



Intro. Computer Control Systems: F9

State-feedback control and observers

Dave Zachariah

Dept. Information Technology, Div. Systems and Control

F8: Quiz!

F8: Quiz!

- 1) For an **observable** system
 - a the effect of all $x(t)$ can be observed in $y(t)$ ↑
 - b we have $\det \mathcal{O} = 0$ ↑
 - c we have stability ↓

F8: Quiz!

- 1) For an **observable** system
 - a the effect of all $x(t)$ can be observed in $y(t)$ ↑
 - b we have $\det \mathcal{O} = 0$ ↑
 - c we have stability ↓

- 2) If a state-space form of $G(s)$ is a **minimal realization**,
 - a A :s eigenvalues < $G(s)$:s poles ↑
 - b A :s eigenvalues = $G(s)$:s poles ↑
 - c there exist more compact state-space forms ↓



F8: Quiz!

- 1) For an **observable** system
 - a the effect of all $x(t)$ can be observed in $y(t)$ ↑
 - b we have $\det \mathcal{O} = 0$ ↑
 - c we have stability ↓
- 2) If a state-space form of $G(s)$ is a **minimal realization**,
 - a A :s eigenvalues < $G(s)$:s poles ↑
 - b A :s eigenvalues = $G(s)$:s poles ↑
 - c there exist more compact state-space forms ↓
- 3) For a **controllable** system with state-feedback control
 - a the closed-loop system is stable ↑
 - b the poles of the closed-loop system can be designed arbitrarily ↑
 - c no information about the system is required ↓

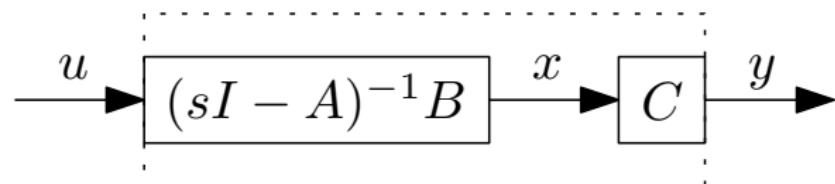


State-feedback control

State-feedback control

State-space form of linear time-invariant system

$$\begin{aligned}\dot{x} &= \textcolor{blue}{A}x + \textcolor{blue}{B}u \\ y &= \textcolor{blue}{C}x\end{aligned}\quad \Rightarrow \quad G(s) = C(sI - A)^{-1}B$$



State-feedback control

Controller using state feedback

$$u = -\textcolor{magenta}{L}x + \ell_0 r$$

gives **closed-loop system**

$$\begin{aligned}\dot{x} &= (A - B\textcolor{magenta}{L})x + B\ell_0 r \\ y &= Cx\end{aligned}$$

where r is the reference signal.

State-feedback control

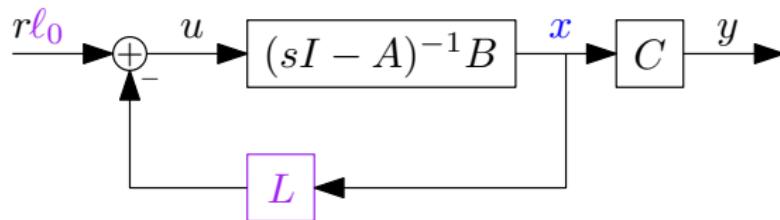
Controller using state feedback

$$u = -\textcolor{magenta}{L}x + \ell_0 r$$

gives **closed-loop system**

$$\begin{aligned}\dot{x} &= (A - B\textcolor{magenta}{L})x + B\ell_0 r \\ y &= Cx\end{aligned}$$

where r is the reference signal.

