Sujay Jakka

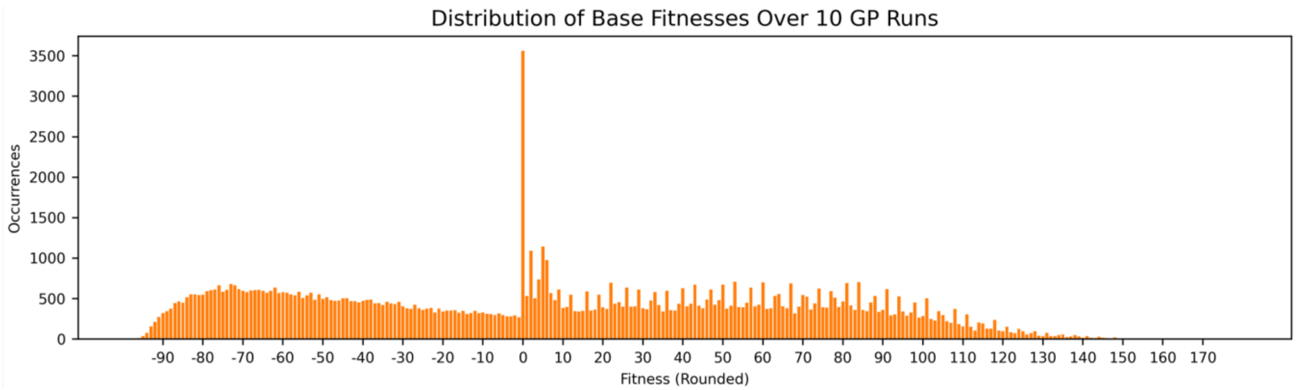
Svj0007@auburn.edu

COMP 5660 Fall 2024 Assignment 2b

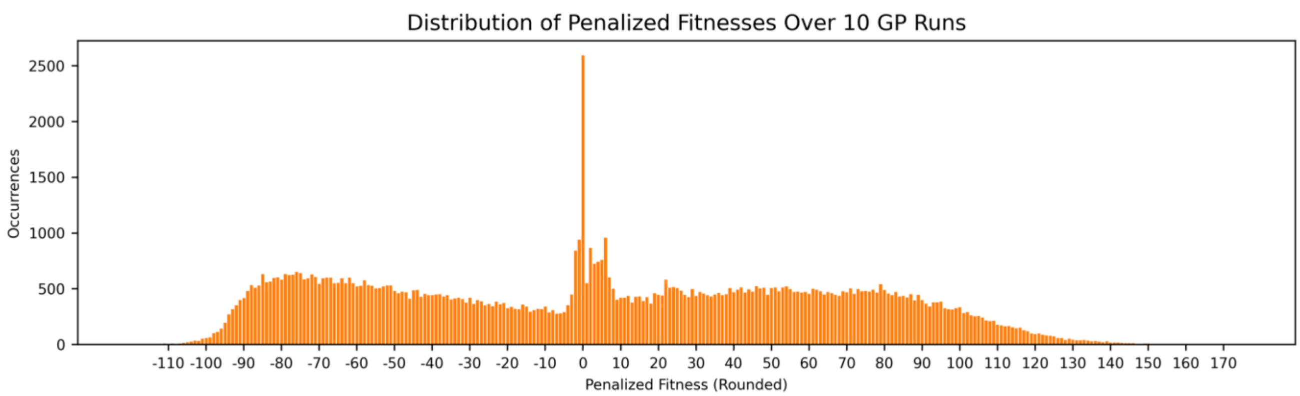
17 November 2024

Assignment 2b Report

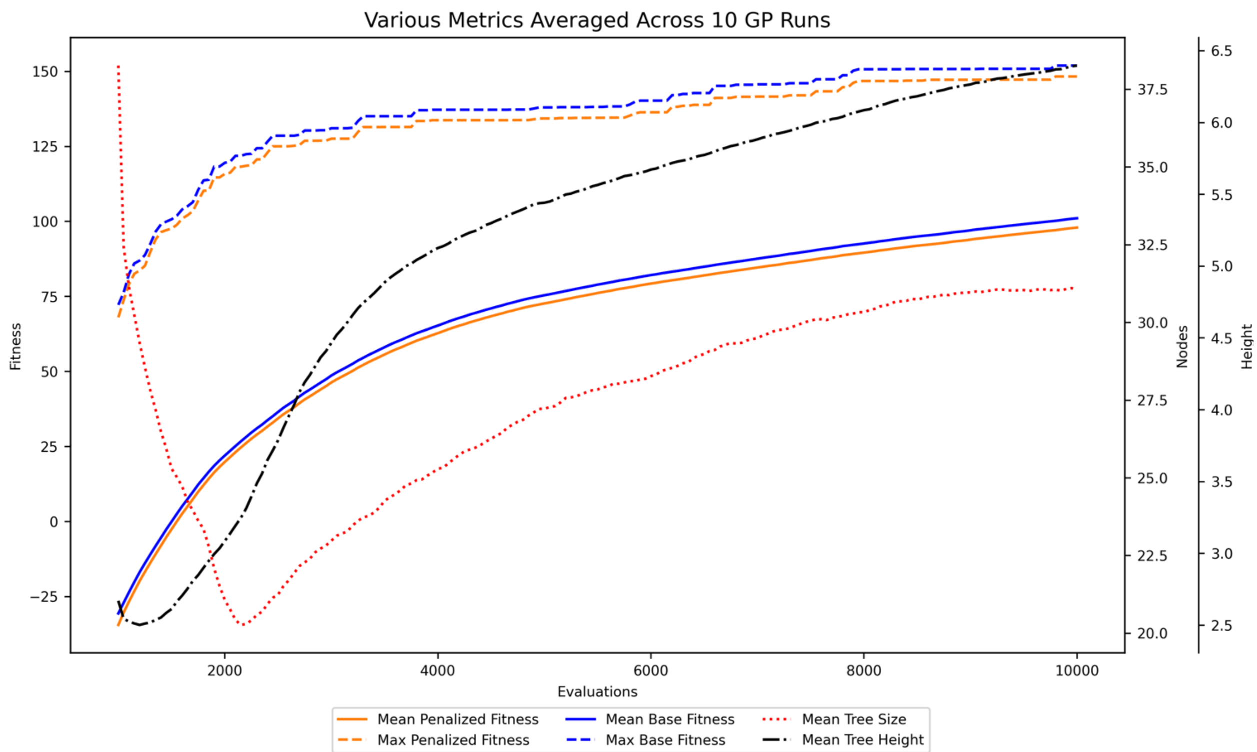
The histogram below is the distribution of base fitness across all 10 runs in the experiment.



Similarly, the following histogram is the distribution of penalized fitness across the experiment.



Lastly, the plot below is of the evals-vs-metrics plot showing the progress of evolution averaged over the experiment.



The best Pac-Man controller that came out of the random search algorithm did a relatively good job of heading towards the pills and the fruits and moving away from the ghosts. However, one of the issues I saw with this controller was that it would be too afraid to head towards the pills if the ghosts happen to be in the same direction as the remaining pills. This would cause Pac-Man to exhibit behavior such as heading toward a corner or staying in the same region until the ghosts moved. This behavior can be problematic once the ghosts actually start to move intelligently. Looking at my visualization of the best 2b Pac-Man controller, it seems to do a better job at being aggressive. There were many occasions during the game where Pac-Man would be right next to a ghost. Furthermore, Pac-Man was able to acquire 8 fruits and only left 3 pills on the board. The ultra aggressive behavior the controller exhibited could be troublesome once the ghosts start to move intelligently. It would be ideal to find a solution that is more passive, but because the controllers were evolved playing a game where the ghosts played randomly it makes sense how we landed on this type of solution. For reference the best controller from 2a had a score of 129.127, while the best controller from 2b had a score of 177.674. Additionally, the 2a and 2b controller