Highlights of genetic programming 2020 events

Miguel Nicolau

Received: 6 October 2021 / Accepted: 7 October 2021 / Published online: 16 October 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

This special issue highlights outstanding work published at two prestigious Genetic Programming (GP) events: EuroGP 2020, and the GECCO 2020 GP Track. Given that both were scheduled to take place in 2020, they had to be moved online, at the very last minute. The quality of the work presented here is a testament to the resilience of GP researchers, and the healthy research drive of the field.

The European Conference on Genetic Programming (EuroGP) is the world’s only conference devoted to GP. In 2020, it was chaired by Nuno Lourenço and Ting Hu. The conference proceedings published 18 papers (12 oral presentations and 6 poster presentations), from a total of 36 submissions. The Genetic and Evolutionary Computation Conference (GECCO) is the largest conference in the field of Evolutionary Computation, and the flagship conference of the Association for Computing Machinery (ACM) Special Interest Group on Genetic and Evolutionary Computation (SIGEVO). One of its main tracks specifically focuses on GP. In 2020, the GP Track was chaired by Miguel Nicolau and Mengjie Zhang. From a total of 37 submissions, 15 papers were accepted for oral presentation, and 11 were accepted as posters.

The goal of this special issue is to promote diverse and promising research presented at these two GP events. From each event, we selected four of the highest rated papers, according to the chairs and the reviewing process. Their authors were invited to expand their original work and submit an extended version to this special issue. These submissions were re-reviewed by the original conference program committee reviewers, and went through the journal’s extensive review process. This ensures that the highlighted research and their extended versions meet the highest scholarly standards, and helps the research reach a wider audience.

In Lones’s article, a GP algorithm was used to explore the space of continuous optimisers, with the goal of discovering novel ways of performing optimisation. Specifically, the optimisers were evolved using PushGP and were found to be diverse and to explore their optimisation landscape using a variety of interesting, and sometimes unusual, strategies. These evolved optimisers were also found

---

* Miguel Nicolau
miguel.nicolau@ucd.ie

1 University College Dublin, Dublin, Ireland
to generalise well to unseen problems. This work highlights GP’s ability to evolve robust optimisers that can be applied to a broad variety of problem types and sizes.

The paper written by Mariot et al. tackled the search for a class of reversible cellular automata (RCA) whose local update rules are defined on conserved landscape, as an optimization problem using GA and GP. Three search scenarios were designed, namely single-objective, multi-objective, and lexicographic approach. Both GA and GP were shown to be effective on the multi-objective and lexicographic approach to discover a trade-off between the reversibility and the Hamming weight of conserved landscape rules, as well as to observe that conserved landscape CA cannot be used in symmetric cryptography given its low Hamming weight.

The topic of using Semantic Genetic Programming with Dynamic Targets was covered in the article by Ruberto et al. In this article, the authors propose a method based on Semantic Genetic Programming, SGP-DT, that divides the problem being solved into multiple runs. The quality of the solutions in each run is measured using a dynamic target based on the residuals errors of previous executions. To obtain the final solution to the problem, the proposed method relies on a linear scaling method to combine the solutions of each run. The performance of the approach is assessed in eleven benchmarks and compared with other Machine Learning techniques. The results show that SGP-DT achieves small error values. Additionally, the authors presented a study on how parameter tuning affects the performance in terms of efficiency and effectiveness.

Saini et al. presents a study on the relationship between parent selection methods, looping constructs and success rate when using Genetic Programming to tackle the problem of Program Synthesis. In concrete, they analyse if the parent selection methods that lead to higher success rates in the program synthesis also lead to an increasing number of individuals with loops, and why. The experiments are conducted using Push Genetic Programming (PushGP) and the General Program Synthesis Benchmark Suite. The results show that the parent selection methods impact the quality of the solutions.

In their paper, Schmitt et al. tackle the complex issue of evolving efficient multigrid solvers for the discretization of partial differential equations. Their approach, EvoStencils, uses grammar-guided genetic programming to represent and optimize trees of mathematical expressions, which are evaluated in terms of convergence and compute performance. In this extended version, the authors introduce an evaluation process based on code generation, which allows for an efficient distributed computation evaluation process, across multiple compute nodes. Their results outperform hand-crafted solvers for Poisson’s equation and a problem with up to 16 million unknowns, by using multi-core processor architectures.

Hodan, Mrazeck and Vasicek’s work proposes the introduction of a semantically-oriented mutation operator for Cartesian GP, in line with recent advances in semantic operators for tree-based GP. This extended version presents a modification of the original semantic operator, by allowing more than one gene to be modified, when the operator is used. A detailed study of the computational complexity of both the
original and modified operators is also performed. Finally, an extended set of benchmark circuits and experiments are used, to validate their approach.

The work authored by Kelly, Voegerl, Banzhaf and Gondro describes an approach to evolving hierarchical memory-prediction machines in a multi-task reinforcement learning environment. Built on a GP framework called Tangled Program Graph (TPG), this extended version proposes a highly-modular memory structure that manages the temporal properties of a task and enables operation in problems with continuous action spaces, and shows that TPG can evolve hierarchical multi-task behaviors by combining several agents which were initially adapted independently. The new developments show that the resulting agents are very competitive with task-specific agents in six environments and that the hierarchical structure of programs allows for dynamic run-time complexity, which results in relatively efficient operations.

Finally, Sotto et al. investigate graph representations in GP, including CGP, LGP, EGGP and standard GP, when the same evolutionary algorithm (EA) is used. The extended version concentrates on how EAs affect the performance of used for these graph GP techniques and standard tree-based GP, whether a graph representation provides an advantage over trees, and whether there exists a relationship between the frequency of reusing intermediate results in LGP and CGP and the performance on parity circuits problems. The investigations show that the best choice of representation, genetic operator and EA depends on the problem domain, that graph GP methods can increase search performance on complex real-world regression problems, and that the reuse of intermediate results by tuning LGP’s number of registers and CGP’s levels back parameter is of utmost importance.

We would like to extend our sincere appreciation to everyone involved in making this special issue happen, despite many setbacks caused by the COVID-19 pandemic. To the authors, for taking the time to extend their research for publication in this issue; to the reviewers, for dedicating their time to ensure the scientific rigor of the submissions; to the Springer editorial team, for their help throughout the publishing process; and finally to the Editor-In-Chief, Lee Spector, ever present to guide us through the finishing line. A big thank you to you all.

Guest Editors

Miguel Nicolau
Ting Hu
Mengjie Zhang
Nuno Lourenço

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.