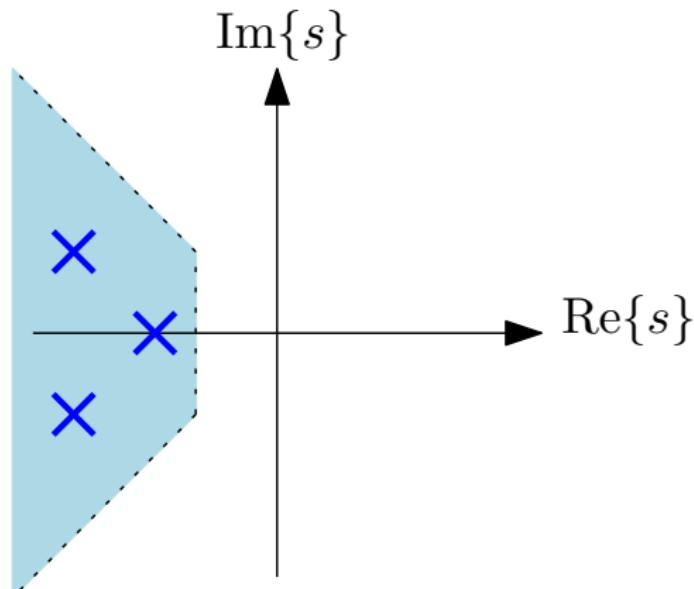


$$G_c(s) = C(sI - A + B\textcolor{magenta}{L})^{-1}B\ell_0$$

Pole placement

Rules of thumb for designing L

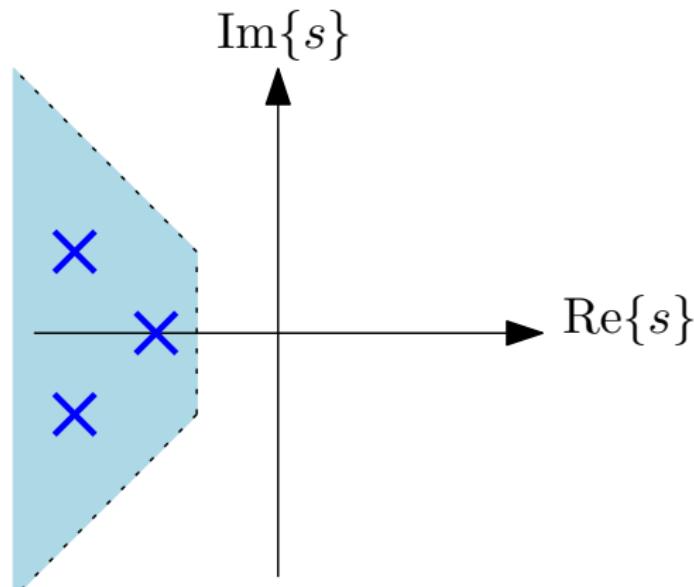
Eigenvalues/poles given by $\det(sI - A + BL) = 0$, which we can design



Pole placement

Rules of thumb for designing L

Eigenvalues/**poles** given by $\det(sI - A + BL) = 0$, which we can *design*



Distance to the origin: Quick system but also sensitive to disturbances



Estimating the states via simulation

Estimating the states

Via simulation

- ▶ Controller

$$u = -L\mathbf{x} + \ell_0 r$$

requires states \mathbf{x} which are often **unknown**.

Estimating the states

Via simulation

- ▶ Controller

$$u = -L\mathbf{x} + \ell_0 r$$

requires states \mathbf{x} which are often **unknown**.

- ▶ In practice, feedback using

$$u = -L\hat{\mathbf{x}} + \ell_0 r$$

where $\hat{\mathbf{x}}$ is an estimate of \mathbf{x} .

Estimating the states

Via simulation

- ▶ Controller

$$u = -L\mathbf{x} + \ell_0 r$$

requires states \mathbf{x} which are often **unknown**.

- ▶ In practice, feedback using

$$u = -L\hat{\mathbf{x}} + \ell_0 r$$

where $\hat{\mathbf{x}}$ is an estimate of \mathbf{x} .

- ▶ *Naive idea:* Estimate x by *simulating* the states

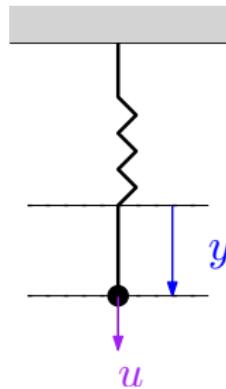
$$\dot{\hat{\mathbf{x}}} = A\hat{\mathbf{x}} + Bu, \quad \hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_0$$

where $\hat{\mathbf{x}}_0$ is an **initial guess**.

Build intuition from simple systems

State estimation via simulation

Ex.: Damper



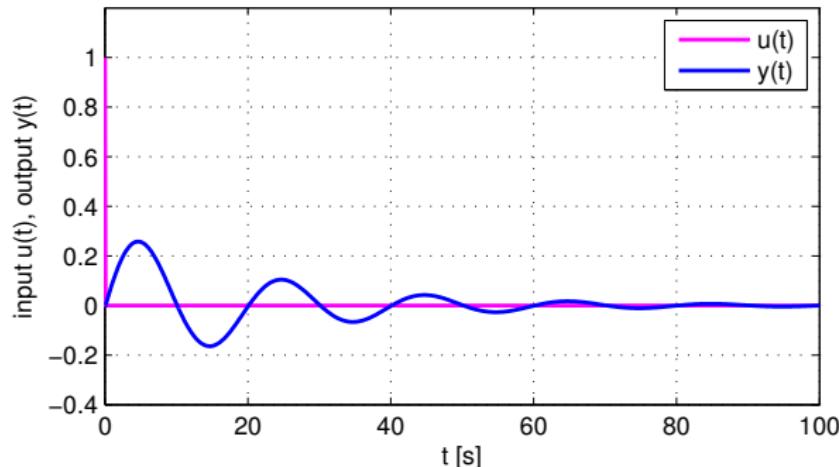
State-space form:

$$\begin{aligned}\dot{x}(t) &= \begin{bmatrix} 0 & 1 \\ -k/m & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} u(t), \quad x(0) = x_0 \\ y(t) &= [1 \quad 0] x(t)\end{aligned}$$

