Module 09
From s-Domain to time-domain
From ODEs, TFs to State-Space — Modern Control

Ahmad F. Taha

EE 3413: Analysis and Desgin of Control Systems

Email: ahmad.taha@utsa.edu

Webpage: http://engineering.utsa.edu/ataha

January 20, 2021

Modern Control

- Readings: 9.1–9.4 Ogata; 3.1–3.3 Dorf & Bishop
- In the previous modules, we discussed the analysis and design of control systems via frequency-domain techniques
  - Root locus, PID controllers, compensators, state-feedback control, etc...
  - These studies are considered as the classical control theory—based on the s-domain
- This module: we'll introduce time-domain techniques
  - Theory is based on *State-Space Representations*—modern control
- Why do we need that? Many reasons

ODEs & Transfer Functions

- For linear systems, we can often represent the system dynamics through an nth order ordinary differential equation (ODE):

$$y^{(n)}(t) + a_1 y^{(n-1)}(t) + a_2 y^{(n-2)}(t) + \cdots + a_{n-1}\dot{y}(t) + a_n y(t) =$$

$$b_0 u^{(n)}(t) + b_1 u^{(n-1)}(t) + b_2 u^{(b-2)}(t) + \cdots + b_{n-1}\dot{u}(t) + b_n u(t)$$

- The $y^{(k)}$ notation means we're taking the kth derivative of $y(t)$
- Input: $u(t)$; Output: $y(t)$—What if we have MIMO system?
- Given that ODE description, we can take the LT (assuming zero initial conditions for all signals):

$$\boxed{H(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_{n-1}s + b_n}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n}}$$

ODEs & TFs

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_{n-1}s + b_n}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n}$$

- This equation represents relationship between one system input and one system output
- This relationship, however, does not show me the internal states of the system, nor does it explain the case with multi-input system
- For that (and other reasons), we discuss the notion of **system state**
- **Definition:** $\boldsymbol{x}(t)$ is a state-vector that belongs to $\mathbb{R}^n$: $\boldsymbol{x}(t) \in \mathbb{R}^n$
- $\boldsymbol{x}(t)$ is an internal state of a system
- Examples: voltages and currents of circuit components

ODEs, TFs to State-Space Representations

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_{n-1} s + b_n}{s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n}$$

- **State-space** (SS) theory: representing the above TF of a system by a **vector-form first order ODE**:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \quad \boldsymbol{x}_{\text{initial}} = \boldsymbol{x}_{t_0}, \tag{1}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t), \tag{2}$$

  - $\boldsymbol{x}(t) \in \mathbb{R}^n$: **dynamic state-vector of the LTI system**, $\boldsymbol{u}(t)$: **control input-vector**, $n =$ order of the TF/ODE
  - $\boldsymbol{y}(t)$: output-vector and $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$ are constant matrices
  - For the above transfer function, we have one input $U(s)$ and one output $Y(s)$, hence the size of $y(t)$ and $u(t)$ is only one (scalars)
- **Module Objectives:** learn how to construct matrices $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$ given a transfer function

State-Space Representation 1

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_{n-1} s + b_n}{s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n}$$

- Given the above TF/ODE, we want to find

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{aligned}$$

- The above two equations represent a relationship between the input and output of the system via the internal system states
- The above 2 equations are nothing but a first order differential equation
- Wait, WHAT? But the TF/ODE was an nth order ODE. How do we have a **first order ODE** now?
- Well, because this equation is vector-matrix equation, whereas the ODE/TF was a scalar equation
- Next, we'll learn how to get to these 2 equations from any TF

State-Space Representation 2 [Ogata, P. 689]

$$\frac{Y(s)}{U(s)} = b_0 + \frac{(b_1 - a_1 b_0)s^{n-1} + \cdots + (b_{n-1} - a_{n-1}b_0)s + (b_n - a_n b_0)}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n}$$

which can be modified to

$$Y(s) = b_0 U(s) + \hat{Y}(s) \tag{9–71}$$

where

$$\hat{Y}(s) = \frac{(b_1 - a_1 b_0)s^{n-1} + \cdots + (b_{n-1} - a_{n-1}b_0)s + (b_n - a_n b_0)}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n} U(s)$$

