ambiguous sensor data \((P_D < 1, \rho_F > 0)\)

\[ n_k + 1 \text{ possible interpretations of the sensor data } Z_k = \{z_k^j\}_{j=1}^{n_k} \]

- \(E_0\): the object was not detected; \(n_k\) false data in the Field of View (FoV)
- \(E_j, j = 1, \ldots, n_k\): Object detected; \(z_k^j\) is object measurement; \(n_k - 1\) false measurements

Consider the interpretations in the likelihood function \(p(Z_k, n_k | x_k)\)!
ambiguous sensor data \((P_D < 1, \rho_F > 0)\)

\(n_k + 1\) possible interpretations of the sensor data \(Z_k = \{z_{jk}^j\}_{j=1}^{n_k}\)

- \(E_0\): the object was not detected; \(n_k\) false data in the Field of View (FoV)
- \(E_j, j = 1, \ldots, n_k\): Object detected; \(z_{jk}^j\) is object measurement; \(n_k - 1\) false measurements

Consider the interpretations in the likelihood function \(p(Z_k, n_k|x_k)\)

\[
p(Z_k, n_k|x_k) = p(Z_k, n_k, \neg D|x_k) + p(Z_k, n_k, D|x_k)
\]

\(D = \text{“object was detected”}\)
ambiguous sensor data \((P_D < 1, \rho_F > 0)\)

\(n_k + 1\) possible interpretations of the sensor data \(Z_k = \{z_k^j\}_{j=1}^{n_k}\):

- \(E_0\): the object was not detected; \(n_k\) false data in the Field of View (FoV)
- \(E_j, j = 1, \ldots, n_k\): Object detected; \(z_k^j\) is object measurement; \(n_k - 1\) false measurements

\[ p(Z_k, n_k|x_k) = p(Z_k, n_k, \neg D|x_k) + p(Z_k, n_k, D|x_k) \]

\(D\) = “object was detected”

\[ = p(Z_k, n_k|\neg D, x_k) \frac{P(\neg D|x_k)}{1-P_D} + p(Z_k, n_k|D, x_k) \frac{P(D|x_k)}{P_D} \]

sensor parameter: detection probability \(P_D\)
ambiguous sensor data \((P_D < 1, \rho_F > 0)\)

\[ n_k + 1 \text{ possible interpretations of the sensor data } Z_k = \{z^j_k\}_{j=1}^{n_k} \]

- \(E_0\): the object was not detected; \(n_k\) false data in the Field of View (FoV)
- \(E_j, j = 1, \ldots, n_k\): Object detected; \(z^j_k\) is object measurement; \(n_k - 1\) false measurements

Consider the interpretations in the likelihood function \(p(Z_k, n_k | x_k)\):

\[
p(Z_k, n_k | x_k) = p(Z_k, n_k, \neg D | x_k) + p(Z_k, n_k, D | x_k) \quad D = \text{“object was detected”}
\]

\[
= p(Z_k, n_k | \neg D, x_k) P(\neg D | x_k) + p(Z_k, n_k | D, x_k) p(D | x_k)
\]

\[
= \frac{p(Z_k | n_k, \neg D, x_k)}{\text{|FoV|}^{n_k}} p(n_k | \neg D, x_k) (1 - P_D) + P_D \sum_{j=1}^{n_k} p(Z_k, n_k, j | D, x_k)
\]

false measurements: Poisson distributed in #, uniformly distributed in the FoV
ambiguous sensor data \((P_D < 1, \rho_F > 0)\)

\[ n_k + 1 \text{ possible interpretations of the sensor data } Z_k = \{z^j_k\}_{j=1}^{n_k}! \]

- \(E_0\): the object was not detected; \(n_k\) false data in the Field of View (FoV)
- \(E_j, j = 1, \ldots, n_k\): Object detected; \(z^j_k\) is object measurement; \(n_k - 1\) false measurements

Consider the interpretations in the likelihood function \(p(Z_k, n_k|x_k)\)!

\[
p(Z_k, n_k|x_k) = p(Z_k, n_k, \neg D|x_k) + p(Z_k, n_k, D|x_k) \quad D = \text{“object was detected”}
\]

\[
= p(Z_k, n_k, \neg D, x_k) P(\neg D|x_k) + p(Z_k, n_k, D, x_k) p(D|x_k)
\]

\[
= p(Z_k|n_k, \neg D, x_k) p(n_k|\neg D, x_k) (1 - P_D) + P_D \sum_{j=1}^{n_k} p(Z_k, n_k, j|D, x_k)
\]

\[
= |\text{FoV}|^{-n_k} p_F(n_k) (1 - P_D) + P_D \sum_{j=1}^{n_k} \frac{p(Z_k|n_k, j, D, x_k)}{|\text{FoV}|^{-(n_k-1)}N(z^j_n;Hx_k,R)} \frac{p(j|n_k, D)}{1/n_k} = p_F(n_k-1)
\]

Insert Poisson distribution: \(p_F(n_k) = \frac{(\rho_F|\text{FoV}|^{-n_k})}{n_k!} e^{-\rho_F|\text{FoV}|}\)
ambiguous sensor data \((P_D < 1, \rho_F > 0)\)

\[ n_k + 1 \text{ possible interpretations of the sensor data } Z_k = \{z_k^j\}_{j=1}^{n_k}! \]

- \(E_0\): the object was not detected; \(n_k\) false data in the Field of View (FoV)
- \(E_j, j = 1, \ldots, n_k\): Object detected; \(z_k^j\) is object measurement; \(n_k - 1\) false measurements

