CDS 101/110a: Lecture 6-1
Transfer Functions

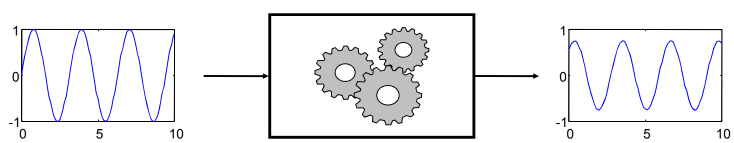
Douglas MacMynowski
1 November 2010

Goals:
- Motivate and define the input/output transfer function of a linear system
- Understand the relationships among frequency response (Bode plot), transfer function, and state-space model
- Introduce block diagram algebra for transfer functions of interconnected systems

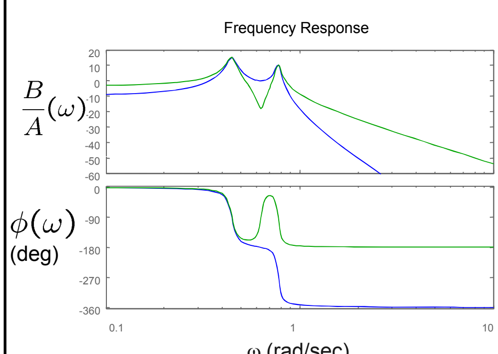
Reading:
- Åström and Murray, *Feedback Systems,* Ch 8
- *Advanced*: Lewis, Chapters 3-4 or DFT, Chapter 2

Frequency Domain Modeling

Defn. The *frequency response* of a linear system is the relationship between the gain and phase of a sinusoidal input and the corresponding steady state (sinusoidal) output.



$u = A\sin(\omega t)$ $\quad$ $y = B\sin(\omega t + \phi)$



Bode plot (1940; Henrik Bode)
- Plot gain and phase vs input frequency
- Gain is plotting using log-log plot
- Phase is plotting with log-linear plot
- Can read off the system response to a sinusoid – in the lab or in simulations
- Linearity ⇒ can construct response to any input (via Fourier decomposition)
- Key idea: do all computations in terms of gain and phase (frequency domain)

Transfer Function and Frequency Response

Exponential response of a linear state space system (from convolution)

$$y = Ce^{At}\left(x(0) - (sI - A)^{-1}B\right) + \left(C(sI - A)^{-1}B + D\right)e^{st}$$

transient   steady state

Transfer function

- Steady state response is proportional to exponential input => look at input/output ratio
- $G(s) = C(sI - A)^{-1}B + D$ is the *transfer function* between input and output

Frequency response

$$u(t) = A \sin \omega t = \frac{A}{2i}\left(e^{i\omega t} - e^{-i\omega t}\right)$$

$$y_{ss}(t) = \frac{A}{2i}\left(G(i\omega)e^{i\omega t} - G(-i\omega)e^{-i\omega t}\right)$$
$$= A \cdot |G(i\omega)| \sin\left(\omega t + \arg G(i\omega)\right)$$

gain   phase

Common transfer functions

$\dot{y} = u$	$\frac{1}{s}$
$y = \dot{u}$	s
$\dot{y} + ay = u$	$\frac{1}{s+a}$
$\ddot{y} = u$	$\frac{1}{s^2}$
$\ddot{y} + 2\zeta\omega_n\dot{y} + \omega_n^2 y = u$	$\frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2}$
$y = k_p u + k_d \dot{u} + k_i \int u$	$k_p + k_d s + \frac{k_i}{s}$
$y(t) = u(t - \tau)$	$e^{-\tau s}$

Poles and Zeros

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

$$H(s) = \frac{n(s)}{d(s)}$$
$$d(s) = \det(sI - A)$$

- Roots of $d(s)$ are called *poles* of $H(s)$
- Roots of $n(s)$ are called *zeros* of $H(s)$

Poles of $H(s)$ determine the stability of the (closed loop) system

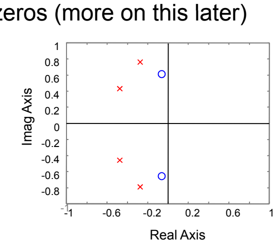
- Denominator of transfer function = characteristic polynomial of state space system
- Provides easy method for computing stability of systems
- Right half plane (RHP) poles (Re > 0) correspond to unstable systems

