Who Found What: Searches of the Genetic Programming Bibliography

W. B. Langdon

Abstract

The genetic programming bibliography key word search has been used hundreds of times, with an average response of 21 milliseconds and more than 10 000 GP papers found. We summarise recent queries, top GP papers retrieved, user devices and updates to `https://gpbib.cs.ucl.ac.uk/gp-search/free_text_search.html`

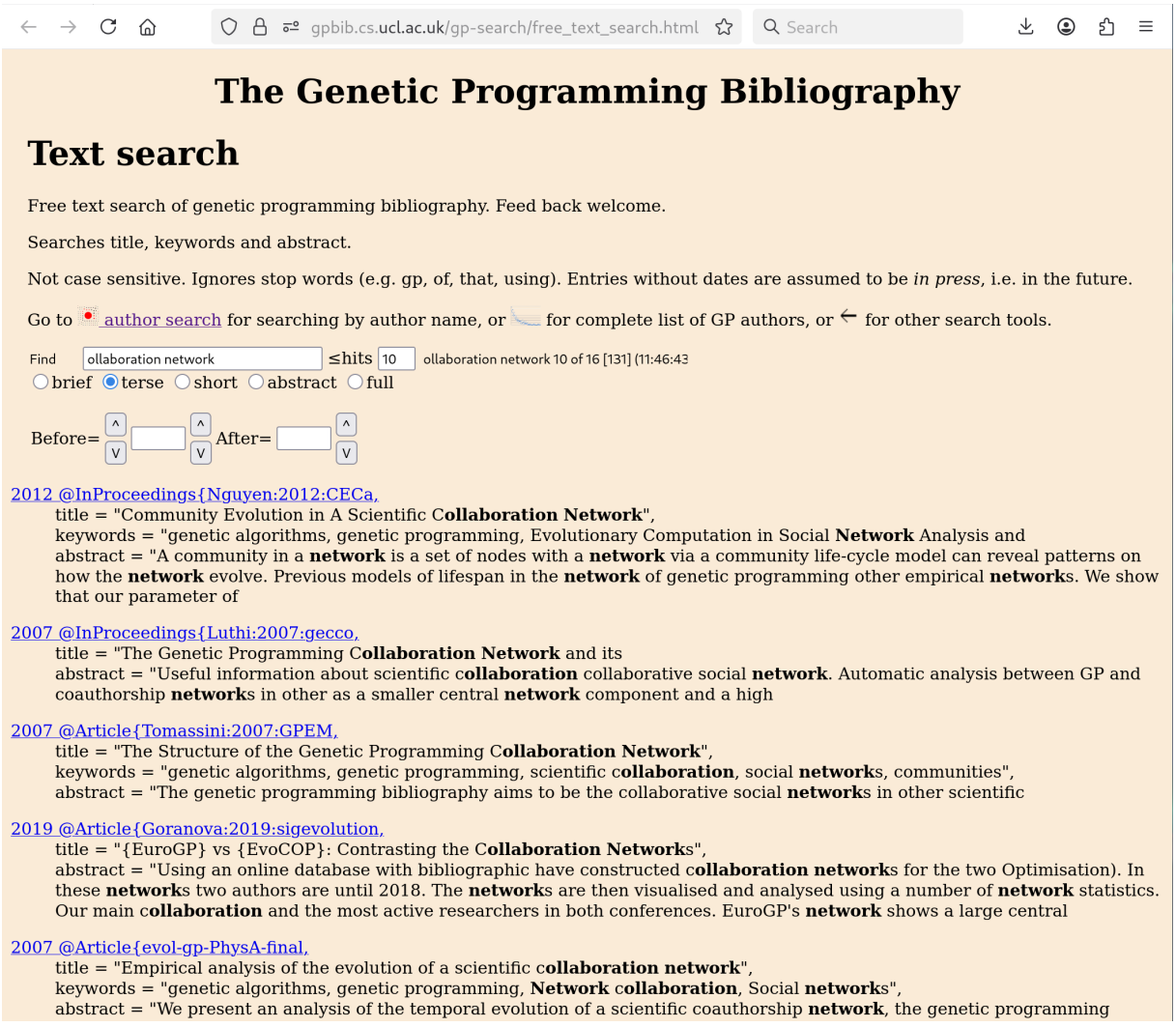

Figure 1: Updated GP Bibliography Web Browser Keyword Query Interface

Table 1: Locations of GP search users (7 Dec 2024 to 26 Feb 2026)

