Cooperative Problem Decomposition in Pareto Competitive Classifier Models of Coevolution*

Andrew R. McIntyre and Malcolm I. Heywood†

October 20, 2008

Abstract

Pareto competitive models of coevolution have the potential to provide a number of distinct advantages over the canonical approach to training under the Genetic Programming (GP) classifier domain. Recent work has specifically focused on the reformulation of training as a two-population competition, that is learners versus training exemplars. Such a scheme affords, for example, the capacity to decouple the fitness evaluation overhead from the data set size through sub sampling while naturally encouraging 'teams' or composite solutions as opposed to solutions based on a single individual alone. One outstanding question with respect to the latter characteristic is with regards to the nature of the team (archive) behavior in terms of pattern coverage. That is to say, which models are used when, and what are the implications for solution modularity as it relates, for example, to the assignment of exemplars to solution participants. The current work investigates two Pareto competitive approaches to classification under GP, with one configured to employ an explicitly cooperative multi-objective cost function based and the other employing the classical (error-based) cost function. We empirically demonstrate a critical distinction between the two with regards to problem decomposition, with the capacity to provide a decomposition into unique behaviors being much more prevalent when co-operative mechanisms are explicitly supported.

1 Introduction

Pareto competitive coevolution has begun to make the transition from the genetic algorithm (GA) to genetic programming (GP) domain. The basic Pareto competitive coevolutionary model of relevance to the classification domain encourages evolution to take place between one or more populations such that one set of individuals represent a set of test cases (training exemplar subset) and the other a set of learners. A Pareto ranking model is established in which the test

*2008 EuroGP, LNCS 4971 – Copyright 2008 Springer-Verlag

†Faculty of Computer Science, Dalhousie University. Halifax. NS. Canada.

cases are rewarded for their ability to distinguish between the learners, and the learners are rewarded for their ability to remain non-dominated (with respect to the set of test conditions) [2], [10], [1]. The end result is that two Pareto fronts are built, one in which a set of non-dominated test points exist (corresponding to the minimal set of tests necessary to distinguish between the current set of learners) and the second details the corresponding set of non-dominated learners. The advantage that such a model provides is twofold. Fitness evaluation need no longer be conducted over the entire set of training exemplars, and a formal process has been established for identifying the 'best' training points and learners. Moreover, it is also possible to interpret convergence in terms of the behavior of the learner Pareto front, thus addressing the issue of domain independent stop criterion[1]. To date, however, the GA bias to this research has emphasized the issues of diversity maintenance, monotonic progress, and niching within the learner population.

In a GP setting the same issues exist, enabling some of the solutions to be carried over from a GA context. In particular, GA Pareto competitive models have been relatively successful in establishing a basis for coevolving two population models under the GP classification domain [5], [6], [12], [8]. In this work we will examine one of the basic problems unique to the GP classification domain under a Pareto competitive model, and illustrate how it can be dealt with by introducing a cooperative model of fitness assignment in addition to the Pareto competitive coevolutionary model.

Specifically, under the classification domain the Pareto competitive model results in a front of solutions for both learner and (test) point populations. This means that the solution will no longer be in the form of a single classifier, but in terms of a set of classifiers (those in the Pareto front). The basic issue at stake here is how to determine which model to use when. The problem does not appear in a GA setting for the most part because individuals take the form of a co-ordinate in a multi-dimensional space, thus application of an appropriate distance metric is sufficient to resolve which individual to apply when. Moreover, the GA domain may also introduce diversity mechanisms based on Euclidean distance metrics to encourage desirable 'coverage' properties in the Pareto front itself. Finally, although GA domains might utilize a Pareto method to evolve a front of solutions, they often assume that only one is chosen for deployment by the user.

Conversely, under a GP classification domain we demonstrate in this work that a lot depends on how the fitness function and associated wrapper operator are designed. In particular, the most straightforward approach might assume a binary (hit) based model on account of the ease with which the associated Pareto dominance test might be made. We show in this work that this results in a weak learner type of association between learners and exemplars, with significant overlap between the exemplars and responding learners. Conversely, by assuming a cooperative multi-objective model to fitness evaluation (in addition to the Pareto competitive model of evolution) we are able to establish an explicit

[1] Originally proposed under the GA paradigm [4], and reapplied under GP [8].

decomposition of exemplars to learners. As such, the post training assignment of learners to exemplars merely takes the form of utilizing the learner providing maximum class membership.