Let us rewrite this last equation in the following form:

$$\frac{\hat{Y}(s)}{(b_1 - a_1 b_0)s^{n-1} + \cdots + (b_{n-1} - a_{n-1}b_0)s + (b_n - a_n b_0)} = \frac{U(s)}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n} = Q(s)$$

From this last equation, the following two equations may be obtained:

$$s^n Q(s) = -a_1 s^{n-1} Q(s) - \cdots - a_{n-1} s Q(s) - a_n Q(s) + U(s) \tag{9–72}$$

$$\hat{Y}(s) = (b_1 - a_1 b_0)s^{n-1} Q(s) + \cdots + (b_{n-1} - a_{n-1}b_0) s Q(s) + (b_n - a_n b_0) Q(s) \tag{9–73}$$

State-Space Representation 3 [Ogata, P. 689]

Now define state variables as follows:

$$
\begin{aligned}
X_1(s) &= Q(s) \\
X_2(s) &= sQ(s) \\
&\cdot \\
&\cdot \\
&\cdot \\
X_{n-1}(s) &= s^{n-2}Q(s) \\
X_n(s) &= s^{n-1}Q(s)
\end{aligned}
$$

Then, clearly,

$$
\begin{aligned}
sX_1(s) &= X_2(s) \\
sX_2(s) &= X_3(s) \\
&\cdot \\
&\cdot \\
&\cdot \\
sX_{n-1}(s) &= X_n(s)
\end{aligned}
$$

State-Space Representation 4 [Ogata, P. 689]

which may be rewritten as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \end{aligned} \qquad (9\text{–}74)$$

Noting that $s^nQ(s) = sX_n(s)$, we can rewrite Equation (9–72) as

$$sX_n(s) = -a_1X_n(s) - \cdots - a_{n-1}X_2(s) - a_nX_1(s) + U(s)$$

or

$$\dot{x}_n = -a_nx_1 - a_{n-1}x_2 - \cdots - a_1x_n + u \qquad (9\text{–}75)$$

Also, from Equations (9–71) and (9–73), we obtain

$$\begin{aligned} Y(s) &= b_0U(s) + (b_1 - a_1b_0)s^{n-1}Q(s) + \cdots + (b_{n-1} - a_{n-1}b_0)sQ(s) \\ &\quad + (b_n - a_nb_0)Q(s) \\ &= b_0U(s) + (b_1 - a_1b_0)X_n(s) + \cdots + (b_{n-1} - a_{n-1}b_0)X_2(s) \\ &\quad + (b_n - a_nb_0)X_1(s) \end{aligned}$$

The inverse Laplace transform of this output equation becomes

$$y = (b_n - a_nb_0)x_1 + (b_{n-1} - a_{n-1}b_0)x_2 + \cdots + (b_1 - a_1b_0)x_n + b_0u \qquad (9\text{–}76)$$

Final Solution

- Combining equations (9-74,75,76), we can obtain the following **vector-matrix first order differential equation:**

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_{n-1}(t) \\ \dot{x}_n(t) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{bmatrix}}_{\boldsymbol{Ax}(t)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t)}_{\boldsymbol{B}u(t)}$$

$$y(t) = \underbrace{\begin{bmatrix} b_n - a_n b_0 | & b_{n-1} - a_{n-1} b_0 | & \cdots | & b_1 - a_1 b_0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{bmatrix}}_{\boldsymbol{Cx}(t)} + \underbrace{b_0 u(t)}_{Du(t)}$$

Remarks

- For any TF with order n (order of the denominator), with one input and one output:
  - $\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times 1}, \boldsymbol{C} \in \mathbb{R}^{1\times n}, \boldsymbol{D} \in \mathbb{R}$
  - Above matrices are constant $\Rightarrow$ system is linear **time-invariant** (LTI)
  - If one term of the TF/ODE (i.e., the a's and b's) change as a function of time, the matrices derived above will also change in time $\Rightarrow$ system is linear **time-varying** (LTV)
- The above state-space form is called the *controllable canonical form*
- You can come up with different forms of $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$ matrices given a different transformation

Example 1

- Find a state-space representation (i.e., the state-space matrices) for the system represented by this second order transfer function:

$$\frac{Y(s)}{U(s)} = \frac{s+3}{s^2+3s+2}$$

- **Solution:** look at the previous slides with the matrices:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_{n-1}s + b_n}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n} = \frac{\overbrace{0}^{b_0} s^2 + \overbrace{1}^{b_1} s + \overbrace{3}^{b_2}}{s^2 + \underbrace{3}_{a_1} s + \underbrace{2}_{a_2}}$$

