جامعة جدة
University of Jeddah

Faculty of Engineering
Department of Electrical & Computer Engineering (ECE)

Control Systems (ECE 331)

Experiment No: 03

"Introduction to Modeling in State Space Domain"

Submitted By:

University ID No:

Date of Submission:

Signature of Faculty: Marks:

Name of Faculty: Ankit Patel

Control Systems   ECE 331

Experiment 03
Introduction to Modeling in State Space Domain

Contents

1 Introduction to State Space

The state space or modern or time domain approach is a unified method for modeling, analyzing and designing a wide range of systems. For example: This approach can be used to represent nonlinear systems have backlash, saturation, and dead zone. Also, it can handle easily systems with nonzero initial conditions. The time domain approach can be used to represent systems with a digital computer in the loop or to model systems for digital simulation. With simulated systems, system response can be obtained for changes in system parameters - an important design tool. This approach is also attractive because of the availability of numerous state-space approach packages for the personal computer.

The time domain approach can also be used for the same class of systems modeled by the classical approach. This alternate model gives the control systems designer another perspective from which to create a design. While the state space approach can be applied to a wide range of systems, it is not as intuitive as the classical approach.

1.1 The General State Space Representation

- **Linear Combinations:** A linear combination of n variables, x_i, for $i = 1\ to\ n$, is given by the following sum, $S = K_n x_n + K_{n-1} x_{n-1} + ... + K_1 x_1$; Where K_i is a constant.

- **Linear Independent:** A set of variables is said to be linearly independent if none of the variables can be written as a linear combination if the others. `For example:` given $x_1, x_2 and x_3$, if $x_2 = 5x_1 + 6x_3$, then the variables are not linearly independent, since one of them can be written as a linear combination of the other two. Now, what must be true so that one variable can not be written as a linear combination of the other variables? Consider the example:: $k_2 x_2 = K_1 x_1 + K_3 x3$. If no $x_i = 0$, then any x_i can be written as a linear combinations of other variables, unless all $K_i = 0$. Formally, then, variables x_i, for $i = 1\ to\ n$, are said to be linearly independent if their linear combination, S, equals zero only if every $K_i = 0$ and no $x_i = 0$.

- **System Variable:** Any variable that responds to an input or initial conditions in a systems.

- **State Variables:** The smallest set of linearly independent system variables such that the values of the members of the set at time t_0 along with known forcing functions completely determine the value of all system variables for all $t \geq t_0$.

- **State Vector:** A vector whose elements are the state variables.

- **State Space:** The n dimensional space whose axes are the state variables.

  Figure 1 shows the state variables are assumed to be a resistor voltage v_R and a capacitor voltage v_C. These variables form the axes of the state space. A trajectory can be thought of as being mapped out by the state vector, $x(t)$, for a range of t. Also shown is the state vector at the particular time $t = 4$.

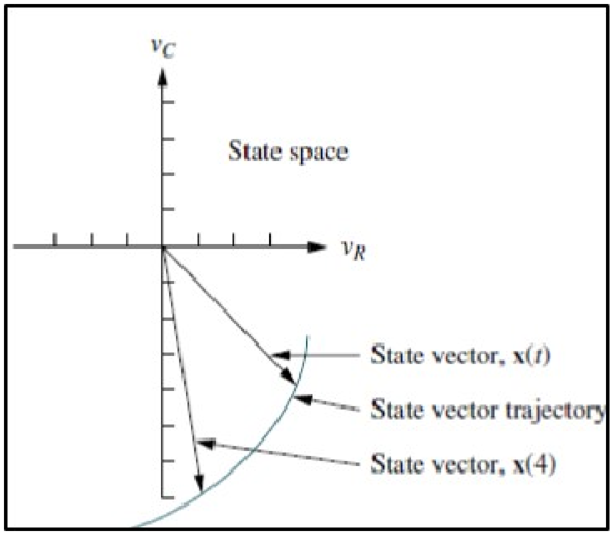

Figure 1: State Space Representation

- **State Equations:** A set of n simultaneous, first order differential equations with n variables, where the n variables to be solved are the state variables.
- **Output Equation:** The algebraic equation that expresses the output variables of a system as linear combinations of the state variables and the inputs.

