Lecture 13: Discrete-time State Space Equations

ELEN 472: Introduction to Digital Control

Lingxiao Wang, Ph.D.
Assistant Professor of Electrical Engineering
Louisiana Tech University

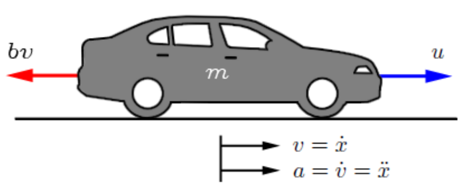
System States

- In this lecture, we discuss an alternative system representation in terms of the system state variables, known as the **state-space representation.**
- Definition of System States:

> The state of a system is the **minimal** set of numbers $\{x_i(t_0), i = 1, 2, \ldots, n\}$ needed to **uniquely determine the behavior of the system** in the interval t_0 to t_f. The number of n is known as the **order of the system**.

- **Example**:
  - Consider a cruise control problem with the dynamic equation:

bv: ***Resistive force due to wind drag, rolling resistance, etc.***



u: ***Force generated by the engine***

v: ***vehicle speed***
m: ***vehicle mass***
b: ***damping coefficient***

$$\sum F_x = u - bv = m\dot{v}$$

We can define the system state as the vehicle velocity $x = v$

Example 2

- Consider the equation of motion of a point mass m driven by a force f

$$m\ddot{y} = f$$

where y is the displacement of the point mass. Define state variables of this system.

- **Solution:**
  - We can define state variables as $\mathbf{x}(\mathrm{t}) = [y(t), \dot{y}(t)]^T$ given the fact that the system is a **second-order system**.
  - These state variables are governed by two first-order differential equations, i.e., $\mathbf{x}(t) = [x_1(t), x_2(t)]$:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = u/m$$

    - where $u = f$
  - Notice that the choice of state variables is not unique.
    - We can also define $\mathbf{x}(t) = [y, y + \dot{y}]$ as the state variables.

State-Space Representation

- In the previous example, two first-order equations governing the state variables were obtained from the second-order input-output differential equation and the definitions of the state variables.

$$
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= u/m
\end{aligned}
$$

  - These equations are known as **state equations**.

- In general, there are ***<u>n state equations for an nth-order system</u>***.
- The state and output equations together provide a complete representation for the system described by the differential equation, which is known as the **state-space representation**.
- The state-space representation of the previous example question is

$$
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= u/m
\end{aligned}
$$

$$
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{m} \end{bmatrix} u
$$

$$
y = [1 \quad 0] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}
$$

State-Space Representation

- The general form of the state space equations for linear systems is

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t)$$
$$\mathbf{y}(t) = C\mathbf{x}(t) + D\mathbf{u}(t)$$

where $\mathbf{x}(t)$ is an $n \times 1$ real vector, $u(t)$ is an $m \times 1$ real vector, and $y(t)$ is an $l \times 1$ real vector.

- The matrices in the equations are

$A = n \times n$ **state matrix**

$B = n \times m$ **input** or **control matrix**

$C = l \times n$ **output matrix**

$D = l \times m$ **direct transmission matrix**

- The orders of the matrices are dictated by the dimensions of the vectors and the rules of vector-matrix multiplication.
  - For example, in the single-input (SI) case, B is a column matrix, and in the single-output (SO) case, both C and D are row matrices.
  - For the SISO case, D is a scalar.

Example

- The following are examples of state-space equations for linear systems:
  - A third-order 2-input 2-output (MIMO) linear **time-invariant** system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 1.1 & 0.3 & -1.5 \\ 0.1 & 3.5 & 2.2 \\ 0.4 & 2.4 & -1.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0.1 & 0 \\ 0 & 1.1 \\ 1.0 & 1.0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

  - A second-order 2-output single-input (SIMO) linear **time-varying** system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \sin(t) & \cos(t) \\ 1 & e^{-2t} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Linear versus nonlinear state space equations

- It is important to remember that the following state-space representation is only valid for **linear** state equations.

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t)$$
$$\mathbf{y}(t) = C\mathbf{x}(t) + D\mathbf{u}(t)$$

- Nonlinear state equations involve nonlinear functions and cannot be written in terms of the matrix (A, B, C, D).