In the following, models for Pareto competitive coevolutionary and Pareto cooperative-competitive models of GP coevolution are introduced, Section 2. In doing so, we contrast the utility of local and global wrapper operators, and make the case for establishing reward mechanisms that explicitly encourage co-operative behaviors. Both algorithms provide solutions in the form of a 'front' of solutions. Section 3 investigates the nature of the interaction between individuals within the front under three multi-class classification problems. The effectiveness of the cooperative–coevolutionary model is now clear, with solutions typically taking the form of a clear behavioral decomposition of the problem domain. Conversely, the competitive–descriminator based model typically results in solutions in which a complex mixture of classifiers takes place, without any improvement over the classification accuracy of the former model.

2 Pareto competitive and Pareto cooperative–competitive Coevolution

In order to illustrate the aforementioned property of competitive coevolutionary GP classifiers, we compare the operation of two recent frameworks that utilize a Pareto based model of interaction between points (exemplars) and learners (classifiers): the Pareto-coevolutionary Genetic Programming Classifier [5], and Competitive Multi-objective Grammatical Evolution [8], [7]. For consistency both are implemented in terms of a canonical model of Grammatical Evolution (GE) [11], although both are entirely independent of the model of evolution on which they are based. Hereafter we refer to them as PGEC and CMGE respectively. In the following we establish the principle differences between the two models, and refer the reader to the original works for the detailed algorithmic descriptions.

The basic features of the CMGE classifier are summarized as follows relative to the pseudo code listing provided in Algorithm 1.

1. Identification of the subset of exemplars over which individual (learner) evaluation will take place (steps 2(a), 2(b));
2. Identification of the local membership function (wrapper operator) for each individual, relative to the associated *gpOut* distribution (steps 2(c)ii.A to D);
3. Fitness evaluation of individuals relative to the learning objectives under a multiobjective methodology (lines 2(c)ii.E to G);
4. Identification and archiving of the most valuable individual classifiers and exemplars (steps 2(d));
5. Class-wise assessment of early stopping criteria (steps 2(e)).

Conversely, the PGEC model is limited to: steps 2(a) and (b), define the content of the learner and point archives, after which the outcome vector for each individual is established, and then step 2(d) is performed, that is the Pareto assessment for establishing archive content.

2.1 Competitive Multi-objective Grammatical Evolution

The standard initialization process of step 1, Algorithm 1 stochastically creates GP population members (learners) and prepares the relevant data structures, including archives for both learners and exemplars (data points or simply, points). A while loop (step 2) encloses the main sections of the algorithm, ensuring that the training of GP is repeated until stopping conditions are met (as evaluated at the end of each iteration in step 2(e)). Steps 2(a) and 2(b) set up the training subset at each iteration ensuring a balanced view of data, thus enabling robustness against problems having unbalanced class distributions.

Algorithm 1 *High-level Pareto Coevolution.*

1. *Initialize Learner Population (LP);*
2. *WHILE ! (Stop criteria)*
   - (a) *Point Population (PP) := random balanced sample of training partition;*
   - (b) *Training Subset (TS) := PP concatenated with Point Archive contents (PA);*
   - (c) *FOR i := 1 to sizeof(LP)*
     - i. *Apply variation operators to Produce Children (C)*
     - ii. *FOR j := 1 to sizeof(C)*
       - A. *Establish phenotype of individual C[j];*
       - B. *Map TS to 1-d number line 'gpOut' of C[j];*
       - C. *Cluster gpOut of C[j];*
       - D. *Parameterize Gaussian Local Membership Function (LMF) of child C[j];*
       - E. *Evaluate C[j] with respect to:*
SSE, Overlap wrt. Learner Archive (LA), Parsimony.
       - F. *Rank C[j] with respect to LP and assign fitness;*
       - G. *Replacement (insert C[j] into LP);*
   - (d) *Archive PP, LP members based on outcomes (according to IPCA)*
     - i. *Points in PP enter PA if they provide a new distinction;*
     - ii. *Learners in LP enter LA if they are non-dominated wrt. LA;*
   - (e) *Evaluate Stop Criteria (method of Rank Histograms);*

