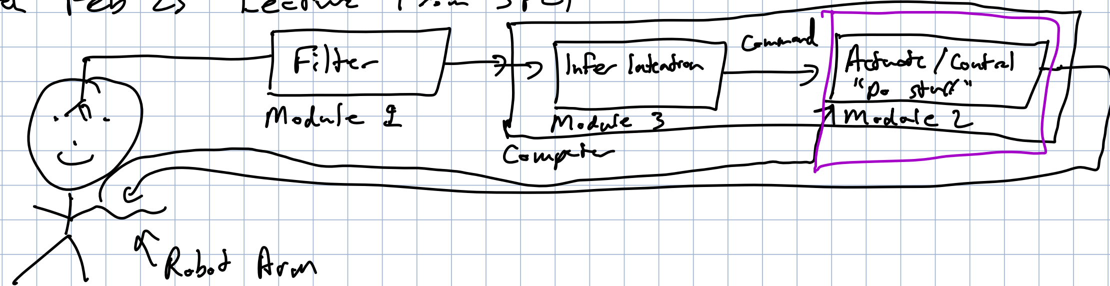
16B 9/30/21 Prof Anant Sahai

Today: Intro to control
Continuous time → discrete-time
System Identification from data.

Midterm: Oct 18^{th}

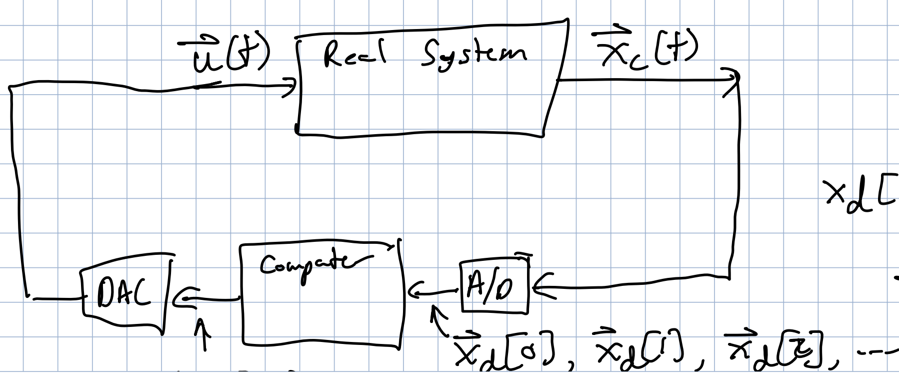
Watch Feb 25th Lecture from SP21



Foundations of Machine learning to interact with a dynamic world.

Modules 1,2,3   1 2 3   1 2 3   1 2 3

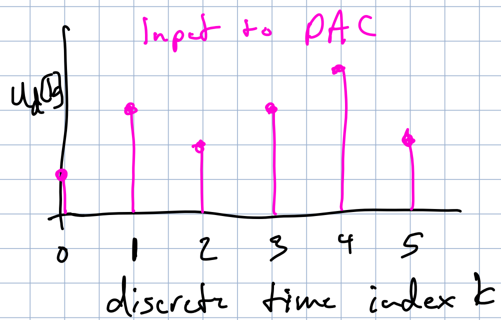
Real World	Computational World "Software"
Continuous time	Discrete time
time flows time infinitely divisible	time "ticks" a clock tick
Differential Equations.	Recurrence Relations
$\frac{d}{dt}\vec{x}(t) = f(\vec{x}(t), \dots)$	$\vec{x}_d[k+1] = g(\vec{x}_d[k], \dots)$

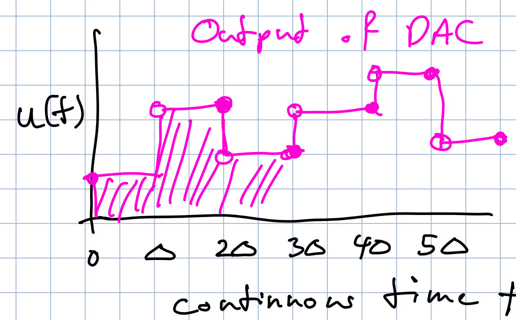


$x_d[k] = x_c(k\Delta)$

sampling

$\vec{u}_d[0], \vec{u}_d[1], \dots$





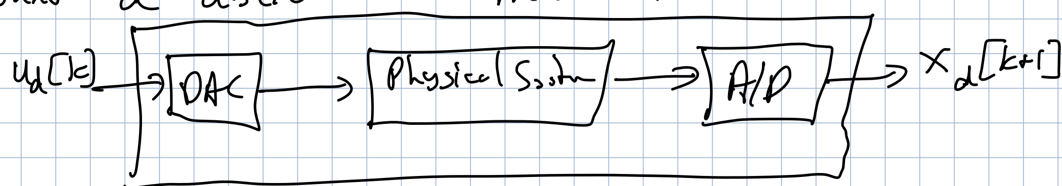
Diff Eq. for Real System

state — Engineer's choice of input — Disturbance

$$\frac{d}{dt}\vec{x}_c(t) = A\,\vec{x}_c(t) + B\,\vec{u}(t) + \vec{w}(t)$$

Nature of the physical system — Nature's input

Want a discrete-time model for



Want / Goal: Discrete-time model:

$$\vec{x}_d[k+1] = A_d\,\vec{x}_d[k] + B_d\,\vec{u}_d[k] + \vec{w}_d[k]$$

Remember Discussion 2B

$$\frac{d}{dt}x(t) = \lambda x(t) + b\,u(t)$$

If you applied controls $u(t) = u_d[i]$ if $t \in (i\Delta, (i+1)\Delta]$ (open, closed)

