MERBIS - A Multi-Objective Evolutionary Rule Base Induction System

Christian Setzkorn, Ray C. Paton

Department of Computer Science, University of Liverpool, UK

Abstract. Classifier induction is a specific data-mining task that is part of the knowledge discovery process. Its objectives are to extract accurate, comprehensible and interesting knowledge. However, most classifier induction approaches focus only on one of these objectives. This paper introduces a Multi-Objective Evolutionary Rule Base Induction System (MERBIS) that is capable of optimising both, accuracy and comprehensibility objectives at the same time. We investigate different parameter sets for this approach and validate its performance on several benchmark data sets.

1 Introduction

Classifier induction involves determining a classifier D from data that is capable of assigning to objects a class ω_j from a predefined class set $\Omega = \{\omega_1, \ldots, \omega_M\}$. The generated classifiers should, as accurately as possible, assign classes to objects that were not necessarily encountered during the induction process. Objects are described by features and stored within a data set. Features can be categorical (e.g. colours, yes/no) or numerical (e.g. age). Classifier induction is a specific data-mining task [20] that is part of the knowledge discovery process whose aim is the extraction of accurate, comprehensible and interesting knowledge [18, 20], thus suggesting that several objectives have to be optimised while inducing a classifier.

The comprehensibility of a classifier has practical importance. Some researchers argue that only comprehensible classifiers are actually adopted in practice [27, 28, 41]. One reason for this might be that domain experts are very wary and distrustful of the incomprehensible results generated by a computer [53]. Comprehensibly can be achieved by symbolic classifiers, which correspond to an explicit knowledge representation form [1, 39]. MERBIS produces Fuzzy Classification Rule Bases (FCRBs), which are a specific type of symbolic classifiers, which can exhibit high transparency [35].

The performance of a classifier on unseen objects is also very important. Unfortunately, the objectives of performance and comprehensibility often conflict with each other [52]. It is therefore important to deploy a supervised classification approach (i.e. inducer) that is capable of finding trade-off solutions between these objectives to provide a domain expert is provided with a choice between several solutions such that she/he can pick one solution depending on her/his preferences.

To generate trade-off solutions in the form of FCRBs we use a multi-objective evolutionary approach called MERBIS (**M**ulti-Objective **E**volutionary **R**ule **B**ase **I**nduction **S**ystem). Multi-objective evolutionary algorithms (MOEAs) belong to the group of problem-independent heuristic search algorithms [51], that have been used successfully in areas such as medicine, biotechnology, robotics, telecommunications, and economics to mention only a few [2].

The majority of existing evolutionary approaches for the induction of FCRBs either focus upon the optimisation of comprehensibility or performance [6, 25]. We are only aware of a few approaches that actually attempt to find trade-off solutions between performance and comprehensibility utilising MOEAs (e.g. Ishibuchi et al. [25, 33] and Bentley [4]). We believe that our study is novel because we investigate different parameter settings, evaluate the approach on a number of data sets, deploy state of the art MOEA components and apply a new performance measure. This new performance measure has several advantages over other measures that have been previously deployed.

The remainder of this paper is organised as follows. Section 2 describes how FCRBs can be used to classify objects. Section 3 introduces the MERBIS system. Here we provide details of the chosen representation scheme, genetic operators, and objective functions. Section 4 describes the methodology used to compare different parameter settings and to establish the performance of MERBIS on different benchmark data sets. Section 5 provides results and discuses them. Section 6 provides some conclusions and suggest avenues for future research.

2 Classification using Fuzzy Classification Rule Bases

This section describes how fuzzy rule bases can be used to classify objects. Rule systems consist of rules of the following form:

$$IF\ cond_1\ AND\ cond_2\ \ldots\ AND\ cond_n\ THEN\ Class\ =\ \omega_i\ with\ CF$$

The antecedent (IF-part) usually consists of conjunctions of conditions. Conditions (e.g. $cond_1$) correspond to sets defined upon the domain of features (e.g. $bloodpressure = low, greater(income, expenditure)$). Conjunctions (logical operator AND) combine different sets (e.g. $bloodpressure = low\ AND\ smoking = yes$). The consequent (THEN-part) associates a class with the antecedents. Each rule also contains the value CF which measures the rule's past performance. A rule base can contain several antecedents with the same consequent. This makes it possible to describe multi-modal distributions within the feature space, which can cause problems to other inducers [13, 20, 44]. Antecedents with the same consequents are usually combined using a disjunction (logical OR operator). This type of rule base is also referred to as disjunctive normal form, the most common form of rule base [13]. Normally an antecedent's conditions correspond to classical sets that use crisp decision thresholds. For example, the set of patients with low blood pressure could be defined as everyone whose blood pressure is exactly below equal 70 mmHg. However, classical sets like this can lead to instable systems that may produce very different responses (classifications) to similar inputs (objects) [14, 50]. For example, someone who has a blood pressure of 71 mmHg would not belong to the group of patients with low blood pressure even though her/his blood pressure is only slightly higher than 70 mmHg. One way to tackle this limitation is the use of fuzzy sets [54]. Fuzzy sets can be described by membership functions [31] that assign values between zero and one to each domain value, thus permitting smooth decision thresholds. As fuzzy sets produce values between zero and one, other conjunction and disjunction operators have to be defined to combine these sets. The min and $prod$ operators are often used instead of the AND operator and the max instead of the OR operator. Combined fuzzy sets define high-dimensional prototypical clusters within the feature space whose boundaries do not need to be axis parallel [40]. This is a further advantage of FCRBs.

3 The MERBIS System

Figure 1 summarises the structure of the MERBIS system. This structure corresponds to a general evolutionary algorithm whose processes are: genetic operators, fitness evaluation and selection. The interiors of some processes were inspired by the SPEA2 approach [57].

