ASEN 5044, Fall 2024

Statistical Estimation for Dynamical Systems

Lecture 04:
Time domain Solutions for LTI Systems:
Matrix Exponential and Properties

Prof. Nisar Ahmed (Nisar.Ahmed@Colorado.edu)

Thursday 09/05/2024

Ann and H.J. Smead
Aerospace Engineering Sciences
UNIVERSITY OF COLORADO BOULDER

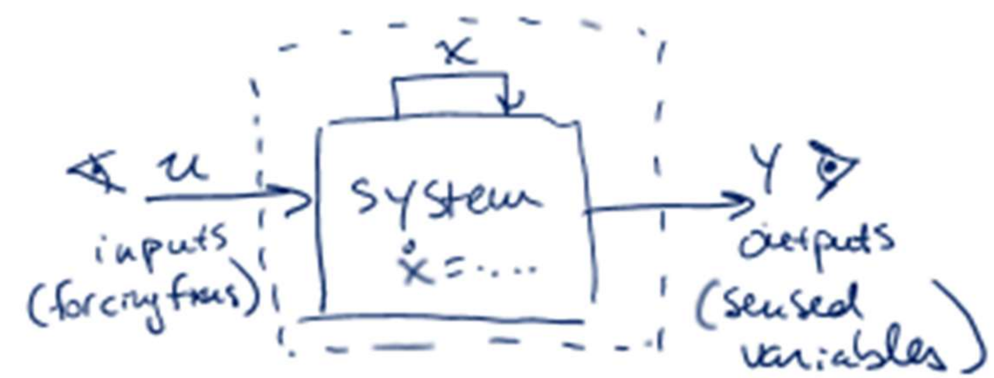
Announcements

- **Quiz 2 will be posted Fri (tomorrow), due Tues 09/10 at 10:00 am**
- **HW 1 Posted: Due Thurs 9/12 at 11:59 pm**
  - **Submit via Gradescope (linked on Canvas) –**
  - **All submissions must be legible!!! – zero credit otherwise**
  - **All submissions must have your name on them!!! – zero credit otherwise**
  - Advanced Question:
    - optional/extra credit (please follow instructions – submit response to designated email address separately from rest of assignment)

Office hours: regular days/times posted, starting next week:
  - **Prof. Ahmed: Wed 4:30-6 pm, AERO N353**
  - **TF Aidan Bagley: Wed 12-1:30 pm, AERO N353**
  - **TF Collin Hudson: Mon 1-2 pm, AERO 303**
  - **TF Jiho Lee: Tues 2:30-3:30 pm, AERO N253**
- **Zoom link for remote office hours participation: use same link as for lectures (posted on Canvas)**

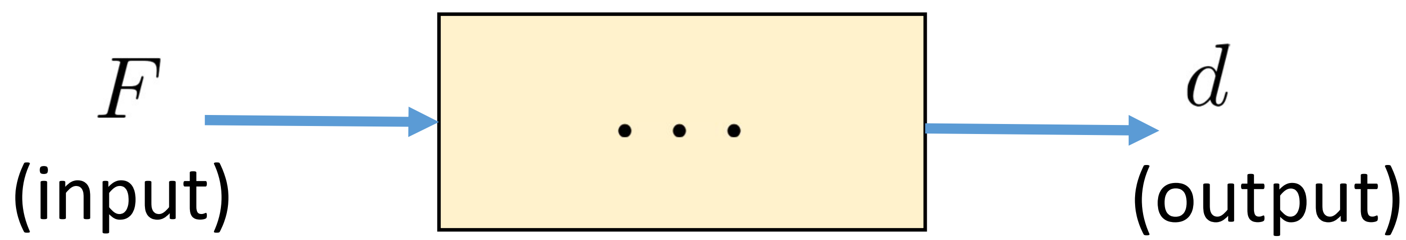
Last Time…

- Introduction to state space models
  - Introduced concept of system “state”
- Review of scalar linear ODEs - behaviors and solutions
  - With and without forcing (input) functions
- Extension to vector-matrix linear ODEs
  - Coupled linear ODEs
  - “State vector”



