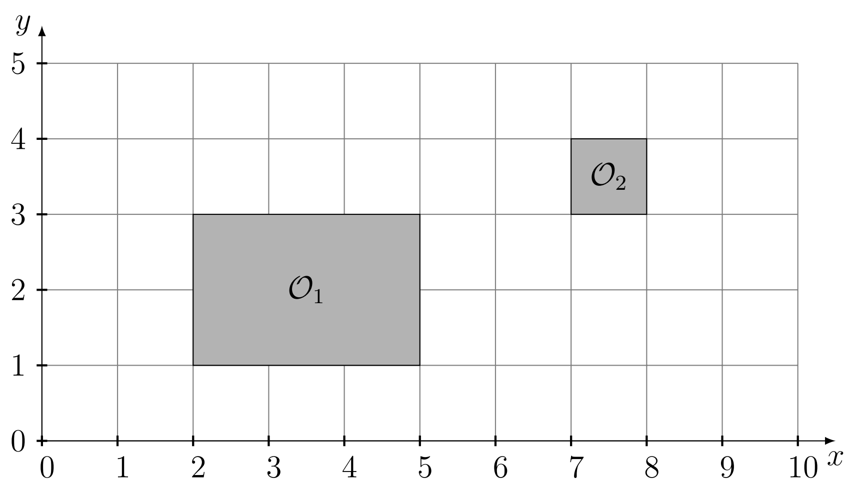
Motion Planning — Exercise 6

Wolfgang Hönig and Andreas Orthey, TU Berlin SS 2024

Non-Programming

1. Your goal is to find 3 samples for an angle, such that the dispersion with respect to the circular metric (see lecture 2) is minimized. Your first sample is $\mathbf{q}_1 = \frac{\pi}{2}$. What should the other two samples be, and what is the resulting dispersion?

2. Consider the following bounded workspace $\mathcal{W}$ with two obstacles (shown in gray).



   (a) Assume you are sampling the configuration of a point robot in 2D (i.e., $\mathcal{Q} \subset \mathbb{R}^2$) uniformly. What is the probability that your sample $\mathbf{q}$ lies on top of $\mathcal{O}_1$? Similarly, what is the probability that $\mathbf{q}$ lies on top of $\mathcal{O}_2$?

   (b) You are using rejection sampling. In what order should you test for collisions with the obstacles? Explain.

3. Consider the following set of 2D points with their (x, y) coordinates:

$$\{(4,3),(7,2),(2,9),(5,8),(3,1),(6,2)\}$$

   (a) The runtime of a query in an existing kd-tree depends on the depth of the tree. Draw an example of a "bad" kd-tree, i.e., a tree where the queries might take a long time. You can visualize the tree similar to the lecture, slide 29.

   (b) Draw an example of a "good" kd-tree. Explain what property of the tree makes it better than your solution in a).

(c) Based on your examples above, explain why the kd-tree construction is done offline given a given fixed dataset, rather than incremental point-by-point.

(d) List the steps needed to query the three nearest neighbors to the query point $(5, 5)$.

Programming

4. Implement and compare two samplers for a car-like robot with state $(x, y, \theta) \in SE(2)$.

   (a) Implement a distance metric for a car-like robot with state $(x, y, \theta) \in SE(2)$.
   Hint: For the position-component you can use the Euclidean distance, for the angular component you should use the circular metric (see lecture 2).

   (b) Implement a uniform sampler for a car-like robot.

   (c) Implement a Halton-sequence sampler for a car-like robot.

   (d) Implement a function that numerically estimates the dispersion of your samplers in b) and c). Compare the results for different numbers of samples (100, 1000, 10000).
   Hint: You may sample $\mathbf{q} \in \mathcal{Q}$ uniformly for the max-part of the dispersion equation.