Linear Control Systems
25411

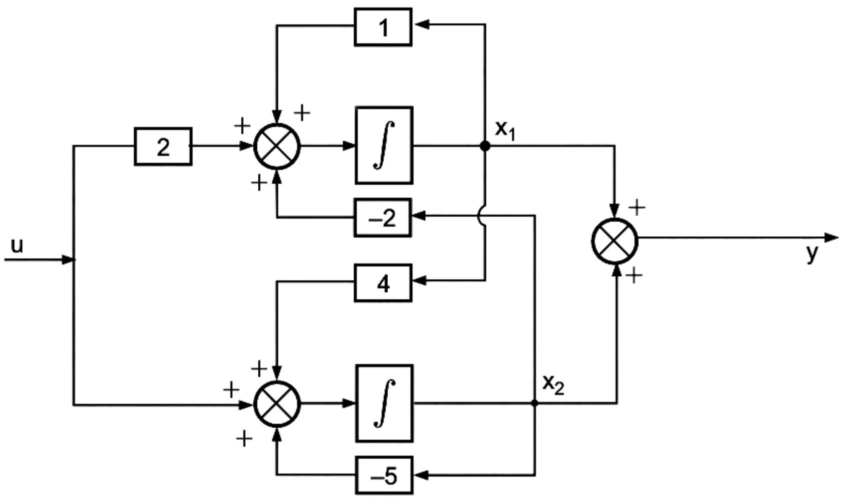
Instructor. Behzad Ahi **Assignment 2**

Fall 1402 **Due Date: 1402/8/12**

---

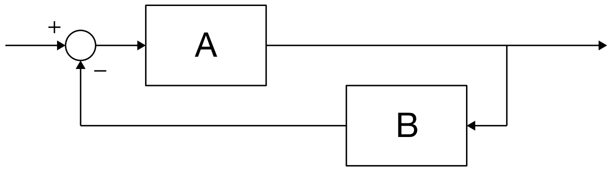
1 Block Diagram to State Transition Matrix

The state block diagram of a linear system is shown below. Write state equations and determine state transition matrix e^{At}.



2 Feedback Combination

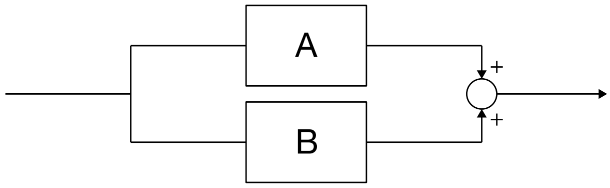
Consider two linear systems with state space models A and B. Derive the state space model of the feedback combination of these systems.



$$A : \begin{cases} \dot{x}(t) = \begin{bmatrix} -3 & 7 \\ -6 & 4 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ -1 \end{bmatrix} u(t) \\ \\ y(t) = \begin{bmatrix} 8 & 5 \end{bmatrix} x(t) \end{cases} \qquad B : \begin{cases} \dot{x}(t) = \begin{bmatrix} 2 & 1 \\ -1 & 5 \end{bmatrix} x(t) + \begin{bmatrix} 3 \\ 1 \end{bmatrix} u(t) \\ \\ y(t) = \begin{bmatrix} 2 & -7 \end{bmatrix} x(t) \end{cases}$$

3 Parallel Combination

Consider two linear systems with state space models A and B. Derive the state space model of the parallel combination of these systems. Then, obtain the transfer function of the combined system and simplify it. Finally, convert the simplified transfer function back to a state space model and compare it with the derived one.
Hint: you can use MATLAB to find the transfer function of the combined system.



$$A: \begin{cases} \dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -2 & -3 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t) \\ \\ y(t) = \begin{bmatrix} 1 & -5 & 3 \end{bmatrix} x(t) \end{cases} \qquad B: \begin{cases} \dot{x}(t) = \begin{bmatrix} -3 & 1 & 0 \\ -2 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} -2 \\ 8 \\ 1 \end{bmatrix} u(t) \\ \\ y(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x(t) \end{cases}$$

4 Response of LTI State-Space Models

Given the state space representation of following linear system, find the output of the system when it is subjected to a unit step input regarding $x(0) = [1 \;\; 0]^T$ as the initial condition.