  – First, $n = 2 \Rightarrow \boldsymbol{A} \in \mathbb{R}^{2\times 2}, \boldsymbol{B} \in \mathbb{R}^{2\times 1}, \boldsymbol{C} \in \mathbb{R}^{1\times 2}, \boldsymbol{D} \in \mathbb{R}$

$$\dot{\boldsymbol{x}}(t) = \underbrace{\begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}}_{\boldsymbol{A}} \boldsymbol{x}(t) + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{\boldsymbol{B}} u(t)$$

$$y(t) = \underbrace{\begin{bmatrix} 3 & 1 \end{bmatrix}}_{\boldsymbol{C}} x(t) + \underbrace{0}_{D} u(t)$$

Other State-Space Forms Given a TF/ODE[1]

Observable Canonical Form:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \cdot \\ \cdot \\ \cdot \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 & -a_n \\ 1 & 0 & \cdots & 0 & -a_{n-1} \\ \cdot & \cdot & & \cdot & \cdot \\ \cdot & \cdot & & \cdot & \cdot \\ \cdot & \cdot & & \cdot & \cdot \\ 0 & 0 & \cdots & 1 & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} + \begin{bmatrix} b_n - a_n b_0 \\ b_{n-1} - a_{n-1} b_0 \\ \cdot \\ \cdot \\ \cdot \\ b_1 - a_1 b_0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{n-1} \\ x_n \end{bmatrix} + b_0 u$$

[1] Derivation from Ogata, but similar to the controllable canonical form.

Other State-Space Forms Given a TF/ODE

Diagonal Canonical Form[2]:

$$\frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_{n-1} s + b_n}{(s+p_1)(s+p_2)\cdots(s+p_n)}$$

$$= b_0 + \frac{c_1}{s+p_1} + \frac{c_2}{s+p_2} + \cdots + \frac{c_n}{s+p_n}$$

$$\Downarrow \qquad \Downarrow \qquad \Downarrow$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \cdot \\ \cdot \\ \cdot \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} -p_1 & & & & & 0 \\ & -p_2 & & & & \\ & & \cdot & & & \\ & & & \cdot & & \\ & & & & \cdot & \\ 0 & & & & & -p_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix} u$$

$$y = \begin{bmatrix} c_1 & c_2 & \cdots & c_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} + b_0 u$$

[2] This factorization assumes that the TF has only distinct real poles.

Example 1 Solution for other Canonical Forms

- Find the observable and diagonal forms for

$$\frac{Y(s)}{U(s)} = \frac{\overbrace{0}^{b_0} s^2 + \overbrace{1}^{b_1} s + \overbrace{3}^{b_2}}{s^2 + \underbrace{3}_{a_1} s + \underbrace{2}_{a_2}}$$

- **Solution:** look at the previous slides with the constructed state-space matrices:

  - **Observable Canonical Form:**

$$\dot{\boldsymbol{x}}(t) = \underbrace{\begin{bmatrix} 0 & -2 \\ 1 & -3 \end{bmatrix}}_{\boldsymbol{A}} \boldsymbol{x}(t) + \underbrace{\begin{bmatrix} 3 \\ 1 \end{bmatrix}}_{\boldsymbol{B}} u(t), \quad y(t) = \underbrace{\begin{bmatrix} 0 & 1 \end{bmatrix}}_{\boldsymbol{C}} x(t) + \underbrace{0}_{D} u(t)$$

  - **Diagonal Canonical Form:**

$$\dot{\boldsymbol{x}}(t) = \underbrace{\begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}}_{\boldsymbol{A}} \boldsymbol{x}(t) + \underbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}_{\boldsymbol{B}} u(t), \quad y(t) = \underbrace{\begin{bmatrix} 2 & -1 \end{bmatrix}}_{\boldsymbol{C}} x(t) + \underbrace{0}_{D} u(t)$$

State-Space to Transfer Functions

- Given a state-space representation:

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)\\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t)\end{aligned}$$

can we obtain the transfer function back? **Yes**:

$$\boxed{\frac{Y(s)}{U(s)} = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}}$$