3. Class-wise LA denote solution: Build appropriate weighting scheme;

Line 2(c) of Algorithm 1 begins the cooperative coevolution training loop which employs an Evolutionary Multi-objective Optimization (EMO) model loosely based on that of [4] to train GP. On each pass of the loop, selection and variation operators are applied to the GP population and children are produced (line 2(c)i). Next, individuals are decoded to their respective phenotype (line 2(c)ii.A) and the current selection of exemplars are mapped to the *gpOut* axis. We now require a mechanism to identify the local membership function (wrapper operator) neighborhood without resorting to inappropriate or arbitrary predefinitions of regions along the *gpOut* axis. In order to achieve this goal we assume that the neighborhoods of most relevance are those having the highest density, a requirement satisfied by a clustering algorithm (step 2(c)ii.C). The clustering algorithm returns the location of the mid point associated with the 'most dense' set of points and exemplars associated with this cluster, M. The process itself is independent of class label. We now have the properties for the local membership function (LMF) defined in terms of a Gaussian with mean, μ, and variance, σ (line 2(c)ii.D).

A fitness function is now applied to the subset of points of the neighborhood, M (line 2(c)ii.E, Algorithm 1). The objectives are designed to encourage: least ambiguity in cluster membership, non overlapping behavior of the exemplars mapped to different individuals, maximization of the number of in-class exemplars mapped to an individual, and simplicity of the GP mapping. Note that, in common with the findings of other EMO research, we establish a set of objectives that have a degree of implicit 'tension' between them. In doing so we are able to encourage mappings that reduce the likelihood of degenerate solutions. Moreover, in order to measure these objectives, the mapping is assigned a class, where this is assumed to correspond to the class of the point at the center of the LMF. In taking this route we avoid making any assumptions regarding which individuals are mapping which classes, and effectively let individuals compete for the right to map exemplars. The inner loop is completed by returning to the generic PCGA EMO algorithm of [4] in order to complete the Pareto ranking and replacement policy (lines 2(c)ii.F and G respectively). The significance of the Pareto ranking and ensuing fitness assignment is that selection operators proportionately favor individuals of higher fitness (lower ranking) over those having lower fitness (higher ranking). This tends to encourage the GP algorithm to more frequently sample material corresponding to individuals that lie closer to the Pareto front, with the goal of establishing improvements in the objectives. The PCGA model also provides the concept of rank histograms, which essentially summarizes the content of the population (in objective space) in terms of the Pareto ranks, so that content can be readily compared between training epochs. When calculated for each class, this provides the basis for the detection of class wise early stopping, line 2(e).

The inner loop defined by line 2(c) denotes the cooperative EMO model.

This portion of the main loop is performed in combination with the Pareto competitive model[2] for the purpose of adapting learner and test point archives as memories at line 2(d). That is to say, the competitive coevolution model's evaluation is conducted over the contents of the subset of training exemplars, *TS* (step 2(b)), dynamically identified by a competitive co-operative model for archiving the most discriminatory test points (step 2(d)i) and non dominated learners (step 2(d)ii), both from the perspective of a Pareto front. The competitive model thus plays a primarily archival role, acting as a memory for the cooperative model. The archive entry criteria are evaluated in terms of GP classification 'outcomes' which are directly related to the LMF definition and it's associated performance on the training set i.e., the outcome vector is takes on real values as opposed to the binary case of IPCA.

Deployment of the classifier (step 3) takes the form of copying the contents of the learner archives and assignment of weights to each on the basis of the training data. A winner-take-all policy with respect to LMF response determines the assignment of class labels among the team individuals.

2.2 Discussion

The critical differences between the PGEC and CMGE models are the use of EMO fitness evaluation in which cooperative behavior is explicitly sought in the mapping between exemplars and class membership, step 2(c), Algorithm 1. PGEC instead relies on the standard sigmoid based global wrapper operator for the purpose of mapping *gpOut* to class labels. This also implies that PGEC is a binary classifier, requiring multiple runs to evolve classifiers for each class, whereas CMGE provides classifiers for all classes from a single run. The interface to IPCA, line 2(d) remains unchanged i.e., the outcome vector. As such the principle mechanism for encouraging problem decomposition is the competitive model of Pareto dominance, as expressed between points and learners. Given that there is no explicitly cooperative mechanism for establishing population diversity, we maintain that this will generally result in PGEC producing classifiers with overlapping behaviors. That is to say, learners need only differ in one outcome in order to satisfy the Pareto dominance criterion and appear in the learner archive.