Zeros of $H(s)$ related to frequency ranges with limited transmission

- A pure imaginary zero at $s=j\omega_z$ blocks any output at that frequency ($G(j\omega_z) = 0$)
- Zeros provide limits on performance, especially RHP zeros (more on this later)

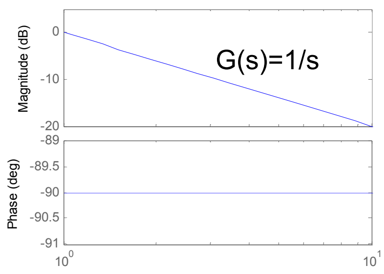
$$H(s) = k\frac{s^2 + b_1 s + b_2}{s^4 + a_1 s^3 + a_2 s^2 + a_3 s + a_4}$$

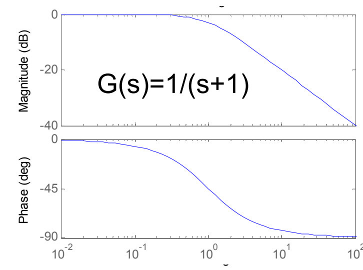
pzmap →



Plotting Bode Plots

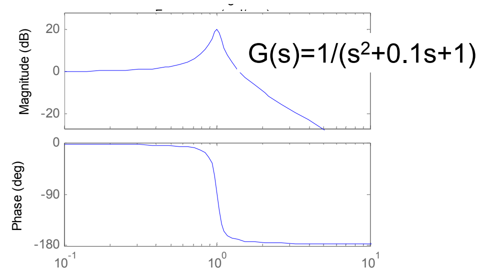
- **Evaluate the transfer function on the imaginary axis**
  - Aside: This is sufficient to characterize the transfer function, follows from analyticity
- **At frequency ω, then $G(i\omega) = r^{i\theta}$**



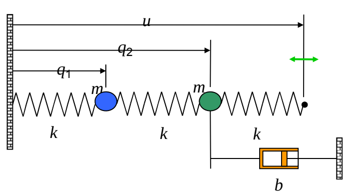


Some useful matlab commands:

```
sys=ss(A,B,C,D);
G=tf(sys);
G=ss2tf(A,B,C,D);
n=[0 0 1];d=[1 0.1 1],G=tf(n,d)
s=tf('s');G=1/(s^2+0.1*s+1);
bode(G)
```



Example: Coupled Masses



$$H_{q_1 f}(s) = k\frac{0.04}{s^4 + 0.08s^3 + 0.8016s^2 + 0.032s + 0.12}$$

$$H_{q_2 f}(s) = k\frac{0.2s^2 + 0.008s + 0.08}{s^4 + 0.08s^3 + 0.8016s^2 + 0.032s + 0.12}$$

Poles (H_{q1f} and H_{q2f})

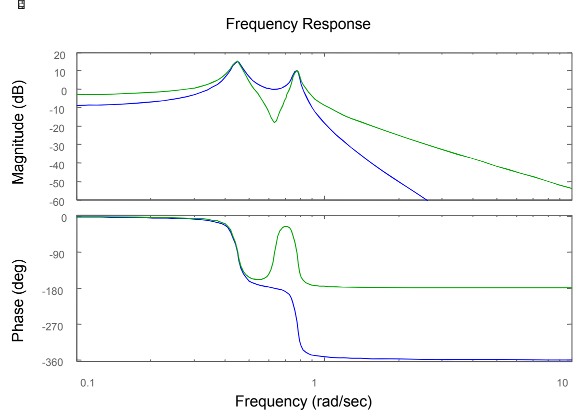
- -0.0200 ± 0.7743j
- -0.0200 ± 0.4468j

Zeros (H_{q2f})

