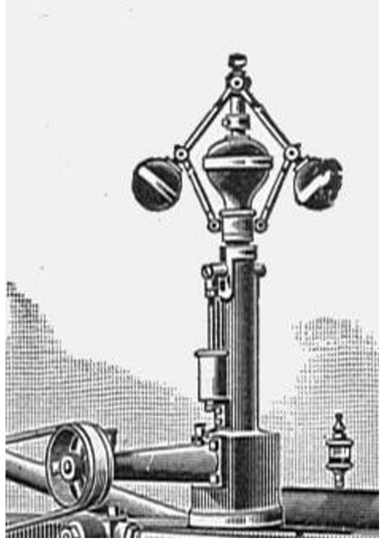
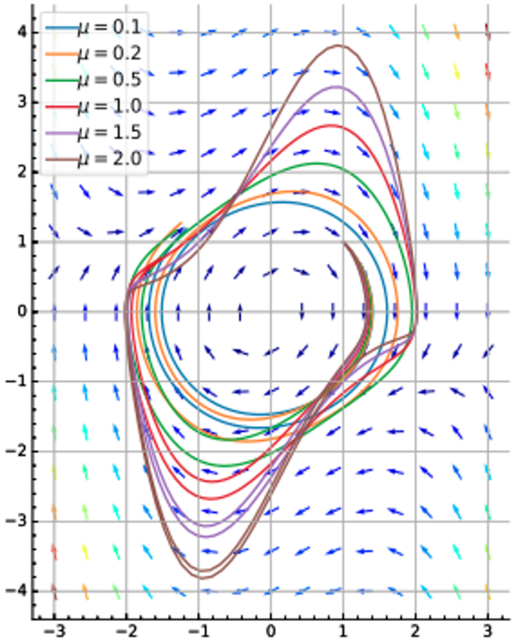
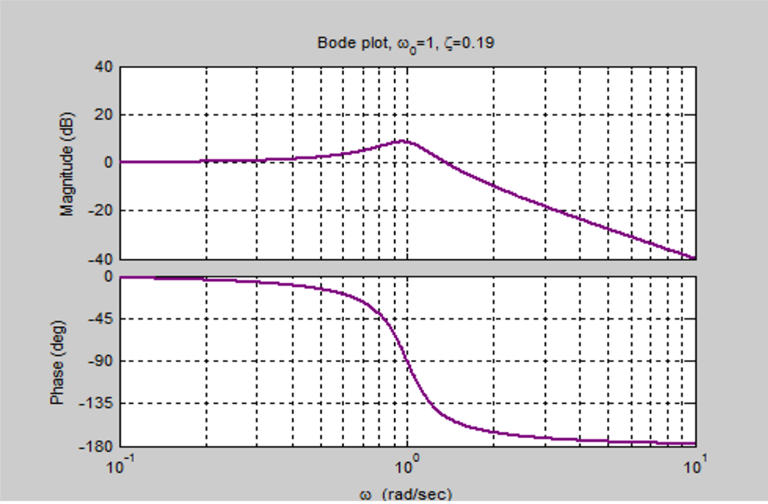


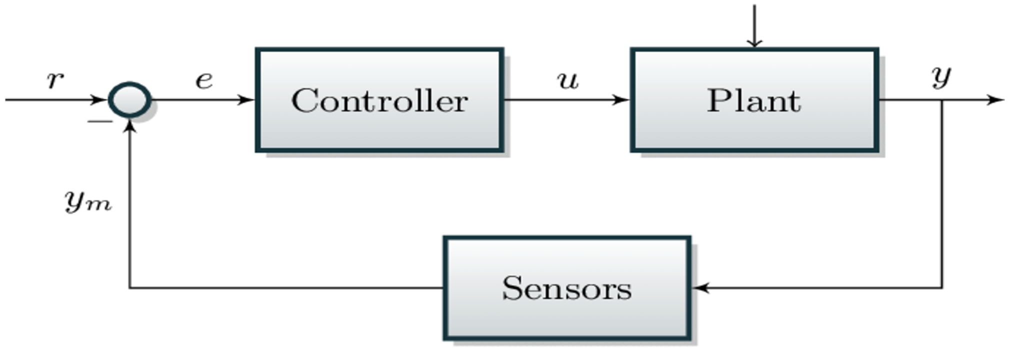


Introduction to Control Systems

Theory and applications

Enrico Regolin / Laura Nenzi





Course Overview (1)

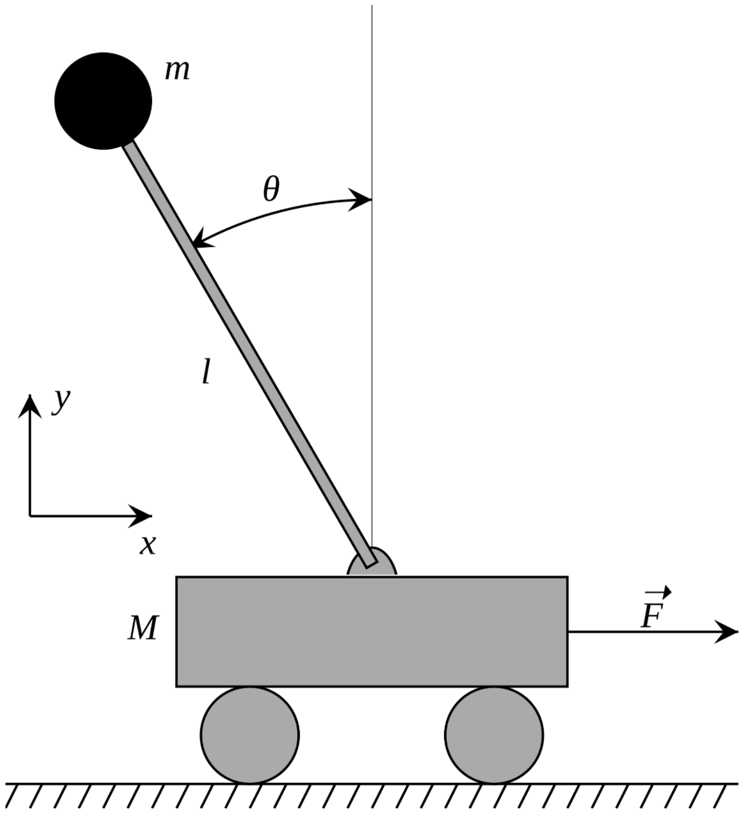
- **Linear Control (time domain)**
- Introduction
- Dynamical Linear Systems
- Observability & Controllability
- PID Controllers
- Luenberger Observer

- **Linear Control (frequency domain)**
- From State-space to Transfer Function
- Classic Control Elements (Bode Diagram / Root Locus)
- Ctrl Lab (days 1,2)

Course Overview (2)

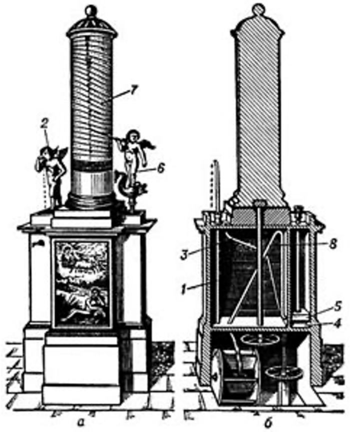
- Optimal Control and KF Estimation
  - Optimal Control (LQR)
  - Model Predictive Control
  - Kalman Filtering

- Control Laboratory
  - Matlab/Simulink
  - Kalman Filtering and Optimal Control
  - Cart-pole



Control Systems History

- Water Clock
  - Alexandria
    (Ctesibius, 3rd century BC)



- Centrifugal Governor
  - Windmills
    (C. Huygeens, 17th century)
  - Steam Engine
    (J. Watt, 1788)

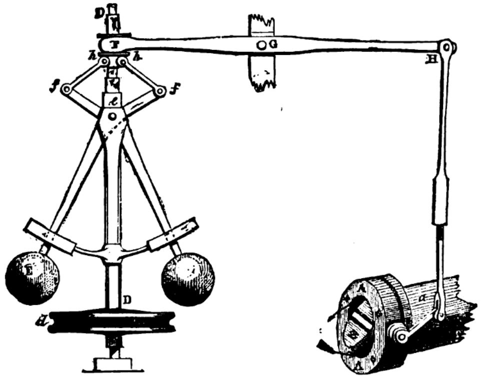
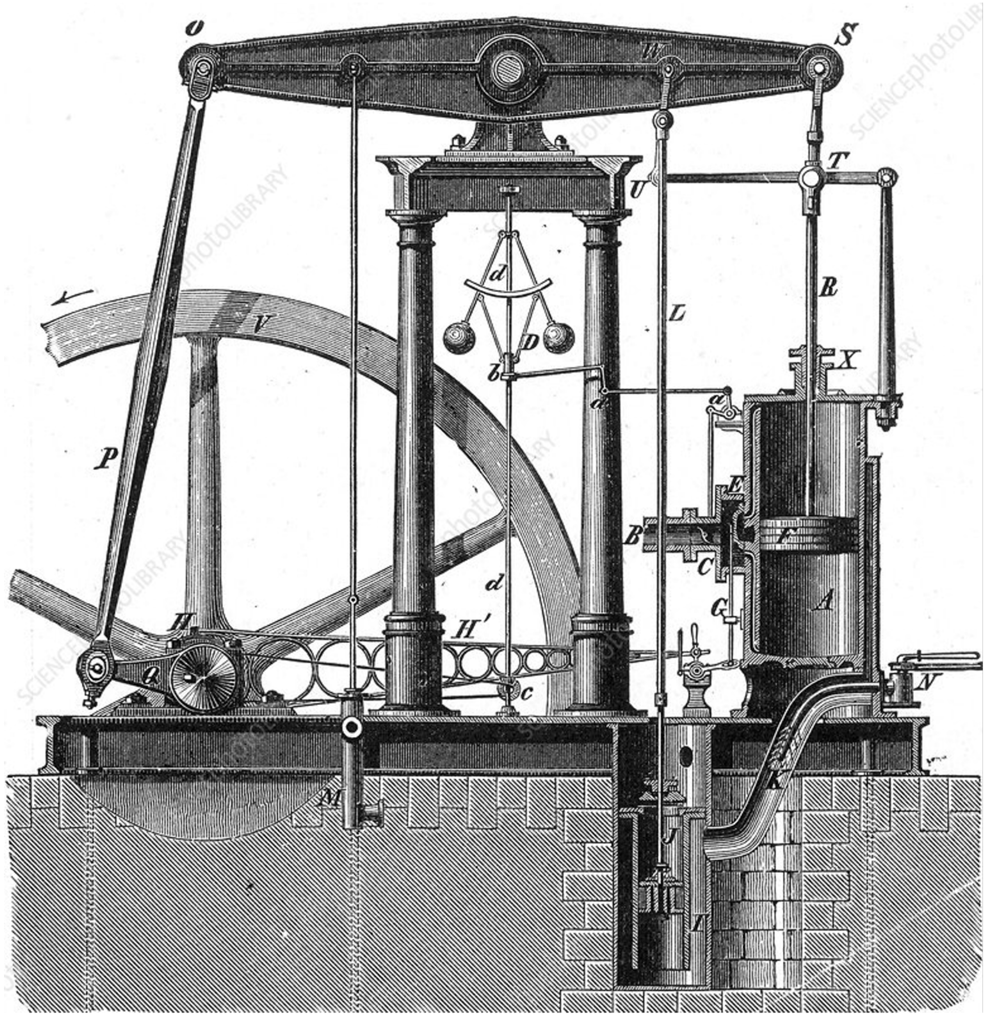
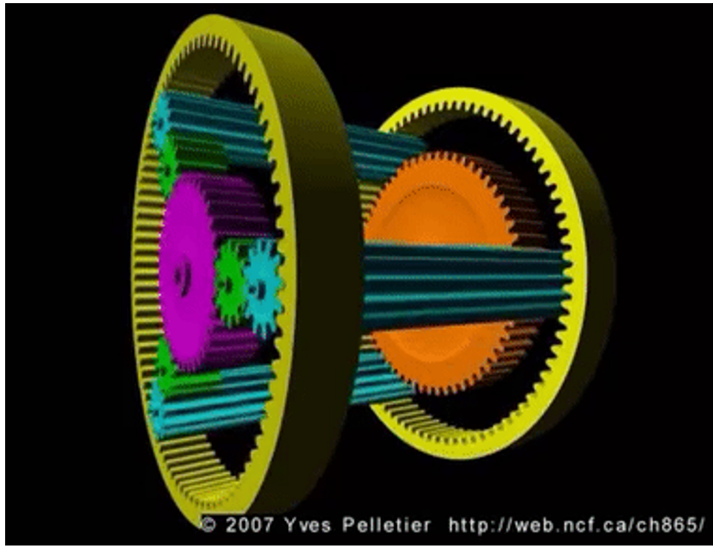


