Lin Alg Review

column space: span of matrix columns — subspace reached by cols of A
$A\vec{x}=\vec{b}$

(lin indep) = (trivial nullspace) = ($\det(A)\neq 0$)

(lin dep) = (nontrivial nullspace) = ($\det(A)=0$)

Nullspace: values of $\vec{x}$ that satisfy $A\vec{x}=\vec{0}$
$A\vec{x}=\vec{0} = [A][\vec{x}]=\begin{bmatrix}0\\ \vdots\\ 0\end{bmatrix} \rightarrow \left[A \,\middle|\, \begin{matrix}0\\ \vdots\\ 0\end{matrix}\right]$

rank-nullity: # of cols = dim(col(A)) + dim(null A)

eigenstuff: $A\vec{v}=\lambda\vec{v}$

$(A-\lambda I)\vec{x}=0$

solving for nullspace of $(A-\lambda I)$

Plug eigenvalues into $(A-\lambda I)\vec{x}=0$ & solve for $\vec{x}$

$$\begin{bmatrix}1 & 1\\ -2 & -3+1\end{bmatrix}\vec{x}=0$$
$$=\left[\begin{matrix}1 & 1\\ -2 & -2\end{matrix}\,\middle|\,\begin{matrix}0\\ 0\end{matrix}\right]$$
$$=\left[\begin{matrix}1 & 1\\ 0 & 0\end{matrix}\,\middle|\,\begin{matrix}0\\ 0\end{matrix}\right]$$
$$\vec{v}_1=\begin{bmatrix}-1\\ 1\end{bmatrix}$$

Circuits Review

NVA:
1. Label elements (nodes, currents, voltages)
2. Write KCL eqs
3. Write elements in terms of nodes
4. Replace currents & solve

Voltage divider:
$$V_x=\frac{R_2}{R_2+R_1}V_s$$

Resistors: $V=IR$
series: R_1+R_2
parallel: $\frac{R_1R_2}{R_1+R_2}$

capacitors: $Q=CV_c$
$\frac{d}{dt}Q=\frac{d}{dt}CV_c$
$\boxed{I_c=C\frac{dV_c}{dt}}$
series: $\frac{C_1C_2}{C_1+C_2}$
parallel: C_1+C_2
$C=\frac{\varepsilon_0 A}{d}$

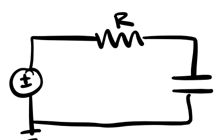
op-amps
1. $i^+=i^-=0\,A$
2. $V^+=V^-$

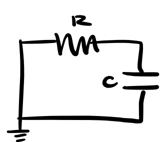
Neg feedback
1. $\uparrow V_{out} \rightarrow \downarrow V_{out}$
2. $\downarrow V_{out} \rightarrow \uparrow V_{out}$

Power: $p(t)=i(t)v(t)=Cv(t)\frac{dv(t)}{dt}$

Energy: $w(t)=C\left[\frac{v}{2}\right]\Big|_0^{v(t)}=\frac{1}{2}Cv(t)^2$
$=\frac{q^2(t)}{2}$

RC circuits


$t<0$


$t\geq 0$

$i_R(t)=i_c(t)$
$\frac{V_R(t)}{R}=C\frac{dV_c(t)}{dt}$
$\frac{0-V_c(t)}{R}=C\frac{dV_c(t)}{dt}$

$$\frac{dV_c}{dt}=-\frac{1}{RC}V_c(t)$$

*pattern match
$x(t)=Ae^{bt}$
$\frac{d}{dt}(Ae^{bt})=Abe^{bt}$
$=bV_c(t)$
$=-\frac{1}{RC}V_c(t)$

*initial cond
$V_c(0)=Ae^{-\frac{1}{RC}0}$
$=A\cdot 1$
$V_c(t)=V_se^{-\frac{1}{RC}t}$

solving diff eq

$\frac{d}{dt} x(t) + ax(t) = b(t) \qquad x(0) = x_0$

Homogeneous w/ $b(t) = 0$

$\frac{d}{dt} x(t) + ax(t) = 0$

$\frac{d}{dt} x(t) = -ax(t)$

Guess: $x(t) = Ae^{bt}$

Plug into diff eq:

$\frac{d}{dt}(Ae^{bt}) - Abe^{bt} = bx(t) \rightarrow b = -a$

initial condn: $x(t) = Ae^{-at}$

$x(0) = Ae^{0} = A$

$x(t) = x(0)e^{-at}$

Given diff eq in form

$\frac{d}{dt} x(t) + ax(t) = b(t)$

solution for $x(t)$ is

$x(t) = x_0 e^{-at} + e^{-at} \int_0^t e^{at'} b(t') dt'$

From discussion:

$\frac{dV_c(t)}{dt} = \lambda V_c(t) + U(t) = -\frac{1}{RC} V_c(t) + \frac{V_{in}(t)}{RC}$

→ homogeneous

$V_h(t) = Ae^{bt} = Ae^{\lambda t}$
$= Ae^{(-1/RC)t}$

→ particular soln * look @ $\underbrace{\text{steady state!}}$

inductor = short
capacitor = open

$V_p(t) = V_{in}(t)$

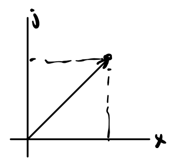
$V_c(t) = Ae^{(-1/RC)t} + V_{in} t$

→ use initial condition to solve for A

complex numbers

Euler's identity $e^{j\theta} = \cos\theta + j\sin\theta$

$\sin\theta = \frac{e^{j\theta} - e^{-j\theta}}{2j} \qquad \cos\theta = \frac{e^{j\theta} + e^{-j\theta}}{2}$



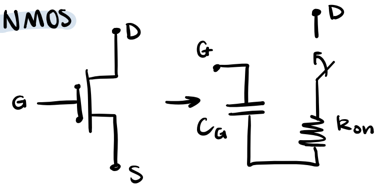
Magnitude: $|a| = \sqrt{x^2 + y^2}$
Phase: $\arctan\left(\frac{y}{x}\right)$

$x = r\cos\theta, \; y = r\sin\theta$

Transistors

acts like a switch or amplifier

NMOS

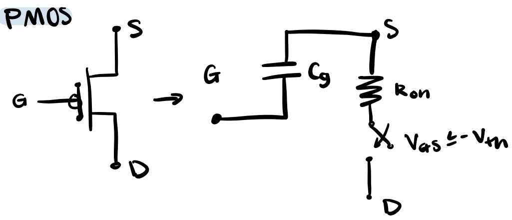


condition for switch to close:

$V_{GS} = V_{gate} - V_{source} \geq V_{tn}$

high gate allows current to flow between drain (D) & source (S)

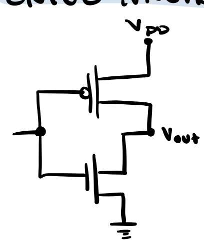
PMOS



$V_{GS} = V_{gate} - V_{source} \leq -|V_{tn}|$

circle means PMOS

CMOS inverter



Behaves like NOT gate

oscillator w/ odd prime # of inverters in loop

inductors : deals w/ magnetic field effects

$$\boxed{V_L(t) = L\frac{di(t)}{dt}}$$

① inductor current can't change immediately

② $t=0 \rightarrow$ open circuit

$t=\infty \rightarrow$ short circuit

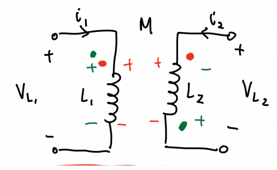
Energy: $p(t) = v(t)i(t) = Li(t)\frac{di(t)}{dt}$

$w(t) = \frac{1}{2}Li^2$

Mutual inductance: one coil induces voltage in another inductor

series: $L_1 + L_2$ parallel $= \frac{L_1L_2}{L_1+L_2}$

total = self + mutual



$$\boxed{\begin{aligned} V_{L1} &= L_1\frac{di_1}{dt} + M\frac{di_2}{dt} \\ V_{L2} &= M\frac{di_1}{dt} + L_2\frac{di_2}{dt}\end{aligned}}$$

$$\boxed{\begin{aligned} V_{L1} &= L_1\frac{di_1}{dt} - M\frac{di_2}{dt} \\ V_{L2} &= -M\frac{di_1}{dt} + L_2\frac{di_2}{dt}\end{aligned}}$$