Consider the interpretations in the likelihood function \(p(Z_k, n_k|x_k)!\)

\[
p(Z_k, n_k|x_k) = p(Z_k, n_k, \neg D|x_k) + p(Z_k, n_k, D|x_k) \quad D = \text{“object was detected”}
\]

\[
= p(Z_k, n_k | \neg D, x_k) P(\neg D|x_k) + p(Z_k, n_k | D, x_k) p(D|x_k)
\]

\[
= p(Z_k | n_k, \neg D, x_k) p(n_k | \neg D, x_k) (1 - P_D) + P_D \sum_{j=1}^{n_k} p(Z_k, n_k, j | D, x_k)
\]

\[
= |\text{FoV}|^{-n_k} p_F(n_k) (1 - P_D) + P_D \sum_{j=1}^{n_k} p(Z_k | n_k, j, D, x_k) p(j | n_k, D) p(n_k | D)
\]

\[
= \frac{e^{-\rho_F|\text{FoV}|}}{n_k!} \rho_F^{n_k-1} \left( (1 - P_D) \rho_F + P_D \sum_{j=1}^{n_k} \mathcal{N}(z_k^j; Hx_k, R) \right)
\]
Likelihood Functions

The likelihood function answers the question:
What does the sensor tell about the state $x$ of the object?

(input: sensor data, sensor model)

- **ideal conditions, one object:** $P_D = 1$, $\rho_F = 0$
  
  at each time one measurement:
  \[
p(z_k|x_k) = \mathcal{N}(z_k; Hx_k, R)
  \]

- **real conditions, one object:** $P_D < 1$, $\rho_F > 0$
  
  at each time $n_k$ measurements $Z_k = \{z_{k1}^1, \ldots, z_{kn_k}^n\}$
  
  \[
p(Z_k, n_k|x_k) \propto (1 - P_D)\rho_F + P_D \sum_{j=1}^{n_k} \mathcal{N}(z_{kj}^i; Hx_k, R)
  \]
Bayes Filtering for: $P_D < 1, \rho_F > 0$, well-separated objects

state $x_k$, current data $Z_k = \{z^j_k\}_{j=1}^{m_k}$, accumulated data $\mathcal{Z}^k = \{Z_k, \mathcal{Z}^{k-1}\}$

interpretation hypotheses $E_k$ for $Z_k$

object not detected, $1 - P_D$

$z_k \in Z_k$ from object, $P_D$

$m_k + 1$ interpretations

interpretation histories $H_k$ for $\mathcal{Z}^k$

- tree structure: $H_k = (E_{H_k}, H_{k-1}) \in \mathcal{H}^k$
- current: $E_{H_k}$, prehistories: $H_{k-i}$

\[
p(x_k|Z^k) = \sum_{H_k} p(x_k, H_k|Z^k) = \sum_{H_k} \left( \frac{p(H_k|Z^k)}{\text{weight!}} \right) \cdot \left( p(x_k|H_k, Z^k) \text{ given } H_k \right)
\]

‘mixture’ density
Closer look: \( P_D < 1, \rho_F > 0 \), well-separated targets

filtering (at time \( t_{k-1} \)):
\[
p(x_{k-1}|\mathcal{Z}^{k-1}) = \sum_{H_{k-1}} p_{H_{k-1}} \mathcal{N}(x_{k-1}; x_{H_{k-1}}, P_{H_{k-1}})
\]

prediction (for time \( t_k \)):
\[
p(x_k|\mathcal{Z}^{k-1}) = \int dx_{k-1} p(x_k|x_{k-1}) p(x_{k-1}|\mathcal{Z}^{k-1}) \quad \text{(MARKOV model)}
\]
\[
= \sum_{H_{k-1}} p_{H_{k-1}} \mathcal{N}(x_k; Fx_{H_{k-1}}, FP_{H_{k-1}}F^\top + D) \quad \text{(IMM also possible)}
\]

measurement likelihood:
\[
p(Z_k, m_k|x_k) = \sum_{j=0}^{m_k} p(Z_k|E^j_k, x_k, m_k) P(E^j_k|x_k, m_k) \quad \text{\( E^j_k \): interpretations)}
\]
\[
\propto (1 - P_D) \rho_F + P_D \sum_{j=1}^{m_k} \mathcal{N}(z^j_k; Hx_k, R) \quad \text{(H, R, P_D, \rho_F)}
\]

filtering (at time \( t_k \)):
\[
p(x_k|\mathcal{Z}^k) \propto p(Z_k, m_k|x_k) p(x_k|\mathcal{Z}^{k-1}) \quad \text{(BAYES’ rule)}
\]
\[
= \sum_{H_k} p_{H_k} \mathcal{N}(x_k; x_{H_k}, P_{H_k}) \quad \text{(Exploit product formula)}
\]
Problem: Growing Memory Disaster:

\( m \) data, \( N \) hypotheses \( \rightarrow N^{m+1} \) continuations

radical solution: mono-hypothesis approximation
Problem: Growing Memory Disaster:

\[ m \text{ data, } N \text{ hypotheses } \rightarrow N^{m+1} \text{ continuations} \]

radical solution: mono-hypothesis approximation

- **gating:** Exclude competing data with \(|\nu^i_{k|k-1}| > \lambda!\)

  \[ \overset{\text{KALMAN filter (KF)}}{\rightarrow} \]

  + very simple, – \( \lambda \) too small: loss of target measurement
Problem: Growing Memory Disaster:

$m$ data, $N$ hypotheses $\rightarrow N^{m+1}$ continuations

radical solution: mono-hypothesis approximation

- **gating:** Exclude competing data with $||\nu_{k|k-1}^i|| > \lambda$!

