POLITECNICO
MILANO 1863

EXERCISE 9 – KALMAN BUCY FILTER

EMANUELE Riva
Course: Mechatronics systems

Department of Mechanical Engineering, Politecnico di Milano, Italy

AGENDA

- THEORY REVIEW: Discrete Kalman filter
- THEORY REVIEW: Kalman-Bucy filter
- EXAMPLE
- HANDS-ON: Matlab

Recursive least square estimation

$$K_k = P_{k-1} H_k^\top \left(H_k P_{k-1} H_k^\top + R_k \right)^{-1}$$

$$\boxed{\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} + K_k \left(\mathbf{y}_k - H_k \hat{\mathbf{x}}_{k-1} \right)}$$

$$P_k = (I - K_k H_k) P_{k-1} (I - K_k H_k)^\top + K_k R_k K_k^\top$$

Definitions:

Measurement residual: $\epsilon_{y,k} = \mathbf{y}_k - H \hat{\mathbf{x}}_k$

Estimation error: $\epsilon_x = \mathbf{x} - \hat{\mathbf{x}}$

Estimation error covariance: $P_k = E(\epsilon_{x,k}, \epsilon_{x,k}^\top)$

Measurement covariance: $R_k = E(\mathbf{v}_k, \mathbf{v}_k^\top)$

$E(v_i^2) = \sigma_i^2 \;\; (i = 1, \dots, n)$

$$= \begin{bmatrix} \sigma_1^2 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & \sigma_n^2 \end{bmatrix}$$

Recursive least square estimation

$$\boxed{\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} + K_k \left(\mathbf{y}_k - H_k \hat{\mathbf{x}}_{k-1} \right)}$$

From two consecutive measurements coming in:

$$J_1 = \epsilon_y^\top R^{-1} \epsilon = (\mathbf{y}_1 - H_1 \hat{\mathbf{x}}_1)^\top R_1^{-1} (\mathbf{y}_1 - H_1 \hat{\mathbf{x}}_1)$$

$$J_2 = (\mathbf{y}_1 - H_1 \hat{\mathbf{x}}_2)^\top R_1^{-1} (\mathbf{y}_1 - H_1 \hat{\mathbf{x}}_2) + (\mathbf{y}_2 - H_2 \hat{\mathbf{x}}_2)^\top R_2^{-1} (\mathbf{y}_2 - H_2 \hat{\mathbf{x}}_2)$$

$\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ are found by minimizing J_1 and J_2 w.r.t. $\hat{\mathbf{x}}_{1,2} \longrightarrow \dfrac{\partial J_1}{\partial \hat{\mathbf{x}}_1} = 0 \quad \dfrac{\partial J_2}{\partial \hat{\mathbf{x}}_2} = 0$

$$\longrightarrow \quad \hat{\mathbf{x}}_1 = \left(H_1^\top R_1^{-1} H_1 \right)^{-1} H_1^\top R_1^{-1} \mathbf{y}_1$$

$$\hat{\mathbf{x}}_2 = \left(H_1^\top R_1^{-1} H_1 + H_2^\top R_2^{-1} H_2 \right)^{-1} \left(H_1^\top R_1^{-1} \mathbf{z}_1 + H_2^\top R_2^{-1} \mathbf{y}_2 \right)$$

The combination of these equations gives the above form form the recursive algorithm.

Recursive least square estimation

In order to evaluate K_k, the following cost is employed:

$$
\begin{aligned}
J_k &= E\left[(\mathbf{x}_1 - \hat{\mathbf{x}}_1)^2\right] + \ldots + E\left[(\mathbf{x}_n - \hat{\mathbf{x}}_n\right]^2 \\
&= E({\epsilon_{x_1,k}}^2 + \ldots + {\epsilon_{x_n,k}}^2) \\
&= E(\epsilon_{x_1,k}^\top \epsilon_{x_1,k}) \\
&= E\left[Tr(\epsilon_{x_1,k}\epsilon_{x_1,k}^\top)\right] \\
&= Tr P_k
\end{aligned}
$$

Where P_k is evaluated iteratively: $P_k = E(\epsilon_{x,k}\epsilon_{x,k}^\top)$ where:

$$
\begin{aligned}
\epsilon_{x,k} &= \mathbf{x} - \hat{\mathbf{x}}_k \\
\hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_{k-1} + K_k(\mathbf{y}_k - H_k\hat{\mathbf{x}}_{k-1}) \\
\mathbf{y}_k &= H_k\mathbf{x}
\end{aligned}
$$

$$
\xrightarrow[\text{Knowing that}]{E(\mathbf{v}_k\epsilon_{x,k-1}^\top) = 0}
\quad
\boxed{P_k = (I - K_kH_k)P_{k-1}(I - K_kH_k)^\top + K_kR_kK_k^\top}
$$

Recursive least square estimation

K_k is evaluated by minimizing the cost J_k :

$$\left(\frac{\partial Tr(ABA^\top)}{\partial A} = 2AB \right) \text{ If } B \text{ is symmetric}$$

$$\frac{\partial J_k}{\partial K_k} = 2(I - K_k H_k) P_{k-1} (-H_k^\top) + 2 K_k R_k = 0$$

$$\longrightarrow \boxed{K_k = P_{k-1} H_k^\top \left(H_k P_{k-1} H_k^\top + R_k \right)^{-1}}$$