Ireland	48	Austria	7	Sweden	1
UK	41	Czech Republic	6	Singapore	1
USA	35	New Zealand	4	Russian Federation	1
Canada	28	Mexico	3	Portugal	1
Unknown	24	Italy	3	Netherlands	1
Germany	21	Australia	3	Maldives	1
Brazil	15	Turkey	2	Korea	1
India	11	Spain	2	Israel	1
Romania	10	Malaysia	2	Denmark	1
Japan	10	France	2		
Total					286

1 Background

The Genetic Programming bibliography aims to cover all papers, books, PhD thesis, etc. on genetic programming. It was started by John Koza and first published in 1994 as Appendix F of his second GP book [Koza(1994)], with an expanded version as Appendix B in Advances in GP 2 [Langdon(1996)]. It is now fully online and hosted by the Computer Science department of University College, London `http://gpbib.cs.ucl.ac.uk`, with a automated mirror hosted by the Perelman School of Medicine in the University of Pennsylvania `http://gpbib.pmacs.upenn.edu`.

Two years ago the bibliography, particularly its then new search interface, was described in the ACM special interest group on evolutionary computing (SIGEVO)'s newsletter SIGEVOlution [Langdon(2024)]. Since the search (Figure 1) differs from some early approaches, in being run in the user's web browser or smart phone rather than a remote server, and provides free text search, rather than syntax based database queries, we review its recent use and performance. Section 2 shows it has been used in many countries, Section 3 shows it is amazingly fast, whilst Sections 4 and 5 shows the sort of queries people prefer and the results they get. We finish by listing some limitations (Section 6), changes since [Langdon(2024)] (Section 7) and suggestions for additional features (Section 8).

2 Usage of Keyword Search

Although free_text_search can be used off line (and indeed via the University of Pennsylvania mirror) we report usage based on the data logged at UCL, feeling that this will give a reasonable indication. (Although running for more than two years, due to upgrades, we present data from the end of December 2024 onwards.)

Table 1 show the search facility has been used several hundred times from across the world, notably from Ireland, Europe and North America.

So far three types of browser have been used for GP keyword searches. Table 2 shows Apple's Safari web browser is the most common (52%), with Firefox almost as popular (38%), followed by Microsoft's Edge web browser (9%). Less than 2% of queries are run from the user's smart phone.

2.1 Usage of Options

Figure 1 shows the user interface allows people to: alter the number of matches displayed (hits), select one of five display modes, and choose the papers to be displayed by their publication date (e.g. "Before" 2010 and "After" 2000, i.e. during 2000–2010, inclusive). Notice, if "Before" and "After" are set to the same

Table 2: GP keyword search by user browser

Desktop	Microsoft Edge	55
Desktop	Firefox	223
Desktop	Apple Safari	292
iPhone	Apple Safari	3
Android	Apple Safari	8
Total		581

Table 3: Number of times user interface controls are used

Hits		Display mode		Years	
default (top 10 matches)	1109	brief	45	Before	28
others	119	terse (default)	1084	After	43
		short	80	default (any time)	1157
		abstract	7		
		full	12		
Total	1228	Total	1228	Total	1228

value, only matching papers from that year are displayed. The central up or down arrow buttons can be used to simultaneously change both before and after years, allowing easy scrolling through multiple matches according to which year they came out.

