Multiple Sensor Target Tracking: Basic Idea

Iterative updating of conditional probability densities!

kinematic target state $x_k$ at time $t_k$, accumulated sensor data $Z^k$

a priori knowledge: target dynamics models, sensor model

• prediction: $p(x_{k-1} | Z^{k-1})$ \[\xrightarrow{\text{dynamics model}}\] $p(x_k | Z^{k-1})$

• filtering: $p(x_k | Z^{k-1})$ \[\xrightarrow{\text{sensor model}}\] $p(x_k | Z^k)$

• retrodiction: $p(x_{l-1} | Z^k)$ \[\xleftarrow{\text{filtering output}}\] $p(x_l | Z^k)$

A first look at retrodiction today!
Recapitulation: An Important Data Fusion Algorithm

Kalman filter: \( x_k = (r_k^\top, \dot{r}_k^\top)^\top, \mathcal{Z}^k = \{z_k, \mathcal{Z}^{k-1}\} \)

**initiation:**
\[
p(x_0) = \mathcal{N}(x_0; x_{0|0}, P_{0|0}) , \quad \text{initial ignorance: } P_{0|0} \text{ 'large'}
\]

**prediction:**
\[
\mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1}) \xrightarrow{\text{dynamics model}} \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1})
\]
\[
x_{k|k-1} = F_{k|k-1} x_{k-1|k-1} \\
P_{k|k-1} = F_{k|k-1} P_{k-1|k-1} F_{k|k-1}^\top + D_{k|k-1}
\]

**filtering:**
\[
\mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \xrightarrow{\text{current measurement } z_k} \mathcal{N}(x_k; x_{k|k}, P_{k|k})
\]
\[
x_{k|k} = x_{k|k-1} + W_{k|k-1} \nu_{k|k-1} , \quad \nu_{k|k-1} = z_k - H_k x_{k|k-1} \\
P_{k|k} = P_{k|k-1} - W_{k|k-1} S_{k|k-1} W_{k|k-1}^\top , \quad S_{k|k-1} = H_k P_{k|k-1} H_k^\top + R_k \\
W_{k|k-1} = P_{k|k-1} H_k^\top S_{k|k-1}^{-1}
\]

‘KALMAN gain matrix’

A deeper look into the dynamics and sensor models necessary!
Consider a car moving on a mountain pass road modeled by:

\[
\mathbf{r}(t) = \begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} = \begin{pmatrix} vt \\ a_y \sin\left(\frac{4\pi v}{a_x} t\right) \\ a_z \sin\left(\frac{\pi v}{a_z} t\right) \end{pmatrix}
\]

\[
v = 20\text{ km/h}, \quad a_x = 10 \text{ km}, \quad a_y = a_z = 1 \text{ km}, \quad t \in [0, a_x/v].
\]

1. Plot the trajectory. Are the parameters reasonable? Try alternatives.
2. Calculate and plot the velocity and acceleration vectors:
   \[
   \mathbf{\dot{r}}(t) = \begin{pmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \end{pmatrix}, \quad \mathbf{\ddot{r}}(t) = -q \begin{pmatrix} \ddot{x}(t) \\ \ddot{y}(t) \\ \ddot{z}(t) \end{pmatrix}.
   \]
3. Calculate for each instance of time \( t \) the tangential vectors in \( \mathbf{r}(t) \):
   \[
   \mathbf{t}(t) = \frac{1}{|\mathbf{\dot{r}}(t)|} \mathbf{\dot{r}}(t).
   \]
4. Plot \(|\mathbf{\dot{r}}(t)|, |\mathbf{\ddot{r}}(t)|, \) and \( \mathbf{\ddot{r}}(t) \mathbf{t}(t) \) over the time interval.
5. Discuss the temporal behaviour based on the trajectory \( \mathbf{r}(t) \)!
Create your own sensor simulator!

