Assignment 2: Robot navigation

a)

From the subject we know that the robot is moving along the X-axis.

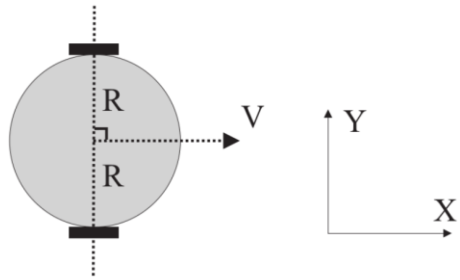

Figure 1.1 Robot position

We assume that ICR is located at a distance L from the center of the robot, so

$$v_L = \dot{\varphi}(L - R)$$

$$v_R = \dot{\varphi}(L + R)$$

So, $v_L = V - \dot{\varphi}R \rightarrow \dot{\varphi}R = V - v_L \rightarrow \dot{\varphi} = \frac{V - v_L}{R}$

Then, put $\dot{\varphi}$ into v_R, we solve the Robot velocity $V = \frac{(v_R + v_L)}{2}$

$v_L = V - \dot{\varphi}R \rightarrow V = v_L + \dot{\varphi}R$ put V into v_R, $v_R = v_L + 2\dot{\varphi}R$

We get the angle velocity $\dot{\varphi} = \frac{v_R - v_L}{2R}$

Finally, we calculate each velocity separately,

$$V_x = V cos\varphi$$

$$V_y = V sin\varphi$$

From the robotic kinematics we know that, at time t_1, the pose of the robot is (x_1, y_1, φ_1). By using the following formulas, we can calculate the exact value.

$$x_1 = x_0 + \int_{t_0}^{t_1} V_x(t)dt = x_0 + \int_{t_0}^{t_1} \frac{v_R(t) + v_L(t)}{2} cos\varphi(t)dt \quad (1-1)$$

$$y_1 = y_0 + \int_{t_0}^{t_1} V_y(t)dt = y_0 + \int_{t_0}^{t_1} \frac{v_R(t) + v_L(t)}{2} sin\varphi(t)dt \quad (1-2)$$

$$\varphi_1 = \varphi_0 + \int_{t_0}^{t_1} \dot{\varphi}(t)dt = \varphi_0 + \int_{t_0}^{t_1} \frac{v_R(t) - v_L(t)}{2R} dt \quad (1-3)$$

The question gives us the following conditions, so we know the left wheel speed and right wheel speed,

$$v_L(t) = v_0(t/t_1) \quad (1-4)$$

$$v_R(t) = v_0(t/t_2) \quad (1-5)$$

Then, the robot starts at $x_0 = 0, y_0 = 0$, $\varphi_0 = 0, t \in [0, 10]$, We bring the known conditions into the above equation.

$$x_1 = \int_0^{t_1} V_x(t)dt = \int_0^{t_1} \frac{v_0\left(\frac{1}{t_2}+\frac{1}{t_1}\right)*t}{2} cos\varphi(t)dt = \frac{v_0\left(\frac{1}{t_2}+\frac{1}{t_1}\right)}{2}\int_0^{t_1} t\cdot cos\varphi(t)dt \quad (1-6)$$

$$y_1 = \int_0^{t_1} V_y(t)dt = \int_0^{t_1} \frac{v_0\left(\frac{1}{t_2}+\frac{1}{t_1}\right)*t}{2} sin\varphi(t)dt = \frac{v_0\left(\frac{1}{t_2}+\frac{1}{t_1}\right)}{2}\int_0^{t_1} t\cdot sin\varphi(t)dt \quad (1-7)$$

$$\varphi_1 = \int_0^{t_1} \dot{\varphi}(t)dt = \int_0^{t_1} \frac{v_0\left(\frac{1}{t_2}-\frac{1}{t_1}\right)*t}{2R} dt = \frac{v_0\left(\frac{1}{t_2}-\frac{1}{t_1}\right)}{2R}\int_0^{t_1} t\,dt \quad (1-8)$$

calculate φ_1

$$\varphi_t = \frac{v_0\left(\frac{1}{t_1}-\frac{1}{t_2}\right)}{2R}\cdot\frac{t^2}{2} = \frac{v_0\left(\frac{1}{t_2}-\frac{1}{t_1}\right)}{4R}\cdot t^2 \quad (1-9)$$

