MAE291 - Spring 2005

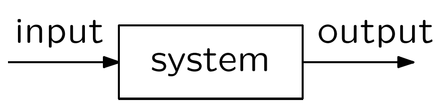
discrete-time system & servo control

short overview of contents of this lecture

- definition of dynamic input/output system
- analysis of dynamic systems
  - discrete & continuous time systems
  - transfer functions
  - example: HDD actuator model
  - stability
- control systems
  - feedforward architecture
  - feedback design
  - loopgain and Nyquist stability criterion
  - loopshaping using a lead/lag controller
  - example: control of inertial system

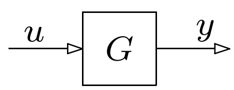
Input/output system

System with an input (signal) and an output (signal).



block diagram

Shorthand notation that we use: $y = Gu$



input u, control output y

Distinction between input and output in the block diagram is based on causality principle.

Input/output system

Causality in input/output system: change in output at current time is a reaction to the input in the past.

Notation: $y(t)$ = output at time t, $u(t)$ = input at time t

$$
\begin{aligned}
\text{then:} \quad y(t_2) &= G(u(t_1)), \quad t_2 > 0, \quad t_1 \in [0, \ldots, t_2] \\
\text{or:} \quad y(t) &= G(u(t), u(t-1), y(t-1), u(t-2), y(t-2), \ldots)
\end{aligned}
$$

G denotes an operator (or function) that relates past inputs and outputs to the output $y(t)$ at time instant t

G can be a differential operator (from a differential equation) or a difference operator (from a difference equation)

In short: G is *a model* that describes *the dynamics of our system* to be controlled

Input/output system

Important properties on input/output systems:

- Causality (as defined before)
- Stability $\leftrightarrow$ instability

  Let $y = Gu$ then G is stable if $\|y\| < \infty$ for all possible inputs u with $\|u\| < \infty$
- Linearity $\leftrightarrow$ non-linearity

  Let $u = \alpha_1 u_1 + \alpha_2 u_2$ then G is linear if $G(\alpha_1 u_1 + \alpha_2 u_2) = \alpha_1 G u_1 + \alpha_2 G u_2$
- Time invariant $\leftrightarrow$ time variant

  Let $u(t+T) = u(t)$, then G is time invariant if $G(u(t+T)) = G(u(t))$
- Discrete time $\leftrightarrow$ continuous time

  Let $y(t) = G(u(t))$ then G is a discrete system if $t \in \mathcal{Z}$

Dynamical systems: discrete time

In many dynamical systems, only discrete measurements of the input/output values are available (computer controlled systems, a HDD with servo sectors).

To describe *linear time invariant causal discrete time system*, we can use an ordinary linear difference equation:

$$\sum_{k=1}^{n} c_k q^k y(t) = \sum_{k=1}^{n} d_k q^k u(t)$$

with $qy(t) := y(t+1)$, $t \in \mathcal{Z}$, and appropriate initial conditions.

Example
Money $y(t)$ *in a simplified account at 5% interest, subjected to withdrawals or deposits* $u(t)$ *and compounded annually:*

$$y(t+1) = 1.05y(t) + u(t)$$

with y(0)=0

Dynamical systems: discrete time

Similarly as Laplace transform for continuous time signals, we introduce the z-transform for discrete time signals:

$$y(z) := \mathcal{Z}\{y(t)\} = \sum_{t=0}^{\infty} y(t) z^{-t}$$

that converts linear difference equations into algebraic expressions, as

$$\mathcal{Z}\{\sum_{k=1}^{n} c_k q^k y(t)\} = y(z) \sum_{k=1}^{n} c_k z^k$$

rewriting

$$\sum_{k=1}^{n} c_k q^k y(t) = \sum_{k=1}^{n} d_k q^k u(t)$$

into

$$y(z) = G(z)u(z), \text{ with } G(z) = \frac{\sum_{k=1}^{n} d_k z^k}{\sum_{k=1}^{n} c_k z^k}$$

Dynamical systems: continuous & discrete time

Discrete time system often due to sampling or approximation of continuous time system.

Consider the first order differential equation

$$\frac{\partial}{\partial t}y(t) - ay(t) = bu(t),\ y(0) = 0$$

or

$$y(s) = G(s)u(s),\ \text{with } G(s) = \frac{b}{s-a}$$

How to obtain a discrete time model from a continuous time model?

Difference between discrete time models is due to assumptions made to approximate input behavior, differentiation operator or integration operator.

Dynamical systems: continuous & discrete time

Consider again first order model

$$\frac{\partial}{\partial t}y(t) = ay(t) + bu(t) \text{ or } y(s) = G(s)u(s) \text{ with } G(s) = \frac{b}{s-a}$$

One possibility: approximation of the derivative
This is know as the **Euler approximation**, where

$$\frac{\partial}{\partial t}y(t) \approx \frac{y(t+\Delta t) - y(t)}{\Delta t}$$

With $\Delta t := 1$ and $t \in \mathcal{Z}$ and the Euler approximation we find

$$y(t+1) = y(t) + ay(t) + bu(t)$$

or

$$y(z) = G(z)u(z) \text{ with } G(z) = \frac{b}{(z-1)-a}$$

a discrete time model by substitution of $s = z - 1$.

