THU-70250403, Convex Optimization (Fall 2020)	**Homework: 8**

Complete Coverage Path Planning

Lecturer: Li Li li-li@tsinghua.edu.cn

Student:

Problem 1

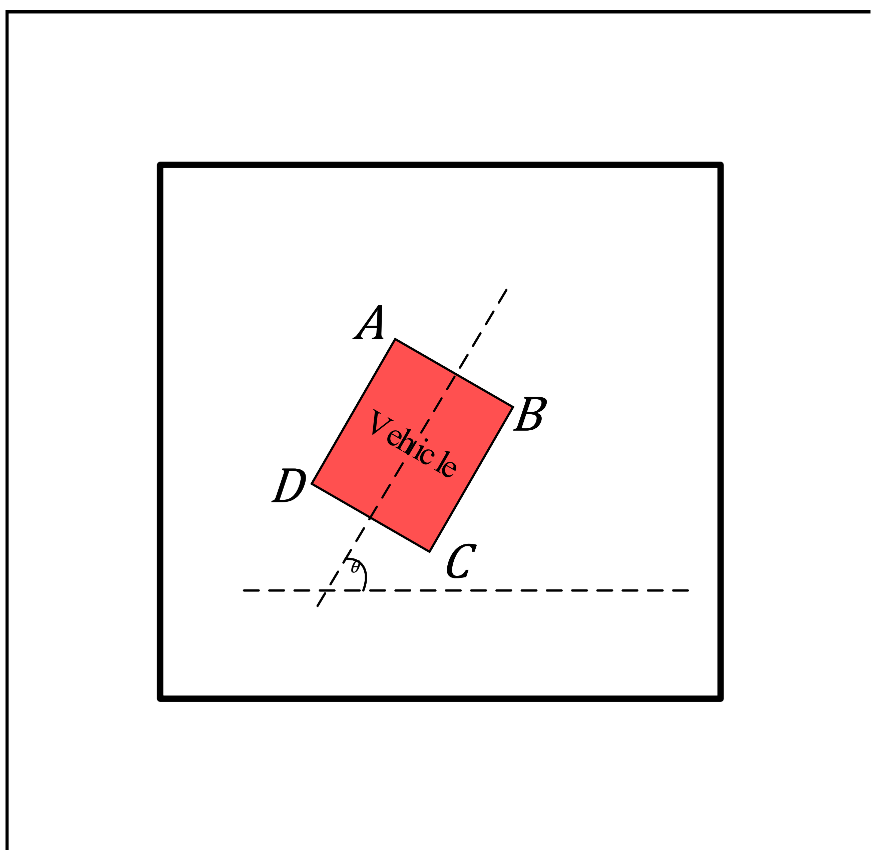

Figure 1: An illustration of the vehicle.

There is a cleaning vehicle of the size 0.57m×0.7m, see Fig.1 for an illustration. For the scenario shown in Fig.2, we set the scenario as a rectangular region of 10m×10m and set up a coordinate system. Four gray polygons labeled by letters represent obstacles and 14 points labeled by numbers represent "rubbish" which should be cleaned by the vehicle in Fig.2. Detailed information is shown in Appendix I.

Please plan a collision-free and smooth path for the vehicle to clean all rubbishes in this scenario, and minimize the path length as much as possible.

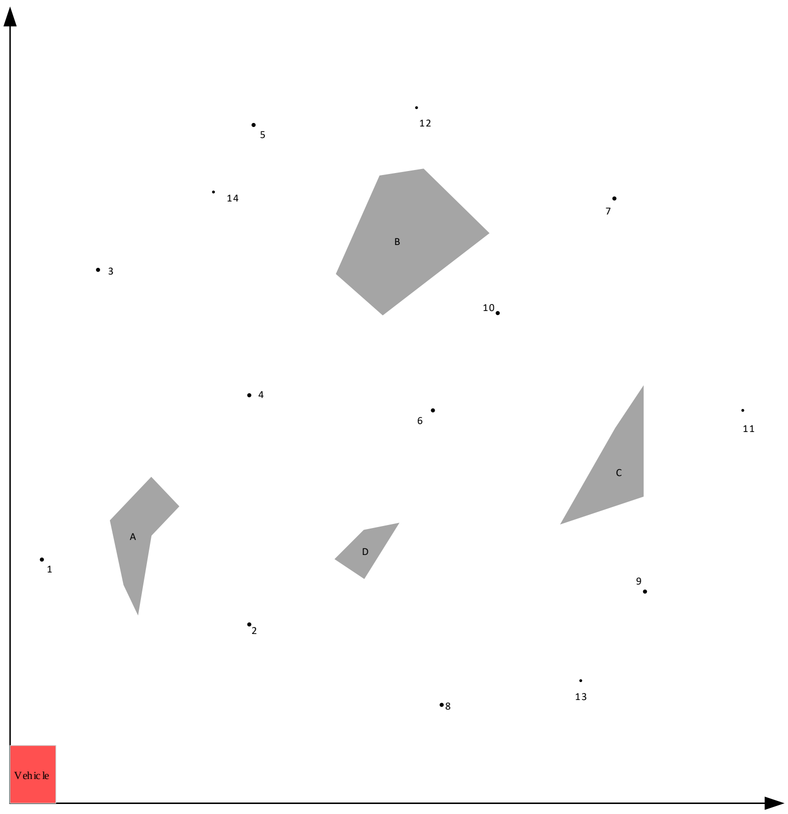

Figure 2: An illustration of the scenario.

