

RBE 550 Advanced Search Algorithms Implementation Documentation

Alex Tacescu — Spring 2021

I. INTRODUCTION

This document accompanies my homework 3 submission. It is designed to explain my thought process when developing my code. It will also explain the implementation of two different algorithms: Weighted A* and Rapidly-Exploring Random Tree (RRG).

II. WEIGHTED A*

Weighted A* is simply just A* with different weights for each of the two heuristics. A normal A* equation looks like:

$$f(n) = g(n) + h(n) \quad (1)$$

$g(n)$ is the cost from the start to current node and $h(n)$ is the estimated cost from the current node to the goal node. Weighted A* adds weights to each of these heuristics:

$$f(n) = G * g(n) + H * h(n) \quad (2)$$

G would be the weight for $g(n)$ while H would be the weight for $h(n)$. Weighted A* allows for better fine tuning to get faster or more precise results than traditional A*. If computation time isn't a concern, you can reduce the H weight and increase the G weight and the function will work closer to Dijkstra's algorithm. If you want a fast result and don't mind the final path to be the most optimal, then you can do the reverse and increase H while reducing G .

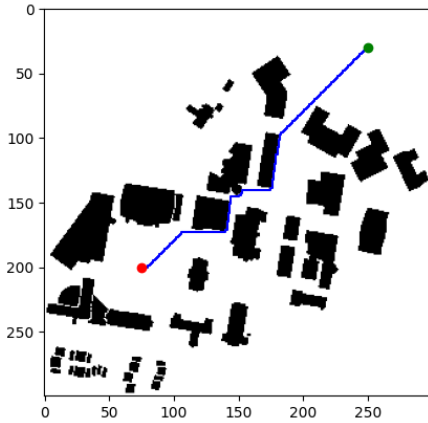


Fig. 1. A WA* run, with $G = 5$ and $H = 1$

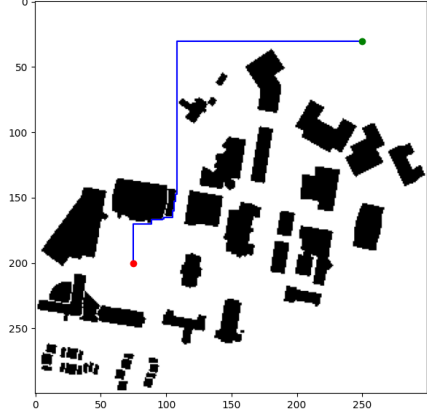


Fig. 2. A WA* run, with $G = 1$ and $H = 5$

III. RRG

RRG allows for more paths to be connected compared to a traditional RRT algorithm. The increase in connections between nodes allows for more possibilities and paths than RRT. This will give the final search algorithm more options to choose from, but it also adds more time for computation.

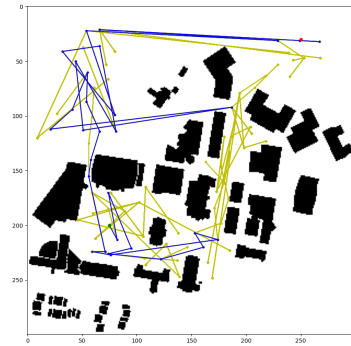


Fig. 3. Results from RRG Algorithm run