Long-Term Evolution of Genetic Programming Populations

William B. Langdon
Department of Computer Science, University College London, Gower Street, WC1E 6BT, UK.

ABSTRACT

Evolving binary mux-6 trees for up to 100 000 generations, during which some programs grow to more than a hundred million nodes, suggests the landscape which GP explores contains some very smooth regions. Although the GP population evolves under crossover, our unbounded GP appears not to evolve building blocks. We do see periods of tens even hundreds of generations where even although each member of the population occupies a different point in the genotypic search space, they are lie at exactly the same point in the phenotypic landscape. Phenotypic convergence whilst retaining genotypic diversity is typical of GP and, we suggest inherent in highly redundant variable length representations.

Based on technical report RN/17/05 arXiv:1703.08481

ACM Reference format:
William B. Langdon . 2017. Long-Term Evolution of Genetic Programming Populations. In *Proceedings of GECCO '17 Companion, Berlin, Germany, July 15-19, 2017,* 2 pages.
DOI: http://dx.doi.org/10.1145/3067695.3075965

1 INTRODUCTION

Rich Lenski's experiments in long term evolution in which the BEACON team evolved bacteria for more than 60 000 generations and found continued beneficial adaptive mutations, prompts the same question in computation based evolution. We report what happens if we allow genetic programming to evolve for tens of thousands, even a hundred thousand generations.

As expected, we see bloat but after a few hundred generations we start to see surprises. Tree growth is not continuous. Indeed we see many generations where the trees get smaller. As we look at the very long term distribution of fitness in Section 2.2 we do not get the whole population reaching one fitness value and everyone having that fitness in all subsequent generations.

Although introns are well known in GP, we find constants are also required to explain the observed long term evolution. We confirm the asymmetry of the GP crossover and the importance of the first parent, which gives the offspring's root node.

The fraction of constants in highly evolved trees is fairly stable despite huge fluctuations in the size of the trees containing them.

GECCO '17 Companion, Berlin, Germany
 978-1-4503-4939-0/17/07...$15.00
DOI: http://dx.doi.org/10.1145/3067695.3075965

We demonstrate the common assumption that useful code clusters around the root node and show that in the longer term this code becomes highly stable and is surrounded and is protected by many thousands of useless instructions composed of both introns and constants.

2 RESULTS

2.1 Evolution of Size: Gambler's Ruin

In all experiments we do see enormous increases in size (see Figure 1). If everyone in the population has the same fitness, selection appears to become random and children are as likely to be smaller than their parents as they are likely to be bigger. If selection is switched off for many generations, we see an apparently random walk in average tree size, with falls and rises. (We return to this in Section 3.) However there are substantial falls in size which might be due to the discovery of smaller than average individuals of max fitness but with a higher than average effective fitness. Nonetheless over thousands of generations the removal of smaller trees is sufficient to continue to bloat the population.

2.2 Fitness Convergence

As expected, the number of individuals with maximum fitness in the population grows rapidly towards 100%. However as Figure 1 shows, typically it does not remain at 100% but hovers slightly below 100%. Figure 1 also shows a general downward trend towards fitness convergence.

2.3 Evolution of Size and Shape

Even after evolving for 1000s of gens in small populations, we continue to see the impact of fitness selection on the distribution of tree sizes. And, although the distribution of tree sizes versus their depths is close to that of random trees, the distribution of tree sizes does not approach the limiting distribution Poli predicted assuming no fitness EuroGP-2007

2.4 Evolution of Effective Code

Figure 2 (top right) shows that the size of effective code is fairly stable. If we look at the effective code in every tree in the population at generation 500–4000 (Figure 2) we see they are very similar. Typically the effective part of the code lies in a few hundred nodes around the root node (yellow) which is protected against crossover by evolved constants. The constants head large sacrificial subtrees of ineffective code. Figure 2 shows the effective code is conserved over many generations. We see this in all pop=500 runs (except 102, which gets stuck at fitness=62). However the details of the evolved effective code differ from run to run.