The planning requirements are as follows:

1) minimize the length of the planned path as much as possible;

2) vehicle must avoid all obstacles;

3) vehicle must clean all "rubbishes";

4) the turning angle of the vehicle should **not be larger than 0.2π**.

We can characterize a path $\boldsymbol{z}$ by a series of $(n+1)$ waypoints (**less than 100**) sampled at equal intervals, i.e. $\boldsymbol{z} = [\boldsymbol{z}(0)^T, \boldsymbol{z}(1)^T, \ldots, \boldsymbol{z}(n)^T]^T$. $\boldsymbol{z}(k) = [x(k), y(k), \theta(k)]^T$ represents the state of the vehicle at the kth sampling moment, $P(k) = [x(k), y(k)]^T$ refers to the position of the center of the vehicle, $\theta(k)$ refers to the orientation angle of the vehicle,as shown in Fig.1.

Define the decision variables as $\boldsymbol{z}$ and consider the above requirements, the corresponding optimization problem can be formulated as below.

The Objective Function

To achieve the first requirement that minimize the path, we can define the objective function as

$$\min_{\boldsymbol{z}} \ J(\boldsymbol{z}) = \sum_{k=0}^{n-1} \sqrt{\big(x(k+1) - x(k)\big)^2 + \big(y(k+1) - y(k)\big)^2} \tag{1}$$

Constraints

To meet the second requirement, the rectangular vehicle should not intersect with all polygonal obstacles at all sampling moment. If the obstacles are convex, since both the rectangular vehicle and polygonal obstacles can be defined as intersecting halfspaces, we can check collisions by judging whether the feasible regions constructed by these halfspaces are empty. For non-convex obstacles, we can split them into convex sub-obstacles or approximate these non-convex obstacles by their convex hull.

To meet the third requirement, we can define the constraints as

$$\forall m_i, \exists j, \text{s.t.} m_i \in V(j). \tag{2}$$

where m_i refers to the position of the ith "rubbish" and $V(j)$ refers to the area occupied by the vehicle at the jth sampling moment.

As shown in Fig.3, we define the turning angle of the vehicle, $\alpha\big(P(k)-P(k-1), P(k+1)-P(k)\big)$, as the intersection angle between the vector $(P(k) - P(k-1))$ and $(P(k+1) - P(k))$. As mentioned above, $P(k) = [x(k), y(k)]^T$ refers to the position of the center of the vehicle at the kth sampling moment. Therefore, to meet the last requirement, we can define the constraints as

$$\alpha\big(P(k) - P(k-1), P(k+1) - P(k)\big) \leq \alpha_{max}. \tag{3}$$

where α_{max} is the maximum permittable turning angle.

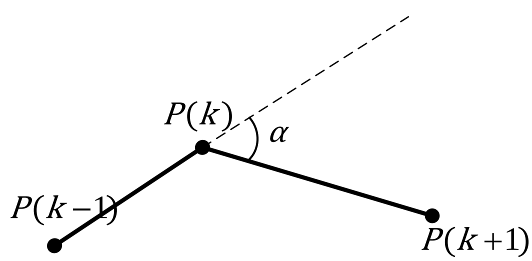

Figure 3: An illustration of the turning angle.

Notably, if necessary, you can sometimes penalize some constraints into the objective function, to simplify the optimization problem.

Appendix I

The vertex coordinates you need are prepared as follows:

Vehicle's start position: (0.00,0.00),(0.00,0.70),(0.57,0.00),(0.57,0.70);

Point 1:(0.40,3.00);Point 2:(2.90,2.20);Point 3:(1.10,6.50) ;Point 4:(2.90,5.00)

Point 5:(3.00,8.30) ;Point 6:(5.20,4.80) ;Point 7:(7.40,7.40) ;Point 8:(5.30,1.20);

Point 9:(7.80,2.60) ;Point 10:(6.00,6.00) ;Point 11:(9.00,4.80) ;Point 12:(5.00,8.50);

Point 13:(7.00,1.50);Point 14:(2.50,7.50);

Polygon A:(1.23,3.47),(1.40,2.67),(1.58,2.30),(1.75,4.00),(2.10,3.63);

Polygon B:(4.00,6.48),(4.52,7.68),(4.65,5.98),(5.06,7.73),(5.90,6.95);

Polygon C:(6.78,3.40),(7.78,3.76),(7.78,5.10);

Polygon D:(4.00,3.00),(4.35,3.35),(4.37,2.75),(4.80,3.45);