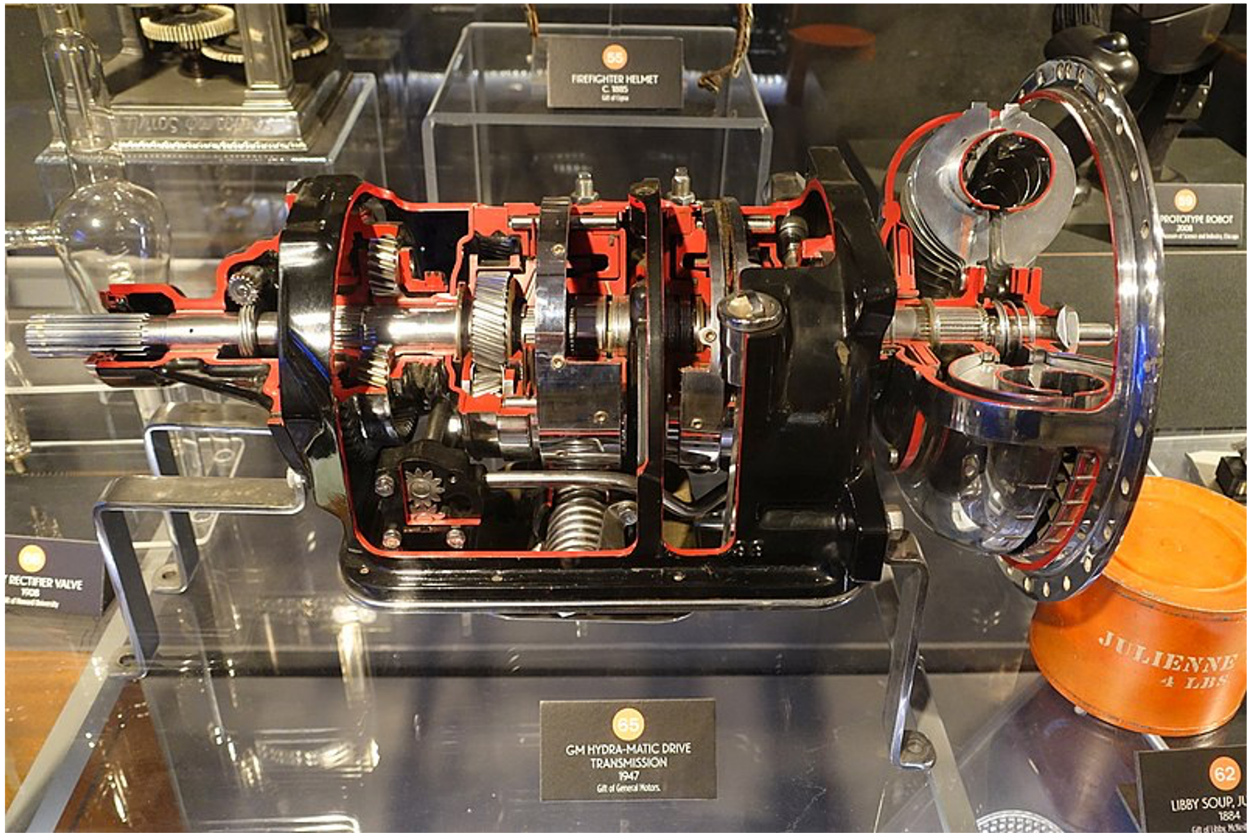
FIG. 4.---*Governor and Throttle-Valve.*



Control Systems History

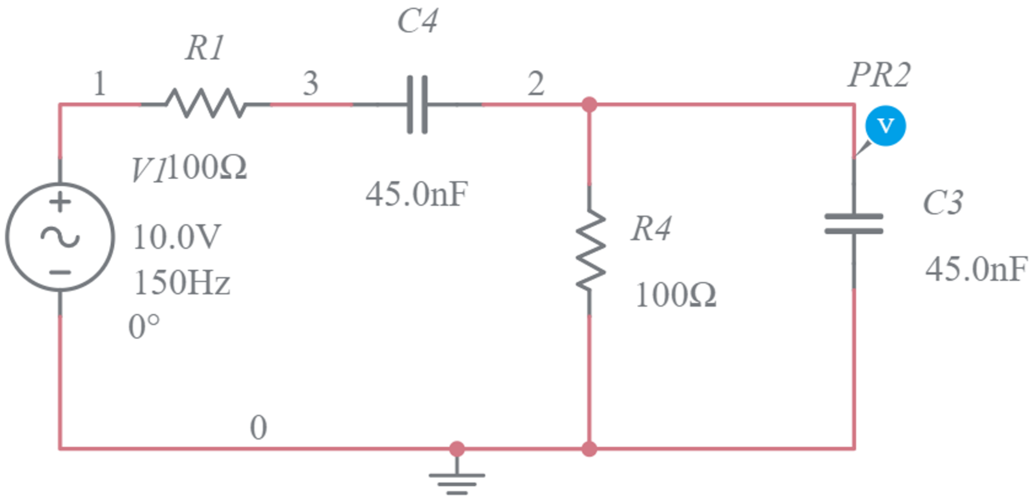
- First Automatic Transmission (Hydramatic, 1939)

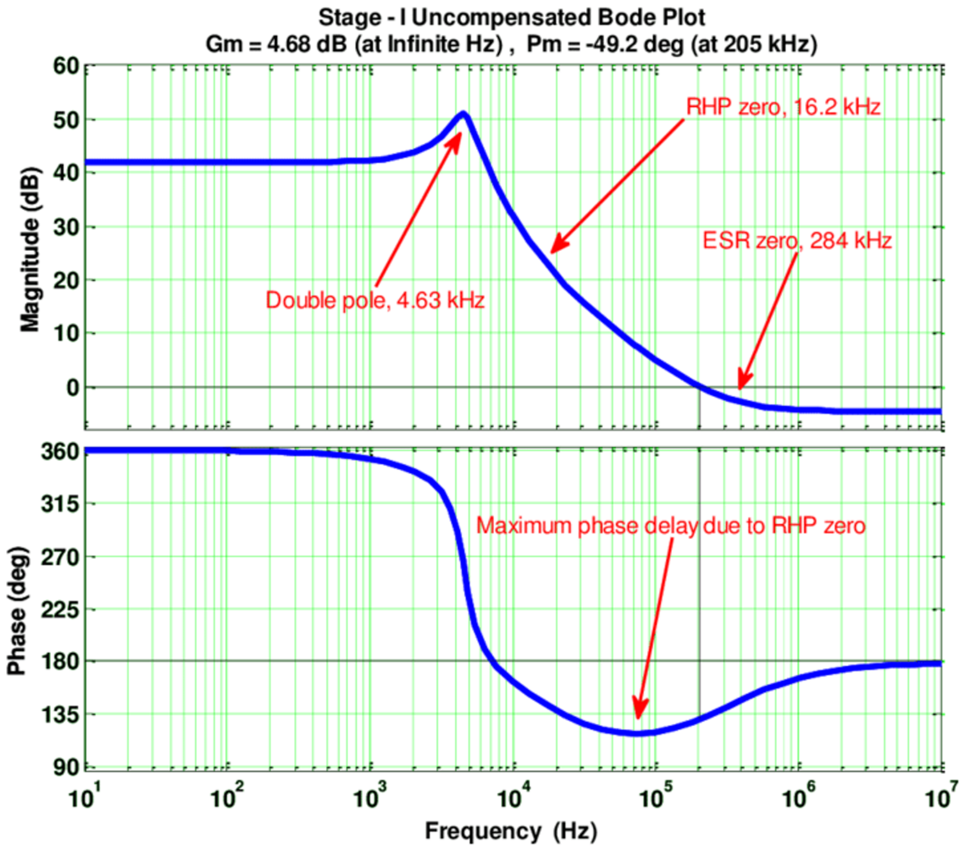




Control Systems History

- Classical control theory formalized from circuits theory





Tacoma Bridge Collapse

Linear Control (time domain)

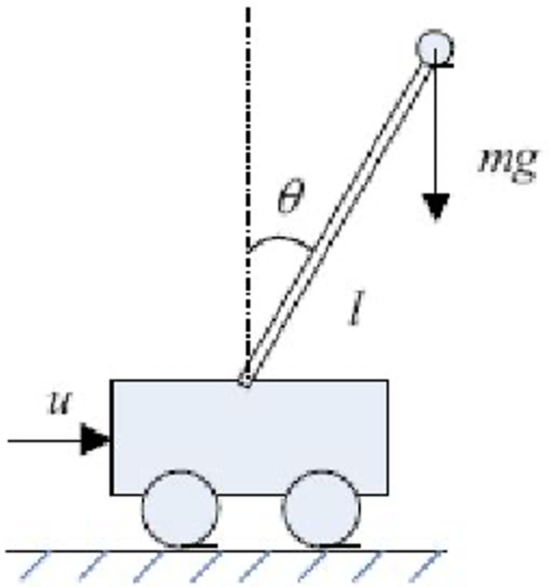
Control Systems Fundamentals

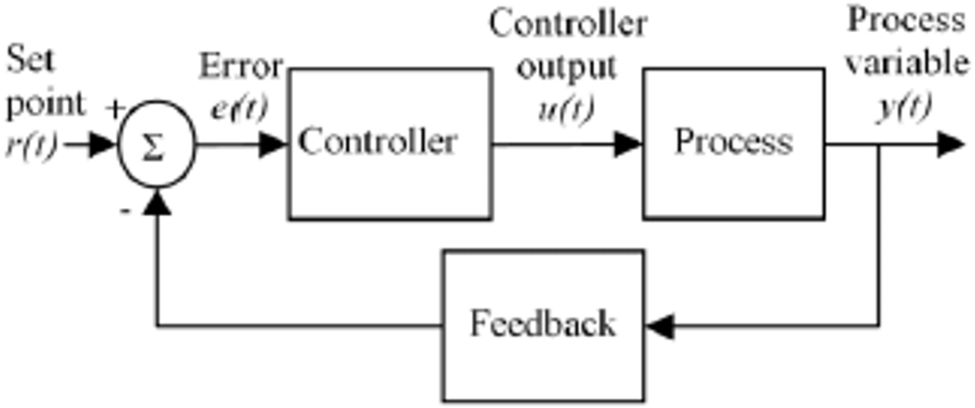
REQUIRED

- Dynamical System MODEL
- Control Input (non-autonomous systems)
- Reference Signal
-

CHALLANGES

- Missing/Noisy Information
- Physical limitations





Dynamical Systems (1)

Past history (state) influences future output

- **Continuous Time** vs. **Discrete Time**

$$\dot{x} = f(x), \quad t \in [0, \infty)$$

$$x(k+1) = f(x(k)), \quad k = 0, 1, 2, \ldots$$

- **Autonomous** vs. **Non-autonomous**

$$\dot{x} = f(x)$$

$$\dot{x} = f(x, u)$$

- **Linear** vs. **Non-linear**

$$\dot{x}_1 = -2x_2$$
$$\dot{x}_2 = 0.5x_1 + x_2 + 0.4u$$

$$\dot{x}_1 = -x_1 x_2$$
$$\dot{x}_2 = 0.5x_1^2 + \sin(x_2) + \frac{0.4}{u}$$

Dynamical Systems (2)

- **SISO** vs. **MIMO**

$$\dot{x} = Ax + b \cdot u$$
$$y = Cx (= 0.5x_1)$$

$$\dot{x} = Ax + B\mathbf{u}$$
$$\mathbf{y} = Cx$$

- **Time Invariant** vs. **Time Variant**

$$\dot{x} = f(x, u)$$
$$\dot{x} = Ax + Bu$$

$$\dot{x}(t) = f(x(t), u(t), t)$$
$$\dot{x}(t) = A(t)x(t) + B(t)u(t)$$

- **Deterministic** vs. **Non-Deterministic (Stochastic, noisy, etc.)**

$$\dot{x} = -x^2 - x + u$$
$$y = 0.5x$$

$$x(k+1) = -(2+\nu)x(k)^2 - x(k) + u(k)$$
$$y(k) = 0.5x(k) + \eta$$
$$\nu \sim N(\mu, \sigma), \eta \sim U(0, 1)$$

