Nonlinear observers

design techniques, part 1

Erik Frisk

Dept. of Electrical Engineering, Linköping University

December 7, 2018

LINKÖPING UNIVERSITY

Today's lecture, objectives

1. The objective of the course is not to understand, in detail, a large number of design techniques. Too small course for that.
2. The objective is to give common knowledge and insight into general principles and some design experience in Matlab.

A few observer approaches that are (more or less) common

- Stochastic filters KF/EKF/EKF-iterated/UKF/partikelfilter/...
- Direct Lyapunov design
- Linearized transformations into observable canonical form (or similar)
- Observer design using Lyapunov's auxiliary theorem
- Observers with (sufficiently) high gain (high gain observers)
- Sliding mode observers
- ...

Today: focus on stochastic filters.

Reading material

1. References [3] and [4] survey papers (a bit dated). [17] page 55-82 gives a quick overview of stochastic filters.
2. KF/EKF [17] + references therein
3. Array implementation of Kalman filter [22]
4. Constant gain Extended Kalman Filter [9]
5. Unscented Kalman Filter [14, 23] (with introduction in [17])
6. Particle filter [12,13]
7. Sliding mode observer [10]
8. High gain observer [11] + references therein

Today: Methods under bullets 2-6.

Outline

- *Reading material*
- *Kalman Filter*
  - *Linear Kalman filter*
  - *Extended Kalman Filter (EKF)*
  - *Iterated Extended Kalman Filter*
  - *Constant Gain Extended Kalman Filter*
- *Implementation, array/square-root algorithms*
- *Unscented Kalman Filter*

Outline

- Reading material
- Kalman Filter
  - Linear Kalman filter
  - Extended Kalman Filter (EKF)
  - Iterated Extended Kalman Filter
  - Constant Gain Extended Kalman Filter
- Implementation, array/square-root algorithms
- Unscented Kalman Filter

Kalman filter

For the linear Kalman filter, typically state-space models in the form

$$x_{t+1} = F_t x_t + G_t w_t$$
$$y_t = H_t x_t + e_t$$

Use the model without input u, direct to introduce.

Basic idea

The Kalman filter is a linear estimator that updates expected value and covariance for a minimum-variance estimator.

Optimality

If the noises e_t and w_t are Gaussian, the Kalman filter is the optimal filter (not just in the class of linear filters).

Kalman filter

The filter is often written as:

1. Initialize the filter with apriori information
$$\hat{x}_{0|-1} = x_0, \quad P_{0|-1} = P_0$$
2. Measurement update; take a new measurement y_t and compute
$$\hat{x}_{t|t}, \quad P_{t|t}$$
3. Time update; proceed to next time-step and compute
$$\hat{x}_{t+1|t}, \quad P_{t+1|t}$$

Iterate from step 2.

Kalman filter, the equations

If you look up the equations, the might look like

Measurement update

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(y_t - H_t\hat{x}_{t|t-1})$$
$$K_t = P_{t|t-1}H_t^T(H_tP_{t|t-1}H_t^T + R_t)^{-1}$$
$$P_{t|t} = P_{t|t-1} - K_tH_tP_{t|t-1}$$

Time update

$$\hat{x}_{t+1|t} = F_t\hat{x}_{t|t}$$
$$P_{t+1|t} = F_tP_{t|t}F_t^T + G_tQ_tG_t^T$$

Again, trivial to extend with input signal u.

Now, an illustration where the equations come from and links to non-linear extensions.

Hand wave the time update equations

The dynamic equations is

$$x_{t+1} = F_t x_t + G_t w_t$$

and the objective of the time update is to

$$\hat{x}_{t|t}, P_{t|t} \to \hat{x}_{t+1|t}, P_{t+1|t}$$

where

$$P_{t|t} = \mathrm{cov}(\hat{x}_{t|t} - x_t), \quad Q_t = \mathrm{cov} w_t$$

For the state, since $E(w_t) = 0$,

$$\hat{x}_{t+1|t} = F_t \hat{x}_{t|t} + E(G_t w_t) = F_t \hat{x}_{t|t}$$

and

$$\begin{aligned} P_{t+1|t} = \mathrm{cov}\,(\hat{x}_{t+1|t} - x_{t+1}) = \mathrm{cov}\,(F_t \hat{x}_{t|t} - F_t x_t - G_t w_t) = \\ = \mathrm{cov}\,(F_t(\hat{x}_{t|t} - x_t)) + \mathrm{cov}\,(G_t w_t) = F_t P_{t|t} F_t^T + G_t \mathrm{cov}\; w_t G_t^T = \\ = F_t P_{t|t} F_t^T + G_t Q_t G_t^T \end{aligned}$$