- **Example**
    - Obtain a state space representation for the s-degree-of-freedom (s-D.O.F.) robotic manipulator from the equation of motion

$$M(\mathbf{q})\ddot{\mathbf{q}} + V(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \tau$$

where

$\mathbf{q}$ = vector of generalized coordinates

$M(\mathrm{q}) = s \times s$ positive definite inertia matrix

$V(\mathbf{q}, \dot{\mathbf{q}}) = s \times s$ matrix of velocity-related terms

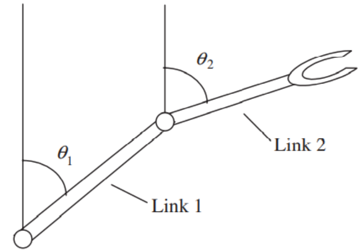
$\mathbf{g}(\mathbf{q}) = s \times 1$ vector of gravitational terms

τ = vector of generalized forces

The output of the manipulator is the position vector $\mathbf{q}$.

- The system is a second-order system. Thus, we need to define two states.
- The most natural choice of state variables is the vector: $\mathbf{x} = \mathrm{col}\{\mathbf{x}_1, \mathbf{x}_2\} = \mathrm{col}\{\mathbf{q}, \dot{\mathbf{q}}\}$
- The associated state equations are

$$\begin{bmatrix} \dot{\mathbf{x}}_1 \\ \hline \dot{\mathbf{x}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_2 \\ \hline -M^{-1}(\mathbf{x}_1)\{V(\mathbf{x}_1, \mathbf{x}_2)\mathbf{x}_2 + \mathbf{g}(\mathbf{x}_1)\} \end{bmatrix} + \begin{bmatrix} 0 \\ \hline M^{-1}(\mathbf{x}_1) \end{bmatrix}\mathbf{u}$$



Nonlinear State-Space Equations

- The general form of nonlinear state space equations is

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\ \mathbf{u})$$
$$\mathbf{y} = \mathbf{g}(\mathbf{x},\ \mathbf{u})$$

where $\mathbf{f}(.)$ $(n \times 1)$ and $\mathbf{g}(.)$ $(l \times 1)$ are vectors of functions satisfying mathematical conditions that guarantee the existence and uniqueness of solution.

- A form that is often encountered in practice and includes the equations of robotic manipulators is

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + B(\mathbf{x})\mathbf{u}$$
$$\mathbf{y} = \mathbf{g}(\mathbf{x}) + D(\mathbf{x})\mathbf{u}$$

Linearization of Nonlinear State Equations

- Nonlinear state equations can be approximated by linear state equations for small ranges of the control and state variables.

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x},\ \mathbf{u})\\ \mathbf{y} &= \mathbf{g}(\mathbf{x},\ \mathbf{u})\end{aligned} \quad \Longrightarrow \quad \begin{aligned}\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t)\\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t)\end{aligned}$$

- The linear equations are based on the first-order approximation

$$f(x) = f(x_0) + \left.\frac{df}{dx}\right|_{x_0} \Delta x + O(\Delta^2 x)$$

  - where x_0 is a constant and $\Delta x = x - x_0$ is a perturbation from the constant.
  - The error associated with the approximation is of order $\Delta^2 x$ and is therefore acceptable for small perturbations.

- Apply the above equation to the nonlinear state equations, i.e.,

$$f(x) = f(x_0) + \left.\frac{df}{dx}\right|_{x_0} \Delta x + O(\Delta^2 x) \quad + \quad \begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x},\ \mathbf{u})\\ \mathbf{y} &= \mathbf{g}(\mathbf{x},\ \mathbf{u})\end{aligned}$$

  - We have the following results:

Linearization of Nonlinear State Equations (Continued)

