Genetic and Evolutionary Computation

Series Editors:
Wolfgang Banzhaf, Michigan State University, East Lansing, MI, USA
Kalyanmoy Deb, Michigan State University, East Lansing, MI, USA

More information about this series at http://www.springer.com/series/7373

Wolfgang Banzhaf • Erik Goodman
Leigh Sheneman • Leonardo Trujillo • Bill Worzel
Editors

Genetic Programming Theory and Practice XVII

Springer

Editors
Wolfgang Banzhaf
Computer Science and Engineering
John R. Koza Chair, Michigan State University
East Lansing, MI, USA

Erik Goodman
BEACON Center
Michigan State University
East Lansing, MI, USA

Leigh Sheneman
Department of Computer Science and Engineering
Michigan State University
Okemos, MI, USA

Leonardo Trujillo
Depto Ingenieria en Electronic Electrica
Tecnológico Nacional de México/
IT de Tijuana
Baja California
Tijuana, Mexico

Bill Worzel
Evolution Enterprises
Ann Arbor, MI, USA

ISSN 1932-0167 ISSN 1932-0175 (electronic)
Genetic and Evolutionary Computation
ISBN 978-3-030-39957-3 ISBN 978-3-030-39958-0 (eBook)
https://doi.org/10.1007/978-3-030-39958-0

This Springer imprint is published by the registered company Springer Nature Switzerland AG.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

We dedicate this book to the memory of the co-founder of the Workshop series on Genetic Programming—Theory and Practice, Rick Riolo, who passed away on August 25, 2018.

Foreword

It is a genuine pleasure to write this brief foreword to the collected proceedings of GPTP XVII. It was my privilege to act as opening keynote speaker at the gathering, returning after a 16-year break from playing the same role for GPTP I in 2003. In both cases, I was a fascinated outsider learning about a community that seemed at once oddly similar and yet weirdly different from the computational evolutionary biologists who comprise my own academic tribe (specifically those concerned with the origin and early evolution of life).

On both occasions, I was struck immediately by the potential for the Genetic Programming Theory and Practice (GPTP) community to answer questions that "my people" struggle to frame. How and why did the computational basis of biology evolve to comprise the particular set of rules and pieces which freshmen biologists now strive to memorize, some four billion years later (4 genetic letters, 20 amino acid building blocks of proteins and their interactions)?

But this year, just as in 2003, careful listening soon brought a far deeper conviction that the questions of evolutionary computing are not and should not be limited to those which happen to interest me, or indeed anyone else. There is something too fresh, vibrant, and exploratory about the border formed by introducing evolutionary principles into programming. The diverse works which follow will grow, within the reader, an inescapable sense that it would be to the detriment of human knowledge and technological progress for anyone to presume, at this early stage, any particular purpose or direction for the field. There's simply too much exploration to be done first!

This truth only highlights a more urgent and somber note which must rightfully dominate my remaining words. While it would be nice to write here only a tourist's guide to the series of locations along the border between evolution and computing which populate the following pages, something far more serious dominated the gathering and must be spoken openly. When I, a nosy outsider, asked participants to bring me up to speed on the history of their field "while I was away," one message united all answers: Deep Learning has emerged to pose a deep and perhaps existential threat to our community. The numerous directions in which this particular form of neural network can find answers are undeniable. Equally undeniable is the attractiveness of a simple, reliable, and user-friendly product developed by the financial might and business acumen of Google. But just as, at least within the USA, the emergence of "big box" stores brought reliability, cost savings, and convenience only at the cost of conformity which eroded a far richer consumer ecosystem, so it is very clear from the pages that follow that Deep Learning is flattening something far richer.

Both implicitly and explicitly, the pages which follow demonstrate that Deep Learning is not the answer to every problem. From industry to computing theory, genetic programming and genetic algorithms can help where neural networks and other forms of machine learning struggle. A subtler, deeper message to be found between their lines is one familiar throughout research science. Surprisingly often, it turns out that an answer to the question, as originally posed, is downright unhelpful. We needed, instead, to understand why the question was badly framed. That need not be expressed in the past tense. Any history of science suggests that we progress less by obtaining answers than by forming better questions. Douglas Adams satirized this important truth famously within the Hitchhiker's Guide to the Galaxy when he told the story of an unimaginably advanced civilization which built planet Earth as a supercomputer with which to calculate the answer to life in the universe and everything. Only when this answer arrived in the form of the number 42 did the civilization reflect that perhaps the question had not been well formed.

