Proceedings of the 2007 IEEE Symposium on Computational
Intelligence in Multicriteria Decision Making (MCDM 2007)

Managing Population Diversity Through the Use of Weighted Objectives and Modified Dominance: An Example from Data Mining

Alan P. Reynolds

School of Computing Sciences
University of East Anglia
Norwich
England
Telephone: +44 (0)1603 593220
Fax: +44 (0)1603 593345
Email: ar@cmp.uea.ac.uk

Beatriz de la Iglesia

School of Computing Sciences
University of East Anglia
Norwich
England
Telephone: +44 (0)1603 592961
Fax: +44 (0)1603 593345
Email: bli@cmp.uea.ac.uk

***Abstract*— The most successful multi-objective metaheuristics, such as NSGA II and SPEA 2, usually apply a form of elitism in the search. However, there are multi-objective problems where this approach leads to a major loss of population diversity early in the search. In earlier work, the authors applied a multi-objective metaheuristic to the problem of rule induction for predictive classification, minimizing rule complexity and misclassification costs. While high quality results were obtained, this problem was found to suffer from such a loss of diversity. This paper describes the use of both linear combinations of objectives and modified dominance relations to control population diversity, producing higher quality results in shorter run times.**

I. Introduction

In earlier work, the authors [1], [2], [3], [4] applied a range of multi-objective metaheuristics to the problem of partial classification, endeavoring to find accurate descriptions of subsets of a class of interest in a database. This work in *descriptive* data mining was extended by applying NSGA II [5] to the generation of rules in the form of expression trees to act as *predictive* classifiers for two class problems [6]. Considering this as a two objective problem, minimizing rule complexity and misclassification costs, produced useful rules with different tradeoffs between the two objectives. However, preliminary investigations revealed a major loss of population diversity early in the search process.

At this point, it should be noted that research into diversity in multi-objective optimization can be divided into two areas. In the first, diversity refers to the spread of solutions across the Pareto-front, with emphasis being on the presentation of a solution set to the client. In this case, research may focus on how diversity is maintained in an external archive [7], [8] or on the application of crowding techniques after ranking solutions using dominance [9], [10]. In the second area of research, diversity refers to the range of genetic material in the population that may be usefully combined to create new solutions, with emphasis being on the quality of the search process [11], [12]. Throughout this paper, the term diversity is used in this second sense

This paper discusses two approaches for better management of population diversity. The first involves the use of three objectives: rule complexity and two carefully selected linear combinations of the number of false positives and the number of false negatives. Such an choice of objectives is suitable when the client is unsure of the precise cost of a false positive compared with a false negative. However, it will also be shown that through adaptive modification of the objectives used, three objectives can be used to control the amount of diversity in the population, resulting in a more effective search even when the client knows precisely how to calculate misclassification costs.

The second approach to diversity management requires no additional objectives but uses a modification to the dominance relation, where rules that would previously have been dominated are given a degree of leeway. Again, adaptive modification of the amount of leeway results in improved performance.

Sections II–IV provide a brief summary of previous work applying NSGA II to the problem of optimizing expression trees for predictive classification, with section II describing the problem and section III discussing the application of NSGA II. Section IV gives a sample of the results and leads into a discussion of the population diversity throughout the search.

Section V discusses, in general terms, the relationship between the dominance relation and the balance between search intensity and population diversity. It proposes a reason for the loss in diversity and suggests that NSGA II is likely to suffer a similar loss of diversity on other multi-objective optimization problems.

The use of three objectives when the client is unsure of the misclassification costs is described in section VI. Analysis of the results of this section leads into the use of three adaptive objectives to manage diversity is outlined in section VII. The alternative method of managing diversity,

This research was funded by the EPSRC, grant number GR/T04298/01.

using two objectives and modifying the dominance relation, is the subject of section VIII. The paper finishes with some conclusions in section IX and a discussion of further research in section X.

II. Multi-Objective Predictive Classification

In previous work [6], NSGA II was applied to optimize rules in the form of expression trees. A typical expression tree is shown in figure 1. Here the internal nodes contain

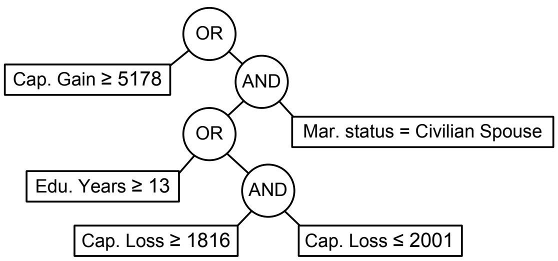

Fig. 1. A typical expression tree.

boolean operators, which are restricted to be 'AND' or 'OR' in this paper, while the leaf nodes contain simple *attribute tests*. In this work, an attribute test (AT) on a categorical field may indicate that a matching record is either of a particular category or not, e.g. `sex = male` or `color` $\neq$ `red`, while an AT on a numeric field supplies a simple bound on the field value, e.g. `age` $\geq$ `29`. (See previous work [1], [6] for more details about ATs and alternative AT types.)

