Covariance of Plant-Informed State Residual under Steady-State Kalman Filtering

Gabriele Gualandi

1 Control system

We consider a linear time-invariant (LTI) system:

$$\dot{\boldsymbol{x}}(t) = A_c \boldsymbol{x}(t) + B_c \boldsymbol{u}(t) \tag{1}$$

$$\boldsymbol{y}_m(t) = C_c \boldsymbol{x}(t) \tag{2}$$

and its discretized version:

$$\boldsymbol{x}_{k+1} = A\boldsymbol{x}_k + B\boldsymbol{u}_k + \boldsymbol{w}_k, \tag{3}$$

$$\boldsymbol{y}_k = C\boldsymbol{x}_k + \boldsymbol{v}_k. \tag{4}$$

Assume $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are zero-mean, mutually independent white sequences with

$$\mathrm{Cov}(\boldsymbol{w}_k) = Q, \qquad \mathrm{Cov}(\boldsymbol{v}_k) = R, \qquad \mathrm{Cov}(\boldsymbol{w}_k, \boldsymbol{v}_k) = 0,$$

and independent of the initial state (equivalently, of $\boldsymbol{z}_0$).

A state feedback controller and a steady-state Kalman filter are employed:

$$\text{Observer-controller:} \quad \begin{cases} \boldsymbol{r}_k &= \boldsymbol{y}_k - C\hat{\boldsymbol{x}}_k \\ \hat{\boldsymbol{x}}_{k+1} &= A\hat{\boldsymbol{x}}_k + B\boldsymbol{u}_k + L\boldsymbol{r}_k \\ \boldsymbol{u}_k &= -K\hat{\boldsymbol{x}}_k \end{cases} \tag{5}$$

where $\hat{\boldsymbol{x}}_k$ is the estimated state and $\boldsymbol{r}_k$ is the innovation.

The observer gain L is derived from the solution of the steady-state discrete algebraic Riccati equation (DARE):

$$P_\infty = AP_\infty A^\top - AP_\infty C^\top (CP_\infty C^\top + R)^{-1} CP_\infty A^\top + Q \tag{6}$$

The unique positive semidefinite stabilizing solution P_∞ is the steady-state *a priori* (prediction) estimation error covariance matrix:

$$P_\infty = \lim_{k\to\infty} \mathbb{E}\left[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^\top\right].$$

The corresponding steady-state Kalman gain is:

$$L = AP_\infty C^\top (CP_\infty C^\top + R)^{-1}. \tag{7}$$

The state feedback gain K is computed via LQR design.

2 Finite Horizon Sensing-free Covariance

We define a *feedforward simulated state* trajectory, $\widetilde{\boldsymbol{x}}_k$, which evolves without measurement updates:

$$\widetilde{\boldsymbol{x}}_{k+1} = A\,\widetilde{\boldsymbol{x}}_k + B\,\boldsymbol{u}_k. \tag{8}$$

This provides a reference for computing the *sensing-free* state residual:

$$\boldsymbol{\delta}_k := \hat{\boldsymbol{x}}_k - \widetilde{\boldsymbol{x}}_k. \tag{9}$$

Our goal is to compute the covariance of this residual after h steps, $\mathrm{Cov}(\boldsymbol{\delta}_h)$. We assume that at the start of the window ($k = 0$), the Kalman filter is in steady-state and we initialize the simulation to match the state estimate, i.e., $\widetilde{\boldsymbol{x}}_0 = \hat{\boldsymbol{x}}_0$, which implies $\boldsymbol{\delta}_0 = \mathbf{0}$.

Let the estimation error be $\boldsymbol{e}_k := \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$. The dynamics of the error and the residual can be combined into an augmented state-space system with state vector $\boldsymbol{z}_k = [\boldsymbol{e}_k^\top, \boldsymbol{\delta}_k^\top]^\top$:

$$\boldsymbol{z}_{k+1} = \underbrace{\begin{bmatrix} A - LC & 0 \\ LC & A \end{bmatrix}}_{F} \boldsymbol{z}_k + \underbrace{\begin{bmatrix} I & -L \\ 0 & L \end{bmatrix}}_{G} \begin{bmatrix} \boldsymbol{w}_k \\ \boldsymbol{v}_k \end{bmatrix}. \tag{10}$$

The covariance of the augmented state, $P_{z,k} = \mathbb{E}[\boldsymbol{z}_k \boldsymbol{z}_k^\top]$, propagates according to the discrete-time Lyapunov equation:

$$P_{z,k+1} = F P_{z,k} F^\top + G \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} G^\top. \tag{11}$$

Under our assumptions, the initial covariance is $P_{z,0} = \mathrm{diag}(P_\infty, 0)$. The desired covariance after h steps, $\mathrm{Cov}(\boldsymbol{\delta}_h)$, is the bottom-right $n \times n$ block of the matrix $P_{z,h}$, which is computed by iterating (11) h times starting from $P_{z,0}$.