DC steady state

$i_C(t) = C\frac{dV_C(t)}{dt}$ $V_L(t) = L\frac{di_L(t)}{dt}$

$e^{-at} \rightarrow 0$ decays

constant voltage across $C \rightarrow \frac{dv_C(t)}{dt} = 0$

↳ no current = open

constant current across $L \rightarrow \frac{di_L(t)}{dt} = 0$

↳ no voltage = short

second order diff eq

$$\frac{d^2x(t)}{dt^2} + 2\alpha\frac{dx(t)}{dt} + w_0^2x(t) = f(t)$$

$s_1 = -\alpha + \sqrt{\alpha^2 - w_0^2}$

$s_2 = -\alpha - \sqrt{\alpha^2 - w_0^2}$

$\frac{\alpha}{w_0}$ = damping ratio

α = damping coefficient

w_0 = undamped resonant frequency

① overdamped $\left(\frac{\alpha}{w_0} > 1\right)$

$\frac{a^2}{w_0} - 1 \rightarrow$ real & nonzero

$x_n(t) = K_1e^{s_1t} + K_2e^{s_2t}$

② critically damped $\left(\frac{\alpha}{w_0} = 1\right)$

$\frac{\alpha^2}{w_0} - 1 = 0 \rightarrow$ real & equal

$x_n(t) = K_1e^{s_1t} + K_2te^{s_1t}$

③ underdamped $\left(\frac{\alpha}{w_0} < 1\right)$

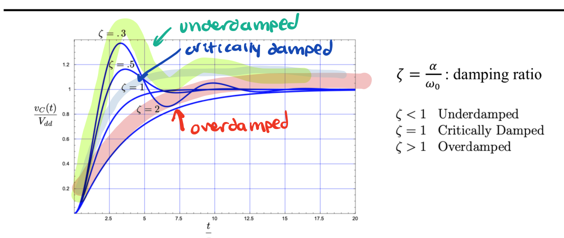
$\frac{\alpha^2}{w_0} - 1 < 0 \rightarrow$ complex & distinct

$x_n(t) = K_1e^{-\alpha t}\cos(\omega_n t) + K_2e^{-\alpha t}\sin(\omega_n t)$

$\omega_n = \sqrt{\omega_0^2 - \alpha^2}$

- KCL: Σ current in $= \Sigma$ current out
- KVL: $\sum_{loop}$ voltage $= 0$
- Capacitors: $i_c(t) = C\frac{dv_c(t)}{dt}$ $V_c(t) = \frac{1}{C}\int_0^t i_c(t)\,dt' + V_c(0)$
- Inductors: $V_L(t) = L\frac{di_L(t)}{dt}$ $i_L(t) = \frac{1}{L}\int_0^t V_L(t')\,dt' + i_L(0)$
- Resistors: $V_R = IR$
- DC steady state (constant voltage/current)
  - capacitors → open circuit b/c $\frac{dv_c(t)}{dt} = 0$
  - inductors → short circuit b/c $\frac{di_L(t)}{dt} = 0$
- parallel/series equivalence for R, L, C
- Thevenin/Norton equivalence

Under vs Over Damped



- Overdamped solutions don't oscillate.

Vector Diff eq

Change of basis:

$$\vec{x} = I\vec{x} = \begin{bmatrix}1&0\\0&1\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix} = \begin{bmatrix}1\\0\end{bmatrix}x_1 + \begin{bmatrix}0\\1\end{bmatrix}x_2$$

$$V\vec{y} = \begin{bmatrix}V_{11}&V_{21}\\V_{21}&V_{22}\end{bmatrix}\begin{bmatrix}y_1\\y_2\end{bmatrix} = \begin{bmatrix}V_{11}\\V_{21}\end{bmatrix}y_1 + \begin{bmatrix}V_{21}\\V_{22}\end{bmatrix}y_2 \rightarrow \text{cols of } V = \text{basis vectors}$$

Diagonalization

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t)$$

$$\frac{d}{dt}\left(V\tilde{x}(t)\right) = A\left(V\tilde{x}(t)\right)$$

$$V\frac{d}{dt}\tilde{x}(t) = AV\tilde{x}(t)$$

$$V^{-1}\left(V\frac{d}{dt}\tilde{x}(t)\right) = V^{-1}\left(AV\tilde{x}(t)\right)$$

$$\frac{d}{dt}\tilde{x}(t) = \underbrace{V^{-1}AV}_{\Lambda}\tilde{x}(t)$$

$$\Lambda = \begin{bmatrix}\lambda_1&0\\0&\lambda_2\end{bmatrix}$$

$$\begin{tikzcd}
\frac{d}{dt}\vec{x}(t)=A\vec{x}(t) \arrow[r, squiggly, "\text{difficult}"] \arrow[d, "\text{change of basis}"'] & \vec{x}(t)= \\
\frac{d}{dt}\tilde{x}(t)=\Lambda\tilde{x}(t) \arrow[r, "\text{solve diagonal system}"] & \tilde{x}(t)= \arrow[u]
\end{tikzcd}$$

change of basis
① $\vec{x}(t) = V\tilde{x}(t)$
② $\tilde{x}(t) = V^{-1}\vec{x}(t)$

original coordinates

nice coordinates

diagonal

$$\frac{d}{dt}\begin{bmatrix}\tilde{x}_1(t)\\\tilde{x}_2(t)\end{bmatrix} = \begin{bmatrix}\lambda_1&0\\0&\lambda_2\end{bmatrix}\begin{bmatrix}\tilde{x}_1 t\\\tilde{x}_2 t\end{bmatrix} \rightarrow \begin{bmatrix}\frac{d}{dt}\tilde{x}_1(t)\\\frac{d}{dt}\tilde{x}_2(t)\end{bmatrix} = \begin{bmatrix}\lambda_1\tilde{x}_1(t)\\\lambda_2\tilde{x}_2(t)\end{bmatrix} \rightarrow \begin{matrix}\tilde{x}_1(t) = \tilde{x}_1(0)e^{\lambda_1 t}\\\tilde{x}_2(t) = \tilde{x}_2(0)e^{\lambda_2 t}\end{matrix}$$

$$\tilde{x}(t) = \begin{bmatrix}\tilde{x}_1(t)\\\tilde{x}_2(t)\end{bmatrix} = \begin{bmatrix}\tilde{x}_1(0)e^{\lambda_1 t}\\\tilde{x}_2(0)e^{\lambda_2 t}\end{bmatrix} \xrightarrow[\vec{x}(t)=V\tilde{x}(t)]{\text{change of basis}} \vec{x}(t) = V\tilde{x}(t) = \begin{bmatrix}V_{11}&V_{12}\\V_{21}&V_{22}\end{bmatrix}\begin{bmatrix}\tilde{x}_1(0)e^{\lambda_1 t}\\\tilde{x}_2(0)e^{\lambda_2 t}\end{bmatrix}$$

$$= \begin{bmatrix}V_{11}\tilde{x}_1(0)e^{\lambda_1 t} + V_{12}\tilde{x}_2(0)e^{\lambda_2 t}\\V_{21}\tilde{x}_1(0)e^{\lambda_1 t} + V_{22}\tilde{x}_2(0)e^{\lambda_2 t}\end{bmatrix}$$

solving vector diff eq

① Set up scalar diff eq & initial cond

② Write system in matrix form

$$\frac{d}{dt}\vec{x}(t) = \frac{d}{dt}\begin{bmatrix}x_1(t)\\x_2(t)\end{bmatrix} = \begin{bmatrix}\frac{d}{dt}x_1(t)\\\frac{d}{dt}x_2(t)\end{bmatrix}$$

③ Calculate eigenvalues & eigenvectors

④ Define change of variables

$V = [\vec{v}_1 \cdots \vec{v}_n]$ *using eigenbasis

$\vec{x}(t) = V\tilde{x}(t)$ $\quad \tilde{x}(t) = V^{-1}\vec{x}(t)$

⑤ Apply change of variables

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + \vec{b}$$

$$\frac{d}{dt}\left(V\tilde{x}(t)\right) = A\left(V\tilde{x}(t)\right) + \vec{b}$$

$$V\frac{d}{dt}\tilde{x}(t) = AV\tilde{x}(t) + \vec{b}$$

$$V^{-1}V\frac{d}{dt}\tilde{x}(t) = V^{-1}\left(AV\tilde{x}(t) + \vec{b}\right)$$

$$\frac{d}{dt}\tilde{x}(t) = \underbrace{V^{-1}AV}_{\tilde{A}}\tilde{x}(t) + \underbrace{V^{-1}\vec{b}}_{\tilde{b}}$$

