L1, Monday

Outline

Monday: Overview

- What is control theory? Why? Apps?
- dynamics and dynamical systems
- linearization + linear systems
- frequency-domain control (PID)
- Time-domain control
  - controllability, observability, duality
  - full-state control
  - observers + output control

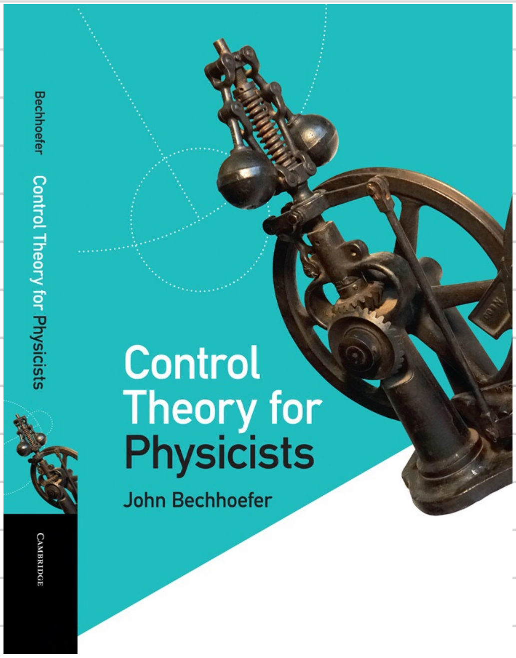
Tuesday: Optimal control

- parametric → Lagrangian (calc. variations)
- Bellman + dynamic prog., LQR
- constraints

Wednesday: Stochastic Control

- state estimation
- Kalman filter, part 1
- Bayesian methods (Kalman filter, part 2)

- Overall reference
- Cambridge Univ. Press 2021
- www.sfu.ca/chaos
  → books → CUP
  + exercises, sol'ns
  + Math Appendix



- Any sufficiently advanced technology is indistinguishable from magic."
  — Arthur C. Clarke, 1973

- dynamics with a purpose
  – profound implications
  – not just math! (a way of thinking)

  purpose: – "intelligent design"
  – evolution (biology, technological)

– control is "the hidden technology" (Karl Åström 1999)
– engineered systems as "robust yet fragile" (John Doyle)

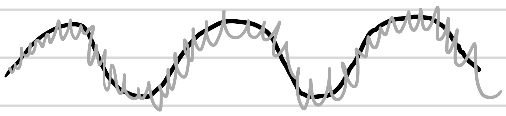
Applications

- Technology ↔ better experiments for physicists
- Physics ↔ control can change physical dynamics (stabilize unstable state / Paul trap)
  - fundamentals of thermo.
  - quantum systems
  - control of complex networks
- Biology ↔
  - large scale (physiology)
  - small scale (genetic regulation)
  - single-mol. control
  - evolutionary time scales (pop. control)

Goals of control

- Regulation

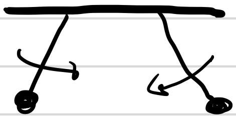
- Tracking



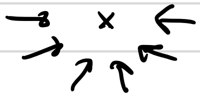
Our focus

- State - state transitions

- Collective motion
  synchronization, swarming, . . .

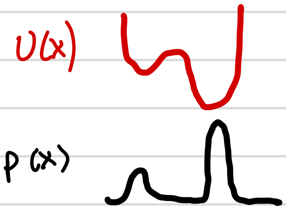


or



- stabilization, virtual potentials ❤️ + forces, change attractor,...

Types of Control



- **passive** : energy landscape

$$U(x) \longrightarrow p(x) \sim e^{-\frac{U(x)}{k_BT}}$$

↳ choose U(x) to get desired p(x)

ie $U = U(x,\lambda) \rightarrow p(x,\lambda)$ for fixed parameter λ

↓

- **active** : alter p(x) by going out of eq.

→ **no sensors** ↔ **open loop** or **feedforward**

$$U(x, \lambda(t)) \longrightarrow p(x, \lambda(t))$$

$\lambda(t)$ ↔ "control par."

