Scientific Experimentation and Evaluation

Assignment: 04

Anees Khan (9030423)
Debaraj Barua (9030412)
Md Zahiduzzaman (9030432)

17-May-2018

Contents

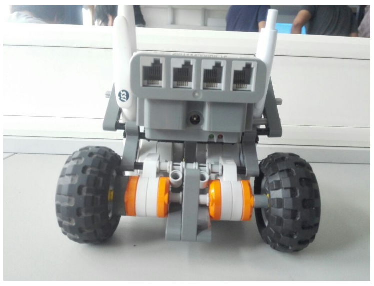
List of Figures

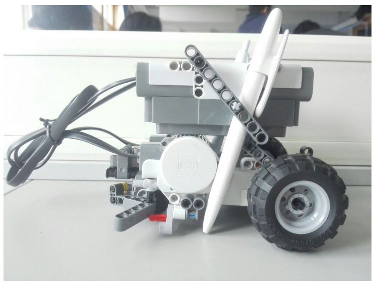
List of Tables

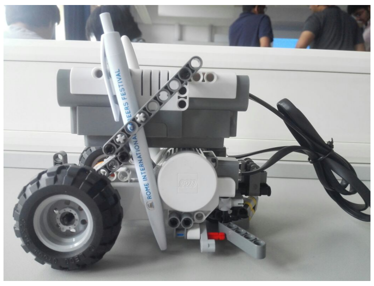
1 Relevant Aspects of Experiment

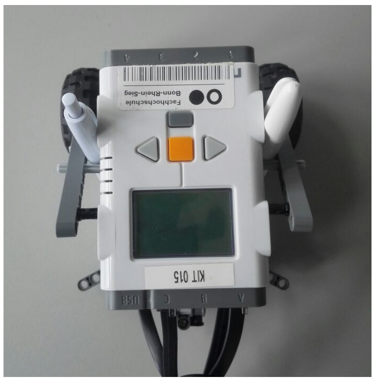
1.1 Design of Robot

- The robot has been designed with three wheels.
- Two of these are driving wheels and are connected to the motors, thus enabling a differential drive systems; and the third is a caster wheel at the back.
- Wheels are oriented as shown in the images below.


(a) Front View


(b) Right View


(c) Left View


(d) Top View

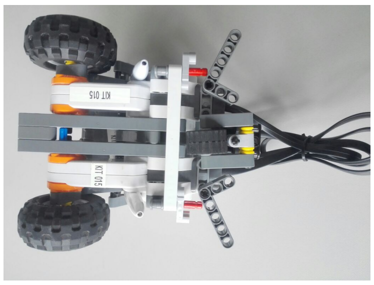

(e) Bottom View

Figure 1: Robot Design

- In addition, it is ensured that the caster wheel cannot rotate completely, and its rotation is limited as shown in the bottom view.

1.2 Formulas used for calculation of ground truth

$$R = \frac{l}{2}\frac{V_l + V_r}{V_r - V_l} \tag{1}$$

$$\omega = \frac{V_r - V_l}{l} \tag{2}$$

$$ICC = [x - Rsin(\theta), y + Rcos(\theta)] \tag{3}$$

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} cos(\omega\delta t) & -sin(\omega\delta t) & 0 \\ sin(\omega\delta t) & -cos(\omega\delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - ICC_x \\ y - ICC_y \\ \theta \end{bmatrix} + \begin{bmatrix} ICC_x \\ ICC_y \\ \omega\delta t \end{bmatrix} \tag{4}$$

Where,

- l is the distance between the centers of the two wheels, (here, 11.1 cm)
- V_r, V_l are the right and left wheel velocities along the ground
- R is the signed distance from the Instantaneous Center of Curvature (ICC) to the midpoint between the wheels
- ω is the rate of rotation about the ICC