Exercise 4.1

Simulate normally distributed radar measurements!

\[ \Delta T = 2 \text{ s}, \text{ 2 radars at } r_{s1}^{1,2} = (x_{s1}^{1,2}, y_{s1}^{1,2}, z_{s1}^{1,2})^T, \]
\[ x_{s1}^{1,2} = 0, \text{ 100 km}, \ y_{s1}^{1,2} = 100, 0 \text{ km}, \ z_{s1}^{1,2} = 10 \text{ km}. \]

State at time \( t_k = k \Delta T, k \in \mathbb{Z} \): \( x_k = (r_k^T, \dot{r}_k^T)^T \)

1. Simulate range and azimuth measurements of the target position \( r_k \) with a random number generator \( \text{normrnd}(0, 1) \) producing normally distributed zero-mean and unit-variance random numbers:

\[
\mathbf{z}_k^p = \begin{pmatrix} z_{k}^r \\ z_{k}^\varphi \end{pmatrix} = \begin{pmatrix} \sqrt{(x_k-x_s)^2+(y_k-y_s)^2+(z_k-z_s)^2-z_s^2} \\ \arctan\left(\frac{y_k-y_s}{x_k-x_s}\right) \end{pmatrix} + \begin{pmatrix} \sigma_r \text{ normrnd}(0,1) \\ \sigma_\varphi \text{ normrnd}(0,1) \end{pmatrix}
\]

with \( \sigma_r = 10 \text{ m}, \sigma_\varphi = 0.1^\circ \) denoting the standard deviations in range and azimuth. Assume that the radars are not able to measure the elevation angle (see discussion on the whiteboard!).

2. Transform the measurements in \( x-y \)-Cartesian coordinates \( z_k^T(r_k \cos z_k^\varphi, r_k \sin z_k^\varphi)^T + r_s \) and plot them over \( x-y \) projection of the true target trajectory! Play with sensor positions and measurement error standard deviations!
Recapitulation: Piecewise Constant White Acceleration

Consider state vectors: \( x_k = (r_k^\top, \dot{r}_k^\top)^\top \) (position, velocity)

For known \( x_{k-1} \) and without external influences we have with \( \Delta T_k = t_k - t_{k-1} \):

\[
x_k = \begin{pmatrix} I & \Delta T_k & I \\ O & I \\ I \\ \dot{r}_{k-1} \\ I \\ \end{pmatrix} \begin{pmatrix} r_{k-1} \\ \end{pmatrix} =: F_{k|k-1} x_{k-1}, \text{ see blackboard!}
\]

Assume during the interval \( \Delta T_k \) a constant acceleration \( a_k \) causing the state evolution:

\[
\begin{pmatrix} \frac{1}{2} \Delta T_k^2 & I \\ \Delta T_k & I \\ \end{pmatrix} a_k =: G_k a_k, \text{ linear transform!}
\]

Let \( a_k \) be a Gaussian RV with pdf: \( p(a_k) = \mathcal{N}(a_k; \mu, \Sigma_k^2 I) \), we therefore have:

\[
p(G_k a_k) = \mathcal{N}(G_k a_k; \mu, \Sigma_k^2 G_k G_k^\top).
\]
Therefore: 
\[ p(x_k|x_{k-1}) = \mathcal{N}(x_k; F_{k|k-1}x_{k-1}, D_{k|k-1}) \]

with

\[
F_{k|k-1} = \begin{pmatrix} I & \Delta T_k I \\ O & I \end{pmatrix}, \quad D_{k|k-1} = \sum_k^2 \begin{pmatrix} \frac{1}{4} \Delta T_k^4 I & \frac{1}{2} \Delta T_k^3 I \\ \frac{1}{2} \Delta T_k^3 I & \Delta T_k^2 I \end{pmatrix}
\]
Recapitulation Range, Azimuth Measurements

- measurements in polar coordinates:
  \[ z_k = (r_k, \varphi_k)^\top, \text{ measurement error: } R = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_{\varphi}^2 \end{pmatrix}, \]  \( r, \varphi \) independent

- in Cartesian coord.: expand around truth position:
  \[ t[z_k] = r_k \begin{pmatrix} \cos \varphi_k \\ \sin \varphi_k \end{pmatrix} \approx t[r_k|k-1] + T(z_k - r_k) \]

  \[ T = \begin{pmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & r \end{pmatrix} \]

  \( T \) = rotation \( D_\varphi \) \quad \text{dilation } S_r

- Cartesian error covariance (time dependent):

  \[ TRT^\top = D_\varphi S_r R S_r D_\varphi^\top = D_\varphi \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & (r \sigma_{\varphi})^2 \end{pmatrix} D_\varphi^\top \]

