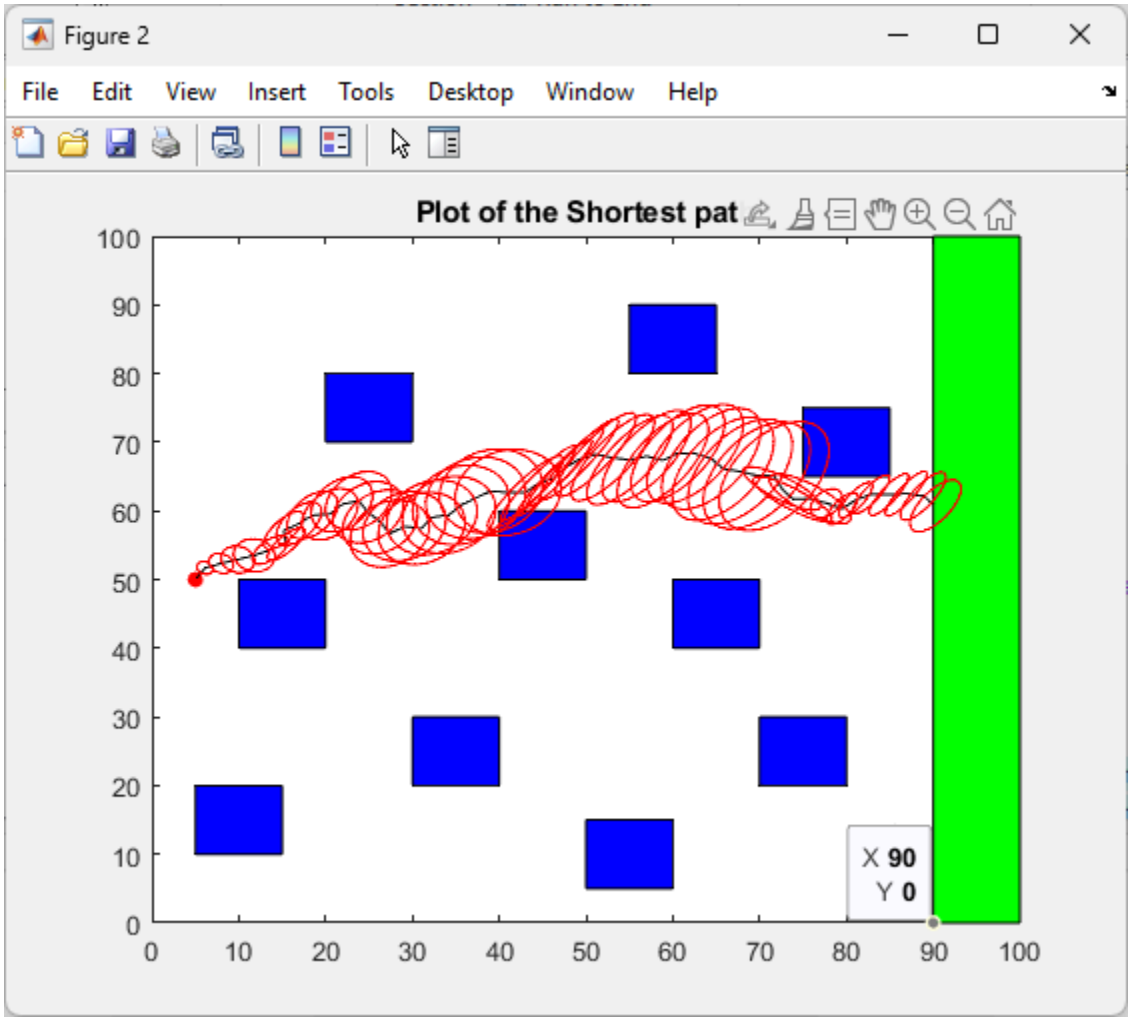
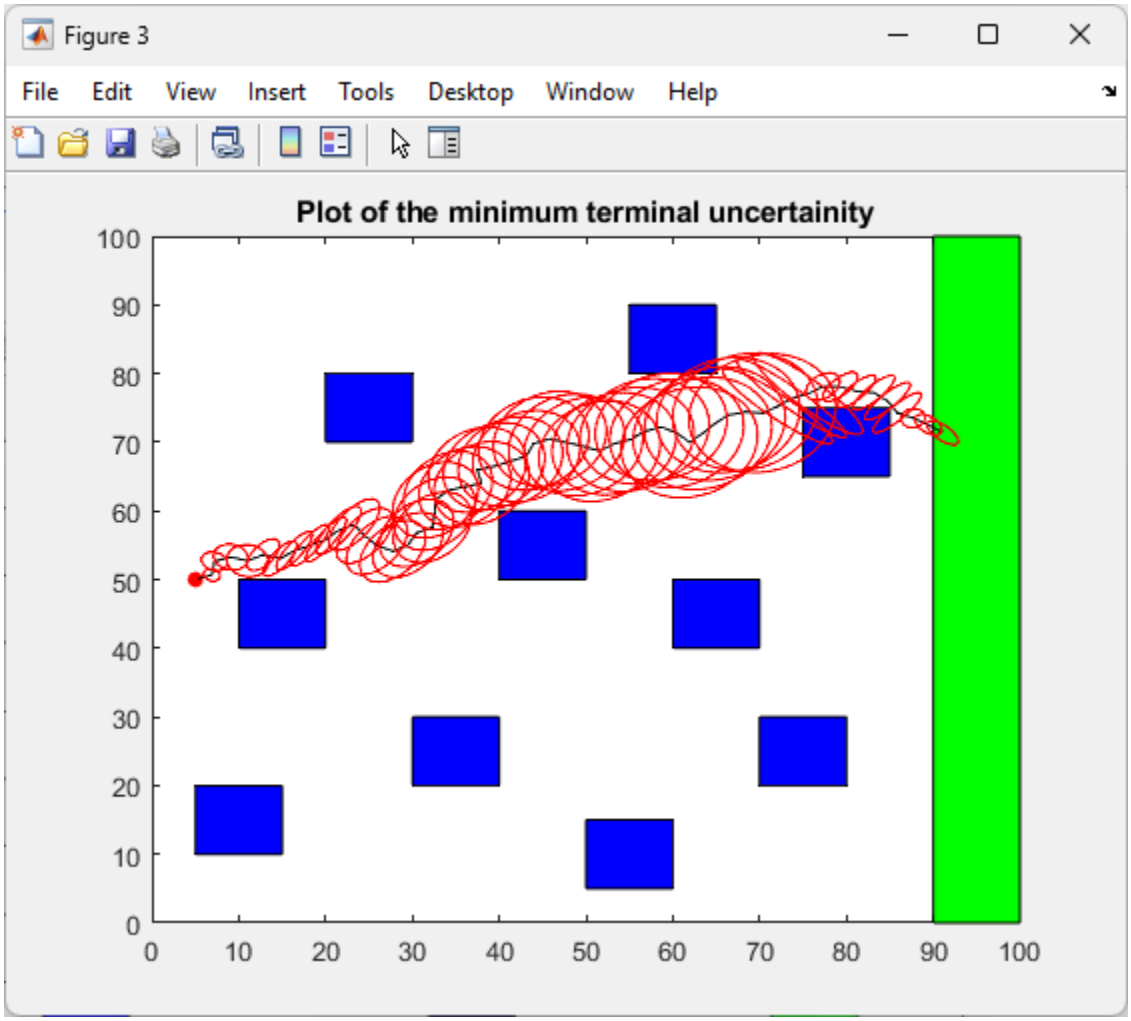
a) The matlab code for the RRT algorithm can be found in the ‘RRT.m’ matlab file in the folder, and the implemented kalman filter can be found in the ‘propogate_KF_path.m’ matlab file, and the ‘main.m’ matlab file will run the codes for the solutions and gives the output plots.

