

Introduction to Robotics

Assignment #6

Due: 11.07.2017, 23.59

Task 6.1 (8 points) Configuration Space – Programming Task:

Given is the planar 2-DOF manipulator as seen in Figure 1.

- The base is mounted at position $(500, 500)$ of the reference coordinate system.
- The manipulator is defined as
 - link length: $l_1 = l_2 = 200$
 - joint range: $\rho_1, \rho_2 \in [0, 2\pi)$
- The reachable workspace has a radius of 400
- The workspace holds two circular obstacles
- Obstacle o_1
 - origin: $(270, 620)$
 - radius: 50
- Obstacle o_2
 - origin: $(250, 200)$
 - radius: 200

6.1.1 (6 points): Use the mechanics of configuration space to shrink the manipulator to a single point. Assume the manipulator to have no physical links and the TCP to be a circle with radius 2. Plot the configuration space with the two C-obstacles and the point-sized manipulator. Discretize the configuration space to a precision of at least 1° .

6.1.2 (2 points): Plot the start area (circle with radius 10 around $s = (900, 500)$) and the two goal areas (circles with radius 10 around $g_1 = (580, 150)$ and $g_2 = (230, 470)$) in the workspace and in the configuration space. Draw a path from s to g_1 and from s to g_2 .

Task 6.2 (7 points) Configuration Space with Links – Programming Task:

Extend your program to include the physical properties of the links. Assume the links to be mounted at the very edge in the center and to have a width of 10. Draw a path from s to g_1 and from s to g_2 .

Task 6.3 (5 points) Arbitrary C-Obstacles – Programming Task:

Extend the program to support arbitrary polygon obstacles. Assume two new obstacles

- Obstacle o_3 : a rectangle with the following corner points
 - $(650, 450), (650, 200), (800, 200), (800, 450)$
- Obstacle o_4 : a polygon with the following corner points
 - $(600, 800), (550, 900), (750, 900), (700, 800), (650, 750), (600, 800)$

Plot the configuration space with C-obstacles $o_1 - o_4$. Draw a path from s to g_1 and from s to g_2 .

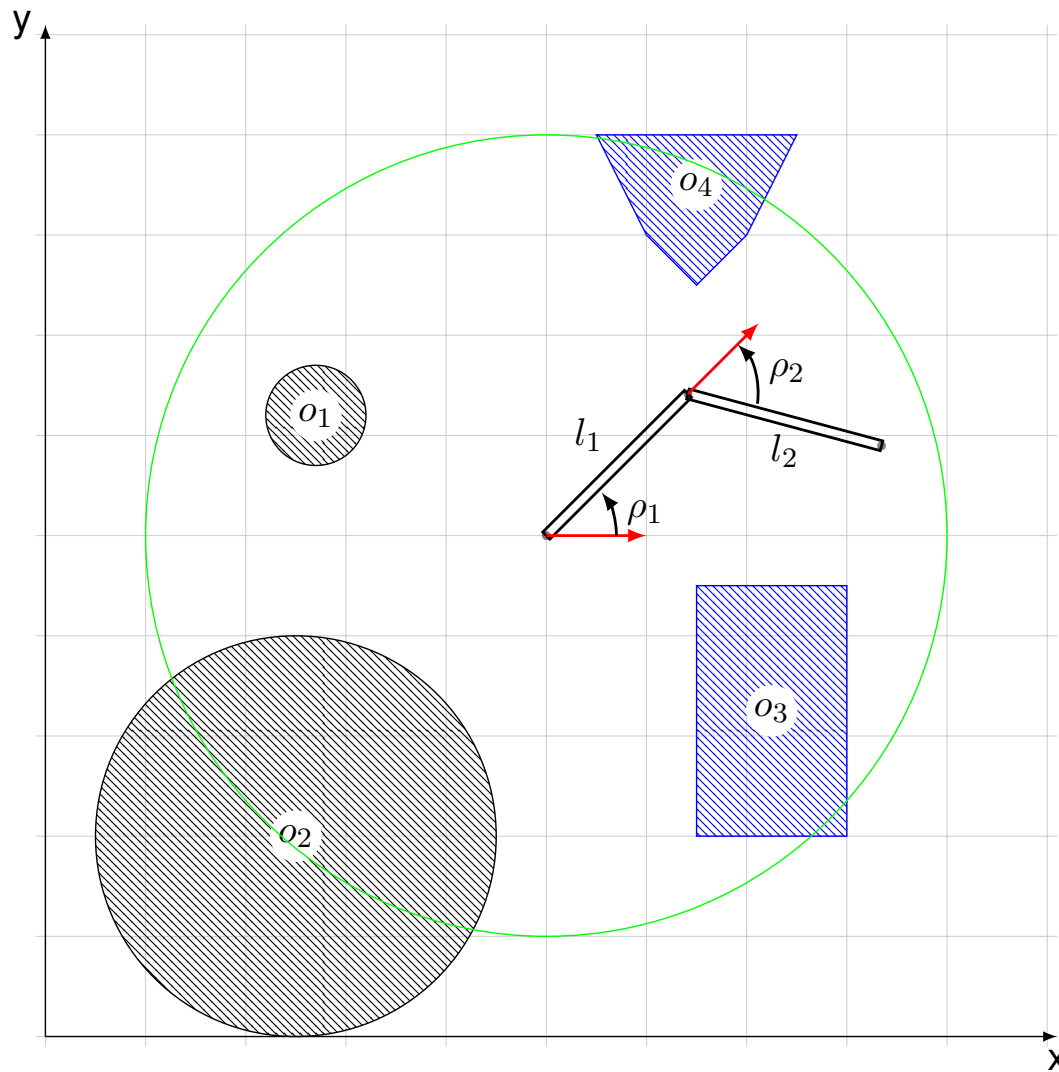


Figure 1: 2-DOF planar manipulator in workspace with obstacles. Grid distance is 100

Pay attention to the depicted direction of ρ_1 and ρ_2

The red arrows represent the zero position of the joints, the black arrows point in the positive rotation direction

Present the resulting plots as a PDF and pack your executable code (with all required libraries) as a ZIP file.

Hints:

When using python you might find useful:

Shapely library allows for polygons, circles (buffered points), affine transformations and intersections of objects <https://pypi.python.org/pypi/Shapely>

numpy is a powerful scientific library with fast array transformations and math functions <http://www.numpy.org/>

math math library grants access to sine, cosine, degree and radians <https://docs.python.org/2/library/math.html>

PIL(low) is the python imaging library which can export arrays to images <https://pypi.python.org/pypi/Pillow/2.2.1>

multiprocessing allows to speed up your code by using multiple processes <https://docs.python.org/3/library/multiprocessing.html>