Theorem 2.1 (h-Step Covariance of the Observer-Model Residual)**.** *Let the system and steady-state Kalman filter be defined as above. Let the residual be $\boldsymbol{\delta}_k := \hat{\boldsymbol{x}}_k - \widetilde{\boldsymbol{x}}_k$, where the simulated state evolves according to* (8) *with initialization $\widetilde{\boldsymbol{x}}_0 = \hat{\boldsymbol{x}}_0$. Then, the covariance of the residual after h steps, $\Sigma_{\delta,h} = \mathrm{Cov}(\boldsymbol{\delta}_h)$, is given by the* $(2, 2)$ *block of the matrix $P_{z,h}$, where $P_{z,h}$ is obtained by iterating the discrete-time Lyapunov equation h times:*

$$P_{z,k+1} = F P_{z,k} F^\top + \Pi, \quad k = 0, \ldots, h-1,$$

starting from the initial condition $P_{z,0} = \mathrm{diag}(P_\infty, 0)$, where

$$F = \begin{bmatrix} A - LC & 0 \\ LC & A \end{bmatrix}, \quad G = \begin{bmatrix} I & -L \\ 0 & L \end{bmatrix}, \\ \Pi = G\,\mathrm{diag}(Q, R) G^\top.$$

Proof. Through the system dynamics equations, the one-step evolution of the estimation error, $\boldsymbol{e}_k := \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$, and the residual, $\boldsymbol{\delta}_k := \hat{\boldsymbol{x}}_k - \widetilde{\boldsymbol{x}}_k$, are derived as:

$$\begin{aligned} \boldsymbol{e}_{k+1} &= (A - LC)\boldsymbol{e}_k + \boldsymbol{w}_k - L\boldsymbol{v}_k, \\ \boldsymbol{\delta}_{k+1} &= A\boldsymbol{\delta}_k + LC\boldsymbol{e}_k + L\boldsymbol{v}_k. \end{aligned}$$

These dynamics are combined into a single state-space representation for the augmented state $\boldsymbol{z}_k := [\boldsymbol{e}_k^\top, \boldsymbol{\delta}_k^\top]^\top$, yielding the linear system:

$$\boldsymbol{z}_{k+1} = F\boldsymbol{z}_k + G\boldsymbol{\eta}_k,$$

with $F \triangleq \begin{bmatrix} A - LC & 0 \\ LC & A \end{bmatrix}$, $G \triangleq \begin{bmatrix} I & -L \\ 0 & L \end{bmatrix}$, and the stacked noise vector $\boldsymbol{\eta}_k \triangleq [\boldsymbol{w}_k^\top, \boldsymbol{v}_k^\top]^\top$.

The propagation of the augmented state covariance, $P_{z,k} \triangleq \mathbb{E}[\boldsymbol{z}_k \boldsymbol{z}_k^\top]$, is found by evaluating $P_{z,k+1} = \mathbb{E}[\boldsymbol{z}_{k+1} \boldsymbol{z}_{k+1}^\top]$:

$$P_{z,k+1} = F\mathbb{E}[\boldsymbol{z}_k \boldsymbol{z}_k^\top]F^\top + F\mathbb{E}[\boldsymbol{z}_k \boldsymbol{\eta}_k^\top]G^\top + G\mathbb{E}[\boldsymbol{\eta}_k \boldsymbol{z}_k^\top]F^\top + G\mathbb{E}[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^\top]G^\top.$$

This expression simplifies based on the following properties:

- By definition, $\mathbb{E}[\boldsymbol{z}_k \boldsymbol{z}_k^\top] = P_{z,k}$.
- Since the noise $\boldsymbol{\eta}_k$ is white and zero-mean, it is uncorrelated with the past state $\boldsymbol{z}_k$, causing the cross-terms to be zero: $\mathbb{E}[\boldsymbol{z}_k \boldsymbol{\eta}_k^\top] = 0$.
- As $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are mutually independent, the noise covariance is $\mathbb{E}[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^\top] = \mathrm{diag}(Q, R)$.