Table 3 shows almost all people used the default settings. However in 10% of queries the maximum number of matching GP papers ("Hits") was changed from its default value (ten), typically to its maximum value (100).

Similarly the results of all but 12% of queries were displayed in the default mode ("terse"). The "brief" mode can be used standalone, i.e. without connection to UCL, and so may not be logged. Nevertheless Table 3 shows (when it is logged) it accounts for less than 4% of queries. The "abstract" display mode (which is like "full" but only shows the matching title, keywords and abstract) was recently added in response to a user request (Section 7). Since it has been available for less time than the other display modes, this may in part explain why it only used in half a percent of queries. However, the logs continue to show a very strong preference for users to accept the defaults.

3 Performance

Figure 2 shows search is effectively instantaneous (median response is 21 milliseconds). As expected, the distribution of search times has a long tail, nevertheless 95% of searches complete within $1/10^{\text{th}}$ of a second. Figure 2 shows the speed recorded by user's device, while it includes the time to update the search interface's own display (upper part of Figure 1). Except for the "brief" display option, Figure 2 does not include the time to fetch and display results across the Internet. From the Apache logs it seems fetching the result web pages typically takes less than a second.

Since the "brief" option does not rely on immediate response across the Internet, as well as giving compact answers, it might also be a useful option on slow connections.

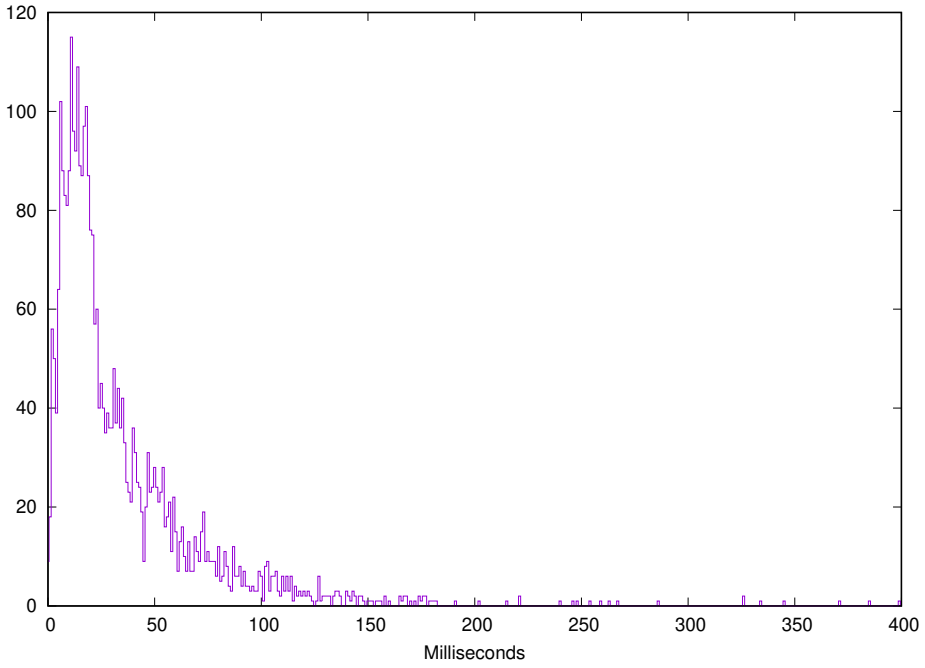

Figure 2: Speed of GP keyword search on user's device. Half queries take $\leq$ 21 milliseconds.

Table 4: Number of words in Free Text Searches from 7 Dec 2024 to 26 Feb 2026

Words	Count	Words	Count	Words	Count
1	249	9	7	15	1
2	149	10	4	16	1
3	65	11	3	18	1
4	25	12	2	19	1
5	21	13	5	21	1
6	18	14	2	69	1
7	13			79	1
8	11				
Total					581

4 What People Look For

Table 4 shows almost half the queries consist of a single keyword with the number of compound searches falling rapidly with their length. There are 7 single cases of very long queries (which must have been copy-pasted into the search box) that are composed of more than 15 keywords and which are clearly targeted very precisely at a single paper e.g. [Brabazon et al.(2020), La Cava et al.(2021), Crary et al.(2025), Burke et al.(2004)].

