

GP-DMD: A Genetic Programming variant with Dynamic Management of Diversity

Ricardo Nieto Fuentes (✉ nifr91@gmail.com)

Centro de Investigacion en Matematicas AC <https://orcid.org/0000-0002-7559-8620>

Carlos Segura

Centro de Investigacion en Matematicas AC

Research Article

Keywords: Genetic Programming, Diversity Management, Exploration, Intensification, Bloat

Posted Date: March 19th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-342085/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

GP-DMD: A Genetic Programming variant with Dynamic Management of Diversity

Ricardo Nieto-Fuentes · Carlos Segura

Received: date / Accepted: date

Abstract The proper management of diversity is essential to the success of Evolutionary Algorithms. Specifically, methods that explicitly relate the amount of diversity maintained in the population to the stopping criterion and elapsed period of execution, with the aim of attaining a gradual shift from exploration to exploitation, have been particularly successful. However, in the area of Genetic Programming (GP), the performance of this design principle has not been studied. In this paper, a novel GP variant, Genetic Programming with Dynamic Management of Diversity (GP-DMD), is presented. GP-DMD applies this novel design principle through a replacement strategy that combines penalties based on distance-like functions with a multi-objective Pareto selection based on maximizing accuracy and minimizing tree size. The experimental validation based on the symbolic regression task considers several state-of-the-art approaches and shows clear improvements both in terms of accuracy and simplicity. The effects of GP-DMD on the dynamics of the population are also analyzed, revealing the reasons for the superiority of GP-DMD. As in other fields of Evolutionary Computation, this design principle contributes significantly to the area of GP.

Keywords Genetic Programming · Diversity Management · Exploration · Intensification · Bloat

1 Introduction

Genetic Programming [30] (GP) is a successful paradigm of Evolutionary Algorithms (EAs) that has been applied to a wide variety of domains [2, 20, 25]. In this paradigm, a population of computer codes or mathematical models is evolved through the iterative application of selection, genetic and replacement operators. In the broader field of Evolutionary Computation (EC), the proper balance between exploration and exploitation

Carlos Segura
Jalisco S/N Col. Valenciana, Guanajuato, Gto, CP: 3623, México
E-mail: carlos.segura@cimat.mx

R. Nieto-Fuentes
Jalisco S/N Col. Valenciana, Guanajuato, Gto, CP: 3623, México
E-mail: nift91@gmail.com

is considered a cornerstone for proper performance [33]. Since there is a clear relationship between this balance and the amount of diversity maintained in the population, several strategies to alter the amount of population's diversity have been devised [35]. They are usually classified in terms of the component that is altered, and they can alleviate numerous inconveniences [34], such as premature convergence and oversampling of neutral networks, among others.

In recent years, a novel design paradigm that explicitly relates the amount of diversity maintained in the population to the stopping criterion and elapsed period of execution has yielded important advances both for single-objective continuous and combinatorial optimization [9, 34]. The main principle behind this design paradigm is that the internal operations of EAs should depend on the amount of computational resources available, and in particular, that the initial phases should focus on exploration while the final phases should be devoted to exploitation, with a gradual transition between those stages. This principle has been successfully used to design new components, such as replacement strategies [34], and crossover operators [8]. In fact, this principle was used to design the winning strategy of the extended round of Google Hash Code 2020¹, which featured more than 100,000 participants, thus indicating the potential of this principle.

In the case of GP, several authors have already noted the impact of diversity management on performance. Several ways of measuring diversity and strategies to alter the amount of diversity maintained in the population have been devised [3]. However, to our knowledge, the design principle previously described has never been applied in the area of GP. This work analyzes the hypothesis that this design principle is also useful for GP and that state-of-the-art strategies can be advanced further. Accordingly, this paper presents a new GP algorithm that considers the explicit and dynamic management of diversity that takes into account the stopping criterion and elapsed period. The novel proposal is called Genetic Programming with Dynamic Management of Diversity (GP-DMD) and, similarly to some of the successful single-objective optimizers [34], it incorporates the principle discussed through a novel replacement strategy. This replacement strategy extends the ones applied in single-objective optimization domains by introducing a multi-objective selection that considers that in the GP case, both the maximization of accuracy and the minimization of tree sizes are important aspects. However, the main core is similar to the one applied in the single-objective optimization case. Symbolic regression, which is one of the most studied tasks in GP, is used to show the benefits of GP-DMD. Comparisons against a large number of GP strategies, including some diversity-aware strategies, show the important benefits of our proposal. In addition to analyses related to accuracy and tree sizes, the dynamics of the population in terms of several features are studied, which reveals the reasons for the superior performance of GP-DMD.

The rest of this paper is organized as follows. Section 2 provides a summary of the most relevant strategies for diversity management in the field of GP, including a taxonomy for diversity measures. Our proposal, the Genetic Programming with Dynamic Management of Diversity (GP-DMD), is described in Section 3. Then, the experimental validation is presented and discussed in Section 4. Several state-of-the-art strategies are used to illustrate the benefits of GP-DMD and, in addition to our results, some analyses related to population dynamics are provided. Finally, the main conclusion and some lines of future work are given in Section 5.

¹ <https://codingcompetitions.withgoogle.com/hashcode/>

2 Diversity in Genetic Programming

Since the proper management of diversity is considered so important in the Evolutionary Computation area, quite a large number of strategies to alter the amount of diversity maintained in the population have been devised [35]. GP is no exception, so as a result several diversity-aware proposals that are inspired by well established diversity management strategies [15], as well as techniques that are specific to GP, have been developed [5]. Due to the particular kinds of individuals evolved in GP, measuring the diversity of a population of individuals is a complex task. This section provides a representative sample of diversity measures and strategies to alter the amount of diversity maintained in GP. Most of the techniques discussed are used to validate the proposal put forth in this paper.

2.1 Diversity measures

Several ways of calculating diversity in GP, both for analysis and design, have been devised. However, different nomenclatures have been used to refer to similar measures and probably worse, the same term has been used to refer to different measures. For instance, the term phenotypic distance has been used both to refer to distances based on the fitness values [4] and distances based on the mathematical models [18]. In order to avoid misunderstandings, we present a classification of diversity measures that considers the most typical use of the terms.

Structural diversity is related to the variety of mathematical models associated to each individual. Note that in this paper, our mathematical models are trees, so the discussion is restricted to this kind of structure. Since calculating tree-based diversity is complex and expensive [19], several ways of estimating it have been devised. These approximations can be categorized into the following classes:

- In edit distances [4], a set of edit transformations and associated costs are established. Then, the edit distance is defined as the minimum-cost sequence of edit operations that transform a given tree into another one. One of the most popular is the *ed2* distance. In this case, the two trees compared (t_1 and t_2) are first overlapped and brought to the same structure by adding null nodes. Then, the distance is defined as $ed2(t_1, t_2) = dist(p, q) + w \sum_{l=1}^m ed2(t_{1,l}, t_{2,l})$, where p and q are the roots of t_1 and t_2 , m is the number of children of the roots, $t_{i,j}$ is the subtree number j of the root of t_i and $w \in \mathbb{R}$ is a constant. $dist(p, q)$ is usually defined as one for unequal nodes p and q and zero for equal nodes [4] and this was the decision adopted in this paper. Note that the constant w sets the importance of each level. Specifically, the values $w < 1$, $w > 1$ and $w = 1$ grant more importance to upper, lower levels or the same for all levels, respectively. The value $w = 0.5$ is broadly used in GP and is the value adopted in this paper.
- Subtree based distances take into account the set of subtrees appearing in each candidate solution to estimate their differences. The proposal by Keijzer (*kj*) [17] is probably the most popular belonging to this group. This distance is defined as follows: $kj(t_1, t_2) = |S_{t_1} \cup S_{t_2}| - |S_{t_1} \cap S_{t_2}|$, where S_{t_i} is the set of subtrees of t_i .

Behavioral diversity is based on checking the output of the mathematical models with the training cases. There has been an increasing interest in analyzing behavioral diversity because structural diversity does not guarantee behavioral diversity [4]. These approximations can be categorized into the following classes:

- 118 – In fitness-based measures, the fitness, which summarizes in a single scalar the overall
 119 performance of the individual, is used in some way to estimate the diversity. For
 120 instance, it can be done through the variance of the fitness values appearing in the
 121 population [11].
- 122 – In semantic-based measures, the semantic of a program t ($s(t)$) is defined as a vector
 123 containing, for each training case, its output or a value related to its performance. Then,
 124 this vector is used to establish the differences between individuals [12]. The *Sampling*
 125 *Semantic Distance* (SSD) [26] is a popular example belonging to this category, and it is
 126 calculated as $SSD(t_1, t_2) = T^{-1}(|s(t_1)[1] - s(t_2)[1]| + \dots + |s(t_1)[T] - s(t_2)[T]|)$, where
 127 T is the number of training cases.

