Introduction to Control

Karl-Erik Årzén & Anton Cervin
Lund University

Outline

1. Introduction
2. Basic concepts
3. Modeling and design
4. Empirical PID control
5. Digital control

Automatic control

The silent technology:

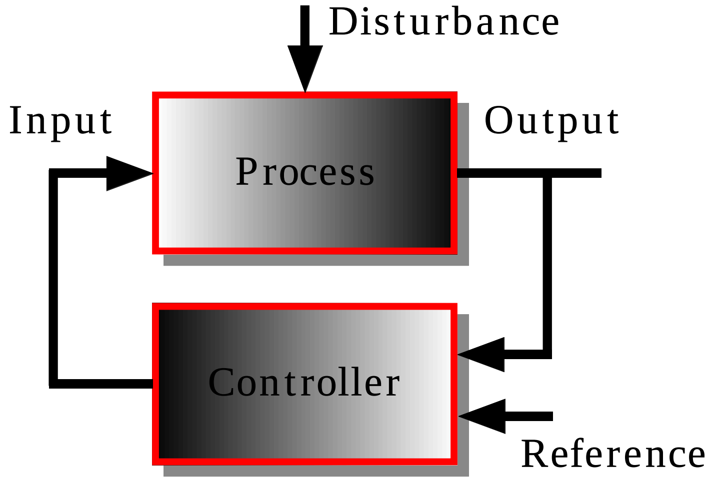
- Widely used
- Very successful
- Seldom talked about, except when disaster strikes!

Automatic control

Use of **models** and **feedback**

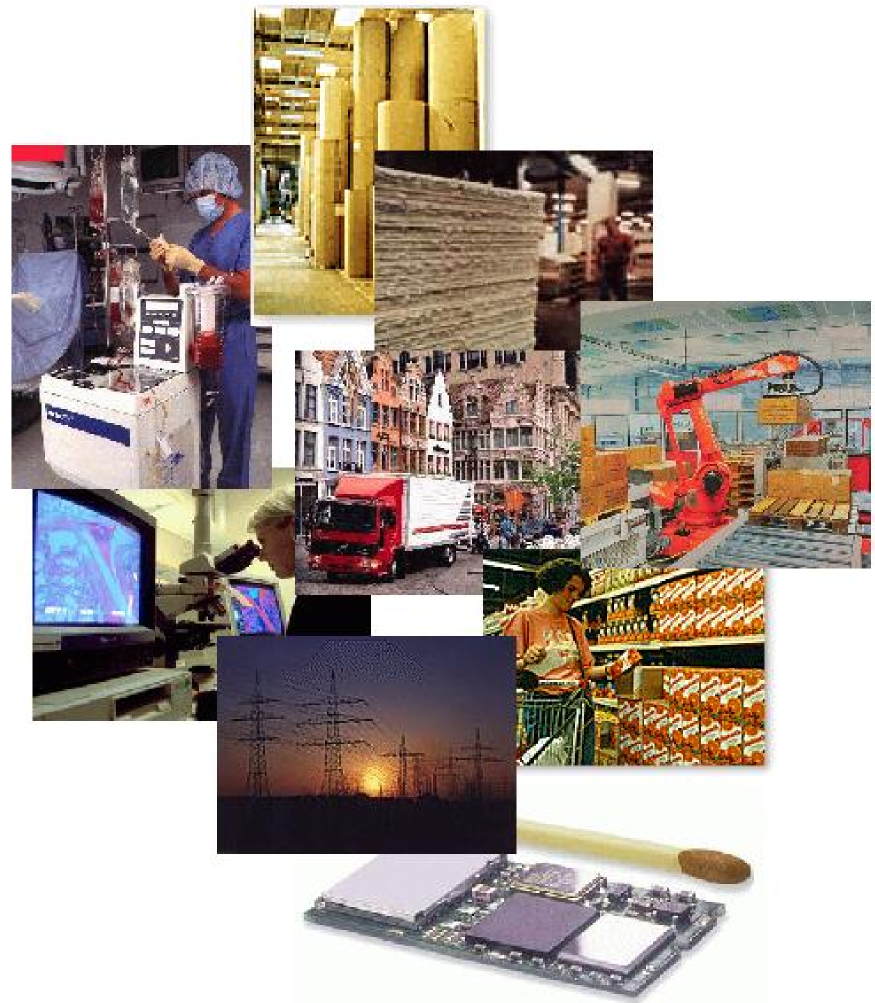
Activities:

- Modeling
- Analysis and simulation
- Control design
- Implementation



Applications

- Automotive systems
- Robotics
- Biotechnology
- Power systems
- Process control
- Communications
- Consumer electronics
- ...

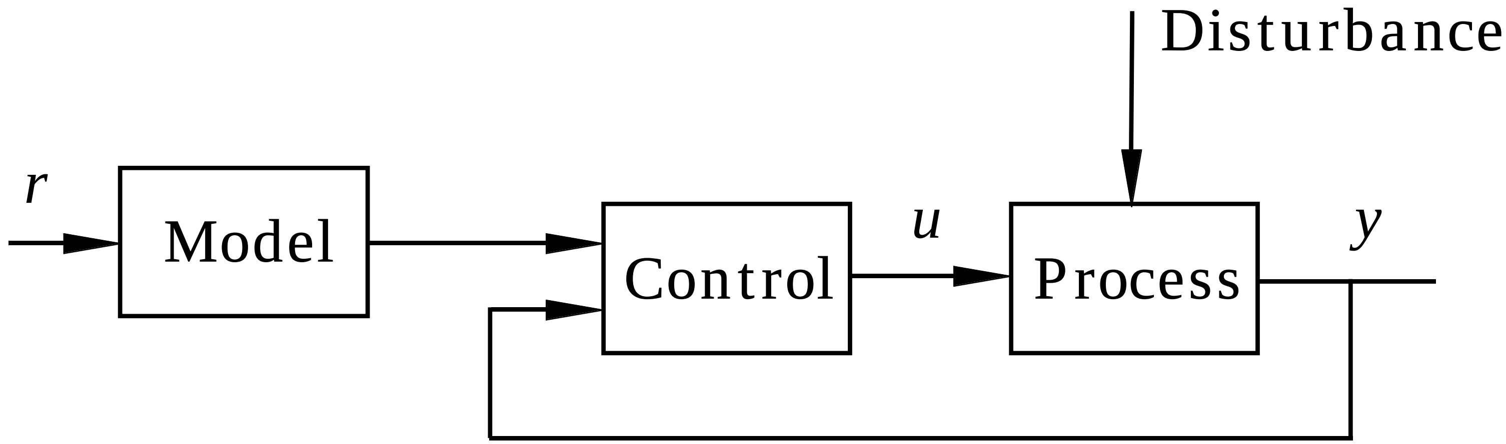


artist

Outline

1. Introduction
2. **Basic concepts**
3. Modeling and design
4. Empirical PID control
5. Digital control

Basic setting



Must handle two tasks:

- Follow reference signals, r
- Compensate for disturbances

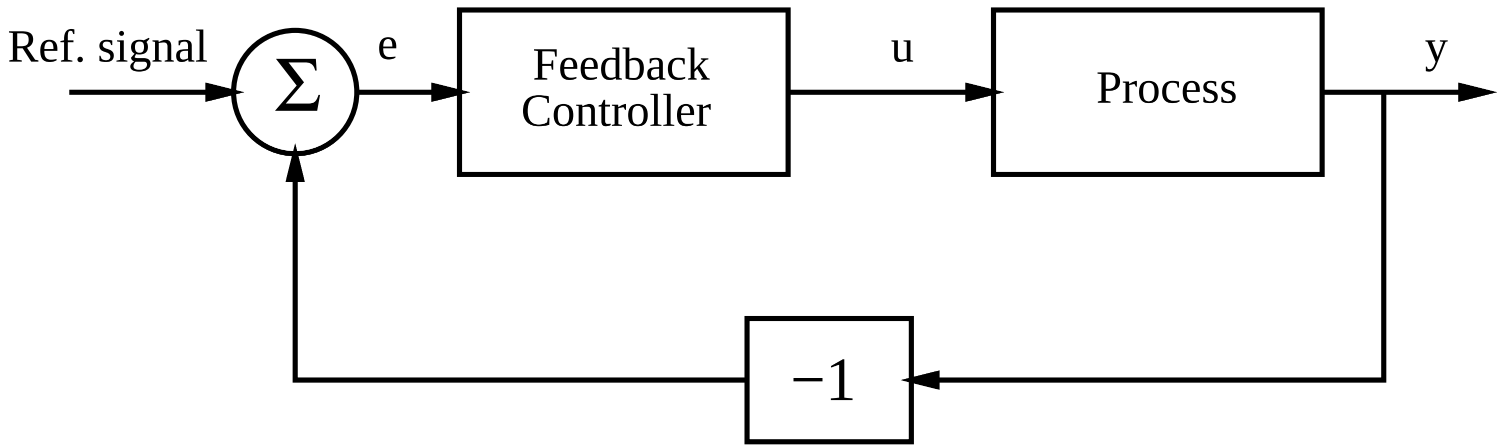
How to

- do several things with the control signal u

The feedback principle

A very powerful idea, that often leads to revolutionary changes in the way systems are designed.

The primary paradigm in automatic control.

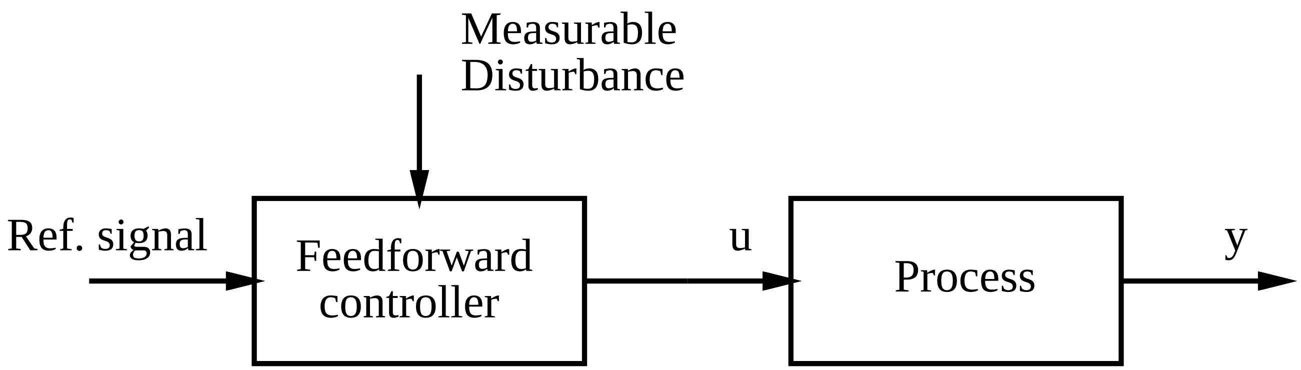


- Base corrective action on an error that has occurred
- Closed loop

Properties of feedback

+ Reduces influence of disturbances
+ Reduces effect of process variations
+ Does not require exact models
− Feeds sensor noise into the system
− May lead to instability, e.g.:
  – if the controller has too high gain
  – if the feedback loop contains too large time delays
    * from the process
    * from the controller implementation

The feedforward principle



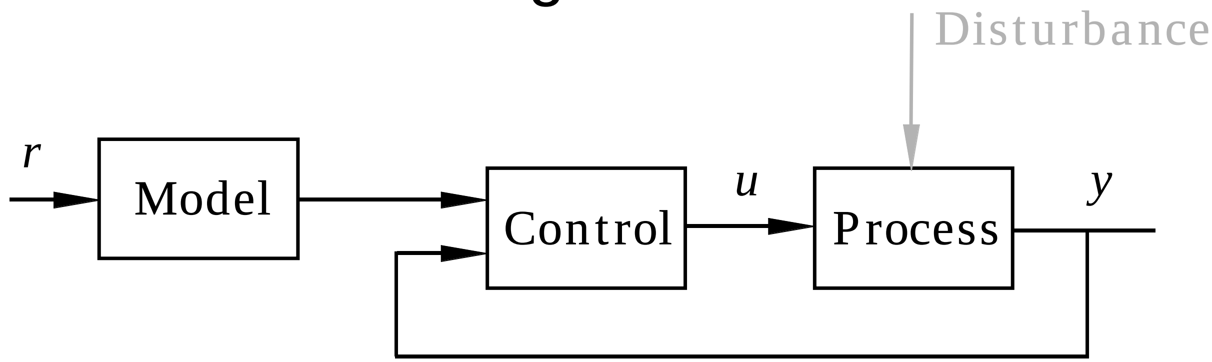
- Take corrective action before an error has occurred
- Measure the disturbance and compensate for it
- Use the fact that the reference signal is known and adjust the control signal to the reference signal
- Open loop

Properties of feedforward

- $+$ Reduces effect of disturbances that cannot be reduced by feedback
- $+$ Measurable signals that are related to disturbances
- $+$ Allows faster set-point changes, without introducing control errors
- $-$ Requires good models
- $-$ Requires stable systems

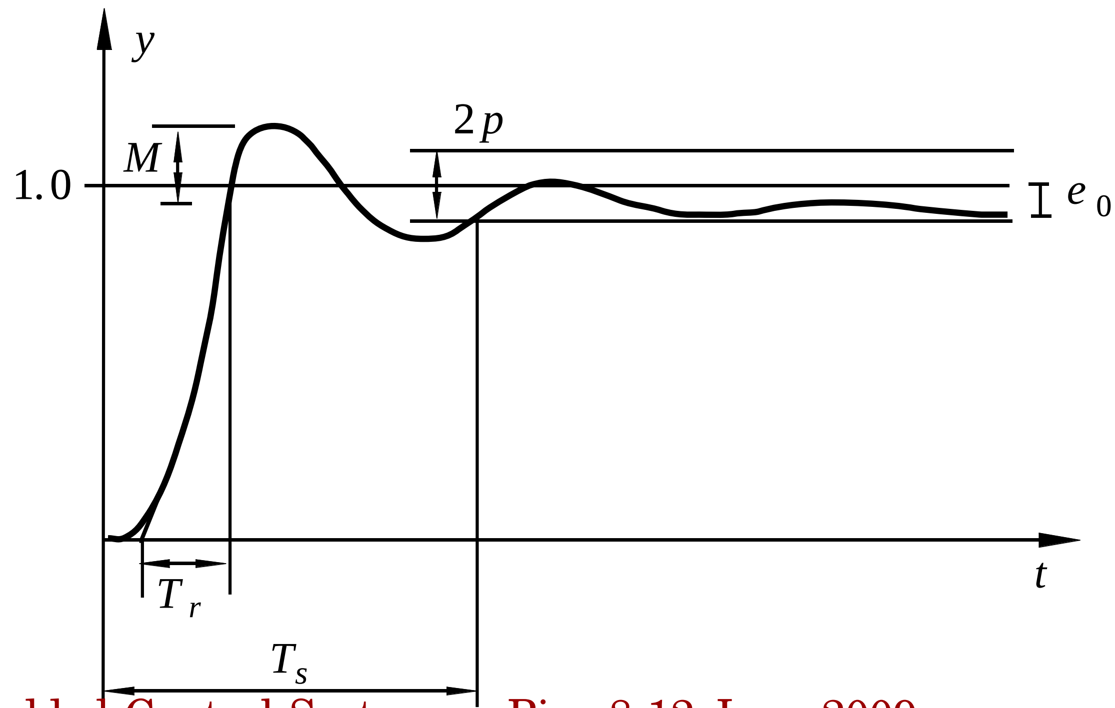
The servo problem

Focus on reference value changes:



Typical design criteria:

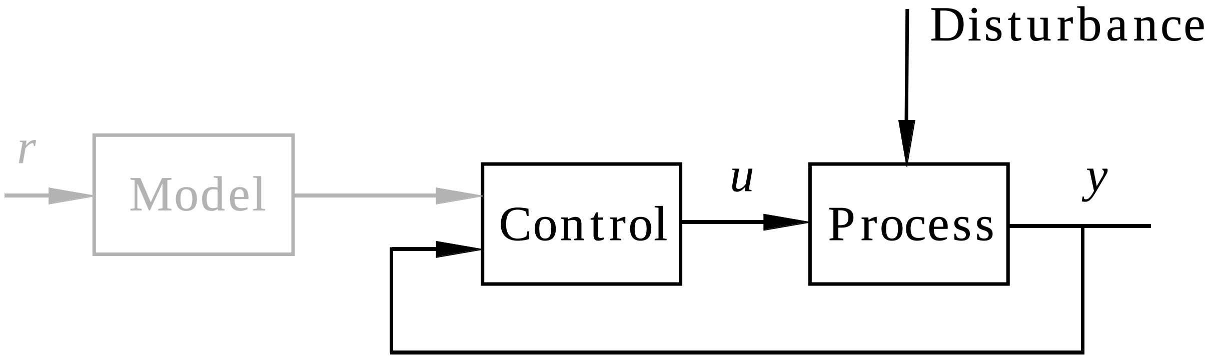
- Rise time, T_r
- Overshoot, M
- Settling time, T_s
- Steady-state error, e_0
- ...



artist

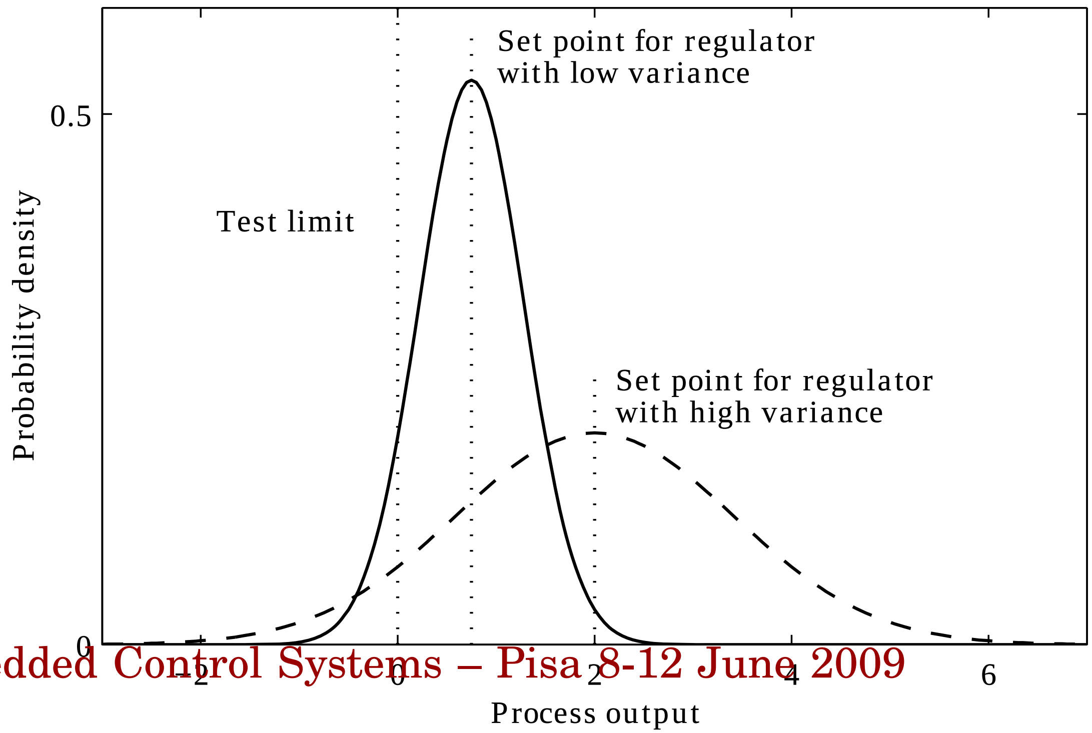
The regulator problem

Focus on process disturbances:

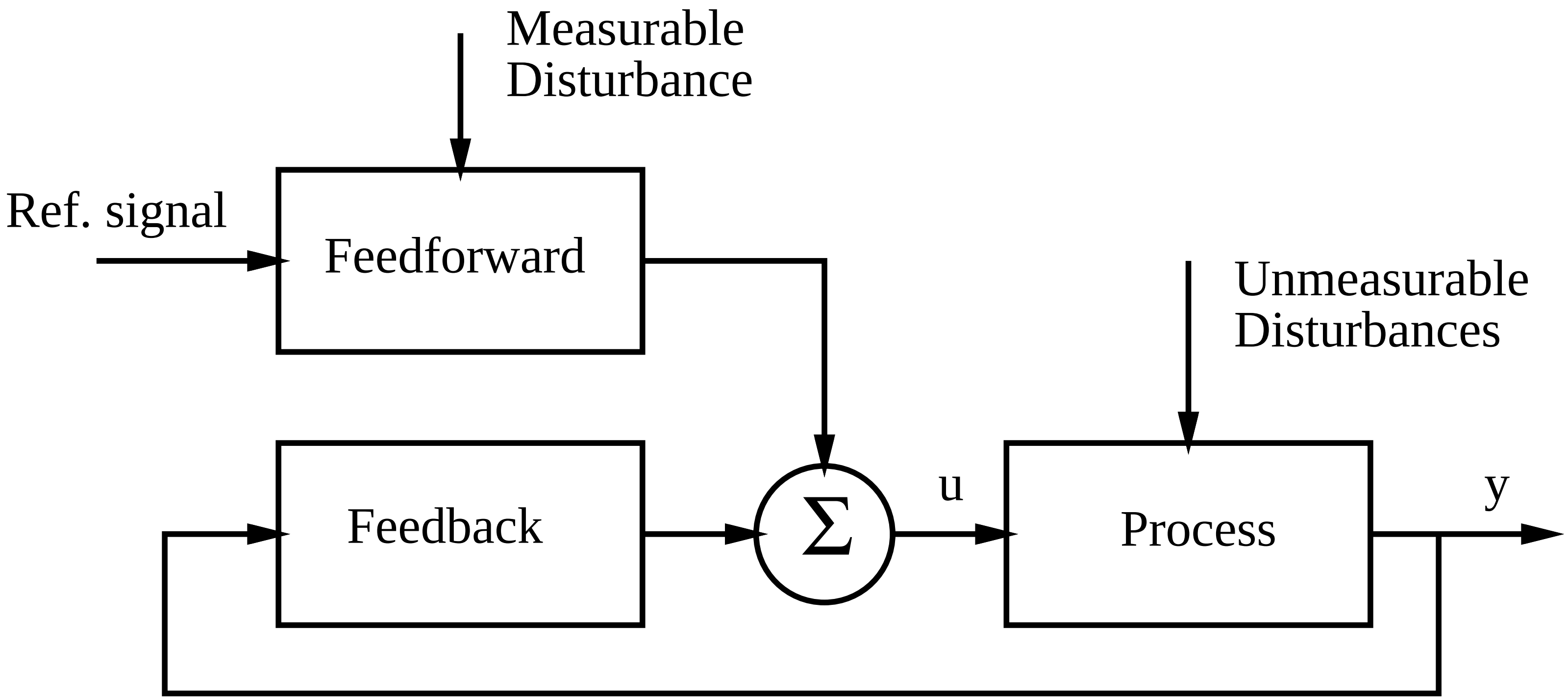


Typical design criteria:

- Output variance
- Control signal variance



Putting it all together

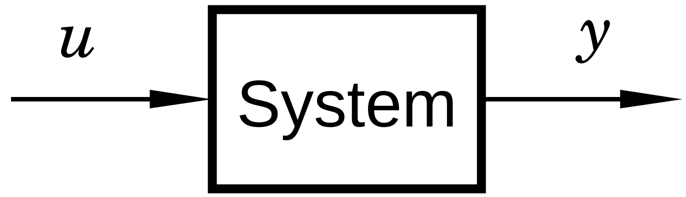


Combination of **feedback** and **feedforward**

Outline

1. Introduction
2. Basic concepts
3. **Modeling and design**
4. Empirical PID control
5. Digital control

Dynamical systems



Static system:

$$y(t) = f\big(u(t)\big)$$

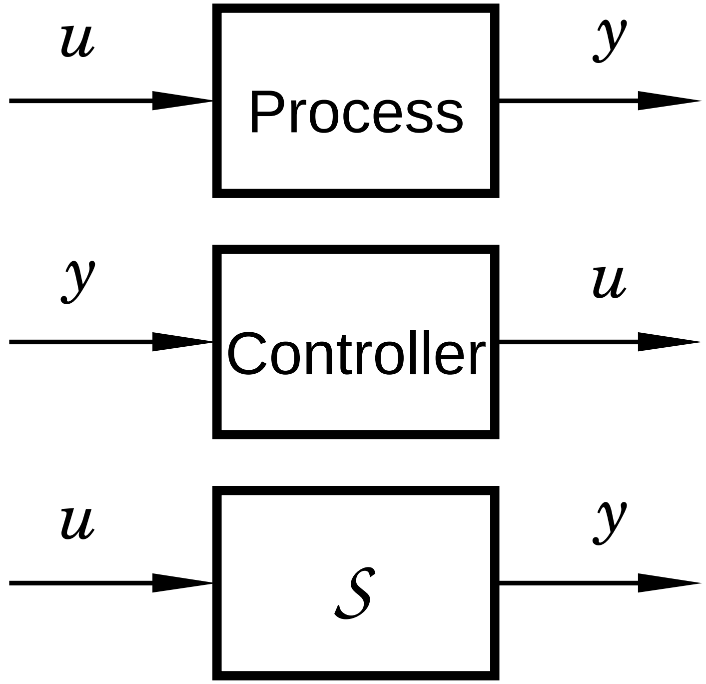
(The output at time t only depends on the input at time t.)

Dynamical system:

$$y(t) = f\big(x(0),\, u_{[0,t]}\big)$$

(The output at time t depends on the initial state $x(0)$ and the input from time 0 to t.)

Modeling

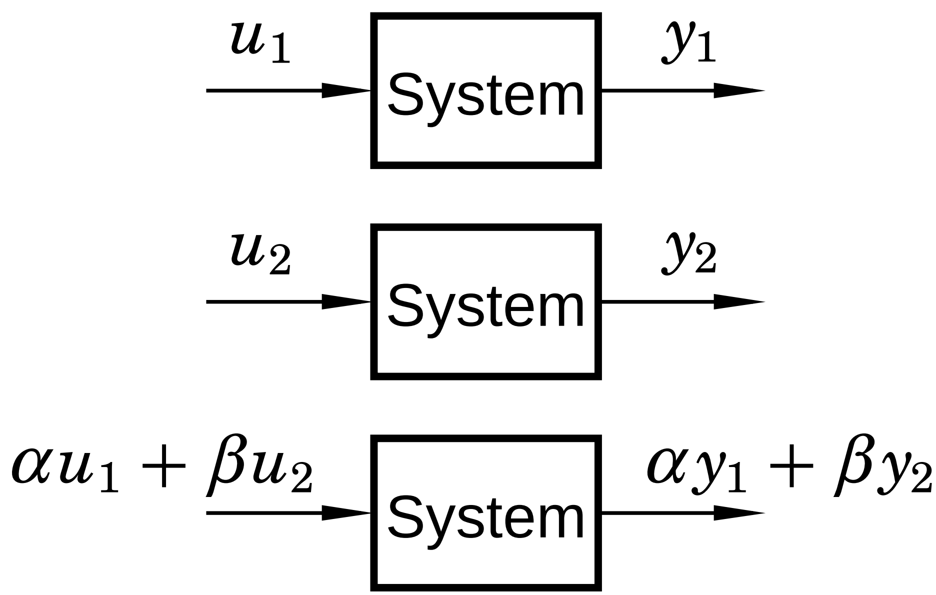


- View all subsystems as "boxes" with inputs and outputs
- Linear, time-invariant (LTI) dynamical systems
- Continuous or discrete time

Linear systems

We will mainly deal with linear, time-invariant (LTI) systems

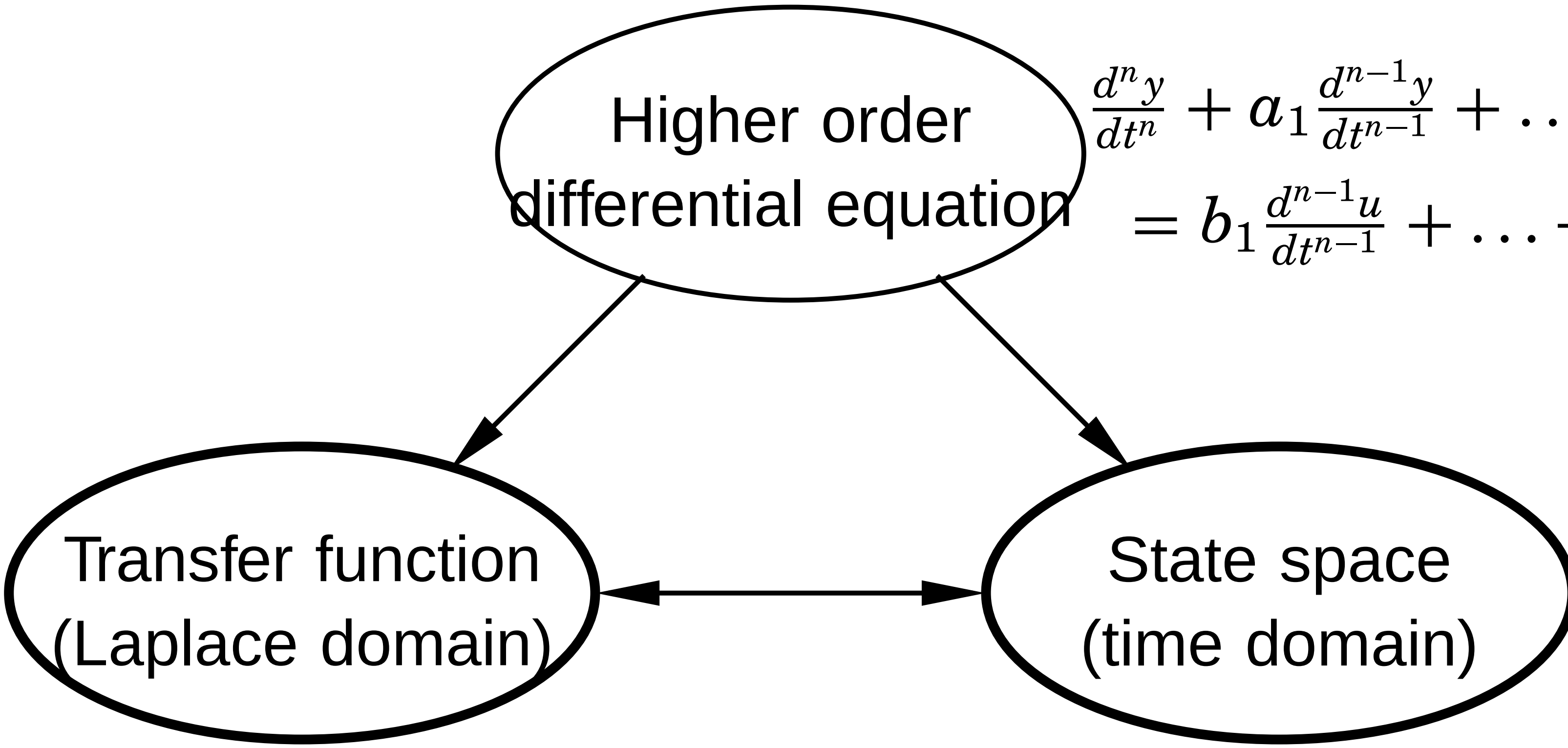
For linear systems, the principle of superposition holds:



Nonlinear systems

- Almost all real systems are nonlinear
  - limited input and output signals
  - nonlinear process geometry
  - friction, turbulence, . . .
- Can be linearized around an operating point
- If there is feedback, a simple linear model is often enough
- But, always remember the limitations of the model!

