

Promoting Semantic Diversity in Multi-objective Genetic Programming

Edgar Galván

Department of Computer Science,
Maynooth University, Ireland
edgar.galvan@mu.ie

Marc Schoenauer

TAU, INRIA and LRI, CNRS &
U. Paris-Sud, Université Paris-Saclay, France
marc.schoenauer@inria.fr

ABSTRACT

The study of semantics in Genetic Programming (GP) has increased dramatically over the last years due to the fact that researchers tend to report a performance increase in GP when semantic diversity is promoted. However, the adoption of semantics in Evolutionary Multi-objective Optimisation (EMO), at large, and in Multi-objective GP (MOGP), in particular, has been very limited and this paper intends to fill this challenging research area. We propose a mechanism wherein a semantic-based distance is used instead of the widely known crowding distance and is also used as an objective to be optimised. To this end, we use two well-known EMO algorithms: NSGA-II and SPEA2. Results on highly unbalanced binary classification tasks indicate that the proposed approach produces more and better results than the rest of the three other approaches used in this work, including the canonical aforementioned EMO algorithms.

CCS CONCEPTS

• **Computing methodologies** → Genetic programming;

KEYWORDS

Multi-objective Genetic Programming, Semantics

ACM Reference Format:

Edgar Galván and Marc Schoenauer. 2019. Promoting Semantic Diversity in Multi-objective Genetic Programming. In *Proceedings of the Genetic and Evolutionary Computation Conference 2019 (GECCO '19)*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3321707.3321854>

1 INTRODUCTION

Genetic Programming (GP) [12] has been successfully applied in a variety of challenging problems. Despite the popularity of GP and its proven effectiveness in the face of challenging problems' features, it is also well-known that vanilla GP has some serious limitations and researchers have been interested on various GP aspects to make it more reliable.

One popular element studied by GP researchers is semantics which has been constantly reported to improve GP performance

by e.g., encouraging semantic diversity via crossover [19], selection [9] and geometric operators [14].

Semantics is a broad concept that has been studied in different fields, making it hard to give a precise definition of the concept. Moreover, the way semantics has been adopted in GP varies significantly. This work uses a popular version of semantics GP, as originally proposed in [13], in which the *semantics* of a (sub)tree is defined as the vector of output values computed by this (sub)tree for each set of input values in turn (also known as each fitness case). Several semantic-based approaches have been proposed for GP which take semantics into account when e.g., choosing and modifying subtrees, such as the one that has been demonstrated beneficial in [19, 20] and it is adopted in this work too.

To the best of our knowledge, however, there is no scientific study on the adoption of semantics in Evolutionary Multi-Objective Optimisation, at large, and in Multi-objective GP (MOGP), in particular, except from the work conducted by Galván et al. [10, 11].

The goal of this paper is to incorporate semantics into a MOGP paradigm. We use two well-known and widely popular MO paradigms: the Nondominated Sorting Genetic Algorithm II (NSGA-II) [6] and the Strength Pareto Evolutionary Algorithm (SPEA2) [25]. We use three different forms of incorporating semantics in a MO approach: (a) semantic similarity-based crossover (SSC) that is based on a widely known, but computationally costly, form of semantic-based single objective GP approach where the goal is to promote semantic diversity by applying, potentially, multiple times crossover between two parents [19], (b) semantic-based crowding distance (SBCD), which is based on the adoption of a semantic distance incorporated into the core of the MO paradigms that replaces the crowding distance [10], and finally, the proposed approach, (c) semantic-based crowding distance as an objective (SBCDO) that is based on SBCD and uses the resulting distance values as another objective in a MO paradigm encouraging diversity.

To study the effects of semantic-based MOGP paradigms, we use challenging unbalanced binary classification problems. We show how the adoption of semantics in MOGP optimisation is beneficial. In particular, the SBCDO is able to achieve better results on the average hypervolume and the Pareto optimal front compared to the rest of the approaches used in this work including their corresponding canonical MO paradigms.

This paper is organised as follows. Next, we present some related work. In Section 3, we introduce the approaches used in this work. Section 4 provides details on the experimental setup used. The results presented in this paper are discussed in Section 5, and finally, conclusions and future work are drawn in Section 6.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GECCO '19, July 13–17, 2019, Prague, Czech Republic

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-6111-8/19/07...\$15.00

<https://doi.org/10.1145/3321707.3321854>

2 RELATED WORK

2.1 Semantics in Genetic Programming

Studies of semantics in GP have increased dramatically over the last years given that it has been consistently reported to be beneficial in GP search [10, 14, 19, 21, 22].

Even though researchers have proposed different mechanisms to use the semantics of GP programs to guide search, it is commonly accepted that semantics refers to the behaviour of a GP program once this is executed on a data set (also known as fitness cases in the GP literature). We give a formal treat of semantics in Section 3.

Semantics can be categorised in one of two main groups: indirect or direct semantics. Indirect semantics approaches refer to those methods that act on the syntax (genotype) of GP individuals to indirectly promote semantic diversity. On the other hand, direct semantic approaches refer to those mechanisms that adapt genetic operators to work directly on the semantics of GP individuals. A survey of semantic methods in GP can be found in [22]. Next, we present a few papers in the area relevant to our work.

The work conducted by McPhee et al. [13] paved the way for the proliferation of indirect semantics works. In their work, McPhee et al. studied the semantics of subtrees and the semantics of context (the remainder of a tree after the removal of a subtree). The authors pointed out how a high proportion of individuals created by the widely used 90-10 crossover operator are semantically equivalent. That is, the crossover operator does not have any useful impact in the semantic space of GP, which in consequence leads to a lack of performance increase as evolution continues.

With the goal of overcoming the lack of semantic diversity reported by McPhee et al. [13], Beadle and Johnson [2] proposed a Semantically Driven Crossover (SDC) operator, that promotes semantic diversity. More specifically, they used reduced ordered binary decision diagrams (ROBDD) on Boolean problems to check for semantic similarity between parents and offspring. The authors showed a significant improvement in terms of fitness increased when promoting semantic diversity using the SDC operator. They also showed that by using ROBDD on these particular problems, the SDC operator was able to considerably reduce bloat.