Unlike IPCA, both CMGE and PGEC make use of heuristics to enforce finite archive sizes for point and learner archives, *PA* and *LA*. In the case of the point archive, both PGEC and CMGE replace points once the archive limit is reached using an Euclidean distance metric in which the nearest current point is replaced [5]. In the case of the learner archive both PGEC and CMGE replace the individual currently within the archive with largest overall error (as estimated against the current training subset, *TS*).

A second difference resulting from the two models appears in the post training voting mechanism, step 3, Algorithm 1. PGEC relies on a majority policy,

[2] A variant of de Jong's IPCA algorithm [12], although any of this class of algorithm would be appropriate.

where this is designed to make use of the expected overlap in learner archive classifier behavior. Conversely, CMGE learners are expected to be unique, thus a winner takes all policy is assumed. In the case of PGEC, the merit of assuming a particular policy is expected to be more significant, as the degree of interaction between learners is likely to be data set specific. Conversely, under CMGE a winner takes all policy is a natural consequence of the explicitly cooperative model of evolution, reinforced by the action of the Gaussian local membership function.

3 Results

The PGEC and CMGE models are implemented using a common grammar and set of evolutionary parameters, Table 1. The grammar is capable of specifying zero argument (exemplar features), single argument (cosine, square root, natural log, exponential), and double argument (plus, minus, multiply, divide) operators. Variation operators take the form of one point crossover and mutation (PXO and PM respectively) and their corresponding context aware variants (PCXO and PCM respectively) [3]. Classifiers are implemented as a 'parallel model' in which a 'k' class problem implies that 'k' learner archives are evolved. A larger study, [7], conducted an evaluation over nine additional multi-class data sets taken from the UCI repository [9]. In this work, we focus on three interesting cases that characterized the behavior of all nine cases: Iris (IRIS), Boston Housing (BOST), and Contraceptive (CONT). Table 2 characterizes the basic properties of the data sets as deployed in this study. All three data sets are three class problems, and results employ ten fold cross validation with fifty runs per fold. The only pre-processing performed involved removing duplicate and incomplete exemplars from the original data set.

Table 1: GE Parameterization.

Parameter	Value	Parameter	Value
Max Generation	500	Learner Pop Size	50
Max Codon	256	Learner Archive Size	30
Max Codon Trans.	4,096	Point Pop Size	30
–	–	Point Archive Size	30
PXO (PCXO)	0.5 (0.9)	PM (PCM)	0.01 (0.9)

3.1 Evaluating Intra-class Voting Behavior

In order to investigate the effectiveness of the cooperative mechanism in CMGE versus the PGEC model, a metric for intra-class voting behavior is derived. In particular we wish to measure the degree to which learners comprising the Pareto front form constructive interactions, that is decompose the problem into

Table 2: Data Set Characterization.

Data set	num. Exemplars	num. Features	Class distribution (percent)
IRIS	147	3	33–33–33
BOST	506	12	33–33–33
CONT	1,425	8	43–22–35

non overlapping associations between learners and exemplars. This is interpreted in terms of the strength of the class membership operators, Gaussian and Sigmoid (local and global) for CMGE and PGEC respectively, relative to the median performance of the set of individuals constituting the 'team' of the same class. That is to say, given a winning classifier (the individual with maximum membership on an exemplar) we measure the difference in membership of the winner relative to the median membership of other classifiers associated with the same class. Differences would be distributed over the unit interval, and results in the metric characterizing performance in terms of three generic outcomes,

- Differences tending towards zero: indicates that there is little difference between membership of winning classifier and the median classifier performance. Needless to say, this could be associated with the majority of individuals labeling an in class exemplar correctly or incorrectly;

- Differences tending towards the mid point (0.5): indicates an individual with strong winning class membership, but with the majority of 'runner up' in-class individuals responding with a 'fifty percent' membership. Thus, the winning individual had a membership in the interval [0.5, 1], with the majority of the remaining intra-class classifiers responding with membership in the interval [0.25, 0.75]. Such behavior is considered undesirable as it is no longer possible to establish a clear difference between individuals labeling in class behaviors and those associated with out of class behavior.

