Control Systems
Lecture 6 State Space Representation

Daro VAN

Laboratory of Dynamics and Control
Department of Industrial and Mechanical Engineering
Institute of Technology of Cambodia

Phnom Penh, May 7th, 2020

Outline

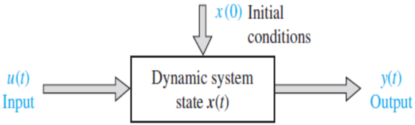
Outline

definition

The State of dynamical system is the minimal set of its variables, whose knowledge enables its future behavior to be predicted under a known input stimuli.

definition

The State of dynamical system is the minimal set of its variables, whose knowledge enables its future behavior to be predicted under a known input stimuli.



- State: The state of a dynamics system is the smallest set of variables (called state variables) such that knowledge of these variables at $t = t_0$, together with knowledge of the input for $t \geqq t_0$, completely determines the behavior of the system for any time $t \geqq t_0$.

- State Variables: The state variables of a dynamical system are the variables making up the smallest set of variables that determine the state of the dynamical system.
- State Vector: if n state variables are needed to completely describe the behavoir of a given system, then these n state variables can be considered the n components of a vector $x \in \mathbb{R}^n$.
- State Space: The n-dimensional space whose coordinate axes consist of x_1 axis, x_2 axis,...,x_n axis, where $x_1, x_2, \ldots, x_n$ are state variables, is called a state space. Any state can be represented by a point in the state space.

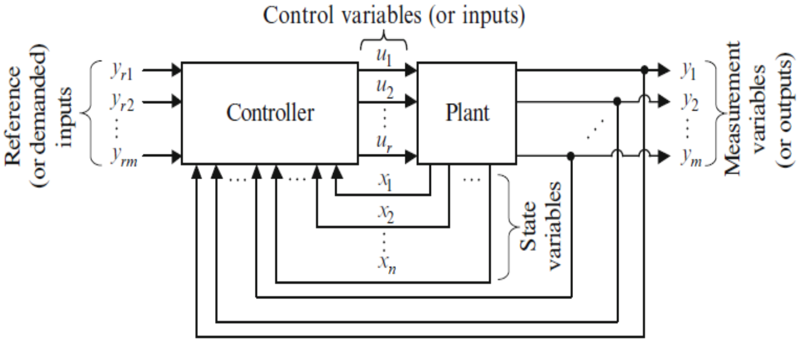
- Know how to form the state-space model

- Know how to form the state-space model
- be able to do numerical integration in the software

- Know how to form the state-space model
- be able to do numerical integration in the software
- know how to transform from state-space form to transfer function and vise versa.

Outline

The General State-Space Model



Assume also that there are r input $u_1(t), u_2(t), ..., u_r(t)$ and m output $y_1(t), y_2(t), ..., y_m(t)$. Define n outputs of the integrators as state variables: $x_1(t), x_2(t), ..., x_n(t)$. Then the system may be described by

$$
\begin{aligned}
\dot{x}_1 &= f_1(x_1, x_2, ..., x_n; u_1, u_2, ..., u_r; t) \\
\dot{x}_2 &= f_2(x_1, x_2, ..., x_n; u_1, u_2, ..., u_r; t) \\
&\vdots \\
\dot{x}_n &= f_n(x_1, x_2, ..., x_n; u_1, u_2, ..., u_r; t),
\end{aligned} \tag{1}
$$

Assume also that there are r input $u_1(t), u_2(t), ..., u_r(t)$ and m output $y_1(t), y_2(t), ..., y_m(t)$. Define n outputs of the integrators as state variables: $x_1(t), x_2(t), ..., x_n(t)$. Then the system may be described by

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2, ..., x_n; u_1, u_2, ..., u_r; t) \\ \dot{x}_2 &= f_2(x_1, x_2, ..., x_n; u_1, u_2, ..., u_r; t) \\ &\vdots \\ \dot{x}_n &= f_n(x_1, x_2, ..., x_n; u_1, u_2, ..., u_r; t), \end{aligned} \tag{1}$$

The outputs $y_1(t), y_2(t), ..., y_m(t)$ of the system may be given by

$$\begin{aligned} y_1(t) &= g_1(x_1, x_2, ..., x_n; u_1, u_2, ..., u_r; t) \\ y_2(t) &= g_2(x_1, x_2, ..., x_n; u_1, u_2, ..., u_r; t) \\ &\vdots \\ y_m(t) &= g_m(x_1, x_2, ..., x_m; u_1, u_2, ..., u_r; t) \end{aligned} \tag{2}$$

The General State-Space Model

For simplicities, Equations (1) and (2) can be written as

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \\ \boldsymbol{y}(t) &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, t) \end{aligned} \tag{3}$$

where the first equation is called state equation, and the second equation is called the output equation.

