Advanced Sensor Data Fusion in Distributed Systems

SS 2019
Felix Govaers,
Dr. rer. nat., Dipl.-Math.
STOCHASTIC DIFFERENTIAL EQUATION (SDE)
The \( \hat{\text{Ito}} \) Integral

Consider the discretization of the time with step size \( \Delta t \) such that \( t_0 < t_1 < \ldots < t_k \). Then:

\[
x_{i+1} = x_i + \Delta x_i, \quad \text{where} \quad \Delta x_i = x_{i+1} - x_i.
\]

The \( \hat{\text{Ito}} \) integral is defined as taking the limit \( \Delta t \to 0 \):

\[
x_k = \int_{t_0}^{t_k} dx_t
\]

\[
:= \lim_{\Delta t \to 0} \sum_{i=0}^{k-1} x_i + \Delta x_i.
\]
Ito Process

The stochastic process $x_t$ is called an Ito process, if its integral formulation is divided into a deterministic and a stochastic part:

$$x_k = x_0 + \int_{t_0}^{t_k} dt f(x_t) + \int_{t_0}^{t_k} dw_t B(x_t).$$

- Initial value
- Deterministic part
- Stochastic part
Stochastic Differential Equation (SDE)

The SDE of an \(\text{\textit{\`{I}to}}\) process

\[
x_k = x_0 + \int_{t_0}^{t_k} \, dt \, f(x_t) + \int_{t_0}^{t_k} \, dw_t \, B(x_t).
\]

is given by taking the limit \(t_k \to t_0\). This yields the infinitesimal increment of \(x\). We write:

\[
dx_k = f(x_k) \, dt + B(x_k) \, dw_k
\]
Example 1: Constant Velocity

Consider the state space \( \mathbf{x} = (x, y, x', y')^T \).

Thus, the incremental differential is given by \( d\mathbf{x} = (x', y', 0, 0)^T \).

We obtain:

\[
\mathbf{f(x)} = \mathbf{Ax}
\]

\[
\mathbf{A} = \begin{pmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}
\]
Example 2: Coordinated Turn

Consider a constant turn acceleration given by the transition model:

\[ f(x) = Ax \]

\[
A = \begin{pmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & -\omega \\
0 & 0 & \omega & 0
\end{pmatrix}
\]
Fokker-Planck Equation

Consider the \( \dot{x}_k \) process described by the SDE:

\[
\dot{x}_k = f(x_k) \, dt + B(x_k) \, dw_k 
\]

If we assume a multivariate Brownian motion for the stochastic increments \( dw \), such that:

\[
w_{k+1} = w_k \pm he_j
\]

Then, we obtain a generalized multivariate Bernoulli process where the pdf evolves according to the following differential equation:

\[
\partial_t p(x_k) = \left( - \sum_{i=1}^{n} \partial_{x_i} [f(x_k)]_i + \frac{1}{2} \sum_{i,j=1}^{n} \partial_{x_i} \partial_{x_j} [B(x_k)B(x_k)\top]_{ij} \right) p(x_k).
\]

Fokker-Planck Equation (FPE)
Statistical Models for

SENSORS
Linear Gaussian Case

- The measurement $z$ at time $k$ is modeled as a RV.
- Relationship to state is described by the
  
  "measurement equation":

  \[ z_k = H_k x_k + v_k \]

  where:
  - $H$ is the measurement function
  - $v$ is the measurement noise.
Ideal Sensor for Tracking

An ideal sensor is given if

- $\mathbf{H}$ is linear.
- $\mathbf{v}$ is zero-mean and Gaussian distributed, that is
  - $E[\mathbf{v}_k] = \mathbf{0}$
  - $p(\mathbf{v}_k) = \mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$
  - in short: $\mathbf{v}_k \sim \mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$

$$\Leftrightarrow \mathbf{z}_k \sim \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k \mathbf{x}_k, \mathbf{R}_k) = p(\mathbf{z}_k|\mathbf{x}_k)$$

Why is this sensor ideal?
If transition model is also linear and Gaussian, an optimal closed solution can be given -> Kalman filter
Example 1

Assume the state is given by

\[
x_k = \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix}
\]

then, the measurement equation for a position sensor (GPS e.g.) is

\[
z_k = H_k(x_k) + v_k
\]

\[
= \begin{pmatrix} x \\ y \end{pmatrix} + \text{noise}
\]

therefore

\[
H_k = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}
\]
TIME DISCRETE ÎTO PROCESS
Linear Îto Process