These studies used discrete fitness-valued cases, impeding their findings to be generalised in continuous fitness-valued cases. Uy et al. [15] addressed this major limitation and multiple works were inspired by their approach.

Uy et al. [15] measured the semantic equivalence of two given expressions by measuring them against a random set of points sampled from the domain. If the resulting outputs of these two expressions were close to each other, subject to a threshold value, these expressions were regarded as semantically equivalent. In their experimental design, the authors focused their attention on the semantics of subtrees. More specifically, the authors tried to encourage semantic diversity by executing crossover, for a number of trials, if two subtrees were not semantically equivalent. They showed, for a number of symbolic regression problems, that by promoting semantic diversity, they obtained better results compared when no semantic diversity was encouraged.

More recently, Forstenlechner et al. [8] showed how it is possible to compute the semantics of GP individuals for program synthesis. This operates on a range of different data types as opposed to those working on a single type of data. They computed the semantics of a GP individual by tracing it. To promote semantic diversity, the authors used two metrics, named partial change, used in the first instance, and any change, used only if the first instance failed to be satisfied to try to avoid using standard crossover. Partial change checks for every variable if there is at least one difference between the semantics from the subtrees in a single entry in the vector. However, the vectors are not allowed to be completely different. Whereas any change does not have the constraint shown in partial change. The authors reported that a semantic-based GP system achieved better results in 4 out of 8 problems used in their studies.

Indirect semantic has also been used in local search methods. For example, Dou and Rockett [7] focused their attention on how the performance of a GP system improves when semantically-aware local search methods are used. To this end, the authors used a MOGP paradigm handling both ‘goodness-of-fit’ and model complexity. In their analysis performed on a wide variety of GP and local search variants, that included the use three subtree selection methods and four replacement mechanisms to perform local search, the authors reported interesting findings including the fact that a given local search algorithm following a steady-state global search performs better than the corresponding algorithm that used generational local search. It was also reported that when GP is used as a local search operator, the steady-state GP does not show any consistent advantage over the generational GP.

One of the first direct semantic approaches that discusses the idea of acting directly on the semantic space is the one conducted by Moraglio et al. [14]. The motivation to propose a direct semantic approach was because many of the proposed indirect semantic approaches are wasteful e.g., [15, 19]. To address this, the authors used previous theoretical results to define transformations on the structure of tree-like GP individuals that correspond to geometric operators, and by doing so, being able to inherit their properties.

Moraglio’s findings are encouraging, however there are some limitations. One of them is the fact that their approach allows the presence of neutrality which can be harmful on problems with certain properties (e.g., unimodal landscapes [17]). Another limitation is that the transformations applied to GP individuals makes them significantly larger. To deal with this, the authors proposed to simplify the expressions while maintaining the same computed function. They showed how, by doing so, one can use semantic operators. However, as pointed by Vanneschi et al. [21], the simplification step proposed by Moraglio et al. is computationally expensive. To overcome this problem, they proposed an efficient cache implementation of the geometric semantic operators. The approach is based, among other things, on storing the semantics of individuals into a table, which makes the process efficient indeed. As acknowledged by Vanneschi et al., there is at least one limitation with their approach: because they use tables to store e.g., the semantics of individuals, syntax of trees; the reconstruction of individuals is difficult or impossible.

2.2 Works on Multi-Objective Genetic Programming

In a Multi-objective optimisation (MO) problem, one optimises with respect to multiple goals or multiple fitness functions. Thus, the task of the algorithm is to find acceptable solutions by considering all the criteria simultaneously.

One can achieve this in various ways. Broadly speaking, MO in GP has been adopted in two forms: (a) combining multiple objectives into an aggregate scalar fitness function, and (b) keeping the objectives separate. The former type of MO has been almost abandoned by the GP community given its naive form. Instead, in this work, we adopt the latter, more elegant, approach.

A natural form of adopting a MO approach in GP is by keeping the objectives separate and use the notion of *Pareto dominance* as adopted in Evolutionary MO (EMO) [5].

EMO offers an elegant solution to the problem of optimising two or more conflicting objectives. As will be seen in Section 3.2, the aim of EMO is to simultaneously evolve a front of the best trade-off solutions along the objectives in a single run. Next, we discuss just a few GP works that have adopted EMO in their studies.

Bloat (dramatic increase of tree sizes as evolution proceeds) control has been tackled via the aggregation of objectives into an aggregate fitness function [12]. However, Bleuler et al. [4] took a more natural approach to control bloat. The authors defined two, potentially, conflicting objectives: the functionality (fitness) of a program and the size of the tree. They compared their, at the time, innovative approach against well-adopted techniques (e.g., constant parsimony) to control GP bloat. They showed how their MO approach successfully controls bloat which results in finding shorter solutions, so the proposed mechanism evaluates expressions faster and achieves better performance.

MOGP has been used to classify highly unbalanced binary data [3, 10]. To do so, the authors treated each objective (class) ‘separately’ using well-known EMO optimisation approaches [6, 25]. Bhowan et al. [3] and Galván et al. [10] shown, independently, how MOGP was able to achieve high accuracy in classifying binary data in conflicting learning objectives (i.e., normally a high accuracy of one class results in lower accuracy on the other).

In the same vein, Zhao showed how MOGP can be successfully used to specify partial preferences on the conflicting objectives by embedding these preferences into the fitness function [23]. The motivation to do this, as specified by the author, is due to the fact that very often misclassification errors are not equally costly (e.g., it is more costly to approve a bad loan than to deny a good loan).

MOGP has also successfully been used in computer vision problems. For example, Shao et al. showed how it is possible to automatically generate domain-adaptive global feature descriptors for the classification of images [18]. Similarly to the work conducted by Bleuler et al. [4], Shao and co-authors also considered the size of the trees as one of objectives in MOGP, where the other natural objective for image classification considered in their studies was the classification error rate. They tested their approach in four well-known data sets. The authors showed how their proposed MOGP approach always produced better results compared to other 14 methods used in their studies.

3 SEMANTIC MOGP METHODS

3.1 Background on Semantics

For clarity purposes, we first briefly give some definitions on semantics, based on Pawlak et al. work [16], that will allow us to describe our approach later in this section.

