Modelling Systems in State Space

Dr Chris Jobling (c.p.jobling@swansea.ac.uk)

Digital Technium 123

Office Hours: 12:00-13:00 Mondays

You can view the notes for this presentation in HTML and PDF.

The source code of this presentation is available in Markdown format from GitHub: Introduction.md.

The GitHub repository EG-247 Resources also contains the source code for all the Matlab/Simulink examples and the Laboratory Exercises.

Scope and Background Reading

This session is an introduction to state variables and state equations as they apply to circuit analysis. The state transition matrix is defined and the state-space to transfer function equivalence is presented. Several examples are given to illustrate their application.

Matlab has particularly strong support for both numeric and symbolic computation of models based on the state-space model and transfer functions so we will not present the detailed mathematical proofs nor bother with the hand-solution of state-space problems. (Nor will we expect you to!)

You will however gain some insight into the power of the state-space model for modelling and simulation in the lab to follow.

If you are interested in the hand calculations, Chapter 5 of Steven T. Karris, Signals and Systems: with Matlab Computation and Simulink Modelling, 5th Edition. gives full details.

Agenda

- The Idea of a State Space Model
- Solving the First-Order Case
- Outline of the General Solution
- Circuit Analysis with State-Space Models
- Transfer Function from State-Space Model

Lab

- Using the Computer for Analysis
- Modelling and Simulation with Simulink
- The LTI Model in Matlab

The Idea of a State-Space Model

The Idea of a State-Space Model

- In circuits when doing analysis with KCL and KVL we obtain integro-differential equations as a result of the energy-storing devices (inductors and capacitors)
- When a circuit contains one such device, it is said to be a *first-order circuit*
- When a circuit contains two such devices, it is said to be a *second-order circuit*, and so on.

First-order linear time invariant system

A first-order linear, time-invariant (LTI) system can be represented by a differential equation of the form

$$a_1 \frac{d}{dt} y(t) + a_0 y(t) = u(t)$$

Second-order LTI system

A second-order LTI system can be represented by a differential equation of the form

$$a_2 \frac{d^2}{dt^2} y(t) + a_1 \frac{d}{dt} y(t) + a_0 y(t) = u(t)$$

nth-order LTI system

By extension, an nth-order LTI system can be represented by a differential equation of the form

$$a_n \frac{d^n}{dt^n} y(t) + a_{n-1} \frac{d^{n-1}}{dt^{n-1}} y(t) + \cdots + a_1 \frac{d}{dt} y(t) + a_0 y(t) = u(t)$$

Example 1

A series RLC circuit with excitation

$$v_s(t) = e^{j\omega t}$$

is described the the integro-differential equation

$$Ri(t) + L\frac{d}{dt}i(t) + \frac{1}{C}\int_{-\infty}^{t} i(t)dt = e^{j\omega t}$$

Convert the equation into a second-order differential form and from this derive a state-space model.

Solution 1 - Step 1

Differentiating both sides of the equation and dividing by L we get

$$\frac{d^2}{dt^2}i(t) + \frac{R}{L}\frac{d}{dt}i(t) + \frac{1}{LC}i(t) = \frac{1}{L}j\omega e^{j\omega t}$$

or

$$\frac{d^2}{dt^2}i(t) = -\frac{R}{L}\frac{d}{dt}i(t) - \frac{1}{LC}i(t) + \frac{1}{L}j\omega e^{j\omega t}$$

Step 2: Define state variables

Let us define x_1 and x_2 to be two *state variables* such that

$$x_1 = i$$

and

$$x_2 = \frac{di}{dt} = \frac{dx_1}{dt} = \dot{x}_1$$

then

$$\dot{x}_2 = \frac{dx_1}{dt} = \frac{d^2i}{dt^2}$$

Step 3: Recast differential equation in terms of the state variables

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -\frac{R}{L}x_2 - \frac{1}{LC}x_1 + \frac{1}{L}j\omega e^{j\omega t}$$

Step 4: Put into matrix-vector form

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{LC} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} j\omega e^{j\omega t}$$

Step 5: State-space form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$

where

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} \quad \mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{LC} & -\frac{R}{L} \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} \text{ and } u = j\omega e^{j\omega t}.$$

The output $y(t)$ is expressed by the state-equation

$$y = \mathbf{C}\mathbf{x} + du$$

where $\mathbf{C}$ is another matrix, and d is a scalar.