⑥ Convert init into eigenbasis

$$\vec{x}(t) = V\tilde{x}(t) \rightarrow \tilde{x}(t) = V^{-1}\vec{x}(t)$$

$$\tilde{x}(0) = V^{-1}\vec{x}(0)$$

⑦

$$\frac{d}{dt}\begin{bmatrix}\tilde{x}_1(t)\\\vdots\\\tilde{x}_n(t)\end{bmatrix} = \begin{bmatrix}\lambda_1&&0\\&\ddots&\\0&&\lambda_n\end{bmatrix}\begin{bmatrix}\tilde{x}_1(t)\\\vdots\\\tilde{x}_n(t)\end{bmatrix} + \begin{bmatrix}\tilde{b}_1\\\vdots\\\tilde{b}_n\end{bmatrix}$$

⑧ $\vec{x}(t) = V\tilde{x}(t)$
convert back to standard basis

Diagonalized: $\Lambda = V^{-1}AV$, $A = V\Lambda V^{-1}$

- A must be square
- V must be invertible
  - ↳ lin indep eigenvalues

Phasors

impedance: $Z = \frac{\hat{V}}{\hat{I}}$ $\frac{\text{voltage phasor}}{\text{current phasor}}$

$$\boxed{Z_R = R \quad Z_C = \frac{1}{j\omega C} \quad Z_L = j\omega L}$$

*impedances like resistors in phasor domain

$\omega = 0$ (DC input)

→ $Z_C = \infty$ capacitor = open

→ $Z_L = 0$ inductor = short

$\omega = \infty$ (high freq)

→ $Z_C = 0$ capacitor = short

→ $Z_L = \infty$ inductor = open

$$\boxed{\begin{aligned} V_0 \cos(\omega t + \phi) &= V_0 e^{j\phi} \\ V_0 \sin(\omega t + \phi) &= \frac{V_0 e^{j\phi}}{j} \end{aligned}}$$

V_0 = amplitude

ϕ = phase shift

① sinusoidal sources

↓

phasor representation

② Replace impedances

③ NVA

④ convert back to sinusoidal

Transfer Functions

$\tilde{V}_{in}(j\omega) \rightarrow \boxed{H(j\omega)} \rightarrow \tilde{V}_{out}(j\omega)$

$$\boxed{H(j\omega) = \frac{\tilde{V}_{out}(j\omega)}{\tilde{V}_{in}(j\omega)} = |H(j\omega)| e^{j\angle H(j\omega)}}$$

magnitude $|H(j\omega)|$

phase $\angle H(j\omega) = \angle \text{numerator} - \angle \text{denom}$

$V_{out}(t) = |H(j\omega)| \cdot V_i \cos(j\omega + \phi + \angle H(j\omega))$

cutoff freq (ω_c): $|H(j\omega_c)| = \frac{|H(j\omega)|_{max}}{\sqrt{2}}$

Poles: $\frac{1}{\omega_p + j\omega}$ $\quad Z = \frac{1}{\omega_p + j\omega}$

↑

pole frequency

magnitude:

$|Z| = \frac{|1|}{|\omega_p + j\omega|}$

$= \frac{1}{\sqrt{\omega_p^2 + \omega^2}}$

phase:

$\angle Z = \frac{\angle 1}{\angle \omega_p + j\omega}$

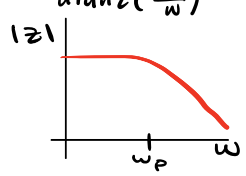
$\angle$phase = $\angle$numerator − $\angle$denom

$= 0 - \text{atan2}\left(\frac{\omega_p}{\omega}\right)$

$= -\text{atan2}\left(\frac{\omega_p}{\omega}\right)$

$\omega < \omega_p \rightarrow |Z| \approx \frac{1}{\omega_p}$

$\omega > \omega_p \rightarrow |Z| \approx \frac{1}{\omega}$



zeros: $\omega_z + j\omega$

↑

zero frequency

magnitude of zero:

$Z = \omega_z + j\omega$

$|Z| = \sqrt{\omega_z^2 + \omega^2}$

phase of zero

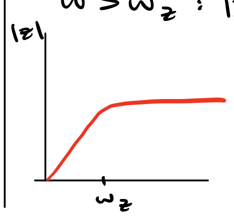
$Z = \omega_z + j\omega$

$\angle Z = \text{atan2}(\omega, \omega_z)$

$= \tan^{-1}\left(\frac{\omega}{\omega_z}\right)$

$\omega < \omega_z$: $|Z| \approx \omega_z$

$\omega > \omega_z$: $|Z| \approx \omega$ (goes to 0 as $\omega \to 0$)



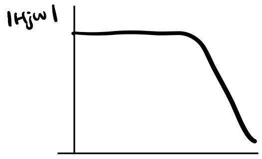
Filters

Lowpass: $H_{LP}(j\omega) = \frac{1}{1 + \frac{j\omega}{\omega_c}}$

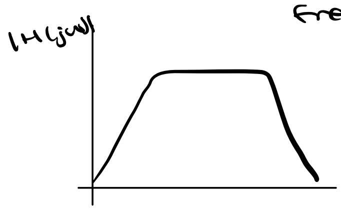
↓

remove high frequency
allow low frequency

$\omega \to 0 \quad |H(j\omega)| \to 1$
$\to \infty \quad \to 0$



Bandpass: allows band of frequencies thru

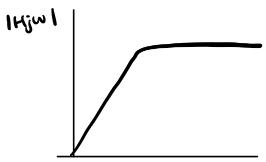


Highpass: $H_{HP}(j\omega) = \frac{\frac{j\omega}{\omega_c}}{1 + \frac{j\omega}{\omega_c}}$

↓

remove low frequency
allow high frequency

$\omega \to 0 \quad |H(j\omega)| \to 0$
$\to \infty \quad \to 1$



$H_{BP}(j\omega) = H_{LP}(j\omega) \cdot H_{HP}(j\omega)$

$\omega = \frac{1}{RC}$ angular cutoff

$f_c = \frac{1}{2\pi RC}$ cutoff frequency

Bode Plots

$|H(j\omega)|_{dB} = 20\log_{10}(|H(j\omega)|)$

$\log|H(j\omega)| = \log|H_1(j\omega)| + \log|H_2(j\omega)|$
$\angle H(j\omega) = \angle H_1(j\omega) + \angle H_2(j\omega)$

rational transfer fn: $H(j\omega) = K \cdot \frac{N(j\omega)}{D(j\omega)}$

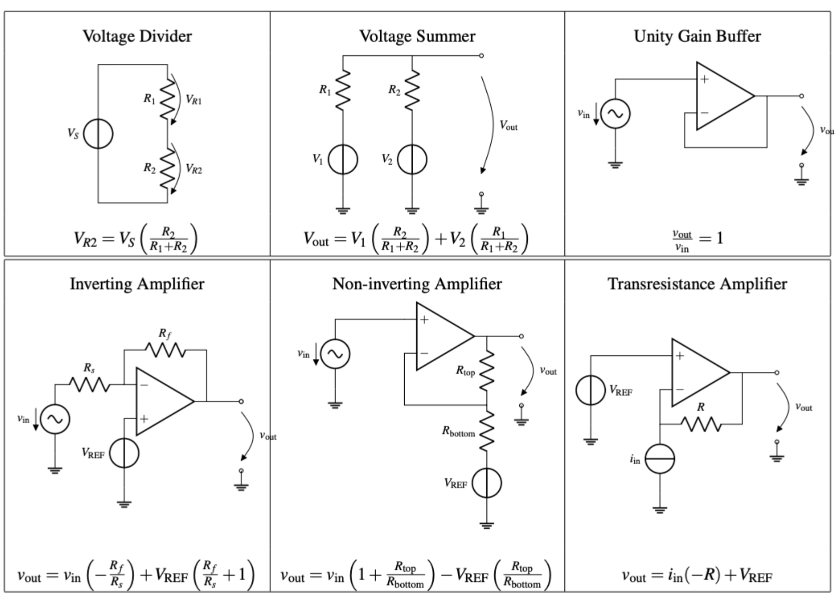
Drops w/ slope of 1 after pole
rises w/ slope of 1 after zero