  $\Rightarrow$ KALMAN filter (KF)

  + very simple, $-$ $\lambda$ too small: loss of target measurement

- Force a **unique interpretation** in case of a conflict!

  look for *smallest statistical distance*: $\min_i ||\nu_{k|k-1}^i||$

  $\Rightarrow$ Nearest-Neighbor filter (NN)
Problem: Growing Memory Disaster:

\[ m \text{ data, } N \text{ hypotheses } \rightarrow N^{m+1} \text{ continuations} \]

radical solution: mono-hypothesis approximation

- **gating**: Exclude competing data with \[ ||\nu_{k|k-1}^i|| \geq \lambda! \]

  \[ \text{KALMAN filter (KF)} \]

  + very simple, - \( \lambda \) too small: loss of target measurement

- Force a **unique interpretation** in case of a conflict!

  look for *smallest statistical distance*: \[ \min_i ||\nu_{k|k-1}^i|| \]

  \[ \text{Nearest-Neighbor filter (NN)} \]

  + one hypothesis, - hard decision, - not adaptive

- **global combining**: Merge all hypotheses!

  \[ \text{PDAF, JPDAF filter} \]

  + all data, + adaptive, - reduced applicability
**Moment Matching:** Approximate an arbitrary pdf $p(x)$ with $\mathbb{E}[x] = x$, $\mathbb{C}[x] = P$ by $p(x) \approx \mathcal{N}(x; x, P)$!

here especially: $p(x) = \sum_{H} p_{H} \mathcal{N}(x; x_{H}, P_{H})$ (normal mixtures)

$$x = \sum_{H} p_{H} x_{H}$$

$$P = \sum_{H} p_{H} \{ P_{H} + (x_{H} - x)(x_{H} - x)^{T} \}$$
PDAF Filter: formally analogous to Kalman Filter

Filtering (scan $k−1$):  
\[ p(x_{k−1}|Z^{k−1}) = \mathcal{N}(x_{k−1}; x_{k−1|k−1}, P_{k−1|k−1}) \]  
(→ initiation)

Prediction (scan $k$):  
\[ p(x_k|Z^{k−1}) \approx \mathcal{N}(x_k; x_{k|k−1}, P_{k|k−1}) \]  
(like Kalman)

Filtering (scan $k$):  
\[ p(x_k|Z^k) \approx \sum_{j=0}^{m_k} p^j_k \mathcal{N}(x_k; x_{j|k}, P_{j|k}) \]  
\[ \approx \mathcal{N}(x_k; x_{k|k}, P_{k|k}) \]
PDAF Filter: formally analogous to Kalman Filter

Filtering (scan \( k - 1 \)):  
\[ p(x_{k-1} | Z^{k-1}) = \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1}) \]  
(\( \rightarrow \) initiation)

Prediction (scan \( k \)):  
\[ p(x_k | Z^{k-1}) \approx \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \]  
(like Kalman)

Filtering (scan \( k \)):  
\[ p(x_k | Z^k) \approx \sum_{j=0}^{m_k} p^j_k \mathcal{N}(x_k; x^j_k|k, P^j_k|k) \]

\[ x^j_k|k = \begin{cases} x_k|k-1, & j = 0 \\ x_k|k-1 + W_k \nu^j_k, & j \neq 0 \end{cases} \]

\[ P^j_k|k = \begin{cases} P_{k|k-1}, & j = 0 \\ P_{k|k-1} - W_k S_k W_k^\top, & j \neq 0 \end{cases} \]

\[ \nu^j_k = z^j_k - H x_k, \quad W_k = P_{k|k-1} H^\top S_k^{-1}, \quad S_k = H P_{k|k-1} H^\top + R_k \]

\[ p^j_k = \frac{p^*_k}{\sum_j p^*_k}, \quad p^*_k = \begin{cases} (1 - P_D) \rho F_k, & j = 0 \\ \frac{P_D}{\sqrt{2\pi S_{k|k}}} e^{-\frac{1}{2} \nu^j_{k|k} S_{k|k} \nu^j_{k|k}}, & j \neq 0 \end{cases} \]
Second-order Approximation of the Mixture Density:

\[
\sum_{j=1}^{m_k} p_j^k \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k}^j, \mathbf{P}_{k|k}^j) \approx \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k}, \mathbf{P}_{k|k})
\]

mit:

\[
\mathbf{x}_{k|k} = \sum_{j=0}^{m_k} p_j^k \mathbf{x}_{k|k}^j
\]

\[
\mathbf{P}_{k|k} = \sum_{j=0}^{m_k} p_j^k \left( \mathbf{P}_{k|k}^j + (\mathbf{x}_{k|k}^j - \mathbf{x}_{k|k})(\mathbf{x}_{k|k}^j - \mathbf{x}_{k|k})^\top \right)
\]
\[
\begin{align*}
x_{k|k} &= \sum_{j=0}^{m_k} p^j_k x^j_{k|k}, \quad x^0_{k|k} = x_{k|k-1}, \quad x^j_{k|k} = x_{k|k-1} + W_k \nu^j_k \\
P_{k|k} &= \sum_{j=0}^{m_k} p^j_k (P^j_{k|k} + (x^j_{k|k} - x_{k|k})(x^j_{k|k} - x_{k|k})^\top)
\end{align*}
\]
\[ \mathbf{x}_{k|k} = \sum_{j=0}^{m_k} p^j_k \mathbf{x}^j_{k|k} \]