As Table 4 shows, there have been more than 500 searches. This is too many to mention them all, so instead next we highlight some keywords that are used multiple times. The most popular query is "Symbolic Regression". Other popular keywords include: "Selection", "Model" and "Diversity". Whilst "machine" appears in composite queries, such as "machine learning" and "machine code" and as part of the journal name "evolvable machines"[1]. Whilst "Learning" appears in composite queries such as "machine learning" (just mentioned) and "Reinforcement Learning". Similarly "Evolutionary" appears in multiple composites, while "Neural" appears in composites associated with artificial neural networks

[1]Genetic Programming and Evolvable Machines

(ANNs). Again "Network" is in ANN composite queries but also searches for Genetic Network Programming, interaction networks and neutral networks. "Guide" and "Field" came up in searches for A field guide to genetic programming [Poli et al.(2008)]. "boolean" is commonly associated with composites involving "boolean function". Other popular searches include "Graph" (in composites) "Analysis" (in composites) "Search" (in composites) "geometric semantic" "linear genetic programming" (note use of year to reduce volume of matches, Section 2.1) "lexicase selection" "grammatical evolution" "Ant-Based" "intrusion detection" "hydrology" "Fitness" "Cartesian Genetic Programming" "recursion" "Population" (in composites) "image classification" "GAN" (i.e. Generative Adversarial Networks) and "Dynamic" in composites such as "dynamic programming", and "dynamical system", etc. Other searches include "Algorithm design" and "Automatic design" "Ant colony optimization" "Cancer", "theory", "Testing", "surrogate", "mutation", "fuzzy", "elitism", "differential equations", "classification", and "benchmark".

5 People Found 10 290 GP papers

Putting aside the problem that the logs cannot be complete, e.g. due web pages being cached or brief output not being logged (Section 2.1 above) or particularly at busy periods data may not be logged, from 7 Dec 2024 to 26 Feb 2026 some 10 290 GP papers have been found. Figure 3 shows, as expected, a few are much more common than others and there is a long tail. For example, 1230 papers were only found once. As shown with dotted line in Figure 3, the frequency falls approximately as a power law. Table 5 lists the papers that were found more than twenty times.

Table 5: Top GP papers found by free text search

count	paper
40	banzhaf:2024:GECCOcomp [Banzhaf and Ting Hu(2024)]
35	hafner:1996:GGP [Hafner et al.(1996)]
33	al-afandi:2021:Algorithms [Al-Afandi and Horvath(2021)]
29	Bautu2012 [Bautu and Barbulescu(2012)]
28	poli08:fieldguide [Poli et al.(2008)]
26	GeyerSchulz96g [Geyer-Schulz(1996)]
25	Kofod-petersen:masters [Kofod-Petersen(2002)]
24	banzhaf:1998:GP [Banzhaf et al.(1998)]
23	Megane:2021:EuroGP [Megane et al.(2021)]
22	banzhaf:2025:GECCOcomp [Banzhaf and Ting Hu(2025)]
22	anderson:1994:profile [Anderson(1994)]
21	spector:2005:GPTP [Spector and Klein(2005)]
21	langdon:2008:bookshelf [Langdon(2008)]
21	Kifetew:2014:ICST [Kifetew et al.(2014)]
21	DBLP:phd/dnb/Keller12a [Keller(2012)]
20	ROMANO:2023:comtox [Romano et al.(2023)]
20	Penaloza-Mejia:2017:GECCO [Penaloza-Mejia et al.(2017)]