1.3 Transformations

- Perpendicular distance from pen axis to wheel axis = 4.6 cm
- Distance between the mid points of the two wheels (track width) = 11.1 cm
- The axis of the wheel is considered to be parallel to the axis formed by joining the two pens.
- To calculate the center of motion for robot, we use the function as shown below.
- The *row* variable has three components, x and y coordinates of the center between two pens and the orientation angle of the robot.

```
def computeCenterOfMotion(data):
        com = np.zeros((data.shape[0],3))
        for i,row in enumerate(data[:]):
                x = row[0]+distance_between_axes*np.cos(np.deg2rad(row[2]))
                y = row[1]+distance_between_axes*np.sin(np.deg2rad(row[2]))
                com[i,:]=np.array([x,y,row[2]])
        return com
```

1.4 Measurement of Start and Stop Positions

1.4.1 Measurement Instruments

- Distance:
  - We use a scale ruler to measure the distances.
  - In addition, we also utilize the grids on the paper which acts as our workspace and world coordinate systems.
  - Instrument Details:
    - 1 square of the grid paper equals 2.5×2.5 cm
    - The precision of the scale used is 1 mm.
- Angle:
  - A protractor is used as a measuring instrument for the angles.
  - Instrument Details:
    - Precision of the protractor used is $1°$.

1.4.2 Measurement Procedure

- Two pens will be fixed near the two driving wheels.
- The starting position is where the two pens meets the x -axis, with the left pen lying on the origin (right pen at a distance of 8.3 cm from the origin).

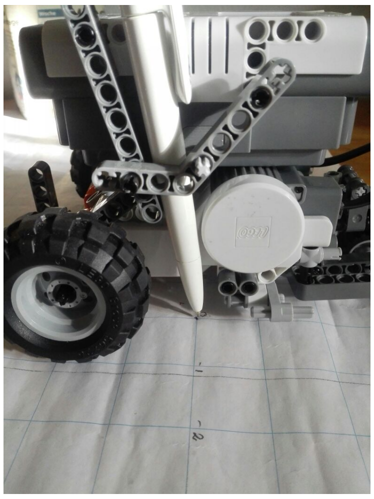


Figure 2: Staring Position

- To measure the orientation of robot, the two points from these pens will be joined by drawing a line, which will be used to mark the pose of robot, with respect to the coordinate system defined.
- For each run, at the initial condition, it is ensured that orientation of rear caster wheel axle is parallel to the driving wheel axis.

- Measuring Position:
  - To get the position of the robot, we measure the distance of the two points from X and Y axis using a scale ruler.
  - These two readings are put in the transformation equation described above to get the actual position of the robot.
- Measuring Angles:
  - The angular pose of the robot is the angle the X-axis makes with the normal to the line joining the two points.
  - This angle is equal to the angle Y-axis makes with the line joining the two points. This can be verified using basic geometry, as shown in the image below.

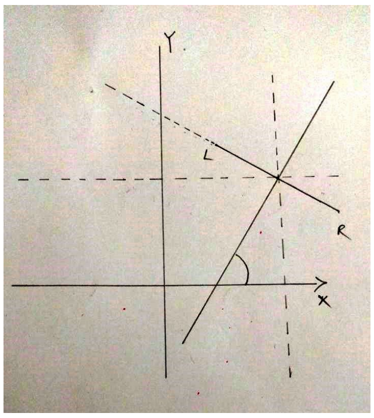


Figure 3: Measurement of angles

  - We now measure the angle using a protractor.

1.5 Parameters used to drive the robot

- Constant angular and translational speed for a fixed time period to describe an arc to left.
- Constant translational speed and no angular speed for a fixed time period to describe a straight line.
- Constant angular and translational speed for a fixed time period to describe an arc to right.
- Speed = 18 cm/s
- Duration = 2.25 seconds
- Turn Rate = 40
- Diameter of wheels: 5.7 cm

1.6 Program used to drive the robot

- Using the JAVA code shared in LEA, we created the scenarios for three run sequences.
- Straight Line:
  - Speed = 18 cm/s
  - Duration: 2.25 Seconds
  - Turn Rate = 0
- Left Arc:
  - Speed = 18 cm/s
  - Duration: 2.25 Seconds
  - Turn Rate = 40
- Right Arc:
  - Speed = 18 cm/s
  - Duration: 2.25 Seconds
  - Turn Rate = -40

