Jacobs University Bremen

Filtering and fusion

Kalman filtering
Extended Kalman filter (EKF)

Automation
CO23-320203

(Class notes by Prof. Dr. Kaustubh Pathak, Spring 2017)

Contents

Recursive State Estimation Filters
- Bayes Filter
- Kalman Filter
- Extended Kalman Filter

Modeling and estimating reality

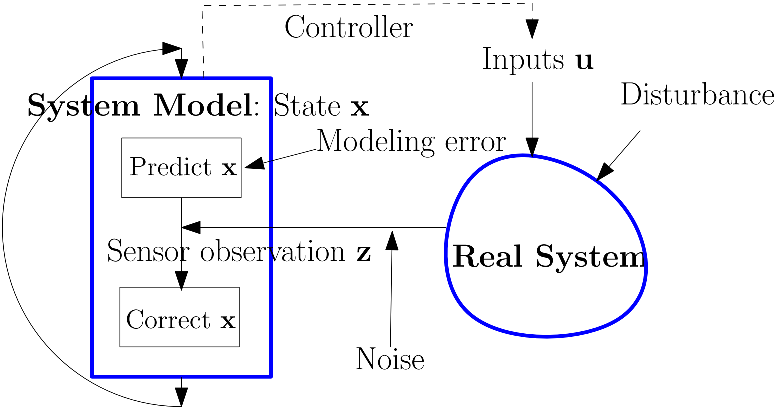

Figure: Recursive stochastic state estimation

Nomenclature

[Thrun et al., 2005], slightly modified.

Definition 14

State $\mathbf{x}_{t_1:t_n} = \{\mathbf{x}_{t_1}, \ldots, \mathbf{x}_{t_n}\}$

Observations/Sensor Measurements $\mathbf{z}_{t_1:t_n}$

Inputs $\mathbf{u}_{t_1:t_n}$

Markov Assumption We only need the current state to predict the next step in future.

$$\rho(\mathbf{x}_t \mid \mathbf{x}_{0:t-1}, \mathbf{z}_{0:t-1}, \mathbf{u}_{0:t-1}) = \rho(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_{t-1}) \tag{2.1}$$

State transition model $\rho(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_{t-1})$.

Measurement model $\rho(\mathbf{z}_t \mid \mathbf{x}_t)$.

Prior belief $\overline{\text{bel}}(\mathbf{x}_t) = \rho(\mathbf{x}_t \mid \mathbf{z}_{0:t-1}, \mathbf{u}_{0:t-1})$, i.e., after using only the state-transition model but without incorporating current measurement.

Posterior belief $\text{bel}(\mathbf{x}_t) = \rho(\mathbf{x}_t \mid \mathbf{z}_{0:t}, \mathbf{u}_{0:t-1})$, i.e., after incorporating current measurement.

Bayes filter

Most general– no assumption about the kind of pdf

Algorithm 1: Bayes filter

input : $\mathrm{bel}(\mathbf{x}_{t-1}) \equiv \rho(\mathbf{x}_{t-1} \mid \mathbf{z}_{0:t-1}, \mathbf{u}_{0:t-2}),\ \mathbf{u}_{t-1},\ \mathbf{z}_t$

output: $\mathrm{bel}(\mathbf{x}_t) \equiv \rho(\mathbf{x}_t \mid \mathbf{z}_{0:t}, \mathbf{u}_{0:t-1})$

The following steps use Markov's assumption

$$\begin{aligned}\overline{\mathrm{bel}}(\mathbf{x}_t) &\equiv \rho(\mathbf{x}_t \mid \mathbf{z}_{0:t-1}, \mathbf{u}_{0:t-1}) \\ &= \int_{\mathbf{x}_{t-1}} \rho(\mathbf{x}_t, \mathbf{x}_{t-1} \mid \mathbf{z}_{0:t-1}, \mathbf{u}_{0:t-1}) d\mathbf{x}_{t-1} \\ &= \int_{\mathbf{x}_{t-1}} \rho(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_{t-1}) \mathrm{bel}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1} \end{aligned} \tag{2.2}$$

$$\mathrm{bel}(\mathbf{x}_t) = \eta \rho(\mathbf{z}_t \mid \mathbf{x}_t) \overline{\mathrm{bel}}(\mathbf{x}_t) \tag{2.3}$$

Eq. (2.2) is called the prediction step and Eq. (2.3) is called the correction step or measurement update step.