- sensor fusion: sensor-to-target-geometry enters into \( TRT^\top \)
initiation: \[ p(x_0) = \mathcal{N}(x_0; x_{0|0}, P_{0|0}) \]

initial ignorance: \( P_{0|0} \) ‘large’

prediction: \[ \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1}) \xrightarrow{\text{dynamics model}} \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \]

\[ x_{k|k-1} = F_{k|k-1} x_{k-1|k-1} \]

\[ P_{k|k-1} = F_{k|k-1} P_{k-1|k-1} F_{k|k-1}^\top + D_{k|k-1} \]

filtering: \[ \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \xrightarrow{\text{current measurement} z_k} \mathcal{N}(x_k; x_{k|k}, P_{k|k}) \]

\[ x_{k|k} = x_{k|k-1} + W_{k|k-1} \nu_{k|k-1} \]

\[ P_{k|k} = P_{k|k-1} - W_{k|k-1} S_{k|k-1} W_{k|k-1}^\top \]

\[ W_{k|k-1} = P_{k|k-1} H_k^\top S_{k|k-1}^{-1} \]

\[ \nu_{k|k-1} = z_k - H_k x_{k|k-1} \]

\[ S_{k|k-1} = H_k P_{k|k-1} H_k^\top + R_k \]

\[ ‘KALMAN gain matrix’ \]

In your sensor simulator, chose a sensor at position \( r_s \) that produces \( x-y \) measurements \( z_k \) of the Cartesian target \( x-y \) positions \( Hx_k \) from your ground truth generator using the measurement matrix \( H \):

\[ Hx_k = \begin{pmatrix} 1,0,0,0,0,0 \\ 0,1,0,0,0,0 \end{pmatrix} x_k \]

Exercise 4.2

Calculate for each measurement the measurement error covariance matrix \( R_k \) based on the true target position. Program your first Kalman filter initiated by the first measurement and reasonably chosen covariance matrices \( P_{1|1} \). What is reasonable? Visualize nicely and compare with the truth and the measurements.
Recapitulation: Create a single effective measurement by preprocessing of the individual measurements!

\[
z_k = R_k \sum_{s=1}^{S_k} \left( R_k^s \right)^{-1} z_k^s
\]

weighted arithmetic mean of measurements

\[
R_k = \left( \sum_{s=1}^{S_k} \left( R_k^s \right)^{-1} \right)^{-1}
\]

harmonic mean of measurement covariances

A typical structure for fusion equations!

With measurement specific measurement error covariances, your Kalman filter already is a multiple sensor fusion algorithms. Use two radar sensors, fuse the measurements, feed them into the Kalman filter and discuss the result!

Exercise 4.3
More general: measurement process

- linear measurement equation:
  \[ z_k = H_k x_k + u_k, \quad p(u_k) = \mathcal{N}(u_k; 0, R_k) \]
  - to be measured: linear functions of the object state
  - measurement error: biasfree, Gaussian distrib. independent for different \( t_k \)
  - \( y_k = z_k - H_k x_k \) has the pdf: \( p(y_k) = p(u_k) \)

- Approach for the requested pdf (‘likelihood fkt.):
  \[ p(z_k | x_k) = \mathcal{N}(z_k; H_k x_k, R_k) \]

- Example: position measurement
  \[ H_k = (I, O, O), \quad H_k x_k = r_k \]
  \( R_k \): measurement error covariance matrix
  possibly depending on the sensor-to-target geometry
Retrodiction: How to calculate the pdf $p(x_l | Z^k)$?

Consider the past: $l < k$!

an observation:

$$p(x_l | Z^k) = \int d\mathbf{x}_{l+1} \, p(x_l, x_{l+1} | Z^k)$$
Retrodiction: How to calculate the pdf $p(x_l | Z^k)$?

Consider the past: $l < k$!

an observation:

$$
p(x_l | Z^k) = \int dx_{l+1} p(x_l, x_{l+1} | Z^k) = \int dx_{l+1} p(x_l | x_{l+1}, Z^k) \underbrace{p(x_{l+1} | Z^k)}_{\text{retrodiction: } t_{l+1}}
$$
Retrodiction: How to calculate the pdf $p(x_l|z^k)$?

Consider the past: $l < k$!

an observation:

$$
p(x_l|z^k) = \int dx_{l+1} p(x_l, x_{l+1}|z^k) = \int dx_{l+1} p(x_l|x_{l+1}, z^k) p(x_{l+1}|z^k)
$$

$$
p(x_l|x_{l+1}, z^k) = \frac{p(Z_k, \ldots, Z_{l+1}|x_{l+1}, x_l, z^l) p(x_l|x_{l+1}, z^l)}{\int dx_l p(Z_k, \ldots, Z_{l+1}|x_{l+1}, x_l, z^l) p(x_l|x_{l+1}, z^l)} = p(x_l|x_{l+1}, z^l)
$$
Retrodiction: How to calculate the pdf $p(x_l | \mathcal{Z}^k)$?

