Matthew Anderson
mia2n4@mst.edu
COMP SCI 5401 FS2015 Assignment 2c

Methodology

Genetic programming can be applied to a variety of different applications and is perhaps the most versatile type of evolutionary computing. The main feature of a genetic program is its utilization of parse trees to represent chromosomes and it seeks to find a model with maximum fit. In this application a genetic program (GP) was developed to create two controllers for a simplified version of Ms. Pacman called GPac. The first controller was meant to choose moves for Ms. Pacman which maximized her fitness that was represented as her score. The second controller was meant to choose moves for the ghosts, in this application the ghost controllers were the same for every ghost on the board.

The GP was implemented using a binary parse tree with two different sets of nodes. The first set was the functional set which represented mathematical operations that could be performed on both the right and left children of a given node. These operations could only be applied to non-terminal nodes and were the same for both Ms. Pacman and the Ghosts. The functional set used in this experiment included the operations: addition, subtraction, multiplication, division, and random(a, b). The functional set could be easily expanded to include any mathematical operator which requires two inputs and returns a floating point number. The GP tree also contained a terminal set which dictated what a terminal node of the GP tree could contain. The terminal set differed for Ms. Pacman consisted of: distance to the nearest pill, distance to the nearest ghost and a random floating point number between 0 and 100. The terminal set for the Ghosts differed slightly and consisted of: distance to Ms. Pacman, distance to the nearest ghost and a random floating point number between 0 and 100. These sets were used to create GP parse trees and an example can be found in Figure 1.

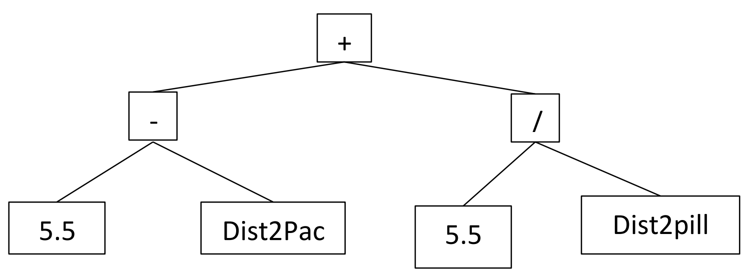

Figure 1 - Example of a Ms. Pacman parse tree

In order to create varied and diverse parse trees initialization was performed using the ramped half-and-half method. This method combines two other types of initialization and selects one or the other with equal probability. The first initialization method in ramped half-and-half is full initialization. This method initializes the tree such that each branch is initialized to the maximum depth and all terminal nodes appear at that depth while every other node is functional. The second method is the grow method where every node has the chance to be a terminal node and each branch is expanded until a terminal is picked or a max depth is reached.

The GP is set up to run in a loop which repeats the same process until a specific ending condition is reached which in this experiment was a maximum fitness evaluation count of 2000 evaluations. The program first performs initialization on an entire population and evaluates these to completion by running a game of GPac and recording the score to be used as fitness. With this population the program then chooses a mating pool using either fitness proportional selection or over-selection, from the mating pool new entities are created using either mutation or crossover based on a mutation rate, the new entities are evaluated and combined with the previous generation. Survivors for the next generation are then picked using either truncation or k-tournament survival selection and the process is repeated on this new generation. In this implementation mutation simply replaced a random node with a new random tree and crossover chooses a random node in two trees and swaps them to create two new trees.

The competitive aspect of this GP is evident in the opposing goals of both Ms. Pacman and the ghosts. The goal for Ms. Pacman is to eat all the pills while the goal for the ghosts is to stop Ms. Pacman from eating all the pills. In order to evolve these two controllers simultaneously the fitness for Ms. Pacman was simply the score from the game while the fitness for the ghost was the negation of that score. This meant that a high score was good for Ms. Pacman but bad for the ghosts and vise-versa. In an ideal situation each individual in the Ms. Pacman population would be run against each individual in the ghost population but this was computationally infeasible. Instead a sampling was used where the populations would be pitted against a single individual from the other population and this evaluation would be used as the fitness. The final aspect of fitness evaluation was the use of a parsimony coefficient. This was used to prevent excessive bloat in the parse tree which could severely slow down computation. The parsimony pressure punished fitness based on the size of the parse tree and used a coefficient to decide how much to punish the solution.

Experimental Setup

In this experiment three different configuration were tested and compared. Each configuration consisted of thirty runs which each consisted of 2000 fitness evaluations on a GPac game board with 10 columns and 15 rows that had a pill density of 50%. These parameters were left unchanged in all configurations which allowed for the testing different parameters on a similar environment. The three sets of parameters that were used in this experiment can be found in Table 1.

