W. B. Langdon
GGGP, Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK (Email: w.langdon@cs.ucl.ac.uk)

Kin Selection with Twin Genetic Programming

1 Twin Genetic Programming

In twin GP two children are created together. They are locked together. If either is selected to be removed from the population, the other is killed at the same time. Two offspring subtree crossover is used. Note the total size of the twins is the same as the total size of their parents.

When children are created by mutation, point mutation is used (which does not change trees sizes or shapes). Again two twins are created, which are again of the same average size as their parents.

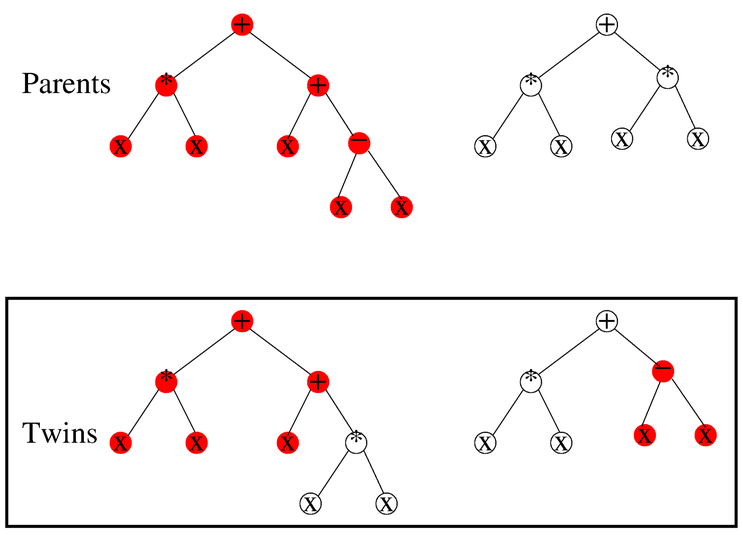

Figure 1: Subtree crossover creating two children which are bound together throughout their lives.

2 Twin Selection

Everyone in population has own fitness. They are run on all 64 test cases. Fitness is number of test cases they get right.

- ✳ Twin. Uses own fitness.
- ✳ Mean: Tournament uses average of the two twins fitnesses.
- ✳ Max. Select using best of two twins.
- ✳ Min: Use fitness of worst of twins.
- ✳ Kin: use the fitness of the other twin.

When selected for overwrite, twin is also removed from population.

3 Kin Selection

Kin selection, fitness of other twin is used. Note own performance has no influence. Although not as efficient as plain GP, kin selection (with a bigger population) can evolve solutions.

4 TinyGP Twin GP Parameters

Table 1. Twin GP parameters for solving 6-mux

Terminals:	6 Boolean inputs D0-D5
Functions:	AND, OR, NAND, NOR
Fitness:	All 64 fitness cases
Selection:	Binary tournaments used for both parent and replacement selection
Population:	1000 (or 10 000)
Initial pop:	Grow, max depth 6
Parameters	80 % subtree crossover. Both crossover points chosen at random, i.e. no function bias
	20 % point mutation. 5 % chance of substitution with primitive of the same arity per primitive. Notice mutants are subjected to zero or more flips and larger programs have proportionately more changes
	No depth or size limits
Termination	Problem is solved, or 100 generation equivalents

5 TwinGP Six Multiplexor (30 runs)

Experiment	Successful runs	
pop size	1000	10000
no twin	21	30
twin	13	29
MEAN	17	30
MAX	19	30
MIN	6	28
KIN	0	11
RAND		0

6 Twin GP bloats

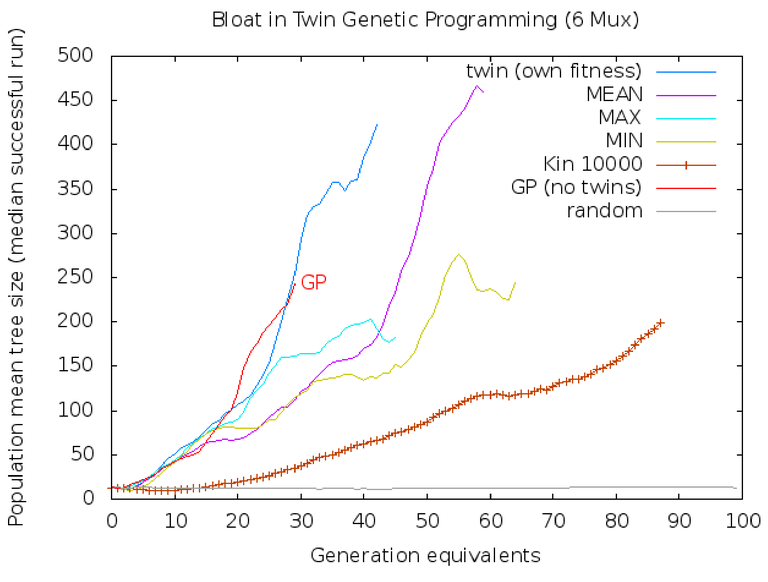

Figure 2: All five types of twin GP bloat, even when binary tournament selection selects parents and who to kill using fitness of other twin. (Bigger population needed for kin selection). Traditional GP and totally random selection shown for comparison.