128 Finally, note that entropy-based measurements are also quite typical, but they can be
 129 considered as special cases of the previous ones. In these kinds of schemes, the
 130 mathematical models are partitioned in some way and the spread of the population over the
 131 partition is taken into account. In order to partition the population, some of the methods
 132 previously discussed are used. For instance, in [11], edit distances to a reference tree are
 133 used to establish the partitions. Then, the entropy of the population P is calculated by
 134 taking into account the fraction of individuals belonging to each group of the partition.
 135 Thus, this is an entropy based on structural diversity. An example where the partitions are
 136 established in terms of the fitness function values, i.e. an entropy based on behavioral
 137 diversity, is presented in [29].

138 2.2 Diversity Management Strategies

139 As in usually done in EC, the diversity management strategies are classified in terms of the
 140 main component that is modified [35]. Most of the strategies discussed in this section are
 141 used to validate our proposal, so both novel as well as more mature strategies are taken into
 142 account.

143 In the *selection-based* techniques, the parent selection procedure is modified, usually
 144 with the aim of biasing the search towards uncrowded regions. Some of the most popular
 145 strategies belonging to this category are the following:

- 146 – ϵ -lexicase selection (ϵ -lex) [21] is an adaption of the well-known lexicase selection to
 147 the symbolic regression case. Lexicase selects parents by iteratively considering single
 148 cases and discarding non-elite solutions, meaning that the semantic diversity is taken
 149 into account. ϵ -lex modules the pass condition with the aim of diversifying further the
 150 selected parents for symbolic regression by considering a threshold value (ϵ) in the filter
 151 step. Several ways of setting ϵ were tested, with the Sum-MAD strategy, which is based
 152 on considering the median of absolute deviations, providing quite robust results.
- 153 – Semantic in Selection (SIS) [13] relies on performing two kinds of selections to promote
 154 quality and behavioral diversity. For each pair of parents, the first one is selected by a
 155 tournament based on the fitness value, whereas the second one is selected by sampling a
 156 pool of PS individuals and then selecting, among the ones with a different semantic than
 157 the first parent, the one with the best fitness value. If none of the sampled individuals
 158 generate a different semantic, the individual is selected randomly. Note that in the case
 159 of symbolic regression, a threshold value (α) is used to compare behaviors. Thus, the
 160 behavior for a test case of two candidate solutions is considered to match when their
 161 differences are lower than α .

- k -nobelty selection (KNOB) [18] is also based on performing two kinds of selections. Specifically, selections based on lexibase (or ϵ -lexibase for continuous semantics) are used with probability $(1 - k)$, whereas the remaining selections are based on novelty. Thus, the first kind of selection comprises both diversity and quality, whereas the second type is based solely on diversity. The novelty is approximated by calculating the mean distance to a sampling of the previously generated solutions. Specifically, a bounded archive with size equal to $|A|$ is maintained and SP individuals are sampled to estimate the novelty. Several ways of setting k were analyzed and schemes based on a logarithmic decay excelled.

Crossover and mutation operators are quite important for the dynamics of the populations. Thus, redefining these operators by taking into account desired effects on diversity is also quite typical. *Variation-based* strategies alter the genetic operators and/or their probabilities. One of most popular strategies that act on the variation phase is the Diverse Partner Selection with Brood Recombination (DPSBR) [24]. DPSBR alters the probabilities of crossover (p_c) and mutation (p_m) dynamically and promotes crossing semantically distant individuals by forcing certain conditions related to the expected improvement. After selecting the parents with a typical tournament based on fitness, those conditions are checked and if they are not fulfilled, alternative parents are tested. After $N/2$ attempts, where N is the population size, the crossover is abandoned and the last pair of selected parents is accepted. Since this might be indicative of a behavioral diversity that is too low, p_c is reduced and p_m is increased by $1/N$. Note that in order to calculate the semantic distance, a binarization scheme is applied. This is done by setting a threshold value to distinguish between success and failure [10]. Additionally, brood recombination is applied, meaning that for each pair of parents, multiple recombinations (MR) are performed.

Replacement-based schemes modify the process of selecting the survivors of the next generation. Since this phase is in charge of erasing information, it is quite a natural component to promote diversity. The following strategies are some of the most popular in this category:

- Find Only and Complete Undominated Sets (FOCUS) [16] applies a multi-objective approach, where the objective triplets consist of fitness, size and the average squared distance to other individuals. Distances consider the extended population made up from the union of the current population and the offspring. Then, it applies the Pareto dominance criterion to identify the non-dominated solutions, which are selected to survive.
- Age Fitness Pareto Optimization (AFPO) [32] also applies multi-objective concepts but using a tuple that consists of fitness and age. The key idea is the concept of genetic age, which is related to the number of generations that the genetic information has been in the evolutionary process. Specifically, the age of solutions of the initial population is set to 0 and at each iteration, the age of every member of the population is increased. The age of the individuals created by crossover is inherited from the oldest parent and, in addition, in each generation a new random individual with its age set to 0 is created. Note that this step is similar to the random immigrants approach [14], but, together with the novel selection process, it further favors young individuals. Finally, the non-dominated individuals survive. However, in this case, there is a target population size (N) meaning that if less than N non-dominated individuals exist, some dominated individuals also survive. The dominated individuals are selected through Pareto tournaments.

209 – Genetic Marker Density Genetic Programming (GMD-GP) [5] is also based on Pareto
210 dominance, but in this case each individual is associated to a tuple that consists of
211 genetic marker density and fitness. Genetic markers are the main novelty of this
212 proposal and they are partial trees generated by traversing the generated trees from the
213 root to a specified depth (MD). Then, the partial tree generated for each individual is
214 used to calculate its density by considering the number of times that such a tree appears
215 in the population. The logic behind this proposal is that, for several cases, the
216 population converges very quickly in the top levels [23]. Note that the same
217 replacement procedure as in AFPO is applied, meaning that all nominated individuals
218 survive and, additionally, some dominated individuals might survive to reach the target
219 population size.

220 In the case of *population-based* diversity-management strategies, the typical panmictic
221 model is modified, for instance, by considering a notion of subpopulation or by altering the
222 population size dynamically. One of the most popular strategies belonging to this category
223 in the GP area is the Age Layered Population Structure [7] (ALPS) scheme. The core idea
224 behind ALPS is to protect recently created individuals in the population from being
225 overshadowed by fitter and more evolved individuals. Thus, only individuals with similar
226 ages compete among them. In order to attain this aim, the population is partitioned into
227 layers and each layer has a capacity (M) and an age limit. Specifically, the age limit of
228 layer i (age_i) is set to $AgeGap \times i^2$, except for the last layer, which has no age limit. Inside
229 each layer, a generational strategy with elitism is applied, meaning that the best EL
230 solutions in terms of their fitness, as well as the offspring, survive. At each generation, the
231 individuals' ages increase and they are promoted to the next layer if their age is greater
232 than their current layer limit. Additionally, individuals in the first layer are regenerated
233 randomly at specific generation intervals ($AgeGap$), meaning that short new individuals
234 enter in the competition. Genetic operators are applied inside layers to generate offspring
235 and the age is inherited from the oldest parent involved.

236 Finally, note that recently some hybrid diversity management strategies that combine
237 variants of several of the schemes presented above have also been devised. For instance,
238 GMD-GP has been combined with lexicase to simultaneously consider the behavioral and
239 structural diversity [6].