→ **sensors** ↔ extra cost to monitor

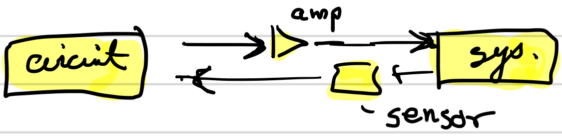
↓

closed-loop or **feedback**

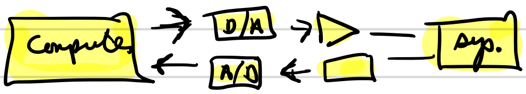
$$U(x, \lambda(x,t)) \rightarrow p(x, \lambda(x,t))$$

Types

- **Gadgets** (flush toilet, governor...)
- **Natural** sys (climate feedbacks ...)
- **Analogue** control



- **Digital** control



The Systems Point of View

nonlinear dyn.

- Control: dynamical systems $\dot{x} = f(x)$ $\in \mathbb{R}^n$

  \+ inputs and outputs

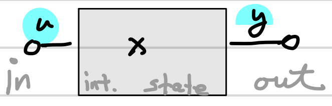
  u y

Control System

x → internal states abstract

u → inputs what we do

y → outputs what we see



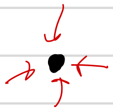
$f(\cdot)$, $g(\cdot)$ → nonlinear functions (usually smooth)

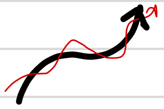
$$\frac{dx}{dt} \equiv \dot{x} = f(x,u) \qquad y = h(x,u) \quad \text{or} \quad h(x)$$

dynamics nonlinear measurements

Linear systems

- easy, w/ general methods
- near fixed point (static)
- near trajectory (time dependent)





Near fixed point

$$\dot{x} = f(x,u) \qquad y = h(x,u)$$

$$f(x_0,u_0) = 0 \qquad h(x_0,u_0) = y_0$$

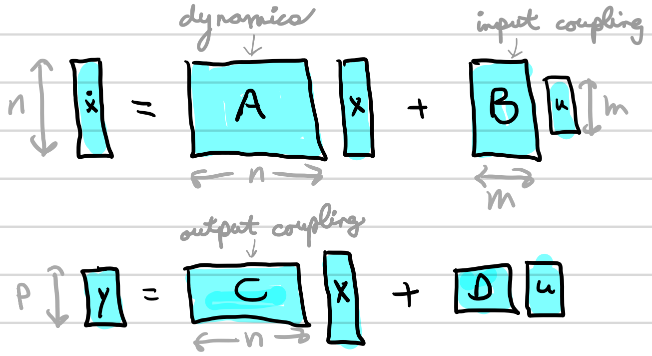
$$x \equiv x_0 + x_1(t) \qquad u \equiv u_0 + u_1(t) \qquad y \equiv y_0 + y_1(t) \qquad \{x_1,u_1,y_1\} \text{ small}$$

Taylor: $\dot{x}_1 = f(x_0 + x_1, u_0 + u_1) \approx \cancelto{0}{f(x_0, u_0)} + \frac{\delta f}{\delta x}\Big|_{x_0,u_0} x_1 + \frac{\partial f}{\partial u}\Big|_{x_0,u_0} u_1 + \cdots$

$$\cancel{y_0} + y_1 = h(x_0 + x_1, u_0 + u_1) \approx \cancel{h(x_0, u_0)} + \frac{\partial h}{\partial x}\Big|_{x_0,u_0} x_1 + \frac{\partial h}{\partial u}\Big|_{x_0,u_0} u_1$$

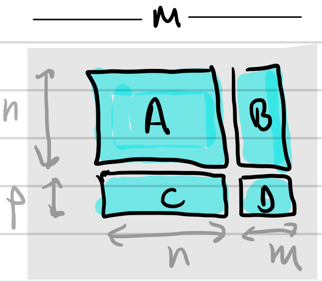
$A = \frac{\partial f}{\partial x}$ $B = \frac{\partial f}{\partial u}$ $C = \frac{\partial h}{\partial x}$ $D = \frac{\partial h}{\partial u}$ all at (x_0, u_0)