Substituting these simplifications yields the discrete-time Lyapunov equation:

$$P_{z,k+1} = F P_{z,k} F^\top + \Pi,$$

where the constant noise contribution is $\Pi \triangleq G\,\mathrm{diag}(Q, R) G^\top$.

The initial covariance $P_{z,0} = \mathbb{E}[\boldsymbol{z}_0 \boldsymbol{z}_0^\top]$ is determined by the conditions at the synchronization point $k_0 = 0$:

- The estimation error covariance is given by the steady-state value, $\mathrm{Cov}(\boldsymbol{e}_0) = P_\infty$.
- By the synchronization rule, the initial residual is a deterministic zero vector, $\boldsymbol{\delta}_0 = \mathbf{0}$. Therefore, its covariance and all cross-covariances involving it are zero: $\mathrm{Cov}(\boldsymbol{\delta}_0) = 0$ and $\mathbb{E}[\boldsymbol{e}_0\boldsymbol{\delta}_0^\top] = 0$.

This results in the initial condition $P_{z,0} = \mathrm{diag}(P_\infty, 0)$. By iterating the Lyapunov equation h times from this state, we obtain $P_{z,h}$. The desired covariance, $\Sigma_{\delta,h}$, is the bottom-right $(2,2)$ block of this matrix. □

2.1 Detailed Derivation

Detailed Derivation. Without loss of generality, we analyze the evolution of the residual over a horizon of h steps starting from a synchronization point at time $k_0 = 0$.

1. Dynamics of the Estimation Error $\boldsymbol{e}_k$ The estimation error is defined as $\boldsymbol{e}_k := \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$. Its one-step evolution is derived as follows:

$$\begin{aligned}
\boldsymbol{e}_{k+1} &= \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} \\
&= (A\boldsymbol{x}_k + B\boldsymbol{u}_k + \boldsymbol{w}_k) - \hat{\boldsymbol{x}}_{k+1} \\
&= (A\boldsymbol{x}_k + B\boldsymbol{u}_k + \boldsymbol{w}_k) - (A\hat{\boldsymbol{x}}_k + B\boldsymbol{u}_k + L(\boldsymbol{y}_k - C\hat{\boldsymbol{x}}_k)) \\
&= A(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + \boldsymbol{w}_k - L(\boldsymbol{y}_k - C\hat{\boldsymbol{x}}_k) \\
&= A\boldsymbol{e}_k + \boldsymbol{w}_k - L((C\boldsymbol{x}_k + \boldsymbol{v}_k) - C\hat{\boldsymbol{x}}_k) \\
&= A\boldsymbol{e}_k + \boldsymbol{w}_k - L(C(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + \boldsymbol{v}_k) \\
\boldsymbol{e}_{k+1} &= (A - LC)\boldsymbol{e}_k + \boldsymbol{w}_k - L\boldsymbol{v}_k.
\end{aligned}$$

2. Dynamics of the Residual $\boldsymbol{\delta}_k$ The residual is defined as $\boldsymbol{\delta}_k := \hat{\boldsymbol{x}}_k - \widetilde{\boldsymbol{x}}_k$. Its one-step evolution is:

$$\begin{aligned}
\boldsymbol{\delta}_{k+1} &= \hat{\boldsymbol{x}}_{k+1} - \widetilde{\boldsymbol{x}}_{k+1} \\
&= (A\hat{\boldsymbol{x}}_k + B\boldsymbol{u}_k + L(\boldsymbol{y}_k - C\hat{\boldsymbol{x}}_k)) - \widetilde{\boldsymbol{x}}_{k+1} \\
&= (A\hat{\boldsymbol{x}}_k + B\boldsymbol{u}_k + L(\boldsymbol{y}_k - C\hat{\boldsymbol{x}}_k)) - (A\widetilde{\boldsymbol{x}}_k + B\boldsymbol{u}_k) \\
&= A(\hat{\boldsymbol{x}}_k - \widetilde{\boldsymbol{x}}_k) + L(\boldsymbol{y}_k - C\hat{\boldsymbol{x}}_k) \\
&= A\boldsymbol{\delta}_k + L((C\boldsymbol{x}_k + \boldsymbol{v}_k) - C\hat{\boldsymbol{x}}_k) \\
&= A\boldsymbol{\delta}_k + L(C(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + \boldsymbol{v}_k) \\
\boldsymbol{\delta}_{k+1} &= A\boldsymbol{\delta}_k + LC\boldsymbol{e}_k + L\boldsymbol{v}_k.
\end{aligned}$$