- **Example**: find the TF corresponding for this SISO system:

$$\dot{\boldsymbol{x}}(t) = \underbrace{\begin{bmatrix}-1 & 0\\ 0 & -2\end{bmatrix}}_{\boldsymbol{A}}\boldsymbol{x}(t) + \underbrace{\begin{bmatrix}1\\1\end{bmatrix}}_{\boldsymbol{B}}u(t), \quad y(t) = \underbrace{\begin{bmatrix}2 & -1\end{bmatrix}}_{\boldsymbol{C}}x(t) + \underbrace{0}_{D}u(t)$$

- **Solution:**

$$\frac{Y(s)}{U(s)} = \boldsymbol{C}(s\boldsymbol{I}_n - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D} = \begin{bmatrix}2 & -1\end{bmatrix}\left(s\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} - \begin{bmatrix}-1 & 0\\ 0 & -2\end{bmatrix}\right)^{-1}\begin{bmatrix}1\\1\end{bmatrix} + 0$$

$$= \frac{s+3}{s^2+3s+2},$$ that's the TF from the previous example!

MATLAB Commands

- `ss2tf(A,B,C,D,iu)`
- `tf2ss(num,den)`

Important Remarks

- So why do we want to go from a transfer function to a time-representation, ODE form of the system?
- There are many benefits for doing so, such as:
  1. Stability analysis for MIMO systems becomes way easier
  2. We have powerful mathematical tools that helps us design controllers
  3. RL and compensator designs were relatively tedious design problems
  4. With state-space representations, we can easily design controllers
  5. Nonlinear systems: cannot use TFs for nonlinear systems
  6. State-space is all about time-domain analysis, which is far more intuitive than frequency-domain analysis
  7. With Laplace transforms and TFs, we had to take inverse Laplace transforms. In many cases, the Laplace transform does not exist, which means time-domain analysis is the only way to go
- We will learn how to get a solution for $y(t)$ for any given $u(t)$ from the state-space representation of the system without Laplace transform—via ODE solutions for matrix-vector equations
- Before that, we need to introduce some linear algebra preliminaries

Linear Algebra Revision

Eigenvalues/Eigenvectors of a matrix

- Evalues/vectors are **only defined for square[3] matrices**
- For a matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, we always have n evalues/evectors
  - Some of these evalues might be distinct, real, repeated, imaginary
  - To find evalues($\boldsymbol{A}$), solve this equation ($\boldsymbol{I}_n$: identity matrix of size n)

$$\det(\lambda \boldsymbol{I}_n - A) = 0 \text{ or } \det(\boldsymbol{A} - \lambda \boldsymbol{I}_n) = 0 \Rightarrow a_0\lambda^n + a_1\lambda^{n-1} + \cdots + a_n = 0$$

- **Example**: $\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc.$
- **Eigenvectors**: A number λ and a non-zero vector $\boldsymbol{v}$ satisfying

$$\boldsymbol{A}\boldsymbol{v} = \lambda \boldsymbol{v} \Rightarrow (\boldsymbol{A} - \lambda \boldsymbol{I}_n)\boldsymbol{v} = 0$$

  are called an eigenvalue and an eigenvector of $\boldsymbol{A}$
  - λ is an eigenvalue of an $n \times n$-matrix $\boldsymbol{A}$ if and only if $\lambda \boldsymbol{I}_n - \boldsymbol{A}$ is not invertible, which is equivalent to

$$\det(\boldsymbol{A} - \lambda \boldsymbol{I}_{\mathrm{n}}) = 0.$$

[3] A square matrix has equal number of rows and columns.

Matrix Inverse

- Inverse of a generic 2by2 matrix:

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

  - Notice that $\boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{A}\boldsymbol{A}^{-1} = I_2$
- Inverse of a generic 3by3 matrix:

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A & B & C \\ D & E & F \\ G & H & I \end{bmatrix}^{T} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A & D & G \\ B & E & H \\ C & F & I \end{bmatrix}$$

$$\begin{array}{lll} A = (ei - fh) & D = -(bi - ch) & G = (bf - ce) \\ B = -(di - fg) & E = (ai - cg) & H = -(af - cd) \\ C = (dh - eg) & F = -(ah - bg) & I = (ae - bd) \end{array}$$

$$\det(\mathbf{A}) = aA + bB + cC.$$

  - Notice that $\boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{A}\boldsymbol{A}^{-1} = I_3$