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x},\ \mathbf{u}) \\ \mathbf{y} &= \mathbf{g}(\mathbf{x},\ \mathbf{u}) \end{aligned} \quad \Longrightarrow \quad \begin{aligned} \dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t) \end{aligned}$$

$$A = \begin{bmatrix} \left.\frac{\partial f_1}{\partial x_1}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} & \cdots & \left.\frac{\partial f_1}{\partial x_n}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} \\ \cdots & \cdots & \cdots \\ \left.\frac{\partial f_n}{\partial x_1}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} & \cdots & \left.\frac{\partial f_n}{\partial x_n}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} \end{bmatrix} \quad B = \begin{bmatrix} \left.\frac{\partial f_1}{\partial u_1}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} & \cdots & \left.\frac{\partial f_1}{\partial u_m}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} \\ \cdots & \cdots & \cdots \\ \left.\frac{\partial f_n}{\partial u_1}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} & \cdots & \left.\frac{\partial f_n}{\partial u_m}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} \end{bmatrix}$$

$$C = \begin{bmatrix} \left.\frac{\partial g_1}{\partial x_1}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} & \cdots & \left.\frac{\partial g_1}{\partial x_n}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} \\ \cdots & \cdots & \cdots \\ \left.\frac{\partial g_n}{\partial x_1}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} & \cdots & \left.\frac{\partial g_n}{\partial x_n}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} \end{bmatrix} \quad D = \begin{bmatrix} \left.\frac{\partial g_1}{\partial u_1}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} & \cdots & \left.\frac{\partial g_1}{\partial u_m}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} \\ \cdots & \cdots & \cdots \\ \left.\frac{\partial g_n}{\partial u_1}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} & \cdots & \left.\frac{\partial g_n}{\partial u_m}\right|_{(\mathbf{x}_0,\ \mathbf{u}_0)} \end{bmatrix}$$

- Th above A, B, C, D matrices are called **Jacobian** matrices, which is the result after first-order linearization.
- $(\mathbf{x}_0, \mathbf{u}_0)$ is called **equilibrium point**, which refers to an initial state where the system remains linear.
  - Equilibrium point can be calculated by $\mathbf{f}(\mathbf{x}, \mathbf{u}) = 0, \mathbf{g}(\mathbf{x}, \mathbf{u}) = 0$

Example Question

- The state equation of a nonlinear system is presented as

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -\frac{x_2^4}{x_1^2} + x_1 + \sqrt{u+1}$$

  - Find the equilibrium point when $u_0 = 3$.
  - Linearize the system at the equilibrium point.
- Solution:
  - The dynamic equations can be described as:

$$\dot{x}_1 = x_2 \qquad = f_1(x_1, x_2, u)$$
$$\dot{x}_2 = -\frac{x_2^4}{x_1^2} + x_1 + \sqrt{u+1} \qquad = f_2(x_1, x_2, u)$$

  - Find the equilibrium point for $u_0 = 3$, i.e., $\dot{x}_1 = 0, \dot{x}_2 = 0$

$$0 = x_2$$
$$0 = -\frac{x_2^4}{x_1^2} + x_1 + \sqrt{3+1}$$

$$\Longrightarrow (x_{10}, x_{20}, u_0) = (-2, 0, 3)$$

Practice Question Solution

- Linearize at the equilibrium point (-2, 0, 3):

$$\frac{\partial f_1}{\partial x_1} = 0, \qquad \frac{\partial f_1}{\partial x_2} = 1, \qquad \frac{\partial f_1}{\partial u} = 0,$$

$$\frac{\partial f_2}{\partial x_1} = +2\frac{x_2^4}{x_1^3} + 1, \qquad \frac{\partial f_2}{\partial x_2} = -4\frac{x_2^3}{x_1^2}, \qquad \frac{\partial f_2}{\partial u} = \frac{1}{2\sqrt{u+1}},$$

$$\frac{\partial f_1}{\partial x_1}_{|\{x_0,\, u_0\}} = 0, \qquad \frac{\partial f_1}{\partial x_2}_{|\{x_0,\, u_0\}} = 1, \qquad \frac{\partial f_1}{\partial u}_{|\{x_0,\, u_0\}} = 0,$$

$$\frac{\partial f_2}{\partial x_1}_{|\{x_0,\, u_0\}} = 1, \qquad \frac{\partial f_2}{\partial x_2}_{|\{x_0,\, u_0\}} = 0, \qquad \frac{\partial f_2}{\partial u}_{|\{x_0,\, u_0\}} = \frac{1}{4},$$

$$\frac{f(x,u)}{\partial x}_{|\{x_0,\, u_0\}} = A \quad = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad \frac{f(x,u)}{\partial u}_{|\{x_0,\, u_0\}} = B \quad = \begin{bmatrix} 0 \\ \frac{1}{4} \end{bmatrix}$$

