Robotarium task 1

Tobias Hallingstad

September 25, 2020

1 Task undertaken

This task was about making a robot (in a simulation and in real life) move along a ploted corse. The corse was going to be like a infinity symbol. I decided to write my own motion algorithm and write a function to draw the corse from a few variables.

2 Motion planning algorithm

Calculate the angle To make the robot follow the given corse, i created the motion algorithm my self. I set out to create a "simple" way of finding the angle difrence from the robot to the next point of the corse. I get the angle by using this function in python:

```python
def calcAngle(rob, x2, y2):
    theta = math.atan2( (y2-rob[1]) , (x2-rob[0]) )

    turn = 3 * np.sin( rob[2] - theta )
    return -1*turn
```

Afther i have calculated the angle I then calculate a extra turning speed. This means that if the next point is straight forward the robot will do small corrections, and bigger corrections if the angle is large.

Calculate the speed I also have a function to calculate the speed of the robot. This function is designed to increas the speed of the robot when the next point is far way and reduse the speed when the point gets close.

```python
def calcSpeed(rob, x2, y2):
    length = math.sqrt( math.pow(x2-rob[0], 2) +
        math.pow( y2-rob[1], 2) )

    return 5 * length if length > 0.2 else 0.2
```

This function is not perfect, the the robot will contantly change speed and it is no max speed defined. The function works fine in the simulation because of the short distances I am working with.

3 Results simulation

Running the simulation the robot works as intended. When the angle between the robot and the goal is large, the robot turns fast, then slows down. When the robot arives at the goal the robot drives to the next goal.

I did have some problems with my code before, i was talking with a friend and i implemented some code he used with out thinking about if that code was going to work for me. This resulted in some time trubleshooting the code, but in the end my algorytm worked fine.

4 Results at Robotarium

At first i had a while loop that looped to a count, i incremented the count when i hit the first goal. This way seemt not to work. I then changes the loop to a for loop and the robot started to move.

The robot is defently not running perfect, it is overcorrecting some, and not always in the middle of the track, this is probaly some of my code not beeing the totaly best, and the robot not having the most acurate sensors. In general the robot is doing what i have told it to do.

When we are going to work on a robot we have then I think it is smart to test more and fine tune the variables, or try to implement a more advansed algorithm.