Expression trees are used to predict when a record belongs to a class of interest to the client. Any record not matching the rule represented are assumed to not belong to the class of interest. The aim is to create expression trees that are accurate on unseen data but also relatively simple for a human to comprehend. Therefore two objectives were minimized:

- **Misclassification cost on training data:** Different forms of misclassification cost, calculated from the number of false positives and the number of false negatives, were used as one of the objectives. This included the simple error rate and the balanced error rate. Note that the costs on the training data provide, at best, an estimate of the misclassification costs on unseen data.
- **Rule complexity:** Rule complexity could also be measured in different ways. In this paper, a simple count of the number of ATs is used, though one could use a count of ATs in the equivalent rule set or a more complex but realistic measure that takes into account the number and type of the internal nodes [6]. While minimizing rule complexity is a goal in its own right, it is also likely to reduce the problem of *overfitting* [13].

A contrasting approach is that taken by Ishibuchi et al. [14], [15], [16], where a set of simpler rules is generated and a multi-objective metaheuristic is used to select a subset to act as a predictive classifier. However, both approaches aim to optimize both rule simplicity and accuracy.

III. Applying NSGA II

NSGA II was selected because it has not only been shown to be effective for multi-objective optimization in general, but also when optimizing classification rules [2], [3], [4], [1]. In order to apply it to the optimization of rules in the form of expression trees, it was necessary to determine how to represent ATs and how to apply genetic operators to the rules produced.

A. Attribute Test Representation

Values occuring in each field of the dataset are stored in reference arrays. The AT representation uses indices into these arrays, as shown in figure 2, rather than the original

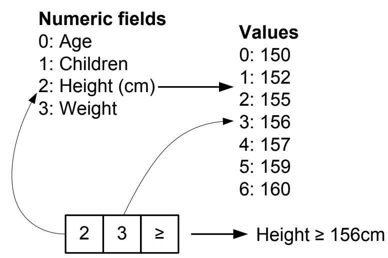

Fig. 2. Representation of an AT providing a bound on a numeric field.

field values. This reduces redundancy in the representation of numeric ATs. For example, `height` $\geq$ `151cm` and `height` $\geq$ `152cm` will match precisely the same records in the dataset, since no record has a height of 151cm. Restricting the bound on height to those values that occur in the dataset ensures that only the latter of these ATs can be generated,

B. Genetic Operators

- **Initialization:** The population is initialized with random balanced trees of depth two, where the root node is at depth zero.
- **Mutation:** When mutation is applied, there is a 50% chance that a random AT is mutated. Categorical ATs are mutated by changing the category index to a random value, while numeric ATs have the bound index changed by up to 20% of the number of values in the database for the field in question. There is a 25% chance that an AT and its parent node is removed and a 25% chance that a random AT and parent node are inserted.
- **Crossover:** Subtree crossover [17] is used, which selects two nodes at random in the tree and swaps the associated subtrees. Note that when a new rule is created, either crossover or mutation is applied, but not both.

C. Rule Simplification and the Rule Size Constraint

As rule evaluation is expensive, rules should be simplified if possible in order to reduce the number of ATs that need to be evaluated. A limited amount of rule simplification is performed after the application of crossover or mutation, as described in previous work [6].

While both rule simplification and the use of rule complexity as an objective help to keep rule sizes down, they are not sufficient to entirely remove the problem of *bloat* [18]. As an additional measure to keep rule complexity within sensible bounds, a constrain on the number of ATs in a rule is imposed. In the reported experiments, this bound is set at 20 ATs. If a rule, after simplification, is found to exceed this limit, ATs and their parent nodes are removed at random until the constraint is satisfied.

IV. Using Two Objectives

The algorithm was applied to find rule trees for a number of datasets from the UCI machine learning repository [19]. Here we summarize the results obtained using the Adult dataset. 80% of the original training set was used to train the expressions trees, with the remaining 20% used as validation data to give the (hypothetical) client some idea of how well the rules might generalize. The rule selected by the client was tested on the original test set. The simple error rate was used as the misclassification costs, with the number of ATs representing rule complexity.

Using a population size of 100 and a crossover rate of 30% — settings found previously to produce good performance [6] — and running the algorithm 30 times produced the results summarized in figure 3. Here the error bars give the standard deviation.

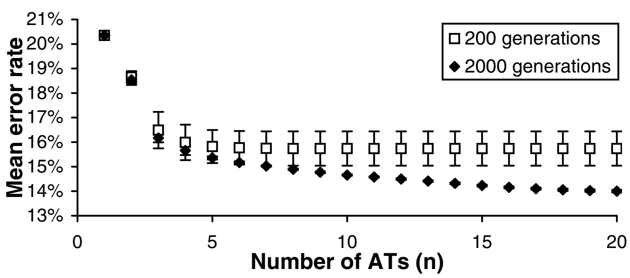

Fig. 3. Training performance of the best rule of up to n ATs, after 200 and 2000 generations.