Table 1 - Three configuration parameters used in this experimanent

	Configuration 1	Configuration 2	Configuration 3
μ_{Pacman}	50	30	30
μ_{Ghost}	30	20	30
λ_{Pacman}	50	30	50
λ_{Ghost}	50	20	50
Mutation rate	5%	15%	5%
Parent selection strategy	over-selection	fitness proportional	over-selection
Survival selection strategy	truncation	truncation	k-tournament
Pacman tournament size	N/A	N/A	10
Ghost tournament size	N/A	N/A	10
Over-selection percentage	80%	N/A	N/A
Pacman parsimony coefficient	0.5	1	0.5
Ghost parsimony coefficient	0.5	1	0.5

The first configuration has a large Pacman population that generates the same number of offspring, this results in a higher selective pressure as there is more competition. The ghost population size is smaller and so there is less selective pressure for the ghosts. There is also a relatively low mutation rate and over-selection is used for parent selection with an over-selection percentage of 80% while truncation is used for survival selection. The second configuration uses smaller Pacman and ghost populations and offspring size meaning there is less selective pressure but there will be more generations. The final configuration uses moderate population sizes and over-selection for the parent selection along with k-tournament selection for survival selection. These parameters all differ in population size and selection strategies and so should help to identify optimal parameters for the GP.

Results

The results are shown below in the form of a box plots which are comparing the maximum fitness of Ms. Pacman at each generation. The results obtained by configuration 1 can be found in Figure 2 and the data can be found in the Appendix. The second configuration file results can be found in Figure 3 and the thrid configuration file results can be found in Figure 4.