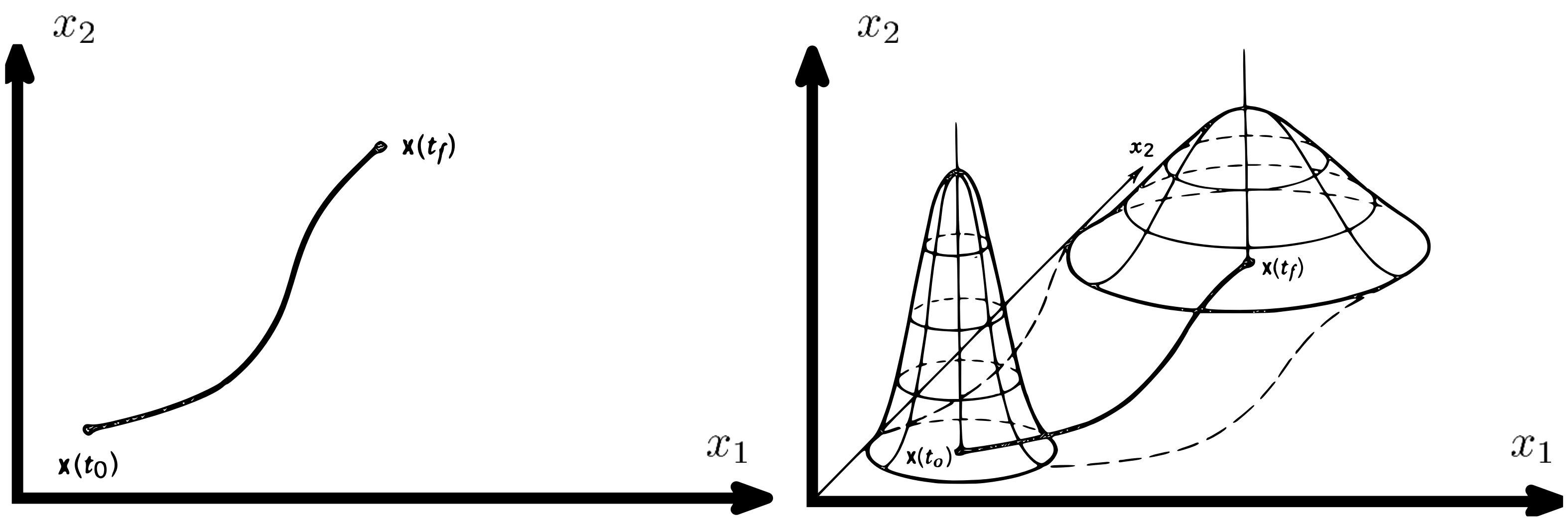
Recursive least square estimation

$$K_k = P_{k-1} H_k^\top \left(H_k P_{k-1} H_k^\top + R_k \right)^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} + K_k \left(\mathbf{y}_k - H_k \hat{\mathbf{x}}_{k-1} \right)$$

$$P_k = (I - K_k H_k) P_{k-1} (I - K_k H_k)^\top + K_k R_k K_k^\top$$

Dynamical model



discrete-time linear time varying (LTV) system:

$$\mathbf{x}_k = F_{k-1}\mathbf{x}_{i-1} + G_{k-1}\mathbf{u}_{\mathbf{i-1}} + \mathbf{w}_{k-1}$$

I. matrices F_{k-1} and G_{k-1}, and the control action $\mathbf{u}_{k-1}$ are known without errors

$$E\left[\mathbf{u_{k-1}}\right] = \overline{\mathbf{u}}_{\mathbf{k-1}} \qquad E\left[\left(\mathbf{u_{k-1}} - \overline{\mathbf{u}}_{\mathbf{k-1}}\right)\left(\mathbf{u_{k-1}} - \overline{\mathbf{u}}_{\mathbf{k-1}}\right)^{\top}\right] = 0$$

II. initial conditions are Gaussian random variables:

$$E\left[\mathbf{x_0}\right] = \overline{\mathbf{x}}_{\mathbf{0}} \qquad E\left[\left(\mathbf{x_0} - \overline{\mathbf{x}}_{\mathbf{0}}\right)\left(\mathbf{x_0} - \overline{\mathbf{x}}_{\mathbf{0}}\right)^{\top}\right] = P_0$$

III. input disturbances are random Gaussian with zero mean and are uncorrelated

$$E\left[\mathbf{w_{i,k}}\right] = 0 \qquad E\left[\mathbf{w_{i,k}}\mathbf{w_{j,k}}^{\top}\right] = Q_k\delta_{ij}$$

IV. cross variances between u_i and w_i are null, i.e. disturbances are completely independent from control actions

$$E\left[\mathbf{u_i}\mathbf{w_i}^{\top}\right] = 0 \qquad E\left[\mathbf{u_i}\mathbf{w_j}^{\top}\right] = 0$$

Ingredients: dynamical model

expected value of the state at a given time instant:

$$E\left(\mathbf{x_k}\right) = E\left(F_{k-1}\mathbf{x_{k-1}} + G_{k-1}\mathbf{u_{k-1}} + \mathbf{w_{k-1}}\right) \quad \rightarrow \quad \boxed{\mathbf{\overline{x}_k} = F_{k-1}\mathbf{\overline{x}_{k-1}} + G_{k-1}\mathbf{\overline{u}_{k-1}}}$$

expected value of the covariance at a given time instant:

$$E\left[\left(\mathbf{x_k} - \mathbf{\overline{x}_k}\right)\left(\mathbf{x_k} - \mathbf{\overline{x}_k}\right)^\top\right] = F_{k-1}E\left[\left(\mathbf{x_{k-1}} - \mathbf{\overline{x}_{k-1}}\right)\left(\mathbf{x_{k-1}} - \mathbf{\overline{x}_{k-1}}\right)^\top\right]F_{k-1}^\top +$$