The truth behind this humor matters when a core limitation of Deep Learning is its lack of transparency. What just happened? How did it reach that answer? Is that really what we needed to know/solve/achieve? In contrast to the black ("big") box of Deep Learning, the diverse "Mom-and-Pop" stores of the GPTP community invite such meta-questions. Through them, we have every reason to believe, a deeper kind of learning proceeds. Let us not wait for Deep Learning to produce all of the answers, only to discover that we now need to dust off, resurrect, or reinvent alternative approaches that it drove extinct along the way. It matters, then, that the community of evolutionary computing spreads this message: through its areas of success and the unexpected insights it uncovers. And if you, the reader, are in any way new to the field represented by GPTP then it matters that you keep reading.

Baltimore, MD, USA
October 2019

Stephen Freeland

Preface

After 16 annual editions of the workshop on Genetic Programming Theory and Practice (GPTP) were held in Ann Arbor, 2019, we saw the workshop venturing out from that location for the first time. This 17th GPTP workshop was held in East Lansing, Michigan, from May 16 to May 19, 2019, at Michigan State University, one of the first land-grant institutions in the USA. It was organized and supported by the NSF-funded BEACON Center for the Study of Evolution in Action, a Science and Technology Center funded by the NSF since 2010.

The collection you hold in hand contains the written final contributions submitted by the workshop's participants. Each contribution was drafted, read, and reviewed by other participants prior to the workshop. Each was then presented at the workshop, and subsequently revised, after the workshop, on the basis of feedback received during the event.

GPTP has long held a special place in the genetic programming community, as an unusually intimate, interdisciplinary, and constructive meeting. It brings together researchers and practitioners who are eager to engage with one another deeply, in thoughtful, unhurried discussions of the major challenges and opportunities in the field. Despite the change in location, the large group of interested individuals at MSU this year resulted in one of the largest groups ever participating in the workshop with approximately 50 regular attendees.

It should be kept in mind that participation at this workshop is by invitation only, and every year the editors make an effort to invite a group of participants that is diverse in several ways, including participants both from academia and industry, junior and senior, local, national, and international. Efforts are also made to include participants in "adjacent" fields such as evolutionary biology.

GPTP is a single-track workshop, with a schedule that provides ample time for presentations and for discussions, both in response to specific presentations and on more general topics. Participants are encouraged to contribute observations from their own, unique perspectives, and to help one another to engage with the presented work. Often, new ideas are developed in these discussions, leading to collaborations after the workshop.

In this year's edition, the regular talks touched on many of the most important issues and research questions in the field, including: opportune application domains for GP-based methods, game playing and co-evolutionary search, symbolic regression and efficient learning strategies, encodings and representations for GP, schema theorems, and new selection mechanisms.

Aside from the presentations of regular contributions, the workshop featured three keynote presentations that were chosen to broaden the group's perspective on the theory and practice of genetic programming. This year, the first keynote speaker was Dr. Stephen Freeland, University of Maryland, on "Alphabets, topologies and optimization." He returned to the workshop after giving a keynote at the first GPTP workshop in 2003, with 15 years of additional research to report on. On the second day, the keynote was presented by Gavin A. Schmidt from the NASA Goddard Institute for Space Studies, on "Some Challenges and Progress in Programming for Climate Science." The third and final keynote was delivered by Indika Rajapakse Associate Professor of Computational Medicine and Bioinformatics, Mathematics and Bioengineering at the University of Michigan in Ann Arbor, on "Cell Reprogramming." As can be gathered from their titles, none of these talks focused explicitly on genetic programming per se. But each presented fascinating developments that connect to the theory and applications of genetic programming in intriguing and possibly influential ways.

While most readers of this volume will not have had the pleasure of attending the workshop itself, our hope is that they will nonetheless be able to appreciate and engage with the ideas that were presented. We also hope that all readers will gain an understanding of the current state of the field, and that those who seek to do so will be able to use the work presented herein to advance their own work, and to make additional contributions to the field in the future.

Acknowledgements

We would like to thank all of the participants for again making GP Theory and Practice a successful workshop 2019. As is always the case, it produced a lot of interesting and high-energy discussions, as well as speculative thoughts and new ideas for further work. The keynote speakers delivered thought-provoking talks from perspectives not usually directly connected to genetic programming.

