GP Classifier Problem Decomposition Using First-Price and Second-Price Auctions*

Peter Lichodzijewski and Malcolm I. Heywood†

April 2007

Abstract

This work details an auction-based model for problem decomposition in Genetic Programming classification. The approach builds on the population-based methodology of Genetic Programming to evolve individuals that bid high for patterns that they can correctly classify. The model returns a set of individuals that decompose the problem by way of this bidding process and is directly applicable to multi-class domains. An investigation of two auction types emphasizes the effect of auction design on the properties of the resulting solution. The work demonstrates that auctions are an effective mechanism for problem decomposition in classification problems and that Genetic Programming is an effective means of evolving the underlying bidding behaviour.

1 Introduction

Genetic Programming (GP) is a population-based search algorithm that classically produces a single 'super' individual by way of a solution [5]. This is a natural effect of the survival of the fittest mechanism implicit in GP and is supported by various theoretical models [7]. The success of a single individual, however, may be limited in scenarios where progress cannot be made without effective problem decomposition. Attempts have been made to encourage GP to provide multiple solutions where these have typically taken the form of diversity maintenance schemes such as niching [10] and coevolution [11]. In this work we take a different approach motivated by the use of market mechanisms in machine learning, and in particular, the Hayek framework of Baum [1] [2]. However, it is apparent that the sheer number of problem-specific parameters endemic to the Hayek model results in a system that is very difficult to replicate [6].

The motivation of the current work is therefore to revisit the market-based approach for problem decomposition with the objective of keeping the model

*Published in EuroGP'07. LNCS 4445. Springer-Verlag.

†Faculty of Computer Science, Dalhousie University, Halifax, Nova Scotia, Canada.

as simple as possible. We begin by considering the problem domain to be discrete, in this case, binary and multi-class classification problems. This implies that actions are also discrete so that individuals may concentrate on identifying an appropriate bidding strategy. Such a strategy need only be profitable, thus individuals are free to identify the subset of training exemplars on which they will concentrate their resources. Key to encouraging the identification of bidding strategies that facilitate problem decomposition is the definition of an appropriate auction mechanism. To this end we concentrate on the design of the auction model where the auction is central to establishing the credit assignment process; if the auction is effective in instigating the relevant reward mechanism then the learning problem as a whole should also be more straightforward.

Auctions have successfully been applied to other problem domains such as the coordination of teams of robots [4]. This work demonstrates that a market-based framework using auctions can also be applied to the machine learning problem of classification. In particular, it is shown that GP is an effective means of producing the underlying bidding behaviour with minimal *a priori* knowledge.

This paper is organized as follows. The following section discusses related systems and motivates the need for a simplification to the Hayek model. Section 3 describes the proposed approach including the two types of auctions that are investigated. Section 4 summarizes model performance on four real-world datasets (two binary and two multi-class), and concluding remarks are made in Section 5.

2 Related Work

The proposed approach is motivated by the market-based Hayek model [1][2] originally applied to a blocks world problem (i.e., reinforcement learning). Hayek was found to be exceptional because it discovered generic solutions capable of producing long chains of actions. In contrast, GP augmented with hand-crafted features was able to solve problems limited in size to at most five blocks [5].

Hayek employs auctions in which individuals bid for the right to act on the environment. Because individuals must pay out their bid, good behaviours will tend to extract sufficient reward and therefore earn wealth while poor behaviours will tend to lose wealth; wealth is therefore treated as a fitness measure and used to guide the search. The key differences between Hayek and the proposed approach are as follows:

1. In Hayek, an individual always bids the same amount (limited only by the individual's wealth) yet its choice of action is a function of the input. Individuals in the proposed approach may bid different amounts, are always associated with a single action, and may posses negative wealth.

2. The Hayek population is of a variable size. Individuals are created when existing individuals accumulate enough wealth and removed when their wealth falls below a threshold. In the proposed approach, the population

size is fixed and the poorest individual is always removed so that survival of the fittest is reinforced and the overhead of maintaining large populations is reduced[1].

3. In Hayek, when a child is created it receives a fraction of its parent's wealth. In return, the offspring must periodically pay to its parent a constant amount plus a fraction of its profit. In the proposed approach, there are no such transfers of wealth thus avoiding the associated problems of parameterizing the payoff feedback loop.

4. The Hayek framework taxes each individual in proportion to its size and the total computational overhead incurred by the system. The proposed approach does not use taxation. However, a fixed-size population with a steady-state selection policy ensures that the worst performing individuals do not linger in the population.

