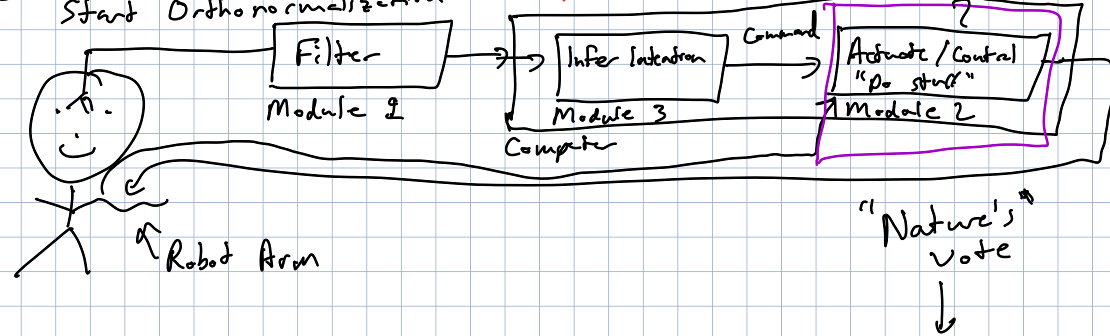
46B Prof Sahai

Midterm Oct 18 7–9 pm

HW Parts Moved today to Bechtel

Today: Finish Controllability
Start Orthonormalization



Discrete model: $\vec{x}[i+1] = A\vec{x}[i] + B\vec{u}[i] + \vec{w}[i]$

Story so far: (1) Can use data to learn the system model using least squares
(2) To achieve our goals using this model, we need to:
(A) Make a plan for controls $\vec{u}[i]$ that achieve our goal
(B) Understand how to reliably execute that plan interactively in the face of disturbances.

Goal: Show that $\text{rank}(B, AB, A^2B, \ldots, A^{n-1}B) = n$ is a reasonable test for controllability.

What do we need? Need to be able to move the state.

$\vec{x}[i+1] = A x[i] + B\vec{u}[i]$ ⟵ ignore disturbance for now.

Can unroll:

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

linear combinations of the columns of ? ↑ our choices

Can see $\text{span}\{B, AB, \ldots, A^{i-1}B\}$ tells us where we can get $\vec{x}[i]$ to go.

If this span were n-dimensional, we could go wherever we want.

Definition A system is controllable if given any target goal $\vec{g}$ we can find a time ℓ s.t. $\exists\ \vec{u}[0], \vec{u}[1], \ldots, \vec{u}[\ell-1]$ so that $\vec{x}[\ell] = \vec{g}$.

How can I check for controllability? First attempt....

Start with $\ell = 1$ $C = B$

if span(C) is n-dimensional, return controllable.
otherwise increment ℓ
Set $C = \{B, AB, \ldots, A^{\ell-1}B\}$
Goto

Try $A = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}$ $B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

Is $\operatorname{span}\left\{\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\}$ 2-dimensional? No

Is $\operatorname{span}\left\{\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 0 \end{bmatrix}\right\}$ 2-dimensional? No

Is $\operatorname{span}\left\{\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \begin{bmatrix} 4 \\ 0 \end{bmatrix}\right\}$ 2-dim? No

$\vdots$

How do stop? Oh no! Infinite loop.

Guess a property: Once span stops growing, it won't grow again

~~Need to state property~~

Thm: If $A^k\vec{b}$ is linearly dependent on $\underbrace{\{\vec{b}, A\vec{b}, \ldots, A^{k-1}\vec{b}\}}$
then $A^{k+1}\vec{b}$ is also linearly dependent on

I know: $A^k\vec{b} = \sum_{i=0}^{k-1} \alpha_i A^i \vec{b}$ for some $\{\alpha_i\}$

I want to prove $\exists\{\beta_i\}$ s.t. $A^{k+1}\vec{b} = \sum_{i=0}^{k-1} \beta_i A^i \vec{b}$

$$A^{k+1}\vec{b} = A(A^k\vec{b}) = A\sum_{i=0}^{k-1}\alpha_i A^i\vec{b}$$
$$= \sum_{i=0}^{k-1}\alpha_i A^{i+1}\vec{b}$$

