Introduction

- There are two main approaches for modelling and design of feedback control systems.
- So far, we have considered only the frequency-domain technique.
- This approach unfortunately can only be applied to single-input, single-output, linear, time-invariant systems or ones that can be approximated by one.
- The more modern, flexible approach is called the state space representation (also called *time-domain technique*).
- This method can also be applied to nonlinear systems, time-varying systems, as well as multiple-input, multiple output systems.

State Space Representation

- For $t \geq t_o$ and initial conditions $\mathbf{x}(t_o)$, the state space representation of a system is:

$$\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} \qquad \textit{state equations} \qquad (1)$$

$$\mathbf{y} = \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} \qquad \textit{output equations} \qquad (2)$$

State equations: for an n^{th} order system, this is a set of n simultaneous, first-order differential equations with n variables, that can be solved to determine the system's n state variables.

For a linear, time-invariant, second order system with a single input $v(t)$, the state equations could have the form:

$$\frac{dx_1}{dt} = a_{11}x_1 + a_{12}x_2 + b_1 v(t)$$

$$\frac{dx_2}{dt} = a_{21}x_1 + a_{22}x_2 + b_2 v(t)$$

State Space Representation - II

$$\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} \qquad \textit{state equations}$$

$$\mathbf{y} = \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} \qquad \textit{output equations}$$

System variables: variables that respond to a system input, or the system's initial conditions.

Linearly independent: if no variables of a set can be written as a linear combination of the other variables, then the set of variables are said to be linearly independant.

State variables: smallest set of linearly independent system variables such that the initial values of these variables (at time t_o) plus any known forcing functions completely determines the future values of all system variables.

State vector: $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ where $x_1, x_2, \ldots, x_n$ are the system's n state variables.

State Space Representation - III

$$\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} \qquad \textit{state equations}$$

$$\mathbf{y} = \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} \qquad \textit{output equations}$$

First derivatives: $\dot{\mathbf{x}} = \frac{d}{dt}\mathbf{x} = [\frac{dx_1}{dt}, \frac{dx_2}{dt}, \ldots, \frac{dx_n}{dt}]^T$

Output vector: $\mathbf{y} = [y_1, y_2, \ldots, y_p]^T$

Input or control vector: $\mathbf{u} = [u_1, u_2, \ldots, u_m]^T$

System matrix: $\mathbf{A}$

Input matrix: $\mathbf{B}$

Output matrix: $\mathbf{C}$

Feedforward matrix: $\mathbf{D}$

State Space Representation eg.

- Derive the state space representation for the system below:

- Using Kirchoff's voltage law, we can write the loop equation:

$$L\frac{di}{dt} + Ri + \frac{1}{C}\int idt = v(t) \tag{3}$$

- If we use $i(t) = \frac{dq}{dt}$, we can see that the system is a second order system:

$$L\frac{d^2q}{dt^2} + R\frac{dq}{dt} + \frac{1}{C}q = v(t) \tag{4}$$

- If we take our state variables to be $i(t)$ and $q(t)$, we can convert equation 4 into two first order differential equations.

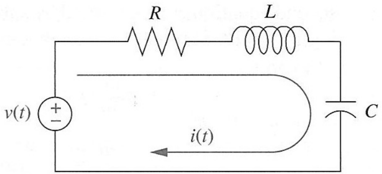


Figure 3.2

State Space Representation eg. II

- We can take the first equation to be:

$$\frac{dq}{dt} = i \qquad (5)$$

- We can get the second eqn by substituting $\int idt = q$ into equation 3 and solving for $\frac{di}{dt}$ gives:

$$\frac{di}{dt} = -\frac{1}{LC}q - \frac{R}{L}i + \frac{1}{L}v(t) \qquad (6)$$

- As our output, we can take the voltage across the inductor, $v_L(t)$.

- Using equation 6 and the relation $v_L(t) = L\frac{di}{dt}$, we get:

$$v_L(t) = -\frac{1}{C}q - Ri + v(t) \qquad (7)$$

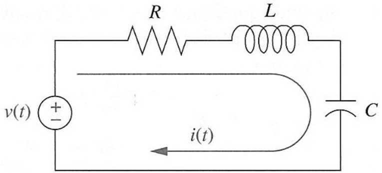


Figure 3.2

State Space Representation eg. III

- We thus have our state equations:

$$\frac{dq}{dt} = i$$
$$\frac{di}{dt} = -\frac{1}{LC}q - \frac{R}{L}i + \frac{1}{L}v(t)$$

which can be represented as $\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,u$, where

$$\dot{\mathbf{x}} = \begin{bmatrix} dq/dt \\ di/dt \end{bmatrix}; \quad \mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1/LC & -R/L \end{bmatrix}$$
$$\mathbf{x} = \begin{bmatrix} q \\ i \end{bmatrix}; \qquad \mathbf{B} = \begin{bmatrix} 0 \\ 1/L \end{bmatrix}; u = v(t)$$

