SENSOR FUSION

Kalman Filter Applications

Sensor Fusion

Fredrik Gustafsson
fredrik.gustafsson@liu.se

Gustaf Hendeby
gustaf.hendeby@liu.se

Linköping University

Purpose

Illustrate the Kalman filter with some practical automotive applications.

- Yaw estimation based on basic odometric model.
- Leaning angle estimation for motorcycles.
- Speed hump characterization.

Virtual Yaw Rate Sensor

- Yaw rate gyroscope measures yaw rate $\dot{\psi}$ directly, subject to bias b_k.

$$y_k^1 = \dot{\psi}_k + b_k + e_k^1$$

- Wheel speeds provide two virtual measurements of speed and yaw rate, respectively

$$y_k^2 = \frac{\omega_3 r_{\text{nom}} + \omega_4 r_{\text{nom}}}{2} \frac{2}{B} \frac{\frac{\omega_3}{\omega_4} \frac{r_{k,3}}{r_{k,4}} - 1}{\frac{\omega_3}{\omega_4} \frac{r_{k,3}}{r_{k,4}} + 1} + e_k^2,$$

$$u_k = \frac{\omega_3 r_{\text{nom}} + \omega_4 r_{\text{nom}}}{2} + w_k.$$

We denote speed as an input for reasons explained later.

From the sensor relations, the state vector needs to include $x_k = \left(\psi_k, \dot{\psi}_k, b_k, \frac{r_{k,3}}{r_{k,4}}\right)$.

Odometric Model

The simplest form of odometric model is

$$
\begin{aligned}
\psi_{k+1} &= \psi_k + T_s \dot{\psi}_k, \\
X_{k+1} &= X_k + T_s v_k \cos(\psi_k), \\
Y_{k+1} &= Y_k + T_s v_k \sin(\psi_k),
\end{aligned}
$$

- The state needs to include position X_k, Y_k.
- We can consider the virtual sensor of speed v_k to be an input or as a measurement if v_k is included in the state vector. This is a bit ambiguous. Generally, a smaller state vector is to prefer, so speed is chosen to be an input.
- However, since we have two measurements of yaw rate, these have to be measurements. We cannot have two inputs of the same thing in the KF.

State Space Model

With the state vector $x = (X, Y, \psi, \dot{\psi}, b, \delta)^T$, where $\delta = r_3/r_4$, we get the state space model

$$
\begin{aligned}
X_{k+1} &= X_k + T_s u_k \cos(\psi_k),\\
Y_{k+1} &= Y_k + T_s u_k \sin(\psi_k),\\
\psi_{k+1} &= \psi_k + T_s \dot{\psi}_k + w_k^{\psi},\\
b_{k+1} &= b_k + w_k^b,\\
\delta_{k+1} &= \delta_k + w_k^{\delta}.
\end{aligned}
$$

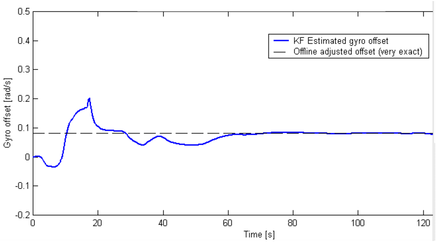
$$
\begin{aligned}
y_k^1 &= \dot{\psi}_k + b_k + e_k^1,\\
y_k^2 &= 0 = \dot{\psi}_k - \frac{\omega_3 r_{\text{nom}} + \omega_4 r_{\text{nom}}}{2} \frac{2}{B} \frac{\frac{\omega_3}{\omega_4}\delta_k - 1}{\frac{\omega_3}{\omega_4}\delta_k + 1} + e_k^2,\\
u_k &= \frac{\omega_3 r_{\text{nom}} + \omega_4 r_{\text{nom}}}{2} + w_k^u.
\end{aligned}
$$

- Note the importance of adding process noise to allow the state to move in the time update.
- The measurement noise of the speed becomes process noise, since it is considered as an input.
- Note how the virtual measurement of yaw rate is included in the model!

Observability

It is always important to check the model before the Kalman filter is applied. Is it invertible?