If, after 2000 generations, the rules are presented to the client and the client selects the rule with the lowest error rate on the *validation* data, the rule selected gives, on average, a 14.45% error rate on the *test* data (standard deviation: 0.12%), which compares favorably with other algorithms. However, run times are approximately 1000 seconds on a 2GHz processor. Since applying the algorithm to larger datasets leads to a reduction in the number of generations that are practicable, performance after fewer generations is also of interest. If only 200 generations are permitted, a significant drop in performance is observed. If the client selects a rule as before, the testing error rate produced is 15.98% (standard deviation: 0.70%). The average complexity of the rule selected is also significantly lower after 200 generations (5.8 ATs) than after 2000 (19.1 ATs).

More detailed analysis of the performance of the algorithm reveals a major loss in population diversity early in the search process. Suppose we use the information entropy of the population as a measure of the its diversity. Here, entropy is defined as:

$$H = -\sum_k p_k \log_2 p_k$$

where k represents a unique solution in the population and p_k is the probability that solution selected at random is solution k. So, if the population contains 100 copies of the same solution, the population entropy is 0. If the population contains 100 unique solutions, the entropy is:

$$-\sum_{k=1}^{100} 0.01 \log_2 0.01 = -\log_2 0.01 \approx 6.644.$$

For the purposes of calculating this entropy, a recursive equality operator is defined for the rules. Two expression trees are considered equal if both the root nodes and the subtrees are equal, regardless of the ordering of the subtrees.

Notice that this is a fairly crude measure of population diversity, since it does not take into account how similar two rules are when not identical and does not consider the diversity of subtrees or attribute tests contained in the rules. However, it is sufficient to demonstrate the loss in population diversity. Figure 4 shows the evolution of population entropy, averaged over 30 runs. The average minimum entropy of the thirty runs was 0.176, with several runs reaching the point of having 99 identical solutions in a population of 100. Alternative methods for population initialization have been tried, for example creating rules of random depths, but this loss of population diversity occurs regardless.

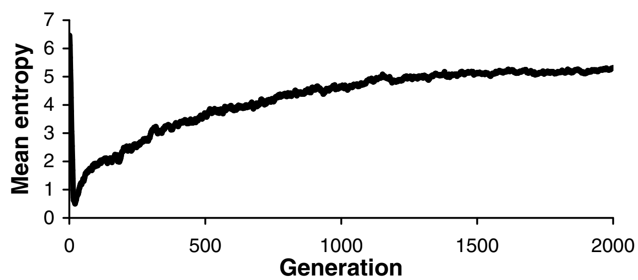

Fig. 4. Plotting population entropy against time reveals a major loss in population diversity, followed by a slow increase due to fortunate mutations.

It is to be expected that such a drastic loss of diversity would impair the performance of the algorithm, especially over shorter runs when there is insufficient time to reintroduce diversity through mutation. It is therefore necessary to look at techniques for preventing or reducing this effect.

V. Population Diversity in Multi-Objective GAs

Typically, a multi-objective problem has the property that a set of solutions contains a reasonable number of non-dominated solutions, contrasting with the case in single objective optimization. The result is that a genetic algorithm performs a less focused search in a multi-objective environment but has little difficulty in maintaining population diversity. In both NSGA II and SPEA 2, elitist measures (such as keeping high quality solutions from generation to generation and eliminating all but the best solutions according to dominance) are used to correct the balance between exploiting promising areas of the search space and maintaining population diversity, providing additional focus to the search.

Early in the search for good expression trees, the algorithm tends to favour smaller rules, since the algorithm has not had the time to build the good subtrees from which a more accurate and larger tree can be built. It is relatively simple to find a single AT with a reasonable error rate, but building larger rules with better error rates is a more complicated matter. The result is that, early in the search, few unique solutions (indeed often only one) are non-dominated. Hence it is no longer possible to rely on the presence of large numbers of non-dominated solutions to provide population diversity and NSGA II and SPEA 2 become *too* elitist, converging prematurely.

This is unlikely to be an issue solely for the problem of expression tree optimization. For example, applying these algorithms to problems such as the optimization of correlated multi-objective minimum spanning trees [20] can be expected to result in the same loss of diversity, due to the correlation between the objectives in the early stages of the search. Optimizing one objective results in the optimization of the other, resulting in few non-dominated solutions in the population.

Much of the remainder of this paper illustrates how the fraction of non-dominated solutions can be artificially increased in order to counteract this loss of diversity, by modifying the objective functions used. It will be shown that, perhaps surprisingly, better performance can be obtained by using these modified objective functions than by using the true objectives.

VI. Using Three Objectives

Up until now, it has been assumed that the client knows precisely how to calculate the misclassification costs, i.e. the client knows the relative costs of a false negative and a false positive. However, in reality the client may be in some doubt as to how to calculate this cost. In such a case it makes sense to treat the problem as having three objectives. This section describes this in more detail, but also describes how the use of three carefully selected objectives can result in improved performance on the two objective problem.

