COMP 5660 Fall 2023 Assignment 2c

Matthew Freestone
maf0083@auburn.edu

18 November 2023

1 Green Deliverables

1.1 GP Parameters

See Table 1 for Green PacMan GP Parameters.

Parameter Name	Value
μ	200
λ	50
Mutation Rate	0.50
Depth Limit	7
Constant Terminal Range	[-5,20]
Bloat Control	Parsimony Pressure with Coefficient 1
Parent Selection Method	K-Tournament With Replacement ($k = 7$)
Survival Selection	K-Tournament Without Replacement ($k = 7$)
Recombination Method	Subtree Crossover
Mutation Method	Subtree Mutation
Tree Size Metric	Number of Nodes
Sample Size	3

Table 1: Green PacMan GP Parameter Values

See Table 2 for Green Ghost GP Parameters.

Parameter Name	Value
μ	400
λ	100
Mutation Rate	0.60
Depth Limit	7
Constant Terminal Range	[-5,20]
Bloat Control	Parsimony Pressure with Coefficient 2
Parent Selection Method	K-Tournament With Replacement ($k = 11$)
Survival Selection	K-Tournament Without Replacement ($k = 9$)
Recombination Method	Subtree Crossover
Mutation Method	Subtree Mutation
Tree Size Metric	Number of Nodes
Sample Size	3

Table 2: Green Ghost GP Parameter Values

1.2 Parse Trees

See Table 3 for the best ghost and PacMan parse trees from the final generations for each of the 10 runs.

Run and Type	Tree Representation
0 pac	$((F - F) - (F/G))$
0 ghost	$(((M * P) * (W/3.186)) - ((M * F) * 13.559))$
1 pac	$((F/(P \text{ RAND } 11.991)) * (((P/G) - P) + (G * 12.771)))$
1 ghost	$((F + P)/M)$
2 pac	$((15.782 \text{ RAND } ((F - P) - P)) * G)$
2 ghost	$((M/F) * (M/(M/(P - F))))$
3 pac	$((F * (G - 1.739)) - F)$
3 ghost	$(((G/M)/(1.515 \text{ RAND } M)) - (M - P))$
4 pac	$((12.905 \text{ RAND } ((P + G) + G)) * G)$
4 ghost	$(((P/P)/M) - M)$
5 pac	$(((W/19.425) \text{ RAND } G) + G)$
5 ghost	$(((P/(F + (F * P))) * P) - M)$
6 pac	$(G/(6.827 + (P + P)))$
6 ghost	$(P - (F * (M + (F/F))))$
7 pac	$(((((1.860 \text{ RAND } G)/(P * 8.025)) * G) * (P + G)) * 1.860)$
7 ghost	$((P \text{ RAND } (((((W/W) * F)/((F + P) + (G - M)))/F)/G)) - M)$
8 pac	$((((F * G) - F) * F) * (((7.267 \text{ RAND } G)/(P * F))/((F - -3.642)/F)))$
8 ghost	$(W - M)$
9 pac	$(G * G)$
9 ghost	$((P + (W - (M * 9.036))) - -4.994)$

Table 3: Green Parse Trees

1.3 Results

The box plots showing the performance of the best PacMan controllers against 100 randomly-generated Ghost controllers can be found in Figure 1