$$+ G_{k-1}E\left[(\mathbf{u_{k-1}} - \mathbf{\overline{u}_{k-1}})(\mathbf{u_{k-1}} - \mathbf{\overline{u}_{k-1}})^\top\right]G_{k-1}^\top + E\left[\mathbf{w_{k-1}}\mathbf{w_{k-1}}^\top\right]$$

$$\rightarrow \quad \boxed{P_k = F_{k-1}P_{k-1}F_{k-1}^\top + Q_{k-1}}$$

1. Propagation of the state estimate:

$$\hat{\mathbf{x}}_{\mathbf{k}}^{-} = F_{k-1}\hat{\mathbf{x}}_{\mathbf{k-1}}^{+} + G_{k-1}\mathbf{u}_{\mathbf{k-1}}$$

2. Covariance estimate propagation:

$$P_k^{-} = F_{k-1}P_{k-1}^{+}F_{k-1}^{\top} + Q_{k-1}$$

3. Filter gain computation:

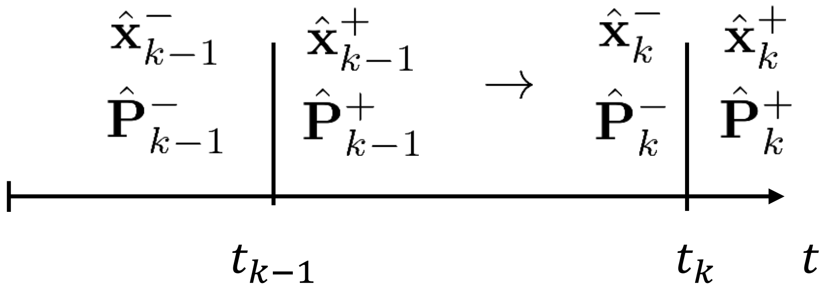
$$\begin{aligned} K_k &= P_k^{-}H_k^{\top}\left(H_kP_k^{-}H_k^{\top} + R_k\right)^{-1} \\ &= P_k^{+}H_k^{\top}R_k^{-1} \end{aligned}$$

4. State estimate update:

$$\hat{\mathbf{x}}_k^{+} = \hat{\mathbf{x}}_k^{-} + K_k\left(\mathbf{y}_k - H_k\hat{\mathbf{x}}_k^{-}\right)$$

5. Covariance estimate update:

$$\begin{aligned} P_k &= (I - K_kH_k)P_k^{-}(I - K_kH_k)^{\top} + K_kR_kK_k^{\top} \\ &= \left[(P_k^{-})^{-1} + H_k^{\top}R_k^{-1}H_k\right]^{-1} \\ &= (I - K_kH_k)\,P_k^{-} \end{aligned}$$



Merging these equations and considering the limit for $\Delta t_k \rightarrow \mathbf{0}$ we get to the continuous time Kalman-Bucy filter.

- Different scenario depending upon the computational time t_c and the reference time for the estimation Δt_k

I. if $t_c = \Delta t_k$ the algorithm is running in real time; the algorithm behaves as a filter (cleans up estimations minimizing uncertainties);

II. if $t_c > \Delta t_k$ the algorithm is running in post-processing; it means it is used to smooth out data;

III. if $t_c < \Delta t_k$ the algorithm is used to estimate future states; thus, it provides a prediction considering the limit for $\Delta t_k \to 0$ the architecture is similar to the Luenberger estimator.

$$\boxed{\dot{\hat{\boldsymbol{x}}} = A\hat{\boldsymbol{x}} + B\boldsymbol{u} + K_{\boldsymbol{o}}(\boldsymbol{y} - C\hat{\boldsymbol{x}}) \qquad \hat{\boldsymbol{x}}(t_0) = \boldsymbol{x_0}}$$

Where the choice of the observer gain matrix is done solving the DRE:

$$\boxed{\dot{P}(t) = Q + AP(t) + P(t)A^T - P(t)C^T R^{-1} CP(t) \qquad P(t_0) = P_0}$$

with: $\boxed{K_o = P(t)C^T R^{-1}}$

If the transitory is negligible (similar to infinite time control problems) the DRE can be approximated (ARE):

$$Q + AP_{ss} + P_{ss}A^T - P_{ss}C^T R^{-1} CP_{ss} = 0 \qquad K = P_{ss}C^T R^{-1}$$

THEORY REVIEW: KBF

Let us consider again the system under the effects of random noise and random disturbances.

Real System

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u} + L\boldsymbol{w}$$
$$\boldsymbol{y} = C\boldsymbol{x} + \boldsymbol{n}$$

State Observer

$$\dot{\hat{\boldsymbol{x}}} = A\hat{\boldsymbol{x}} + B\boldsymbol{u} + K_o(\boldsymbol{z} - C\hat{\boldsymbol{x}})$$
$$\hat{\boldsymbol{y}} = C\hat{\boldsymbol{x}}$$

The dynamic equation of the state estimation error becomes:

$$\dot{\boldsymbol{\varepsilon}} = \dot{\boldsymbol{x}} - \dot{\hat{\boldsymbol{x}}} = (A\boldsymbol{x} + B\boldsymbol{u} + L\boldsymbol{w}) - \left(A\hat{\boldsymbol{x}} + B\boldsymbol{u} + K_{\boldsymbol{o}}(\boldsymbol{z} - \hat{\boldsymbol{z}})\right)$$

$$\dot{\boldsymbol{\varepsilon}} = (A - K_o C)\boldsymbol{\varepsilon} + L\boldsymbol{w} + K_o\boldsymbol{n}$$