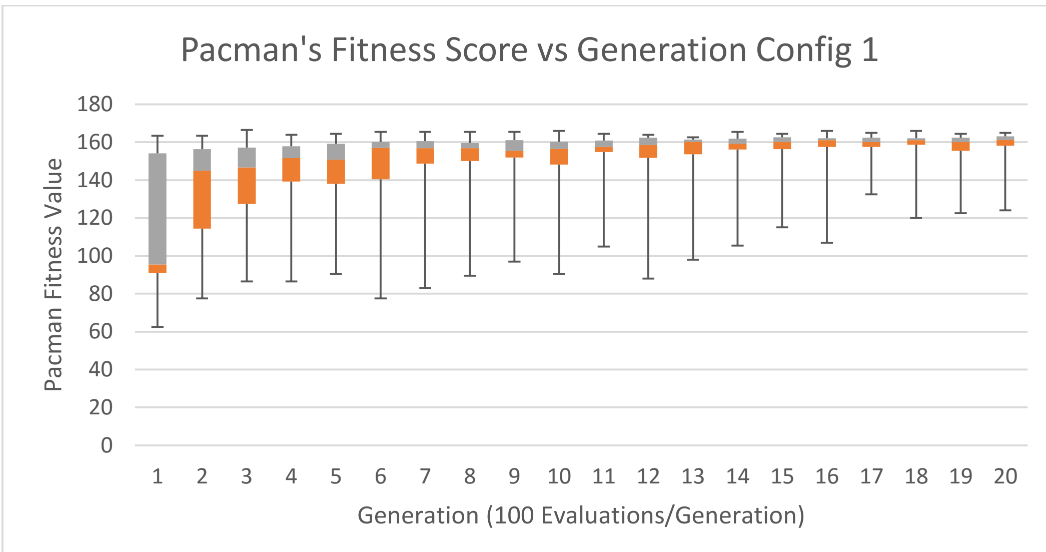

Figure 2 - Configuration file 1 box plot

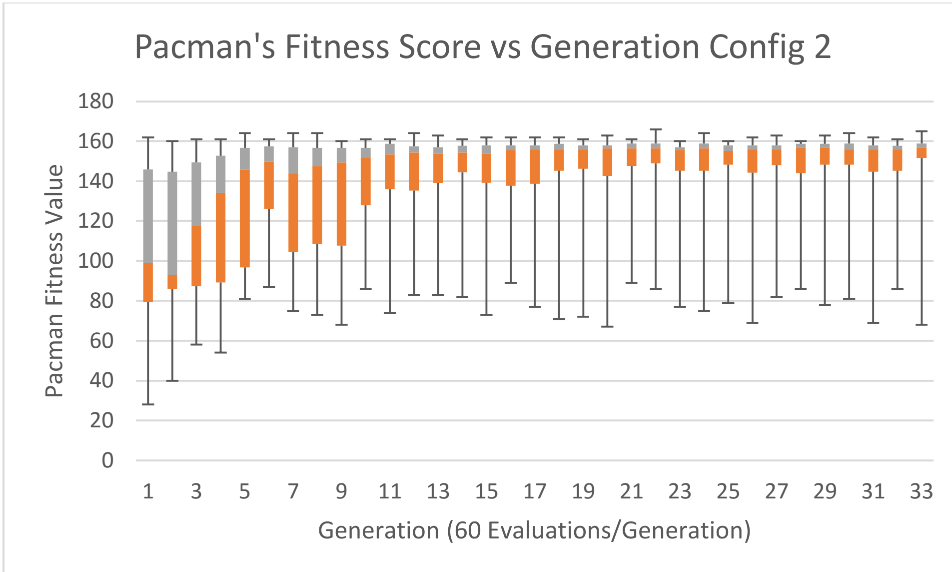

Figure 3 - Configuration file 2 box plot

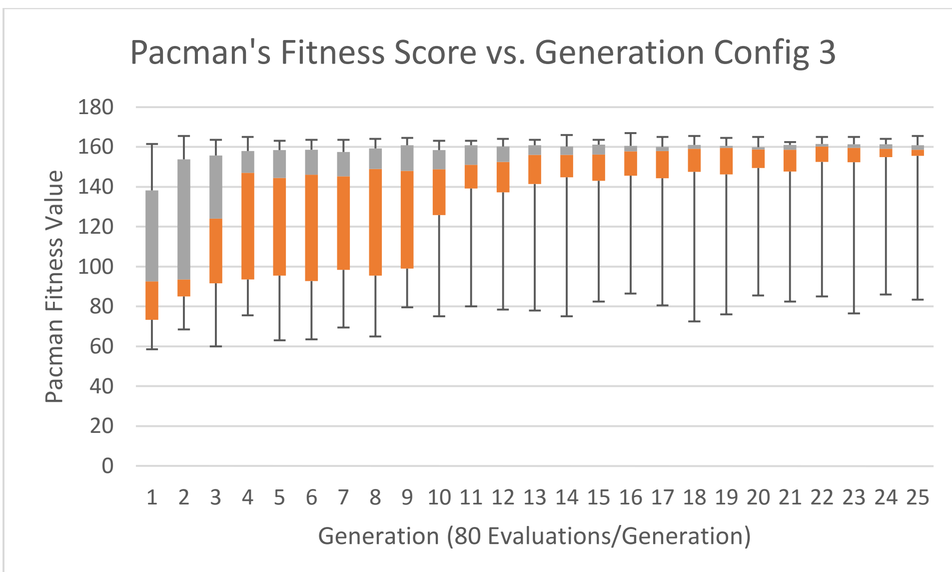

Figure 4 - Configuration file 3 box plot

Discussion

In order to compare configuration files the median fitness of Ms. Pacman was used as it represents the overall performance of both populations. When comparing configuration 1 to configuration 2 an F-Test was performed to determine whether variances were equal or not. The results from the F-Test can be found in in Table 2 - F-Test for configuration 1 vs. configuration 2 and from this test it can be assumed that the variances are equal which means that a t-test with equal variances can be performed. From the t-Test results seen in Table 3 it can be concluded that the results of the two configuration files were equal.

Table 2 - F-Test for configuration 1 vs. configuration 2

	Configuration 1	*Configuration 2*
Mean	153.55	148.9090909
Variance	208.5236842	248.7414773
Observations	20	33
df	19	32
F	0.83831489	
P(F<=f) one-tail	0.349064222	
F Critical one-tail	0.485396375	

Table 3 - t-Test assuming equal variances for config 1 vs. config 2

	Config 1	*Config 2*
Mean	153.55	148.9090909
Variance	208.5236842	248.7414773
Observations	20	33
Pooled Variance	233.7583779	
Hypothesized Mean Difference	0	
df	51	
t Stat	1.071157916	
P(T<=t) one-tail	0.144570741	
t Critical one-tail	1.67528495	
P(T<=t) two-tail	0.289141482	
t Critical two-tail	2.00758377	

The second test compared configuration 2 and configuration 3. The results of the f-test to test for variance equality can be found in Table 4 and based on the results in can be assumed that the two configurations do not have equal variances and so a two tailed t-test assuming unequal variances was used. Based on the results of the t-test which can be found in Table 5 it can be assumed that the mean fitness of the controller for both configuration 2 and configuration 3 were the same for each generation.

