

Problem 1

- What does a population represent in this case?
 - A population would simply represent a set of chromosomes (paths) from the start node to the goal node. A chromosome may be an array of vertex IDs, sorted according to the order that they are traversed in the solutions. The initial populations may initially be formed using an algorithm like DFS, where the order of the children nodes is randomized during each run of DFS and blocked nodes are removed to ensure they are not introduced into our population.
- What is your mutation function?
 - Randomly select a node in the path that is not the start and goal node, since that is fixed in our problem, and replace it with a vertex that the previous one in the path connects to. Note that we will never change a vertex to one that is blocked. If this node directly connects to the next node in the path, then stop mutation here, otherwise, insert the shortest possible sub path between this changed vertex and the next vertex in the path which can be found using the current A* implementation.
- How does crossover happen in this case?
 - Randomly select two sub paths of nodes that are directly connected from both parent chromosomes. The split should occur where the last point in the first parent sub path connects to the first node in the sub path from the second parent chromosome. If the two sub paths can't be directly connected for some reason, we can randomly choose a split point and we can find the shortest path between the last node of the first sub path to the first node of the second sub path using normal A*. Through these sub paths we can create valid offspring paths since we are always combining valid segments whose individual nodes are known to connect to each other, and finding a valid and the shortest possible path between the two segments when a direct connection between their last and first nodes respectively is not possible.
- How do you evaluate the quality of a chromosome in this scenario?
 - We can evaluate the quality of a chromosome (path) by setting the fitness value to the negative of the path length. Through this method, the path that is longer will have a higher magnitude negative value for its fitness value and our algorithm will attempt to select chromosomes with the least negative fitness value.