In the following, we will study the particular linear Îto process:

\[ dx_k = Ax_k \, dt + B \, dw_k, \]

Considering only the deterministic part

\[ \frac{d}{dt} x_t = Ax_t, \]

we obtain a unique time discrete solution with initial value \( x_0 \):

\[ x_k = \exp \{ \Delta_k|0A \} x_0, \]

\[ \exp \{ \Delta_k|0A \} = e^{\Delta_k|0} \sum_{i=0}^{\infty} A^i \]

\[ =: F_{k|0}. \]

Where:

\[ \Delta_k|0 = t_k - t_0 \]
### Time Discrete Motion Model

The general solution for the transition matrix \( F \) is given by

\[
F_{l|0} = \exp(A(l dt)) = \sum_{m=0}^{\infty} \frac{A^m(l dt)^m}{m!}.
\]

Then, the time discrete difference equation is

\[
x_{l+1} = F_{l+1|l} x_l + w_{l+1|l},
\]

\[
w_{l+1|l} = \int_{t_l}^{t_{l+1}} BF_{l+1|\tau} dw_{\tau}
\]
Statistics of Time Discrete Noise

We have:

\[
E\left[ w_{l+1|l} \right] = E \left[ \int_{t_l}^{t_{l+1}} BF_{l+1|\tau} d\omega_\tau \right] \\
= E \left[ \lim_{m \to \infty} \sum_{j=1}^{m} BF_{l+1|\tau_j}(w_{\tau_j} - w_{\tau_{j-1}}) \right] \\
= \lim_{m \to \infty} B \sum_{j=1}^{m} F_{l+1|\tau_j} E \left[ (w_{\tau_j} - w_{\tau_{j-1}}) \right] \\
= 0
\]

\[
E\left[ w_{l+1|l} w_{l+1|l}^\top \right] = E \left[ (\int_{t_l}^{t_{l+1}} BF_{l+1|\tau} d\omega_\tau) \left( \int_{t_l}^{t_{l+1}} BF_{l+1|\tau} d\omega_\tau \right)^\top \right] \\
= B^2 \lim_{m \to \infty} \sum_{j=1}^{m} F_{l+1|\tau_j} E \left[ (w_{\tau_j} - w_{\tau_{j-1}})(w_{\tau_j} - w_{\tau_{j-1}})^\top \right] F_{l+1|\tau_j}^\top \\
= B^2 \lim_{m \to \infty} \sum_{j=1}^{m} F_{l+1|\tau_j} F_{l+1|\tau_j}^\top \cdot \frac{1}{m} \\
= B^2 \int_{t_l}^{t_{l+1}} d\tau F_{l+1|\tau} F_{l+1|\tau}^\top \\
= Q_{l+1|l}
\]

\[
p(x_k|x_{k-1}) = \mathcal{N}\left( x_k; F_{k|k-1} x_{k-1}, Q_{k|k-1} \right)
\]
Parameter estimation for Sensor Data Fusion using the

BAYESIAN APPROACH
Properties of Joint Densities

- **Non-negative:**
  \[ p(x, y) \geq 0 \]

- **Normalized:**
  \[ \int dx \ dy \ p(x, y) = 1 \]

- **Relation between \( x \) and \( y \)**
  \[ p(x) = \int dy \ p(x, y) \]
  \[ p(y) = \int dx \ p(x, y) \]

- **We saw:**
  \[ p(x|y) = \frac{p(x, y)}{p(y)} \]

- **If \( x \) and \( y \) are mutually independent, iff:**
  \[ p(x, y) = p(x) \cdot p(y) \]

- **then:**
  \[ p(x|y) = \frac{p(x, y)}{p(y)} = \frac{p(x)p(y)}{p(y)} = p(x) \]
Objective: Target Tracking

- Given: Set of measurements.
- Recursive expression:
  \[ Z^k = \{ z_k, z_{k-1}, z_{k-2}, \ldots, z_1 \} = \{ z_k Z^{k-1} \} \]
- Multivariate RV \( x \): state at time \( k \)
  \[ x_k = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \]
- Objective: Calculate
  \[ p(x_k \mid Z^k) \]
  \[ x_k|k = E \left[ x_k \mid Z^k \right] \]
  \[ P_{k|k} = \text{cov} \left[ x_k \mid Z^k \right] \]
The BAYES Formula

- Rule (incl. proof):

\[ p(x|y) = \frac{p(x,y)}{p(y)} = \frac{p(y,x)}{p(y)} = \frac{p(y|x)p(x)}{p(y)} \]