\[ = p^0_k \mathbf{x}_{k|k-1} + \sum_{j=1}^{m_k} p^j_k \left( \mathbf{x}_{k|k-1} + \mathbf{W}_k \mathbf{\nu}^j_k \right) \]

\[ \mathbf{P}_{k|k} = \sum_{j=0}^{m_k} p^j_k \left( \mathbf{P}^j_{k|k} + \left( \mathbf{x}^j_{k|k} - \mathbf{x}_{k|k} \right) \left( \mathbf{x}^j_{k|k} - \mathbf{x}_{k|k} \right)^\top \right) \]
\[ x_{k|k} = \sum_{j=0}^{m_k} p_j^k x_{k|k}^j \]

\[ = p_0^k x_{k|k-1} + \sum_{j=1}^{m_k} p_j^k (x_{k|k-1} + W_k \nu_j^k) \]

\[ = x_{k|k-1} \left( p_0^k + \sum_{j=1}^{m_k} p_j^k \right) + W_k \sum_{j=1}^{m_k} p_j^k \nu_j^k \]

\[ \text{mean!} \]

\[ P_{k|k} = \sum_{j=0}^{m_k} p_j^k (P_{k|k}^j + (x_{k|k} - x_{k|k}) (x_{k|k} - x_{k|k})^\top) \]
\begin{align*}
x_{k|k} &= \sum_{j=0}^{m_k} p^j_k x_{k|k}^j \\
&= p^0_k x_{k|k-1} + \sum_{j=1}^{m_k} p^j_k \left( x_{k|k-1} + W_k \nu^j_k \right) = x_{k|k-1} + W_k \nu_k \\

P_{k|k} &= \sum_{j=0}^{m_k} p^j_k \left( P_{k|k}^j + (x_{k|k}^j - x_{k|k})(x_{k|k}^j - x_{k|k})^T \right)
\end{align*}

\textbf{Combined Innovation:} \quad \nu_k = \sum_{j=1}^{m_k} p^j_k \nu^j_k
\[
x_{k|k} = \sum_{j=0}^{m_k} p^j_k x^j_{k|k} \\
= p^0_k x_{k|k-1} + \sum_{j=1}^{m_k} p^j_k (x_{k|k-1} + W_k \nu^j_k) = x_{k|k-1} + W_k \nu_k
\]

\[
P_{k|k} = \sum_{j=0}^{m_k} p^j_k (P^j_{k|k} + (x^j_{k|k} - x_{k|k})(x^j_{k|k} - x_{k|k})^\top), \quad P^0_{k|k} = P_{k|k-1}, \quad P^j_{k|k} = P_{k|k-1} - W_k S_k W_k^\top
\]

\[
= P_{k|k-1} - \sum_{j=1}^{m_k} p^j_k W_k S_k W_k^\top + \sum_{j=1}^{m_k} p^j_k W_k (\nu^j_k - \nu_k)(\nu^j_k - \nu_k)^\top W_k^\top
\]

**Combined Innovation:** \[
\nu_k = \sum_{j=1}^{m_k} p^j_k \nu^j_k
\]
\[ x_{k|k} = \sum_{j=0}^{m_k} p_k^j x_{k|k}^j \]

\[
= p_k^0 x_{k|k-1} + \sum_{j=1}^{m_k} p_k^j (x_{k|k-1} + W_k \nu_k^j) = x_{k|k-1} + W_k \nu_k
\]

\[
P_{k|k} = \sum_{j=0}^{m_k} p_k^j (P_{k|k}^j + (x_{k|k}^j - x_{k|k}) (x_{k|k}^j - x_{k|k})^\top)
\]

\[
= P_{k|k-1} - \sum_{j=1}^{m_k} p_k^j W_k S_k W_k^\top + \sum_{j=1}^{m_k} p_k^j W_k (\nu_k^j - \nu_k) (\nu_k^j - \nu_k)^\top W_k^\top
\]

\[
= P_{k|k-1} - (1 - p_k^0) W_k S_k W_k^\top + W_k \left[ \sum_{j=1}^{m_k} p_k^j \nu_k^j \nu_k^j \top - \nu_k \nu_k^\top \right] W_k^\top
\]

**Combined Innovation:** \[ \nu_k = \sum_{j=1}^{m_k} p_k^j \nu_k^j \]
PDAF Filter: formally analog to Kalman Filter

Filtering (scan $k-1$): \[ p(x_{k-1}|z^{k-1}) = \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1}) \] (→ initiation)

Prediction (scan $k$): \[ p(x_k|z^{k-1}) \approx \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \] (like Kalman)

Filtering (scan $k$): \[ p(x_k|z^k) \approx \sum_{j=0}^{m_k} p_j^k \mathcal{N}(x_k; x_{j|k}^k, P_{j|k}^k) \approx \mathcal{N}(x_k; x_{k|k}, P_{k|k}) \]

\[ \nu_k = \sum_{j=0}^{m_k} p_j^k \nu^j_k, \quad \nu^j_k = z^j_k - Hx_{k|k-1} \] \textit{combined} innovation

\[ W_k = P_{k|k-1}H^\top S_k^{-1}, \quad S_k = HP_{k|k-1}H^\top + R_k \] Kalman gain matrix

\[ p_j^k = p_j^k* / \sum_j p_j^k*, \quad p_j^k* = \left\{ \begin{array}{l} (1 - P_D) \rho_F \frac{R_k}{\sqrt{2\pi S_{\nu_k}}} e^{-\frac{1}{2} \nu_{\nu_k}^\top S_{\nu_k} \nu_{\nu_k}} \end{array} \right. \] weighting factors

\[ x_k = x_{k|k-1} + W_k \nu_k \] (Filtering Update: Kalman)

\[ P_k = P_{k|k-1} - (1-P_D^0) W_k S W_k^\top \] (Kalman part)

\[ + W_k \left\{ \sum_{j=0}^{m_k} p_j^k \nu_j^k \nu_{k}^\top - \nu_k \nu_k^\top \right\} W_k^\top \] (Spread of Innovations)
PDAF: Characteristic Properties

