Marco Ogaz-Vega
Professor Shapiro
CMPM 146
11 February 2025

P5 Writeup

Selection Strategy:

Changes: Implemented a selection strategy that includes elitist selection and tournament selection:

Elitist Selection: Keep the top 10% of the population based on fitness.

Tournament Selection: Select the rest of the population using tournament selection with a tournament size of 5. This ensures diversity while maintaining high-quality individuals.

Crossover Operator:

Changes: Implemented a single-point crossover operator for both Individual_Grid and Individual_DE classes:

Individual_Grid: Perform a single-point crossover between two parents, ensuring proper constraints to prevent pipes in the air and other invalid configurations.

Individual_DE: Perform a crossover by splitting the genome at random points and combining parts from both parents.

Mutation Operator:

Changes: Implemented a mutation operator for both Individual_Grid and Individual_DE classes:

Individual_Grid:

Rule 1: Do not generate wall blocks (X) above the bottom 3 layers.

Rule 2: Generate enemies (E) only on blocks with a standable block below them, with a maximum enemy count of 10. Additionally, do not generate enemies right next to the player.

Rule 3: Ensure pipes (| and T) are complete by generating a base (X) below them and a number of | above them ranging from 1-3, topping it off with a T.

Rule 4: Generate longer stretches of wall ranging from 4-8 blocks.

Rule 5: Small chance to generate breakable blocks, coins, mushrooms, or coin blocks 2 blocks above ground or standable blocks like pipes.

Rule 6: Ensure a playable path with gaps no longer than 4 blocks and no gaps of 1 block.

Rule 7: Ensure no gaps of length 1 in the bottom layer.

Individual_DE:

Implemented mutation for various design elements such as blocks, question mark blocks, coins, pipes, holes, stairs, platforms, and enemies.

Added constraints to prevent generating weird configurations and ensure proper placement of elements.

Fitness Function:

Changes: Modified the fitness function to include additional penalties and rewards:

Individual_Grid:

Bottom Bonus: Reward blocks at the bottom 6 levels.

Penalty: Add a penalty for blocks in the top 4 levels.

Overlap Penalty: Penalize overlapping elements.

Density Penalty: Penalize excessive density.

Spacing Bonus: Reward proper spacing.

Isolation Penalty: Penalize isolated elements.

Enemy Penalty: Penalize more than 2 enemies within a certain radius.

Individual_DE:

Added penalties for too many stairs and other unaesthetic configurations.

Included various metrics such as meaningful jump variance, negative space, path percentage, empty percentage, linearity, and solvability.

Population Initialization:

Changes: Modified the population initialization to ensure proper constraints and weighting of different tile types to prevent pipes in the air and other invalid configurations:

Individual_Grid: Ensured a playable path with gaps no longer than 4 blocks and no gaps of 1 block. Also ensured no gaps of length 1 in the bottom layer.

Individual_DE: Enhanced the random individual generation to include various design elements such as holes, platforms, enemies, coins, blocks, question mark blocks, stairs, and pipes.

Technical Implementation

Selection Strategy:

Elitist Selection: The top 10% of the population is retained based on fitness to ensure that the best individuals are preserved.

Tournament Selection: The rest of the population is selected using tournament selection with a tournament size of 5. This helps maintain diversity while selecting high-quality individuals.

Crossover Operator:

Individual_Grid: A single-point crossover is performed between two parents, ensuring that the crossover point is within valid bounds. This helps combine the genetic material of both parents while maintaining valid configurations.

Individual_DE: The genome is split at random points, and parts from both parents are combined to create new offspring. This helps generate diverse levels with different design elements.

Mutation Operator:

Individual_Grid: The mutation operator ensures proper placement of elements, limits gaps in the floor, and integrates isolated elements. Specific rules are implemented to prevent invalid configurations and ensure a playable path.

Individual_DE: The mutation operator modifies various design elements such as blocks, question mark blocks, coins, pipes, holes, stairs, platforms, and enemies. Constraints are added to prevent generating weird configurations and ensure proper placement of elements.