Continuous-time systems



$$\frac{d^n y}{dt^n} + a_1 \frac{d^{n-1} y}{dt^{n-1}} + \ldots + a_n y = b_1 \frac{d^{n-1} u}{dt^{n-1}} + \ldots + b_n u$$

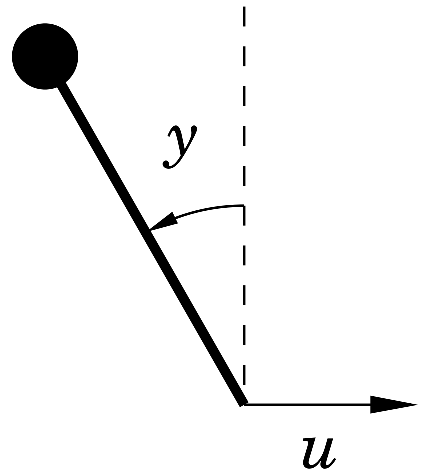
$$Y(s) = G(s)U(s)$$

$$\frac{dx}{dt} = Ax + Bu$$
$$y = Cx + Du$$

Standard system forms

- State space form
  - A number of first-order differential equations
  - Describes what happens "inside" the system and how inputs and output are connected to this
  - Numerically superior
  - The heritage of mechanics
- Transfer function form
  - The transform of a higher-order linear differential equation
  - Describes the relationship between the input and the output
  - The system is a "black box"
  - Compact notation, convenient for hand calculations
  - The heritage of electrical engineering

Example: Inverted pendulum



Nonlinear differential equation from physical modeling:

$$\frac{d^2y}{dt^2} = \omega_0^2 \sin y + ku \cos y$$

Linearized model around $y^0 = 0$ $(\sin y \approx y,\ \cos y \approx 1)$:

$$\frac{d^2y}{dt^2} = \omega_0^2 y + ku$$

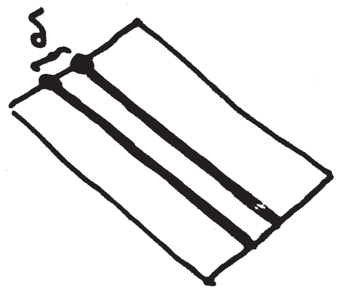
Inverted pendulum in state space form

Introduce state variables

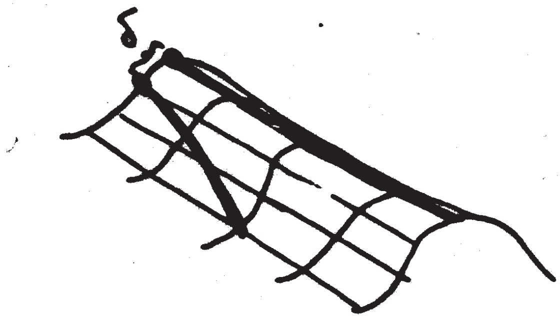
- $x_1 = y$ (pendulum angle)
- $x_2 = \frac{dy}{dt}$ (pendulum angular velocity)

$$
\begin{aligned}
\frac{dx}{dt} &= \begin{pmatrix} 0 & 1 \\ \omega_0^2 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ k \end{pmatrix} u \\
y &= \begin{pmatrix} 1 & 0 \end{pmatrix} x
\end{aligned}
$$

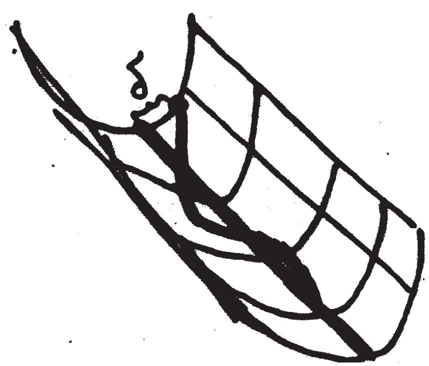
Stability concepts



Stable



Unstable



Asymptotically stable

Stability definitions

Assume

$$\dot{x} = Ax, \quad x(0) = x_0$$

The system is **stable** if $x(t)$ is limited for all x_0.

The system is **asymptotically stable** if $x(t) \to 0$ for all x_0.

The system is **unstable** if $x(t)$ is unlimited for some x_0.

Stability criteria

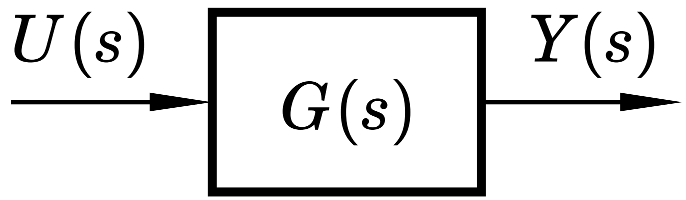
$$\begin{cases} \dot{x} = Ax \\ x(0) = x_0 \end{cases} \quad \Rightarrow \quad x(t) = x_0 e^{At}$$

The behavior of the solution depends on the eigenvalues of A

All eigenvalues have negative real part:	$\Leftrightarrow$	As. stab.
Some eigenvalue has positive real part:	$\Rightarrow$	Unstable
No eigenvalues with positive real part and no multiple eigenvalues on the imaginary axis:	$\Leftrightarrow$	Stable

Transfer function form

Study the system in the (complex) frequency domain:



$U(s)$ – Laplace transform of $u(t)$

$Y(s)$ – Laplace transform of $y(t)$

$G(s)$ – transfer function

$$Y(s) = G(s)U(s)$$

(if the initial state is assumed to be zero)

Some signals and their Laplace transforms

Definition: $$\mathcal{L}f = F(s) = \int_0^\infty e^{-st} f(t)dt$$

Derivative: $$\mathcal{L}\left(\frac{df}{dt}\right) = sF(s)$$

Integral: $$\mathcal{L}\left(\int f dt\right) = \frac{1}{s}F(s)$$

Dirac impulse: $$\mathcal{L}\delta = 1$$

Step function: $$\mathcal{L}\theta = \frac{1}{s}$$

Ramp function: $$\mathcal{L}(t\theta) = \frac{1}{s^2}$$

Exponential function: $$\mathcal{L}(e^{at}\theta) = \frac{1}{s-a}$$

From transfer function to state space form

$$\begin{cases} \dot{x} = Ax + Bu & \qquad x(0) = 0 \\ y = Cx + Du \end{cases}$$

$$\begin{cases} sX(s) = AX(s) + BU(s) \\ Y(s) = CX(s) + DU(s) \end{cases}$$

$$Y(s) = \left[C(sI - A)^{-1}B + D\right] U(s)$$

$$G(s) = C(sI - A)^{-1}B + D = \frac{p(s)}{q(s)}$$

$q(s) = \det(sI - A)$ is called **characteristic polynomial**

Poles and zeros

Often,

$$G(s) = \frac{p(s)}{q(s)}$$

The roots of $p(s)$ are called zeros

The roots of $q(s)$ are called poles

Note that

$$\text{Poles of } G(s) \quad \Leftrightarrow \quad \text{Eigenvalues of } A$$

Inverted pendulum in transfer function form

Apply Laplace transform to differential equation:

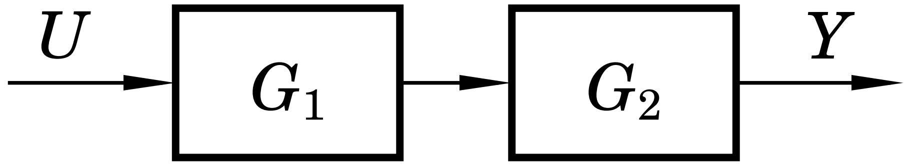
$$s^2 Y(s) = \omega_0^2 Y(s) + kU(s)$$

$$G(s) = \frac{Y(s)}{U(s)} = \frac{k}{s^2 - \omega_0^2}$$

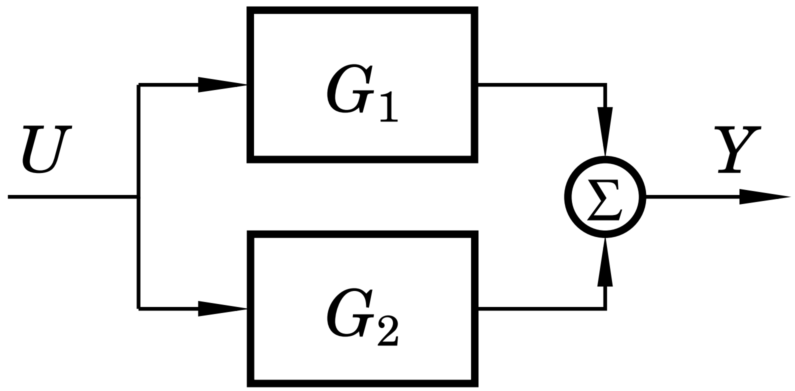
Or, from state space to transfer function:

$$G(s) = C(sI - A)^{-1} B = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} s & -1 \\ -\omega_0^2 & s \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ k \end{pmatrix} = \frac{k}{s^2 - \omega_0^2}$$

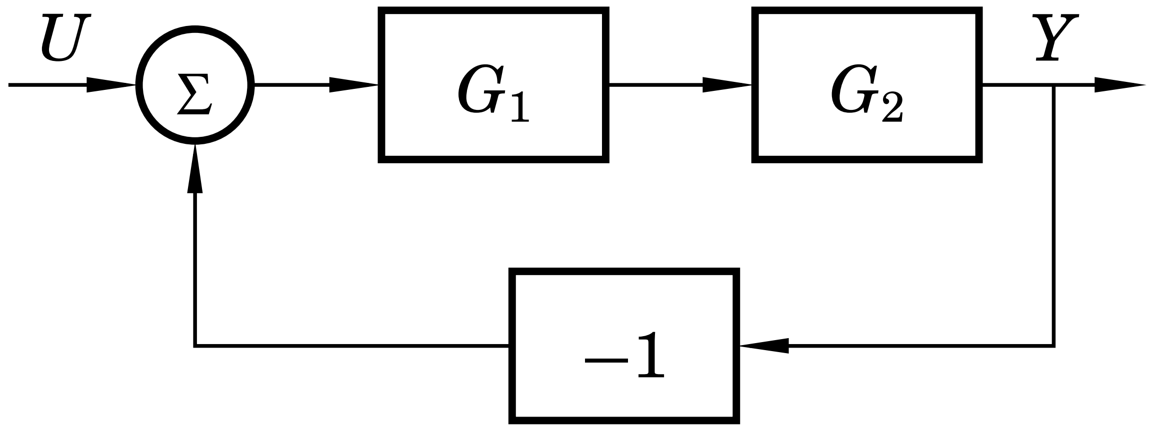
Block diagrams



$$Y = G_2 G_1\, U$$

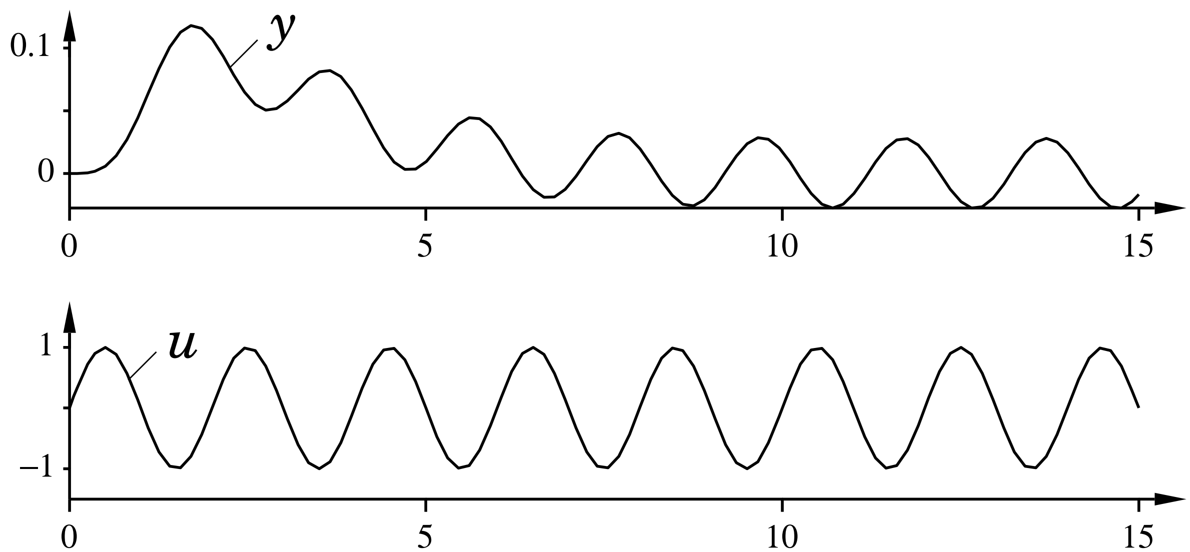


$$Y = \big(G_1 + G_2\big)\, U$$



$$
\begin{aligned}
Y &= G_2 G_1 \big(U - Y\big) \\
Y\big(1 + G_2 G_1\big) &= G_2 G_1 U \\
Y &= \frac{G_2 G_1}{1 + G_2 G_1} U
\end{aligned}
$$

Frequency response



Given a stable system $G(s)$, the input $u(t) = \sin \omega t$ will, after a transient, give the output

$$y(t) = |G(i\omega)| \sin \Big(\omega t + \arg G(i\omega) \Big)$$

The steady-state output is also sinusoidal

Bode diagram

Draw

- $|G(i\omega)|$ as a function of ω (in log-log scale)
  - Amplitude/magnitude/gain diagram
- $\arg G(i\omega)$ as a function of ω (in log-lin scale)
  - Phase/angle diagram

Example: low-pass filter

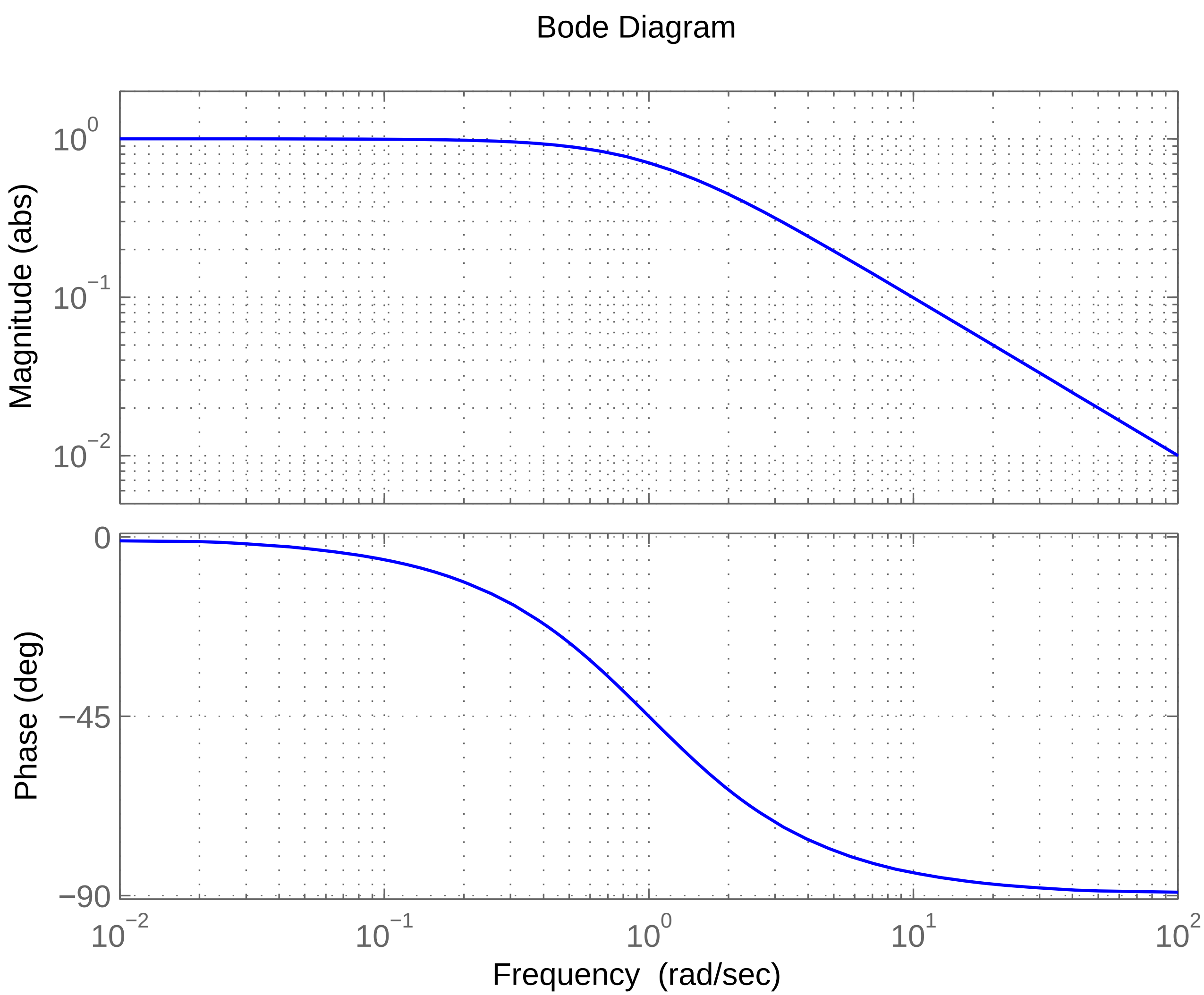
$$\frac{dy(t)}{dt} + y(t) = u(t) \quad \Leftrightarrow \quad G(s) = \frac{1}{s+1}$$