(Last Time) Example #2: Same Mass, Slightly Different Model

Block Diagram:



$$F = m\dot{v} \text{ (ODE physical law)}$$

Different output:

Now want displacement $d(t)$ as fxn of time and $F(t)$ (input)

Easy to Show:

$$v(t) = \frac{1}{m}\int_{t_0}^{t} F(\tau)\,d\tau + \underline{\underline{v(t_0)}}$$

$$\& \quad d(t) = \int_{t_0}^{t} v(\tau)\,d\tau + \underline{\underline{d(t_0)}}$$

Now Need 2 IC's, hence the state vector is $x(t) = \begin{bmatrix} v(t) \\ d(t) \end{bmatrix}$

→ If we don't keep track of $v(t)$ & $d(t)$ in our $x(t)$, then we cannot uniquely specify output $d(t)$ for given $F(t)$ input! [i.e. we must know both IC's for $d(t)$ to be predicted]

→ System has memory: Need to integrate $F(t)$ to get $v(t)$ & integrate $v(t)$ to get $d(t)$!

(Last Time) Re-arrange Dynamics to Reflect States (want d vs. f)

<u>Since everything here is **linear**, can rewrite all this in **general matrix-vector ODE form**:</u>

$$\dot{x} = \begin{bmatrix} \dot{v}(t) \\ \dot{d}(t) \end{bmatrix} = \begin{bmatrix} 0\cdot v(t) + 0\cdot d(t) + \frac{1}{m}F(t) \\ 1\cdot v(t) + 0\cdot d(t) + 0\cdot F(t) \end{bmatrix}$$

($n=2$, # states)

(where $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} v \\ d \end{bmatrix}$

$\rightarrow \dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{v} \\ \dot{d} \end{bmatrix}$)

$$= \begin{bmatrix} 0\cdot x_1 + 0 x_2 + \frac{1}{m}u(t) \\ 1\cdot x_1 + 0\cdot x_2 + 0\cdot u(t) \end{bmatrix}$$

$$\rightarrow \boxed{\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}}_{n\times n = 2\times 2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{1}{m} \\ 0 \end{bmatrix}}_{n\times m = 2\times 1} u(t), \quad \text{w/ IC } x(0) = \begin{bmatrix} v(0) \\ d(0) \end{bmatrix}}$$

"A" "B" ($m=1$ input)

how states track "internal memory" of dyn. sys.

where output $y(t) = d(t) = x_2$ is given by

gives "snapshot" of system via some operation on some subset of states

$$\boxed{\begin{aligned} y(t) &= 0\cdot x_1 + 1\cdot x_2 + 0\cdot u(t) \\ &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]\, u(t) \end{aligned}}$$

"C" "D"

Linear State space model

Today…

- General linear matrix-vector ODES
  - **Linear time-invariant (LTI) (ABCD) matrix parameterization**
  - Extension: linear time varying (LTV)

- **Matrix exponential** as solution to LTI matrix vector IVP (initial value problem)

READ: Chapter 1.3-1.4 in Simon book

State Space Form of Linear Dynamical Systems

- In general, suppose state variables $x_1,\ldots,x_n$ obey linear ODEs, with scalar u(t) (i.e. m=1 for now)

$$\begin{aligned}\dot{x}_1 &= a_{11}x_1(t) + a_{12}x_2(t) + \cdots + a_{1n}x_n(t) + b_1 u(t) \\ &\vdots \\ \dot{x}_n &= a_{n1}x_1(t) + a_{n2}x_2(t) + \cdots + a_{nn}x_n(t) + b_n u(t)\end{aligned} \Leftrightarrow \boxed{\dot{x}(t) = Ax(t) + Bu(t)}$$

$$\text{where} \quad \dot{x} = \begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} u(t)$$

