Multi-Objective Genetic Fuzzy Systems: On the Necessity of Including Expert Knowledge in the MOEA Design Process*

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 analyzes the application of Multi-Objective Evolutionary Algorithms in order to attain the desired accuracy/interpretability balance of Fuzzy Rule-Based Systems by preserving the improved accuracy that a tuning of membership functions could give and obtaining more compact models. The results shows that the use of expert knowledge in the algorithm design process can significantly improve the search ability of these algorithms, obtaining more accurate and at the same time simpler models with respect to the single objective based approach.

Recently, Multi-Objective Evolutionary Algorithms (MOEAs) [6, 9] have been applied to improve the accuracy/interpretability trade-off of linguistic Fuzzy Rule-Based Systems (FRBSs), by obtaining Mamdani type models not only accurate but also interpretable. Since this problem presents a multi-objective nature the use of MOEAs to obtain a set of solutions with different accuracy/interpretability degrees is an interesting way to work [5, 8, 13, 14, 15, 16]. Most of these works were applied to classification problems in order to obtain the complete Pareto (set of non-dominated solutions with different trade-off) by selecting or learning the set of rules better representing the example data, i.e., improving the system accuracy and decreasing the FRBS complexity but not considering learning or tuning [4] of the Membership Functions (MF), one of the most powerful techniques to improve the system accuracy.

Our aim in this work is to analyze different alternatives in order to attain the desired accuracy/interpretability balance by maintaining the improved accuracy that a tuning of MFs could give but trying to obtain more compact models. To do this, this work analyzes the application of MOEAs by performing rule selection and a tuning of the MFs parameters, representing a more complex search space that needs of different considerations with respect to the works in the existing literature. Indeed, to directly apply the most recognized MOEAs to perform together tuning and rule selection could present some important problems.

The main problem is that it is practically impossible to obtain the complete optimal Pareto due to the larger search space and due to a faster tuning of the simplest solutions before exploring more promising rule configurations. In this way, it is necessary to include any expert knowledge in the MOEA design process [3]. An adequate application of standard MOEAs could partially deal with this problem by focusing the search in the most interesting zone of the Pareto frontier. Taking into account that non-dominated solutions with a small number of rules and high errors are usually not interesting for an expert since they have not the desired trade-off between accuracy and interpretability, we could focus the search only in the Pareto zone with the

---

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

L. Magdalena, M. Ojeda-Aciego, J.L. Verdegay (eds): Proceedings of IPMU'08, pp. 1446–1453
Torremolinos (Málaga), June 22–27, 2008

most accurate solutions trying to obtain the least possible number of rules (see Figure 1).

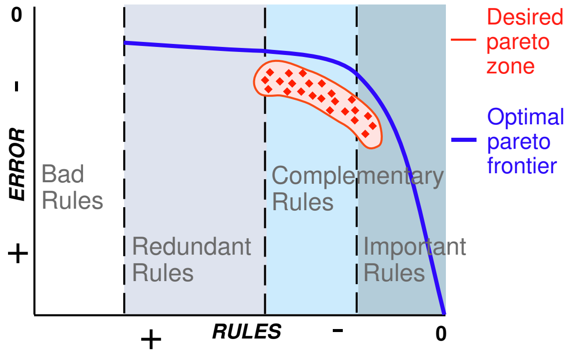

Figure 1: Pareto Frontier in this Problem

In this way, we propose the use of MOEAs as the tool to get almost one improved solution with respect to the classic single objective algorithm (a solution dominating this one). For that, this work presents and analyzes the application of the well-known SPEA2 (standard MOEA applying proper genetic operators) and two extended MOEAs for specific application, SPEA2$_{Acc}$ in [1] and an extension of it proposed in this paper that by applying an intelligent crossover operator (specific for this problem) is able to extract more useful information from different parents, SPEA2$_{Acc^2}$.

Additionally, NSGA-II and two versions of it for finding knees [2] (theoretically the most promising Pareto zones in this problem) are also used with comparative purposes by using the same operators proposed for SPEA2. The results show that the use of expert knowledge in the MOEAs design process significantly improves the search ability of these algorithms.

This work is arranged as follows. Next section presents three specific MOEAs and appropriate genetic operators for their specific application. Section 2 shows an experimental study of these methods in a complex problem. Finally, Section 3 points out some conclusions.

