Robotics
Exercise 10

Lecturer: Jim Mainprice

TAs: Philipp Kratzer, Janik Hager, Yoojin Oh

Machine Learning & Robotics lab, U Stuttgart

Universitätsstraße 38, 70569 Stuttgart, Germany

July 9, 2021

1 Kalman Localization (8 Points)

We consider the same car example as for the last exercise, but track the car using a Kalman filter. Access the code as usual:

1. To make sure you have an updated version of the repository, run '*git pull*' and '*git submodule update*'
2. For python run: *'jupyter-notebook course1-Lectures/08-kalman/kalman.ipynb'*
3. For C++ run: *'cd course1-Lectures/08-kalman'*, *'make'*, *'./x.exe'*

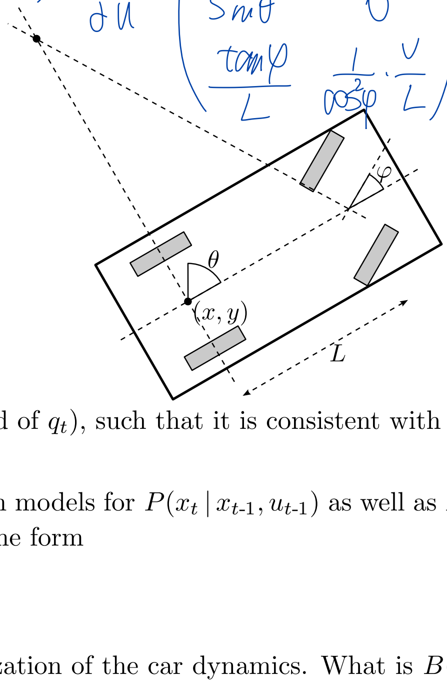
The motion of the car is described by the following:

a) $\frac{\partial f}{\partial u} = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ \frac{\tan\varphi}{L} & \frac{1}{\cos^2\varphi} \cdot \frac{v}{L} \end{pmatrix}$

State $\overset{x_t}{\cancel{q}} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$ **Controls** $u = \begin{pmatrix} v \\ \varphi \end{pmatrix}$

Sytem equation $\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = f(\underset{x_t}{\cancel{q}}, u) = \begin{pmatrix} v \ \cos\theta \\ v \ \sin\theta \\ (v/L) \ \tan\varphi \end{pmatrix}$

$|\varphi| < \Phi$



(In the following, we will denote the state with x_t (instead of q_t), such that it is consistent with the lecture slides.)

a) To apply a Kalman filter (slide 10:26) we need Gaussian models for $P(x_t \,|\, x_{t\text{-}1}, u_{t\text{-}1})$ as well as $P(y_t|x_t)$. We assume that the dynamics model is given as a local Gaussian of the form

$$P(x_{t+1} \,|\, x_t, u_t) = \mathcal{N}(x_{t+1}|x_t + B(x_t)u_t), \sigma_{\text{dynamics}})$$

where the matrix $B(x_t) = \frac{\partial f(x_t, u_t)}{\partial u_t}$ gives the local linearization of the car dynamics. What is $B(x_t)$ (the Jacobian of the state change w.r.t. u) for the car dynamics? (2P)

b) Implement the linearized dynamics model in the function 'getControlJacobian()'. (2P)

c) Concerning the observation likelihood $P(y_t|x_t)$ we assume

$l^c = R\, l^w + t$

$$P(y_t|x_t, \theta_{1:N}) = \mathcal{N}(y_t \,|\, C(x_t)x_t + c(x_t), \sigma_{\text{observation}})$$

$l^w = R^{-1}(l^c - t) = R^{-1} \cdot l^c - R^{-1} t$

What is the matrix $C(x_t)$ (the Jacobian of the landmark positions w.r.t. the car state) in our example? (2P)

$T_{W \to C} = \left(\begin{array}{c|c} R & t \\ \hline 0 & 1 \end{array}\right) = \begin{pmatrix} \cos\theta & -\sin\theta & x \\ \sin\theta & \cos\theta & y \\ 0 & 0 & 1 \end{pmatrix}$

$T_{C\to W} = \begin{pmatrix} R^{-1} & -R^{-1}t \\ 0 & 1 \end{pmatrix}$ $y = l^C = T_{C\to W}\cdot l^W$

$\therefore \frac{\partial g(x_t)}{\partial x_t} = \frac{\partial \cdot T_{C\to W}\cdot l^W}{\partial x_t}$

Hints:
- Assume there is only one landmark in the world.
- The car observes this landmark in its own coordinate frame, $y = l^C \in \mathbb{R}^2$.
- Write down the transformations between world and car coordinates $T_{W\to C}$ and the inverse $T_{C\to W}$.
- Use them to define a mapping $y = g(x_t)$ that maps the state x_t to the observation y.
- Compute the local linearization $C = \frac{\partial g(x_t)}{\partial x_t}$.

d) Implement the Kalman filter (slide 10:26) to track the car (this does not require a solution to question part c). Note that $c(s) = \hat{y}_t - C(s)s$, where $\hat{y}_t$ is the mean observation in the estimated state s. The variables $C, A, Q, W, \hat{y}_t$ of the Kalman filter are already provided in the code. (2P) please see the code implementation.