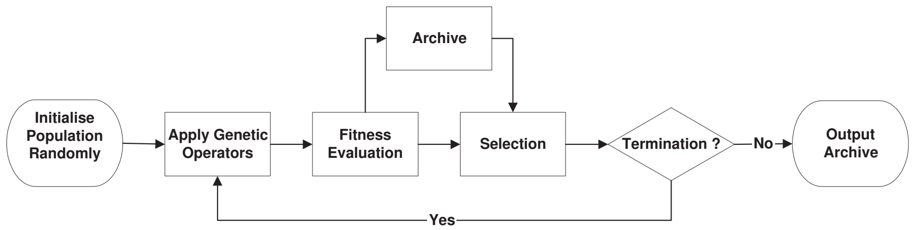


Fig. 1. The structure of MERBIS

Before the genetic operator processes are applied, a number of candidate solutions (i.e. population of individuals) are randomly initialised. In our particular case, individuals take the form of a fuzzy rule base (see section 3.1). After this step, genetic operators recombine and/or slightly change a certain number of individuals within the current population (see section 3.4). It follows the fitness evaluation during which each individual's performance (fitness) is determined. An individual's fitness dependents on its performance on the current data set and its complexity (see section 3.2). The selection process generates a new population of individuals by sampling from the current population and the archive imposing a bias that emulates Darwin's principle of the survival of the fittest [10] (see section 3.3). The archive contains the best (elite) individuals that have been found so far. Deploying an archive ensures that the best individuals are preserved, as they can otherwise get lost due to the randomness of the selection process [57]. The use of an archive is a form of elitism, which increases the likelihood of creating better individuals [12] and has long been considered a beneficial component of EAs [30]. The selection process is succeeded by the termination test during which it is decided whether the algorithm terminates or whether another population (generation) of individuals is produced. If the algorithm does not terminate the current population is passed to the genetic operator processes and evaluation and selection processes are repeated. Each repetition is expected to produce better and better individuals. If the algorithm terminates (e.g. after a maximum number of generations) the individuals within the current population and the archive are evaluated on test data. The final output of the system is the updated set of elite individuals within the archive.

3.1 The Representation Scheme

An individual consists of a number of fuzzy rules of the form described in section 2. Individuals that consist of several rules are often referred to as examples of the Pittsburgh approach [48]. The number of rules is restricted by a minimum and maximum value. Each rule's antecedent takes the form of a tree as shown in figure 2, which is similar to representation scheme of Genetic Programming (GP) [3, 9, 34]. The size of a tree is limited by a maximum number of nodes.

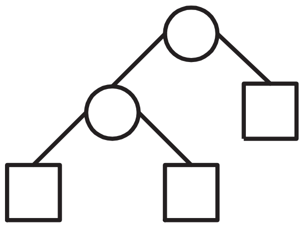


Fig. 2. Antecedent tree

As an antecedent corresponds to a conjunction of fuzzy sets, the non-terminal nodes (depicted as circles in figure 2) of the tree are either the min or the $prod$ operator. The terminal or leaf nodes (depicted as squares in figure 2) are either one of the fuzzy sets depicted in table 1 defined upon the domain of a feature.

Each feature can only be used once within an antecedent to maintain comprehensibility. If an attribute is categorical, a singleton fuzzy set is defined on its domain, as there is no order of the attribute's domain values. Currently, we only allow expressions like: $age = old$. This corresponds to propositional rules that are less expressive than first-order logic expression, which can contain conditions like: $greater(income, expenditure)$ [15]. We restrict ourselves to propositional rules because their induction is less complicated and does not require the features to have compatible domains [20, 21].

When an object is presented to an individual each antecedent generates an output between zero and one. We have mentioned in section 2 that there may exist several antecedents with the same associated class (consequent). If this is the case, the maximum

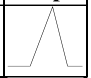
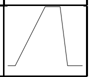
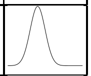
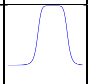
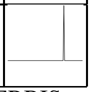
Formula	Shape
$MF1(x;a,b,c) = max(min(\frac{x-a}{b-a}, \frac{c-x}{c-b}), 0)$	
$MF2(x;a,b,c,d) = max(min(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}), 0)$	
$MF3(x;a,c) = e^{\frac{-(x-c)^2}{2a^2}}$	
$MF4(x;a,b,c) = \frac{1}{1+\lvert\frac{x-c}{a}\rvert^{2b}}$	
$MF5(x;a,b) = \begin{cases} a & \text{if } x = b \\ 0 & \text{otherwise} \end{cases}$	

Table 1. The membership function utilised by MERBIS.

value of those antecedents is chosen as the output value for this class. If an individual does not contain an antecedent for a particular class the output value for this class is zero. Hence, the response of an individual to an object is a vector whose dimensionality is equal to the number of classes to be predicted. Each value within this vector indicates the membership degree that an individual associates with the corresponding class. Please note that the position of a response value within this vector is unimportant and that we do not deploy an inference mechanism. The vector output of an individual is exploited by the performance measure described in section 3.2.

3.2 The Fitness Evaluation

The Performance Measure The majority of rule base inducers optimise the accuracy or error respectively of the rule bases. However, such measures, and many others derived from contingency tables, are inappropriate when the costs of misclassification and the class priors are unknown [17]. This is frequently the case in practice. Therefore, we deploy a performance measure originally proposed by Hand et al. [23] that is based on the area under the receiver operating characteristic curve (AUC) (e.g., [5, 17, 24]. Hand et al.'s measure has many advantages because it does not exhibit the above-mentioned shortcomings, works with degrees of memberships (not necessarily probabilities) and can be deployed for multiple classes. The measure is denoted o_1 as it is the first objective to be minimised. It is computed according to formula 1.

