ELEC4410

Control Systems Design

Lecture 18: State Feedback Tracking and State Estimation

Julio H. Braslavsky

julio@ee.newcastle.edu.au

School of Electrical Engineering and Computer Science
The University of Newcastle

Outline

- Regulation and Tracking
- Robust Tracking: Integral Action
- State Estimation
- A tip for Lab 2

State Feedback

In the last lecture we introduced **state feedback** as a technique for **eigenvalue** placement. Briefly, given the **open-loop** state equation

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t)\\ y(t) &= Cx(t),\end{aligned}$$

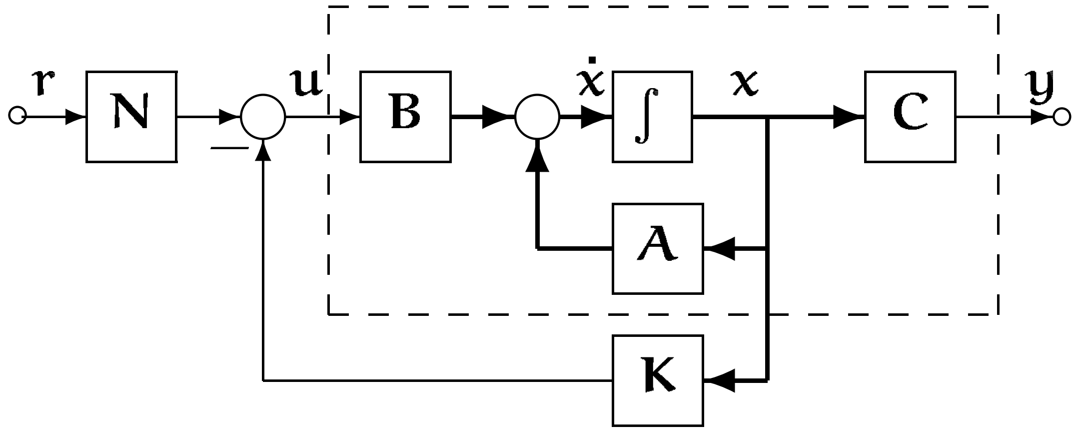
we apply the control law $u(t) = Nr(t) - Kx(t)$ and obtain the **closed-loop** state equation

$$\begin{aligned}\dot{x}(t) &= (A - BK)x(t) + BNr(t)\\ y(t) &= Cx(t),\end{aligned}$$

If the system is **controllable**, by appropriately designing K, we are able to place the eigenvalues of $(A - BK)$ at any desired locations.

Regulation and Tracking

The associated block diagram is the following



Two typical control problems of interest:

- The **regulator problem**, in which $r = 0$ and we aim to keep $\lim_{t\to\infty} y(t) = 0$ (i.e., a pure **stabilisation problem**)
- The **tracking problem**, in which $y(t)$ is specified to **track** $r(t) \neq 0$.

When $r(t) = R \neq 0$, constant, the regulator and tracking problems are essentially the same. Tracking a **nonconstant** reference $r(t)$ is a more difficult problem, called the **servomechanism problem**.

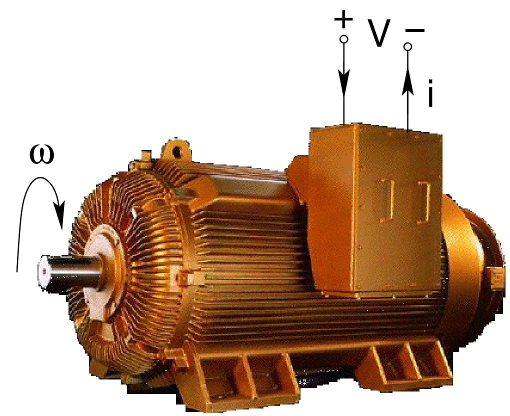
Regulation and Tracking

We review the state feedback design procedure with an example.

Example (Speed control of a DC motor). We consider a DC motor described by the state equations

$$\frac{d}{dt}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix} = \begin{bmatrix}-10 & 1\\ -0.02 & -2\end{bmatrix}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix} + \begin{bmatrix}0\\ 2\end{bmatrix} V(t)$$

$$y = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix}$$



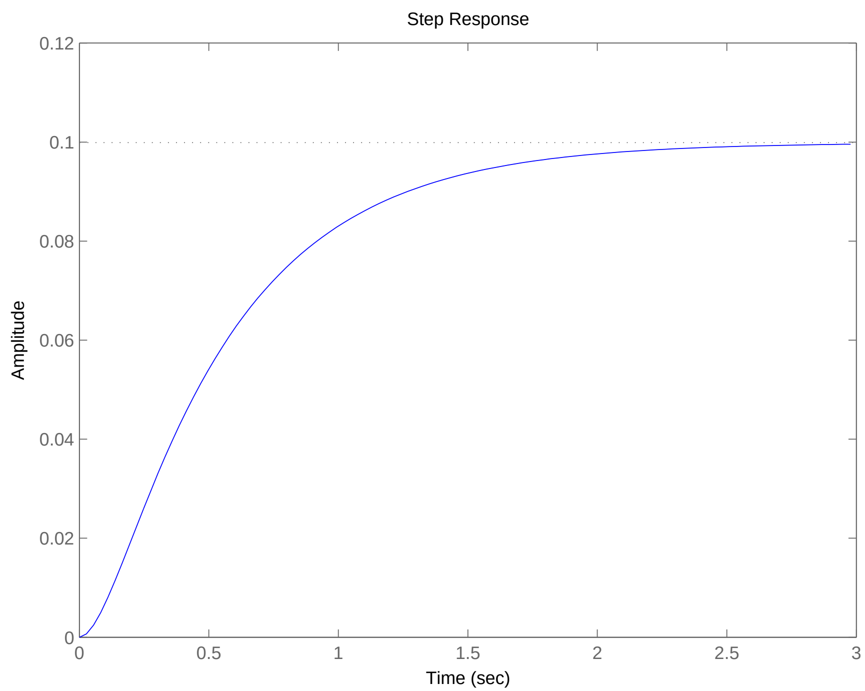
The input to the DC motor is the voltage $V(t)$, and the states are the current $i(t)$ and rotor velocity $\omega(t)$. We assume that we can measure both, and take $\omega(t)$ as the output of interest.

Regulation and Tracking

Example (continuation). ① The open-loop characteristic polynomial is

$$\Delta(s) = \det(sI - A) = \det \begin{bmatrix} s+10 & -1 \\ 0.02 & s+2 \end{bmatrix} = s^2 + 12s + 20.02$$

which has two stable roots at $s = -9.9975$ and $s = -2.0025$. The motor **open-loop step response** is



The system takes about 3s to reach steady-state. The final speed is about $1/10$ the amplitude of the voltage step.

We would like to design a state feedback control to make the motor response faster and obtain tracking of $\omega(t)$ to constant reference inputs r.

Regulation and Tracking

Example (continuation). To design the state feedback gain, we next ② compute the controllability matrix

$$\mathcal{C} = \begin{bmatrix} B & AB \end{bmatrix} = \begin{bmatrix} 0 & 2 \\ 2 & -4 \end{bmatrix}$$

which is full rank $\Rightarrow$ the system is **controllable**.

Also, from the open-loop characteristic polynomial we form **controllability matrix in $\bar{x}$ coordinates** is

$$\bar{\mathcal{C}} = \begin{bmatrix} 1 & \alpha_1 \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 12 \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & -12 \\ 0 & 1 \end{bmatrix}$$

Regulation and Tracking

Example (continuation). We now ③ propose a desired characteristic polynomial. Suppose that we we would like the closed-loop eigenvalues to be at $s = -5 \pm j$, which yield a step response with 0.1% overshoot and about 1s settling time.

The desired (closed-loop) characteristic polynomial is then

$$\Delta_K(s) = (s + 5 - j)(s + 5 + j) = s^2 + 10s + 26$$

With $\Delta_K(s)$ and $\Delta(s)$ we determine the state feedback gain in $\bar{x}$ coordinates

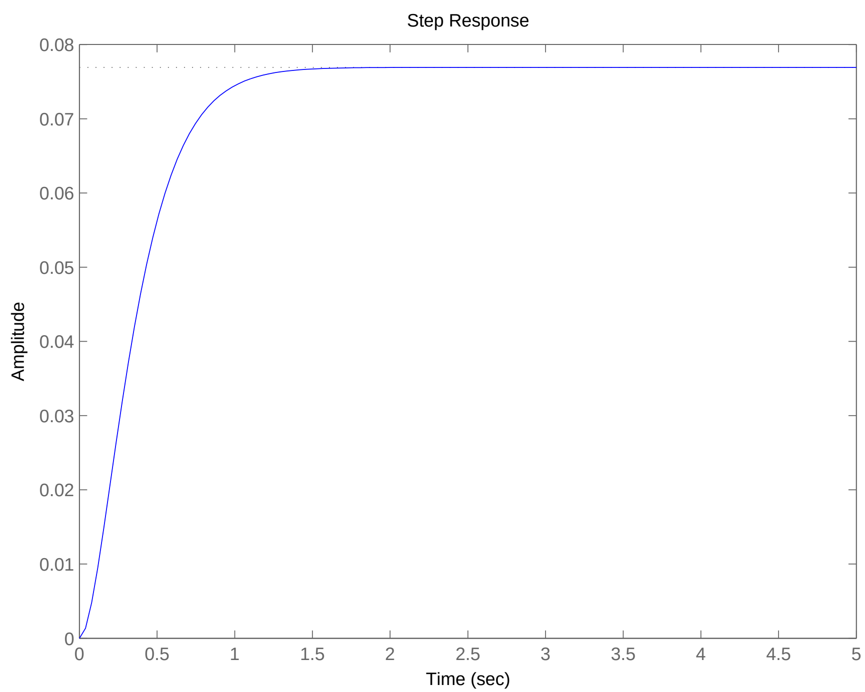
$$\bar{K} = \begin{bmatrix} (\bar{\alpha}_1 - \alpha_1) & (\bar{\alpha}_2 - \alpha_2) \end{bmatrix} = \begin{bmatrix} (10 - 12) & (26 - 20.02) \end{bmatrix}$$
$$= \begin{bmatrix} -2 & 5.98 \end{bmatrix}$$

Regulation and Tracking

Example (continuation). Finally, ④ we obtain the state feedback gain $\mathbf{K}$ in the original coordinates using **Bass-Gura** formula,

$$\mathbf{K} = \bar{\mathbf{K}}\bar{\mathcal{C}}\mathcal{C}^{-1} = \begin{bmatrix} -2 & 5.98 \end{bmatrix} \begin{bmatrix} 1 & -12 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 2 & -4 \end{bmatrix}^{-1}$$
$$= \begin{bmatrix} 12.99 & -1 \end{bmatrix}$$

As can be verified, the eigenvalues of $(\mathbf{A} - \mathbf{BK})$ are as desired.