1 Three Specific MOEAs for Rule Selection and Tuning of MFs

The proposed algorithms will perform rule selection from a given fuzzy rule set together with a parametric tuning of the MFs by including expert knowledge in the design process in a cumulative way. We consider one of the most used MOEAs of the second generation, SPEA2 [18] and two extended ones for specific application to this concrete problem, SPEA2$_{Acc}$ in [1], and an extension of that, called SPEA2$_{Acc^2}$. All of them consider two objectives, system error and number of rules.

In the next, we present SPEA2 and SPEA2$_{Acc}$ algorithms and we propose SPEA2$_{Acc^2}$ applied for linguistic fuzzy modeling. At first, the common components of these algorithms are proposed and then the main steps and specific characteristic of them are described.

1.1 Common Components

As mentioned, we describe three algorithms including expert knowledge in a cumulative way to perform tuning and rule selection. In the next, the common components needed to apply these algorithms in this problem are explained. They are coding scheme, initial gene pool, objectives and genetic operators:

- **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 $C_S^p = (c_{S1}, \ldots, c_{Sm})$ part, the coding scheme consists of binary-coded strings with size m (number of initial rules). Depending on whether a rule is selected or not, values '1' or '0' are 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 in the data base,

  $$C_i = (a_1^i, b_1^i, c_1^i, \ldots, a_{m^i}^i, b_{m^i}^i, c_{m^i}^i),$$
  $$i = 1, \ldots, n, \;\; C_T^p = C_1 C_2 \ldots C_n \;\; .$$

  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 ones are generated at random within the corresponding variation intervals, which 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:

$$\begin{aligned}
[I^l_{a^j}, I^r_{a^j}] &= [a^j - (b^j - a^j)/2, a^j + (b^j - a^j)/2] \\
[I^l_{b^j}, I^r_{b^j}] &= [b^j - (b^j - a^j)/2, b^j + (c^j - b^j)/2] \\
[I^l_{c^j}, I^r_{c^j}] &= [c^j - (c^j - b^j)/2, c^j + (c^j - b^j)/2]
\end{aligned}$$

Besides, we have to highlight the way to generate the C_S part in the initial population. Usually, GAs generate the initial population totally at random (random selection of the initial rules). However, to get solutions with high accuracy we should not lose rules that could present good cooperation once their MFs have been evolved. A way to do this is to start with solutions selecting all the rules, which favors a progressive extraction of bad rules (those that do not improve with the tuning of parameters), only by means of the mutation at the beginning and then by means of the crossover.

- **Objectives**

  Two objectives are minimized: number of rules and Mean Squared Error,

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

  with $|E|$ being the size of a data set E, $F(x^l)$ being the output obtained from the FRBS decoded from the said chromosome when the l-th example is considered and y^l being the known desired output. The fuzzy inference system 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.

- **Genetic Operators**

  The crossover operator depends on the chromosome part where it is applied: BLX-0.5 [12] in C_T and HUX [11] in C_S. Four offspring are generated by combining the two from C_S with the two from C_T (the two best replace to their parent). The mutation operator changes a gene value at random in C_S and C_T (one in each part) with probability P_m.

1.2 SPEA2 Based Approach

Considering the components defined and the descriptions of the authors in [18], the SPEA2 algorithm consists of the next steps:

Input: N (population size), $\overline{N}$ (external population size), T (maximum number of generations).

Output: A (non-dominated set).

1. Generate P_0 (initial population) and create $\overline{P}_0 = \emptyset$ (empty external population).
2. Calculate fitness values of individuals in P_t and $\overline{P}_t$.
3. Copy all non-dominated individuals in $P_t \cup \overline{P}_t$ to $\overline{P}_{t+1}$. If $|\overline{P}_{t+1}| > \overline{N}$ apply truncation operator. If $|\overline{P}_{t+1}| < \overline{N}$ fill with dominated in $P_t \cup \overline{P}_t$.
4. If $t \geq T$, return A and stop.
5. Perform binary tournament selection with replacement on $\overline{P}_{t+1}$ in order to fill the mating pool.
6. Apply recombination (BLX-HUX) and mutation operators to the mating pool and set P_{t+1} to the resulting population. Go to step 2 with $t = t + 1$.

1.3 Accuracy-Oriented Based Approach: SPEA2$_{Acc}$ Algorithm

SPEA2$_{Acc}$ was recently proposed in [1], and is a particularization of SPEA2 to better solve the problem of rule selection and tuning of FRBSs. This algorithm tries to focus the search on the desired Pareto zone, high accuracy and low number of rules, proposing two main changes on SPEA2 for giving more selective pressure to those solutions with higher accuracy (exploration in principle and explotation at the end). These changes were also applied on NSGA-II in [1] showing not so good results. The said changes are described next:

- A restarting operator is applied at the algorithm mid, by maintaining the most accurate individual as the sole individual in $\overline{P}_{t+1}$ (with size 1) and obtaining the remaining individuals in P_{t+1} with its same rule configuration and tuning parameters generated at random (within their respective variation intervals). This operation is performed in step 4 as a second condition, then returning to step 2 with $t = t+1$. The search is then concentrated in the desired Pareto zone (similar solutions in a zone with high accuracy).