Let $p \in P$ be a program from a programming language P . When p is applied to an input $in \in I$, p produces an output $p(in)$.

DEFINITION 1. *The semantic mapping function $s : P \rightarrow S$ maps any program p to its semantics $s(p)$.*

This intuitively means that,

$$s(p_1) = s(p_2) \iff \forall in \in I : p_1(in) = p_2(in).$$

As indicated in [16], the semantics specified in Def. 1 has three properties. Firstly, every program has only and only one semantics. Secondly, two or more programs can have the same semantics. Thirdly, programs that produce different outputs have different semantics.

Def. 1 is general as it does not specify how semantics is represented. This work, as indicated previously, uses a popular version of semantics GP where the semantics of a program is defined as the vector of output values computed by this program for an input set (also known as fitness cases). A fitness case is a pair consisting of a program input and its corresponding desired program output. Thus, a fitness case is a pair from $I \times O$. Assuming we use a finite set of fitness cases, as normally adopted in GP, we can now define the semantics of a program.

DEFINITION 2. *The semantics $s(p)$ of a program p is the vector of values from O obtained by executing p on every in from I ,*

Thus, we have that the semantics of a program is given by,

$$s(p) = [p(in_1), p(in_2), \dots, p(in_l)]$$

where $l = |I|$ is the size of the set of the fitness cases.

This form of semantics has widely been used in the specialised literature e.g., [10, 19] and it is also use in this work.

3.2 Background on MO

Multi-objective optimisation (MO) is concerned with the simultaneous optimisation of several objectives. When these are in conflict, no single solution exists, and trade-offs between the objectives must be sought. The optimal trade-offs are the solutions for which no objective can be further improved without degrading another objective. This idea is captured in the Pareto dominance relation: a point x in the search space is said to *Pareto-dominate* another point y if x is at least as good as y on all objectives and strictly better on at least one objective.

In this work, the objectives are to be maximised. Thus, the *Pareto dominance* concept is defined in Eq. 1. Similarly, solutions are *non-dominated* if they are not dominated by any solution in the population,

$$S_i > S_j \iff \forall_m [(S_i)_m \geq (S_j)_m] \wedge \exists k [(S_i)_k > (S_j)_k] \quad (1)$$

where $(S_i)_m$ indicates the performance of solution S_i on the m th objective.

The set of optimal trade-off solutions of a MO problem can then be defined as the set of points of the search space that are non-dominated by any other point, and is called the *Pareto set* of the problem at hand. The goal of Pareto EMO is to identify the Pareto set, or a good approximation of it. The *Pareto front* is the image of the Pareto set in the objective space.

The notion of Pareto-dominance has inspired different measures. The most well-known are dominance rank [6] and dominance count [4]. The former measure refers to the number of solutions in the population that dominate a given solution (lower is better). The latter measure refers to the number of other solutions that a particular solution dominates (higher is better).

There are two popular EMO approaches that use these measures and that are used in this work: the Nondominated Sorting Genetic Algorithm II (NSGA-II) [6] that uses dominance rank, and the Strength Pareto Evolutionary Algorithm (SPEA2) [4] that employs both dominance rank and dominance count.

In NSGA-II, the dominance rank determines the fitness of solution S_i : the number of other solutions in the population that dominate S_i . This is expressed in Eq. 2. Thus, the best fitness value of a solution is 0 (nondominated solution). Similarly, a solution with high fitness indicates a solution dominated by many individuals.

$$\text{NSGA-II}(S_i) = |\{j|j \in \text{Pop} \wedge S_j \succ S_i\}|. \quad (2)$$

In SPEA2, as aforementioned, both dominance rank and dominance count measures are considered when computing the fitness of a solution. First, each solution in the population is assigned a strength value D . This denotes the number of solutions that S_i dominates (dominance count for solution S_i).

$$D(S_i) = |\{j|j \in \text{Pop} \wedge S_i \succ S_j\}|.$$

To determine the fitness of a solution in SPEA2, we then use the strengths of all its dominators. This is expressed as follows,

$$\text{SPEA2}(S_i) = \sum_{j \in \text{Pop}, S_i \succ S_j} D(S_j). \quad (3)$$

The fitness value for a solution S_i in SPEA2 is then the sum of all dominance counts of other solutions in the population that are dominated by S_i . As in NSGA-II, fitness in SPEA2 is to be minimised where nondominated solutions have the best fitness value of 0.

3.2.1 Crowding Distance. Because Pareto dominance is not a total order, some additional criterion must be used so as to allow the comparison of any pair of points of the search space. To this end, we also use a *crowding distance* measure. Crowding is the Manhattan distance between solutions in objective space, where sparsely populated regions are preferred over densely populated regions. The crowding distance is only used to resolve selection when the primary fitness is equal between two or more individuals. Thus, the crowding distance promotes diversity among individuals having the same Pareto rank: in objective space, for each objective, the individuals in the population are ordered, and the partial crowding distance for each of them is the difference in fitness between its two immediate neighbours. The crowding distance is the sum over all objectives of these partial crowding distances [6]. Intuitively, it can be seen as the Manhattan distance between the extreme vertices of the largest hypercube containing the point at

hand and no other point of the population. Selecting points with the largest crowding distance amounts to favour the low-density regions of the objective space, thus favouring diversity.

3.3 Semantic Similarity-based Crossover MOGP

To incorporate semantics in a MOGP paradigm, we first use a semantic similarity-based crossover (SSC) mechanism originally proposed by Uy et al. [19] which was used in a single-objective GP system. The idea is to promote semantic diversity by computing the sampling semantic distance. Using Def. 2, the authors computed this distance by calculating the average of the addition of difference values for every $in \in I$ between parent and offspring. If the distance value lies within a range, defined by $\alpha = 0.01$ and $\beta = 0.5$, then crossover is promoted to generate offspring. Because this condition may be hard to satisfy, the authors tried to encourage semantic diversity by repeatedly applying crossover up to 20 times. If after this, the condition is not satisfied, then crossover is executed as normal. We use the same approach with the recommended α and β values in both NSGA-II and SPEA2.