(m x 1) input vector

"A" (n x n) state dynamics matrix

"x(t)" (n x 1) state vector

"B" (n x m) input-to-state matrix

Suppose also: only linearly sensed outputs are :

$$\begin{aligned} y_1 &= c_{11}x_1(t) + \cdots + c_{1n}x_n(t) + d_1 u(t) \\ &\vdots \\ y_p &= c_{p1}x_1(t) + \cdots + c_{pn}x_n(t) + d_p u(t)\end{aligned} \Leftrightarrow \boxed{y(t) = Cx(t) + Du(t)}$$

(p = # sensed output variables)

$$C = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{p1} & \cdots & c_{pn} \end{bmatrix}, \qquad D = \begin{bmatrix} d_1 \\ \vdots \\ d_p \end{bmatrix}$$

(p x n) output matrix

(p x m) feed-thru matrix

General Linear State Space Models

Can generalize this formulation for *time-varying* or *time-invariant* dynamics:

$$\text{for: } x(t) \in \mathbb{R}^{n\times 1},\ u(t) \in \mathbb{R}^{m\times 1},\ y(t) \in \mathbb{R}^{p\times 1}$$

- **Linear Time Varying (LTV) State Space Model:**

$$\dot{x}(t) = A(t)x(t) + B(t)u(t),$$
$$y(t) = C(t)x(t) + D(t)u(t)$$
$$A(t) \in \mathbb{R}^{n\times n},\ B(t) \in \mathbb{R}^{n\times m},\ C(t) \in \mathbb{R}^{p\times n},\ D(t) \in \mathbb{R}^{p\times m}$$

Matrix elements are functions of time (very powerful representation, but generally tricky to analyze/solve…)

- **Linear Time Invariant (LTI) State Space Model:**

$$\dot{x}(t) = Ax(t) + Bu(t),$$
$$y(t) = Cx(t) + Du(t)$$
$$A \in \mathbb{R}^{n\times n},\ B \in \mathbb{R}^{n\times m},\ C \in \mathbb{R}^{p\times n},\ D \in \mathbb{R}^{p\times m}$$

[A,B,C,D] parameters constant w.r.t. time

Note: if m>1: $u(t) = [u_1(t), \cdots, u_m(t)]^T$

$$B = \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ \vdots & \vdots & \vdots \\ b_{n1} & \cdots & b_{nm} \end{bmatrix},\ D = \begin{bmatrix} d_{11} & \cdots & d_{1m} \\ \vdots & \vdots & \vdots \\ d_{p1} & \cdots & d_{pm} \end{bmatrix}$$

(likewise for LTV…)

Matrix-Vector Initial Value Problems (IVPs)

- **Given SS model for an LTI system (i.e. given its [A,B,C,D] parameters), how do we solve for x(t)?** (vector sol'n over time!)
- Suppose x(0) given, u(t) =0 (no external forcing) and we ignore output y(t)
- Left with a matrix-vector ODE, i.e. a system of linear ODEs with initial conditions

$$\dot{x}(t) = Ax(t), \quad x(0) = \begin{bmatrix} x_1(0) \\ \vdots \\ x_n(0) \end{bmatrix} \longleftrightarrow \begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{matrix} a_{11}x_1(t) + \cdots + a_{1n}x_n(t) \\ \vdots \\ a_{n1}x_1(t) + \cdots + a_{nn}x_n(t) \end{matrix}$$

→ what is the solution for $x(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix}$?

