Dynamic GP Fitness Cases in Static and Dynamic Optimisation Problems

Edgar Galván-López[1,3], Lucia Vázquez-Mendoza[2], Marc Schoenauer[3], Leonardo Trujillo[4]
[1]Department of Computer Science, National University of Ireland Maynooth, Ireland,
[2]School of Social Sciences and Philosophy, Trinity College Dublin, Ireland
[3]TAU, INRIA and LRI, CNRS & U. Paris-Sud, Université Paris-Saclay, France
[4]Posgrado en Ciencias de la Ingeniería, Instituto Tecnológico de Tijuana, México
edgar.galvan@nuim.ie,lucyvaz@gmail.com,marc.schoenauer@inria.fr,leonardo.trujillo@tectijuana.edu.mx

ABSTRACT

In Genetic Programming (GP), the fitness of individuals is normally computed by using a set of fitness cases (FCs). Research on the use of FCs in GP has primarily focused on how to reduce the size of these sets to, for instance, reduce the fitness evaluation time. However, often, only a small set of FCs is available and there is no need to reduce it. In this work, we are interested in using the whole FCs set, but rather than adopting the commonly used GP approach of presenting the entire set of FCs to the system from the beginning of the search, referred as static FCs, we allow the GP system to build it over time, named as dynamic FCs, to make the search more amenable. Moreover, to the best of our knowledge, there is no study on the use/impact of FCs in Dynamic Optimisation Problems (DOPs). To this end, we also propose the Kendall Tau Distance (KTD) approach, which quantifies pairwise dissimilarities among two lists of fitness values. KTD aims to capture the degree of a change in DOPs and we use this to promote diversity, which has constantly reported to be beneficial in a dynamic setting. Results on eight symbolic regression functions indicate that both approaches are highly beneficial in GP.

ACM Reference format:
Edgar Galván-López[1,3], Lucia Vázquez-Mendoza[2], Marc Schoenauer[3], Leonardo Trujillo[4]. 2017. Dynamic GP Fitness Cases in
Static and Dynamic Optimisation Problems. In *Proceedings of GECCO '17 Companion, Berlin, Germany, July 15-19, 2017,* 2 pages.
DOI: http://dx.doi.org/10.1145/3067695.3076055

1 INTRODUCTION

Normally, the fitness of Genetic Programming (GP) candidate solutions is obtained by using a set of fitness cases (FCs): a fitness case is an input/output pair and the fitness of a GP individual is measured on how well it matches the output(s) from input(s) (raw fitness).

Research on the use of FCs has primarily focused on how to reduce the number of these cases when running a GP system given that this is a major element that affects speed. There are, however, some problems where only a few FCs are available for the GP system to work with.

GECCO '17 Companion, Berlin, Germany

DOI: http://dx.doi.org/10.1145/3067695.3076055

Table 1: Symbolic regression benchmarks problems.

f	Objective function
f_1	$x^3 + x^2 + \alpha x$
f_2	$x^4 + x^3 + x^2 + \alpha x$
f_3	$x^5 + x^4 + x^3 + x^2 + \alpha x$
f_4	$x^6 + x^5 + x^4 + x^3 + x^2 + \alpha x$
f_5	$\sin(x^2)\cos(\alpha) - 1$
f_6	$\sin(\alpha x) + \sin(x + x^2)$
f_7	$\log(\alpha x + 1) + \log(x^2 + 1)$
f_8	$\mathrm{sqrt}(\alpha x)$

In this work, rather than using only a subset of FCs from the entire set, we are interested in using them all in a way to make the search more robust. To do so, we propose an approach called *dynamic fitness cases*, wherein FCs are built by aggregation over generations instead of using the commonly adopted approach of using them all from the beginning of the search, called in this work static FCs.

2 DYNAMIC FITNESS CASES

To make the GP search more amenable, we build the FCs over time. More specifically, at the beginning of an evolutionary run or just after a change has occurred (for the dynamic setting), we use a subset of FCs, $C_{g=0}$ which is chosen from all the FCs C^N of size N, $C_{g=0} \subset C^N, |C_{g=0}| = k$, where k is a constant and $k < N$.

After a few i generations another k FCs of the C^N FCs are added to $C_{g=0}$, $C_{g=0} \cup C_{g=i}, C_{g=0} \cap C_{g=i} = \{\}$

We continue this process until all the FCs have been used. Thus, the complete sequence of FCs is build as follows, $C_{g=0} \cup C_{g=i} \cup \cdots \cup C_{g=M} = C^N$, where M is a constant and $M < K$, where K is either the maximum number of generations or the number of generations that are necessary for a change to take place (for the dynamic scenario). By defining the latter, we guarantee that the GP system accounts for all the FCs before a change takes place and it has all the necessary elements to, potentially, find the optimum solution. The values of the variables are defined at the end of next section and discussed in Section 4.

