Le 4: detection and filter theory

**Whiteboard:**
- Detection theory
  - Notation overview
  - Neyman-Pearson’s lemma
  - Detection tests for no model, linear model and nonlinear model
- Derivation of Bayesian optimal filter

**Slides:**
- Detection summary and example
- Filtering model definitions
- General view of nonlinear filtering
- Overview of optimal and approximate filters
- CRLB for filtering

(Renewed) Lecture Plan

<table>
<thead>
<tr>
<th>Le</th>
<th>Content</th>
<th>Ch</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Course overview. Estimation theory for linear models</td>
<td>1, 2</td>
</tr>
<tr>
<td>2</td>
<td>Estimation theory for nonlinear models</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Cramér-Rao lower bound. Models for sensor network</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>Detection theory. Filter theory</td>
<td>5, 6</td>
</tr>
<tr>
<td>5</td>
<td>Modeling and motion models</td>
<td>12–14</td>
</tr>
<tr>
<td>6</td>
<td>Kalman filter. Kalman filter approximations (EKF, UKF)</td>
<td>7, 8</td>
</tr>
<tr>
<td>7</td>
<td>Point-mass filter and particle filter</td>
<td>9</td>
</tr>
<tr>
<td>8</td>
<td>Simultaneous localization and mapping (SLAM)</td>
<td>11</td>
</tr>
<tr>
<td>9</td>
<td>Particle filter theory. Marginalized particle filter</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>Sensors and filter validation. Industry case study</td>
<td>14, 15</td>
</tr>
</tbody>
</table>

Lecture 3: summary

- Sensor networks: typical models TOA, TDOA, DOA, and RSS
- CRLB Theorem: for any unbiased estimator $\hat{x}$,
  \[
  \text{cov}(\hat{x}) \geq I^{-1}(x^o),
  \]
  Fisher information matrix (FIM) $I^{-1}(x)$
- ML estimate efficient: asymptotically unbiased satisfying CRLB.
- Bayes estimator for estimation in $y = h(x) + e$
  \[
  \hat{x}^\text{Bayes} = \arg\max_x p(x|y) = \arg\max_x \frac{p(x)}{p(y)} \frac{p(e)}{p(y)}
  \]
Chapter 5 Overview

- Detection Theory:
  - (Generalized) likelihood ratio ((G)LR) test
  - Test statistics, $T(y)$
- Classification
  - Choose between many hypothesis, pick the most likely one
- Measurement association problem

Hypothesis Tests

Hypothesis test in statistics:

$H_0 : y = e$

$H_1 : y = x + e$

$e \sim p(e)$

General model-based test (sensor clutter versus target present):

$H_0 : y = e^0$

$H_1 : y = h(x) + e^1$

$e^0 \sim p^0(e^0)$

$e^1 \sim p^1(e^1)$

Special case: Linear model $h(x) = Hx$

First Example: revisited

Detect if a target is present

$y_1 = x + e_1$, $\text{cov}(e_1) = R_1$

$y_2 = x + e_2$, $\text{cov}(e_2) = R_2$

$y = Hx + e$, $\text{cov}(e) = R$, $H = \begin{pmatrix} I \\ I \end{pmatrix}$

$R = \begin{pmatrix} R_1 & 0 \\ 0 & R_2 \end{pmatrix}$

$T(y) = y^T R^{-T/2} \Pi R^{-1/2} y$

$\Pi = R^{-T/2} H (H^T R^{-1} H)^{-1} H R^{-1/2} = \begin{pmatrix} 22.8 & 22.2 & 5.0 & 0.0 \\ 22.2 & 22.8 & 0.0 & 5.0 \\ 5.0 & 0.0 & 22.8 & -22.2 \\ -0.0 & 5.0 & -22.2 & 22.8 \end{pmatrix}$

Numerical Simulation

Threshold for the test:

% NB! This invokes the chi2dist_erfinv function
% that is only remotely related to the
% mathematical erfinv function!

\begin{verbatim}
% NB! This is an approximation!
\end{verbatim}

\begin{verbatim}
h = erfinv(chi2dist(2), 0.999)
\end{verbatim}

Output:

\begin{verbatim}
h = 12.1638
\end{verbatim}

Note:

For no model (standard statistical test), $\Pi = I$. Both methods perform perfect $P_D = 1$. 