Generally speaking: we need a state transition matrix (STM) $\Phi(t, t_0)$

$$x(t) = \Phi(t,0) \cdot x(0) \quad \text{(or)} \quad x(t) = \Phi(t,t_0) \cdot x(t_0)$$

where the STM $\Phi(t,t_0) \in \mathbb{R}^{n \times n}$ such that

$$\frac{d}{dt}[x(t)] = \frac{d}{dt}[\Phi(t,t_0)x(t_0)] = Ax(t) \quad \text{w/ initial condition } \Phi(t_0,t_0) = I_{n\times n} \text{ (identity matrix)}$$

(*)

Matrix-Vector Initial Value Problems (IVPs)

- If we plug the STM into the original matrix-vector ODE, we get:

$$\dot{x}(t) = \frac{\partial}{\partial t}\left[\Phi(t,t_0)\,x(t_0)\right] = \dot{\Phi}(t,t_0)\,x(t_0) + \Phi(t,t_0)\cancel{\frac{d}{dt}(x(t_0))}^{\,0} \quad (*)$$

But also have $\dot{x}(t) = A\,x(t) = A\left[\underbrace{\Phi(t,t_0)\,x(t_0)}_{=x(t)}\right] \quad (**) \quad$ (w/ IC $\Phi(t_0,t_0) = I$)

→ Equating RHS (right hand side) of $(*)$ & $(**)$ to each other gives:

$$\dot{\Phi}(t,t_0)\,x(t_0) = A\,\Phi(t,t_0)\,x(t_0)$$

→ So we need to solve this matrix ODE: $\boxed{\dot{\Phi}(t,t_0) = A\,\Phi(t,t_0), \text{ w/ IC } \Phi(t_0,t_0) = I}$ $(*)$

(to find $x(t)$)

→ How to solve for $\Phi(t,t_0)$??

→ Consider $n=1$ (simplest case): Scalar $A = a$ → $\dot{\Phi}(t,t_0) = a\,\Phi(t,t_0)$, $\Phi(t_0,t_0) = 1$

→ So clearly: $\Phi(t,t_0) = e^{a(t-t_0)}$

(STM is exponential in a & $\Delta t = t - t_0$)

Now what if $n \geq 1$ more generally?

The STM for LTI Systems: the Matrix Exponential

- Remarkably, the STM for any square LTI matrix A is given by the **matrix exponential**

$$\boxed{\Phi(t,t_0) = e^{A(t-t_0)}} \iff \boxed{\Phi(t,0) = e^{At}}$$

where the matrix exponential function is defined as the infinite series:

$$(*) \quad e^{A(t-t_0)} \triangleq I + A(t-t_0) + \frac{A^2}{2!}(t-t_0)^2 + \frac{A^3}{3!}(t-t_0)^3 + \dots + \frac{A^r}{r!}(t-t_0)^r + \dots$$

$$= \sum_{i=0}^{\infty} A^i \frac{(t-t_0)^i}{i!} \quad \left(\text{where } A^i = A \cdot A \cdot A \cdots A = \prod_{k=1}^{i} A = \text{product of } A \text{ } i \text{ times} \right)$$

(*) This provably converges for any $A \in \mathbb{R}^{n \times n}$ (use Picard iteration → Peano-Baker Series → converges for time invariant A).

→ can easily verify that:

$$\frac{d}{dt}\left[e^{A(t-t_0)}\right] = A + A^2(t-t_0) + A^3\frac{(t-t_0)^2}{2!} + \dots$$

$$= A\underbrace{\left[I + A(t-t_0) + A^2\frac{(t-t_0)^2}{2!} + \dots\right]}$$

$$= A\underbrace{e^{A(t-t_0)}}_{\Phi}$$

Properties of the Matrix Exponential

The matrix exponential function of matrix M is generally defined as:

$$(*) \quad \underset{[n\times n]}{e^{M}} \triangleq \sum_{i=0}^{\infty} \frac{M^i}{i!} = I + M + \frac{M^2}{2!} + \frac{M^3}{3!} + \ldots, \quad \text{for } M \in \mathbb{R}^{n\times n}$$

This maps an arbitrary (n x n) matrix M to another (n x n) matrix.