Dynamical systems: continuous & discrete time

Alternative approach: approximation of the integral.
By rewriting

$$\frac{\partial}{\partial t}y(t) - ay(t) = bu(t),\ y(0) = 0$$

into

$$y(t) = \int_{\tau=0}^{\infty}[ay(\tau) + bu(\tau)]d\tau$$

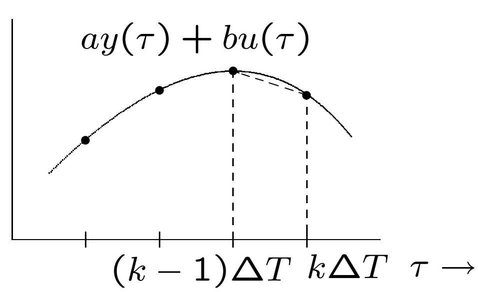
we see that the value of $y(t)$ at the k'th sampling time $k\Delta T$:

$$y(k\Delta T) = y((k-1)\Delta T) + \int_{\tau=(k-1)\Delta T}^{k\Delta T}[ay(\tau) + bu(\tau)]d\tau$$

$\Rightarrow$ approximate the integral from $\tau = (k-1)\Delta T$ to $\tau = k\Delta T$ to obtain a discrete time model equivalent of the continuous time model

Dynamical systems: continuous & discrete time

One choice: trapezoidal approximation of integral
Also know as **Tustin approximation**



Trapezoidal area:

$$[ay(k\Delta T) + bu(k\Delta T)]\Delta T +$$

$$([ay((k-1)\Delta T) + bu((k-1)\Delta T)] -$$

$$[ay(k\Delta T) + bu(k\Delta T)])\Delta T/2$$

Dynamical systems: continuous & discrete time

With the size of this Trapezoidal area we get:

$$y(k\Delta T) \approx y((k-1)\Delta T) +$$
$$\left([ay(k\Delta T) + bu(k\Delta T)] + [ay((k-1)\Delta T) + bu((k-1)\Delta T)]\right)\Delta T/2$$

and by rewriting we recognize

$$y(k\Delta T) = \frac{1+a\Delta T/2}{1-a\Delta T/2} y((k-1)\Delta T) +$$
$$\frac{b\Delta T/2}{1-a\Delta T/2} u((k-1)\Delta T) +$$
$$\frac{b\Delta T/2}{1-a\Delta T/2} u((k)\Delta T)$$

a discrete time model!

Dynamical systems: continuous & discrete time

This discrete time model can be written in a transfer function with the z-transform:

$$y(z) = G(z)u(z), \text{ with } G(z) = \frac{b}{\frac{2}{\Delta T}\frac{(z-1)}{(z+1)} - a}$$

Compare with original continuous time model $G(s)$:

$$G(s) = \frac{b}{s-a}$$

Substitution $s = \frac{2}{\Delta T}\frac{(z-1)}{(z+1)}$ and $z = \frac{(2/\Delta T + s)}{(2/\Delta T - s)}$ are also called bilinear transformation or Tustin's formulae.

Implemented in Matlab in the function `c2d`:

```
SYSD = C2D(SYSC,TS,'TUSTIN')
```

Dynamical systems: transfer function representations

The transfer functions $G(s)$ and $G(z)$ represent models respectively of a continuous time system and a discrete time system

The notation $y = Gu$ (compare with our block diagrams) denotes a *transfer function* representation of the system with G given by

$$G(s) = \frac{\sum_{k=1}^{n} b_k s^k}{\sum_{k=1}^{n} a_k s^k} \text{ or } G(z) = \frac{\sum_{k=1}^{n} c_k z^k}{\sum_{k=1}^{n} d_k z^k}$$

Notes

- the coefficients a_k and b_k or c_k and d_k completely determine the dynamic behavior of the model!
- the response $y(t)$ can be computed by $G(s)$ or $G(z)$ together with the additional initial conditions
- Difference between continuous time and discrete time lies in transformation (Laplace or z-transform)

Dynamical systems: transfer function representations

Important information deducted from a transfer function G (continuous or discrete time):

- Frequency response of the model.

  Let $u(t) = cos(\omega t)$ then for $t >> 1$,
  $y(t) = |G(\omega)| cos(\omega t + \angle G(\omega))$ where

  $$\begin{aligned} G(\omega) &= G(s),\ s = j\omega,\ \omega \in [0, \infty) \\ G(\omega) &= G(z),\ z = \mathrm{e}^{j\omega},\ \omega \in [0, \pi] \end{aligned}$$

  Difference lies in the transformation being used. Matlab functions: `tf`, `bode`, `dbode`
- Stability of the models

  Location of the roots of the denominator of the transfer function determine the stability of the model G

Dynamical systems: example - HDD model

Standard Transfer Function (STF) model of a second order system:

$$G(s) = \frac{\omega_n^2}{s^2 + 2\beta\omega_n s + \omega_n^2}$$

ω_n = undamped frequency in rad/s and β = damping coefficient

- Low frequency mode due to free body motion of actuator with $f_n = 25$Hz, $\beta = 0.15$
- E-block sway mode at $f_n = 5$KHz, $\beta = 0.0125$
- Suspension torsion mode at $f_n = 5.93$KHz, $\beta = 0.0105$ (denominator) and $f_n = 6$KHz, $\beta = 0.0105$
- Suspension sway mode at $f_n = 9$KHz, $\beta = 0.0045$

