

An Effective Diversity Promotion Mechanism in Grammatical Evolution

Eric Medvet
DIA - Università di Trieste
Trieste, Italy
emedvet@units.it

Alberto Bartoli
DIA - Università di Trieste
Trieste, Italy
bartoli.alberto@units.it

Giovanni Squillero
Politecnico di Torino
Torino, Italy
giovanni.squillero@polito.it

ABSTRACT

Grammatical Evolution is an Evolutionary Algorithm which can evolve programs in any language described by a context-free grammar. A sequence of bits (the genotype) is transformed into a string of the language (the phenotype) by means of a mapping function, and eventually into a fitness value. Unfortunately, the flexibility brought by the mapping is also likely to introduce non-locality phenomena, reduce diversity, and consequently hamper the effectiveness of the algorithm. In this paper, we propose a novel technique for promoting diversity, able to operate on three different levels: genotype, phenotype, and fitness. The technique is quite general, independent both from the specific problem being tackled and from other components of the evolutionary algorithm, such as genotype-phenotype mapping, selection criteria, and genetic operators. We experimentally demonstrate its efficacy in a wide range of conditions and from different points of view. The results also confirm the preponderant importance of the phenotype-level analyses in diversity promotion.

CCS CONCEPTS

•Computing methodologies → Genetic programming;

KEYWORDS

Diversity, Representation, Locality, Performance

ACM Reference format:

Eric Medvet, Alberto Bartoli, and Giovanni Squillero. 2017. An Effective Diversity Promotion Mechanism in Grammatical Evolution. In *Proceedings of The Genetic and Evolutionary Computation Conference, Berlin, Germany, July 15–19, 2017 (GECCO’17)*, 2 pages.
DOI: <http://dx.doi.org/10.1145/3067695.3076057>

1 INTRODUCTION

Grammatical Evolution (GE) is an Evolutionary Algorithm proposed by Ryan, Collins, and O’Neill in 1998 [5]. It encodes individuals into genomes as unstructured, variable-length sequences of bits grouped in *codons*, eventually interpreted in the context of a user-supplied grammar. Both advantages and disadvantages of such an approach are apparent: changing the base grammar allows to

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’17, Berlin, Germany

© 2017 ACM. 978-1-4503-4939-0/17/07...\$15.00
DOI: <http://dx.doi.org/10.1145/3067695.3076057>

exploit the very same EA for virtually any possible problem without modifications; on the other hand, the rather indirect mapping between genotypes and phenotypes is likely to impair the evolution process. Not surprisingly, in the past 25 years literature reports several successful application of GE [1], together with scholarly articles that scrutinize its peculiar evolutionary processes, such as [2, 4, 9, 10].

Also related to the mapping procedure, the inherent structure of GE clearly separates three levels: *genotype*, the sequence of bits; *phenotype*, the string resulting from the chosen productions; and *fitness*, the evaluation of the phenotype in the context of the current problem. As the distinction is known to play a significant role when devising and evaluating diversity promotion mechanisms, GE offers an unparalleled testbed in such analyses.

The *lack of diversity* frequently limits the effectiveness of evolutionary algorithms [7]. Scholars agree that the maintaining of a high level of diversity within the population is important and proposed many *niche methods* for promoting diversity [3, 6]. Most of them work at the level of genotypes, as in this level it is usually quite easy to define a distance metric. However, as the natural phenomenon interests specimen, such methods are effective only when the distance between genotypes is related to the distance between phenotypes. On the other hand, working directly at the level of phenotype is usually impracticable, while the connection between fitness values and phenotypes is even more feeble.

This paper proposes a technique for promoting diversity in GE independent from the problem tackled (actual grammar and fitness) and from other GE components (genotype-phenotype mapping, selection criteria, and genetic operators). The technique may be set to operate at a very specific level (i.e., genotype, phenotype, or fitness), and for each level its effect on the performances is evaluated on four different problems. The experimental results also suggest that similar mechanisms could be beneficial for different type EAs.

2 PROPOSED APPROACH

We briefly present our modified version of GE including a Diversity Promotion mechanism, which we called DP-GE. It exploits a niching method, with explicit neighborhood, but using zero as niche’s radius. Indeed, at any level, detecting *clones*, that is, individuals that are exactly alike, is quite easy.

DP-GE evolves a population of n_{pop} sets of individuals, each one corresponding to a 0-radius niche. Sets form a partition \mathcal{I} of a larger population, and individuals in the same niche are equivalent according to a specific relation: each set in the partition has a representative. Whenever an individual has to be selected for reproduction, the selection criterion is applied on the set of representatives. Thus, the niche is selected first, then one individual

Table 1: Best fitness (mean and standard deviation across repetitions) for GE and different DP-GE variants (with the Youngest parent representative criterion) for different genotype sizes $|g|$ and problems. In each row, the best figure is highlighted.

Problem	GE	Geno.	DP-GE	
			Pheno.	Fitness
Harmonic	7 ±1.8	8.3±3.4	5.9±0.6	6.2±1.7
Polynomial	0.9±0.4	1.1±0.6	0.5±0.3	0.6±0.4
Santa-Fe	39.2±5	40 ±6	33.7±2.8	34.9±4.9
Text	3.6±1.5	3.8±1.3	2.1±0.8	4.3±1.6

inside the niche is chosen, reducing the probability to be selected for individuals in crowded niches. We impose a maximum size n_{part} to each partition.

