SENSOR FUSION

Filtering CRLB

Sensor Fusion

Fredrik Gustafsson
fredrik.gustafsson@liu.se

Gustaf Hendeby
gustaf.hendeby@liu.se

Linköping University

Purpose

Derive the corresponding performance bound for filtering that exists for estimation

- Add the dynamic state-space models: $x_{k+1} = f(x_k, v_k)$.
- to the standard estimation model: $y_k = h(x_k, e_k)$.
- Note that x_k might not be directly observable from y_k in filtering, in contrast to estimation.
- There are two kind of filtering bounds: parametric and posterior CRLB.

State-Space Models

Nonlinear model:

$$x_{k+1} = f(x_k, v_k) \qquad \text{or} \qquad x_{k+1}|x_k \sim p(x_{k+1}|x_k)$$
$$y_k = h(x_k, e_k) \qquad \text{or} \qquad y_k|x_k \sim p(y_k|x_k)$$

Nonlinear model with additive noise:

$$x_{k+1} = f(x_k) + v_k \qquad \text{or} \qquad x_{k+1}|x_k \sim p(x_{k+1}|x_k) = p_{v_k}(x_{k+1} - f(x_k))$$
$$y_k = h(x_k) + e_k \qquad \text{or} \qquad y_k|x_k \sim p(y_k|x_k) = p_{e_k}(y_k - h(x_k))$$

Linear model:

$$x_{k+1} = F_k x_k + G_k v_k$$
$$y_k = H_k x_k + e_k$$

Gaussian model: $v_k \sim \mathcal{N}(0, Q_k)$, $e_k \sim \mathcal{N}(0, R_k)$ and $x_0 \sim \mathcal{N}(0, P_0)$

CRLB: estimation

- The Fisher information matrix, $\mathcal{I}(x)$, is defined as

$$\mathcal{I}(x) = \mathrm{E}\Big(\nabla_x^T \log p_{\mathbf{e}}\big(\mathbf{y} - \mathbf{h}(x)\big) \nabla_x \log p_{\mathbf{e}}\big(\mathbf{y} - \mathbf{h}(x)\big)\Big)$$

$$\nabla_x \log p_{\mathbf{e}}\big(\mathbf{y} - \mathbf{h}(x)\big) = \begin{pmatrix} \frac{\partial \log p_{\mathbf{e}}\left(\mathbf{y}-\mathbf{h}(x)\right)}{\partial x_1} & \dots & \frac{\partial \log p_{\mathbf{e}}\left(\mathbf{y}-\mathbf{h}(x)\right)}{\partial x_{n_x}} \end{pmatrix}$$

- For Gaussian $\mathbf{e}$, then (compare with WLS covariance!)

$$\mathcal{I}(x) = \mathbf{H}^T(x)\mathbf{R}^{-1}(x)\mathbf{H}(x),$$
$$\mathbf{H}(x) = \nabla_x \mathbf{h}(x).$$

- Information is *additive*, so if two or more sensors give independent observations $y_k = h_k(x) + e_k$, then $\mathcal{I} = \sum_k \mathcal{I}_k$.
- CRLB provides a lower bound on root mean square error

$$\mathrm{RMSE} = \sqrt{\mathrm{E}\big((x_1^0 - \hat{x}_1)^2 + (x_2^0 - \hat{x}_2)^2\big)} = \sqrt{\mathrm{tr}\big(\mathrm{Cov}(\hat{x})\big)}$$
$$\geq \sqrt{\mathrm{tr}\big(\mathcal{I}^{-1}(x^0)\big)}$$

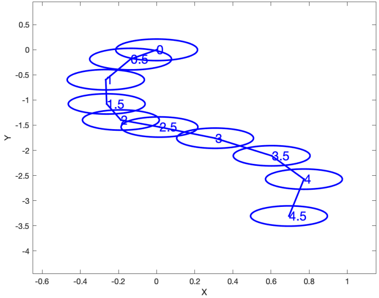
CRLB: filtering

- CRLB developed for static parameter x, with many measurements $y_{1:k}$.
- The filtering CRLB concerns the case where x is replaced with $x_{1:k}$, with the constraints $x_{n+1} = f(x_n) + v_n$, $n = 1, 2, \dots, k-1$.
- Two cases:
  - Parametric CRLB for filtering: $x_{1:k}$ is seen as a parameter with a *true value* $x^0_{1:k}$.
  - Posterior, or Bayesian, CRLB for filtering: $x_{1:k}$ is seen as a stochastic variable with a *prior* $p(x_{1:k})$.
- Parametric CRLB better in practice: easy to calculate, easy to interpret (given a certain trajectory and model, how well can a nonlinear filter estimate this trajectory?)
- Posterior CRLB useful for theoretical studies.

