

Genetic Programming IV

Routine Human-Competitive Machine Intelligence

by

John R. Koza

Stanford University, CA

Martin A. Keane

Econometrics Inc., IL

Matthew J. Streeter

Genetic Programming, Inc., CA

William Mydlowec

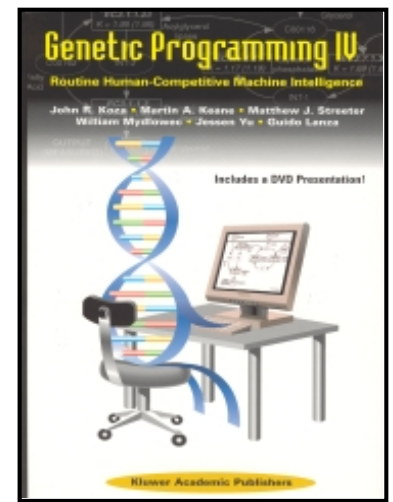
Pharmix Corporation, CA

Jessen Yu

Pharmix Corporation, CA

Guido Lanza

Pharmix Corporation, CA



25% discount off list price!

Genetic programming (GP) is a method for automatically creating computer programs. It starts from a high-level statement of what needs to be done and uses the Darwinian principle of natural selection to breed a population of improving programs over many generations.

Genetic Programming IV: Routine Human-Competitive Machine Intelligence presents the application of GP to a wide variety of problems involving automated synthesis of controllers, circuits, antennas, genetic networks, and metabolic pathways. The book describes fifteen instances where GP has created an entity that either infringes or duplicates the functionality of a previously patented 20th-century invention, six instances where it has done the same with respect to post-2000 patented inventions, two instances where GP has created a patentable new invention, and thirteen other human-competitive results. The book additionally establishes:

- GP now delivers routine human-competitive machine intelligence.
- GP is an automated invention machine.
- GP can create general solutions to problems in the form of parameterized topologies.
- GP has delivered qualitatively more substantial results in synchrony with the relentless iteration of Moore's Law.

Reviews:

"*Genetic Programming IV: Routine Human-Competitive Machine Intelligence* demonstrates the everyday solution of such 'holy grail' problems as the automatic synthesis of analog circuits, the design of automatic controllers, and the automated programming of computers. To specialists in any of the fields covered by this book's sample problem areas, I say read this book and discover the computer-augmented inventions that are your destiny. To remaining skeptics who doubt the inventive competence of genetics and evolution, I say read this book and change your mind or risk the strong possibility that your doubts will soon cause you significant intellectual embarrassment." **David E. Goldberg, University of Illinois**

"The research reported in this book is a tour de force. For the first time since the idea was bandied about in the 1940s and the early 1950s, we have a set of examples of human-competitive automatic programming." **John H. Holland, University of Michigan**

Visit our website at:

www.wkap.nl

For up-to-date information.

quality solutions more effectively. Genetic Programming III: Darwinian Invention and Problem. Solvingâ€”J. R. Koza, F. H Bennett III, D. Andre, and M. A. Keane, Eds.Â Method: The proposed genetic programming algorithm is focused in the induction of mathematical equations, running the process called symbolic regression on data from four male children. Results: It was observed that the generated curve becomes closer of growth curve in the last training points, so with a more number of training data, better the accurate of prediction. Using Genetic Programming in Industrial Statistical Model Building. Pages 31-48. Castillo, Flor (et al.) Preview Buy Chapter 24,95 â‚¬. Population Sizing for Genetic Programming Based on Decision-Making. Pages 49-65. Sastry, Kumara (et al.)Â Using Genetic Programming to Search for Supply Chain Reordering Policies. Pages 207-223. Moore, Scott A. (et al.) Preview. Cartesian Genetic Programming and the Post Docking Filtering Problem. Pages 225-244. Garmendia-Doval, A. Beatriz (et al.) Preview.

Genetic programming is a domain-independent method that genetically breeds a population of computer programs to solve a problem. Specifically, genetic programming iteratively transforms a population of computer programs into a new generation of programs by applying analogs of naturally occurring genetic operations. This process is illustrated in Figure 5.1. In artificial intelligence, genetic programming (GP) is a technique of evolving programs, starting from a population of unfit (usually random) programs, fit for a particular task by applying operations analogous to natural genetic processes to the population of programs. It is essentially a heuristic search technique often described as 'hill climbing', i.e. searching for an optimal or at least suitable program among the space of all programs. Genetic programming is a special field of evolutionary computation that aims at building programs automatically to solve problems independently of their domain. Although there exist diverse representations used to evolve programs, the most common is the syntax tree. For example, the above figure presents the program $(\max(x + 3 * y, x + x))$. For this tree and further examples, the leaves of the tree, in green, are called terminals, while the internal nodes, in red, are called primitives. @article{OReilly1994GeneticPI, title={Genetic Programming II: Automatic Discovery of Reusable Programs.}, author={U. Oâ€™Reilly}, journal={Artificial Life}, year={1994}, volume={1}, pages={439-441} }. U. Oâ€™Reilly. Published 1994. Computer Science. Artificial Life. Reading Genetic Programming IE Automatic Discovery of Reusable Programs (GP II) in its entirety is not a task for the weak-willed because the book without appendices is about 650 pages.