2 Bayes Smoothing (4 points)

In the lecture we derived the Bayesian filter: given information on the past (observations $y_{0:t}$ and controls $u_{0:t\text{-}1}$) it estimates the current state x_t. However, we can use the available information on $y_{0:T}$ and $u_{0:T}$ also to get a Bayes-optimal estimate of a past state x_t at a previous time $t < T$. This estimate should be "better" than the forward filtered $P(x_t \,|\, y_{0:t}, u_{0:t\text{-}1})$ because it uses the additional information on $y_{t+1:T}$ and $u_{t:T}$. This is called Bayes smoothing (slides 10:30 – 10:31).

Derive the backward recursion $\beta_t(x_t) := P(y_{t+1:T}|x_t, u_{t:T})$ (the likelihood of all future observations given x_t and knowledge of all subsequent controls) of Bayes smoothing on slide 10:31. Explain in each step which rule/transformation you applied.

$R^{-1} = R' = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$, because orthogonal

$$g(x_t) = \begin{pmatrix} R^{-1} & -R^{-1}t \\ 0 & 1 \end{pmatrix}\cdot\begin{pmatrix} z \\ 1 \end{pmatrix} = \begin{pmatrix} R^{-1}z - R^{-1}t \\ 1 \end{pmatrix} = \begin{pmatrix} R^{-1}(z-t) \\ 1 \end{pmatrix} = \begin{pmatrix} (z_1-x)\cos\theta + (z_2-y)\sin\theta \\ -(z_1-x)\sin\theta + (z_2-y)\cos\theta \\ 1 \end{pmatrix}$$

$$\therefore \frac{\partial g(x_t)}{\partial x_t} = \frac{\partial g(x_t)}{\partial(x,y,\theta)} = \begin{pmatrix} -\cos\theta & -\sin\theta & -(z_1-x)\cdot\sin\theta + (z_2-y)\cos\theta \\ \sin\theta & -\cos\theta & -(z_1-x)\cos\theta - (z_2-y)\sin\theta \\ 0 & 0 & 0 \end{pmatrix}$$

2. Derivation of Bayes Smoothing.

past observation $P := y_{0:t}$ Observation model $P(y_t|x_t)$

future observation $F := y_{t+1:T}$ for $T > t$. Dynamic model $P(x_{t+1}|x_t, u_t)$ First order Markov.

control

$$P(x_t|P,F,u_{0:T}) = P(F\,|\,x_t, P, u_{0:T})\cdot P(x_t|P, u_{0:T})$$

$$= C\cdot \underbrace{P(F|x_t, \cancel{P}, u_{t:T})}_{\beta_t(x_t)}\cdot \underbrace{P(x_t|P, u_{0:t-1})}_{P_t(x_t)}$$

Markov. $P(F|P, u_{0:T})$ ← unrelevant to x_t, considered as constant. C

$\beta_t(x_t)$ new info from future

$P_t(x_t)$ same as original Bayes Filter form.

prove: $\beta_t(x_t) := P(y_{t+1:T}|x_t, u_{t:T})$

$= \int P(y_{t+2:T}, y_{t+1}, x_{t+1}|x_t, u_{t:T})\, dx_{t+1}$ Marginal probability.

$= \int P(y_{t+2:T}\,|\,y_{t+1}, x_{t+1}, x_t, u_{t:T})\, P(y_{t+1}, x_{t+1}|x_t, u_{t:T})\, dx_{t+1}$ ← $P(AB|C) = P(A|BC)\; P(B|C)$ conditional probability.

observation model

$= \int P(y_{t+2:T}\,|\,x_{t+1}, u_{t+1:T})\; P(y_{t+1}|x_{t+1}, \cancel{x_t}, u_{t:T})\cdot P(x_{t+1}|x_t, u_{t:T})\, dx_{t+1}$

$= \int \beta_{t+1}(x_{t+1})\cdot P(y_{t+1}|x_{t+1})\cdot P(x_{t+1}|x_t, u_t)\, dx_{t+1}$

observation model dynamic model.