3.4 Semantic-based Crowding Distance MOGP

Another form to adopt semantics into a MOGP paradigm is to incorporate it into the core of the MO algorithm itself.

As indicated previously, we use the well-known NSGA-II and the SPEA2 algorithms and the adoption of semantics is performed by replacing the crowding distance (see Section 3.2.1) by a semantic-based indicator denominated Semantic-based Crowding Distance (SBCD) first used in [10]. This is computed in the following way: a *pivot* is chosen, being the individual from the first Pareto front (Rank 1) that is the furthest away from the other individuals of this front using the crowding distance. For each point, its semantic distance is computed using Def. 2. We count the number of absolute difference values between each $in \in I$ from the pivot and each individual that it is greater than 0.5 as suggested in [10]. Once these values are calculated, the SBCD is computed, similarly to the crowding distance, as the average of the semantic distance differences with its closest neighbours in each direction. The higher values yielded by SBCD are favored during the selection step of NSGA-II and SPEA2. This allows us to have a set of individuals that are spread in the semantic space, therefore, promoting semantic diversity, the same way NSGA-II and SPEA2 promote diversity ('spreadness') in the objective space.

3.5 Semantic-based Crowding Distance as an Objective MOGP

Semantic-based Crowding Distance as an Objective (SBCDO) considers elements from the semantic-based distance approach mentioned previously. SBCDO is also adopted into the core of the MO algorithms used in this study (NSGA-II and SPEA2). Before explaining this approach, let us briefly remind the reader a few elements considered in these two canonical EMO approaches. From a given population of size N , N offspring are created using standard variation operators (crossover and mutation). Parents and offspring are

merged, and the resulting population, of size $2N$, is ordered using non-dominated sorting, and a crowding distance is used as secondary criterion. The best N individuals according to this ranking are selected to survive at the next generation.

Because the underlying idea within NSGA-II and SPEA2 is to favour behavioural diversity, but only considering the fitness as a whole, it can be hoped that introducing semantic diversity in these algorithms can only enforce this idea. Thus, this distance is used in two forms, it replaces the crowding distance as done with the SBCD approach defined before and it is used as another objective to be optimised by the algorithm. More specifically, in our proposed SBCDO approach, this new objective is the semantic distance from each individual in $2N$ w.r.t. to the *pivot* taken from the first front as done with the SBCD method. This semantic distance is computed as described in Section 3.4. By doing so, we encourage semantic diversity in the objective space with the hope to favour more ‘spreadness’ in the space. The completion of the new population, if this is the case, is performed using the semantic-based crowding distance explained before.

4 EXPERIMENTAL SETUP

To study the effects of semantics in MOGP, we used challenging binary highly unbalanced classification problems taken from the well-known UCI Machine Learning repository [1]. These problems are of different nature and complexity, e.g., they have from a few features up to dozens of them, these features include binary, integer, and real-valued features. Table 1, adapted from [3], gives the details for all datasets used in this work. These have been used ‘as is’ i.e., we did not try to balance the classes out. For each dataset, half of the data (with the same class balance than in the whole dataset) was used as a training set and the rest as a test set. All reported results are on the latter.

The terminals are the problem features. The function set consists of the typical four standard arithmetic operators: $F = \{+, -, *, /\}$, where the latter operator is the protected division, which returns the numerator if the denominator is zero. These functions are used to build a classifier (e.g., mathematical expression) that returns a single value for a given input (data example to be classified). This number is mapped onto a set of class labels using zero as the class threshold. In our work, an example is assigned to the minority class if the output of the classifier is greater or equal to zero. It is assigned to the majority class, otherwise.

The common way to measure the fitness of a classifier for classification tasks is the overall classification accuracy: for binary classification, the four possible cases are shown in Table 2. Assuming the minority class is the positive class, the accuracy is given by

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

The drawback of using Eq. 4 alone is that it rapidly biases the evolutionary search towards the majority class [3].

A better approach is to treat each objective (class) ‘separately’ using a multi-objective approach. Two objectives are considered: the true positive rate given by

$$TPR = \frac{TP}{TP + FN} \quad (5)$$

and the true negative rate given by

$$TNR = \frac{TN}{TN + FP} \quad (6)$$

These measure the distinct accuracy for the minority (Eq. 5) and majority class (Eq. 6).

The experiments were conducted using a generational approach. The parameters used in this study are shown in Table 3. To obtain meaningful results, we performed an extensive empirical experimentation (2,400 independent runs in total)¹.

5 RESULTS

5.1 Front Hypervolume

In order to compare the different approaches, we use the hypervolume [5] of the evolved Pareto approximations as a measure of performance. For bi-objectives problems, the hypervolume of a set of points in objective space, using reference point $(0, 0)$, is easily computed as the sum of the areas of all trapezoids fitted under each point. Such measure was chosen as being the only known Pareto-compliant indicator to-date [24]: the larger the hypervolume, the better the performance. We also computed the Pareto-optimal front (POF) with respect to all 50 runs, i.e., the set of non-dominated solutions after merging all 50 Pareto-approximated fronts.

Tables 4 and 5 report, for each problem defined in Table 1, both the average hypervolume over 50 runs, and the hypervolume of the POF using the NSGA-II and SPEA2, with their associated semantic-based methods, respectively. In these tables, the highest POF values are underlined. Furthermore, the statistical significance for the results on the average hypervolume was computed using Wilcoxon Test at 95% level of significance, independently comparing each of the semantic-based approaches (SBCDO, SSC and SBCD) against NSGA-II and SPEA2. Those results that are statistically significant are in boldface.

Let us focus first our attention on the results achieved by NSGA-II and the semantic-based MOGP methods. These are shown in Table 4. As can be observed, our proposed SBCDO approach achieves better results when considering the average hypervolume. That is, in all the problems, our approach finds the best results compared to the other three methods (NSGA-II, NSGA-II SSC and NSGA-II SBCD). From these, five out of six are statistically significant and these are indicated in boldface. Moreover, our approach achieves the highest values on the Pareto-optimal front in five out of six problems used in our work (these results are underlined in Table 4). It is, however, worth mentioning that in one problem (Abal₁) our approach finds the same POF value compared to NSGA-II and NSGA-II SSC. On the other hand, when using the Abal₂ dataset, our approach finds the second best POF value among all the methods, although the difference is minimum (0.865 obtained by NSGA-II SSC vs. 0.864 achieved by NSGA-II SBCDO).