Bayes Rule:

\[ p(x|y) = \frac{p(y|x)p(x)}{p(y)} \]
Simplifications of Bayes Theorem

Normalization can be solved by integration:

\[
p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\int dx \ p(y, x)} = \frac{p(y|x)p(x)}{\int dx \ p(y|x)p(x)}
\]

Assume \(p(x|y)\) is a Gaussian density.

\[
p(x|y) = \frac{p(y|x)p(x)}{p(y)}
\]

Gaussian is normalized.

\[\rightarrow \text{sufficient to know:}
\]

\[
p(x|y) \propto p(y|x)p(x)
\]

\[
\text{Posterior} \propto \text{Likelihood function} \cdot \text{Prior}
\]
THE KALMAN FILTER

The core of Linear Estimation
Kalman Filter Initialization

The track of a Kalman filter is initialized by a Gaussian density

\[ p(x_0) = \mathcal{N}(x_0; x_{0\mid 0}, P_{0\mid 0}) \]

with

- Initialization estimate \( x_{0\mid 0} \)
- First measurement
- Track extraction algorithm
- Init. covariance \( P_{0\mid 0} \)
- User parameter
- “Initial ignorance” -> ‘large’ covariance
Prediction in Bayesian Estimation

Prediction is the prior target state pdf for the next time step.

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

\[
p(x_k|Z^{k-1}) \xrightarrow{\text{sensor model}} p(x_k|Z^k)
\]
Derivation of the Prior PDF

Use marginalization:

\[
p(x_k|z^{k-1}) = \int dx_{k-1} p(x_k, x_{k-1}|z^{k-1})
= \int dx_{k-1} p(x_k|x_{k-1}, z^{k-1}) \cdot p(x_{k-1}|z^{k-1})
= \int dx_{k-1} p(x_k|x_{k-1}) \cdot p(x_{k-1}|z^{k-1})
\]

The last equation follows from the Markov property!

But how to solve integral?
The Kalman filter assumptions for dynamics are

- linear Gaussian transition kernel

\[ p(x_k|x_{k-1}) = \mathcal{N}(x_k; F_{k|k-1}x_{k-1}, Q_{k|k-1}) \]

- previous posterior pdf is a Gaussian

\[ p(x_{k-1}|z^{k-1}) = \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1}) \]

Then, the prior for time k is also a Gaussian:

\[ p(x_k|z^{k-1}) = \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \]
Prior Mean for Kalman Filter

The prior mean is given by

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

\[
x_{k|k-1} = \mathbb{E} \left[ x_k | Z^{k-1} \right]
\]

\[
= \mathbb{E} \left[ F_{k|k-1}x_{k-1} + w_k | Z^{k-1} \right]
\]

\[
= F_{k|k-1} \mathbb{E} \left[ x_{k-1} | Z^{k-1} \right] + \mathbb{E} \left[ w_k | Z^{k-1} \right]
\]

\[
= F_{k|k-1} \mathbb{E} \left[ x_{k-1} | Z^{k-1} \right] + \mathbb{E} \left[ w_k \right]
\]

\[
= F_{k|k-1}x_{k-1|k-1}
\]
Prior Covariance for Kalman Filter

The prior covariance is given by

\[ P_{k|k-1} = \text{cov} \left[ x_k | Z^{k-1} \right] \]

\[ = E \left[ \left( x_k - E \left[ x_k | Z^{k-1} \right] \right) \left( x_k - E \left[ x_k | Z^{k-1} \right] \right)^\top \right] | Z^{k-1} \]

\[ = E \left[ (x_k - x_{k|k-1}) (x_k - x_{k|k-1})^\top \right] | Z^{k-1} \]

\[ = E \left[ \left( F_{k|k-1} x_{k-1} + w_k - x_{k|k-1} \right) \left( F_{k|k-1} x_{k-1} + w_k - x_{k|k-1} \right)^\top \right] | Z^{k-1} \]

\[ = E \left[ (F_{k|k-1} x_{k-1} - F_{k|k-1} x_{k-1}) (F_{k|k-1} x_{k-1} - F_{k|k-1} x_{k-1})^\top \right] | Z^{k-1} \] + \[ E \left[ w w^\top \right] \]

\[ = F_{k|k-1} E \left[ (x_{k-1} - x_{k-1|k-1}) (x_{k-1} - x_{k-1|k-1})^\top \right] | Z^{k-1} \] \[ F_{k|k-1}^\top + E \left[ w_k w_k^\top \right] \]

