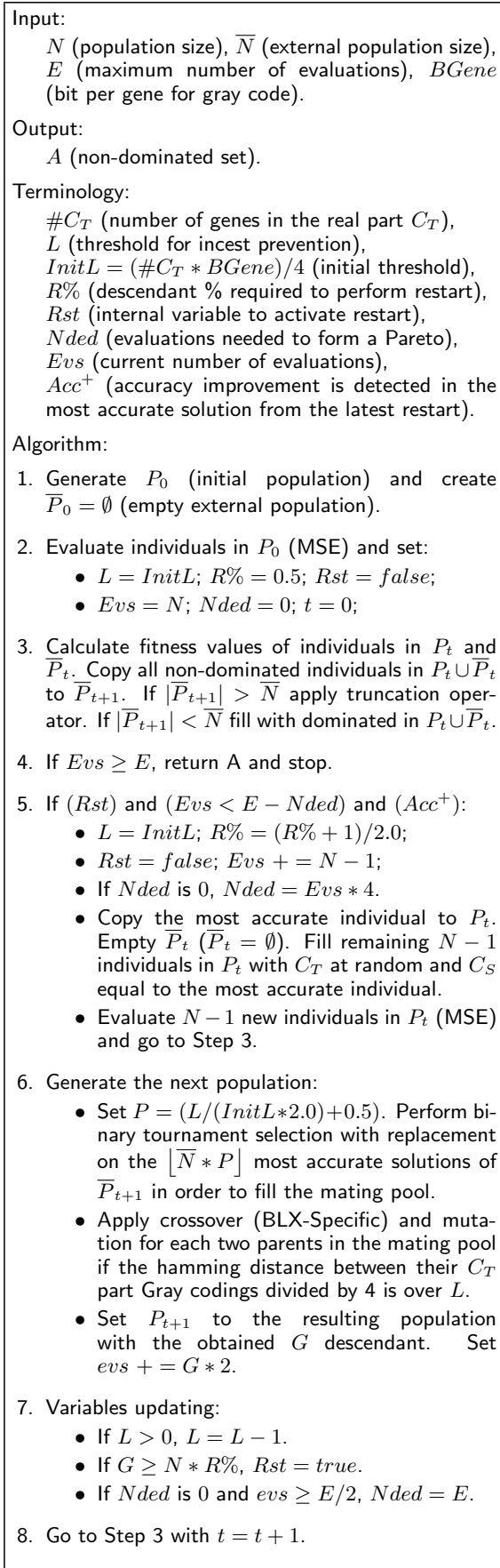


Figure 2: SPEA2_{E/E} Algorithm Scheme

Table 1: Methods Considered for Comparison

Method	Ref.	Description
WM	[12]	Wang & Mendel Algorithm
WM+T	[2]	Tuning of Parameters
WM+S	[2]	Rule Selection
WM+TS	[2]	Tuning and Rule Selection
WM+TS-NSGA-II	[1]	Tuning & Selection by NSGA-II
WM+TS-NSGA-II_{Acc}	[1]	Accuracy-Oriented NSGA-II
WM+TS-SPEA2	[1]	Tuning & Selection by SPEA2
WM+TS-SPEA2_{Acc}	[1]	Accuracy-Oriented SPEA2
WM+TS-SPEA2_{E/E}	—	Exploration/Exploitation based SPEA2

TS are: population size of 61, 100000 evaluations, 0.6 as crossover probability and 0.2 as mutation probability per chromosome. The values of the input parameters considered by the MOEAs are: population size of 200 (61 in the case of NSGA-II based algorithms), external population size of 61 (in the case of SPEA2 based approaches), 100000 evaluations, 0.2 as mutation probability per chromosome and 30 bits per gene for the Gray codification in SPEA2_{E/E}.

3.1 Problem Description

Estimating the maintenance costs of the medium voltage electrical network in a town [5] is a complex but interesting problem. Since a direct measure is very difficult to obtain, it is useful to consider models. These estimations allow electrical companies to justify their expenses. Moreover, the model must be able to explain how a specific value is computed for a certain town. Our objective will be to relate the *maintenance costs of the medium voltage lines* with the following four variables: *sum of the lengths of all streets in the town*, *total area of the town*, *area that is occupied by buildings*, and *energy supply to the town*. We will deal with estimations of minimum maintenance costs based on a model of the optimal electrical network for a town in a sample of 1,059 towns.