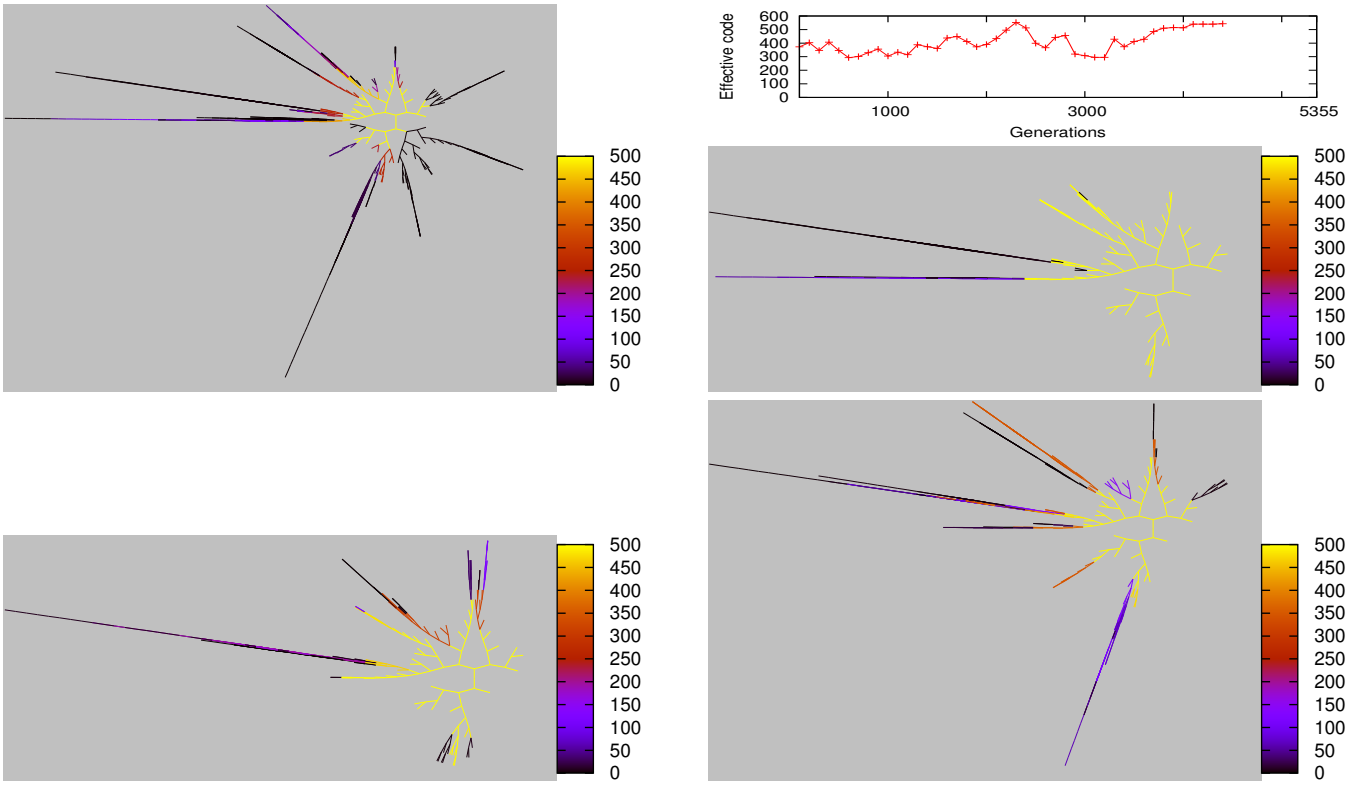

Figure 2: Effective code in a pop of 500 binary trees after 500, 1000, 2000 and 4000 gens. (Each panel showing the 496-500 trees with max fitness.) Note the similarity of the eff. code even though separated by 1000s of gens. By gen 4000 although mean size is 105 880, eff. code is limited to the 721 nodes shown. Almost all the population have 82 eff. nodes in common (yellow). Darker colours indicate eff. code which only occurs in $\leq$148 (blue) or $\leq$22 (black) trees. Code which does not effect any program's output is not plotted.

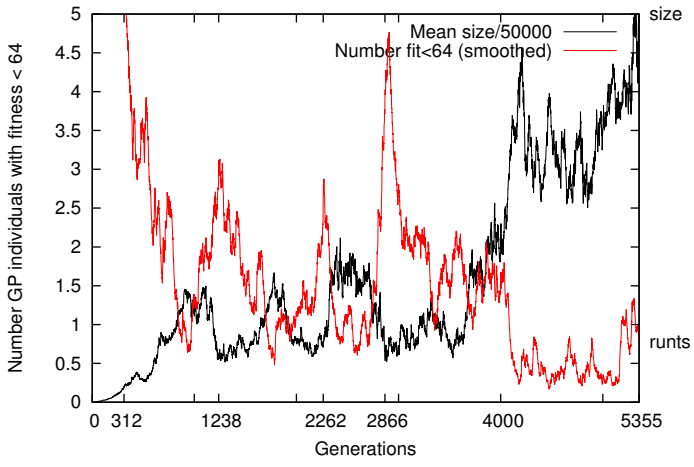

Figure 1: Evolution of program size (black) in 6-mux pop=500 run. Run stops after gen 5355 when 1^{st} tree exceeds 10^6 (mean size 260 130). Evolution of low fitness (red). All low fit (runts) first removed in gen 312. (Plot has been smooth over 30 gens.) After gen 312, there are at most 11 trees with fitness < 64.

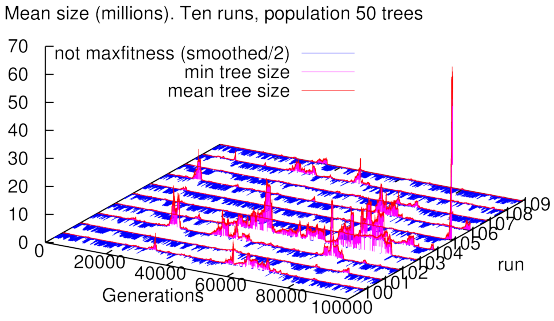

Figure 3: Ten extended population fifty runs.

3 A LIMIT TO BLOAT

When the bloated trees become so large that number of times the high fitness core near the root node is disrupted falls well below one per generation there is no driving force to grow the trees. Hence there is a balancing point near tree size $\approx$ popsize $\times$ core codesize. Averaged over ten extended runs over 100 000 generations, the median mean tree size in the 10 populations was 42 507 (see Figure 3).

GP code `http://www.cs.ucl.ac.uk/.../GPbmux6.tar.gz`