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} -1 & -2 \\ 1 & -4 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \\ \\ y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} x(t) \end{cases}$$

5 State-Space Representation of Differential Equations

Obtain the standard linear state-space representation for the following system.

$$\begin{cases} \ddot{s}_1(t) + 3\dot{s}_2(t) + 2s_1(t) = 3\dot{u}(t) - u(t) \\ \\ \dot{s}_2(t) - 2\dot{s}_1(t) + s_2(t) = 0 \\ \\ y(t) = \dot{s}_1(t) + \dot{s}_2(t) \end{cases}$$

MATLAB Assignments

6 Canonical Form Representation of Transfer Functions

Controllable and Observable canonical forms are defined for the given strictly proper transfer function $T(s)$ as follows (Don't use builtin MATLAB functions for canonical forms)

$$T(s) = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \cdots + a_1 s + a_0} \rightarrow \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ \\ y(t) = Cx(t) \end{cases}$$

1. Controllable Canonical Form:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 & -a_3 & \cdots & -a_{n-3} & -a_{n-2} & -a_{n-1} \end{bmatrix}_{n\times n}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}_{n\times 1}$$

$$C = \begin{bmatrix} b_0 & b_1 & \cdots & b_{n-2} & b_{n-1} \end{bmatrix}_{1\times n}$$

2. Observable Canonical Form:

$$A = \begin{bmatrix} -a_{n-1} & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -a_{n-2} & 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ -a_{n-3} & 0 & 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ -a_3 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ -a_2 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 \\ -a_1 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 \\ -a_0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix}_{n\times n}$$

$$B = \begin{bmatrix} b_{n-1} \\ b_{n-2} \\ \vdots \\ b_1 \\ b_0 \end{bmatrix}_{n\times 1}$$

$$C = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \end{bmatrix}_{1\times n}$$

Assignment: Define a MATLAB function that takes a strictly proper transfer function and first letter of the desired canonical form as the inputs and returns `[A, B, C]` matrices of the given canonical form. For example:

```
[A, B, C] = canonicalForms(sys, "C");     or     [A, B, C] = canonicalForms(sys, "O");
```

7 Response of LTI State-Space Models using `lsim`

The `lsim` function can help us analyze the behavior and performance of a system under different scenarios and conditions. The following steps describe how to use the `lsim` function in MATLAB to find the response of a linear state-space model.

I) Define the state-space system using the `ss` command. For example, if you have a system with matrices A, B, C, and D, you can create a state-space object `sys` using

```
sys = ss(A, B, C, D);
```

II) Define the input signal `u` and the time vector `t`. The input signal `u` should be a matrix with as many rows as time samples and as many columns as system inputs. The time vector `t` should be a vector of regularly spaced time samples. For example, if you want to simulate the system for 10 seconds with a sampling time of $T_s = 0.01s$ and a sinusoidal input of frequency 1 rad/s, you can use

```
t = 0:0.01:10;
u = sin(t);
```

III) Specify the initial conditions of your system, the initial conditions should be a vector that contains the values of the state variables at the initial time. The length of the initial conditions vector should match the number of states in your system. For example, if your system has two states and you want to use zero initial conditions, you can use

```
x0 = [0, 0];
```

IV) Use the `lsim` command to simulate the system response to the input signal. The `lsim` command returns the output response `y`, the time vector `tOut` used for simulation, and the state trajectories `x`. For example

```
[y, tOut, x] = lsim(sys, u, t, x0);
```

Assignment: Given a linear time-invariant (LTI) state-space model and an input signal $u(t)$, use the MATLAB function `lsim` to compute the state trajectories $x_1(t), x_2(t), \cdots$ and the output response $y(t)$ for the time interval $0 \leq t \leq 10$. Show the plots of the state and output variables on the same figure to illustrate how each state affects the system output.