240 3 Proposal

241 The proposal put forth in this paper was designed by considering the hypothesis that
242 relating diversity management to the stopping criterion and elapsed period of execution
243 might bring additional benefits to the GP area. Specifically, this principle was used to
244 design a novel replacement phase. This component is an adaptation of the Replacement
245 with Multi-objective Dynamic Diversity Control (RMDDC), which is a strategy that has
246 been applied successfully in combinatorial single-objective optimization [34]. One of the
247 most important features of RMDDC lies in the incorporation of penalties in the
248 replacement phase with the aim of avoiding the survival of individuals that are too similar.
249 The definition of similarity is based on distance-like functions which are
250 problem-dependent, whereas the notion of being too similar is dynamic. In particular, an
251 initial threshold distance to distinguish between penalized and non-penalized individuals is
252 established. Then, this threshold is decreased linearly during the evolution in such a way
253 that it attains the value 0 at the end of the optimization process. Note that this means that as

Algorithm 1: Main Procedure of Standard GP

```

Input :  $N$  (population size), population-initialization, selection, crossover,
          mutation, replacement, stopping-criterion, evaluation
Output : The best solution found  $t_{best}$ 
1  $P = \text{population-initialization}(N)$ 
2  $\text{evaluation}(P)$ 
3 until  $\text{stopping-criterion}()$ 
4    $t_{best} = \text{best-individual}(\{t_{best}\} \cup P)$ 
5    $P_s = \text{selection}(P)$ 
6    $O' = \text{crossover}(P_s)$ 
7    $O = \text{mutation}(O')$ 
8    $\text{evaluation}(O)$ 
9    $P = \text{replacement}(P, O)$ 
10 end
11 return  $t_{best}$ 

```

the evolution progresses, closer individuals are accepted with the aim of shifting gradually from exploration to exploitation. The penalization promotes the survival of less fit but diverse solutions, but at the same time, the dynamic threshold promotes the search to focus on the most promising regions during the last phases of the optimization.

In order to test the potential of an adaptation of RMDDC to GP, and with the aim of comparing it fairly against other diversity management strategies, we included quite a large number of different strategies into a standard GP template (Algorithm 1). In this standard GP, we consider some default components, and when integrating a specific diversity management strategy, only the components involved in the strategy are altered. Thus, these components are considered as input parameters of the strategy. Note also that, when instantiating additional components, some other input parameters might be required. The default components are the following:

- **population-initialization**: It is the standard ramped half and half strategy, which combines grow and full tree generation techniques.
- **evaluation**: Each individual is evaluated with the MSE (*Mean Squared Error*) function $\frac{1}{T} \sum_i (y_i - \hat{y}_i)^2$, where T is the number of training cases, and y_i , \hat{y}_i are the expected and predicted outputs for the i -th test case of the training set. Note that individuals are only evaluated if necessary, i.e. in cases where crossover or mutation alters the tree.
- **selection**: A tournament selection is used. A pool of tsize individuals are randomly selected from the population and the fittest one is selected, with ties broken randomly.
- **crossover**: The **subtree-exchange** is used with probability p_c . In this operator, two nodes, n_a and n_b , each from a parent, are selected at random. Then, the subtrees rooted at these nodes are exchanged.
- **mutation**: The **subtree-replacement** method is applied with probability p_m . In this mutation, a node n is selected randomly and the subtree rooted in n is replaced by a new tree that is generated by the **grow** method.
- **replacement**: A generational scheme with elitism is used, meaning that the next population consists of the offspring together with the best solution from the previous population, but only if this solution is better than any of the offspring.
- **stopping-criterion**: The iterations are performed until a maximum number of objective functions are evaluated.

Algorithm 2: Replacement Phase of GP-DMD

```

Input   :  $P$  (current population),  $O$  (offspring),  $N$  (number of survivors),  $E_e$  (elapsed evaluations),
            $E_s$  (stopping evaluation),  $D_I$  (initial distance threshold)
Output : New Population (NP)
1   $C = P \cup O$ 
2   $NP = \{\text{best-individual}(C)\}$ 
3   $C = C \setminus NP$ 
4   $T = \text{minimum-required-distance}(D_I, E_s, E_e)$ 
5  while  $|NP| < N$  do
6     $(C_p, C_{np}) = \text{classify-individuals}(NP, C, T)$ 
7    if  $C_{np} \neq \emptyset$  then
8       $nds = \text{non-dominated-set}(C_{np})$                                 # Using Fitness and Size
9       $t = \text{random-sampling}(nds)$ 
10   else
11      $t = \text{farthest}(C_p, NP)$ 
12   end
13    $NP = NP \cup \{t\}$ 
14    $C = C \setminus \{t\}$ 
15 end
16 return  $NP$ 

```

286 In the case of Genetic Programming with Dynamic Management of Diversity
287 (GP-DMD), the only component that does not use the default behavior is the replacement
288 phase. The component used for the replacement is detailed in Algorithm 2. The input to the
289 replacement component is the previous population (P), the offspring (O), the number of
290 desired survivors (N), and some information that is used for the penalization approach,
291 which consists of the number of elapsed evaluations (E_e), the number of function
292 evaluations that are going to be evolved (E_s) and the initial distance threshold (D_I). The aim
293 of this component is to select N members to form the population of the next generation.
294 This replacement strategy initially joins population P and offspring O into a set of eligible
295 candidate individuals C (line 1). It initially selects the best individual in terms of fitness to
296 form part of the new population NP (line 2) and removes it from the set of candidate
297 solutions (line 3). Note that ties in fitness are broken by selecting the smallest individual. In
298 cases where both the fitness and sizes are equal, ties are broken randomly. It then calculates
299 the threshold value (T) that is used to penalize individuals (line 4) and it performs $N - 1$
300 iterations to select the remaining survivors (lines 5-15). The way to calculate the threshold
301 is detailed later. At each of these iterations, the non-selected individuals are categorized as
302 penalized and included in C_p or non-penalized and included in C_{np} (line 6). Specifically,
303 any individual in C whose distance to an already selected individual (Distance to Closest
304 Survival — DCS) is lower than T is categorized as a penalized individual, and the
305 remaining ones are categorized as non-penalized. In each iteration, a non-penalized
306 individual is selected (lines 7-9) and the penalized ones are ignored. Only when no
307 non-penalized individuals exist does the selection process considers the penalized
308 individuals (line 11). Note that the behavior described up to this point is similar to the one
309 applied in RMDDC. However, the process to select among the potential non-penalized
310 individuals differs. In the case of RMDDC and its variants, two different strategies were
311 analyzed. In the first variant [34], a multi-objective approach based on a tuple that consists
312 of fitness and diversity was applied and the selection process chooses survivors at random
313 from among the non-dominated individuals. Note that subsequently, a simpler strategy
314 based only on fitness also behaved properly [31], so the step of distinguishing between

penalized and non-penalized individuals seems to be key to proper performance. In the case of GP, both the accuracy and sizes of trees are important. Thus, in this case, a multi-objective approach based on a tuple that consists of fitness and size is taken into account and at each iteration, a non-dominated individual is selected at random on the basis of these two objectives (line 8-9). Note that in the event that no non-penalized individuals exist, similarly to RMDDC, the individual with the largest DCS value is selected (line 11). The logic behind this decision is that this situation might indicate that the diversity is too low, so selecting the most distant individual seems promising. Finally, the selected individual is added to the new population and removed from the candidates (lines 13-14).

Note that in order to fully specify our proposal, the `minimum-required-distance` functions and the method for calculating distances between individuals must be specified. The `minimum-required-distance` function is used to set the threshold that is applied to distinguish between penalized and non-penalized individuals. This function is similar to the one applied in RMDDC, i.e. it starts from an initial value D_I (a parameter of GP-DMD), which is reduced linearly. Specifically, the threshold T is set as follows: $T = D_I - \frac{D_I \times E_e}{E_s}$. In the case of the distance-like function, we considered the well-known *ed2* distance, which was previously presented. Note that in order to restrict the distance values to the range $[0, 1]$, the *ed2* distance is divided by the maximum distance attainable for a tree with a structure of the expanded tree.

4 Experimental validation

Symbolic Regression (SR) is a well-established benchmark for GP, so we selected it to validate the potential of GP-DMD. The goal of SR is to generate a predictive model as an analytic function, from a set of input/output pairs [36]. GP has been shown to yield impressive results with SR, and both the complexity and accuracy are important [28]. However, simple variants of GP that evolve a single tree have not excelled, and the current state-of-the-art algorithms are hybrid schemes that incorporate other kinds of procedures, such as multiple linear regressions and/or operators specifically devoted to SR but that suffer from exponential tree growth [1, 37]. The focus of this paper is not to develop further the state-of-the-art strategies for SR. Thus, these kinds of ad-hoc and hybrid methods are not considered in our experimental validation. Instead, the aim is to show that recently proposed design principles applied successfully in other optimization areas related to the management of diversity bring important benefits to the field of GP. Thus, this validation is focused on comparing GP-DMD against other strategies that also evolve a single tree and that provide some specific strategies to alter the degree between exploration and intensification.