Build intuition from simple systems

State estimation via simulation

Example using impulse u



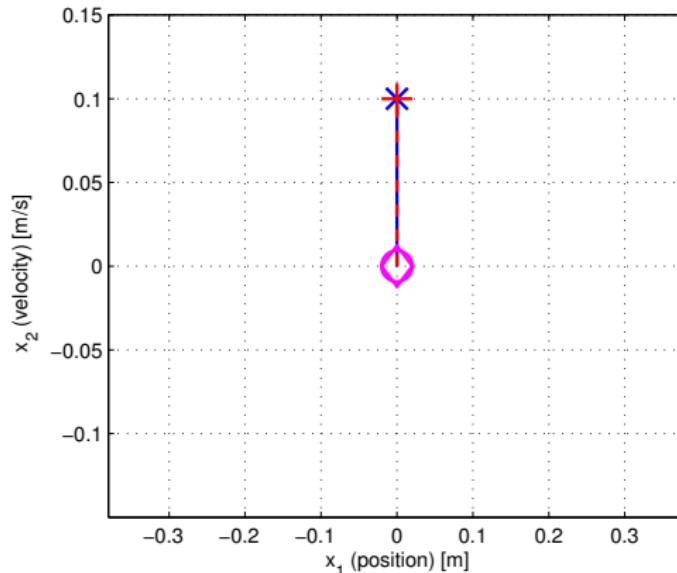
System with unknown initial state x_0

Build intuition from simple systems

State estimation via simulation

Naive estimate using *perfect* initial guess:

$$\dot{\hat{x}} = A\hat{x} + Bu, \quad \boxed{\hat{x}_0 = x_0}$$



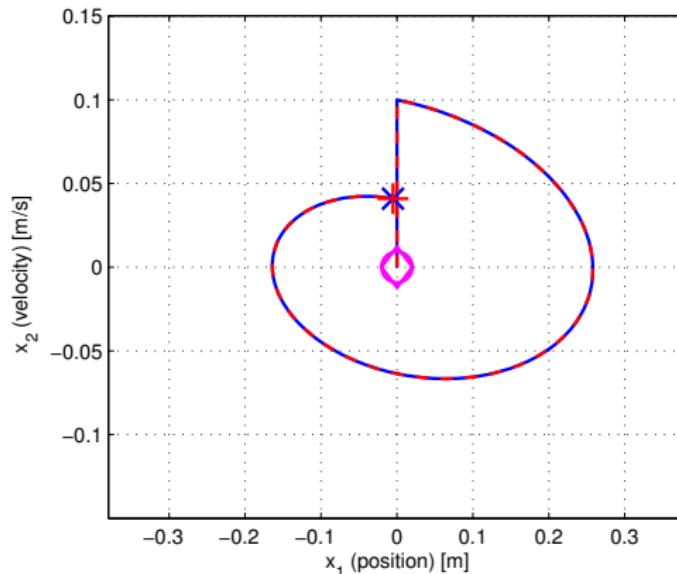
x versus \hat{x} at $t = 0^+$

Build intuition from simple systems

State estimation via simulation

Naive estimate using *perfect* initial guess:

$$\dot{\hat{x}} = A\hat{x} + Bu, \quad \hat{x}_0 = x_0$$



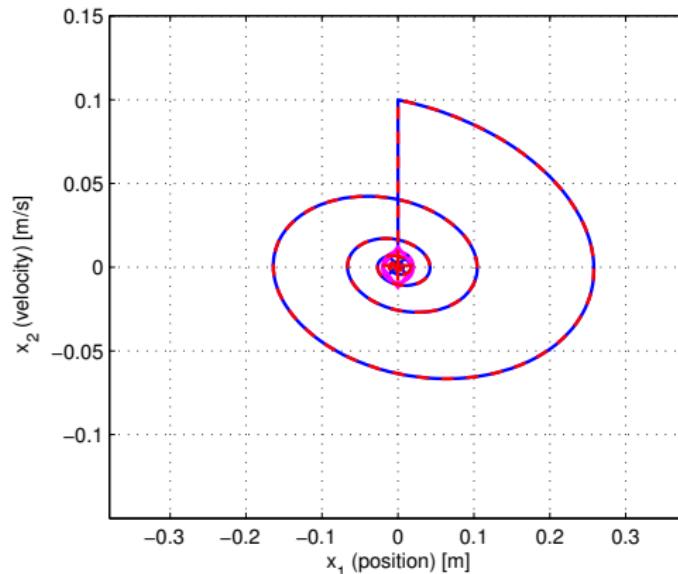
x versus \hat{x} at $t = 20$

Build intuition from simple systems

State estimation via simulation

Naive estimate using *perfect* initial guess:

$$\dot{\hat{x}} = A\hat{x} + Bu, \quad \boxed{\hat{x}_0 = x_0}$$



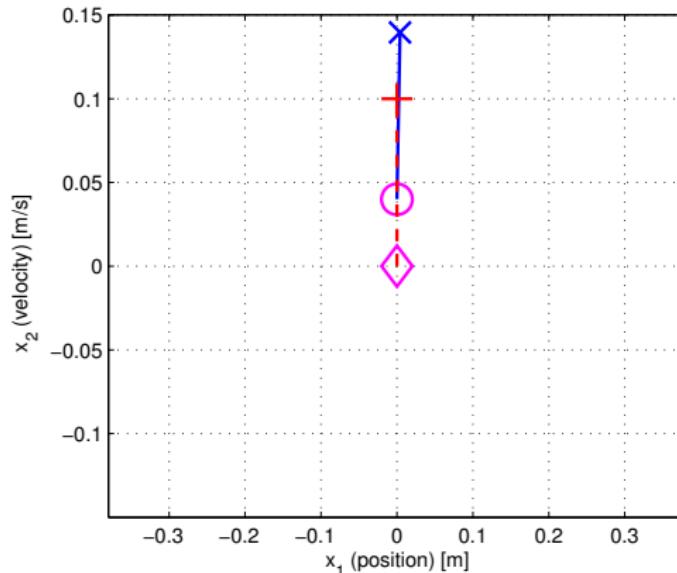
$\textcolor{blue}{x}$ versus $\textcolor{red}{\hat{x}}$ at $t = 100$

Build intuition from simple systems

State estimation via simulation

Naive estimate using *wrong* initial guess:

$$\dot{\hat{x}} = A\hat{x} + Bu, \quad \boxed{\hat{x}_0 \neq x_0}$$



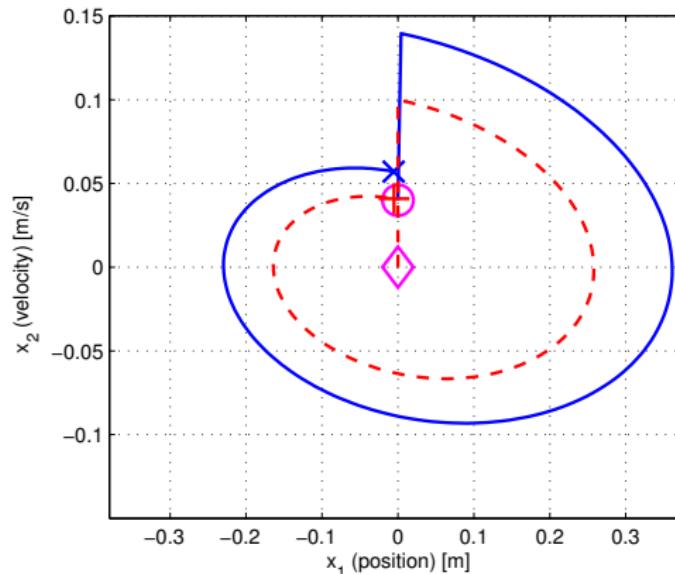
x versus \hat{x} at $t = 0^+$

Build intuition from simple systems

State estimation via simulation

Naive estimate using *wrong* initial guess:

$$\dot{\hat{x}} = A\hat{x} + Bu, \quad \boxed{\hat{x}_0 \neq x_0}$$



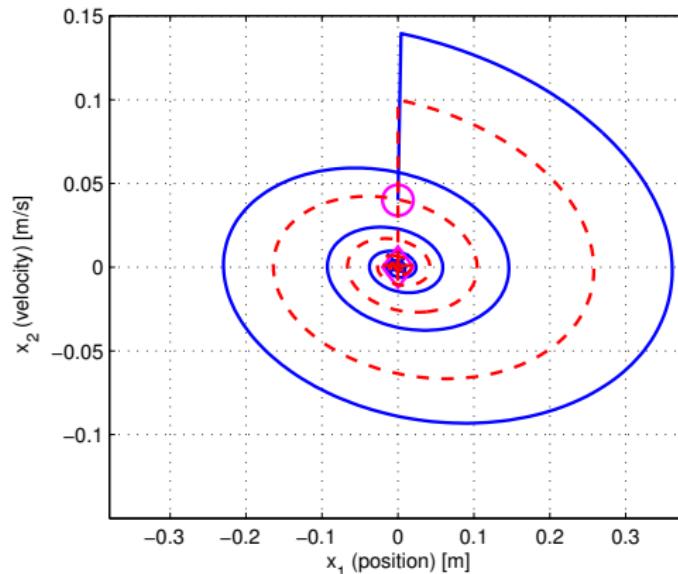
x versus \hat{x} at $t = 20$

Build intuition from simple systems

State estimation via simulation

Naive estimate using *wrong* initial guess:

$$\dot{\hat{x}} = A\hat{x} + Bu, \quad \boxed{\hat{x}_0 \neq x_0}$$



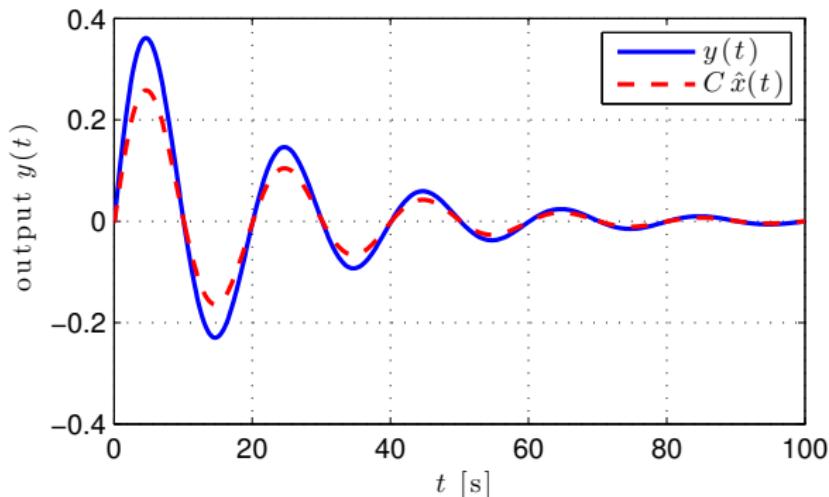
$\textcolor{blue}{x}$ versus $\textcolor{red}{\hat{x}}$ at $t = 100$

Build intuition from simple systems

State estimation via simulation

$\textcolor{blue}{x}$ och \hat{x} correspond to different outputs:

$$\textcolor{blue}{y} = C\textcolor{blue}{x} \quad \text{versus} \quad \hat{y} = C\hat{x}$$





Estimating the states via observer

Estimating the states

Correcting the state estimates

- ▶ *Idea:* Feedback the prediction error $y - C\hat{x}$ to correct \hat{x}
- ▶ *Observer:* an estimator with a *correction term*

$$\dot{\hat{x}} = A\hat{x} + Bu + \underbrace{K(y - C\hat{x})}_{\text{correction}}, \quad \hat{x}(0) = \hat{x}_0$$

Estimating the states

Correcting the state estimates

- ▶ *Idea:* Feedback the prediction error $y - C\hat{x}$ to correct \hat{x}
- ▶ *Observer:* an estimator with a *correction term*

$$\dot{\hat{x}} = A\hat{x} + Bu + \underbrace{K(y - C\hat{x})}_{\text{correction}}, \quad \hat{x}(0) = \hat{x}_0$$

- ▶ Using matrix

$$K = \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}$$

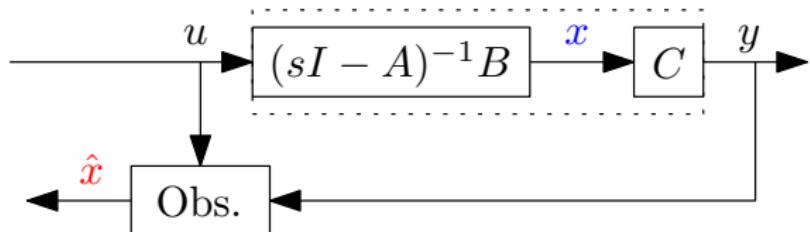
we can *design* the estimator.

Estimating the states

Correcting the state estimates

- ▶ *Idea:* Feedback the prediction error $y - C\hat{x}$ to correct \hat{x}
- ▶ *Observer:* an estimator with a *correction term*

$$\dot{\hat{x}} = A\hat{x} + Bu + \underbrace{K(y - C\hat{x})}_{\text{correction}}, \quad \hat{x}(0) = \hat{x}_0$$

