Evolving a Sorting Algorithm with SNGP

Robin Lockyer

University of Liverpool

April 2019

Abstract

Genetic programming is a technique for creating programmes not by writing them by hand, but instead by creating a population of random programmes and modifying them using an evolutionary algorithm. The desired result is that after several generations a programme that performs well at a given task is generated. GP has previously been used to successfully evolve sorting algorithms.

Single node genetic programming is a variation on GP invented by Dr Jackson which structures the population of programmes in a manner that allows the use of dynamic programming when computing the result of the programmes in an effort to more efficiently generate a working solution.

This project aims to compare the effectiveness of the two methods in evolving a sorting algorithm.

Contents

1 Introduction

This project is being done for my project supervisor Dr David Jackson. The aim of this project is to attempt to evolve a sorting algorithm using node genetic programming (SNGP) and, if successful, compare the effectiveness of evolving sorting algorithms using standard genetic programming (GP) to evolving sorts with SNGP.

The aim of GP is to automate the creation of algorithms and programmes. This is done by applying a genetic algorithm to a population of random programmes so that successive generations of programmes improve at the desired characteristics until a functional programme is created. The standard approach to GP requires evaluating hundreds of programmes per generation over potentially thousands of generations and as such GP can take up a large amount of processing time. Several variations of GP have been created that try to reduce the amount of processing, including Linear Genetic Programming and Parallel Distributed GP [3].

SNGP is one such variation devised by Dr Jackson in *A New, Node-Focused Model for Genetic Programming* [1]. This variation makes use of a form of dynamic programming to re-use results of previously evaluated programmes. It has been shown that SNGP tends to perform better than standard GP in terms of processing time, solution rate, and solution size [1]. SNGP is not as efficient when dealing with problems with side-effects because this prevents re-use of evaluations, although it still compares favourably to GP in some problem domains [2].

2 Background

SNGP was first described by Dr Jackson in A New, Node-Focused Model for Genetic Programming [1].

3 Data Required

4 Design

5 Realisation

6 Evaluation

7 Learning Points

8 Professional Issues

9 Bibliography

References

[1] JACKSON, D. A new, node-focused model for genetic programming. In *Genetic Programming* (2012), A. Moraglio, S. Silva, K. Krawiec, P. Machado, and C. Cotta, Eds., Lecture Notes in Computer Science, Springer Berlin Heidelberg, pp. 49–60.

[2] JACKSON, D. Single node genetic programming on problems with side effects. In *Parallel Problem Solving from Nature - PPSN XII* (2012), C. A. C. Coello, V. Cutello, K. Deb, S. Forrest, G. Nicosia, and M. Pavone, Eds., Lecture Notes in Computer Science, Springer Berlin Heidelberg, pp. 327–336.

[3] POLI, R., LANGDON, W. B., AND MCPHEE, N. F. *A Field Guide to Genetic Programming.* Lulu Enterprises, UK Ltd.

10 Appendices