$$o_1 = 1 - (\frac{2}{c(c-1)} \sum_{i<j} A(i,j)) \tag{1}$$

Here c corresponds the number of classes to be predicted. The value for $A(i,j)$ is computed as follows:

$$A(i,j) = \frac{A(i \mid j) + A(j \mid i)}{2} \tag{2}$$

The values $A(i \mid j)$ and $A(j \mid i)$ estimate the AUC using the Mann-Whitney-Wilcoxon (MWW) two sample test statistic. This statistic compares two arrays. Array one contains the maximum response or membership values from antecedents to objects that belong to the class indicated by the first index (i or j in either $A(i \mid j)$ or $A(j \mid i)$) and the class value within the consequent(s). Array two contains the maximum responses from the same antecedents to objects that belong to the class indicated by the second index (these objects correspond to noise). One would expect that if the antecedents discriminate between objects from the two different classes the first array should contain much higher values than the second array. To determine the MWW statistic, the two arrays are merged and arranged in ascending order and then equation 3 is applied.

$$A = \frac{S_0 - n_0(n_0+1)/2}{n_0 n_1} \tag{3}$$

Here S_0 denotes the sum of the ranks of response values from the first array. The values n_0 and n_1 denote the number of values in array one and two respectively.

As mentioned in section 2, we determine for each rule a certainty degree CF. It measures the past performance of the rule [8] and is computed according to equation 4.

$$CF = \frac{Sj^k}{S^k} \tag{4}$$

Here Sj^k corresponds to the sum of the response values from the k-th rule's antecedent to objects that belong to the class indicated by the corresponding consequent. On the other hand, S^k corresponds to the response values of the k-th rule's antecedent to all objects independent of their class membership.

The Comprehensibility Measures According to Ishibuchi et al. [25] the number of rules (objective o_2) and conditions (objective o_3) within the FCRB can quantify the comprehensibility of a FCRB. Hence, these two objectives have to be minimised as fewer rules and conditions improve the comprehensibility of a system.

The Fitness Assignment The fitness of an individual i is computed according to equation 5.

$$F(i) = R(i) + D(i) \tag{5}$$

Here $R(i)$ and $D(i)$ correspond to dominance (see equation 6 and 7) and density information (see equation 8) associated with individual i.

$$R(i) = \sum_{j \in P_t + \overline{P_t}, j \succ i} S(j) \tag{6}$$

$$S(i) = \mid \{ j \mid j \in P_t + \overline{P_t} \wedge i \succ j \} \mid \tag{7}$$

Here P_t and $\overline{P_t}$ refer to individuals from the population and the archive respectively. The expression $i \succ j$ denotes the dominance relation between individual i and j. An individual i dominates another individual j if i is at least as good as j in all objectives and better with respect to at least one objective. Equation 7 determines how many individuals individual i dominates within P_t and $\overline{P_t}$. Equation 6 determines the number of individuals which are dominated by the individuals that dominate individual i. If the value of R_i is zero the individual i is non-dominated. The density information is computed according to equation 8 and is an adaptation of the k-th nearest neighbour method [47].

$$D(i) = \frac{1}{\sigma_i^k + 2} \tag{8}$$

The value σ_i^k measures the Euclidean distance between the objective values between the k-th and the i-th individual. The value for k is equal to the square root of the sample size: $k = \sqrt{N + \overline{N}}$ [47]. The value N and $\overline{N}$ correspond to the number of individuals in the population and archive respectively.

3.3 Selection

The selection process produces a new population of individuals from the current population and the archive. To generate a new population, we deploy binary tournament selection [57] utilising the fitness value described in section 3.2. During binary tournament selection, two individuals are picked randomly without replacement from either the population or the archive. The probability that an individual is picked from the archive is determined by the elitism degree (ED). The individual with the lowest fitness value (see equation 5) is declared as the winner and inserted into the new population. If a tie occurs one individual is chosen randomly and inserted into the new generation. This procedure is repeated until the new population has reached the size of the old one.

3.4 The Genetic Operators

Genetic operators (GOs) are responsible for the 'movement' of individuals through the search space. We distinguish between GOs that involve two individuals (GO2s) and one individual (GO1s). The GO2s can lead to an exchange of rules or their parts between two individuals. This mimics sexual recombination or crossover. The GO1s can change the structure of an individual. Rules can be added, removed, changed or the whole FCRB can be reinitialised. This mimics natural mutation. We have implemented several GOs of both types because it has been shown that an evolutionary algorithm, which deploys several GOs, can lead to superior results [49]. In addition, we would also like to investigate the effects of the implemented GOs and their teamwork. If we could determine the most effective GOs we may improve them further. This may lead to very problem specific GOs and thus to more effective evolutionary algorithms for the task of data mining [19]. Before we introduce the implemented GOs we would like to explain how we deploy several GOs. Please note that the described procedure is applied for each operator type (GO1 and GO2) separately.

Each operator has a particular probability to be selected from the available operator set, denoted as $IPGO1_i$ or $IPGO2_i$ respectively. The index refers to a specific operator. It is ensured that these probabilities add up to one because roulette wheel selection [11] is used to select a specific operator. This common selection method ensures that operators that have higher probabilities are selected more often on average than operators with lower probabilities. When an operator has been determined one individual, in the case of GO1s, or two individuals, in the case of GO2s, are chosen without replacement from the current population. An operator is applied with a particular probability denoted as MP (mutation probability) or CP (crossover probability), respectively. The resulting individuals are then accumulated in the new population. This process is repeated until the current population is either empty or only contains one individual (in the case of GO2s). In the later case, the remaining individual is added to the new population, which then becomes the current population. It follows a more detailed description of the implemented operators.