The Hayek framework was found to be complex and sensitive to a large number of paremeter setings making it difficult to reproduce previous results [6]. The design of the proposed approach results in a simpler model enabling us to concentrate on establishing the contribution of specific system components (e.g., auctions). In addition, as a starting step and to make the analysis more manageable, here the approach is applied to classification problems only.

The approach presented in this work is also related to Learning Classifier Systems [8]. Whereas classifier systems evolve populations of condition-action-strength rules, in the proposed approach the condition and strength component has been replaced by the bid procedure. This has several implications, for example, it makes the behaviours of each individual deterministic as it does not depend on any dynamic parameters such as strength. Furthermore, in the proposed approach the action set always consists of a single individual making allocation of credit more straightforward. Finally, compared to some popular classifier system formulations, in the proposed approach an individual's fitness is not solely a function of its bid accuracy [14]. Instead, individuals may survive in the population so long as their ratio of the gains made on profitable auctions to the losses sustained during unprofitable auctions is sufficiently high.

3 Methodology

A population of individuals each defining a *bid* and an *action* is evolved. The view is taken that only the bidding behaviour needs be represented as a program. The corresponding action is defined by a scalar selected *a priori* over the range of class labels, i.e., the set of integers $\{0, ..., n-1\}$ in an n-class classification problem. Given that a market model will be utilized as the methodology for problem decomposition, wealth should reflect the success of an individual's bidding behaviour. Thus, when new individuals are initialized in the population, they assume the same wealth as the poorest individual in the population. Such

[1]Populations with tens of thousands of individuals were encountered in Hayek.

a view is taken in order to avoid injecting disproportionally high volumes of wealth into the market. Moreover, the initial population possesses zero wealth. This does not preclude individuals from bidding as negative wealth is possible. Thus, wealth is used as a relative measure of performance as opposed to having any monetary properties.

The following sections describe the generic auction model, two specific auction types, and the form of GP employed.

3.1 Generic Auction Model

The population is of a fixed size. During initialization, Step 1 of Algorithm 1, individuals generated with uniform probability are added to the population until the population limit is reached. Initial wealth values are set to zero. Following initialization, the training algorithm proceeds in a series of epochs, Step 2. In the first stage of an epoch, an auction is held for each pattern in the training dataset, Step 2(a)*i*. During the auction, agents compete for the ownership of the pattern and wealths are adjusted to reflect the outcome of this competition. Reproduction takes place once the auctions have completed. At this time, the individual with the least amount of wealth is replaced. A single parent is selected with uniform probability from the population, Step 2(c), and a child is created from this parent through the application of mutation operators. The wealth of the poorest agent in the remaining population is determined, with the wealth of the child taking this value, Step 2(e). The child is then added to the population. If the new individual is profitable, its ranking with respect to wealth should 'bubble up' relative to the performance of the population in successive epochs.

The goal of training is to produce a population where each individual wins a subset of the training exemplars for which its action is suitable (e.g., in classification, suitable means that the label of the exemplar matches the action of the individual). Following training, the aggregate bidding behaviour of the population will determine how the system acts on an unseen exemplar.

3.2 Auction Types

Two types of auctions, Step 2(a)*i* of Algorithm 1, were investigated. The first represents a vanilla first-price auction in which the winning agent pays the difference between its bid and the reward resulting from its action. The second auction model explicitly encourages individuals representing different actions to minimize their bid values when their actions do not match that of the class label.

First-Price (FP). An exemplar is presented and each individual submits an associated bid. The individual with the highest bid is selected as the auction winner and must pay its bid to the environment (i.e., a payment is made but never collected). The winner's action is then compared to the exemplar label. When the the two match, the winner receives a reward of 1, otherwise, the winner receives a reward of 0.

Algorithm 1 Generic auction-based evolutionary training algorithm.

1. while less than the population size limit do
   (a) seed = newRandomIndividual();
   (b) seed.wealth = 0;
   (c) population.insert(seed);
2. for each epoch do
   (a) for each training exemplar 'p' do
      i. auction(population, p);
   (b) population.delete(findPoorestIndividual(population));
   (c) parent = selectRandomIndividual(population);
   (d) child = newChild(parent);
   (e) child.wealth = findPoorestIndividual(population).wealth;
   (f) population.insert(child);

Second-Price (SP). Individuals again submit a bid for each exemplar and the highest bidder is selected as the auction winner. The winner's action is compared to that of the exemplar, and the FP scheme followed if the winner's action does *not* match that of the label. If the winner's action matches the label, then the highest bidder is identified from individuals representing alternative actions. The winner does not pay out its bid (to the environment) but rather the bid of this runner-up. In addition, the runner-up pays its bid (to the environment). Since the winner's action matched the class of the pattern, the winner receives a reward of 1.