A. The Client is Unsure of Misclassification Costs

When the client is unsure about how misclassification costs should be calculated, it is tempting to simply revert to a three objective problem, minimizing rule complexity, the number of false positives and the number of false negatives. This is appropriate if the client has absolutely no idea how misclassification costs are calculated. However, in practice, the client is likely to have a rough idea and this knowledge should be used to improve the search.

Misclassification costs are typically of the form $\lambda FP + \mu FN$, where FP and FN indicate the number of false positives and false negatives respectively. The client may know that μ/λ should take a value between $2/3$ and $3/2$. In this case, the following three objectives should be used:

- Rule complexity,
- $0.4FP + 0.6FN$,
- $0.6FP + 0.4FN$.

This ensures that rules that are possibly optimal with regards to the true misclassification cost and rule complexity, and only these rules, are Pareto-optimal. Alternatively, the client may indicate how much he is willing to pay in terms of one objective for a specified improvement in the other. This similarly leads to the use of linear combinations of objectives.

The effect of using linear combinations in this way is illustrated in figure 5. This figure shows a slice through the

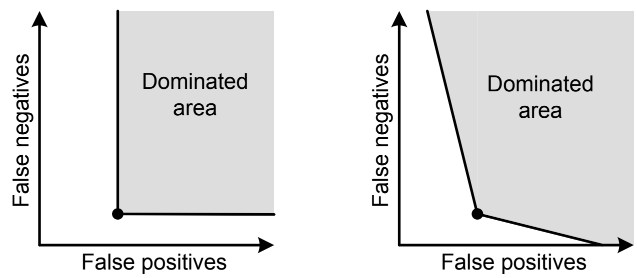

Fig. 5. Dominated areas when minimizing FP and FN and when minimizing $0.8FP + 0.2FN$ and $0.2FP + 0.8FN$.

objective space at constant rule complexity. The first graph indicates how much of the objective space is dominated according to the basic objectives, while the second indicates how much is dominated if the client has merely indicated that μ/λ should be between 0.25 and 4. The modified objectives ensures that the client is presented with only those solutions that he is likely to find interesting and also provides an added push towards the Pareto-front.

B. Using Three Objectives to Maintain Diversity

The algorithm was reapplied to the adult dataset using the same parameter settings but with different sets of three objectives. In each case, one objective was the rule complexity. The two remaining objectives were $\lambda FP + (1 - \lambda)FN$ and $(1 - \lambda)FP + \lambda FN$, with λ equal to 0, 0.1, 0.2, 0.3 and 0.4. A value of zero indicates that the algorithm is using the three basic objectives, while a value of 0.5 is effectively the same as minimizing rule complexity and the simple error rate. Thirty runs were performed for each choice of objectives.

After performing these experiments, the non-dominated rules under the original two objectives of error rate and rule simplicity were extracted. It might be expected that there would be a drop in performance compared with the two objective algorithm, since the algorithm no longer singles out the error rate to be optimized. Figure 6 shows a comparison of the results obtained after 200 generations. The results are scaled with respect to those obtained using just two objectives to improve clarity. Note that using the three basic objectives of number of false positives, number of false

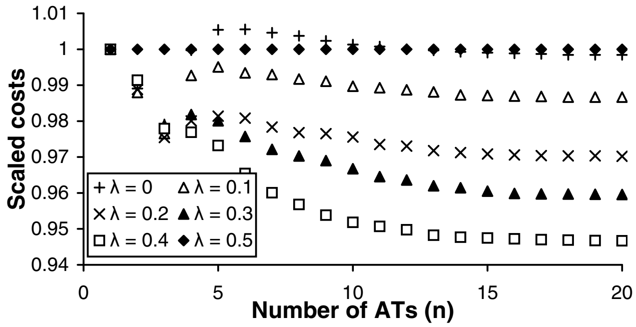

Fig. 6. Comparison of results at 200 generations using different combinations of objectives. Training error rates are scaled with respect to the two objective case ($\lambda = 0.5$).

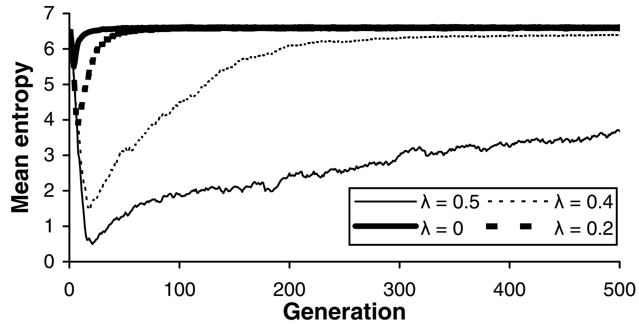

Fig. 8. Population diversity during the search using different combinations of objectives.

negatives and rule complexity ($\lambda = 0$) resulted in rules that performed similarly to those obtained with just two objectives (or $\lambda = 0.5$) when compared on rule complexity and error rate, with better performance on rules of size 2 and 3. Furthermore, using linear combinations for the objectives resulted in better performance across all rule sizes.