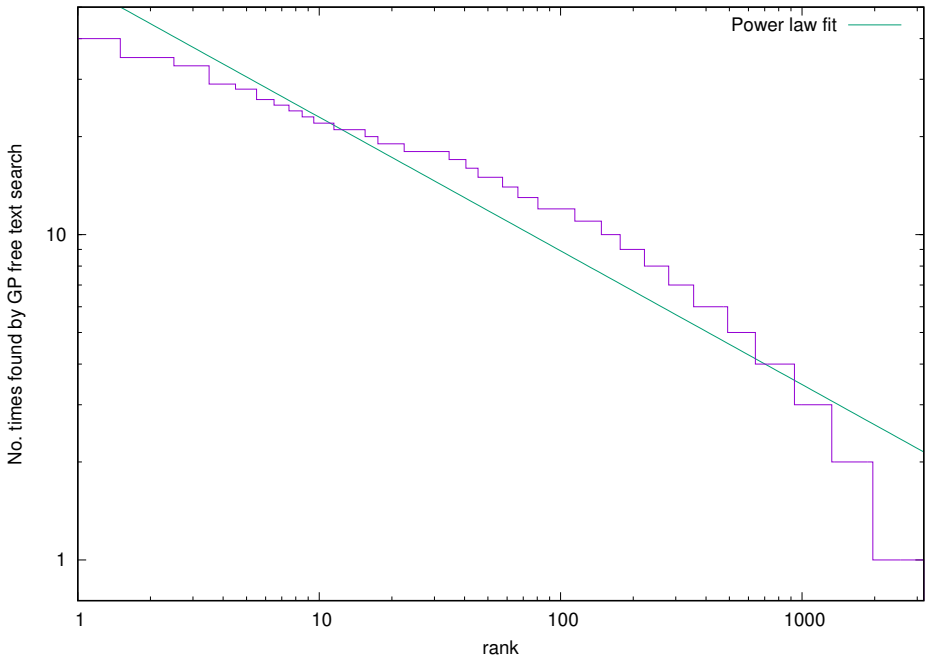

Figure 3: Number of times individual GP papers have been found by keyword searches. Note log scales. See also Table 5.

6 Limitations

6.1 Things it Cannot Do

Of course it cannot find things that are not in the GP Bibliography. But also some queries have simply pasted URLs into the search box. It does its best to interpret this as a sequence of keywords but it is not a web browser.

6.1.1 Searching for GP: Everything is GP

The most popular search is for "Genetic Programming". However naturally enough this will match every entry in the GP bibliography. Such a result is not very useful and internally the search engine limits itself to provide useful results quickly (see Section 3). Another popular search keyword is "GP", but again this is useless. It appears in several thousands of entries and (as mentioned at the top of the free text search web page, Figure 1) it is therefore a stopword (Section 6.2) and being a stop word it will not match anything.

6.1.2 It cannot fix spelling or typing errors

The search interface does not do spelling correction, so there will be some typo failures (e.g. "ensamble" rather than "ensemble").

6.1.3 It searches for all keywords, Stemming for free

The search is deliberately as simple as possible, therefore you cannot link searches with *and* or *or*. (As it searches for all words you can think of it as having an implicit *and* between each one.)

Similarly you cannot bracket search words or link them with quotation marks. At the end of the "Find" line (after the ≤hits input box, see Figure 1), the search interface reflects back the search it has actually conducted, as well as the number of matches found and the number displayed.

Table 6: Stop Words. Words that are ignored in free text searches and so not stored in its database.

an and are as be by can for from gp in is it of on that the this to used using we which with

Since we do partial matches, effectively we get stemming and plurals for free. For example, "plan" will match "plan" and "plans", and also "planning", "re-plan" and "replanning".

6.1.4 It cannot search for authors or publishers

The GP bibliography has multiple ways to search for papers written by specific authors. For example, the main index lists all authors by name and then link's to all their GP papers. Whilst the coauthors graph includes a search box which allows partial matches on both fore and surnames. Nonetheless GP text search will do its best by treating author names as keywords, which can be successful where people's names appear in paper titles or abstracts. For example, Koza, Banzhaf and Nordin.