Linear Algebra — Example 1

- Find the eigenvalues, eigenvectors, and inverse of matrix

$$\boldsymbol{A} = \begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix}$$

  - Eigenvalues: $\lambda_{1,2} = 5, -2$
  - Eigenvectors: $\boldsymbol{v}_1 = \begin{bmatrix} 1 & 1 \end{bmatrix}^\top, \boldsymbol{v}_2 = \begin{bmatrix} -\frac{4}{3} & 1 \end{bmatrix}^\top$
  - Inverse: $\boldsymbol{A}^{-1} = -\frac{1}{10} \begin{bmatrix} 2 & -4 \\ -3 & 1 \end{bmatrix}$

- Write $\boldsymbol{A}$ in the matrix **diagonal transformation,** i.e., $\boldsymbol{A} = \boldsymbol{TDT}^{-1}$ where $\boldsymbol{D}$ is the diagonal matrix containing the eigenvalues of $\boldsymbol{A}$:

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix} \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix}^{-1}$$

  - **Only valid for matrices with distinct, real eigenvalues**

Rank of a Matrix

- Rank of a matrix: rank($\boldsymbol{A}$) is equal to the number of linearly independent rows or columns
  - **Example 1:** rank $\left(\begin{bmatrix} 1 & 1 & 0 & 2 \\ -1 & -1 & 0 & -2 \end{bmatrix}\right) = ?$
  - **Example 2:** rank $\left(\begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix}\right) = ?$
- Rank computation: reduce the matrix to a simpler form, generally row echelon form, by elementary row operations
  - **Example 2 Solution:**

$$\begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix} \rightarrow 2r_1 + r_2 \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 3 & 5 & 0 \end{bmatrix} \rightarrow -3r_1 + r_3 \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 0 & -1 & -3 \end{bmatrix}$$

$$\rightarrow r_2 + r_3 \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 0 \end{bmatrix} \rightarrow -2r_2 + r_1 \begin{bmatrix} 1 & 0 & -5 \\ 0 & 1 & 3 \\ 0 & 0 & 0 \end{bmatrix} \Rightarrow rank(\boldsymbol{A}) = 2$$

- **For a matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$:** rank($\boldsymbol{A}$) $\leq$ min(m, n)

Null Space of a Matrix

- The Null Space of any matrix $\boldsymbol{A}$ is the subspace $\mathcal{K}$ defined as follows:

$$\mathrm{N}(\boldsymbol{A}) = \mathrm{Null}(\boldsymbol{A}) = \ker(\boldsymbol{A}) = \{\boldsymbol{x} \in \mathcal{K} | \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}\}$$

- Null($\boldsymbol{A}$) has the following three properties:
  - Null($\boldsymbol{A}$) always contains the zero vector, since $\boldsymbol{A}\boldsymbol{0} = \boldsymbol{0}$
  - If $\boldsymbol{x} \in \mathrm{Null}(\boldsymbol{A})$ and $\boldsymbol{y} \in \mathrm{Null}(\boldsymbol{A})$, then $\boldsymbol{x} + \boldsymbol{y} \in \mathrm{Null}(\boldsymbol{A})$
  - If $\boldsymbol{x} \in \mathrm{Null}(\boldsymbol{A})$ and c is a scalar, then $c\boldsymbol{x} \in \mathrm{Null}(\boldsymbol{A})$
- **Example:** Find N($\boldsymbol{A}$)

$$\boldsymbol{A} = \begin{bmatrix} 2 & 3 & 5 \\ -4 & 2 & 3 \end{bmatrix} \Rightarrow \begin{bmatrix} 2 & 3 & 5 \\ -4 & 2 & 3 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow \left[\begin{array}{ccc|c} 2 & 3 & 5 & 0 \\ -4 & 2 & 3 & 0 \end{array}\right] \Rightarrow$$

$$\left[\begin{array}{ccc|c} 1 & 0 & 1/16 & 0 \\ 0 & 1 & 13/8 & 0 \end{array}\right] \Rightarrow a = -\frac{1}{16}c, b = -\frac{13}{8}c \;\Rightarrow \boxed{\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \alpha \begin{bmatrix} -1/16 \\ -13/8 \\ 1 \end{bmatrix} = \tilde{\alpha} \begin{bmatrix} -1 \\ -26 \\ 16 \end{bmatrix}}$$

