An Analysis of Genetic Programming

This thesis analyzes Koza's Genetic Programming (GP) paradigm, a genetic algorithm for program discovery. In order to improve upon our understanding of GP and to improve GP, it provides a systematic analysis of GP that is based upon experimentation and theory.

We assess the role of designer expertise in successfully using GP. Our experiments show that its performance is influenced by propitious designer choices of the test suite and primitive set. We also appraise whether GP proceeds in a hierarchical manner. In experiments with the canonical earliest version of GP, GP did not appear to exploit a hierarchical process.

The theoretical analysis develops a schema-based framework for describing GP search behaviour. We formally develop a Schema Theorem for GP, define building blocks and state a GP Building Block Hypothesis. We proceed to methodically question the plausibility that GP exploits a building block process while searching.

We conduct further experimental analysis by comparing GP to alternative algorithms. A mutation-based operator, HVL-Mutate, that generates a syntactically valid and possibly structurally different program from another is introduced. Two adaptive search algorithms, Stochastic Iterated Hill Climbing and Simulated Annealing, which use either HVL-Mutate or GP crossover are implemented to solve exactly the same class of program discovery problems as GP. The resulting algorithms are comparable to GP and sometimes even outperform it on a small suite of these problems.

Because these algorithms are relatively successful at solving the same problems GP solves, we conjecture that synthesizing a localized search strategy into GP will complement its global, population-based search and improve it. Our experiments with our problem suite confirm this insight. When we hybridize GP by adding a hill climbing component, various versions of the hybrid algorithm achieve higher likelihood of success and process less candidate programs than GP.