Annoying

since

$$= \alpha_{k-1}A^k\vec{b} + \sum_{i=1}^{k-1}\alpha_{i-1}A^i\vec{b}$$
$$= \alpha_{k-1}\sum_{i=0}^{k-1}\alpha_i A^i\vec{b} + \sum_{i=1}^{k-1}\alpha_{i-1}A^i\vec{b}$$

$$= \underbrace{\alpha_{k-1}\,\alpha_0}_{\beta_0}\,\vec{b} + \sum_{i=1}^{k-1} \underbrace{(\alpha_{i-1} + \alpha_{k-1}\alpha_i)}_{\beta_i \text{ for } i \geq 1} A^i \vec{b}$$

> Recall how $n = \dim \vec{x}$
> So A is an $n \times n$ matrix

New test for controllability:

Start with $k = 0$ $\quad d = 0$

$C = \{B\}$

If $\dim \operatorname{span}(C) = d$, return NOT CONTROLLABLE

If $\dim \operatorname{span}(C) = n$, return controllable

otherwise

increment k

$d = \dim \operatorname{span}(C)$

$C = \{B, AB, \dots, A^k B]$

Goto

Because n is a finite integer, and the $\dim \operatorname{span}(C)$ grows by at least one in each iteration, we can take at most n iterations before we must stop.

To test for controllability, check $\dim \operatorname{span}[B, AB, \dots, A^{n-1}B]$
if n, Controllable.
if $< n$, Not controllable.

> It turns out to be the case that if a system is controllable, then $\exists$ feedback gains F so that the $A_{CL} = [A + BF]$ has whatever eigenvalues you want.

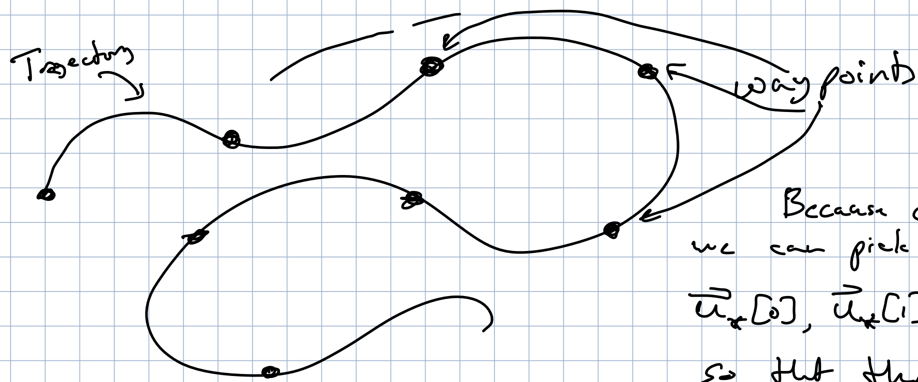
Putting Pieces together

Consider $\vec{x}[i+1] = A\vec{x}[i] + B\vec{u}[i] + \vec{w}[i]$

Goal is to follow a target trajectory "$\vec{x}_*[i]$"

We know system is controllable

e.g. I was given waypoints $\vec{x}_*[0], \vec{x}_*[n], \vec{x}_*[2n], \dots$



Because controllable, we can pick controls $\vec{u}_*[0], \vec{u}_*[1], \ldots$ so that the resulting

$$\vec{x}_*[i+1] = A\vec{x}_*[i] + B\vec{u}_*[i]$$

hits all the way points.

Openloop Trajectory ↗

How can we deal with the disturbance $\vec{w}[i]$?

<u>Idea</u>: Apply the sum of two inputs as our control.

$$\vec{u}[i] = \vec{u}_*[i] + \vec{u}_v[i]$$

To follow plan ↗ ↖ to reject variability due to disturbance

Goal: To keep $\| \underbrace{\vec{x}[i] - \vec{x}_*[i]}_{\text{Call this } \vec{v}[i]} \|$ small.

Subgoal: Design feedback control to keep $\|\vec{v}[i]\|$ small.

i.e. stabilize the system that governs $\vec{v}[i]$.

$$\vec{v}[i+1] = \vec{x}[i+1] - \vec{x}_*[i+1]$$
$$= A\vec{x}[i] + B\vec{u}[i] + \vec{w}[i]$$
$$- A\vec{x}_*[i] - B\vec{u}_*[i]$$

$$\vec{v}[i+1] = A\vec{v}[i] + B\vec{u}_v[i] + \vec{w}[i]$$ ← Dynamics for variation

Can stabilize by appropriate choice of F and applying $\vec{u}_v[i] = F\vec{v}[i]$

$\Longrightarrow A_{cl} = (A + BF) \longleftarrow$ I can pick eigenvalues for A_{cl} to make this stable.

