CSE-483: Mobile Robotics **Monsoon 2019**

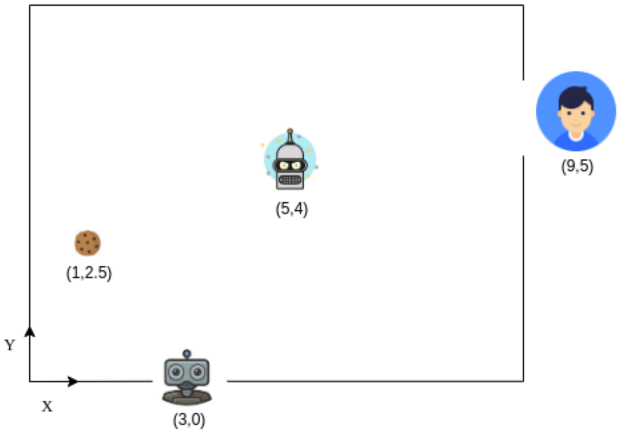
Assignment-5

Released: November 4th *Deadline: November 11th*

Instructions

- This assignment is designed to get you familiar with basic trajectory planning.
- Code can be written in Python or C++ only. Make sure your code is modular.
- Submit your code files and a report (or as a single Jupyter notebook if you wish) as a zip file, named as ⟨team_id⟩_assignment5.zip.
- The report should include all outputs and results and a description of your approach. It should also briefly describe what each member worked on. The scoring will be primarily based on the report.
- Refer to the late days policy and plagiarism policy in the course information document.

Q) Trajectory planning using polynomial functions



A service robot wakes up in a room at coordinates $(3, 0)$ at $t_o = 0$. Within the room, the robot has to visit the point $(1, 2.5)$ in order to collect a cookie for his friend. The robot then has to move to the other end of the room at $(9, 5)$ to give his friend the cookie, at $t_f = 5$. In addition, the robot must not crash into his friend, so his end speed must be equal to 0. Assume the robot is holonomic i.e. he can move along the x and y directions independently. Sample t from t_o to t_f in increments of 0.1.

(a) Compute the trajectory for this desired motion profile using a polynomial function of degree four. Recall that this amounts to formulating a linear system Ax = b based on the given constraints, for both x and y, and simply solving for the coefficients of the polynomial function. Plot the obtained trajectory indicating the start, end, and waypoint clearly. Also plot the corresponding velocity and acceleration profiles.

(b) Next, compute the trajectory as a fifth degree *Bernstein polynomial*, i.e. as a linear combination of the *Bernstein basis polynomials*. Such a polynomial is defined as,

$$f(t) = \sum_{i=0}^{n} {}^{n}C_i \tau^i (1-\tau)^{n-i} p_i$$

where p_i is the scalar coefficient, and $\tau = (\frac{t-t_o}{t_f-t_o})$. Plot the obtained trajectory indicating the start, end, and waypoint clearly. Also plot the corresponding velocity and acceleration profiles.

(c) To complicate things, an evil robot of radius 2 units appears at $(5, 4)$ to steal his cookie. The robot, whose radius is 1 unit, must avoid this evil robot in order to successfully deliver the cookie. Describe how you would compute a trajectory accordingly. [*Bonus*] points for implementing it.

R.O.B.O.T. Comics

"HIS PATH-PLANNING MAY BE SUB-OPTIMAL, BUT IT'S GOT FLAIR."