Time Domain vs Frequency Domain

Typical time-domain representation of I/O systems:

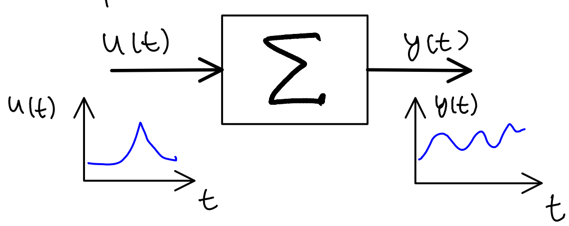
* Linear differential systems: $a_n \frac{d^n y}{dt^n} + \dots + a_0 y = b_m \frac{d^m u}{dt^m} + \dots + b_0 u$

  Example: Mass-spring-friction system (previous lecture)
  RLC circuits, linear filters, etc

  n is called the order of the system. It is related to:

- the number of energy bearing variables
- the complexity of the system

* Delay systems: $y(t) = u(t-\tau)$, $\tau > 0$
* Combination of both.



I/O LTI systems are linear transformations between function spaces.
Think of signals as vectors.

Frequency Domain

There is a linear transformation between time domain representation of the signals and the frequency domain:

Laplace transform: $U(s) = \mathcal{L}\{u(t)\} = \int_0^\infty u(t)e^{-st}\,dt$, $s \in \mathbb{C}$

and its inverse: $u(t) = \mathcal{L}^{-1}\{U(s)\} = \frac{1}{2\pi j}\int_{-\infty}^{\infty} U(s)\,e^{st}\,ds$

$$\text{time domain} \underset{\mathcal{L}^{-1}}{\overset{\mathcal{L}}{\rightleftarrows}} \text{Freq. domain}$$

$$\begin{tikzcd} u(t) \arrow[r, "\Sigma"] \arrow[d] & y(t) \arrow[d] \\ U(s) \arrow[r, "G(s)"] & Y(s) \end{tikzcd}$$

Transfer function is the equivalence of Σ in frequency domain.

$Y(s) = G(s)\,U(s)$, and if u is scalar : $\dfrac{Y(s)}{U(s)} = G(s)$

Some important properties of the transform:

If $u(t) \xrightarrow{\mathcal{L}} U(s)$, then : $\displaystyle\int_0^\infty \frac{du}{dt} e^{-st}\, dt = \underbrace{u e^{-st} \Big|_0^\infty}_{\substack{\shortparallel \\ 0\text{, since the transform exists}}} + \int s\, u(t) e^{-st}\, dt$

$= s\,U(s)$

$\displaystyle\int_0^\infty u(t-\tau)\, e^{-st}\, dt \Rightarrow$ Change variable $t - \tau \triangleq r$

$\shortparallel$

$$\int_{-\tau}^\infty u(r)\, e^{-s(r+\tau)}\, dr = e^{-s\tau} \int_0^\infty u(r)\, e^{-sr}\, dr = e^{-s\tau}\, U(s)$$

Thus: $a_n \dfrac{d^n y}{dt^n} + \dots + a_0 y = b_m \dfrac{d^m u}{dt^m} + \dots + b_0 u$ $\leftarrow$ time domain

$\Downarrow \mathcal{L}$

$$a_n s^n Y(s) + \dots + a_0 Y(s) = b_m s^m U(s) + \dots + b_0 U(s)$$

$$\frac{Y(s)}{U(s)} = \frac{b_m s^m + \dots + b_0}{a_n s^n + \dots + a_0} = \frac{B(s)}{A(s)}$$ $\leftarrow$ freq. domain

$m < n \rightarrow$ proper transfer function

$m > n \rightarrow$ improper transfer function

$m = n \rightarrow$ biproper transfer function

$A(s) = a_n s^n + \cdots + a_0$ is called the characteristic polynomial

The roots of $A(s)$ are called the poles of the transfer function.

The roots of $B(s)$ are called the zeros

zeros

Alternative representation:

$$\frac{Y(s)}{U(s)} = \frac{(s-z_1)^{m_1} \cdot (s-z_2)^{m_2} \cdots}{(s-p_1)^{n_1} \cdot (s-p_2)^{n_2} \cdots}$$

poles

The poles determine the stability of the system: they must have negative real parts for the system to be stable.

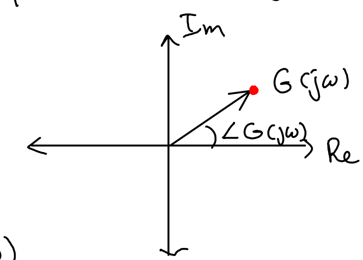
Frequency Response

The frequency domain representation is very convenient for analyzing the spectral properties of the signals and systems

If $u(t)$ is sinusoidal : $u(t) = u_0 \sin(\omega t + \alpha)$, then $y(t)$ is also sinusoidal $y(t) = y_0 \sin(\omega t + \beta)$, $t \in (-\infty, \infty)$

Properties: $\dfrac{y_0}{u_0} = |G(j\omega)|$, and $(\beta - \alpha) = \angle G(j\omega)$

Recall that $G(j\omega)$ is a complex number:



Bode plot: The plot of $|G(j\omega)|$ and $\angle G(j\omega)$ versus ω (in log scale)

The plot of $|G(j\omega)|$ is typically given in dB scale

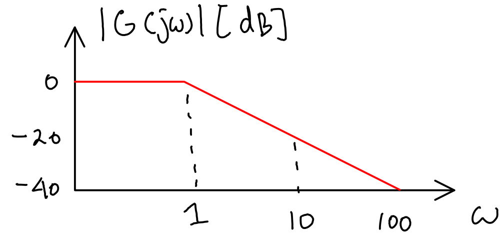
$$|G(j\omega)|[dB] = 20 \log |G(j\omega)|$$