- -0.0200 ± 0.6321j

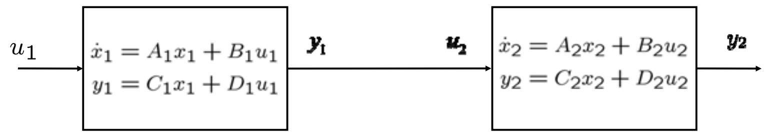
Interpretation

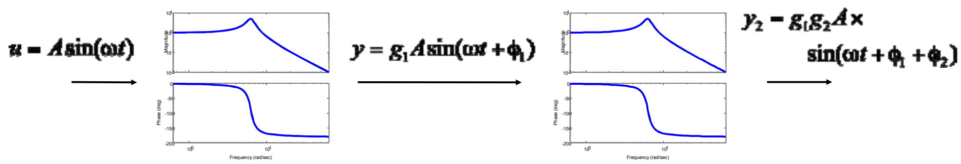
- Zeros in H_{q2f} give low response at $\omega \simeq 0.6321$



Series Interconnections

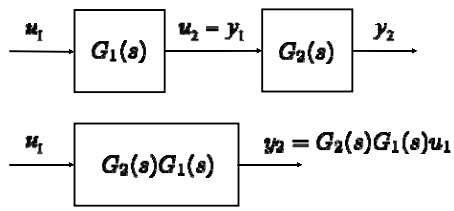
Q: what happens when we connect two systems together *in series?*



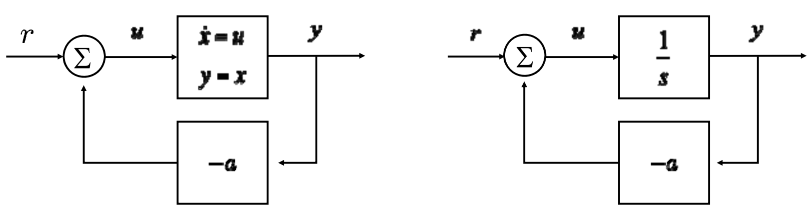


A: Transfer functions *multiply*

- Gains multiply
- Phases add
- Generally: transfer functions well formulated for frequency domain interconnections



Feedback Interconnection



State space derivation

$$\dot{x} = u = r - ay = -ax + r$$
$$y = x$$

Frequency response $r = A\sin(\omega t)$

$$y = \left|\frac{1}{\sqrt{a^2+\omega^2}}\right| \sin\left(\omega t - \tan^{-1}\left(\frac{\omega}{a}\right)\right)$$

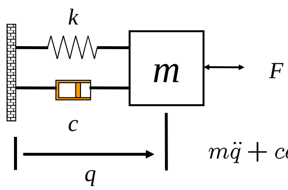
Transfer function derivation

$$y = \frac{u}{s} = \frac{r - ay}{s}$$
$$y = \frac{r}{s+a} = G(s)r$$

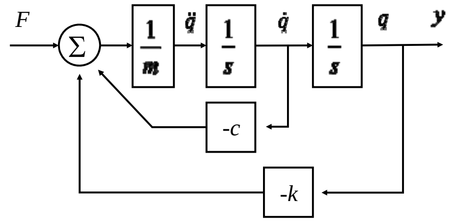
Frequency response

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

Example: mass spring system



$$m\ddot{q} + c\dot{q} + kq = F$$



Rewrite in terms of "block diagram"

- Represent integration using 1/s
- Include spring and damping through feedback terms
- Determine the transfer function through algebraic manipulation
- Claim: resulting transfer function captures the frequency response

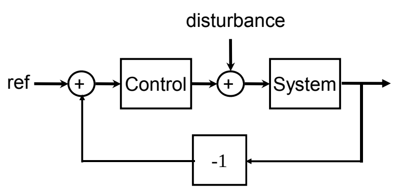
$$y = \frac{1}{m}\cdot\frac{1}{s}\cdot\frac{1}{s}(F - c\dot{q} - kq) = \frac{1}{ms^2}F - \frac{c}{ms}y - \frac{k}{ms^2}y$$

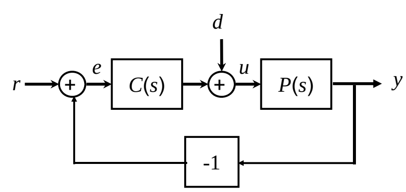
$$\left(1 + \frac{b}{ms} + \frac{k}{ms^2}\right) y = \frac{1}{ms^2}F$$

$$y = \frac{1}{ms^2 + cs + k}F$$

$$\boxed{H(s) = \frac{1}{ms^2 + cs + k}}$$

Control Analysis and Design Using Transfer Functions





Transfer functions provide a method for "block diagram algebra"