- filtering: processing of *combined innovation*
- *all data* $Z_k$ in the gate are considered
- $p_i$ data dependent! Update *not linear*
- missing measurement: $P_{k|k-1}$ with weight $p_0$
- “usual” Kalman covariance according to $(1 - p_0)$
- Spread *positively semidefinite*: larger covariance
- therefore: *data driven adaptivity*
- *non linear estimator*: data dependent error
- Performance prediction *only via simulations*

**Problem:** Multimodality is lost!
The qualitative shape of $p(x_k | \mathcal{Z}^k)$ is often much simpler than its correct representation: *a few pronounced modes*

**adaptive solution: nearly optimal approximation**
The qualitative shape of $p(x_k|Z^k)$ is often much simpler than its correct representation: *a few pronounced modes*

**adaptive solution: nearly optimal approximation**

- **individual gating:** Exclude *irrelevant data*! before continuing existing track hypotheses $H_{k-1}$
  
  $\rightarrow$ *limiting case:* KALMAN filter (KF)
The qualitative shape of \( p(x_k|Z^k) \) is often much simpler than its correct representation: _a few pronounced modes_

**adaptive solution: nearly optimal approximation**

- **individual gating:** Exclude _irrelevant data_!
  before continuing existing track hypotheses \( H_{k-1} \)
  → _limiting case:_ KALMAN filter (KF)

- **pruning:** Kill hypotheses of very _small weight_!
  after calculating the weights \( p_{H_k} \), before filtering
  → _limiting case:_ Nearest Neighbor filter (NN)
The qualitative shape of \( p(x_k | Z_k) \) is often much simpler than its correct representation: *a few pronounced modes*

**adaptive solution: nearly optimal approximation**

- **individual gating:** Exclude *irrelevant data!*  
  before continuing existing track hypotheses \( H_{k-1} \)  
  \( \rightarrow \) *limiting case:* KALMAN filter (KF)

- **pruning:** Kill hypotheses of very *small weight!*  
  after calculating the weights \( p_{H_k} \), before filtering  
  \( \rightarrow \) *limiting case:* Nearest Neighbor filter (NN)

- **local combining:** Merge *similar hypotheses!*  
  after the complete calculation of the pdfs  
  \( \rightarrow \) *limiting case:* PDAF (global combining)
Successive Local Combining

Partial sums of *similar* densities $\rightarrow$ moment matching:

$$\sum_{H_k \in H_k^*} p_{H_k} \mathcal{N}(x_k; x_{H_k}, P_{H_k}) \approx p_{H_k^*} \mathcal{N}(x_k; x_{H_k^*}, P_{H_k^*})$$

$H_k^* \subset H_k \rightarrow H_k^*: \text{effective hypothesis}$
Successive Local Combining

Partial sums of *similar* densities $\rightarrow$ moment matching:

$$
\sum_{H_k \in \mathcal{H}^k} p_{H_k} \mathcal{N}(x_k; x_{H_k}, P_{H_k}) \approx p_{H_k^*} \mathcal{N}(x_k; x_{H_k^*}, P_{H_k^*})
$$

$\mathcal{H}^k \subset \mathcal{H}^k \rightarrow H_k^*$: effective hypothesis

**similarity:**

$$
d(H_1, H_2) < \mu \quad \text{mit (z.B.)}:
$$

$$
d(H_1, H_2) = (x_{H_1} - x_{H_2})^\top (P_{H_1} + P_{H_2})^{-1} (x_{H_1} - x_{H_2})
$$

**Start:** Hypothesis of highest weight $H_1 \rightarrow$ search similar hypothesis $(p_H \triangleright_{\lambda}) \rightarrow$ merge: $(H_1, H) \triangleright H_1^* \rightarrow$ continue search $(p_H \triangleright_{\lambda}) \ldots
$$

$\rightarrow \textbf{restart}:$ hypothesis with next to highest weight $H_2 \rightarrow \ldots$
Successive Local Combining

Partial sums of similar densities → moment matching:

\[ \sum_{H_k \in \mathcal{H}^{k*}} p_{H_k} \mathcal{N}(x_k; x_{H_k}, P_{H_k}) \approx p_{H_k^*} \mathcal{N}(x_k; x_{H_k^*}, P_{H_k^*}) \]

\( \mathcal{H}^{k*} \subset \mathcal{H}^k \rightarrow H_k^*: \text{effective hypothesis} \)

similarity: \( d(H_1, H_2) < \mu \) mit (z.B.):

\[ d(H_1, H_2) = (x_{H_1} - x_{H_2})^\top (P_{H_1} + P_{H_2})^{-1} (x_{H_1} - x_{H_2}) \]

Start: Hypothesis of highest weight \( H_1 \) → search similar hypothesis \((p_H \downarrow \alpha) \) → merge: \((H_1, H) > H_1^* \) → continue search \((p_H \downarrow \alpha) \) . . . → restart: hypothesis with next to highest weight \( H_2 \) → . . .