3. Assembling the Augmented System We combine the two derived dynamics into a single state-space representation for the augmented state $\boldsymbol{z}_k = [\boldsymbol{e}_k^\top, \boldsymbol{\delta}_k^\top]^\top$.

$$\begin{bmatrix} \boldsymbol{e}_{k+1} \\ \boldsymbol{\delta}_{k+1} \end{bmatrix} = \begin{bmatrix} A - LC & 0 \\ LC & A \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_k \\ \boldsymbol{\delta}_k \end{bmatrix} + \begin{bmatrix} I & -L \\ 0 & L \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_k \\ \boldsymbol{v}_k \end{bmatrix}.$$

This is the linear system $\boldsymbol{z}_{k+1} = F\boldsymbol{z}_k + G\boldsymbol{\eta}_k$, where $\boldsymbol{\eta}_k = [\boldsymbol{w}_k^\top, \boldsymbol{v}_k^\top]^\top$ is the stacked noise vector.

4. Derivation of the Covariance Propagation The covariance of the augmented state is $P_{z,k} = \mathbb{E}[\boldsymbol{z}_k\boldsymbol{z}_k^\top]$. Its propagation is derived by taking the expectation of the outer product of $\boldsymbol{z}_{k+1}$:

$$\begin{aligned}
P_{z,k+1} &= \mathbb{E}[\boldsymbol{z}_{k+1}\boldsymbol{z}_{k+1}^\top] \\
&= \mathbb{E}\left[(F\boldsymbol{z}_k + G\boldsymbol{\eta}_k)(F\boldsymbol{z}_k + G\boldsymbol{\eta}_k)^\top\right] \\
&= \mathbb{E}\left[F\boldsymbol{z}_k\boldsymbol{z}_k^\top F^\top + F\boldsymbol{z}_k\boldsymbol{\eta}_k^\top G^\top + G\boldsymbol{\eta}_k\boldsymbol{z}_k^\top F^\top + G\boldsymbol{\eta}_k\boldsymbol{\eta}_k^\top G^\top\right] \\
&= F\mathbb{E}[\boldsymbol{z}_k\boldsymbol{z}_k^\top]F^\top + F\mathbb{E}[\boldsymbol{z}_k\boldsymbol{\eta}_k^\top]G^\top + G\mathbb{E}[\boldsymbol{\eta}_k\boldsymbol{z}_k^\top]F^\top + G\mathbb{E}[\boldsymbol{\eta}_k\boldsymbol{\eta}_k^\top]G^\top.
\end{aligned}$$

We simplify this expression by evaluating each of the four additive terms:

1. **Term 1:** $F\mathbb{E}[\boldsymbol{z}_k\boldsymbol{z}_k^\top]F^\top$. By definition, the covariance of the state is $P_{z,k} = \mathbb{E}[\boldsymbol{z}_k\boldsymbol{z}_k^\top]$. This term propagates the existing uncertainty from step k to $k+1$, becoming $FP_{z,k}F^\top$.

2. **Terms 2 and 3: The Cross-Terms**. The state $\boldsymbol{z}_k$ is a function of noise up to step $k-1$. Since the noise process $\boldsymbol{\eta}_k$ is white, the current noise $\boldsymbol{\eta}_k$ is uncorrelated with all past states. For zero-mean processes, this implies the expectation of the cross-product is zero:

$$\mathbb{E}[\boldsymbol{z}_k \boldsymbol{\eta}_k^\top] = 0.$$

Consequently, both cross-terms in the expansion evaluate to zero.

3. **Term 4:** $G\mathbb{E}[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^\top]G^\top$. This term represents the injection of new uncertainty. The expectation $\mathbb{E}[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^\top]$ is the covariance of the stacked noise vector $\boldsymbol{\eta}_k = [\boldsymbol{w}_k^\top, \boldsymbol{v}_k^\top]^\top$. Since $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are mutually independent, this covariance is a block-diagonal matrix:

$$\mathbb{E}[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^\top] = \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} = \mathrm{diag}(Q, R).$$

Substituting the simplified results for each term back into the expanded equation yields:

$$P_{z,k+1} = FP_{z,k}F^\top + 0 + 0 + G\,\mathrm{diag}(Q, R)G^\top.$$

This simplifies to the final form known as the **discrete-time Lyapunov equation**:

$$P_{z,k+1} = FP_{z,k}F^\top + \Pi,$$

where the constant noise contribution term is defined as $\Pi := G\,\mathrm{diag}(Q, R)G^\top$.

