F2E5216/TS1002 Adaptive Filtering and Change Detection

Fredrik Gustafsson (LiTH) and Bo Wahlberg (KTH)

Linköpings universitet

Lecture 7

State Estimation (Chapter 8,9)

Kalman Filtering Applications

- Introduction to Kalman filtering
- Target tracking applications and motion models
- Change detection
- Sensor fusion

Kalman Filtering in Practise

You need a good model

$$\begin{aligned} x_{t+1} &= A_t x_t + B_{u,t} u_t + B_{v,t} v_t, \quad \mathrm{Cov}(x_0) = P_0, \ \mathrm{Cov}(v_t) = Q_t \\ y_t &= C_t x_t + e_t, \quad \mathrm{Cov}(e_t) = R_t \end{aligned}$$

and understanding of tuning:
- The adaptation gain α which scales Q to αQ (the state variation) or, equivalently, R to R/α (the measurement noise).
More state variation $\Leftrightarrow$ less measurement noise $\Leftrightarrow$ faster filter.
- The transient speed P_0.
Larger initial uncertainty $\Leftrightarrow$ larger P_0 $\Leftrightarrow$ shorter transient.

Trade-off fast tracking to good noise attenuation with α.
Trade-off initial knowledge of x_0 to short transient.

Kalman Filter

$$\hat{x}_t = F(q; \alpha, P_0, A, B, C, D, Q, R) \begin{pmatrix} y_t \\ u_t \end{pmatrix}$$

- Choose implementation (filter, predictor, smoother, square root algorithm). See `adkalman`.
- Iterate optimization of design parameter α, P_0 using subjective or objective (RMS) criteria.
- Iterate modeling until satisfaction.

Target Tracking

Applications:

- Air traffic control (ATC)
- Military applications (e.g. bearings only tracking).
- Surveillance (e.g. highway traffic)
- GPS (sort of target tracking)
- Navigation applications

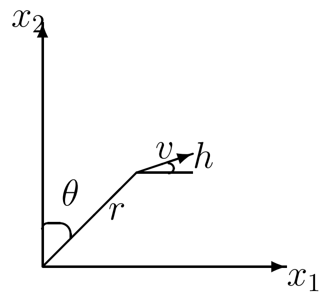
Characteristics:

- Sensors
- Motion models

Sensors in Tracking and Navigation

- Radar gives bearing and range
- Bearing only sensors including infra-blue (IR) and radar warning systems.
- Range only information in GPS.
- Camera and computer vision algorithms.
- Tracking information from other trackers.
- Navigation sensors as baro-altitude, terrain based algorithms using ground radar, inertial navigation systems (INS), GPS, etc.

Motion Models



Basic principle: Newtons law

$$
\begin{aligned}
m\ddot{x}_1 &= F \Rightarrow \\
\dot{x}_1 &= v_1 \\
\dot{v}_1 &= a_1 = F/m = w_1 = \text{white noise}
\end{aligned}
$$

Prior knowledge that can be used: *maximal acceleration much larger in lateral than longitudinal direction*. The **coordinated turn (CT)** assumption.

- A four state linear model with $x = (x_1, x_2, v_1, v_2)^T$.
- A six state linear model with $x = (x_1, x_2, v_1, v_2, a_1, a_2)^T$.
- A four or six state linear model with state dependent covariance matrix $Q(x_t)$ (CT assumption).
- A five state non-linear model with extra turn rate state.

$$x = (x_1, x_2, v_1, v_2, \omega)^T, \quad \text{or } x = (x_1, x_2, v, h, \omega)^T$$

(h is heading) together with an extended Kalman filter.
See nnnplane.

Q for Coordinated Turns

Assumption: let $Q = \mathrm{diag}(q_v, q_w)$ and $q_v = 0.01 q_w$.

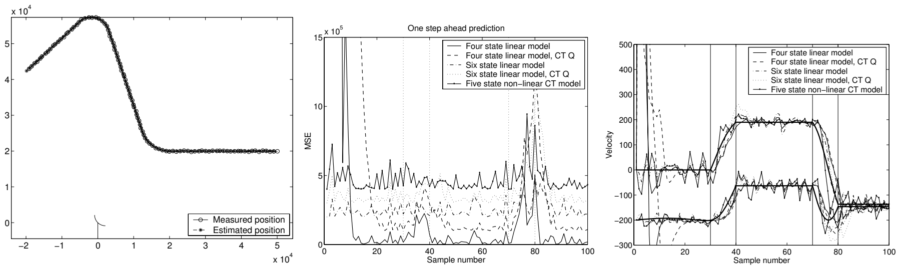
$$\theta = \arctan(x^{(4)}/x^{(3)})$$

$$Q = \begin{pmatrix} q_v \cos^2(\theta) + q_w \sin^2(\theta) & (q_v - q_w)\sin(\theta)\cos(\theta) \\ (q_v - q_w)\sin(\theta)\cos(\theta) & q_v \sin^2(\theta) + q_w \cos^2(\theta) \end{pmatrix}$$

$$B_v = \begin{pmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{pmatrix} \quad \text{or} \quad B_v = \begin{pmatrix} T^3/3 & 0 \\ 0 & T^3/3 \\ T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{pmatrix}$$