- and our output equation $v_L(t) = -\frac{1}{C}q - Ri + v(t)$ which can be represented as $y = \mathbf{C}\,\mathbf{x} + Du$, where

$$y = v_L(t); \quad \mathbf{C} = [-1/C \quad -R]; \quad D = 1;$$

Applying State Space Representation

- First step is to select the state vector.

- In choosing the state vector, one must make sure
  1. The state variables are linearly independent.

  2. A minimum number of state variables must be chosen that is sufficient to completely describe the system.

     The minimum number is the order of the differential equation that describes the system.

     This is equivalent to the order of the denominator of the transfer function after cancelling any common factors in both the numerator and denominator.

     The number needed is usually equal to the number of independent storage elements.

Representing an Electrical Network eg.

- Find a state space representation for the network below with output $i_R(t)$, the current through the resistor.

1. Label branch currents in network (i_L, i_R, i_C).
2. Write derivative equations for all energy storing elements. Select state variables to be the quantities that are differentiated.
3. Rewrite the derivative equations in terms of the state variables.
4. Solve for the output in terms of input and state variables.
5. Express in state space form.

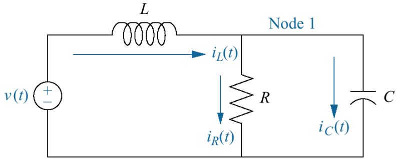
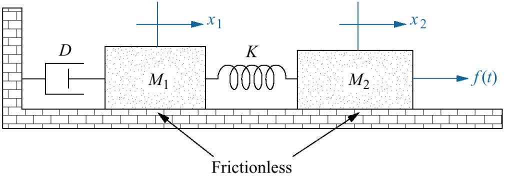


Figure 3.5

Representing a Translational Mechanical System eg.

- Find a state space representation for the system below, if the output is $x_2(t)$.
- For mechanical systems, it is easier to use equations of motions to derive state variables.
- For state variables, use the position and velocity of each linearly independent point of motion.
- Use relations $\dfrac{d^2x}{dt^2} = \dfrac{dv}{dt}$, and $v = \dfrac{dx}{dt}$.

Figure 3.7



Converting a Transfer Function to State Space

- So far, we have derived state space representations directly from the physical system.
- We now examine how to derive a state space representation if we are given a transfer function representation of a system.
- We will use the phase variable approach.
- Assume you are given a differential equation of the form below, where y is the system's output, and u is the system's input.

$$\frac{d^n y}{dt^n} + a_{n-1}\frac{d^{n-1} y}{dt^{n-1}} + \cdots + a_1 \frac{dy}{dt} + a_0 y = b_0 u \qquad (8)$$

- We choose y and its $n-1$ derivatives as our n state variables.

Converting a Transfer Function to State Space - II

- Thus for our state variables $x_1, \ldots, x_n$, we get:

$$x_1 = y,\ x_2 = \frac{dy}{dt},\ x_3 = \frac{d^2y}{dt^2}, \cdots, x_n = \frac{d^{n-1}y}{dt^{n-1}} \tag{9}$$

- Taking the derivatives of both sides of these equations gives:

$$\dot{x_1} = \frac{dy}{dt},\ \dot{x_2} = \frac{d^2y}{dt^2},\ \dot{x_3} = \frac{d^3y}{dt^3}, \cdots, \dot{x_n} = \frac{d^ny}{dt^n} \tag{10}$$

- Substituting into equation 10 from equation 9, as well as solving for $\dot{x_n} = \frac{d^ny}{dt^n}$ in equation 8 gives:

$$\begin{aligned} \dot{x_1} = x_2,\ \dot{x_2} = x_3,\ \dot{x_3} = x_4, \cdots, \dot{x}_{n-1} = x_n, \\ \dot{x_n} = -a_o x_1 - a_1 x_2 \cdots - a_{n-1} x_n + b_0 u \end{aligned} \tag{11}$$

Converting a Transfer Function to State Space - III

- Putting the state equations in matrix form gives:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \vdots \\ \dot{x}_{n-1} \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ \vdots & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & -a_3 & -a_4 & -a_5 & \cdots & -a_{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ b_0 \end{bmatrix} u$$

- Using that our ouput $y(t)$ equals x_1, gives:

$$y = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix}$$

$$\frac{d^n y}{dt^n} + a_{n-1}\frac{d^{n-1} y}{dt^{n-1}} + \cdots + a_1\frac{dy}{dt} + a_0 y = b_0 u$$

Transfer Function to State Space eg.