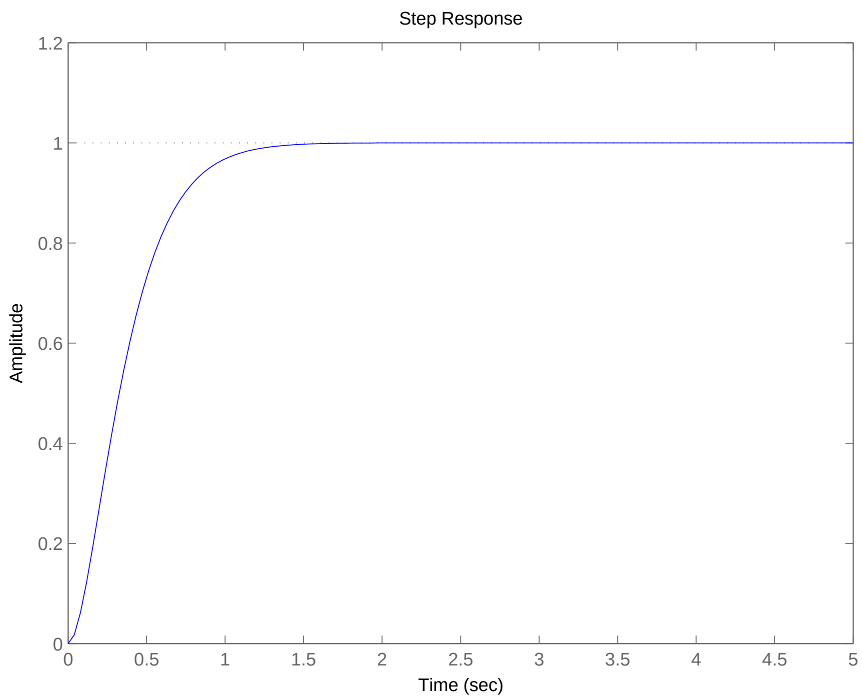


The **closed-loop step response**, as desired, settles in 1s, with no significant overshoot.

Note, however, that we still have steady-state error ($\omega(t) \to 0.0769$). To fix it, we use the feed-forward gain $\mathbf{N}$.

Regulation and Tracking

Example (continuation). The system transfer function does not have a **zero at** $s = 0$, which would prevent tracking of constant references (as we can see in the step response, which otherwise, would asymptotically go to 0).



Thus, ⑤ we determine N with the formula

$$N = -\frac{1}{G_{cl}(0)} = \frac{-1}{C(A - BK)^{-1}B} = 13$$

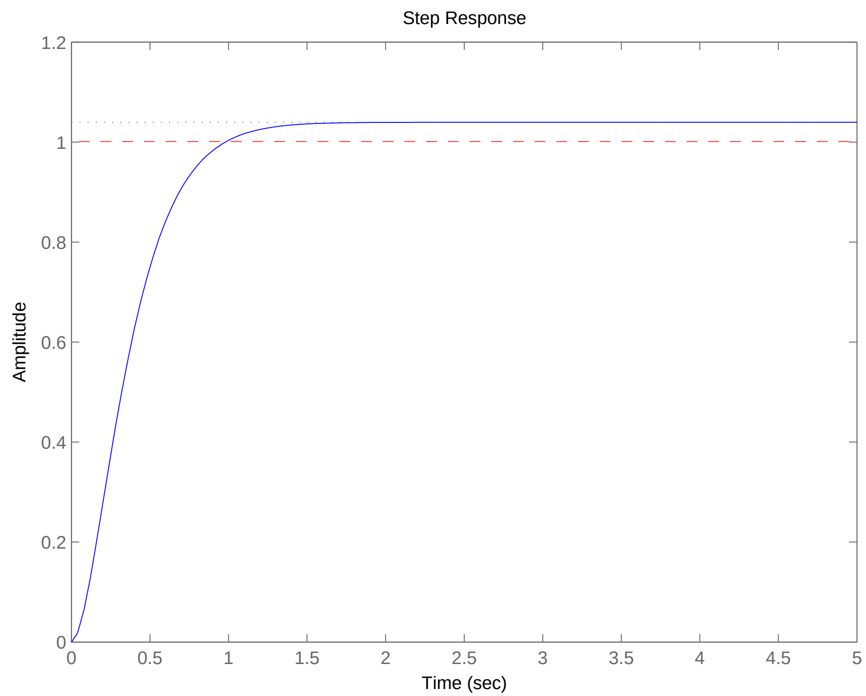
and achieve zero steady-state error in the closed-loop step response.

Regulation and Tracking

Example (continuation). We have designed a state feedback controller for the speed of the DC motor. However, the tracking achieved by **feedforward precompensation** would not tolerate (it is not **robust** to) to uncertainties in the plant model.

To see this, suppose the **real** matrix A in the system is slightly different from the one we used to compute $\mathbf{K}$,

$$\tilde{\mathbf{A}} = \begin{bmatrix} -10 & 1 \\ -0.02 & -2 \end{bmatrix} + \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$



The closed-loop step response given by the designed gains $\mathbf{N}$, $\mathbf{K}$ (based on a **different** A-matrix) doesn't yield tracking. □

Outline

- Regulation and Tracking
- Robust Tracking: Integral Action
- State Estimation

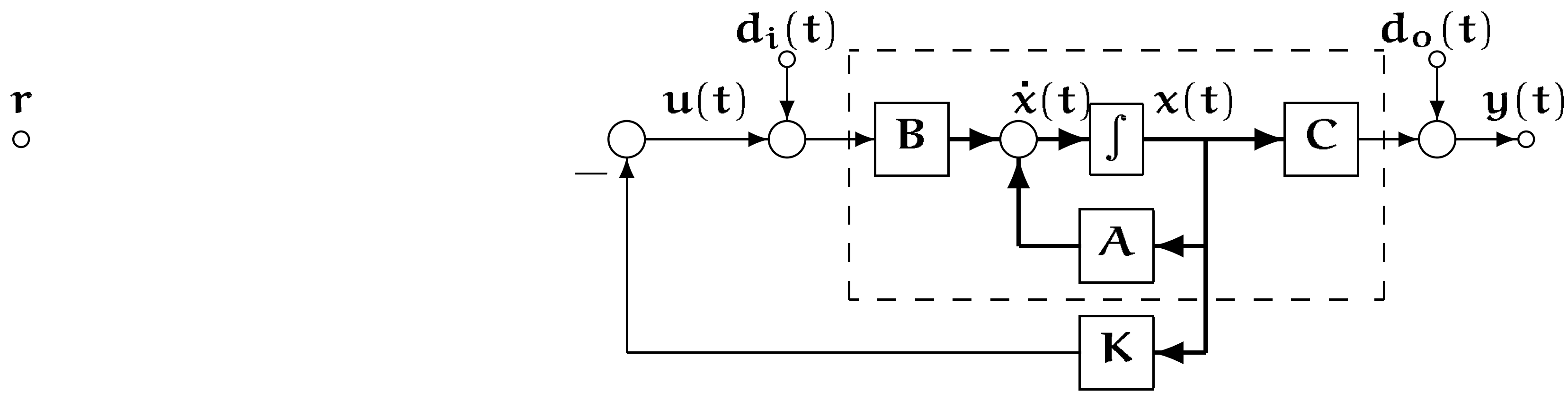
Robust Tracking: Integral Action

We now introduce a **robust** approach to achieve constant reference tracking by state feedback. This approach consists in the **addition of integral action** to the state feedback, so that

- the error $\varepsilon(t) = r - y(t)$ will approach 0 as $t \to \infty$, and this property will be preserved
  - under moderate uncertainties in the plant model
  - under constant input or output disturbance signals.

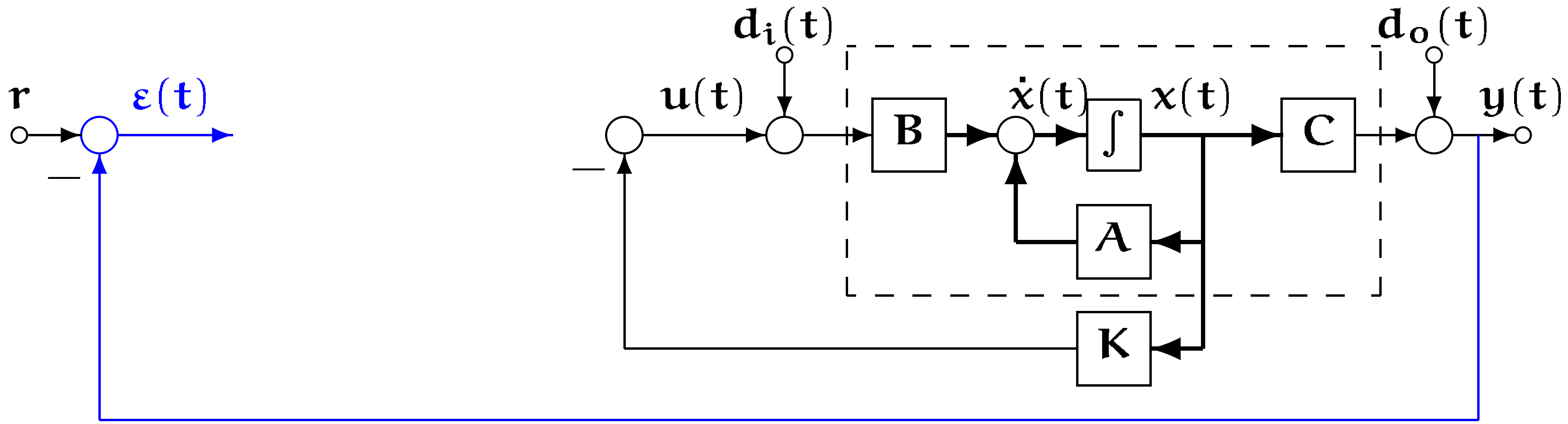
Robust Tracking: Integral Action

The State Feedback with Integral Action scheme:



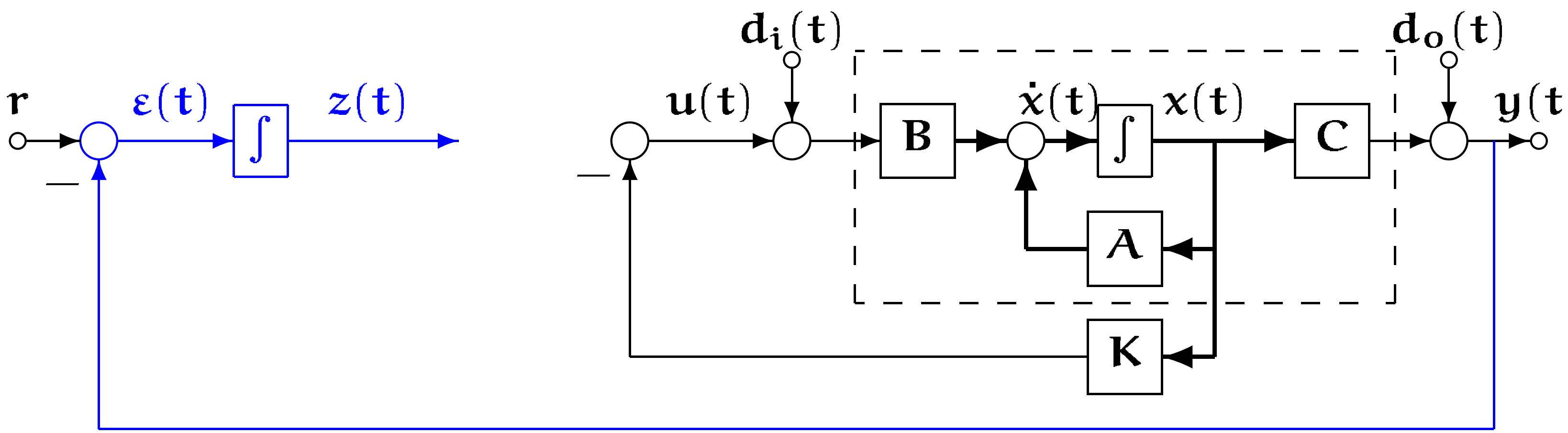
Robust Tracking: Integral Action

The State Feedback with Integral Action scheme:



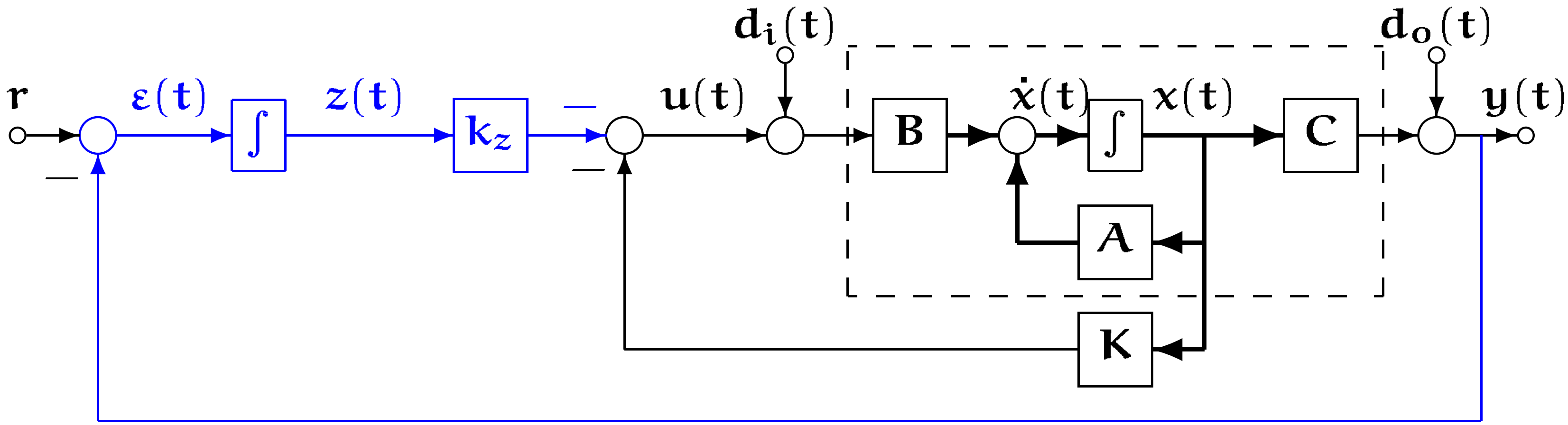
Robust Tracking: Integral Action

The State Feedback with Integral Action scheme:



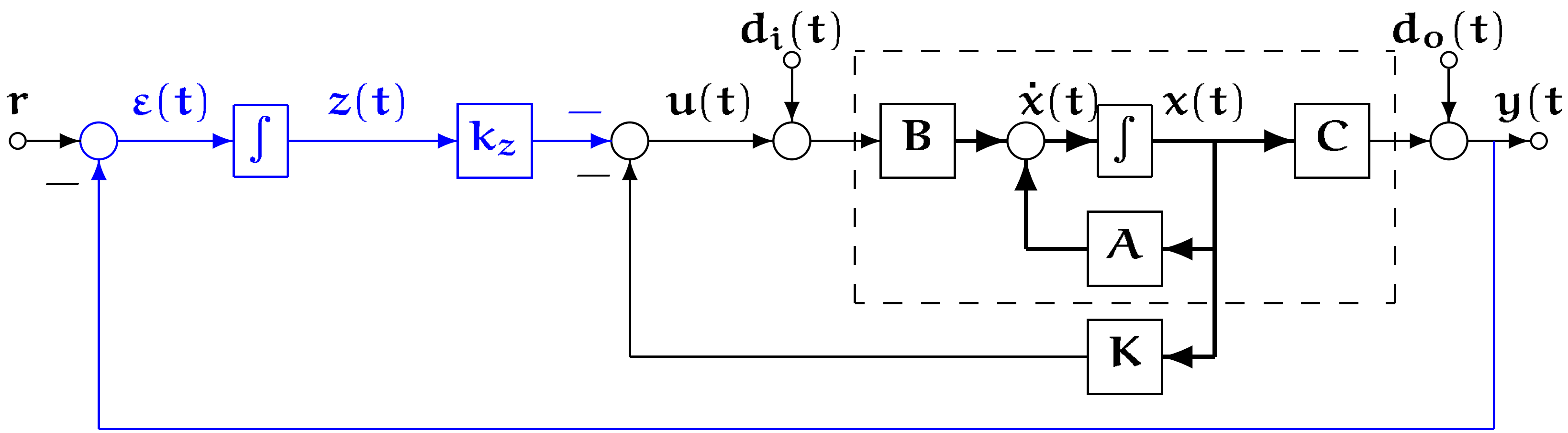
Robust Tracking: Integral Action

The State Feedback with Integral Action scheme:



Robust Tracking: Integral Action

The State Feedback with Integral Action scheme:



The main idea in the addition of integral action is to **augment the plant** with an extra state: the integral of the tracking error $\varepsilon(t)$,

$$\dot{z}(t) = r - y(t) = r - Cx(t) \qquad \text{(IA1)}$$

The control law for the **augmented plant** is then

$$u(t) = -\begin{bmatrix} K & k_z \end{bmatrix} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} \qquad \text{(IA2)}$$

Robust Tracking: Integral Action

The closed-loop state equation with the state feedback control $\mathbf{u}(t)$ given by (IA1) and (IA2) is

$$\begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}}_{A_a} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} - \underbrace{\begin{bmatrix} B \\ 0 \end{bmatrix}}_{B_a} \underbrace{\begin{bmatrix} K & k_z \end{bmatrix}}_{K_a} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$

Robust Tracking: Integral Action

The closed-loop state equation with the state feedback control $\mathbf{u}(t)$ given by (IA1) and (IA2) is

$$\begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}}_{A_a} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} - \underbrace{\begin{bmatrix} B \\ 0 \end{bmatrix}}_{B_a} \underbrace{\begin{bmatrix} K & k_z \end{bmatrix}}_{K_a} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$

$$= (A_a - B_a K_a) \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$

Robust Tracking: Integral Action

The closed-loop state equation with the state feedback control $\mathbf{u}(\mathbf{t})$ given by (IA1) and (IA2) is

$$\begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}}_{A_a} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} - \underbrace{\begin{bmatrix} B \\ 0 \end{bmatrix}}_{B_a} \underbrace{\begin{bmatrix} K & k_z \end{bmatrix}}_{K_a} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$

$$= (A_a - B_a K_a) \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$

The state feedback design with integral action can be done as a normal state feedback design for the **augmented plant**

Robust Tracking: Integral Action

The closed-loop state equation with the state feedback control $u(t)$ given by (IA1) and (IA2) is

$$\begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}}_{A_a} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} - \underbrace{\begin{bmatrix} B \\ 0 \end{bmatrix}}_{B_a} \underbrace{\begin{bmatrix} K & k_z \end{bmatrix}}_{K_a} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$

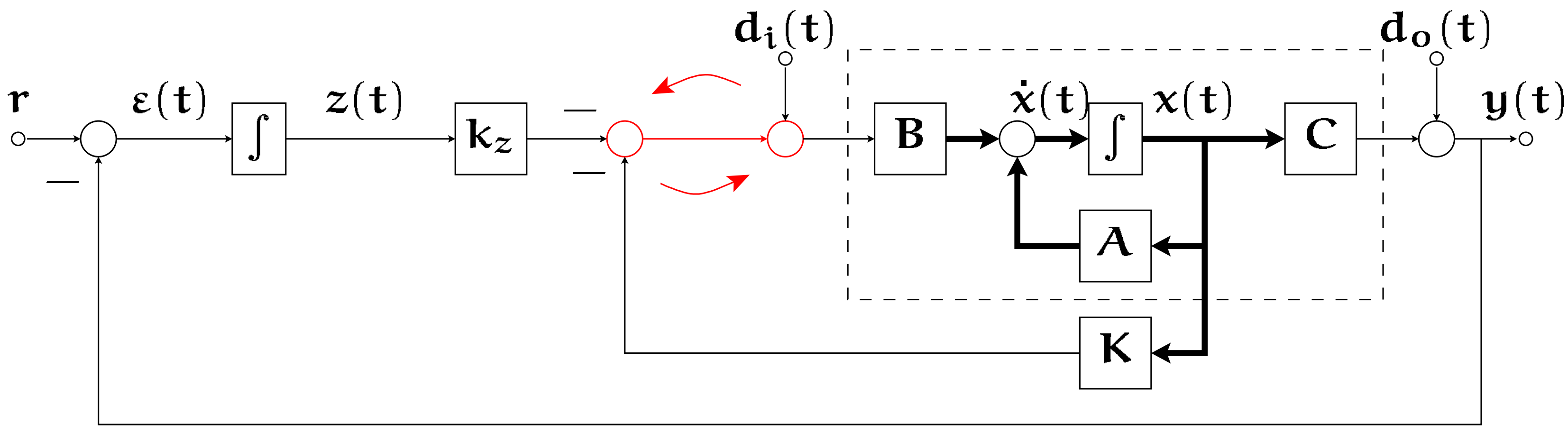
$$= (A_a - B_a K_a) \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$