$$G(i\omega) = \frac{1}{i\omega + 1}$$

$$|G(i\omega)| = \frac{1}{\sqrt{\omega^2 + 1}}$$

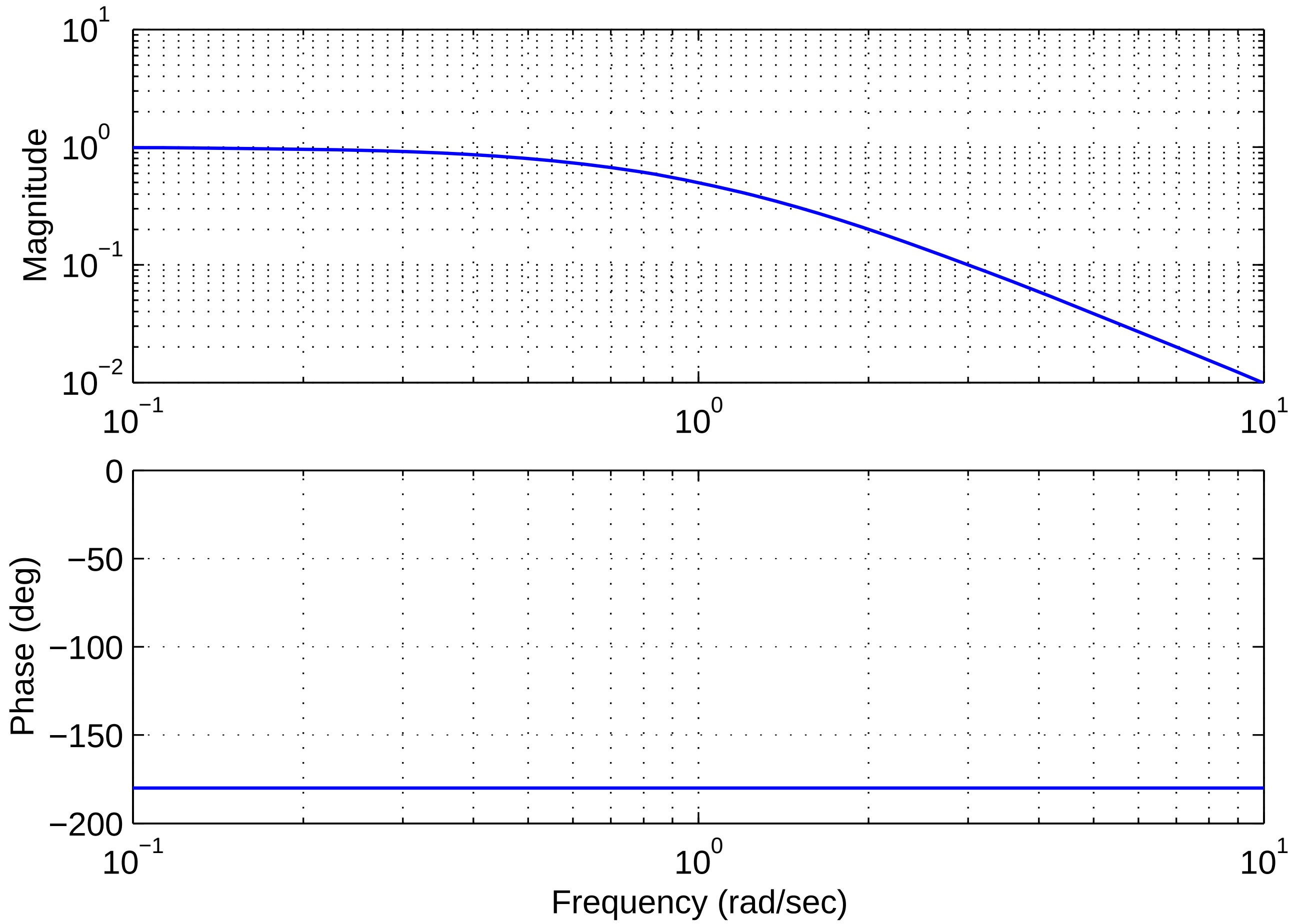
$$\arg G(i\omega) = -\arctan \omega$$

Example: low-pass filter

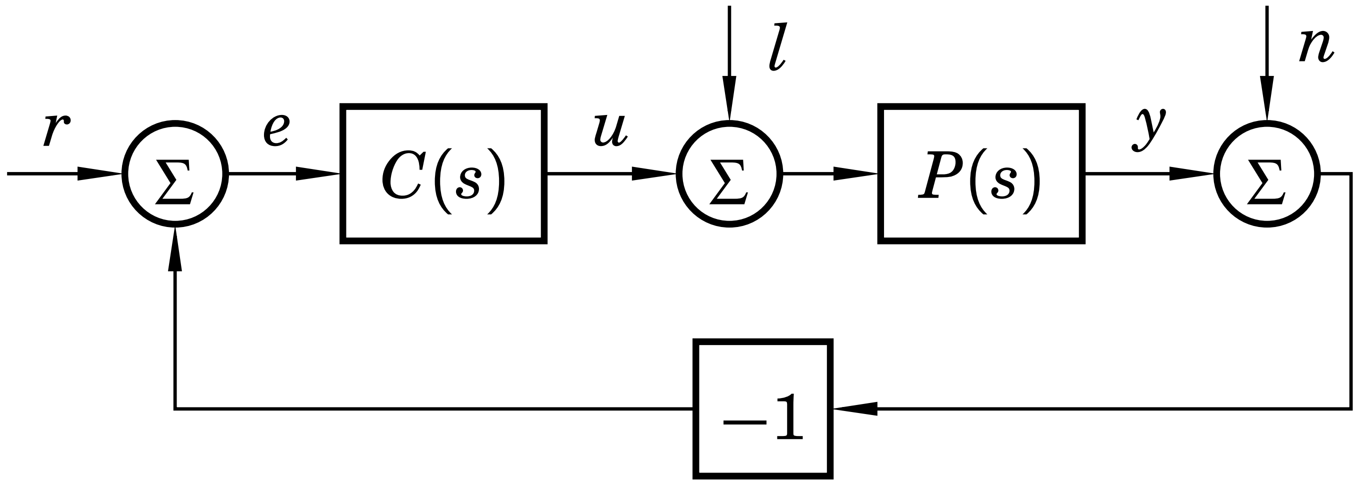


Frequency response for inverted pendulum

Plot $|G(i\omega)|$ and $\arg G(i\omega)$ for $\omega \in [0, \infty]$



Model-based design



Given $P(s)$, determine $C(s)$ such that the specifications on the closed-loop system are met. Common approaches:

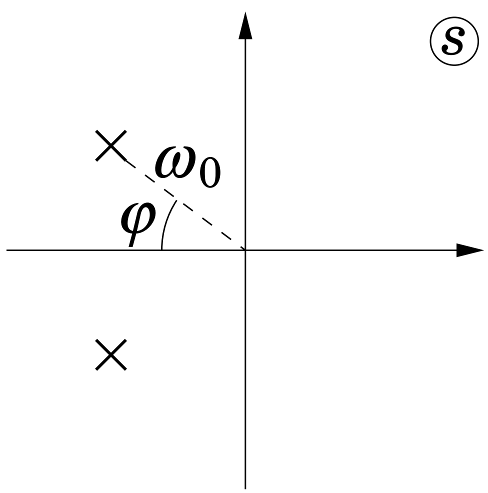
- Frequency domain design (loop shaping)
- Pole placement design
  - transfer function domain
  - state space domain
- Optimization-based methods (H_∞, LQG, ...)

Pole placement – transfer function domain

- Determine the required form of $C(s) = \frac{b_1 s^{n-1} + \ldots + b_n}{s^n + a_1 s^{n-1} + \ldots + a_n}$
- Calculate the closed loop system:

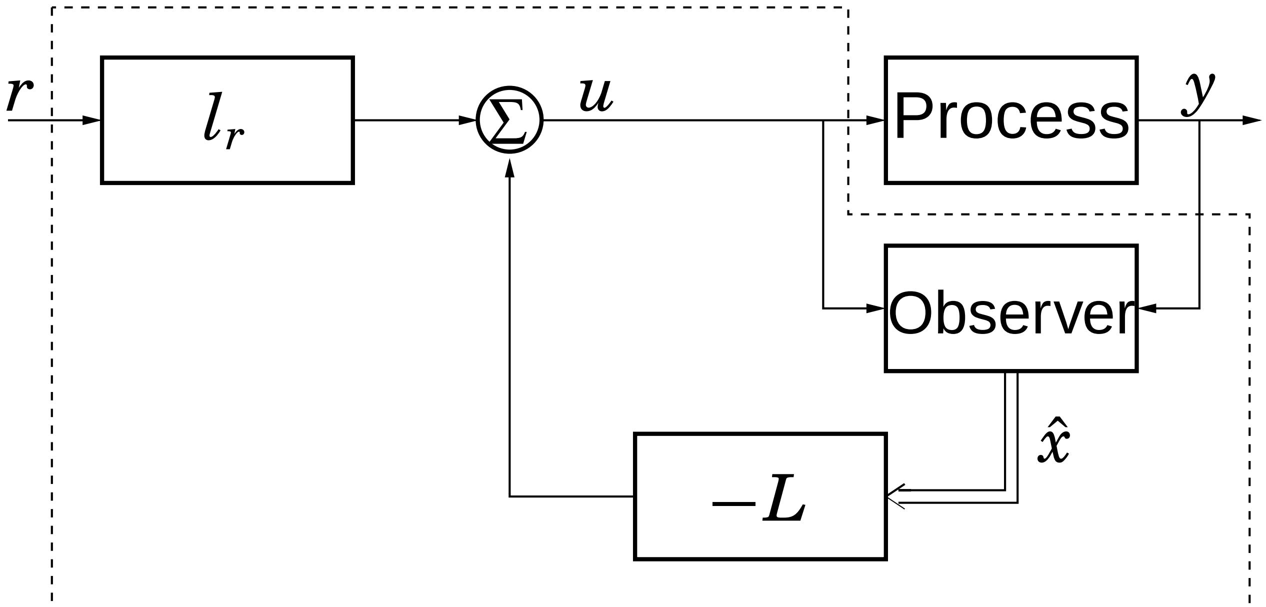
$$G_{cl}(s) = \frac{P(s)C(s)}{1 + P(s)C(s)}$$

- Choose the coefficients of $C(s)$ such that you get the desired closed-loop poles



- Large ω_0 $\Leftrightarrow$ fast system
- Large φ $\Leftrightarrow$ poorly damped system

Pole placement – state space domain



State feedback from an observer:

$$\frac{d\hat{x}}{dt} = A\hat{x} + Bu + K(y - C\hat{x})$$
$$u = -L\hat{x} + l_r r$$

Choose gain vectors L and K to give desired closed-loop poles

Outline

1. Introduction
2. Basic concepts
3. Modeling and design
4. **Empirical PID control**
5. Digital control

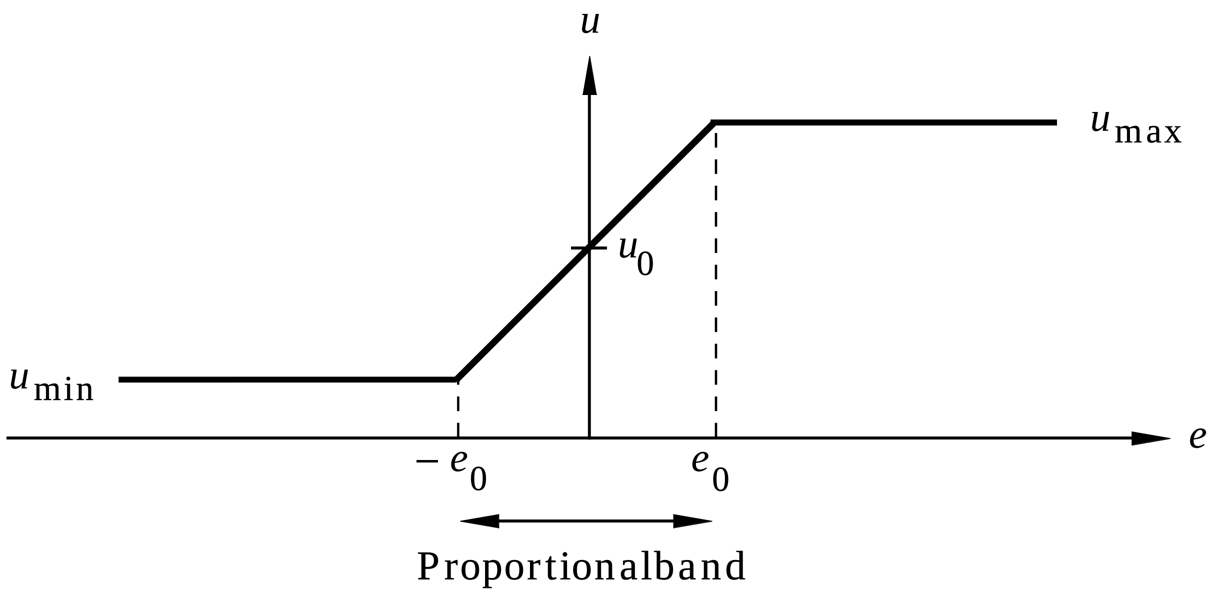
PID control

- The oldest controller type
- The most widely used
  - Pulp & Paper 86%
  - Steel 93%
  - Oil refineries 93%
- Much to learn!!