\[ = F_{k|k-1} \text{cov} \left[ x_{k-1} | Z^{k-1} \right] F_{k|k-1}^\top + \text{cov} [w_k] \]

\[ = F_{k|k-1} P_{k-1|k-1} F_{k|k-1}^\top + Q_{k|k-1} \]
Kalman Prediction

As a conclusion, we have

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

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

\[
\begin{align*}
x_{k|k-1} &= F_{k|k-1} x_{k-1|k-1} \\
P_{k|k-1} &= F_{k|k-1} P_{k-1|k-1} F_{k|k-1}^\top + Q_{k|k-1}
\end{align*}
\]
A small excursion to

LEAST SQUARES FILTERING
Problem Setting

Assume a random variable $z$ is given as a set of observations of $x$. We further have

\[
\begin{align*}
E[z] &= Hx \\
cov[z] &= R
\end{align*}
\]

Observe:

\[
\begin{aligned}
\arg \max_x \mathcal{N}(z; Hx, R) &= \arg \min_x (z - Hx)^\top R^{-1} (z - Hx) \\
\max_{z|z} \mathcal{N}(z; Hx, R) &= \min_{z|z} (z - Hx)^\top R^{-1} (z - Hx)
\end{aligned}
\]

Maximum Likelihood Estimate \( ML(x|z) \)

Least Squares Estimate \( LS(x|z) \)
Computation of Least Squares Solution

The solution of the LS estimator has zero gradient:

\[ 0 = \nabla_x [(z - Hx)^\top R^{-1}(z - Hx)] \]
\[ = 2(z - Hx)^\top R^{-1}H \]

Therefore

\[ H^\top R^{-1}z = H^\top R^{-1}Hx \]

and finally

\[ x = (H^\top R^{-1}H)^{-1}H^\top R^{-1}z \]

Least Squares Estimate \( \hat{x} \)
Covariance of LS Solution

A short computation yields

\[
\text{cov} \left[ \hat{x} \right] = \mathbb{E} \left[ (x - \hat{x})(x - \hat{x})^\top \right] \\
= \mathbb{E} \left[ (x - (H^\top R^{-1}H)^{-1}H^\top R^{-1}z)(x - (H^\top R^{-1}H)^{-1}H^\top R^{-1}z)^\top \right] \\
= \mathbb{E} \left[ (H^\top R^{-1}H)(H^\top R^{-1}H)x - H^\top R^{-1}z)(H^\top R^{-1}H)x - H^\top R^{-1}z)^\top \right] \left( H^\top R^{-1}H \right)^{-1} \\
= \mathbb{E} \left[ (H^\top R^{-1}H)H^\top R^{-1}(Hx - z)(Hx - z)^\top \left( H^\top R^{-1} \right)^\top \left( H^\top R^{-1}H \right)^{-1} \right] \\
= (H^\top R^{-1}H)H^\top R^{-1}\mathbb{E} \left[ (Hx - z)(Hx - z)^\top \right] R^{-1}H(H^\top R^{-1}H)^{-1} \\
= (H^\top R^{-1}H)^{-1}
\]
Least Squares Conclusion

Together, we have

\[ P = (H^\top R^{-1}H)^{-1} \]
\[ \hat{x} = PH^\top R^{-1}z \]

if

\[ E[z] = Hx \]
\[ \text{cov}[z] = R \]

and where

\[ P = \text{cov}[\hat{x}|z] \]
The alternative derivation

KALMAN FILTER AS LEAST SQUARES SOLUTION
Kalman Filter as LS Solution

The posterior estimate is obtained as the solution of the LS where the prior estimate and the measurement are the input data.

We define:

\[
z^* = \begin{pmatrix} x_{k|k-1} \\ z_k \end{pmatrix}
\]

Therefore, it holds that

\[
E[z^* | x_k] = \begin{pmatrix} x_k \\ H_k x_k \end{pmatrix}
\]

which also equals

\[H^* x_k\]

We obtain

\[
H^* = \begin{pmatrix} I \\ H_k \end{pmatrix}
\]

Furthermore, we have

\[
cov[z^* | x_k] = R^* = \begin{pmatrix} P_{k|k-1} & \cdot \\ \cdot & R_k \end{pmatrix}
\]
Posterior Covariance by LS Solution