Parametric CRLB

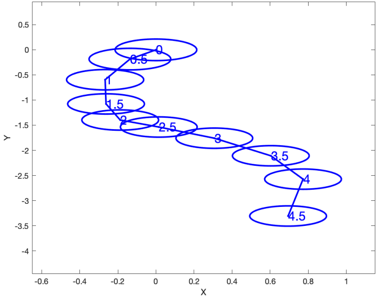
- The parametric CRLB gives a lower bound on estimation error for a fixed trajectory $x_{1:k}$. That is, $\mathrm{Cov}(\hat{x}_{k|k}) \succeq P^{\mathrm{crlb}}_{k|k}$.
- Algorithm identical to the Riccati equation (covariance update) in KF, where the gradients are evaluated along the trajectory $x_{1:k}$:

$$
\begin{aligned}
P_{k+1|k} &= F_k P_{k|k} F_k^T + G_k Q_k G_k, \\
P_{k+1|k+1} &= P_{k+1|k} - P_{k+1|k} H_k^T (H_k P_{k+1|k} H_k^T + R_k)^{-1} H_k P_{k+1|k}, \\
F_k &= \nabla_{x_k} f(x_k, v_k), \\
G_k &= \nabla_{v_k} f(x_k, v_k), \\
H_k &= \nabla_{x_k} h(x_k, e_k).
\end{aligned}
$$



Example

- Define and simulate a a constant velocity model
- Convert to a nonlinear model and compute the CRLB
- Illustrate with confidence ellipsoids




```
mss=exlti('cv2D')
mss =
              / 1   0   0.5    0 \
              | 0   1     0  0.5 |
x[k+1]  =     | 0   0     1    0 |  x[k]  + v[k]
              \ 0   0     0    1 /
             /1   0   0   0\
y[k]  =      \0   1   0   0/   x[k] + e[k]
                    /0.016          0       0.062           0\
                   |     0      0.016           0       0.062 |
Q = Cov(v)  =      | 0.062          0        0.25           0 |
                    \    0      0.062           0        0.25/
                   / 0.01           0\
R = Cov(e)  =      \    0        0.01/
mnl=nl(mss)
mnl  =
NL object: Constant velocity motion model
x[k+1]  = [1 0 0.5 0;0 1 0 0.5;0 0 1 0;0 0 0 1]*x(1:4,:) + v
    y  = [1 0 0;0 1 0 0]*x(1:4,:) + e
    x0'  = [0,0,0,0]
  States:  X      Y      vX      vY
  Outputs: X      Y
y=simulate(mss,10);
xcrlb=crlb(mnl,y);
xplot2(xcrlb,'conf',90)
```

Posterior CRLB

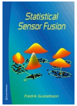
- Average FIM over all possible trajectories $x_{1:k}$ with respect to v_k.
- Much more complicated expressions.
- For linear systems, the parametric and posterior CRLB coincide.

CRLB for non-Gaussian noise

- Replace covariances Q and R with the **intrinsic accuracy** $\mathcal{I}_v^{-1}$ and $\mathcal{I}_e^{-1}$.
- Same Ricatti equation for the parametric CRLB.
- Note $\mathcal{I}_v^{-1} \leq \mathrm{Cov}(v)$ and $\mathcal{I}_e^{-1} \leq \mathrm{Cov}(e)$.
- The Riccati equation monotone function of the covariance matrix.
- Gaussian noise is the least informative distribution, thus the KF $P_{k|k}$ is a conservative bound on the possible performance with nonlinear filters.

Summary

- CRLB standard tool in estimation problems.
- Key trick going from estimation CRLB to filtering CRLB: consider the batch problem with parameter $x_{1:k}$.
- The parametric CRLB can be computed by a Kalman filter, and provides a filtering lower bound for a particular trajectory.
- Gaussian noise is the worst case distribution, meaning it is the least informative giving the largest lower bound.



Section 6.5