3.5 One Individual Genetic Operators

The MERBIS approach deploys the following one individual operators:

- The $GO1_1$ operator reinitialises each condition (fuzzy set) within the FCRB antecedents with the probability MP.
- The $GO1_2$ operator reinitialises each consequent of the FCRB with the probability MP.
- The $GO1_3$ operator reinitialises each consequent of the FCRB with the probability MP. In addition it mutates each antecedent tree with the same probability. Each node of the tree is examined in respect to MP. If a node is mutated, a new sub-tree is created at this point fulfilling the restrictions described in section 3.1.
- The $GO1_4$ Operator does the same as the $GO1_3$ operator with the exception that is not capable of reinitialising each consequent of the FCRB.
- The $GO1_5$ operator removes one rule from the FCRB with the probability MP. It is only applicable if there are currently more than the minimum number of rules within the FCRB.
- The $GO1_6$ operator adds one rule with the probability MP. It is only applicable if there are currently less than the maximum number of rules minus one within the FCRB.
- The $GO1_7$ operator reinitialising each antecedent's condition with the probability MP, this operator slightly changes a condition. The operator either changes the coverage[1] of a condition's fuzzy set or it moves it either to the left or to the right along the corresponding feature's domain.
- The $GO1_8$ operator mutates the antecedent that exhibits the lowest CF with the probability MP. The mutation works after the same principle as for the operator $GO1_3$.
- The $GO1_9$ operator removes the rule with the lowest CF with the probability MP. A rule can only be removed as long as the FCRB contains more than the minimum number of allowed rules.

[1] The coverage refers to the interval of domain values to which the membership function of the fuzzy sets assigns values greater than zero

3.6 Two Individuals Genetic Operators

The MERBIS approach deploys the following two individuals operators:

- The $GO2_1$ operator performs an exchange (crossover) of sub-trees between two antecedent trees that where randomly chosen from each of the two individuals. The exchange takes place with the probability CP. Figure 3 illustrates this. The crossover point is chosen with a uniform probability. Please note that the exchange only takes place if the constraints, described in section 3.1, are not violated.
- The $GO2_2$ operator works after the same principle as the $GO2_1$ operator with the difference that both randomly chosen antecedent tree must have the same consequent.
- The $GO2_3$ operator randomly exchanges one rule between two FCRBs with the probability CP.
- The $GO2_4$ operator removes rules from the first FCRB and inserts them into the second FCRB with probability CP. This is done as long as the first rule system contains more than the minimum number of rules allowed and as long as the second individual's number of rules is less equal to the maximum number of allowed rules.
- The $GO2_5$ operator merges two FCRBs with probability MP. If the resulting FCRBs exceed the maximum number of allowed rules, rules are randomly removed to fulfil this constraint.

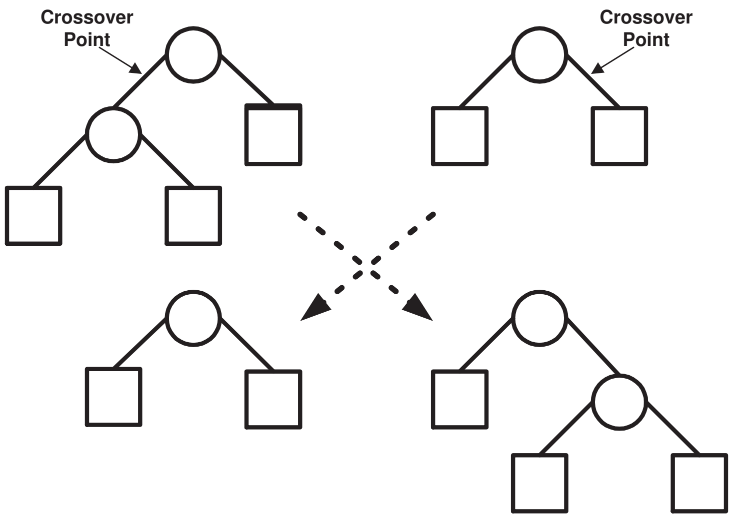


Fig. 3. Crossover between two antecedent trees.

3.7 The Archive

As mentioned in section 3, an archive is an crucial component for a MOEA. However, the deployment of standard archives does not automatically guarantee convergence towards optimal solutions and diversity promotion [36]. Consequently, we utilise a new archive strategy that was originally proposed by Laumanns et al. [36]. It ensures diversity and convergence and in addition limits the size of the archive depending on a chosen resolution. Details of this archive strategy are beyond the scope of this paper. The interested reader is rather referred to [36].

3.8 Parameters

The MERBIS approach needs to be provided with the following parameters:

– **Number of Generations:** constraints the number of generations the system can 'breed' before the archive is presented as the final solution set.
– **Minimum Number of Rules:** determines the minimum number of rules that an individual has at least to contain. This value depends on the number of classes in a data set. For example, for a data set that contains three classes this value would be three.
– **Maximum Number of Rules:** determines the maximum number of rules that an individual could contain.
– **Maximum Number of Antecedent Nodes:** - constrains the maximal number of nodes that can be used within the antecedent.
– **Elitism Degree (ED):** - determines the probability of selecting an individual from the archive rather than the current population while generating a new population.
– **One Individual Genetic Operator Intensity Probability (IPGO1):** - determines the probability that the i-th GO1 operator is chosen from the set of all GO1s.
– **Mutation Probability (MP):** - determines the probability that the chosen operator is applied to the chosen individual.
– **Two Individuals Genetic Operator Intensity Probability (IPGO2):** - determines the probability that the i-th GO2 operator is chosen from the set of all GO2s.
– **Crossover Probability (CP):** - determines the probability that the chosen operator is applied to two randomly chosen individuals.

4 Methodology

The current implementation of our approach requires the adjustment of several parameters. We therefore carried out a performance investigation of our approach for a limited number of parameter values on five data sets taken from the UCI Machine Learning Repository[2]. Table 2 summarises the chosen the parameter values.

[2] http://www.ics.uci.edu/%7Emlearn/MLRepository.html

Name	Value(s)
Population Size	100
Number of Generation	100
Maximum Number of Rules	50
Maximum Number of Antecedent Nodes	3
Elitism Degree (ED)	0.1, 0.85
One Individual Genetic Operator Intensity Probability (IPGO1)	$GO1_1$ or $GO1_2$ ... $GO1_9$ or *all*
Mutation Probability (MP)	0.001, 0.01, 0.3
Two Individuals Genetic Operator Intensity Probability (IPGO2)	$GO2_1$ or $GO2_2$ ... $GO2_5$ or *all*
Crossover Probability (CP)	0.3, 0.6, 0.95

Table 2. The parameter values used for the experiments.

