ME 570: Robot Motion Planning

Homework 4 Report

By Cameron Cipriano

12/02/2021

**Problem 1: Graph Search**

Question 1.1 \_code\_: Graph.heuristic

Implemented using numpy’s linalg.norm() and subtracting the two points’ physical locations in the graph

Question 1.2 \_code\_: Graph.get\_expand\_list

Question 1.3 \_code\_: Graph.expand\_element

Question 1.4 \_code\_: Graph.path

Question 1.5 \_code\_: Graph.search

**Problem 2: Application of A\* to the Sphere World**

Question 2.1 \_code\_: SphereWorldGraph.\_\_init\_\_

Question 2.2 \_code\_: Graph.search\_start\_goal

Question 2.1 \_report\_: nb\_cells discretization

Diagram, engineering drawing

Description automatically generatedA picture containing text, mask, wheel, vector graphics

Description automatically generatedA picture containing text, wheel, gear

Description automatically generated

Top Left: Discretized with 8 Nodes

Top Right: Discretized with 40 Nodes

Bottom Left: Discretized with 80 Nodes

Question 2.2 \_report\_: SphereWorldGraph run\_plot method

Diagram

Description automatically generatedDiagram

Description automatically generated

Left: Discretized with 8 nodes goal location 1

Right: Discretized with 8 nodes goal location 2

A diagram of a heart

Description automatically generated with low confidenceA diagram of a heart

Description automatically generated with low confidence

Left: Discretized with 40 nodes goal location 1

Right: Discretized with 40 nodes goal location 2

A picture containing chart

Description automatically generatedA picture containing chart

Description automatically generated

Left: Discretized with 80 nodes goal location 1

Right: Discretized with 80 nodes goal location 2

Question 2.3 \_report\_: A\* Behavior given choice of nb\_cells

Given the choice of nb\_cells, A\* will either move very close to the optimal path at all times, or will tend to jump around in a zigzag pattern, as seen in the world discretized by 40 nodes. As the resolution increases (nb\_cells increases), the path will tend very close to the goal and not explore too many nodes around it.

Additionally, the higher resolution in the discretization will allow the A\* algorithm to find a path closer to the actual goal location in the real world.

Question 2.4 \_report\_: A\* Behavior with respect to the potential planner

The potential planner from homework 3 has no effect on the behavior of A\*

**Problem 3: Application of A\* to the Two-Link Manipulator**

Question 3.1(a) \_report\_: TwoLinkGraph.load\_free\_space\_graph

Question 3.1(b) \_report\_: TwoLinkGraph.plot

A picture containing diagram

Description automatically generated

Question 3.1(c) \_report\_: TwoLinkGraph.search\_start\_goal

Question 3.2 \_report\_: Plot the points obstacle\_points

Question 3.3 \_report\_: Comment on the *unwinding* phenomenon in the easy case

Question 3.4 \_report\_: Comment on the obstacle closeness issue and practical solutions

Given the fact that A\* is optimal given an optimistic heuristic, it is always going to try and hug the obstacles because that will be the shortest path between any start and the goal. In order to potentially avoid the closeness to obstacles, we can arbitrarily expand the borders of all obstacles. This expansion