I.e. properly weighting how much my system (Q) or the measurements (R) are reliable it is possible to filter out the noise $\boldsymbol{n}$.

$$Q + AP_{ss} + P_{ss}A^T - P_{ss}C^T R^{-1} C P_{ss} = 0 \qquad K = P_{ss}C^T R^{-1}$$

THEORY REVIEW: duality LQR-KBF

Optimal control LQR

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}$$

$$\boldsymbol{y} = C\boldsymbol{x}$$

$$J = \boldsymbol{x}_f^T P \boldsymbol{x}_f + \int_{t_0}^{t_f} L(\boldsymbol{x}, \boldsymbol{u})\, dt$$

$$\boldsymbol{u} = -K\boldsymbol{x}$$

Where:

$$K = R^{-1}B^T P$$

$$\dot{P} = -A^T P - PA - Q + PBR^{-1}B^T P$$

$$P(t_f) = P_f$$

Kalman-Bucy filter

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u} + L\boldsymbol{w}$$

$$\boldsymbol{y} = C\boldsymbol{x} + \boldsymbol{n}$$

$$J = \int_{t_0}^{t_f} \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}\, dt$$

$$\dot{\hat{\boldsymbol{x}}} = A\hat{\boldsymbol{x}} + B\boldsymbol{u} + K_o(\boldsymbol{y} - C\hat{\boldsymbol{x}})$$

Where:

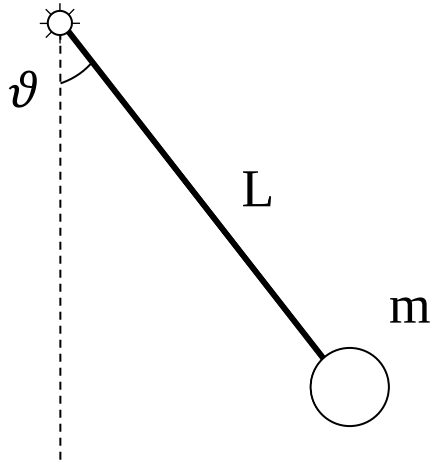
$$K_o = PC^T R^{-1}$$

$$\dot{P} = A^T P + PA + Q - PCR^{-1}C^T P$$

$$P(t_0) = P_0$$

Exercise 1: KBF

From our previous exercise:



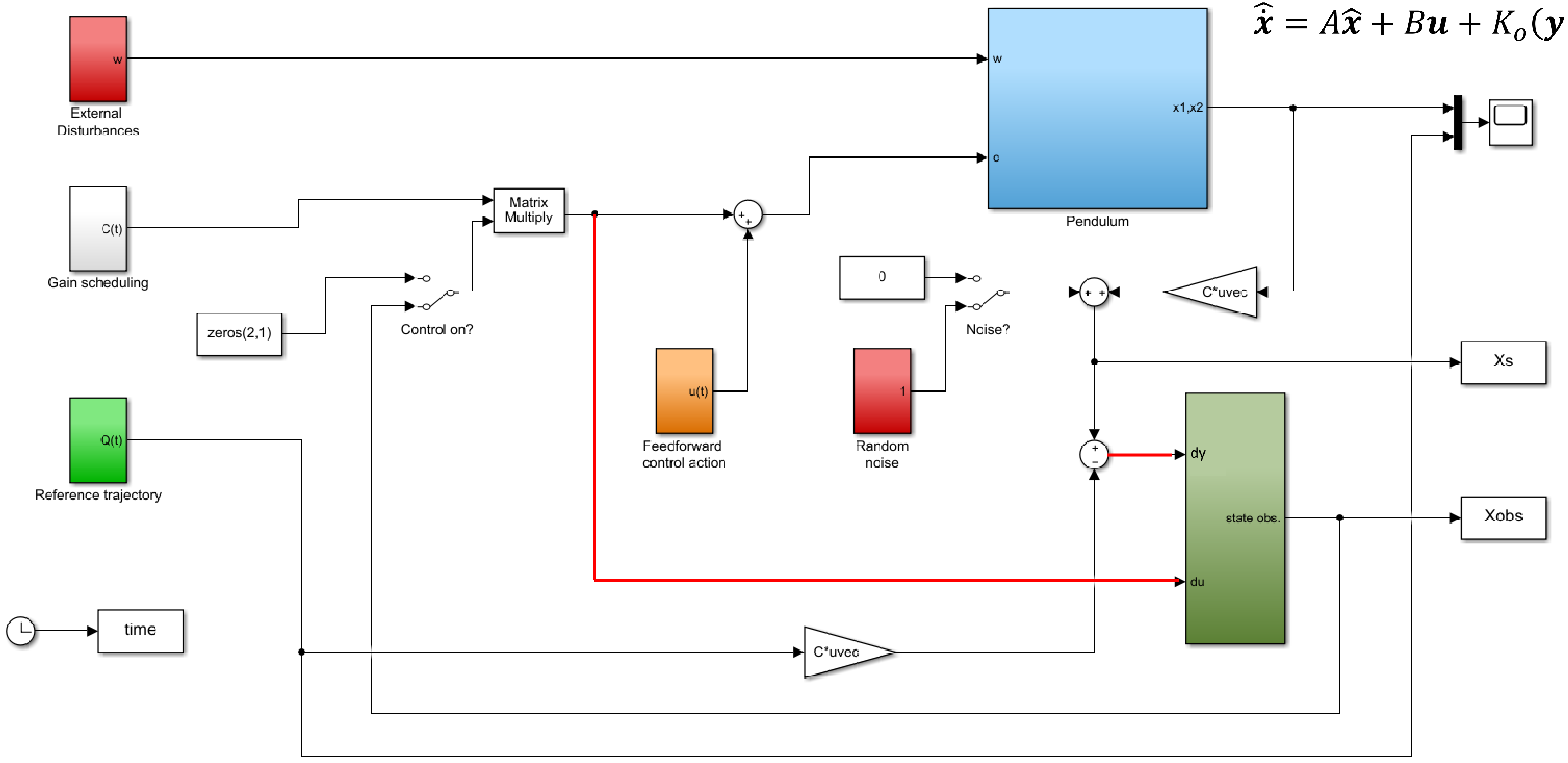
$$\begin{cases} \dot{x}_2 = -2\zeta\omega_0 x_2 - \omega_0^2 \sin(x_1) + \dfrac{c(t)}{mL^2} \\ \dot{x}_1 = x_2 \end{cases}$$

