ME 145
Robotic Planning and Kinematics

Lab Session No. 6

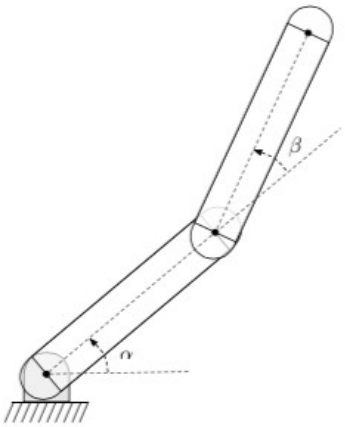
# FREE CONFIGURATION SPACE OF A TWO LINK ROBOT

**Instructions**

Submit your code through iLearn. Your code and reports are due on Monday, June 3rd ,11:59pm. No late submission will be accepted unless informed prior.

E4.3 **Sampling the Free Configuration Space for the 2-Link Robot (35 points)**

In this exercise you will determine the free configuration space for a two-link manipulator in a workspace with a circular obstacle.




The configuration of this manipulator is described by two angles, say α and β, and the configuration space is thus a 2-torus. It is not always easy to directly compute the free configuration space for multi-body robots. In this exercise, you are asked to grid the configuration space and test all grid points to determine whether they are safe (i.e., collision-free).

Assume that:

- $L_1$ and $L_2$ be the length of the first and second link, respectively.
- Each link is shaped like a rectangle with semi-circles at both ends, as in the figure.
- The environment contains 1 circular obstacle, with center $(x_o,y_o)$, and radius r.
- The angles α and β are expressed in radians.
- Each link has width 2W.

Write the following functions:

*plotEnvironment* (10 points)

Input: $L_1$, $L_2$, W, α, β, $(x_o,y_o)$, r

Output: the function plots the two-link manipulator defined by $L_1$, $L_2$, W, α, β, and the obstacle defined by $(x_o,y_o)$ and r

*checkCollisionTwoLink* (10 points)

Input: $L_1$, $L_2$, W, α, β, $(x_o,y_o)$, r

Output: the function returns 1 if the two-link manipulator defined by $L_1$, $L_2$, W, α, β collides with the obstacle defined by $(x_o,y_o)$ and r. The function returns 0 otherwise.

*plotSampleConfigurationSpaceTwoLink* (15 points)

Input: $L_1$, $L_2$, W, $(x_o,y_o)$, r, sampling_method, n

Output: the function plots the sampled free configuration space of the two-link manipulator defined by $L_1$, $L_2$, W. In particular, the function (i) determines n sample points in the configuration space according to the sampling method specified in the parameter sampling_method, (ii) draws a black dot at the sample (α, β) if the first link collides with the obstacle, a red dot at the sample (α, β) if the second link collides with the obstacle, and a blue dot otherwise. Use the sampling methods you developed in Lab 4.

*motionPlanningTwoLink* (20 points) (optional)

Input: $L_1$, $L_2$, W, $\alpha_0$, $\beta_0$, $\alpha_1$, $\beta_1$, $(x_o,y_o)$, r, sampling_method, n

Output: the function computes a collision-free path between the initial configuration $\alpha_0$, $\beta_0$ and the final configuration $\alpha_1$, $\beta_1$. The function first construct a roadmap of the free configuration space, by sampling n configuration points based on the method specified in sampling_method, and by connecting neighboring collision-free points. Assume that the path between two collision-free points is also collision-free. Second, function determines the closest sampled points (say $s_0$ and $s_1$) to the initial and final robot configurations. Third, the function uses the BFS algorithm to compute a path on the roadmap between $s_0$ and $s_1$. Finally, the function plots the environment with the robot and obstacle along the computed path.

Besides your Matlab functions, to show that your code works properly, you should turn in a brief document containing the output and plots of your functions for some example cases.