- In many cases: good approximations → quasi-optimality
- PDAF, JPDAF: \( \mathcal{H}^{k*} = \mathcal{H}^k \) → limited applicability
- robustness → detail mostly irrelevant
Retrodiction for **GAUSSian Mixtures**

wanted: \[ p(x_l | Z^k) \leftarrow p(x_{l+1} | Z^k) \] for \( l < k \)

\[
p(x_l | Z^k) = \sum_{H_k} p(x_l, H_k | Z_k) = \sum_{H_k} p(x_{l+1} | H_k, Z^k) p(H_k | Z^k)
\]

no ambiguities! filtering!

Calculation of \( p(x_l | H_k, Z^k) \) as in case of \( P_D = 1, \rho_F = 0 \):

\[
p(x_l | H_k, Z^k) = \mathcal{N}(x_l; x_{H_k}(l|k), P_{H_k}(l|k))
\]

with parameters given by RAUCH-TUNG-STRIEBEL formulae:

\[
x_{H_k}(l|k) = x_{H_k}(l|l) + W_{H_k}(l|k) \left( x_{H_k}(l+1|k) - x_{H_k}(l+1|l) \right)
\]

\[
P_{H_k}(l|k) = P_{H_k}(l|l) + W_{H_k}(l|k) \left( P_{H_k}(l+1|k) - P_{H_k}(l+1|l) \right) W_{H_k}(l|k)^\top
\]

\[
\text{gain matrix: } W_{H_k}(l|k) = P_{H_k}(l|l) F_{l+1}^\top P_{H_k}(l+1|l)^{-1}
\]
Retrodiction of Hypotheses’ Weights

Consider approximation: neglect RTS step!

\[ p(x_l|H_k, Z^k) = \mathcal{N}(x_l; x_{H_k}(l|k), P_{H_k}(l|k)) \approx \mathcal{N}(x_l; x_{H_k}(l|l), P_{H_k}(l|l)) \]

\[ p(x_l|H_k, Z^k) \approx \sum_{H_l} p^*_H \mathcal{N}(x_l; x_{H_k}(l|l), P_{H_k}(l|l)) \]

with recursively defined weights:

\[ p^*_{H_k} = p_{H_k}, \quad p^*_H = \sum p^*_{H_{l+1}} \]

summation over all histories \( H_{l+1} \) with equal pre-histories!

- Strong sons strengthen weak fathers.
- Weak sons weaken even strong fathers.
- If all sons die, also the father must die.
Track Extraction: Initiation of the PDF Iteration

**extraction of target tracks:** detection on a higher level of abstraction

**start:** data sets \( Z_k = \{ z_{j, k}^m \}_{j=1}^m \) (sensor performance: \( P_D, \rho_F, R \))

**goal:** Detect a target trajectory in a time series: \( \mathcal{Z}^k = \{ Z_i \}_{i=1}^k \)

**at first simplifying assumptions:**

- The targets in the sensors’ field of view (FoV) are well-separated.
- The sensor data in the FoV in scan \( i \) are produced simultaneously.
Track Extraction: Initiation of the PDF Iteration

**extraction of target tracks:** detection on a higher level of abstraction

**start:** data sets $Z_k = \{z^{i}_{k}\}_{j=1}^{m_k}$ (sensor performance: $P_D, \rho_F, R$)

**goal:** Detect a target trajectory in a time series: $\mathcal{Z}^k = \{Z_i\}_{i=1}^k$

**at first simplifying assumptions:**
- The targets in the sensors’ field of view (FoV) are well-separated.
- The sensor data in the FoV in scan $i$ are produced simultaneously.

**decision between two competing hypotheses:**
- $h_1$: Besides false returns $\mathcal{Z}^k$ contains also target measurements.
- $h_0$: There is no target existing in the FoV; all data in $\mathcal{Z}^k$ are false.

**statistical decision errors:**
- $P_1 = \text{Prob}(\text{accept } h_1| h_1)$ analogous to the sensors’ $P_D$
- $P_0 = \text{Prob}(\text{accept } h_1| h_0)$ analogous to the sensors’ $P_F$
Practical Approach: Sequential Likelihood Ratio Test

**Goal:** Decide as fast as possible for given decision errors $P_0, P_1$!

Consider the ratio of the conditional probabilities $p(h_1 | Z^k), p(h_0 | Z^k)$ and the likelihood ratio $LR(k) = p(Z^k | h_1) / p(Z^k | h_0)$ as an intuitive decision function:

$$
\frac{p(h_1 | Z^k)}{p(h_0 | Z^k)} = \frac{p(Z^k | h_1)}{p(Z^k | h_0)} \frac{p(h_1)}{p(h_0)}
$$

a priori: $p(h_1) = p(h_0)$
Practical Approach: Sequential Likelihood Ratio Test

**Goal:** Decide as fast as possible for given decision errors $P_0, P_1$!

Consider the ratio of the conditional probabilities $p(h_1|Z^k)$, $p(h_0|Z^k)$ and the likelihood ratio $LR(k) = p(Z^k|h_1)/p(Z^k|h_0)$ as an intuitive decision function:

$$\frac{p(h_1|Z^k)}{p(h_0|Z^k)} = \frac{p(Z^k|h_1) p(h_1)}{p(Z^k|h_0) p(h_0)} \quad \text{a priori: } p(h_1) = p(h_0)$$

Starting from a time window with length $k = 1$, calculate the test function $LR(k)$ successively and compare it with two thresholds $A$, $B$:

If $LR(k) < A$, accept hypothesis $h_0$ (i.e. no target is existing)!