The solution of linear state space equations

- The state space equations are linear and can therefore be Laplace transformed to obtain their solution.
- Hence, the state equation:

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

- Taking the Laplace transform on both sides, we have:

$$s\mathbf{X}(s) - \mathbf{x}(0) = A\mathbf{x}(s) + B\mathbf{U}(s)$$

$$[sI_n - A]\mathbf{X}(s) = \mathbf{x}(0) + B\mathbf{U}(s)$$

$$\mathbf{X}(s) = [sI_n - A]^{-1}[\mathbf{x}(0) + B\mathbf{U}(s)]$$

- Apply Laplace Transform on both sides, we have

$$\mathbf{x}(t) = e^{At}\mathbf{x}(0) + \int_0^t e^{A(t-\tau)}B\mathbf{u}(\tau)d\tau$$

  - Where $e^{At} \rightarrow \mathscr{L}^{-1}\{[sI_n - A]^{-1}\}$

State Transition Matrix

- e^{At} is called the **state-transition matrix**, which can be found via the following two methods:
  - $e^{At} = \mathcal{L}^{-1}\{[sI_n - A]^{-1}\}$
  - $e^{At} = Me^{(\Lambda t)}M^{-1}$,
    - where $M = [\mathbf{v_1}, \mathbf{v_2}, \dots \mathbf{v_n}]$ denote the matrix containing **eigenvectors** $\boldsymbol{v_i}$ of A;
    - $e^{\Lambda t} = \mathrm{diag}([\mathrm{e}^{\lambda_1 t}, \mathrm{e}^{\lambda_2 t}, \dots \mathrm{e}^{\lambda_n t}])$ is the diagonal matrix and λ is the **eigenvalue** of A.
- How to find **eigenvectors** and **eigenvalues**?
  - Manual Calculation:
    - $\det(A - \lambda I) = 0 \rightarrow$ Eigenvalues
    - $(A - \lambda I)\mathbf{v} = 0 \rightarrow$ Eigenvectors
  - MATLAB Command:

```
% Define the A matrix
A = [1, 2; 3, 4]
% Find the eigenvalues and eigenvectors of A
% Every column in V is a eigenvector
% Every diagonal element in D is a eigenvalue
[V, D] = eig(A)
```

```
V =
  -0.8246   -0.4160
   0.5658   -0.9094

D =
  -0.3723        0
        0   5.3723
```

Example

- The state equations of an armature-controlled DC motor are given by

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}0&1&0\\0&0&1\\0&-10&-11\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}+\begin{bmatrix}0\\0\\10\end{bmatrix}u$$

  - Find the state-transition matrix e^{At}
- Solution:
  - The state-transition matrix is the matrix exponential given by the inverse Laplace transform of the matrix, $\mathscr{L}^{-1}\{[sI_n-A]^{-1}\} \longrightarrow e^{At}$

$$[sI_3-A]^{-1}=\begin{bmatrix}s&-1&0\\0&s&-1\\0&10&s+11\end{bmatrix}^{-1}$$

$$=\frac{\begin{bmatrix}(s+1)(s+10)&s+11&1\\0&s(s+11)&s\\0&-10s&s^2\end{bmatrix}}{s(s+1)(s+10)}$$

$$=\begin{bmatrix}\frac{1}{s}&\frac{s+11}{s(s+1)(s+10)}&\frac{1}{s(s+1)(s+10)}\\0&\frac{s+11}{(s+1)(s+10)}&\frac{1}{(s+1)(s+10)}\\0&\frac{-10}{(s+1)(s+10)}&\frac{s}{(s+1)(s+10)}\end{bmatrix}$$

$$=\begin{bmatrix}\frac{1}{s}&\frac{1}{90}\left(\frac{99}{s}-\frac{100}{s+1}+\frac{1}{s+10}\right)&\frac{1}{90}\left(\frac{9}{s}-\frac{10}{s+1}+\frac{1}{s+10}\right)\\0&\frac{1}{9}\left(\frac{10}{s+1}-\frac{1}{s+10}\right)&\frac{1}{9}\left(\frac{1}{s+1}-\frac{1}{s+10}\right)\\0&-\frac{10}{9}\left(\frac{1}{s+1}-\frac{1}{s+10}\right)&\frac{1}{9}\left(\frac{10}{s+10}-\frac{1}{s+1}\right)\end{bmatrix}$$