We would also like to thank our financial supporters for making the existence of GP Theory and Practice possible for the past 16 years. For 2019, as we moved to another location, we needed additional funds raised from different sponsors. We are grateful to the following sponsors:

- John Koza
- Jason H. Moore
- Babak Hodjat at Sentient
- Mark Kotanchek at Evolved Analytics

- Stuart Card
- The BEACON Center for the Study of Evolution in Action, at MSU

A number of people made key contributions to the organization and assisted our participants during their stay in East Lansing. Foremost among them is Constance James, who made the workshop run smoothly with her diligent efforts behind the scenes before, during, and after the workshop. Special thanks go to Michigan State University, particularly the College of Engineering and its Dean, Professor Leo Kempel, for hosting us in the Engineering Conference room, as well as to the Springer Nature Publishing Company, for producing this book. We are particularly grateful for contractual assistance by Melissa Fearon and Ronan Nugent at Springer.

We would also like to express our gratitude to Carl Simon at the Center for the Study of Complex Systems at the University of Michigan for continued support.

East Lansing, MI, USA	Wolfgang Banzhaf
East Lansing, MI, USA	Erik Goodman
Okemos, MI, USA	Leigh Sheneman
Tijuana, Mexico	Leonardo Trujillo
Ann Arbor, MI, USA	Bill Worzel
October 2019	

Contents

1 Characterizing the Effects of Random Subsampling on Lexicase Selection 1
Austin J. Ferguson, Jose Guadalupe Hernandez, Daniel Junghans, Alexander Lalejini, Emily Dolson, and Charles Ofria

1.1 Introduction 1
1.2 Lexicase Selection 3
1.2.1 Applying Subsampling to Lexicase Selection 3
1.3 Methods 4
1.3.1 Evolutionary System 4
1.3.2 Program Synthesis Problems 5
1.3.3 Experimental Design 6
1.3.4 Statistical Analyses 10
1.4 Results and Discussion 10
1.4.1 Subsampling Improves Lexicase Selection's Problem-Solving Success 10
1.4.2 Deeper Evolutionary Searches Contribute to Subsampling's Success 12
1.4.3 Subsampling Reduces Computational Effort 13
1.4.4 Subsampling Does Not Systematically Decrease Phenotypic Diversity in Lexicase Selection 14
1.4.5 Cohort Lexicase Enables More Phylogenetic Diversity Than Down-Sampled Lexicase 15
1.4.6 Subsampling Degrades Specialist Maintenance 18
1.5 Conclusion 20
References 21

2 It Is Time for New Perspectives on How to Fight Bloat in GP 25
Francisco Fernández de Vega, Gustavo Olague, Francisco Chávez, Daniel Lanza, Wolfgang Banzhaf, and Erik Goodman

2.1 Introduction 25
2.2 The Bloat Phenomenon 26

2.3 Load-Balancing and Parallel GP 27
2.3.1 Structural Complexity of GP Individuals 28
2.4 Methodology 29
2.4.1 Implementation 31
2.4.2 Experiments 32
2.5 Results 33
2.5.1 Parallel Model 33
2.5.2 Sequential Execution 34
2.6 Conclusions 36
References 37

3 Explorations of the Semantic Learning Machine Neuroevolution Algorithm: Dynamic Training Data Use, Ensemble Construction Methods, and Deep Learning Perspectives 39
Ivo Gonçalves, Marta Seca, and Mauro Castelli
3.1 Introduction 39
3.2 Neuroevolution Overview 40
3.3 Semantic Learning Machine 45
3.3.1 Algorithm 45
3.3.2 Previous Comparisons with Other Neuroevolution Methods 46
3.4 Experimental Methodology 47
3.4.1 Datasets and Parameter Tuning 47
3.4.2 SLM Variants 48
3.4.3 MLP Variants 49
3.5 Results and Analysis 50
3.5.1 SLM 50
3.5.2 MLP 52
3.5.3 Generalization and Ensemble Analysis 54
3.6 Toward the Deep Semantic Learning Machine 58
References 59

4 Can Genetic Programming Perform Explainable Machine Learning for Bioinformatics? 63
Ting Hu
4.1 Introduction 63
4.2 Methods 64
4.2.1 Metabolomics Data for Osteoarthritis 64
4.2.2 Linear Genetic Programming Algorithm 65
4.2.3 Training Using the Full and the Focused Feature Sets 67
4.2.4 Feature Synergy Analysis 67
4.3 Results and Discussion 68
4.3.1 Best Genetic Programs Evolved on the Full Feature Set 68
4.3.2 Identification of Important Features 69