If $LR(k) > B$, accept hypothesis $h_1$ (i.e. target exists in FoV)!

If $A < LR(k) < B$, wait for new data $Z_{k+1}$, repeat with $LR(k + 1)$!
Iterative Calculation of the Likelihood Ratio

\[
\text{LR}(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} = \frac{\int dx_k p(Z_k, m_k, x_k, Z^{k-1}|h_1)}{p(Z_k, m_k, Z^{k-1}, h_0)}
\]
Iterative Calculation of the Likelihood Ratio

\[
\text{LR}(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} = \frac{\int dx_k \ p(Z_k, m_k, x_k, Z^{k-1}|h_1)}{p(Z_k, m_k, Z^{k-1}, h_0)} = \frac{\int dx_k \ p(Z_k, m_k|x_k) p(x_k|Z^{k-1}, h_1) p(Z^{k-1}|h_1)}{|\text{FoV}|-m_k \ p_F(m_k) p(Z^{k-1}|h_0)}
\]
Iterative Calculation of the Likelihood Ratio

\[
\text{LR}(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} = \frac{\int d\mathbf{x}_k p(Z_k, m_k, \mathbf{x}_k, Z^{k-1}|h_1)}{p(Z_k, m_k, Z^{k-1}, h_0)} = \frac{\int d\mathbf{x}_k p(Z_k, m_k|\mathbf{x}_k) p(\mathbf{x}_k|Z^{k-1}, h_1) p(Z^{k-1}|h_1)}{|\text{FoV}|^{-m_k} p_F(m_k) p(Z^{k-1}|h_0)}
\]

\[
= \frac{\int d\mathbf{x}_k p(Z_k, m_k|\mathbf{x}_k, h_1) p(\mathbf{x}_k|Z^{k-1}, h_1)}{|\text{FoV}|^{-m_k} p_F(m_k)} \text{LR}(k-1)
\]

basic idea: iterative calculation!
Iterative Calculation of the Likelihood Ratio

\[
\text{LR}(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} = \int dx_k p(Z_k, m_k, x_k, Z^{k-1}|h_1) p(Z_k, m_k, Z^{k-1}, h_0) = \int dx_k p(Z_k, m_k|x_k) p(x_k|Z^{k-1}, h_1) p(Z^{k-1}|h_1) \quad \text{LR}(k-1)
\]

basic idea: iterative calculation!

Let \( H_k = \{E_k, H_{k-1}\} \) be an interpretation history of the time series \( Z^k = \{Z_k, Z^{k-1}\} \).

- \( E_k = E^0_k \): target was not detected,
- \( E_k = E^j_k \): \( z^j_k \in Z_k \) is a target measurement.

\[
p(x_k|Z^{k-1}, h_1) = \sum_{H_{k-1}} p(x_k|H_{k-1}Z^{k-1}, h_1) p(H_{k-1}|Z^{k-1}, h_1)
\]

The standard MHT prediction!

\[
p(Z_k, m_k|x_k, h_1, h_1) = \sum_{E_k} p(Z_k, E_k|x_k, h_1)
\]

The standard MHT likelihood function!

The calculation of the likelihood ratio is just a by-product of Bayesian MHT tracking.
Iteration Formula for $LR(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)}$

**initiation:**
$k = 0, \ j_0 = 0, \ \lambda_{j_0} = 1$

**recursion:**
$LR(k+1) = \sum_{j_{k+1}} \lambda_{j_{k+1}} = \sum_{j_{k+1}=0}^{m_{k+1}} \sum_{j_k} \lambda_{j_{k+1}j_k} \lambda_{j_k}$

with:
$\lambda_{j_{k+1}j_k} = \begin{cases} 1 - P_D & \text{for } j_{k+1} = 0 \\ \frac{P_D}{\rho_F} \mathcal{N}(\nu_{j_{k+1}j_k}, S_{j_{k+1}j_k}) & \text{for } j_{k+1} \neq 0 \end{cases}$

**convenient notation:**
with $j_k = (j_k, \ldots, j_1)$ let
$\sum_{j_k} \lambda_{j_k} = \sum_{j_k=0}^{m_k} \cdots \sum_{j_1=0}^{m_1} \lambda_{j_k \ldots j_1}$
Iteration Formula for $LR(k) = p(Z^k|h_1)/p(Z^k|h_0)$

**initiation:**

$k = 0, \quad j_0 = 0, \quad \lambda_{j_0} = 1$

**recursion:**

$$LR(k + 1) = \sum_{j_{k+1}} \lambda_{j_{k+1}} = \sum_{j_{k+1}=0}^{m_{k+1}} \lambda_{j_{k+1}j_k} \lambda_{j_k}$$

with:

$$\lambda_{j_{k+1}j_k} = \begin{cases} 1 - P_D & \text{for } j_{k+1} = 0 \\ \frac{P_D}{\rho_F} N(\nu_{j_{k+1}j_k}, S_{j_{k+1}j_k}) & \text{for } j_{k+1} \neq 0 \end{cases}$$

**innovation:**

$$\nu_{j_{k+1}j_k} = z_{j_{k+1}} - H_{j_{k+1}} x_{j_{k+1}|k}$$

**innov. cov.:**

$$S_{j_{k+1}j_k} = H_{j_{k+1}} P_{j_{k+1}|k} H_{j_{k+1}}^\top + R_{j_{k+1}}$$

**state update:**

$$x_{j_{k+1}|k} = F_{j_{k+1}} x_j \quad \quad \quad \quad \quad \quad \quad x_j = x_{j_{k+1}|k-1} + W_{j_{k+1}} \nu_{j_{k+1}|k-1}$$