* look @ asymptotic behavior

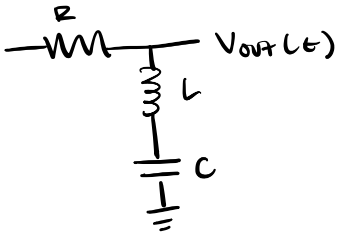
$\underbrace{\begin{matrix}\omega < \omega_p \\ \omega > \omega_p\end{matrix}}_{\text{magnitude}} \qquad \underbrace{\begin{matrix}\omega < \frac{\omega_p}{10} \\ \omega = \omega_p \\ \omega > 10\omega_p\end{matrix}}_{\text{phase}}$

pole @ origin → phase = −90, magnitude $\frac{1}{j\omega}$
zero @ origin → phase = 90, magnitude $j\omega$

positive constants = 0°
negative constants = −180



Notch Filter

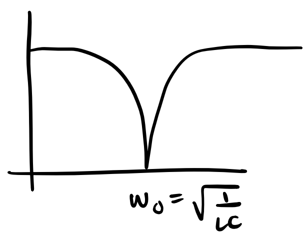


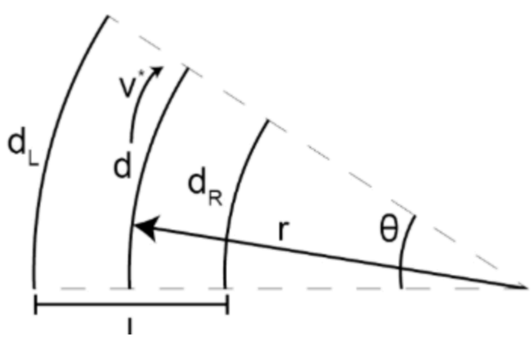
$$H(j\omega) = \frac{\tilde{V}_{out}}{\tilde{V}_{in}} = \frac{j(\omega L - \frac{1}{\omega C})}{R + j(\omega L - \frac{1}{\omega C})} = \frac{-\omega^2 LC + 1}{j\omega RC - \omega^2 LC}$$

$Q = \frac{\omega_0 L}{R}$ quality of notch filter → higher Q = smaller bandwidth

$= \frac{1}{\omega_0 RC}$

$\omega_0 = \sqrt{\frac{1}{LC}}$





$d_L = \theta(r + \frac{L}{2}) = \theta r + \frac{\theta L}{2}$

$d_R = \theta(r - \frac{L}{2}) = \theta r - \frac{\theta L}{2}$

$v^* = \frac{r\theta}{i}$ so $\theta = \frac{v^* i}{r}$

$\delta_{ref}[i] = d_L - d_R = \theta L$

$= \boxed{\frac{v^* i L}{r}}$

POST MIDTERM CONTENT !

controls

- real world = continuous
  computers = discrete

  convert continuous → discrete

- state ($\vec{x}$) = collection of vars
- control ($\vec{u}$) = control input that can change state

control model

continuous time:

$$\frac{\partial}{\partial t}\vec{x}(t) = A\vec{x}(t) + B\vec{u}(t)$$

$$\vec{x}(0) = \vec{x}_0$$

Discrete time

$$\vec{x}[i+1] = A_d\vec{x}[i] + B_d\vec{u}[i]$$

$$\vec{x}[0] = \vec{x}_0$$

A, B = matrices representing how state transitions

state trajectories

① continuous (scalar) ①

$$\vec{x}(t) = x_{t_0}e^{at} + \int_{t_0}^{t} e^{a(t-\tau)} bu(\tau)d\tau$$

Discrete (*unroll recursion)

$$\vec{x}[i] = A^i\vec{x}[0] + \sum_{k=0}^{i-1} A^{i-k-1} B\vec{u}[k]$$

---

② continuous (diagonal)

$$\rightarrow \vec{x}(t) = e^{At}\vec{x}(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}\vec{b}u(\tau)d\tau$$

$$\frac{d}{dt}\vec{x}(t) = \Lambda\vec{x}(t) + \vec{b}u(t) \qquad \Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$$

$$\frac{d}{dt}\begin{bmatrix} x_1 t \\ \vdots \\ x_n t \end{bmatrix} = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}\begin{bmatrix} x_1 t \\ \vdots \\ x_n t \end{bmatrix} + \begin{bmatrix} b_1 u(t) \\ \vdots \\ b_n u(t) \end{bmatrix}$$

$$\begin{bmatrix} \frac{\partial}{\partial t} x_1(t) \\ \vdots \\ \frac{\partial}{\partial t} x_n(t) \end{bmatrix} = \begin{bmatrix} \lambda_1 x_1(t) + b_1 u(t) \\ \vdots \\ \lambda_n x_n(t) + b_n u(t) \end{bmatrix}$$

* each $x_i(t) = x_i(t_0)e^{\lambda_i t} + \int_{t_0}^{t} e^{\lambda_i(t-\tau)} b_i u(\tau)d\tau$

---

continuous (diagonalizable)

$$\vec{x}(t) = Ve^{\Lambda t}V^{-1}\vec{x}(t) + \int_{t_0}^{t} Ve^{\Lambda(t-\tau)}V^{-1}\vec{b}u(\tau)d\tau$$

$$\frac{\partial}{\partial t}\vec{x}(t) = A\vec{x}(t) + \vec{b}u(t)$$

) diagonalize

$$\frac{\partial}{\partial t}\tilde{\vec{x}}(t) = \Lambda\tilde{\vec{x}}(t) + V^{-1}\vec{b}u(t)$$

$$\tilde{\vec{x}}(t) = e^{\Lambda t}\tilde{\vec{x}}(t_0) + \int_{t_0}^{t} e^{\Lambda(t-\tau)}V^{-1}\vec{b}u(\tau)d\tau$$

) apply diagonal case

$$V\tilde{\vec{x}}(t) = Ve^{\Lambda t}\tilde{\vec{x}}(t_0) + \int_{t_0}^{t} Ve^{\Lambda(t-\tau)}V^{-1}\vec{b}u(\tau)d\tau$$

) convert form $\tilde{\vec{x}}(t) \rightarrow \vec{x}(t)$

① Discretization = approximate CT model w/ DT model

② Disturbances / noise = handle / account for random noise in models

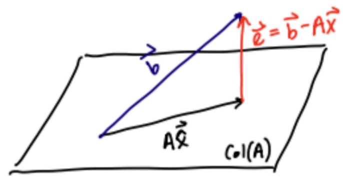
③ System identification = learn estimates for model parameters from data

④ Validation = quantify quality of model w/ loss function
use extra data to measure loss

Least squares

Goal: find "closest" vector to $\vec{b}$ in span of A $\longrightarrow$ $A\vec{x}=\vec{b}$

* minimizes norm of $\vec{e}$
  ↳ perpendicular to $A\vec{x}$



* Dot product $\langle \vec{e}, \vec{a}_i \rangle = \vec{a}_i^T \vec{e} = 0$ when perpendicular

$\hat{x} = (A^TA)^{-1}A^Tb$ * assuming A^TA invertible

Matrix edition: $DP = S$ ← desired vector
↑ matrix of parameters

$D[\vec{p}_1 \cdots \vec{p}_n] = [\vec{s}_1 \cdots \vec{s}_n]$

* Split into n different least squares problems: $\boxed{\begin{array}{l} D\vec{p}_i = \vec{s}_i \\ \hat{\vec{p}}_i = (D^TD)^{-1}D^T\vec{s}_i \end{array}}$

* Stack together: $\boxed{\hat{P} = (D^TD)^{-1}D^TS}$

System ID = find model parameters A_d & B_d

① quantify accuracy of model

minimize $\left\| \begin{bmatrix} A_d \\ B_d \end{bmatrix} - \begin{bmatrix} \hat{A}_d \\ \hat{B}_d \end{bmatrix} \right\|^2 \xrightarrow{\text{use Frobenius norm}} \|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |A_{ij}|^2}$ sqrt of all elements squared

Given
$\vec{x}_a[i+1] = A_d\vec{x}_d[i] + \vec{b}_d u_d[i]$
$\vec{x}_a[i+1] = [A_d \ \vec{b}_d] \begin{bmatrix} \vec{x}_d[i] \\ \vec{u}_d[i] \end{bmatrix}$

