ELEE 4200/5200: Autonomous Mobility Robotics
Term I, 2018
Homework 4: Drive Robot to Goal (Quick & Smooth)

Guidelines:

- Due date: Thursday, October 18, 2018 by 12 Noon.
- Each group of no more than two students must work on its own in completing this assignment! Feel free to consult with others (and with me and the TAs) in developing solution ideas, but the final implemented code must be your work product alone. Refer to the Syllabus, where the policy on academic integrity is clearly outlined, our classroom discussion on this topic, and consult with me if you have any questions!
- State the full names and T# of the students in the group on the cover page of every document that you submit.
- Submit the report by responding to this assignment posting in Blackboard.
- The submission should at least include the following documents, bundled together into a single zip file with the name *YourNameHW4* (use one of the group member names).
  - The main report (following the template provided).
  - The main report in ‘pdf’ form.
  - The MATLAB program code.
- A hard copy (printout) of the ‘pdf’ report with MATLAB code; staple all pages together and follow the TA’s instructions on how to submit. If you don’t submit the hard copy, you will not get a grade for this homework!

Broad Objectives:

a) To construct a program in MATLAB that drives a simulated Turtlebot (in Gazebo) smoothly to a given goal. This assignment is a logical extension of the previous one, where we made the robot drive a path contour, but there was no specific target position.
b) To do so with the use of motion feedback (output) rather than calculations based on drive commands (input).
c) To understand and deal with the wrap-around aspect of angle measurements, due to their finite range of representation of ($-\pi$ to $+\pi$).

Specific Task Description: (read carefully and separate background problem information from requirements!)

a) Consider the available field to be a 10 m by 10 m square area. Think of it in terms of the “floor tiles” in Gazebo, each 1 m by 1 m. Remove all obstacles! The Turtlebot initially faces East and is in tile (10, 8), with the top left corner tile being (1, 1), which is also the goal (see Figure 1).

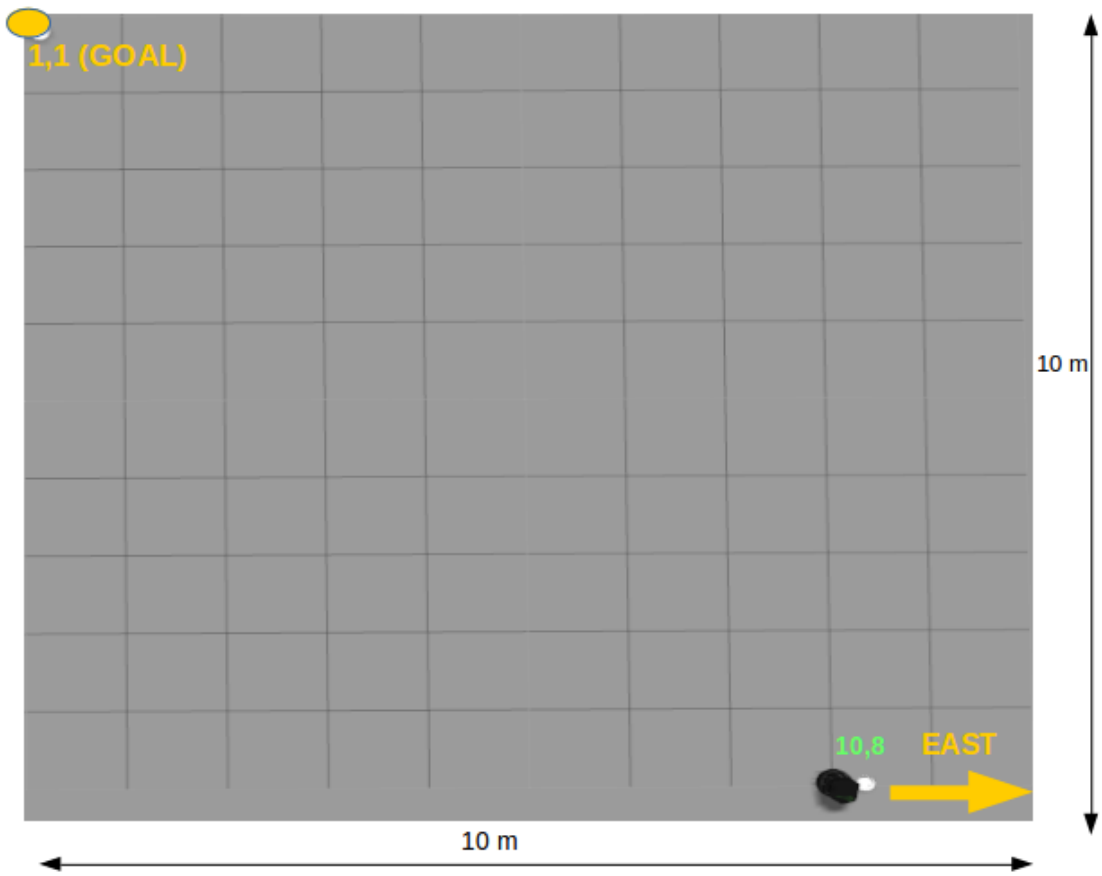

Figure 1

b) Drive the robot from the start position to the goal position as fast as possible and with as smoothly contoured a path as possible. The robot needs to be brought to a stop when it reaches the goal.

   Note: This is not necessarily a single circle or combination of circles. Think of how a human being would accomplish the move were he/she driving. This might give you some ideas. A pivot in place, followed by straight line motion to the goal is not allowed! That is, you need to be moving forward as you turn towards the target to. You cannot exit the field at any time!

c) You must use motion feedback (through odometry) to drive, not just input velocity commands. The former is an output, while the latter is an input! Once again, just like in an earlier assignment, you need to construct the solution as a simple control loop based on basic principles.

d) Plot the path (x, y), the forward velocity, v(t), and the heading angle θ(t), as a function of time during the robot travel. These are quantities based on what the robot is doing, not what you are commanding it to do!

e) Estimate the total travel time from start to goal and the total length of the path.

k

f) Using the results of part (d), make a focused argument on the smoothness of your drive. For example, the rate of change of $v(t)$ and $\theta(t)$ might serve as good indicators of smoothness!

Other information:

- Deal with angles carefully!
- Explain your algorithm clearly in your report!
- An important point to note is that *the process of arriving at the results is more important than obtaining results that appear correct, but you don't have any idea of how you got them!* This is embodied in the following sequence of actions – an algorithmic idea translated correctly to code, examination of the attendant behavior of the robot in multiple experiments, modification of the algorithm to change or correct behavior desirably, then repeat this sequence as necessary.
- Produce some good videos of robot runs and include them in your submission.
- If you want to make sure that, for repeated runs, the robot is being started with the same pose, you need to use the "set model state" command.