In both auctions, an individual can profit only by winning auctions for patterns whose class is the same as its action without overbidding. As such, the goal is to evolve *winning* bids that reflect the reward associated with individuals' actions on each pattern. In addition, the SP auction is designed to increase the winner's profit (especially during the later stages of training when high bids are expected) and produce more robust behaviours by driving down bids of non-winning classes.

3.3 Linear Genetic Programming

Bid procedures were evolved using a linear GP representation [3]. The Sigmoid function $f(y) = (1-e^{-y})^{-1}$ was used to obtain a bid over the unit interval given a raw (real-valued) GP output y. Since the possible reward values were restricted to the set $\{0, 1\}$, an individual could overbid only for instances of the wrong class. A null-initialized set of registers was made available for storing intermediate results and the final output was extracted from a predefined register. Denoting registers by R, inputs by I, and operations by op, the instructions themselves could be of the form $R_x \leftarrow op\, R_x\, R_y$ or $R_x \leftarrow op\, R_x\, I_y$. Both unary and binary operators were allowed, and in the cases where op was unary it was applied to the y operand.

Five stochastic search operators were used to mutate a parent to generate a child of which the first four affected the bid program: (1) bid delete removed an instruction at an arbitrary position, (2) bid add inserted an arbitrary instruction at a randomly chosen position, (3) bid mutate flipped a random bit of an instruction at an arbitrarily chosen position, (4) bid swap exchanged the positions of two arbitrarily chosen instructions, and (5) action mutate changed the action associated with an individual to a randomly chosen value. Each of these operators was applied with a specified probability and the application of the operators was not exclusive.

During initialization, the individual bid program sizes were selected from a predefined range with uniform probability (i.e., a fixed length representation). The bid delete and add operators were therefore included to allow change to the complexity of a program. The bid swap operator was added for situations when a program had the right instructions but in the wrong order. Finally, the action mutate operator was employed in case individuals appeared that exhibited the right bidding behaviour but for the wrong class. In addition, this operator proved useful for ensuring an action always had a chance of appearing in the population (e.g., in situations where all individuals advocating an action were extinct).

4 Evaluation and Results

4.1 Datasets and Parameterization

Four real-world datasets from the UCI Machine Learning Repository [12] were used to evaluate the approach, Table 1. The test partitions were generated by randomly selecting instances from the entire dataset to approximately preserve class distributions. With the exception of the BCW dataset where a 50/50 split was used to reduce training times, in all cases one-quarter of all the patterns were held out for testing. BCW refers to the Wisconsin Breast Cancer dataset with patterns containing missing attributes removed. BUPA is the liver disorders databases. Classes 1, 2, and 3 in the Iris dataset correspond to flowers iris-setosa, iris-versicolor, and iris-virginica respectively. Finally, Housing refers to a three-class version of the Boston Housing dataset [9]. The BUPA and Housing datasets are considered to be representative of the more difficult classification problems. It should be noted that individuals in the population were initialized with a single action selected a priori from the set of possible class labels, Table 1, with uniform probability.

The parameters used in all of the experiments are given in Table 2. Thirty different initializations were performed for each configuration to account for the

Table 1: Datasets used in evaluating the proposed approach. Distribution refers to the number of patterns of each class and is given in the same order as the class labels in the 'Labels' column. All labels are shown as they appear in the original datasets.

Dataset	Features	Labels	Train Size	Train Distribution	Test Size	Test Distribution
BCW	9	(2, 4)	342	(222, 120)	341	(222, 119)
BUPA	6	(1, 2)	259	(108, 151)	86	(37, 49)
Iris	4	(1, 2, 3)	113	(37, 37, 39)	37	(13, 13, 11)
Housing	13	(1, 2, 3)	380	(123, 140, 117)	126	(44, 33, 49)

Table 2: Parameter values used in the experiments.

Parameter	Value
Minimum program size	1
Maximum program size	256
Bid delete/add/mutate/swap probability	0.5
Action mutate probability	0.5
Number of registers	4
Function set	$\{+, \times, -, \div, \cos, \sin, \exp, \log\}$
Population size	100
Epochs	100 000
Number of initializations	30

dependence of the algorithm on the starting conditions.