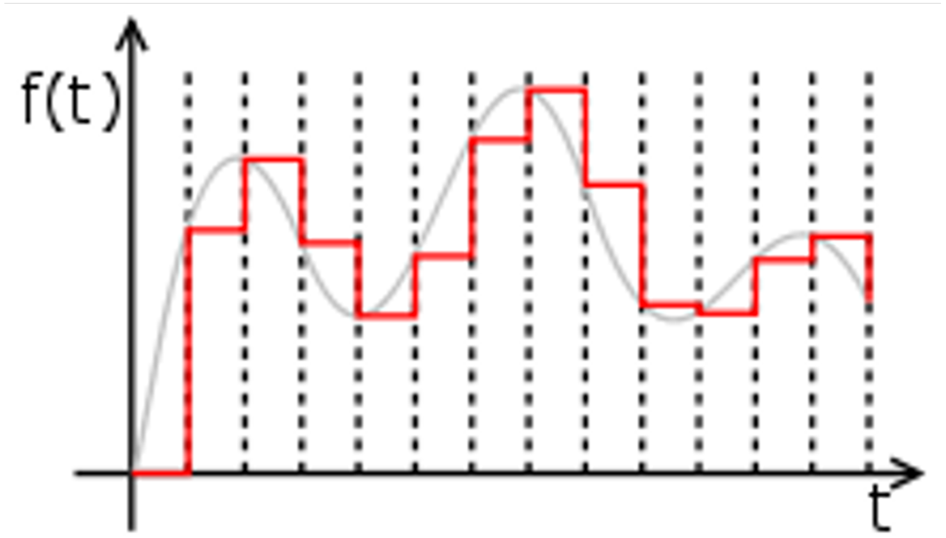
Dynamical Systems (3)

•LTI systems --- State-Space representation $x(0) = x_0,\ x \in \mathbb{R}^n$

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

$$A_d = e^{A\Delta T}$$
$$B_d = A^{-1}(e^{A\Delta T} - 1)B$$

$$x(k+1) = A_d x(k) + B_d u(k)$$
$$y(k) = Cx(k) + Du(k)$$



Dynamical Systems (3)

- **LTI systems --- State-Space representation** $x(0) = x_0, \; x \in \mathbb{R}^n$

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

$$A_d = e^{A\Delta T}$$
$$B_d = A^{-1}(e^{A\Delta T} - 1)B$$

→

$$x(k+1) = A_d x(k) + B_d u(k)$$
$$y(k) = Cx(k) + Du(k)$$

- **Output response (continuous time)**

$$y(t) = \boxed{Ce^{At}x_0} + \boxed{C\int_0^t e^{A(t-\tau)}Bu(\tau)d\tau} + Du(t)$$

Free Response (homogeneous solution)

Effect of input

- **Output response (discrete time)**

$$y(k) = CA_d^k x_0 + C\sum_{i=0}^{k-1} A_d^{k-1-i}B_d u(i) + Du(k)$$

Stability condition (Hurwitz)

$$x(t) = e^{at}$$

$a < 0$ $a > 0$

$$real(eig(A)) < 0$$

$$x(k) = a^k$$

$|a| < 1$ $|a| > 1$

$$|eig(A_d)| < 1$$

State-Space Realizations

Similarity Transformations

- The choice of a state-space model for a given system is not unique.
- For example, let T be an invertible matrix, and consider a coordinate transpormation $x = T\tilde{x}$, i.e., $\tilde{x} = T^{-1}x$. This is called a similarity transformation.
- The standard state-space model can be written as

$$\begin{cases} \dot{x} &= Ax + Bu, \\ y &= Cx + Du. \end{cases} \Rightarrow \begin{cases} T\dot{\tilde{x}} &= AT\tilde{x} + Bu, \\ y &= CT\tilde{x} + Du. \end{cases}$$

i.e.,

$$\begin{aligned} \dot{\tilde{x}} &= (T^{-1}AT)\tilde{x} + (T^{-1}B)u = \tilde{A}\tilde{x} + \tilde{B}u \\ y &= (CT)\tilde{x} + Du = \tilde{C}\tilde{x} + \tilde{D}u. \end{aligned}$$

- You can check that the time response is exactly the same for the two models (A, B, C, D) and $(\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D})$!

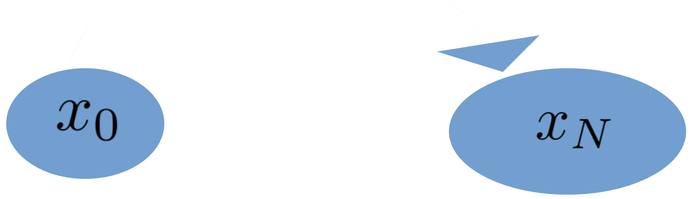
LTI Systems Properties

Discrete case

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k)$$

Reaching a state

$$u_0, u_1, \dots u_{N-1}$$



"Observing" the initial state

$$y_N, y_{N-1}, \dots y_0$$

LTI Systems Properties

Conditions for all LTI systems:

- Controllability $\iff rank(\mathcal{C}) = n$

$$\mathcal{C} = \left[B, AB, A^2B, \ldots, A^{n-1}B\right]$$

- Observability $\iff rank(\mathcal{O}) = n$

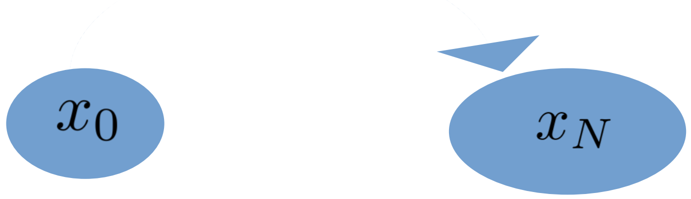
$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \cdots \\ CA^{n-1} \end{bmatrix}$$

Discrete case

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k)$$

Reaching a state

$$u_0, u_1, \ldots u_{N-1}$$



"Observing" the initial state

$$y_N, y_{N-1}, \cdots y_0$$

LTI Systems Properties

- Pair (A,B) is "Controllable" $\Leftrightarrow rank(\mathcal{C}) = n$
- Pair (A,C) is "Observable" $\Leftrightarrow rank(\mathcal{O}) = n$
- LTI System $\mathcal{S} : \{A, B, C\}$ is a "**minimal state-space realization**" if it is both observable and controllable.
- Example:

$$\mathcal{S}_0 : \{A_0, B, C\}, \quad \mathcal{S}_1 : \{A_1, B, C\}$$

$$B = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \qquad C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

$$A_0 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & -1 & 2 \end{bmatrix}$$

$$\mathcal{C}_0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \mathcal{O}_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

$$rank(\mathcal{C}_0) = 1 \qquad rank(\mathcal{O}_0) = 2$$

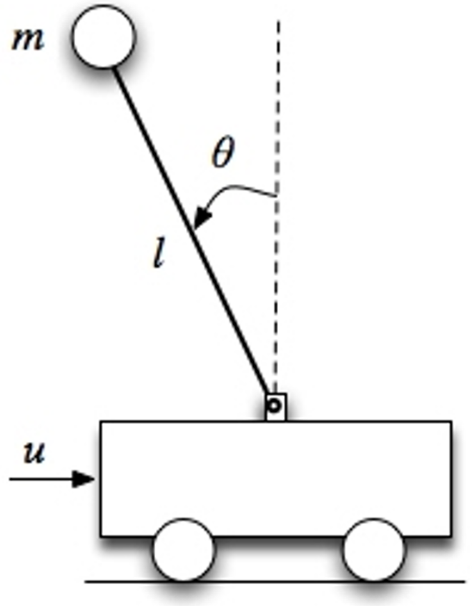
$$\mathcal{C}_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 2 \\ 1 & 2 & 3 \end{bmatrix} \quad \mathcal{O}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$rank(\mathcal{C}_1) = 3 \qquad rank(\mathcal{O}_1) = 3$$

non-LTI Systems (example)

Is the inverted pendulum (cartpole) controllable?

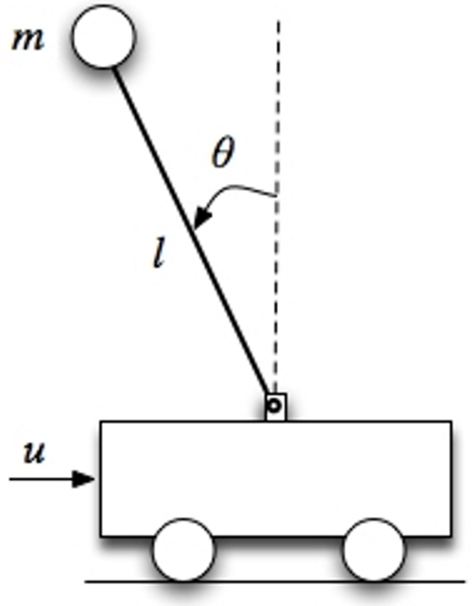
$$\begin{cases} \ddot{p} & = \frac{u + m\,l\,\dot{\theta}^2\ \sin\theta - m\,g\ \cos\theta\sin\theta}{M + m\sin\theta^2} \\ \ddot{\theta} & = \frac{g\ \sin\theta - \cos\theta\ddot{p}}{l} \end{cases}$$



In non-linear systems Controllability and Observability Matrices represent LOCAL properties.

non-LTI Systems (example)

Is the inverted pendulum (cartpole) controllable?



$$\begin{cases} \ddot{p} & = \frac{u + m\, l\, \dot{\theta}^2 \; \sin\theta - m\, g \; \cos\theta \sin\theta}{M + m \sin\theta^2} \\ \ddot{\theta} & = \frac{g \; \sin\theta - \cos\theta \ddot{p}}{l} \end{cases}$$

In non-linear systems Controllability and Observability Matrices represent LOCAL properties.

$$\dot{x} = f(x,u), \quad \text{eq.point } x_0, u_0$$

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

$$\underline{A} = \frac{\partial f(x,u)}{\partial x}|_{x=x_0, u=u_0}$$

$$\underline{B} = \frac{\partial f(x,u)}{\partial u}|_{x=x_0, u=u_0}$$

$$x = \left[p,\ \dot{p},\ \theta,\ \dot{\theta}\right]^T$$

$$\frac{\partial f}{\partial u} = \left[0,\ \frac{1}{(M + m(1 - \cos^2(\theta))},\ 0,\ \frac{-\cos(\theta)}{l\,(M + m(1 - \cos^2(\theta))}\right]^T$$

non-LTI Systems (example)

Linearization

$$\dot{x} = f(x,u), \quad \text{eq.point } x_0, u_0$$

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

$$\underline{A} = \frac{\partial f(x,u)}{\partial x}|_{x=x_0,u=u_0}$$

$$\underline{B} = \frac{\partial f(x,u)}{\partial u}|_{x=x_0,u=u_0}$$

$$(\dot{x} = 0,\ \theta_0 = 0,\ \dot{\theta}_0 = 0,\ u_0 = 0)$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -gm/M & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \alpha & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1/M \\ 0 \\ -1/(Ml) \end{bmatrix}$$

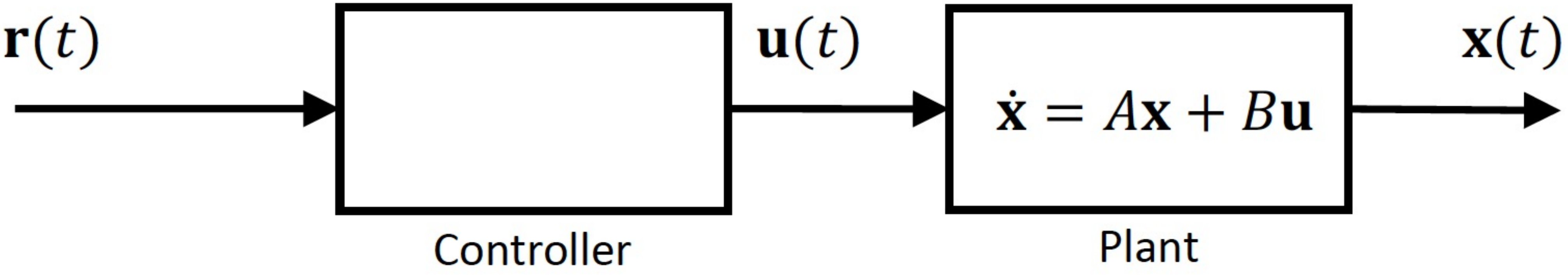
$$\alpha = \frac{(m+M)g}{Ml}$$

$$M = 1,\ m = 0.1,\ g = 9.81,\ l = 0.5$$

$$\mathcal{C} \approx \begin{bmatrix} 0 & 1 & 0 & 2 \\ 1 & 0 & 2 & 0 \\ 0 & -2 & 0 & -43 \\ -2 & 0 & -43 & 0 \end{bmatrix}$$

$$rank(\mathcal{C}) = 4$$

Reference Tracking



Given a reference trajectory r(t), design u(t) such that x(t) closely follows r(t)