The General State-Space Model

For simplicities, Equations (1) and (2) can be written as

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \\ \boldsymbol{y}(t) &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, t) \end{aligned} \tag{3}$$

where the first equation is called state equation, and the second equation is called the output equation.
If equations (3) are linearized about the operating state, then we have the following linearized state equation and output equation:

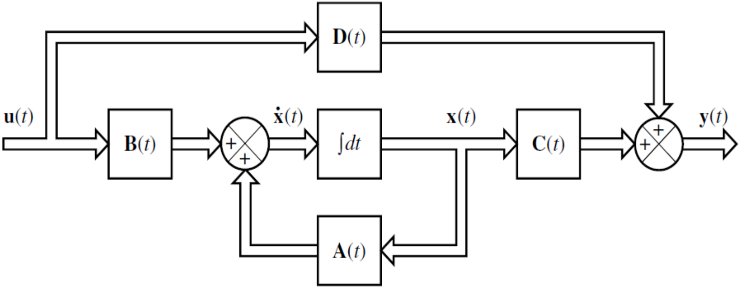
$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}(t)\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}(t)\boldsymbol{x}(t) + \boldsymbol{D}(t)\boldsymbol{u}(t) \end{aligned} \tag{4}$$

where $\boldsymbol{A}(t)$ is called state matrix, $\boldsymbol{B}(t)$ is called input matrix, $\boldsymbol{C}(t)$ is called the output matrix, and $\boldsymbol{D}(t)$ is called the direct transmission matrix.
Note: Later we no longer use bold letter to denote matrix and vector, and t will be dropped for convenient. This means that $\boldsymbol{A}(t) = A, \boldsymbol{x}(t) = x$

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

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

$$\Downarrow$$



Outline

- Step 1: Find equation of motion

Forming a State Space Model

- Step 1: Find equation of motion
- Step 2: Choose state variables (Ex. position and velocity)

Forming a State Space Model

- Step 1: Find equation of motion
- Step 2: Choose state variables (Ex. position and velocity)
- Step 3: Take derivative of state vector

Forming a State Space Model

- Step 1: Find equation of motion
- Step 2: Choose state variables (Ex. position and velocity)
- Step 3: Take derivative of state vector
- Step 4: Write in state-space form

Forming a State Space Model

- Step 1: Find equation of motion
- Step 2: Choose state variables (Ex. position and velocity)
- Step 3: Take derivative of state vector
- Step 4: Write in state-space form
- Step 5: Write output equation

Write the state-space representation for the system described by

$$\ddot{y} + 4\dot{y} + 3y = 2u$$

Forming a State-Space Model

Example

Write the state-space representation for the system described by

$$\ddot{y} + 4\dot{y} + 3y = 2u$$

- Step 1: (given)

Write the state-space representation for the system described by

$$\ddot{y} + 4\dot{y} + 3y = 2u$$

- Step 1: (given)
- Step 2: Choose the state variable

  Choose $x_1 = y$ and $x_2 = \dot{y}$, then the state vector is $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

Forming a State-Space Model

Example

Write the state-space representation for the system described by

$$\ddot{y} + 4\dot{y} + 3y = 2u$$

- Step 1: (given)
- Step 2: Choose the state variable

  Choose $x_1 = y$ and $x_2 = \dot{y}$, then the state vector is $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$
- Step 3: Take the derivative of the state vector.

  $$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Rightarrow \dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{y} \\ 2u - 4\dot{y} - 3y \end{bmatrix}$$

Forming a State-Space Model

Example

Write the state-space representation for the system described by

$$\ddot{y} + 4\dot{y} + 3y = 2u$$

- Step 1: (given)
- Step 2: Choose the state variable

  Choose $x_1 = y$ and $x_2 = \dot{y}$, then the state vector is $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$
- Step 3: Take the derivative of the state vector.

  $$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Rightarrow \dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{y} \\ 2u - 4\dot{y} - 3y \end{bmatrix}$$
- Step 4: Write the state space form.

  $$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ 2u - 4x_2 - 3x_1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -3 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 2 \end{bmatrix} u$$

Write the state-space representation for the system described by

$$\ddot{y} + 4\dot{y} + 3y = 2u$$

- Step 1: (given)
- Step 2: Choose the state variable

  Choose $x_1 = y$ and $x_2 = \dot{y}$, then the state vector is $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$
- Step 3: Take the derivative of the state vector.

  $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Rightarrow \dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{y} \\ 2u - 4\dot{y} - 3y \end{bmatrix}$
- Step 4: Write the state space form.

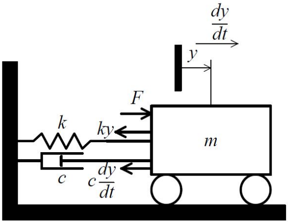
  $\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ 2u - 4x_2 - 3x_1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -3 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 2 \end{bmatrix} u$
- Step 5: Write output equation

  y can be chosen as $y = Cx = \begin{bmatrix} 1 & 0 \end{bmatrix} x$

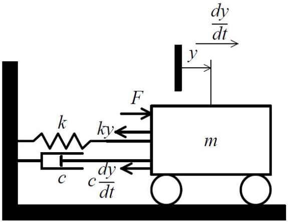
Forming a State-Space Model

Example: Mass Spring Damper

Find the state space representation for the system shown below. Output is displacement of the mass.

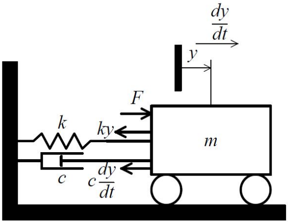


Find the state space representation for the system shown below. Output is displacement of the mass.



- Step 1: find equation of motion
  $m\ddot{y} = f - ky - c\dot{y}$

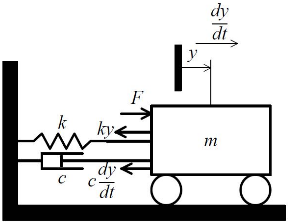
Find the state space representation for the system shown below. Output is displacement of the mass.