After training, individuals that won zero auctions on the training data were marked as inactive and not used on the test partition. Results were then compiled in terms of the number of *active* individuals, classification accuracy, and bidding behaviour. The results shown are averaged over the thirty initializations performed for each pairing of auction type and dataset.

4.2 Results

Figure 1 summarizes the number of active individuals. Compared to the SP auction, the FP auction uses more individuals one the easier datasets and fewer on the difficult ones. Conversely, the SP auction allocates more resources to the more difficult problems and less to the easier problems. Both approaches assign significantly more resources to the more difficult BUPA and Housing datasets.

As seen in the summary of the accuracy results on the test data, Figure 2, there is no clear preference for either auction type. In addition, neither approach

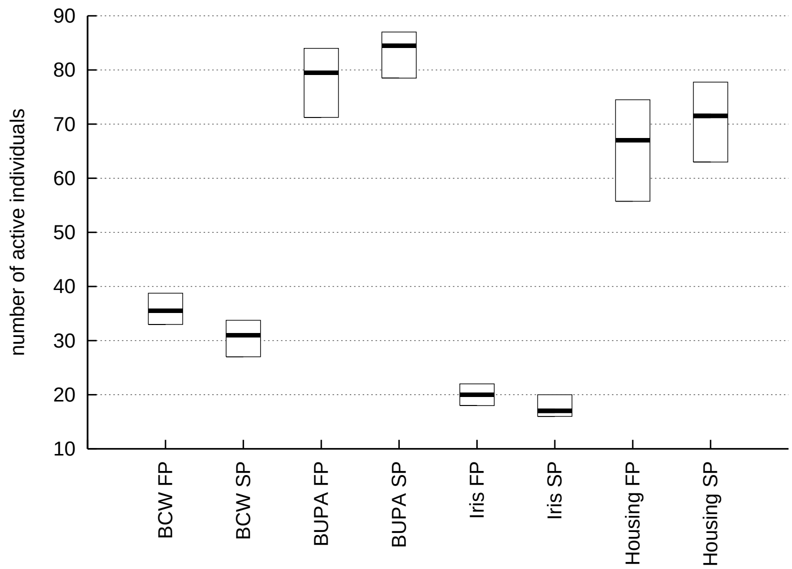


Figure 1: Active individual counts. The box boundaries denote the first and third quartiles, the thick horizontal line the median.

is seen to be superior with respect to consistency (i.e., spread of test accuracies). Even though the overall accuracy of the FP and SP approaches appears to be similar, the two differ with respect to their per class accuracies, Figure 3. In the case of the BCW and Iris datasets, the task appears to be straightforward as all classes are represented with a high degree of precision under both auction schemes. On the more difficult BUPA and Housing datasets, however, the SP approach yields more balanced results (i.e., the SP approach is more effective at profiling *all* classes). This suggest that the SP approach will be less biased in situations where the class distributions are unbalanced.

Figure 4 shows aggregate bidding behaviour on the training data. For each case, the mean bid value is calculated by considering the bids of all individuals of a given action on all patterns of a given class. The mean maximum bid is calculated in a similar fashion except that for each pattern only the winning individual is considered (from all individuals advocating a given action). For individuals of action x bidding on instances of class y, the desired behaviour corresponds to bidding high only if $x = y$ and low otherwise. This corresponds to individuals bidding high for patterns that match their actions and low otherwise. Desirable behaviour does not necessarily imply that the mean bid is high whenever $x = y$. Given a class x, certain individuals of action x may bid high only for some of the patterns of class x thus identifying a subclass within a single class (as labeled in the dataset).

Figure 4 shows that the SP approach yields better bidding behaviour. Using the FP approach, the maximum bid always tends to be high; individuals bid