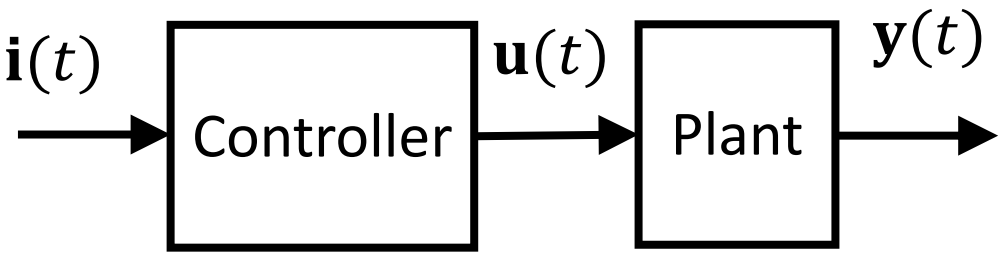
Control objectives:

- Reject disturbances (if there is some perturbation in state, making it get back to initial state)
- Follow reference trajectories (if we want the system to have a certain $\boldsymbol{x_{ref}}$)
- Make system follow some other “desired behavior”

Open-loop vs. Closed-loop control

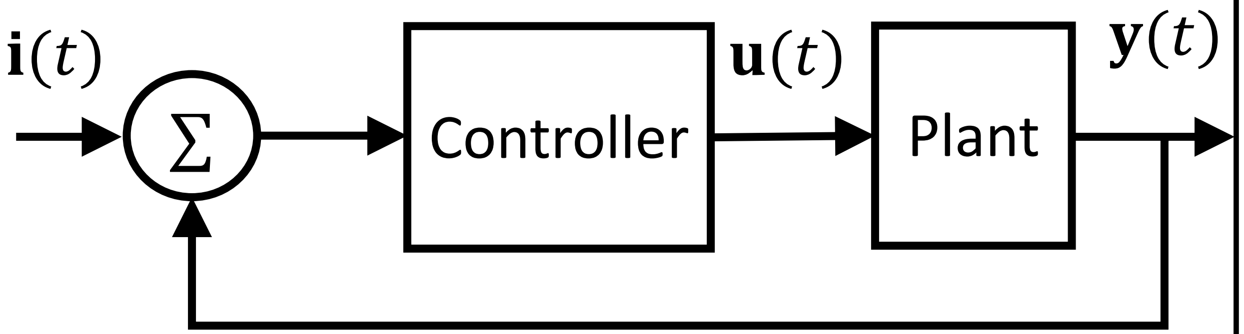
Open-loop or feed-forward control

- Control action does not depend on plant output
- Cheaper, no sensors required.
- Quality of control generally poor without human intervention

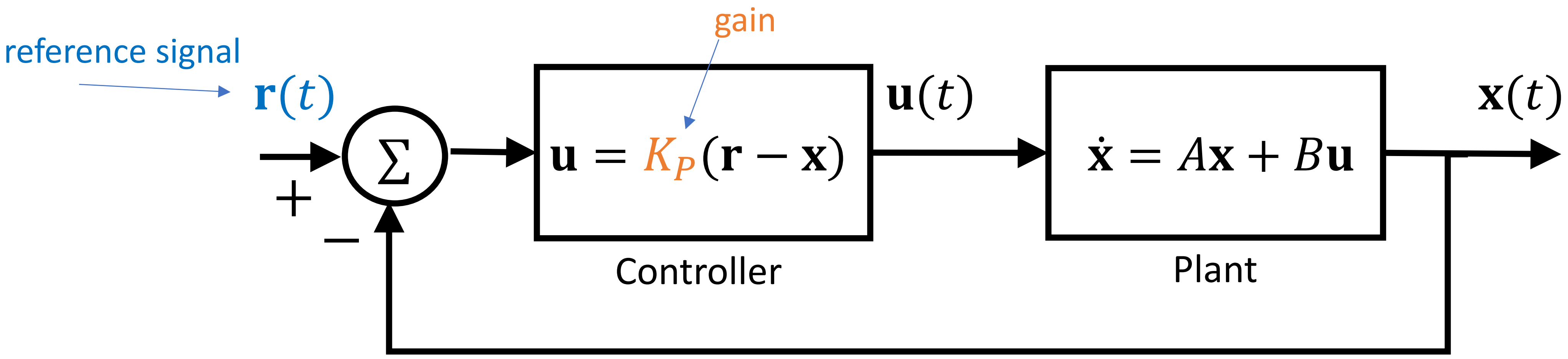


Feed-back control

- Controller adjusts controllable inputs in response to observed outputs
- Can respond better to variations in disturbances
- Performance depends on how well outputs can be sensed, and how quickly controller can track changes in output

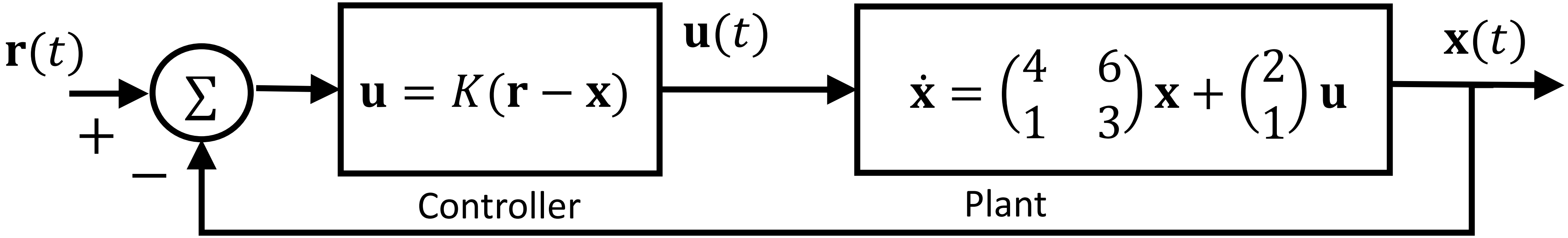


Proportional Controller



- Common objective: make plant state *track* the reference signal $\mathbf{r}(t)$
- $e = r - x$ is the error signal
- Closed-loop dynamics: $\dot{\mathbf{x}} = A\mathbf{x} + BK_P(\mathbf{r} - \mathbf{x}) = (A - BK_P)\mathbf{x} + BK_P\mathbf{r}$
- pick K_P s.t. the composite system is asymptotically stable, i.e. pick K_P such that eigenvalues of $(A - BK)$ have negative real-parts

Designing a pole placement controller

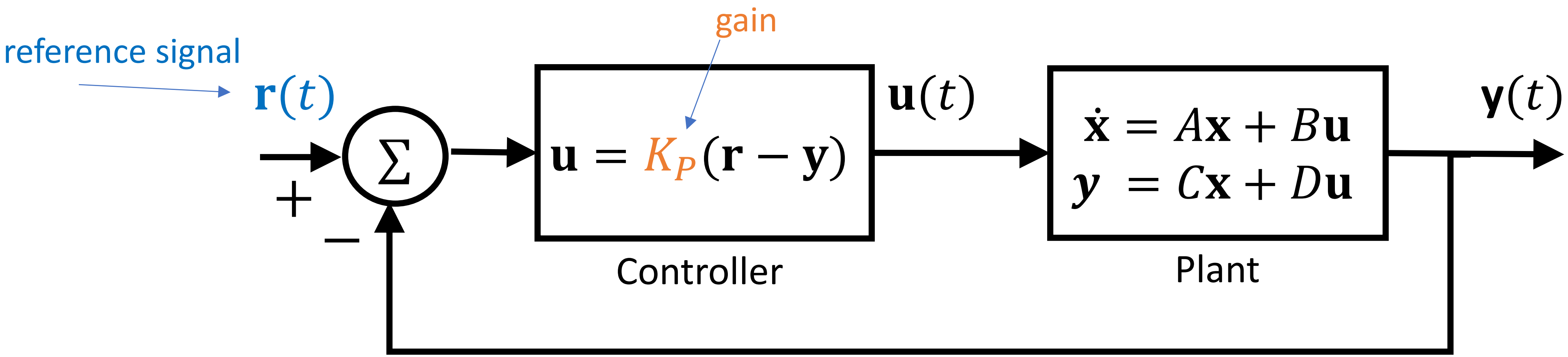


- eigs(A) are values of λ that satisfy the equation $\det(A - \lambda I) = 0$
- Note eigs$(A) = 6, 1 \Rightarrow$ unstable plant!

- Let $K = (k_1 \quad k_2)$. Then, $A - BK = \begin{pmatrix} 4 - 2k_1 & 6 - 2k_2 \\ 1 - k_1 & 3 - k_2 \end{pmatrix}$

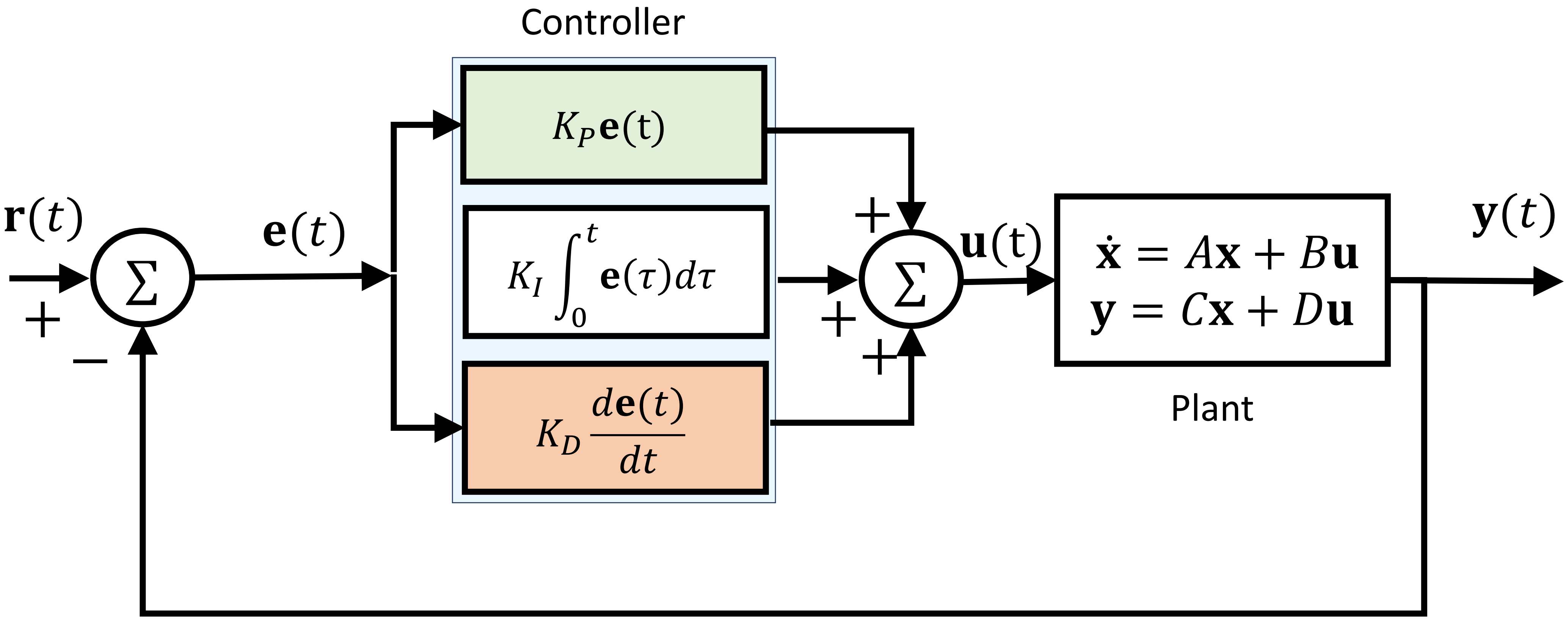
- eigs$(A - BK)$ satisfy equation $\lambda^2 + (2k_1 + k_2 - 7)\lambda + (6 - 2k_2) = 0$
  - two distinct solutions λ_1, λ_2 if $(\lambda - \lambda_1)\,(\lambda - \lambda_2) = \lambda^2 + (-\lambda_1 - \lambda_2)\lambda + \lambda_1\lambda_2$
  - That means $2k_1 + k_2 - 7 = -\lambda_1 - \lambda_2$ and $6 - 2k_2 = \lambda_1\lambda_2$
  - E.g. $\lambda_1 = -1$ and $\lambda_2 = -2$ gives $k_1 = 4, k_2 = 2$. Thus controller with $K = (4 \quad 2)$ stabilizes the plant!