1.7 Expected Problems and Performance

- Axis connecting the two pens might not be parallel to the wheel axle.
- Start position of each run may not be exactly similar owing to inaccurate positioning of the robot, this will result in lower precision.
- Pens may slip of move during the run, as such may not result in accurate positions, thus affecting the precision of our readings.
- The constant angular and translational speeds that we assume, may be inaccurate. The actual speed may differ and thus our estimate from the time will be inaccurate.
- The initial acceleration and final deceleration of the robot has not been considered in the experiments, resulting in low accuracy.
- In addition of the two previous points, slippage in the wheels and motors will also affect the accuracy of readings.
- The caster wheel will also result in the bot to drift and also decrease the distance it reaches.
- In addition, the calculation of expected position may result in an overestimate because actual power output may depend on the charge in batteries and the efficiency of motor.

2 Observations and Data

2.1 Readings

	X_L(cm)	Y_L(cm)	X_R(cm)	Y_R(cm)	Angle(degrees)
0	0	0	8.3	0	90

Table 1: Intial Position

	X_L (cm)	Y_L (cm)	X_R (cm)	Y_R (cm)	Theta (degrees)
0	1.2	39.7	9.3	39.6	89
1	3.0	39.9	11.2	39.5	87
2	2.6	39.9	10.8	39.5	88
3	2.2	39.8	10.4	39.5	88
4	1.4	39.8	9.6	39.7	89
5	0.5	39.9	8.1	39.8	90
6	-0.5	39.9	7.6	39.9	90
7	2.5	40.0	10.6	39.6	87
8	1.2	40.0	9.3	39.9	89
9	1.2	40.0	9.2	40.0	88
10	1.5	39.8	9.7	39.5	88
11	2.0	39.8	10.1	39.4	88
12	1.4	39.8	9.5	39.8	89
13	-1.1	39.7	7.1	39.9	91
14	0.0	40.1	8.2	40.1	90
15	0.7	39.8	8.8	39.7	89
16	0.8	40.0	8.9	39.8	89
17	1.4	39.8	9.5	39.7	89
18	1.7	40.0	9.8	39.7	88
19	2.0	40.0	10.1	39.7	88
20	0.5	39.3	8.5	39.6	92
21	1.3	40.0	9.3	39.7	88
22	2.2	40.3	10.4	39.7	87

Table 2: Straight Line Position

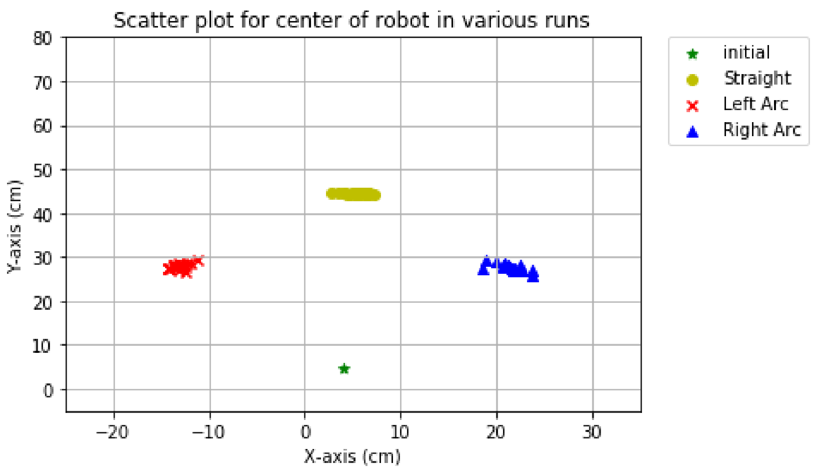
	X_L (cm)	Y_L (cm)	X_R (cm)	Y_R (cm)	Theta (degrees)
0	-9.5	22.5	-9.1	30.8	165
1	-9.1	23.0	-8.5	31.1	162
2	-9.4	22.9	-8.9	31.1	165
3	-7.9	23.0	-7.1	31.4	161
4	-9.3	22.7	-8.8	30.8	164
5	-9.8	22.4	-9.5	30.5	168
6	-10.0	22.2	-9.6	30.3	164
7	-7.3	24.0	-6.3	31.9	162
8	-8.4	23.2	-8.4	31.1	165
9	-9.0	23.1	-8.1	31.2	166
10	-9.1	22.8	-8.5	30.9	167
11	-8.3	23.1	-7.4	31.3	165
12	-9.9	22.2	-9.4	30.4	168
13	-9.3	22.7	-8.9	30.8	167
14	-9.6	22.1	-9.4	30.4	168
15	-8.5	23.1	-7.7	31.2	165
16	-9.2	22.5	-8.5	30.8	166
17	-9.3	22.0	-8.1	30.3	169
18	-9.1	22.8	-8.1	30.9	165
19	-9.3	21.6	-6.5	30.0	169
20	-9.4	22.5	-8.2	30.7	167
21	-7.8	22.8	-8.0	30.9	167
22	-10.0	22.8	-8.7	30.9	163

