# Assignment no. 4

due: Thursday, December 23rd, 2021

The exercises in this assignment refer to the setting of a disc (or two) moving among (not necessarily convex) polygonal obstacles in the plane, where the workspace is bounded within an axis-parallel rectangle.
Let $p$ be a free placement of the (center of the) disc robot, namely a placement where the disc does not intersect the obstacles. We assume that the robot is an open set and the polygonal obstacles are closed sets.
We define the *upward* (respectively *downward*) *vertical clearance* of a free placement $p$ of the disc as the maximal distance it can be moved vertically upward (respectively, downward) before hitting an obstacle, and denote it by $u(p)$ (respectively, $d(p)$).
The *vertical clearance* of such a free placement, $v(p)$, is defined as follows $v(p) = \min(u(p), d(p))$. The vertical clearance of a free path $\gamma$ is defined to be $v(\gamma) = \min_{p\in\gamma} v(p)$.

The code for Exercises 4.1 and 4.2 is in the `mrmp` module of the `DiscoPygal` suite. Installation instructions and guide are provided in the course's Moodle page.

Wherever you are asked to submit scenes, please provide figures of the scenes as well as scene files.

**Exercise 4.1 (p2)** Motion planning for a single disc robot.
**(a, 10 points)** Compare the running time of the exact solver vs. the PRM solver (both are given) on three different scenes. Then design one scene with a narrow passage, and make it progressively more narrow. Determine a measure of the *width* of the narrow passage, and plot a graph of the running time of the two algorithms as a function of this width of the narrow passage. For each fixed width, average the running time of the PRM over ten runs. Report on your findings.

The exact solver can be found in `mrmp/solvers/exact_single_disc.py` and the multi-disc version of PRM can be found in `mrmp/solvers/prm_discs.py`. For this exercise you should provide images of the scenes that you designed, together with an explanation of the choice of scenes, and tables + graphs of the comparison.

**(b, 30 points)** Modify the PRM implementation such that it will compute paths with high vertical clearance. Describe your solution in detail and report on experimental results.

Submit the description of your solution and experiments in a pdf file. Submit the implementation as a single Python file — `prm_hvc.py`.

Along the explanations in the pdf, submit also the Python file `exact_hvc.py`

**Exercise 4.2 (p2), 40 points** Motion planning for two unit disc robots.
Devise and implement a planner that aims to minimize the total distance traveled by the two robots, namely the objective is to minimize the sum of the lengths of the two paths for the robots. You can start with the given PRM (which supports $n$ disc robots, but you can of course assume there are only two) and modify and improve it with respect to this objective function, or you can implement an alternative solution altogether. Design three scenes on which you will demonstrate the quality of your solution. Describe your solution in detail, describe the scenes that you designed and explain why they were chosen, and report on experimental results.

Submit your results as pdf with explanations, images of the scenes and table/graphs of your choice. Also submit the Python file `prm_2_minlen.py`.

**Exercise 4.3, 20 points** Two unit disc robots are moving in the plane. One robot moves along the polygonal path $\pi_1 = u_0, u_1, \ldots, u_m$, namely, the robot starts with its center at the point $u_0$, moves along the line segment $u_0u_1$ and so on until it reaches $u_m$. The other robot moves along the polygonal path $\pi_2 = v_0, \ldots, v_n$. The robots are not allowed to move backward along their respective paths. In this exercise there are **no obstacles** in the workspace and the robots just need to avoid collision with one another. We are given a parameter $\Delta > 0$ and we need to determine whether there is a coordinated motion of the robots along their respective paths, such that at all times they are at least $\Delta$ away from one another. Design an efficient algorithm to decide whether such a motion exists. A possible way to describe a coordinated motion is to indicate for every position of the first robot along $\pi_1$, what is the position (which can be a point or a sub-path) of the other robot along $\pi_2$.

**Exercise 4.4** Choose a topic for your final personal project. It should include an experimental section. Write a short text, between 5 to 15 lines, about the problem you plan to address, and upload it to the Moodle page at the assigned spot. (Wait for approval of the project by the course's team before you start working on it.)