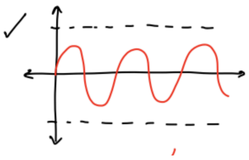
** Block transpose = apply transpose & reverse order

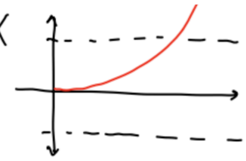
TRANSPOSE
$\vec{x}_a^T[i+1] = [\vec{x}_a^T[i] \ \ u_d^T[i]] \begin{bmatrix} A_d \\ \vec{b}_d^T \end{bmatrix}$

$\underbrace{S \approx D}_{\text{values we can choose}}$ $P \longrightarrow$ Least squares $\rightarrow \boxed{\hat{P} = (D^TD)^{-1}D^TS}$

Stability

Discrete $\|x_d[i]\| \le R_d$ or continuous $\|x(t)\| \le R_d$

✓


X


BIBO = bounded input, bounded output stability
- bounded input function $\vec{u}$ & every initial condition $\vec{x}_0$
  ↳ state trajectory is bounded

State trajectories

continuous

$$\begin{cases} \frac{d}{dt}x(t) = \lambda x(t) + bu(t) \\ x(0) = x_0 \end{cases}$$

$$x(t) = x_0 e^{-\lambda t} + e^{-\lambda t}\int_0^t e^{\lambda t} bu(\tau)\,d\tau$$

Discrete

$$\begin{cases} x[i+1] = \lambda_d x[i] + b_d u[i] \\ x[0] = x_0 \end{cases}$$

$$x[i] = \lambda_d^{\,i} x_0 + \sum_{j=0}^{i-1} \lambda_d^{\,i-j-1} b_d u[j]$$

When do these blow up

① continuous

$x(t)$ dominated by $e^{-\lambda t}$

- want to avoid $e^{-\lambda t}$ from blowing up
  ↳ exponential decay $\lambda < 0$
- if λ = complex : $\lambda = a + jb$

$$e^{-(a+jb)t} = e^{-at} \cdot \underbrace{e^{-jbt}}_{\text{magnitude}=1}$$

* only care abt $Re\{\lambda\}$ for stability

$\boxed{\text{stable: } Re\{\lambda\} < 0}$

② Discrete

* geometric serie-ish behavior

→ $\lambda_d^{\,i} x_0$: blows up $\lambda_d > 1$

→ $\sum_{j=0}^{i-1} \lambda_d^{\,i-1-j} \underbrace{bu[j]}_{\text{bounded so } \le R} \le$

$\le Rb \sum_{j=0}^{i-1} \lambda_d^{\,i-1-j}$] explodes when $|\lambda_d| > 1$

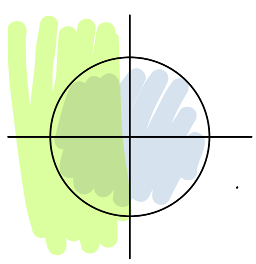
$\boxed{\text{stable: } |\lambda_d| < 1}$

** $\underbrace{\lambda = 0 \ \& \ \lambda_d = 1}$

marginal stability (may or may not blow up

Discrete stable: $|\lambda_i| < 1$

Continuous stable: $Re\{\lambda_i\} < 0$



vector control models w/ multidimensional state space

① $\frac{\partial}{\partial t}\vec{x}(t) = A\vec{x}(t) + \vec{b}u(t)$

$\downarrow$ DIAGONALIZE

$\frac{\partial}{\partial t}\tilde{\vec{x}}(t) = \Lambda\tilde{\vec{x}}(t) + V^{-1}\vec{b}u(t)$

$$\begin{bmatrix} \frac{\partial}{\partial t}x_1(t) \\ \vdots \\ \frac{\partial}{\partial t}x_n(t) \end{bmatrix} = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix} + \begin{bmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_n \end{bmatrix} u(t)$$

* each row becomes scalar case
* $Re\{\lambda_i\} < 0$
* all $x_1(t) \cdots x_n(t)$ must be stable

$\boxed{Re\{\lambda_i\} < 0 \text{ for all } \lambda_i \text{ where } \lambda_i = A \text{ eigenvalues}}$

② $\vec{x}[i+1] = A_d\vec{x}[i] + \vec{b}_d u[i]$

$\downarrow$ Diagonalize

$\tilde{\vec{x}}[i+1] = \Lambda_d\tilde{\vec{x}}[i] + V^{-1}\vec{b}_d u[i]$

$$\begin{bmatrix} x_1[i+1] \\ \vdots \\ x_n[i+1] \end{bmatrix} = \begin{bmatrix} \lambda_{d_1} & & 0 \\ & \ddots & \\ 0 & & \lambda_{d,n} \end{bmatrix} \begin{bmatrix} x_1[i] \\ \vdots \\ x_n[i] \end{bmatrix} + \begin{bmatrix} \tilde{b}_{d,1} \\ \vdots \\ \tilde{b}_{d,n} \end{bmatrix} u[i]$$

* each row = scalar
* All $x_1[i] \cdots x_n[i]$ must be stable

$\boxed{|\lambda_{d,i}| < 1 \text{ for all } \lambda_{d,i} \text{ where } \lambda_{d,i} = \text{eigenvalues of } A_d}$

Feedback control

choose $u(t)$ to be some function of current state

$u(t) = f x(t)$

$\rightarrow \frac{\partial}{\partial t}x(t) = \lambda x(t) + bu(t)$

$\rightarrow \frac{\partial}{\partial t}x(t) = \lambda x(t) + b(f x(t))$

) substitute feedback

$$\boxed{\frac{\partial}{\partial t}x(t) = \underbrace{(\lambda + bf)}_{}x(t)}$$

NEW EIGENVALUE: we can choose f (feedback coefficient)

TO MAKE $\boxed{\lambda + bf \text{ stable}}$

Reachability / controllability

Reachability: provide inputs that push model state to some target given some initial state

controllability: model can reach ANY given target state from ANY initial state

DT state trajectory: $\vec{x}[i] = A^i \vec{x}[0] + \sum_{k=0}^{i-1} A^{i-1-k} B\vec{u}[k]$

$$= A^i \vec{x}[0] + \underbrace{\begin{bmatrix} A^{i-1}B & A^{i-2}B & \cdots & AB & B \end{bmatrix}}_{\text{controllability matrix}} \begin{bmatrix} \vec{u}[0] \\ \vec{u}[1] \\ \vdots \\ \vec{u}[i-2] \\ \vec{u}[i-1] \end{bmatrix}$$

controllability matrix @ timestep i

$$C_i = \begin{bmatrix} A^{i-1}B & A^{i-2}B & \cdots & AB & B \end{bmatrix} \longrightarrow \boxed{\vec{x}[i] = A^i \vec{x}[0] + C_i \underbrace{\begin{bmatrix} \vec{u}[0] \\ \vec{u}[1] \\ \vdots \\ \vec{u}[i-1] \end{bmatrix}}_{\text{choose these values to reach "anything" in span of } C_i}}$$

Reachability

Input: given fixed initial state $\vec{x}_0 \in \mathbb{R}^n$
fixed target state $\vec{x}^* \in \mathbb{R}^n$

* $\vec{x}^*$ reachable in i timesteps from $\vec{x}_0 \longleftrightarrow \boxed{\vec{x}^* - A^i \vec{x}_0 \in \underbrace{\text{col}(C_i)}_{\text{span\{cols of } C_i\}}}$

① solve using Gaussian elimination
$\longmapsto$ multiple solns

② no solns = not reachable

Controllability

Input: ANY initial state $\vec{x}_0 \in \mathbb{R}^n$
ANY target state $\vec{x}^* \in \mathbb{R}^n$

① controllable in i timesteps $\rightarrow \text{col}(C_i) = \mathbb{R}^n$
② controllable (no time constraint) $\rightarrow \text{col}(C_n) = \mathbb{R}^n$