Matrix exponential has following useful properties:

- Always invertible, even if M itself is singular

$$\text{ie } (e^{M})^{-1} \text{ always exists: } (e^{M})^{-1} = e^{-M} \text{ s.t. } (e^{M})(e^{-M}) = (e^{-M})(e^{M}) = I$$

- Product of two matrix exponentials commutes $\underset{\text{if \& only if}}{\underline{\text{iff}}}$ input matrices commute

$$\Phi \in \mathbb{R}^{n\times n}, \; \Psi \in \mathbb{R}^{n\times n} \longrightarrow e^{\Phi} \cdot e^{\Psi} = e^{\Phi+\Psi} = e^{\Psi+\Phi} \underset{\text{iff}}{\Longleftrightarrow} \Phi\Psi = \Psi\Phi$$

Computing the Matrix Exponential/STM

The matrix exponential is the STM for LTI state space initial value problems:

$$\dot{x} = Ax, \quad x(t_0) = x_0 \longleftrightarrow x(t) = \Phi(t, t_0) \cdot x(t_0)$$

$$\text{where } \Phi(t, t_0) = e^{A(t-t_0)} = \sum_{i=0}^{\infty} A^i \frac{(t-t_0)^i}{i!}$$

The STM is extremely useful for doing computer simulations of LTI systems --

but how to actually compute an infinite series of matrix powers?

- Brute force: truncated series, or lucky properties of matrix
- Eigenvalue/Jordan decomposition
- Laplace transforms
- *Cayley-Hamilton theorem*
- **Matlab: "expm"** command

→ for any matrix A: $|A - \lambda I| = \lambda^n + c_{n-1}\lambda^{n-1} + \dots + c_1\lambda + c_0 = 0$

= char. equ. for $n \times n$ matrix A

→ C-H theorem: says:

$$A^n + c_{n-1}A^{n-1} + \dots + c_1 A + c_0 \cdot I = 0 \text{ (matrix)}$$

ie every square matrix satisfies its own char. equ!

→ So what? This means $\sum_{i=0}^{\infty} A^i = \sum_{i=0}^{n-1} q_i \cdot A^i$ for $q_i \in \mathbb{R}$

(ie all matrix powers $i \geq n$ are just linear combos of lower powers!)

Example: STM Computation for 1D Mass System IVP

- Recall: LTI SS model for displacement d(t) vs. t (let force F(t)=0)

$\rightarrow F = 0$

$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} v \\ d \end{bmatrix}$

$\dot{x} = \begin{bmatrix} \dot{v} \\ \dot{d} \end{bmatrix} = Ax$

$A = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \longrightarrow$ so what is $\Phi(t, t_0) = ?$

$\longrightarrow$ Suppose $t - t_0 \triangleq \Delta t$ for some constant t_0

$\longrightarrow \Phi(t, t_0) = e^{A\Delta t} = \sum_{i=0}^{\infty} A^i \frac{(\Delta t)^i}{i!} \longrightarrow$ we know that: $i=0: A^0 = I$

$i=1: A^1 = A$

$\longrightarrow$ for $i=2$: $A^2 = A \cdot A = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

for $i=3$: $A^3 = A \cdot A \cdot A = A^2 \cdot A = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

$\Rightarrow A^i = 0$ for all $i \geq 2$ (!!)

$\longrightarrow \Phi(t, t_0) = e^{A\Delta t} = A^0 \frac{(\Delta t)^0}{0!} + A^1 \frac{(\Delta t)^1}{1!} + 0 \cdots + 0 = I + A\Delta t = \boxed{\begin{bmatrix} 1 & 0 \\ \Delta t & 1 \end{bmatrix} = \Phi(t, t_0)}$

$\longrightarrow$ So: $x(t) = x(t_0 + \Delta t) = \Phi(t, t_0)\, x(t_0) = \Phi(t, t_0) \begin{bmatrix} v(t_0) \\ d(t_0) \end{bmatrix}$

$= \begin{bmatrix} v(t_0) \\ v(t_0)\Delta t + d(t_0) \end{bmatrix}$ ✓