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 & -1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} u(t) \\ \\ y(t) = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \end{bmatrix} x(t) \end{cases}$$

$$u(t) = \begin{cases} 1 & 2 \leq t \leq 4 \\ -1 & 6 \leq t \leq 8 \\ 0 & O.W. \end{cases}$$

8 Response of LTI State-Space Models using Euler Method

To find a numerical solution of an LTI state-space model using the Euler method iteratively in MATLAB, you need to follow these steps

I) Define the state-space model using the A, B, C, and D matrices

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}$$

For example

```
A = [-1, 1; 0, -2];
B = [0; 1];
C = [1, 0];
D = 0;
```

II) Define the input signal `u` and the time vector `t`. The input signal `u` should be a matrix with as many rows as time samples and as many columns as system inputs. The time vector `t` should be a vector of regularly spaced time samples. For example, if you want to simulate the system for 10 seconds with a sampling time of 0.01 seconds and a step input, you can use

```
Ts = 0.01;
t = 0:0.01:10;
u = ones(size(t));
u(1:100) = 0;
```

III) Define a matrix to store the states samples `X` and specify the initial conditions

```
X = zeros([length(A), length(t)]);
x0 = [0; 0];
X(:, 1) = x0;
```

IV) Define a matrix to store the output samples `Y`

```
Y = zeros(size(t));
Y(1) = C * x0;
```

V) Use the Euler's formula to approximate the system response to the input signal. The Euler's formula is

$$x[i+1] = x[i] + T_s \dot{x}[i]$$

where

$$\dot{x}[i] = Ax[i] + Bu[i]$$

So

$$x[i+1] = x[i] + T_s \left(Ax[i] + Bu[i] \right)$$

VI) Implement the Euler's formula using a `for` loop

```
for i=2:length(t)
     X(:, i) = X(:, i-1) + Ts * (A * X(:, i-1) + B * u(i-1));
     Y(i) = C * X(:, i) + D * u(i);
end
```

Assignment: Given a linear time-invariant (LTI) state-space model and an input signal $u(t)$, use the Euler method to compute the state trajectories $x_1(t), x_2(t), \cdots$ and the output response $y(t)$ for the time interval $0 \leq t \leq 10$. Show the plots of the state and output variables on the same figure to illustrate how each state affects the system output. Compare the results with the previous question.

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 & -1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} u(t) \\ \\ y(t) = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \end{bmatrix} x(t) \end{cases}$$

$$u(t) = \begin{cases} 1 & 2 \leq t \leq 4 \\ -1 & 6 \leq t \leq 8 \\ 0 & O.W. \end{cases}$$

9 Response of Nonlinear State-Space Models

To numerically find the response of a nonlinear system with state-space representation and an arbitrary input signal using Matlab, you can follow these steps

I) Define the state-space representation of the nonlinear system. This includes the state equations and output equation. Let's assume that the system has three states, $x_1(t)$, $x_2(t)$, and $x_3(t)$, and the input is $u(t)$.

The state equations can be written as

$$\begin{aligned}\frac{dx_1(t)}{dt} &= f_1(x_1(t), x_2(t), x_3(t), u(t))\\ \frac{dx_2(t)}{dt} &= f_2(x_1(t), x_2(t), x_3(t), u(t))\\ \frac{dx_3(t)}{dt} &= f_3(x_1(t), x_2(t), x_3(t), u(t))\end{aligned}$$

The output equation can be written as

$$y(t) = g(x_1(t), x_2(t), x_3(t), u(t))$$

II) Define the time span and initial conditions. You need to specify the time span over which you want to simulate the system and the initial conditions of the states. Let's assume that you want to simulate the system from $t = 0$ to $t = 10$ seconds with an initial condition of $x_1(0) = 0$, $x_2(0) = 0$, and $x_3(0) = 0$

```
Ts = 0.01;
t = 0:0.01:10;
x0 = [0; 0; 0];
X = zeros([length(x0), length(t)]);
X(:, 1) = x0;
```