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

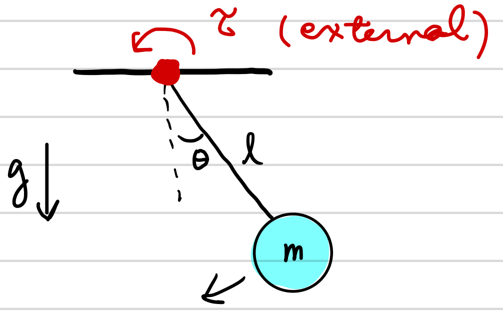


Nice packing of matrices:

$$\begin{pmatrix}\dot{x}\\ y\end{pmatrix} = M \begin{pmatrix}x\\ u\end{pmatrix}$$



Ex: Pendulum



$$\ddot{\theta} + \sin\theta = \tau(t)$$

– scaled (inv.) time by $\omega_0 = \sqrt{g/l}$, etc.

state $x = \begin{pmatrix}\theta\\ \dot{\theta}\end{pmatrix}$ input $u = \tau$ (torque)

output $y = \theta$ (angle)

$$\longrightarrow \quad \dot{x} = \frac{d}{dt}\begin{pmatrix}x_1\\ x_2\end{pmatrix} = \begin{pmatrix}x_2\\ -\sin x_1 + u\end{pmatrix} = f(x, u)$$

Fixed points: $f(x^*) = 0 \longrightarrow x_2 = 0, \ \sin x_1 = 0, \ u = 0$

$(\dot{\theta} = 0, \ \theta = \{0, \pi\})$

$$\frac{d}{dt}\begin{pmatrix}x_1\\ x_2\end{pmatrix} = \overset{A}{\begin{pmatrix}0 & 1\\ -1 & 0\end{pmatrix}}\begin{pmatrix}x_1\\ x_2\end{pmatrix} + \overset{B}{\begin{pmatrix}0\\ 1\end{pmatrix}} u \qquad y = \overset{C}{(1\ 0)}\begin{pmatrix}x_1\\ x_2\end{pmatrix} + \overset{D}{0}$$

Frequency-domain analysis

Laplace transform

Transfer functions

(dynamical response)

$$\circ \xrightarrow[\text{in}]{u(s)} \boxed{G(s)} \xrightarrow[\text{out}]{y(s)} \circ \qquad G(s) = \frac{y(s)}{u(s)} \qquad y(s) = \int_0^\infty dt \cdot y(t)\, e^{-st}$$

- intuitive approach for linear, time-invariant (LTI) systems
- leads to proportional-integral-derivative (PID) control

$$u(t) = \underset{\substack{P \\ \text{present}}}{K_p\, e(t)} + \underset{\substack{I \\ \text{past}}}{K_i \int_{-\infty}^{t} dt'\, e(t')} + \underset{\substack{D \\ \text{future}}}{K_d\, \dot{e}(t)} \qquad \underset{\text{"error"}}{e = r - y}$$

- Developed (mostly) in US 1920-1960 (Bell Labs)
  - Nyquist, Bode, etc.
- what experimentalists and engineers use...

Time-domain Control

Recall: state vector $x(t)$ $\qquad \dot{x} = f(x, \overset{\text{in}}{u}) \qquad \overset{\text{out}}{y} = h(x,u)$

$\dot{x} = Ax + Bu, \quad y = Cx + Du \qquad\qquad A = \partial_x f|_{x_0, u_0}, \quad B = \dots$

Why study the time domain?

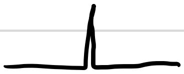
① easily handle multiple inputs + multiple outputs ("MIMO")
② handle noise in a natural way
③ leads to more sophisticated view of dynamics, control
④ generalize to time varying and nonlinear cases

$$\dot{x} = Ax + Bu \longrightarrow x(t) = e^{At}\, \overset{\text{initial condition}}{x(0)} + \int_0^t dt'\, e^{A(t-t')}\, Bu(t')$$

↳ matrix exponential $e^{At} = I + At + A^2 \frac{t^2}{2!} + \dots$

Time - domain response

- Impulse response $u = \delta(t)$



$$y(s) = G(s)\,u(s) \quad \Rightarrow \quad y = G * u = G * \delta = G(t)$$

$\Rightarrow$ $G(t)$ = impulse response function [Green function in physics]

states : $x_{imp}(t) = 0 + \int_0^t dt' \, e^{A(t-t')} B \, \delta(t') = e^{At} B$