Table 4 - f-Test for config 2 vs config 3

	Configuration 2	*Configuration 3*
Mean	148.9090909	147.96
Variance	248.7414773	336.0764583
Observations	33	25
df	32	24
F	0.740133595	
P(F<=f) one-tail	0.210880314	
F Critical one-tail	0.536600903	

Table 5 - t-Test assuming unequal variances for config 2 vs config 3

	Config 2	*Config 3*
Mean	148.9090909	147.96
Variance	248.7414773	336.0764583
Observations	33	25
Hypothesized Mean Difference	0	
df	47	
t Stat	0.207203957	
P(T<=t) one-tail	0.418372859	
t Critical one-tail	1.677926722	
P(T<=t) two-tail	0.836745718	
t Critical two-tail	2.011740514	

Conclusion

It can be concluded that from these experiments a viable controller for both Ms. Pacman and the ghosts can be created. The competitive coevolution of both of these controllers in this experiment resulted in a skew towards Ms. Pacman based on the results mostly because much of the comparison was based off of the maximum fitness value of the Ms. Pacman controller. The results are also somewhat flawed due to the fact that only a sampling of the competition's population was used for computation feasibility. If the entire population was competing with the entire other population a more stable equilibrium would have been reached. As it stands in this implementation it is entirely possible that a Ms. Pacman controller competes with a ghost controller which performs very poorly against the employed strategy.

Based on the results of the analysis it can be concluded that all three configuration files performed at about the same level. Each configuration file resulted in a very similar box plot that starts with a large variance in the population and slowly converges towards a maximum.

Appendix

Table 6 - Data for configuration 1

Fit Evals	Min	Q1	Median	Q2	Max
150	62.5	91.125	95.5	154.125	163.5
250	77.5	114.375	145	156.25	163.5
350	86.5	127.375	146.75	157.125	166.5
450	86.5	139.25	151.5	157.875	164
550	90.5	138	150.75	159.25	164.5
650	77.5	140.375	157	160	165.5
750	83	148.625	156.75	160.5	165.5
850	89.5	150	156.75	159.5	165.5
950	97	152	155.5	161	165.5
1050	90.5	148.125	156.5	160.25	166
1150	105	154.75	157.5	160.875	164.5
1250	88	151.75	158.5	162.375	164
1350	98	153.625	160.25	161.375	162.5
1450	105.5	156.125	159.25	161.875	165.5
1550	115	156.25	160.25	162.5	164.5
1650	107	157.5	161	162	166
1750	132.5	157.5	160	162.375	165
1850	120	158.625	161	162	166
1950	122.5	155.5	160	162.375	164.5
2050	124	158.125	161.25	163	165

Table 7 - Data for configuration 2

Fit Evals	Min	Q1	Median	Q3	Max
90	28	79.5	99	146	162
150	40	86	93	144.75	160
210	58	87.25	117.5	149.5	161
270	54	89.25	134	152.75	161
330	81	96.75	146	156.75	164
390	87	126	150	157.5	161
450	75	104.5	144	157	164
510	73	108.5	147.5	156.75	164
570	68	107.75	149.5	156.75	160
630	86	128	152	156.75	161
690	74	136	153.5	158.75	161
750	83	135.25	154.5	157.5	164
810	83	139	154	157	163
870	82	144.5	154.5	157.75	161
930	73	139.25	154	158	162
990	89	137.75	155.5	158	162
1050	77	138.75	156	158	162
1110	71	145.25	156	158.75	162
1170	72	146.25	156	158	161
1230	67	142.5	156.5	158	163
1290	89	147.5	156.5	159	161
1350	86	149	156.5	159	166
1410	77	145.25	155.5	157	160
1470	75	145.25	156.5	159	164
1530	79	148.25	155	158	160
1590	69	144.25	156	158	162
1650	82	148	156	158	163
1710	86	144	157	158.75	160
1770	78	148.25	157	158.75	163
1830	81	148.25	156	159	164
1890	69	144.75	156	158	162
1950	86	145.25	156	157.75	161
2010	68	151.5	157	159	165

Table 8 - Data for configuration 3

Fit Evals	Min	Q1	Median	Q3	Max
110	58.5	73.375	92.5	138.25	161.5
190	68.5	85	93.5	153.75	165.5
270	60	91.625	124	155.75	163.5
350	75.5	93.5	147	157.875	165
430	63	95.5	144.5	158.375	163
510	63.5	92.75	146	158.5	163.5
590	69.5	98.375	145.25	157.5	163.5
670	65	95.5	149	159.25	164
750	79.5	99	148	160.875	164.5
830	75	125.75	148.75	158.375	163
910	80	139.125	151	160.875	163
990	78.5	137.25	152.5	160.25	164
1070	78	141.375	156	160.875	163.5
1150	75	144.75	156	160.125	166
1230	82.5	143	156.25	161.125	163.5
1310	86.5	145.625	157.75	160.5	167
1390	80.5	144.25	158	160	165
1470	72.5	147.5	159	161	165.5
1550	76	146.25	159.5	160.5	164.5
1630	85.5	149.375	158.75	159.875	165
1710	82.5	147.625	158.5	161	162.5
1790	85	152.5	160.25	161.5	165
1870	76.5	152.25	159.5	161.375	165
1950	86	154.875	159	161.25	164
2030	83.5	155.5	158.5	160.875	165.5