III) Define the input signal. You need to specify the input signal $u(t)$ that you want to apply to the system. This can be a step function, a sinusoidal function, or any other arbitrary function

IV) Define the functions for the state equations and input equation. You can define function handles for the state equations and input equation using anonymous functions in Matlab. For example, you can define the function handles as follows for the following system

$$\begin{cases}\dot{x}_1(t) = -x_1(t)x_2(t) + \sin(x_3(t))\\[2ex] \dot{x}_2(t) = 3e^{-x_1(t)} - \sqrt{x_3(t)}\\[2ex] \dot{x}_3(t) = -x_3^3(t) + \dfrac{u(t)}{x_1(t) + x_2(t)}\\[6ex] y(t) = x_1^2(t) - x_3(t)\end{cases}$$

```
function dxdt = f(x, u)
     dxdt = [-x(1)*x(2) + sin(x(3));
             3*exp(-x(1)) - sqrt(x(e));
             -x(3)^3 + u/(x(1) + x(2))];
end
```

```
function y = g(x, u)
     y = x(1)^2 - x(3);
end
```

V) Implement the Euler method using a `for` loop

```
for i=2:length(t)
     X(:, i) = X(:, i-1) + Ts * f(X(:, i-1), u(i-1));
     Y(i) = g(X(:, i));
end
```

Assignment: Given a nonlinear state-space model of a fermenter and an input signal $u(t)$, use the Euler method to compute the state trajectories $x_1(t), x_2(t), \cdots$ and the output response $y(t)$ for the time interval $0 \leq t \leq 500$. Show the plots of the state and output variables on the same figure to illustrate how each state affects the system output.

$$\begin{cases} \dot{x}_1(t) = -0.15x_1(t) + \mu(x(t))x_1(t) \\ \dot{x}_2(t) = 0.15(u(t) - x_2(t)) - 2.5\mu(x(t))x_1(t) \\ \dot{x}_3(t) = -0.15x_3(t) + (2.2\mu(x(t)) + 0.2)x_1(t) \\ \mu(x(t)) = \dfrac{0.48(1 - 0.02x_3(t))x_2(t)}{1.2 + x_2(t) + \dfrac{x_2^2(t)}{22}} \\ y(t) = x_3(t) \end{cases}$$

$$u(t) = \begin{cases} 30 & 100 \leq t \leq 200 \\ 20 & 300 \leq t \leq 400 \\ 25 & O.W. \end{cases}$$

10 State-Space Implementation using Simulink

Given a nonlinear state-space model of a fermenter and an input signal $u(t)$, use the Simulink to compute the state trajectories $x_1(t), x_2(t), \cdots$ and the output response $y(t)$ for the time interval $0 \leq t \leq 500$. Show the plots of the state and output variables on the same figure to illustrate how each state affects the system output. Try 3 different sampling times (slow, suitable and fast) and 3 different solvers ($3 \times 3 = 9$ figures). Compare the results with the previous question.

$$
\begin{cases}
\dot{x}_1(t) = -0.15x_1(t) + \mu(x(t))x_1(t) \\[2ex]
\dot{x}_2(t) = 0.15(u(t) - x_2(t)) - 2.5\mu(x(t))x_1(t) \\[2ex]
\dot{x}_3(t) = -0.15x_3(t) + (2.2\mu(x(t)) + 0.2)x_1(t) \\[4ex]
\mu(x(t)) = \dfrac{0.48(1 - 0.02x_3(t))x_2(t)}{1.2 + x_2(t) + \dfrac{x_2^2(t)}{22}} \\[6ex]
y(t) = x_3(t)
\end{cases}
$$

$$
u(t) = \begin{cases} 30 & 100 \leq t \leq 200 \\ 20 & 300 \leq t \leq 400 \\ 25 & O.W. \end{cases}
$$