Proportional Controller

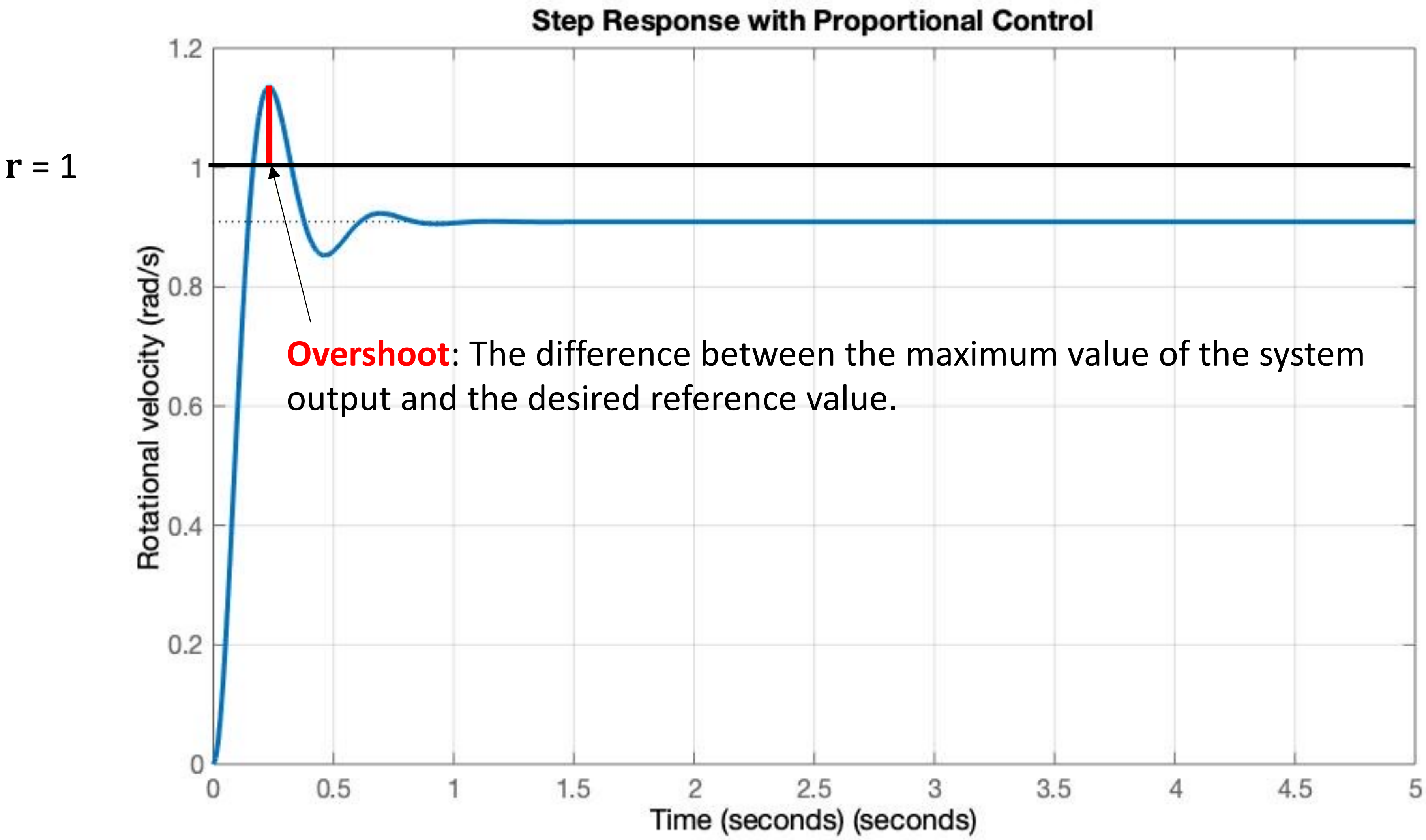


Proportional Integral Derivative (PID) controllers

eigs(A) are values of λ that satisfy the equation $\det(A - \lambda I) = 0$
Note eigs$(A) = 6, 1 \Rightarrow$ unstable plant!

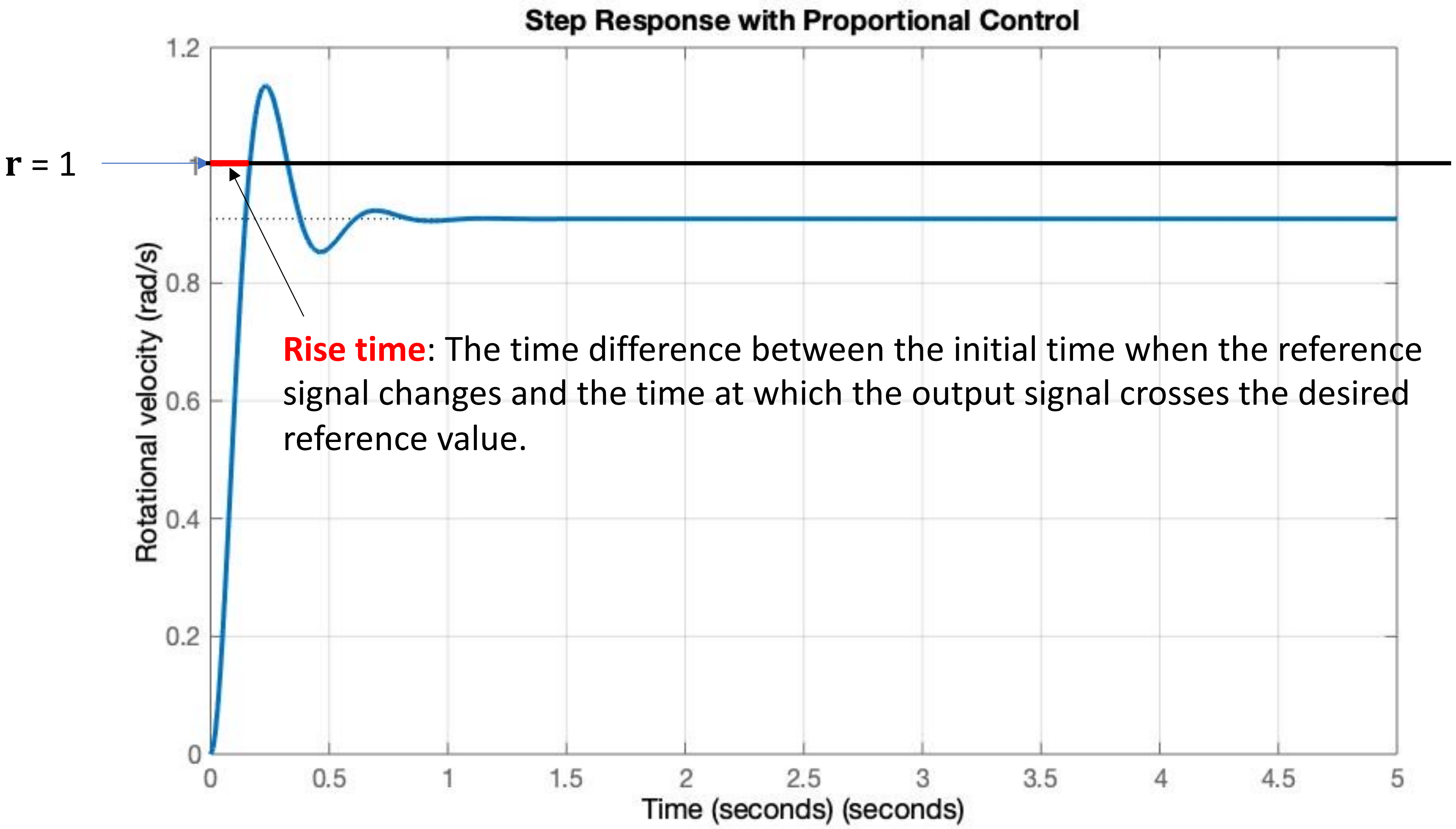


Measuring control performance



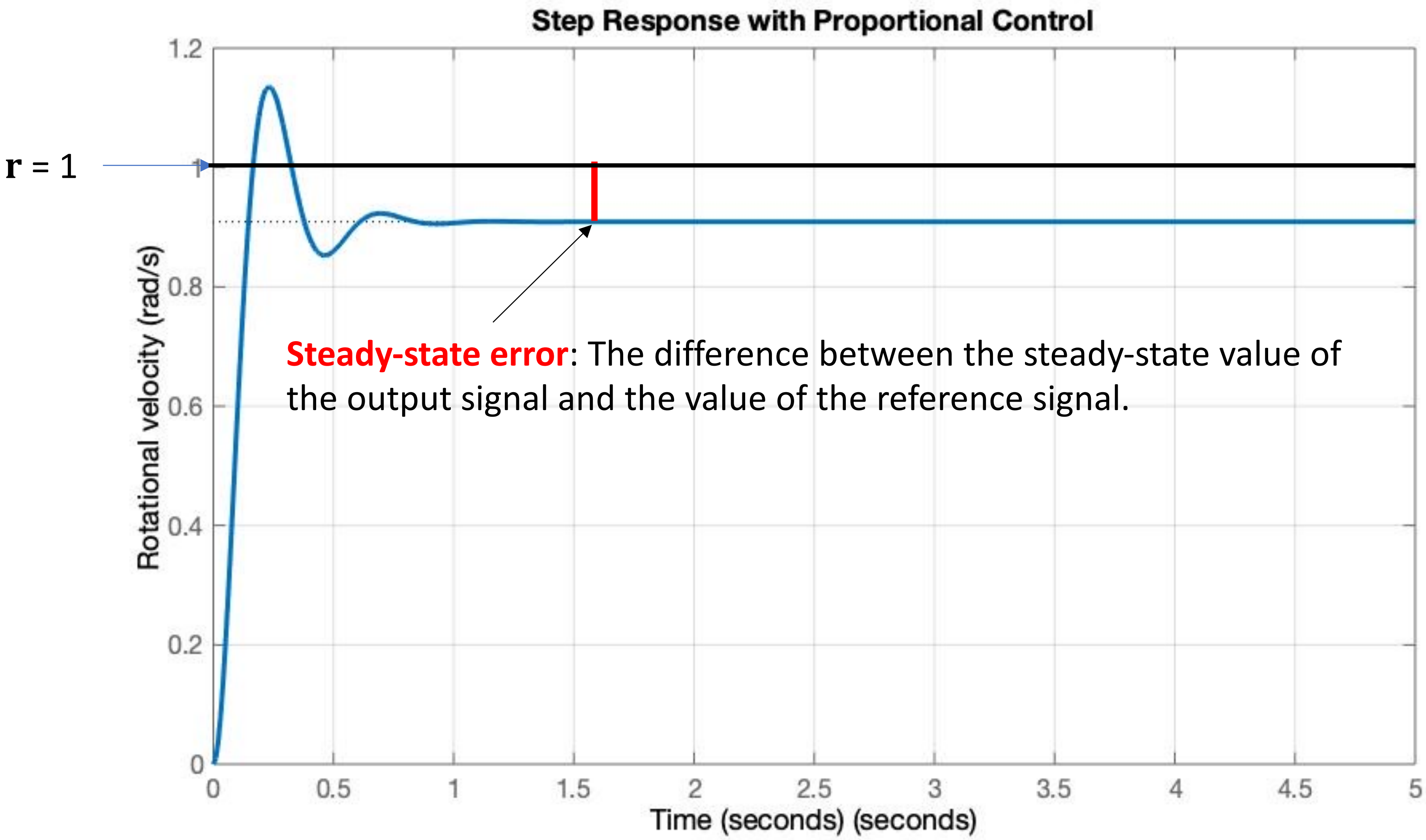
Overshoot: The difference between the maximum value of the system output and the desired reference value.