- In each algorithm stage (before/after restarting), the number of solutions in $\overline{P}_{t+1}$ considered to form the mating pool is progressively reduced. 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 progressively reduced from 100% at the beginning to 50% at the end of each stage ($|\overline{P}_{t+1}| \cdot \frac{TotalEvals-Evals}{TotalEvals}$, with $Evals$ being the number of evaluations from the last restarting).

1.4 SPEA2$_{Acc}$ Extension: SPEA2$_{Acc^2}$

SPEA2$_{Acc}$ represents a good way to obtain more accurate solutions by maintaining only a few more rules with respect to its counterpart (SPEA2). However, sometimes this fact represents a problem since there are problems in which to obtain accurate solutions could be easy but not so easy to remove unnecessary rules. In this subsection, we extend this algorithm in order to solve this problem.

To do that, we replace the HUX crossover (C_S part) by an intelligent crossover based on our experience in this concrete problem, which is able to profit even more from the corresponding parents. To obtain each offspring the following steps are applied:

1. The BLX crossover is applied to obtain the C_T part of the offspring.
2. Once the real parameters are obtained for the whole data base, for each gene in C_S the corresponding rule is independently extracted from each individual involved in the crossover (offspring and parents 1 and 2). The same rule is then obtained three times with different MFs (those concerning these three individuals).
3. Euclidean normalized distances are computed between offspring and each parent by only using the center points (vertex) of the MFs involved in the extracted rules. The differences between each two points are normalized by the amplitude of their respective variation intervals.
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 by repeating this process four times (after applying mutation, only the two most accurate are taken as descendent). By using this operator, exploration is performed in the C_T part and the C_S part is obtained based on the previous knowledge each parent has about the use or not of a specific configuration of MFs for each rule.

Since a better exploration is performed for the C_S part, the mutation operator does not need to add rules (rules eliminated in the parents for a similar bad configuration of the MFs involved in them). Thus, once an offspring is generated the mutation operator changes a gene value at random in C_T (as in the previous algorithm) and directly sets to zero a gene selected at random in C_S (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 Experiments

In this section, we present an example on the use of MOEAs to obtain linguistic models with a good trade-off between interpretability and accuracy in a real-world problem [7]

Table 1: Methods Considered for Comparison

Method	Ref.	Description
WM	[17]	Wang & Mendel Algorithm
WM+T (T)	[4]	Tuning of Parameters
WM+S (S)	[4]	Rule Selection
WM+TS (TS)	[4]	Tuning & Selection
Application of standard MOEAs for general use		
WM+TS-SPEA2 (SP)	[18]*	Tuning & Selection by SPEA2
WM+TS-NSGA-II (NS)	[10]*	Tuning & Selection by NSGA-II
WM+TS-NSGA-II$_A$ **(NS**$_A$**)**	[2]*	Tuning & Selection by NSGA-II with angle-measure
WM+TS-NSGA-II$_U$ **(NS**$_U$**)**	[2]*	Tuning & Selection by NSGA-II with utility-measure
Extended MOEAs for specific application		
WM+TS-SPEA2$_{Acc}$ **(SP**$_{Acc}$**)**	[1]	Accuracy-Oriented SPEA2
WM+TS-SPEA2$_{Acc^2}$ **(SP**$_{Acc^2}$**)**		Extended SPEA2$_{Acc}$

* based on that algorithm

with 4 input variables that consists of estimating the maintenance costs of medium voltage lines in a town. To do that, we also compare the studied algorithms with the paradigm of MOEAs, NSGA-II [10] and two versions of NSGA-II for finding knees [2] (that theoretically should obtain the most promising Pareto zones) in order to show the good behavior of SPEA2 in this framework. They consider the same components described in section 1.1.