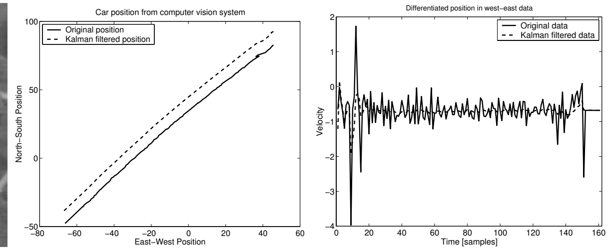
Application: Air Traffic Control

Comparison of the five different models for ATC.



Application: Highway Surveillance

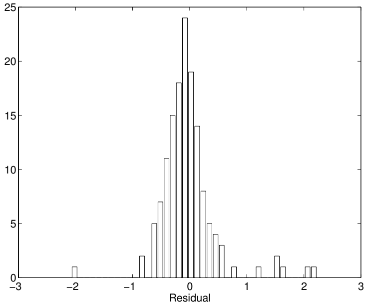
A helicopter hovering over a highway measures the position of all cars for surveillance purposes (the WITAS project on LiTH)



The bias in the middle plot is just there to highlight the illustration

Noise Distribution

Residuals from highway filtering.



- Measurement noise can in applications often be considered as Gaussian noise.
- Validation of KF optimality
- Validation of assumptions in change detection algorithms

Statistical Derivation

Use conditional expectation.

Assume that X and Y are jointly gaussian

$$\begin{pmatrix} X \\ Y \end{pmatrix} \in \mathrm{N}\left(\begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}, \begin{pmatrix} P_{xx} & P_{xy} \\ P_{xy} & P_{yy} \end{pmatrix} \right) = \mathrm{N}\left(\mu, P\right)$$

Conditional distribution of X, given $Y = y$:

$$(X|Y=y) \in N(\mu_x + P_{xy}P_{yy}^{-1}(y-\mu_y), P_{xx} - P_{xy}P_{yy}^{-1}P_{yx})$$

Conditional Mean

$$\begin{aligned} \hat{x}^{CM} &= \mathrm{E}(X|Y=y) = \mu_x + P_{xy}P_{yy}^{-1}(y-\mu_y) \\ \mathrm{Cov}(\hat{x}^{CM}) &= P_{xx} - P_{xy}P_{yy}^{-1}P_{yx} \end{aligned}$$

Conditional Mean= Minimum Variance Estimator
= Maximum A Posteriori Estimator (only for Gaussian)

Kalman Filter

- Let $X = x(t)|y(1), .., y(t-1)$ and $Y = y(t)|y(1), .., y(t-1)$.
- Only P_{xx}, P_{yy}, P_{xy} are need. They can be determined from the model.

$$\begin{aligned} x_{t+1} &= A_t x_t + B_{u,t}u_t + B_{v,t}v_t, \quad \mathrm{Cov}(x_0) = P_0, \ \mathrm{Cov}(v_t) = Q_t \\ y_t &= C_t x_t + e_t, \quad \mathrm{Cov}(e_t) = R_t \end{aligned}$$

The Algorithm

1. Let $x_0 \in N(\hat{x}_{0|0}, P_{0|0})$
2. Assume $x_{t-1}|y^{t-1} \in N(\hat{x}_{t-1|t-1}, P_{t-1|t-1})$
3. **Time update (just simulation):**

$$\begin{aligned} \hat{x}_{t|t-1} &= A_{t-1}\hat{x}_{t-1|t-1} + B_{u,t-1}u_{t-1} \\ P_{t|t-1} &= A_{t-1}P_{t-1|t-1}A_{t-1}^T + B_{v,t-1}Q_{t-1}B_{v,t-1}^T \quad \Rightarrow \\ (x_t|y^{t-1}) &\in N(\hat{x}_{t|t-1}, P_{t|t-1}) \end{aligned}$$