Similarly it cannot search for papers published by particular publishers and again matches are mostly accidental, e.g. where a publisher's name occurs in a paper's title e.g. [Togelius et al.(2008), Lewis et al.(2006), Langdon(2019), Kechagia et al.(2022), Petke et al.(2019), Petke et al.(2018)].

6.2 Stop Words

Stop words are words that are so common that they are ignored. The use of stop words is very common in text search. For example, searching for "the" in english prose is seldom useful as it occurs too many times. Therefore "the" is often treated as a stop word. By excluding stop words from keyword searches, they can also be removed from the online database, giving useful network bandwidth savings. Our stop words are calculated based on the text in GP bibliography, see Table 6. A few (particularly "GP") are mentioned on the web page itself (see Figure 1).

7 Changes Since 2024

Primarily to simplify things for the user, ten words which were originally stopwords are no longer stop words. They are: algorithm, algorithms, approach, based, data, evolutionary, genetic, model, programming, results.

The number of genetic programming papers continues to grow, and although the query's database has grown, e.g. due to fewer stop words, by 17% it is still less than 12MB.

As mentioned in Section 2.1, since September 2025 we have an additional "abstract" display mode. This reduces the volume of text displayed by showing only the abstract of matching GP papers.

8 Additional Features?

Do we need new features? For example, would you like to limit searches to just PhD thesis?

Comments and suggestions are welcome.

Acknowledgements

I would like to thank Stjepan Picek for suggesting the new abstract display mode feature.

References

[Al-Afandi and Horvath(2021)] Jalal Al-Afandi and Andras Horvath. 2021. Adaptive Gene Level Mutation. *Algorithms* 14, 1 (2021). http://dx.doi.org/10.3390/a14010016

[Anderson(1994)] Kenneth R. Anderson. 1994. *Courage in Profiling.* Technical Report. BBN. https://gpbib.cs.ucl.ac.uk/gp-html/anderson_1994_profile.html

[Banzhaf et al.(1998)] Wolfgang Banzhaf, Riccardo Poli, Marc Schoenauer, and Terence C. Fogarty (Eds.). 1998. *Genetic Programming.* LNCS, Vol. 1391. Springer-Verlag, Paris. http://dx.doi.org/10.1007/BFb0055923

[Banzhaf and Ting Hu(2024)] Wolfgang Banzhaf and Ting Hu. 2024. Linear Genetic Programming. In *Proceedings of the 2024 Genetic and Evolutionary Computation Conference Companion (GECCO '24)*, Mengjie Zhang and Emma Hart (Eds.). Association for Computing Machinery, Melbourne, Australia, 759–771. http://dx.doi.org/10.1145/3638530.3648422 Tutorial.

[Banzhaf and Ting Hu(2025)] Wolfgang Banzhaf and Ting Hu. 2025. Linear Genetic Programming. In *Proceedings of the 2025 Genetic and Evolutionary Computation Conference Companion (GECCO '25 Companion)*, Mengjie Zhang and Emma Hart (Eds.). Association for Computing Machinery, Malaga, Spain, 1749–1766. http://dx.doi.org/10.1145/3712255.3716528 Tutorial.

[Bautu and Barbulescu(2012)] Elena Bautu and Alina Barbulescu. 2012. A Hybrid Approach for Modelling Financial Time Series. *The International Arab Journal of Information Technology (IAJIT)* 9, 4 (July 2012), 327–335. http://www.ccis2k.org/iajit/PDF/vol.9,no.4/2806-5.pdf

[Brabazon et al.(2020)] Anthony Brabazon, Michael Kampouridis, and Michael O'Neill. 2020. Applications of genetic programming to finance and economics: past, present, future. *Genetic Programming and Evolvable Machines* 21, 1-2 (June 2020), 33–53. http://dx.doi.org/10.1007/s10710-019-09359-z Twentieth Anniversary Issue.