**covariances:**

$$P_{j_{k+1}|k} = F_{j_{k+1}} P_j F_{j_{k+1}}^\top + D_{j_{k+1}} \quad \quad P_j = P_{j_{k+1}|k-1} - W_{j_{k+1}} S_{j_{k+1}|k-1} W_{j_{k+1}}^\top$$
Iteration Formula for \( LR(k) = p(Z^k|h_1)/p(Z^k|h_0) \)

initiation: \( k = 0, \quad j_0 = 0, \quad \lambda_{j_0} = 1 \)

recursion: \( LR(k + 1) = \sum_{j_{k+1}} \lambda_{j_{k+1}} = \sum_{j_{k+1}=0}^{m_{k+1}} \sum_{j_k} \lambda_{j_{k+1}j_k} \lambda_{j_k} \)

with: \( \lambda_{j_{k+1}j_k} = \begin{cases} 1 - P_D & \text{for } j_{k+1} = 0 \\ \frac{P_D}{\rho_F} \mathcal{N}(\nu_{j_{k+1}j_k}, S_{j_{k+1}j_k}) & \text{for } j_{k+1} \neq 0 \end{cases} \)

innovation: \( \nu_{j_{k+1}j_k} = z_{j_{k+1}} - H_{j_{k+1}} x_{j_{k+1}} \)

innov. cov.: \( S_{j_{k+1}j_k} = H_{j_{k+1}} P_{j_{k+1}} H_{j_{k+1}}^\top + R_{j_{k+1}} \)

state update: \( x_{j_{k+1}|k} = F_{j_{k+1}} x_{j_k} \quad x_{j_k} = x_{j_k|k-1} + W_{j_k} \nu_{j_k|j_{k-1}} \)

covariances: \( P_{j_{k+1}|k} = F_{j_{k+1}} P_{j_k} F_{j_{k+1}}^\top + D_{j_{k+1}} \quad P_{j_k} = P_{j_k|k-1} - W_{j_k} S_{j_k} W_{j_k}^\top \)

Exercise Show that this recursion formulae for calculating the decision function is true.
Sequential Track Extraction: Discussion

- $LR(k)$ is given by a growing number of summands, each related to a particular interpretation history. The tuple $\{\lambda_j, x_j, P_j\}$ is called a sub-track.
Sequential Track Extraction: Discussion

- $LR(k)$ is given by a growing number of summands, each related to a particular interpretation history. The tuple $\{\lambda_{jk}, x_{jk}, P_{jk}\}$ is called a sub-track.

- For mitigating growing memory problems all approximations discussed for track maintenance can be used if they do not significantly affect $LR(k)$:
  - *individual gating*: Exclude data not likely to be associated.
  - *pruning*: Kill sub-tacks contributing marginally to the test function.
  - *local combining*: Merge similar sub tracks:

\[
\{\lambda_i, x_i, P_i\}_i \rightarrow \{\lambda, x, P\} \quad \text{with:} \quad \lambda = \sum_i \lambda_i,
\]
\[
x = \frac{1}{\lambda} \sum_i \lambda_i x_i, \quad P = \frac{1}{\lambda} \sum_i \lambda_i [P_i + (x_i - x)(\ldots)^\top].
\]
Sequential Track Extraction: Discussion

- $LR(k)$ is given by a growing number of summands, each related to a particular interpretation history. The tuple $\{\lambda_{j_k}, x_{j_k} P_{j_k}\}$ is called a sub-track.

- For mitigating growing memory problems all approximations discussed for track maintenance can be used if they do not significantly affect $LR(k)$:
  
  - **individual gating**: Exclude data not likely to be associated.
  - **pruning**: Kill sub-tacks contributing marginally to the test function.
  - **local combining**: Merge similar sub tracks:
    
    $$\{\lambda_i, x_i, P_i\} \rightarrow \{\lambda, x, P\} \quad \text{with:} \quad \lambda = \sum_i \lambda_i,$$
    $$x = \frac{1}{\lambda} \sum_i \lambda_i x_i, \quad P = \frac{1}{\lambda} \sum_i \lambda_i [P_i + (x_i - x)(\ldots)^\top].$$

- The LR test ends with a decision in favor of or against the hypotheses: $h_0$ (no target) or $h_1$ (target existing). Intuitive interpretation of the thresholds!
track extraction at $t_k$: Decide in favor of $h_1$!

initiation of pdf iteration (track maintenance):

Normalize coefficients $\lambda_{jk}$:

$$p_{jk} = \frac{\lambda_{jk}}{\sum_{j} \lambda_{jk}}$$

$$(\lambda_{jk}, x_{jk}, P_{jk}) \rightarrow p(x_k | Z^k) = \sum_{j_k} p_{jk} \mathcal{N}(x_k; x_{jk}, P_{jk})$$

Continue track extraction with the remaining sensor data!

sequential LR test for track monitoring:

After deciding in favor of $h_1$ reset $\text{LR}(0) = 1$! Calculate $\text{LR}(k)$ from $p(x_k | Z^k)$!

track confirmation: $\text{LR}(k) > \frac{P_1}{P_0}$: reset $\text{LR}(0) = 1$!

track deletion: $\text{LR}(k) < \frac{1-P_1}{1-P_0}$; ev. track re-initiation
DEMONSTRATION (simulated)

Exercise (voluntary)

Simulate a detection process with a given $P_D$, target measurements with a given $R$, a detection process with a given $P_D$ and realize the track extraction procedure.