- Convert the transfer function below into a state space representation.
- As numerator is not a constant, we need to first split the transfer function into two cascading boxes, such that the first has a constant numerator.
- We can now apply the phase variable approach to the first box, with $X_1(s)$ as its output.
- To determine the system's output equation, solve for $C(s)$ in terms of the state variables by evaluating output of second block.

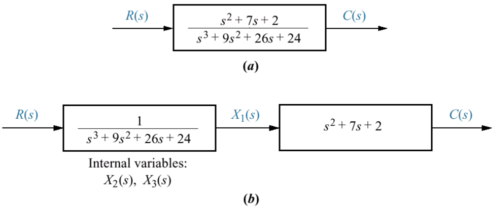


Figure 3.12.

Transfer Function to State Space eg. - I

- Figure shows the state space representation as a block diagram by using integrator blocks.

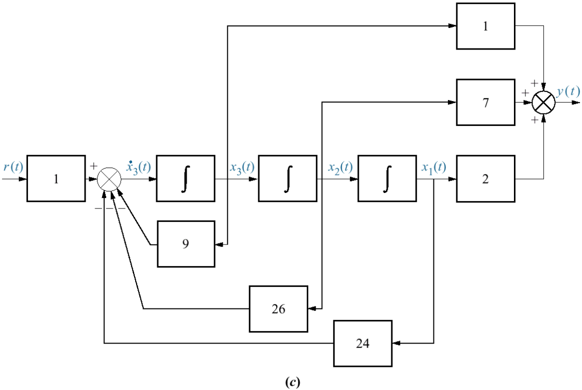


Figure 3.12.

$$\dot{x_3} = -24x_1 - 26x_2 - 9x_3 + r, \quad y = 2x_1 + 7x_2 + x_3$$

Transfer Function to State Space: Controller Form

Given

$$G(s) = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \ldots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \ldots + a_1 s + a_0}$$

We get Controller Canonical Statespace form:

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 & 0 \\ 0 & 0 & 1 & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 & 0 \\ 0 & 0 & 0 & \ldots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \ldots & -a_{n-2} & -a_{n-1} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{u}(t) \\
\mathbf{y}(t) &= \begin{bmatrix} b_0 & b_1 & \ldots & b_{n-2} & b_{n-1} \end{bmatrix} \mathbf{x}(t).
\end{aligned}$$

Transfer Function to Controller Form in $n = 4$ Case

Consider the following transfer function:

$$G(s) = \frac{b_3 s^3 + b_2 s^2 + b_1 s + b_0}{s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}$$

The state-space Controller Canonical Form for the transfer function is:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 & -a_3 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{u}(t) \\ \mathbf{y}(t) &= \begin{bmatrix} b_0 & b_1 & b_2 & b_3 \end{bmatrix} \mathbf{x}(t). \end{aligned}$$

This state-space realization is called *controllable canonical form* because the resulting model is guaranteed to be controllable (i.e., because the control enters a chain of integrators, it has the ability to move every state).

Transfer Function to State Space: Observer Form

Given

$$G(s) = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \ldots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \ldots + a_1 s + a_0}$$

We get *Observer Canonical Statespace Form*:

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \begin{bmatrix} -a_{n-1} & 1 & 0 & \ldots & 0 & 0 \\ -a_{n-2} & 0 & 1 & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ -a_2 & 0 & 0 & \ldots & 1 & 0 \\ -a_1 & 0 & 0 & \ldots & 0 & 1 \\ -a_0 & 0 & 0 & \ldots & 0 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} b_{n-1} \\ b_{n-2} \\ \vdots \\ b_2 \\ b_1 \\ b_0 \end{bmatrix} \mathbf{u}(t) \\
\mathbf{y}(t) &= \begin{bmatrix} 1 & 0 & \ldots & 0 & 0 & 0 \end{bmatrix} \mathbf{x}(t)
\end{aligned}$$

Observer Cannonical Form $n = 4$ Case

Consider the following transfer function:

$$\mathbf{G}(s) = \frac{b_3 s^3 + b_2 s^2 + b_1 s + b_0}{s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}$$

The state-space Observer Canonical Form for the transfer function is:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \begin{bmatrix} -a_3 & 1 & 0 & 0 \\ -a_2 & 0 & 1 & 0 \\ -a_1 & 0 & 0 & 1 \\ -a_0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} b_3 \\ b_2 \\ b_1 \\ b_0 \end{bmatrix} \mathbf{u}(t) \\ \mathbf{y}(t) &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}(t) \end{aligned}$$

This state-space realization is called *observable canonical form* because the resulting model is guaranteed to be observable (i.e., because the output exits from a chain of integrators, every state has an effect on the output).