Measuring control performance

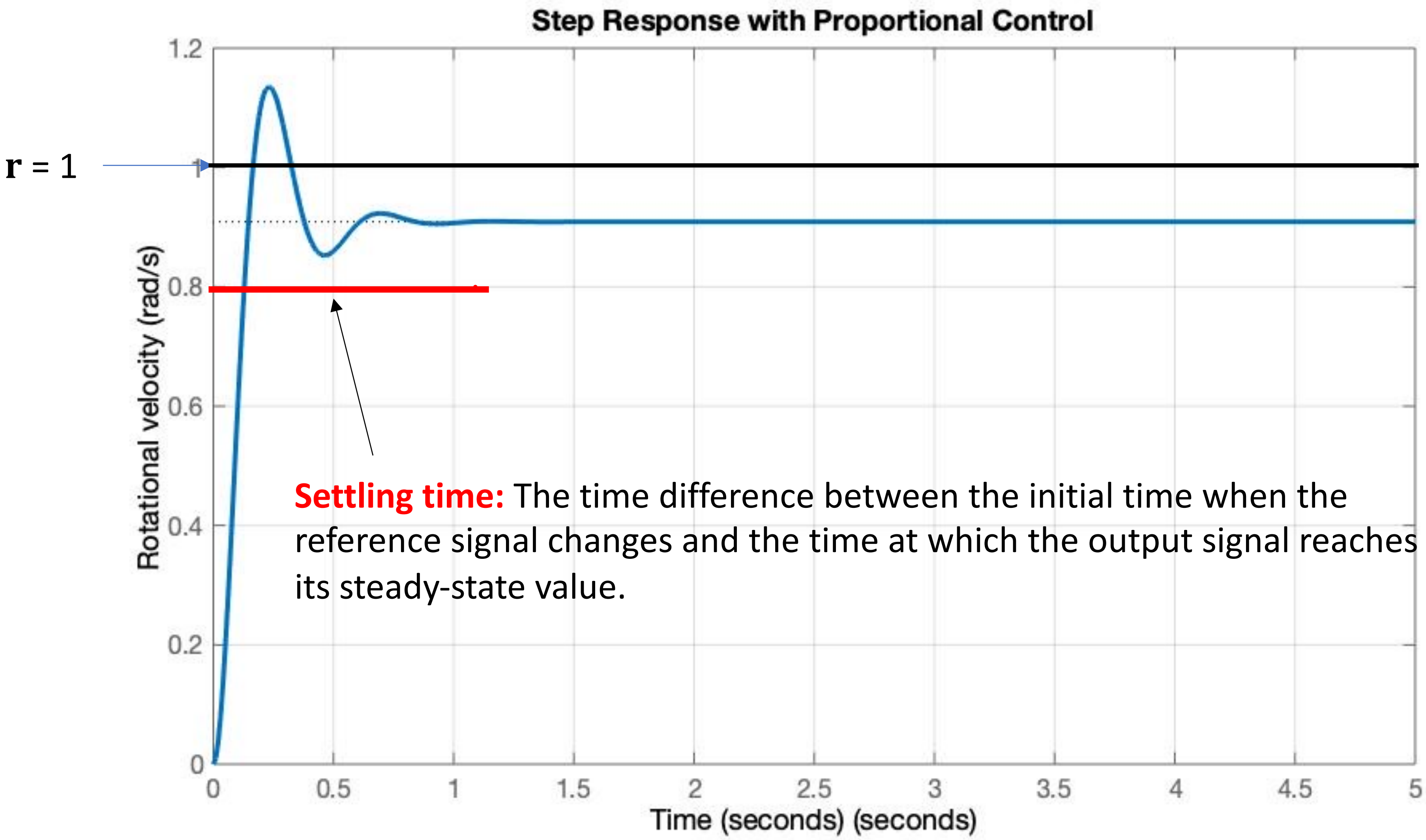


Rise time: The time difference between the initial time when the reference signal changes and the time at which the output signal crosses the desired reference value.

Measuring control performance

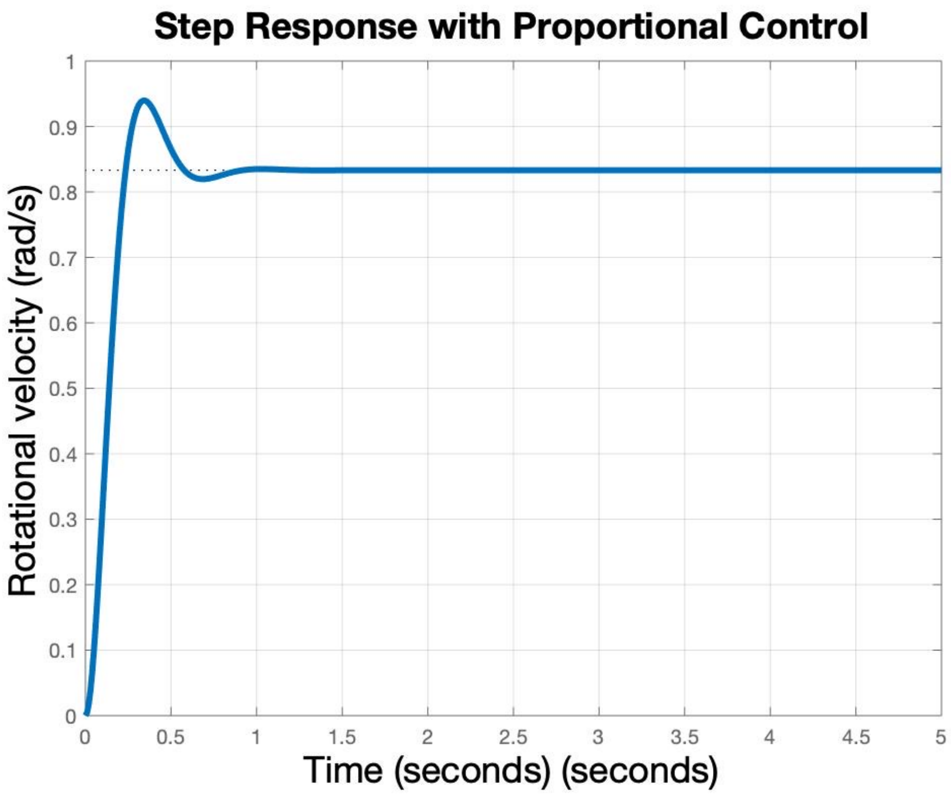


Measuring control performance

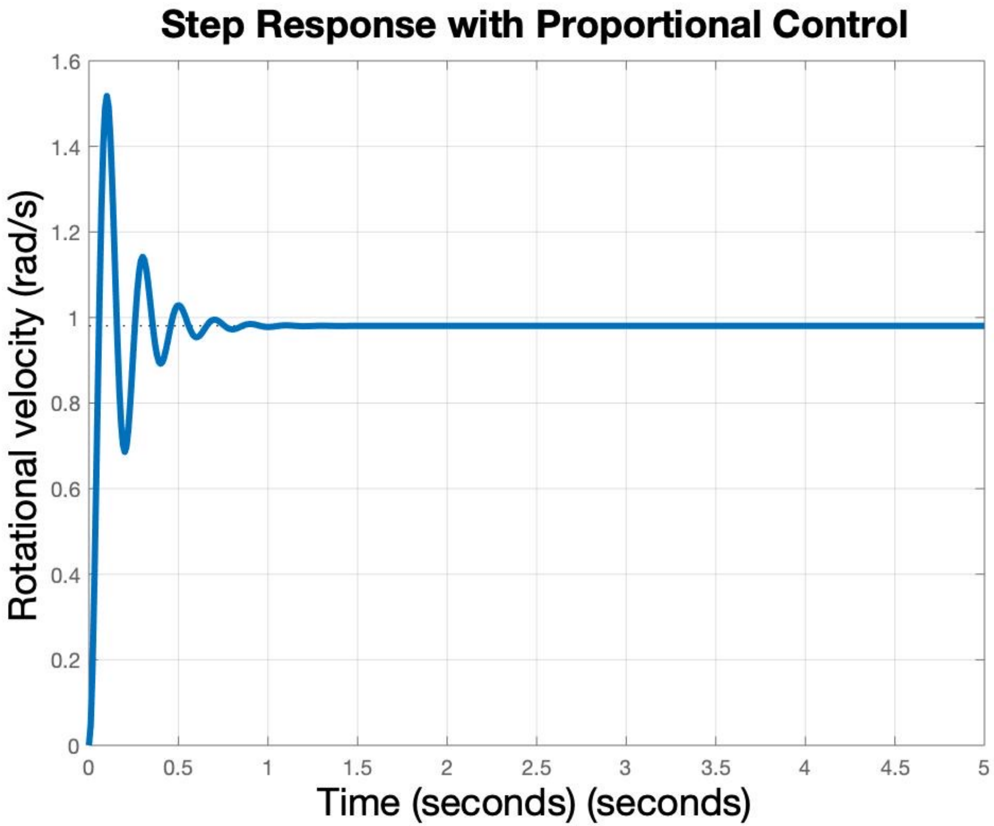


Settling time: The time difference between the initial time when the reference signal changes and the time at which the output signal reaches its steady-state value.

Measuring control performance



$\mathbf{K_P}$ = 50



$\mathbf{K_P}$ = 500

P-only controller

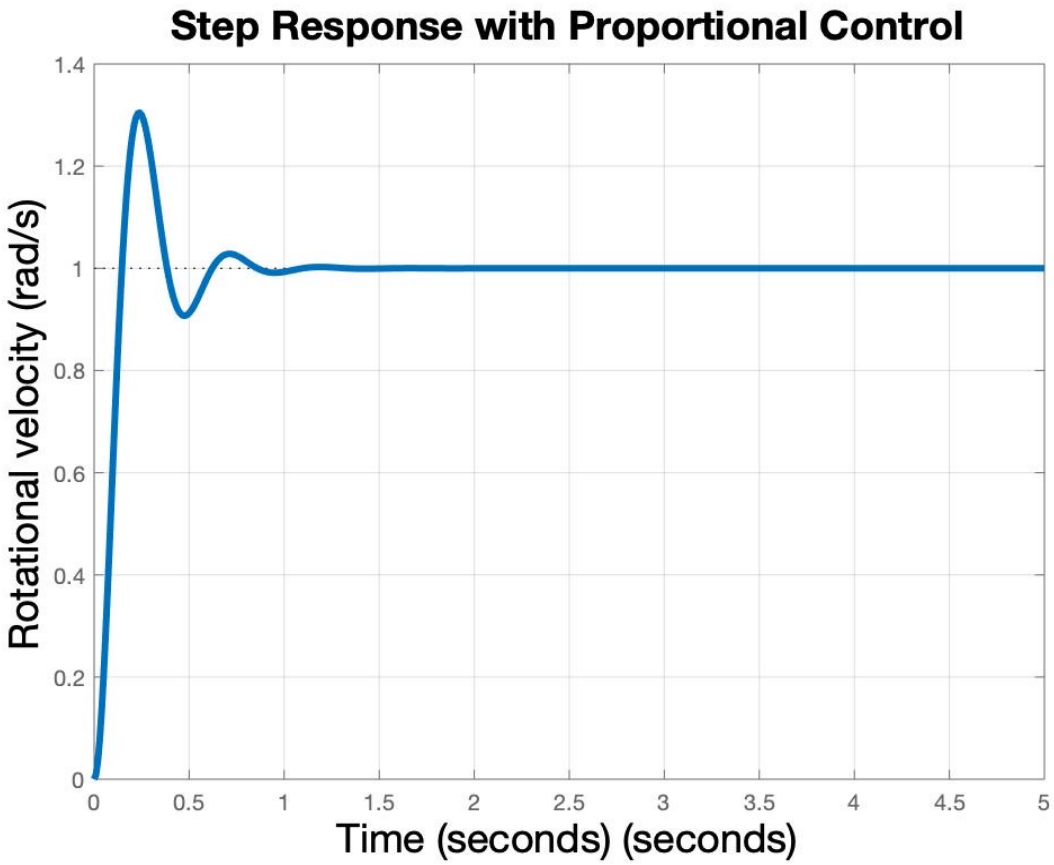
- Compute error signal $\mathbf{e} = \mathbf{r} - \boldsymbol{y}$
- Proportional term $K_p\mathbf{e}$:
  - K_p proportional gain;
  - Feedback correction proportional to error
- Cons:
  - If K_p is small, error can be large! [undercompensation]
  - If K_p is large,
    - system may oscillate (i.e. unstable) [overcompensation]
    - may not converge to set-point fast enough
  - P-controller always has steady state error or offset error

