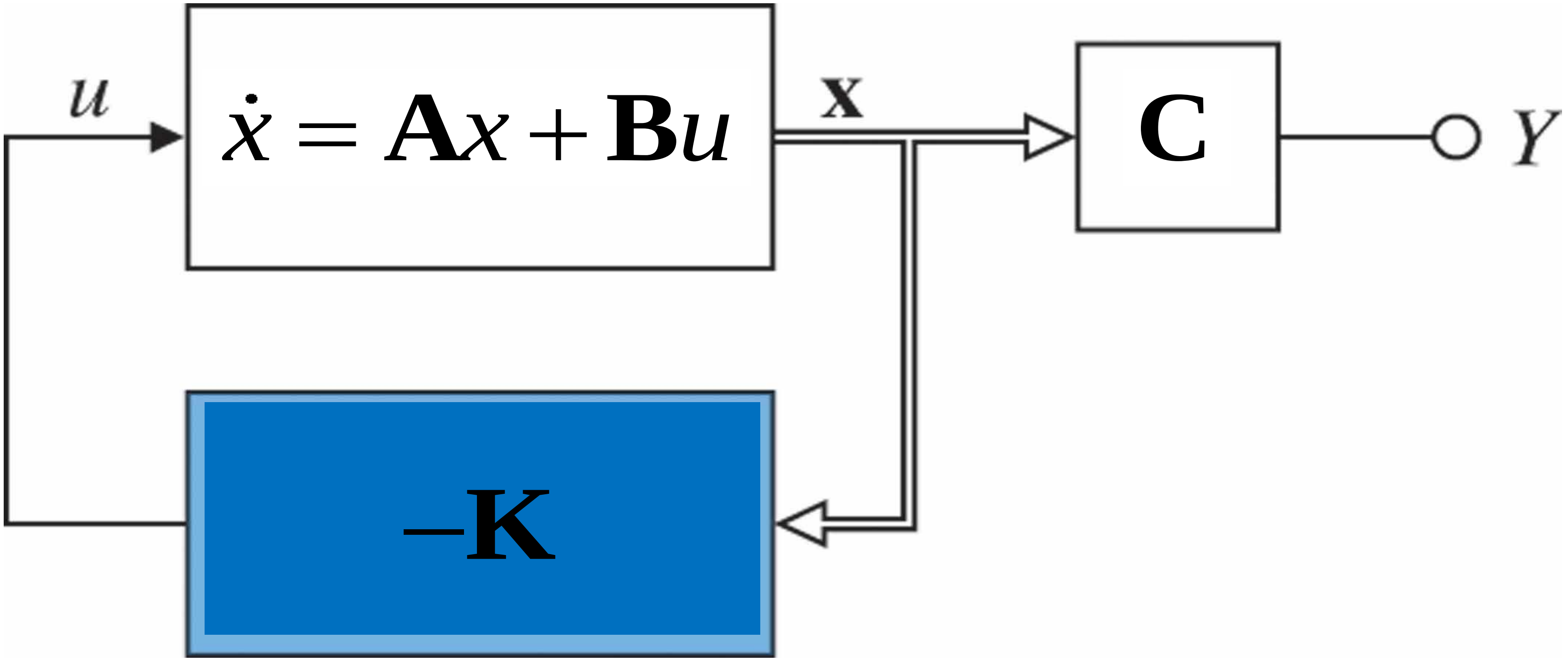
State feedback control

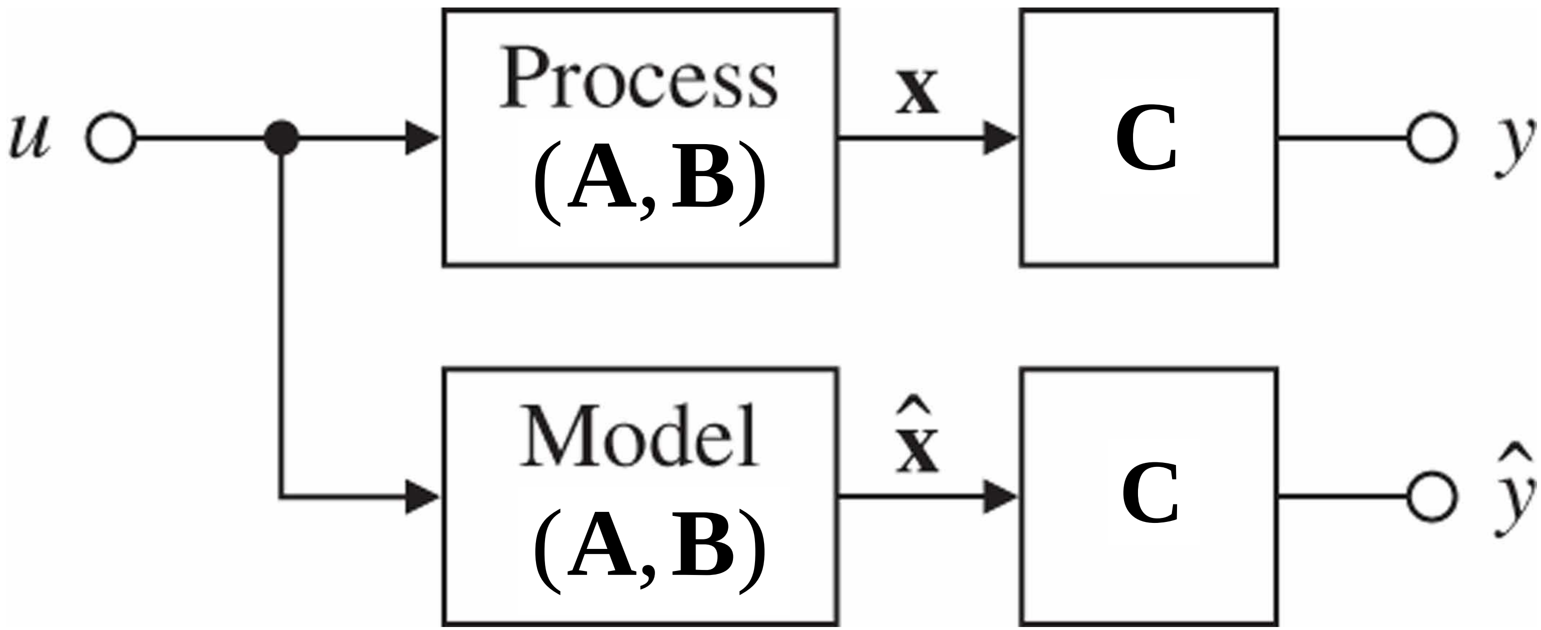


Observer design

State feedback encompasses the essence of many fundamental ideas in control design and lies at the core of many design strategies. However, this approach requires that all states be measured. In most cases, this is an unrealistic requirement. For that reason, the idea of observers is introduced next, as a mechanism for estimating the states from the available measurements.

Note: The terms *observer* and *estimator* are used interchangeably here.

Open-loop observer



- Model and process outputs are only identical if **model and process** are identical, **initial conditions** are know, and **disturbances** are zero!

Observers (Estimators)

Consider the state space model

$$\dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}u(t); \quad x(0) = x_0$$

$$y(t) = \mathbf{C}x(t)$$

Define the **error in the estimated state** as

$$\tilde{x}(t) = x(t) - \hat{x}(t)$$

where $\hat{x}(t)$ is the ***state estimate***.