Distribution of test statistics ($h=12.1638$)

Linear model $T(y)$
Sensor Network Example

```
my = 5; nx = 2;
a = sensor('tow', my, 1); % Default network
s.pe = 0.2 * my; % Set noise level
plot(s)
```

Chapter 6 overview

- Dynamic state-space models: $x_{k+1} = f(x_k, v_k)$
- Measurement model: $y_k = h(x_k, e_k)$
- General Bayesian solution
- Filtering bounds: parametric and posterior CRLB

Simple and Generic Motion Models (1/2)

Newton’s force law $F = ma$ gives the “nearly constant velocity model” in $n$ dimensions:

$$x_{k+1} = F_k x_k + G_k v_k = \begin{pmatrix} I_n & T_l \end{pmatrix} x_k + \begin{pmatrix} T_l^2 I_n \end{pmatrix} v_k$$

where $x_k = (p_k^T, V_k^T)^T$.

Interpretation:

- $p_{k+1} = p_k + TV_k + \frac{T_l^2}{2} v_k$
- $V_{k+1} = V_k + TV_k$

where process noise corresponds to acceleration, $v_k = a_k$.

Linear transformation of independent stochastic vectors implies:

$$x_{k+1} = F_k x_k + G_k v_k, \quad x_k \sim \mathcal{N}(\hat{x}_{k|k}, P_{k|k}), \quad v_k \sim \mathcal{N}(0, Q_k)$$

$$\implies x_{k+1} \sim \mathcal{N}(F \hat{x}_{k|k}, F_k P_{k|k} F_k^T + G_k Q_k G_k^T)$$

State-Space Models

Nonlinear model:

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

Nonlinear model with additive noise:

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

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 $x_0 \sim \mathcal{N}(0, P_0)$
Similarly, a nearly constant acceleration model is
\[
x_{k+1} = F_k x_k + G_k v_k = \begin{pmatrix} I_n & T l_n & \frac{T^2}{2} l_n \\ 0_n & l_n & T l_n \\ 0_n & 0_n & l_n \end{pmatrix} x_k + \begin{pmatrix} \frac{I_n}{2} l_n \\ \frac{T^2}{2} l_n \\ T l_n \end{pmatrix} v_k
\]

More motion models in the next lecture.

Bayes’ Solution: nonlinear model with additive noise

Bayes law provides the recursion (measurement and time updates):
\[
\alpha = \int_{\mathbb{R}^n} p_{e_i}(y_k - h(x_k)|x_k) p(x_k|y_{1:k-1}) \, dx_k,
\]
\[
p(x_k|y_{1:k}) = \frac{1}{\alpha} p_{e_i}(y_k - h(x_k)) p(x_k|y_{1:k-1})
\]
\[
p(x_{k+1}|y_{1:k}) = \int_{\mathbb{R}^n} p_{e_i}(x_{k+1} - f(x_k)) p(x_k|y_{1:k}) \, dx_k
\]

To get analytical solution, we need a model that keeps the same functional form of the posterior during:
- the nonlinear transformation \(f(x_k)\)
- the addition of \(f(x_k)\) and \(v_k\)
- the inference of \(x_k\) from \(y_k\) done in the measurement update

A General Bayesian Filter Framework

1. **Estimation**: Provides the complete distribution \(p(x_k|y_k)\)
2. **Fusion**: Estimated information \(p(x_k|y_k)\) is merged with the prior information \(p(x_k|y_{1:k-1})\) to obtain \(p(x_k|y_{1:k})\)
3. **Transformation**: Propagate information through the dynamics \(z = f(x_k, u_k)\). This gives \(p(z|y_{1:k})\).
4. **Diffusion**: Add uncertainty from the process noise. This gives \(p(x_{k+1}|y_{1:k})\).
Practical Cases with Analytic Solution

Bayes solution can be represented with finite dimensional statistics analytically in the following cases:

- Linear Gaussian model (Kalman filter)
- Hidden Markov model (HMM)
- Linear-Gaussian mixture (Kalman filter filterbank; however exponential complexity in time)

General Approximation Approaches

1. Approximate the model to a case where an optimal algorithm exists.
   - Extended KF (EKF) which approximates the model with a linear one.
   - Unscented KF (UKF) and EKF2 that apply higher order approximations.

