EN.530.663: Robot Motion Planning
Final Project

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

out on 04/27/2022 (Wed); due on 05/09/2022 (Mon) by midnight EST

Instruction: You may use the textbook, your class notes, homework assignments, and materials provided in classes (e.g., handouts), as well as your Matlab codes used in all homework assignments. You can use a computer only for the indicated/necessary calculations (not to search the web for answers). No collaboration with other students or help from people outside the course (other than your team members) is allowed.

Task 1

In this task, you will perform the path planning of an n-link planar serial manipulator, with the proximal end fixed in space. Specific conditions are:

- $n \in \{4, 5\}$, i.e., choose one from the set for n.
- Each rigid link of the manipulator is considered as a line segment.
- There must be at least three convex-polygonal obstacles in $\mathcal{W}$, and the configuration must not be trivial.

Choose the dimensions, number of links (n), and non-trivial initial/goal configurations, $\mathbf{q}_I$ and $\mathbf{q}_G$, as you like. Then your task is:

(a) apply PRM algorithm; and

(b) apply RRT algorithm

to do the path planning. Each joint is assumed to be fully rotary (i.e., ignore self collision between the links).

Task 2

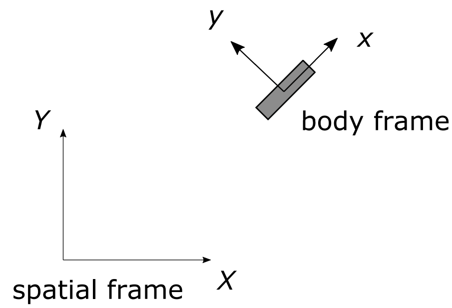

Figure 1: Task 2

In this task, you will perform the path planning of a holonomic planar rigid robot. The robot is represented as a rectangle. The body-fixed frame is attached as in Fig. 1 (i.e., the origin is located at the geometric center of the rectangle). There must be at least three convex-polygonal obstacles in $\mathcal{W}$, and the configuration must not be trivial. Specific dimensions and non-trivial initial/goal configurations, $\mathbf{q}_I$ and $\mathbf{q}_G$, can be freely chosen by your team. Then apply the artificial potential field (APF) method to do the path planning. Note that in the APF method, you don't need to fix the orientation of the robot during the planning, i.e., $\mathbf{q} = [x, y, \theta]^T$. Then you will have to use the APF method on non-Euclidean spaces. In other words, you have to choose control points $\{\mathbf{r}_i(\mathbf{q})\}$, especially for the repulsive potential functions. As discussed in the class, you will need to choose at least two or more control points on the robot. As one of the outputs, you generate a figure as in Fig. 2.

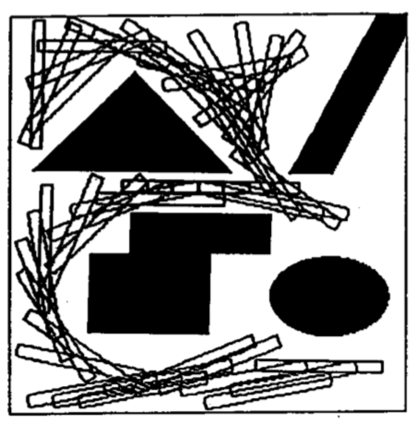

Figure 2: Task 2: an example of the plots to be submitted. This is from [1].

Task 3

In this task, you will consider *the path planning of a flexible needle* (planar problem), where the asymmetric needle tip is modeled as a nonholonomic mobile robot (geometrically modeled as an isosceles triangle) that moves in the plane. The configuration is shown in Fig. 3. The kinematic model of the needle tip is a unicycle. Then the trajectory of the unicycle robot (i.e., plot the

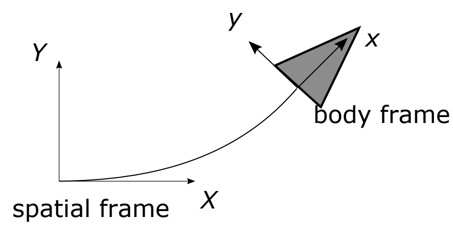

Figure 3: Task 3

trajectory of the origin of the body fixed frame) becomes the needle trajectory. For the unicycle model, see Section 13.1.2.3. The unicycle is simple yet is effective to other car-like system models (see Section 13.1.2 for the corresponding discussions). The direction of the unicycle (insertion of the needle) is along the x-axis of the body fixed frame.