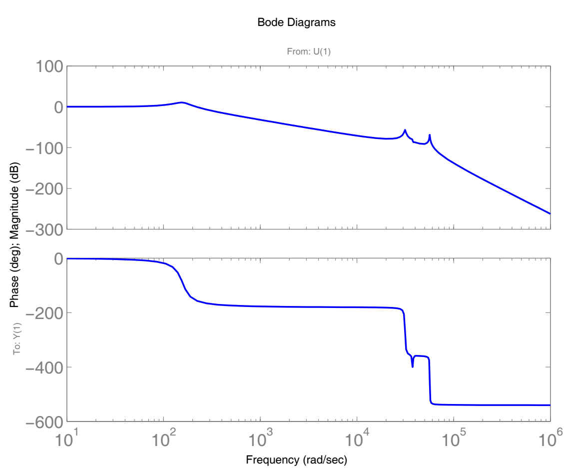
Combination of STF's yields $G(s)$ of simple HDD model:

$$\frac{7.6 \cdot 10^{22}(s^2 + 791.7s + 1.4 \cdot 10^9)}{s^8 + 2127s^7 + 5.5 \cdot 10^9 s^6 + 8.3 \cdot 10^{12} s^5 + 9 \cdot 10^{18} s^4 + 7.1 \cdot 10^{21} + 4.4 \cdot 10^{27} s^2 + 2 \cdot 10^{29} s + 1 \cdot 10^{32}}$$

Dynamical systems: example - HDD model

Bode plot of TF model

$$\frac{7.6 \cdot 10^{22}(s^2 + 791.7s + 1.4 \cdot 10^9)}{s^8 + 2127s^7 + 5.5 \cdot 10^9 s^6 + 8.3 \cdot 10^{12} s^5 + 9 \cdot 10^{18} s^4 + 7.1 \cdot 10^{21} + 4.4 \cdot 10^{27} s^2 + 2 \cdot 10^{29} s + 1 \cdot 10^{32}}$$



Note: observe bad numerical conditioning of TF model! Improvement possible by formulation in **State Space form**.

Dynamical systems: stability of continuous time models

Consider again continuous time first order model

$$\frac{\partial}{\partial t}y(t) = ay(t) + bu(t) \text{ or } y(s) = G(s)u(s) \text{ with } G(s) = \frac{b}{s-a}$$

A homogeneous solution to difference equation:

$$y(t) = \mathrm{e}^{at}$$

satisfies $\|y\| < \infty$ iff $a < 0$. Equivalent to condition that the root s_1 of $(s-a) = 0$ satisfies $s_1 < 0$.

Stability statement can be generalized to higher order differential equations (degree n denominator).

> For continuous time systems, the roots s_k of denominator of $G(s)$ should satisfy $\mathrm{Re}\{s_k\} < 0, \ k = 1, \ldots, n$. Alternatively: poles of $G(s)$ should lie in the open-left half of the complex plane.

Dynamical systems: stability of discrete time models

Consider discrete time first order model

$$y(t+1) = cy(t) + du(t) \text{ or } y(z) = G(z)u(z) \text{ with } G(z) = \frac{d}{z-c}$$

A homogeneous solution to difference equation:

$$y(t) = c^t \ t \in \mathcal{Z}$$

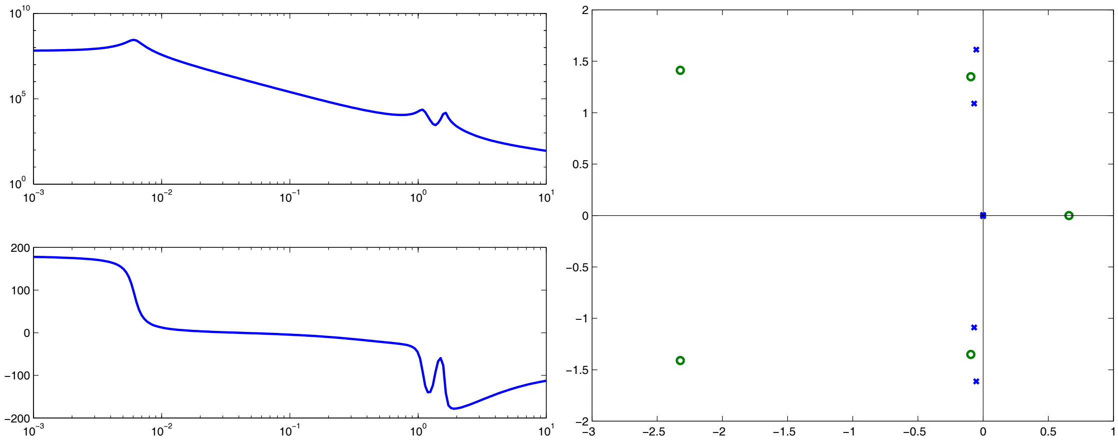
satisfies $\|y\| < \infty$ iff $c < 1$. Equivalent to condition that the root z_1 of $(z-a) = 0$ satisfies $z_1 < 1$.

Stability statement can be generalized to higher order difference equations (degree n denominator).

> For discrete time systems, the roots z_k of the denominator of $G(z)$ should satisfy $|z_k| < 1, \ k = 1, \ldots, n$. Alternatively: poles of $G(z)$ should lie inside the unit disk centered around the origin.