- Step response $u(t) = \Theta(t)$

Controllability and observability

Two "technical" questions:

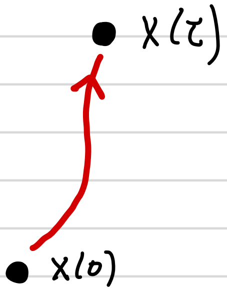
1) Given input(s) $u(t)$, can you "control" all elements of x? → controllability

2) Given observation(s) $y(t)$, can you "infer" all elements of x? → Observability

I. Controllability

- begin w/ SISO case (scalar u, y; $x \in \mathbb{R}^n$)

Controllable solution : $\exists\, u(t)$ makes sys evolve from $x(0) = x_0 \rightarrow x(\tau) = x_\tau$ in finite time τ



Controllable set: set of all x_τ that can be reached from x_0 in time τ

Controllable system: For any x_0, can reach any $x_\tau \in \mathbb{R}^n$ for some τ and $u(t)$ $\exists u: x_0 \to x_\tau$

Ex 1: $\dot{x} = -x + u(t)$ Yes!

"inverse engineering" $x_d(t)$ = desired traj

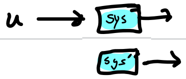
$$u(t) = \dot{x}_d(t) + x_d(t)$$

(not usually poss.)

→ trajectory x_d tracked perfectly (asking more)

Ex 2:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -\lambda_1 & 0 \\ 0 & -\lambda_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u$$

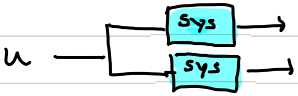


$$\begin{cases} \dot{x}_1 = -\lambda_1 x_1 + u \\ \dot{x}_2 = -\lambda_2 x_2 \end{cases}$$

$x_2(t)$ is completely unaffected by $u(t)$

→ No!

Ex 3:



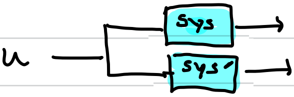
same input to two identical sys. ⇒ No!

Ex 4:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -\lambda_1 & 0 \\ 0 & -\lambda_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u(t) \to$$

Q?

yes*! See next page.



* for $\lambda_1 \neq \lambda_2$

4.1 **Controllability of nearly identical systems**. Consider two first-order systems with relaxation rates $\lambda_1 = 1$ and $\lambda_2 = 2$ that are driven by identical inputs (Eq. 4.7). Find an input $u(t)$ that takes the system from an initial state $\boldsymbol{x}_0 = \binom{0}{0}$ to a final state $\boldsymbol{x}_\tau = \binom{1}{1}$, for $\tau = 1$. Plot your solution. Hint: Try a step function with two parameters.

Solution.

We need to find a solution $x_1(1) = x_2(1)$ for

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -\lambda_1 & 0 \\ 0 & -\lambda_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u(t) \quad \Longrightarrow \quad \begin{aligned} \dot{x}_1 &= -\lambda_1 x_1 + u \\ \dot{x}_2 &= -\lambda_2 x_2 + u \end{aligned},$$

with initial state ($t = 0$) and final states ($\tau = 1$) given by

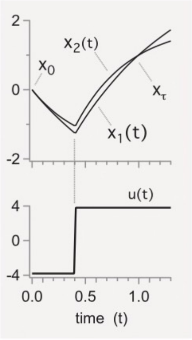
$$x_0 = \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad x_\tau = \begin{pmatrix} x_1(1) \\ x_2(1) \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

There are an infinite number of ways to do this. A basic requirement is that there be two free parameters, since we are trying to fix two conditions ($x_1 = x_2$ at $\tau = 1$). One simple route is to use piecewise constant $u(t)$ functions. Let us try the simple form

$$u(t) = \begin{cases} -u_0 & 0 < t < \tau_0 \\ +u_0 & \tau_0 < t < 1 \end{cases}$$

We can explicitly integrate the x_1 and x_2 equations (they are the same, substituting 2 for 1, etc.) Denoting $x_{1,2}$ by $x(t)$ and $\lambda_{1,2}$ by λ, we get

$$x(t) = \begin{cases} \frac{-u_0}{\lambda}\left(1 - e^{-\lambda t}\right) & 0 < t < \tau_0 \\ \frac{u_0}{\lambda}\left(1 - e^{-\lambda \tau_0}\right) \mathrm{e}^{-\lambda(t-\tau_0)} - (1 - e^{-\lambda(t-\tau_0)}) & \tau_0 < t < 1\,. \end{cases}$$