General State-Space Form

For a single-input single-output (SISO) system:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$
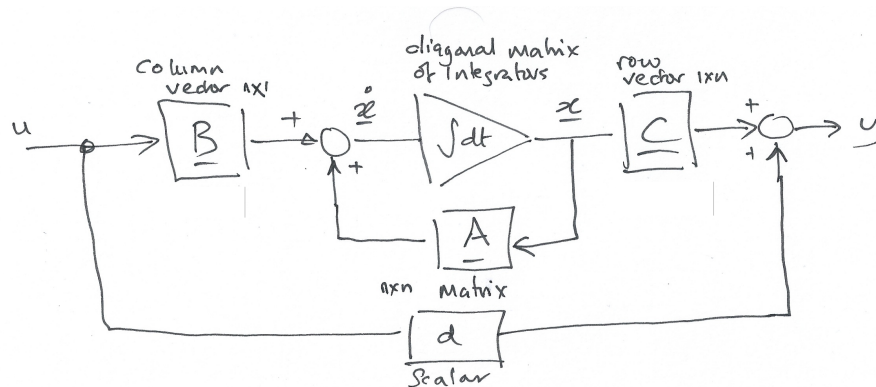$$\mathbf{y} = \mathbf{C}\mathbf{x} + du$$

Figure 1: General state-space form

Example 2

A fourth-order network is decsribed by the differential equation

$$\frac{d^4y}{dt^4} + a_3\frac{d^3y}{dt^3} + a_2\frac{d^2y}{dt^2} + a_1\frac{dy}{dt} + a_0y(t) = u(t)$$

where $y(t)$ represents the circuit output as a current or voltage, and $u(t)$ is an arbitrary input.

Express this system in state-space form.

Example 3

Write the state equations for the circuit below, given that $V_c(0^-) = 0$, and $u_0(t)$ is the unit step.
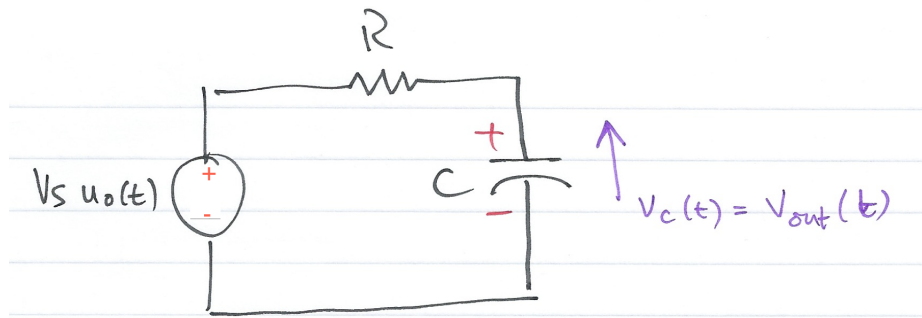

Figure 2: Example 3

Example 4

See Example 5.4 on Pages 5-5—5-6 in the textbook.

Solving the First-Order Case

Solving the First-Order Case

If a circuit contains only one energy-storing device, the state equations are written as:

$$\dot{x} = ax + bu$$
$$y = cx + du$$

where a, b, c and d are scalar constants.