4. **Measurement update:**

$$\begin{aligned} \mathrm{E}(X|Y) &= \mu_x + P_{xy}P_{yy}^{-1}(Y-\mu_\gamma) \\ \mathrm{Cov}(X|Y) &= P_{xx} - P_{xy}P_{yy}^{-1}P_{yx}. \end{aligned}$$

$$\begin{aligned} P_{xy} &= E[(x_t - E[x_t])(y_t - E[y_t])^T|y^{t-1}] \\ &= E((x_t - \hat{x}_{t|t-1})(C_t x_t + B_{v,t}v_t - C_t\hat{x}_{t|t-1})^T) \\ &= P_{t|t-1}C_t^T = P_{yx}^T \end{aligned}$$

This gives

$$\begin{aligned} \hat{x}_{t|t} &= \hat{x}_{t|t-1} + P_{t|t-1}C_t^T(C_tP_{t|t-1}C_t^T + R_t)^{-1}(y_t - C_t\hat{x}_{t|t-1}) \\ P_{t|t} &= P_{t|t-1} - P_{t|t-1}C_t^T(C_tP_{t|t-1}C_t^T + R_t)^{-1}C_tP_{t|t-1} \end{aligned}$$

$$\Rightarrow\ (x_t|y^t) \in N(\hat{x}_{t|t}, P_{t|t})$$

Innovations

$$\varepsilon_t = y_t - C_t\hat{x}_{t|t-1}, \quad \mathrm{Cov}(\varepsilon_t) = S_t = C_tP_{t|t-1}C_t^T + R_t$$

Independent (or uncorrelated)!

Innovation model:

$$y_t = C_t\hat{x}_{t|t-1} + \varepsilon_t$$

Conditional distribution:

$$(y_t|y^{t-1}) \in N(C_t\hat{x}_{t|t-1}, S_t)$$

Iterate Bayes' rule

$$p(y^t) = p(y_t|y^{t-1})p(y^{t-1})$$

The Information Filter

Kalman filter has simple time update and complicated measurement update.

Information filter has simple measurement update and complicated time update.

Compare least squares algorithm ($A = I$ and $Q = 0$):

$$\begin{aligned} f_t &= f_{t-1} + C_t^T R_t^{-1} y_t \\ P_t^{-1} &= P_{t-1}^{-1} + C_t^T R_t^{-1} C_t \\ \hat{x}_{t|t} &= P_t f_t. \end{aligned}$$

Introduce the transformed state vector a (cf the vector f in LS):

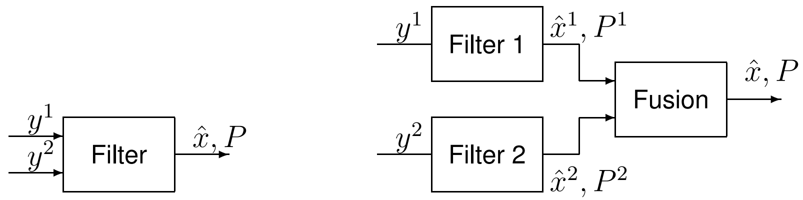
$$\begin{aligned} \hat{a}_{t|t} &= P_{t|t}^{-1}\hat{x}_{t|t} \\ \hat{a}_{t|t-1} &= P_{t|t-1}^{-1}\hat{x}_{t|t-1} \end{aligned}$$

The Kalman filter time update is now very messy but measurement update is simple

$$\begin{aligned} \hat{a}_{t|t} &= \hat{a}_{t|t-1} + C_t^T R_t^{-1} y_t \\ P_{t|t}^{-1} &= P_{t|t-1}^{-1} + C_t^T R_t^{-1} C_t \end{aligned}$$

Sensor Fusion

Centralized filtering **Decentralized filtering**



Advantage: fault detection by "voting" possible.
Disadvantage: heavy signaling.
Practical constraint: built-in KF in sensors.

Centralized Filtering

Simple concept. Just concatenate the measurements.

$$y_t = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix} = \begin{pmatrix} C_1 \\ C_2 \\ \vdots \\ C_m \end{pmatrix} x_t + \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_m \end{pmatrix}$$

General Fusion Formula

Suppose two independent state estimates

$$\hat{x}^1, \quad \hat{x}^2, \quad P^1, \quad P^2$$

Fusion straightforward:

$$\begin{aligned} \hat{x} &= P\left((P^1)^{-1}\hat{x}^1 + (P^2)^{-1}\hat{x}^2\right) \\ P &= \left((P^1)^{-1} + (P^2)^{-1}\right)^{-1}. \end{aligned}$$

