SC 627 Assignment 3 Instructions

S Deepak Mallya

March 2022

Reference

"Motion Planning in Dynamic Environments using Velocity Obstacles" by Fiorini, et. al (Find a copy in MS Teams under General > Files > Reference Books)

Problem Statement

Plan and execute a trajectory to goal location in a dynamic environment using velocity obstacles method.

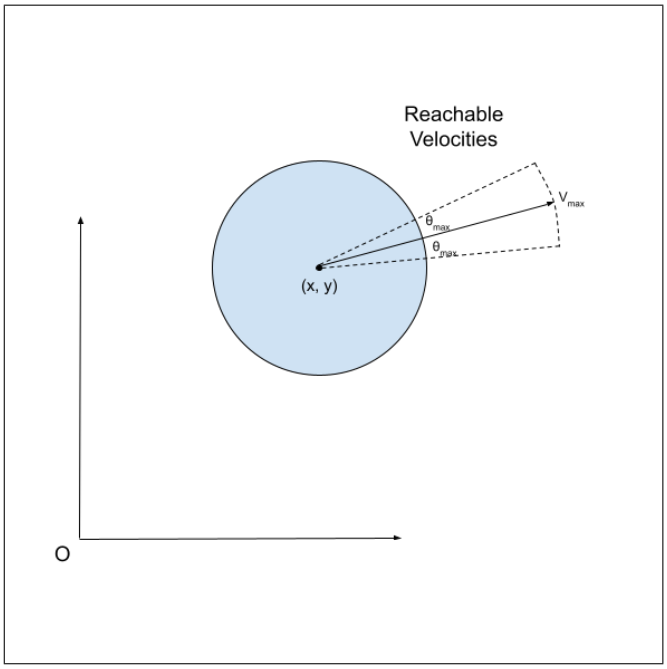

Figure 1: Reachable Velocities (V_{max} is shown towards the current robot heading)

- Environment and Robot Definition
  - Start = $(0, 0)$, Goal = $(5, 0)$
  - Robot and Obstacle Diameters, $D = 0.15$
  - Magnitude of Maximum Robot Velocity $V_{max} = 0.15m/s$
  - Maximum Deviation in Orientation $\theta_{max} = 10^{\circ}$
- At each instance, the obstacle data is available containing the pose (x_i, y_i) as well as velocities (V_{x_i}, V_{y_i}) of each obstacle i w.r.t to the fixed (global) frame of reference (Elaborated in Implementation Section)
- At each instance (as frequently as possible) command a velocity vector (V_x, V_y) that is outside the instantaneous velocity obstacle. You are free to use either TG, MV or ST strategy (check reference).
- Plot the robot's path and elaborate your work in the form of a report.

Requirements

- **Note:** Run the commands to get updated 'sc627_helper' folder
  - cd catkin_ws/src/sc627_helper
  - git pull origin master
  - cd ../..
  - catkin_make
  - source devel/setup.bash
- You'll find two new folders 'launch' and 'msg' in 'sc627_helper' folder and a few additional files in 'scripts' folder
- Create *assignment_3* folder (henceforth referred to as the folder) in 'sc627_assignments'
- The folder must contain the following files:
  - collision_avoidance.py
    This file contains your collision avoidance script (refer to assign3_skeleton.py in sc627_helper/scripts for outline)
  - robot_path.png
    Plot of robot path
  - Plot pos_x vs. time
    Plot of robot's x-coordinate vs time
  - report.pdf
    A detailed report of your implementation

Implementation

- The expected execution is as follows -
  - roslaunch sc627_helper collision_cone.launch
  - rosrun sc627_assignments collision_avoidance.py
- In order to implement the algorithm you can use 3 topics (refer assign3_skeleton.py file in sc627_helper/scripts)
  - /obs_data
    Contains obstacle position and velocity data
  - /bot_1/odom
    Contains robot's position and velocity (in local frame) data
  - /bot_1/cmd_vel
    Command robot's velocity
- You are also provided with a function 'velocity_convert' which you can use to convert velocities along cartesian coordinates to linear and angular velocities.
- NOTE: Kindly change the first line of each script in sc627_helper/scripts folder from 'python3' to 'python' if you are not using python3 and ensure that all the scripts are executable.

Submission

- Submission Deadline: $\mathbf{8^{th}}$ *April*, 2022 (For both Assignments 3 and 4)
- After creating the 'assignment_3' folder as described above run the following commands from the terminal
  - cd path_to_catkin_ws/src/sc627_assignments
  - git add .
  - git commit -m "assignment3_final"
    Assign this message only to the final version of your submission
  - git push -u origin master
    Verify that the folder is added to your github repository (online)
  - git log -pretty=oneline
    Copy the first string (the long one!) to the spreadsheet against your name under the appropriate column (`https://docs.google.com/spreadsheets/d/1bZN23JUzaHuUMvjP4L_9tu9Io85-VPG4_kNK7A25fTY/edit?usp=sharing`)