CSE 276A - Homework 1 - Report

Motor Calibration

Wang, Shou-Yu[A69030868][a], Huang, Yao-Ting[A69030676][a]

[a] *University of California, San Diego.*

October, 2024

Abstract

In this assignment, we calculate the kinematic model ourselves with different approaches, and design a navigation algorithm. At the end, we discuss the reasons cause the imperfect result.

1. Result

Check out the video of our car's navigation at link.

2. Calibration

2.1. Tuning and data collection

First, we found the correct connect motor ports.

Second, we tuned the four-wheels-speeds Ω passed to the motor controller API, and made our car moves straight in both x-axis and y-axis directions. For each four-wheels-speed $\vec{\omega_i}$, we calculated the speed and the angular velocity $\vec{v_i}$ of the car.

$$\vec{\omega_i} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix}, \Omega = \begin{bmatrix} \vec{\omega_1} & \vec{\omega_2} & \dots & \vec{\omega_n} \end{bmatrix}$$

$$\vec{v_i} = \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix}, V = \begin{bmatrix} \vec{v_1} & \vec{v_2} & \dots & \vec{v_n} \end{bmatrix}$$

2.2. Kinematic model

In this assignment, kinematic model gives the motors' angular velocity given the desired velocity. We expanded the kinematic model K to metric format, so their relation became

$$\Omega = K \cdot V \tag{1}$$

If $n = 3$ we can calculate the kinematic model by multiply the Ω with the inverse of V. But, we can collect more and use ***least-squares solution*** to calculate the optimized kinematic model.

$$K = \begin{cases} \Omega \cdot V^{-1}, & \text{if } n = 3 \\ \Omega \cdot V^T \cdot (V \cdot V^T)^{-1}, & \text{if } n > 3 \end{cases} \tag{2}$$

3. Navigation Algorithm

1. Initialize the car's position and direction in world frame.
2. Calculate the difference of position $(\Delta x, \Delta y)$ and direction $\Delta\phi$. Divide those three by a proper transition time and get the velocities.
3. We move the car first and then rotate it to the desired direction.
4. Transform the velocities from world frame to car's frame.
5. Convert the velocities to the motors' speeds by Kinematic model.
6. After sending the motor signal, update the car's position and direction.
7. Repeat for each given waypoint.

4. Discussion

Accuracy: Fairly accurate in direction and reaching waypoints but with small deviations in mixing movement on both x-axis & y-axis, as well as rotating.

Sources of Error: battery level, surface texture, wheel slip, imperfect calibration, and also imperfect kinematic model. For those potential source of error, we divide them into two categories and used several strategies to address.

4.1. Hardware Imperfection

To address these issues, we ensure the battery is charged above 80%, test the car in a consistent location, and regularly clean any hair caught in the wheels. So, the car's movement maintains high consistency with same motor signal.

4.2. Imperfect Calibration

We observed that the car's angular velocity is not directly proportional to the motor speeds. Specifically, doubling the motor speed does not result in the car rotating twice as many degrees within the same time interval.

To address this issue, we

1. choose the most common rotation angle: 90°. We sample go-forward, go-left, and left-turn-90-degree movements with the corresponding motor speeds to build the kinematic model.
2. sample more car's velocities $\vec{v_i}$ and their corresponding motor speeds $\vec{\omega_i}$ and use the ***least-squares solution*** to estimate the optimal kinematic model.

From the result, we observed that the first attempt makes the car's movement highly consistent, the car made fairly accurate 90° turns every time.