Converting from State Space to a Transfer Function

- We now examine how to derive a transfer function if we are given a state space representation of a system.
- We assume we are given the state and output equations below

$$\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} \qquad \textit{state equations}$$
$$\mathbf{y} = \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} \qquad \textit{output equations}$$

- We first note that

$$\mathcal{L}\{\mathbf{x}(t)\} = \mathcal{L}\left\{\begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix}\right\} = \begin{bmatrix} \mathcal{L}\{x_1(t)\} \\ \vdots \\ \mathcal{L}\{x_n(t)\} \end{bmatrix}$$

- Taking the Laplace transform of both sides, assuming zero initial conditions gives:

$$s\mathbf{X}(s) = \mathbf{A}\,\mathbf{X}(s) + \mathbf{B}\,\mathbf{U}(s) \qquad (12)$$
$$\mathbf{Y}(s) = \mathbf{C}\,\mathbf{X}(s) + \mathbf{D}\,\mathbf{U}(s) \qquad (13)$$

Converting from State Space ... - II

- Collecting $\mathbf{X}(s)$ terms in equation 12 gives:

$$(s\mathbf{I} - \mathbf{A})\mathbf{X}(s) = \mathbf{B}\,\mathbf{U}(s) \tag{14}$$

  where $\mathbf{I}$ is the identity matrix.
- Solving for $\mathbf{X}(s)$ gives:

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\,\mathbf{U}(s) \tag{15}$$

- substituting into our output equation, gives:

$$\mathbf{Y}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\,\mathbf{U}(s) + \mathbf{D}\,\mathbf{U}(s) \tag{16}$$

$$= [\mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}]\mathbf{U}(s) \tag{17}$$

- If we have a single input, single output system, we get the transfer function:

$$G(s) = \frac{Y(s)}{U(s)} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \tag{18}$$

Converting from State Space eg.

- Given the system below, find the transfer function $\frac{Y(s)}{U(s)}$.

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -24 & -26 & -9 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$

$$y = [2 \;\; 7 \;\; 1]\, \mathbf{x}$$

Matrix Review - Cofactors

- Given the matrix below, we say the minor of entry a_{ij}, denoted by M_{ij}, is the determinant of the matrix that remains after row i and column j are deleted.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

- For example, $M_{11} = \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix}$ and $M_{32} = \begin{vmatrix} a_{11} & a_{13} \\ a_{21} & a_{23} \end{vmatrix}$.
- We define the cofactor of entry a_{ij}, denoted C_{ij}, to be the number $(-1)^{i+j}M_{ij}$.

Matrix Review - Cofactor Expansion

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Definition

The determinant of an $n \times n$ matrix $\mathbf{A}$ can be computed by multiplying the entries in any row (or column) by their cofactors and adding the resulting products. We thus have for $1 \leq i \leq n$ and $1 \leq j \leq n$,

cofactor expansion along the jth column

$$\det(\mathbf{A}) = a_{1j}C_{1j} + a_{2j}C_{2j} + \cdots + a_{nj}C_{nj}$$

cofactor expansion along the ith row

$$\det(\mathbf{A}) = a_{i1}C_{i1} + a_{i2}C_{i2} + \cdots + a_{in}C_{in}$$

Matrix Review - Adjoints

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

Definition

If $\mathbf{A}$ is any $n \times n$ matrix and C_{ij} is the cofactor for a_{ij}, then the matrix of cofactors from $\mathbf{A}$ is

$$\begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & & \vdots \\ C_{n1} & C_{n2} & \cdots & C_{nn} \end{bmatrix}$$

The adjoint of $\mathbf{A}$, denoted adj$(\mathbf{A})$, is the transpose of the above matrix.

Matrix Review - Inverses

Definition
The inverse (if it exists) of a $n \times n$ matrix $\mathbf{A}$, denoted $\mathbf{A}^{-1}$, is the matrix that makes the following equation true: $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$

Theorem
If $\mathbf{A}$ *is an invertible matrix, then*

$$\mathbf{A}^{-1} = \frac{adj(\mathbf{A})}{det(\mathbf{A})}$$

Linearization and Case Studies

- Please read Section 3.7 (Linearization) on your own.