Figure 7 shows a comparison of performance at 2000 generations, including 95% confidence intervals. Here the two objective approach outperforms all of the three objective approaches, with the exception of $\lambda = 0.4$, for the larger rules. However, using three objectives still results in better performance for the smaller rules.

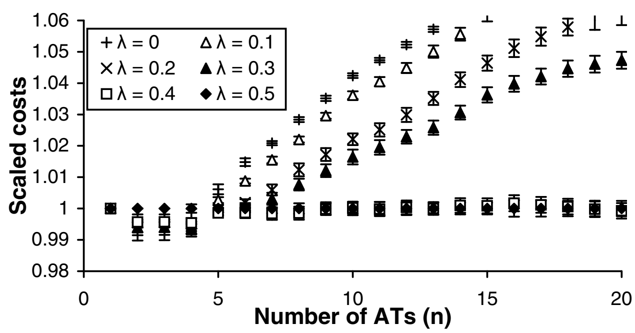

Fig. 7. Comparison of results at 2000 generations using different combinations of objectives. Training costs are scaled with respect to the two objective case ($\lambda = 0.5$).

The hypothesis is that the use of three objectives reduces the loss of diversity that occurs early in the search. This loss of diversity severely reduces the effectiveness of the search when using only two objectives. However, using the basic set of three objectives provides insufficient direction to the search to find the best large rules under error rate and rule simplicity. Using two cost objectives close enough to the true error rate to provide direction but different enough to maintain diversity produces improved results. Figure 8 shows how population entropy varies with time for the different sets of objectives, confirming that the use of three objectives reduces the drop in population diversity.

VII. Using Adaptive Objectives

Results obtained using three objectives suggest that when attempting to minimize error rate and rule complexity, three objectives should be used early in the search, with the two cost objectives some distance from the true error rate, in order to encourage diversity in the population. However, later in the search, the cost objectives should be closer (or identical) to the true error rate, in order to push the search towards the true Pareto-front. Two approaches have been considered.

- **Fixed objective schedule:** A schedule giving the objectives to be used at each stage in the search could be provided in advance. A suitable schedule results in additional emphasis on population diversity and exploring the search space in the early parts of the search and more emphasis on exploitation of good solution features later on, much like in simulated annealing. However, it is not obvious how to create a suitable schedule.
- **Adaptive objectives:** The approach taken in this paper is to use the population diversity measure — in this work, the entropy — to determine when it is necessary to move cost objectives away from the true cost in order to increase diversity or when it is possible to move the objectives towards the true cost without too great a loss of diversity, guiding the search more towards those solutions that are truly of interest.

In the experiments reported, the search starts with λ set to zero. The entropy of the population is measured at each iteration and λ is increased by 0.01, up to a value of 0.5, whenever the entropy is greater than five and decreased by 0.01, provided it is greater than zero, whenever the entropy is less than three.

Figure 9 shows a comparison of the results obtained after 2000 generations using two objectives and using adaptive objectives. For clarity, results have been scaled to emphasize the difference between algorithms rather than the difference between rule sizes. It can be seen that the use of adaptive objectives results in a significant improvement in performance. If the client selects the rule with the lowest validation error rate, the use of adaptive objectives gives an improvement from a mean error rate of 14.45% (standard deviation: 0.12%) to 14.17% (standard deviation: 0.15%).

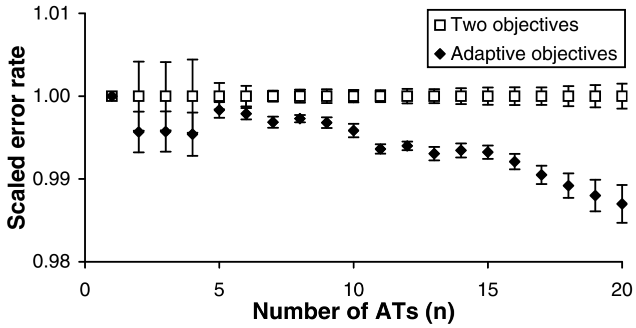

Fig. 9. Mean error rates on training data, obtained using 2000 generations using either two objectives or three adaptive objectives. Results rates are scaled with respect to those obtained for two objectives and error bars give 95% confidence intervals.

The improvement obtained after only 200 generations can be clearly seen in figure 10 without any scaling of the

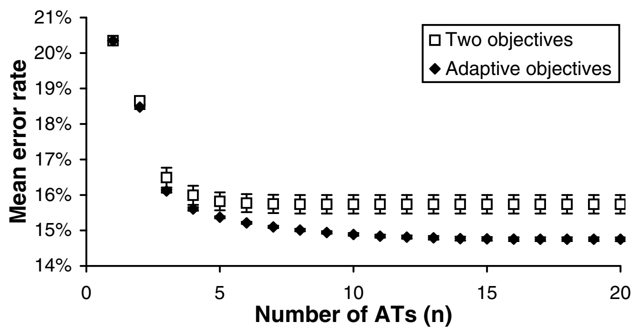

Fig. 10. Mean error rates on training data, obtained after 200 generations using either two objectives or three adaptive objectives, with 95% confidence intervals.

objectives. If the client selects the rule with the lowest validation error rate, the use of adaptive objectives gives an improvement from a mean error rate of 15.98% (standard deviation: 0.70%) to 15.00% (standard deviation: 0.25%).