4.1 Experimental setup

In order to validate the achievements of GP-DMD, our extensive comparison considers the following proposals: ϵ -LEX, SIS, DPSBR, FOCUS, GMD-GP, AFPO, KNOBELTY, ALPS and standard genetic programming (STDGP). Note that STDGP is just Algorithm 1 with its default components. All these schemes were previously presented and they were implemented by changing some of the components of Algorithm 1 to incorporate the specific changes devised in each of these strategies. Table 1 details the parameters applied in each scheme and clarifies the component that each of the methods altered with respect to

Table 1 Component modified by each method with respect to STDGP and parameterization applied

Method	Altered Component	Parameters
ε -LEX	Selection	$\varepsilon = Sum - MAD$
SIS	Selection	$\alpha = 0.01$ $PS = 3$
KNOBELTY	Selection	$k = \exp(-\lambda E_e)$ $\lambda = 1.3 \times 10^{-4}$ $\varepsilon = Sum - MAD$ $ A = 2000$ $SP = 100$
DPSBR	Crossover	$MR = 5$
FOCUS	Replacement	
AFPO	Replacement	
GMD-GP	Replacement	$MD = 3$
ALPS	Population	$AgeGap = 3$ $M = N/10$ $EL = 3$ $age_i = AgeGap \times i^2$
GP-DMD	Replacement	$D_I = 0.5$

Table 2 Common GP parameters used in all the algorithms tested

N	200
stop-criteria	76800 evaluations
tsize	3
p_m	0.2
p_c	0.6
terminals	$int \in [0, 9]$, $float \in [0, 1]$, x_i (The Variables)
functions	$+$, $-$, $*$, \sin , \cos , \exp , \ln , x_i^2 , $\frac{1}{x_i}$

358 STDGP. These parameters are similar to those put forth in the original proposals, with
 359 some minor changes to adapt each proposal to the specific problems that are addressed in
 360 this paper.

361 In order to facilitate the reproducibility, free software implementations are available for
 362 public download at the repository [27]. Instructions for replicating the experiments presented
 363 in this paper are provided. Table 2 details the common parameters used in all the algorithms.

364 Experiments were performed on the first 17 models from the benchmark proposed
 365 in [22]. Note that we performed some preliminary executions with additional models. Since
 366 similar conclusions could be drawn, we restricted the experimentation to these 17 models,
 367 which are among the most popular ones. The symbolic regression problems are listed in
 368 Table 3. For each problem, the training and validation sets were created as specified in their
 369 original papers, and in order to provide a fair comparison, the same data were considered
 370 for all the executions. Note that \mathcal{E} indicates an equidistant point sampling, and \mathcal{U} a
 371 uniform sampling. In order to facilitate future comparisons, codes to generate the training
 372 and validation sets are included in our repository.

373 Finally note that, since stochastic algorithms were considered in this study, each
 374 execution was repeated 30 times and comparisons were carried out by applying a set of
 375 statistical tests. Specifically, the following tests were applied, assuming a significance level

Table 3 Symbolic Regression problems

Name	Objective Function	Training Set	Samples	Validation Set	Samples
Koza-1	$x^4 + x^3 + x^2 + x$	$\mathcal{U}(-1, 1)$	20	$\mathcal{U}(-1, 1)$	100
Koza-2	$x^5 - 2x^3 + x$	$\mathcal{U}(-1, 1)$	20	$\mathcal{U}(-1, 1)$	100
Koza-3	$x^6 - 2x^4 + x^2$	$\mathcal{U}(-1, 1)$	20	$\mathcal{U}(-1, 1)$	100
Nguyen-1	$x^3 + x^2 + x$	$\mathcal{U}(-1, 1)$	20	$\mathcal{U}(-1, 1)$	100
Nguyen-3	$x^5 + x^4 + x^3 + x^2 + x$	$\mathcal{U}(-1, 1)$	20	$\mathcal{U}(-1, 1)$	100
Nguyen-4	$x^6 + x^5 + x^4 + x^3 + x^2 + x$	$\mathcal{U}(-1, 1)$	20	$\mathcal{U}(-1, 1)$	100
Nguyen-5	$\sin(x^2)\cos(x) - 1$	$\mathcal{U}(-1, 1)$	20	$\mathcal{U}(-1, 1)$	100
Nguyen-6	$\sin(x) + \sin(x + x^2)$	$\mathcal{U}(-1, 1)$	20	$\mathcal{U}(-1, 1)$	100
Nguyen-7	$\ln(x + 1) + \ln(x^2 + 1)$	$\mathcal{U}(0, 2)$	20	$\mathcal{U}(0, 2)$	100
Nguyen-8	\sqrt{x}	$\mathcal{U}(0, 4)$	20	$\mathcal{U}(0, 4)$	100
Nguyen-9	$\sin(x) + \sin(y^2)$	$\mathcal{U}(0, 1)$	20	$\mathcal{U}(0, 1)$	100
Nguyen-10	$2\sin(x)\cos(y)$	$\mathcal{U}(0, 1)$	20	$\mathcal{U}(0, 1)$	100
Keijzer-12	$x^4 - x^3 + \frac{y^2}{2} - y$	$\mathcal{U}(-3, 3)$	20	$\mathcal{E}(-3, 3, 0.01)$	360000
Keijzer-10	x^y	$\mathcal{U}(0, 1)$	100	$\mathcal{E}(0, 1, 0.01)$	10000
Keijzer-6	$\sum_{i=1}^x \frac{1}{i}$	$\mathcal{E}(1, 50, 1)$	50	$\mathcal{E}(1, 120, 1)$	120
Pagie-1	$\frac{1}{1+x^4} + \frac{1}{1+y^4}$	$\mathcal{E}(-5, 5, 0.4)$	25	$\mathcal{E}(-5, 5, 0.1)$	10000
Vladislavleva-4	$\frac{10}{5+(x-3)^2+(y-3)^2+(z-3)^2+(v-3)^2+(w-3)^2}$	$\mathcal{U}(0.05, 6.06)$	1024	$\mathcal{U}(-0.25, 6.35)$	5000

of 5%. First a Shapiro-Wilk test was applied to check if the values of the results followed a Gaussian distribution. If so, the Levene test was used to check for the homogeneity of the variance. When similar variances were confirmed, an ANOVA test was done; otherwise, a Welch test was performed. For non-Gaussian distributions, the non-parametric Kruskal-Wallis test was applied. The statement “algorithm A is superior than algorithm B” means that the differences between them are statistically significant and that the median obtained by A is lower than the median achieved by B. Note that the same kinds of statistical tests are used to compare fitness and sizes. The following subsections include analyses related to accuracy and sizes, as well as some studies regarding the dynamics of the population which contribute to a better understanding of the internal behavior of the set of GP strategies considered.

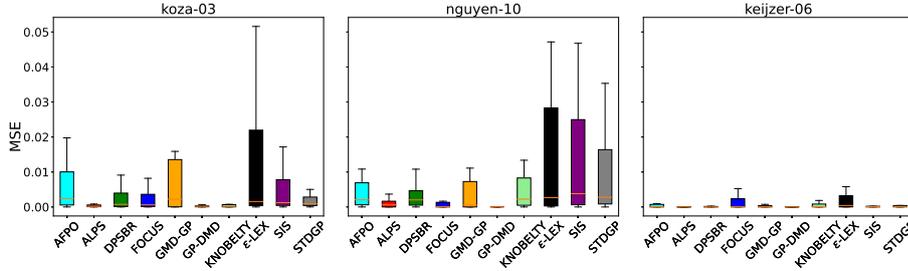
4.2 Fitness comparison

First, we compared the proposed algorithm in terms of accuracy by comparing it against all the methods implemented. Table 4 summarizes the performance of the algorithms. Specifically, it shows the number of instances where each algorithm exhibited no statistically significant difference with the algorithm that reported the lowest median among all algorithms (for detailed pairwise comparisons and the specific fitness values obtained, see the supplementary material). This is reported both for the training and validation sets. It should be noted that GP-DMD was in the group of proper algorithms in more instances than any other approach, and probably more importantly, when considering the validation set, it was in the group of best performers in 16 of the 17 instances. Thus, not only is the accuracy of GP-DMD competitive, but it is attained in quite a robust way. Note also that three of the best four approaches do modifications in the replacement phase; so, as in the case of combinatorial optimization, it seems that including diversity management in this component is more advantageous [34].