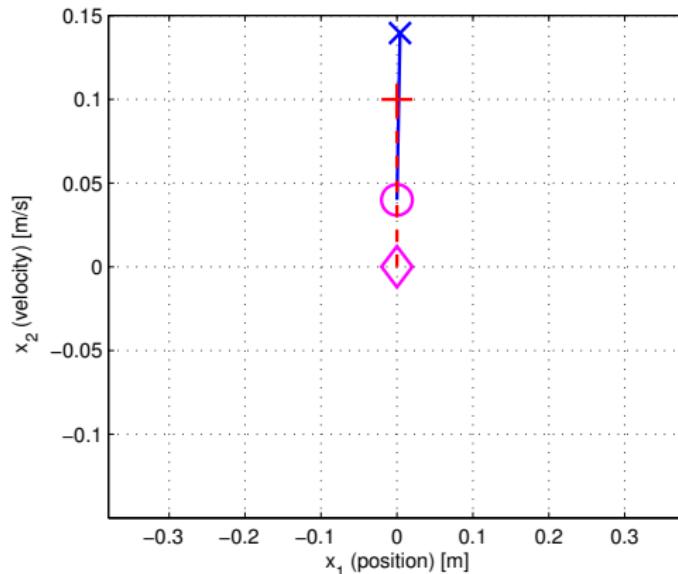


Build intuition using simple systems

State estimation using observer

Estimation using observer:

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}), \quad \hat{x}_0 \neq x_0$$



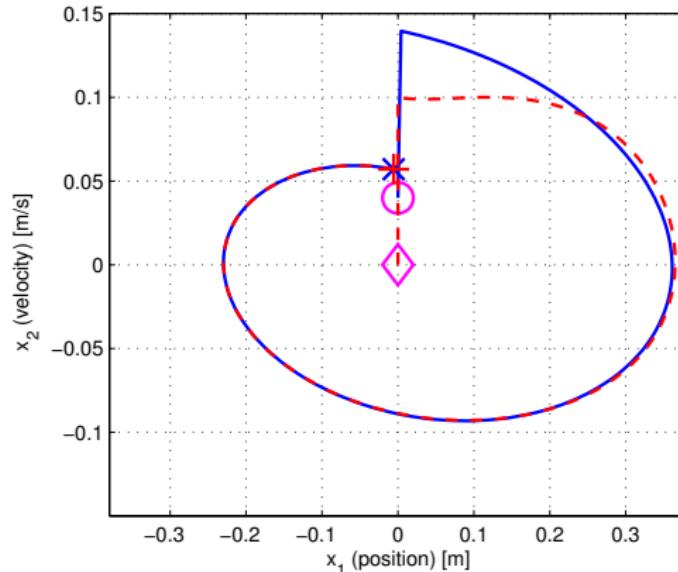
x versus \hat{x} at $t = 0^+$

Build intuition using simple systems

State estimation using observer

Estimation using observer:

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}), \quad \hat{x}_0 \neq x_0$$



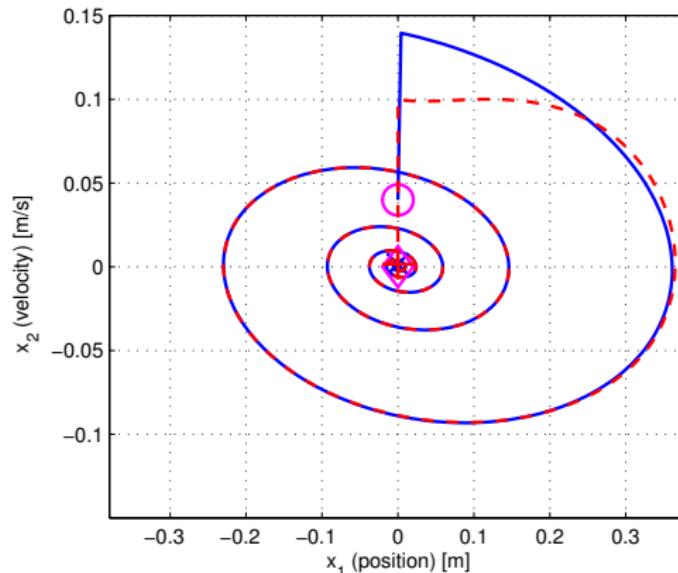
x versus \hat{x} at $t = 20$

Build intuition using simple systems

State estimation using observer

Estimation using observer:

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}), \quad \hat{x}_0 \neq x_0$$



$\textcolor{blue}{x}$ versus $\textcolor{red}{\hat{x}}$ at $t = 100$

State estimation

Estimation error and observability

Estimation error:

$$\tilde{x} \triangleq x - \hat{x}$$

[Board: derive evolution of estimation errors]

State estimation

Estimation error and observability

Estimation error:

$$\tilde{x} \triangleq x - \hat{x}$$

[Board: derive evolution of estimation errors]

Result

Errors of observer described as system

$$\tilde{x}(t) = e^{(A - \textcolor{blue}{K}C)t} \tilde{x}(0)$$

and therefore $\|\tilde{x}(t)\|$ decays at a rate given by maximum

$$\operatorname{Re}\{\tilde{s}_i\}$$

where \tilde{s}_i are observer poles/eigenvalues of $(A - \textcolor{blue}{K}C)$.