[Burke et al.(2004)] Edmund K. Burke, Steven Gustafson, and Graham Kendall. 2004. Diversity in Genetic Programming: An Analysis of Measures and Correlation with Fitness. *IEEE Transactions on Evolutionary Computation* 8, 1 (Feb. 2004), 47–62. http://dx.doi.org/10.1109/TEVC.2003.819263

[Crary et al.(2025)] Christopher Crary, Wesley Piard, Greg Stitt, Benjamin Hicks, Caleb Bean, Bogdan Burlacu, and Wolfgang Banzhaf. 2025. Using FPGA devices to accelerate the evaluation phase of tree-based genetic programming: an extended analysis. *Genetic Programming and Evolvable Machines* 26 (2025), Article no 8. http://dx.doi.org/10.1007/s10710-024-09505-2 Online first.

[Geyer-Schulz(1996)] Andreas Geyer-Schulz. 1996. Fuzzy Genetic Algorithms. *Handbook of Fuzzy Systems* (April 1996). Work in progress.

[Hafner et al.(1996)] Christian Hafner, Juerg Froehlich, and Hansueli Gerber. 1996. Generalized Genetic Program. (1996). Submitted to the 'Evolutionary Computation' Journal.

[Kechagia et al.(2022)] Maria Kechagia, Shin Hwei Tan, Sergey Mechtaev, and Lin Tan (Eds.). 2022. *2022 IEEE/ACM International Workshop on Automated Program Repair (APR).* IEEE, Pittsburgh, PA, USA. https://ieeexplore.ieee.org/xpl/conhome/9474454/proceeding

[Keller(2012)] Robert E. Keller. 2012. *Toward autopoietic programming.* Ph. D. Dissertation. LS11, Technical University Dortmund, Germany. http://dx.doi.org/10.17877/DE290R-17173

[Kifetew et al.(2014)] Fitsum Meshesha Kifetew, Wei Jin, Roberto Tiella, Alessandro Orso, and Paolo Tonella. 2014. Reproducing Field Failures for Programs with Complex Grammar-Based Input. In *Seventh International Conference on Software Testing, Verification and Validation, ICST 2014*. IEEE, Cleveland, Ohio, USA, 163–172. http://dx.doi.org/10.1109/ICST.2014.29

[Kofod-Petersen(2002)] Anders Kofod-Petersen. 2002. *Adaptive Behaviour Based Robotics using On-Board Genetic Programming*. Cand. Scient. Norwegian University of Science and Technology, Norway. https://hdl.handle.net/11250/249931

[Koza(1994)] John R. Koza. 1994. *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, Cambridge Massachusetts. http://www.genetic-programming.org/gpbook2toc.html

[La Cava et al.(2021)] William La Cava et al. 2021. Contemporary Symbolic Regression Methods and their Relative Performance. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, Joaquin Vanschoren and Sai-Kit Yeung (Eds.), Vol. 1. Curran. https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/c0c7c76d30bd3dcaefc96f40275bdc0a-Abstract-round1.html

[Langdon(2019)] Bill Langdon. 2019. Mark Harman wins both IEEE and ACM Awards in 2019. *SIGEVOlution newsletter of the ACM Special Interest Group on Genetic and Evolutionary Computation* 12, 2 (Aug. 2019), 10–11. http://dx.doi.org/10.1145/3357514.3357517

[Langdon(1996)] William B. Langdon. 1996. A Bibliography for Genetic Programming. In *Advances in Genetic Programming 2*, Peter J. Angeline and K. E. Kinnear, Jr. (Eds.). MIT Press, Cambridge, MA, USA, Appendix B, 507–531. http://dx.doi.org/10.7551/mitpress/1109.003.0033