$$\boldsymbol{x} = [x_2, x_1]^T = \left[\dot{\theta}, \theta\right]^T \qquad \boldsymbol{x}_f = [0, \pi/4]^T$$

$$A = \frac{\partial f}{\partial \boldsymbol{x}} = \begin{bmatrix} -2\zeta\omega_0 & -\omega_0^2 \cos(x_1) \\ 1 & 0 \end{bmatrix} \qquad B = \frac{\partial f}{\partial u} = \begin{bmatrix} \dfrac{1}{mL^2} \\ 0 \end{bmatrix}$$

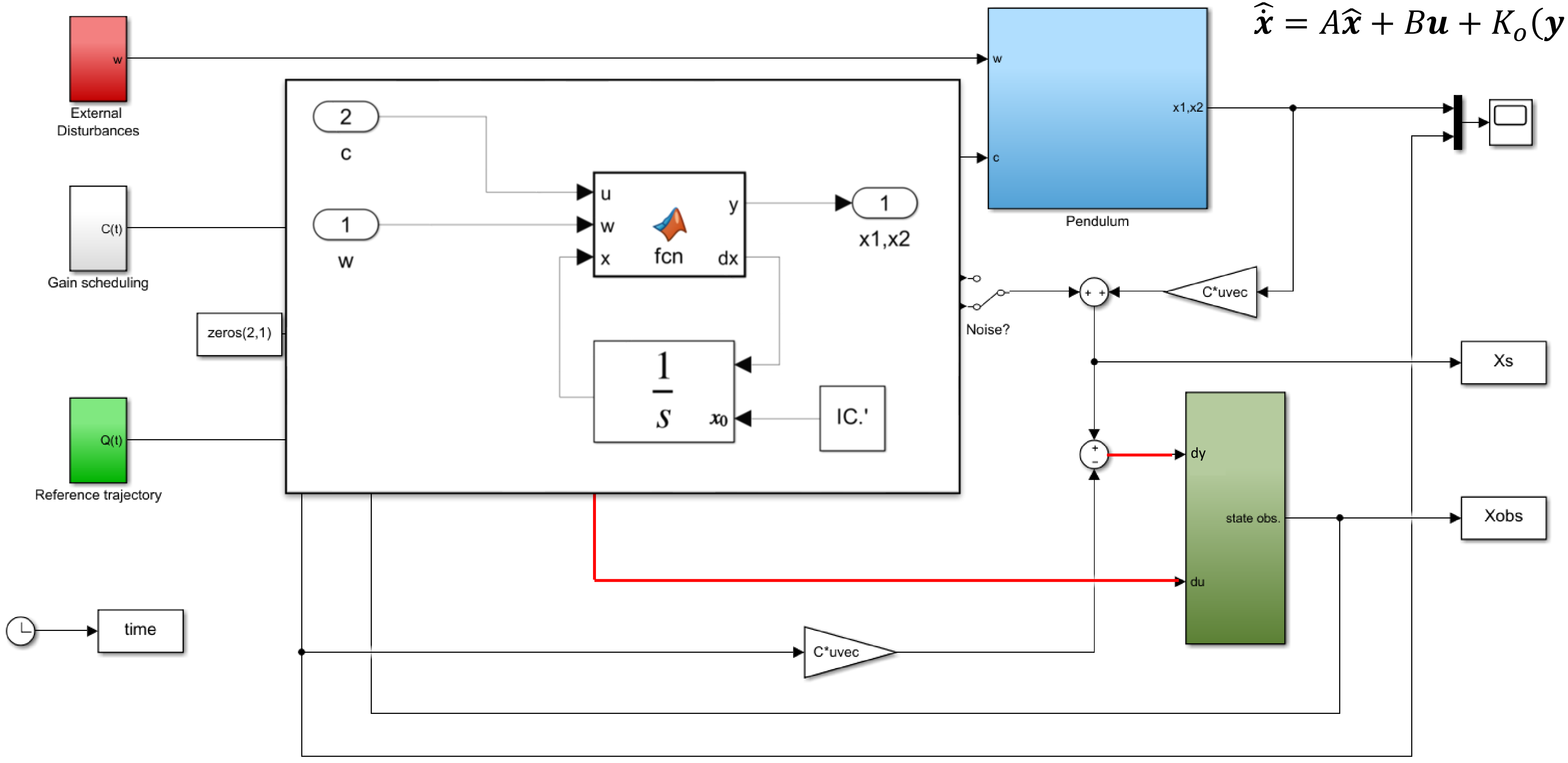
Exercise 1: KBF

$$\dot{\hat{\boldsymbol{x}}} = A\hat{\boldsymbol{x}} + B\boldsymbol{u} + K_o(\boldsymbol{y} - C\hat{\boldsymbol{x}})$$