4.3.3 Best Genetic Programs Evolved on the Focused Feature Subset ... 73
4.4 Conclusion ... 75
References ... 76

5 Symbolic Regression by Exhaustive Search: Reducing the Search Space Using Syntactical Constraints and Efficient Semantic Structure Deduplication ... 79
Lukas Kammerer, Gabriel Kronberger, Bogdan Burlacu, Stephan M. Winkler, Michael Kommenda, and Michael Affenzeller
5.1 Introduction ... 79
5.1.1 Motivation ... 80
5.1.2 Prior Work ... 80
5.1.3 Organization of This Chapter ... 81
5.2 Definition of the Search Space ... 81
5.2.1 Grammar for Mathematical Expressions ... 82
5.2.2 Expression Hashing ... 85
5.3 Exploring the Search Space ... 87
5.3.1 Symbolic Regression as Graph Search Problem ... 88
5.3.2 Guiding the Search ... 89
5.4 Steering the Search ... 90
5.4.1 Quality Estimation ... 90
5.4.2 Priority Calculation ... 91
5.5 Experiments ... 92
5.5.1 Results ... 93
5.6 Discussion ... 95
5.6.1 Limitations ... 96
5.7 Outlook ... 96
References ... 97

6 Temporal Memory Sharing in Visual Reinforcement Learning ... 101
Stephen Kelly and Wolfgang Banzhaf
6.1 Introduction ... 101
6.2 Background ... 102
6.2.1 Temporal Memory ... 103
6.2.2 Heterogeneous Policies and Modularity ... 105
6.3 Evolving Heterogeneous Tangled Program Graphs ... 105
6.3.1 Programs and Shared Temporal Memory ... 106
6.3.2 Cooperative Decision-Making with Teams of Programs ... 108
6.3.3 Compositional Evolution of Tangled Program Graphs ... 109
6.4 Empirical Study ... 110
6.4.1 Problem Environments ... 111
6.4.2 Ball Catching: Training Performance ... 112
6.4.3 Ball Catching: Solution Analysis ... 114

6.4.4 Atari Breakout ... 114
6.5 Conclusions and Future Work ... 116
References ... 117

7 The Evolution of Representations in Genetic Programming Trees ... 121
Douglas Kirkpatrick and Arend Hintze
7.1 Introduction ... 121
7.2 Material and Methods ... 124
7.2.1 Representations and the Neuro-Correlate *R* ... 124
7.2.2 Smearedness of Representations ... 126
7.2.3 Active Categorical Perception Task ... 127
7.2.4 Number Discrimination Task ... 127
7.2.5 The Perception-Action Loop for Stateful Machines ... 128
7.2.6 Markov GP Brains Using CGP Nodes ... 129
7.2.7 Genetic Encoding of GP Brains in a Tree-Like Fashion ... 130
7.2.8 GP-Forest Brain ... 131
7.2.9 GP-Vector Brain ... 133
7.2.10 Evolutionary Process ... 136
7.2.11 Augmenting with R ... 136
7.3 Results ... 137
7.3.1 GP Trees Evolve to Have Representations ... 137
7.3.2 Does Augmentation Using *R* Improve the Performance of a GA? ... 138
7.3.3 Smeared Representations ... 139
7.4 Discussion ... 141
7.5 Conclusions ... 141
References ... 142

8 How Competitive Is Genetic Programming in Business Data Science Applications? ... 145
Arthur Kordon, Theresa Kotanchek, and Mark Kotanchek
8.1 Introduction ... 145
8.2 Business Needs for Data Science ... 146
8.2.1 Business Forecasting ... 146
8.2.2 Effective Operation ... 147
8.2.3 Growth Opportunities ... 148
8.2.4 Multi-Objective Optimization and Decision Making ... 149
8.3 Data Science Competitive Landscape ... 149
8.3.1 Defining Key Competitors for Data Science Applications ... 149
8.3.2 Comparison on Business Needs Satisfaction ... 150
8.3.3 How Popular Is GP in the Data Science Community? ... 150