The textbook algorithm

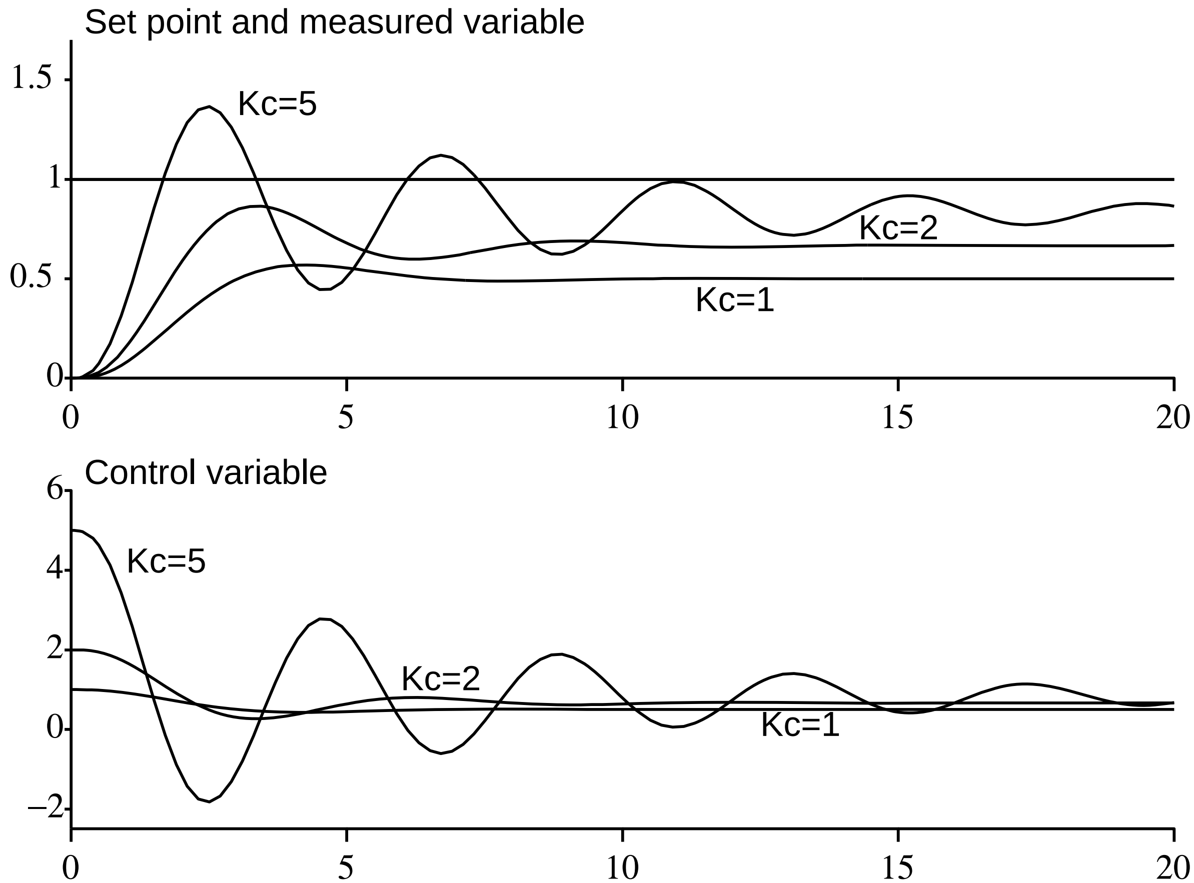
$$
\begin{aligned}
u(t) &= K\left(e(t) + \frac{1}{T_i}\int^{t} e(\tau)d\tau + T_d\frac{de(t)}{dt}\right) \\
U(s) &= K\left(E(s) + \frac{1}{sT_i}E(s) + T_d s E(s)\right) \\
&= P + I + D
\end{aligned}
$$

Proportional term



$$u = \begin{cases} u_{\max} & e > e_0 \\ Ke + u_0 & -e_0 < e < e_0 \\ u_{\min} & e < -e_0 \end{cases}$$

Properties of P-control



- stationary error
- increased K means faster speed, increased noise sensitivity, worse stability

Errors with P-control

Control signal:

$$u = Ke + u_0$$

Error:

$$e = \frac{u - u_0}{K}$$

Error removed if:

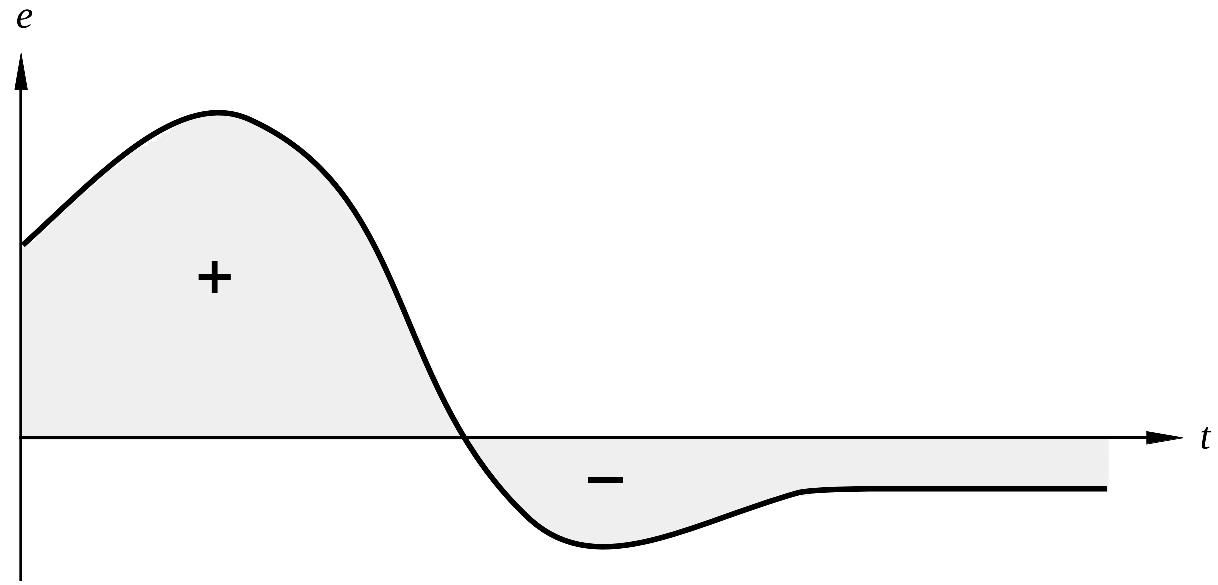
- K equals infinity
- $u_0 = u$

Solution: Automatic way to obtain u_0

Integral term

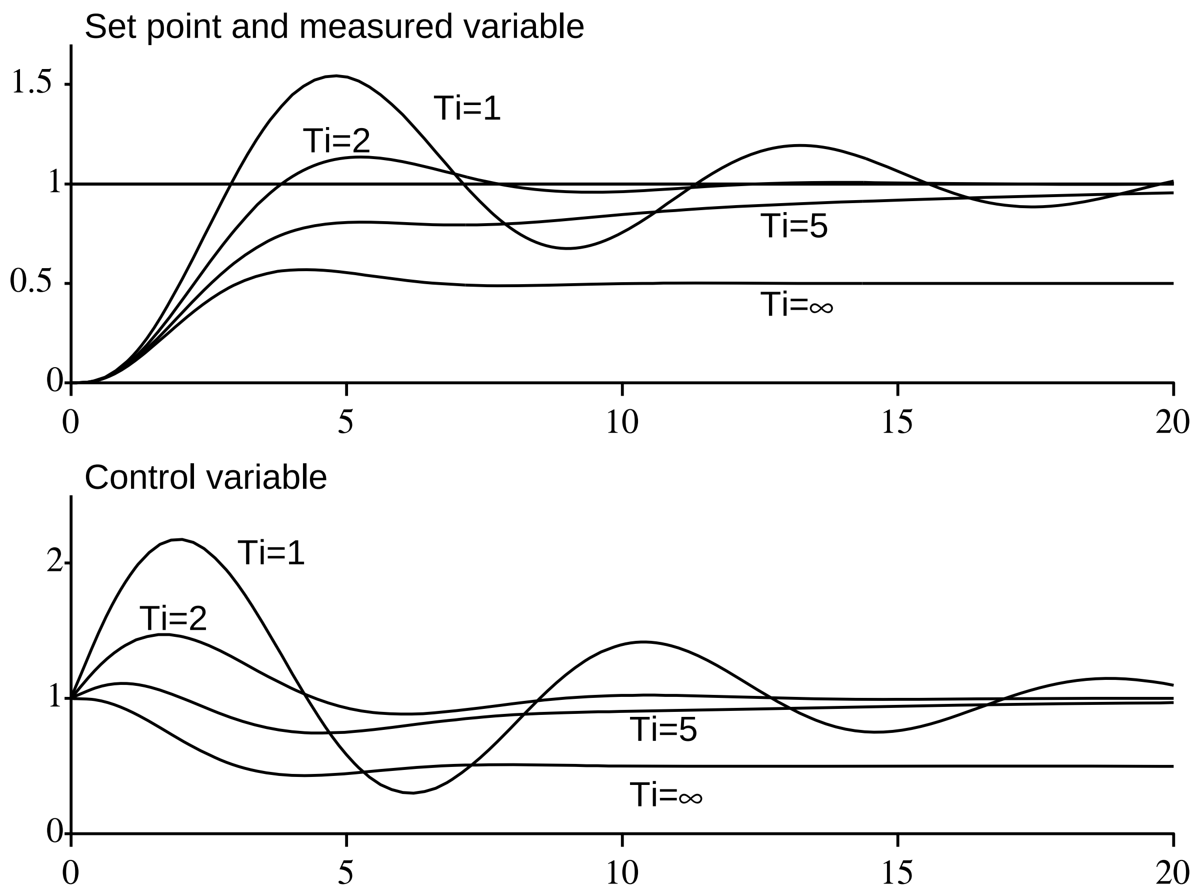
$$u = Ke + u_0$$

$$u = K\left(e + \frac{1}{T_i}\int e(t)dt\right) \quad \text{(PI)}$$



Stationary error present $\rightarrow \int edt$ increases $\rightarrow u$ increases $\rightarrow y$ increases $\rightarrow$ the error is not stationary

Properties of PI-control

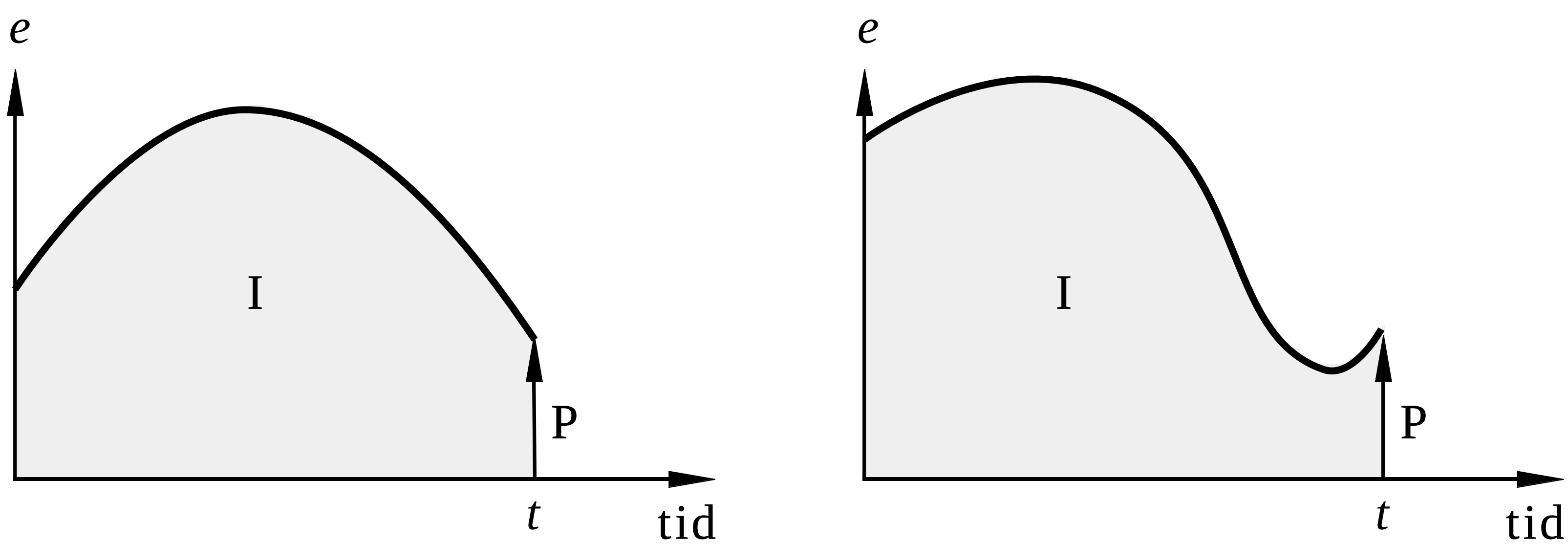


- removes stationary error
- smaller T_i implies worse stability, faster steady-state error removal

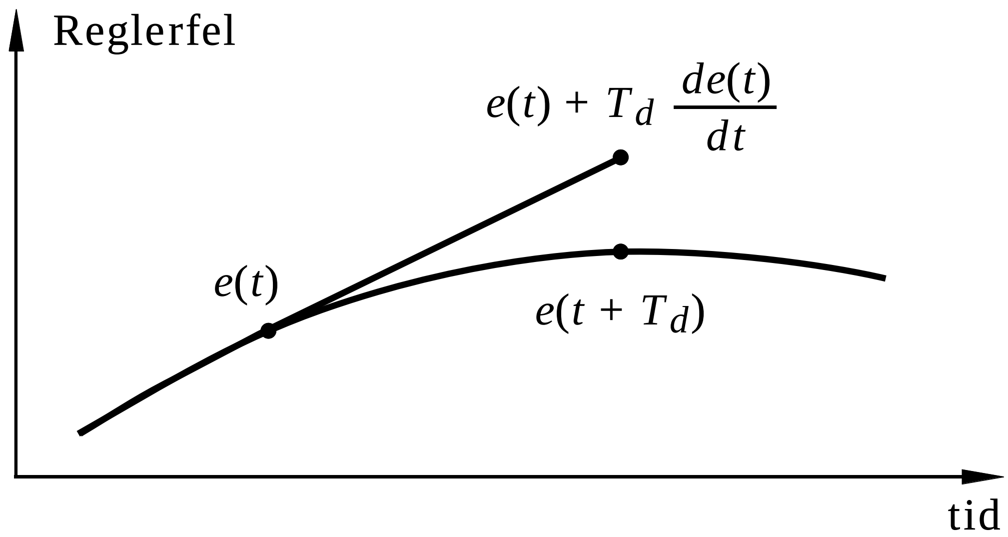
Prediction

A PI-controller contains no prediction

The same control signal is obtained for both these cases:



Derivative part



P:

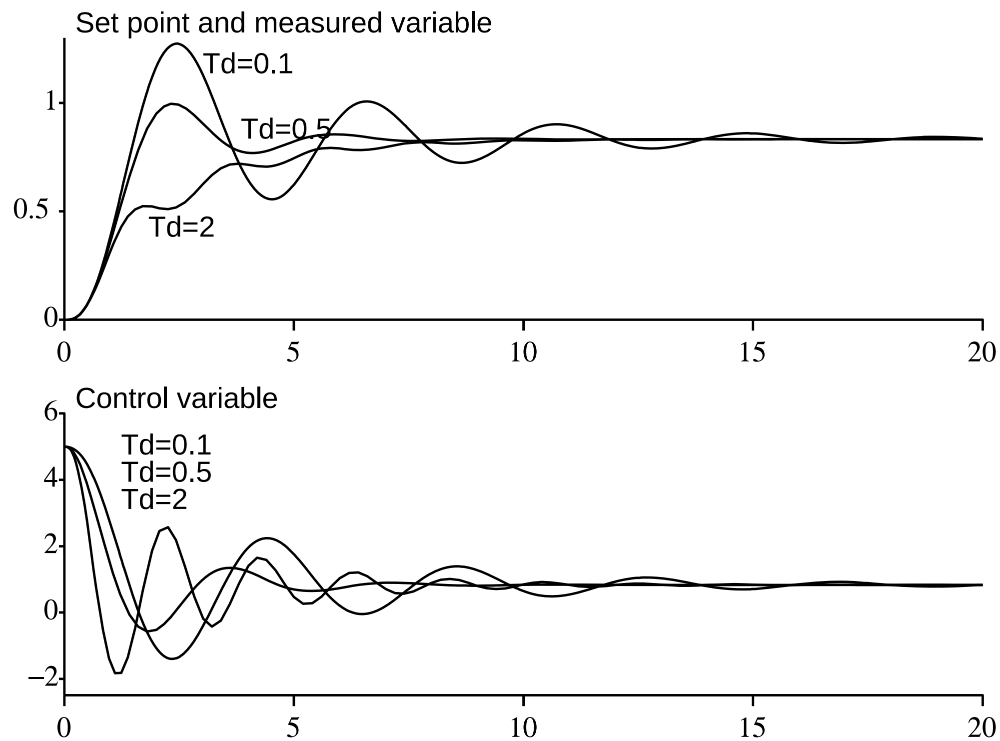
$$u(t) = Ke(t)$$

PD:

$$u(t) = K\left(e(t) + T_d \frac{de(t)}{dt}\right) \approx Ke(t + T_d)$$

T_d = Prediction horizon

Properties of PD-control



- T_d too small, no influence
- T_d too large, decreased performance

In industrial practice the D-term is often turned off.

Algorithm modifications

Modifications are needed to make the controller practically useful

- Limitations of derivative gain
- Derivative weighting
- Handle control signal limitations

Limitations of derivative gain

We do not want to apply derivation to high frequency measurement noise, therefore the following modification is used:

$$sT_d \approx \frac{sT_d}{1 + sT_d/N}$$

$N =$ maximum derivative gain, often $10 - 20$

Derivative weighting

The reference is often constant for long periods of time

Reference often changed in steps $\rightarrow$ D-part becomes very large.