Consider the past: $l < k$!

an observation:

$$p(x_l | \mathcal{Z}^k) = \int dx_{l+1} \ p(x_l, x_{l+1} | \mathcal{Z}^k) = \int dx_{l+1} \ p(x_l | x_{l+1}, \mathcal{Z}^k) \underbrace{p(x_{l+1} | \mathcal{Z}^k)}_{\text{retrodiction: } t_{l+1}}$$

$$p(x_l | x_{l+1}, \mathcal{Z}^k) = p(x_l | x_{l+1}, \mathcal{Z}^l) = \frac{p(x_{l+1} | x_l) \ p(x_l | \mathcal{Z}^l)}{\int d x_l \ p(x_{l+1} | x_l) \ p(x_l | \mathcal{Z}^l)}$$

$$= \frac{\ p(x_{l+1} | x_l) \ p(x_l | \mathcal{Z}^l)}{\ p(x_{l+1} | x_l) \ p(x_l | \mathcal{Z}^l)}$$
Retrodiction: How to calculate the pdf $p(x_l|Z^k)$?

Consider the past: $l < k$!

**an observation:**

$$p(x_l|Z^k) = \int dx_{l+1} p(x_l, x_{l+1}|Z^k) = \int dx_{l+1} \frac{p(x_{l+1}|x_l) p(x_l|Z^l)}{\int dx_l \frac{p(x_{l+1}|x_l) p(x_l|Z^l)}{p(x_l|Z^l)}} \frac{p(x_{l+1}|Z^k)}{\text{retrodiction: } t_{l+1}}$$

- $p(x_{l+1}|Z^k)$ retrodiction: last iteration step
- $p(x_k|x_{k-1})$ dynamic object behavior
- $p(x_l|Z^l)$ filtering at the time considered
- **GAUSSians, GAUSSian mixtures:** Exploit product formula!
- **linear GAUSSian likelihood/dynamics:** Rauch-Tung-Striebel smoothing
Exercise 4.4  Derive the *Rauch-Tung-Striebel* formulae

by using the Kalman filter assumptions

and the product formula (twice)!

---

**retrodiction:**  \( \mathcal{N}(x_{l|k}; x_{l|k}, P_{l|k}) \xleftarrow{\text{filtering, prediction, dynamics model}} \mathcal{N}(x_{l+1|k}; x_{l+1|k}, P_{l+1|k}) \)

\[
\begin{align*}
    x_{l|k} &= x_{l|l} + W_{l|l+1}(x_{l+1|k} - x_{l+1|l}), \\
    P_{l|k} &= P_{l|l} + W_{l|l+1}(P_{l+1|k} - P_{l+1|l})W_{l|l+1}^\top
\end{align*}
\]
Kalman filter: linear Gaussian likelihood/dynamics, $x_k = (r_k^T, \dot{r}_k^T, \ddot{r}_k^T)^T$, $Z^k = \{z_k, Z^{k-1}\}$

**initiation:**

$$p(x_0) = \mathcal{N}(x_0; x_{0|0}, P_{0|0})$$

**initial ignorance:**

$P_{0|0}$ 'large'

**prediction:**

$$\mathcal{N}(x_{k-1}^k; x_{k-1|k-1}, P_{k-1|k-1}) \xrightarrow{\text{dynamics model}} \mathcal{N}(x_{k}^k; x_{k|k-1}, P_{k|k-1})$$

$$x_{k|k-1} = F_{k|k-1} x_{k-1|k-1}$$

$$P_{k|k-1} = F_{k|k-1} P_{k-1|k-1} F_{k|k-1}^T + D_{k|k-1}$$

**filtering:**

$$\mathcal{N}(x_{k}^k; x_{k|k-1}, P_{k|k-1}) \xrightarrow{\text{current measurement } z_k} \mathcal{N}(x_{k}^k; x_{k|k}, P_{k|k})$$

$$x_{k|k} = x_{k|k-1} + W_{k|k-1} \nu_{k|k-1}, \quad \nu_{k|k-1} = z_k - H_k x_{k|k-1}$$

$$P_{k|k} = P_{k|k-1} - W_{k|k-1} S_{k|k-1} W_{k|k-1}^T, \quad S_{k|k-1} = H_k P_{k|k-1} H_k^T + R_k$$