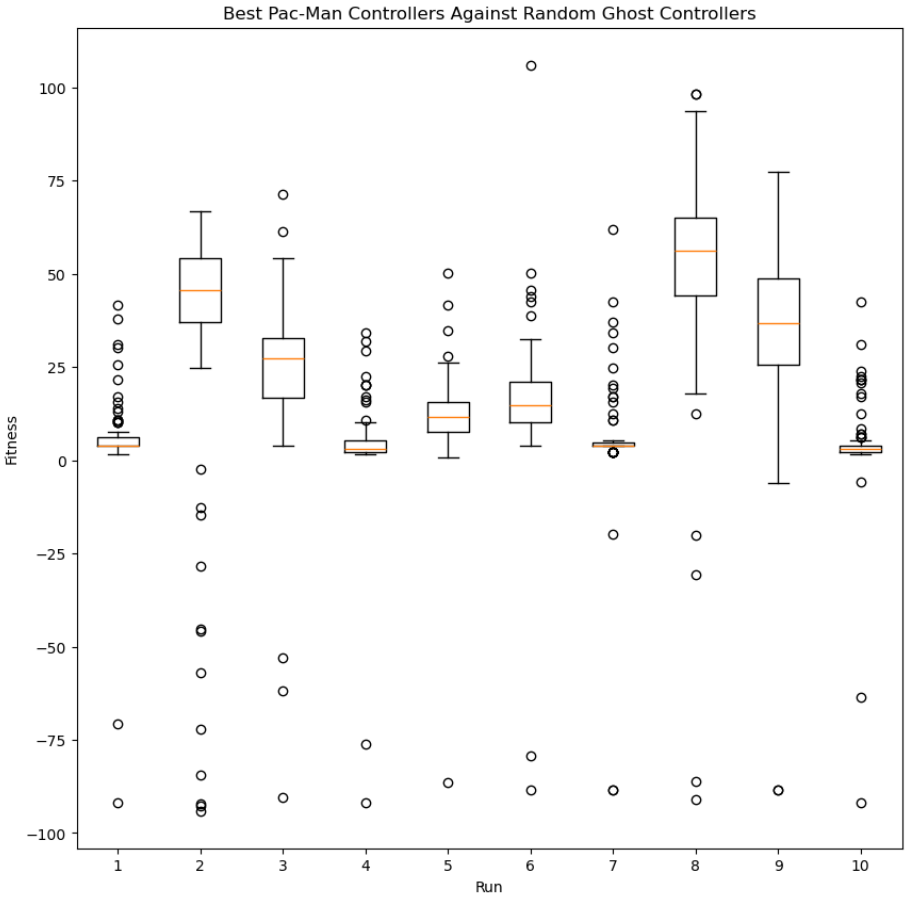

Figure 1: Performance of best PacMan against 100 random Ghosts

The box plots showing the performance of the best Ghost controllers against 100 randomly-generated PacMan controllers can be found in Figure 2

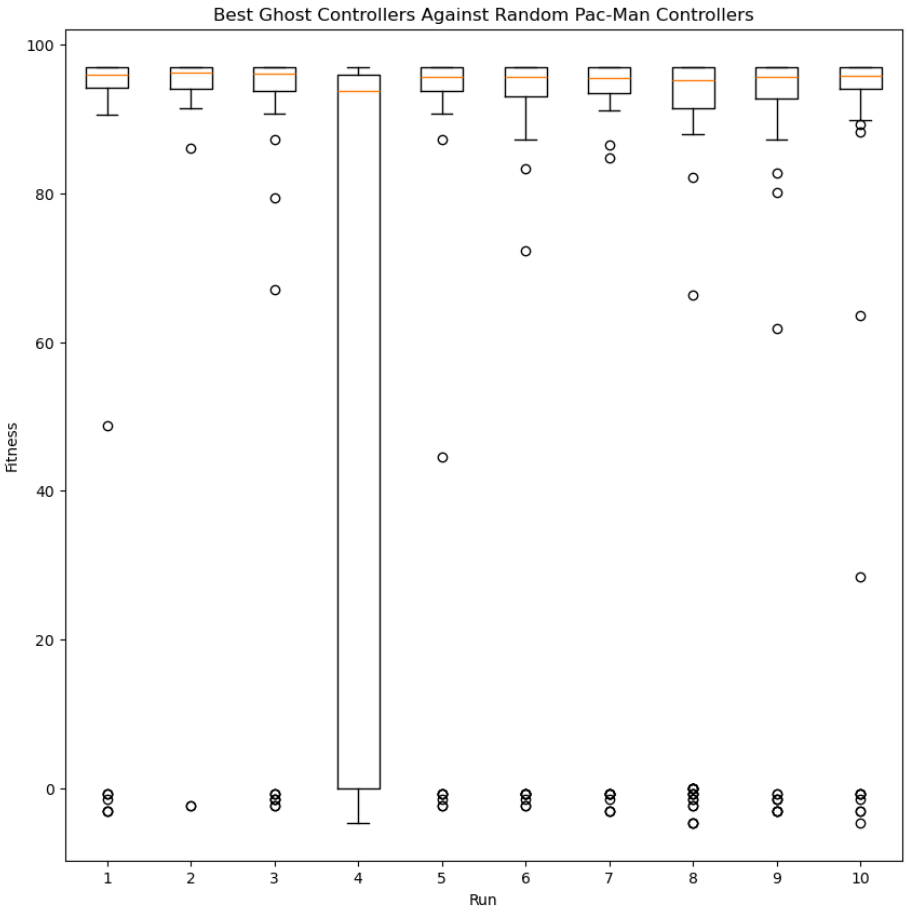

Figure 2: Performance of best Ghosts against 100 random PacMani

The best base fitness PacMan from the 10 runs came from run 7 with score 59.39. The same run's best ghost score was -96. An exhibition game was played between these two controllers, and the log is saved in the 'data/2b/green' directory. The genes for these two controller is also shown below.
PacMan: $(((((1.860 \text{ RAND } G)/(P * 8.025)) * G) * (P + G)) * 1.860)$
Ghost: $((P \text{ RAND } (((((W/W) * F)/((F + P) + (G - M)))/F)/G)) - M)$

The tree for PacMan contains a large number of G and P primitives, showing that evolution encouraged taking the distance to ghosts into great consideration when choosing where to move, as well as distance to nearest pill. It still contains RAND to allow PacMan to escape corners.
The ghost tree is significantly larger/deeper than most of the best run's ghosts's parse trees. It contains a $-M$ section at the first level of the tree, and this term encourages the agent to pick the direction that prints it closer to PacMan.