The choice of parameters follows from previous work of De Jong [30], Grefenstette [22] and Laumanns et al. [37]. For example, De Jong recommended a crossover probability (parameter CP) of 0.6 and a mutation probability (parameter MP) of 0.001. Grefenstette, on the other hand, suggested a CP of 0.95 and a MP of 0.01. Laumanns et al. recommended high mutation probabilities together with high elitism degrees. We have chosen to deploy either only one of the possible genetic operators, assigning a probability of one to a corresponding $IPGO$ (denoted as $GO1_1$ or $GO1_2$ ... $GO1_9$ in table 2), or we deploy all implemented genetic operators with the same probability, assigning the same probability to each $IPGO$ (denoted as *all* in table 2).

Because some of the parameters in table 3.8 contain more than one value (ED, MP, IPGO1, CP, IPGO2), there are 2x3x10x3x6=1080 possible parameter sets. Consequently, we performed 1080 experiments for each data set. Ten-fold stratified cross validation was deployed to estimate the performance of our approach for each parameter set. As the output of a MOEA is a set of non-dominated solutions we use the hypervolume measure [56] to obtain a point estimate. In a nutshell, the hypervolume measures the volume between the approximate Pareto front (defined by the current non-dominated solutions) and an anti-ideal reference point. We chose an anti-ideal reference point such that we obtain a hypervolume value between zero (lowest performance) and one (highest performance). The hypervolume is one of the best currently available measures for the performance evaluation of MOEAs [42] but it also has some limitations [32, 55].

To investigate the performance results of different parameter sets we used Factorial Analysis of Variance (e.g. [26]). This method enables one to determine the effect of a parameter (main effect) on the hypervolume value (while ignoring the other parameters) and has been used previously to investigate different parameter sets for EAs (e.g. [45]). In addition, this method also enables one to detect interactions between parameters. Interaction effects exist if one parameter affects the performance depending on the value of one or several other parameters [29]. To investigate the main and interaction effects we deployed the statistical package SAS[3] (Version 8.0) and the Ryan/Einot

[3] SAS is a trademark of SAS Institute Inc.

and Gabriel/Welsch procedure (REGW) to perform multiple comparisons. The REGW procedure is recommended in [26]. We intend to answer the following questions:

1. Does one particular parameter set exist that leads to better performance of the approach than other parameter sets on all data sets?
2. Are the recommendations of De Jong, Greffenstette and Laumanns et al. applicable for our approach?
3. Is it beneficial to deploy several genetic operators at the same time?
4. Are some of the genetic operators better than others?

Finally, the performance of our approach was evaluated on thirteen data sets from the UCI Machine Learning Repository using ten-fold stratified cross-validation and one parameter set derived from table 2. We have chosen these data sets because they were also used by Fawcett [17] to estimate the performance of different rule inducers in terms of the AUC. The results of the performance evaluation are presented in section 5.2.

5 Results and Discussion

5.1 Results Parameter Investigation

Using ANOVA we could detect significant [4] differences between the obtained mean hypervolume values. Table 3 summarises all significant main and some interactions effects. We could also detect higher order interaction (e.g. CPxMPxIPGO1) but we do not report them here.

Data Set	Main Effects	Interaction Effects
BCW	IPGO1, IPGO2, CP, ED, MP	IPGO1xIPGO2, CPxIPGO1, EDxIPGO1, MPxIPGO1, CPxIPGO2, EDxIPGO2, MPx-IPGO2, CPxMP, EDxMP
Promoter	IPGO1, IPGO2, CP, ED, MP	IPGO1xIPGO2, EDxIPGO1, MPxIPGO1, CPxIPGO2, MPxIPGO2
Image	IPGO1, IPGO2, CP, MP	IPGO1xIPGO2, CPxIPGO1, EDxIPGO1, MPxIPGO1, CPxIPGO2, EDxIPGO2, MPx-IPGO2, CPxMP, EDxMP
Glass	IPGO1, IPGO2, ED, MP	IPGO1xIPGO2, CPxIPGO1, EDxIPGO1, MPxIPGO1, CPxIPGO2, EDxIPGO2, MPx-IPGO2, CPxMP, EDxMP
German	IPGO1, IPGO2, ED, MP	IPGO1xIPGO2, EDxIPGO1, MPxIPGO1, CPxIPGO2, EDxIPGO2, MPxIPGO2

Table 3. Main and interaction effects.

[4] We refer to everything as significant at the $\alpha < 0.01$ level

Table 4 reports the parameter values for main effect means (mean hypervolume values) that are significantly better than other main effect means (Best column). We also report the parameter values for main effect means (mean hypervolume values) that are significantly worse than other main effect means (Worst column).

	Parameter Values									
	Best					Worst				
Data Set	ED	MP	IPGO1	CP	IPGO2	ED	MP	IPGO1	CP	IPGO2
BCW	0.1	0.3	$GO1_1$, $GO1_3$, $GO1_4$, $GO1_7$, all	0.6	all	0.85	0.001	$GO1_6$	0.3, 0.95	$GO2_5$
Promoter	0.1	0.3	$GO1_1$, $GO1_3$, $GO1_4$, all	0.95, 0.6	all	0.85	0.001	$GO1_2$, $GO1_5$, $GO1_6$, $GO1_7$, $GO1_9$	0.3	$GO2_5$
Image		0.01	all	0.6	all		0.001	$GO1_6$	0.3, 0.95	$GO2_5$
Glass	0.1	0.01, 0.3	all		all	0.85	0.001	$GO1_6$		$GO2_5$
German	0.1	0.3	$GO1_1$, $GO1_3$, $GO1_4$, all		all	0.85	0.01	$GO1_2$, $GO1_5$, $GO1_6$, $GO1_7$, $GO1_8$, $GO1_9$		$GO2_5$

Table 4. Main effect values.