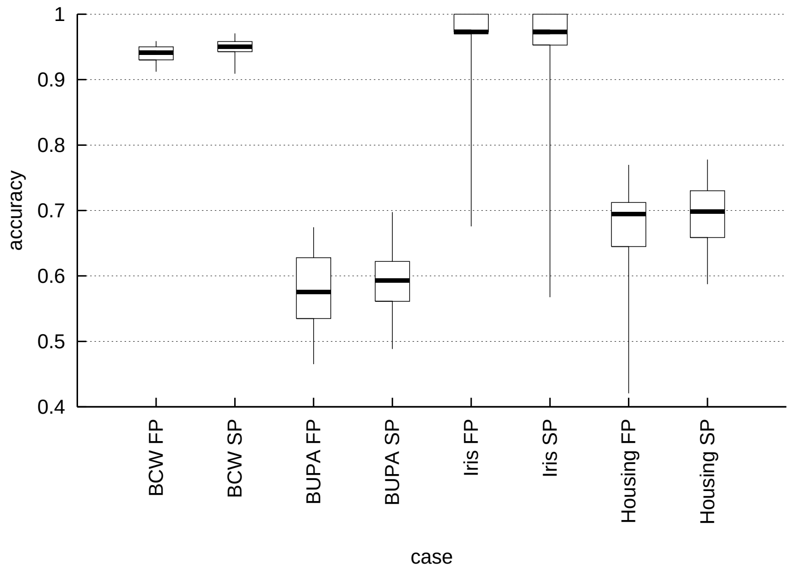
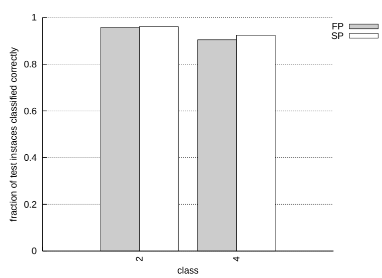
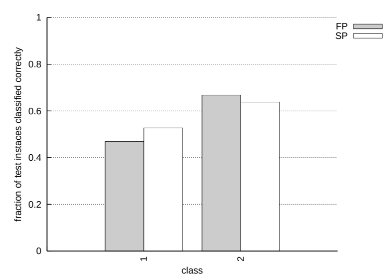


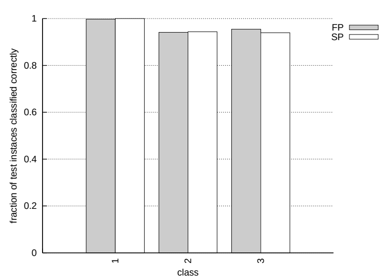
Figure 2: Accuracy results on the test data. The thick horizontal line denotes the median, the box boundaries the first and third quartiles, and the line endpoints the minimum and maximum.



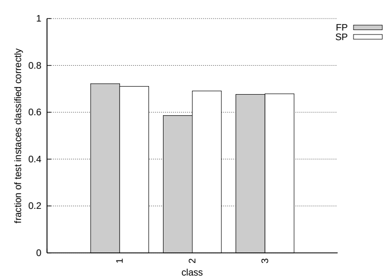
(a) BCW.



(b) BUPA.

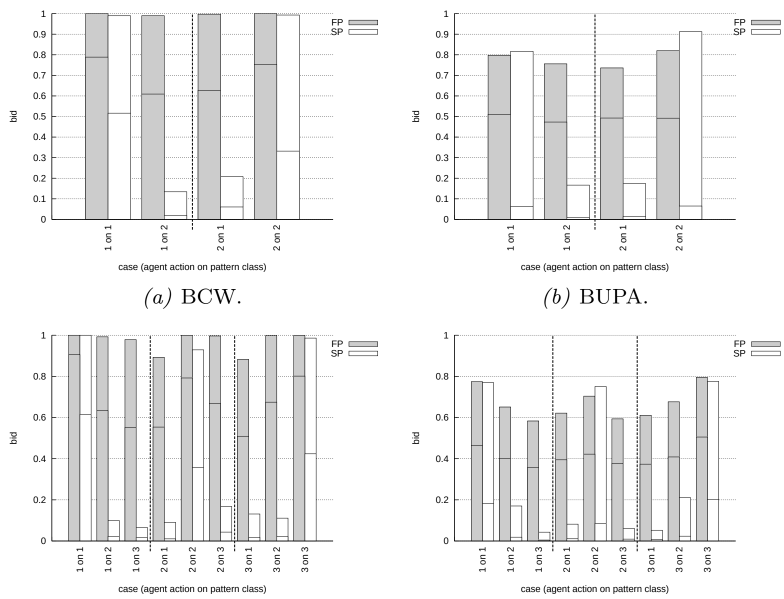


(c) Iris.



(d) Housing.

Figure 3: Per class accuracies on the test data. Each bar shows the accuracy of the FP or SP approach on the class denoted on the x-axis.



(a) BCW. *(b)* BUPA. *(c)* Iris. *(d)* Housing.