8.4 Current State-of-the-Art of Genetic Programming as Business Application Method ... 151
8.4.1 Competitive Advantages of GP ... 151
8.4.2 Key Weaknesses of GP ... 153
8.4.3 Successful Genetic Programming Applications ... 154
8.5 How to Increase Competitive Impact of Genetic Programming in Data Science Applications? ... 157
8.5.1 Develop a Successful Marketing Strategy ... 157
8.5.2 Broaden Application Areas ... 160
8.5.3 Improved Professional Development Tools ... 160
8.5.4 Increase GP Visibility and Teaching in Data Science Classes ... 160
8.6 Conclusions ... 161
References ... 162

9 Using Modularity Metrics as Design Features to Guide Evolution in Genetic Programming ... 165
Anil Kumar Saini and Lee Spector
9.1 Introduction ... 165
9.2 Modularity in Genetic Programming ... 166
9.3 Modularity Metrics ... 167
9.3.1 Module ... 168
9.3.2 Design Principles for Modularity Metrics ... 168
9.3.3 Reuse and Repetition ... 169
9.3.4 Reuse and Repetition from Execution Trace ... 169
9.4 Using Modularity Metrics to Guide Evolution ... 171
9.4.1 Using Design Features During Parent Selection ... 172
9.4.2 Using Design Features During Variation ... 172
9.5 Experiments and Results ... 173
9.5.1 Extracting Modules from Push Programs ... 173
9.5.2 Autosimplification ... 175
9.5.3 Experimental Set-up and Results ... 175
9.6 Conclusions and Future Work ... 178
References ... 179

10 Evolutionary Computation and AI Safety ... 181
Joel Lehman
10.1 Introduction ... 181
10.2 Background ... 183
10.2.1 AI Safety ... 183
10.2.2 EC and the Real World ... 185
10.3 EC and Concrete AI Safety Problems ... 187
10.3.1 Avoiding Negative Side Effects ... 187
10.3.2 Reward Hacking ... 188
10.3.3 Scalable Oversight ... 190
10.3.4 Safe Exploration ... 191

10.3.5 Robustness to Distributional Drift 193
10.4 Discussion 194
10.5 Conclusion 196
References 196

11 Genetic Programming Symbolic Regression: What Is the Prior on the Prediction? 201
Miguel Nicolau and James McDermott
11.1 Introduction 201
11.2 Motivation 203
11.2.1 Distribution Mismatch, Problem Difficulty, and Performance 203
11.2.2 Algorithm Configuration 204
11.2.3 Understanding the Behaviour of Search Operators 205
11.3 Previous Work on GP Biases 205
11.4 Methodology, Experiments, and Results 206
11.4.1 Reasoning from First Principles 206
11.4.2 Setup 207
11.4.3 Initialisation Prior 207
11.4.4 GPSR Prior 209
11.4.5 Effect of Tree Depth on Initialisation Prior 210
11.4.6 Effect of Problem Dimension on Initialisation Prior 211
11.4.7 Effect of X Range on Initialisation Prior 212
11.4.8 Comparing the y and $\hat{y}$ Distributions Across Problems 213
11.5 Applications 215
11.5.1 Algorithm Behaviour and Performance 215
11.5.2 Algorithm Configuration 216
11.5.3 Understanding GSGP Mutation 217
11.6 Conclusions 219
11.6.1 Limitations and Future Work 220
References 223

12 Hands-on Artificial Evolution Through Brain Programming 227
Gustavo Olague and Mariana Chan-Ley
12.1 Introduction 227
12.2 Evolution of Visual Attention Programs 228
12.2.1 Evolution of Visual Recognition Programs 229
12.3 Problem Statement 231
12.4 Classification of Digitized Art 232
12.5 Experiments 237
12.5.1 Beyond Random Search in Genetic Programming 239
12.5.2 Ideas for a New Kind of Evolutionary Learning 241
12.5.3 Running the Algorithm with Fewer Images 242
12.5.4 Running the Algorithm with 100 Images 244
12.5.5 Ensemble Techniques and Genetic Programming 246

12.6 Conclusions 251
References 251

13 Comparison of Linear Genome Representations for Software Synthesis 255
Edward Pantridge, Thomas Helmuth, and Lee Spector
13.1 Introduction 255
13.2 Linear Genomes: Plush vs. Plushy 256
13.2.1 Random Genome Generation 259
13.2.2 Genetic Operators 260
13.3 Impact on Search Performance 260
13.3.1 Benchmarks 260
13.3.2 Benchmark Results 262
13.4 Genome and Program Structure 262
13.4.1 Sizes 262
13.4.2 Presence of "Closing" Genes 265
13.5 Other Considerations 268
13.5.1 Hyperparameter Fitting 268
13.5.2 Applicable Search Methods 269
13.5.3 Automatic Simplification 270
13.5.4 Serialization 271
13.5.5 New Epigenetic Markers for Plush 271
13.6 Conclusion 272
References 272