Dynamical systems: stability and oscillations

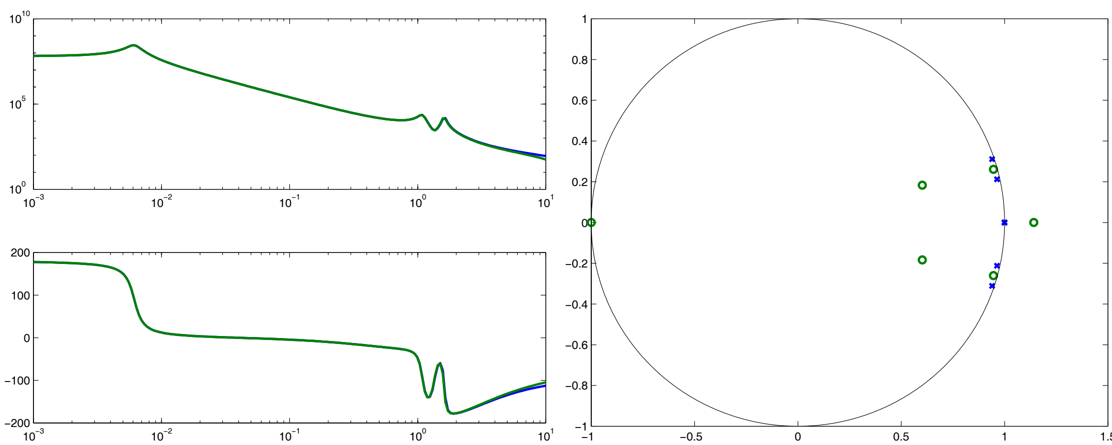
Examining frequency domain plot of G and accompanying pole/zero plot will give information on poorly damped poles (and zeros) that will play an important role in control system design.



Continuous time model of a hard disk drive rotary actuator: Bode plot (left) and pole/zero plot (right)

Dynamical systems: stability and oscillations

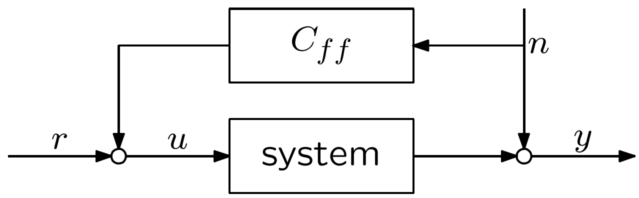
Comparison between continuous time model and discrete time model of hard disk drive rotary actuator



Bode plot of continuous and discrete time model (left) and polse/zero plot of discrete time model (right)

Control systems

General form of *feedforward* control



block diagram of feed forward

Signals:

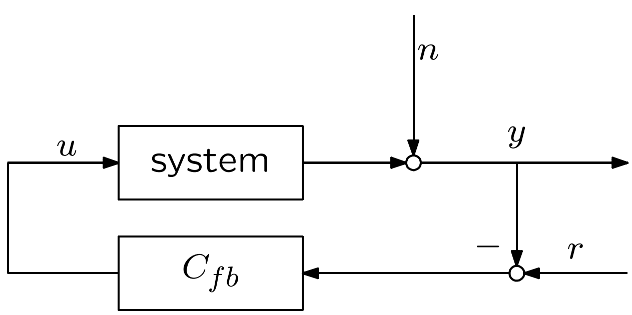
- reference signal $= r$
- input $= u$
- output $= y$
- perturbation or noise $= n$

Design considerations:

- measure perturbation or noise n
- implements ways to change input u
- reduce effect of disturbance n on output y

Control systems

General form of *feedback* control:



block diagram of feedback

Design considerations:

- measure output y
- implements ways to change input u
- reduce effect of disturbance n on output y
- modify open-loop to closed-loop dynamic properties

Control systems: feedforward

Feedforward control can be described by:

$$\begin{aligned} y &= Gu + n \\ u &= r + C_{ff}n \end{aligned}$$

which yields

$$\boxed{y = Gr + (1 + GC_{ff})n}$$

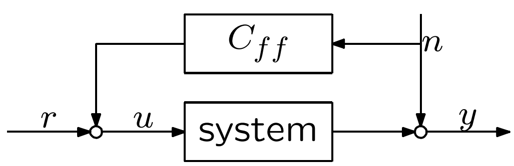
Note:
– r is new input signal (reference)
– perturbation or noise n acts on output y via $(1 + GC_{ff})$.

Hence:
– dynamic properties of $n \to y$ change
– dynamic properties of $r \to y$ do not change
– influence of noise n can be altered by design C_{ff}:

$$C_{ff} = -\frac{1}{G}$$

Control systems: feedforward

Design considerations and trade-offs in feedforward control



$$y = Gr + (1 + GC_{ff})n, \; C_{ff} = -\frac{1}{G}$$

If indeed $C_{ff} = -1/G$ and n = actual track on hard disk, then y = PES would be zero (perfect track following). However:

- $C_{ff} = -1/G$ might be non-causal or unstable
- G is just *a model of the system*! What if a small error is made while modeling the system?
- What if disturbances n cannot be measured or cannot be measured completely?
- The control signal $u = -1/G\,n$ might be very large.

Control systems: feedback

Feedback control can be described by:

$$\left.\begin{array}{rcl} y &=& Gu + n \\ u &=& C_{fb}(r - y) \end{array}\right\} \Rightarrow \; y = GC_{fb}r - GC_{fb}y + n$$

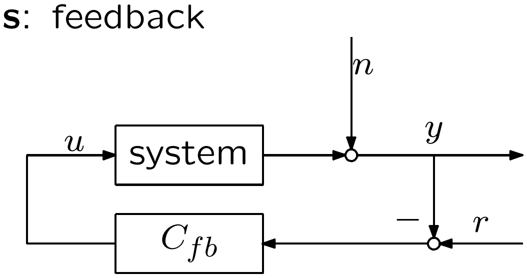
which yields

$$\boxed{y = \frac{GC_{fb}}{1 + GC_{fb}} r + \frac{1}{1 + GC_{fb}} n}$$

Note:
– r is new input signal (reference)
– noise n act on output in a dynamic way via $\dfrac{1}{1 + GC_{fb}}$.

