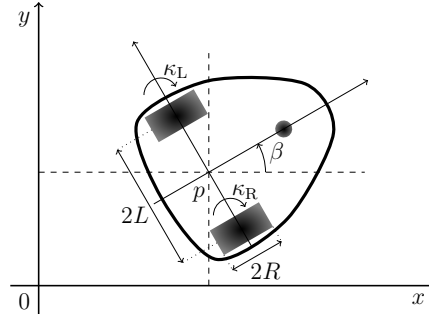


Exercise 11: Extended Kalman Filter
 (to be returned by Feb 12th, 2020, 8:30 in HS 00 036 (Schick-Saal),
 or in building 102, 1st floor, 'Anbau' until 10:00)

Prof. Dr. Moritz Diehl, Tobias Schöls, Naya Baslan, Jakob Harzer, Bryan Ramos

In this exercise, we perform state estimation for a nonlinear system using the *Extended Kalman Filter*.



We model the robot's state $x \in \mathbb{R}^5$ by $x = [p_1, p_2, \beta, v, \omega]^\top$ where $p = (p_1, p_2) \in \mathbb{R}^2$ denotes the position of the robot in m, $\beta \in [0, 2\pi]$ denotes its orientation in rad, $v \in \mathbb{R}$ denotes its forward velocity in $\frac{\text{m}}{\text{s}}$, and $\omega \in \mathbb{R}$ denotes the angular velocity in $\frac{\text{rad}}{\text{s}}$. Note that we omitted the time dependence for cleaner notation, i.e. $x = x(t)$. The time derivatives are given by

$$\dot{p}_1 = v \cos(\beta), \quad \dot{p}_2 = v \sin(\beta), \quad \dot{\beta} = \omega, \quad \dot{v} = c_1 \left(\frac{\tau_R + \tau_L}{R} \right), \quad \dot{\omega} = c_2 \left(\frac{\tau_R - \tau_L}{R} \right)$$

with control inputs $u = (\tau_R, \tau_L)^\top \in \mathbb{R}^2$ and constants c_1 and c_2 given by

$$c_1 = \left(m + \frac{2I_w}{R^2} \right)^{-1}, \quad c_2 = L \cdot \left(I + \frac{2L^2}{R^2} I_w \right)^{-1}$$

where $m = 220$ kg denotes the mass of the robot, $R = 0.16$ m denotes the wheel radius, $2 \cdot L = 0.64$ m is the axis width, $I = 9.6$ kg·m² denotes the moment of inertia around the center of mass, and $I_w = 0.1$ kg·m² denotes the combined moment of inertia of one driving wheel and an actuator's rotating parts about a wheel's axis. We assume that the control inputs u are perfectly known.

1. ON PAPER: Discretize the continuous time model for the robot's dynamics using a one-step Euler integrator with step length $h = 0.05$ s. (1 point)

We assume that the discrete time state dynamics, which we denote by $F(x_k, u_k)$, are perturbed by additive zero-mean Gaussian noise, i.e.

$$x_{k+1} = F(x_k, u_k) + \chi_k$$

where $\chi_k \sim \mathcal{N}(0, \Sigma_\chi)$. We assume

$$\Sigma_{\chi_p} = 10^{-3} \cdot \mathbb{I}_2 \text{ m}^2, \quad \Sigma_{\chi_\beta} = 2 \cdot 10^{-4} \text{ rad}^2, \quad \Sigma_{\chi_v} = 10^{-6} \text{ m}^2/\text{s}^2, \quad \Sigma_{\chi_\omega} = 10^{-6} \text{ rad}^2/\text{s}^2.$$

The function $F(x_k, u_k)$ is nonlinear. Thus, we can not apply the predict function from last week's exercise sheet. In order to predict the next state of the system $x_{[k|k-1]}$ given the current estimate $x_{[k-1|k-1]}$, we evaluate F at $x_{[k-1|k-1]}$ and u_{k-1} :

$$x_{[k|k-1]} = F(x_{[k-1|k-1]}, u_{k-1})$$

We approximate the covariance matrix of $x_{[k|k-1]}$ using a first order Taylor expansion:

$$P_{[k|k-1]} = A_{k-1}P_{[k-1|k-1]}A_{k-1}^\top + \Sigma_\chi \quad \text{where} \quad A_{k-1} = \nabla_x F(x_{[k-1|k-1]}, u_{k-1})$$

2. PAPER: Specify the matrix Σ_χ and compute the Jacobian $\nabla_x F(x, u)$. (1 point)

3. MATLAB: Write a function

`[x_predict, P_predict] = predict(x_estimate, P_estimate, u, F, F_jacobian, W)`
 that computes the prediction $x_{[k|k-1]}$, $P_{[k|k-1]}$. Also implement functions that compute $F(x, u)$ and its Jacobian $\nabla_x F(x, u)$. (1 point)

We cannot observe the robot's state directly. However, GPS data as well as measurements of the rotational speeds of the robot's wheels are available.

The rotational speeds of the left and right wheel, denoted by κ_L and κ_R respectively, are linked to the robot's forward velocity v and angular velocity ω via the following equations:

$$v = \frac{R\kappa_R + R\kappa_L}{2}, \quad \omega = \frac{R\kappa_R - R\kappa_L}{2L}$$

Both GPS measurements and the measurements of the rotational speeds are perturbed by additive zero-mean Gaussian noise with covariance matrices

$$\Sigma_{\gamma_p} = 10 \cdot \mathbb{I}_2 \text{ m}^2 \quad \Sigma_{\gamma_\kappa} = 10^{-6} \cdot \mathbb{I}_2 \text{ rad}^2/\text{s}^2$$

4. ON PAPER: For all $i \in \{1, 2, 3\}$ specify matrices $C^{(i)}$ and covariance matrices $\Sigma_\gamma^{(i)}$ such that the measurements $y_k^{(i)}$ given by

$$y_k^{(i)} = C^{(i)}x_k + \gamma_k^{(i)},$$

with $\gamma_k^{(i)} \sim \mathcal{N}(0, \Sigma_\gamma^{(i)})$ correspond to

$i = 1$: the noisy GPS measurements of the robot's position p .

$i = 2$: the noisy measurements of the angular velocities $[\kappa_R, \kappa_L]^\top$.

$i = 3$: the noisy GPS measurements and the noisy measurements of the angular velocities.

(3 points)

5. MATLAB: Compute the state estimates $x_{[k|k]}$ and state predictions $x_{[k|k-1]}$ where we assume an initial state $x_0 \sim \mathcal{N}(x_{\text{init}}, \Sigma_{\text{init}})$ with

$$x_{\text{init}} = [0, 0, -\pi/2, 0, 0]^\top, \quad \Sigma_{p_{\text{init}}} = \Sigma_{\gamma_p}, \quad \Sigma_{\beta_{\text{init}}} = \Sigma_{v_{\text{init}}} = \Sigma_{\omega_{\text{init}}} = 10^{-1}$$

Hint: The measurement model is linear, thus, you can simply apply the update function from last week's exercise sheet. (1 point)

6. ON PAPER: Compare (and explain) the results you obtained for the different measurement models. (1 point)

7. MATLAB: Let's investigate what happens if the initial guess of the robot's state is wrong. Run your code with the *incorrect* initial state

$$x_{\text{init}} = [-4, 2, -\pi/2, 0, 0]^\top$$

ON PAPER: Compare (and explain) your results.

(1 point)

This sheet gives in total 9 points.