Derivative part applied on part of the reference or only on the measurement signal.

$$D(s) = \frac{sT_d}{1 + sT_d/N}(\gamma R(s) - Y(s))$$

Often, $\gamma = 0$

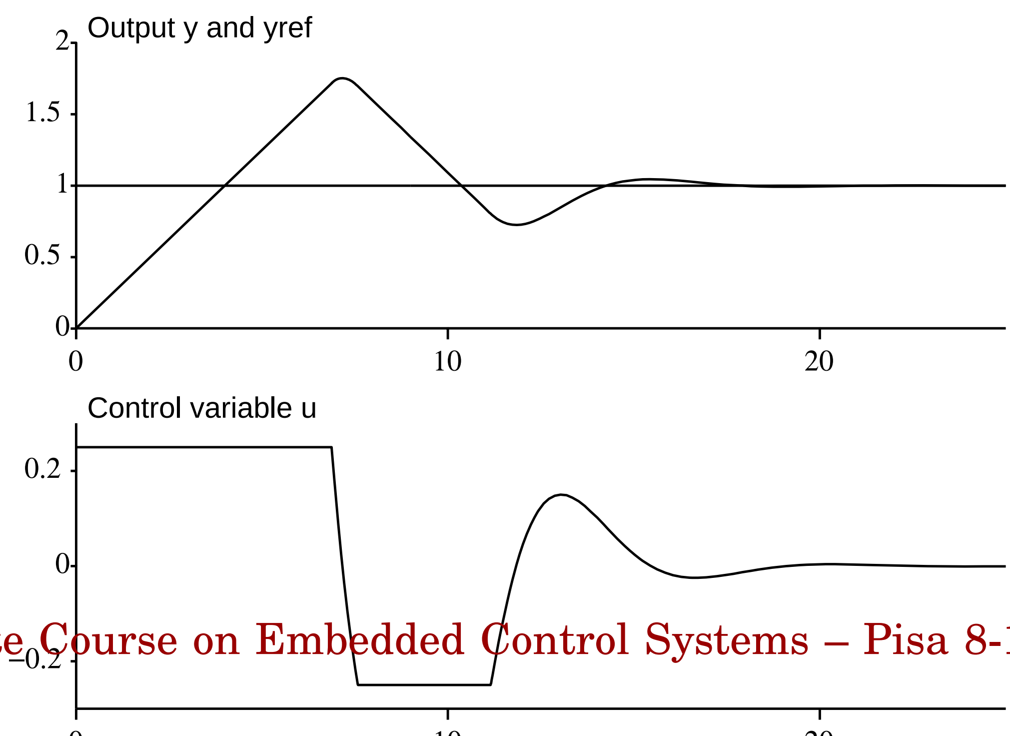
Control signal limitations

All actuators saturate.

Problems for controllers with integration.

When the control signal saturates the integral part will continue to grow – integrator windup.

When the control signal saturates the integral part will integrate up to a very large value. This may cause large overshoots.



Anti-windup

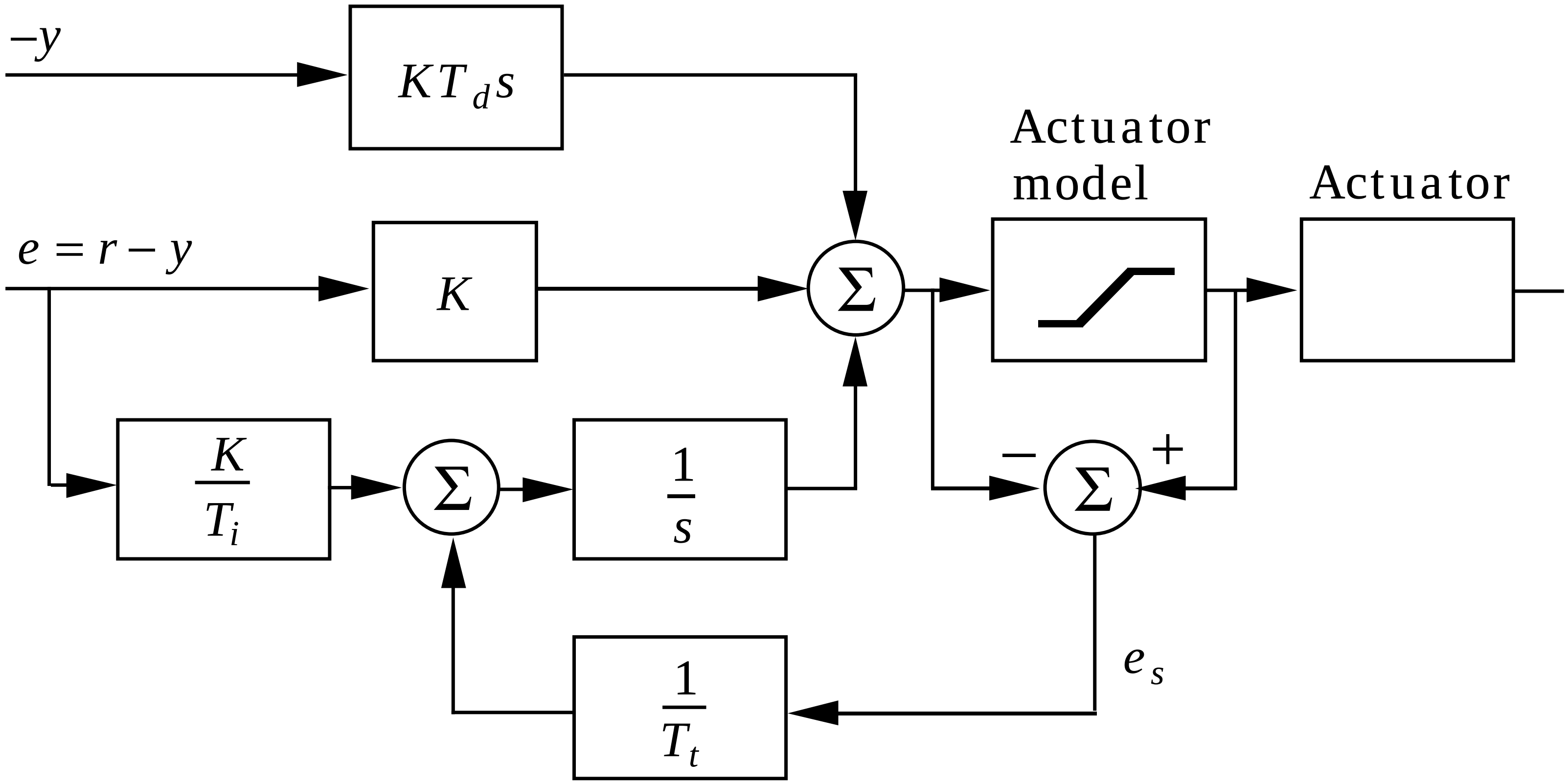
Several solutions exist:

- limit the reference variations (saturation never reached)
- conditional integration (integration is switched off when the control is far from the steady-state)
- tracking (back-calculation)

Tracking

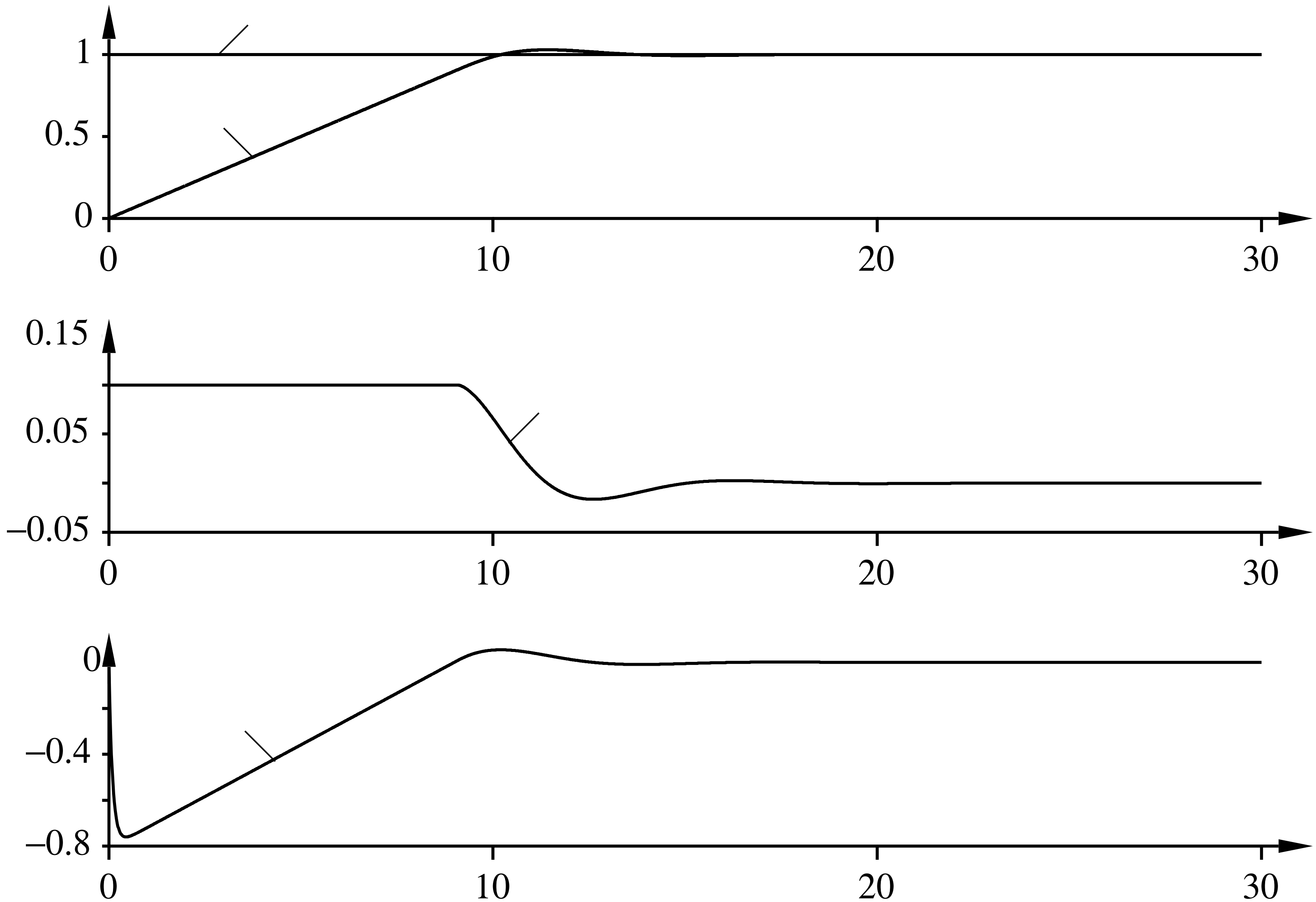
- when the control signal saturates, the integral is recomputed so that its new value gives a control signal at the saturation limit
- to avoid resetting the integral due to, e.g., measurement noise, the recomputation is done dynamically, i.e., through a LP-filter with a time constant T_t.

Tracking



$-y$
$KT_d s$
Actuator
model
Actuator
$e = r - y$
K
Σ
$\frac{K}{T_i}$
Σ
$\frac{1}{s}$
$-$
$+$
Σ
e_s
$\frac{1}{T_t}$

Tracking



Tuning

Parameters: $K, T_i, T_d, N, \gamma, T_t$

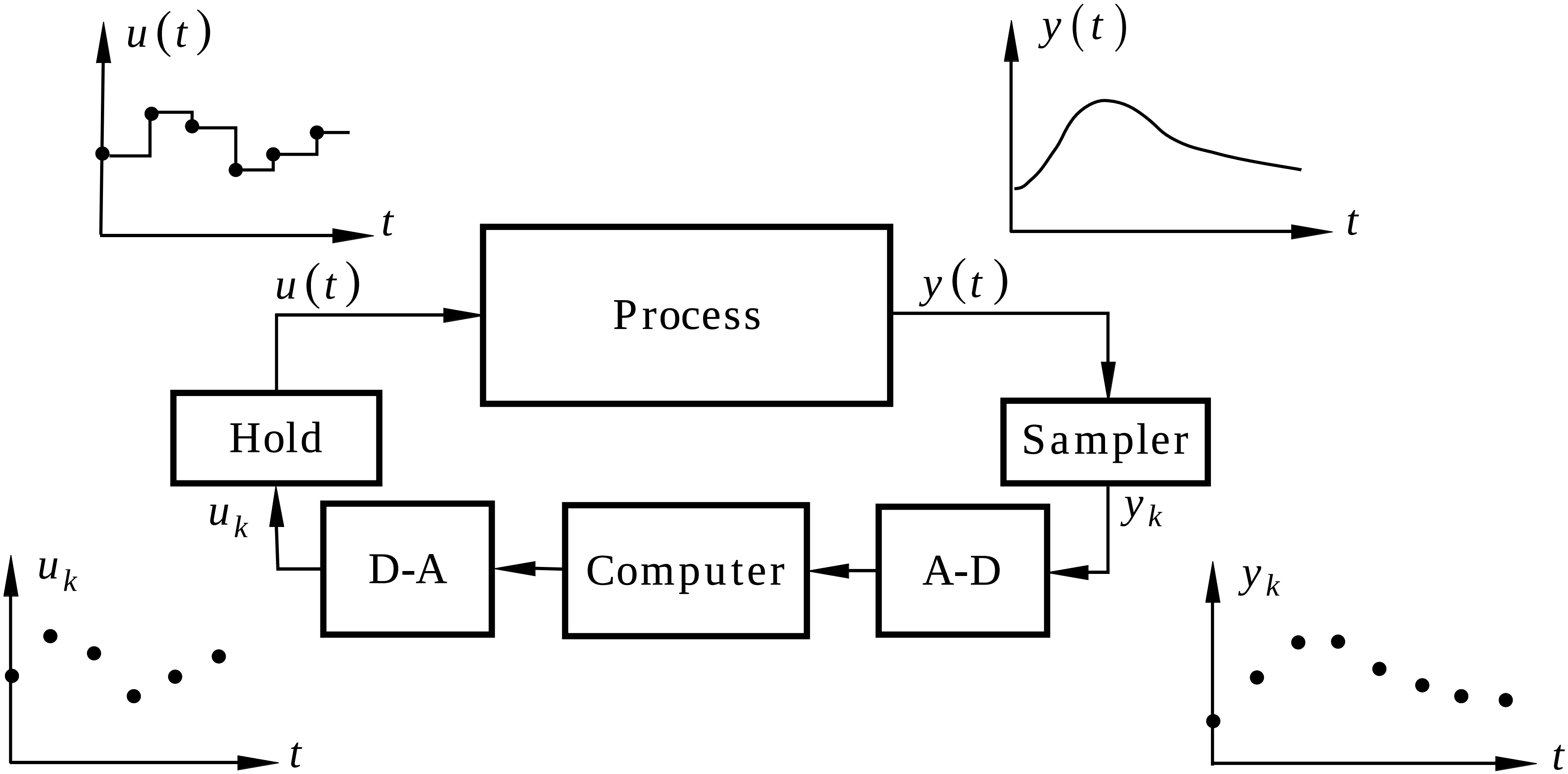
Methods:

- empirically, rules of thumb, tuning charts
- model-based tuning, e.g., pole-placement
- automatic tuning experiment
  - Ziegler-Nichols method
    - step response method
    - ultimate sensitivity method
  - relay method