Methods considered for the experiments are briefly described in Table 1. WM method is considered to obtain the initial rule base that should be improved by the proposed post-processing methods. Although any other method could be considered (even a rule base obtained from experts), WM is quick and simple, and usually generates a good set of candidate rules (we have not found very significant differences by using more sophisticated algorithms). 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. T, S and TS use the same or a part of the coding scheme presented in Section 1.1, C_T, C_S and $C_S C_T$ respectively (as they were proposed in [4]). In this way, TS is the single objective counterpart of the proposed algorithms. Additionally, S and T are considered as illustrative references of both techniques (rule selection or tuning) when they are independently applied. On the other hand, MOEAs considered perform rule selection from the initial fuzzy rule set (obtained from WM) together with the parametric tuning of the MFs considering two objectives, system error and number of rules.

2.1 Problem Description

Estimating the maintenance costs of the medium voltage electrical network in a town [7] is a complex but interesting problem. Since a direct measure is very difficult to obtain, it is useful the use of 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 post-processing methods have been run 6 times (6 different seeds), showing for each problem the averaged results of a total of 30 runs. In the case of methods with multi-objective nature, the averaged values are calculated considering the most accurate solution from each Pareto obtained. In this way, MOEAs can be compared with several single objective based methods. This way to work differs with the previous works in the specialized literature in which one or several Pareto fronts are presented and an expert should after select one solution. Our main aim following this approach is to compare the same technique when only the accuracy objective is considered (algorithm WM+TS) with the most accurate solution found by the proposed multi-objective algorithms in order to see if the Pareto fronts obtained are not only wide but also optimal (almost similar solutions to that obtained by WM+TS should be found in the final Pareto).

The initial linguistic partitions are comprised by *five linguistic terms* with equally distributed triangular shape MFs. The values of the input parameters for S, T and TS (single objective oriented algorithms) are[1]: population size of 61, 100000 evaluations, 0.6 as crossover probability and 0.2 as mutation probability per chromosome. In the case of MOEAs, the best results were obtained by taking similar sizes than those considered by S, T and TS (with single objective) in these kinds of problems (i.e., when the size of the population used for parent selection takes these values). We recommend the use of this simple rule of thumb to fix the population size in these kinds of problems. The input parameters considered by the MOEAs are shown in the next: 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 and 0.2 as mutation probability per chromosome.

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. The interpretation of this column is:

Table 2: Results of the studied methods

Met.	**#R**	**MSE**$_{tra/tst}$	$\sigma_{tra/tst}$	$t_{tra/tst}$
WM	65.0	57605/57934	2841/4733	+/+
T	65.0	17020/21027	1893/4225	+/+
S	40.9	41158/42988	1167/4441	+/+
TS	41.3	13387/17784	1153/3344	+/+
SP	**28.9**	11630/15387	1283/3108	+/=‡
NS	31.4	11826/16047	1354/4070	+/+
NS_A	29.7	11798/16156	1615/4091	+/+
NS_U	30.7	11954/15879	1768/4866	+/+
SP_{Acc}	32.3	10714/14252	1392/3181	=/=
SP_{Acc^2}	29.8	**10325/13935**	**1121/2759**	*/*

‡ + with 94% confidence

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

2.2 Results and Analysis

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

- The best results were obtained by TS-SPEA2$_{Acc^2}$ and TS-SPEA2$_{Acc}$, showing that the use of expert knowledge in the design process can help to obtain more optimal Pareto fronts. Moreover, TS-SPEA2$_{Acc^2}$ is able to obtain the best results with less rules than TS-SPEA2$_{Acc}$.
- All MOEAs studied get significantly simpler models that those obtained by only considering the accuracy based objective and almost the same results (minor average values in all the cases and statistical differences in the case of the extended MOEAs). This is a positive fact since a proper use of MOEAs can improve the desired trade-off with respect to the classic accuracy-based approaches, and specific adaptations can help to improve the performance of standard MOEAs.

---

[1] Standard common parameters that work well in most cases

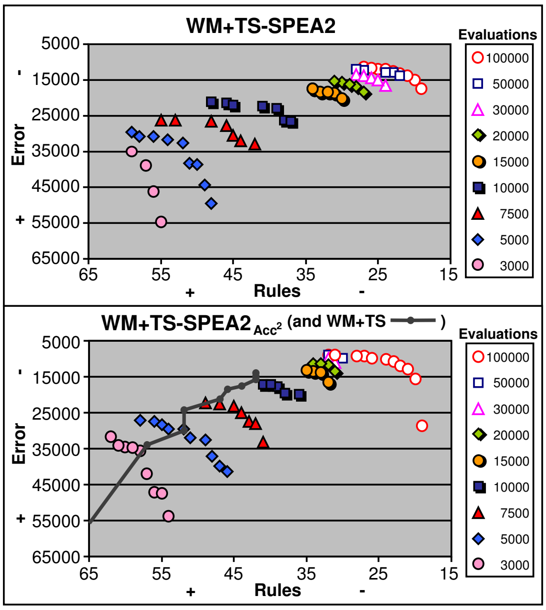

Figure 2: Evolution of Pareto with (SP) and (SP_{Acc^2}), and of the best solution with (TS)

In Figure 2, we can see the Pareto evolution in a representative run with TS-SPEA2 and TS-SPEA2$_{Acc^2}$, 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.