The solution will be of the form

$$x(t) = e^{a(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{a(t-\tau)}bu(\tau)d\tau$$

(For proof see text-book)

Example 5

Use the previous result to find the capacitor voltage $v_c(t)$ of the circuit below for $t > 0$, given that the initial condition is $v_c(0^-) = 1$ V, $R = 2$ Ω, $C = 0.5$ F and $v_s(t) = 2u_0(t)$.
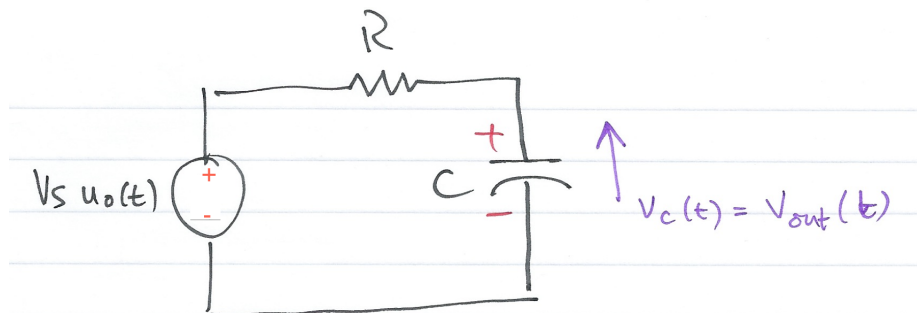

Figure 3: Example 5

(*Solution:* $y(t) = x(t) = v_c(t) = (2 - e^{-t})u_0(t)$)

Matlab Solution

```
t = linspace(0,10,1000);
vc = 2-exp(-t);
plot(t,vc),axis([0,10,0,3]),grid
```
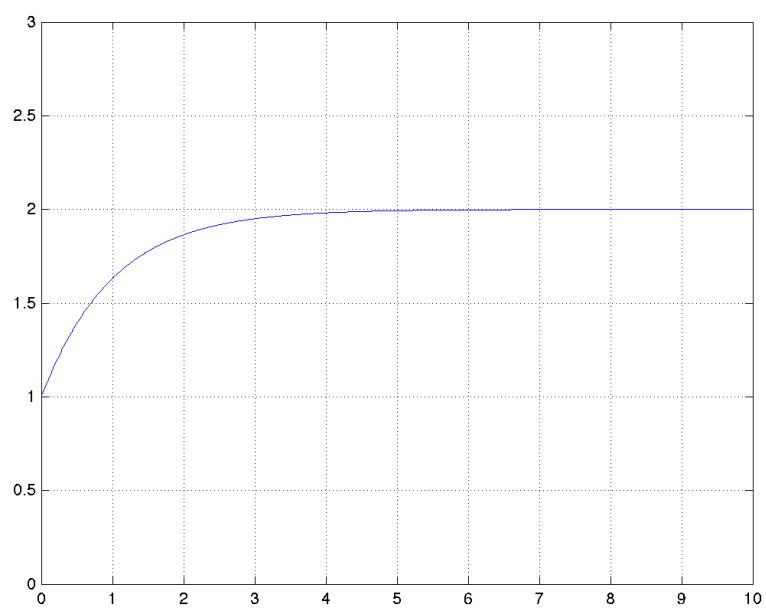
Figure 4: Matlab solution

Outline of the General Solution

Outline of the General Solution

If

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + du$$

the the solution of

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$

with initial condtions $\mathbf{x}(t_0) = \mathbf{x_0}$

is the matrix generalization of the previous result:

$$\mathbf{x} = e^{\mathbf{A}t}\mathbf{x}_0 + \int_{t_0}^{t} e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau$$

Transition Matrix

The term

$$\phi(t) = e^{\mathbf{A}t}$$

is called the *state transition matrix* and it can be computed by hand (at least for the 2x2 case) if you have knowledge of the *eigenvalues* and *eigenvectors* of the state-matrix $\mathbf{A}$.

If you want to study the proof, see Section 5.4 **Computation of the State Transition Matrix** in the text book.

Computation of the State Transition Matrix in Matlab

Matlab provides the function `expm` which computes the matrix exponential

Example 6

Compute the state transition matrix $\phi = \mathbf{e}^{\mathbf{At}}$ for the state matrix

$$\mathbf{A} = \left[\begin{array}{cc} -2 & 1 \\ 0 & -1 \end{array} \right]$$

```
syms t;
A = [-2, 1; 0, -1];
phi = expm(A*t)

phi =

  [ exp(-2*t), exp(-t) - exp(-2*t)]
  [        0,             exp(-t)]
```

$$\phi(t) = \left[\begin{array}{cc} e^{-2t} & e^{-t} - e^{-2t} \\ 0 & e^{-t} \end{array} \right]$$