To develop the different experiments, we consider a *5-folder cross-validation model*, i.e., 5 random partitions of data each with 20%, and the combination of 4 of them (80%) as

training and the remaining one as test. For each one of the 5 data partitions, the tuning methods have been run 6 times, showing for each problem the averaged results of a total of 30 runs. In the case of methods with multi-objective approach (the last five), the averaged values have been calculated considering the most accurate solution from each Pareto obtained. In this way, since our main objective is to reduce the number of rules but maintaining or improving the accuracy of the obtained models we propose as final solution the most accurate one of each Pareto.

The results obtained by the analyzed methods are shown in Table 2, where $\#R$ stands for the number of rules, $MSE_{tra/tst}$ for the averaged error obtained over the training/test data, σ for their respective standard deviations and t for the results of applying a *test t-student* (with 95 percent confidence) in order to ascertain whether differences in the performance of the best results are significant when compared with that of the other algorithms in the table. A run of the algorithms in the table takes approximately 8 minutes in a Core 2 Duo 2GHz CPU (by only using one of the two processors). The interpretation of this column is:

- ★ represents the best averaged result.
- + means that the best result has better performance than that of the related row.

3.2 Results and Analysis

Analysing the results showed in table 2 we can highlight the following facts:

Table 2: Results obtained by the studied methods

Method	#R	MSE _{tra}	σ_{tra}	t-test	MSE _{tst}	σ_{tst}	t-test
WM	65	57605	2841	+	57934	4733	+
T	65	17020	1894	+	21027	4225	+
S	40.9	41158	1167	+	42988	4442	+
TS	41.3	13387	1153	+	17784	3344	+
TS-NSGAI	31.4	11827	1354	+	16047	4070	+
TS-NSGAI _{Acc}	36.4	11167	1464	+	14986	2769	+
TS-SPEA2	28.9	11630	1283	+	15387	3108	+
TS-SPEA2 _{Acc}	32.3	10714	1392	+	14252	3181	+
TS-SPEA2 _{E/E}	29.8	9688	942	*	12671	1796	*
TS-SPEA2 [†] _{E/E}	31.0	10276	974	=	13318	1999	=

[†] Results after 50000 evaluations

- The models obtained by SPEA2_{E/E} show an important reduction of the error in training and test with respect with those obtained by the others methods.
- A large number of rules have been removed from the initial rule base (more or less 35 rules have been eliminated). In this way, SPEA2_{E/E} improves the accuracy and obtain simpler models than TS and SPEA2_{Acc}, being the second method with less rules.
- Although it is natural to expect a slow convergence of SPEA2_{E/E} (forcing exploration before allowing exploitation), this method is faster than the remaining, presenting very good results even after 50000 evaluations.
- Moreover, SPEA2_{E/E} presents the lowest standard deviations in both, training and specially in test.

In Figure 3, we can see the Pareto evolution in a representative run with SPEA2_{Acc} and SPEA2_{E/E}, and also the evolution of the best solution in the population in a representative run of WM+TS. We can observe as the Pareto moves along without having a wide extension but dominating the solution obtained by WM+TS at the end. Pareto at 30000 in SPEA2_{E/E} represents a still not formed Pareto after a restarting point.