- Use MATLAB command 'bode', or
- use linear approximation (read textbook p.19 – p.20)

Example: $G(s) = \dfrac{1}{s+1} \to G(j\omega) = \dfrac{1}{1+j\omega}$

$$|G(j\omega)| = \sqrt{\frac{1}{1+\omega^2}} \to \text{if } \begin{array}{l} \omega \ll 1, \ |G(j\omega)| \approx 1 \\ \omega \gg 1, \ |G(j\omega)| \approx \frac{1}{\omega} \end{array}$$

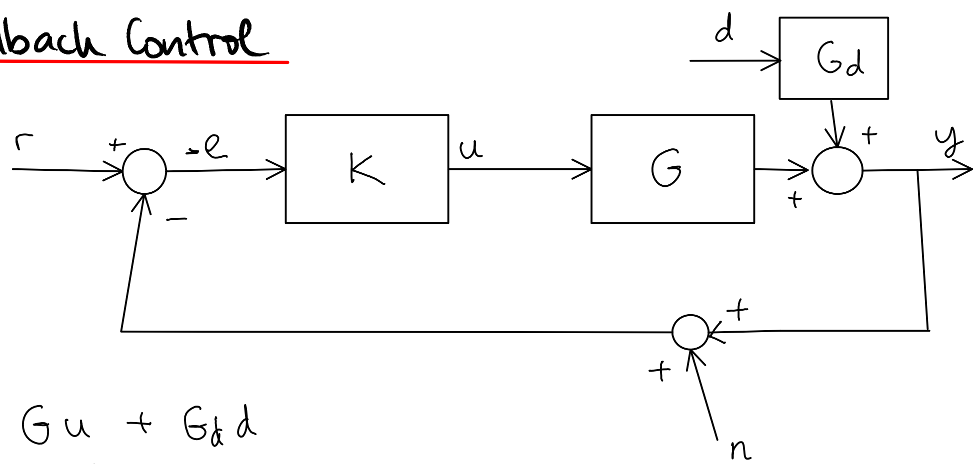
Linear approximation:



$$\angle G(j\omega) = -\angle(1+j\omega) = \begin{array}{l} 0 \text{ if } \omega \ll 1 \\ -90^\circ \text{ if } \omega \gg 1 \end{array}$$

In case of multiple zeros and poles, the corresponding Bode plot is the superposition of the individual plots.

Feedback Control



$y = Gu + G_d d$

$u = K(r - y - n)$

$y = GKr - GKy - GKn + G_d d$

$(I+GK)y = GKr - GKn + G_d d$

$y = (I+GK)^{-1}GKr - (I+GK)^{-1}GKn + (I+GK)^{-1}G_d d$

Definitions: $L \triangleq GK \rightarrow$ loop transfer function

$S \triangleq (I+GK)^{-1} \rightarrow$ sensitivity function

$T \triangleq (I+GK)^{-1}GK \rightarrow$ complementary sensitivity function $\rightarrow$ closed loop transfer function

Notice that: $S+T = (I+GK)^{-1}(I+GK) = I$

Consider SISO case: $T = \dfrac{GK}{1+GK} \rightarrow dT = \dfrac{(1+GK)K - GK \cdot K}{(1+GK)^2} dG$

$$\frac{dT}{T} = \frac{K}{(1+GK)^2} \cdot \frac{1+GK}{GK} dG = \frac{1}{(1+GK)G} \cdot dG \text{ , thus: } \frac{\frac{dT}{T}}{\frac{dG}{G}} = \frac{1}{1+GK} = S$$

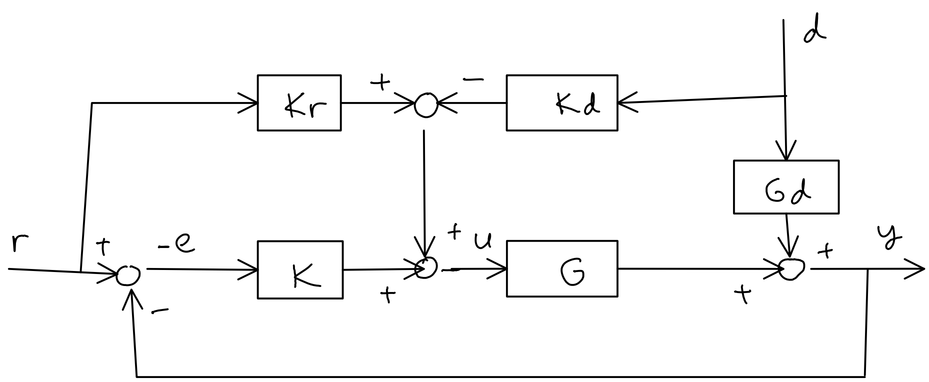
Hence sensitivity function!

Assume perfect measurement: $n \equiv 0$; then:

$e = y - r = (Tr + SG_d d) - r = -Sr + SG_d d$

ideally S is small for good tracking

Two degrees of freedom and feedforward configuration



$y = Gu + G_d d$

$u = K(r-y) + K_r r - K_d d$

$$y = (I + GK)^{-1}\left[G(K + K_r)r + (G_d - GK_d)d\right]$$

$$= (T + SGK_r)r + S(G_d - GK_d)d$$

$$e = -S(I - GK_r)r + S(G_d - GK_d)d = -SS_r r + SS_d G_d d$$

$$S_r \triangleq I - GK_r \quad ; \quad S_d \triangleq (I - GK_d G_d^{-1})$$

Extra DOF helps designing controllers for better tracking.

Perfect feedforward control: $K_r = G^{-1}$; $K_d = G^{-1}G_d$

requires inverse plant model $\Rightarrow$ not feasible