- Step 1: find equation of motion
  $m\ddot{y} = f - ky - c\dot{y}$

- Step 2: define state variable
  Choose $x_1 = y, x_2 = \dot{y}$

Find the state space representation for the system shown below. Output is displacement of the mass.

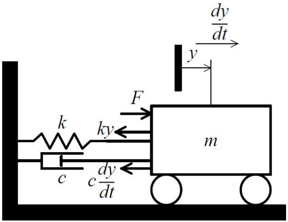


- Step 1: find equation of motion
  $m\ddot{y} = f - ky - c\dot{y}$

- Step 2: define state variable
  Choose $x_1 = y, x_2 = \dot{y}$
- Step 3: Take the derivative

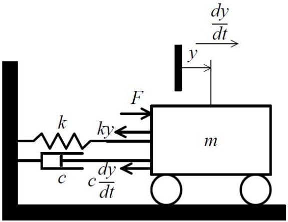
$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{f}{m} - \frac{k}{m}x_1 - \frac{c}{m}x_2 \end{bmatrix}$$

Find the state space representation for the system shown below. Output is displacement of the mass.



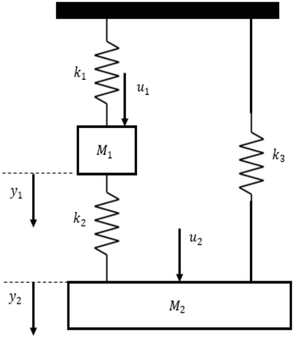
- Step 1: find equation of motion
  $m\ddot{y} = f - ky - c\dot{y}$
- Step 2: define state variable
  Choose $x_1 = y, x_2 = \dot{y}$
- Step 3: Take the derivative
  $$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{f}{m} - \frac{k}{m}x_1 - \frac{c}{m}x_2 \end{bmatrix}$$
- Step 4 : Write the State-space form
  $$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k/m & -c/m \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} f$$

Find the state space representation for the system shown below. Output is displacement of the mass.

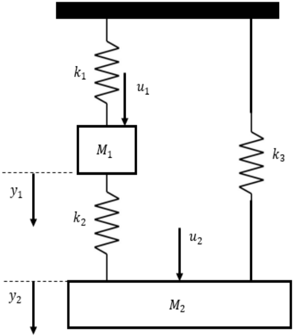


- Step 1: find equation of motion
  $m\ddot{y} = f - ky - c\dot{y}$
- Step 2: define state variable
  Choose $x_1 = y, x_2 = \dot{y}$
- Step 3: Take the derivative
  $$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{f}{m} - \frac{k}{m}x_1 - \frac{c}{m}x_2 \end{bmatrix}$$
- Step 4 : Write the State-space form
  $$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k/m & -c/m \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} f$$
- Step 5: Write the output equation
  $$y = Cx = \begin{bmatrix} 0 & 1 \end{bmatrix} x$$

Find the state space representation for the system shown below. Output is displacement of the mass.

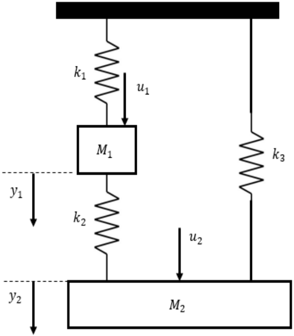


Find the state space representation for the system shown below. Output is displacement of the mass.



- Step 1: find equation of motion
  $M_1\ddot{y}_1 = -(k_1 + k_2)y_1 + k_2y_2 + u_1$
  $M_2\ddot{y}_2 = k_2y_1 - (k_2 + k_3)y_2 + u_2$

Find the state space representation for the system shown below. Output is displacement of the mass.



- Step 1: find equation of motion
  $M_1\ddot{y}_1 = -(k_1 + k_2)y_1 + k_2y_2 + u_1$
  $M_2\ddot{y}_2 = k_2y_1 - (k_2 + k_3)y_2 + u_2$

- Step 2: define state variable
  Choose $x_1 = y_1, x_2 = \dot{y}_1, x_3 = y_2, x_4 = \dot{y}_2$
  The state vector
  $x = x = [x_1, x_2, x_3, x_4]^T = [y_1, \dot{y}_1, y_2, \dot{y}_2]$

- Step 3: take the derivative (do it by yourself)

Forming a State-Space Model

Example: Two Masses with Spring

- Step 3: take the derivative (do it by yourself)
- Step 4: Write the state-space form

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{(k_1+k_2)}{M_1} & 0 & \frac{k_2}{M_1} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_2}{M_2} & 0 - \frac{(k_2+k_3)}{M_2} & 0 & \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ \frac{1}{M_1} & 0 \\ 0 & 0 \\ 0 & \frac{1}{M_2} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

- Step 3: take the derivative (do it by yourself)
- Step 4: Write the state-space form

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{(k_1+k_2)}{M_1} & 0 & \frac{k_2}{M_1} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_2}{M_2} & 0 - \frac{(k_2+k_3)}{M_2} & 0 & \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ \frac{1}{M_1} & 0 \\ 0 & 0 \\ 0 & \frac{1}{M_2} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

- Step 5: Write the output equation

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = Cx = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x$$

Outline

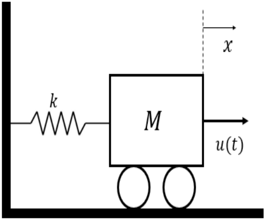
To numerical integrate in MATLAB, we use ode45 (Runge Kutta method)

$$[t, x] = ode45(@f, tspan, x_0)$$

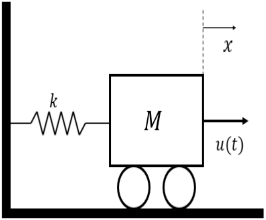
where,

- t = time
- x = state vector
- f = the function that contain state-space information
- $tspan = [t_0, t_f]$ = the time span
- x_0 = Initial conditions

Consider the system shown below



Consider the system shown below



The state space equation can be found as

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -k/m & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} u$$

where k is the spring coefficient, m is a mass and u is the input. The state can be found by numerical integration using MATLAB using the solver(ode45).