Figure 4: Bidding behaviour on the training data. Each bar shows the maximum mean bid (height of the bar) and the mean bid (segment crossing the area of the bar) for the FP or the SP approach. The bars are grouped by bidding scenarios.

high even for patterns associated with the wrong class. This is best illustrated on the BCW dataset where the mean maximum bid using the FP approach is always virtually unity regardless of the true class of the pattern. Using the SP approach, if a pattern does not match the individual's action then that individual's bid tends to be significantly lower. This behaviour appears to be a direct result of the penalty applied to the runner-up in the SP wealth adjustment process and suggests more robust decision boundaries.

4.3 Comparison with C5.0

To summarize and to put the difficulty of the learning task into context, Table 3 shows a comparison of the results obtained using the FP and SP auction types and C5.0 [13]. C5.0 is an established data mining algorithm that can be used to build classifiers in the form of decision trees. In setting up C5.0, all BCW attributes were defined as ordered discrete, the 'CHAS' Housing attribute was defined as discrete, all other attributes were defined as continuous, and default learning parameters were used. The table shows that the proposed approach typically outperforms C5.0.

Table 3: Test accuracies (in percent) of the FP and SP schemes compared to C5.0.

	BCW		BUPA		Iris		Housing	
	Best	Mean	Best	Mean	Best	Mean	Best	Mean
FP	96	94	67	58	100	96	77	67
SP	97	95	70	59	100	96	78	70
C5.0	94	-	67	-	97	-	75	-

5 Conclusion

A market-based model for decomposing classification problems between multiple GP individuals was presented. The central mechanism in this model is the auction where an individual can profit only by correctly classifying a problem instance. The proposed approach requires a single population to be evolved and can be directly applied to problems with more than two classes.

Two auction types were examined and the SP formulation found to be superior for several reasons. First, it allocated more resources to the more difficult problems and fewer resources to the easier problems. Second, it yielded more balanced per class classification accuracies. Finally, it produced a wider margin between correct and incorrect bids suggesting more robust decision boundaries.

This work demonstrates that auctions are an effective means of partitioning the instance space in classification problems and that they can be tailored to achieve desired system behaviour. By using bids to associate individuals with patterns, problem decomposition can be achieved. In addition, GP was shown to be able to successfully evolve the bidding behaviour underlying every auction. In this regard, wealth was shown to be an effective measure of individual fitness.

As demonstrated by Hayek, the idea of using an auction to select an appropriate action given an input is also applicable to reinforcement learning scenarios. One obstacle in this problem domain is the high number of possible test instances. Future work should therefore focus on incorporating active learning to select a manageable and informative subset of test cases during training.

Acknowledgments

The authors would like to thank the Killam Trusts, the Natural Sciences and Engineering Research Council of Canada, and the Canada Foundation for Innovation for their financial support.

References

[1] E. B. Baum and I. Durdanovic. An evolutionary post production system. *Advances in Learning Classifier Systems*, pages 3–21, 2000.

[2] E. B. Baum and I. Durdanovic. Toward code evolution by artificial economies. *Evolution as Computation*, pages 314–332, 2002.

[3] M. Brameier and W. Banzhaf. *Linear Genetic Programming*. Springer, 2007.

[4] S. Koenig, C. Tovey, M. Lagoudakis, V. Markakis, D. Kempe, P. Keskinocak, A. Kleywegt, A. Meyerson, and S. Jain. The power of sequential single-item auctions for agent coordination. In *Proceedings of the National Conference on Artificial Intelligence*, 2006.

[5] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[6] I. Kwee, M. Hutter, and J. Schmidhuber. Market-based reinforcement learning in partially observable worlds. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 865–873, 2001.

[7] W. B. Langdon and R. Poli. *Foundations of Genetic Programming*. Springer-Verlag, 2002.

[8] P. L. Lanzi and R. L. Riolo. Recent trends in learning classifier systems research. *Advances in Evolutionary Computing: Theory and Applications*, pages 955–988, 2003.

[9] T. Lim, W. Loh, and Y. Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine Learning*, 40:203–228, 2000.

[10] A. R. McIntyre and M. I. Heywood. On multi-class classification by way of niching. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 581–592, 2004.

[11] A. R. McIntyre and M. I. Heywood. MOGE: GP classification problem decomposition using multi-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 863–870, 2006.

[12] 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.

[13] Rulequest Research. Data mining tools See5 and C5.0. [http://www.rulequest.com/see5-info.html], 2005.

[14] S. Wilson. Classifier fitness based on accuracy. *Evolutionary Computation*, 3(2):149–175, 1995.