PI-controller

Compute error signal $\mathbf{e} = \mathbf{r} - \mathbf{y}$

Integral term: $K_I \int_0^t \mathbf{e}(\tau) d\tau$

- K_I integral gain;
- Feedback action proportional to cumulative error over time
- If a small error persists, it will add up over time and push the system towards eliminating this error): **eliminates offset/steady-state error**

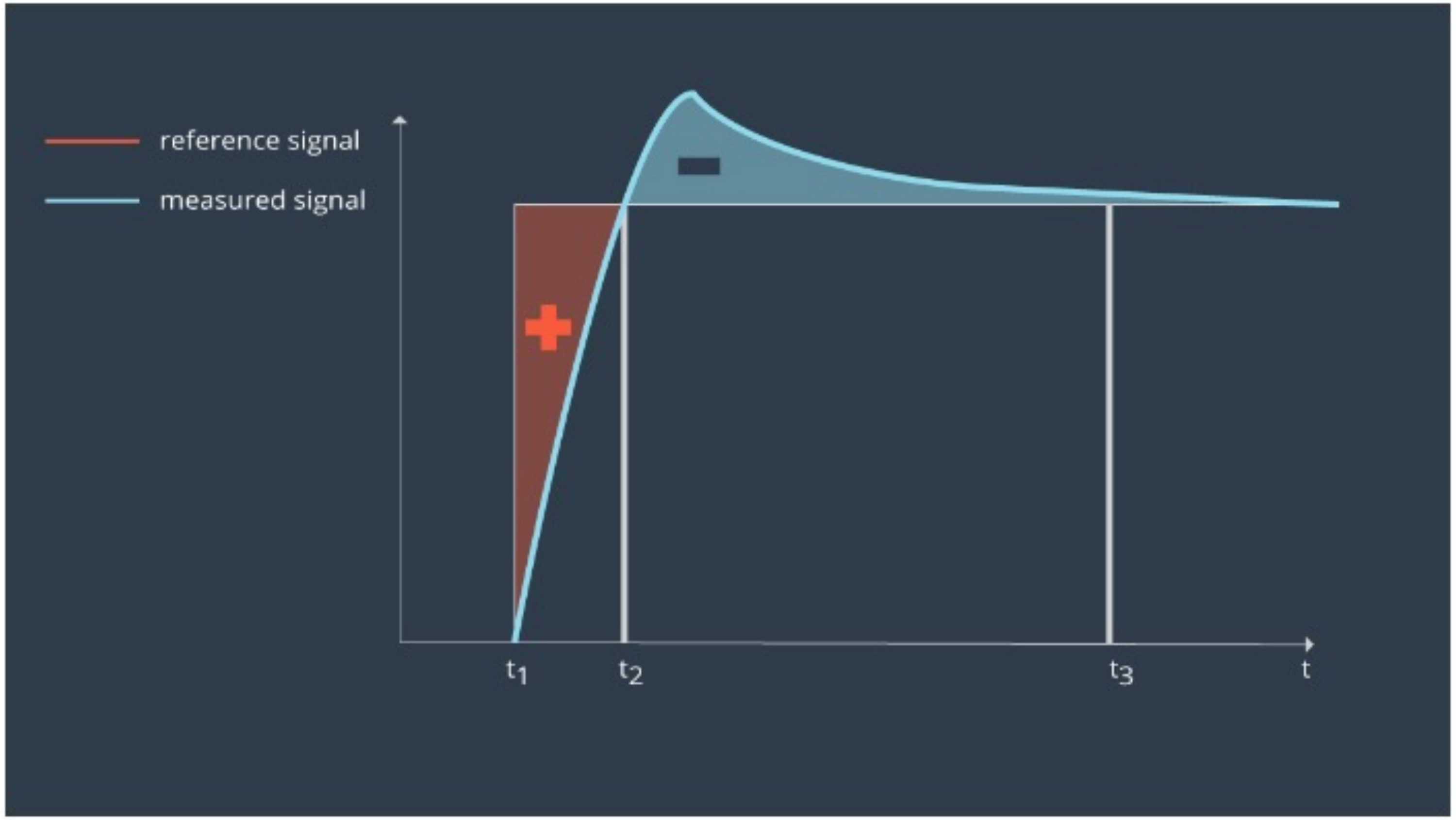


Disadvantages:

- Integral action by itself can increase instability
- Integrator term can accumulate error and suggest corrections that are not feasible for the actuators (integrator windup)
  - Real systems "saturate" the integrator beyond a certain value

PI-controller

Integrator windup



PD-controller

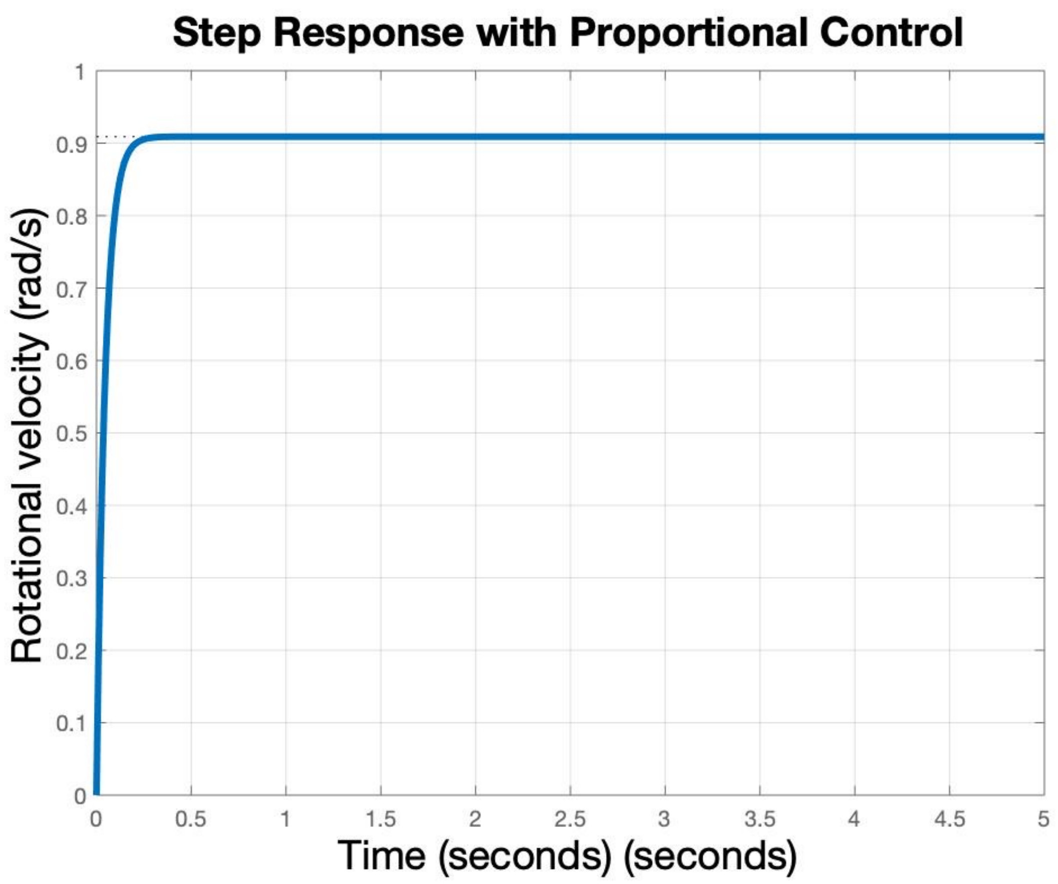
Compute error signal $\mathbf{e} = \mathbf{r} - \mathbf{y}$

Derivative term $K_d\dot{\mathbf{e}}$:

- K_d derivative gain;
- Feedback proportional to how fast the error is increasing/decreasing

Purpose:

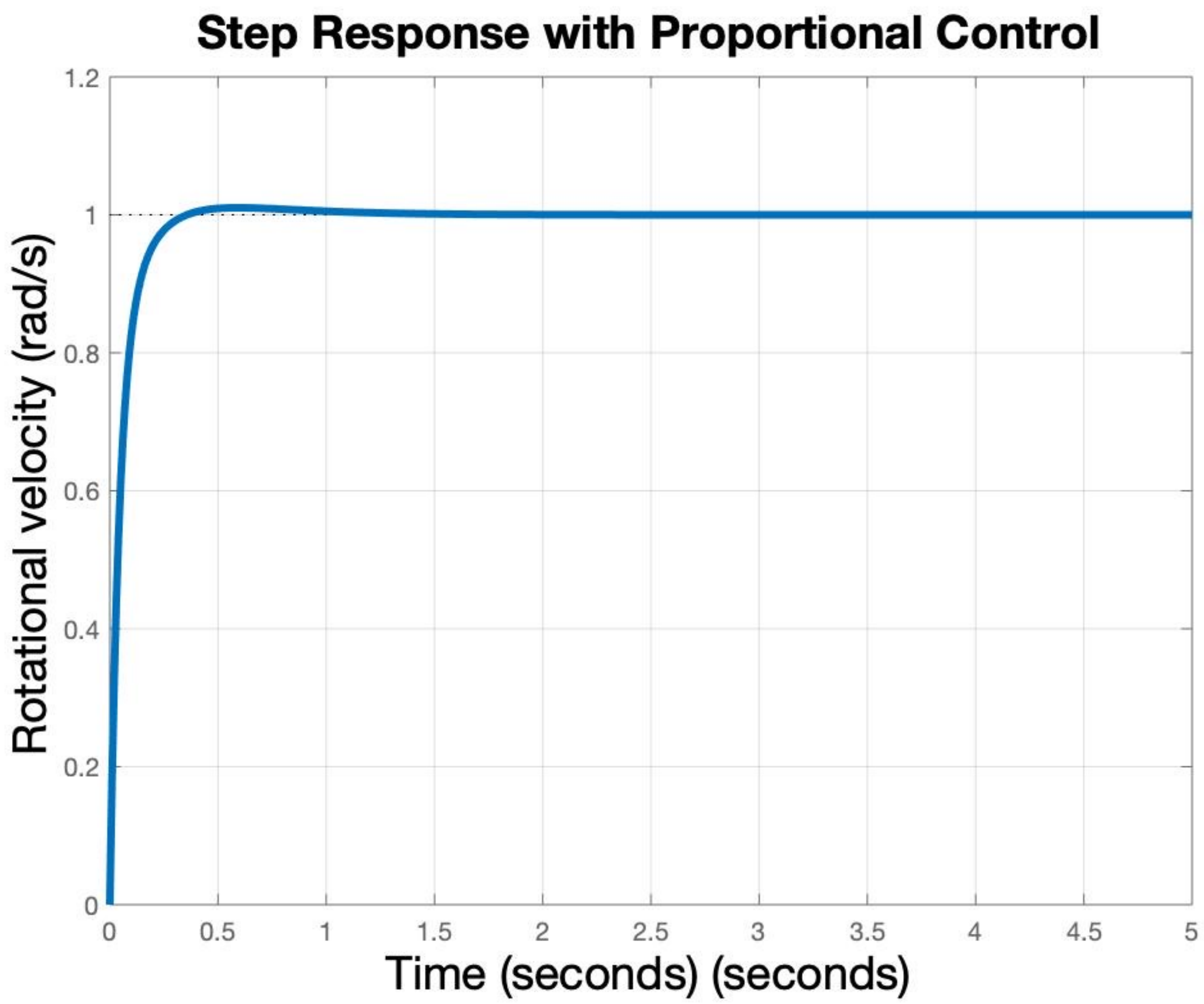
- “Predictive” term, can reduce overshoot: if error is decreasing slowly, feedback is slower
- Can improve tolerance to disturbances



Disadvantages:

- Still cannot eliminate steady-state error
- High frequency disturbances can get amplified

PID-controller



PID controller in practice

May often use only PI or PD control

Many heuristics to *tune* PID controllers, i.e., find values of K_P, K_I, K_D

Several *recipes* to tune, usually rely on designer expertise

E.g. *Ziegler-Nichols* method: increase K_P till system starts oscillating with period T (say till $K_P = K^*$), then set $K_P = 0.6K^*, K_I = \frac{1.2K^*}{T}, K_D = \frac{3}{40}K^*T$