- Differences tending towards unity: implies that the class winner responds with a class membership tending towards unity, whereas the majority of the other individuals respond with low class membership. Naturally, this implies a strong uniqueness in the classifier to exemplar decomposition.

The following comparison will first establish the baseline performance of each model in terms of detection rate, false positive rate and the number of participating models. This establishes that nothing is lost by assuming a model that enforces cooperative problem decomposition. The second evaluation characterizes the nature of the intra-class decomposition.

3.1.1 Complexity and Classification performance

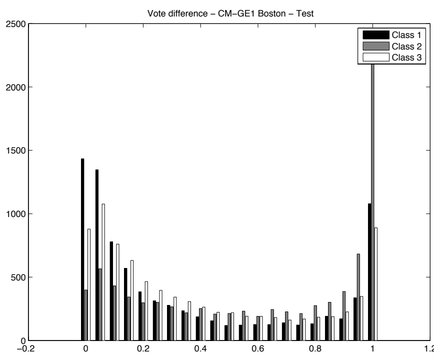
Table 3 establishes the number of classifiers participating and ensuing detection / false positive rates over the test partition in terms of the class-wise median. Both models clearly utilize multiple classifiers per class. It is also clear that the CMGE model provides a more reliable classifier, with the significantly higher per class detection rates more than out weighing any increases in false positive rate. Moreover, the PGEC model was uncompetitive on both the 'balanced' Iris data set as well as the larger unbalanced data sets. Also of significance is that this is typically achieved while CMGE utilizes the entire contents of the learner archive. Thus only on class one of IRIS did CMGE employ a much lower count of classifiers than that in the remaining cases (all of which tend to the archive limit of thirty).

Table 3: Median Test Set Performance.

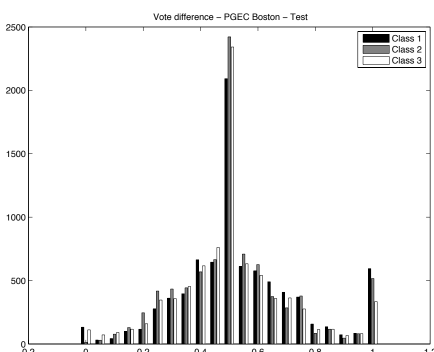
Classifiers per Class			
Data set	IRIS	BOST	CONT
Class	1–2–3	1–2–3	1–2–3
CMGE	3–30–30	30–30–30	29–30–30
PGEC	1–5–4	10–14.5–10	14–15–21
Detection Rate			
CMGE	100–100–100	87.5–35.3–82.4	73.8–31.2–24
PGEC	0–100–40	17.6–52.9–6.2	18–11.1–32.7
False Positive Rate			
CMGE	0–0–0	22.9–9.4–15.2	53.7–14.5–16.3
PGEC	0–50–0	6.1–44.1–2.9	11.1–9.4–28

3.1.2 Intra-class Coverage

The above section established that the explicitly cooperative model of GMGE is able to build on the competitive coevolutionary paradigm shared by both models. In this section we characterize the form of the decomposition using the aforementioned coverage metric. Specifically, we build histograms of the CMGE and PGEC intra-class coverage over the test partition (no significant differences appearing between training and test histograms). Figures 1, 2 and 3 summarizing the basic behaviors on the Boston Housing, Iris and Contraceptive data sets respectively. In the case of the Boston Housing data set, CMGE results in a bimodal distribution in which there is either a considerable differentiation between winning classifier and the remainder of the same class classifiers (the right peek at unity), or the majority of the classifiers have a similar class membership behavior (the left peek at zero). Moreover, class 2 appears to result in classifiers demonstrating most behavioral uniqueness, whereas classes 1 and 3 result in behavior distributed equally at the two peeks. PGEC on the other hand demonstrates a strong preference for multiple individuals responding at an intermediate level of class membership (i.e., the peak at 0.5). As such it is not possible to establish that the majority of in-class individuals respond with a strong in-class preference or a strong differentiation between in and out of class behavior, Figure 1(b).