Outline

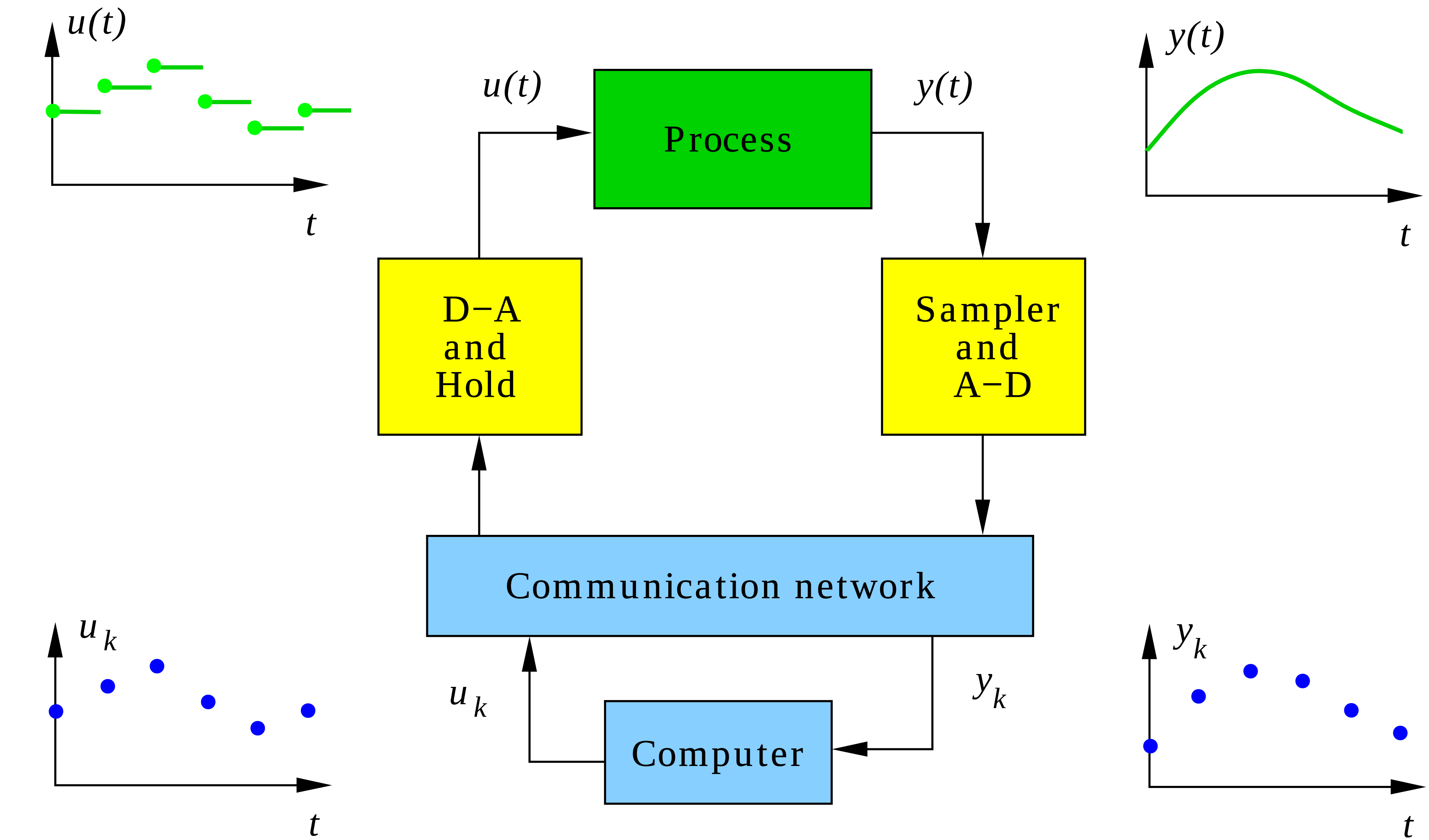
1. Introduction
2. Basic concepts
3. Modeling and design
4. Empirical PID control
5. **Digital control**

Sampled-data control systems



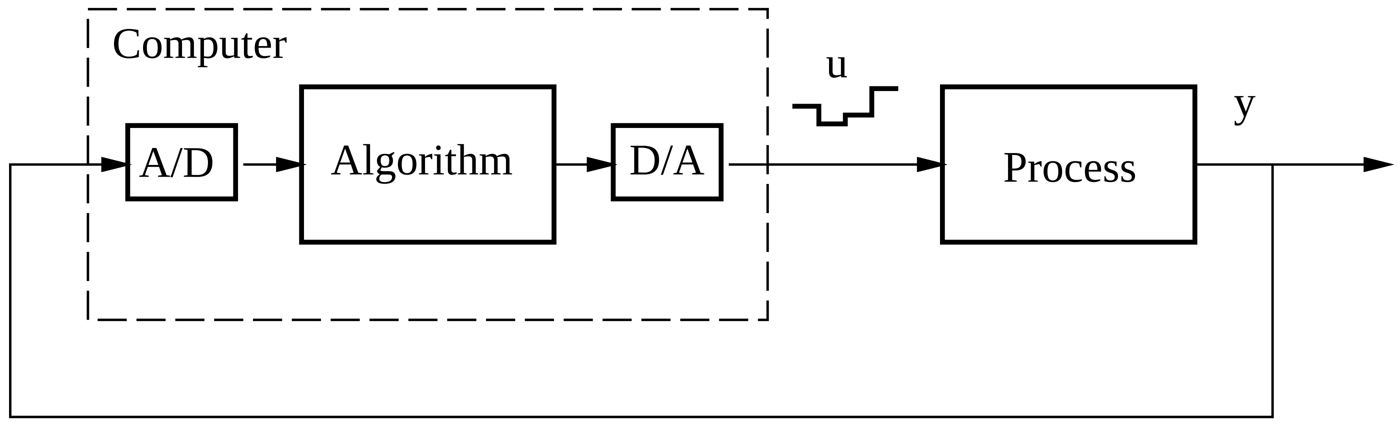
- Mix of continuous-time and discrete-time signals

Networked control systems

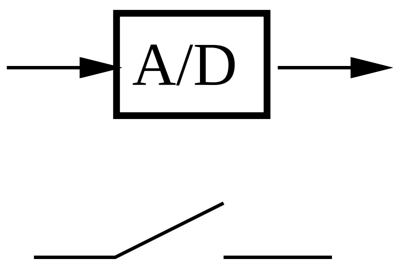


xtra delay, possibly lost packets

Sampling



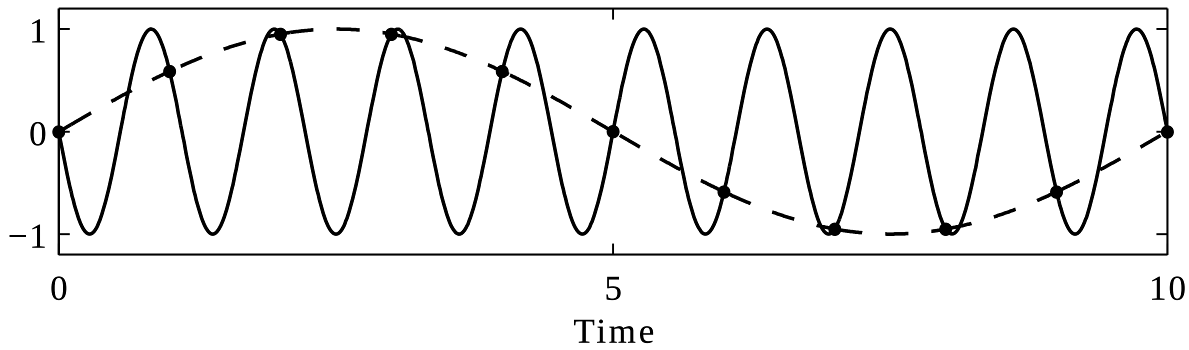
AD-converter acts as sampler



DA-converter acts as a hold device

Normally, zero-order-hold is used $\Rightarrow$ piecewise constant control signals

Aliasing



$$\omega_s = \frac{2\pi}{h} = \text{sampling frequency}$$

$$\omega_N = \omega_s/2 = \text{Nyquist frequency}$$

Frequencies above the Nyquist frequency are folded and appear as low-frequency signals.

The fundamental alias frequency for a frequency f_1 is given by

$$f = |(f_1 + f_N) \bmod (f_s) - f_N|$$

$f_1 = 0.9, f_s = 1, f_N = 0.5, f = 0.1$

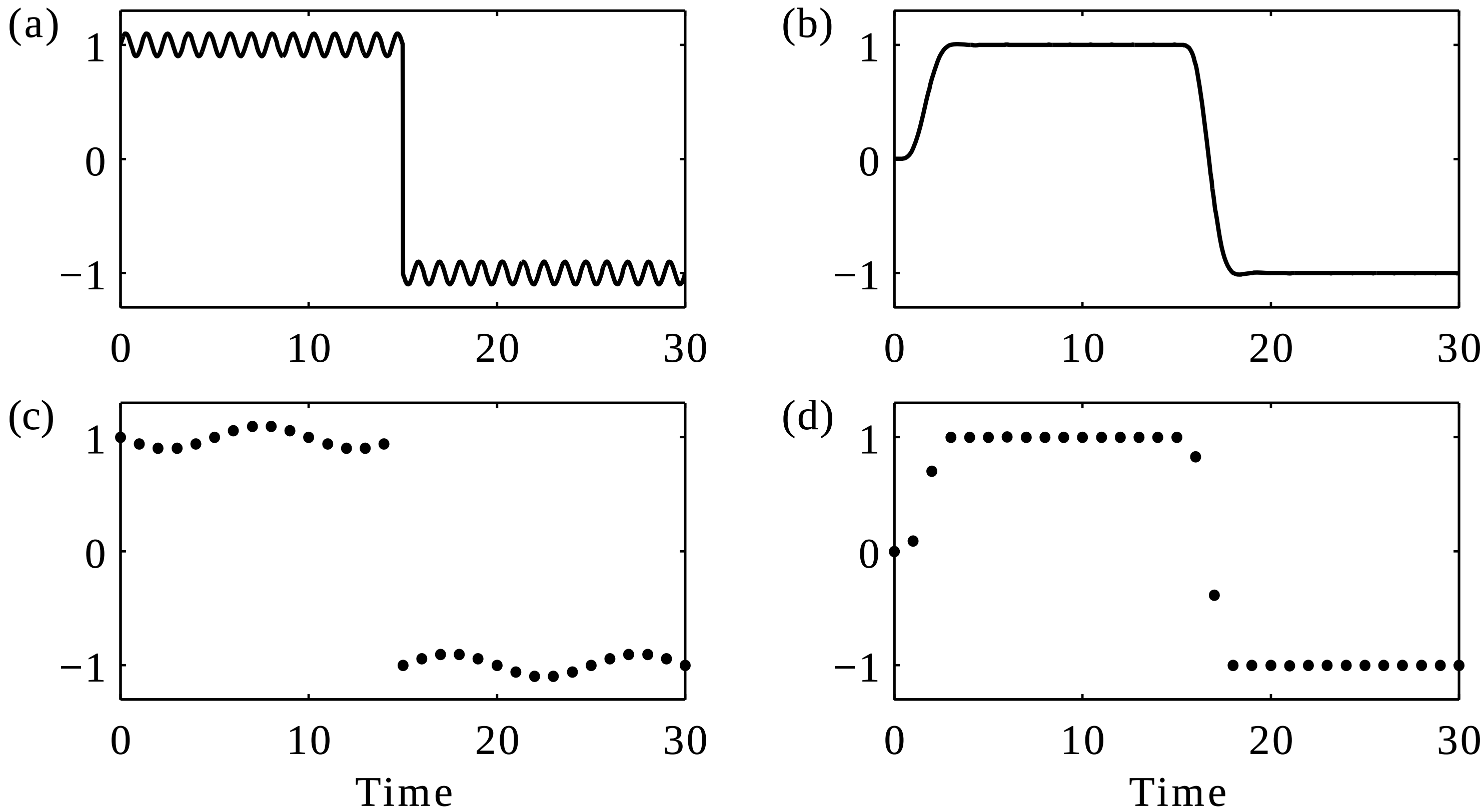
Anti-aliasing filter

Analog low-pass filter that eliminates all frequencies above the Nyquist frequency

- Analog filter
  - 2-6th order Bessel or Butterworth filter
  - Difficulties with changing h (sampling interval)
- Analog + digital filter
  - Fixed, fast sampling with fixed analog filter
  - Downsampling using digital LP-filter
  - Control algorithm at the lower rate
  - Easy to change sampling interval

The filter may have to be included in the control design

Example – Prefiltering



$\omega_d = 0.9$, $\omega_N = 0.5$, $\omega_{alias} = 0.1$

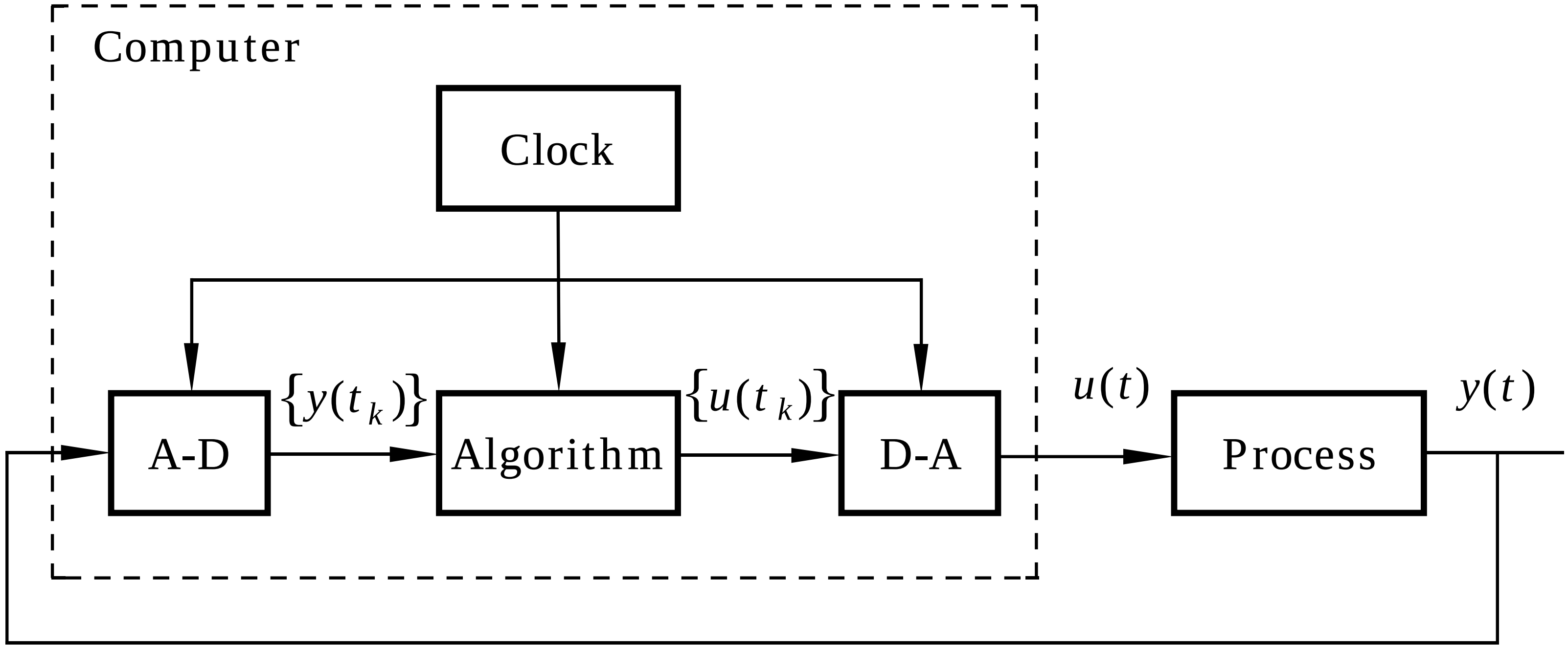
6th order Bessel with $\omega_B = 0.25$

Design approaches

Digital controllers can be designed in two different ways:

- Discrete-time design – sampled control theory
  - Sample the continuous system
  - Design a digital controller for the sampled system
    - Z-transform domain
    - state-space domain
- Continuous time design + discretization
  - Design a continuous controller for the continuous system
  - Approximate the continuous design
  - Use fast sampling