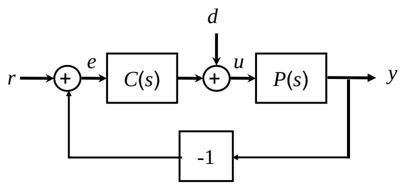
- Easy to compute transfer functions between various inputs and outputs
  - $H_{er}(s)$ is the transfer function between the reference and the error
  - $H_{ed}(s)$ is the transfer function between the disturbance and the error

Transfer functions provide a method for performance specification

- Since transfer functions provide frequency response directly, it is convenient to work in the "frequency domain"
  - $H_{er}(s)$ should be small in the frequency range 0 to 10 Hz (good tracking)

Block Diagram Algebra

Basic idea: treat transfer functions as multiplication, write down equations



$$y = P(s)u$$
$$u = d + C(s)e$$
$$e = r - y$$

Manipulate equations to compute desired signals

$$\begin{aligned} e &= r - y \\ &= r - P(s)u \\ &= r - P(s)(d + C(s)e) \end{aligned}$$

$$(1 + P(s)C(s))e = r - P(s)d$$

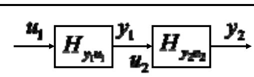
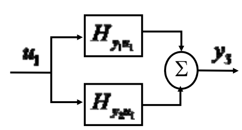
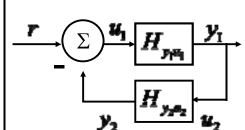
$$e = \underbrace{\frac{1}{1 + P(s)C(s)}}_{H_{er}} r - \underbrace{\frac{P(s)}{1 + P(s)C(s)}}_{H_{ed}} d$$

Note: linearity gives super-position of terms

Algebra works because we are working in frequency domain

- Time domain (ODE) representations are not as easy to work with
- Formally, all of this works because of Laplace transforms

Block Diagram Algebra

Type	Diagram	Transfer function
Series		$H_{y_2u_1} = H_{y_2u_2} H_{y_1u_1} = \frac{n_1 n_2}{d_1 d_2}$
Parallel		$H_{y_3u_1} = H_{y_2u_1} + H_{y_1u_1} = \frac{n_1 d_2 + n_2 d_1}{d_1 d_2}$
Feedback		$H_{y_1r} = \frac{H_{y_1u_1}}{1 + H_{y_1u_1} H_{y_2u_2}} = \frac{n_1 d_2}{n_1 n_2 + d_1 d_2}$

- These are the basic manipulations needed; some others are possible
- Formally, could work all of this out using the original ODEs (nothing *really* new)

MATLAB manipulation of transfer functions

Creating transfer functions

- [num, den] = ss2tf(A, B, C, D)
- sys = tf(num, den)
- num, den = [1 a b] $\rightarrow s^2 + as + b$

```
» tf(sys)
   Transfer function:
           1
   ---------------
   s^2 + 0.2 s + 1
```

Interconnecting blocks

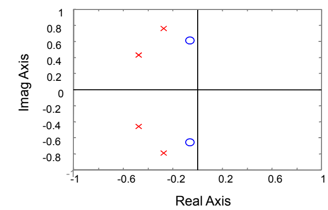
- sys= series(sys1, sys2), parallel, feedback

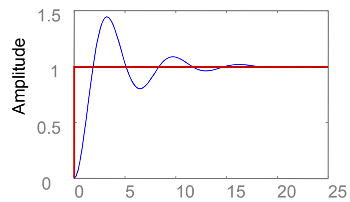
Computing poles and zeros

- pole(sys), zero(sys)
- pzmap(sys)

I/O response

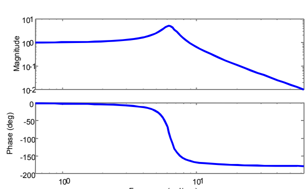
- step(sys), bode(sys)





Summary: Frequency Response & Transfer Functions

$$u = A\sin(\omega t) \rightarrow \begin{array}{c}\dot{x} = Ax + Bu \\ y = Cx + Du \\ x(0) = 0\end{array} \rightarrow y_{ss} = A \cdot |G(i\omega)| \times \sin(\omega t + \arg G(i\omega))$$



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

$$G_{y_2u_1} = G_{y_2u_2}G_{y_1u_1} = \frac{n_1n_2}{d_1d_2}$$