VIII. Modifying the Dominance Relation

While the techniques of the previous two sections have proved to be successful at maintaining diversity and improving the quality of the expression trees produced, there remains the question of how widely these techniques can be applied, since they requires the ability to break one objective into two or more parts. With ingenuity, this may be possible more frequently that appears at first sight. For example, when solving multi-objective minimum spanning tree problems, an objective can be split by randomly partitioning the edges of the graph and taking the sum of the edge costs in each partition. However, this will not always be the case.

An alternative is to retain the original objectives and modify the dominance relation in a way that might encourage population diversity. The approach taken here is to allow a certain amount of leeway in the misclassification cost of a rule, as shown in figure 11. The resulting dominance relation

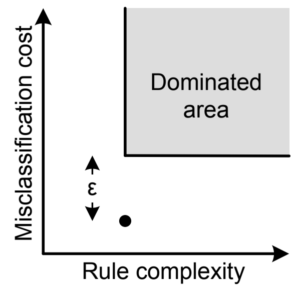

Fig. 11. Modifying the dominance relation to allow some leeway of rule misclassification cost.

states that rule r_1 dominates rule r_2 if and only if:

$$cost(r_1) < cost(r_2) - \epsilon \text{ and } comp(r_1) \leq comp(r_2) \text{ or}$$
$$cost(r_1) \leq cost(r_2) - \epsilon \text{ and } comp(r_1) < comp(r_2),$$

where $cost$ represents the misclassification costs and $comp$ is the rule complexity.

Note that this use of a leeway works in the opposite way to ϵ-dominance [8], [21]. When using ϵ-dominance, a solution is permitted to dominate more of the objective space, including solutions that may have slightly better values for one or more objectives. This is typically used to control the spread — across the Pareto-front — of solutions stored in an external archive. Here, however, we wish to increase the number of non-dominated solutions to maintain population diversity, so solutions are permitted to dominate less of the objective space.

Figures 12 and 13 show the effect of different leeway

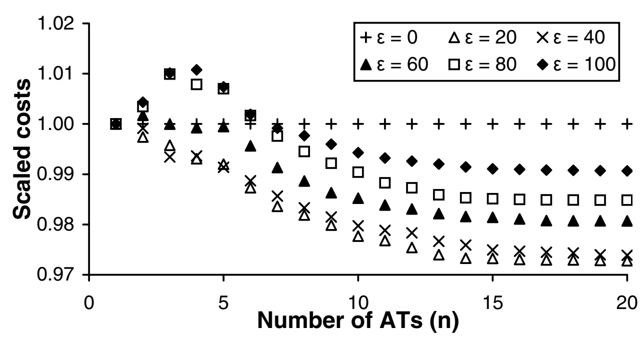

Fig. 12. Comparison of results at 200 generations, using different amounts of leeway, ϵ. Training costs are scaled with respect to the case when there is no leeway ($\epsilon = 0$).

sizes on the results obtained at 200 and 2000 generations respectively. Here, the value for ϵ indicates the leeway size in terms of the number of extra false outcomes permitted on the training data. These results are broadly similar to those obtained when encouraging diversity through the use of three objectives. Furthermore, the drop in population diversity is also affected in a similar way, as shown in figure 14.

As before, these results suggest giving a lot of leeway early in the search, in order to encourage diversity, but less in later stages of the search, to focus the search on the rules that are truly of interest. Therefore, the value

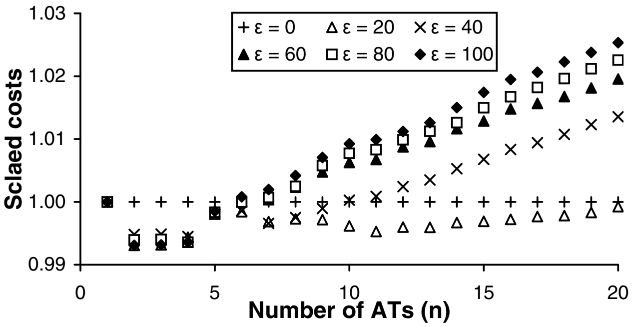

Fig. 13. Comparison of results at 2000 generations, using different amounts of leeway, ϵ. Training costs are scaled with respect to the case when there is no leeway ($\epsilon = 0$).