After a certain amount of playing around, I found the solution illustrated at right, where $u_0 = 3.8$ and $\tau_0 = 0.405$ works for $\lambda_1 = 1$ and $\lambda_2 = 2$. Again, we emphasize that we use the same $u(t)$ for both $x_1(t)$ and $x_2(t)$. It is possible to find explicit algebraic solutions for u_0 and τ_0 in terms of $\lambda_{1,2}$, etc., but the main point here is to understand intuitively how a solution works and why it

Test for Controllability

"the recipe"

1) Define controllability matrix $W_c \equiv (B \ \ AB \ \ A^2B \cdots A^{n-1}B)$

SISO → B a column matrix ⇒ W_c is $n \times n$

2) W_c invertible $(\det W_c \neq 0)$ ⇒ $\{A, B\}$ controllable

Ex 2: $A = \begin{pmatrix} -\lambda_1 & 0 \\ 0 & -\lambda_2 \end{pmatrix}$ $B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $AB = \begin{pmatrix} -\lambda_1 \\ 0 \end{pmatrix}$ ⇒ $W_c = \begin{pmatrix} 1 & -\lambda_1 \\ 0 & 0 \end{pmatrix}$ ⇒ $\det = 0$ ⇒ No

Ex 4: $A = \begin{pmatrix} -\lambda_1 & \\ & -\lambda_2 \end{pmatrix}$ $B = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ $AB = \begin{pmatrix} -\lambda_1 \\ -\lambda_2 \end{pmatrix}$ $W_c = \begin{pmatrix} 1 & -\lambda_1 \\ 1 & -\lambda_2 \end{pmatrix}$

$\det W_c = \lambda_1 - \lambda_2$ ⇒ controllable if $\lambda_1 \neq \lambda_2$

Why it works

1) Can set $x_0 = 0$ (by defining $x_\tau \to x_\tau - e^{A\tau} x_0$)

2) Cayley-Hamilton: matrix A obeys its nth-order characteristic equation

⇒ A^ℓ can be re-expressed as $O(n-1)$ polynomial

⇒ $e^{At} = \sum_{j=0}^{n-1} \alpha_j(t) A^j$

3) impulse response $x_{imp} = e^{At} B = \sum_{j=0}^{n-1} \alpha_j(t) A^j B$

controllability ⇒ impulse response spans $\mathbb{R}^n$

⇒ set of n vectors $B, AB, A^2B, \dots A^{n-1}B$ are all linearly independent ⇒ $\det W_c \neq 0$

4) in gen.

$$\dot{x}_\tau = \underbrace{\begin{pmatrix} B & AB & A^2B & \cdots \end{pmatrix}}_{W_c} \begin{pmatrix} \int_0^\tau dt\; \alpha_0(\tau-t)\, u(t) \\ \int_0^\tau dt\; \alpha_1(\tau-t)\, u(t) \\ \vdots \end{pmatrix}$$

5) **MIMO:** B is a matrix, u a vector

W_c needs to have rank $= n$ "full rank"

Notes

1) Controllability $\not\Rightarrow$ prescribed trajectory

- recall inverse engineering $u = \dot{x}_d + x_d$

2) Reach x_τ at τ $\not\Longrightarrow$ hold x_τ for $t > \tau$

3) Might be good enough to control only some states or to control in a region only

4) Control trajectories can be nonlocal 

5) nonlinear if input saturates $(|u| \leq U_{max})$

→ reachable set

6) Nonlinearity ↔ no general formula

- weak nonlin → follows linearization
- geometrical formulation (Lie brackets)

7) Condition # of $W_c = \frac{\max|\lambda|}{\min|\lambda|} \sim e^n$

$\Rightarrow$ hard to decide and control for $n \gg 1$

8) Control Gramian: $X_{imp}(t) = e^{At} B$

$P(t) = |X_{imp}\rangle\langle X_{imp}| = \int_0^T dt\, e^{At} B B^T e^{A^T t}$

controllability ↔ $P(t)$ invertible ; eigs ↔ controllability

9) structured controllability "SC" (Lin, 1974)

structure $\{A, B\}$ ← set of zero elements

SC → $\exists\ \{A', B'\}$ w/ struct $\{A, B)$ that is controllable

→ controllable for almost all parameters

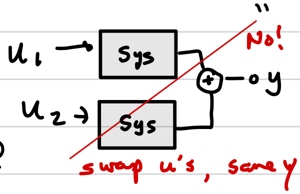
10) Inverse problem: Given A, find B so $\{A, B\}$ controll.