Numerical Integration in MATLAB

Example

There are two files. One is the `main.m` and another one is the `f.m` file The code are given.

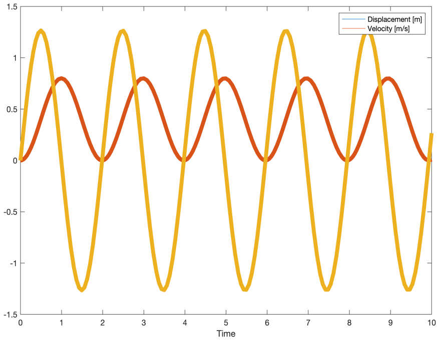
———————————————————————————————————————-

```
main.m
-------
clear all
clc
tspan = [0,10];
x0 = [0,0];
[t,x]= ode45(@f,tspan,x0);
% plot the state
plot(t,x(:,1))
hold on
plot(t,x(:,2))
legend('Displacement(m)','Velocity(m/s)')
```

```
f.m
----
function dx = f(t,x)
% define A, B and u
A = [0,1;-k/m,0];
B = [0;1/m];
u = 2
dx = A*x + B*u;
```

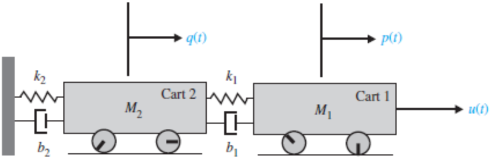
———————————————————————————————————————

The plot of the displacement and the velocity of the system is shown below

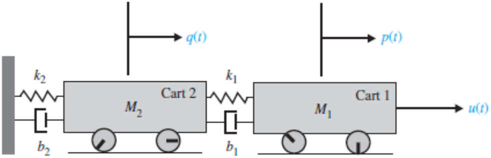


This can also easily be done using Simulink.

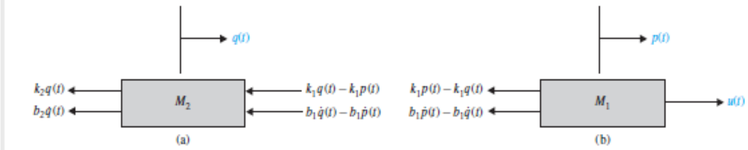
Consider the system shown below. $p(t)$ and $q(t)$ are positions of the carts, $u(t)$ is external force acting on the system, k_1, k_2 are spring constant, b_1, b_2 are damping coefficient.



Consider the system shown below. $p(t)$ and $q(t)$ are positions of the carts, $u(t)$ is external force acting on the system, k_1, k_2 are spring constant, b_1, b_2 are damping coefficient.



$$\Downarrow$$



The equations of motion can be obtained as

$$M_1\ddot{p}(t) + b_1\dot{p}(t) + k_1 p(t) = u(t) + k_1 q(t) + b_1\dot{q}(t) \tag{5}$$

$$M_2\ddot{q}(t) + (k_1 + k_2)q(t) + (b_1 + b_2)\dot{q}(t) = k_1 p(t) + b_1\dot{p}(t) \tag{6}$$

Using the same procedure as described in the previous section, we obtain

The equations of motion can be obtained as

$$M_1\ddot{p}(t) + b_1\dot{p}(t) + k_1 p(t) = u(t) + k_1 q(t) + b_1\dot{q}(t) \tag{5}$$

$$M_2\ddot{q}(t) + (k_1 + k_2)q(t) + (b_1 + b_2)\dot{q}(t) = k_1 p(t) + b_1\dot{p}(t) \tag{6}$$

Using the same procedure as described in the previous section, we obtain

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{7}$$

where $x = [x_1, x_2, x_3, x_4]^T = [p, q, \dot{p}, \dot{q}]^T$

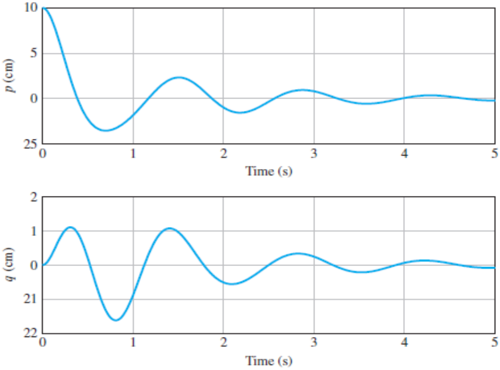
$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k_1/M_1 & k_1/M_1 & -b_1/M_1 & b_1/M_1 \\ k_1/M_2 & -(k_1+k_2)/M_2 & b_1/M_2 & -(b_1+b_2)/M_2 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1/M_1 \\ 0 \end{bmatrix}$$

$$C = [1, 0, 0, 0], D = \mathbf{0}$$

Suppose that the two rolling carts have the following parameter values: $k_1 = 150N/m$; $k_2 = 700N/m$; $b_1 = 15Ns/m$; $b_2 = 30Ns/m$; $M_1 = 5kg$; and $M_2 = 20kg$.