14 Enhanced Optimization with Composite Objectives and Novelty Pulsation 275
Hormoz Shahrzad, Babak Hodjat, Camille Dollé, Andrei Denissov, Simon Lau, Donn Goodhew, Justin Dyer, and Risto Miikkulainen
14.1 Introduction 275
14.2 Background and Related Work 276
14.2.1 Single-Objective Optimization 276
14.2.2 Multi-Objective Optimization 277
14.2.3 Novelty Search 278
14.2.4 Exploration Versus Exploitation 278
14.2.5 Sorting Networks 279
14.2.6 Stock Trading 280
14.3 Methods 281
14.3.1 Representation 281
14.3.2 Single-Objective Approach 281
14.3.3 Multi-Objective Approach 282
14.3.4 Composite Multi-Objective Approach 282
14.3.5 Novelty Selection Method 283
14.3.6 Novelty Pulsation Method 285
14.4 Experiment 286
14.4.1 Experimental Setup 286

14.4.2 Sorting Networks Results ... 287
14.4.3 Stock Trading Results ... 288
14.5 Discussion and Future Work ... 289
14.6 Conclusion ... 290
References ... 291

15 New Pathways in Coevolutionary Computation ... 295
Moshe Sipper, Jason H. Moore, and Ryan J. Urbanowicz
15.1 Coevolutionary Computation ... 295
15.2 OMNIREP ... 297
15.3 SAFE ... 299
15.4 Concluding Remarks ... 303
References ... 304

16 2019 Evolutionary Algorithms Review ... 307
Andrew N. Sloss and Steven Gustafson
16.1 Preface ... 307
16.2 Introduction ... 310
16.2.1 Applications ... 312
16.3 Fundamentals of Digital Evolution ... 313
16.3.1 Population ... 314
16.3.2 Population Entities ... 315
16.3.3 Generation ... 315
16.3.4 Representation and the Grammar ... 316
16.3.5 Fitness ... 316
16.3.6 Selection ... 317
16.3.7 Multi-Objective ... 317
16.3.8 Constraints ... 318
16.3.9 Exploitative-Exploratory Search ... 318
16.3.10 Execution Environment, Modularity and System Scale ... 318
16.3.11 Code Bloat and Clean-Up ... 319
16.3.12 Non-convergence, or Early Local Optima ... 319
16.3.13 Other Useful Terms ... 320
16.4 Traditional Techniques ... 321
16.4.1 Evolutionary Strategy, ES ... 321
16.4.2 Genetic Algorithms, GA ... 322
16.4.3 Genetic Programming, GP ... 322
16.4.4 Genetic Improvement, GI ... 323
16.4.5 Grammatical Evolution, GE ... 323
16.4.6 Linear Genetic Programming, LGP ... 324
16.4.7 Cartesian Genetic Programming, CGP ... 324
16.4.8 Differential Evolution, DE ... 324
16.4.9 Gene Expression Programming, GEP ... 325
16.5 Specialized Techniques and Concepts ... 325
16.5.1 Auto-Constructive Evolution ... 325

16.5.2 Neuroevolution, or Deep Neuroevolution 326
16.5.3 Self-Replicating Neural Networks 327
16.5.4 Markov Brains 327
16.5.5 PushGP 328
16.5.6 Simulated Annealing 328
16.5.7 Tangled Program Graph, TPG............................. 328
16.5.8 Tabu Search 329
16.5.9 Animal Inspired Algorithms............................. 329
16.6 Problem-Domain Mapping 329
16.6.1 Specific Problem-Domain Mappings 330
16.6.2 Unusual and Interesting Problem-Domain Mappings 333
16.7 Challenges.. 335
16.8 Predictions 337
16.9 Final Discussion and Conclusion 338
16.10 Feedback 340
References.. 340

