Kalman Filter and Extended Kalman Filter

Kalman Filter: The following motion and observation equations are defined for a simplified 2D robot in discrete time.

$$x_k = \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\begin{bmatrix}x_{k-1}\\y_{k-1}\end{bmatrix} + \begin{bmatrix}dt & 0\\0 & dt\end{bmatrix}\begin{bmatrix}r/2(u_r+u_l)\\r/2(u_r+u_l)\end{bmatrix} + \begin{bmatrix}w_x\\w_y\end{bmatrix} \tag{1}$$

$$y_k = \begin{bmatrix}1 & 0\\0 & 2\end{bmatrix} + \begin{bmatrix}r_x & 0\\0 & r_y\end{bmatrix} \tag{2}$$

$r = 0.1m$, is the radius of the wheel, u_r and ul are control signal applied to the right and left wheels. $w_x = N(0, 0.1)$, $w_y = N(0, 0.15)$, $r_x = N(0, 0.05)$, and $r_y = N(0, 0.075)$ are considered. k is the discrete-time index and K its maximum. The variables have the following definitions:

- system state: x_k
- initial state: x_0
- input:v_k
- process noise: w_k
- measurement: y_k
- measurement noise: n_k

The speed of each wheel is fixed and is 10 m/s, and the initial values for x and P are zero.

The Kalman Filter algorithm consists of two steps:

Predictor:

$$\hat{P}_k = A_{k-1}\hat{P}_{k-1}A_{k-1}^T + Q_k \tag{3}$$

$$\hat{x}_k = A_{k-1}\hat{x}_{k-1} + v_k \tag{4}$$

Update:

$$K_k = \hat{P}_k C_k^T (C_k \hat{P}_k C_k^T + R_k)^{-1} \tag{5}$$

$$\hat{P}_k = (1 - K_k C_k)\hat{P}_k \tag{6}$$

$$\hat{x}_k = \hat{x}_{k-1} + K_k(y_k - C_k\hat{x}_k) \tag{7}$$

Simulation for the system such that the robot is driven 1 m to right is done in Python, and the result is according to "Fig 1".

The green line is ground truth. The red circle is robot position, the error ellipse is shown with black, and blue line connects the estimated position of robot at each time step.

Extended Kalman Filter:

In this part the classic motion model and range observation from a landmark located at $L = [10, 10]$ is considered. The motion model is as follows:

$$\begin{bmatrix}x_k\\y_k\\\theta_k\end{bmatrix} = \begin{bmatrix}ru_w cos(\theta_{k-1})\\ru_w sin(\theta_{k-1})\\r/Lu_\psi\end{bmatrix} + \begin{bmatrix}w_w\\w_w\\w_\psi\end{bmatrix} \tag{8}$$

where $w_w = N(0, 0.1)$ and $w_\psi = N(0, 0.01)$. We consider two different kind of measurements: the linear measurement

Fig. 1. Estimated 2D robot moving 1m to the right.

model in the previous part and range/bearing measurements as Eq(9).

$$y_k = \begin{bmatrix}\sqrt{(x_k - x_{landmark})^2 + (y_k - y_{landmark})^2}\\atan2(y_k - y_{landmark}, x_k - x_{landmark}) - \theta_k\end{bmatrix} \tag{9}$$

The Extended Kalman Filter's equation for a nonlinear model is as follows:

Predictor:

$$\hat{P}_k = F_x\hat{P}_{k-1}F_x^T + Q_k' \tag{10}$$

$$\hat{x}_k = f(\hat{x}_{k-1}, v_k) \tag{11}$$

Update:

$$K_k = \hat{P}_k G_k^T (G_k \hat{P}_k G_k^T + R_k')^{-1} \tag{12}$$

$$\hat{P}_k = (1 - K_k G_k)\hat{P}_k \tag{13}$$

$$\hat{x}_k = \hat{x}_{k-1} + K_k(y_k - g(\hat{x}_k)) \tag{14}$$

F_x, F_u, and can be calculated from next equations:

$$F_x = \frac{\partial f}{\partial x_k} = \begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}, F_u = \frac{\partial f}{\partial v_k} = \begin{bmatrix}dtcos(\theta_k) & 0\\dtsin(\theta_k) & 0\\0 & dt\end{bmatrix} \tag{15}$$

$$G = \frac{\partial g}{\partial x_k} = \begin{bmatrix} x_k - x_L/\sqrt{(x_k - x_L)^2 + (y_k - y_L)^2} & y_k - y_L/\sqrt{(x_k - x_L)^2 + (y_k - y_L)^2} & 0 \\ -(y_k - y_L)/\sqrt{(x_k - x_L)^2 + (y_k - y_L)^2} & x_k - x_L/\sqrt{(x_k - x_L)^2 + (y_k - y_L)^2} & -1 \end{bmatrix} \tag{16}$$

Also R' and Q' can be found from Eq (16).

$$R' = F_u R F_u^T, Q' = GQG^T \tag{17}$$

The result of simulating EKF for a classic motion of 2D robot is shown in “Fig 2, 3, and 4”. The measurements in those motions are linear, which means that matrix C is an identity matrix. By considering nonlinear measurements according to Eq (9), we have “Fig 5 and 6. The red point is the landmark at which the measurements occur.

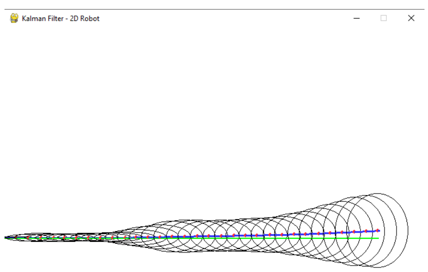

Fig. 2. Estimated 2D robot moving 1 m to the right, considering the linear measurement.

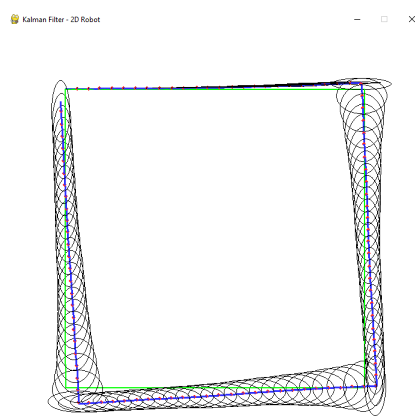

Fig. 3. Estimated 2D robot moving in a square path, considering the linear measurement.

Fig. 4. Estimated 2D robot moving around a landmark, considering the linear measurement.

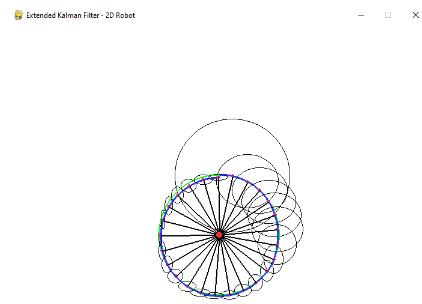

Fig. 5. Estimated 2D robot moving around a landmark, considering the nonlinear measurement from that landmark.

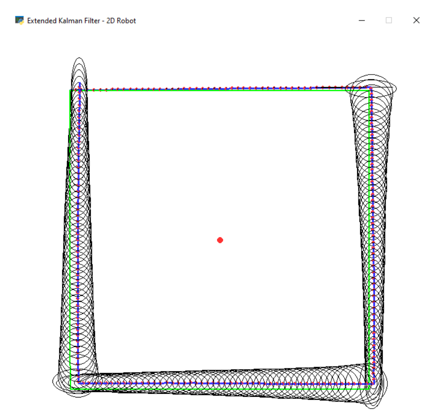

Fig. 6. Estimated 2D robot moving around a landmark in a square path, considering the nonlinear measurement from that landmark.