It is interesting to notice that a high elitism degree (ED = 0.85), a very low mutation rate (MP = 0.001) and the $GO2_5$ operator consistently lead to worse results. On the other hand, a high mutation probability (MP = 0.3), low elitism degree (ED = 0.1) seems to lead to good results. A high mutation rate results in superior performance emphasizing the importance of the genetic operator mutation. This is interesting because usually the genetic operator mutation is only supposed to play a minor role for genetic programming style EAs, which utilise tree-shaped representation schemes. On the other hand, the genetic operator crossover is suppose to play a major role. Please note that, in the context of this study, the crossover probability has no beneficial main effect on the performance for some data sets. Table 4 also shows that the deployment of several genetic operators can lead to superior results. This is especially true for the GO2 operators. However, as we have also detected significant interaction effects, we have to be cautious in interpreting the obtained results because the parameter values have always to be considered in conjunction with other parameter values. For example, we have noticed (when considering interaction plots) that high mutation probabilities (MP=0.3) are only beneficial when the elitism degree is also high (ED=0.85). On the other hand, lower mutation probabilities (MP=0.01) lead to better results for lower elitism degrees (ED=0.1), confirming the findings of Laumanns et al. [36]. Hence one has to be careful to conclude that an elitism degree of 0.1 is generally beneficial. In summary we can answer the question posed in section 4 as follows:

1. There exist no unique parameter set that leads to better performance of the approach on all data sets.
2. None of the recommendations of De Jong, Greffenstette and Laumanns et al. are directly applicable for our approach.
3. The deployment of several genetic operators can be beneficially.
4. We could not determine any genetic operators that are superior to other genetic operators. However, it seems that the operator $GO2_5$ (when deployed alone) leads to inferior results.

5.2 Results Performance Evaluation

This section compares MERBIS with some of the approaches investigated in [17]. Unfortunately, we could not determine a unique parameter set that results in superior performance of the approach (see section 5.1). We decided to use the parameter values summarised in table 5. These parameter values occurred relatively consistently in conjunction with good performance (see table 4 in section 3).

Name	**Value**
Population Size	100
Number of Generation	100
Maximum Number of Rules	50
Maximum Number of Antecedent Nodes	3
Elitism Degree (ED)	0.1
One Individual Genetic Operator Intensity Probability (IPGO1)	*all*
Mutation Probability (MP)	0.3
Two Individuals Genetic Operator Intensity Probability	*all*
Crossover Probability (CP)	0.6

Table 5. The parameter values used for the experiments.

We report the performance of MERBIS for the chosen data sets in table 6.

As the output of MERBIS is not one solution but rather a set of solutions (the non-dominated solutions within the archive) we report the average AUC values of the worst individuals (MIN) and the best individuals (MAX), together with the standard deviation. We also report two different average AUC values because we have actually utilised the AUC measure put forward by Hand et al. [23] (denoted as Hand AUC in table 6) as our performance measure (see equation 1 in section 3.2) rather than the AUC measure put forward by Fawcett [17] (denoted as Fawcett AUC in table 6). To report both measures enables us to compare MERBIS with the approaches in [17]. The results of this comparison are shown in table 7.

Data Set	Fawcett AUC		Hand AUC		Generated Rules	
	Min	Max	Min	Max	Min	Max
Breast-Wisc	97.2 ± 2.2	98.6 ± 1.3	97 ± 2.7	98.8 ± 1.1	2.0	6.4
Car	66.6 ± 7.7	79.8 ± 3.7	76.2 ± 3.7	86.1 ± 1.6	4.0	21.0
Cmc	57.1 ± 3.4	60.8 ± 3.8	66.8 ± 2.9	70.0 ± 3.2	3.0	8.8
Crx	92.7 ± 3.4	93.2 ± 3.3	92.3 ± 3.6	93.1 ± 3.5	3.0	11.3
German	60.1 ± 3.4	73.6 ± 3.3	67.3 ± 3.3	73.3 ± 3.7	2.0	8.7
Glass	55.2 ± 14.2	72.9 ± 5.8	76.4 ± 3.7	82.6 ± 2.9	6.0	11.5
Image	81.5 ± 6.8	86.0 ± 5.0	87.7 ± 5.5	91.8 ± 4.5	7.0	10.2
Kr-vs-kp	67.0 ± 2.6	88.0 ± 4.3	67.6 ± 2.2	88.0 ± 4.0	2.0	6.4
Mushroom	88.1 ± 3.0	98.2 ± 1.4	85.2 ± 2.8	98.0 ± 1.4	2.0	18.1
Nursery	64.2 ± 3.1	87.8 ± 1.2	75.6 ± 3.0	94.9 ± 1.0	5.0	31.4
Promoters	61.5 ± 12.6	89.3 ± 8.4	61.5 ± 12.6	89.3 ± 8.4	2.0	21.4
Sonar	75.5 ± 9.5	80.0 ± 8.0	75.2 ± 10.1	79.7 ± 7.8	2.0	5.3
Splice	50.5 ± 5.3	82.9 ± 4.0	59.8 ± 5.3	83.8 ± 3.6	3.0	34.6

Table 6. Results generated by the MERBIS approach.