The state feedback design with integral action can be done as a normal state feedback design for the **augmented plant**

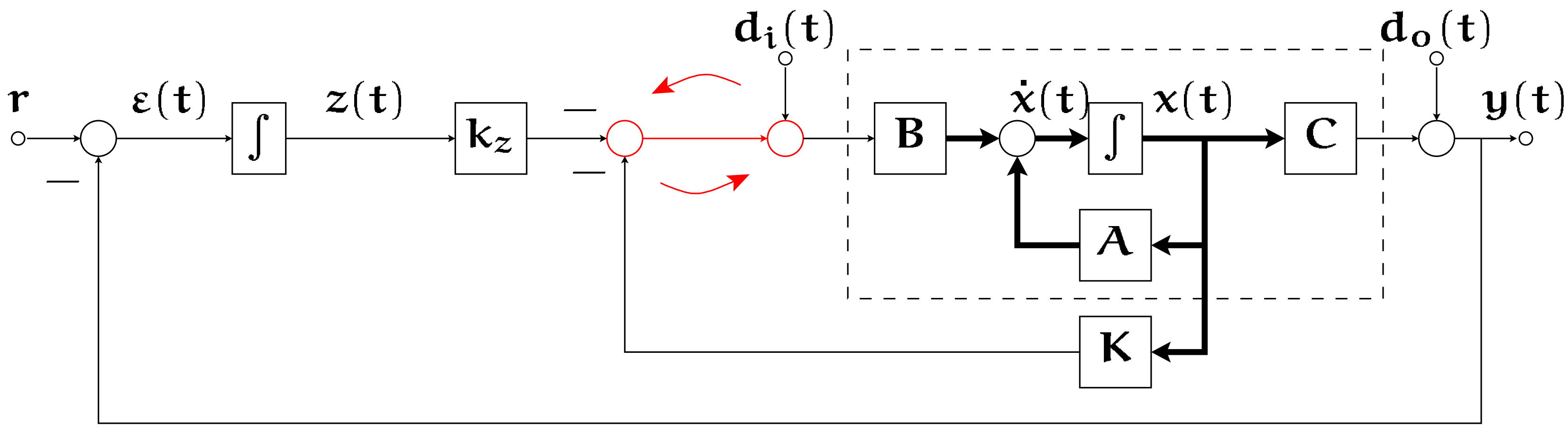
If K_a is designed such that the closed-loop augmented matrix $(A_a - B_a K_a)$ is rendered Hurwitz, then necessarily in steady-state

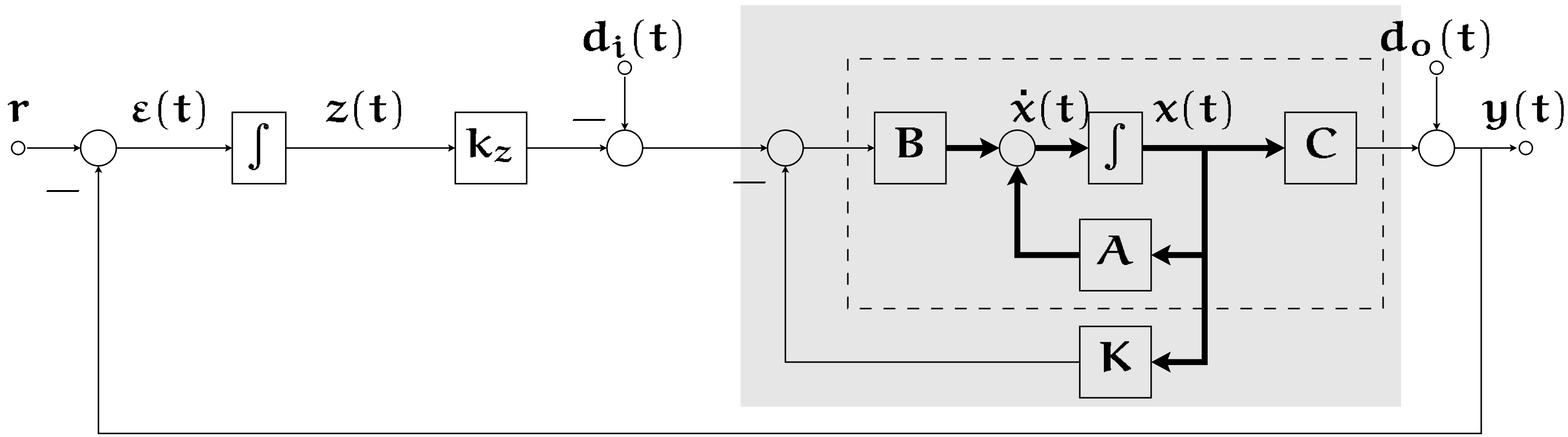
$$\lim_{t\to\infty} \dot{z}(t) = 0 \quad \Rightarrow \quad \lim_{t\to\infty} y(t) = r, \quad \text{achieving tracking.}$$

Integral Action — How does it work?

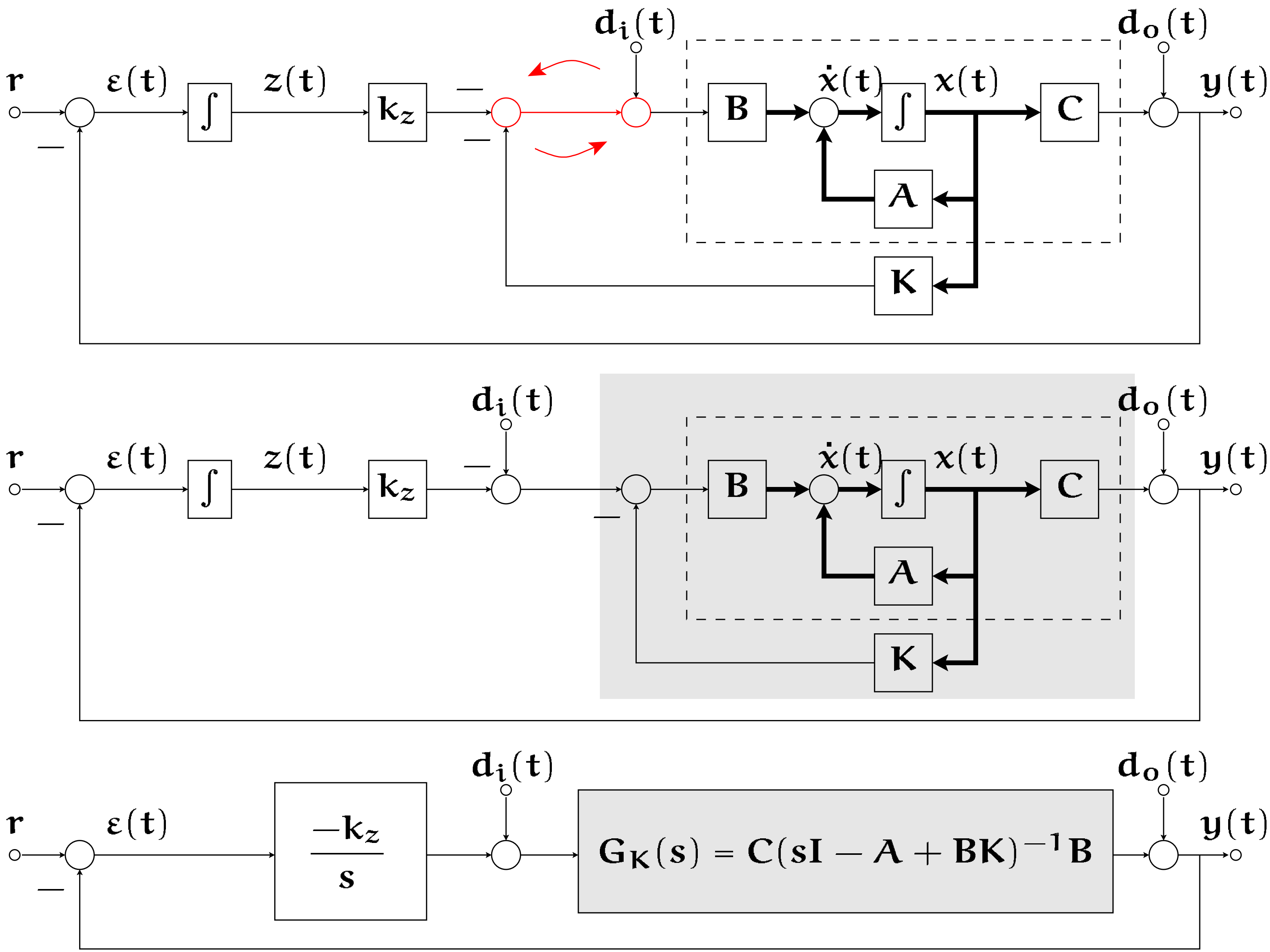


Integral Action — How does it work?



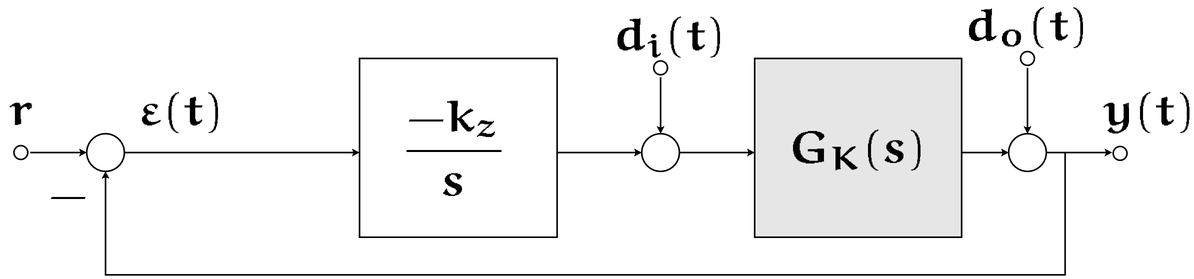


Integral Action — How does it work?