Kalman filter

A specialization of Bayes filter for linear systems and Gaussian pdfs

References [Thrun et al., 2005, Welch and Bishop, 2007]

Definition 15 (State transition model)

System evolution is linear. $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$.

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1} + \varepsilon \tag{2.4}$$

The random variable $\varepsilon \in \mathbb{R}^n$ is Gaussian noise $\mathcal{N}(\varepsilon\,;\,\mathbf{0}, \mathbf{R})$ and denotes modeling error among other things. The state-transition model is therefore,

$$\begin{aligned}\rho(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_{t-1}) &= \mathcal{N}(\mathbf{x}_t\,;\,\mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1}, \mathbf{R}) \\ &= \frac{\exp\left\{-\frac{1}{2}(\mathbf{x}_t - (\mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1}))^T \mathbf{R}^{-1}(\mathbf{x}_t - (\mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1}))\right\}}{(2\pi)^{n/2} |\det \mathbf{R}|^{1/2}}.\end{aligned} \tag{2.5}$$

Kalman filter

Measurement model The measured sensor output $\mathbf{z} \in \mathbb{R}^k$ is linearly related to the state.

$$\mathbf{z}_t = \mathbf{H}\mathbf{x}_t + \delta, \tag{2.6}$$

where, $\delta \in \mathbb{R}^k$ is the Gaussian sensor noise $\mathcal{N}(\delta\ ;\ \mathbf{0}, \mathbf{Q})$. The measurement model is therefore,

$$\begin{aligned} \rho(\mathbf{z}_t \mid \mathbf{x}_t) &= \mathcal{N}(\mathbf{z}_t\ ;\ \mathbf{H}\mathbf{x}_t, \mathbf{Q}) \\ &= \frac{\exp\left\{-\frac{1}{2}(\mathbf{z}_t-(\mathbf{H}\mathbf{x}_t))^T\mathbf{Q}^{-1}(\mathbf{z}_t-(\mathbf{H}\mathbf{x}_t))\right\}}{(2\pi)^{k/2}|\det\mathbf{Q}|^{1/2}}. \end{aligned} \tag{2.7}$$

Initial distribution is also Gaussian.

$$\begin{aligned} \mathrm{bel}(\mathbf{x}_0) = \rho(\mathbf{x}_0) &= \mathcal{N}(\mathbf{x}_0\ ;\ \mu_0, \mathbf{P}_0) \\ &= \frac{\exp\left\{-\frac{1}{2}(\mathbf{x}_0-(\mu_0))^T\mathbf{P}_0^{-1}(\mathbf{x}_0-(\mu_0))\right\}}{(2\pi)^{n/2}|\det\mathbf{P}_0|^{1/2}}. \end{aligned} \tag{2.8}$$

Kalman filter

Algorithm 2: Kalman filter

input : $\text{bel}(\mathbf{x}_{t-1}) \equiv \mathcal{N}\left(\mathbf{x}_{t-1}\,;\ \mu_{t-1}, \mathbf{P}_{t-1}\right),\ \mathbf{u}_{t-1},\ \mathbf{z}_t$

output: $\text{bel}(\mathbf{x}_t) \equiv \mathcal{N}\left(\mathbf{x}_t\,;\ \mu_t, \mathbf{P}_t\right)$

Prediction step;

$$\bar{\mu}_t = \mathbf{A}\mu_{t-1} + \mathbf{B}\mathbf{u}_{t-1} \tag{2.9}$$

$$\bar{\mathbf{P}}_t = \mathbf{A}\mathbf{P}_{t-1}\mathbf{A}^T + \mathbf{R} \tag{2.10}$$