State estimation

Estimation error and observability

Estimation error:

$$\tilde{x} \triangleq x - \hat{x}$$

Result 9.2

State-space form is **observable** (cf. $\det \mathcal{O} \neq 0$) \Rightarrow matrix K can be chosen such that \tilde{x} vanish arbitrarily quick

State estimation

Estimation error and observability

Estimation error:

$$\tilde{x} \triangleq x - \hat{x}$$

Result 9.2

State-space form is **observable** (cf. $\det \mathcal{O} \neq 0$) \Rightarrow matrix K can be chosen such that \tilde{x} vanish arbitrarily quick

- K is solved by polynomial $\det(sI - A + KC) = 0$ with desired roots in left halfplane

State estimation

Estimation error and observability

Estimation error:

$$\tilde{x} \triangleq x - \hat{x}$$

Result 9.2

State-space form is **observable** (cf. $\det \mathcal{O} \neq 0$) \Rightarrow matrix K can be chosen such that \tilde{x} vanish arbitrarily quick

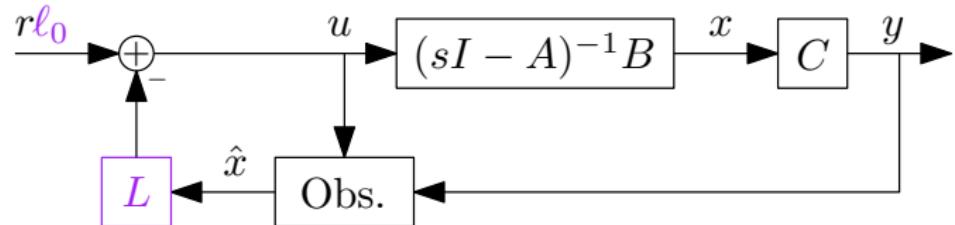
- ▶ K is solved by polynomial $\det(sI - A + KC) = 0$ with desired roots in left halfplane
- ▶ Quick observer \hat{x} is however sensitive to measurement noise!



Combining feedback with estimated states

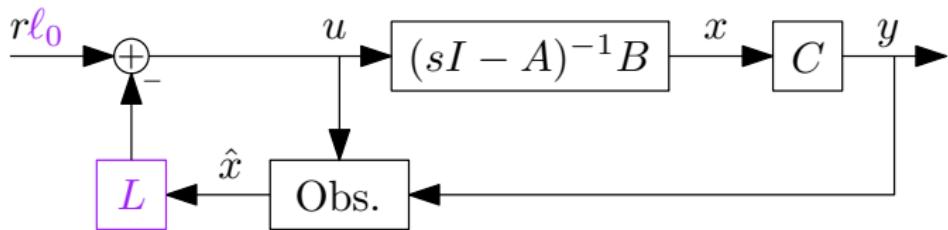
Feedback using estimated states

Controller in the Laplace domain



Feedback using estimated states

Controller in the Laplace domain

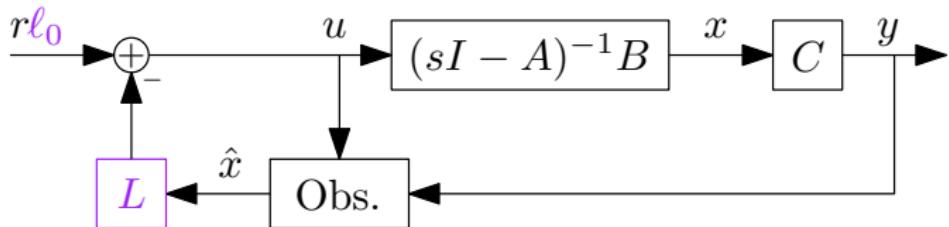


System and controller with observer:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \quad \text{and} \quad \begin{cases} u = -L\hat{x} + \ell_0 r \\ \dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}) \end{cases}$$

Feedback using estimated states

Controller the Laplace domain



Controller with observer:

$$\mathcal{L} : \begin{cases} U(s) &= -L\hat{X}(s) + \ell_0 R(s) \\ s\hat{X}(s) &= AX(s) + BU(s) + K(Y(s) - CX(s)) \end{cases}$$

[Board: solve for controller]