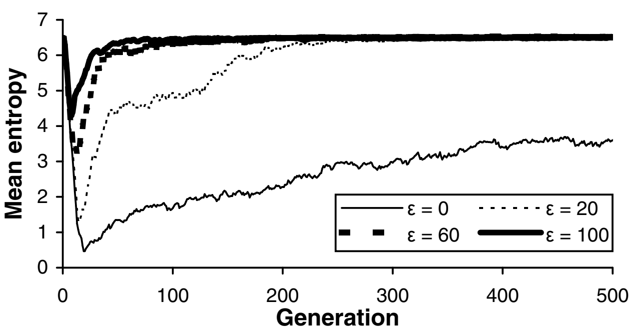

Fig. 14. Population diversity during the search using different degrees of leeway, ϵ.

of ϵ was initialized at 100 false outcomes, reduced by two false outcomes whenever the entropy exceeded five and increased by two false outcomes whenever the entropy dropped below three. Figures 15 and 16 show a comparison

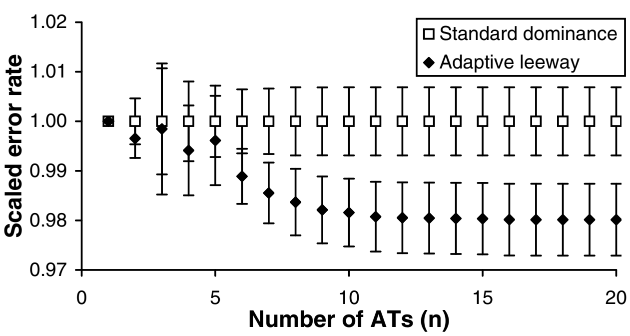

Fig. 15. Comparison of results obtained after 200 generations, using either the standard dominance relation ($\epsilon = 0$) or the adaptive dominance relation. Training error rates are scaled with respect to those obtained with the standard dominance relation. Error bars show 95% confidence intervals.

of the results obtained after 200 and 2000 generations. The use of an adaptive leeway has resulted in improved results at both 200 and 2000 generations, though the improvement is less marked than when three adaptive objectives were used. If the client selects the rule with the lowest validation error rate, the use of an adaptive leeway produces a mean error rate of 14.25% (standard deviation: 0.19%) after 2000

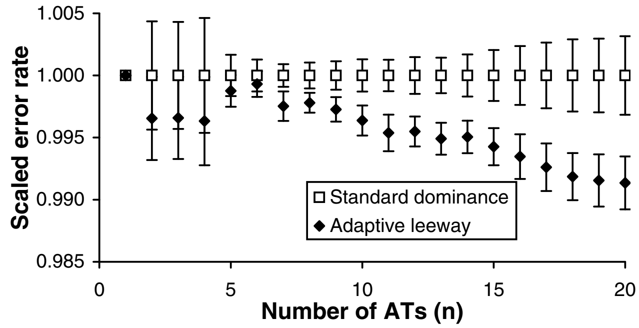

Fig. 16. Comparison of results obtained at 2000 generations, either the standard dominance relation ($\epsilon = 0$) or the adaptive dominance relation. Training error rates are scaled with respect to those obtained with the standard dominance relation. Error bars show 95% confidence intervals.

generations and 15.41% (standard deviation: 0.34%) if only 200 generations are performed.

IX. Conclusions

When optimizing expression tree based rules using NSGA II, there is a severe loss of population diversity early in the search process. At this stage in the search, there is little reason to expect a large rule to have lower misclassification costs, with the result that one small rule with low costs can dominate most and sometimes all of the remaining population. The resultant lack of an appreciable number of non-dominated solutions results in NSGA II being *too* elitist.

This reduction in population diversity can also be expected whenever a multi-objective problem has few, positively correlated objectives, since optimizing one objective will tend to optimize the others early in the search. Again, the result is that there are few non-dominated solutions.

We have illustrated two methods for maintaining population diversity, each of which can be applied without making major changes to the algorithm being applied. The first splits an objective into two component parts, each of which can be separately optimized. By using different linear combinations of the new objectives, a balance between search intensity and population diversity can be maintained. In the example problem, this approach has an additional advantage: the new non-dominated solutions are those that might be truly non-dominated if the client is unsure of the misclassification costs. In any case, such rules are likely to contain useful subtrees that might otherwise be eliminated from the population.

While it may not always be apparent how to apply this first approach to a multi-objective problem, the second approach, in which the dominance relation is modified in order to give solutions a little leeway in terms of one objective, is applicable to any multi-objective problem. Both approaches provide a parameter that may be adjusted depending on the amount of diversity in the population, resulting in a feedback control mechanism that permits increased control of population diversity. The result is the production of algorithms showing significant improvements in performance.

X. Further Research

There are a number of areas for further research:

- The results presented were all obtained using the same values for population size and crossover rate. These settings were those found to produce the best results using the standard two objective algorithm. However, the severe loss of diversity is likely to have influenced the values for these parameters. For example, the relatively low crossover rate may be partly due to the fact that crossover will be ineffective when the population consists primarily of many copies of one solution. When population diversity is maintained, improved results might be obtained with a higher crossover rate.
- We have performed little experimentation with how the feedback mechanisms in the adaptive algorithms work. For example, a preliminary examination of the progress of the three objective algorithm, with adaptive objectives, suggests that the effect of changing λ by 0.01 is greater when λ is closer to 0.5. This might suggest the use of smaller step sizes in this case.
- Similarly, the entropy values at which λ and ϵ change were chosen intuitively. Research and experimentation into the ideal range of population entropy, and how this range changes with population size and problem type, would be useful. Also, it has been noted that the entropy measure used was fairly crude; research into more suitable measure of population diversity would be useful.
- The effects of population diversity loss in other problem domains should be studied. The diversity maintenance techniques described and others should be applied in those domains where loss of population diversity appears to be a problem.