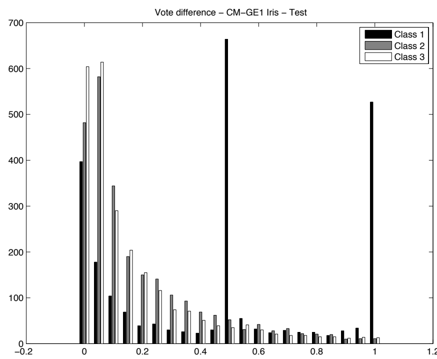
(a) CMGE (Test)



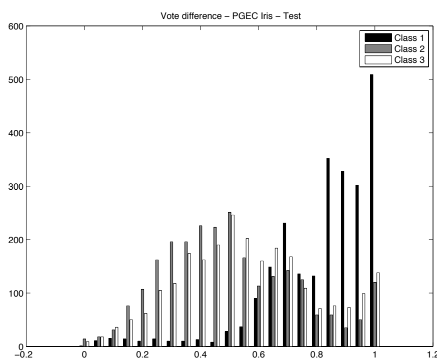
(b) PGEC (Test)

Figure 1: Team behavior: Boston data set.

Under the Iris data set, CMGE demonstrates two distinct distributions. In the case of the linearly separable class (one) a distribution similar to that for the Boston Housing data set is returned i.e., strong similarity or strong differentiation. The single peak at the mid point is, in this case, produced as an artifact of an equal number of in-class classifiers returning both maximum (1) and minimum (0) differences. On the two non-linearly separable classes the intermixing of the class boundary results in a bias towards a higher similarity in behavior between in-class classifier membership. Given the strong classification performance of the model as a whole, Table 3, this implies that multiple classifiers are involved in supplying a correct label. The PGEC model also produces two distinct distributions under the Iris data set. However, although the linearly separable class does result in a desirable peak at unity, there are also secondary



(a) CMGE (Test)

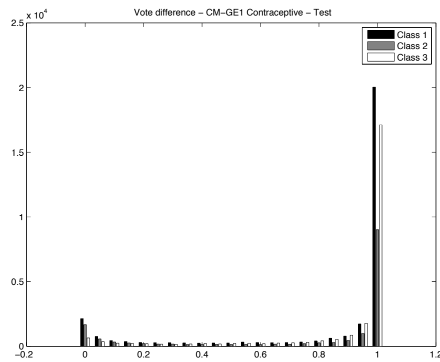


(b) PGEC (Test)

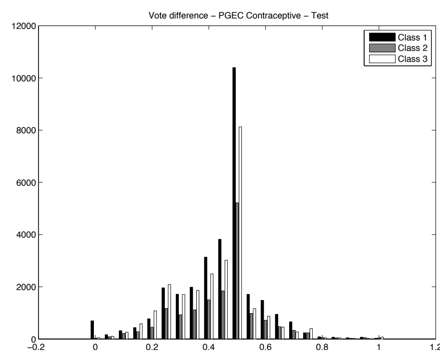
Figure 2: Team behavior: Iris data set.

distributions in the interval 0.8 to 1 and around 0.7. The two non linearly separable classes (two and three) appear to rely on a complex intermixing of class votes with no clear differentiation between winner and other in-class classifiers. In short PGEC appears to find it difficult to encourage individual classifiers to take a definite policy regarding the subset of exemplars on which they will respond. This is reinforced by the poor overall classification results, Table 3.

Finally, the Contraceptive data set, Figure 3, is representative of the most difficult problem domain considered and in the case of CMGE results in a clear emphasis towards classifiers that respond to very different subsets of exemplars (high count of large differences). Conversely the PGEC model is generally unable to establish a clear differentiation in class membership values, with most of the distribution again sitting in the mid region of the histogram.



(a) CMGE (Test)



(b) PGEC (Test)

Figure 3: Team behavior: Contraceptive data set.