Sampled control theory

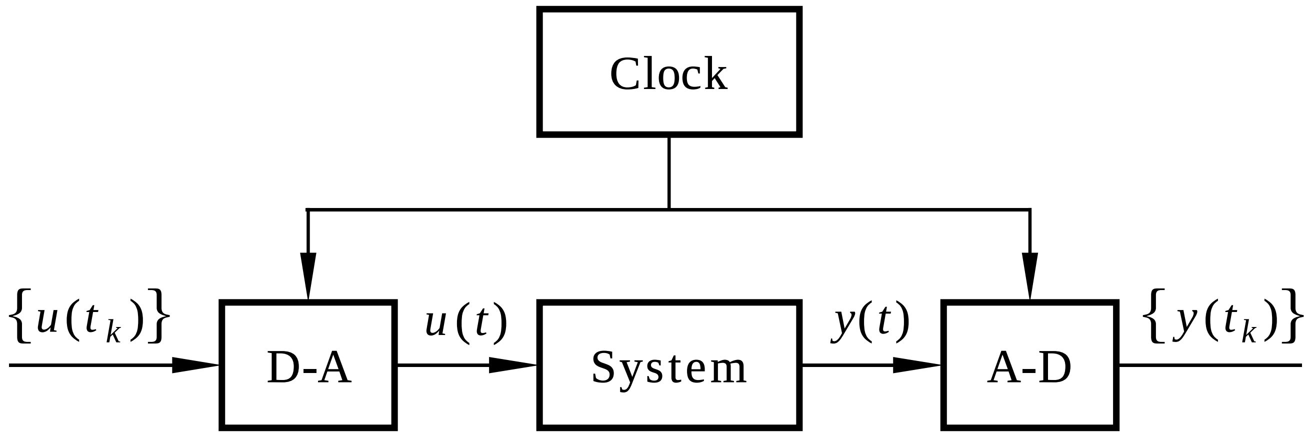


Basic idea: look at the sampling instances only

- System theory analogous to continuous-time systems
- Better performance can be achieved
- Potential problem with intersample behaviour

Sampling of systems

Look at the system from the point of view of the computer



Zero-order-hold sampling of a system

- Let the inputs be piecewise constant
- Look at the sampling points only
- Solve the system equation

Sampling a continuous-time system

System description

$$\begin{aligned} \frac{dx}{dt} &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

Solve the system equation

$$\begin{aligned} x(t) &= e^{A(t-t_k)}x(t_k) + \int_{t_k}^{t} e^{A(t-s')}Bu(s')\,ds' \\ &= e^{A(t-t_k)}x(t_k) + \int_{t_k}^{t} e^{A(t-s')}\,ds'\,Bu(t_k) \quad (u \text{ const.}) \\ &= e^{A(t-t_k)}x(t_k) + \int_{0}^{t-t_k} e^{As}\,ds\,Bu(t_k) \quad \text{(variable change)} \\ &= \Phi(t,t_k)x(t_k) + \Gamma(t,t_k)u(t_k) \end{aligned}$$

Periodic sampling

Assume periodic sampling, i.e. $t_k = k \cdot h$, then

$$\begin{aligned} x(kh+h) &= \Phi x(kh) + \Gamma u(kh) \\ y(kh) &= Cx(kh) + Du(kh) \end{aligned}$$

where

$$\begin{aligned} \Phi &= e^{Ah} \\ \Gamma &= \int_0^h e^{As}\, ds\, B \end{aligned}$$

Time-invariant linear system!

Example: Sampling of inverted pendulum

$$\begin{aligned} \frac{dx}{dt} &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u \\ y &= \begin{pmatrix} 1 & 0 \end{pmatrix} x \end{aligned}$$

We get

$$\begin{aligned} \Phi &= e^{Ah} = \begin{pmatrix} \cosh h & \sinh h \\ \sinh h & \cosh h \end{pmatrix} \\ \Gamma &= \int_0^h \begin{pmatrix} \sinh s \\ \cosh s \end{pmatrix} ds = \begin{pmatrix} \cosh h - 1 \\ \sinh h \end{pmatrix} \end{aligned}$$

Several ways to calculate Φ and Γ. Matlab

Sampling a system with a time delay

Sampling the system

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t-\tau), \quad \tau \leq h$$

we get the discrete-time system

$$x(kh+h) = \Phi x(kh) + \Gamma_0 u(kh) + \Gamma_1 u(kh-h)$$

where

$$\Phi = e^{Ah}$$

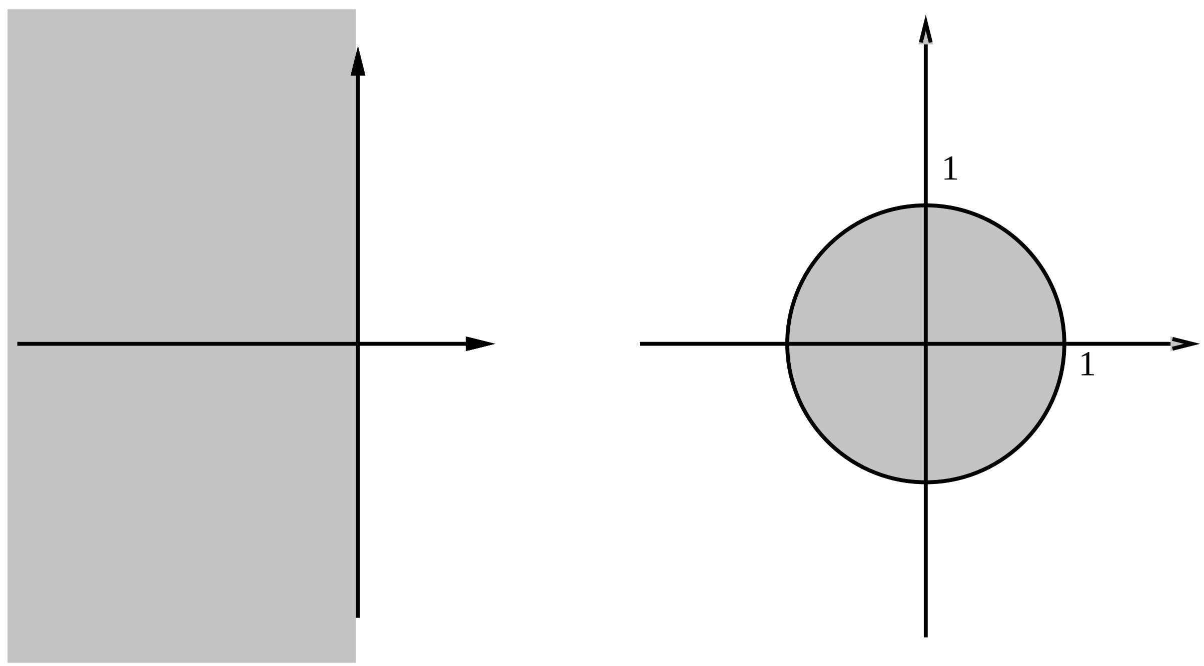
$$\Gamma_0 = \int_0^{h-\tau} e^{As}\, ds\ B$$

$$\Gamma_1 = e^{A(h-\tau)} \int_0^{\tau} e^{As}\, ds\ B$$

We get one extra state $\big(u(kh-h)\big)$ in the sampled system

Stability region

- In continuous time the stability region is the complex left half plane, i.e,, the system is stable if all the poles are in the left half plane.
- In discrete time the stability region is the unit circle.



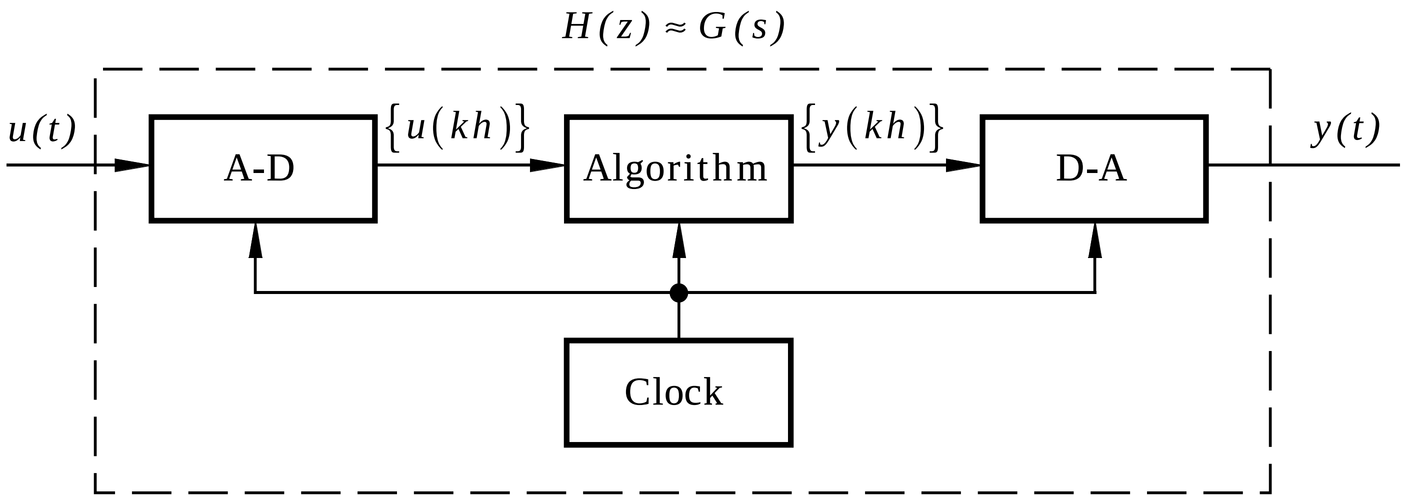
Digital control design

Similar to the continuous-time case, we can choose between

- frequency-domain design (loop shaping)
- pole-placement design
  - transfer function domain
  - state space domain
  - the poles are placed inside the unit circle
- optimal design methods (e.g. LQG)

Approximation of continuous-time design

Basic idea: Reuse the design



$G(s)$ is designed based on analog techniques

Want to get:

- A/D + Algorithm + D/A $\approx G(s)$

Methods:

- proximate s, i.e., $H(z) = G(s')$
- Other methods (Matlab)

Approximation methods

Forward Difference (Euler's method)

$$\frac{dx(t)}{dt} \approx \frac{x(t+h) - x(t)}{h}$$

$$s' = \frac{z-1}{h}$$

Backward Difference

$$\frac{dx(t)}{dt} \approx \frac{x(t) - x(t-h)}{h}$$

$$s' = \frac{z-1}{zh}$$

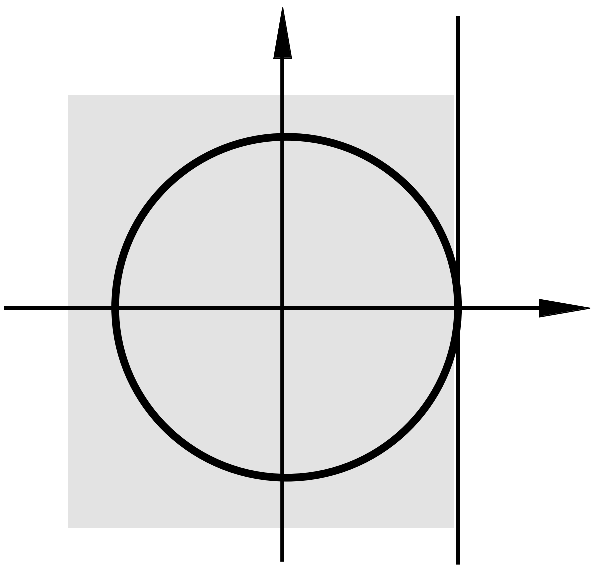
Tustin

$$\frac{\frac{dx(t)}{dt} + \frac{dx(t+h)}{dt}}{2} \approx \frac{x(t+h) - x(t)}{h}$$

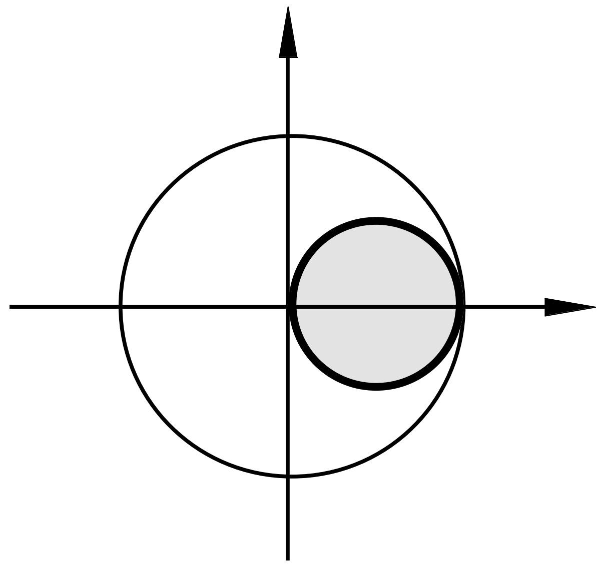
$$s' = \frac{2}{h}\,\frac{z-1}{z+1}$$

Stability of approximations

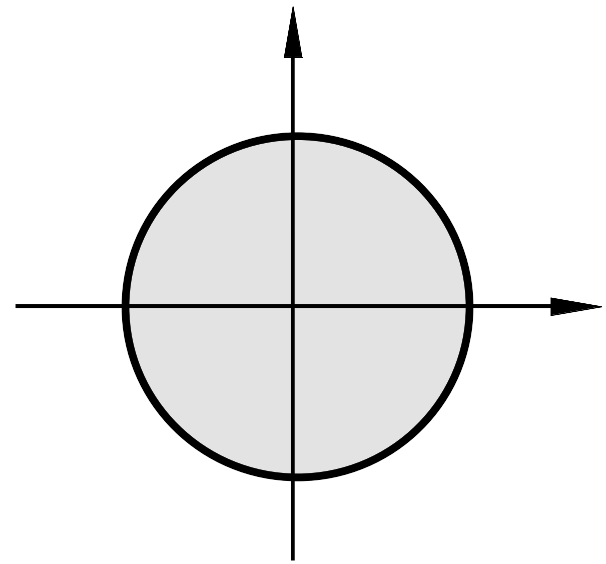
How is the continuous-time stability region (left half plane) mapped?



Forward differences



Backward differences



Tustin

Discretization of the PID controller

Continuous PID controller with $\gamma = 0$:

$$U(s) = K\left(R(s) - Y(s)) + \frac{1}{sT_i}\left(R(s) - Y(s)\right) - \frac{sT_d}{1 + sT_d/N}\,Y(s)\right)$$

Discretization

P-part:

$$P(k) = K(r(k) - y(k))$$

Discretization

I-part:

$$I(t) = \frac{K}{T_I} \int_0^t (r(\tau) - y(\tau)) d\tau$$

$$\frac{dI}{dt} = \frac{K}{T_I}(r(t) - y(t))$$

- Forward difference

$$\frac{I(k+1) - I(k)}{h} = \frac{K}{T_I}(r(k) - y(k))$$

```
I(k+1) := I(k) + (K*h/Ti)*(r(k)-y(k))
```

The I-part can be precalculated

- Backward difference

The I-part cannot be precalculated, $I(k) = f(r(k), y(k))$

Discretization

D-part (assume $\gamma = 0$):

$$D = K \frac{sT_D}{1 + sT_D/N}(-Y(s))$$

$$\frac{T_D}{N}\frac{dD}{dt} + D = -KT_D\frac{dy}{dt}$$

- Forward difference (unstable for small T_D)
- Backward difference

$$\frac{T_D}{N}\frac{D(k) - D(k-1)}{h} + D(k) = -KT_D\frac{y(k) - y(k-1)}{h}$$

$$D(k) = \frac{T_D}{T_D + Nh}D(k-1) - \frac{KT_DN}{T_D + Nh}(y(k) - y(k-1))$$

Discretization

Tracking:

```
v := P + I + D;
u := sat(v,umax,umin);
I := I + (K*h/Ti)*(r-y) + (h/Tt)*(u - v);
```

PID code

PID-controller with anti-windup ($\gamma = 0$).

```
r = ref.get();
y = yIn.get();
D = ad * D - bd * (y - yold);
v = K*(r - y) + I + D;
u = sat(v,umax,umin);
uOut.put(u);
I = I + (K*h/Ti)*(r - y) + (h/Tt)*(u - v);
yold = y;
```

`ad` and `bd` are precalculated parameters given by the backward difference approximation of the D-term.

Further Reading

- B. Wittenmark, K. J. Åström, K.-E. Årzén: "Computer Control: An Overview." IFAC Professional Brief, 2002.
  (93 pages, available at http://www.control.lth.se)
- K. J. Åström, Tore Hägglund: "Advanced PID Control." The Instrumentation, Systems, and Automation Society, 2005.
- A. Cervin: "Integrated Control and Real-Time Scheduling." PhD Thesis, Lund University, 2003. (Available at http://www.control.lth.se)