Estimation, some basics

Assume two random variables x and y with expected value 0. A linear estimator that estimates x from y is then

$$\hat{x} = Ky$$

With the notation $P_{xy} = E\{x\,y^T\}$, the covariance for the estimation error is then

$$\begin{aligned} E\{(\hat{x} - x)(\hat{x} - x)^T\} = E\{(Ky - x)(Ky - x)^T\} = \\ = KP_{yy}K^T - KP_{xy}^T - P_{xy}K^T + P_{xx} = \\ = P_{xx} - P_{xy}P_{yy}^{-1}P_{xy}^T + (P_{xy} - KP_{yy})P_{yy}^{-1}(P_{xy} - KP_{yy})^T \end{aligned}$$

The last term is a positive definite matrix, which gives that the optimal estimator K and minimal covariance is given by:

$$K = P_{xy}P_{yy}^{-1}, \quad P_{\hat{x}\hat{x}} = P_{xx} - P_{xy}P_{yy}^{-1}P_{xy}^T$$

Hand wave the measurement update equations

We have the measurement equation

$$y_t = H_t x_t + e_t$$

and

$$\hat{x}_{t|t-1} \sim \mathcal{N}(x_t, P_{t|t-1}), \quad e_t \sim \mathcal{N}(0, R_t)$$

and the objective of the measurement update is to

$$\hat{x}_{t|t-1}, P_{t|t-1} \to \hat{x}_{t|t}, P_{t|t}$$

Introduce $\tilde{x}_t = x_t - \hat{x}_{t|t-1}$ and rewrite

$$z_t = y_t - H_t \hat{x}_{t|t-1} = H \tilde{x}_t + e_t,$$

We now have the exact same situations as before $\tilde{x}_t$ and z_t, with

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(y_t - H_t \hat{x}_{t|t-1})$$

where

$$K_t = P_{\tilde{x}z}P_{zz}^{-1}$$

Direct computations (do them!) gives

$$P_{zz} = H_t P_{t|t-1} H_t^T + R_t, \quad P_{\tilde{x}z} = P_{t|t-1} H_t^T$$

Kalman-Bucy - Kalman filtering in continuous time

A continuous time model

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + G(t)w(t) \\ y(t) &= H(t)x(t) + e(t) \end{aligned}$$

with the covariance function

$$\mathrm{E}\{\begin{pmatrix} w(t) \\ v(t) \end{pmatrix} \begin{pmatrix} w(s) \\ v(s) \end{pmatrix}^T\} = \begin{pmatrix} Q(t)\delta(t-s) & 0 \\ 0 & R(t)\delta(t-s) \end{pmatrix}$$

then the Kalman filter is given by

$$\begin{aligned} \dot{\hat{x}}(t) &= A(t)x(t) + K(y(t) - H(t)\hat{x}(t)) \\ K(t) &= P(t)H^T(t)R^{-1}(t) \\ \dot{P}(t) &= F(t)P(t) + P(t)F^T(t) + G(t)Q(t)G^T(t) - K(t)R(t)K^T(t) \end{aligned}$$

with

$$\hat{x}(0) = x_0, \quad P(0) = P_0$$

Outline

- Reading material
- Kalman Filter
  - Linear Kalman filter
  - Extended Kalman Filter (EKF)
  - Iterated Extended Kalman Filter
  - Constant Gain Extended Kalman Filter
- Implementation, array/square-root algorithms
- Unscented Kalman Filter

Extended Kalman Filter (EKF)

Extended Kalman Filter is a method where the same methodology is used for non-linear systems. Again, left out u but it is straightforward to include.

$$
\begin{aligned}
x_{t+1} &= f(x_t, w_t) \\
y_t &= h(x_t) + e_t
\end{aligned}
$$

Difficulty 1

Expected value and variance in linear transformations are easy to express explicitly.

This is not true for nonlinear transforms.

Idea

Compute the observer gain and covariance matrices by linearizing around the current state x_t.

The problem is that we do not know x_t, instead linearize around our best guess $\hat{x}$.