The initial conditions are $p(0) = 10m, q(0) = 0$,and$\dot{p}(0) = \dot{q}(0) = 0$ and there is no input driving force, that is, $u(0) = 0$.

Suppose that the two rolling carts have the following parameter values: $k_1 = 150N/m$; $k_2 = 700N/m$; $b_1 = 15Ns/m$; $b_2 = 30Ns/m$; $M_1 = 5kg$; and $M_2 = 20kg$.
The initial conditions are $p(0) = 10m$, $q(0) = 0$,and$\dot{p}(0) = \dot{q}(0) = 0$ and there is no input driving force, that is, $u(0) = 0$.



Outline

State Space Representation of Scalar Differential Equation Systems

- State space representation of nth-Oder systems of linear differential equations in which the force function does not involve derivative terms:

  Consider the following nth-order systems:

  $$y^{(n)} + a_1 y^{n-1} + \ldots + a_{n-1}\dot{y} + a_n y = u$$

  The knowledge of $y(0), \dot{y}(0), \ldots, y^{(n-1)}(0)$, together with the input $u(t)$ for $t \geqq 0$, determines completely the future behavior of the system, we may take $y(t), \dot{y}(t), \ldots, y^{(n-1)}(t)$ as a set of n state variables.

State Space Representation of Scalar Differential Equation Systems

Let define

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} y \\ dy/dt \\ \vdots \\ d^{n-2}y/dt^{n-2} \\ d^{n-1}y/dt^{n-1} \end{bmatrix}$$

State Space Representation of Scalar Differential Equation Systems

Let define

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} y \\ dy/dt \\ \vdots \\ d^{n-2}y/dt^{n-2} \\ d^{n-1}y/dt^{n-1} \end{bmatrix}$$

Then,

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ -a_0x_1 - a_1x_2 - \ldots - a_nx_n + u \end{bmatrix}$$

State Space Representation of Scalar Differential Equation Systems

and the state-space equations become

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots \\ -a_n & -a_{n-1} & a_{n-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} u$$

The output can be given by

$$y = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Its corresponding TF is $\frac{Y(s)}{U(s)} = \frac{1}{s^n + a_1 s^{n-1} + \dots + a_{n-1}s + a_n}$

State Space Representation of Scalar Differential Equation Systems

- State space representation of nth order systems of linear differential equations in which the forcing function involves derivative terms:

Consider the differential equation system that involves derivatives of the forcing function such as

$$y^{(n)} + a_1^{n-1} y + \dots + a_{n-1}\dot{y} + a_n y = b_0 u^n + b_1 u^{(n-1)} + \dots + b_{n-1}\dot{u} + b_n u$$

State Space Representation of Scalar Differential Equation Systems

- State space representation of nth order systems of linear differential equations in which the forcing function involves derivative terms:

Consider the differential equation system that involves derivatives of the forcing function such as

$$y^{(n)} + a_1^{n-1}y + \dots + a_{n-1}\dot{y} + a_n y = b_0 u^n + b_1 u^{(n-1)} + \dots + b_{n-1}\dot{u} + b_n u$$

One way to obtain a state equation and output equation for this case is to define the following n variables as a set of n state variable:

$$\begin{aligned}
x_1 &= y - \beta_0 u \\
x_2 &= \dot{y} - \beta_0 \dot{u} - \beta_1 u = \dot{x}_1 - \beta_1 u \\
&\vdots \\
x_n &= y^{(n-1)} - \beta_0 u^{(n-1)} - \beta_1 u^{(n-2)} - \dots - \beta_{n-2}\dot{u} - \beta_{n-1}u = \dot{x}_{n-1} - \beta_{n-1}u
\end{aligned}$$

State Space Representation of Scalar Differential Equation systems

where $\beta_0, \beta_1, \beta_3, ..., \beta_{n-1}$ are determine from

$$\begin{aligned}
\beta_0 &= b_0 \\
\beta_1 &= b_1 - a_1\beta_0 \\
\beta_2 &= b_2 - a_1\beta_1 - a_2\beta_0 \\
\beta_3 &= b_3 - a_1\beta_2 - a_2\beta_1 - a_3\beta_0 \\
&\vdots \\
\beta_{(n-1)} &= b_{(n-1)} - a_1\beta_{n-2} - ... - a_{(n-2)}\beta_1 - a_{(n-1)}\beta_0
\end{aligned}$$

With this choice of state variables the existence and uniqueness of the solution of the state equation is guaranteed. (Note that this is not the only choice of a set of state variables.)