The state equations are now defined as under:

$$\dot{x} = Ax + Bu \quad StateEquation$$
$$y = Cx + Du \quad OutputEquation$$

Where,

x = State vector, $\dot{x}$ = Derivation of the state vector with respect to time, y = Output vector, u = Input or Control vector, A = System matrix, B = Input matrix, C = Output matrix, D = Feedforward matrix

1.2 Some Observations of State Space Approach:

1. We select a particular subset of all possible system variables and call the variables in this subset state variables.
2. For the n^{th} order system, we write n simultaneous, first order differential equations in terms of the state variables. We call this system of simultaneous differential equations state equations.
3. If we know the initial conditions of all of the state variables at t_0 as well as the system input for $t \geq t_0$, we can solve the simultaneous differential equations for the state variables for $t \geq t_0$.

4. We algebraically combine the state variables with the systems input and find all the other system variables for $t \geq t_0$. We call this algebraic equation the output equation.
5. We consider the state equations and output equations a viable representation of the system. We call this representation of the system a state-space representation.

2 Modeling in State Space Domain

Command Function	**Converting From**	**Converting To**	**Input Arguments**	**Output Arguments**
`ss2tf`	state space	transfer function	`[A,B,C,D]`	`[num,den]`
`ss2zp`		zero pole gain		`[z,p,k]`
`tf2ss`	transfer function	state space	`[num.den]`	`[A,B,C,D]`
`tf2zp`		zero pole gain		`[z,p,k]`
`zp2ss`	zero pole gain	state space	`[z,p,k]`	`[A,B,C,D]`
`zp2tf`		transfer function		`[num,den]`

Figure 2: MATLAB Command for State Space Representation

`Example:01     Obtain the state equation in phase variable form for the following differential equation:`

$$2\frac{d^3y}{dt^3} + 4\frac{d^2y}{dt^2} + 6\frac{dy}{dt} + 8 = 10u(t)$$

`Solution:-`

The differential equation is third order, and thus there are three state variables.

$x_1 = y, x_2 = \dot{y}, x_3 = \ddot{y}.$

The first derivatives are:

$$\dot{x_1} = x_2$$
$$\dot{x_2} = x_3$$
$$\dot{x_3} = -4x_1 - 3x_2 - 2x_3 + 5u(t)$$

or in matrix form:

$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -4 & -3 & -2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 5 \end{pmatrix} u(t)$$

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

The .m file was developed to convert an n^{th} order ordinary differential equation to the state space phase variable form. The syntax is $[ABC] = ode2phv(ai, k)$, and returns the typical three matrices. The input ai is a row vector containing the coefficients of the equation in descending order, and k is the coefficient on the right hand side.

```
>> ai = [2 4 6 8];
>> k = 10;
>> [A, B, C] = ode2phv(ai,k)
A =
    0   1   0
    0   0   1
    -4 -3 -2


B =
    0
    0
    5


C =
    1  0  0
```

Creating State Space Model.

`Example: 02  Given the set of linear differential equations.`

$$\frac{dh_1}{dt} = -2h_1 + 3h_2 + 2v$$
$$\frac{dh_2}{dt} = 4h_1 - h_2 - v$$
$$q = h_1 - 2h_2 + \frac{1}{2}v$$

where the states are h_1 and h_2, the input is v and the output is q.

`Solution:`

This can be rewritten as in the form of matrix,

$$\frac{d}{dt}x = Ax + Bu$$
$$y = Cx + Du$$
$$\text{where, } x = \begin{pmatrix} h_1 \\ h_2 \end{pmatrix};$$
$$u = (v);\ y = (q)$$
$$A = \begin{pmatrix} -2 & 3 \\ 4 & -1 \end{pmatrix}; \qquad B = \begin{pmatrix} 2 \\ -1 \end{pmatrix};$$
$$C = \begin{pmatrix} 1 & -2 \end{pmatrix};$$
$$D = \begin{pmatrix} \frac{1}{2} \end{pmatrix}$$

The above matrices are written in `MATLAB` as:

```
>> A = [-2,3; 4,-1];
>> B = [2;-1];
>> C = [1,-2];
>> D = [1/2];
```

To create a state space model or object, use the `ss` command,

```
>> ht_model = ss(A,B,C,D)
```