Project 3

Implementation of A* algorithm on a differential drive (non-holonomic) robot

A* on Differential Drive

- Navigate a differential drive robot (TurtleBot 2 / TurtleBot 3) in a Robotics Realization Lab's virtual environment from a given start point to a given goal point.
- Consider differential drive constraints while implementing the A* algorithm, with 8-connected action space (shown on slide 5).
- Either V-Rep or Gazebo can be used for simulation.
- The map of the RRL lab has been provided for both V-Rep and Gazebo. The dimensions of the map are also provided as an image and a PDF file for ease. The dimensions are in metres.

Differential Drive Constraints

- For this project you consider the robot as a non-holonomic robot, which means the robot cannot move in y-direction independently.
- You will have to define smooth moves for the robot by providing left and right wheel velocities. The time for each move will have to be fixed, and the time for each move defines the resolution.
- The equations for differential drive robot

$$\dot{x} = \frac{r}{2}(u_l + u_r)\cos\theta$$

$$\dot{y} = \frac{r}{2}(u_l + u_r)\sin\theta$$

$$\dot{\theta} = \frac{r}{L}(u_r - u_l)$$

Differential Drive Constraints (Continued..)

Where, $\dot{x}$ and $\dot{y}$ are the velocities in x and y directions

u_l and u_r are left and right wheel velocities

r is a wheel radius and L is the distance between two wheels

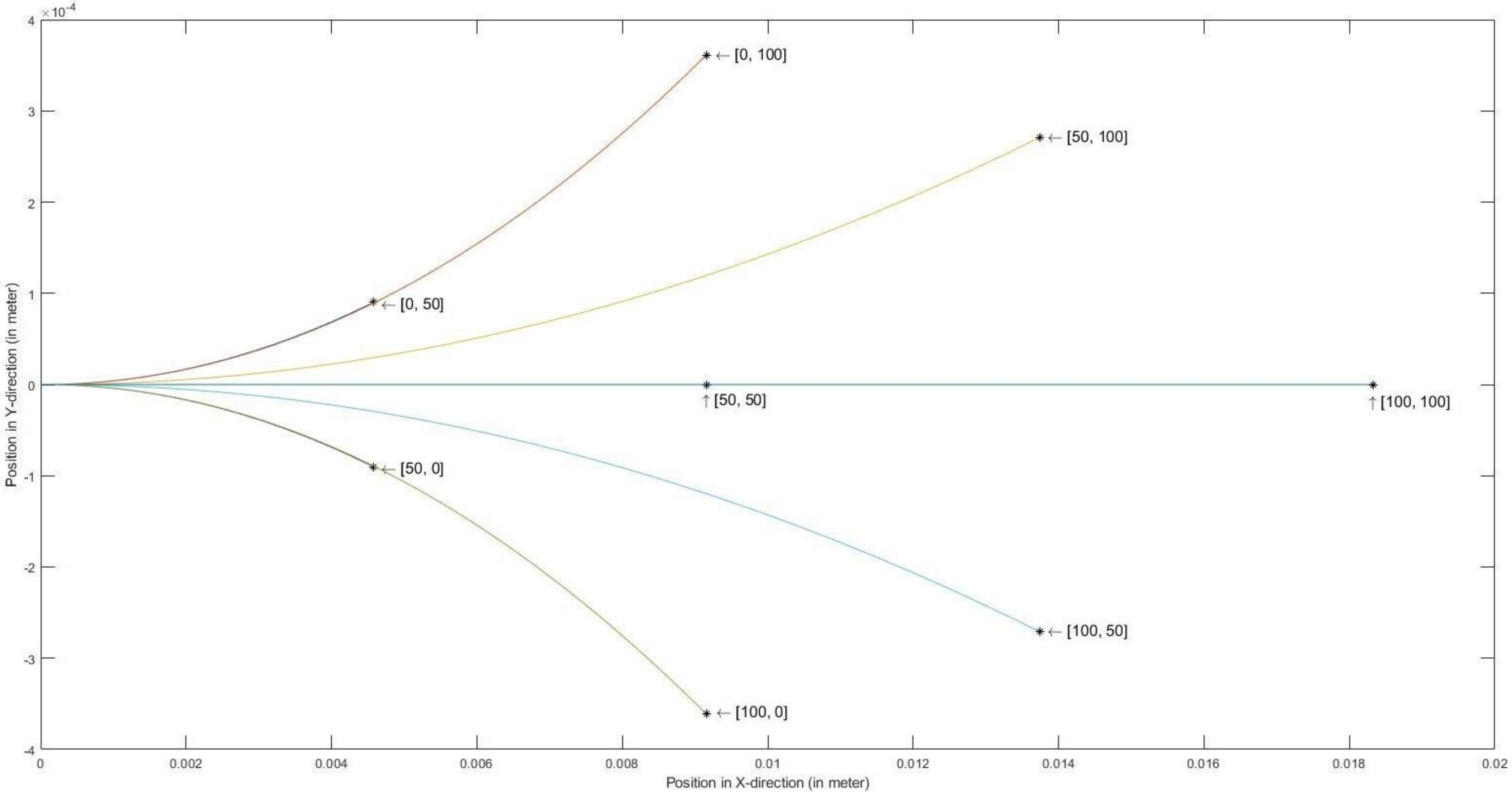
- From the velocity equations we can calculate the distance travelled and angle covered in each time step