Correction/Measurement-update step;

$$\mathbf{K} = \bar{\mathbf{P}}_t\mathbf{H}^T\left(\mathbf{H}\bar{\mathbf{P}}_t\mathbf{H}^T + \mathbf{Q}\right)^{-1} \tag{2.11a}$$

$$\mu_t = \bar{\mu}_t + \mathbf{K}\left(\mathbf{z}_t - \mathbf{H}\bar{\mu}_t\right) \tag{2.11b}$$

$$\mathbf{P}_t = \left(\mathbf{I}_n - \mathbf{K}\mathbf{H}\right)\bar{\mathbf{P}}_t \tag{2.11c}$$

Notational differences

- $\mathbf{P}_t$ may be written as $\boldsymbol{\Sigma}_t$.
- $\mathbf{P}_t$ is also written $\mathbf{P}_{k+1|k+1}$. Then $\bar{\mathbf{P}}_t$ is written as $\mathbf{P}_{k+1|k}$.
- The definitions of $\mathbf{Q}$ and $\mathbf{R}$ can be reversed! Happens in EKF SLAM paper [Dissanayake et al., 2001].
- For the observation matrix $\mathbf{H}$, another common notation is $\mathbf{C}$.
- The discrete time can be written as $t = k\Delta t$ where Δt is the sampling time. Therefore, $\mathbf{x}_{t-1}$ will also be written as $\mathbf{x}(k-1)$.

Proof

The proof follows by:

- Substituting the Gaussian state transition Eq. (2.5) in Bayes prediction step (2.2),
- and the Gaussian measurement model (2.7) into Bayes correction step (2.3).

Application: fusing two estimates I

Example 16 (Sensor fusion)

Suppose you have two sensors to measure the same state $\mathbf{x} \in \mathbb{R}^n$. They gave you two estimates of the states $\mathcal{N}(\mathbf{x}\ ;\ \mu_1, \mathbf{\Psi}_1)$, and $\mathcal{N}(\mathbf{x}\ ;\ \mu_2, \mathbf{\Psi}_2)$. How would you fuse these results together to produce a better estimate?

- We merely need to apply one correction step of a Kalman filter. For this filter there is no system transition model, i.e., $\mathbf{x}_t = \mathbf{x}_{t-1}$.
- The measurement model is simply $\mathbf{z}_t = \mathbf{x}_t + \delta$, where $\delta \in \mathbb{R}^n$ is a random variable with pdf $\mathcal{N}(\delta\ ;\ \mathbf{0}, \mathbf{\Psi}_2)$. Therefore, referring to (2.7), $\mathbf{H} = \mathbf{I}_n$ and $\mathbf{Q} \equiv \mathbf{\Psi}_2$.
- Referring to Eq. (2.8), we initialize our estimate as $\mathcal{N}(\mathbf{x}\ ;\ \mu_1, \mathbf{\Psi}_1)$.

Application: fusing two estimates II

- No prediction step is required: $\bar{\mathbf{P}} = \mathbf{\Psi}_1, \bar{\mu} = \mu_1$. After one application of the correction step Eq. (2.11) for $\mathbf{z} = \mu_2$, we get

$$\mu = \mu_1 + \mathbf{\Psi}_1 \left(\mathbf{\Psi}_1 + \mathbf{\Psi}_2\right)^{-1} \left(\mu_2 - \mu_1\right) \tag{2.12}$$

$$\mathbf{\Psi} \equiv \mathbf{P} = \mathbf{\Psi}_1 - \mathbf{\Psi}_1 \left(\mathbf{\Psi}_1 + \mathbf{\Psi}_2\right)^{-1} \mathbf{\Psi}_1 \tag{2.13}$$

These represent our fused estimates.

Example 17 (Specialization to a scalar state $x \in \mathbb{R}$)

In this case, we can take $\mathbf{\Psi}_1 \equiv \sigma_1^2$ and $\mathbf{\Psi}_2 \equiv \sigma_2^2$. Show that the above fusion equations reduce to the well known results

$$\frac{1}{\sigma^2} = \frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} \qquad \mu = \frac{\sigma_1^2 \mu_2 + \sigma_2^2 \mu_1}{\sigma_1^2 + \sigma_2^2}. \tag{2.14}$$

Note that $\sigma^2 \leq \sigma_1^2$ and $\sigma^2 \leq \sigma_2^2$, i.e., we have a better estimate!