* need span of cols in C_i to be $\mathbb{R}^n$

$C_n = \begin{bmatrix} A^{n-1}B & A^{n-2}B & \cdots & AB & A \end{bmatrix}$
where $\vec{x}[i] \in \mathbb{R}^n$

$$A_{cont} = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \\ -\alpha_0 & -\alpha_1 & -\alpha_2 & -\alpha_3 & \dots & -\alpha_{n-1} \end{bmatrix}, \quad B_{cont} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix},$$

$$C_{cont} = \begin{bmatrix} \beta_0 & \beta_1 & \dots & \beta_{n-1} \end{bmatrix}, \quad D_{cont} = d_0.$$

controllable canonical form

LINEAR ALGEBRA

Definition 13 (Controllable Canonical Form for Discrete-Time LTI Difference Equation Model)
A Discrete-Time LTI Difference Equation Model is in *controllable canonical form (CCF)* if $m = 1$, i.e.,

$$\vec{x}[i+1] = A\vec{x}[i] + \vec{b}u[i], \tag{7}$$

and the coefficients have the following special structure:

$$A = \begin{bmatrix} \vec{0}_{n-1} & I_{n-1} \\ a_1 \quad a_2 & \cdots \quad a_n \end{bmatrix} \in \mathbb{R}^{n\times n} \quad \text{and} \quad \vec{b} = \begin{bmatrix} \vec{0}_{n-1} \\ 1 \end{bmatrix} \in \mathbb{R}^n. \tag{8}$$

Span

If $\vec{v} \in \text{span}\{\vec{s}_1, \vec{s}_2\} \rightarrow \vec{v} = \alpha\vec{s}_1 + \beta\vec{s}_2$ for some $\alpha, \beta \in \mathbb{R}$

$\vec{v}$ = linear combo of α, β

Orthonormality = orthogonal + normal

① $\vec{x}, \vec{y}$ orthogonal $\longleftrightarrow \langle \vec{x}, \vec{y} \rangle = \vec{y}^T \vec{x} = 0$

For set of vectors $S = \{\vec{s}_1 \cdots \vec{s}_m\}$

$\langle \vec{s}_i, \vec{s}_j \rangle = 0$ for all $\vec{s}_i \neq \vec{s}_j$

$$\langle \vec{x}, \vec{y} \rangle = \begin{cases} 1 & \text{if } \vec{x} = \vec{y} \\ 0 & \text{if } \vec{x} \neq \vec{y} \end{cases}$$

② Normalized : unit norm $\|\vec{x}\| = 1$

orthonormal sets of vectors → ① All vectors unit norm
② All vectors orthogonal to each other

3 cases

① Square $Q \in \mathbb{R}^{n\times n}$ (cols & rows are orthonormal sets)
"orthonormal matrix"

$$\boxed{\begin{aligned} Q^TQ &= I_n \\ QQ^T &= I_n \\ Q^T &= Q^{-1} \end{aligned}}$$

② Tall $Q \in \mathbb{R}^{m\times n}$ $(m \geq n)$ (cols = orthonormal set)

$$\boxed{Q^TQ = I_n}$$

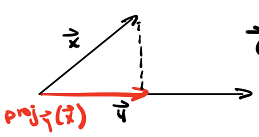
③ Wide $Q \in \mathbb{R}^{m\times n}$ $(m \leq n)$ (rows = orthonormal set)

$$\boxed{QQ^T = I_m}$$

add this!!

Projections

Find component of $\vec{x}$ in direction of $\vec{y}$



$\vec{e} = \vec{x} - \text{proj}_{\vec{y}}(\vec{x}) = \vec{x} - \underbrace{c\vec{y}}_{\text{scalar multiple of } \vec{y} \text{ (span of } \vec{y})}$

$\vec{e} \perp \vec{y} \rightarrow \langle \vec{e}, \vec{y} \rangle = 0$

$\langle \vec{x} - c\vec{y}, \vec{y} \rangle = 0$

$\langle \vec{x}, \vec{y} \rangle - c\langle \vec{y}, \vec{y} \rangle = 0$

$c = \dfrac{\langle \vec{x}, \vec{y} \rangle}{\langle \vec{y}, \vec{y} \rangle}$

$$\boxed{\text{proj}_{\vec{y}}(\vec{x}) = c\vec{y} = \frac{\langle \vec{x}, \vec{y} \rangle}{\langle \vec{y}, \vec{y} \rangle}\vec{y} = \frac{\langle \vec{x}, \vec{y} \rangle}{\|\vec{y}\|^2}\vec{y}}$$

If $\vec{y}$ is normalized $\|\vec{y}\|^2 = 1 \rightarrow \boxed{\text{proj}_{\vec{y}}(\vec{x}) = \langle \vec{x}, \vec{y} \rangle \vec{y}}$

Gram-Schmidt orthonormalization =

process for converting set of vectors / matrix into orthonormal set of vectors w/ same span
↓
aka matrix w/ orthonormal cols w/ same col space

$S = \{\vec{s}_1 \cdots \vec{s}_n\} \longrightarrow Q = \{\vec{q}_1 \cdots \vec{q}_n\}$

① Q = orthnormal set of vectors
② span $\{\vec{s}_1 \cdots \vec{s}_n\}$ = span $\{\vec{q}_1 \cdots \vec{q}_n\}$

Main idea
① Normalize first vector & add to Q
② For each subsequent vector:
  a) Project vector onto all vectors in Q
  b) Subtract projections from vector
  c) normalize new vector & add to Q
③ Repeat step 2 until desired subspace reached

a) $\vec{q}_1 = \dfrac{\vec{s}_1}{\|\vec{s}_1\|}$

b) $\vec{q}_2 = \dfrac{\vec{z}_2}{\|\vec{z}_2\|}$ where $\vec{z}_2 = \vec{s}_2 - \langle \vec{s}_2, \vec{q}_1 \rangle \vec{q}_1$

* $\vec{z}_i$ = component of $\vec{s}_i$ that's orthogonal to all prev vectors

c) $\vec{q}_3 = \dfrac{\vec{z}_3}{\|\vec{z}_3\|}$ where $\vec{z}_3 = \vec{s}_3 - \langle \vec{s}_3, \vec{q}_1 \rangle \vec{q}_1 - \langle \vec{s}_3, \vec{q}_2 \rangle \vec{q}_2$

d) $\vec{q}_4 = \dfrac{\vec{z}_4}{\|\vec{z}_4\|}$ where $\vec{z}_4 = \underbrace{\vec{s}_4 - \langle \vec{s}_4, \vec{q}_1 \rangle \vec{q}_1 - \langle \vec{s}_4, \vec{q}_2 \rangle \vec{q}_2 - \langle \vec{s}_4, \vec{q}_3 \rangle \vec{q}_3}$

subtract projections of s_i onto all prev set vectors

$$\boxed{\vec{q}_i = \frac{\vec{z}_i}{\|\vec{z}_i\|} \quad \text{where} \quad \vec{z}_i = \vec{s}_i - \sum_{j=1}^{i-1} \langle \vec{s}_i, \vec{q}_j \rangle \vec{q}_j}$$

Basis Extension

Goal: orthonormal basis for $\mathbb{R}^n$ that has some basis vectors of og S
* append standard basis vectors to S
* Run GS on all vectors

ie extend basis of $\mathbb{R}^3$ from $S = \left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \right\}$

① span $\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \right\} \neq \mathbb{R}^3$

② Add standard basis of $\mathbb{R}^3$ & run GS & run GS

$GS\left(\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\} \right) \rightarrow$ basis for $\mathbb{R}^3$

Upper triangularization

when solving for state trajectories / stability

$\vec{x}[i+1] = A\vec{x}[i] + \vec{b}u[i]$

$\downarrow$ diagonalize

$\tilde{\vec{x}}[i+1] = \Lambda\tilde{\vec{x}}[i] + V^{-1}\vec{b}u[i]$

→ decompose into indep scalar equations

→ scalar state trajectory → scalar cond for BIBO → combine for vector case

If A NOT diagonalizable → upper triangular matrix $T = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1n} \\ 0 & t_{22} & & \\ \vdots & & \ddots & \\ 0 & & & t_{nn} \end{bmatrix}$

characteristics
- eigenvalues along diagonal
- last row decoupled: solve for $x_n \to x_{n-1} \cdots x_1$
- don't need lin indep eigenvalues to be upper Δ

$$\frac{d}{dt}\vec{x}(t) = \begin{bmatrix} t_{11} & t_{21} & \cdots & t_{1n} \\ & t_{22} & & \vdots \\ & & \ddots & \\ 0 & & & t_{nn} \end{bmatrix}\vec{x}(t)$$

last n^{th} row: $\frac{d}{dt}x_n(t) = t_{nn}x_n(t)$

$x_n(t) = x_n(0)e^{t_{nn} \cdot t}$

→ sub into $(n-1)^{th}$ row: $\frac{d}{dt}x_{n-1}(t) = t_{n-1,n-1}x_{n-1}(t) + t_{n-1,n}x_n(t)$

