Transfer
Function and State Variables

THE TRANSFER FUNCTION FROM THE STATE EQUATION

The transfer function of a single input-single output (SISO) system can be obtained from the state variable equations.

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

where y is the single output and u is the single input. The Laplace transform of the equations

$$sX(s) = AX(s) + BU(s)$$
$$Y(s) = CX(s)$$

where B is an nx1 matrix, since u is a single input. We do not include initial conditions, since we seek the transfer function. Reordering the equation

$$[sI - A]X(s) = BU(s)$$

$$X(s) = [sI - A]^{-1}BU(s) = \phi(s)BU(s)$$

$$Y(s) = C\phi(s)BU(s)$$

Therefore, the transfer function G(s)=Y(s)/U(s) is

$$G(s) = C\phi(s)B$$

Example:

Determine the transfer function G(s)=Y(s)/U(s) for the RLC circuit as described by the state differential function

$$\dot{x} = \begin{bmatrix} 0 & -\frac{1}{C} \\ \frac{1}{L} & -\frac{R}{L} \end{bmatrix} x + \begin{bmatrix} \frac{1}{C} \\ 0 \end{bmatrix} u \quad , \quad y = \begin{bmatrix} 0 & R \end{bmatrix} x$$

$$[sI - A] = \begin{bmatrix} s & \frac{1}{C} \\ -\frac{1}{L} & s + \frac{R}{L} \end{bmatrix}$$

$$\phi(s) = [sI - A]^{-1} = \frac{1}{\Delta(s)} \begin{bmatrix} s + \frac{R}{L} & -\frac{1}{C} \\ \frac{1}{L} & s \end{bmatrix}$$

$$\Delta(s) = s^2 + \frac{R}{L}s + \frac{1}{LC}$$

Then the transfer function is

$$G(s) = [0 \quad R] \begin{bmatrix} \frac{s + \frac{R}{L}}{\Delta(s)} & -\frac{1}{C\Delta(s)} \\ \frac{1}{L\Delta(s)} & \frac{s}{\Delta(s)} \end{bmatrix} \begin{bmatrix} \frac{1}{C} \\ 0 \end{bmatrix}$$

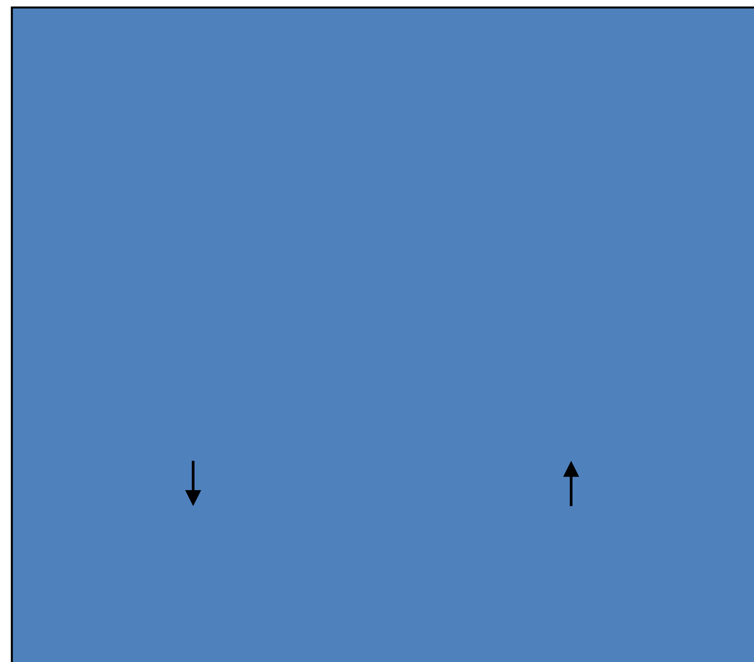
$$G(s) = \frac{R / LC}{\Delta(s)} = \frac{R / LC}{s^2 + \frac{R}{L}s + \frac{1}{LC}}$$

Dorf and Bishop, Modern Control Systems

ANALYSIS OF STATE VARIABLE MODELS USING MATLAB

Given a transfer function, we can obtain an equivalent state-space representation and vice versa. The function **tf** can be used to convert a state-space representation to a transfer function representation; the function **ss** can be used to convert a transfer function representation to a state-space representation. The functions are shown in Figure 4, where sys_tf represents a transfer function model and sys_ss is a state space representation.

The **ss** function



Linear system model conversion

Figure 4.

