Prathamesh Mehta

RBE 550 :Motion Planning

Assignment - Transmission

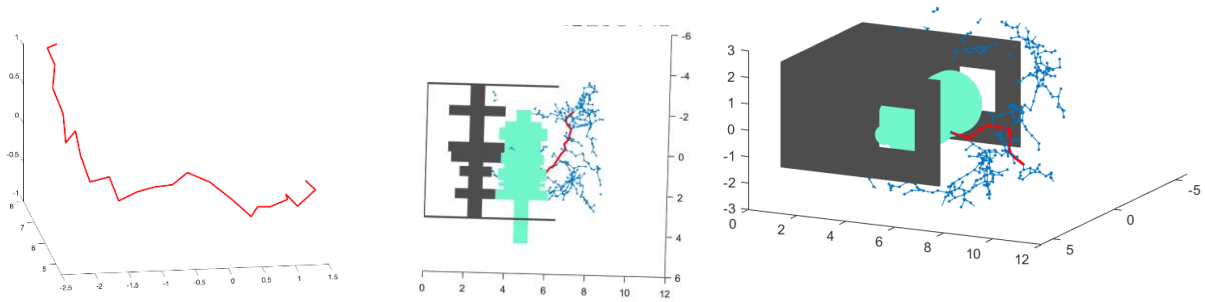
In this study, we propose a motion planning algorithm for a 6-degree-of-freedom rigid body Shaft to navigate out of the transmission engine's body. Our approach utilizes an RRT or RRT variant algorithm to construct and explore the roadmap in the given environment, accounting for the 6DoFs of the rigid body, which include x, y, z, roll, pitch, and yaw. The motion plan trajectory is then simulated in 3D visualization space. The proposed algorithm has the potential to improve the efficiency and accuracy of the motion planning process for similar applications.

Matlab was chosen as the simulation tool for this project over Python due to its user-friendly features for simulating 3D geometry and its built-in collision checking functions. In particular, the collisionbox() and collisioncylinder() functions were employed to define the transmission environment and shaft rigid body for this simulation. The use of these functions within Matlab allowed for a streamlined approach to accurately represent and simulate the motion planning environment while also enhancing the accuracy and efficiency of the collision detection process.

In this study, a custom class called "MakeEngine" was used to define the transmission engine object as a static engine block. This object was created using the collisionbox() and collisioncylinder() functions in Matlab, allowing for efficient collision detection. A variable "MakeShaft" was defined to store the main shaft element, which was also scaled down by a factor of 100. The configuration space bounds were established and stored in a variable called "c_space_bounds," which was then used to define a state space variable called "ss" using the built-in Matlab function StateSpaceSE3().

An occupancy map was created and stored in SV, and the plannerRRTStar function in Matlab was used to generate a roadmap and find the path from start to goal. The resulting motion path was then smoothed, and the animation function was utilized to visualize the trajectory.

The project was developed based on an example code for 2D rrt for furniture moving in 2D space found on the Matlab website. The utilization of custom classes and built-in functions in Matlab allowed for efficient implementation and accurate representation of the motion planning environment.



References:

http://smclassiccars.com/other-makes/195755-1971-fj40-vintage-mod-bbc-sm465-trans-lifted-33s-custom-ralph-lauren-interior.html

https://www.mathworks.com/help/nav/ug/moving-furniture-in-a-cluttered-room-with-rrt.html