Fitness Function:

Individual_Grid: The fitness function includes various metrics and penalties to ensure the generated levels are playable and generate interesting structures. Metrics such as meaningful jump variance, negative space, path percentage, empty percentage, linearity, and solvability are used to calculate fitness.

Individual_DE: The fitness function includes penalties for unaesthetic configurations such as too many stairs. Various metrics are used to calculate fitness, ensuring the generated levels are playable and aesthetically pleasing.

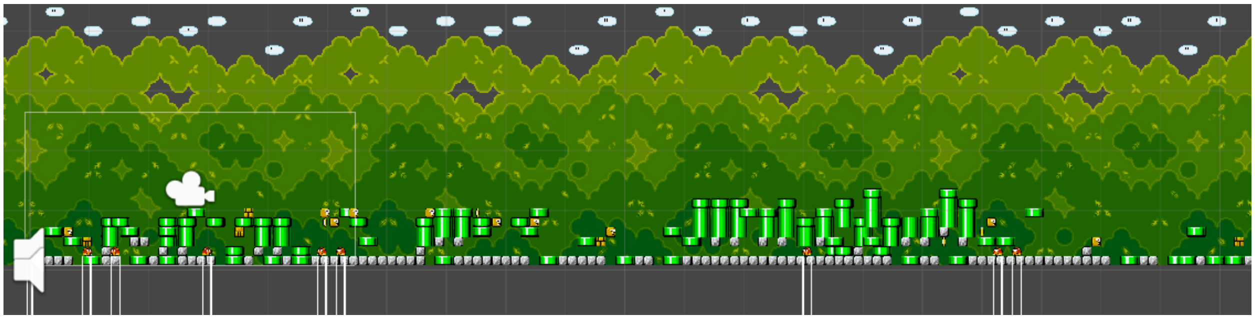
Population Initialization:

Individual_Grid: The population initialization ensures a playable path with gaps no longer than 4 blocks and no gaps of 1 block. It also ensures no gaps of length 1 in the bottom layer.

Individual_DE: The random individual generation includes various design elements such as holes, platforms, enemies, coins, blocks, question mark blocks, stairs, and pipes. This helps generate diverse levels with different design elements.

Favorite Levels

Level 1:



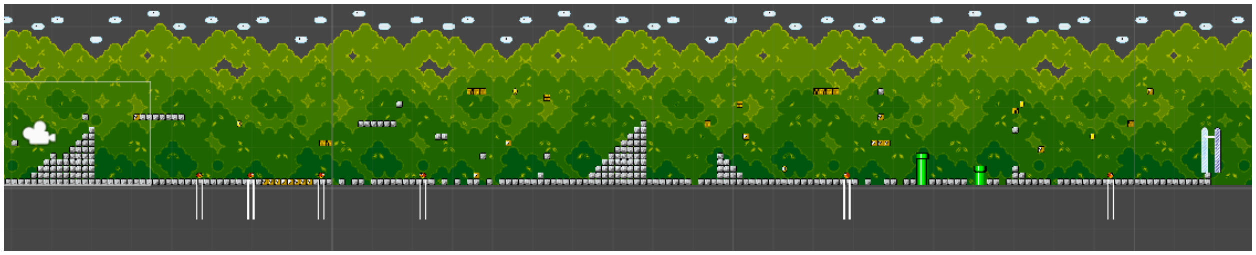
Description: This level was generated using the Individual Grid system. The reason I chose it is because of the design elements of the pipes. The pipes lay over the bottom layer of the level with some elements like enemies and coins and blocks under them. This contrast and gameplay gives the level a feeling of jumping rooftop to rooftop in order to get to the end of the level.
Generation: 199

Max fitness: 98.71459852273028

Average generation time: 4.958992176918528

Total time: 986.8394432067871

Level 2:



Description: This level was generated using my Individua_DE implementation and the reason I chose this level as one of my favorites is because it feels very handmade. The flow is very pleasant with this level and there are even skill based areas for the player to explore above the main part at the bottom. There are also challenging elements to this one like a few odd enemy placements that make the player stop and assess.

Generation: 10

Max fitness: 3.036391110046572

Average generation time: 4.503851008415222

Total time: 45.03851008415222