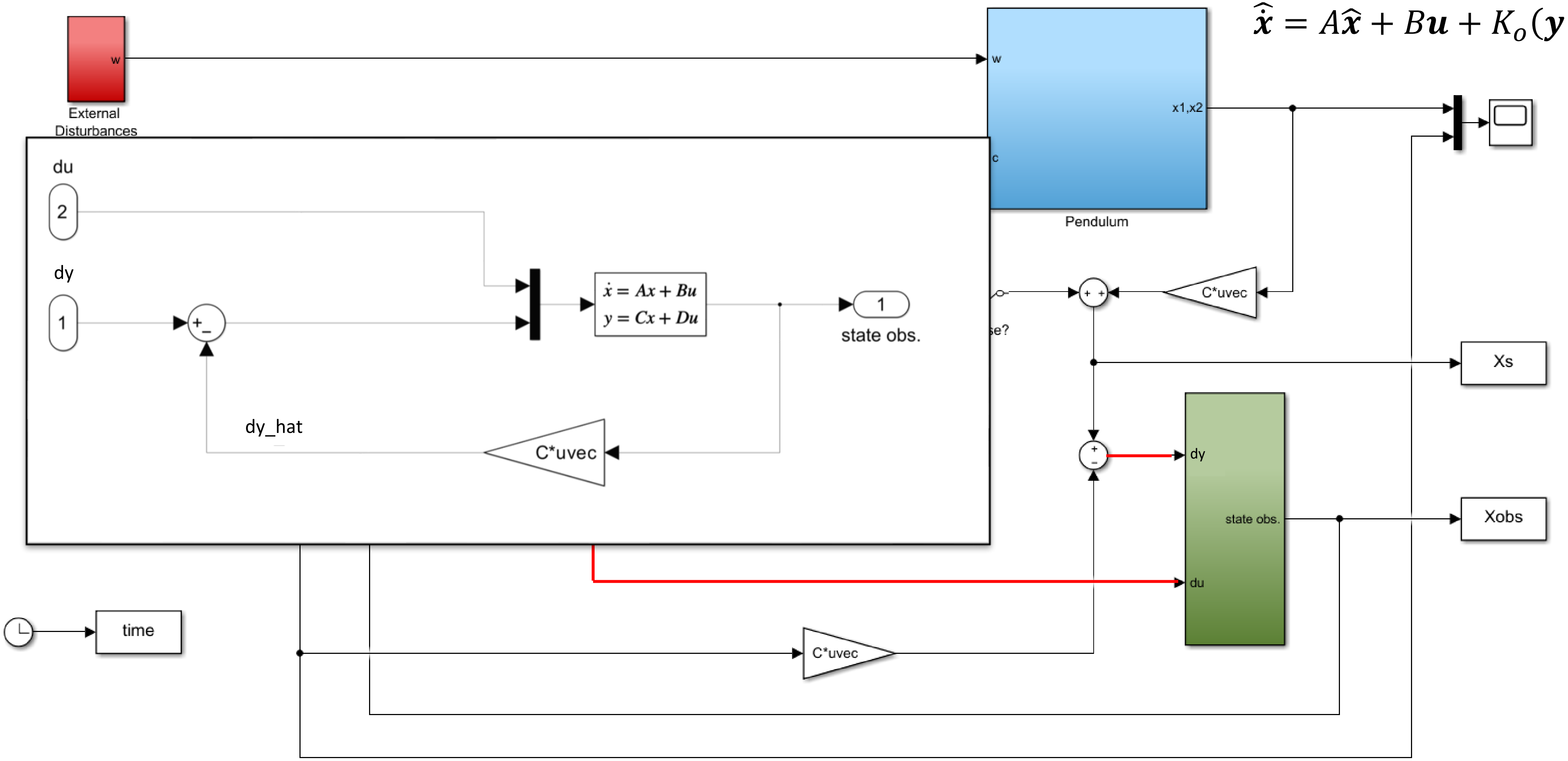
Exercise 1: KBF

$$\dot{\hat{\boldsymbol{x}}} = A\hat{\boldsymbol{x}} + B\boldsymbol{u} + K_o(\boldsymbol{y} - C\hat{\boldsymbol{x}})$$



Exercise 1: KBF

$$\dot{\hat{\boldsymbol{x}}} = A\hat{\boldsymbol{x}} + B\boldsymbol{u} + K_o(\boldsymbol{y} - C\hat{\boldsymbol{x}})$$



Exercise 1: KBF

Matlab function care.m: $[P, \boldsymbol{\rho}, K] = \boldsymbol{care}(A, B, Q, R)$

Where:

- K is the gain matrix.

