Assignment 4
Genetic Programming a One Dimensional Bin Packing Objective Function

Quinton Weenink, 13176545[1]

[1]Department of Computer Science – University of Pretoria
Pretoria, South Africa

1. The function set

Function	Inputs
add	2
subtract	2
multiply	2
divide	2
square	1
power	2

2. The terminal set

A Sequence defines terminals that will change over each iteration but refer to the same thing each time. A Constant remains unchanged no matter the iteration of current individual. A Variable can change depending on individual but will not change per iteration.

Terminal	Type
missing spaces	Sequence
consumed space	Sequence
weights in a bin	Sequence
bins required	Variable
bin capacity	Constant
1	Constant
2	Constant

In order to aid in having a smoother search space the Genetic Programme had to only train for the results inside of the summation. A summation was used as each bin should contribute to fitness evenly. The collection of fitness's for each bin should indicate to the fitness of the individual.

3. Evaluation

Evaluation of the GP individuals was determined through one run of the bin packing problem used in Assignment 3. An easy bin packing problem was used due to computational complexity of such a problem.

The GP was sampled over one iteration due to the time.

4. Selection method

Tournament selection was used with a tournament size of 3 and reverse tournament selection was used to select the individuals that should be replaced.

As discussed above the fitness used for selection was determined though getting the resulting bins required for one run of a trivial one dimensional bin packing problem.

5. Parameters

A population size of 30 was used for this research. Due to the computational complexity only 30 generations were used to train the individuals. Due to the low amount of generations and the desire to converge quickly on Crossover was used.

5.1. Mutation

No mutation was used in this research.

5.2. Crossover

After the best individuals were selected using tournament selection random indices were selected from two parents and branches were selected at random.

5.3. Tree generation

Trees were generated using Ramped half-and-half meaning that at each depth half of the trees were generated using grow and half with full. Trees were generated with a depth between 3 and 5 with trivial trees being at depth 0.

Terminal and Function Nodes were chosen at random from the the terminal and function set respectively until the intended depth was achieved. Once the intended tree depth was achieved only terminal nodes were selected.

6. Training vs. Testing

6.1. Training

Problem	m*
N1C1W1$_A$	25
N1C1W1$_O$	32
N1C1W1$_P$	26
N1C1W1$_T$	28

Fig. (1) depicts how the population moves to wards a better objective function after each generation. The solve ratio is simple the minimum number of bins the individual achieved in GA training over the known minimum bins.

7. Testing

Problem	m*
N1C1W1$_G$	25
N1C1W1$_K$	26
N1C1W1$_S$	28
N1C1W1$_F$	27

In order to prevent overfitting multiple problem instances of difference difficulties should be used. For this research only different instances of the easy solution were used

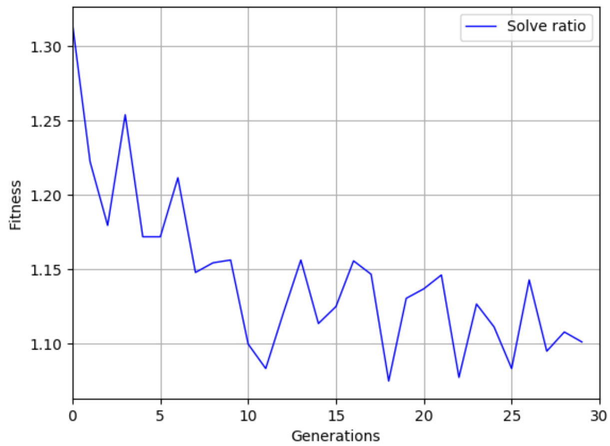

Figure 1. Training over 30 generations

8. Best evolved function

Due to the use of multiple testing files the mean over each could not be reported. The mean over the solution ratio will be reported.

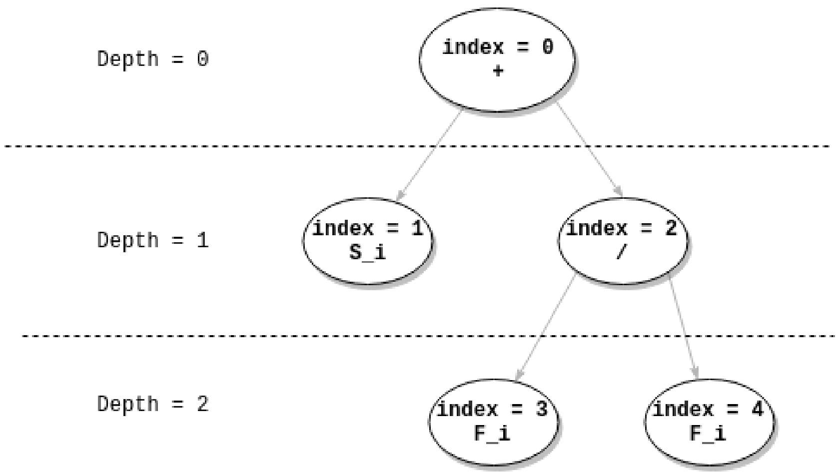

Figure 2. Training over 30 generations

As can be seen then in 2 the tree does contain some redudent branches such as S_i/S_i

9. Comparison to Falkenauer, E

The manner in which this solution was implementation limed function preventing the use of functions and terminals outside of the summation if I had to implemented this again I would include the summation operation in the functional set. As for the rest I shall compare the internals.

While the generated equation 1 doesn't represent Falkenauer's proposed equation 2 it does highlight that even over very few runs it was able to establish the importance of F_i.

$$f_{BPP} = \sum_{i=1}^{nb} (F_i) \tag{1}$$

$$f_{BPP} = \frac{\sum_{i=1}^{nb} (F_i/C)^k}{N} \tag{2}$$

2