Hence:
– dynamic properties of $n \rightarrow y$ change
– dynamic properties of $r \rightarrow y$ *also* change
– noise influence can be altered by design $C_{fb} >> 1$

Control systems: feedback



$$y = \frac{GC_{fb}}{1 + GC_{fb}} r + \frac{1}{1 + GC_{fb}} n, C_{fb} >> 1$$

If indeed $C_{fb} >> 1$ and $r =$ actual track on hard disk, then $y = r$ (PES would be zero) and effect of noise n is eliminated, but:

- For $C_{fb} >> 1$, the control signal $u = C_{fb}(r - y)$ might be very large.
- For $C_{fb} >> 1$, the feedback control system might exhibit oscillations or (even stronger) instabilities

Design considerations are less restricting than in feedforward! A model G is needed to study stability and control energy of the feedback control system.

Control systems: feedforward & feedback

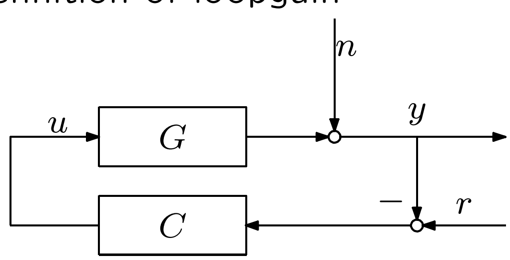
In modern harddisk drives (or any mechanical data storage system for that matter), the control system is usually a combination of feedforward and feedback.

- track seeking: feedforward control $u = C_{ff}n$, where n = servo track measurement and $C_{ff} = -1/G$ based on a relatively simple model G of the servo actuator.
- track following: feedback control $u = C_{fb}(r - y)$ where $(r - y)$ is a PES measurement and C_{fb} is designed on a more complicated model of the servo actuator to include stability analysis.

In the remaining part of this lecture:

- focus on the design of servo feedback controllers
- illustration via lead/lag based control

Loopshaping: definition of loopgain



Loopgain L is series connection of model G and controller C:

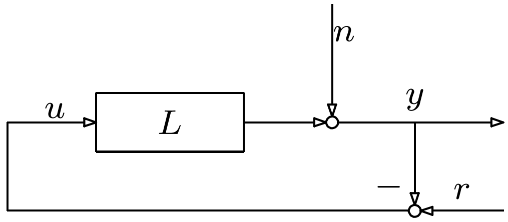
$$L(s) = G(s)C(s) \text{ or } L(z) = G(z)C(z)$$

and is found by 'cutting the feedback loop open'

Loopgain plays crucial role in feedback:

- transfer function from n to y: $\frac{1}{1+L}$ (sensitivity function)
- transfer function from r to y: $\frac{L}{1+L}$ (complementary sensitivity function)
- stability of the feedback loop: Nyquist criterion

Loopshaping: Nyquist stability criterion



Suppose n or r is a sinusoidal function $\cos(\omega t)$

Obviously, the signals in the feedback loop grow if a signal $|L|\cos(\omega t + \angle L)$ is both amplified in the loop and is provide with the right phase shift.

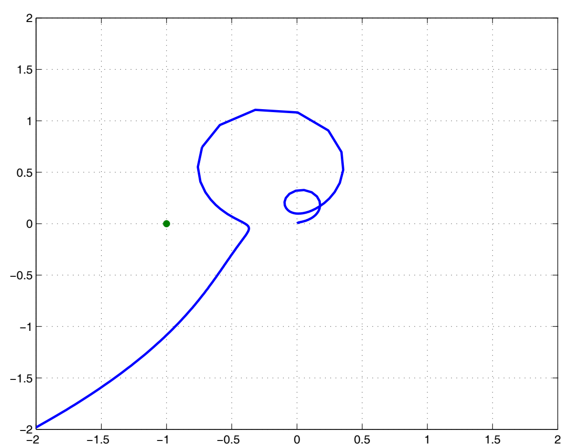
Amplification means: $|L(\omega)| > 1$, right phase shift means: $\angle L(\omega) = -180 - k360,\ k = 0, 1, \ldots$.

Nyquist stability criterion checks possibility of the encirclement of the singularity point $L(s) = -1$. In that case $|S| = \left|\frac{1}{(1+L)}\right| = \infty$ and closed-loop system is unstable.

Loopshaping: Nyquist stability criterion

Check of Nyquist stability criterion is done, by plotting the complex function $L(j\omega)$ or $L(e^{j\omega})$ in the complex plane

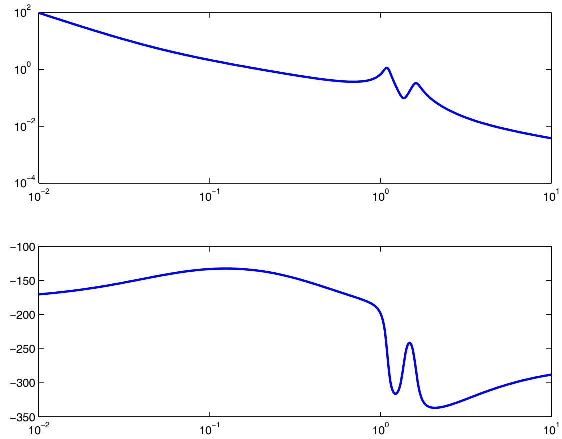
Example:



Stable or unstable?