Table 3: Left Arc Position

	X_L (cm)	Y_L (cm)	X_R (cm)	Y_R (cm)	Theta (degrees)
0	18.8	30.0	16.3	21.8	16
1	17.5	31.0	15.3	22.9	17
2	17.8	30.4	15.7	22.3	17
3	18.3	30.3	16.1	22.0	17
4	16.6	31.5	14.6	23.5	18
5	15.4	31.3	13.8	23.5	22
6	17.7	30.8	15.5	22.8	23
7	18.2	30.5	16.0	22.3	17
8	19.3	29.9	16.9	21.6	16
9	18.0	30.8	15.8	22.6	17
10	18.6	30.1	16.3	21.9	16
11	20.6	28.8	18.0	20.9	13
12	17.9	30.5	15.7	22.4	17
13	19.3	30.0	16.9	21.7	15
14	17.5	30.4	15.3	22.3	17
15	18.5	29.9	16.2	21.7	16
16	19.5	29.9	17.0	21.5	15
17	19.6	30.0	17.0	21.7	15
18	18.0	30.6	15.8	22.5	16
19	19.5	29.9	17.0	21.5	14
20	17.6	30.5	19.0	22.3	24
21	18.8	30.3	9.6	22.1	15
22	18.7	30.2	19.7	22.1	12

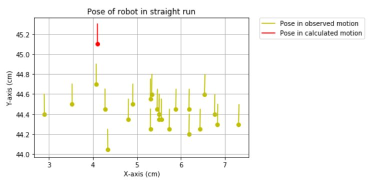
Table 4: Right Arc Position

2.2 Visualization

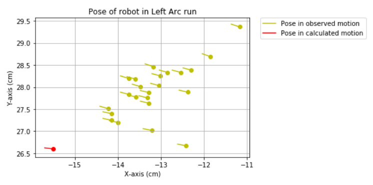


(a) Scatter plot for center of robot in various runs

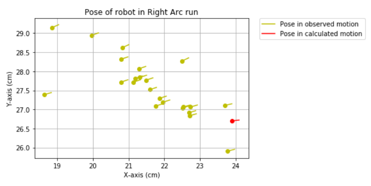
Figure 4: Visualizing Robot Position



(a) Pose of robot in Straight run



(b) Pose of robot in Left Arc run



(c) Pose of robot in Right Arc run

Figure 5: Zoomed View of Robot Positions

All the plots and code is available in the file "SEE_Experiment03_calculations.ipynb"

2.3 Software and Libraries used

- **LibreOffice Calc** for data recording and initial transformation calculations.
- **Python** for data visualization and calculations
- Python libraries:
  - pandas
  - numpy
  - matplotlib
  - seaborn
  - scipy.stats

3 Results

3.1 Final Position & Accuracy

- Straight Run:
  - Mean X value: 5.4 cm
  - Mean Y value: 44.4 cm
  - Mean Angular value: 89 degrees
  - Standard Deviation in X value: 1.0 cm
  - Standard Deviation in Y value: 0.1 cm
  - Standard Deviation in Angular value: 1 degrees
  - Accuracy in X-coordinate: 67.26%