Feedback using estimated states

General linear feedback control

General linear feedback form, ch.9.5 G&L

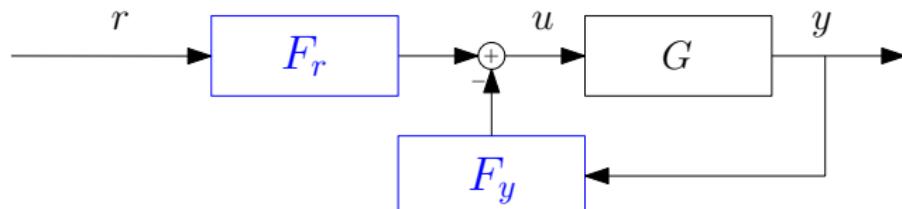
Controller with observer can be written as

$$U(s) = F_r(s)R(s) - F_y(s)Y(s),$$

where

$$F_r(s) = (1 - L(sI - A + KC + BL)^{-1}B)\ell_0$$

$$F_y(s) = L(sI - A + KC + BL)^{-1}K$$

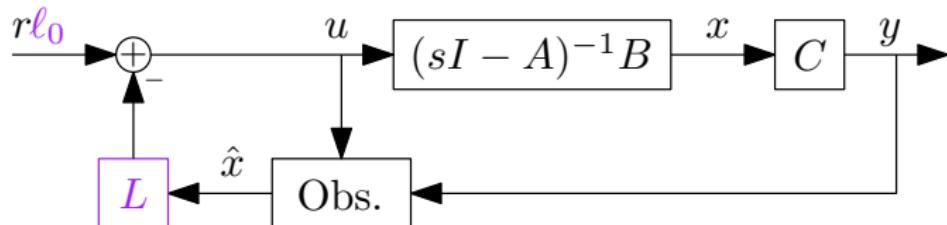




Resulting closed-loop system

Feedback using estimated states

Effect of estimation error



Study system and controller with observer:

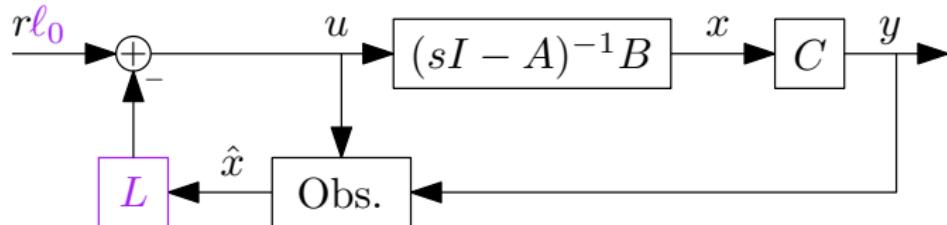
$$\begin{cases} \dot{\hat{x}} = Ax + Bu \\ y = Cx \end{cases} \quad \text{and} \quad u = -L\hat{x} + \ell_0 r$$

by substituting $\hat{x} = x - \tilde{x}$

[Board: derive the closed-loop system with estimation error \tilde{x}]

Feedback using estimated states

Effect of estimation error



Yields **closed-loop system**:

effect of estimation error

$$\dot{x} = (A - BL)x + \widehat{BL\tilde{x}} + B\ell_0 r$$

$$y = Cx$$

with **additional error states**

$$\dot{\tilde{x}} = (A - KC)\tilde{x} \rightarrow 0$$

[Board: write the closed-loop system in state-space form]

Feedback using estimated states

The closed-loop system with observer

The closed-loop system with estimation error can be written as

$$\begin{bmatrix} \dot{x} \\ \dot{\tilde{x}} \end{bmatrix} = \underbrace{\begin{bmatrix} A - BL & BL \\ 0 & A - KC \end{bmatrix}}_{\tilde{A}} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + \underbrace{\begin{bmatrix} B \\ 0 \end{bmatrix}}_{\tilde{B}} \ell_0 r$$
$$y = \underbrace{\begin{bmatrix} C & 0 \end{bmatrix}}_{\tilde{C}} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix}$$

with extended state vector.

This yields transfer function from r to y :

$$\Rightarrow G_c(s) = \tilde{C}(sI - \tilde{A})^{-1}\tilde{B}$$

Feedback using estimated states

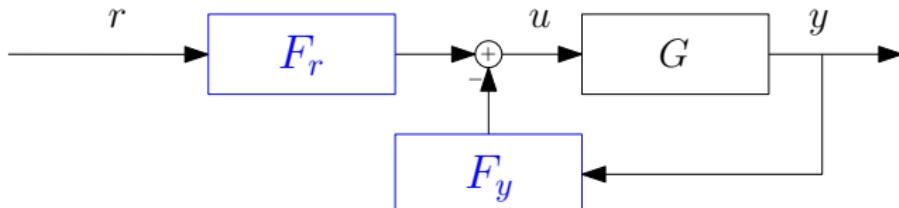
The closed-loop system with observer

Closed-loop system transfer function, ch.9.5 G&L

Insert matrices \tilde{A} , \tilde{B} and \tilde{C} yields

$$\begin{aligned}G_c(s) &= \tilde{C}(sI - \tilde{A})^{-1}\tilde{B} \\&= C(sI - A + BL)^{-1}BL_0\end{aligned}$$

with same poles as if states were known and K is gone!



Summary and recap

- ▶ Rules of thumb for pole placement
- ▶ Estimation using observer
- ▶ Feedback using estimated states
- ▶ Closed-loop system with observer