Let us now focus our attention on the results achieved by SPEA2 and the semantic-based MOGP methods. These are shown in Table 5. We can see, again, that our proposed SPEA2 SBCDO achieves the highest results in all the problems when considering the average hypervolume and in four of the six problems, these results are

¹50 independent runs, 6 problems, 4 MOGP approaches, 2 canonical MO methods (NSGA-II, SPEA2).

Table 1: Binary imbalanced classification data sets used in our research.

Data set	Classes Positive/Negative (Brief description)	Number of examples			Imb. Ratio	Features	
		Total	Positive	Negative		No.	Type
Ion	Good/bad (ionsphere radar signal)	351	126 (35.8%)	225 (64.2%)	1:3	34	Real
Spect	Abnormal/normal (cardiac tom. scan)	267	55 (20.6%)	212 (79.4%)	1:4	22	Binary
Yeast ₁	mit/other (protein sequence)	1482	244 (16.5%)	1238 (83.5%)	1:6	8	Real
Yeast ₂	me3/other (protein sequence)	1482	163 (10.9%)	1319 (89.1%)	1:9	8	Real
Abal ₁	9/18 (biology of abalone)	731	42 (5.75%)	689 (94.25%)	1:17	8	Real
Abal ₂	19/other (biology of abalone)	4177	32 (0.77%)	4145 (99.23%)	1:130	8	Real

Table 2: Confusion Matrix.

	Predicted positive	Predicted negative
Actual positive	True Positive (TP)	False Negative (FN)
Actual negative	False Positive (FP)	True Negative (TN)

Table 3: Summary of parameters.

Parameter	Value
Population Size	500
Generations	50
Type of Crossover	90% internal nodes, 10% leaves
Crossover Rate	0.60
Type of Mutation	Subtree
Mutation Rate	0.40
Selection	Tournament (size = 7)
Initialisation Method	Ramped half-and-half
Initialisation Depths:	
Initial Depth	1 (Root = 0)
Final Depth	5
Maximum Length	800
Maximum Final Depth	8
Independent Runs	50
Semantic Thresholds	0.5 (SBCDO, SBCD) $\alpha = 0.01$ and $\beta = 0.5$ (SSC)

statistically significant (indicated in boldface). As for the Pareto optimal front, SPEA2 SBCDO achieves the highest results in all six problems.

5.2 Pareto-optimal Front

Let us now focus on more detail on the coverage of the objective space achieved by the MOGP semantic-based methods and canonical NSGA-II and SPEA2, shown in the top and in the bottom of Figure 1, respectively. Due to space constraints, we only show the results on the Ion, Spect and Abal₂ datasets. These plots show the 50 Pareto front approximations obtained from the 50 independent runs for each of the methods and each of the datasets used in this work.

We decided to show the results on the aforementioned datasets because it is relatively clear to see how our proposed NSGA-II SBCDO approach, denoted by red hollow square symbols, has a

better coverage of the objective space compared with the rest of the semantic-based approaches as well as the canonical NSGA-II approach (denoted by black hollow circles). The coverage of the objective space achieved by our approach compared to the other methods in the other three data sets, Yeast₁, Yeast₂ and Abal₁, not shown in the paper, is fairly similar between each other as inferred by the POF indicators reported in Table 4.

If we now turn our attention on the coverage of the objective space achieved using SPEA2, shown at the bottom of Figure 1, we can see a similar trend: our proposed SPEA2 SBCDO, denoted by red hollow circles, has a better coverage compared to the other three methods, including the canonical SPEA2. This is clearer when using the Ion and Abal₂ datasets and less clear when using the Spect dataset.

The POF indicator is useful to determine, for instance, what approach has better coverage compared to another. However, it does not provide detailed information on all the Pareto fronts generated in all independent runs by a given method used in this work. We address this by plotting evolved solutions that were exclusively found by either one method or another and this is why some blank areas may be visible in some plots, in particular when using the Spect dataset. This is depicted in Figure 2. We only report the evolved solution using the NSGA-II against any of the semantic-based methods used in this work and using the Ion, Spect and Abal₂ datasets.

Let us see how our proposed SBCDO approach behaves in comparison to the NSGA-II shown in the left-hand side of Figure 2. It is clear to see that our SBCDO solutions (denoted by green hollow squares) yields significantly more solutions compared to those produced by NSGA-II (denoted by black hollow circles), regardless of the dataset used (the same is observed for the Yeast₁, Yeast₂ and Abal₁ datasets not shown in the paper). Moreover, it is important to note that there are more solutions yield by SBCDO that have higher accuracy in both the majority (x-axis) and the minority (y-axis) compared to the results produced by NSGA-II as well as the results produced by NSGA-II SSC and NSGA-II SBCD. The same tendency is observed when using SPEA2: our proposed SBCDO produces many more and better solutions compared to the rest of the algorithms.

When we continue comparing, now NSGA-II vs. NSGA-II SSC (centre) and NSGA-II vs. NSGA-II SBCD (right-hand side), we can see a mirror image: the canonical NSGA-II tends to produce better solutions compared to these semantic-based methods. The same trend is also observed when using SPEA2, again not shown in the paper due to space limitations.

Table 4: Average (\pm standard deviation) hypervolume, where the reference point is (0,0), of evolved Pareto-approximated fronts, Pareto optimal front (POF) for the four MOGP used in this work: NSGA-II, NSGA-II SBCDO, NSGA-II SSC and NSGA-II SBCD, over 50 runs. Boldface numbers are statistical significant (read text) compared to the results yield by canonical NSGA-II. The highest POF values are underlined.