had very different parse trees. The parse tree from 2a had a max-depth of 4 while the parse tree from 2b had a max-depth of 7 reaching the depth-limit. Both trees only had one constant value terminal node and both had many G, P, and F terminal nodes. It seems that the W value(number of adjacent walls) may not be as important as the other terminal values as the 2a solution has none of these terminal nodes and the 2b solution only has one.

Lastly, the mean of the sample of 10 runs(sample size 10) of the EA was 151.946, while the mean of the sample of 10 runs(sample size 10) of the random search algorithm was 114.254. The standard deviation of the EA and random search sample was 14.443 and 13.367 respectively. I performed a two-sample independent t-test where the null hypothesis was that the two algorithms are equally effective at solving the problem, and the alternative hypothesis is that the two algorithms are not equally effective at solving the green problem instance. I performed the t-test with a significance level(alpha) of 0.05. I got a p-value of 1.029e-5, which is significantly smaller than the significance level of 0.05. We can then reject the null hypothesis, and claim that the two algorithms have statistically significant differences in performance. Because it is highly unlikely that both sample distributions have the same population mean and that the EA sample has a higher mean than the random search sample, we can conclude that the EA performed better. The parameters I used for the green problem instance are shown below.

[ea]

mu = 1_000

num_children = 50

mutation_rate = 0.02

parent_selection = k_tournament_with_replacement

survival_selection = k_tournament_without_replacement

individual_class = TreeGenotype

[parent_selection_kwargs]

k = 50

[survival_selection_kwargs]

k = 40

[fitness_kwargs]

parsimony_coefficient = 1/10

experiment = green