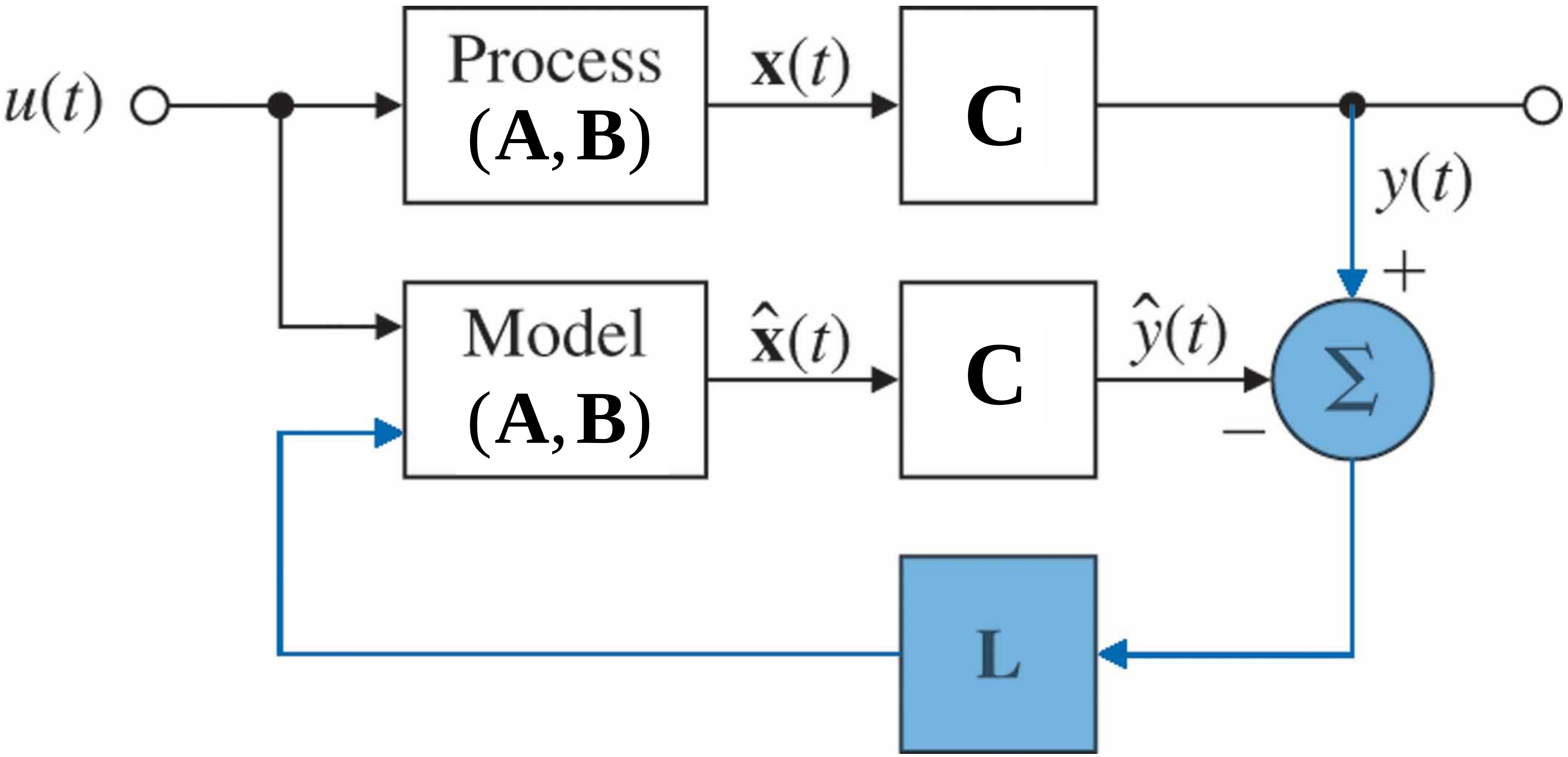
Then the dynamics of the state error are given by

$$\dot{\tilde{x}}(t) = \mathbf{A}\tilde{x}(t) \text{ with } \tilde{x}(t) = x(0) - \hat{x}(0)$$