Data Set	RL			C4.5rules WVote		MERBIS	
	Best	Rules	Naive Bayes	Results	Rules	Results	Rules
Breast-wisc	97.6 ± 1.3	306.5	93.1 ± 5.5	97.4 ± 3.6	8.2	98.6 ± 1.3	6.4
Car	94.3 ± 1.4	107.6	92.3 ± 2.2	98.3 ± 0.7	78.6	79.8 ± 3.7	21.0
Cmc	63.9 ± 4.0	196.6	64.1 ± 5.8	66.5 ± 4.9	39.1	60.8 ± 3.8	8.8
Crx	90.2 ± 4.2	758.5	87.6 ± 4.3	90.1 ± 3.4	12.9	93.2 ± 3.3	11.3
German	71.9 ± 4.9	807.5	77.1 ± 4.5	67.9 ± 9.6	23.4	73.6 ± 3.3	8.7
Glass	74.4 ± 10.0	183.7	74.0 ± 8.7	75.7 ± 5.9	12.2	72.9 ± 5.8	11.5
Image	93.3 ± 1.4	811.4	95.6 ± 0.9	99.0 ± 0.5	28.6	86.0 ± 5.0	10.2
Kr-v-kp	92.6 ± 1.5	2328.3	95.1 ± 0.8	99.7 ± 0.2	26.3	88.0 ± 4.3	6.4
Mushroom	100.0 ± 0.0	2362.2	99.8 ± 0.1	100.0 ± 0.0	11.5	98.2 ± 1.4	18.1
Nursery	97.1 ± 0.2	606.6	98.0 ± 0.2	99.8 ± 0.1	336.8	87.8 ± 1.2	31.4
Promoters	83.5 ± 16.2	7432.2	97.7 ± 4.0	88.4 ± 12.8	8.0	89.3 ± 8.4	21.4
Sonar	65.8 ± 12.8	10075.7	76.1 ± 13.0	77.8 ± 13.7	9.1	80.0 ± 8.0	5.3
Splice	87.3 ± 1.6	8406.8	99.2 ± 0.6	97.2 ± 0.7	76.2	82.9 ± 4.0	34.6

Table 7. Comparison of MERBIS with other approaches.

The second column in table 7 contains the best results that Fawcett obtained for the RL approach [7]. The columns three and four contain the results that Fawcett obtained for the Naive Bayes and the C4.5rules approach [43]. The last column contains the results of the MERBIS approach. Our approach exhibits higher average AUC values for the Breast-wisc, Crx and the Sonar data sets. Unfortunately, for the remaining data sets we obtained lower average AUC values. This might be attributed to a sub-optimal parameter set. Table 7 shows that our approach produced fewer rules in comparison to the RL approach. This is also true for most data sets in comparison to the C4.5rules approach.

6 Concluding Remarks and Further Developments

This paper has introduced MERBIS. We have demonstrated that this approach is capable of producing comprehensible and accurate fuzzy classification rule systems. We have attempted to determine an advantageous parameter set for our approach and demonstrated that there is no unique parameter set that leads to good performance of our approach on several data sets. It was also shown that parameter values recommended in the literature are not directly applicable for our approach and that the deployment of several genetic operators can be beneficial. Future research will therefore focus upon the deployment of adaptable parameters. In brief, when deploying adaptable parameters, the EA not only searches for good solutions for the given problem but also for the right parameters at the same time (e.g. [16]). Future research will also focus upon the implementation of a coarse-grained parallel version of the MERBIS approach because the current implementation does not scale up for large data sets. Coarse-grained parallel EAs are a special type of parallel EAs and we shown in an earlier study that a coarse-grained approach scales for large data sets [46]. We also intend to compare our approach with more classification approaches reported in [38] on a larger number of data sets.

References

1. C. Apte and S. Weiss. Data mining with decision trees and decision rules. *Future Generation Computer Systems*, 13:197–210, 1997.
2. Thomas Bäck, Frank Hoffmeister, and Hans-Paul Schwefel. Applications of evolutionary algorithms. Technical Report SYS-2/92, 1993.
3. Wolfgang Banzhaf, Peter Nordin, and Frank D. Francone Robert E. Keller. *Genetic Programming : An Introduction - On the Automatic Evolution of Computer Programs and Its Applications*. Morgan Kaufmann Publishers, 1998.
4. Peter J. Bentley. Evolutionary, my dear watson - investigating committee-based evolution of fuzzy rules for the detection of suspicious insurance claims. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000)*, pages 702–709, 2000.
5. A. Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.
6. Jorge Casillas, Oscar Cordon, Francisco Herrera, and Luis Magdalena. Finding a balance between interpretability and accuracy in fuzzy rule-based modelling: An overview. Technical report, 2001.
7. S. Clearwater and F. Provost. Rl4: A tool for knowledge-based induction. In *Proceedings of the Second International IEEE Conference on Tools for Artificial Intelligence*, pages 24–30, 1990.
8. O. Cordon, M. Jesus, and F. Herrera. A proposal on reasoning methods in fuzzy rule-based classification systems. *International Journal of Approximate Reasoning*, 20:21–45, 1999.
9. Nichael Lynn Cramer. A representation for the adaptive generation of simple sequential programs. In *PROCEEDINGS OF AN INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS AND THEIR APPLICATIONS*, 1985.
10. Charles Robert Darwin. *On the origin of species by means of natural selection*. Murray, London, 1860.
11. Lawrence Davis. *Handbook of Genetic Algorithms*. International Thomson Computer Press, 1991.