$W_{k|k-1} = P_{k|k-1} H_k S_{k|k-1}^{-1}$

‘KALMAN gain matrix’

**retrodiction:**

$$\mathcal{N}(x_{l}^k; x_{l|k}, P_{l|k}) \xleftarrow{\text{filtering, prediction}} \mathcal{N}(x_{l+1}^k; x_{l+1|k}, P_{l+1|k})$$

$$x_{l|k} = x_{l|l} + W_{l|l+1}(x_{l+1|k} - x_{l+1|l}), \quad W_{l|l+1} = P_{l|l} F_{l+1|l}^T P_{l+1|l}^{-1}$$

$$P_{l|k} = P_{l|l} + W_{l|l+1}(P_{l+1|k} - P_{l+1|l}) W_{l|l+1}^T$$

**Exercise 4.5** Implement the Rauch-Tung-Striebel formulae in your simulator (in the course of the semester ...)!
Continuous Time Retrodiction for $t_l < t_{l+\theta} < t_{l+1}$ with $0 < \theta < 1$

Interpolate between $p(x_l|\mathcal{Z}^k)$ and $p(x_{l+1}|\mathcal{Z}^k)$ based on the evolution model:

$$p(x_{l+\theta}|\mathcal{Z}^k) = \int dx_{l+1} p(x_{l+\theta}, x_{l+1}|\mathcal{Z}^k)$$

$$= \int dx_{l+1} p(x_{l+\theta}|x_{l+1}, \mathcal{Z}^k) p(x_{l+1}|\mathcal{Z}^k)$$
Continuous Time Retrodiction for $t_l < t_{l+\theta} < t_{l+1}$ with $0 < \theta < 1$

Interpolate between $p(x_l|Z^k)$ and $p(x_{l+1}|Z^k)$ based on the evolution model:

$$p(x_{l+\theta}|Z^k) = \int dx_{l+1} p(x_{l+\theta}, x_{l+1}|Z^k)$$

$$= \int dx_{l+1} p(x_{l+\theta}|x_{l+1}, Z^k) p(x_{l+1}|Z^k)$$

where:

$$p(x_{l+\theta}|x_{l+1}, Z^k) = \frac{p(x_{l+1}|x_{l+\theta}) p(x_{l+\theta}|Z^l)}{\int dx_{l+\theta} p(x_{l+1}|x_{l+\theta}) p(x_{l+\theta}|Z^l)}$$

with:

$$p(x_{l+1}|x_{l+\theta}) = \mathcal{N}(x_{l+1}; F_{l+1|l+\theta}x_{l+\theta}, D_{l+1|l+\theta})$$

$$p(x_{l+\theta}|Z^l) = \int dx_l p(x_{l+\theta}|x_l) p(x_l|Z^l)$$

$$p(x_{l+1}|Z^l) = \int dx_{l+\theta} p(x_{l+1}|x_{l+\theta}) p(x_{l+\theta}|Z^l)$$

$$= \mathcal{N}(x_{l+1}; x_{l+1}|l, P_{l+1|l})$$
Continuous Time Retrodiction for $t_l < t_l + \theta < t_{l+1}$ with $0 < \theta < 1$

Interpolate between $p(x_l|Z^k)$ and $p(x_{l+1}|Z^k)$ based on the evolution model:

$$p(x_{l+\theta}|Z^k) = \int dx_{l+1} p(x_{l+\theta}, x_{l+1}|Z^k)$$
$$= \int dx_{l+1} p(x_{l+\theta}|x_{l+1}, Z^k) p(x_{l+1}|Z^k)$$

where:

$$p(x_{l+\theta}|x_{l+1}, Z^k) = \frac{p(x_{l+1}|x_{l+\theta}) p(x_{l+\theta}|Z^l)}{\int dx_{l+\theta} p(x_{l+1}|x_{l+\theta}) p(x_{l+\theta}|Z^l)}$$

with:

$$p(x_{l+1}|x_{l+\theta}) = \mathcal{N}(x_{l+1}; F_{l+1|l+\theta} x_{l+\theta}, D_{l+1|l+\theta})$$

$$p(x_{l+\theta}|Z^l) = \int dx_l p(x_{l+\theta}|x_l) p(x_l|Z^l)$$

$$p(x_{l+1}|Z^l) = \int dx_{l+\theta} p(x_{l+1}|x_{l+\theta}) p(x_{l+\theta}|Z^l)$$
$$= \mathcal{N}(x_{l+1}; x_{l+1|l}, P_{l+1|l})$$