$\Longrightarrow$ If $\|\vec{w}[i]\| \le \epsilon \xRightarrow{\text{Because BIBO}} \|\vec{v}[i]\| \le K\epsilon$

$\Longrightarrow \vec{x}[i]$ follows $\vec{x}_*[i]$ closely.

$\vec{u}[i] = \vec{u}_*[i] + F(\vec{x}[i] - \vec{x}_*[i])$

Closed-loop since it depends on current state.

Done with controllability

Loose ends?? In the course so far.

a) What about stability when we have repeated eigenvalues without enough eigenvectors?
How do we know that we can find a nice basis where $\tilde{A}$ is upper-triangular?

b) What if we don't have $x_*[n], x_*[2n], \ldots$?
What if we just have $x_*[l] = \vec{g}$.
Specific time
How can we plan in a natural way?

c) How can we infer intentions from brain signals? Do classification?

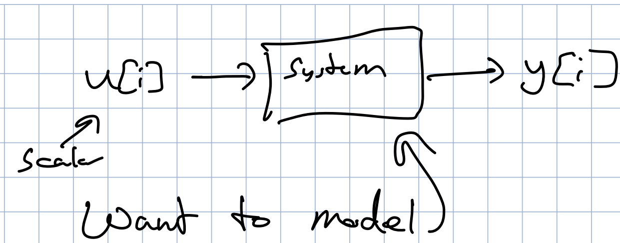
Secret: All the math is connected

Consider Problem (d) $\leftarrow$ How can I efficiently do lots of related least-squares problems?

Chosen to bring out the tool we need

Motivating Example:

Related to HW 6 DemoSystemId problem Example 4.



The model could be: $y[i+1] = b\,u[i]$

or $y[i+1] = b\,u[i] + a_0\,y[i]$

or $y[i+1] = b\,u[i] + a_0\,y[i] + a_1\,y[i-1]$

or $y[i+1] = b\,u[i] + a_0\,y[i] + a_1\,y[i-1] + a_2\,y[i-2]$

or

$\vdots$ $y[i+1] = b\,u[i] + \sum_{j=0}^{k-1} a_j\,y[i-j]$

Called auto-regressive models or Markov of order k.

Want to do System-ID.

a) Try to fit all the models

b) Test them to see which one works best.

Gold Standard: Works in practice

Silver Standard: predicts well on test data.

To do ⓐ, need to solve a family of nested least-squares problems

Approach: $D\,p \simeq S$ ← Set up approximate equations

D: data, p: unknown parameters, S: data.

$$S = \begin{bmatrix} \vdots \\ y[7] \\ y[8] \\ \vdots \end{bmatrix} \qquad p = \begin{bmatrix} b \end{bmatrix} \text{ or } \begin{bmatrix} b \\ a_0 \end{bmatrix} \text{ or } \begin{bmatrix} b \\ a_0 \\ a_1 \end{bmatrix} \cdots\cdots$$

$$D = \underbrace{\begin{bmatrix} \vec{d}_1 & \vec{d}_2 & \cdots & \vec{d}_{k+1} \end{bmatrix}}_{\#\text{ matches dim } p}$$

Other places this can happen:

1) Polynomial fitting.

2) OMP

L.S. Problem 0:

$$\begin{bmatrix} \vec{d}_1 \end{bmatrix} \vec{p}_1 \approx \vec{s}$$

LS Problem 1

$$\begin{bmatrix} \vec{d}_1 & \vec{d}_2 \end{bmatrix} \vec{p}_2 \approx \vec{s}$$

LS Problem 2

$$\begin{bmatrix} \vec{d}_1 & \vec{d}_2 & \vec{d}_3 \end{bmatrix} \vec{p}_3 \approx \vec{s}$$

How do you solve least-squares?

$$D\vec{p} \approx \vec{s} \Longrightarrow \hat{\vec{p}} = (D^T D)^{-1} D^T \vec{s}$$

$k \times k$

Keeps growing.

Inverting takes $O(k^3)$ steps.

Gets painful.

Can we make things faster?