12. Kalyanmoy Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley Europe, 2001.
13. Vasant Dhar, Dashin Chou, and Foster J. Provost. Discovering interesting patterns for investment decision making with glower - a genetic learner overlaid with entropy reduction. *Data Mining and Knowledge Discovery*, 4(4):251–280, 2000.
14. W. Duch, N. Jankowski, K. Grabczewski, and Rafał Adamczak. Optimization and interpretation of rule-based classifiers. In *Intelligent Information Systems*, Advances in Soft Computing, pages 1–14, 2000.
15. R. O. Duda, P. E. Hart, and H. G. Stork. *Pattern Classification*. Wiley-Interscience, New York, 2000.
16. Ágoston Endre Eiben, Robert Hinterding, and Zbigniew Michalewicz. Parameter control in evolutionary algorithms. *IEEE Trans. on Evolutionary Computation*, 3(2):124–141, 1999.
17. Tom Fawcett. Using rule sets to maximize roc performance, 2001. Presented at the 2001 IEEE International Conference on Data Mining (ICDM-01).
18. U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. In *Advances in Knowledge Discovery and Data Mining*, pages 1–36. 1996.
19. A. Freitas. Evolutionary algorithms. In *Handbook of Data Mining and Knowledge Discovery*. 2000.
20. A. Freitas. *Data Mining and Knowledge Discovery With Evolutionary Algorithms*. Springer Verlag, 2002.
21. Johannes Fürnkranz. Separate-and-conquer rule learning. *Artificial Intelligence Review*, 13(1):3–54, February 1999.
22. J. Grefenstette. Optimization of control parameters for genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics, SMC16*, 1:122–128, 1986.
23. David J. Hand and Robert J. Till. A simple generalization of the area under the roc curve for multiple class classification problems. *Machine Learning*, 45:171–186, 2001.
24. J. Hanley and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143:29–36, 1982.
25. H.Ishibuchi, T.Nakashima, and T. Murata. Three-objective genetics-based machine learning for linguistic rule extraction. *Information Sciences*, 136:109–133, 2001.
26. David C. Howell. *Statistical Methods for Psychology*. Wadsworth, 1996.
27. M. Humphrey, S. Cunningham, and I. Witten. Knowledge visualization techniques for machine learning. *Intelligent Data Analysis*, 2:333–347, 1998.
28. John F. Elder IV and Daryl Pregibon. A statistical perspective on knowledge discovery in databases. In *Advances in Knowledge Discovery and Data Mining*, pages 83–113. 1996.
29. James Jaccard. *Interaction effects in factorial analysis of variance*. Thousand Oaks : Sage Publications, 1998.
30. K. A. De Jong. An analysis of the behaviour of a class of genetic adaptive systems, 1975. PhD thesis, University of Michigan.
31. George J. Klir, Ute S. Clair, and Bo Yuan. *Fuzzy Set Theory: Foundations and Applications*. Prentice Hall, 1997.
32. J. Knowles and D. Corne. On metrics for comparing non-dominated sets. In *In Proceedings Congress on Evolutionary Computation (CEC 2002)*, 2002.
33. M. Korver and P. Lucas. Converting a rule-based expert system into a belief network. *Medical Informatics*, 18(3):219–241, 1993.
34. John R. Koza. Genetic programming. In James G. Williams and Allen Kent, editors, *Encyclopedia of Computer Science and Technology*, volume 39, pages 29–43. Marcel-Dekker, 1998.
35. Ludmila I. Kuncheva. *Fuzzy classifier design*. Heidelberg : Physica-Verlag, 2000.

36. M. Laumanns, L. Thiele, K. Deb, and E. Zitzler. Archiving with guaranteed convergence and diversity in multi-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 439–447. Morgan Kaufmann Publishers, 2002.
37. Marco Laumanns, Eckart Zitzler, and Lothar Thiele. On the effects of archiving, elitism, and density based selection in evolutionary multi-objective optimization. In *Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization*, pages 181–196. Springer Verlang, 2001.
38. Tjen-Sien Lim, Wei-Yin Loh, and Yu-Shan Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine Learning*, 40(3):203–228, 2000.
39. Ryszard S. Michalski, Ivan Bratko, and Miroslav Kubat. *Machine learning and data mining : methods and applications*. Chichester : Wiley, 1998.
40. Andreas Nürnberger, Aljoscha Klose, and Rudolf Kruse. Analyzing borders between partially contradicting fuzzy classification rules. In *Proceedings of 19th International Conference of the North American Fuzzy Information Processing Society (NAFIPS 2000)*, pages 59–63, 2000.
41. M. Pazzani, S. Mani, and W. Shankle. Comprehensible knowledge discovery in databases. In *Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society*, pages 596–601, 1997.
42. Robin C. Purshouse and P.J. Fleming. An adaptive divide-and-conquer methodology for evolutionary multi-criterion optimisation. In *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization*, pages 133–147, 2003.
43. J. R. Quinlan. *C4.5 : programs for machine learning*. Morgan Kaufmann, 1994.
44. Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1994.
45. J. David Schaffer, R.A. Carana, L.J. Eshelman, and R. Das. A study of control parameters affecting online performance of genetic algorithms for function optimization, 1989.
46. C. Setzkorn and R.C. Paton. The scalability of javaspaces for the implementation of different parallel evolutionary algorithms, 2002. submitted for publication.
47. Bernard W. Silverman. *Density estimation for statistics and data analysis*. Chapman and Hall, 1999.
48. S.F. Smith. A learning system based on genetic algorithm, 1980. Ph.D. dissertation, University of Pittburgh.
49. William M. Spears. Adapting crossover in evolutionary algorithms. In J. R. McDonnell, R. G. Reynolds, and D. B. Fogel, editors, *Proc. of the Fourth Annual Conference on Evolutionary Programming*, pages 367–384, Cambridge, MA, 1995. MIT Press.
50. Friedrich Steimann. Fuzzy set theory in medicine. *Artificial Intelligence in Medicine*, 11(1):1–7, 1997.
51. Ingo Wegener. On the design and analysis of evolutionary algorithms. In *Proceedings of Workshop on Algorithm Engineering as a New Paradigm*, pages 36–47, 2000.
52. S. Weiss and N. Indurkhya. Reduced complexity rule induction. In *In Proceedings of the Twelfth IJCAI*, pages 678–684, 1991.
53. S. Weiss and C. Kulikowski. *Computer Systems That Learn*. Morgan Kaufmann, 1991.
54. L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.
55. E. Zitzler, M. Laumanns, L. Thiele, C. M. Fonseca, and V. Grunert da Fonseca. Why quality assessment of multiobjective optimizers is difficult. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2002)*, pages 666–674, 2002.
56. E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. *IEEE-EC*, 3(4):257, November 1999.

57. Eckart Zitzler, Marco Laumanns, , and Lothar Thiele. Spea2: Improving the strength pareto evolutionary algorithm. In *EUROGEN 2001 - Evolutionary Methods for Design, Optimisation and Control with Applications to Industrial Problems*, 2001.