Circuit Analysis with State-Space Models

Example 7

For the RLC circuit shown below, the initial conditions ar $i_L(0^-)$=0, and $v_c(0^-) = 0.5$ V. Use the state variable method to compute $i_L(t)$ and $v_c(t)$.
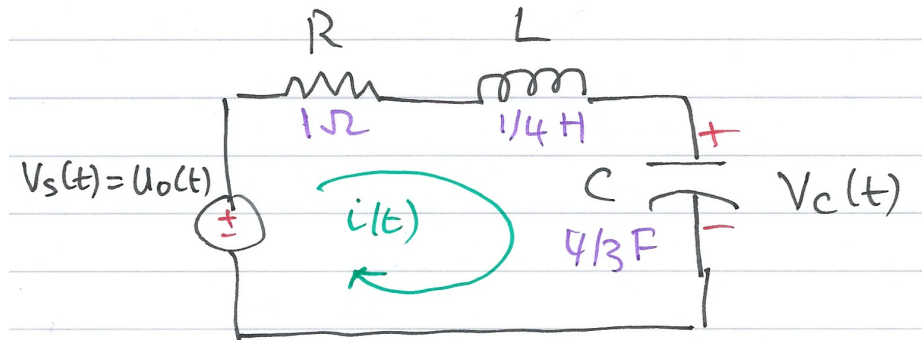

Figure 5: Example 7

Sketch of solution

See Example 5.10 for full solution.

We chose $x_1 = i_L$ and $x_2 = v_c$ as our state variables which yields state equations

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -4 & -4 \\ 3/4 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_0(t)$$

State-transition matrix

```
syms t;
A = [-4, -4; 3/4, 0];
phi = expm(A*t);
pretty(phi)


  +-                                                    -+
  |    3 exp(-3 t)    exp(-t)                             |
  |    -----------  - -------,   2 exp(-3 t) - 2 exp(-t)  |
  |         2            2                                |
  |                                                      |
  |   3 exp(-t)    3 exp(-3 t)    3 exp(-t)    exp(-3 t)  |
  |   ---------  - -----------,   ---------  - ---------  |
  |       8            8              2            2      |
  +-                                                    -+
```

Giving

The state transition matrix is

$$\phi(t) = \begin{bmatrix} -\frac{1}{2}e^{-t} + \frac{3}{2}e^{-3t} & -2e^{-t} + 2e^{-3t} \\ -\frac{3}{8}e^{-t} + \frac{8}{8}e^{-3t} & -\frac{3}{2}e^{-t} - \frac{1}{2}e^{-3t} \end{bmatrix}$$

Unforced response

The *unforced response* is the response of the state variables to the initial conditions only.

We multiply this by the initial condition matrix, to get the *unforced response*

$$\mathbf{x}_u(t) = e^{\mathbf{A}t} \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix}$$

```
xu = phi*[0; 1/2];
pretty(xu)

  +-                      -+
  |   exp(-3 t)  - exp(-t)  |
  |                         |
  |  3 exp(-t)     exp(-3 t)  |
  |  ---------  -  ---------  |
  |      4             4      |
  +-                      -+
```

$$\mathbf{x}_u(t) = \begin{bmatrix} -e^{-t} + e^{-3t} \\ \frac{3}{4}e^{-t} - \frac{3}{4}e^{-3t} \end{bmatrix}$$

Forced response

The *forced response* (the response to the input only) is trickier to compute by hand

$$\mathbf{x}_f(t) = \int_0^t e^{\mathbf{A(t-\tau)}}\mathbf{B}u_0(\tau)d\tau$$

---

but is easy with Matlab

```
syms t tau
B = [4; 0];
xf = int(expm(A*(t-tau))*B*heaviside(tau),tau,0,t)

xf =

     2*exp(-t)  -  2*exp(-3*t)
  exp(-3*t)/2  -  (3*exp(-t))/2 + 1
```

$$\mathbf{x}_f(t) = \begin{bmatrix} 2e^{-t} - 2e^{-3t} \\ 1 - \frac{3}{2}e^{-t} + \frac{1}{2}e^{-3t} \end{bmatrix}$$