State Space Representation of Scalar Differential Equation Systems

With the present choice of state variables, we obtain

$$\begin{aligned}
\dot{x}_1 &= x_2 + \beta_1 u \\
\dot{x}_2 &= x_3 + \beta_2 u \\
&\vdots \\
\dot{x}_{n-1} &= x_n + \beta_{n-1} u \\
\dot{x}_n &= -a_n x_1 - a_{n-1} x_2 - \ldots - a_1 x_n + \beta_n u
\end{aligned}$$

where, β_n is given by

$$\beta_n = b_n - a_1 \beta_{n-1} - \ldots - a_{n-1}\beta_1 - a_{n-1}\beta_0$$

State Space Representation of Scalar Differential Equation Systems

In terms of vector-matrix equations the state equation and the output equation can be written as

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots \\ -a_n & -a_{n-1} & a_{n-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_n \end{bmatrix} u$$

State Space Representation of Scalar Differential Equation Systems

In terms of vector-matrix equations the state equation and the output equation can be written as

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots \\ -a_n & -a_{n-1} & a_{n-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_n \end{bmatrix} u$$

and,

$$y = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \beta_0 u$$

State Space Representation of Scalar Differential Equation Systems

In terms of vector-matrix equations the state equation and the output equation can be written as

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots \\ -a_n & -a_{n-1} & a_{n-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_n \end{bmatrix} u$$

and,

$$y = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \beta_0 u$$

Its corresponding TF is $\frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \ldots + b_{n-1} s + b_n}{s^n + a_1 s^{n-1} + \ldots + a_{n-1} s + a_n}$

Outline

Transformation from Transfer Function to State-Space

Find the state space of the system

$$\frac{Y(s)}{U(s)} = \frac{100}{s^4 + 20s^3 + 10s^2 + 7s + 100}$$

Transformation from Transfer Function to State-Space

Find the state space of the system

$$\frac{Y(s)}{U(s)} = \frac{100}{s^4 + 20s^3 + 10s^2 + 7s + 100}$$

From the above equation, we obtain

$$(s^4 + 20s^3 + 10s^2 + 7s + 100)Y(s) = 100U(s)$$

Transformation from Transfer Function to State-Space

Find the state space of the system

$$\frac{Y(s)}{U(s)} = \frac{100}{s^4 + 20s^3 + 10s^2 + 7s + 100}$$

From the above equation, we obtain

$$(s^4 + 20s^3 + 10s^2 + 7s + 100)Y(s) = 100U(s)$$

Take the inverse Laplace transform, then we obtain

$$y^4 + 20y^3 + 10\ddot{y} + 7\dot{y} + 100y = 100u$$

Transformation from Transfer Function to State-Space

Find the state space of the system

$$\frac{Y(s)}{U(s)} = \frac{100}{s^4 + 20s^3 + 10s^2 + 7s + 100}$$

From the above equation, we obtain

$$(s^4 + 20s^3 + 10s^2 + 7s + 100)Y(s) = 100U(s)$$

Take the inverse Laplace transform, then we obtain

$$y^4 + 20y^3 + 10\ddot{y} + 7\dot{y} + 100y = 100u$$

Define

$$x_1 = y, x_2 = \dot{y}, x_3 = \ddot{y}, x_4 = y^{(3)} \Rightarrow \dot{x}_1 = x_2, \dot{x_2} = x_3, \dot{x}_3 = x_4$$

Transformation from Transfer Function to State-Space

Find the state space of the system

$$\frac{Y(s)}{U(s)} = \frac{100}{s^4 + 20s^3 + 10s^2 + 7s + 100}$$

From the above equation, we obtain

$$(s^4 + 20s^3 + 10s^2 + 7s + 100)Y(s) = 100U(s)$$

Take the inverse Laplace transform, then we obtain

$$y^4 + 20y^3 + 10\ddot{y} + 7\dot{y} + 100y = 100u$$

Define

$$x_1 = y, x_2 = \dot{y}, x_3 = \ddot{y}, x_4 = y^{(3)} \Rightarrow \dot{x}_1 = x_2, \dot{x_2} = x_3, \dot{x}_3 = x_4$$

Then the state-space form is given as

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_3 \\ x_4 \\ -20x_4 - 10x_3 - 7x_2 - 100x_1 + 100u \end{bmatrix}$$

We obtain

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -100 & -7 & -10 & -20 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 100 \end{bmatrix} r$$

We obtain

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -100 & -7 & -10 & -20 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 100 \end{bmatrix} r$$

and,

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Consider the transfer function systems given as

$$\frac{Y(s)}{U(s)} = \frac{s}{s^3+14^2+56s+160}$$

There are my possible state space representations for this system. One possible state space representation is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -160 & -56 & -14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ -14 \end{bmatrix} u$$

and,

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [0]u$$

Transformation from Transfer Function to State Space

Let us write the closed loop transfer function as

$$\frac{Y(s)}{U(s)} = \frac{num}{den}$$

Once we have this transfer function expression,

Transformation from Transfer Function to State Space

Let us write the closed loop transfer function as

$$\frac{Y(s)}{U(s)} = \frac{num}{den}$$

Once we have this transfer function expression, the MATLAB command

$$[A, B, C, D] = tf2ss(num, den)$$

will give a state space representation. It is important that the state space representation for any system is not unique. There are many (infinitely many) state space representation for the same system.

Transformation from Transfer Function to State Space

Let us write the closed loop transfer function as

$$\frac{Y(s)}{U(s)} = \frac{num}{den}$$

Once we have this transfer function expression, the MATLAB command

$$[A, B, C, D] = tf2ss(num, den)$$

will give a state space representation. It is important that the state space representation for any system is not unique. There are many (infinitely many) state space representation for the same system.

```
MATLAB PROGRAM FOR THE ABOVE EXAMPLE
-----------------------
clear all
clc
num = [1, 0];
den = [1, 14,56, 160]
[A,B,C,D] = tf2ss(num, den)
```

Outline

State-Space to Transfer Function

Let us consider the system whose transfer function is given by

$$\frac{Y(s)}{U(s)} = G(s) \tag{8}$$

This system may be represented in state space by the following equations

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{9}$$

where x is the state vector, u is the input and y is the output.

