

It appears that the latency of the program is more dependent upon the size of the genome than the size of the maze. For example, compared to a 7x7 maze with a genome size of 50, you would only get a slightly higher latency with a 21x21 maze with the same genome size (usually something like 4 seconds to 7-8 seconds). However, you get an increase of almost 15 seconds when you increase the size of the genome to 200. This makes sense because the majority of the work is being done on the genome, not the maze. The program has to much more frequently traverse the genome than the maze.

If I were to alter this program to potentially improve its performance, I would first change how much random mutation there is. I believe that 40% is too low for the mutation, especially if you are only changing a single move each time. I would also like to include a function that mutates the genome in a non-random way. For example, plenty of genomes include instances where they are going left then right or up then down (meaning they aren't actually moving). I think it could be beneficial to isolate these cases and randomly set one of the two to zero. Regardless, it is hard to increase performance under such strict specifications.