$\downarrow$ sub soln for $x_n(t)$

$$\underbrace{\frac{d}{dt}x_{n-1}(t) = t_{n-1,n-1}x_n(t) + t_{n-1,n}\left(x_n(0)e^{t_{n,n}t}\right)}_{\text{non-homogeneous 1st order scalar diff eq}}$$

Schur Decomposition

$A \in \mathbb{R}^{n\times n}$ w/ real λ_i $\rightarrow$ orthonormal change of basis $U \in \mathbb{R}^{n\times n}$
upper triangle matrix $T \in \mathbb{R}^{n\times n}$

$$\boxed{A = UTU^T \longleftrightarrow T = U^TAU}$$

Q = orthonormal basis extended from $\vec{v}_1$: $Q = [\vec{v}_1 \;\; R]$

R: remainder of orthonormal basis

$$Q^TAQ = [\vec{v}_1 \;\; R]^T [A][\vec{v}_1 \;\; R]$$

$$= \begin{bmatrix} \vec{v}_1^T \\ R^T \end{bmatrix} \underbrace{[A][\vec{v}_1 \;\; R]}_{\text{combine}}$$

$$= \begin{bmatrix} \vec{v}_1^TA\vec{v}_1 & \vec{v}_1AR \\ R^TA\vec{v}_1 & R^TAR \end{bmatrix}$$

sub $A\vec{v}_1 = \lambda\vec{v}_1$

$$= \underbrace{\begin{bmatrix} \vec{v}_1^T\lambda\vec{v}_1 & \vec{v}_1AR \\ R^T\lambda\vec{v}_1 & R^TAR \end{bmatrix}}_{\text{Factor } \lambda}$$

$$= \begin{bmatrix} \lambda(\vec{v}_1^T\vec{v}_1) & \vec{v}_1AR \\ \lambda(R^T\vec{v}_1) & R^TAR \end{bmatrix}$$

* $v_1^Tv_1 = \langle \vec{v}_1, \vec{v}_1 \rangle = \|\vec{v}_1\|^2 = 1$ (since normalized eigenvector)

* $R^T\vec{v}_1 = \begin{bmatrix} - & r_2^T & - \\ & \vdots & \\ - & \vec{r}_n^T & - \end{bmatrix} \vec{v}_1 = \begin{bmatrix} \vec{r}_2^T\vec{v}_1 \\ \vdots \\ \vec{r}_n^T\vec{v}_1 \end{bmatrix} = \begin{bmatrix} \langle \vec{v}_1, \vec{r}_2 \rangle \\ \vdots \\ \langle \vec{v}_1, \vec{r}_n \rangle \end{bmatrix} = \vec{0}_{n\times 1}$

$$= \begin{bmatrix} \lambda_1 & -\vec{v}_1^TAR- \\ \vec{0} & R^TAR \end{bmatrix} \qquad \begin{bmatrix} 1\times 1 & 1\times(n-1) \\ (n-1)\times 1 & (n-1)\times(n-1) \end{bmatrix}$$

Use R^TAR as indicator

first column is "upper triangularized"

$\rightarrow$ if R^TAR is upper triangular then we can fully upper triangularize A

* repeat on A until reach a 1×1 matrix

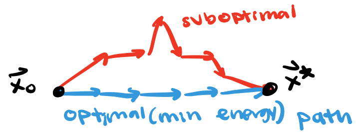
$\boxed{\text{RECURSION}}$

- Base case: 1×1 already upper triangular
- Reduce to subproblem : Apply Q to A $\rightarrow$ UT first column (upper triangularize)
- Recursive step: UT the $(n-1)\times(n-1)$ matrix

Minimum Energy Control

Reachable if $(\vec{x}^* - A^{i^*}\vec{x}_0) \in \text{col}(C_i^*)$ → could exist INFINITE solns

want to pick soln that minimizes energy



Energy $\vec{w} = \begin{bmatrix} \vec{u}[0] \\ \vdots \\ \vec{u}[i^*-1] \end{bmatrix} \rightarrow \|\vec{w}\|^2 = \sum_{i=0}^{i-1} \|\vec{u}[i]\|^2$

SQUARED NORM

★ choose some $\vec{w}$ with $\boxed{\text{NO}}$ component in Null(C) → doesn't help us reach goal

OPTIMIZATION PROBLEM $\min_{\vec{w}} \|\vec{w}\|^2 \quad \text{st} \quad C\vec{w} = \vec{z}$

Spectral Theorem

$A \in \mathbb{R}^{n \times n}$ real & symmetric → v. nice properties

① real eigenvalues
② A is diagonalizable
③ Exists orthonormal basis of $\mathbb{R}^n$ consisting of eigenvectors of A

→ A may be "orthonormally diagonalized"

$A = V\Lambda V^T \longleftrightarrow \Lambda = V^T A V$

↳ orthonormal matrix of eigenvectors

diagonal matrix of eigenvalues

★ eigenvectors corresponding to 0 eigenvalues span matrix nullspace

SVD

$\boxed{A = U\Sigma V^T}$

U = orthonormal matrix $\in \mathbb{R}^{m \times m}$
Σ = nonsquare diagonal matrix w/ singular values $\sigma_i = \sqrt{\lambda_i}$ ↳ eigenvalue of AA^T or A^TA
V^T = orthonormal matrix $\in \mathbb{R}^{n \times n}$

wide matrix

Full SVD: $A = U\Sigma V^T$

$$A = \begin{bmatrix} \vec{u}_1 & \cdots & \vec{u}_m \end{bmatrix} \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ & & \sigma_m \end{bmatrix} \begin{bmatrix} \vec{v}_1 \\ \vdots \\ \vec{v}_n \end{bmatrix}$$

Compact SVD: $A = U_r \Sigma_r V^T$

$\text{Rank}(A) = r$

$$A = \left[\vec{u}_1 \cdots \vec{u}_r \,\middle|\, \vec{u}_{r+1} \cdots \vec{u}_m\right] \left[\begin{array}{c|c} \sigma_1 \ddots \sigma_r & 0 \\ \hline 0 & 0 \end{array}\right] \left[\begin{array}{c} \vec{v}_1^T \\ \vdots \\ \vec{v}_r^T \\ \hline \vec{v}_{r+1}^T \\ \vdots \\ \vec{v}_n \end{array}\right]$$

$$= \left[U_r \mid U_{m-r}\right] \left[\begin{array}{c|c} \Sigma_r & 0_{r\times(n-r)} \\ \hline 0_{(m-r)\times r} & 0_{(m-r)\times(n-r)} \end{array}\right] \left[\begin{array}{c} V_r^T \\ \hline V_{n-r}^T \end{array}\right]$$

$$= \left[V_r \mid U_{m-r}\right] \begin{bmatrix} \Sigma_r V_r^T + 0 \cdot V_{n-r}^T \\ 0 V_r^T + 0 \cdot V_{n-r}^T \end{bmatrix}$$

$$= \left[U_r \mid U_{m-r}\right] \begin{bmatrix} \Sigma_r V_r^T \\ 0 \end{bmatrix} = U_r \Sigma_r V_r^T + U_{m-r} \cdot 0 = U_r \Sigma_r V_r^T$$

Full SVD: $A = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^T$

Full SVD: $A = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^T$
Compact SVD: $A = U_r \Sigma_r V^T$
wide matrix
Full SVD: $A = U \Sigma V^T$

$$A = \begin{bmatrix} \vec{u}_1 \cdots \vec{u}_m \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_m \end{bmatrix} \begin{bmatrix} \vec{v}_1^T \\ \vdots \\ \vec{v}_n^T \end{bmatrix}$$

$m \times m$ $\quad$ $m \times n$ $\quad$ $n \times n$

$$= \begin{bmatrix} \vec{u}_1 \cdots \vec{u}_m \end{bmatrix} \begin{bmatrix} \sigma_1 v_1^T \\ \vdots \\ \sigma_m v_m^T \end{bmatrix}$$