State-Space to Transfer Function

Let us consider the system whose transfer function is given by

$$\frac{Y(s)}{U(s)} = G(s) \tag{8}$$

This system may be represented in state space by the following equations

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{9}$$

where x is the state vector, u is the input and y is the output. The Laplace transform of (9) are given by

$$\begin{aligned} sX(s) - X(0) &= AX(s) + BU(s) \\ Y(s) &= CX(s) + DU(s) \end{aligned} \tag{10}$$

Transformation from State Space to Transfer Function

Since the transfer function was previously defined as the ratio of the Laplace transform of the output to the Laplace transform of the input when the initial conditions were zero, we set $X(0)$ to be zero.

Transformation from State Space to Transfer Function

Since the transfer function was previously defined as the ratio of the Laplace transform of the output to the Laplace transform of the input when the initial conditions were zero, we set $X(0)$ to be zero. Then, we have

$$sX(s) - AX(s) = BU(s) \text{ or } (sI - A)X(s) = BU(s)$$

Transformation from State Space to Transfer Function

Since the transfer function was previously defined as the ratio of the Laplace transform of the output to the Laplace transform of the input when the initial conditions were zero, we set $X(0)$ to be zero. Then, we have

$$sX(s) - AX(s) = BU(s) \text{ or } (sI - A)X(s) = BU(s)$$

By pre-multiplying $(sI - A)^{-1}$ to both sides of the last equation, we obtain

$$X(s) = (sI - A)^{-1}BU(s) \tag{11}$$

Transformation from State Space to Transfer Function

Since the transfer function was previously defined as the ratio of the Laplace transform of the output to the Laplace transform of the input when the initial conditions were zero, we set $X(0)$ to be zero. Then, we have

$$sX(s) - AX(s) = BU(s) \text{ or } (sI - A)X(s) = BU(s)$$

By pre-multiplying $(sI - A)^{-1}$ to both sides of the last equation, we obtain

$$X(s) = (sI - A)^{-1}BU(s) \tag{11}$$

By substitute Equation (11) into the output equation of (9), we get

$$Y(s) = [C(sI - A)^{-1}B + D]U(s) \tag{12}$$

Transformation from State Space to Transfer Function

Since the transfer function was previously defined as the ratio of the Laplace transform of the output to the Laplace transform of the input when the initial conditions were zero, we set $X(0)$ to be zero. Then, we have

$$sX(s) - AX(s) = BU(s) \text{ or } (sI - A)X(s) = BU(s)$$

By pre-multiplying $(sI - A)^{-1}$ to both sides of the last equation, we obtain

$$X(s) = (sI - A)^{-1}BU(s) \tag{11}$$

By substitute Equation (11) into the output equation of (9), we get

$$Y(s) = [C(sI - A)^{-1}B + D]U(s) \tag{12}$$

then,

$$G(s) = C(sI - A)^{-1}B + D \tag{13}$$

Transformation from State Space to Transfer Function

Since the transfer function was previously defined as the ratio of the Laplace transform of the output to the Laplace transform of the input when the initial conditions were zero, we set $X(0)$ to be zero. Then, we have

$$sX(s) - AX(s) = BU(s) \text{ or } (sI - A)X(s) = BU(s)$$

By pre-multiplying $(sI - A)^{-1}$ to both sides of the last equation, we obtain

$$X(s) = (sI - A)^{-1}BU(s) \tag{11}$$

By substitute Equation (11) into the output equation of (9), we get

$$Y(s) = [C(sI - A)^{-1}B + D]U(s) \tag{12}$$

then,

$$G(s) = C(sI - A)^{-1}B + D \tag{13}$$

This is the transfer function express in terms of A, B, C and D.

Find the transfer function given the following:

$$\dot{x} = \begin{bmatrix} -3 & 1 \\ -2 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \qquad y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

Answer:

$$G(s) = \frac{1}{(s+2)(s+1)}$$

Transformation from State Space to Transfer Function

To obtain the transfer function from state space equations in MATLAB, use the following command:

```
[num,den]=ss2tf(A,B,C,D,iu)
```

`iu` must be specified for systems with more than one input. For example,if the system has three inputs (u_1, u_2, u_3), then `iu` must be either 1, 2, or 3, where 1 implies u_1, 2 implies u_2, and 3 implies u_3.
if the system has only one input, then either

```
[num,den]=ss2tf(A,B,C,D)
```

may be used.

Obtain the transfer function of the system defined by the following state-space equations

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5 & -25 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 25 \\ -120 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Transformation from State Space to Transfer Function

Example

Obtain the transfer function of the system defined by the following state-space equations

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5 & -25 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 25 \\ -120 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

We can easily obtain the transfer function using MATLAB.

```
MATLAB Program
A = [0,1,0;0,0,1;-5,-25,-5];
B = [0;25;120];
C = [1,0,0];
D = [0]
[num,den]=ss2tf(A,B,C,D)
```

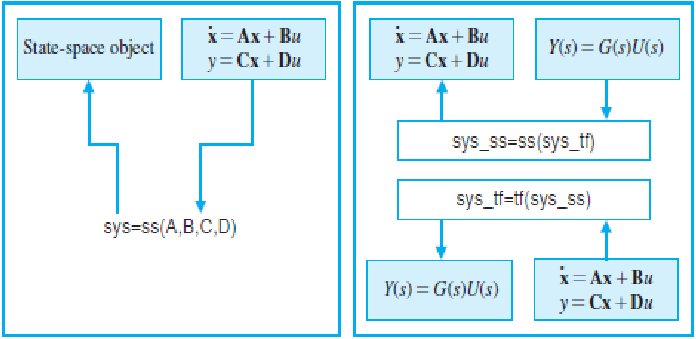
The command

```
[num, den]=ss2tf(A,B,C,D,1)
```

gave the same result.