Overall response

The overall response is

$$\mathbf{x}(t) = \mathbf{x}_u(t) + \mathbf{x}_f(t)$$

```
x = xu + xf

  x =

              exp(-t) - exp(-3*t)
   exp(-3*t)/4 - (3*exp(-t))/4 + 1
```

$$\mathbf{x}(t) = \left[\begin{array}{c} e^{-t} - e^{-3t} \\ 1 - \frac{3}{4}e^{-t} + \frac{1}{4}e^{-3t} \end{array} \right]$$

Complete solution

So

$$x_1 = i_L = e^{-t} + e^{-3t}$$

and

$$x_2 = v_c = 1 - 0.75e^{-t} + 0.25e^{-3t}$$

Which agrees with the text book.

Quick MATLAB solution

As a four-liner

```
syms t tau;
A = [-4, -4; 3/4, 0]; B = [4; 0]; x0 = [0; 1/2];
x = expm(A*t)*x0 + ..
  int(expm(A*(t-tau))*B*heaviside(tau),tau,0,t);
il = x(1), vc = x(2)

il =

  exp(-t) - exp(-3*t)
```

```
vc =

  exp(-3*t)/4 - (3*exp(-t))/4 + 1
```

Transfer Function from State-Space Model

Transfer Function from State-Space Model

We state, without proof, that the laplace transform of the state equations are:

$$Y(s) = \left[\mathbf{C}(\mathbf{sI} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{d} \right] U(s)$$

where $\mathbf{I}$ is the *identity matrix* (Matlab `eye(n)`)

So the transfer fuction of a state-space model with input $U(s)$ and output $Y(s)$ is

$$G(s) = \frac{Y(s)}{U(s)} = \mathbf{C}(\mathbf{sI} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{d}$$

Example 7

Give the transfer function for Example 6 assuming that the output $y(t) = v_c(t)$ and find the step response of the circuit.

Solution

if $y(t) = v_c(t)$

$$\mathbf{C} = [\ 0 \quad 1\ ]$$

and

$$d = 0$$

```
syms t s
C = [0, 1]; d = 0;
G = C*inv(s*eye(2)-A)*B + d
```

```
G =

    3/(s^2 + 4*s + 3)
```

```
U = laplace(heaviside(t));
vc = ilaplace(G*U)

  vc =

    exp(-3*t)/2 - (3*exp(-t))/2 + 1
```

Another way of Looking at the State Transition Matrix

It can be shown that

$$\mathcal{L}e^{\mathbf{At}} = (s\mathbf{I} - \mathbf{A})^{-1}$$

and this gives us another way to find $e^{\mathbf{At}}$.

Example 8

Use Laplace transforms to find the state transition matrix for Example 6.

Solution

```
syms s t
A = [-4, -4; 3/4, 0];
phi_s = inv(s*eye(2)-A)
phi_t = ilaplace(phi_s)

phi_s =

    [     s/(s^2 + 4*s + 3),       -4/(s^2 + 4*s + 3)]
    [ 3/(4*(s^2 + 4*s + 3)), (s + 4)/(s^2 + 4*s + 3)]


phi_t =

    [     (3*exp(-3*t))/2 - exp(-t)/2,     2*exp(-3*t) - 2*exp(-t)]
    [ (3*exp(-t))/8 - (3*exp(-3*t))/8, (3*exp(-t))/2 - exp(-3*t)/2]
```

Homework

Try the end of the chapter exercises (Section 5.9) from the textbook. You can omit Question 6 and may solve Questions 7, 8, and 9 using Matlab. Don't look at the answers until you have attempted the problems.

Lab Work

In the lab, a week on Friday, we will give further examples of computing the state- transition matrix, and its use in the solution of systems defined by state-space models and transfer functions.

We will also introduce the *Linear Time Invariant* (LTI) model which can be used to model and analyse state-space and transfer functions.

We will also demonstrate the use of Simulink to model and simulate circuits

Next Session

- Time Convolution (Part 1) (Slides PDF)