- need to try 2^n combos of B for $x \in \mathbb{R}^n$
- YY Liu + A.-L. Barabasi 2011 → A ↔ network

→ graph-theory techniques → poly. time ...

II. Observability and Duality

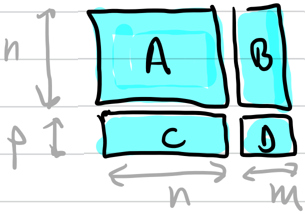
controllability: input u "coupled" to entire state vec. x
observability: output y "



- to control need to set $u(t)$ over interval
- to observe, need to measure $y(t)$ over interval

test: $$W_o = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix}$$

full rank



- can derive directly, but here: duality

$W_c = (B\ AB\ A^2B \cdots A^{n-1}B)$ is same as W_o if $A \to -A^T$, $B \to C^T$

adjoint operator (int. by parts)

Notice: $\dot{x} = -A^T x + C^T u$ $\quad y = B^T x$

minus sign on A doesn't affect rank W_c

so $W_c = (C^T\ A^TC^T\ (A^T)^2 C^T \cdots) = W_o^T$

$u(t)$: affects state $x(t)$ at future times

$y(t)$: determines $x(t)$ based on past obs.

> "We can know the past but not control it;
> we can control the future but do not know it."
> ~ Shannon, 1959

Part 2: Observers, controller design

Control based on the state

Divide control into two problems

1) given $x(t)$; choose $u[x(t)]$
2) estimate $\hat{x}(t)$ from past observations $y|_0^t$

$\Big\}\ u[\hat{x}(t)]$

Claim: If $\{A, B\}$ is controllable, then the closed-loop dyn. based on $u = -Kx$ $[\dot{x} = (A - BK)x \equiv A'x]$ will have n eigenvalues (poles) that can be chosen freely using the n gains in row vector K [SISO case!]

Sketch of proof:

$$A = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \\ 0 & 0 & \cdots & & 1 \\ -a_n & -a_{n-1} & \cdots & & -a_1 \end{pmatrix} \qquad B = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$

"Controller canonical form"

$\dot{x}_1 = x_2$, $\dot{x}_2 = x_3$, ..., $\dot{x}_n = -a_n x_1 - a_{n-1} x_2 - \cdots a_1 x_1 + u$

Let $u = -Kx = -(k_1 \; k_2 \; \cdots \; k_n)$ $\quad BK = \begin{pmatrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ k_1 & \cdots & k_n \end{pmatrix}$

$$A' = \begin{pmatrix} 0 & 1 & 0 & \cdots \\ & & 1 & \\ & & & \ddots \\ -a_n - k_1 & -a_{n-1} k_2 & & 1 \end{pmatrix}$$

$a_i \rightarrow a_i - k_{n-i+1}$

Thus we see that the dynamics of A' are set K
↪ arb. eigs!

Full-State Control

Ex: Damped harmonic osc. $\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix} = \begin{pmatrix}0 & 1\\ -1 & -2\zeta\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix} + \begin{pmatrix}0\\ 1\end{pmatrix}u$

$$u = -\begin{pmatrix}k_1 & k_2\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix} = -\underset{P}{k_1} x_1 - \underset{D}{k_2} x_2$$

$$BK = \begin{pmatrix}0\\ 1\end{pmatrix}\begin{pmatrix}k_1 & k_2\end{pmatrix} = \begin{pmatrix}0 & 0\\ k_1 & k_2\end{pmatrix} \qquad A' = A - BK = \begin{pmatrix}0 & 1\\ -1-k_1 & -2\zeta - k_2\end{pmatrix}$$