Linear Algebra — Example 2

- Find the determinant, rank, and null-space set of this matrix:

$$\boldsymbol{B} = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 7 & 8 \end{bmatrix}$$

  - $\det(\boldsymbol{B}) = 0$
  - $\text{rank}(\boldsymbol{B}) = 2$
  - $\text{null}(\boldsymbol{B}) = \alpha \begin{bmatrix} 3 \\ -2 \\ 1 \end{bmatrix}, \forall\ \alpha \in \mathbb{R}$
- Is there a relationship between the determinant and the rank of a matrix?
  - Yes! Matrix drops rank if determinant = zero $\Rightarrow$ 1 zero evalue
- True or False?
  - $\boldsymbol{AB} = \boldsymbol{BA}$ for all $\boldsymbol{A}$ and $\boldsymbol{B}$—**FALSE!**
  - $\boldsymbol{A}$ and $\boldsymbol{B}$ are invertible $\rightarrow (\boldsymbol{A} + \boldsymbol{B})$ is invertible—**FALSE!**

Matrix Exponential — 1

- Exponential of scalar variable:

$$e^a = \sum_{i=0}^{\infty} \frac{a^i}{i!} = 1 + a + \frac{a^2}{2!} + \frac{a^3}{3!} + \frac{a^4}{4!} + \cdots$$

- Power series converges $\forall\ a \in \mathbb{R}$
- How about matrices? For $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, matrix exponential:

$$e^{\boldsymbol{A}} = \sum_{i=0}^{\infty} \frac{\boldsymbol{A}^i}{i!} = \boldsymbol{I}_n + \boldsymbol{A} + \frac{\boldsymbol{A}^2}{2!} + \frac{\boldsymbol{A}^3}{3!} + \frac{\boldsymbol{A}^4}{4!} + \cdots$$

- What if we have a time-variable?

$$e^{t\boldsymbol{A}} = \sum_{i=0}^{\infty} \frac{(t\boldsymbol{A})^i}{i!} = \boldsymbol{I}_n + t\boldsymbol{A} + \frac{(t\boldsymbol{A})^2}{2!} + \frac{(t\boldsymbol{A})^3}{3!} + \frac{(t\boldsymbol{A})^4}{4!} + \cdots$$

Matrix Exponential — 2

- **Objective**: computing matrix exponential $e^{\boldsymbol{A}}$ for square matrices $\boldsymbol{A} \in \mathbb{R}^{n \times n}$

- **Assumption**: matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ is diagonalizable, hence:

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix} \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix}^{-1}$$

- We can then show that

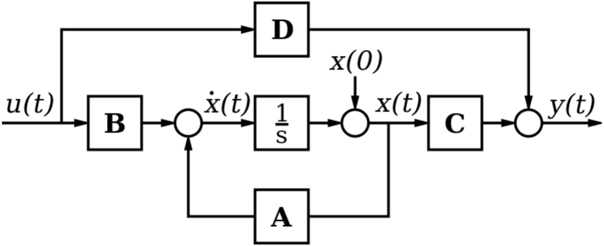
$$e^{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix} \begin{bmatrix} e^{\lambda_1} & & & \\ & e^{\lambda_2} & & \\ & & \ddots & \\ & & & e^{\lambda_n} \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix}^{-1}$$

Solution to the State-Space Equation

- In the next few slides, we'll answer this question: what is a solution to this vector-matrix first order ODE:

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)\\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t)\end{aligned}$$

- By solution, we mean a closed-form solution for $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ given:
  - An initial condition for the system, i.e., $\boldsymbol{x}(t_{initial}) = \boldsymbol{x}(0)$
  - A given control input signal, $\boldsymbol{u}(t)$, such as a step-input ($u(t) = 1$), ramp ($u(t) = t$), or anything else



The Curious Case of Autonomous Systems—Case 1

- Let's assume that we seek solution to this system first:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t), \boldsymbol{x}(0) = \boldsymbol{x}_0 = \text{given} \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) \end{aligned}$$