In this task, two or more polygonal obstacles are to be used. Again, the configurations must not be trivial.

(a) Derive the kinematics equations (or state-transition equations) for the unicycle assuming that we have two actions: u_ϕ (or u_σ in Section 12.1.2.3) for the wheel rotation rate (or $\dot{\phi}$) and u_ω for $\dot{\theta}$. You can read the paragraphs around (13.18) and try yourself. Also derive the Lie-group-theoretic forms of the kinematic equations.

(b) Here in this task, assume that $u_\phi = 1$. Also assume that $u_\omega \in \{0, -\alpha, \alpha\}$. Define your own parameters for the radius of the wheel, α, and other dimensions. Then apply RRT to do the path planning with non-trivial goal configurations, $\mathbf{q}_G$ (specifically (x, y) of the needle tip; recall the explicit form of $g(x, y, \theta)$). Allow relatively large enough X_G for reaching the target (but not too large.) Note that $g_I = \mathbb{I}_3$, i.e., $\mathbf{q}_I = [0, 0, 0]^T$.

Note: In this case, u_ϕ is related to the needle insertion speed. Also, $u_\omega = \alpha$ becomes the constant curvature that the needle tip follows. Then $u_\omega = -\alpha$ corresponds to the case where you stop inserting the needle, rotate it by 180°, and then re-insert the needle. The case $u_\omega = 0$ corresponds to the case where you insert the needle while spinning it fast (100% duty cycle), which results in the straight needle insertion.

Deliverables

- Project report (in pdf format, maximum 8 pages) that contains detailed information on your system (i.e., dimensions of obstacles and the world, parameters, and so on), necessary modeling, resulting path plots (the sequence of robot configurations and obstacles in the workspace (world); trajectory of a needle in the world space by collecting the position of the tip), and the answers to the question. Authors should be the team members. Also the report should contain team workload distribution.
- *All* Matlab codes used in the project. Note that Matlab codes includes the following:

  - A main script file for each task that contains all the preambles, the main function, postprocessing of the results, and the process of movie generation. Other necessary functions may also be included
  - the main function file for each task:
    * Input: $\mathbf{q}_I$, $\mathbf{q}_G$; robot configuration A as a rectangle for task 2 (contains vertex coordinates in the body-fixed frame); B (a cell array that contains all convex polygonal obstacles); and other inputs necessary for each task.
    * Output: a resulting path (i.e., a sequence of $\mathbf{q}'s$); Vertex/Edge set and/or Weighted Adjacency matrix for the graph (choose any as necessary); and other outputs necessary for the task.
    * Regarding the format of polygons, use the convention that has been used in the class, i.e., CCW ordering of the vertices, and $B_i = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_k]$, not in $[\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_k, \mathbf{v}_1]$ format.
  - all the sub-function files.

- Movie files that show the path generated for task 1. Movies should show the robot path in the world (or workspace), not in the configuration space.

Project Submission Guideline

- Your team must submit a single zip file that contains all the deliverables. Name the single zip file submission as "Team#_NameInitials.zip". For example, "Team3_MX_ML.zip", if Michael Xu and Mingxu Liu are the team 3. Submission will be done through the Gradescope.

- Just in case when the zip file is too big to submit, then you can send your movie files via email to the TAs and the instructor. In all cases, name your movie file in the format of "Name_TaskNumber" (e.g., "Team3_MX_ML_Task1.mov").

- Please make sure to include *all the necessary files*, even files that were submitted in the previous homework assignments (of course the codes must be updated if necessary so as to be error-free). 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 your codes are readable.

References

[1] J. Barraquand, B. Langlois, and J.-C. Latombe. Numerical Potential Field Techniques for Robot Path Planning. *IEEE Transactions on Systems, Man, and Cybernetics*, **22**(2):224–241, 1992.