Problem 1

Given a set of k cells, each containing vertices describing unique configuration space obstacles

$$CB_1 = \begin{pmatrix} x_1 & x_2 & \dots & x_{n_1} \\ y_1 & y_2 & \dots & y_{n_1} \end{pmatrix} \tag{1}$$

$$CB_2 = \begin{pmatrix} x_1 & x_2 & \dots & x_{n_2} \\ y_1 & y_2 & \dots & y_{n_2} \end{pmatrix} \tag{2}$$

$$CB_i = \begin{pmatrix} x_1 & x_2 & \dots & x_{n_i} \\ y_1 & y_2 & \dots & y_{n_i} \end{pmatrix} \tag{3}$$

$$CB_k = \begin{pmatrix} x_1 & x_2 & \dots & x_{n_k} \\ y_1 & y_2 & \dots & y_{n_k} \end{pmatrix} \tag{4}$$

where CB_i, contained in cell i, represents the vertices of the ith configuration space obstacle which is a set of n_i vertices in $\Re^2$ relative to frame F_W ordered in a "CCW" fashion.

Assume you are given an initial and final robot position (noting that CB is defined for a fixed orientation) defined as q_{init} and q_{goal} respectively, and assume your environment is bounded by a *bounds* polygon defined:

$$bounds = \begin{pmatrix} x_1 & x_2 & \dots & x_p \\ y_1 & y_2 & \dots & y_p \end{pmatrix} \tag{5}$$

Create individual MATLAB functions to accomplish the following:

(a)

Define the approximate cell decomposition graph of the system. Use the function name "approxCellGraph" with the inputs q_{init}, q_{goal}, CB, and *bounds* as defined above. The function should return an $m \times m$ adjacency matrix Adj, an $m \times m$ weighted adjacency matrix $wAdj$, and a $2 \times m$ set of xy coordinates corresponding to the index values of Adj and $wAdj$. Use center of empty cells to define node locations and use the Euclidean Norm to define the weights for the weighted adjacency graph.

Note: Book keeping is the most difficult portion of this algorithm. Work out a solution by hand before implementing.

Hint: Use the same common index value for q_{init} and q_goal as you did with your "visibilityGraph" and "vCellGraph" method to avoid confusion.

Hint: Using the plotting tools created in previous homework assignments can be a very useful tool in debugging this algorithm!