17 Evolving a Dota 2 Hero Bot with a Probabilistic Shared Memory Model.......... 345
Robert J. Smith and Malcolm I. Heywood
17.1 Introduction 345
17.2 The Dota 2 1-on-1 Mid-lane Task............................. 347
17.3 Related Work.. 348
17.3.1 Memory in Neural Networks.............................. 348
17.3.2 Memory in Genetic Programming 350
17.4 Tangled Program Graphs 351
17.5 Indexed Memory for TPG 353
17.6 Dota 2 Game Engine Interface 354
17.6.1 Developing the Dota 2 Interface 354
17.6.2 Defining State Space................................... 356
17.6.3 Defining the Shadow Fiend Action Space 357
17.6.4 Fitness Function 357
17.7 Results... 358
17.7.1 TPG Set Up 358
17.7.2 Training Performance................................... 359
17.7.3 Assessing Champion TPG Agents Post Training............ 361
17.7.4 Characterization of Memory Behaviour 362
17.8 Conclusion 363
References.. 364

18 Modelling Genetic Programming as a Simple Sampling Algorithm .. 367
David R. White, Benjamin Fowler, Wolfgang Banzhaf, and Earl T. Barr
18.1 Introduction 367
18.2 Rationale for Modelling Simple Schemata 369

18.3 Modelling GP ... 371
18.3.1 Change in Schema Prevalence Due to Selection ... 371
18.3.2 Change in Schema Prevalence Due to Operators ... 372
18.4 Empirical Data Supporting the Model ... 373
18.5 Ways to Improve GP ... 379
18.6 Related Work ... 380
18.7 Conclusion ... 380
References ... 381

19 An Evolutionary System for Better Automatic Software Repair ... 383
Yuan Yuan and Wolfgang Banzhaf
19.1 Introduction ... 383
19.2 Background and Motivation ... 385
19.2.1 Related Work ... 385
19.2.2 Motivating Examples ... 386
19.3 Overview of ARJA-e ... 387
19.4 Shaping the Search Space ... 387
19.4.1 Exploiting the Statement-Level Redundancy Assumption ... 387
19.4.2 Exploiting Repair Templates ... 388
19.4.3 Initialization of Operation Types ... 390
19.5 Multi-Objective Evolution of Patches ... 391
19.5.1 Patch Representation ... 391
19.5.2 Finer-Grained Fitness Function ... 391
19.5.3 Genetic Operators ... 392
19.5.4 Multi-Objective Search ... 393
19.6 Alleviating Patch Overfitting ... 393
19.6.1 Overfit Detection ... 393
19.6.2 Patch Ranking ... 395
19.7 Experimental Design ... 395
19.7.1 Research Questions ... 395
19.7.2 Dataset of Bugs ... 396
19.7.3 Parameter Setting ... 396
19.8 Results and Discussions ... 397
19.8.1 Performance Evaluation (RQ1) ... 397
19.8.2 Novelty in Generated Repairs (RQ2) ... 398
19.8.3 Effectiveness of Overfit Detection (RQ3) ... 400
19.9 Conclusion ... 402
References ... 403

Index ... 407

Contributors

Michael Affenzeller Heuristic and Evolutionary Algorithms Laboratory (HEAL), University of Applied Sciences Upper Austria, Hagenberg, Austria
Department of Computer Science, Johannes Kepler University, Linz, Austria

Wolfgang Banzhaf Department of Computer Science and Engineering & Beacon Center, Michigan State University, East Lansing, MI, USA

Earl T. Barr CREST, University College London, London, UK

Bogdan Burlacu Heuristic and Evolutionary Algorithms Laboratory (HEAL), University of Applied Sciences Upper Austria, Hagenberg, Austria
Josef Ressel Center for Symbolic Regression, University of Applied Sciences Upper Austria, Hagenberg, Austria

Mauro Castelli NOVA IMS, Universidade Nova de Lisboa, Lisboa, Portugal

Mariana Chan-Ley EvoVisión Laboratory, Ensenada, BC, Mexico

Francisco Chávez University of Extremadura, Badajoz, Spain

Andrei Denissov Sentient Investment Management, San Francisco, CA, USA

Camille Dollé Sentient Investment Management, San Francisco, CA, USA

Emily Dolson Department of Translational Hematology and Oncology Research, Cleveland Clinic, Cleveland, OH, USA

Justin Dyer Sentient Investment Management, San Francisco, CA, USA

Austin J. Ferguson The BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI, USA

Francisco Fernández de Vega University of Extremadura, Badajoz, Spain

Benjamin Fowler Department of Computer Science, Memorial University of Newfoundland, St. John's, NL, Canada