Looks like a Kalman filtering update!
\[ p(x_{l+\theta} | x_{l+1}, Z^k) \propto p(x_{l+1} | x_{l+\theta}) \, p(x_{l+\theta} | Z^l) \]  
Looks like filtering!

\[ p(x_{l+\theta} | Z^k) = \int dx_{l+1} \, p(x_{l+\theta} | x_{l+1}, Z^k) \, p(x_{l+1} | Z^k) \]  
Looks like prediction!
\[ p(x_{l+\theta}|x_{l+1}, Z^k) \propto p(x_{l+1}|x_{l+\theta}) \, p(x_{l+\theta}|Z^l) \]

Looks like filtering!

\[ = \mathcal{N}(x_{l+\theta}; a_{l+\theta|l+1}, \Delta_{l+\theta|l+1}) \]

\[ a_{l+\theta|l+1} = x_{l+\theta|l} + \Phi_{l+\theta|l+1}(x_{l+1} - F_{l+1|l+\theta} x_{l+\theta|l}) \]

\[ = x_{l+\theta|l} - \Phi_{l+\theta|l+1} x_{l+1|l} + \Phi_{l+\theta|l+1} x_{l+1} \]

\[ \Delta_{l+\theta|l+1} = P_{l+\theta|l} - \Phi_{l+\theta|l+1} P_{l+1|l} \Phi_{l+\theta|l+1}^{\top} \]

\[ \Phi_{l+\theta|l+1} = P_{l+\theta|l} F_{l+1|l+\theta}^{\top} P_{l+1|l}^{-1} \]

\[ P_{l+1|l} = F_{l+1|l+\theta} P_{l+\theta|l} F_{l+1|l+\theta}^{\top} + D_{l+1|l+\theta}. \]

\[ p(x_{l+\theta}|Z^k) = \int dx_{l+1} \, p(x_{l+\theta}|x_{l+1}, Z^k) \, p(x_{l+1}|Z^k) \]

Looks like prediction!
\[
p(x_{l+\theta}|x_{l+1}, Z^k) \propto p(x_{l+1}|x_{l+\theta}) p(x_{l+\theta}|Z^l)
\]

Looks like filtering!

\[
= \mathcal{N}(x_{l+\theta}; a_{l+\theta|l+1}, \Delta_{l+\theta|l+1})
\]

\[
= \mathcal{N}(b_{l+\theta|l+1}; \Phi_{l+\theta|l+1} x_{l+1}, \Delta_{l+\theta|l+1})
\]

\[
a_{l+\theta|l+1} = x_{l+\theta|l} + \Phi_{l+\theta|l+1} (x_{l+1} - F_{l+1|l+\theta} x_{l+\theta|l})
\]

\[
= x_{l+\theta|l} - \Phi_{l+\theta|l+1} x_{l+1} x_{l+\theta|l} + \Phi_{l+\theta|l+1} x_{l+1}
\]

\[
b_{l+\theta|l+1} = x_{l+\theta} - x_{l+\theta|l} + \Phi_{l+\theta|l+1} x_{l+1} x_{l+\theta|l}
\]

\[
\Delta_{l+\theta|l+1} = P_{l+\theta|l} - \Phi_{l+\theta|l+1} P_{l+1|l} \Phi_{l+\theta|l+1}^T
\]

\[
\Phi_{l+\theta|l+1} = P_{l+\theta|l} F_{l+1|l+\theta}^T P_{l+1|l}^{-1} P_{l+1|l}
\]

\[
P_{l+1|l} = F_{l+1|l+\theta} P_{l+\theta|l} F_{l+1|l+\theta}^T + D_{l+1|l+\theta}.
\]

\[
p(x_{l+\theta}|Z^k) = \int dx_{l+1} p(x_{l+\theta}|x_{l+1}, Z^k) p(x_{l+1}|Z^k)
\]

Looks like prediction!
\[
p(x_{l+\theta}|x_{l+1}, Z^k) \propto p(x_{l+1}|x_{l+\theta}) \cdot p(x_{l+\theta}|Z^l)
\]

Looks like filtering!