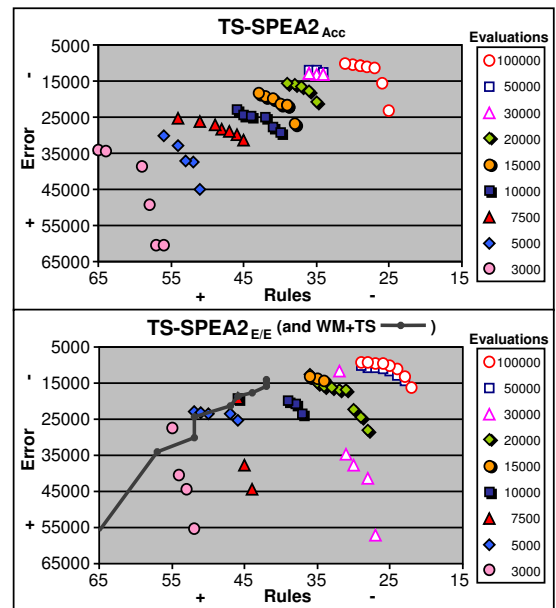


Figure 3: Pareto evolution with SPEA2_{Acc} & SPEA2_{E/E}, and of the best solution with TS.

4 Concluding Remarks

Taking into account the obtained results, we can conclude that the models obtained by the proposed method present a better trade-off between interpretability and accuracy than the remaining ones. By searching for a good configuration of rules and by tuning the parameters for a proper subset of rules, the proposed algorithm has obtained models even with a better accuracy than those obtained by methods only guided by accuracy measures. Moreover, the results show that to im-

prove the trade-off between exploration and exploitation by means of incest prevention in the search process allows the derivation of more global optima.

On the other hand, the proposed algorithm could be of interest in problems that considering real coding and although presenting a multi-objective nature, need as solution not all the Pareto frontier but only a specific area of it.

References

- [1] M.J. Gacto, R. Alcalá, F. Herrera, J. Alcalá-Fdez (2007). A multi-objective genetic algorithm for tuning and rule selection to obtain accurate and compact linguistic fuzzy rule-based systems. *Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems* 15:5, pages 521–537, 2007.
- [2] J. Casillas, O. Cordón, M.J. del Jesus, and F. Herrera (2005). Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction. *IEEE Trans. Fuzzy Systems* 13:1, pages 13–29, 2005.
- [3] M. Cococcioni, P. Ducange, B. Lazzarini, F. Marcelloni (2007). A pareto-based multi-objective evolutionary approach to the identification of mamdani fuzzy systems. *Soft Computing* 11, pages 1013–1031, 2007.
- [4] C.A. Coello, D.A. Van Veldhuizen, and G.B. Lamont (2002). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers, 2002.
- [5] O. Cordón, F. Herrera, and L. Sánchez (1999). Solving electrical distribution problems using hybrid evolutionary data analysis techniques. *Applied Intelligence* 10, pages 5–24, 1999.
- [6] O. Cordon, F. Herrera, M.J. del Jesus, and P. Villar (2001). A multiobjective genetic algorithm for feature selection and granularity learning in fuzzy-rule based classification systems. In *Proc. IX IFSA World Congress and XX NAFIPS Int. Conf.*, pages 1253–1258, Vancouver, Canada, 2001.
- [7] K. Deb, S. Agrawal, A. Pratab, and T. Meyarivan (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evolutionary Computation* 6:2, pages 182–197, 2002.
- [8] L.J. Eshelman (1991). The CHC adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination. In *Foundations of genetic Algorithms*, volume 1, pages 265–283, 1991.
- [9] L.J. Eshelman, and J.D. Schaffer (1993). Real-coded genetic algorithms and interval-schemata. In *Foundations of Genetic Algorithms*, volume 2, pages 187–202, 1993.
- [10] H. Ishibuchi, T. Murata, and I.B. Turksen (1997). Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems. *Fuzzy Sets and Systems* 89:2, pages 135–150, 1997.
- [11] H. Ishibuchi, and Y. Nojima (2007). Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning. *Int. J. of Approximate Reasoning* 44:1, pages 4–31, 2007.
- [12] L.X. Wang, and J.M. Mendel (1992). Generating fuzzy rules by learning from examples. *IEEE Trans. Systems, Man, and Cybernetics* 22:6, pages 1414–1427, 1992.
- [13] E. Zitzler, M. Laumanns, and L. Thiele (2001). SPEA2: Improving the strength Pareto evolutionary algorithm for multi-objective optimization. In *Proc. Evolutionary Methods for Design, Optimization and Control with App. to Industrial Problems*, pages 95–100, Barcelona, Spain, 2001.