References

[1] Alan Reynolds and Beatriz de la Iglesia. Rule Induction Using Multi-Objective Metaheuristics: Encouraging Rule Diversity. In *Proceedings of the 2006 IEEE World Congress on Computational Intelligence*, pages 6375–6382. IEEE, 2006.

[2] B. de la Iglesia, M. S. Philpott, A. J. Bagnall, and V. J. Rayward-Smith. Data Mining Rules Using Multi-Objective Evolutionary Algorithms. In *Proceedings of 2003 IEEE Congress on Evolutionary Computation*, pages 1552–1559, 2003.

[3] Beatriz de la Iglesia, Alan Reynolds, and Vic J Rayward-Smith. Developments on a Multi-Objective Metaheuristic (MOMH) Algorithm for Finding Interesting Sets of Classification Rules. In *Evolutionary Multi-Criterion Optimization: Third International Conference, EMO 2005*, number 3410 in Lecture Notes in Computer Science, pages 826–840, March 2005.

[4] B. de la Iglesia, G. Richards, M. S. Philpott, and V. J. Rayward-Smith. The application and effectiveness of a multi-objective metaheuristic algorithm for partial classification. *European Journal of Operational Research*, 169(3):898–917, 2006.

[5] Kalyanmoy Deb, Samir Agrawal, Amrit Pratab, and T. Meyarivan. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In *Proceedings of PPSN VI*, number 1917 in Lecture Notes in Computer Science, pages 849–858. Springer, 2000.

[6] Alan Reynolds and Beatriz de la Iglesia. Rule Induction for Classification Using Multi-Objective Genetic Programming. In *The Fourth International Conference on Evolutionary Multi-Criterion Optimization (EMO 2007) (submitted)*, 2007.

[7] Marco Laumanns, Lothar Thiele, Kalyanmoy Deb, and Eckart Zitzler. Combining Convergence and Diversity in Evolutionary Multi-Objective Optimization. *Evolutionary Computation*, 10(3):263–282, 2002.

[8] Marco Laumanns, Lothar Thiele, Eckart Zitzler, and Kalyanmoy Deb. Archiving with Guaranteed Convergence and Diversity in Multi-Objective Optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2002)*, pages 439–447. Morgan Kaufmann, 2002.

[9] Nasreddine Hallam, Peter Blanchfield, and Graham Kendall. Handling Diversity in Evolutionary Multiobjective Optimisation. In *Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC 2005)*, pages 2233–2240. IEEE Press, 2005.

[10] Da Kuang and Jinhua Zheng. Strategies Based on Polar Coordinates to Keep Diversity in Multi-Objective Genetic Algorithm. In *Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC 2005)*, pages 1276–1281. IEEE Press, 2005.

[11] Edwin D. de Jong, Richard A. Watson, and Jordan B. Pollack. Reducing Bloat and Promoting Diversity using Multi-Objective Methods. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2001)*, pages 11–18. Morgan Kaufmann, 2001.

[12] Xunxue Cui, Miao Li, and Tingjian Fang. Study of Population Diversity of Multiobjective Evolutionary Algorithm Based on Immune and Entropy Principles. In *Proceedings of the 2001 Congress of Evolutionary Computation (CEC 2001)*, pages 1316–1321. IEEE Press, 2001.

[13] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Elsevier, second edition, 2005.

[14] Hisao Ishibuchi, Tadahiko Murata, and I. B. Türkşen. Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems. *Fuzzy Sets and Systems*, 89:135–150, 1997.

[15] Hisao Ishibuchi and Yusuke Nojima. Accuracy-Complexity Tradeoff Analysis by Multiobjective Rule Selection. In *Proceedings of the ICDM 2005 Workshop on Computational Intelligence in Data Mining*, pages 39–48, 2005.

[16] Hisao Ishibuchi, Isao Kuwajima, and Yusuke Nojima. Multiobjective Association Rule Mining. In *Proceedings of the PPSN Workshop on Multiobjective Problem Solving from Nature*, 2006.

[17] John Koza. *Genetic programming: on the programming of computers by means of natural selection*. MIT Press, 1992.

[18] William B. Langdon and Riccardo Poli. *Foundations of Genetic Programming*. Springer, 1998.

[19] C. J Merz and P. M. Murphy. UCI repository of machine learning databases. Univ. California, Irvine, 1998.

[20] J. D. Knowles and D. W. Corne. Benchmark Problem Generators and Results for the Multiobjective Degree-Constrained Minimum Spanning Tree Problem. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2001)*, pages 424–431. Morgan Kaufmann, 2001.

[21] Siegfried Helbig and Darinka Pateva. On several concepts for ϵ efficiency. *OR Spektrum*, 16:179–186, 1994.