5. Initial Condition At the synchronization point $k_0 = 0$, the initial conditions are:

- The Kalman filter is in steady state, so the estimation error covariance is $\mathrm{Cov}(\boldsymbol{e}_0) = \mathbb{E}[\boldsymbol{e}_0 \boldsymbol{e}_0^\top] = P_\infty$.
- The residual is initialized to zero by definition: $\boldsymbol{\delta}_0 = \hat{\boldsymbol{x}}_0 - \widetilde{\boldsymbol{x}}_0 = \mathbf{0}$. As a deterministic value, its covariance is $\mathrm{Cov}(\boldsymbol{\delta}_0) = 0$.
- The cross-covariance is $\mathbb{E}[\boldsymbol{e}_0 \boldsymbol{\delta}_0^\top] = \mathbb{E}[\boldsymbol{e}_0 \mathbf{0}^\top] = 0$. Specifically, the cross-covariance involves the initial residual $\boldsymbol{\delta}_0$. By the initialization rule, $\widetilde{\boldsymbol{x}}_0 = \hat{\boldsymbol{x}}_0$, the initial residual is deterministically zero:

  $$\boldsymbol{\delta}_0 = \hat{\boldsymbol{x}}_0 - \widetilde{\boldsymbol{x}}_0 = \mathbf{0}.$$

  The expectation of the product of the random error vector $\boldsymbol{e}_0$ and this constant zero vector is the zero matrix:

  $$\mathbb{E}[\boldsymbol{e}_0 \boldsymbol{\delta}_0^\top] = \mathbb{E}[\boldsymbol{e}_0 \mathbf{0}^\top] = \mathbb{E}[0_{n\times n}] = 0.$$

Assembling these blocks gives the initial augmented covariance matrix:

$$P_{z,0} = \begin{bmatrix} \mathbb{E}[\boldsymbol{e}_0 \boldsymbol{e}_0^\top] & \mathbb{E}[\boldsymbol{e}_0 \boldsymbol{\delta}_0^\top] \\ \mathbb{E}[\boldsymbol{\delta}_0 \boldsymbol{e}_0^\top] & \mathbb{E}[\boldsymbol{\delta}_0 \boldsymbol{\delta}_0^\top] \end{bmatrix} = \begin{bmatrix} P_\infty & 0 \\ 0 & 0 \end{bmatrix} = \mathrm{diag}(P_\infty, 0).$$

6. Conclusion By iterating the Lyapunov equation h times starting from $P_{z,0}$, we obtain $P_{z,h}$. The desired covariance, $\Sigma_{\delta,h} = \mathrm{Cov}(\boldsymbol{\delta}_h)$, is the bottom-right $(2, 2)$ block of this resulting matrix $P_{z,h}$. □

Implementation-Oriented Form (MATLAB). For numerical propagation, we use the matrix recursion directly:

$$P_{z,k+1} = FP_{z,k}F^\top + \Pi,$$

with one-time initialization

$$P_{z,0} = \mathrm{diag}(P_\infty, 0), \qquad \Pi = G\,\mathrm{diag}(Q, R)G^\top.$$

This is implemented by two MATLAB functions:

- `initialize_covariance_step_propagator(A, C, Q, R, P_inf)`
  Input: discrete-time model/noise matrices and steady-state covariance.
  Output: structure `propagator_data` containing `F`, `Pi`, `Pz0`, and state dimension `n`.

- `propagate_covariance_onestep(propagator_data, Pz_current)`
  Input: precomputed structure and current covariance $P_{z,k}$.
  Output: next covariance $P_{z,k+1}$.

The practical workflow is:

1. Call `initialize_covariance_step_propagator` once to compute L, build F, G, and store Π and $P_{z,0}$.
2. Set $P_z \leftarrow P_{z,0}$.
3. For each horizon step, call `propagate_covariance_onestep` to apply $P_z \leftarrow F P_z F^\top + \Pi$.
4. Extract the residual covariance $\Sigma_{\delta,h}$ as the $(2,2)$ block of $P_{z,h}$.