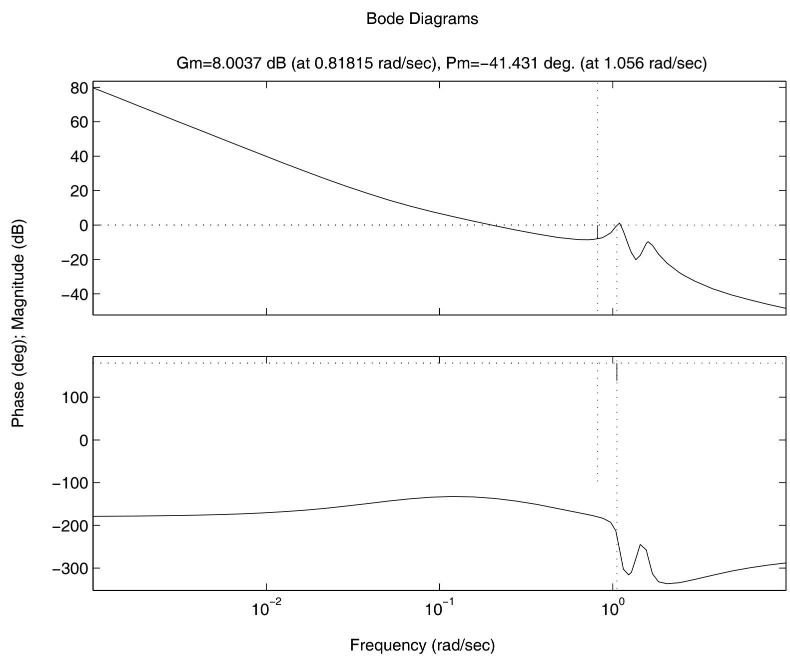
Loopshaping: Nyquist stability criterion

Instead of plotting real value of L against imaginary value of L, one can also simply plot the Bode plot:



Bode plot gives good indication of stability margins!

Loopshaping: crossover frequency, amplitude and phase margin



crossover frequency: smallest frequency where $|L| = 1$

Loopshaping: lead/lag compensation

Important question: how to 'shape the loopgain'?

As $L = GC$, obviously, the dominant behavior of L is dictated by our system (or our model) G. With the controller C we can perform additional shaping.

In general, C is a nth order transfer function (continuous time or discrete time) and is quite hard to design ($2n$ parameters)

For servo systems, typical (minimal) controller is:

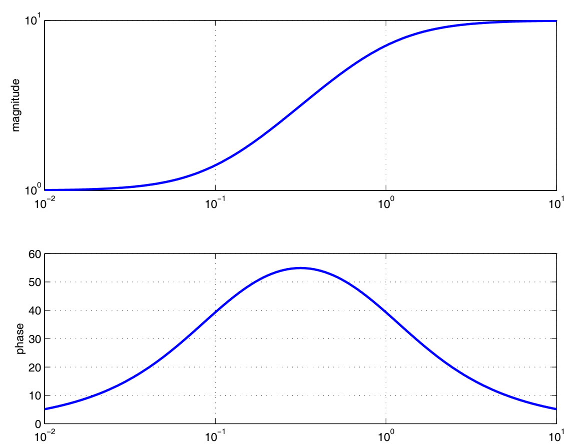
$$C(s) = K\frac{\tau_1 s + 1}{\tau_2 s + 1} \quad \text{or} \quad C(z) = K\frac{(1-a)}{(1-b)} \cdot \frac{(z-b)}{(z-a)}$$

and called a lead/lag compensator with three design parameters

Loopshaping: lead/lag compensation

$$C(s) = K\frac{\tau_1 s + 1}{\tau_2 s + 1}$$

Typical Bode plot of $C(s)$ for $\tau_1 > \tau_2$ and $K > 0$:



Bode plot for lead/lag compensator with parameters
$\tau_1 = 10$, $\tau_2 = 1$, $K = 1$

Loopshaping: lead/lag compensation

from transfer function of lead/lag compensator

$$C(s) = K\frac{\tau_1 s + 1}{\tau_2 s + 1}$$

we have

$$\begin{aligned} \tau_1 s + 1 = 0 &\;\Rightarrow\; s = -1/\tau_1 \text{ zero!} \\ \tau_2 s + 1 = 0 &\;\Rightarrow\; s = -1/\tau_2 \text{ pole!} \\ C(0) = C(j\omega),\ \omega = 0 &\;\Rightarrow\; C(0) = K \end{aligned}$$

Hence:

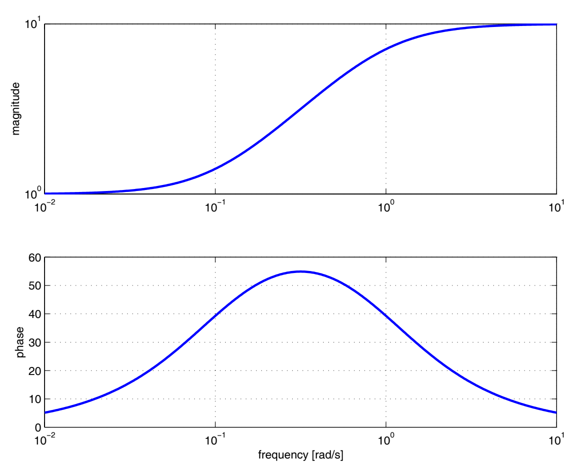
- static gain is K
- Amplitude Bode plot increases at cut-off frequency $\omega = 1/\tau_1$
- Amplitude Bode plot decreases at cut-off frequency $\omega = 1/\tau_2$
- Maximum phase shift at approximately $(1/\tau_1 + 1/\tau_2)/2$

Loopshaping: lead/lag compensation

This is confirmed in the Bode plot for lead/lag compensator

$$C(s) = K\frac{\tau_1 s + 1}{\tau_2 s + 1}$$

with parameters $\tau_1 = 10$, $\tau_2 = 1$, $K = 1$



Loopshaping: lead/lag compensation

In general $\tau_1 = 10\tau_2$, so that

$$C(s) = K\frac{10\tau s + 1}{\tau s + 1}$$

and design freedom is just two parameters: τ and K

How to design of lead/lag compensator? Choose K and τ to put the phase advance where you need it! Usually around the cross over frequency where you need $\angle L > -180$ for stability requirements.

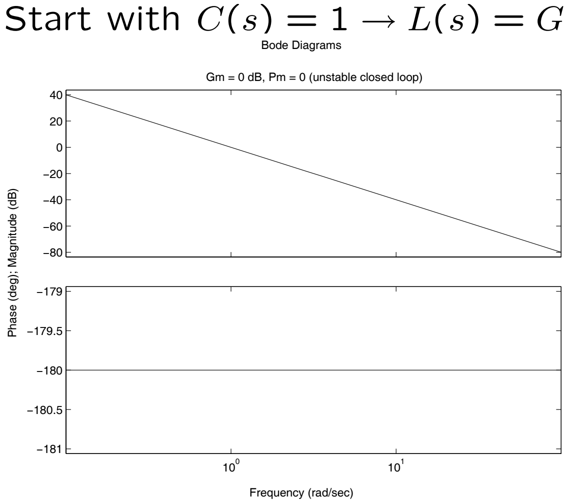
Design can be done in continuous time and converted to discrete time or directly in discrete time using K, a and b parameters.

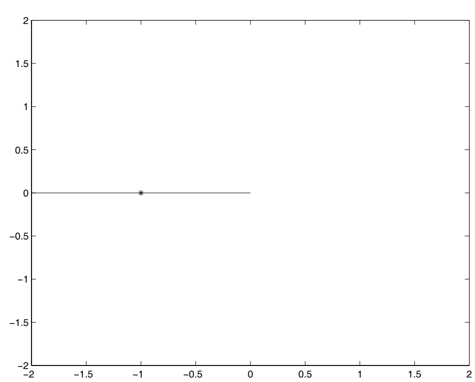
For illustration purposes: design simple lead/lag feedback controller to control the position y of a mass m with a force u (inertial system).

Loopshaping: lead/lag compensation (example)

Model: $m\frac{\partial^2}{\partial t^2}y(t) = u(t)$ or $y(s) = G(s)u(s)$ with $G(s) = 1/(ms^2)$ (let's choose $m = 1$)

Start with $C(s) = 1 \rightarrow L(s) = G(s)$, Bode/Nyquist plot:





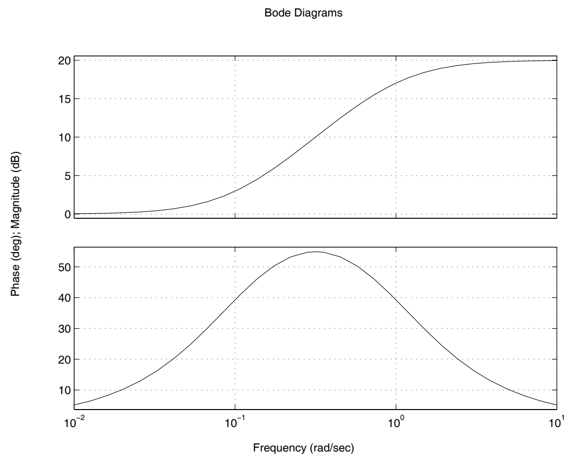
Matlab commands: `G=tf(1,[1 0 0]);L=1*G;margin(L)` and `[re,im]=nyquist(L);plot(re(:),im(:),-1,0,'*');axis([-2 2 -2 2])`

Loopshaping: lead/lag compensation (example)

Lead/lag compensator will increase phase in specified area. Choosing $K = 1$ and $\tau = 1$

$$C(s) = K\frac{10\tau s + 1}{\tau s + 1}$$

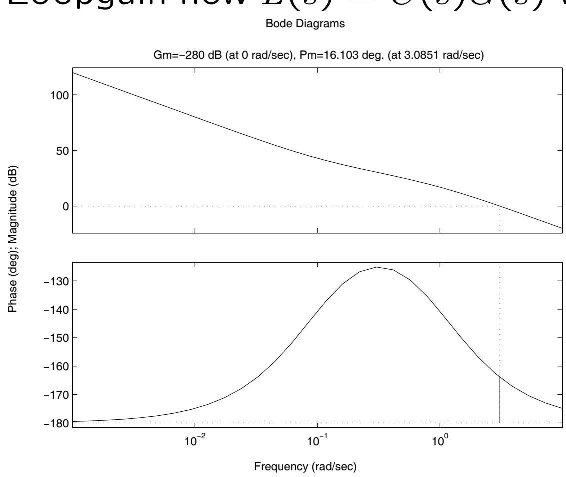
Resulting Bode plot of lead/lag controller $C(s)$