  - Accuracy in Y-coordinate: 98.43%
  - Accuracy in angular: 98.60%

- Left Arc:
  - Mean X value: -13.2 cm
  - Mean Y value: 27.9 cm
  - Mean Angular value: 166 degrees
  - Standard Deviation in X value: 0.8 cm
  - Standard Deviation in Y value: 0.6 cm
  - Standard Deviation in Angular value: 2 degrees
  - Accuracy in X-coordinate: 84.85%
  - Accuracy in Y-coordinate: 95.03%
  - Accuracy in angular: 89.96%
- Right Arc :
  - Mean X value: 21.6 cm
  - Mean Y value: 27.6 cm
  - Mean Angular value: 17 degrees
  - Standard Deviation in X value: 1.3 cm
  - Standard Deviation in Y value: 0.7 cm
  - Standard Deviation in Angular value: 3 degrees
  - Accuracy in X-coordinate: 90.38%
  - Accuracy in Y-coordinate: 96.73%
  - Accuracy in angular: 87.22%

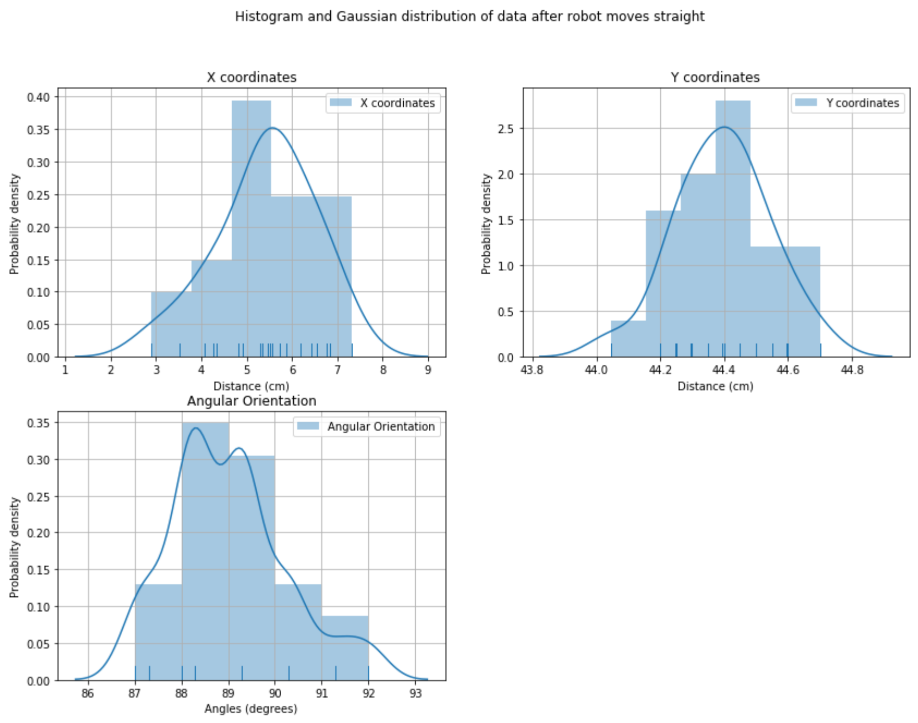
The details of calculations of these values is available in the file "SEE_Experiment03_calculations.ipynb"

3.2 Compare Data with Gaussian

To verify if the observed data follows a gaussian distribution, we use a the *scipy* library. The *scipy.stats.normaltest* function is used to check if the data follows a normal distribution.