Matlab/Simulink has PID controller blocks + PID auto-tuning capabilities

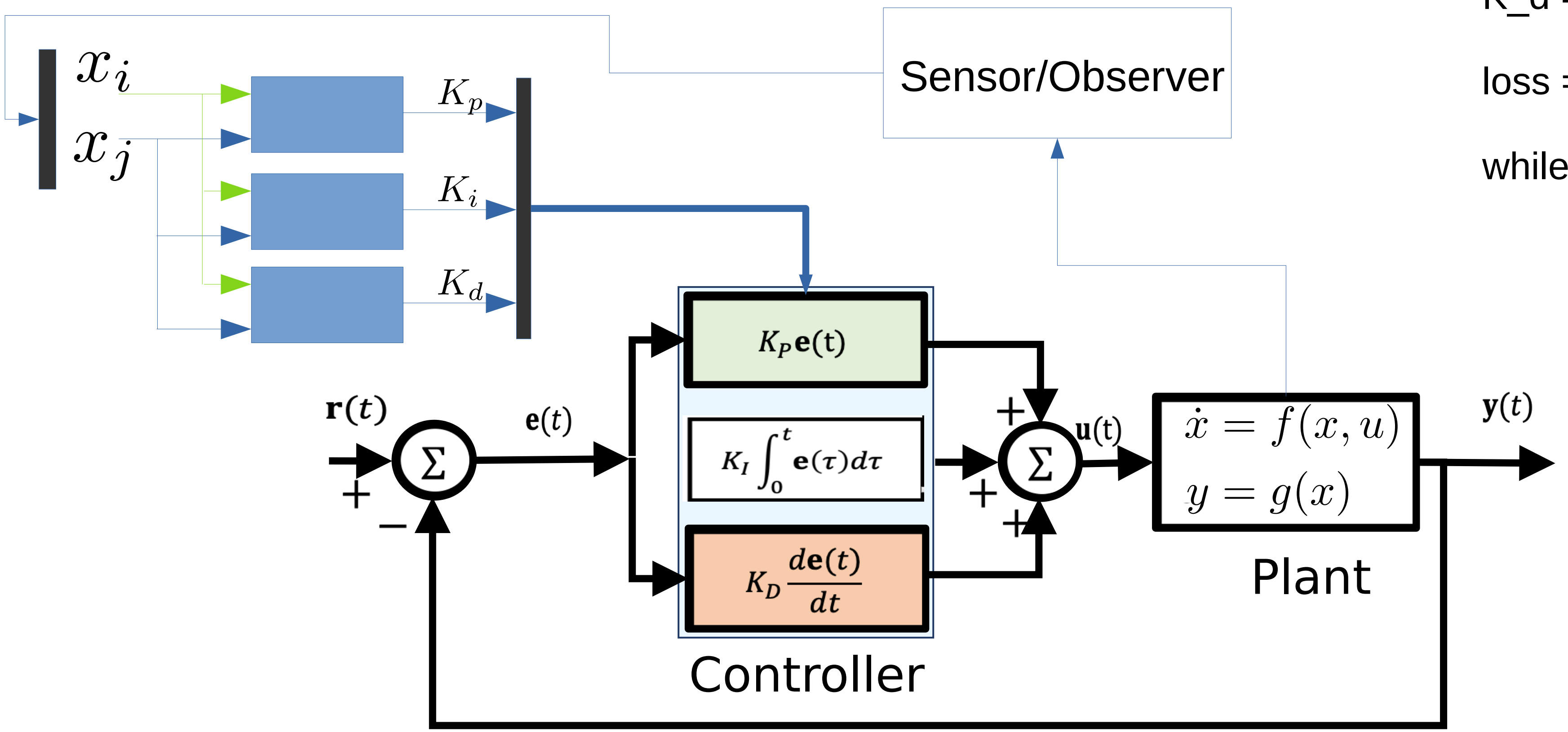
Work well with linear systems or for small perturbations,

For non-linear systems use "gain-scheduling"

- (i.e. using different K_P, K_I, K_D gains in different operating regimes)

Gain Scheduling Example

Used for NONLINEAR / unknown systems



Calibration Routine Example

K_p = f_p (state, param_set)
K_i = f_i (state, param_set)
K_d = f_d (state, param_set)

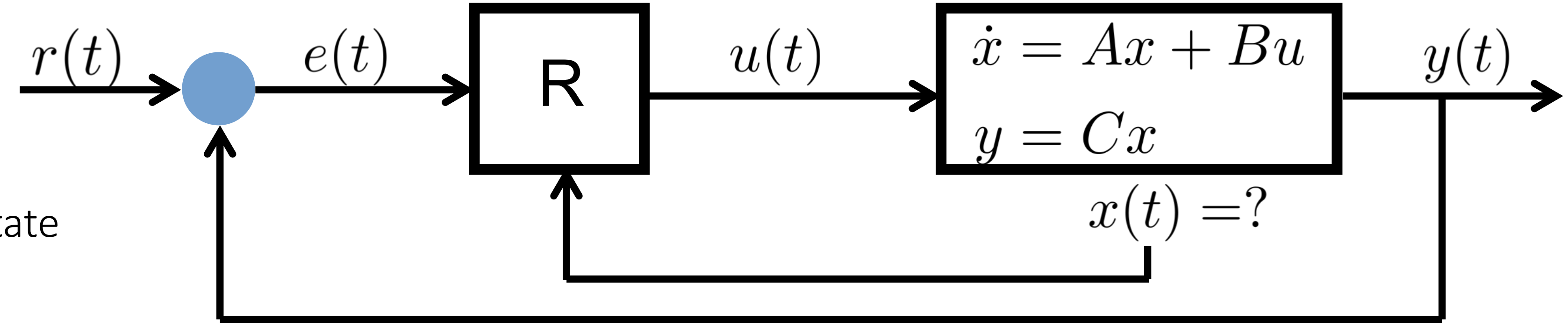
loss = g(stability, risetime, overshoot, etc.)

while not (end condition):

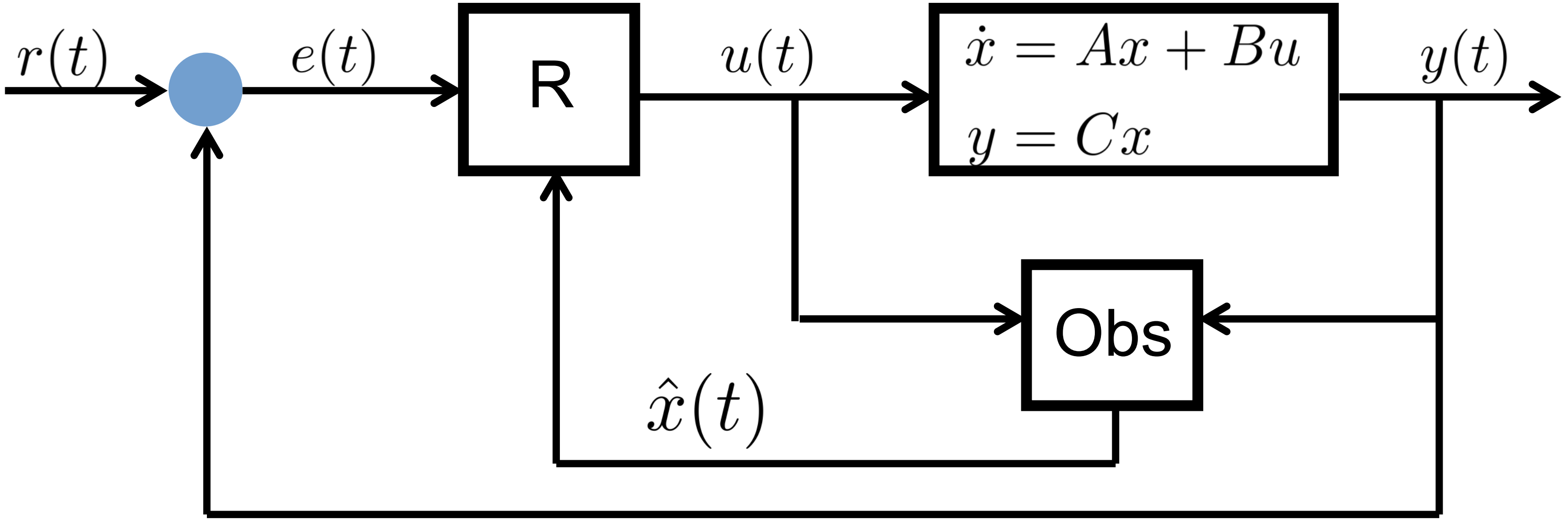
    loss = run_system (param_set)
    optimization_step(param_set)

Observation

- Problem:Control
  - design with (partially) unknown state



- Solution:
  - Luenberger Observer



Luenberger Observer

- State-space representation
- $$\dot{x} = Ax + Bu$$
  $$y = Cx$$

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y})$$
$$\hat{y} = C\hat{x}$$
$$u = K(x_{ref} - \hat{x})$$

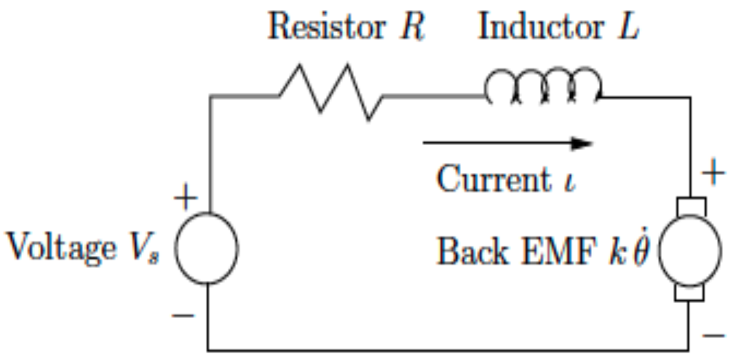
Control design parameters

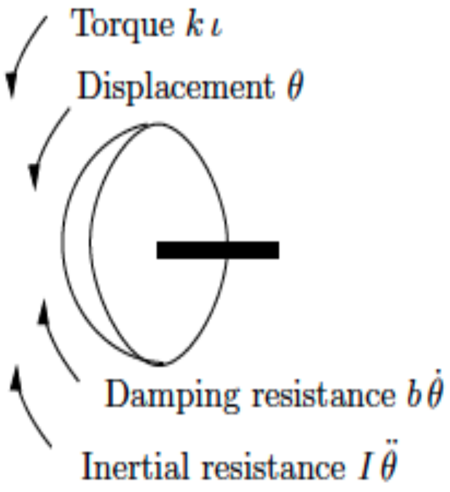
- Observer Error satisfies: $\dot{e} = (A - LC)e$
- Required: Observability, Controllability
- Pole Placement

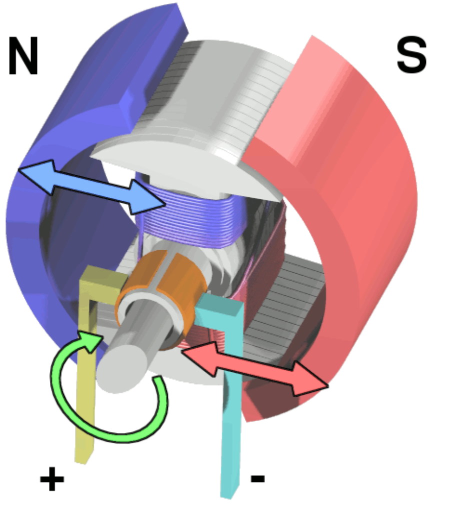
$$K : eig(A - BK) = \{\lambda_{c1}, \ldots, \lambda_{cn}\}$$
$$L : eig(A^T - LC) = \{\lambda_{o1}, \ldots, \lambda_{on}\}$$

Overall system is stable iff both observer and controller are stable

Example - DC Motor







b = 0.1 # friction coefficient (Nm/(rad/sec))
I = 0.01 # mechanical inertia (Kg*m^2)
k = 0.01 # motor torque constant (Nm/A)
R = 1   # armature resistance (Ohm)
L = 0.5 # armature inductance (H)

$$V_s = Ri + L\frac{di(t)}{dt} + k\theta_v$$
$$I\frac{d\theta_v}{dt} + b\theta_v = ki$$

State-space representation
$\dot{x} = Ax + Bu$

$$A = \begin{bmatrix} -b/I & k \\ -k/L & -R \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$x = \begin{bmatrix} \theta_v \\ i \end{bmatrix} \qquad u = V_s$$

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$