There are 4 selection criteria in DP-GE, each operating on an input set of individuals and giving in output a single individual: SELECTPARENT, SELECTUNSURVIVAL, PARENTREP, and UNSURVIVALREP. The latter two criteria are used to build the set of representatives of the partitions in \mathcal{I} , respectively in the reproduction phase and in the deletion phase. The former two criteria are used on the set of representatives in the same way they are used in the standard GE algorithm.

We considered different options for the DP-GE specific components PARENTREP and UNSURVIVALREP: for space constraints we here present the results obtained with youngest individual and oldest individual criteria, respectively. Concerning the equivalence relation, we explored 3 options consisting in considering two individuals equivalent if they have the same genotype, or phenotype, or fitness.

3 EXPERIMENTAL EVALUATION

We performed an experimental evaluation with the 4 benchmark problems used in [4] and the following parameters: population of 500 randomly initialized individuals, 50 generations, genotype size of 256 bit, two-points crossover with 0.8 probability, bit flip mutation ($p_{mut} = 0.01$) with 0.2 probability, tournament parent selection with size 5, worst fitness unsurvival selection, steady-state replacement, codon size of 8 bit, and max wrappings of 5.

For GE and each of the design variants of DP-GE we performed 30 independent evolutionary runs on each problem. For DP-GE, we set the partition size n_{part} to 20—we experimentally verified that small perturbations to this parameter do not alter substantially the overall findings.

Table 1 shows the mean value and the standard deviation, across the 30 runs, of the fitness of the best individual at the end of the evolution, with the Youngest criterion. It can be seen that DP-GE outperforms GE in all cases. In particular, the variant Youngest-Phenotype obtains better fitness values for all the problems. The difference w.r.t. GE is statistically significant ($p < 0.01$ with the Mann-Withney test) for Youngest-Phenotype DP-GE on all problems.

We carefully analyzed the raw results of our experimentation in order to understand why DP-GE, and in particular its variant

Youngest-Phenotype, consistently outperformed GE. We believe that one of the main motivations is in the ability of DP-GE to generate larger phenotypes. From an high level point of view, our claim is consistent with the findings of [8]. In facts, the cited paper demonstrates that GE is highly redundant and, moreover, that the redundancy is not uniform, being instead biased towards small phenotypes. Promoting diversity and, thus, decreasing redundancy may help to obtain larger phenotypes.

4 CONCLUDING REMARKS

GE is a modular EA promoting a sharp separation between genotypes and phenotypes: the former are sequences of bits while the latter are programs in any language described by a context-free grammar. A key issue with GE is preserving an adequate amount of diversity during evolution. While diversity is a potential issue in many EAs, GE tends to be particularly prone to magnifying such issue due to its inherent structure.

In this work, we have proposed a mechanism for promoting diversity in GE, called DP-GE, that is a form of niching in which each niche (partition) is formed by clones. Genetic operators are applied to individuals that belong to different partitions and may be selected based on various criteria.

We assessed experimentally DP-GE and compared it against GE in terms of effectiveness: our proposal leads to consistent improvements in the fitness of the solutions, often due to its ability to construct solutions consisting of longer phenotypes. Although our experiments did not identify a clear winner among the design variants and parameter values that we analyzed, we believe that our proposal may indeed constitute an effective framework for building practical GE applications.

REFERENCES

- [1] Anthony Brabazon, Michael O'Neill, and Seán McGarraghy. 2015. Grammatical Evolution. In *Natural Computing Algorithms*. Springer Nature, 357–373. DOI: http://dx.doi.org/10.1007/978-3-662-43631-8_19
- [2] Tom Castle and Colin G Johnson. 2010. Positional effect of crossover and mutation in grammatical evolution. In *European Conference on Genetic Programming*. Springer, 26–37.
- [3] Samir W Mahfoud. 1995. Niching methods for genetic algorithms. *Urbana* 51, 95001 (1995), 62–94.
- [4] Eric Medvet. 2017. A Comparative Analysis of Dynamic Locality and Redundancy in Grammatical Evolution. In *Genetic Programming: 20th European Conference, EuroGP 2017, Amsterdam, Netherlands, April 19-21, 2017, Proceedings*. Springer International Publishing, Cham, to appear.
- [5] Conor Ryan, JJ Collins, and Michael O'Neill. 1998. Grammatical evolution: Evolving programs for an arbitrary language. In *European Conference on Genetic Programming*. Springer, 83–96.
- [6] Bruno Sareni and Laurent Krahenbuhl. 1998. Fitness sharing and niching methods revisited. *Evolutionary Computation, IEEE Transactions on* 2, 3 (1998), 97–106. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=735432
- [7] Giovanni Squillero and Alberto Paolo Tonda. 2016. Divergence of character and premature convergence: A survey of methodologies for promoting diversity in evolutionary optimization. *Information Sciences* 329 (2 2016), 782–799. DOI: <http://dx.doi.org/10.1016/j.ins.2015.09.056>
- [8] Ann Thorhauer. 2016. On the Non-uniform Redundancy in Grammatical Evolution. In *International Conference on Parallel Problem Solving from Nature*. Springer, 292–302.
- [9] Ann Thorhauer and Franz Rothlauf. 2014. On the locality of standard search operators in grammatical evolution. In *International Conference on Parallel Problem Solving from Nature*. Springer, 465–475.
- [10] Nguyen Quang Uy, Nguyen Xuan Hoai, Michael O'Neill, R.I. McKay, and Dao Ngoc Phong. 2013. On the roles of semantic locality of crossover in genetic programming. *Information Sciences* 235 (jun 2013), 195–213. DOI: <http://dx.doi.org/10.1016/j.ins.2013.02.008>