$$dx = \frac{r}{2}(u_l + u_r)\cos\theta\ dt$$

$$dy = \frac{r}{2}(u_l + u_r)\sin\theta\ dt$$

$$d\theta = \frac{r}{L}(u_r - u_l)\ dt$$

Differential Drive Constraints (Continued..)



- The figure shows various curvatures obtained by changing left and right wheel velocities.

Action Space

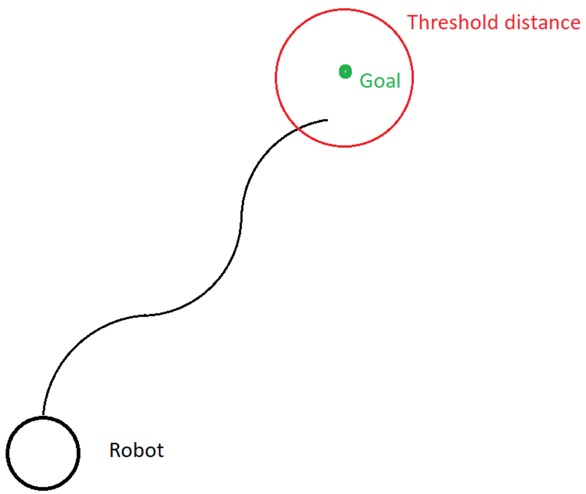
- Let the two RPMs provided by the user are RPM1 and RPM2 (see slide 8 for all inputs from the user). Then the action space for the A* algorithm are:

1. [0, RPM1]
2. [RPM1, 0]
3. [RPM1, RPM1]
4. [0, RPM2]
5. [RPM2, 0]
6. [RPM2, RPM2]
7. [RPM1, RPM2]
8. [RPM2, RPM1]

- Here the first element corresponds to the left wheel RPM and the second element corresponds to the right wheel RPM.

Differential Drive Constraints (Continued..)

- Note - You have to define a reasonable threshold value for the distance to the goal point. Due to the limited number of moves, the robot cannot reach the exact goal location, so to terminate the program a threshold distance has to be defined.



Inputs from the User

- Your code must take following values from the user:

1) Start Point Co-ordinates (2-element vector)
2) Goal Point Co-ordinates (2-element vector)
3) Wheel RPMs (2-element vector) => Two possible values for the wheel RPMs

Parameters to be Defined

- Your code must take the following parameters into consideration:

1) Robot Diameter (from the datasheet)
2) Wheel Distance (to be computed using the datasheet)
3) Reasonable Clearance
4) Sampling Frequency (required for simulation)

Note that, these parameters are not defined by the user. These are the parameters you need to consider while developing the code.

Deliverables

- Source code for the A* implementation on differential drive robot
- Simulation results (video of the simulation)
- Students who will be working on practical implementation on TurtleBot 2 / TurtleBot 3 have to convert velocity of left and right wheel into robot velocities ($\dot{x}$, $\dot{y}$, $\dot{z}$, $\dot{\alpha}$, $\dot{\beta}$, $\dot{\gamma}$) and save the velocities into a text file. Note that, for implementation a fixed sampling frequency will be provided.

Submission Details

- You are required to submit a zip file with the file structure as shown

proj3_firstname_lastname_codingLanguage_simulationSoftware

— codes

— readme.txt