4 Conclusion

Coevolutionary models of classification using GP are beginning to appear in which the test point competitive coevolutionary Pareto models developed under a GA setting by Watson, Ficici, de Jong, and Pollock frequently serve as the starting point. The model provides many useful properties, not least that the inner loop of GP is now decoupled from the size of the original training data set. In this work we emphasize that there are also several GP and classification domain specific problems that were not especially relevant in the original GA domain. In particular, just because the solution (typically) takes the form of a set of classifiers (the contents of the learner archive or Pareto front), this is not sufficient to encourage distinct behaviors between the learners themselves. Related to this property is the need to introduce a mechanism for establishing post-training class labels from the ensuing classifiers. In this work we revisit the original Pareto competitive classification model of Lemczyk [5] in which a global membership function and majority voting are used to establish class labels, and compare with a local membership function in which members of the

Pareto front are required to explicitly cooperate under a local wrapper operator [8],[7]. In order to investigate this property a 'coverage' metric is introduced to establish the degree of differentiation between 'winning' class behavior and the median performance of the remaining classifiers. The resulting evaluation clearly demonstrates that the competitive-coevolutionary CMGE model is able to associate specific exemplar subsets with specific classifiers, whereas PGEC is unable to provide a clear separation between classifier behaviors. Moreover, this is achieved without compromising the performance of the ensuing classifiers.

Future work will revisit the representation used within the context of the point population. In particular the GP domain typically employs a GA population for the points in which exemplars are directly represented by indices. The basic problem with this is that although directly supporting the competitive coevolutionary model, there is no natural mechanism for establishing context on which a crossover operator could operate. Thus, to date the most effective model appears to simply re-establish the point population at each generation using uniform selection with a class balance enforcing heuristic, whereas finding a representation that permits variation operator context might provide a more elegant solution.

Acknowledgements

The authors gratefullly acknowledge the support of PRECARN, NSERC Discovery, MITACS, and CFI New Opportunities programs; and industrial funding through the Telecom Applications Research Alliance (Canada), and SwissCom Innovations AG (Switzerland).

References

[1] E. D. de Jong and J. B. Pollack. Ideal evaluation from coevolution. *Evolutionary Computation*, 12(2):159–192, 2004.

[2] S. G. Ficici and J. B. Pollack. Pareto optimality in coevolutionary learning. In *European Conference on Artificial Life*, pages 286–297, 2001.

[3] R. Harper and A. Blair. A structure preserving crossover in Grammatical Evolution. In *IEEE Congress on Evolutionary Computation*, volume 3, pages 2537–2544, 2005.

[4] R. Kumar and P. Rockett. Improved sampling of the pareto-front in multiobjective genetic optimizations by steady-state evolution. *Evolutionary Computation*, 10(3):283–314, 2002.

[5] M. Lemczyk and M. I. Heywood. Training binary GP classifiers efficiently: a pareto-coevolutionary approach. In *Proceedings of the European Conference on Genetic Programming (EuroGP)*, volume 4445 of *LNCS*, pages 229–240, 2007.

[6] P. Lichodzijewski and M. I. Heywood. Pareto-coevolutionary Genetic Programming for problem decomposition in multi-class classification. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, volume 1, pages 464–471, 2007.

[7] A. R. McIntyre. *Novelty Detection + Coevolution = Automatic Problem Decomposition: A Framework for Scalable Genetic Programming Classifiers*. PhD thesis, Faculty of Computer Science, Dalhousie University, 2007. [http://www.cs.dal.ca/∼mheywood/Thesis].

[8] A. R. McIntyre and M. I. Heywood. Multi-objective competitive coevolution for efficient GP classifier problem decomposition. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1930–1937, 2007.

[9] D. J. Newman, S. Hettich, C. L. Blake, and C. J. Merz. UCI repository of machine learning databases. [http://www.ics.uci.edu/∼mlearn/mlrepository.html], 1998.

[10] J. Noble and R. Watson. Pareto coevolution: Using performance against coevolved opponents in a game as dimensions for pareto selection. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 493–500, 2001.

[11] Michael O'Neill and Conor Ryan. *Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language*. Springer, May 2003.

[12] T.-S. Yo and E. D. de Jong. A comparison of evaluation methods in coevolution. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, volume 1 of *SIGEVO*, pages 479–486, 2007.