Computational Robotics, Fall 2019

Lab 3 - Slide Supplement

Name: Andrew Choi, UID: 205348339

Github Link: **https://github.com/QuantuMope/ucla-computational-robotics**

1 Preliminaries

- Collaborators: Yuki Shirai, Yusuke Tanaka, Viacheslav Inderiakin
- All members made equal contributions of 25%

2 Problem Statement

The goal of this lab was to implement state estimation for a two-wheeled non-holonomic differential drive robot with realistic hardware. To do so, sensor and actuator models were derived for the state estimator. Due to the nonlinear dynamics of the robot, the Extended Kalman Filter (EKF) was implemented.

3 Mathematical Setup

In order to use EKF, a motion model was first derived using the robot's differential drive kinematic equations and are shown below.

$$x_{t+1} = x_t + \frac{r}{2}(u_{l,t} + u_{r,t}) sin\theta_t \Delta t + w_{x,t} \tag{1}$$

$$y_{t+1} = y_t + \frac{r}{2}(u_{l,t} + u_{r,t}) cos\theta_t \Delta t + w_{y,t} \tag{2}$$

$$\theta_{t+1} = \theta_t + \frac{r}{L}(u_r - u_l)\Delta t + w_{\theta,t} \tag{3}$$

where r is the wheel radius, u_l and u_r are the angular wheel velocities, Δt is the time step, and w is zero mean Gaussian noise from motor variance.

In addition to the motion model, a sensor model was also derived using the hardware of the robot. Sensors included two laser range finders, one pointed straight and the other 90° to the right. The robot was also equipped with an inertial measurement unit that outputs the angular pose and velocity estimates. Below the sensor model is shown

$$\begin{bmatrix} \phi_t \\ \psi_t \\ r_t \\ \omega_t \end{bmatrix} = \begin{bmatrix} \sqrt{(x_t - \lambda_{f,x})^2 + (y_t - \lambda_{f,y})^2} \\ \sqrt{(x_t - \lambda_{r,x})^2 + (y_t - \lambda_{r,y})^2} \\ \tan^{-1}\left(\frac{y_t - \lambda_{f,x}}{x_t - \lambda_{f,y}}\right) - \theta_t \\ \frac{R}{L}(u_{r,t} - u_{l,t}) \end{bmatrix} + \begin{bmatrix} v_{\phi,t} \\ v_{\psi,t} \\ v_{r,t} \\ v_{\omega,t} \end{bmatrix} \tag{4}$$

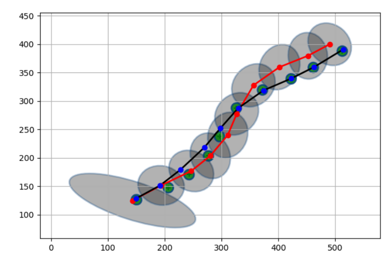
where $(\lambda_{f,x},\, \lambda_{f,y})$ and $(\lambda_{r,x},\, \lambda_{r,y})$ represents the x and y element of a position of a landmark with the front and right laser sensor, respectively.

The first two columns show the distance from the current position of the robot to observed landmarks. The third and fourth columns represent the angular pose and velocity of the robot respectively. All have added Gaussian noise v subject to their respective sensor variances.

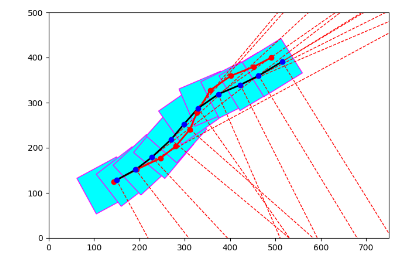
4 Experimental Results

Once the EKF was implemented, state estimation was performed on several trajectories. Two such trials are shown below where the *true trajectory* is shown in blue and the *predicted trajectory* in red. Excellent results can be seen where the robot was able to adapt to high motor noise during a linear trajectory and realign its state to that of the true state in Fig. 1(a).

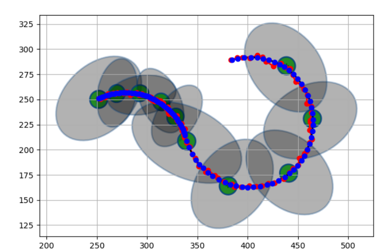
Furthermore, a trade-off between computational efficiency and estimation accuracy was observed as shown in Fig. 1(c) where the scan rate was increased to every tenth of a second versus every second. Although this led to a much lower prediction error, the computational time increased roughly by a magnitude of 5.5 leading to the necessity of proper tuning for optimal performance. Regardless, both results showcase the robustness of the EKF as a state estimator.



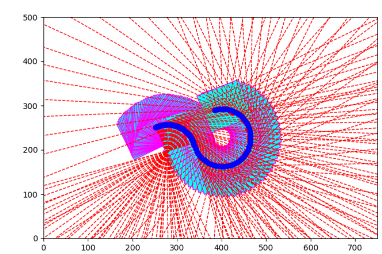
(a) Noisy Linear Trajectory Uncertainty



(b) Noisy Linear Trajectory Robot Pose



(c) High Scan Curve Trajectory Uncertainty



(d) High Scan Curve Trajectory Robot Pose

Figure 1: EKF Results for Several Trajectories Scan Rate

5 Key Contributions

Wrote the experimental results and analysis portion of the report up until section 4.5. Contributed to the code and wrote comments. Derived motion model. Came up with the performance metrics for EKF performance.