eigs: $|s\mathbb{I} - A'| = \begin{vmatrix} s & -1\\ 1+k_1 & s+2\zeta+k_2\end{vmatrix} = s^2 + (2\zeta + k_2)s + (1+k_1) = 0$

choose k_1, k_2 to put eigs anywhere

eg closed-loop poles at $s = (-2, -2)$

$(s+2)^2 = s^2 + 4s + 4 \Rightarrow k_2 = 4 - 2\zeta \quad k_1 = 3$

or $s = (-a, -a) \Rightarrow k_1 = a^2 - 1 \quad k_2 = 2(a - \zeta)$

Notice how gains increase with a (how far poles are moved)

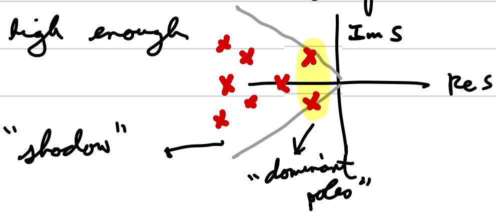
- In general: use control software to find gain K given $\{A, B,$ desired pole positions$\}$
- n poles is a lot of choice! ☹

1) high gains inject noise (and u may be too big)

2) unstable system ⇒ <u>must</u> move RHP poles → LHP

So: Move as few poles as possible as little as possible to stabilize, get ω_c high enough

(order by ζ...)



Output control

- if $\{A, C\}$ is observable, then $y(t \le t_0) \to \hat{x}(t_0)$
- naive differentiation too noisy

Observer

$y \to \hat{x} \to u(\hat{x})$ "synchronize"

Naive obs.

$\dot{x} = Ax + Bu$ $e \equiv x - \hat{x}$ (estimation err.)

$\dot{\hat{x}} = A\hat{x} + Bu$ assume we know A, B!

subtract ⇒ $\dot{e} = Ae$ inputs cancel!

No good ☹ i) if A is unstable, $e \to \infty$

ii) need $e \to 0$ faster than A dynamics

Observer

feedback on deviations betw. observations y and predictions $\hat{y} = C\hat{x}$

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

$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \underbrace{C\hat{x}}_{\hat{y}})$ ⇒ $\dot{e} = (A - LC)e$

L = observer gains ⟷ $A' = A - LC$ obs. poles.

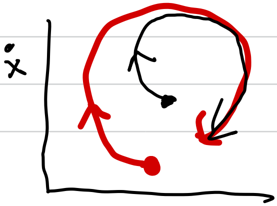
Ex: Harmonic osc. $A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$ $C = (1\ 0)$

$L = \begin{pmatrix} l_1 \\ l_2 \end{pmatrix}$ $LC = \begin{pmatrix} l_1 & 0 \\ l_2 & 0 \end{pmatrix}$ $A' = A - LC = \begin{pmatrix} -l_1 & 1 \\ -1-l_2 & 0 \end{pmatrix}$

eigs → $s^2 + l_1 s + (l_2 + 1) = 0$

eg $s = (-2, -2)$ ⟷ $L = \begin{pmatrix} 4 \\ 3 \end{pmatrix}$

↳ a bit slow...



Observer - controller

$u = -K\hat{x} + k_r r$

$$\dot{x} = Ax + B(-K\hat{x} + k_r r), \qquad y = Cx$$

$$\dot{\hat{x}} = A\hat{x} + B(-K\hat{x} + k_r r) + L(y - C\hat{x})$$

$$\dot{e} = Ae - LCe = (A-LC)e$$

$$\begin{pmatrix} \dot{x} \\ \dot{e} \end{pmatrix} = \underbrace{\begin{pmatrix} A-BK & BK \\ 0 & A-LC \end{pmatrix}}_{M} \begin{pmatrix} x \\ e \end{pmatrix} + \begin{pmatrix} B k_r \\ 0 \end{pmatrix} r$$

$\hat{x} = x - e$

M is block triangular ⇒ eigs given by

$$\det(sI - A + BK)\,\det(sI - A + LC) = 0$$

→ **Separation principle** (design K, L sep.)
- only for lin sys.

- design separately feedback (K) + observer (L)
- use state estimate $\hat{x}$ for feedback