Data Set	NSGA-II		NSGA-II SBCDO		NSGA-II SSC		NSGA-II SBCD	
	Hypervolume		Hypervolume		Hypervolume		Hypervolume	
	Average	POF	Average	POF	Average	POF	Average	POF
Ion	0.766 \pm 0.114	0.938	0.860 \pm 0.031	<u>0.953</u>	0.753 \pm 0.124	0.935	0.793 \pm 0.076	0.939
Spect	0.534 \pm 0.024	0.647	0.588 \pm 0.019	<u>0.675</u>	0.539 \pm 0.030	0.635	0.536 \pm 0.058	0.644
Yeast ₁	0.838 \pm 0.011	0.876	0.850 \pm 0.006	<u>0.881</u>	0.834 \pm 0.029	0.875	0.804 \pm 0.093	0.875
Yeast ₂	0.950 \pm 0.009	0.976	0.960 \pm 0.008	<u>0.980</u>	0.944 \pm 0.028	0.976	0.946 \pm 0.017	0.979
Abal ₁	0.847 \pm 0.058	<u>0.961</u>	0.875 \pm 0.066	<u>0.961</u>	0.812 \pm 0.086	<u>0.961</u>	0.809 \pm 0.098	0.956
Abal ₂	0.576 \pm 0.122	0.842	0.631 \pm 0.126	0.864	0.534 \pm 0.102	<u>0.865</u>	0.564 \pm 0.112	0.828

Table 5: Average (\pm standard deviation) hypervolume, where the reference point is (0,0), of evolved Pareto-approximated fronts, Pareto optimal front (POF) for the four MOGP used in this work: SPEA2, SPEA2 SBCDO, SPEA2 SSC and SPEA2 SBCD, over 50 runs. Boldface numbers are statistical significant (read text) compared to the results yield by canonical SPEA2. The highest POF values are underlined.

Data Set	SPEA2		SPEA2 SBCDO		SPEA2 SSC		SPEA2 SBCD	
	Hypervolume		Hypervolume		Hypervolume		Hypervolume	
	Average	POF	Average	POF	Average	POF	Average	POF
Ion	0.776 \pm 0.105	0.948	0.865 \pm 0.032	<u>0.955</u>	0.767 \pm 0.121	0.936	0.801 \pm 0.066	0.934
Spect	0.539 \pm 0.026	0.657	0.591 \pm 0.021	<u>0.674</u>	0.536 \pm 0.022	0.659	0.530 \pm 0.045	0.644
Yeast ₁	0.834 \pm 0.021	0.876	0.847 \pm 0.006	<u>0.880</u>	0.824 \pm 0.062	0.875	0.789 \pm 0.160	0.878
Yeast ₂	0.947 \pm 0.015	0.977	0.962 \pm 0.008	<u>0.979</u>	0.947 \pm 0.010	0.977	0.950 \pm 0.011	0.978
Abal ₁	0.832 \pm 0.085	0.960	0.862 \pm 0.097	<u>0.966</u>	0.854 \pm 0.082	0.965	0.803 \pm 0.122	0.961
Abal ₂	0.552 \pm 0.124	0.837	0.594 \pm 0.151	<u>0.851</u>	0.518 \pm 0.125	0.829	0.546 \pm 0.141	0.842

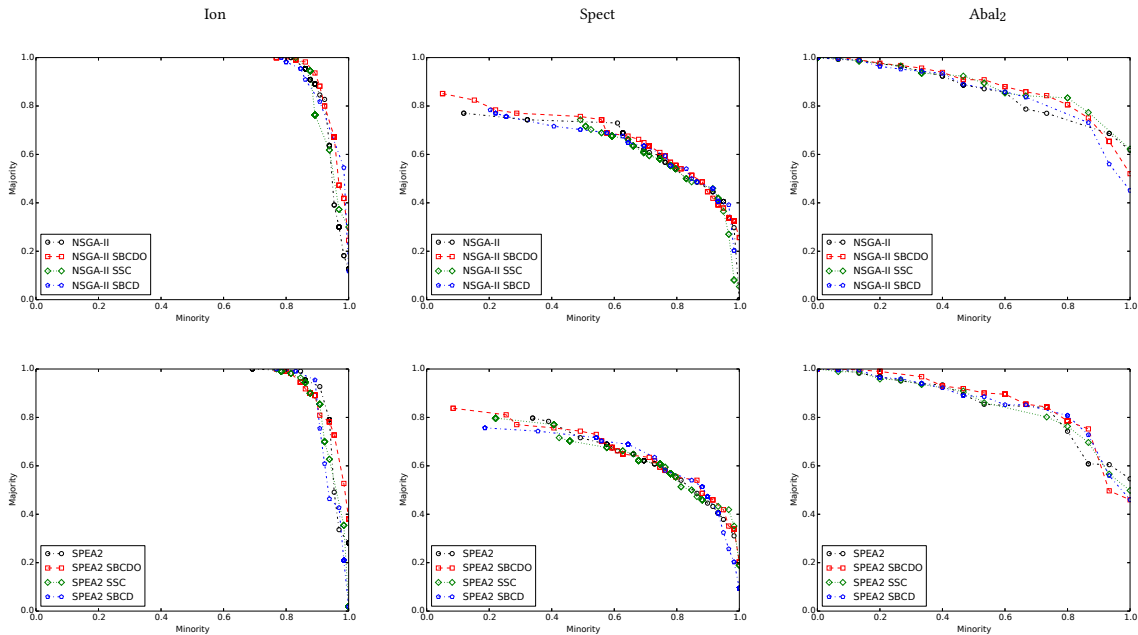


Figure 1: Pareto-Optimal fronts for Ion, Spect and Abal₂ using NSGA-II and its semantic-based MO variants (top row) and using SPEA2 and its semantic-based MO variants (bottom row).