Figure 1 shows boxplots of the fitness in the validation set for 30 independent executions in three selected instances. Note that GP-DMD achieved a very low median and

Table 4 Number of instances where each method had no statistical difference with respect to the best (lowest median fitness) algorithm, for the training and validation sets

	SIS	STDGP	DPSBR	AFPO	FOCUS	ϵ -LEX	ALPS	NOBELTY	GMD-GP	GP-DMD
training	1	1	1	1	2	8	9	10	10	11
validation	3	4	4	5	4	3	14	4	8	16

Fig. 1 Boxplots of the fitness attained in 30 independent executions for the validation set in three instances**Table 5** Number of instances where each method had no statistical difference with respect to the algorithm that reported the best (smallest) size median

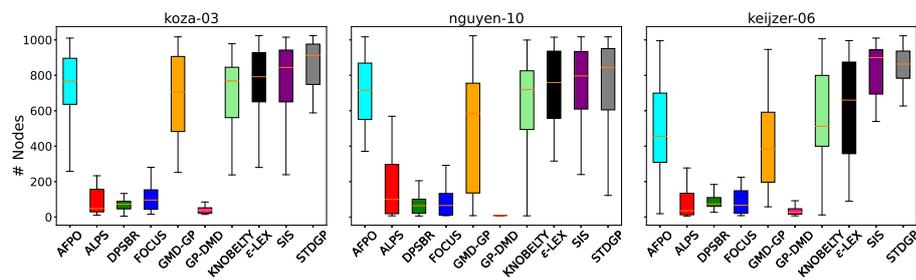
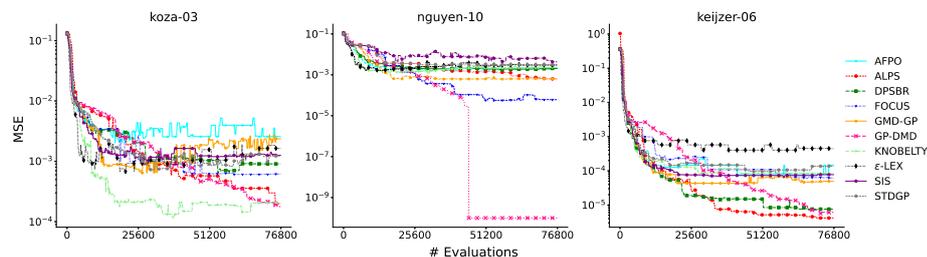
AFPO	ϵ -LEX	STDGP	FOCUS	SIS	NOBELTY	GMD-GP	DPSBR	ALPS	GP-DMD
0	0	1	1	1	1	2	3	10	17

403 the lowest variability. While other methods eventually yield good solutions, GP-DMD
 404 consistently generates proper models, so the way to explore the search space provided by
 405 GP-DMD, with a more explicit control of the search, results in a much more robust
 406 behavior.

407 4.3 Solution size

408 An important aspect in GP is the concept of bloat, i.e. the uncontrolled growth of code
 409 without a significant performance improvement [16]. Avoiding this uncontrolled growth is
 410 important mainly for two reasons. First, concise final trees are usually desired for
 411 numerous reasons, such as reduced computational complexity, increased generalization and
 412 easier examination of the structure. Second, in the presence of bloat, the structural diversity
 413 measures become an unreal measure of the difference between individuals [18]. Thus,
 414 diversity management strategies, and especially those based on structural diversity, should
 415 take size into account. This section analyzes the sizes of the final solutions attained by each
 416 algorithm tested.

417 Table 5 presents a summary of the statistical tests when applied to compare tree sizes
 418 (number of nodes). It shows the number of instances where each algorithm exhibited no
 419 statistically significant differences with respect to the algorithm that reported the lowest
 420 median size among all algorithms (for detailed pairwise comparisons and the sizes obtained,
 421 see the supplementary material).

Fig. 2 Boxplots of the sizes of the models generated in the 30 independent executions**Fig. 3** Trend of the median for 30 independent executions of the fitness in the validation set (logarithmic scale)

The clear winners in terms of size are ALPS and GP-DMD, with a clear advantage in favor of GP-DMD. In fact, no algorithm was able to attain lower trees (statistically significant) than GP-DMD in any of the instances tested.

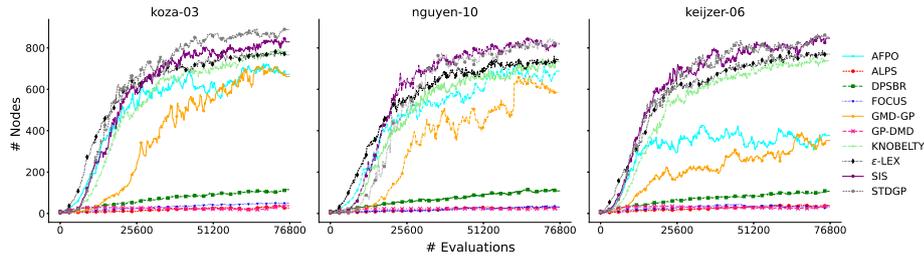
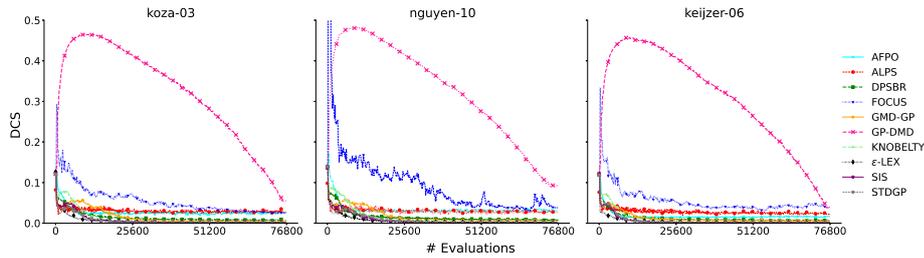
Figure 2 shows boxplots of the solution sizes for 30 independent executions in three selected instances. It shows a clear separation among ALPS, GP-DMD, DPSBR, and FOCUS and the remaining methods. However, the additional benefits contributed by GP-DMD in terms of the model's complexity are also clear.

These results show that the proposed method has a positive and significant impact on the size of the solutions, promoting robustness and generalization, as also illustrated with the analyses of the performance in the validation set.

4.4 Population Dynamics

In order to shed some light on the reasons behind the good performance of our proposed method, a discussion of some of the properties related to the dynamics of the population is presented. To this end, below is a set of graphs with the trend, through the optimization process, of several measurements for three test cases. The behavior observed in the remaining 14 cases is quite similar.

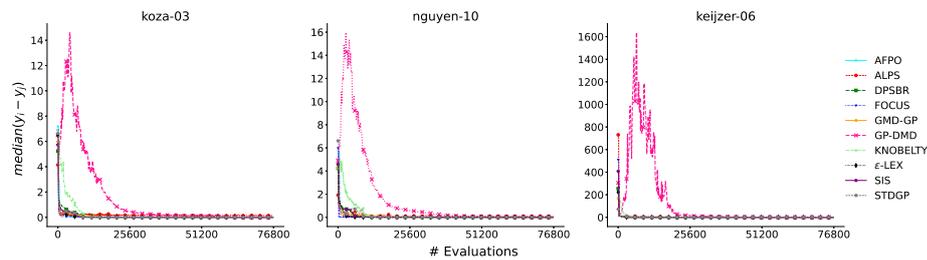
Figure 3 shows the median among executions of the fitness for the validation set. Note that GP-DMD yields improvements in a continuous and gradual manner, which indicates that the algorithm does not suffer from premature convergence. This contrasts with the remaining methods, where the improvements are not very significant for a large period of the execution.

Fig. 4 Trend in the median for 30 independent executions of the mean population's tree sizes**Fig. 5** Trend in the median for 30 independent executions of the mean ed2 distance to the closest individual in the population

443 Figure 4 depicts the median among executions of the mean size (number of nodes) of the
 444 solutions in the population. It is clear that ALPS, GP-DMD, DPSBR, and FOCUS exhibit a
 445 slow but steady increase in the sizes, whereas for the remaining methods, there is initially
 446 a fast increase in the population size that never shrinks, meaning they are searching much
 447 more complex models. Note also that those schemes that maintain lower sizes are faster,
 448 meaning larger populations or executions could be used to potentially improve the quality
 449 of solutions further.

450 Figure 5 shows the median among executions of the mean ed2 distance to the closest
 451 individual in the population. As expected, in the case of GP-DMD there is quite a linear
 452 decrease in this measurement. It is also important to point out that the diversity resulting
 453 from the population initialization method is not very high, and GP-DMD is capable of
 454 promoting the diversity to the desired value despite the initial diversity value. This behavior
 455 is quite different to the ones exhibited in the remaining methods, where in most cases, a
 456 relatively constant amount of diversity is maintained for a large period of the execution.
 457 This means that GP-DMD is in fact moving from exploration to exploitation, which is the
 458 main aim behind the design principle studied in this paper.