\[
= \mathcal{N}(x_{l+\theta}; a_{l+\theta|l+1}, \Delta_{l+\theta|l+1})
\]

\[
= \mathcal{N}(b_{l+\theta|l+1}; \Phi_{l+\theta|l+1}x_{l+1}, \Delta_{l+\theta|l+1})
\]

\[
a_{l+\theta|l+1} = x_{l+\theta|l} + \Phi_{l+\theta|l+1}(x_{l+1} - F_{l+1|l+\theta}x_{l+\theta|l})
\]

\[
= x_{l+\theta|l} - \Phi_{l+\theta|l+1}x_{l+1} + \Phi_{l+\theta|l+1}x_{l+1}
\]

\[
b_{l+\theta|l+1} = x_{l+\theta} - x_{l+\theta|l} + \Phi_{l+\theta|l+1}x_{l+1}
\]

\[
\Delta_{l+\theta|l+1} = P_{l+\theta|l} - \Phi_{l+\theta|l+1}P_{l+1|\phi}\Phi_{l+\theta|l+1}
\]

\[
\Phi_{l+\theta|l+1} = P_{l+\theta|l}F_{l+1|l+\theta}P_{l+1|l}^{-1}
\]

\[
P_{l+1|l} = F_{l+1|l+\theta}P_{l+\theta|l}F_{l+1|l+\theta}P_{l+1|l}^{-1} + D_{l+1|l+\theta}.
\]

\[
p(x_{l+\theta}|Z^k) = \int dx_{l+1} \cdot p(x_{l+\theta}|x_{l+1}, Z^k) \cdot p(x_{l+1}|Z^k)
\]

Looks like prediction!

\[
= \mathcal{N}(x_{l+\theta}; x_{l+\theta|k}, x_{l+\theta|k})
\]

\[
x_{l+\theta|k} = x_{l+\theta|l} + \Phi_{l+\theta|l+1}(x_{l+1|k} - x_{l+1|l})
\]

\[
P_{l+\theta|k} = x_{l+\theta|l} + \Phi_{l+\theta|l+1}(P_{l+1|k} - P_{l+1|l})\Phi_{l+\theta|l+1}
Kalman filter: linear Gaussian likelihood/dynamics, \( x_k = (r_k^\top, \dot{r}_k^\top, \ddot{r}_k^\top)^\top, \mathcal{Z}^k = \{z_k, \mathcal{Z}^{k-1}\} \)

**initiation:**
\[
p(x_0) = \mathcal{N}(x_0; x_{0|0}, P_{0|0}), \quad \text{initial ignorance: } P_{0|0} \text{ 'large'}
\]

**prediction:**
\[
\mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1}) \xrightarrow{\text{dynamics model}} \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \]
\[
x_{k|k-1} = F_{k|k-1} x_{k-1|k-1} \\
P_{k|k-1} = F_{k|k-1} P_{k-1|k-1} F_{k|k-1}^\top + D_{k|k-1}
\]

**filtering:**
\[
\mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \xrightarrow{\text{current measurement } z_k} \mathcal{N}(x_k; x_k|k, P_k|k) \\
x_k|k = x_{k|k-1} + W_{k|k-1} \nu_{k|k-1}, \quad \nu_{k|k-1} = z_k - H_k x_{k|k-1} \\
P_k|k = P_{k|k-1} - W_{k|k-1} S_{k|k-1} W_{k|k-1}^\top, \quad S_{k|k-1} = H_k^\top P_{k|k-1} H_k + R_k \\
W_{k|k-1} = P_{k|k-1} H_k^\top S_{k|k-1}^{-1}
\]

‘KALMAN gain matrix’

**retrodiction:**
\[
\mathcal{N}(x_l; x_{l|k}, P_l|k) \xleftarrow{\text{filtering, prediction}} \mathcal{N}(x_{l+1}; x_{l+1|k}, P_{l+1|k}) \xrightarrow{\text{dynamics model}} \mathcal{N}(x_l; x_{l|l+1}, P_l|l+1)
\]
\[
x_{l|k} = x_{l|l} + W_{l|l+1}(x_{l+1|k} - x_{l+1|l}), \quad W_{l|l+1} = P_l|l F_{l+1|l}^\top P_{l+1|l}^{-1}
\\
P_{l|k} = P_{l|l} + W_{l|l+1}(P_{l+1|k} - P_{l+1|l}) W_{l|l+1}^\top
\]