EN.530.663: Robot Motion Planning
Homework 6

Jin Seob Kim, Ph.D.
Senior Lecturer, ME Dept., LCSR, JHU

due: 03/12/2021, midnight to Gradescope

This is exclusively used for Spring 2021 EN.530.663 RMP students, and is not to be posted, shared, or otherwise distributed.

1. Solve Exercises problem 5 of Chapter 4.

2. Solve Exercises problem 8 of Chapter 4.

3. Prove that the configuration space obstacle of a convex mobile robot translating in a plane with a convex obstacle is convex.

4. Solve exercise problem 14 of Chapter 4.

5. (Exercises problem 11 of Chapter 4)

Suppose that $\mathcal{A}(0)$ and $\mathcal{O}$ each are defined as equilateral triangles with coordinates $(0, 0)$, $(2, 0)$, and $(1, \sqrt{3})$. Determine the C-space obstacle by using the primitive forms. Do not use any softwares or computer codes yet.

6. Write a Matlab function code for computing C-space obstacle of given convex polygonal robot and obstacles. In doing so, you will need to write a function code for the Minkowski difference that uses the Minkowski sum algorithm given in the class. For comparison, you can write a Matlab code that uses the brute-force approach (direct application of the difference vertex-by-vertex and use convex hull) to make sure that your function for the Minkowski difference works correctly.

The specification of the code is the following:

- The inputs include:
  - a cell array in which each element is a $2 \times n$ array (n: number of vertices) that contains the coordinates of the vertices of the obstacle (in CCW order).
  - a $2 \times m$ array (m: number of vertices) that contains the coordinates of vertices in $\mathcal{A}(0)$.
  - robot parameter $\mathbf{q} = (x_t,\ y_t,\ \theta)^T$.
- The output includes:

- C-obstacles as a cell array

Specifically,

(a) Write a function that computes C-space obstacle using Minkowski difference:
`[C_obs, (···)] = fn_c_obstacles(obs_vertices, A0_vertices, q)`
  - Input:
    - `Obs_vertices`: a cell array. Each element is a $2 \times n$ array (n: number of vertices) that contains the coordinates of the vertices of the obstacle (in CCW order).
    - `A0_vertices`: a $2 \times m$ array (m: number of vertices) that contains the coordinates of vertices in $\mathcal{A}(0)$.
    - `q`: a 3×1 vector of robot parameters $(x_t,\ y_t,\ \theta)^T$.
  - Output:
    - `C_obs`: a cell array of C-obstacles
    - Other outputs if necessary. Please put other outputs after `C_obs`.

(b) Write a main script that
  (i) Contains at least one example to test your `fn_c_obstacles()` function
  (ii) For debugging purpose, you can use brute-force approach for comparison.

For your code submission, please follow the guideline below.

Submission Guideline

- Submit your answers to problems 1 to 5 in a single pdf file to "HW6_analytical" on the Gradescope.
- Submit all your Matlab codes for problem 6 in a single .zip file to "HW6_computational" on the Gradescope. Name your single zip file submission as "YourName_HW6code.zip". For example, "JinSeobKim_HW6code.zip" for a single zip file.
- Please make sure to include *all the necessary files.* If TAs try to run your function and it does not run, then your submission will have a significant points deduction.
- Make as much comments as possible so that the TAs can easily read your codes.