Ivo Gonçalves INESC Coimbra, DEEC, University of Coimbra, Coimbra, Portugal

Donn Goodhew Sentient Investment Management, San Francisco, CA, USA

Erik Goodman BEACON Center, Michigan State University, East Lansing, MI, USA

Steven Gustafson MAANA Inc., Bellevue, WA, USA

Thomas Helmuth Hamilton College, Clinton, NY, USA

Jose Guadalupe Hernandez The BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI, USA

Malcolm I. Heywood Faculty of Computer Science, Dalhousie University, Halifax, NS, Canada

Arend Hintze Department of Integrative Biology, Michigan State University, East Lansing, MI, USA
Department of Computer Science and Engineering, BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI, USA

Babak Hodjat Cognizant Technology Solutions, Dublin, CA, USA

Ting Hu School of Computing, Queen's University, Kingston, ON, Canada
Department of Computer Science, Memorial University, St. John's, NL, Canada

Daniel Junghans The BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI, USA

Lukas Kammerer Heuristic and Evolutionary Algorithms Laboratory (HEAL), University of Applied Sciences Upper Austria, Hagenberg, Austria
Department of Computer Science, Johannes Kepler University, Linz, Austria
Josef Ressel Center for Symbolic Regression, University of Applied Sciences Upper Austria, Hagenberg, Austria

Stephen Kelly Department of Computer Science and Engineering & Beacon Center, Michigan State University, East Lansing, MI, USA

Douglas Kirkpatrick Department of Computer Science and Engineering, BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI, USA

Michael Kommenda Heuristic and Evolutionary Algorithms Laboratory (HEAL), University of Applied Sciences Upper Austria, Hagenberg, Austria
Josef Ressel Center for Symbolic Regression, University of Applied Sciences Upper Austria, Hagenberg, Austria

Arthur Kordon Kordon Consulting LLC, Fort Lauderdale, FL, USA

Theresa Kotanchek Evolved Analytics LLC, Midland, MI, USA

Mark Kotanchek Evolved Analytics LLC, Midland, MI, USA

Gabriel Kronberger Heuristic and Evolutionary Algorithms Laboratory (HEAL), University of Applied Sciences Upper Austria, Hagenberg, Austria
Josef Ressel Center for Symbolic Regression, University of Applied Sciences Upper Austria, Hagenberg, Austria

Alexander Lalejini The BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI, USA

Daniel Lanza University of Extremadura, Badajoz, Spain

Simon Lau Sentient Investment Management, San Francisco, CA, USA

Joel Lehman Uber AI, San Francisco, CA, USA

James McDermott National University of Ireland, Galway, Ireland

Risto Miikkulainen Cognizant Technology Solutions, Dublin, TX, USA
The University of Texas at Austin, Austin, CA, USA

Jason H. Moore Institute for Biomedical Informatics, University of Pennsylvania Philadelphia, PA, USA

Miguel Nicolau University College Dublin, Quinn School of Business, Belfield, Dublin, Ireland

Charles Ofria The BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI, USA

Gustavo Olague CICESE, Ensenada, BC, Mexico

Edward Pantridge Swoop, Inc., Cambridge, MA, USA

Anil Kumar Saini College of Information and Computer Sciences, University of Massachusetts, Amherst, MA, USA

Marta Seca NOVA IMS, Universidade Nova de Lisboa, Lisboa, Portugal

Hormoz Shahrzad Cognizant Technology Solutions, Dublin, CA, USA

Moshe Sipper Institute for Biomedical Informatics, University of Pennsylvania, Philadelphia, PA, USA
Department of Computer Science, Ben-Gurion University, Beer Sheva, Israel

Andrew N. Sloss Arm Inc., Bellevue, WA, USA

Robert J. Smith Faculty of Computer Science, Dalhousie University, Halifax, NS, Canada

Lee Spector Department of Computer Science, Amherst College, Amherst, MA, USA
School of Cognitive Science, Hampshire College, Amherst, MA, USA

College of Information and Computer Sciences, University of Massachusetts, Amherst, MA, USA

Ryan J. Urbanowicz Institute for Biomedical Informatics, University of Pennsylvania, Philadelphia, PA, USA

David R. White Department of Physics, University of Sheffield, Sheffield, UK

Stephan M. Winkler Heuristic and Evolutionary Algorithms Laboratory (HEAL), University of Applied Sciences Upper Austria, Hagenberg, Austria
Department of Computer Science, Johannes Kepler University, Linz, Austria

Yuan Yuan Department of Computer Science and Engineering & Beacon Center, Michigan State University, East Lansing, MI, USA