$m \times m$ $\quad$ $m \times n$

$$= \vec{u}_1 (\sigma_1 \vec{v}_1^T) + \vec{u}_2 (\sigma_2 \vec{v}_2^T) + \dots + \vec{u}_m \vec{\sigma}_m \vec{v}_m^T$$

$$= \sigma_1 (\vec{u}_1 \vec{v}_1^T) + \dots + \sigma_m \vec{u}_m \vec{v}_m^T$$

* If $\text{rank}(A) = r \rightarrow \sigma_{r+1} \cdots \sigma_m = 0$

$$= \sigma_1 (\vec{u}_1 \vec{v}_1^T) + \dots + \sigma_r \vec{u}_r \vec{v}_r^T$$

$$= \sum_{i=0}^{r} \sigma_i \vec{u}_i \vec{v}_i^T$$

Spectral Thm

A is square & symmetric

↳ $A = V \Lambda V^T$

Λ = square & diagonal w/ A's eigenvalues

$V = [V_r \quad V_{n-r}]$

V_r: $\text{col}(A) = \text{col}(A^T)$

V_{n-r}: $\text{Null}(A) = \text{Null}(A^T)$

SVD

A is NOT symmetric / square

$A = U \Sigma V^T$

↳ non-square diagonal matrix of singular values

$U = [U_r \quad U_{m-r}] \in \mathbb{R}^{m \times m}$ ORTHONORMAL

* U_r spans $\text{col}(A)$
* U_{m-r} orthogonal to $\text{col}(A)$ spans $\text{Null}(A^T)$

$V = [V_r \quad V_{n-r}] \in \mathbb{R}^{n \times n}$ orthonormal

* V_r spans $\text{col}(A^T)$
* V_{n-r} orthogonal to $\text{col}(A^T)$ spans $\text{Null}(A)$

SVD Applications

- U is orthonormal $\in \mathbb{R}^{m \times m}$

$U = [U_r \;\; U_{m-r}]$

$U_r \to Col(A)$, $\quad U_{m-r} \to Null(A^T)$

V orthonormal $\in \mathbb{R}^{n \times n}$

$V = [V_r \;\; V_{n-r}]$

$V_r \to Col(A^T)$ rowspace, $\quad V_{n-r} \to Null(A)$

Σ is diagonal $\in \mathbb{R}^{m \times n}$

$$\left[\begin{array}{c|c} \sigma_1 \ddots \sigma_r & 0 \\ \hline 0 & 0 \end{array}\right] = \left[\begin{array}{c|c} \Sigma & 0 \\ \hline 0 & 0 \end{array}\right]$$

① A is square, invertible

$\vec{x} = A^{-1}\vec{b}$

② A is tall, linear independent cols

$\vec{x} = (A^TA)^{-1}A^T\vec{b}$

③ A is tall/wide w/ linear indep/dep cols

$\vec{x} = A^+\vec{b}$

↳ pseudoinverse $A^+ = V_r \Sigma_r^{-1} U_r^T$

Geometry of matrices

- Diagonal = scaling
- Orthonormal = rotation / reflection
  norm = constant
- other matrices: combo of scaling, rotation, reflection

SVD: $U\Sigma V^T \to$ rotate/reflect + scaling + rotation/reflection

Comparison of SVD Forms

(Note 14)

$A \in \mathbb{R}^{m \times n}$, $rank(A) = r$

		Strengths	Weaknesses
① Full SVD	$A = U\Sigma V^T$	· U, V are orthonormal (square) · $Null(A^T)$ as $Col(U_{m-r})$ · $Null(A)$ as $Col(V_{n-r})$	· computationally intensive ↳ need GS (twice) · Σ non-square, not invertible
② Compact SVD	$A = U_r \Sigma_r V_r^T$	· Σ_r is square, invertible · easier to construct than Full SVD	· U_r, V_r not square, invertible · no characterization of Null
③ Outer Product SVD	$A = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^T$	· computationally efficient · easier to construct than Full SVD	· summation notation messy · no characterization of Null

Projections = mapping of least squares soln

$A\vec{x} = \vec{b}$ (tall lin indep cols)

$\vec{x}^* = (A^TA)^{-1}A^T\vec{b}$ ← least squares soln

$A\vec{x}^* = \vec{p}$

$\text{proj}_{\text{col}(A)}\vec{b} = A(A^TA)^{-1}A^T\vec{b}$

When A is orthonormal

$\text{proj}_{\text{col}(A)}\vec{b} = AA^T\vec{b}$

Minimum Norm Solution

wide w/ full column rank

$\boxed{\vec{x}^* = A^T(AA^T)^{-1}\vec{b}}$

Pseudoinverse

1. Full pseudoinverse : $A^+ = V\Sigma^+U^+$

2. compact pseudoinverse : $A^+ = V_r\Sigma_r^{-1}U_r^T$

dagger

inverts non-zero diagonal elements

$$\Sigma^+ = \left[\begin{array}{c|c} \Sigma_r & 0 \\ \hline 0 & 0 \end{array}\right]^+ = \left[\begin{array}{c|c} \Sigma_r^{-1} & 0 \\ \hline 0 & 0 \end{array}\right]$$

disregards vectors in Null(A)

Properties of Pseudo-inverse (prove on own time ◡̈)

- If A is invertible, $A^{-1} = A^+$
- $(A^+)^+ = A$
- $(A^T)^+ = (A^+)^T$
- $\alpha \neq 0$, $(\alpha A)^+ = \alpha^{-1}A^+$
- $AA^+A = A$
- $A^+AA^+ = A^+$
- $AA^+ = U_rU_r^T$ (projection onto Col(A))
- $A^+A = V_rV_r^T$ (projection onto Col(A^T))

we will show these in Dis 12B

Low Rank Approximation

$A \in \mathbb{R}^{m \times n} \rightarrow \text{rank}(A) = r$

* want $A_\ell \in \mathbb{R}^{m \times n} \rightarrow \text{rank}(A_\ell) = \ell <<< r$
that is "close" to A ($\|A - A_\ell\|$ small)

$$A = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^\top$$
$$= \sum_{i=1}^{\ell} \sigma_i \vec{u}_i \vec{v}_i^\top + \sum_{i=\ell+1}^{r} \underbrace{\sigma_i}_{\approx 0} \vec{u}_i \vec{v}_i^\top$$
$$\approx \sum_{i=1}^{\ell} \sigma_i \vec{u}_i \vec{v}_i^\top .$$

PCA

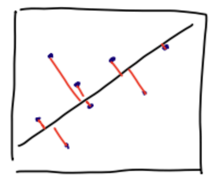
Data $= \vec{x}_1 \cdots \vec{x}_n \in \mathbb{R}^d$
$A = [\vec{x}_1 \cdots \vec{x}_n]$
↳ SVD → $A = U \Sigma V^T$

→ $\boxed{U_\ell \in \underset{W \in \mathbb{R}^{d \times \ell}}{\text{argmin}} \sum_{i=1}^{n} \|\vec{x}_i - WW^T \vec{x}_i\|^2}$

$[\vec{v}_1 \cdots \vec{v}_\ell] = 1^{st}\ \ell$ cols of U

Geometric

$$\boxed{S_{PCA} \in \underset{S \in \mathbb{R}^d}{\text{argmin}} \sum_{i=1}^{n} \|\vec{x}_i - \text{proj}_S(\vec{x}_i)\|^2 \quad \text{st} \quad \dim(S) \le \ell}$$



$\vec{x}_i - \text{proj}_S(\vec{x}_i)$
↳ trying to minimize this error across all data points

* $U_\ell = \ell$ orthnormal vectors w/ "best" subspace closest to all points

COL data:	PC = eigenvectors of AA^T (non-decreasing order by value of λ)	U_ℓ has principal components
ROW data:	PC = eigenvectors of A^TA (order of value of λ)	V_ℓ has principal components

Outer product

$$\sigma_i \vec{u}_i \vec{v}_i^T = \sigma_i \begin{bmatrix} | \\ \vec{u}_i \\ | \end{bmatrix} [-\vec{v}_i^T-]$$

2 interpretations:

(1) $\vec{u}_i$ is the data and you are scaling each column by the components of $\vec{v}_i^T$
⇒ interpreting data in columns

(2) $\vec{v}_i^T$ is the data & you are scaling each row by the components of $\vec{u}_i$
⇒ interpreting data in rows

→ σ_i dictates how much this outer product $\vec{u}_i \vec{v}_i^T$ contributes to overall data matrix A