Now, we apply the LS equations:

\[
P_{k|k} = (H^*\top R^*^{-1} H^*)^{-1}
\]

\[
= \left( \begin{pmatrix} I & H_k^\top \end{pmatrix} \begin{pmatrix} P_{k|k-1} & R_k \end{pmatrix}^{-1} \begin{pmatrix} I \\ H_k \end{pmatrix} \right)^{-1}
\]

\[
= \left( \begin{pmatrix} I & H_k^\top \end{pmatrix} \begin{pmatrix} P_{k|k-1}^{-1} & R_k^{-1} \end{pmatrix} \begin{pmatrix} I \\ H_k \end{pmatrix} \right)^{-1}
\]

\[
= \left( \begin{pmatrix} P_{k|k-1}^{-1} & H_k^\top R_k^{-1} \end{pmatrix} \begin{pmatrix} I \\ H_k \end{pmatrix} \right)^{-1}
\]

\[
= \left( P_{k|k-1}^{-1} + H_k^\top R_k^{-1} H_k \right)^{-1}
\]
Posterior Estimate by LS Solution

And for the state estimate, we have

$$ x_{k|k} = P_{k|k}(H^* R^* - z^*) $$

$$ = P_{k|k} \left( (I \ H_k^T) \left( P_{k|k-1}^{-1} \ R_k^{-1} \right) \left( x_{k|k-1} \ z_k \right) \right) $$

$$ = P_{k|k} \left( \left( P_{k|k-1}^{-1} \ H_k^T \ R_k^{-1} \right) \left( x_{k|k-1} \ z_k \right) \right) $$

$$ = P_{k|k} \left( P_{k|k-1}^{-1} x_{k|k-1} + H_k^T R_k^{-1} z_k \right) $$

Together, we have obtained the information filter form of the Kalman filter:

$$ P_{k|k} = (P_{k|k-1}^{-1} + H_k^T R_k^{-1} H_k)^{-1} $$

$$ x_{k|k} = P_{k|k} \left( P_{k|k-1}^{-1} x_{k|k-1} + H_k^T R_k^{-1} z_k \right) $$
Information Filter Form Summary

We obtained the following formulas which are equivalent to the Kalman filter update:

\[ P_{k|k} = \left( P_{k|k-1}^{-1} - H_k^T R_k^{-1} H_k \right)^{-1} \]

\[ x_{k|k} = P_{k|k} \left( P_{k|k-1}^{-1} x_{k|k-1} + H_k^T R_k^{-1} z_k \right) \]

Note: The inverse of a covariance matrix is also called Fisher Information Matrix.

There is a one – one correspondence between Kalman filter – Information filter.
Filtering

Input

\[ x_{k|k-1} \]
\[ P_{k|k-1} \]
\[ z_k \]
\[ R_k \]

Likelihood

Output

\[ x_{k|k} \]
\[ P_{k|k} \]

Least Squares

\[ P_{k|k} = \left( P_{k|k-1}^{-1} - H_k^T R_k^{-1} H_k \right)^{-1} \]
\[ x_{k|k} = P_{k|k} \left( P_{k|k-1}^{-1} x_{k|k-1} + H_k^T R_k^{-1} z_k \right) \]
Kalman Filter – Information Filter

**Kalman Filter**
- Parameters:
  - State
  - Covariance
- Prediction:
  - slim
- Update:
  - More effort

**Information Filter**
- Parameters:
  - Information
  - Information state
- Prediction:
  - Much effort
- Update:
  - Very slim
Information Filter – Properties

- Easy to Initialize
  - \( I = 0 \).
- Matrix inversion in dimension of state.
- In multi-sensor systems:
  - Required parameters for Track-to-Track fusion (T2TF).
- Equivalent to Kalman Filter.
- Numerically more consistent.

Information Filter for Multiple Sensors

Assume $S$ sensors provide synchronized measurements at time $t_k$:

$$\{(z_k^1, R_k^1), \ldots, (z_k^S, R_k^S)\}$$

We define:

$$x_{ik|k}^i \rightarrow E[x_k|z^{k-1}_k, z_k^1, \ldots, z_k^i]$$

$$P_{ik|k}^i \rightarrow \text{cov}[x_k|z^{k-1}_k, z_k^1, \ldots, z_k^i]$$

According to the Information Filter, we have:

$$P_{1k|k} = (P_{k|k-1}^{-1} + H_k^T (R_k^1)^{-1} H_k)^{-1}$$

$$P_{2k|k} = ((P_{k|k}^{-1})^{-1} + H_k^T (R_k^2)^{-1} H_k)^{-1}$$

$$= (P_{k|k-1}^{-1} + H_k^T (R_k^1)^{-1} H_k + H_k^T (R_k^2)^{-1} H_k)^{-1}$$

$$\vdots$$

$$P_{Sk|k} = P_{Sk|k}^S = (P_{k|k-1}^{-1} + \sum_{s=1}^{S} H_k^T (R_k^s)^{-1} H_k)^{-1}$$
..and for the state:

Analogously:

\[ x_{k|k}^1 = P_{k|k}^1 (P_{k|k-1}^{-1} x_{k|k-1} + H_{k}^T (R_{k}^1)^{-1} z_{k}^1) \]

\[ x_{k|k}^2 = P_{k|k}^2 ((P_{k|k}^1)^{-1} x_{k|k-1} + H_{k}^T (R_{k}^2)^{-1} z_{k}^2) \]

\[ = P_{k|k}^2 (P_{k|k-1}^{-1} x_{k|k-1} + H_{k}^T (R_{k}^1)^{-1} z_{k}^1 + H_{k}^T (R_{k}^2)^{-1} z_{k}^2) \]

\[ \vdots \]

\[ x_{k|k} = x_{k|k}^S = P_{k|k} (P_{k|k-1}^{-1} x_{k|k-1} + \sum_{s=1}^{S} H_{k}^T (R_{k}^s)^{-1} z_{k}^s) \]
A small parenthesis

MATRIX INVERSION LEMMA
Matrix Inversion Lemma – Problem Formulation

Assume, we are looking for the inverse of a block matrix:

\[
\begin{pmatrix}
A & B \\
C & D
\end{pmatrix}^{-1} = ?
\]

where \(A\) and \(D\) are invertable (square) matrices.
Matrix Inversion Lemma – Solution Approach

Given that the inverse is represented by $E,F,G,$ and $H$, we have:

$$
\begin{pmatrix}
A & B \\
C & D
\end{pmatrix}
\cdot
\begin{pmatrix}
E & F \\
G & H
\end{pmatrix}
=
\begin{pmatrix}
I & O \\
O & I
\end{pmatrix}
\quad \text{and} \quad
\begin{pmatrix}
E & F \\
G & H
\end{pmatrix}
\cdot
\begin{pmatrix}
A & B \\
C & D
\end{pmatrix}
=
\begin{pmatrix}
I & O \\
O & I
\end{pmatrix}
$$

This yields the following equations.

- $AE + BG = I$
- $AF + BH = O$
- $CE + DG = O$
- $CF + DH = I$
- $EA + FC = I$
- $EB + FD = O$
- $GA + HC = O$
- $GB + HD = I$
Matrix Inversion – Computed Solution

From equation block A, we obtain

\[
BG = -BD^{-1}CE
\]
\[
AE - BD^{-1}CE = I
\]
\[
E = (A - BD^{-1}C)^{-1}
\]
\[
G = -D^{-1}CE
\]
\[
= -D^{-1}(A - BD^{-1}C)^{-1}
\]
\[
H = D^{-1} - GBD^{-1}
\]
\[
= D^{-1} + D^{-1}(A - BD^{-1}C)^{-1}BD^{-1}
\]
\[
CF = -CA^{-1}BH
\]
\[
-CA^{-1}BH + DH = I
\]
\[
F = -A^{-1}B(D - CA^{-1}B)^{-1}
\]

From equation block B, we obtain

\[
E = A^{-1} - FCA^{-1}
\]
\[
= A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}
\]
\[
F = -(A - BD^{-1}C)^{-1}BD^{-1}
\]
\[
GB = -HCA^{-1}B
\]
\[
-HCA^{-1}B + HD = I
\]
\[
H = (D - CA^{-1}B)^{-1}
\]
\[
G = -(D - CA^{-1}B)^{-1}CA^{-1}
\]

Putting everything together:

\[
\begin{pmatrix}
A & B \\
C & D
\end{pmatrix}^{-1}
= \begin{pmatrix}
(A - BD^{-1}C)^{-1} & -(A - BD^{-1}C)^{-1}BD^{-1} \\
-D^{-1}C(A - BD^{-1}C)^{-1} & D^{-1} + D^{-1}C(A - BD^{-1}C)^{-1}BD^{-1}
\end{pmatrix}
\]
\[
= \begin{pmatrix}
A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1} & -A^{-1}B(D - CA^{-1}B)^{-1} \\
-(D - CA^{-1}B)^{-1}CA^{-1} & (D - CA^{-1}B)^{-1}
\end{pmatrix}
\]