Integral Action — How does it work?

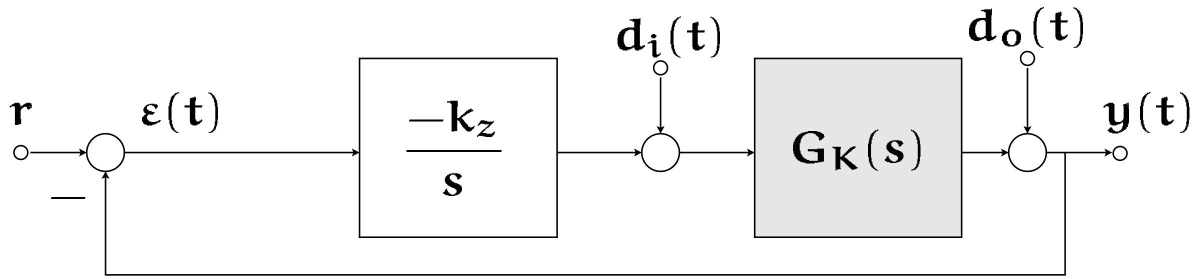
The block diagram of the closed-loop system controlled by state feedback with integral action thus collapses to



where $G_K(s)$ is a BIBO stable transfer function, by design, and **also the overall closed-loop is BIBO stable**.

Integral Action — How does it work?

The block diagram of the closed-loop system controlled by state feedback with integral action thus collapses to



where $G_K(s)$ is a BIBO stable transfer function, by design, and **also the overall closed-loop is BIBO stable**. Let's express $G_K(s)$ in terms of its numerator and denominator polynomials,

$$G_K(s) = \frac{N(s)}{D(s)}$$

Then, from the block diagram above,

$$Y(s) = \frac{\frac{(-k_z)N(s)}{sD(s)}}{1 + \frac{(-k_z)N(s)}{sD(s)}} R(s) + \frac{\frac{N(s)}{D(s)}}{1 + \frac{(-k_z)N(s)}{sD(s)}} D_i(s) + \frac{1}{1 + \frac{(-k_z)N(s)}{sD(s)}} D_o(s)$$

Integral Action — How does it work?

In other words,

$$Y(s) = \frac{(-k_z)N(s)}{sD(s) + (-k_z)N(s)}\, R(s) + \frac{sN(s)}{sD(s) + (-k_z)N(s)}\, D_i(s) + \frac{sD(s)}{sD(s) + (-k_z)N(s)}\, D_o(s)$$

Integral Action — How does it work?

In other words,

$$Y(s) = \frac{(-k_z)N(s)}{sD(s) + (-k_z)N(s)}\,R(s) + \frac{sN(s)}{sD(s) + (-k_z)N(s)}\,D_i(s) + \frac{sD(s)}{sD(s) + (-k_z)N(s)}\,D_o(s)$$

If the reference and the disturbances are constants, say r, d_i and d_o then, because the closed-loop is BIBO stable, the steady state value of $y(t)$ is determined by the 3 transfer functions above evaluated at $s = 0$,

$$\lim_{t\to\infty} y(t) = \frac{(-k_z)N(0)}{0\cdot D(0) + (-k_z)N(0)}\,r + \frac{0\cdot N(0)}{0\cdot D(0) + (-k_z)N(0)}\,d_i + \frac{0\cdot D(0)}{0\cdot D(0) + (-k_z)N(0)}\,d_o$$

Integral Action — How does it work?

In other words,

$$Y(s) = \frac{(-k_z)N(s)}{sD(s) + (-k_z)N(s)}\, R(s) + \frac{sN(s)}{sD(s) + (-k_z)N(s)}\, D_i(s) + \frac{sD(s)}{sD(s) + (-k_z)N(s)}\, D_o(s)$$

If the reference and the disturbances are constants, say r, d_i and d_o then, because the closed-loop is BIBO stable, the steady state value of $y(t)$ is determined by the 3 transfer functions above evaluated at $s = 0$,

$$\begin{aligned}\lim_{t\to\infty} y(t) &= \frac{(-k_z)N(0)}{0\cdot D(0) + (-k_z)N(0)}\, r + \frac{0\cdot N(0)}{0\cdot D(0) + (-k_z)N(0)}\, d_i + \frac{0\cdot D(0)}{0\cdot D(0) + (-k_z)N(0)}\, d_o \\ &= r + 0 + 0.\end{aligned}$$

That is: the output will asymptotically track constant references and reject constant disturbances irrespective of the values r, d_i and d_o.

Robust Tracking Example

Example (Robust speed tracking in a DC motor). Let's go back to the DC motor example and design a state feedback with integral action to achieve robust speed tracking.

We have to **redesign** the state feedback gain K — it must be computed together with k_z for the **augmented plant** given by

$$A_a = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} = \begin{bmatrix} -10 & 1 & 0 \\ -0.02 & -2 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \quad B_a = \begin{bmatrix} B \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix}$$

If we compute the characteristic polynomial of the augmented matrix A_a we find

$$\Delta_a(s) = s^3 + 12s^2 + 20.02s = s\Delta(s).$$

Unsurprisingly, it just adds a root at $s = 0$ to the original.

Robust Tracking Example

Example (continuation). We now move on to compute $\mathcal{C}_a$ and $\bar{\mathcal{C}}_a$ — for the augmented pair (A_a, B_a),

$$\mathcal{C}_a = \begin{bmatrix} 0 & 2 & -24 \\ 2 & -4 & 7.96 \\ 0 & 0 & -2 \end{bmatrix}, \qquad \bar{\mathcal{C}}_a = \begin{bmatrix} 1 & 12 & 20.02 \\ 0 & 1 & 12 \\ 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & -12 & 123.98 \\ 0 & 1 & -12 \\ 0 & 0 & 1 \end{bmatrix}$$

Note that the augmented pair (A_a, B_a) will always be **controllable** as long as the original pair (A, B) is controllable and the system **has no zeros at $s = 0$.**

We select as **desired augmented characteristic polynomial**

$$\Delta_{K_a}(s) = (s + 6)\Delta_K(s) = s^3 + 16s^2 + 86s + 156$$

(We keep the original desired characteristic polynomial and let the extra pole be faster, to keep the same specifications in the response of $y(t)$.)

Robust Tracking Example

Example (continuation). As before, from $\Delta_a(s)$ and $\Delta_{aK}(s)$ we compute $\bar{K}_a$ and now obtain

$$\bar{K}_a = \begin{bmatrix}(16 - 12) & (86 - 20.02) & (156 - 0)\end{bmatrix} = \begin{bmatrix}4 & 65.98 & 156\end{bmatrix}$$

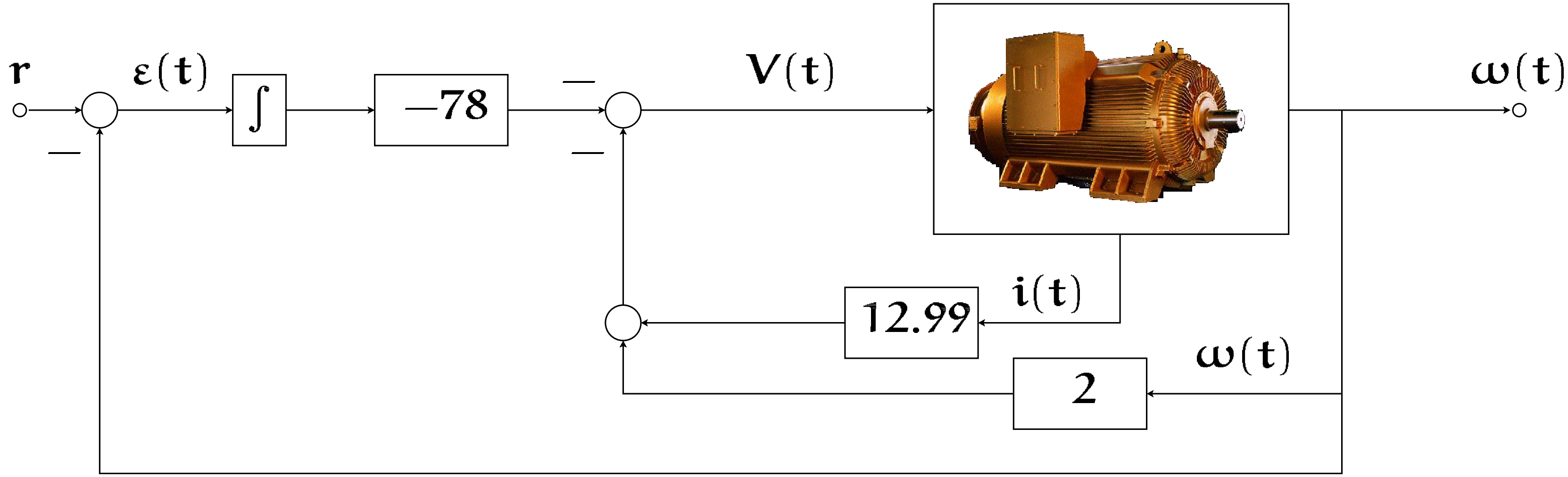
and finally,

$$K_a = \bar{K}_a \bar{\mathcal{C}}_a \mathcal{C}^{-1} = \Big[\underbrace{\begin{bmatrix}12.99 & 2\end{bmatrix}}_{K} \; \underbrace{\begin{bmatrix}-78\end{bmatrix}}_{k_z}\Big]$$