[Langdon(2008)] W. B. Langdon. 2008. A Field Guide to Genetic Programing. 8 pages. https://gpbib.cs.ucl.ac.uk/gp-html/langdon_2008_bookshelf.html

[Langdon(2024)] W. B. Langdon. 2024. Searching the Genetic Programming Bibliography. *SIGEVOlution newsletter of the ACM Special Interest Group on Genetic and Evolutionary Computation* 17, 1 (March 2024), Article No: 2. http://dx.doi.org/10.1145/3687242.3687244

[Lewis et al.(2006)] Tim Lewis, Neil Fanning, and Gary Clemo. 2006. Enhancing IEEE802.11 DCF using Genetic Programming. In *IEEE 63rd Vehicular Technology Conference, VTC 2006-Spring*, Vol. 3. IEEE, Melbourne, Australia, 1261–1265. http://dx.doi.org/10.1109/VETECS.2006.1683037

[Megane et al.(2021)] Jessica Megane, Nuno Lourenco, and Penousal Machado. 2021. Probabilistic Grammatical Evolution. In *EuroGP 2021: Proceedings of the 24th European Conference on Genetic Programming (LNCS, Vol. 12691)*, Ting Hu, Nuno Lourenco, and Eric Medvet (Eds.). Springer Verlag, Virtual Event, 198–213. http://dx.doi.org/10.1007/978-3-030-72812-0_13

[Penaloza-Mejia et al.(2017)] Ollin Penaloza-Mejia, Eddie Clemente, Marlen Meza-Sanchez, Cynthia B. Perez, and Francisco Chavez. 2017. GP-based Motion Control Design for the Double-integrator System Subject to Velocity Constraint. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO '17)*. ACM, Berlin, Germany, 73–74. http://dx.doi.org/10.1145/3067695.3076094

[Petke et al.(2019)] Justyna Petke, Shin Hwei Tan, William B. Langdon, and Westley Weimer (Eds.). 2019. *Proceedings 2019 ACM/IEEE 6th International Genetic Improvement Workshop, GI 2019*. Montreal. http://www.cs.ucl.ac.uk/staff/W.Langdon/icse2019/GI_2019_frontmatter.pdf

[Petke et al.(2018)] Justyna Petke, Kathryn Stolee, William B. Langdon, and Westley Weimer (Eds.). 2018. *Proceedings 2018 ACM/IEEE 4th International Genetic Improvement Workshop, GI 2018*.

Gothenburg, Sweden. http://www.cs.ucl.ac.uk/staff/W.Langdon/icse2018/GI_2018_FM_to_chair_04-26-18.pdf

[Poli et al.(2008)] Riccardo Poli, William B. Langdon, and Nicholas Freitag McPhee. 2008. *A field guide to genetic programming*. Published via http://lulu.com and freely available at. http://www.gp-field-guide.org.uk (With contributions by J. R. Koza).

[Romano et al.(2023)] Joseph D. Romano, Liang Mei, Jonathan Senn, Jason H. Moore, and Holly M. Mortensen. 2023. Exploring genetic influences on adverse outcome pathways using heuristic simulation and graph data science. *Computational Toxicology* 25 (2023), 100261. http://dx.doi.org/10.1016/j.comtox.2023.100261

[Spector and Klein(2005)] Lee Spector and Jon Klein. 2005. Trivial Geography in Genetic Programming. In *Genetic Programming Theory and Practice III*, Tina Yu, Rick L. Riolo, and Bill Worzel (Eds.). Genetic Programming, Vol. 9. Springer, Ann Arbor, Chapter 8, 109–123. http://dx.doi.org/10.1007/0-387-28111-8_8

[Togelius et al.(2008)] Julian Togelius et al. 2008. The 2007 IEEE CEC simulated car racing competition. *Genetic Programming and Evolvable Machines* 9, 4 (Dec. 2008), 295–329. http://dx.doi.org/10.1007/s10710-008-9063-0