459 Finally, in order to analyze the trend in the behavioral diversity, Figure 6 shows the
 460 median among executions of the median pair-wise fitness differences appearing in the
 461 population. While GP-DMD does not explicitly control the behavioral diversity, it is the
 462 only method that initially attains a large increase in this metric. Note also that the reduction
 463 in GP-DMD is gradual, so the explicit management of structural diversity indirectly
 464 manages the behavioral diversity in a proper way in the case of GP-DMD. This contrasts
 465 with the remaining methods, where a much faster convergence of behavioral diversity
 466 appears.

Fig. 6 Trend in the median for 30 independent executions of the median pair-wise fitness distances in the validation

These trends show that the search features are quite different in the proposal put forth in this paper than in other methods. These trends are quite similar to the ones observed in the field of combinatorial optimization, meaning that these interesting features could be translated to the realm of GP and that in this case, it also provides significant benefits to the results. Specifically, there are important benefits in terms of robustness, which is also one of advantages found in complex combinatorial optimization problems.

5 Conclusions and Future Work

The proper balance between exploration and exploitation is one of the keys to designing effective Evolutionary Algorithms. Recently, a design principle based on relating the amount of diversity maintained in the population to the elapsed period of execution and stopping criterion has yielded significant benefits in the field of combinatorial optimization. This work studies the application of this design principle in the area of GP; specifically, this paper presents a novel GP variant called GP-DMD. GP-DMD manages diversity explicitly through a novel replacement strategy that incorporates the previously mentioned design principle and at the same time favors fit and simple trees by applying multi-objective concepts.

The novel proposal was compared to a diverse set of current and well-established algorithms, including several diversity-aware techniques. Specifically, the validation is carried out using the symbolic regression task. The experimental validation shows that the approach presented allows for the generation of high-quality solutions with impressively small sizes, significantly improving the algorithm's robustness. The dynamics of the population in terms of size, fitness and diversity shows the remarkable differences in GP-DMD when compared to other related algorithms. Moreover, the findings are somewhat similar to those achieved in the field of combinatorial optimization, meaning that this important advance recently proposed in that area could be successfully transferred to GP.

One of the weaknesses of our proposal is that it requires setting a stopping criterion based on time or evaluations in order to promote the gradual shift from exploration to exploitation. In practical terms, using a stopping criterion related to quality is also promising, so in a future work, we plan to analyze the application of the ideas explored in this paper but in a way that is not incompatible with setting a stopping criterion based on quality. We would also like to apply GP-DMD to other kinds of applications.

498 **Acknowledgements** Authors acknowledge the financial support from CONACyT through the “Ciencia
499 Básica” project no. 285599 and the support from “Laboratorio de Supercómputo del Bajío” through the
500 project 300832 from CONACyT.

501 Conflict of interest

502 The authors declare that they have no conflict of interest.

503 References

- 504 1. Arnaldo, I., Krawiec, K., O’Reilly, U.M.: Multiple regression genetic programming. In: Proceedings
505 of the 2014 Annual Conference on Genetic and Evolutionary Computation, GECCO ’14, p. 879–886.
506 Association for Computing Machinery, New York, NY, USA (2014). DOI 10.1145/2576768.2598291
- 507 2. Azaria, I., Elyasaf, A., Sipper, M.: Evolving artificial general intelligence for video game controllers. In:
508 R. Riolo, B. Worzel, B. Goldman, B. Tozier (eds.) Genetic Programming Theory and Practice XIV, pp.
509 53–63. Springer International Publishing, Cham (2018). DOI 10.1007/978-3-319-97088-2_4
- 510 3. Burke, E., Gustafson, S., Kendall, G., Krasnogor, N.: Advanced population diversity measures in genetic
511 programming. In: J.J.M. Guervós, P. Adamidis, H.G. Beyer, H.P. Schwefel, J.L. Fernández-Villacañas
512 (eds.) Parallel Problem Solving from Nature — PPSN VII, pp. 341–350. Springer Berlin Heidelberg,
513 Berlin, Heidelberg (2002). DOI 10.1007/3-540-45712-7_33
- 514 4. Burke, E.K., Gustafson, S., Kendall, G.: Diversity in genetic programming: an analysis of measures and
515 correlation with fitness. *IEEE Transactions on Evolutionary Computation* **8**(1), 47–62 (2004). DOI
516 10.1109/TEVC.2003.819263
- 517 5. Burks, A.R., Punch, W.F.: An efficient structural diversity technique for genetic programming. In:
518 Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation, GECCO
519 ’15, p. 991–998. Association for Computing Machinery, New York, NY, USA (2015). DOI
520 10.1145/2739480.2754649
- 521 6. Burks, A.R., Punch, W.F.: An investigation of hybrid structural and behavioral diversity methods in
522 genetic programming. In: R. Riolo, B. Worzel, B. Goldman, B. Tozier (eds.) Genetic Programming
523 Theory and Practice XIV, pp. 19–34. Springer International Publishing, Cham (2018). DOI 10.1007/978-
524 3-319-97088-2_2
- 525 7. Cao, B., Jiang, Z.: Increasing diversity and controlling bloat in linear genetic programming. In: 2016
526 3rd International Conference on Information Science and Control Engineering (ICISCE), pp. 414–419
527 (2016). DOI 10.1109/ICISCE.2016.97
- 528 8. Chacón, J., Segura, C.: Analysis and enhancement of simulated binary crossover. In: 2018 IEEE
529 Congress on Evolutionary Computation (CEC), pp. 1–8 (2018). DOI 10.1109/CEC.2018.8477746
- 530 9. Chacón Castillo, J., Segura, C.: Differential evolution with enhanced diversity maintenance.
531 *Optimization Letters* **14**(6), 1471–1490 (2020). DOI 10.1007/s11590-019-01454-5
- 532 10. Day, P., Nandi, A.K.: Binary string fitness characterization and comparative partner selection in genetic
533 programming. *IEEE Transactions on Evolutionary Computation* **12**(6), 724–735 (2008). DOI
534 10.1109/TEVC.2008.917201
- 535 11. Folino, G., Pizzuti, C., Spezzano, G., Vanneschi, L., Tomassini, M.: Diversity analysis in cellular and
536 multipopulation genetic programming. In: The 2003 Congress on Evolutionary Computation, 2003.
537 CEC ’03., vol. 1, pp. 305–311 Vol.1 (2003). DOI 10.1109/CEC.2003.1299589
- 538 12. Galván, E., Schoenauer, M.: Promoting semantic diversity in multi-objective genetic programming. In:
539 Proceedings of the Genetic and Evolutionary Computation Conference, GECCO ’19, p. 1021–1029.
540 Association for Computing Machinery, New York, NY, USA (2019). DOI 10.1145/3321707.3321854
- 541 13. Galván-López, E., Cody-Kenny, B., Trujillo, L., Kattan, A.: Using semantics in the selection mechanism
542 in genetic programming: A simple method for promoting semantic diversity. In: 2013 IEEE Congress
543 on Evolutionary Computation, pp. 2972–2979 (2013). DOI 10.1109/CEC.2013.6557931
- 544 14. Grefenstette, J.: Genetic algorithms for changing environments. In: Parallel Problem Solving from
545 Nature 2, pp. 137–144. Elsevier (1992)
- 546 15. Hien, N.T., Hoai, N.X.: A brief overview of population diversity measures in genetic programming. In:
547 Proc. 3rd Asian-Pacific Workshop on Genetic Programming, Hanoi, Vietnam, pp. 128–139. Citeseer
548 (2006)
- 549 16. de Jong, E.D., Watson, R.A., Pollack, J.B.: Reducing bloat and promoting diversity using multi-objective
550 methods. In: Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation,
551 GECCO’01, p. 11–18. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001)