EKF, the equations

Measurement update

$$
\begin{aligned}
\hat{x}_{t|t} &= \hat{x}_{t|t-1} + K_t(y_t - h(\hat{x}_{t|t-1})) \\
K_t &= P_{t|t-1}H_t^T(H_t P_{t|t-1} H_t^T + R_t)^{-1} \\
P_{t|t} &= P_{t|t-1} - K_t H_t P_{t|t-1}
\end{aligned}
\qquad
H_t = \frac{\partial}{\partial x} h(x)|_{x=\hat{x}_{t|t-1}}
$$

Time update

$$
\begin{aligned}
\hat{x}_{t+1|t} &= f(\hat{x}_{t|t}, 0) \\
P_{t+1|t} &= F_t P_{t|t} F_t^T + G_t Q_t G_t^T
\end{aligned}
\qquad
\begin{aligned}
F_t &= \frac{\partial}{\partial x} f(x, w)|_{x=\hat{x}_{t|t}, w=0} \\
G_t &= \frac{\partial}{\partial w} f(x, w)|_{x=\hat{x}_{t|t}, w=0}
\end{aligned}
$$

EKF - some comments

- very common approach and have, according to reports, worked well in many applications
- requires availability of gradients
- unclear robustness properties
- a bad initial guess (or if you temporarily gets lost), or possibly strong non-linearities, could lead to divergence since it relies on a linerization.
- It is possible to derive higher order, at least 2:nd order, EKF. But then you have to have access to Hessians.
- several formulations of the (E)KF equations. The most common, shown here, is called the covariance form. It is possible to formulate using the inverse of P and is then called the information form. Have a little different numerical properties.

Outline

- Reading material
- Kalman Filter
  - Linear Kalman filter
  - Extended Kalman Filter (EKF)
  - Iterated Extended Kalman Filter
  - Constant Gain Extended Kalman Filter
- Implementation, array/square-root algorithms
- Unscented Kalman Filter

Iterated Extended Kalman Filter

With strong non-linearities, you could get into problems with convergence with the EKF.

A simple idea that could help

In the measurement update the filter linearizes around $\hat{x}_{t|t-1}$ to compute $\hat{x}_{t|t}$. Why not iterate and linearize around the new, and improved, estimate of x_t?

IEKF, measurement update

$$\hat{x}_{t|t} = \hat{x}_{t|t}^{(m)}, \quad P_{t|t} = P_{t|t-1} - K_t^{(m)} H_t^{(m)} P_{t|t-1}$$

where $\hat{x}_{t|t}^{(0)} = \hat{x}_{t|t-1}$

$$\begin{aligned}\hat{x}_{t|t}^{(i)} &= \hat{x}_{t|t-1} + K_t^{(i)}(y_t - h(x_{t|t-1}))\\ K_t^{(i)} &= P_{t|t-1}H_t^{(i)T}(H_t^{(i)}P_{t|t-1}H_t^{(i)T} + R_t)^{-1}\\ H_t^{(i)} &= \frac{\partial}{\partial x}h(x)|_{x=\hat{x}_{t|t}^{(i)}}\end{aligned}$$

Outline

- Reading material
- Kalman Filter
  - Linear Kalman filter
  - Extended Kalman Filter (EKF)
  - Iterated Extended Kalman Filter
  - Constant Gain Extended Kalman Filter
- Implementation, array/square-root algorithms
- Unscented Kalman Filter

Constant Gain Extended Kalman Filter

To reduced the computational burden, sometimes what is referred to as Constant Gain Extended Kalman Filter (CGEKF) is used. For a system

$$\begin{aligned}\dot{x} &= F(z)x + G(z)w\\ y &= H(z)x + e\end{aligned}$$

where z is a known time-function (typically y, u etc.). Then the CGEKF is

$$\dot{\hat{x}} = F(z)\hat{x} + K(z)(y - h(\hat{x}))$$

where $K(z)$ is computed as the stationary Kalman gain

$$\begin{aligned}K(z) &= P(z)H^T(z)R^{-1}\\ 0 &= F(z)P(z) + P(z)F^T(z) + G(z)QG^T(z) - K(z)RK^T(z)\end{aligned}$$

- Similar to a gain scheduled observer design
- For a simple alternative, pre-compute the gains in a number of operating points and interpolate the corresponding $K(z)$.
- Can be formulated for more general non-linearities
- IS even more sensitive than an EKF for non-linearities and bad initial guesses.