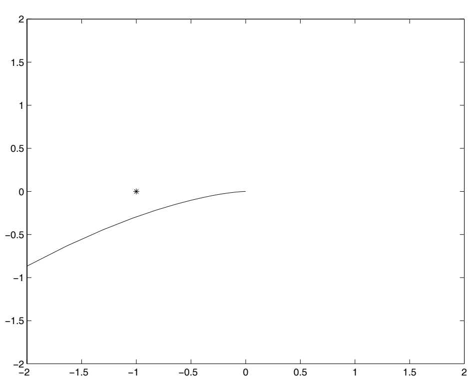


Matlab commands: `C=tf(1*[10 1],[1 1]);bode(C)`

Loopshaping: lead/lag compensation (example)

Loopgain now $L(s) = C(s)G(s)$ with the resulting Bode/Nyquist:





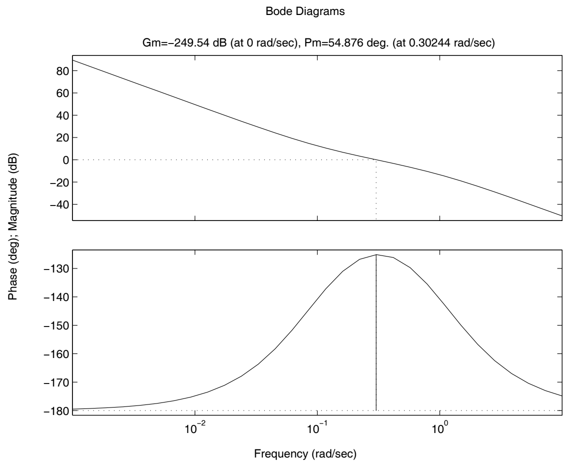
Matlab commands: `L=C*G;margin(L)` and

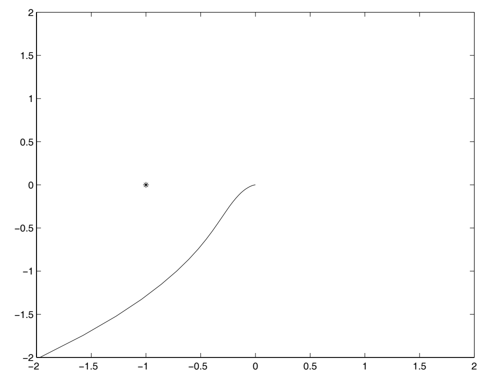
`[re,im]=nyquist(L);plot(re(:),im(:),-1,0,'*');axis([-2 2 -2 2])`

We can still optimize gain K of lead/lag compensator to put maximum phase advancement at frequency where $|L| = 1$

Loopshaping: lead/lag compensation (example)

With choice $\tau = 1$ and $K = 0.03$ we find loopgain $L(s) = C(s)G(s)$ with the resulting Bode/Nyquist:



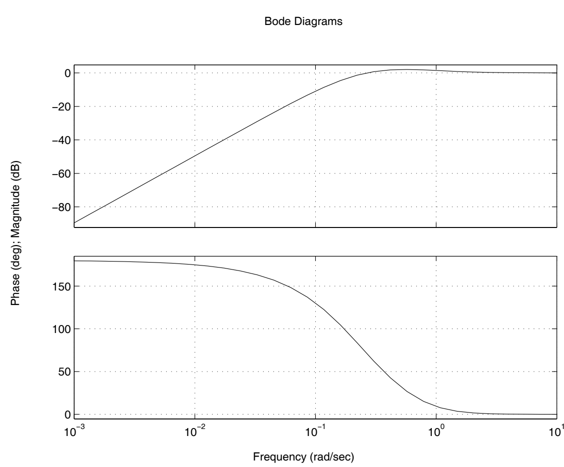


Matlab commands: `C=tf(0.03*[10 1],[1 1]);L=C*G;margin(L)` and `[re,im]=nyquist(L);plot(re(:),im(:),-1,0,'*');axis([-2 2 -2 2])`

Stability: O.K., cross-over frequency $\approx$ 0.2 rad/s

Loopshaping: lead/lag compensation (example)

Other important function to look at: sensitivity function $\frac{1}{1+CG}$ that tells us how much disturbances attenuation we have achieved.



Matlab commands: `S=inv(1+C*G);bode(S)`

Disturbances attenuation until approx. the cross-over frequency!

Final remarks

- Linear I/O dynamical systems can be represented by transfer function models (be careful w.r.t. numerical conditioning)
- Linear Time Invariant Discrete Time (LTIDT) models are modeled by difference equations - a linear relationship between past outputs and past inputs
- Stability of LTIDT models: poles inside unit disk
- Illustration of control design for servo systems: typically lead/lag controllers needed to enable closed-loop stability
- Closed-loop stability can be checked with frequency domain tools such as Nyquist Stability criterion, which can be used for both discrete- and continuous-time systems!
- More complicated control design methods: state-space methods with observer/state feedback and optimal control design methods (H_2- or H_∞- optimization) can also be done on the models presented in this lecture