Application: fusing two estimates

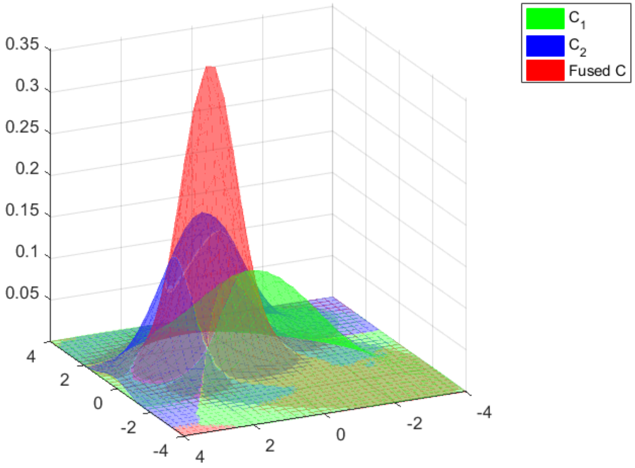

Figure: The original normal pdfs and the fused pdf.

Application: fusing two estimates

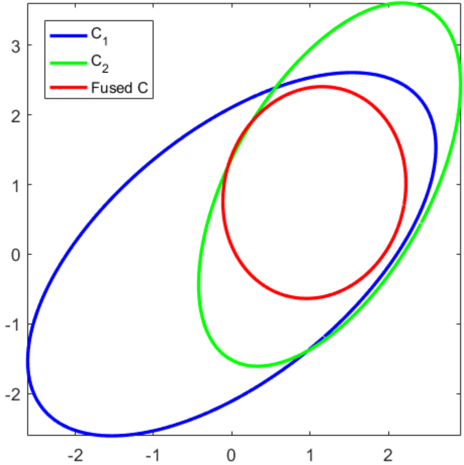

Figure: The original covariances and the fused covariance.

Extended Kalman filter (EKF)

Extension of KF to nonlinear systems by linearization

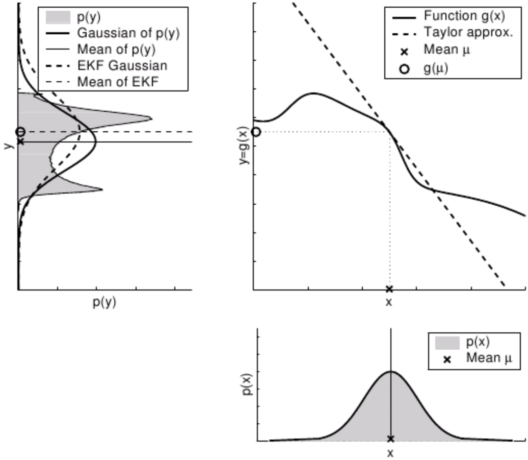

Figure: Effects of nonlinearity. From `probabilistic-robotics.org`

Nonlinear models and their linearization

Most practical problems in robotics are nonlinear.

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}, \varepsilon_{t-1}) \quad \varepsilon \equiv \mathcal{N}(\varepsilon\ ;\ \mathbf{0}, \mathbf{R}) \quad \text{State transition model} \tag{2.15}$$

$$\mathbf{z}_t = \mathbf{h}(\mathbf{x}_t, \delta_t) \quad \delta \equiv \mathcal{N}(\delta\ ;\ \mathbf{0}, \mathbf{Q}) \quad \text{Measurement model} \tag{2.16}$$

On linearization using Taylor's series, one gets

$$\mathbf{x}_t \approx \mathbf{f}(\mu_{t-1}, \mathbf{u}_{t-1}, \mathbf{0}) + \underbrace{\left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mu_{t-1}, \mathbf{u}_{t-1}, \mathbf{0}}}_{\text{Jacobian } \mathbf{F}_{t-1}} (\mathbf{x}_{t-1} - \mu_{t-1}) + \underbrace{\left.\frac{\partial \mathbf{f}}{\partial \varepsilon}\right|_{\mu_{t-1}, \mathbf{u}_{t-1}, \mathbf{0}}}_{\text{Jacobian } \mathbf{E}_{t-1}} \varepsilon_{t-1} \tag{2.17}$$

$$\mathbf{z}_t \approx \mathbf{h}(\bar{\mu}_t, \mathbf{0}) + \underbrace{\left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\bar{\mu}_t, \mathbf{0}}}_{\text{Jacobian } \mathbf{H}_t} (\mathbf{x}_t - \bar{\mu}_t) + \underbrace{\left.\frac{\partial \mathbf{h}}{\partial \delta}\right|_{\bar{\mu}_t, \mathbf{0}}}_{\text{Jacobian } \mathbf{D}_t} \delta_t \tag{2.18}$$

