



My Programs ▶ ... ▶ Project: 3D Motion Planning ▶ Submit Project

Project: 3D Motion Planning

Submission Results

Submission Date: March 17, 2021



Submission Passed

Download Submission

Feedback Details

Specification Review Code Review

Reviewer Note

Greetings Learner Congratulations on passing your project! 🏂 Your project was very impressive and shows your outstanding efforts! We very much appreciate the time and interest you have put into this project for passing it. This application is very intelligently developed and you should be proud that you have been able to do so! I direly hope that you keep up the spirit and continue giving this amazing performance! Wishing you all the best.

Further resources:

- A Real-Time 3D Path Planning Solution
- Advanced Robotic Systems 3D Dynamic Motion Planning

Writeup

② Provide a Writeup / README that includes all the rubric points and how you addressed each one. You can submit your writeup as markdown or pdf.

Reviewer Note

Good job student! You have provided an amazing write up explaining each rubric point in a very detailed manner with images. 💍

The writeup / README should include a statement and supporting figures / images that explain how each rubric item was addressed, and specifically where in the code each step was handled.

Explain the Starter Code

Test that motion_planning.py is a modified version of backyard_flyer_solution.py for simple path planning. Verify that both scripts work. Then, compare them side by side and describe in words how each of the modifications implemented in motion_planning.py is functioning.

Reviewer Note

Bravo! You have properly explained the starter code which shows you have properly understood it as well!

The goal here is to understand the starter code. We've provided you with a functional yet super basic path planning implementation and in this step, your task is to explain how it works! Have a look at the code, particularly in the plan_path() method and functions provided in planning_utils.py and describe what's going on there. This need not be a lengthy essay, just a concise description of the functionality of the starter code.

Implementing Your Path Planning Algorithm

Write your search algorithm. Minimum requirement here is to add diagonal motions to the A* implementation provided, and assign them a cost of sqrt(2). However, you're (2) encouraged to get creative and try other methods from the lessons and beyond!

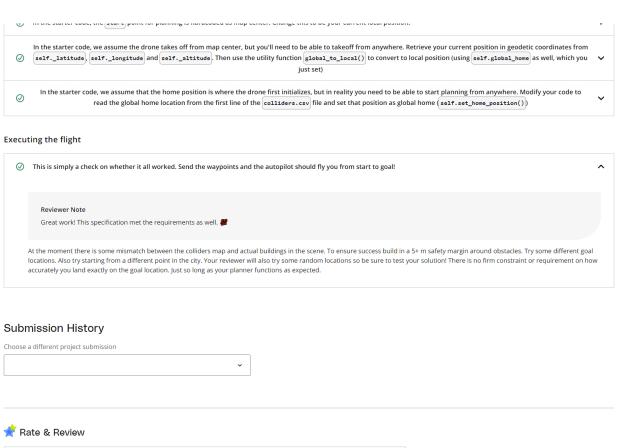
Reviewer Note

Excellent job here! You have used a great technique for this part and explained it very well. Hats off to you! 💍

Minimal requirement here is to modify the code in planning_utils() to update the A* implementation to include diagonal motions on the grid that have a cost of sqrt(2), but more creative solutions are welcome. In your writeup, explain the code you used to accomplish this step.

Cull waypoints from the path you determine using search.

In the starter code, the goal position is hardcoded as some location 10 m north and 10 m east of map center. Modify this to be set as some arbitrary position on the grid given any geodetic coordinates (latitude, longitude)





 ${\sf Click}\ {\sf next}\ {\sf to}\ {\sf continue}\ {\sf to}\ {\sf the}\ {\sf next}\ {\sf lesson}.\ {\sf You}\ {\sf may}\ {\sf return}\ {\sf and}\ {\sf submit}\ {\sf or}\ {\sf view}\ {\sf a}\ {\sf project}\ {\sf at}\ {\sf any}\ {\sf time}.$