- This means that the system operates without any control input—**autonomous system** (e.g., autonomous vehicles)
- First, let's look at $\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t)$—what's the solution to this first order ODE?
  - First case: $\boldsymbol{A} = a$ is a scalar $\Rightarrow x(t) = e^{at}x_0$
  - Second case: $\boldsymbol{A}$ is a matrix

$$\boxed{\Rightarrow \boldsymbol{x}(t) = e^{\boldsymbol{A}t}\boldsymbol{x}_0 \Rightarrow \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) = \boldsymbol{C}e^{\boldsymbol{A}t}\boldsymbol{x}_0}$$

- Exponential of scalars is very easy, but exponentials of matrices can be very challenging
- Hence, for an nth order system, where $n \geq 2$, we need to compute the matrix exponential in order to get a solution for the above system—we learned that in the linear algebra revision section

Example (Case 1)

$$\boxed{\boldsymbol{x}(t) = e^{\boldsymbol{A}t}\boldsymbol{x}_0, \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) = \boldsymbol{C}e^{\boldsymbol{A}t}\boldsymbol{x}_0}$$

- Find the solution for these two autonomous systems separately:

$$\boldsymbol{A}_1 = \begin{bmatrix} 1 & 0 \\ 0 & -2 \end{bmatrix}, \boldsymbol{C}_1 = \begin{bmatrix} 1 & 2 \end{bmatrix}, \boldsymbol{x}_0^{(1)} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$\boldsymbol{A}_2 = \begin{bmatrix} 0 & 1 \\ 0 & -2 \end{bmatrix}, \boldsymbol{C}_2 = \begin{bmatrix} 2 & 0 \end{bmatrix}, \boldsymbol{x}_0^{(2)} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

- **Solution:**

Case 2—Systems with Inputs

- MIMO (or SISO) LTI dynamical system:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \boldsymbol{x}(t_0) = \boldsymbol{x}_{t_0} = \text{given} \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{aligned}$$

- The to the above ODE is given by:

$$\boxed{\boldsymbol{x}(t) = e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}_{t_0} + \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)\, d\tau}$$

- Clearly the output solution is:

$$\boxed{\boldsymbol{y}(t) = \underbrace{\boldsymbol{C}\left(e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}_{t_0}\right)}_{\text{zero input response}} + \underbrace{\boldsymbol{C}\left(\int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)\, d\tau\right) + \boldsymbol{D}\boldsymbol{u}(t)}_{\text{zero state response}}}$$

- **Question:** how do I analytically compute $\boldsymbol{y}(t)$ and $\boldsymbol{x}(t)$?

- **Answer:** you need to (a) **integrate** and (b) **compute matrix exponentials** (given $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}, \boldsymbol{x}_{t_0}, \boldsymbol{u}(t)$)

Example (Case 2)

$$\boldsymbol{x}(t) = e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}_{t_0} + \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)\, d\tau$$

$$\boldsymbol{y}(t) = \underbrace{\boldsymbol{C}\left(e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}_{t_0}\right)}_{\text{zero input response}} + \underbrace{\boldsymbol{C}\left(\int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)\, d\tau\right) + \boldsymbol{D}\boldsymbol{u}(t)}_{\text{zero state response}}$$

- Find the solution for these two LTI systems with inputs:

$$\boldsymbol{A}_1 = \begin{bmatrix} 1 & 0 \\ 0 & -2 \end{bmatrix}, \boldsymbol{B}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \boldsymbol{C}_1 = \begin{bmatrix} 1 & 2 \end{bmatrix}, \boldsymbol{x}_0^{(1)} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, D_1 = 0, u_1(t) = 1$$

$$\boldsymbol{A}_2 = \begin{bmatrix} 0 & 1 \\ 0 & -2 \end{bmatrix}, \boldsymbol{B}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \boldsymbol{C}_2 = \begin{bmatrix} 2 & 0 \end{bmatrix}, \boldsymbol{x}_0^{(2)} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, D_2 = 1, u_2(t) = 2e^{-2t}$$

- **Solution:**

Stability of LTI Systems

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \boldsymbol{x}(t_0) = \boldsymbol{x}_{t_0} = \text{given} \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t)\end{aligned}$$

- The above system is stable if and only if all the eigenvalues of matrix $\boldsymbol{A}$ are **strictly negative**
- REALLY? No need for other matrices? No!
- How does that relate to the stability from CLTF?
- **Answer:** the poles of CLTF are equal to the eigenvalues of $\boldsymbol{A}$
- Example:

Controllability — 1

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \boldsymbol{x}(0) = \boldsymbol{x}_0$$

- **Controllability**: the ability to move a system (i.e., its states $\boldsymbol{x}(t)$) from one point in space to another via certain control signals $\boldsymbol{u}(t)$
- **Rigorous definition**: Over the time interval $[0, t_f]$, control input $\boldsymbol{u}(t)\ \forall\ t \in [0, t_f]$ steers the state from $\boldsymbol{x}_0$ to $\boldsymbol{x}_{t_f}$:

$$\boldsymbol{x}(t_f) = \boldsymbol{e}^{\boldsymbol{A}t_f}\boldsymbol{x}_0 + \int_0^{t_f} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)\, d\tau$$

Controllability Definition

LTI system is controllable at time $t_f > 0$ if for any initial state and for any target state (x_{t_f}), a control input $\boldsymbol{u}(t)$ exists that can steer the system states from $x(0)$ to $x(t_f)$ over the defined interval.

- Physical examples

Controllability — 2

Controllability Test

For a system with n states and m control inputs, the test for controllability is that matrix

$$\mathcal{C} = \begin{bmatrix} \boldsymbol{B} & \boldsymbol{AB} & \boldsymbol{A}^2\boldsymbol{B} & \cdots & \boldsymbol{A}^{n-1}\boldsymbol{B} \end{bmatrix} \in \mathbb{R}^{n \times nm}$$

has full row rank (i.e., $\text{rank}(\mathcal{C}) = n$).

Theorem

The following statements are equivalent:

1. $\mathcal{C}$ is full rank
2. PBH Test: for all $\lambda_i \in eig(\boldsymbol{A})$, $\text{rank}\,[\lambda_i \boldsymbol{I} - \boldsymbol{A} \;\; \boldsymbol{B}] = n$
3. Eigenvector Test: for any evector $\boldsymbol{v}_i$ of $\boldsymbol{A}$, $\boldsymbol{v}^T\boldsymbol{B} \neq 0$

Observability — 1

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \boldsymbol{x}(t_0) = \boldsymbol{x}_{t_0} = \text{given} \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{aligned}$$

- **Observability:** can we figure out internal states $\boldsymbol{x}(t)$ from only measuring outputs $\boldsymbol{y}(t)$?
- Above system is **observable** $\Rightarrow$ current state can be determined using only the outputs ($\boldsymbol{y}(t)$)
- Hence, from $\boldsymbol{y}(t)$, we can obtain $\boldsymbol{x}(t)$ $\Rightarrow$ determine the behavior of the entire system (wow)
- A system is **not observable** $\Rightarrow$ $\boldsymbol{x}(t)$ cannot be determined uniquely from $\boldsymbol{y}(t)$
- What does that mean in reality?
- Examples....

Observability — 2

Observability Test

For a system with n states and p outputs, the test for observability is that matrix $\mathcal{O} = \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{CA} \\ \vdots \\ \boldsymbol{CA}^{n-1} \end{bmatrix} \in \mathbb{R}^{np \times n}$ has full column rank (i.e., $\text{rank}(\mathcal{C}) = n$).

Theorem

The following statements are equivalent:

1. $\mathcal{O}$ is full rank, system is observable
2. PBH Test: for all $\lambda_i \in eig(\boldsymbol{A})$, rank $\begin{bmatrix} \lambda_i \boldsymbol{I} - \boldsymbol{A} \\ \boldsymbol{C} \end{bmatrix} = n$
3. Eigenvector Test: for any evector $\boldsymbol{v}_i$ of $\boldsymbol{A}$, $\boldsymbol{Cv} \neq 0$

Examples

- Consider two dynamical systems defined by:

– First system:

$$\boldsymbol{A} = \begin{bmatrix} 1 & -1 \\ 1 & -2 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

- Second system:

$$\boldsymbol{A} = \begin{bmatrix} 1 & -1 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Are these two systems controllable?
- Are these two system observable?
- **Solution:** you'll have to compute matrices $\mathcal{O}, \mathcal{C}$ for these systems and then find the rank of these two matrices
- **Answers:**
- MATLAB commands: `ctrb, obsv`

Questions And Suggestions?

Thank You!

Please visit

engineering.utsa.edu/ataha

IFF you want to know more ☺