- The error converges to zero for stable systems (**A** stable), but we have no ability to influence the rate at which the state estimate converges to the true state.
- Furthermore, the error is converging to zero at the same rate as the natural dynamics **A**, which may not be satisfactory.
- We can now invoke the Golden Rule:

When in trouble, use feedback

Closed-loop observer



- We can feed back the difference between the measured and the estimated output

$$v(t) = y(t) - C\hat{x}(t)$$

using a feedback vector **L**

$$\mathbf{L} = \begin{bmatrix} l_1 & l_2 & \cdots & l_n \end{bmatrix}^T$$

- The equation for this scheme is

$$\dot{\hat{x}}(t) = \mathbf{A}\hat{x}(t) + \mathbf{B}u(t) + \mathbf{L}\left(y(t) - \mathbf{C}\hat{x}(t)\right)$$

Lemma: The estimation error $\tilde{x}(t)$ satisfies

$$\dot{\tilde{x}}(t) = \left(\mathbf{A} - \mathbf{LC}\right)\tilde{x}(t)$$

Moreover, provided the model is **completely observable**, then the eigenvalues of (**A** - **LC**) can be arbitrarily assigned by choice of **L**.

The characteristic equation of the error is now given by

$$\det\left[s\mathbf{I} - \left(\mathbf{A} - \mathbf{LC}\right)\right] = 0$$

- If we choose **L** so that (**A-LC**) has stable and reasonable fast eigenvalues, then the state error will decay to zero and remain there, independent of the known forcing function *u(t)*.
- This means that the estimated state will converge to the true state, regardless of initial conditions.
- **Note:** we have assumed that **A**, **B**, and **C** are identical in the physical plant and in the estimator. If we don't have an accurate model, then we can at least choose **L** so that the error system is still stable.

Example: Observer design

$$G(s)=\frac{B(s)}{A(s)}=\frac{2}{s^2+7s+12}=\frac{2}{(s+4)(s+3)}$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\mathbf{A}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\mathbf{B}u=\begin{bmatrix}-7 & 1\\ -12 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}0\\ 2\end{bmatrix}u$$

$$y=\mathbf{C}\begin{bmatrix}x_1\\ x_2\end{bmatrix}=\begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

Example: Observer design

- Using Matlab, we can verify that both descriptions are equivalent:
- Implement the transfer function:
  - B_tf = [2]; A_tf=[1 7 12];
  - sys1= tf(B_tf, A_tf)
- Implement the state space description in **observer canonical form**:
  - A= [-7 1; -12 0]; B=[0;2]; C=[1 0]; D=0;
  - sys2=ss (A,B,C,D)
- Compare both implementations: ltiview(sys1,sys2)

Example: Observer design

- The poles of the original system are at -4 and -3
- Choose observer gains such that the closed loop poles are located at -50 and -60
  - $A_{obsv}(s)=(s+50)(s+60)=s^2+110s+3000$
- Solution: $L = \begin{bmatrix} 103 \\ 2988 \end{bmatrix}$
- This can be verified by calculating the eigenvalues of the closed-loop observer matrix:
  - eig(A-LC)

Observer design

- Thus, to calculate the observer feedback vector **L**, we need to
  1. Calculate the desired observer polynomial
     $$A_{obsv}(s) = s^n + \alpha_n s^{n-1} + \ldots + \alpha_2 s + \alpha_1$$
  2. transform the system into observer canonical form,
  3. Compare elements of the matrix **A** with the corresponding elements of $A_{obsv}(s)$ to calculate the elements of **L**

Observer design: Matlab commands

- We can also use Ackermann's formula for the observer design.
- However, we need to use the transpose of **A** and **C**:
  - L = acker(A',C',P_obsv)';
- *P_obsv* is a vector giving the desired locations of the observer poles
- We can also use the alternative *place* function
  - L = place(A',C',P_obsv)'