- ρ are the poles of the controlled stability matrix.

$$\rho = \boldsymbol{eig}(A - BK)$$

- P is the solution of the control algebraic equation stated as:

$$Q + A^T P + PA - PB^T RBP = 0$$

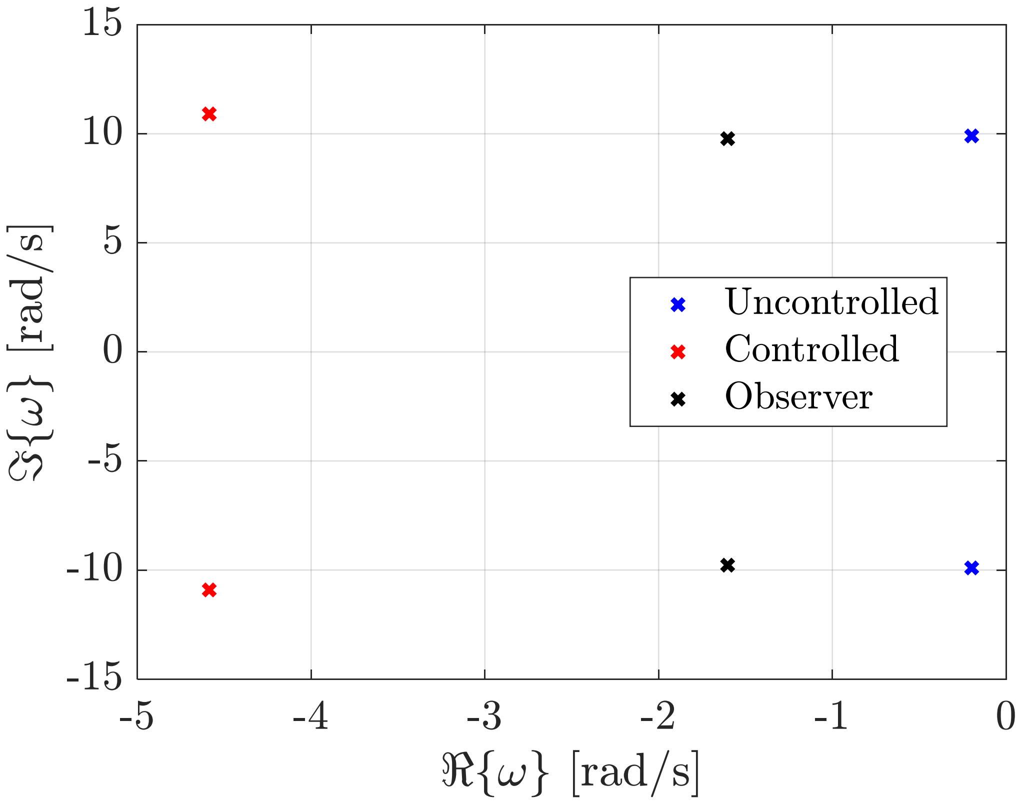
- The dual problem for the observer algebraic equation is:

$$Q + A^T P + PA - PC^T R^{-1} CP = 0$$

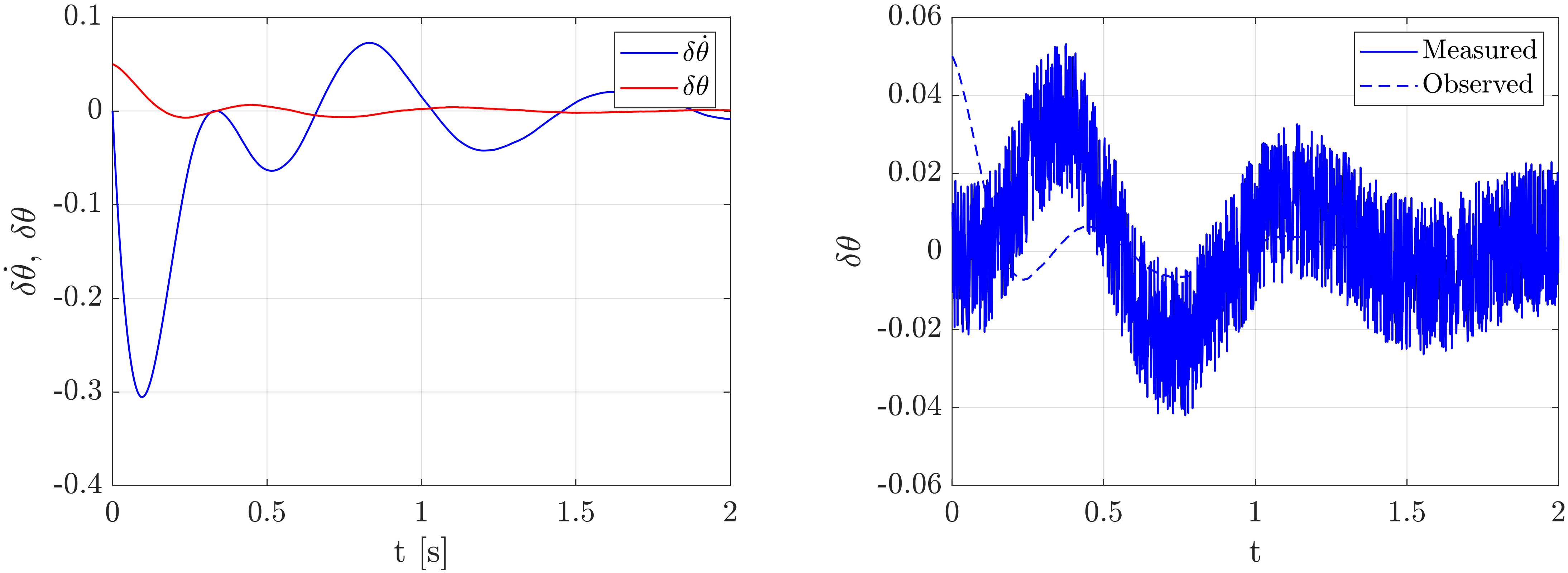
and the correspondent solution is: $[P, \boldsymbol{\rho_o}, K_o^T] = care(A^T, C^T, Q, R)$