2. Approximate the optimal nonlinear filter for the original model.
   - Point-mass filter (PMF) which uses a regular grid of the state space and applies the Bayesian recursion.
   - Particle filter (PF) which uses a random grid of the state space and applies the Bayesian recursion.

CRLB: estimation

- The Fisher information matrix, $I(x)$, is defined as
  $$I(x) = E \left( \nabla_x^T \log p_e(y - h(x)) \nabla_x \log p_e(y - h(x)) \right)$$
  $$\nabla_x \log p_e(y - h(x)) = \left( \frac{\partial \log p_e(y - h(x))}{\partial x_1} \ldots \frac{\partial \log p_e(y - h(x))}{\partial x_n} \right)$$
- For Gaussian $e$, then (compare with WLS covariance!)
  $$I(x) = h^T(x) R^{-1} h(x), \quad h(x) = \nabla_x h(x).$$
- Information is additive, so if two or more sensors give independent observations $y_k = h_k(x) + e_k$, then $I = \sum_k I_k$.
- CRLB provides a lower bound on root mean square error
  $$\text{RMSE} = \sqrt{E((x_0^1 - \hat{x}_1)^2 + (x_0^2 - \hat{x}_2)^2)} = \sqrt{\text{tr}(\text{cov}(\hat{x}))}$$
  $$\geq \sqrt{\text{tr}(I^{-1}(x^0))}$$

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 $x_{n+1} = f(x_n) + v_n$, $n = 1, 2, \ldots, k - 1$.
- 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, $\text{cov}(\hat{x}_k|k) \succeq P_{k|k}^{\text{CRLB}}$.
- Algorithm identical to the Riccati equation (covariance update) in KF, where the gradients are evaluated along the trajectory $x_{1:k}$:
  
  $P_{k+1|k} = F_k P_{k|k} F_k^T + G_k Q_k G_k$,
  
  $P_{k+1|k+1} = P_{k+1|k} - P_{k+1|k} H_k^T (H_k P_{k+1|k} H_k^T + R_k)^{-1} H_k P_{k+1|k}$,
  
  $F_k = \nabla_{x_k} f(x_k, v_k)$,
  
  $G_k = \nabla_{v_k} f(x_k, v_k)$,
  
  $H_k = \nabla_{x_k} h(x_k, e_k)$.

**Posterior CRLB**

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

**Summary Lecture 4**

- Detection problems as hypothesis tests:
  
  $H_0 : y = e$,
  
  $H_1 : y = \hat{x} + e = h(x) + e$.

- Neyman-Pearson’s lemma: $T(y) = p_e(y - h(x^0)) / p_e(y)$ maximizes $P_D$ for given $P_{\lambda}$ (best ROC curve)
- In general case
  
  $T(y) = 2 \log p_e(y - h(x^0)) - 2 \log p_e(y) \sim \chi_n^2 (x^{0,T} I(x^0) x^0)$

- Bayes optimal filter
  
  $p(x_k|y_{1:k}) \propto p_e(y_k - h(x_k)) p(x_k|y_{1:k-1})$
  
  $p(x_{k+1}|y_{1:k}) = \int p_v(x_{k+1} - f(x_k)) p(x_k|y_{1:k}) dx_k$

- Intuitive work flow of nonlinear filter:
  - MU: estimation from $y_k = h(x_k) + e_k$ and fusion with $\hat{x}_{k|k-1}$
  - TU: nonlinear transformation $z = f(x_k)$ and diffusion from $x_{k-1} = z_k + v_k$.