Outline

- Reading material
- Kalman Filter
  - Linear Kalman filter
  - Extended Kalman Filter (EKF)
  - Iterated Extended Kalman Filter
  - Constant Gain Extended Kalman Filter
- Implementation, array/square-root algorithms
- Unscented Kalman Filter

Array/square-root algorithms

Source of problems

Badly scaled problems might lead to divergence in (E)KF due to

- P_t non-symmetric
- P_t not positive definite

Array/square-root algorithms

Square root algorithms circumvent the problem by only computing the square root $P_t^{1/2}$ instead of P_t

Notation: Matrix A is a square root of a matrix P if

$$P = AA^T$$

Array/square-root algorithms

There are many algorithms, but here's a simple one I like

Square root measurement update

- Form matrix $\mathcal{A}$ and make upper triangular, e.g., by QR factorization of $\mathcal{A}^T$

$$\mathcal{A} = \begin{pmatrix} R_t^{1/2} & H_t P_{t|t-1}^{1/2} \\ 0 & P_{t|t-1}^{1/2} \end{pmatrix}, \quad \mathcal{A}\Theta = \begin{pmatrix} X & 0 \\ Y & Z \end{pmatrix}$$

- Then

$$X = R_{e,t}^{1/2}, \quad Y = P_{t|t-1} H_t^T R_{e,t}^{-T/2}, \quad Z = P_{t|t}^{1/2}$$

i.e.

$$P_{t|t}^{1/2} = Z, \quad K_t = P_{t|t-1} H_t^T R_{e,t}^{-1} = YX^{-1}$$

The inverse of X is simple to compute since it is triangular.

See Chapter 12 in *Linear Estimation* by Kailath, Sayed, Hassibi for more information (can be downloaded via reference [22] on the course page).

Outline

- Reading material
- Kalman Filter
  - Linear Kalman filter
  - Extended Kalman Filter (EKF)
  - Iterated Extended Kalman Filter
  - Constant Gain Extended Kalman Filter
- Implementation, array/square-root algorithms
- Unscented Kalman Filter

Unscented Kalman Filter

Advantages with (E)KF

- Propagates expected value μ and covariance P, which gives relatively few paramaters to compute in each step.
- Linear transformations of μ/P is simple.
- Is possible to generalize to sets of expected values/covariances for multi-modal distributions etc.

Disadvantagew with EKF

- works well if the linearization for most purposes well describes the nonlinear dynamics
- Requires access to gradientents of f_x and h_x. Thee can be expensive/hard to compute, or maybe they are even non-differentiable.
- Primarily unimodal distributions.

UKF

Unscented Kalman Filter is an algorithm that avoids linearization and approaches the filtering more direct.

Unscented Transform

Basic idea

The difficulty in nonlinear filtering is to describe distributions under nonlinear transformations

$$z = f(x)$$

Idea is to **carefully** choose a set of points, called sigma points, to represent the distribution for x. Send these through the non-linearity and compute a weighted average and covariance for the transformed points

Unscented transform - a standard version

There are many ways (and this is important!), to choose sigma points. $2N + 1$ points are chosen as

$$x^{(0)} = E\{x\}, \qquad w^{0} \text{ parameter}$$

$$x^{(\pm i)} = x^{(0)} \pm \sqrt{\frac{n_x}{1 - w^{(0)}}} u_i, \qquad w^{(\pm i)} = \frac{1 - w^{(0)}}{2n_x}$$

u_i is the i:th column in the square root of $cov\{x\}$.

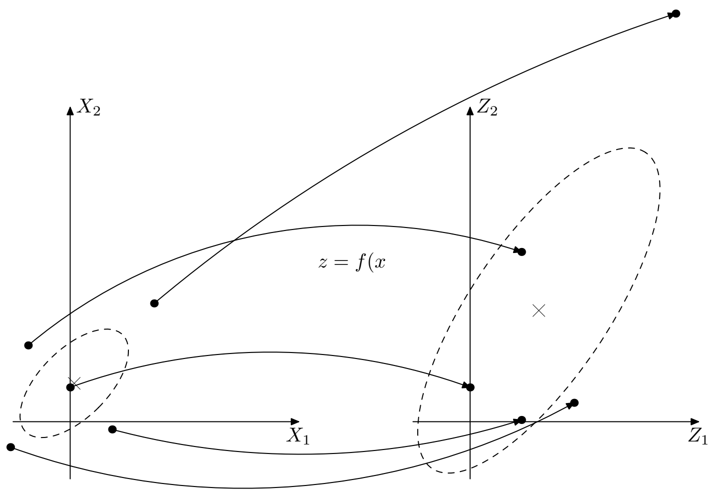
Unscented Transform

We have a random variable x that is transformed through a nonlinear functionz $= f(x)$. What is the expected value and covariance of z?