Then $$x_d[i+1] = e^{\lambda\Delta}\,x_d[i] + b\left(\frac{e^{\lambda\Delta}-1}{\lambda}\right)u_d[i] \quad \text{if } \lambda \neq 0$$

Let's do $\lambda = 0$ ourselves.

$$\frac{d}{dt} x(t) = 0 \cdot x(t) + b\,u(t) = b\,u(t)$$

$$x(t) = x(0) + \int_0^t b\,u(\tau)\,d\tau$$

$$x((i+1)\Delta) = x(0) + b \sum_{k=0}^{i} \Delta\, u_d[k]$$

$$\parallel$$

$$x_d[i+1] = x_d[i] + b\Delta\, u_d[i]$$

In Discussion on Monday, extend to the vector case.

$$A \longrightarrow A_d \qquad \text{given } \Delta$$

$$B \longrightarrow B_d$$

How do we get A_d & B_d if we don't have a Diff Eq?

Learn it from experiments & collected data

System Identification: Given a "trace" of the input/output behavior of a system, figure out the system model.

$$x[k+1] = \lambda\, x[k] + b\, u[k] + \underbrace{w[k]}$$

Don't know

disturbance
I hope it is small
Treat like "noise".

I know $u[k]$ because I chose these

I know $x[k]$ because I measured them.

2 unknowns. How many equations?

Suppose I collect from $k = 0, \ldots, k = T$

I know $x[0], \ldots, x[T]$
I know $u[0], \ldots, u[T-1]$

I have T "equations."

$$\left.\begin{aligned} x[1] &\approx \lambda x[0] + b\,u[0] \\ x[2] &\approx \lambda x[1] + b\,u[1] \\ &\vdots \\ x[T] &\approx \lambda x[T-1] + b\,u[T-1] \end{aligned}\right\} T \text{ "equations"}$$

Write as:

$$\begin{bmatrix} x[0] & u[0] \\ x[1] & u[1] \\ \vdots & \\ x[T-1] & u[T-1] \end{bmatrix} \begin{bmatrix} \lambda \\ b \end{bmatrix} \approx \begin{bmatrix} x[1] \\ \vdots \\ \vdots \\ x[T] \end{bmatrix}$$

Tall Matrix

$$D \vec{p} \approx \vec{s}$$

data, parameters, states.

Least squares solution: $\hat{\vec{p}} = (D^T D)^{-1} D^T \vec{s}$

estimated parameters

Need $D^T D$ to be invertible.

Reasonable if we believe disturbances w are "small"

Because $\hat{\vec{p}}$ minimizes $\| \vec{s} - D\hat{\vec{p}} \|^2$

Vector Case

$\vec{x} \leftarrow n$-dim $\quad \vec{u} \leftarrow m$-dim

Presumed Model: $\vec{x}[t+1] = A\vec{x}[t] + B\vec{u}[t] + \vec{w}[t]$

unknown

$$\begin{array}{cc} & n \\ n & A \end{array} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & & & \\ \vdots & & & \\ a_{n1} & - & - & a_{nn} \end{bmatrix} \qquad B = \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ \vdots & & \vdots \\ b_{n1} & & b_{nm} \end{bmatrix}$$

How many unknowns? $n \cdot n + n \cdot m = n^2 + nm$

Remember: All matrix vector equations are just systems of equations.

$$A = \begin{bmatrix} \vec{a}_1^T \\ \vec{a}_2^T \\ \vdots \\ \vec{a}_n^T \end{bmatrix} \qquad B = \begin{bmatrix} \vec{b}_1^T \\ \vec{b}_2^T \\ \vdots \\ \vec{b}_n^T \end{bmatrix} \leftarrow r^{st} \text{ row}$$

Focus on r^{st} row

$$\vec{a}_r^T = [a_{r1}\ a_{r2}\ \cdots\ a_{rn}] \qquad \vec{b}_r^T = [b_{r1}\ \cdots\ b_{rm}]$$

$$x_r[t+1] \approx \underbrace{\vec{a}_r^T}_{\text{unknowns}} \vec{x}[t] + \vec{b}_r \vec{u}[t]$$

know

Stack unknowns vertically

$$T\left\{\begin{bmatrix} \vec{x}[0]^T & \vec{u}[0]^T \\ & \vdots \\ \vec{x}[T-1]^T & \vec{u}[T-1]^T \end{bmatrix}\right. \begin{array}{l} \begin{bmatrix} \vec{a}_r \\ \vec{b}_r \end{bmatrix} \begin{matrix} n \\ m \end{matrix} \end{array} \approx \begin{bmatrix} x_r[1] \\ x_r[2] \\ \vdots \\ x_r[T] \end{bmatrix}$$

$n + m$

tall matrix $T\ \overset{n+m}{D}$ $\qquad \vec{p}_r\,]^{n+m} \qquad \vec{s}_r\,]\,T$

$$D \vec{p}_r \approx \vec{s}_r$$

Least Squares $\hat{\vec{p}}_r = \underbrace{(D^T D)^{-1} D^T}_{} \vec{s}_r$

No r's here!

$$\hat{P} = (D^T D)^{-1} D^T S$$

$$\begin{bmatrix} \hat{\vec{p}}_1 & \hat{\vec{p}}_2 & \cdots & \hat{\vec{p}}_n \end{bmatrix} = \left(\quad \right) \begin{bmatrix} \vec{s}_1 & \vec{s}_2 & \cdots & \vec{s}_n \end{bmatrix}$$

$$\begin{bmatrix} \vec{x}[1]^T \\ \vdots \\ \vec{x}[T]^T \end{bmatrix}$$

We can solve for all the parameters !! :)