Finally, by taking the inverse Laplace transform, we can get e^{At}

$$e^{At}=\begin{bmatrix}1&\frac{1}{90}(99-100e^{-t}+e^{-10t})&\frac{1}{90}(9-10e^{-t}+e^{-10t})\\0&\frac{1}{9}(10e^{-t}-e^{-10t})&\frac{1}{9}(e^{-t}-e^{-10t})\\0&-\frac{10}{9}(e^{-t}-e^{-10t})&\frac{1}{9}(10e^{-10t}-e^{-t})\end{bmatrix}$$

Example Revise

- The state equations of an armature-controlled DC motor are given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -10 & -11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 10 \end{bmatrix} u$$

  - Find the state-transition matrix e^{At}

- **MATLAB Solution**:

```
clear all
% Define the time variable t
syms t

% Define the A matrix
A = [0 1 0;
0 0 1;
0 -10 -11];

% Find the eigenvalues and eigenvectors
of A
[V, D] = eig(A);

% Define the diagonal matrix
e_diagonal = diag(exp(diag(D*t)));

% Define State Transition Matrix
STM = V * e_diagonal * inv(V);

% Simplify the expression
simplify(STM)
```

$e^{\Lambda t} = \mathrm{diag}([\mathrm{e}^{\lambda_1 t}, \mathrm{e}^{\lambda_2 t}, \ldots \mathrm{e}^{\lambda_n t}])$ is the diagonal matrix and λ is the **eigenvalue** of A.

$$e^{At} = Me^{(\Lambda t)}M^{-1}$$

Discrete-time state space equations

- Similar to continuous-time state space equations, we can also get state space equations for **discrete-time** systems.

$$\mathbf{x}[k+1] = \mathbf{A}_d\mathbf{x}[k] + \mathbf{B}_d\mathbf{f}[k]$$

$$\mathbf{y}[k] = \mathbf{C}_d\mathbf{x}[k] + \mathbf{D}_d\mathbf{f}[k]$$

  - x is the state vector
  - $\mathrm{A_d} = \mathrm{e}^{\mathrm{A}T}$
  - $\mathrm{B_d} = (\mathrm{A_d} - \mathrm{I})\mathrm{A}^{-1}\mathrm{B}$
  - $\mathrm{C_d} = \mathrm{C}$
  - $\mathrm{D_d} = \mathrm{D}$

Example

- Find the discrete-time state-space model from the continuous-time system: (sampling period $T = 0.1$ s)

$$\dot{\mathrm{x}}(t) = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}\mathrm{x}(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} f(t)$$

$$y(t) = [1 \quad 0]\,\mathrm{x}(t)$$

- **Solution:**

$$\mathrm{A}_d = \Phi(T) = \begin{bmatrix} 2e^{-T} - e^{-2T} & e^{-T} - e^{-2T} \\ 2e^{-2T} - 2e^{-T} & 2e^{-2T} - e^{-T} \end{bmatrix} = \begin{bmatrix} 0.9909 & 0.0861 \\ -0.1722 & 0.7326 \end{bmatrix}$$

e^{AT}

$$\mathrm{B}_d = (\mathrm{A}_d - \mathrm{I})\mathrm{A}^{-1}\mathrm{B} = \begin{bmatrix} \frac{1}{2}(1 + e^{-2T}) - e^{-T} \\ e^{-T} - e^{-2T} \end{bmatrix} = \begin{bmatrix} 0.0045 \\ 0.0861 \end{bmatrix}$$

$$\mathrm{C}_d = \mathrm{C} = [1 \quad 0], \qquad \mathrm{D}_d = \mathrm{D} = 0$$

Example Continued

- **MATLAB Solution**

e^{AT}

```
clear all

% Define A, B, C, D matrix
A = [0 1; -2 -3];
B = [0; 1];
C = [1 0];
D = 0

% Define the sampling time
T = 0.1;

% Find Ad
[V, D] = eig(A * T);
e_diag = diag(exp(diag(D)));
Ad = V * e_diag * inv(V)

% Find Bd
I = eye(2);
Bd = (Ad-I) * inv(A) * B

% Find Cd
Cd = C

% Find Dd
Dd = D
```