Exercise 1: KBF

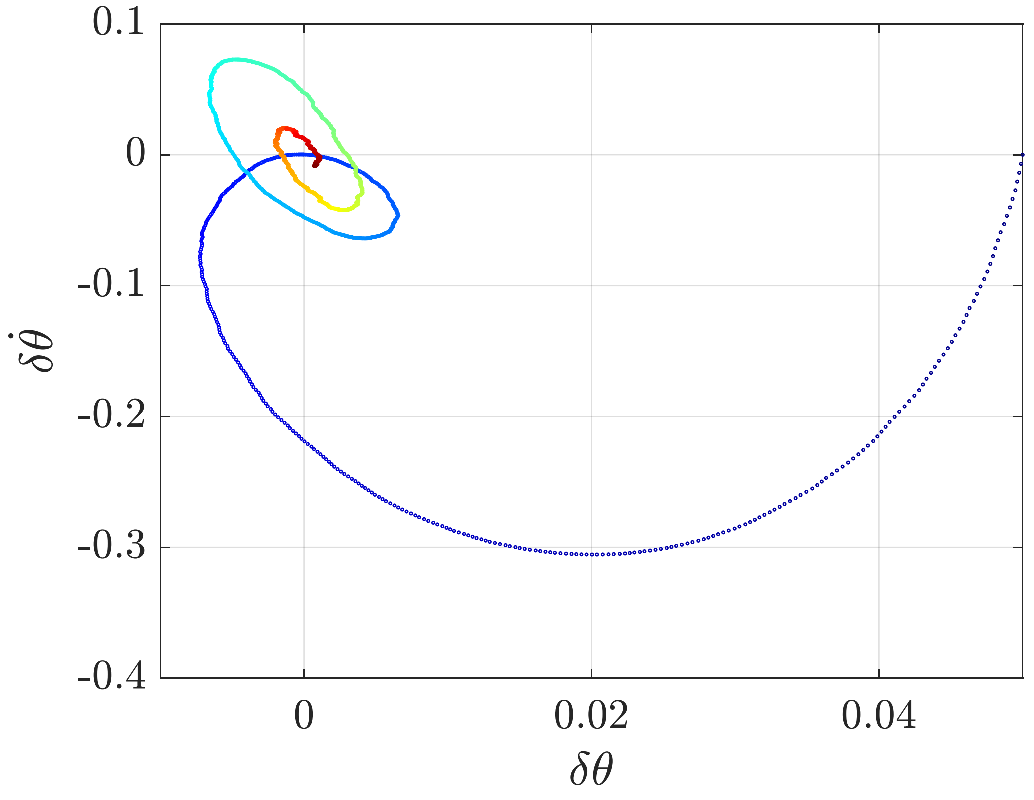
```
% Gain matrix and Poles of the controlled system
[K,PP,PolesC] = lqr(A,B,Q,R);


%% OPTIMAL OBSERVER DESIGN
[~,~,Ko] = care(A',C',Qk,Rk);
Ko = Ko';
```

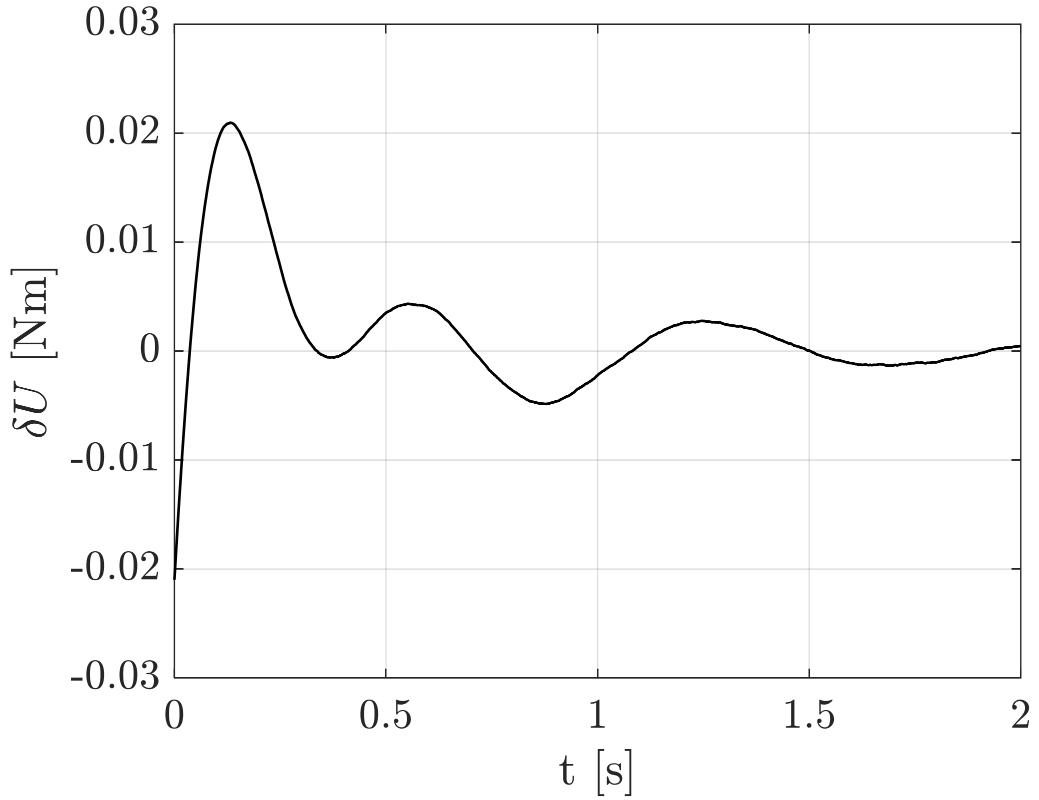


Exercise 1: Results



Exercise 1: Results





HANDS-ON