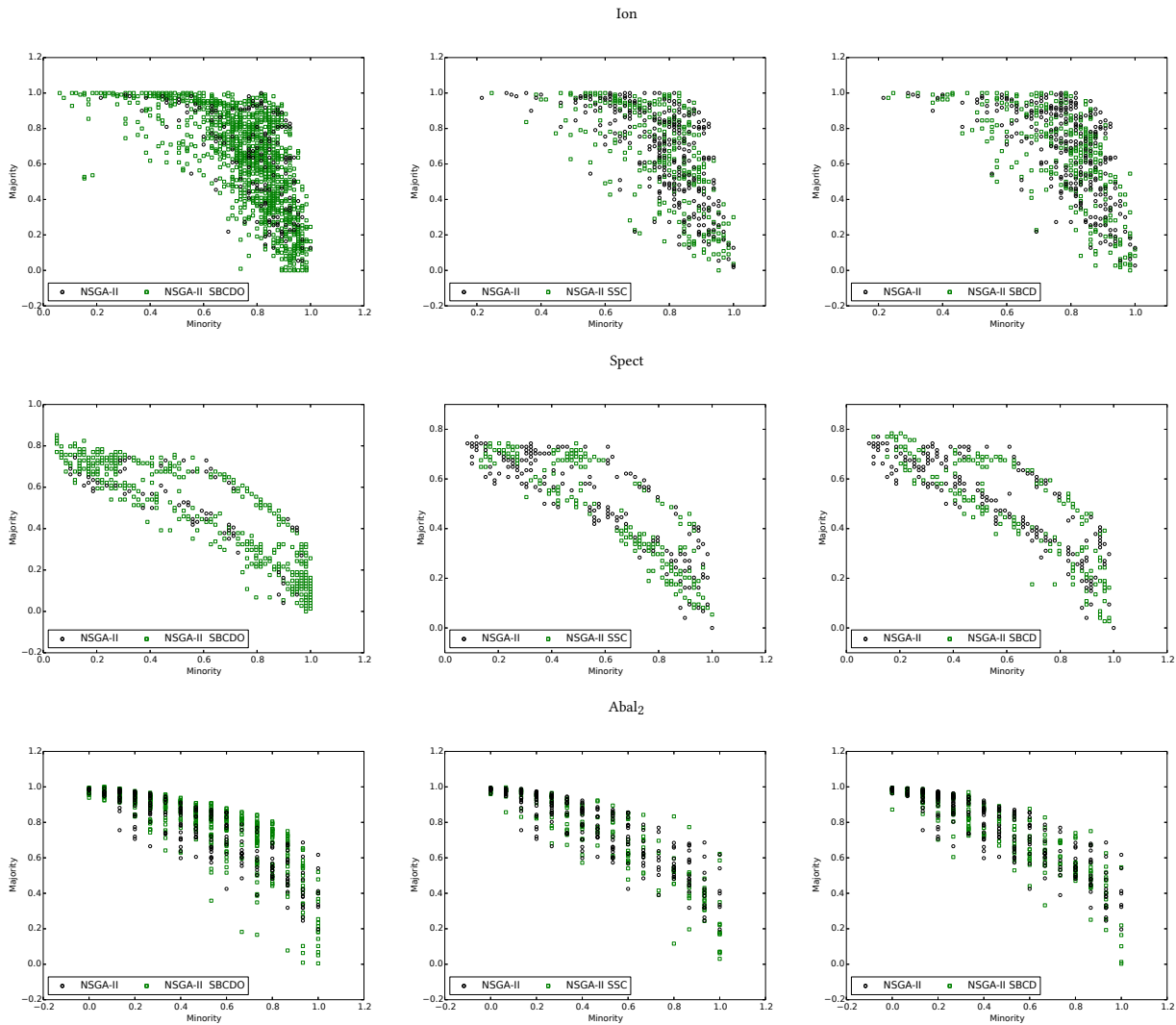


Figure 2: Evolved solutions that were exclusively found by either NSGA-II (indicated by black hollow circles symbols) and the rest of each of the methods used in this work: SBCDO (left-hand side), SSC (middle), SBCD (right-hand side), indicated by green hollow square symbols using the Ion dataset (top row), Spect dataset (middle row) or Abal₂ dataset (bottom row).

6 CONCLUSIONS AND FUTURE WORK

In Genetic Programming, semantics is commonly defined as the behaviour of syntactically correct programs. In canonical GP, semantics is represented by the output vector of the tree for different known inputs and the similarity between the semantics of two trees gives a much smoother idea of the similarity between the trees than either the syntactic description of the trees or their raw fitness.

This work proposes a new form to add semantics into the core of a Multi-objective Genetic Programming (MOGP) using the well-known the Nondominated Sorting Genetic Algorithm II (NSGA-II) and the Strength Pareto Evolutionary Algorithm (SPEA2). Our approach, named semantic-based crowding distance as an objective (SBCDO) consists of using a semantic distance as another objective

in the MOGP with the hope to promote more ‘spreadness’ in the objective space. We also use this distance in lieu of the crowding distance at the heart of the aforementioned EMO algorithms.

We have shown that this form of semantics achieves better results compared to other forms of semantics used in EMO approaches as well as the canonical NSGA-II and SPEA2 algorithms. We have learned how it is feasible to promote semantic diversity in a MOGP by using a well-defined semantic-based distance.

Our immediate goal is to perform an in-depth study on the threshold values used in semantic-based methods that are necessary to continuous fitness-valued cases problems with the hope to make sound suggestions on the values that should be used in these type of problems.