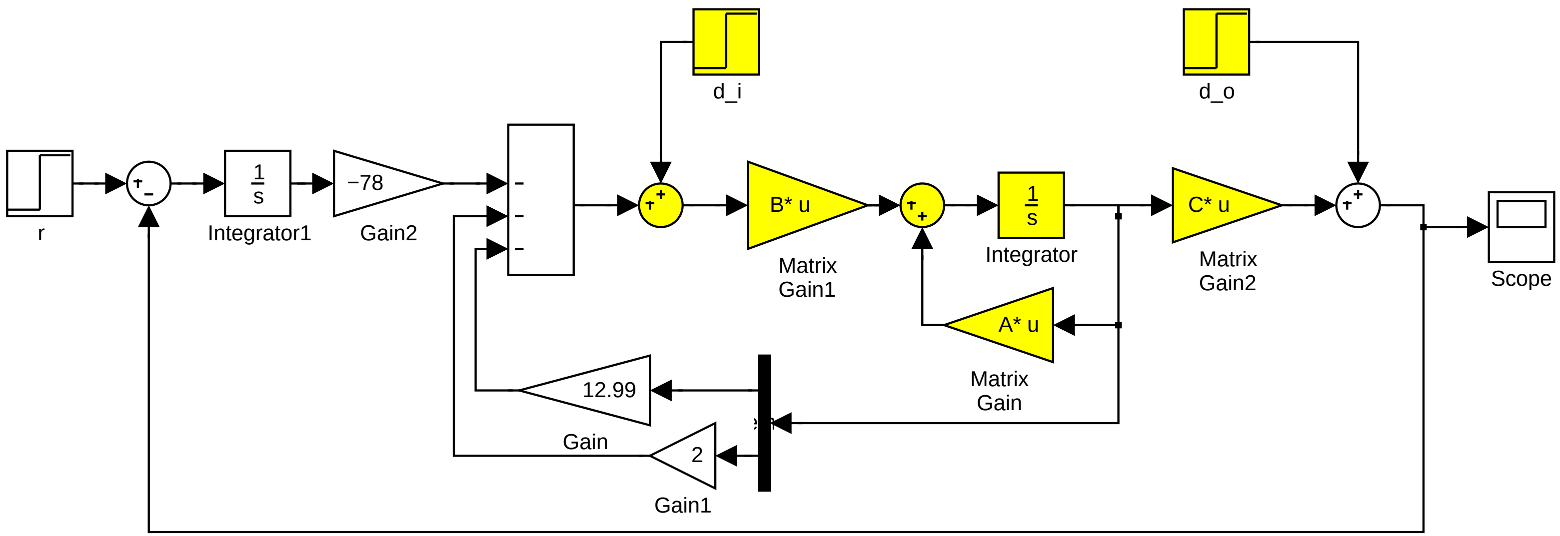
Note that the first two elements of the augmented K_a correspond to the **new** state feedback gain K for the motor state $\begin{bmatrix}\omega(t) \\ i(t)\end{bmatrix}$, while the last element is the state feedback gain k_z for the augmented state $z(t)$.

Robust Tracking Example

Example (continuation). A block diagram for implementation:

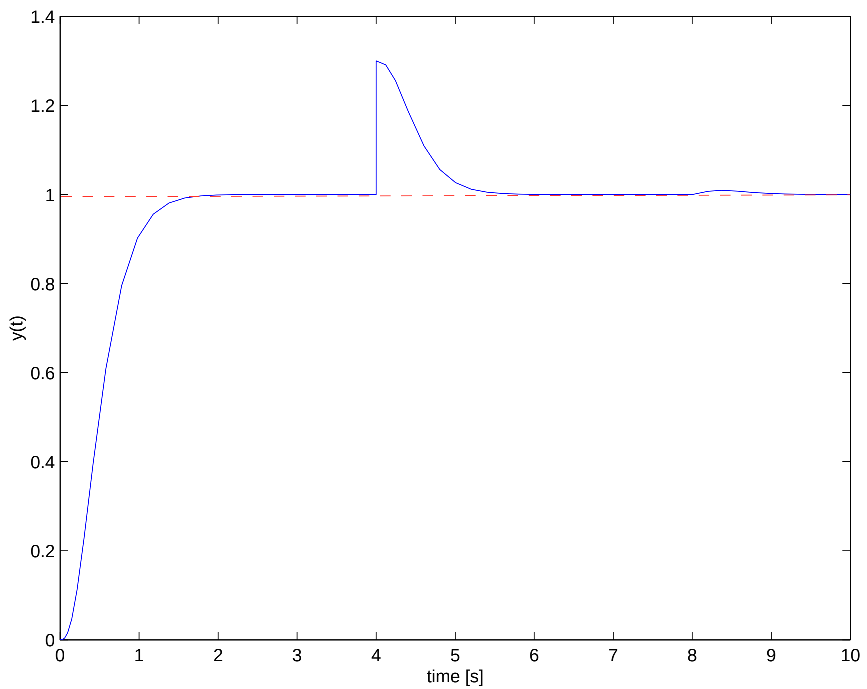


And a SIMULINK diagram (including disturbances):



Robust Tracking Example

Example (continuation). We simulate the closed loop system to a unit step input applied at $t = 0$, an input disturbance $d = 0.5$ applied at $t = 8$s, and an output disturbance $d_o = 0.3$ applied at $t = 4$s.



We can see the asymptotic tracking of the reference despite the disturbances. □

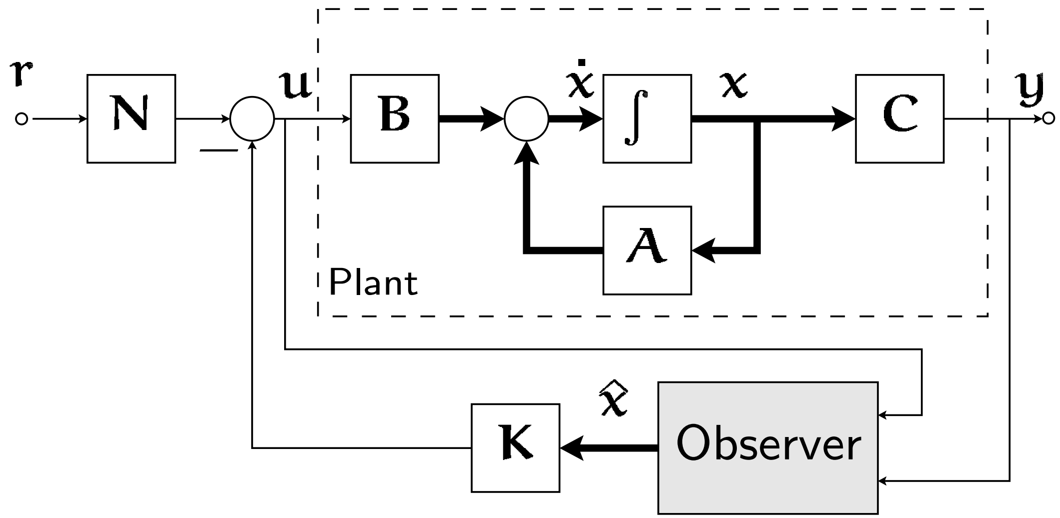
Outline

- Regulation and Tracking
- Robust Tracking: Integral Action
- State Estimation

State Estimation

State feedback requires measuring the states but, normally, we do not have direct access to *all* the states. Then, how do we implement state feedback?

If the system is observable the states can be estimated by an **observer**.

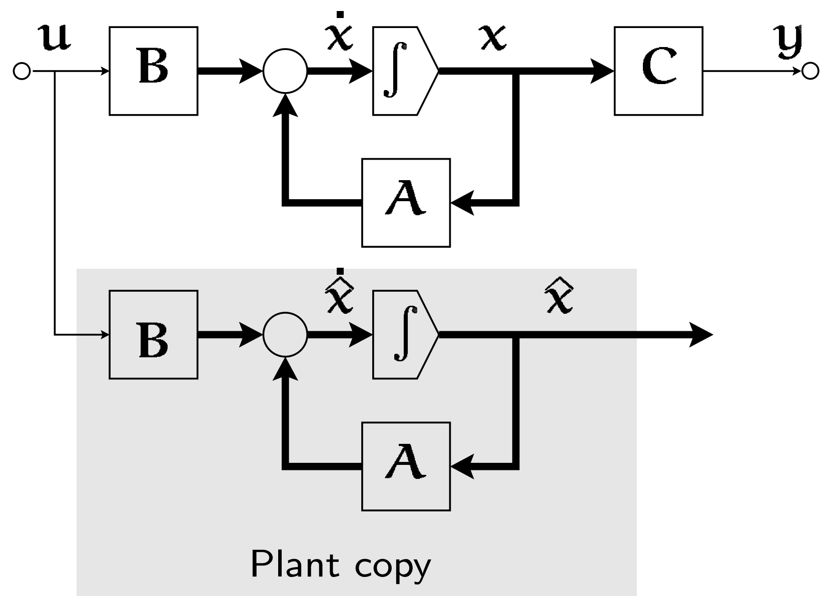


State feedback from estimated states

An observer is a dynamic system that estimates the states of the plant based on the measurement of its inputs and outputs.

State Estimation: A "Naive" Observer

How would we build an observer? One intuitive way would be to **reproduce a model of the plant** and run it simultaneously to obtain a state estimate $\hat{x}(t)$.

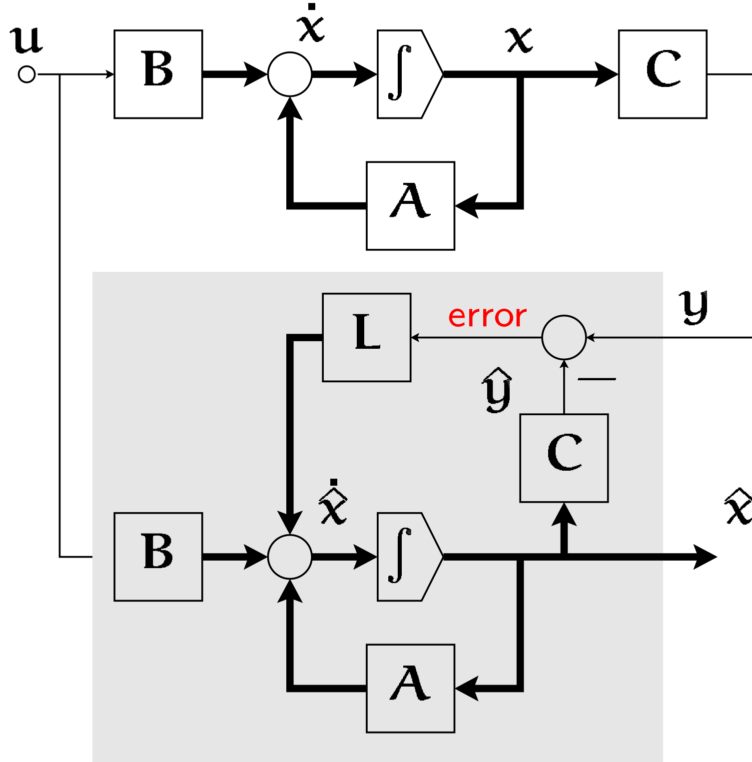


A "naive" design of an observer.