PacMan starts the game by collecting the nearby pills, and the ghosts head toward the player, as we've seen in most runs. However, when the ghosts all approach the player, we see some interesting behavior. The ghosts all run into each other, and the continuously move toward the direction of the other ghosts, completely ignoring the player. After they get stuck, the PacMan then goes around the rest of the map collecting pills.
The strange ghost behavior is likely due to the $/G$ term in its equation, shown here in with unnecessary terms hidden:

$$\left(\left(P \text{ RAND } \frac{(\ldots)}{G}\right) - M\right)$$

As the ghosts approach each other, if RAND selects the right term, it approaches infinity with the distance to another ghost being 0. This dominates any other part of the equation. When the other side of the RAND is picked, it is the ghost might move toward a pill or the PacMan, but it quickly returns to following the other ghosts.

2 Yellow Deliverables

In the yellow section, the results from the Green runs were analyzed by splitting the population into two halves. The best PacMan controllers from the first half play against the Ghost controllers from the other half, and vice-versa, creating a test-set. These controllers played against each other 10 times, and the average games scores are shown in Figure 3.

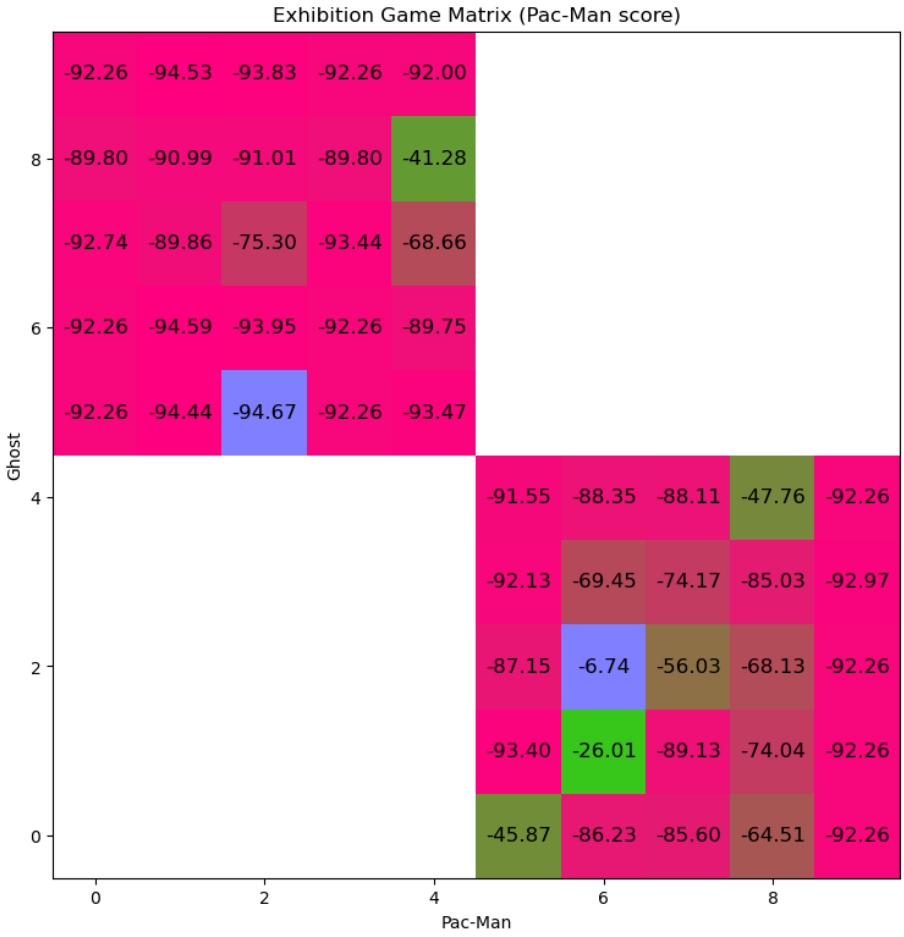

Figure 3: Performance Matrix for Test Set

2.1 Analysis

1. Are you able to see any signs of intransitivity in this visualization?
   Yes, some intransitivity is seen in this visualization. PacMan 5 performs better against Ghost 0 than PacMan 6, but PacMan 6 performs better against Ghost 1 than Pacman 5.

2. Do the results differ meaningfully from those obtained using the GREEN analysis method?
   Yes. This visualization shows the best ghosts found were, as a whole, much better than the best PacMani found. The randomly generated controllers did not seem representative of space of valid phenotypes. This also shows that some of the best found PanMan controllers were likely only there by

chance, as they performed extremely poorly against non-random ghosts.