REFERENCES

- [1] D.J. Newman A. Asuncion. 2007. UCI Machine Learning Repository. (2007). [http://www.ics.uci.edu/\\$sim\\$learn/](http://www.ics.uci.edu/simlearn/)
- [2] L. Beadle and C.G. Johnson. 2008. Semantically driven crossover in genetic programming. In *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence)*, 111–116. <https://doi.org/10.1109/CEC.2008.4630784>
- [3] U. Bhowan, M. Johnston, M. Zhang, and X. Yao. 2013. Evolving Diverse Ensembles Using Genetic Programming for Classification With Unbalanced Data. *IEEE Transactions on Evolutionary Computation* 17, 3 (June 2013), 368–386. <https://doi.org/10.1109/TEVC.2012.2199119>
- [4] S. Bleuler, M. Brack, L. Thiele, and E. Zitzler. 2001. Multiobjective genetic programming: reducing bloat using SPEA2. In *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, Vol. 1. 536–543 vol. 1. <https://doi.org/10.1109/CEC.2001.934438>
- [5] Carlos A. Coello Coello, Gary B. Lamont, and David A. Van Veldhuizen. 2006. *Evolutionary Algorithms for Solving Multi-Objective Problems (Genetic and Evolutionary Computation)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- [6] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6 (2002), 182–197.
- [7] Tiantian Dou and Peter Rockett. 2018. Comparison of semantic-based local search methods for multiobjective genetic programming. *Genetic Programming and Evolvable Machines* 19, 4 (01 Dec 2018), 535–563. <https://doi.org/10.1007/s10710-018-9325-4>
- [8] Stefan Forstenechner, David Fagan, Miguel Nicolau, and Michael O'Neill. 2018. Towards Effective Semantic Operators for Program Synthesis in Genetic Programming. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '18)*. ACM, New York, NY, USA, 1119–1126. <https://doi.org/10.1145/3205455.3205592>
- [9] E. Galván-López, B. Cody-Kenny, L. Trujillo, and A. Kattan. 2013. Using semantics in the selection mechanism in Genetic Programming: A simple method for promoting semantic diversity. In *2013 IEEE Congress on Evolutionary Computation. 2972–2979*.
- [10] Edgar Galván-López, Efrén Mezura-Montes, Ouassim Ait ElHara, and Marc Schoenauer. 2016. On the Use of Semantics in Multi-objective Genetic Programming. In *Parallel Problem Solving from Nature – PPSN XIV: 14th International Conference, Edinburgh, UK, September 17-21, 2016, Proceedings*, Julia Handl et al. (Eds.). Springer, 353–363. https://doi.org/10.1007/978-3-319-45823-6_33
- [11] Edgar Galván-López, Lucia Vázquez-Mendoza, and Leonardo Trujillo. 2016. Stochastic Semantic-Based Multi-Objective Genetic Programming Optimisation for Classification of Imbalanced Data. In *Advances in Soft Computing*, Obdulia Pichardo-Lagunas and Sabino Miranda-Jiménez (Eds.). Springer, Chapter 22, 261–272.
- [12] John R. Koza. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA.
- [13] Nicholas Freitag McPhee, Brian Ohs, and Tyler Hutchison. 2008. Semantic building blocks in genetic programming. In *Proceedings of the 11th European conference on Genetic programming (EuroGP'08)*. Springer-Verlag, Berlin, Heidelberg, 134–145. <http://dl.acm.org/citation.cfm?id=1792694.1792707>
- [14] Alberto Moraglio, Krzysztof Krawiec, and Colin G. Johnson. 2012. Geometric Semantic Genetic Programming. In *PPSN (1) (LNCS)*, Carlos A. Coello Coello, Vincenzo Cutello, Kalyanmoy Deb, Stephanie Forrest, Giuseppe Nicosia, and Mario Pavone (Eds.), Vol. 7491. Springer, 21–31.
- [15] Quang Uy Nguyen, Xuan Hoai Nguyen, and Michael O'Neill. 2009. Semantic Aware Crossover for Genetic Programming: The Case for Real-Valued Function Regression. In *Proceedings of the 12th European Conference on Genetic Programming (EuroGP '09)*. Springer-Verlag, Berlin, Heidelberg, 292–302. https://doi.org/10.1007/978-3-642-01181-8_25
- [16] T. P. Pawlak, B. Wieloch, and K. Krawiec. 2015. Semantic Back-propagation for Designing Search Operators in Genetic Programming. *IEEE Transactions on Evolutionary Computation* 19, 3 (June 2015), 326–340. <https://doi.org/10.1109/TEVC.2014.2321259>
- [17] Riccardo Poli and Edgar Galván-López. 2007. On the Effects of Bit-Wise Neutrality on Fitness Distance Correlation, Phenotypic Mutation Rates and Problem Hardness. In *Foundations of Genetic Algorithms*, Christopher R. Stephens, Marc Toussaint, Darrell Whitley, and Peter F. Stadler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 138–164.
- [18] L. Shao, L. Liu, and X. Li. 2014. Feature Learning for Image Classification Via Multiobjective Genetic Programming. *IEEE Transactions on Neural Networks and Learning Systems* 25, 7 (July 2014), 1359–1371. <https://doi.org/10.1109/TNNLS.2013.2293418>
- [19] Nguyen Quang Uy, Nguyen Xuan Hoai, Michael O'Neill, R. I. McKay, and Edgar Galván-López. 2011. Semantically-based crossover in genetic programming: application to real-valued symbolic regression. *Genetic Programming and Evolvable Machines* 12, 2 (2011), 91–119. <https://doi.org/10.1007/s10710-010-9121-2>
- [20] Nguyen Quang Uy, Michael O'Neill, Nguyen Xuan Hoai, Bob McKay, and Edgar Galván-López. 2010. *Semantic Similarity Based Crossover in GP: The Case for Real-Valued Function Regression*. Springer Berlin Heidelberg, Berlin, Heidelberg, 170–181. https://doi.org/10.1007/978-3-642-14156-0_15
- [21] Leonardo Vanneschi, Mauro Castelli, Luca Manzoni, and Sara Silva. 2013. *A New Implementation of Geometric Semantic GP and Its Application to Problems in Pharmacokinetics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 205–216. https://doi.org/10.1007/978-3-642-37207-0_18
- [22] Leonardo Vanneschi, Mauro Castelli, and Sara Silva. 2014. A Survey of Semantic Methods in Genetic Programming. *Genetic Programming and Evolvable Machines* 15, 2 (June 2014), 195–214. <https://doi.org/10.1007/s10710-013-9210-0>
- [23] Huimin Zhao. 2007. A multi-objective genetic programming approach to developing Pareto optimal decision trees. *Decision Support Systems* 43, 3 (2007), 809–826. <https://doi.org/10.1016/j.dss.2006.12.011> Integrated Decision Support.
- [24] Eckart Zitzler, Dimo Brockhoff, and Lothar Thiele. 2007. *The Hypervolume Indicator Revisited: On the Design of Pareto-compliant Indicators Via Weighted Integration*. Springer Berlin Heidelberg, Berlin, Heidelberg, 862–876. https://doi.org/10.1007/978-3-540-70928-2_64
- [25] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. 2001. *SPEA2: Improving the Strength Pareto Evolutionary Algorithm*. Technical Report.