The problem with this "naive" design **is that if the plant and its "copy" in the observer have different initial conditions**, the estimates will generally **not** converge to the true values.

State Estimation: A "Feedback" Observer

A better observer structure includes error feedback correction.



A "self-corrected" (feedback) design of an observer.

By appropriately designing the **matrix gain** $\mathbf{L}$, we could adjust the observer to give a state estimate that will asymptotically converge to the true state.

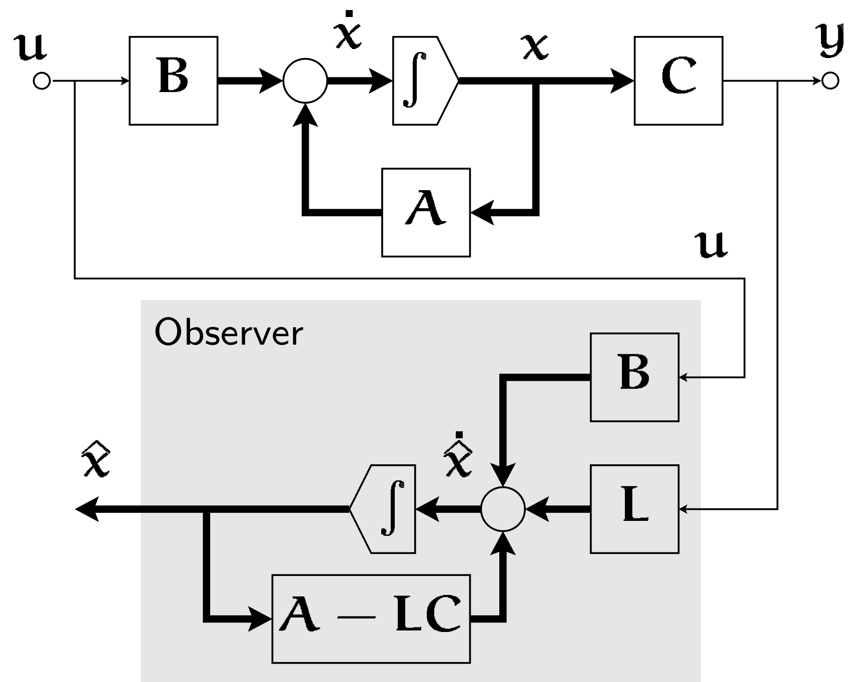
State Estimation: Final Observer Structure

By rearranging the previous block diagram, we get to the final structure of the observer

If the system is **observable**, we can then choose the gain $\mathbf{L}$ to ascribe the eigenvalues of $(\mathbf{A} - \mathbf{L}\mathbf{C})$ arbitrarily.

We certainly want the observer to be stable!

From the block diagram, the observer equations are



Observer structure

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

$$= (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{L}\mathbf{y}(t) \qquad \text{(O)}$$

State Estimation

From the observer state equation (O), and the plant state equation

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

We can obtain a state equation for the **estimation error** $\varepsilon = x - \hat{x}$

$$\begin{aligned}\dot{\varepsilon} &= \dot{x} - \dot{\hat{x}} \\ &= Ax + Bu - A\hat{x} - Bu - LC(x - \hat{x}) \\ &= A(x - \hat{x}) - LC(x - \hat{x}) \\ &= (A - LC)\varepsilon \quad \Rightarrow \boxed{\varepsilon(t) = e^{(A-LC)t}\varepsilon(0)}.\end{aligned}$$

Thus, we see that for the error to asymptotically converge to zero $\varepsilon(t) \to 0$, (and so $\hat{x}(t) \to x(t)$) we **need** $(A - LC)$ to be **Hurwitz**.

Observer Design

In summary, to build an observer, we use the matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ from the plant and form the state equation

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

where $\mathbf{L}$ is such that the eigenvalues of $(\mathbf{A} - \mathbf{L}\mathbf{C})$ have negative real part.

How to choose $\mathbf{L}$? We can use, by **duality**, the same procedure that we already know for the design of a state feedback gain $\mathbf{K}$ to render $\mathbf{A} - \mathbf{B}\mathbf{K}$ Hurwitz. Notice that the matrix transpose

$$\begin{aligned}(\mathbf{A} - \mathbf{L}\mathbf{C})^T &= \mathbf{A}^T - \mathbf{C}^T\mathbf{L}^T \\ &= \mathbf{A}_{\text{dual}} - \mathbf{B}_{\text{dual}}\mathbf{K}_{\text{dual}}\end{aligned}$$

We choose $\mathbf{K}_{\text{dual}}$ to make $\mathbf{A}_{\text{dual}} - \mathbf{B}_{\text{dual}}\mathbf{K}_{\text{dual}}$ Hurwitz, and finally

$$\boxed{\mathbf{L} = \mathbf{K}_{\text{dual}}^T}$$

State Estimation Example

Example (Current estimation in a DC motor). We revisit the DC motor example seen earlier. Before we used state feedback to achieve robust reference tracking and disturbance rejection. This required **measurement of both states:** current $i(t)$ and velocity $\omega(t)$.

Suppose now that we don't measure current, but only the motor speed. We will construct an observer to estimate $i(t)$.

Recall the plant equations

$$\frac{d}{dt}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix} = \begin{bmatrix}-10 & 1\\ -0.02 & -2\end{bmatrix}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix} + \begin{bmatrix}0\\ 2\end{bmatrix} V(t)$$

$$y = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix}$$

State Estimation Example

Example (continuation). We first check for **observability** (otherwise, we won't be able to build the observer)

$$\mathcal{O} = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -10 & 1 \end{bmatrix}$$

which is full rank, and hence the system **is observable**.

By duality, in the procedure to compute K_{dual}, the role of $\mathcal{C}$ is played by $\mathcal{C}_{\text{dual}} = \mathcal{O}^T$, and $\bar{\mathcal{C}}_{\text{dual}}$ **is the same**

$$\bar{\mathcal{C}}_{\text{dual}} = \bar{\mathcal{C}} = \begin{bmatrix} 1 & \alpha_1 \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & -12 \\ 0 & 1 \end{bmatrix}$$

since $\bar{\mathcal{C}}$ only depends on the characteristic polynomial of A, which is the same as that of A^T.

State Estimation Example

Example (continuation). Say that the desired eigenvalues for the observer are $s = -6 \pm j2$, (slightly **faster** than those set for the closed-loop plant, which is standard) which yields

$$\Delta_{K_{\text{dual}}} = s^2 + 12s + 40$$

Thus, from the coefficients of $\Delta_{K_{\text{dual}}}$ and those of $\Delta(s)$ we have

$$\bar{K}_{\text{dual}} = \begin{bmatrix}(12 - 12) & (40 - 20.02)\end{bmatrix} = \begin{bmatrix}0 & 19.98\end{bmatrix}$$

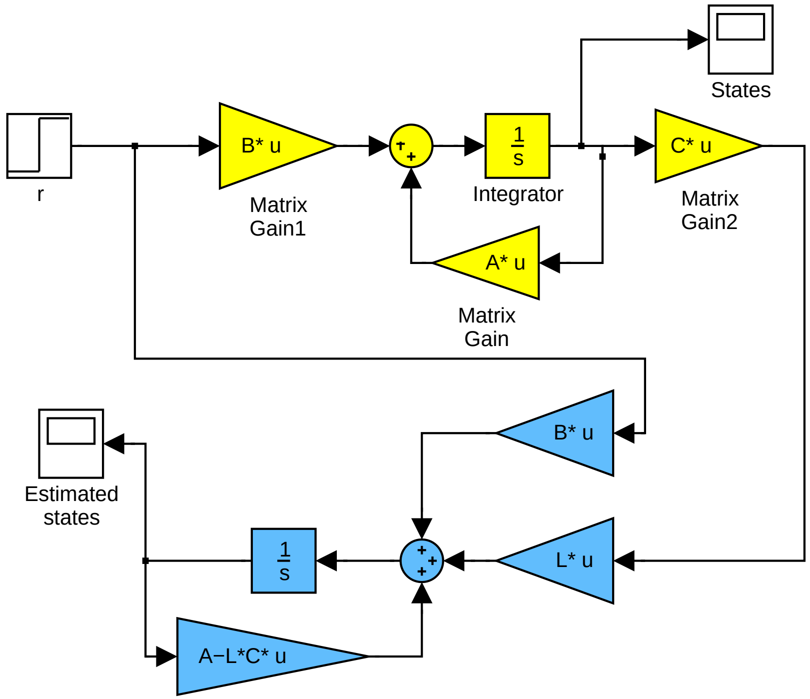
We now return to the original coordinates and get K_{dual},

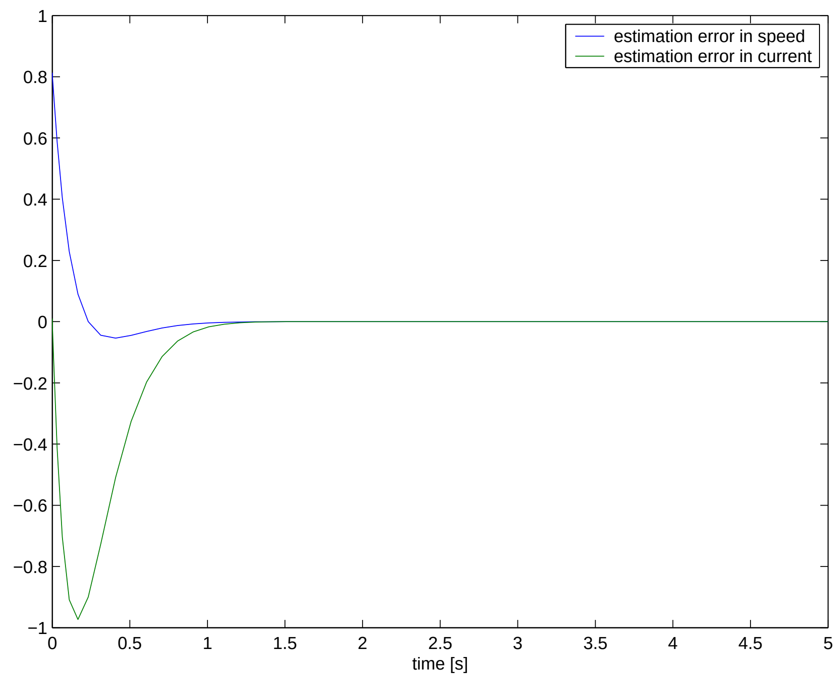
$$K_{\text{dual}} = \bar{K}_{\text{dual}}\bar{\mathcal{C}}\mathcal{C}^{-1}_{\text{dual}}, = \begin{bmatrix}0 & 19.98\end{bmatrix}, \qquad \text{by chance, the same as } \bar{K}_{\text{dual}}$$

Finally, $\boxed{L = K^T_{\text{dual}} = \begin{bmatrix}0 \\ 19.98\end{bmatrix}}$. It can be checked with MATLAB that L effectively place the eigenvalues of $(A - LC)$ at the desired locations.