Then, bringing φ_1 into x_t, y_t

$$x_t = \frac{v_0}{2}\left(\frac{1}{t_2}+\frac{1}{t_1}\right)\int_0^{t_1} t\cdot\cos\left(\frac{v_0}{4R}\left(\frac{1}{t_2}-\frac{1}{t_1}\right)\cdot t^2\right)dt \quad (1-10)$$

$$y_t = \frac{v_0}{2}\left(\frac{1}{t_2}+\frac{1}{t_1}\right)\int_0^{t_1} t\cdot\sin\left(\frac{v_0}{4R}\left(\frac{1}{t_2}-\frac{1}{t_1}\right)\cdot t^2\right)dt \quad (1-11)$$

Then bring in the constants,

$$\varphi_t = \frac{v_0\left(\frac{1}{t_1}-\frac{1}{t_2}\right)}{2R}\cdot\frac{t^2}{2} = \frac{5}{48}\cdot t^2 \quad (1-12)$$

$$x_t = \frac{3}{40}\int_0^{t_1} t\cdot\cos(\frac{5}{48}\cdot t^2)dt \quad (1-13)$$

$$y_t = \frac{3}{40}\int_0^{t_1} t\cdot\sin(\frac{5}{48}\cdot t^2)dt \quad (1-14)$$

I plot the resulting trajectory by MATLAB as follow:

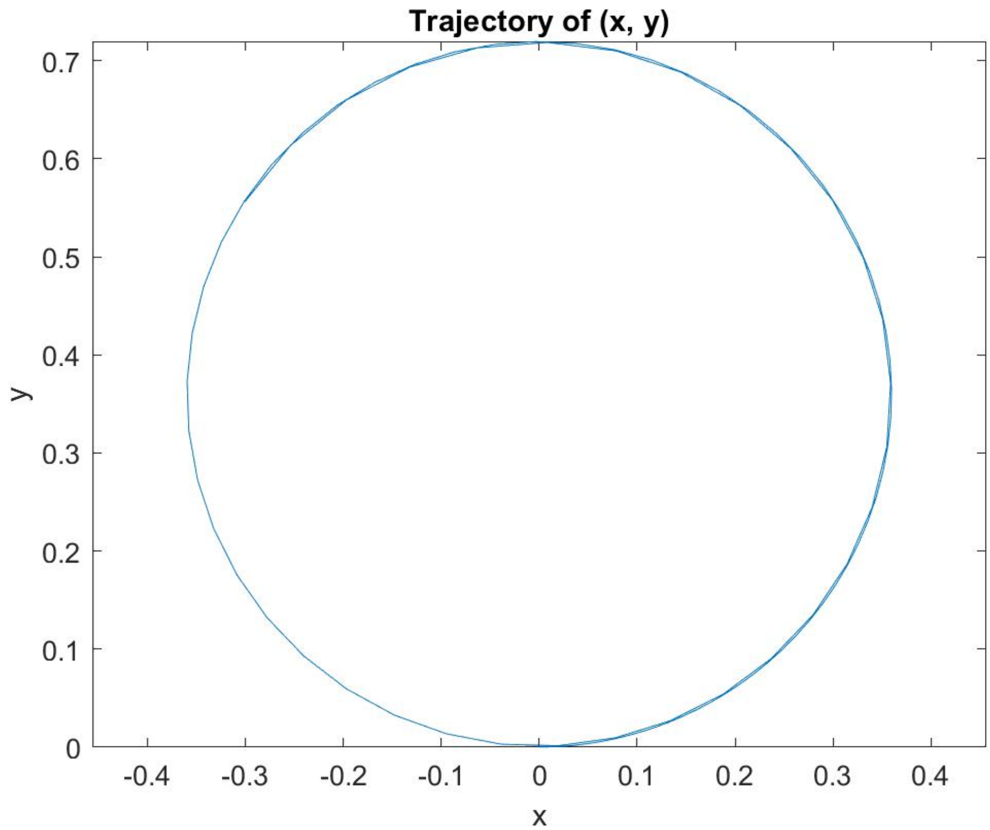

Figure 1.2 Trajectory of the car

b)

First, we need to build two docker images, one for ***opendlv-virtual-motor-kiwi***, run the following command:

cd opendlv-virtual-motor-kiwi

docker build -t registry.opendlv.org/community/ opendlv-virtual-motor-kiwi:1.0 .

the other for ***opendlv-logic-test-kiwi***, run the following command:

cd opendlv-logic-test-kiwi

docker build -t registry.opendlv.org/community/ opendlv-logic-test-kiwi:1.1 .

Then, open two terminals, both have to run the following command:

cd kiwi-simulation-2023

in one terminal, run

docker compose -f interface-kiwi.yml up

open the web in OpenDLV, point your browser to *http://localhost:8081*

in other terminal, run

docker-compose -f simulate-kiwi.yml up

Finally, we will see the simulation result, the coordinate is (0.18, -0.39).

Then, I plot the path of robot movement

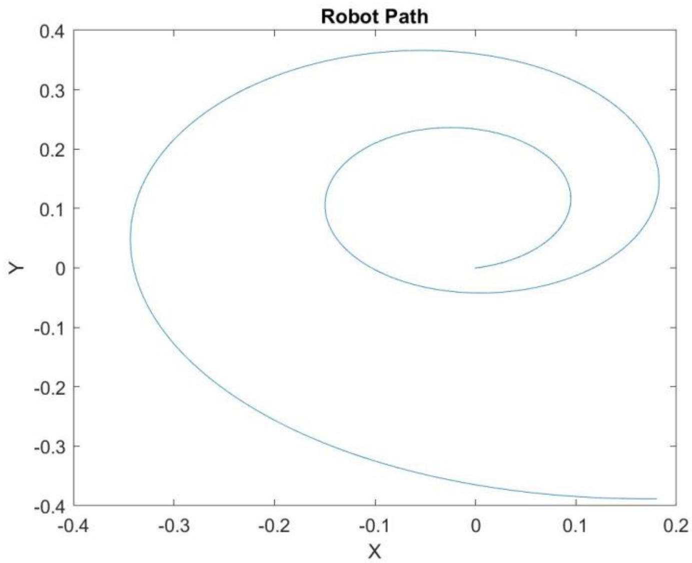

Figure 2.1 Path

action(s)	ID	senderStamp	message name	sample timestamp [μs]	signal(s)
	1002	0	opendlv_sim_KinematicState	1682672229342767	vx: 0 vy: 0 vz: 0 rollRate: 0 pitchRate: 0 yawRate: 0
	1039	0	opendlv_proxy_DistanceReading	1682672229272028	distance: 1.617842
	1094	0	opendlv_proxy_AxleAngularVelocityRequest	1682672229305980	axleAngularVelocity: 0
	1094	1	opendlv_proxy_AxleAngularVelocityRequest	1682672229305980	axleAngularVelocity: 0
	1037	3	opendlv_proxy_VoltageReading	1682672229330853	voltage: 0
	1039	3	opendlv_proxy_DistanceReading	1682672229331068	distance: 0.34
	1039	1	opendlv_proxy_DistanceReading	1682672229324204	distance: 1.982415
	1001	0	opendlv_sim_Frame	1682672229335109	x: 0.1822912 y: -0.389835 z: 0 roll: 0 pitch: 0 yaw: 0.01210688