1. Compute sigma points/weights $x^{(i)}/w^{(i)}$ and the corresponding $z^{(i)} = f(x^{(i)})$.
2. the expected value is estimated as
$$\mu_z = \sum_i w^{(i)} z^{(i)}$$
3. the covariance is estimated as
$$\Sigma_z = \sum_i w^{(i)} (z^{(i)} - \mu_z)(z^{(i)} - \mu_z)^T$$

Often, different weights are used for estimating expected value and covariance. See Julier/Uhlmann for more details.

Unscented Transform (figure borrowed from G. Hendeby)



$$\mu_z = \sum_i w^{(i)} z^{(i)}, \quad \Sigma_z = \sum_i w^{(i)} (z^{(i)} - \mu_z)(z^{(i)} - \mu_z)^T$$

UKF - Unscented Transform to propagate μ and P

In the model

$$x_{t+1} = f(x_t, w_t)$$
$$y_t = h(x_t) + e_t$$

it is the extended random variable

$$\mathcal{X}_e = \begin{pmatrix} x_t \\ w_t \\ e_t \end{pmatrix}$$

that is passed through nonlinear functions. Here, there are opportunities for simplifications, there are typically $2N+1$ sigma points.

For example in

$$x_{t+1} = f(x_t) + Gw_t)$$
$$y_t = h(x_t) + e_t$$

it is only x that goes through non-linearities which reduces the number of sigma points substantially from $4n_x + 2m + 1$ to $2n_x + 1$.

UKF

Measurement update

From the beginning of the lecture, we had

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + P^{xy}_{t|t-1} P^{-yy}_{t|t-1}(y_t - \hat{y}_t)$$
$$P_{t|t} = P_{t|t-1} - P^{xy}_{t|t-1} P^{-yy}_{t|t-1} P^{xyT}_{t|t-1}$$

where

$$\hat{y}_t = \sum_i w^{(i)} y_t^{(i)}, \quad y_t^{(i)} = h(x^{(i)}_{t|t-1}, e_t^{(i)})$$
$$P^{yy}_{t|t-1} = \sum_i w^{(i)} (y_t^{(i)} - \hat{y}_t)(y_t^{(i)} - \hat{y}_t)^T$$
$$P^{xy}_{t|t-1} = \sum_i w^{(i)} (x^{(i)}_{t|t-1} - \hat{x}_{t|t-1})(y_t^{(i)} - \hat{y}_t)^T$$

Time update is done in a corresponding way, see Julier/Uhlmann or G. Hendeby.

UKF, some comments

- Doesn't need any gradients, but more evaluations of functions $f()$ and $h()$.
- Can be compared to a numerical approximation approximation of the gradients that typically requires (at least) $2n$ evaluations (2 per dimension).
- Tuning parameters are, as usual, matrices $P(0)$, Q, R *and* choice of sigma points.
- Choice of sigma points can have significant impact on filter performance.
- The choice shown here is a simple first approach, more sophisticated chjoices are possible. For example, more points and weight to preserve higher order moments.

Exercise

Objective

The objective is to implement and evaluate general purpose EKF and UKF algorithms.

Tasks

For the EKF you should implement both a standard, and a square root algorithm. To help evaluate your implementation, two skeleton files are provided: `task1.m` and `EKF.m`. The file `EKF.m` is where you can implement your EKF. The file `task1.m` generates data, and gives a skeleton for calling the EKF function for a robot position estimation task.
The UKF can be implemented using a similar design.

A scenario will be described on the next slide.

Exercise, cont'

The scenario is a mobile robot, travelling at constant speed V, with range and heading measurements. The model, in discrete time form, is

$$\begin{aligned} x_{t+1} &= x_t + T_s V \cos(\theta_t) & \qquad y_{1,t} &= \sqrt{x_t^2 + y_t^2} \\ y_{t+1} &= y_t + T_s V \sin(\theta_t) & \qquad y_{2,t} &= \tan^{-1}(y_t/x_t) \\ \theta_{t+1} &= \theta_t + T_s u_t & & \end{aligned}$$

where u_t is the control-signal (steer rate). The sampling rate $T_s = 0.1$ sec and velocity $V = 3$ m/s. Tune the EKF to obtain plots similar to (or better than)