State Estimation Example

Example (continuation). We simulated the observer with SIMULINK, setting some initial conditions to the plant (and none to the observer)

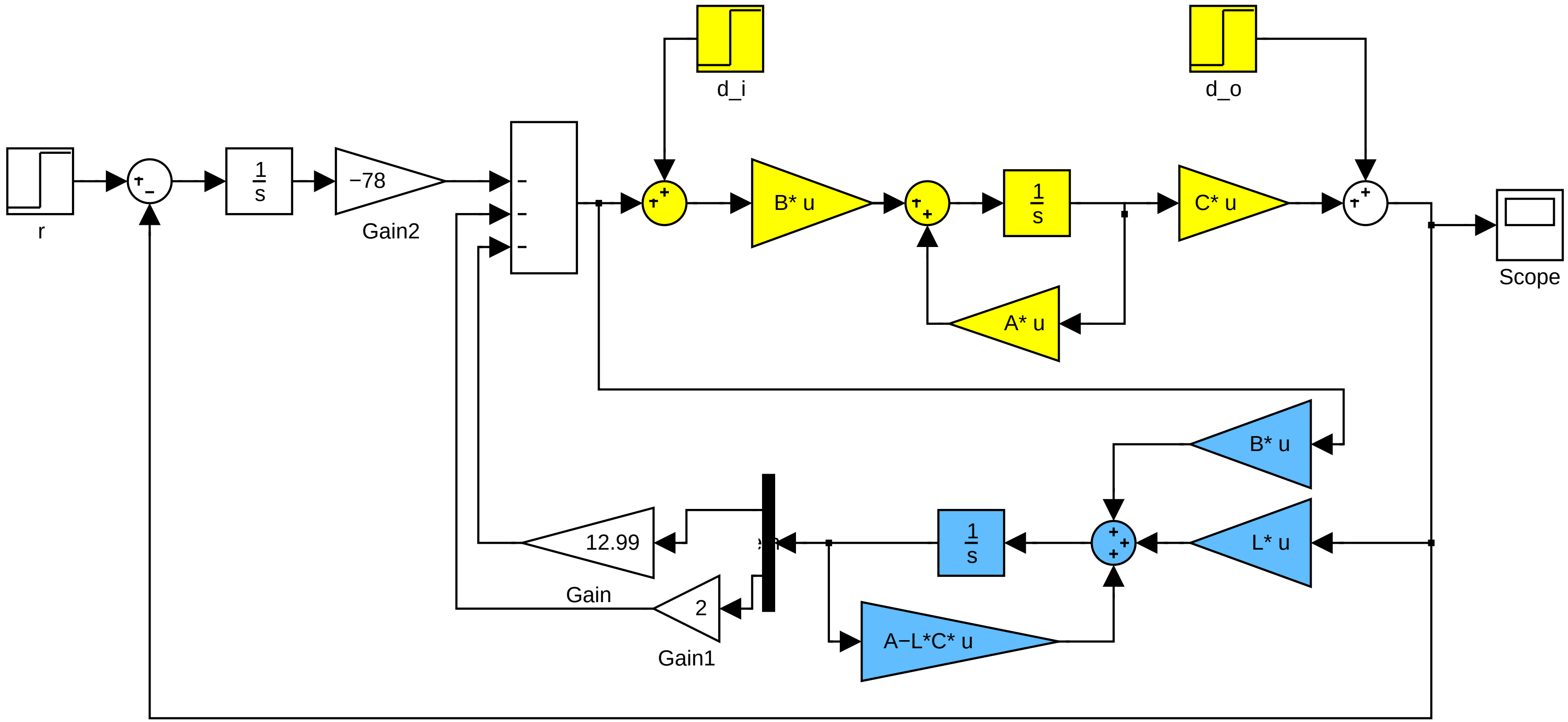




We can see how the estimation errors effectively converge to zero in about 1s.

State Estimation Example

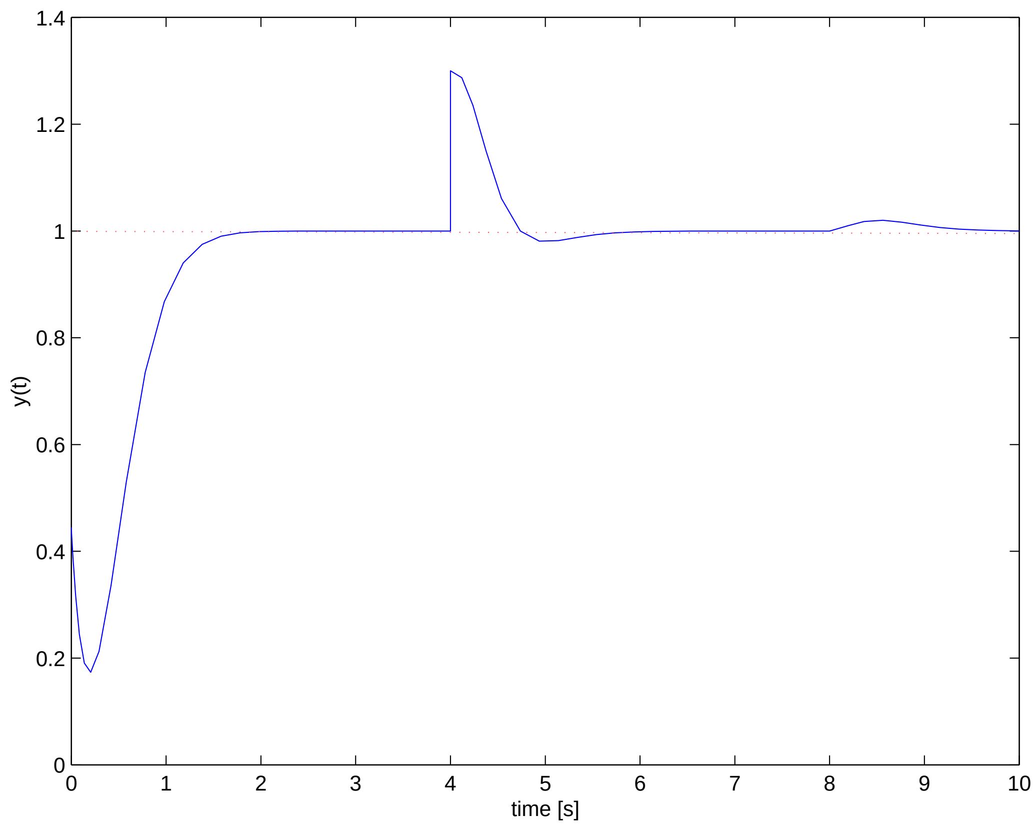
Example (Feedback from Estimated States). The observer can be combined with the previous state feedback design; **we just need to replace the state measurements by the estimated states**.



Note that for the integral action part we **still need to measure the real output** (its estimate won't work).

State Estimation Example

Example (continuation). The figure below shows the results of simulating the complete closed-loop system, with feedback from estimated states and integral action for robust reference tracking and disturbance rejection.



□

One Tip for Lab 2

Although we will further discuss state feedback and observers, we have now all the ingredients necessary for Laboratory 2 and Assignment 3.

One trick that may come handy in the state design for Lab 2 is **plant augmentation**. It consists of prefiltering the plant before carrying out the state feedback design.

The system for Lab 2 has a transfer function of the form

$$G(s) = \frac{k}{s(s+a)(s^2 + 2\zeta\omega s + \omega^2)}$$

When the damping ζ is small, the plant has a resonance at ω, as is the case in the system of Lab 2.

One Tip for Lab 2

Control design for a system with resonances is tricky. One way to deal with them is **prefilter** the plant with a **notch** filter of the form

$$F(s) = \frac{s^2 + 2\zeta\omega s + \omega^2}{s^2 + 2\bar{\zeta}\omega s + \omega^2}$$

with better damping $\bar{\zeta} = 0.7$ (say) and the **same natural frequency**. The notch filter cancels the resonant poles and replaces them by a pair of more damped poles. Of course, this can only be done for **stable** poles.

The augmented plant then becomes

$$G_a(s) = G(s)F(s) = \frac{k}{s(s+a)(s^2 + 2\bar{\zeta}\omega s + \omega^2)}$$

We then do state feedback design for this **augmented plant**.

Summary

- For a plant with a controllable state-space description, it is possible to design a state feedback controller $\mathbf{u}(\mathbf{t}) = -\mathbf{K}\mathbf{x}(\mathbf{t})$ that will place the closed-loop poles in any desired location.

Summary

- For a plant with a controllable state-space description, it is possible to design a state feedback controller $\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t)$ that will place the closed-loop poles in any desired location.
- State feedback control can incorporate **integral action**, which yields robust reference tracking and disturbance rejection for constant references and disturbances.

Summary

- For a plant with a controllable state-space description, it is possible to design a state feedback controller $\mathbf{u(t)} = -\mathbf{Kx(t)}$ that will place the closed-loop poles in any desired location.
- State feedback control can incorporate **integral action**, which yields robust reference tracking and disturbance rejection for constant references and disturbances.
- If the states are not all measurable, but the state-space description of the plant is also observable, then it is possible to estimate the states with an **observer**.

Summary

- For a plant with a controllable state-space description, it is possible to design a state feedback controller $\mathbf{u}(\mathbf{t}) = -\mathbf{K}\mathbf{x}(\mathbf{t})$ that will place the closed-loop poles in any desired location.
- State feedback control can incorporate **integral action**, which yields robust reference tracking and disturbance rejection for constant references and disturbances.
- If the states are not all measurable, but the state-space description of the plant is also observable, then it is possible to estimate the states with an **observer**.
- The observer can be used in conjunction with the state feedback by feeding back estimated estates.