Table 3: Results obtained by random initialization of the C_S part (rule part)

Met.	**#R**	**MSE**$_{tra/tst}$	$\sigma_{tra/tst}$	$t_{tra/tst}$
SP	21.9	18768/23951	2256/**5198**	+/+
NS	27.7	17688/23762	2333/7681	+/+
NS_A	**19.5**	23981/29442	3709/7058	+/+
NS_U	24.4	18728/24148	2071/5397	+/+
SP_{Acc}	23.2	**14175/18289**	**1752**/5571	*/*
SP_{Acc^2}	20.2	16539/21977	2729/5625	+/+

A study has been also performed on the importance of the initialization component for the rule selection part in the chromosome (considering a random initialization instead of that presented in Section 1.1). Table 3 presents the results obtained. By using random initialization the results obtained present too low numbers of rules with much worse results especially in the test. In any case, there are two important facts in these results:

- TS-SPEA2$_{Acc}$ and TS-SPEA2$_{Acc^2}$ methods were not very affected by the random initialization (very low number of rules and a good accuracy).
- Fixing the percentage of selected rules in the initial population (100% in Table 2 and 50% in Table 3) can be a way to regulate the desired trade-off since this biases the number of rules in the final solutions.

3 Concluding Remarks

We have analyzed the application of different MOEAs to obtain simpler but still accurate linguistic fuzzy models by performing rule selection and a tuning of the MF parameters. Since combining rule selection and tuning of the system parameters represents a very complex search space, some considerations based on the experience are needed in the MOEA design process in order to get good solutions.

The results obtained have shown that a proper use of MOEAs can represent a way to get even more accurate and simpler linguistic models than those obtained by only considering performance measures. Besides, population initialization has demonstrated to be an important component that can help to regulate the desired trade-off since this biases the number of rules in the final solutions. The best results obtained by TS-SPEA2$_{Acc^2}$ show that the use of experience based knowledge in the MOEAs design process can significantly improve the search ability of these algorithms.

For future works, a lexicographic fitness accuracy-size could be also considered. The main aim would be to evolve a preferably small, but mostly accurate rule base.

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. Branke, K. Deb, H. Dierolf, and M. Osswald (2004). Finding Knees in Multi-objective Optimization. In *Proc. VIII PPSN Conf., LNCS*, volume 3242, pages 722–731, Birmingham, UK, 2004.

[3] P.P. Bonissone, R. Subbu, N. Eklund, T.R. Kiehl (2006). Evolutionary algorithms + domain knowledge = real-world evolutionary computation. *IEEE Trans. Evolutionary Comput.* 10:3, pages 256–280, 2006.

[4] 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.

[5] M. Cococcioni, P. Ducange, B. Lazzerini, F. Marcelloni (2007). A pareto-based multiobjective evolutionary approach to the identification of mamdani fuzzy systems. *Soft Computing* 11, pages 1013–1031, 2007.

[6] C.A. Coello, D.A. Van Veldhuizen, and G.B. Lamont (2002). Evolutionary Algorithms for Solving Multi-Objective Problems. Kluwer Academic Publishers, 2002.

[7] 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.

[8] 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.

[9] K. Deb (2001). Multi-objective optimization using evolutionary algorithms. John Wiley & Sons, NY, USA, 2001.

[10] 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.

[11] 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* 1, pages 265–283, 1991.

[12] 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.

[13] 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.

[14] H. Ishibuchi, T. Nakashima, and T. Murata (2001). Three-objective genetics-based machine learning for linguistic rule extraction. *Inf. Sciences* 136, pages 109-133, 2001.

[15] H. Ishibuchi, and T. Yamamoto (2004). Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. *Fuzzy Sets and Systems* 141:1, pages 59–88, 2004.

[16] H. Ishibuchi, and Y. Nojima (2007). Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning. *Int. J. of Approx. Reasoning* 44:1, pages 4–31, 2007.

[17] 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.

[18] E. Zitzler, M. Laumanns, and L. Thiele (2001). SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In *Proc. Evolutionary Methods for Design, Optimization and Control with App. to Industrial Problems*, pages 95–100, Barcelona, Spain, 2001.