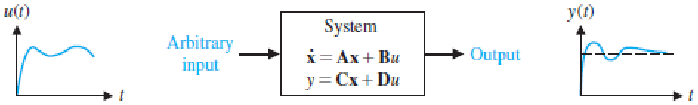
$$\frac{Y(s)}{U(s)} = \frac{25s+5}{s^3+5s^2+25s+5}$$

State-Space to Transfer Function and Vice Versa



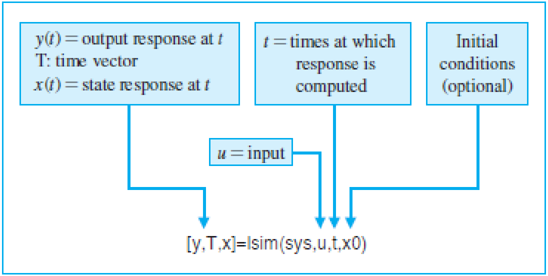
Example: Obtain the state-space representation of:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{2s^2+8s+6}{s^3+8s^2+16s+6}$$



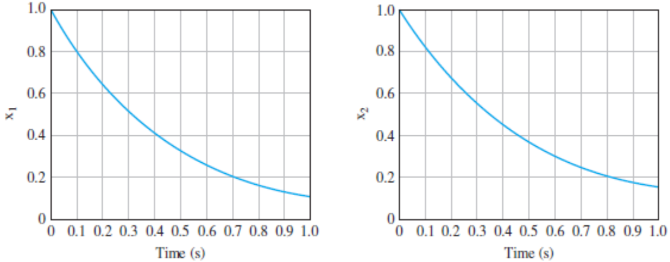
$u(t)$
t
Arbitrary input
System
$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{B}u$
$y = \mathbf{Cx} + \mathbf{D}u$
Output
$y(t)$
t

(a)



$y(t)$ = output response at t
T: time vector
$x(t)$ = state response at t
t = times at which response is computed
Initial conditions (optional)
u = input
[y,T,x]=lsim(sys,u,t,x0)

(b)




```
A=[0 -2;1 -3]; B=[2;0]; C=[1 0]; D=[0];
sys=ss(A,B,C,D);
x0=[1 1];
t=[0:0.01:1];
u=0*t;
[y,T,x]=lsim(sys,u,t,x0);
subplot(121), plot(T,x(:,1))
xlabel('Time (s)'), ylabel('x_1')
subplot(122), plot(T,x(:,2))
xlabel('Time (s)'), ylabel('x_2')
```

State-space model

Initial conditions

Zero input

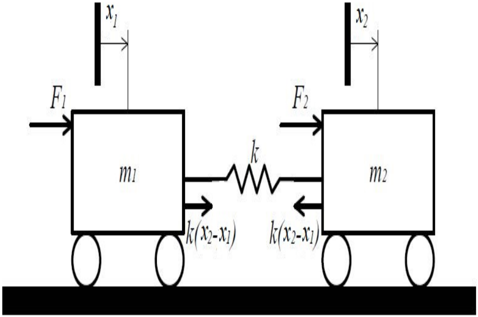
Outline

Homework

Consider a two-mass system as shown below. The spring constant between the two masses is k, and the masses are m_1, m_2. When the displacement from the equilibrium point of the mass is y_1, y_2, and the external forces are F_1, F_2, respectively. The equation of motion is given as

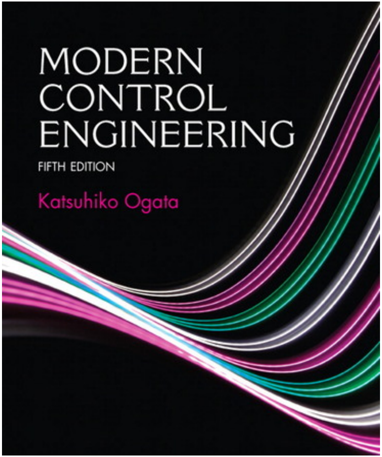
$$\begin{aligned} m_1\frac{d^2x_1}{dt^2} - k(x_2 - x_1) = F_1 \\ m_2\frac{d^2x_2}{dt^2} + k(x_2 - x_1) == F_2 \end{aligned} \tag{14}$$

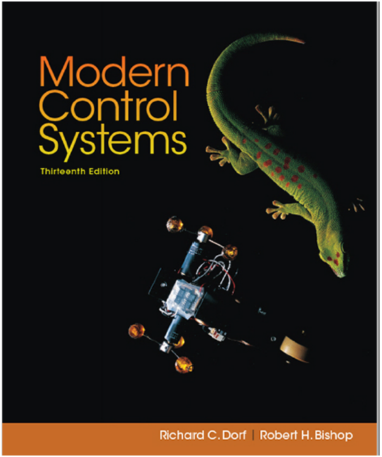
Homework



Find the state space equation of the system.

References


MODERN
CONTROL
ENGINEERING
FIFTH EDITION
Katsuhiko Ogata


Modern
Control
Systems
Thirteenth Edition
Richard C. Dorf | Robert H. Bishop