3 EXPERIMENTAL SETUP

To test our approach, we use eight symbolic regression functions of various difficulties, shown in Table 1. The fitness function is computed as the sum of absolute errors of the Euclidean distance to the output vector of the target uni-variate function queried on 20 inputs in the range $[-1, 1]$ (equally drawn). A solution is regarded

Table 2: Success rate and avg. of best fitness using either SFCs or DFCs in the face of a static scenario. Higher is better.

f	Success Rate		Avg. Best Fitness	
	SFC	DFC	SFC	DFC
f_1	92.0%	100.0%	0.9371	1.0000
f_2	54.0%	88.0%	0.6656	0.9969
f_3	18.0%	70.0%	0.4501	0.9915
f_4	4.0%	72.0%	0.3280	0.9895
f_5	0.0%	60.0%	0.4580	0.9896
f_6	0.0%	64.0%	0.3438	0.9893
f_7	0.0%	36.0%	0.4988	0.9739
f_8	0.0%	16.0%	0.3068	0.9665

Table 3: Success rate using either SFCs or DFCs in the face of a change. Higher is better.

f	Smooth Change		Random Change		Abrupt Change	
	SFCs	DFCs	SFCs	DFCs	SFCs	DFCs
f_1	21.5%	25.0%	24.5%	33.5%	21.5%	29.0%
f_2	10.0%	92.5%	11.0%	90.0%	10.5%	91.5%
f_3	2.5%	79.0%	4.5%	89.0%	2.5%	82.0%
f_4	0.5%	87.0%	1.5%	86.0%	0.5%	90.5%
f_5	0.0%	49.0%	0.0%	60.5%	0.0%	57.0%
f_6	0.0%	68.0%	0.5%	78.5%	0.0%	77.5%
f_7	0.0%	76.0%	0.0%	78.0%	2.5%	60.0%
f_8	0.0%	53.0%	0.5%	64.0%	1.0%	61.0%

as correct when its fitness is less than a threshold set at 0.01. The function set is $F = \{+, -, *, /\}$, where / is protected division.

Furthermore, we use a static and a dynamic setting to test our approach. We define three different type of changes for the latter: we use α as a variable (see Table 1) that can be tuned to achieve this along with a constant L, set at 50, that denotes when α changes to simulate a change (in this work, the maximum number of generations is set at 200, hence only three values for α are required for a dynamic setting, as defined next). For the static scenario, $\alpha = 1$. For the dynamic setting, we define a smooth, an 'abrupt' and a random change, where $\alpha = \{0.9, 0.8, 0.7\}$, $\alpha = \{0.1, 0.9, 0.1\}$, and finally, α is set with a random value between 0 and 1 every L generations, respectively.

For comparative purposes, we use a static fitness case-scenario and our proposed dynamic fitness case-approach, where all the cases are presented to the system at the beginning of the search as commonly adopted in the GP community and where the cases are built over time, respectively.

The experiments were conducted using a generational approach, using 800 individuals, 200 generations, standard crossover (80%), subtree mutation (20%), tournament size of 7, ramped half-and-half initialisation method with initial and final depth set at 2 and 5, respectively. To control bloat we set 1200 nodes or a maximum depth of 8, whatever occurs first. For the dynamic fitness cases variables defined in Sect. 2, we set $k = 1, i = 2, M = 39$. To obtain meaningful results, we performed an extensive empirical experimentation (50 * 3 * 8 runs, plus 50 runs for each static and dynamic setting; 1300 independent runs in total)[1].

4 RESULTS AND DISCUSSION

Performance on a Static Setting

We compare the results of our proposed approach, dynamic fitness cases (DFCs), against the widely adopted mechanism of using all the FCs at the beginning of the search, denominated in this work as static FCs (SFCs) when no changes are presented in the GP system.

Table 2 shows the success rate, defined as the number of times that the GP system was able to find the global optimum solution and the average of the best fitness at the end of each independent run.

It is clear to see that DFCs achieves good results in terms of finding the global optimum solution. The traditional SFCs has a good performance only on the relatively easy f_1 and its performance decreases significantly with the rest of the functions used in this work, where SFCs is not able to find a single optimum solution for functions f_5 - f_8 in any of the independent runs. Our proposed DFCs, on the other hand, achieves better results e.g., 60%, 64%, 36% and 16% for functions f_5 - f_8, respectively.

The results on the average of the best individuals' fitness values at the end of each run, are aligned to the performance achieved by SFCs and FDCs. These results are all statistically significant, Wilcoxon Test set at 95% level of significance.

Performance on a Dynamic Setting

Now, let us focus our attention on the presence of a dynamic scenario, where we encourage structural diversity by replacing part of the population based on the Kendall Tau Distance [1] whenever a change takes place (every 50 gens).

The results using either SFCs or DFCs are shown in Table 3. These are similar to those discussed above: the SFCs approach has a poor performance: less than 3.0%, for functions f_5 - f_8, regardless of the type of change used (see Sect. 3 for a description on the type of a change). These results are significantly better when using the proposed DFCs. For example, the proposed approach achieves more than 48%, 67%, 59% and 52% for functions f_5, f_6, f_7 and f_8, respectively, regardless of the change presented to the system. The average of best fitness values (not shown due to space constraints) just before a change takes place, are all statistically significant (Wilcoxon Test set at 95% level of significance).

5 CONCLUSIONS

We propose a DFCs approach, wherein the FCs are built by aggregation over time. We showed that the proposed DFCs approach has much better performance compared to the SFCs in both static and dynamic settings.

ACKNOWLEDGEMENTS

EGL's research is funded by an ELEVATE Fellowship, the Irish Research Council's Career Develop. Fellowship co-funded by Marie Curie Actions. EGL would like to thank the TAU group at INRIA Saclay for hosting him during the outgoing phase of the fellowship.

REFERENCES

[1] E. Galván-López and O. A. ElHara. Using fitness comparison disagreements as a metric for promoting diversity in dynamic optimisation problems. In *2016 IEEE Symposium Series on Computational Intelligence, SSCI 2016, Athens, Greece, December 6-9, 2016*, pages 1–8. IEEE, 2016.

[1] 50 independent runs, 3 types of changes (a smooth, a random, an abrupt change), 8 problems.