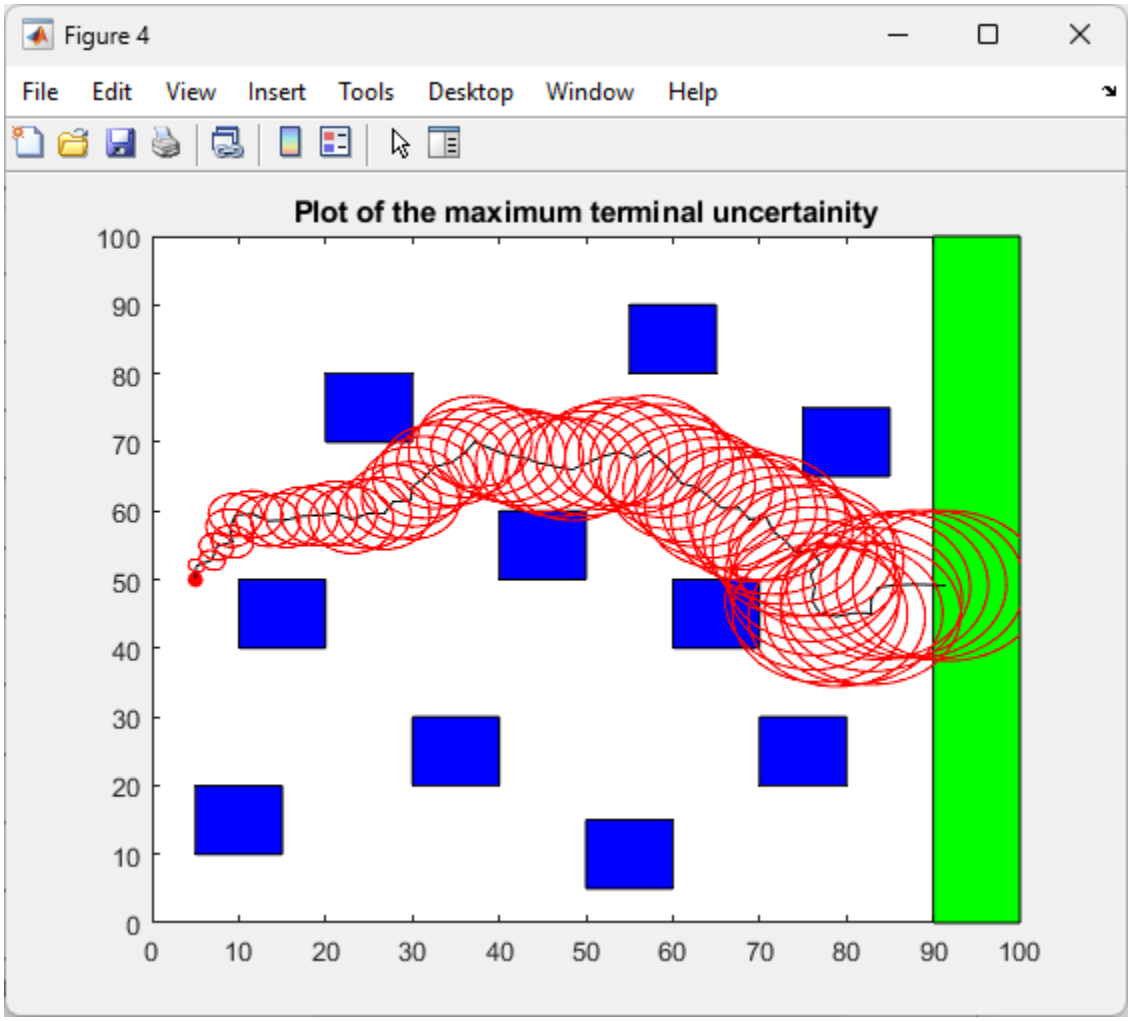
1. Plot of the shortest path obtained



2. Plot of the the path of minimum uncertainty at the terminal state



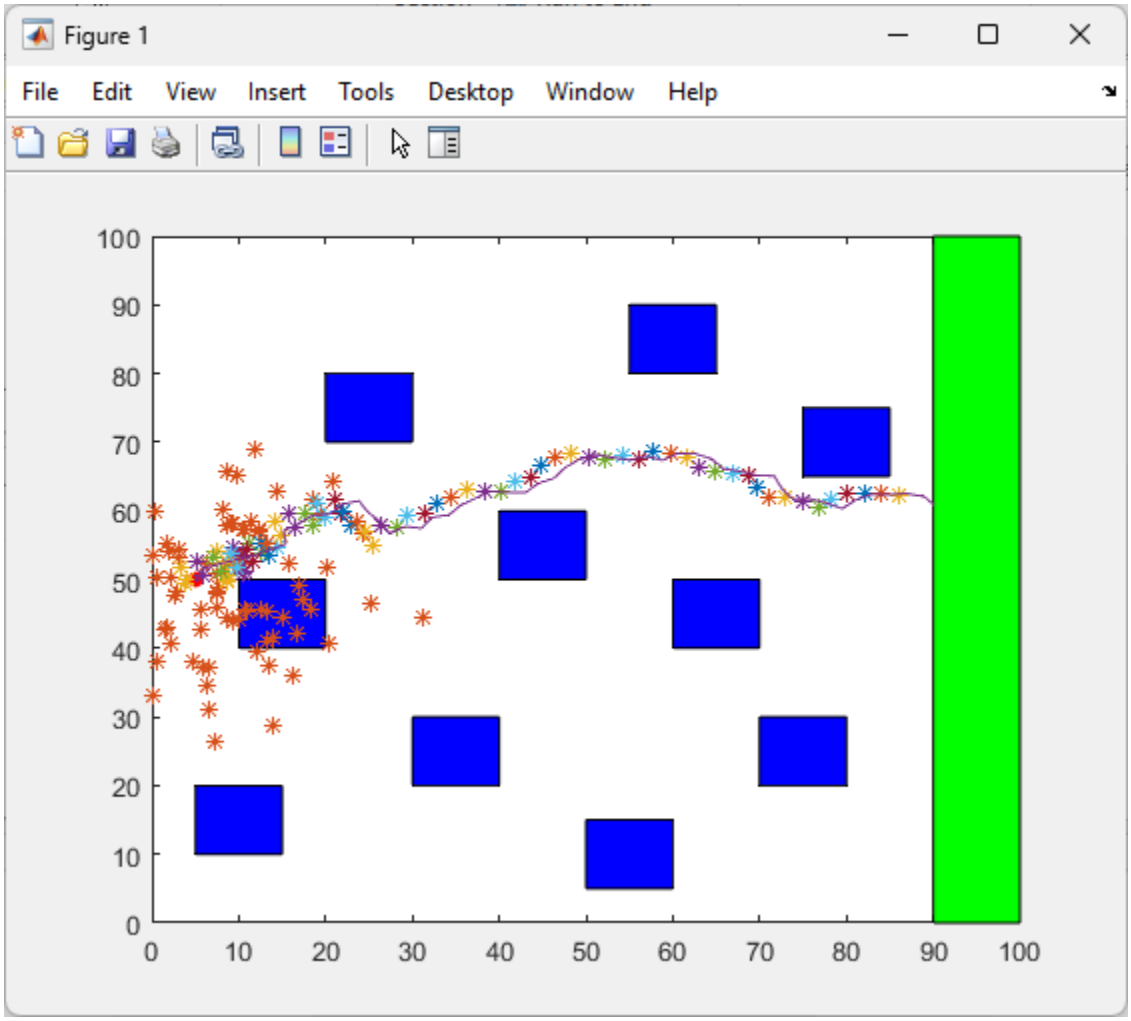
3. Plot of the path of maximum uncertainty at the terminal state



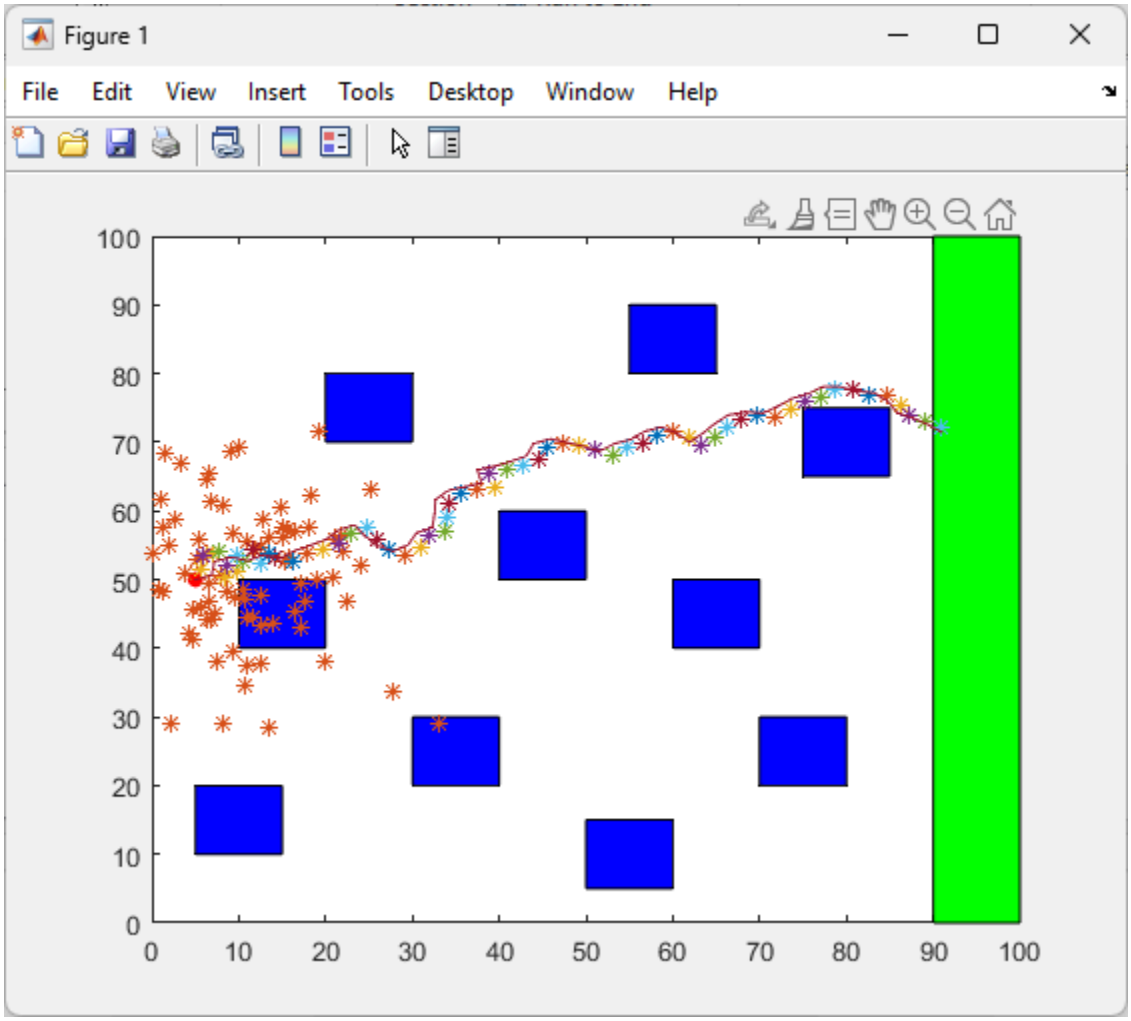
b) We can use a desired goal node to reduce the additional distance traveled by the robot in the goal region.
We can use more landmarks, so that the localization uncertainty induced along the robot's path can be reduced, and if we know the robot's dimensions we can include these dimensions in the path planning algorithm, to avoid it being in the close proximity of the obstacles, or else we can use the artificial potential fields algorithm for the path planning which will find a path equidistant from two obstacles, and therefore avoiding them.
To identify a reliable path we need to use as many as possible, since we are using the random states in each instance, we can get the necessary path as we desire by getting the path with the shortest length or with the minimum uncertainty.

c) The particle filter is implemented in the 'particle_filter.m' matlab file which takes the path as the input to implement the particle filter along the path, once the paths of the shortest length, min uncertainty and max uncertainty are obtained they can be used to call the particle_filter function.

1. Particle filter for the shortest path.



2. Particle filter for least uncertain path.



3. Particle filter for Max uncertain path