Remark 18

- *The state transition Jacobians* $\mathbf{F}_{t-1}$ *and* $\mathbf{E}_{t-1}$*, and also the Observation Jacobians* $\mathbf{H}_t$*, and* $\mathbf{D}_t$ *are not constant but are functions of* $(\mu_{t-1}, \mathbf{u}_{t-1})$ *and* $\bar{\mu}_t$ *respectively. Hence, they need to be recalculated in each time-step.*
- *The linearized model and sensor noise random variables are*

$$\varepsilon'_t \equiv \mathcal{N}\left(\varepsilon'_t\ ;\ \mathbf{0}, \mathbf{E}_t \mathbf{R} \mathbf{E}_t^T\right) \tag{2.19}$$

$$\delta'_t \equiv \mathcal{N}\left(\delta'_t\ ;\ \mathbf{0}, \mathbf{D}_t \mathbf{Q} \mathbf{D}_t^T\right) \tag{2.20}$$

- *Refer to the tutorial at [Welch and Bishop, 2007] for more detailed explanation.*

Extended Kalman filter

Algorithm 3: EKF

input : $\text{bel}(\mathbf{x}_{t-1}) \equiv \mathcal{N}\left(\mathbf{x}_{t-1}\,;\,\mu_{t-1}, \mathbf{P}_{t-1}\right),\ \mathbf{u}_{t-1},\ \mathbf{z}_t$

output: $\text{bel}(\mathbf{x}_t) \equiv \mathcal{N}\left(\mathbf{x}_t\,;\,\mu_t, \mathbf{P}_t\right)$

Prediction step;

$$\bar{\mu}_t = \mathbf{f}(\mu_{t-1}, \mathbf{u}_{t-1}, \mathbf{0}) \tag{2.21}$$

$$\bar{\mathbf{P}}_t = \mathbf{F}_{t-1}\mathbf{P}_{t-1}\mathbf{F}_{t-1}^T + \mathbf{E}_{t-1}\mathbf{R}\mathbf{E}_{t-1}^T \tag{2.22}$$

Correction/Measurement-update step;

$$\mathbf{K} = \bar{\mathbf{P}}_t\mathbf{H}_t^T\left(\mathbf{H}_t\bar{\mathbf{P}}_t\mathbf{H}_t^T + \mathbf{D}_t\mathbf{Q}\mathbf{D}_t^T\right)^{-1} \tag{2.23}$$

$$\mu_t = \bar{\mu}_t + \mathbf{K}\left(\mathbf{z}_t - \mathbf{h}(\bar{\mu}_t, \mathbf{0})\right) \tag{2.24}$$

$$\mathbf{P}_t = \left(\mathbf{I}_n - \mathbf{K}\mathbf{H}_t\right)\bar{\mathbf{P}}_t \tag{2.25}$$

References I

Dissanayake, M. W. M. G., Newman, P., Clark, S., and Durrant-Whyte, H. F. (2001).
A solution to the simultaneous localization and map building (SLAM) problem.
IEEE Trans. on Robotics and Automation, 17(3):229–241.

Jaynes, E. T. (2003).
Probability theory: the logic of science.
Cambridge University Press, `http://bayes.wustl.edu/etj/prob/book.pdf`.

Sivia, D. S. (2002).
Data Analysis: A Bayesian Tutorial.
Oxford Science Publications.

References II

Thrun, S., Burgard, W., and Fox, D. (2005).
Probabilistic robotics.
The MIT Press.

Welch, G. and Bishop, G. (2007).
An introduction to the Kalman filter.
Technical Report TR 95-041, Dept. Computer Science, University of North Carolina at Chapel Hill,
http://www.cs.unc.edu/~welch/kalman/.