- Linearize around a working point and compute the observability matrix.
- In this case, one can motivate that the two bias states b_k, δ_k are observable *if and only if* the car changes yaw rate.
- Why? b_k is constant, but the influence of δ is multiplied with ω_3/ω_4, so this ratio needs to change to separate them from eachother.
- Formal analysis gives the same result.
- And it works in practice!



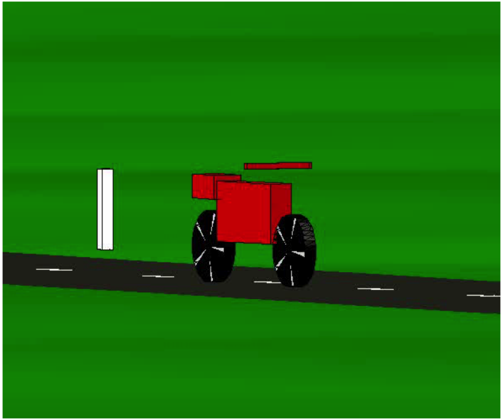
Result

- Test in Valla-rondellen
- Red curve: odometric model just supported with speed and gyroscope, not the virtual yaw rate gyro.
- Blue curve: odometric model supported with speed, gyroscope and virtual yaw rate gyro.
- Gyro bias $b_k \approx 0.8$ rad/s, ≈ 5 deg/s. After 100 seconds, the integrated error in yaw angle will be 500 degrees.

http://youtu.be/d9rzCCIBS9I

MC Leaning Angle

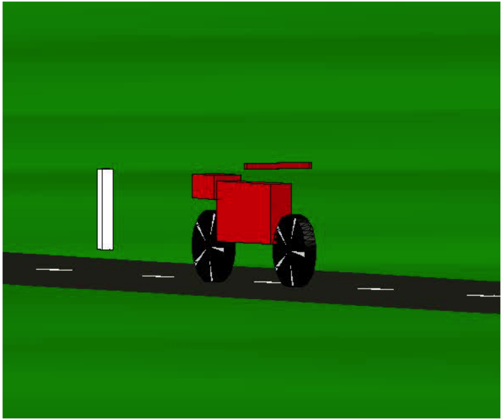
- Headlight steering, ABS and anti-spin systems require leaning angle.
- Gyro very expensive for this application.
- Combination of accelerometers investigated, lateral and downward acc worked fine in EKF.



http://youtu.be/hT6S1FgHx0c0

MC Leaning Angle

- Headlight steering, ABS and anti-spin systems require leaning angle.
- Gyro very expensive for this application.
- Combination of accelerometers investigated, lateral and downward acc worked fine in EKF.



http://youtu.be/hT6S1FgHx0c0

MC Model

- *State vector:*

$$x = \begin{pmatrix} \varphi & \dot{\varphi} & \ddot{\varphi} & \dot{\psi} & \ddot{\psi} & \delta_{ay} & \delta_{az} & \delta_{\dot{\varphi}} \end{pmatrix}^T.$$

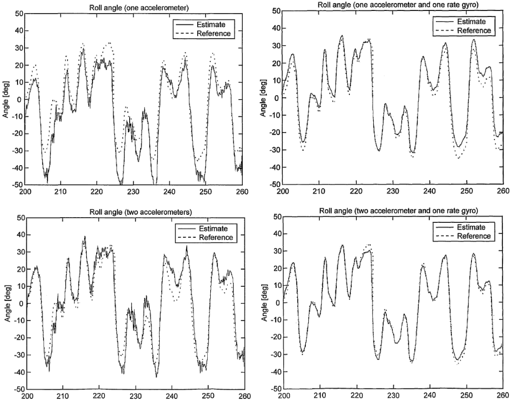
- *State dynamics:* triple integrator for φ, double integrator for ψ and sensor bias states.
- *Sensor model:* Mechanical relations give the three sensor models

$$y = h(x) = \begin{pmatrix} a_y \\ a_z \\ \dot{\varphi} \end{pmatrix} = \begin{pmatrix} ux_4 - z_y x_3 + z_y x_4^2 \tan(x_1) + g \sin(x_1) + x_6 \\ -ux_4 \tan(x_1) - z_z \left(x_2^2 + x_4^2 \tan^2(x_1)\right) + g \cos(x_1) + x_7 \\ -a_1 x_3 + a_2 x_4^2 \tan(x_1) - ux_4 J + x_6 \end{pmatrix}$$

- *Input:* Speed $u = v_x$.
- *Parameters:* z_y, z_z, a_1, a_2, J are constants relating to geometry and inertias of the motorcycle.