17. Keijzer, M.: Efficiently Representing Populations in Genetic Programming, p. 259–278. MIT Press, Cambridge, MA, USA (1996) 552
18. Kelly Johnathan Hemberg Erick, O.U.M.: Improving genetic programming with novel exploration - exploitation control. In: European Conference on Genetic Programming, vol. 11451, pp. 64–80 (2019). DOI https://doi.org/10.1007/978-3-030-16670-0_5 554
19. Koza, J.R., Koza, J.R.: Genetic programming: on the programming of computers by means of natural selection, vol. 1. MIT press (1992) 555
20. Krawiec, K.: Behavioral program synthesis with genetic programming, vol. 618. Springer (2016) 558
21. La Cava, W., Spector, L., Danai, K.: Epsilon-lexicase selection for regression. In: Proceedings of the Genetic and Evolutionary Computation Conference 2016, GECCO '16, p. 741–748. Association for Computing Machinery, New York, NY, USA (2016). DOI 10.1145/2908812.2908898 560
22. McDermott, J., White, D.R., Luke, S., Manzoni, L., Castelli, M., Vanneschi, L., Jaskowski, W., Krawiec, K., Harper, R., De Jong, K., O'Reilly, U.M.: Genetic programming needs better benchmarks. In: Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation, GECCO '12, pp. 791–798. ACM, New York, NY, USA (2012). DOI 10.1145/2330163.2330273 561
23. McPhee, N.F., Hopper, N.J.: Analysis of genetic diversity through population history. In: Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation - Volume 2, GECCO'99, p. 1112–1120. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1999) 562
24. Muhammad Waqar Aslam Zhechen Zhu, A.K.N.: Diverse partner selection with brood recombination in genetic programming. *Applied Soft Computing* **67**, 558–566 (2018). DOI <https://doi.org/10.1016/j.asoc.2018.03.035> 563
25. Nag, K., Pal, N.R.: Genetic programming for classification and feature selection. In: Evolutionary and Swarm Intelligence Algorithms, pp. 119–141. Springer International Publishing, Cham (2019). DOI 10.1007/978-3-319-91341-4_7 564
26. Nguyen, Q.U., Nguyen, X.H., O'Neill, M., Agapitos, A.: An investigation of fitness sharing with semantic and syntactic distance metrics. In: European Conference on Genetic Programming, vol. 7244, pp. 109–120. Springer (2012). DOI https://doi.org/10.1007/978-3-642-29139-5_10 565
27. Nieto-Fuentes, R., Segura, C.: Genetic Programming framework in Crystal (2021). URL <https://drive.google.com/file/d/1g-JLWqyGMZDmZuBhh-Xx6ozNfT64NjS0/view?usp=sharing>. Available after acceptance at <https://gitlab.com/nifr91/genetic-programming> 566
28. Orzechowski, P., La Cava, W., Moore, J.H.: Where are we now? a large benchmark study of recent symbolic regression methods. In: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '18, p. 1183–1190. Association for Computing Machinery, New York, NY, USA (2018). DOI 10.1145/3205455.3205539 567
29. P, R.J.: Entropy-driven adaptive representation **9** (1995) 568
30. Poli, R., Koza, J.: Genetic programming. In: E.K. Burke, G. Kendall (eds.) *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, pp. 143–185. Springer US, Boston, MA (2014). DOI 10.1007/978-1-4614-6940-7_6 587
31. Romero Ruiz, E., Segura, C.: Memetic algorithm with hungarian matching based crossover and diversity preservation. *Computación y Sistemas* **22**(2) (2018). DOI 10.13053/cys-22-2-2951 588
32. Schmidt, M.D., Lipson, H.: Age-fitness pareto optimization. In: Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, GECCO '10, p. 543–544. Association for Computing Machinery, New York, NY, USA (2010). DOI 10.1145/1830483.1830584 589
33. Segura, C., Aguirre, A.H., Peña, S.I.V., Rionda, S.B.: The importance of proper diversity management in evolutionary algorithms for combinatorial optimization. In: O. Schütze, L. Trujillo, P. Legrand, Y. Maldonado (eds.) *NEO 2015: Results of the Numerical and Evolutionary Optimization Workshop NEO 2015 held at September 23-25 2015 in Tijuana, Mexico*, pp. 121–148. Springer International Publishing, Cham (2017). DOI 10.1007/978-3-319-44003-3_6 590
34. Segura, C., Hernández-Aguirre, A., Luna, F., Alba, E.: Improving diversity in evolutionary algorithms: New best solutions for frequency assignment. *IEEE Transactions on Evolutionary Computation* **21**(4), 539–553 (2017). DOI 10.1109/TEVC.2016.2641477 591
35. Črepinšek, M., Liu, S.H., Mernik, M.: Exploration and exploitation in evolutionary algorithms: A survey. *ACM Comput. Surv.* **45**(3) (2013). DOI 10.1145/2480741.2480752 592
36. White, D.R., McDermott, J., Castelli, M., Manzoni, L., Goldman, B.W., Kronberger, G., Jaskowski, W., O'Reilly, U.M., Luke, S.: Better gp benchmarks: community survey results and proposals. *Genetic Programming and Evolvable Machines* **14**(1), 3–29 (2013). DOI 10.1007/s10710-012-9177-2 593
37. Žegklitz, J., Poš'ik, P.: Benchmarking state-of-the-art symbolic regression alrogirhms. *Genetic Programming and Evolvable Machines* pp. 1–29 (2020). DOI <https://doi.org/10.1007/s10710-020-09387-0> 594

Figures

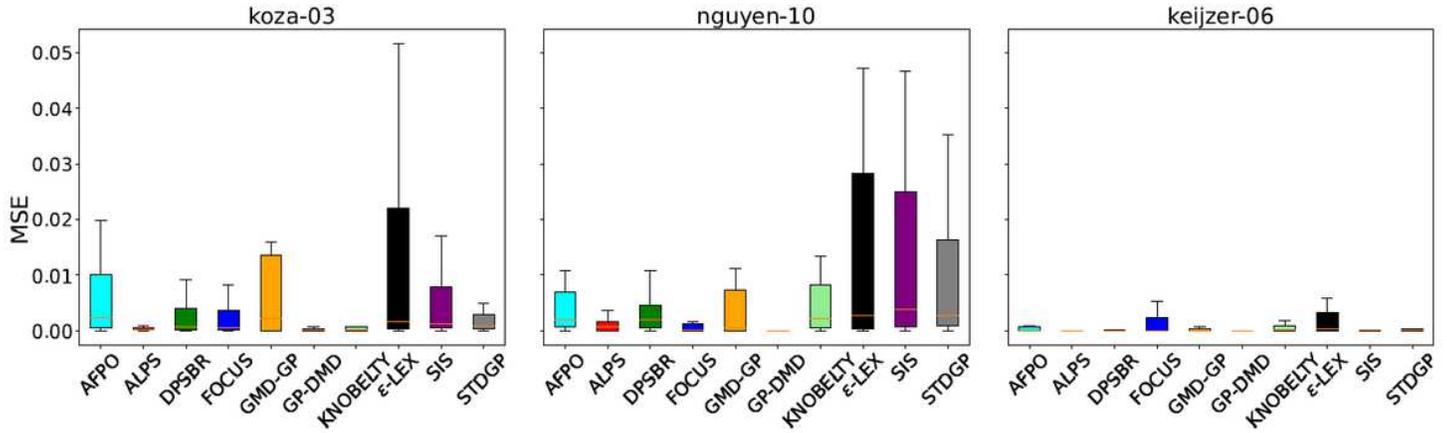


Figure 1

Boxplots of the fitness attained in 30 independent executions for the validation set in three instances