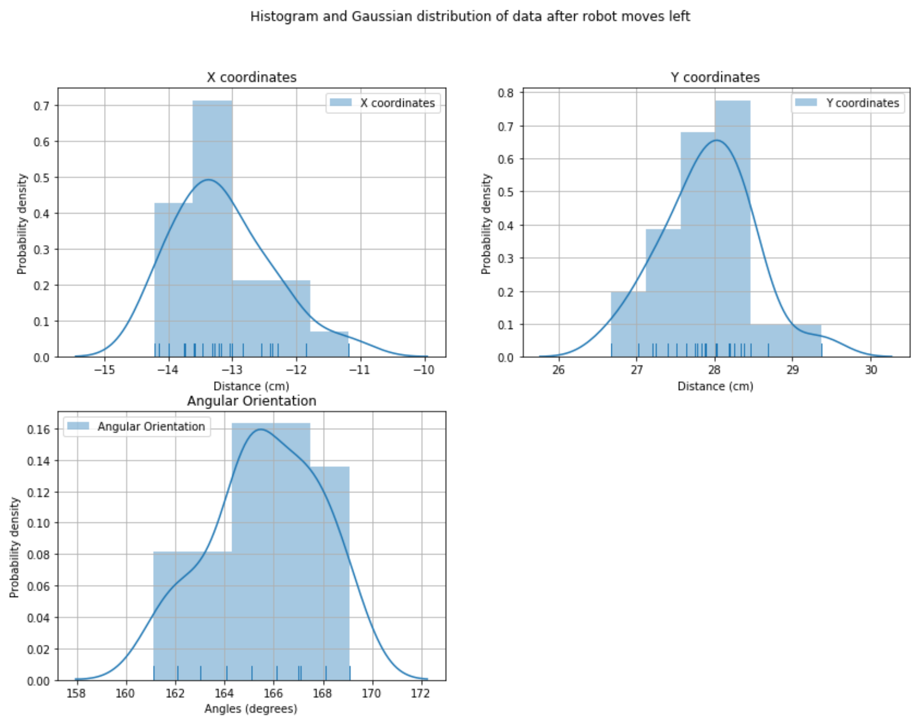
We observe that the threshold value of p value is more than 0.05,indicating that the data is normally distributed.

However, for the right arc motion, we see that the data observed do not follow a Gaussian distribution, this can be attributed to some outlying data points. The reason for these data points has been explored in the expected problems and performance section.



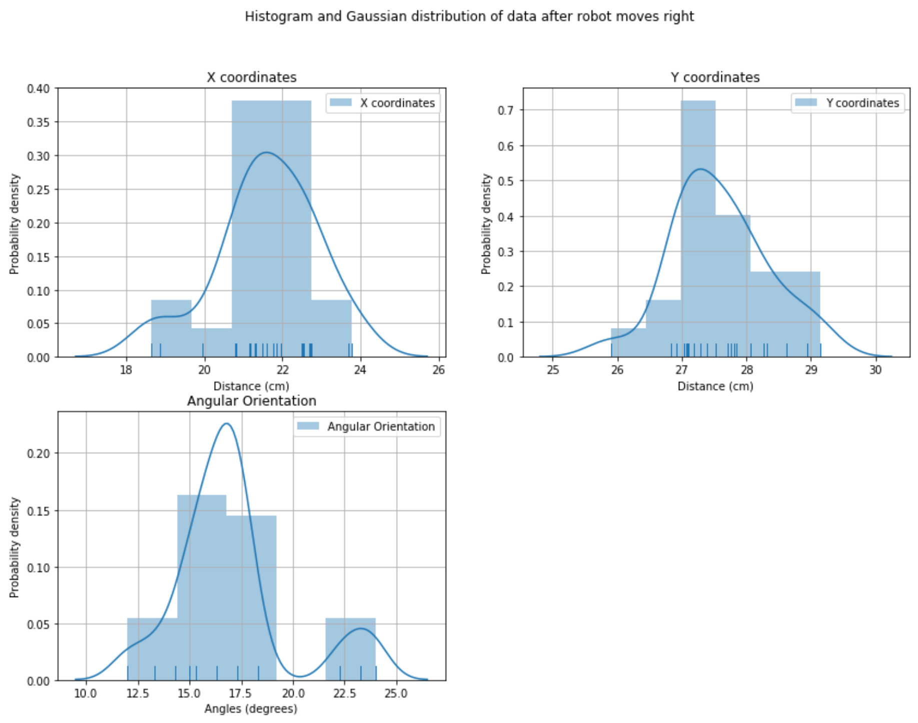
(a) Histogram and Gaussian distribution of data for straight run

Figure 6



(a) Histogram and Gaussian distribution of data for left run

Figure 7



(a) Histogram and Gaussian distribution of data for right run

Figure 8