MC Results

- Tests with reference sensor based on an extra wheel on the rear axle that provides the leaning angle with high accuracy.
- Note the importance of having a *ground truth* to compare to!
- Off-line evaluation of different subsets of the sensors: all, all but gyro, only a_y, only a_z.
- Result: a_y and a_z is a cost-efficient combination that meets the performance requirement from the applications.

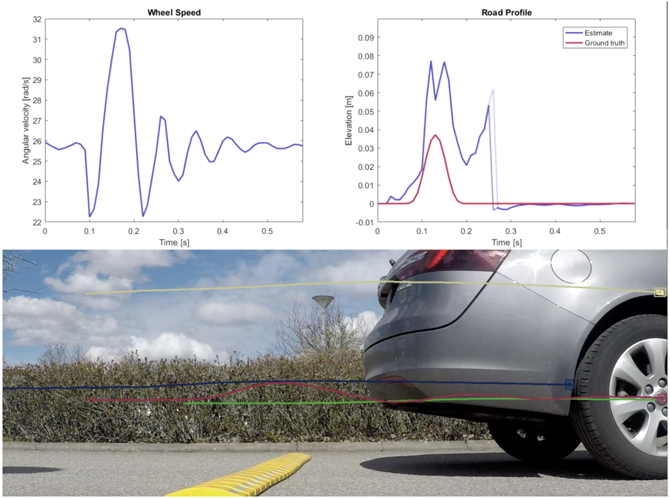


Speed bump estimation

- Goal: estimate the profile of a speed bump only based on the wheel speed measurements.
- A dynamical model is used for how the wheel speed depends on the road profile.
- A Kalman smoother eliminates an unavoidable vertical drift by using the assumption that the absolute height before and after the bump is equal.
- The ground truth of the vertical position of the car is computed based on the markers on the car and the video stream, while the road profile is measured with a ruler.
- Result of a MSc thesis by Lage Ragnarsson 2017 at NIRA Dynamics

Speed bump estimation

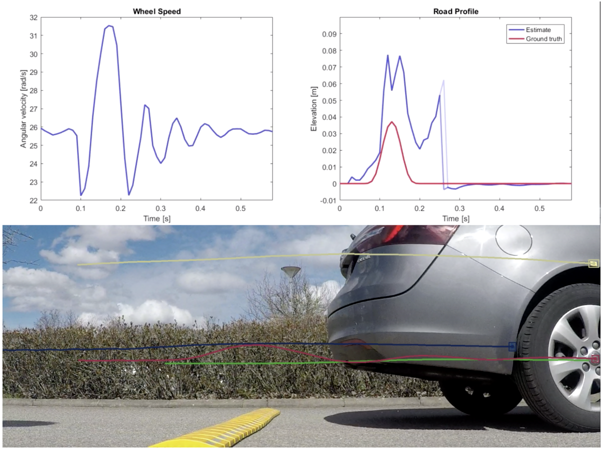
- Red, blue and yellow line in video follows markers on the car using computer vision detectors.
- Upper left plot: blue line shows variation in wheel rotational speed.
- Upper right plot: red is ground truth, blue is the Kalman filter and subsequently smoother.



https://youtu.be/1t6YKUffZiw

Speed bump estimation

- Red, blue and yellow line in video follows markers on the car using computer vision detectors.
- Upper left plot: blue line shows variation in wheel rotational speed.
- Upper right plot: red is ground truth, blue is the Kalman filter and subsequently smoother.

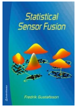


https://youtu.be/1t6YKUffZiw

Summary

Automotive applications illustrating some important concepts:

- *Ground truth:* The importance of having something to evaluate against.
- *Observability:* is the problem realistic with the given measurements?
- *Virtual measurements:* can be both constraints and indirect measurements.
- *Input or output ambiguity:* In some applications, a measured quantity can be used both as an input u or as an output y in the KF.
- *Smoothing:* Do you get important information in the end? Then, a backward sweep can improve the estimate drastically.
- *Cost effective solutions:* Expansive sensors can be replaced with cheap sensors in a KF framework.



Section 16.2