7 Why Does Twin GP Bloat

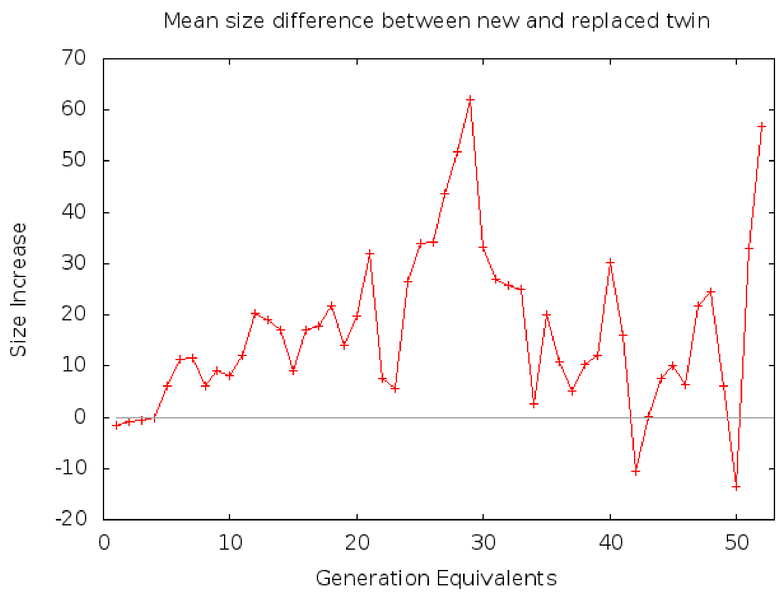

Figure 3: Twin GP runs bloat because (like genetic programming runs) on average selection removes smaller trees and keeps bigger ones.

8 Why does Kin Selection Work

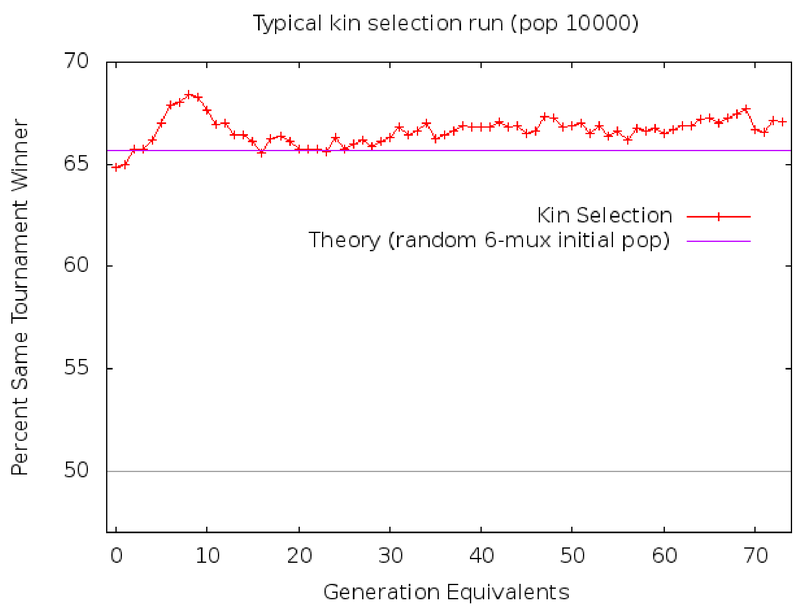

Figure 4: Kin selection works (albeit with larger population) because mostly selecting using twin's fitness gives same answer as using own fitness would have done.

9 Future: Fixed total twin size

Should we fix genetic operators so average size of twins cannot increase

- ✳ Use only point mutation?
- ✳ Devise subtree crossover (e.g. with more crossover points) to ensure average size of two children is exactly the target size?

Implementation:
http://www.cs.ucl.ac.uk/staff/W.Langdon/ftp/gp-code/tiny_gp_twin.c

Reference: PPSN-2016, LNCS 9921, pp313-323, DOI:10.1007/978-3-319-45823-6_29