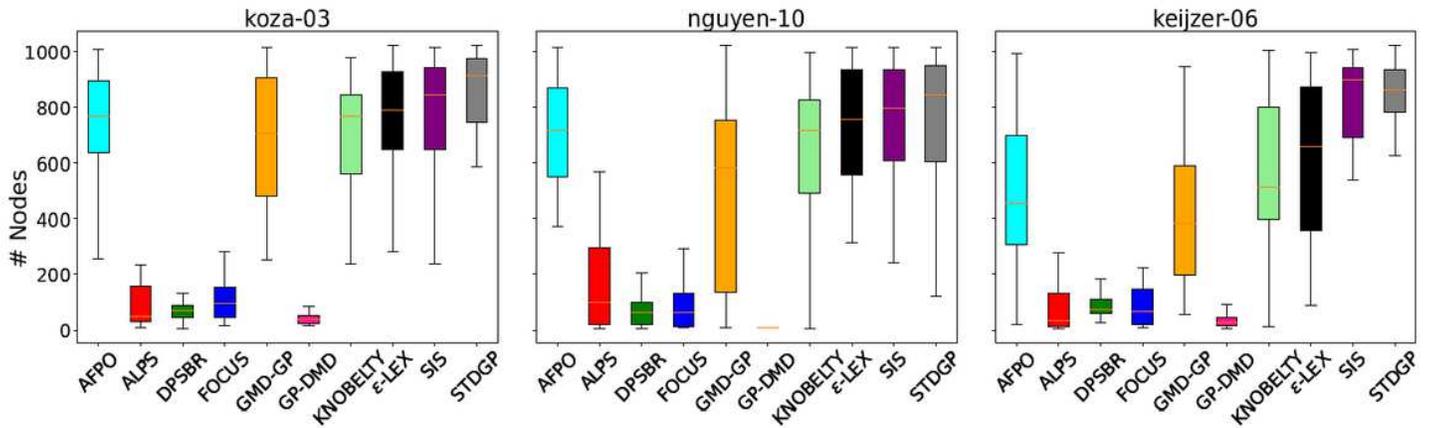


Figure 2

Boxplots of the sizes of the models generated in the 30 independent executions

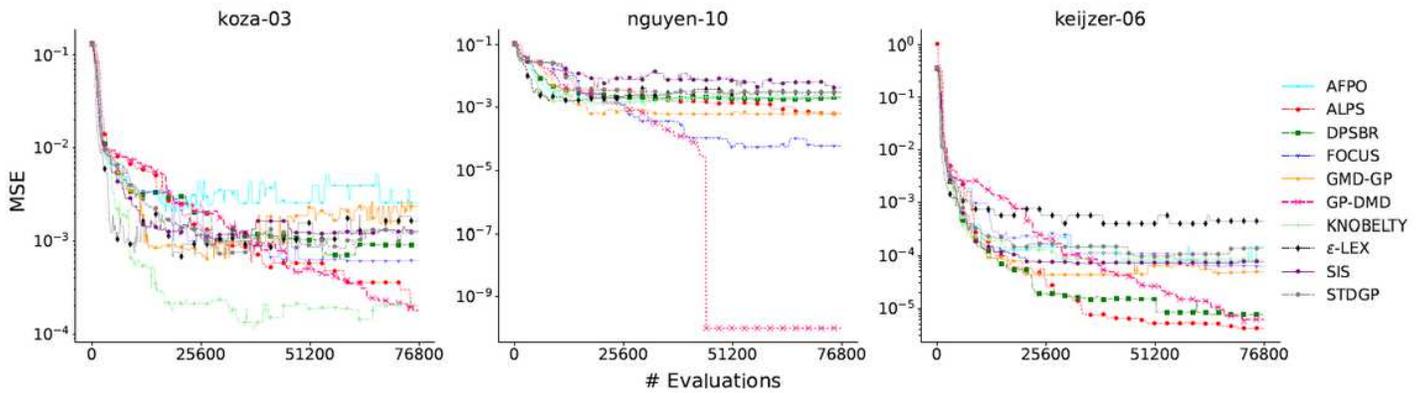


Figure 3

Trend of the median for 30 independent executions of the fitness in the validation set (logarithmic scale)

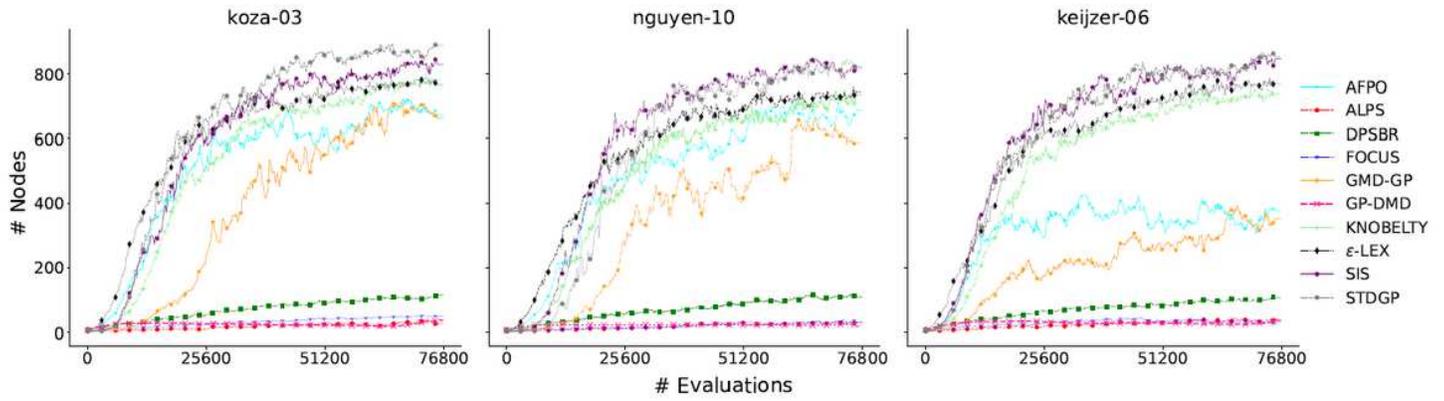


Figure 4

Trend in the median for 30 independent executions of the mean population's tree sizes

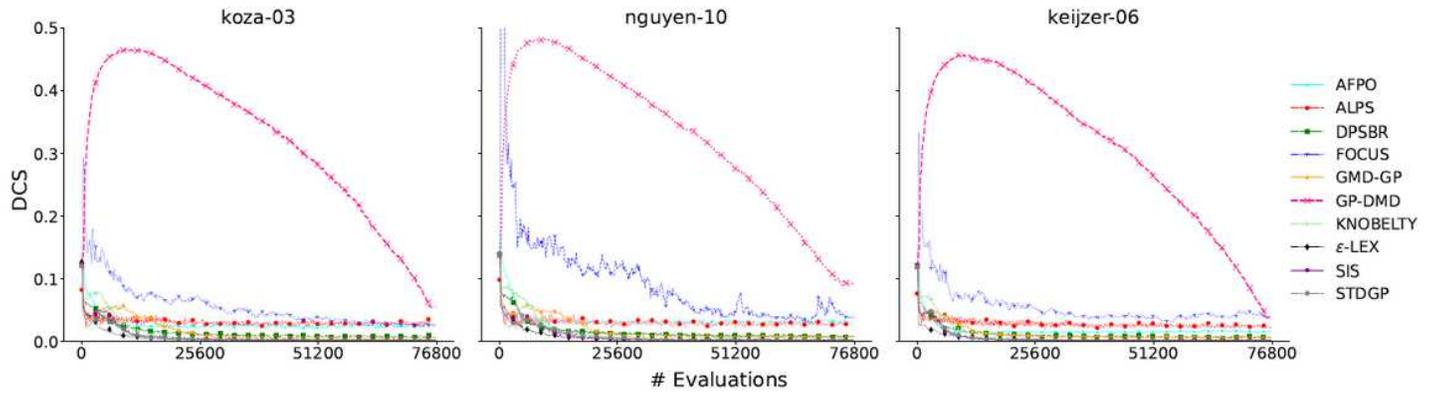


Figure 5

Trend in the median for 30 independent executions of the mean ed2 distance to the closest individual in the population

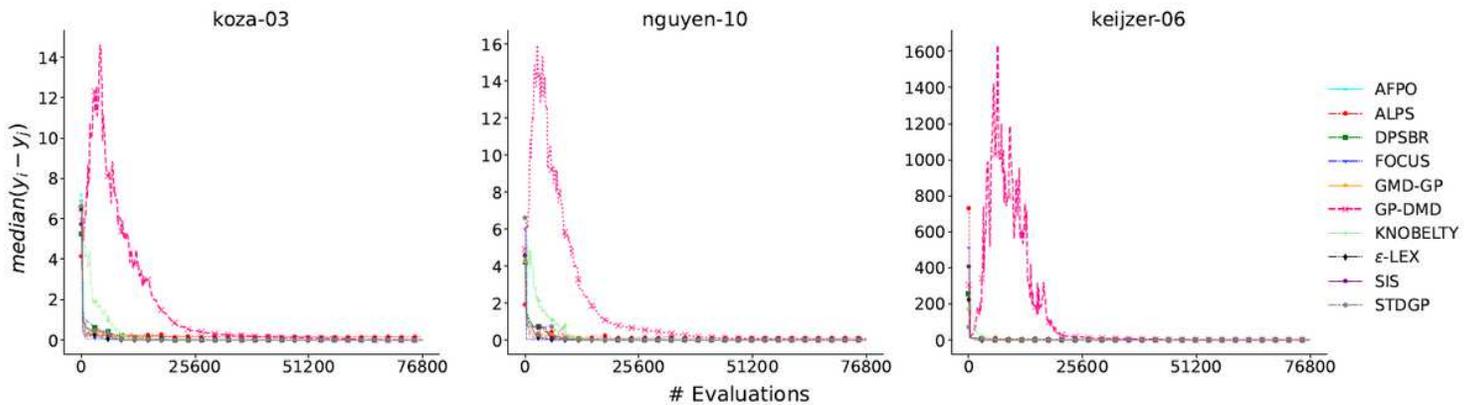


Figure 6

Trend in the median for 30 independent executions of the median pair-wise fitness distances in the validation

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [supplementarymaterial.pdf](#)