Decentralized Fusion

The sub-optimal approach. State space model:

$$\begin{aligned} \begin{pmatrix} x^1_{t+1} \\ x^2_{t+1} \end{pmatrix} &= \begin{pmatrix} A_t & 0 \\ 0 & A_t \end{pmatrix} \begin{pmatrix} x^1_t \\ x^2_t \end{pmatrix} + \begin{pmatrix} B_t & 0 \\ 0 & B_t \end{pmatrix} \begin{pmatrix} v^1_t \\ v^2_t \end{pmatrix}, \quad \mathrm{Cov}(v^i_t) = Q \\ y_t &= \begin{pmatrix} C^1_t & 0 \\ 0 & C^2_t \end{pmatrix} \begin{pmatrix} x^1_t \\ x^2_t \end{pmatrix} + \begin{pmatrix} e^1_t \\ e^2_t \end{pmatrix} \end{aligned}$$

The diagonal forms of $\bar{A}$, $\bar{B}$ and $\bar{C}$ imply:
1. KF decentralized.
2. The state estimates will be independent. Thus, the fusion formula applies.
3. There is only one system, and thus only one state noise! The state noise should be

$$\begin{pmatrix} B_t \\ B_t \end{pmatrix} v_t$$

Consequence: P is *too small* in the suboptimal approach.

The Optimal Approach

The Central Filter must recover the measurements (“inverse KF”) or rather their information.

Derivation from the information filter. Study the measurement update:

$$\hat{a}_{t|t} = \hat{a}_{t|t-1} + C_t^T R_t^{-1} y_t$$

where $\hat{a}_{t|t} = \hat{P}_{t|t}^{-1}\hat{x}_{t|t}$. Measurement error covariance block diagonal:

$$\hat{a}_{t|t} = \hat{a}_{t|t-1} + (C_t^1)^T (R_t^1)^{-1} y_t^1 + (C_t^2)^T (R_t^2)^{-1} y_t^2$$

Note that this can be done without forming larger matrices

Each decentralized KF has computed a measurement update

$$\hat{a}_{t|t}^i = \hat{a}_{t|t-1}^i + (C_t^i)^T (R_t^i)^{-1} y_t^i$$

Recover the information from available KF quantities:

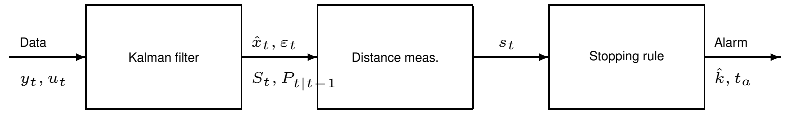
$$\begin{aligned}(C_t^i)^T (R_t^i)^{-1} y_t^i &= \hat{a}_{t|t}^i - \hat{a}_{t|t-1}^i \\ &= (\hat{P}_{t|t}^i)^{-1}\hat{x}_{t|t}^i - (\hat{P}_{t|t-1}^i)^{-1}\hat{x}_{t|t-1}^i\end{aligned}$$

Optimal decentralized fusion:

$$\hat{P}_{t|t}^{-1}\hat{x}_{t|t} = \hat{P}_{t|t-1}^{-1}\hat{x}_{t|t-1} + \sum_{i=1}^{2}\left((\hat{P}_{t|t}^i)^{-1}\hat{x}_{t|t}^i - (\hat{P}_{t|t-1}^i)^{-1}\hat{x}_{t|t-1}^i\right)$$

State covariance in the same way.
Time update according to the Central KF.

Change Detection



Distance measure s_t:
1. Normalized innovations

$$s_t = S_t^{-1/2}\varepsilon_t.$$

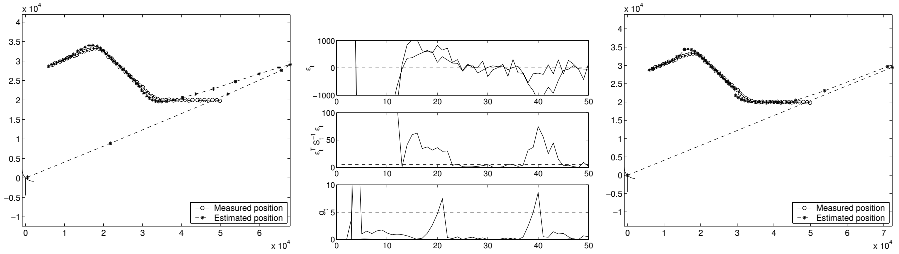
Disadvantage: Not sensitive to variance changes.
2. The squared normalized innovations

$$s_t = \varepsilon_t^T S_t^{-1}\varepsilon_t - n_y.$$

Disadvantage: Sensitive to scalings of covariance matrices.

Example

Target tracking example. Innovations from a Kalman filter.



1. The 2D position error.
2. The normalized square (without subtraction of n_y).
3. The test statistics in the two-sided CUSUM test using the sum of innovations as input.

Next Time: Filter Banks for State Changes

- Explicit modeling of additive change: GLR and MLR
- Multiple models: pruning, merging and off-line algorithms