Dorf and Bishop, Modern Control Systems

For instance, consider the third-order system

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

We can obtain a state-space representation using the **ss** function. The state-space representation of the system given by G(s) is

Matlab code

```
num=[2 8 6];den=[1 8 16 6];

sys_tf=tf(num,den)

sys_ss=ss(sys_tf)
```

Answer

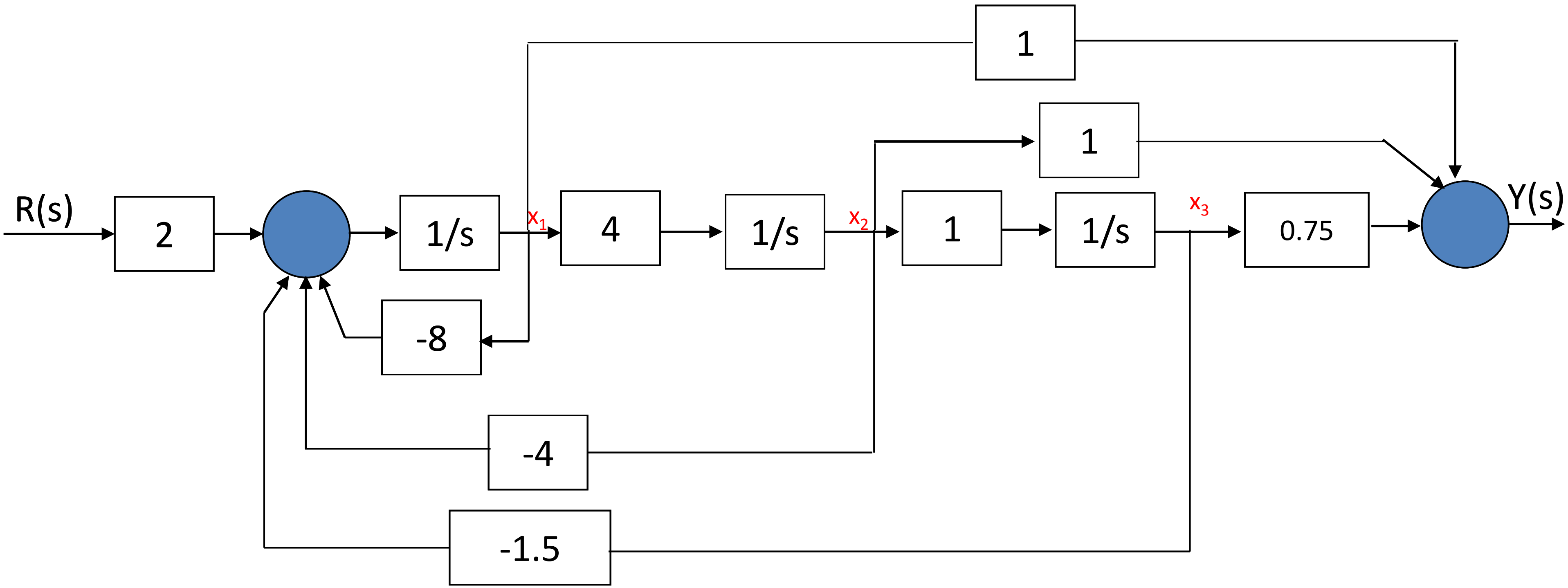
```
Transfer function:
  2 s^2 + 8 s + 6
----------------------
s^3 + 8 s^2 + 16 s + 6


a =
       x1    x2    x3
  x1   -8    -4  -1.5
  x2    4     0     0
  x3    0     1     0

b =
     u1
  x1  2
  x2  0
  x3  0

c =
       x1    x2    x3
  y1    1     1  0.75

d =
     u1
  y1  0

Continuous-time model.
```

$$A = \begin{bmatrix} -8 & -4 & -1.5 \\ 4 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 1 & 0.75 \end{bmatrix} \text{and } D = \begin{bmatrix} 0 \end{bmatrix}$$

$$A = \begin{bmatrix} -8 & -4 & -1.5 \\ 4 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 1 & 0.75 \end{bmatrix} \text{and } D = \begin{bmatrix} 0 \end{bmatrix}$$



Block diagram with x_1 defined as the leftmost state variable.

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)} B u(\tau)\, d\tau$$

$$x(t) = \phi(t)\, x(0) + \int_0^t \phi(t-\tau)\, B u(\tau)\, d\tau$$

We can use the function **expm** to compute the transition matrix for a given time. The **expm(A)** function computes the matrix exponential. By contrast the exp(A) function calculates $e^{a_{ij}}$ for each of the elements $a_{ij} \epsilon A$.

For the RLC network, the state-space representation is given as:

$$A = \begin{bmatrix} 0 & -2 \\ 1 & -3 \end{bmatrix}, B = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix} \text{and } D = \begin{bmatrix} 0 \end{bmatrix}$$

The initial conditions are $x_1(0)=x_2(0)=1$ and the input u(t)=0. At t=0.2, the state transition matrix is calculated as

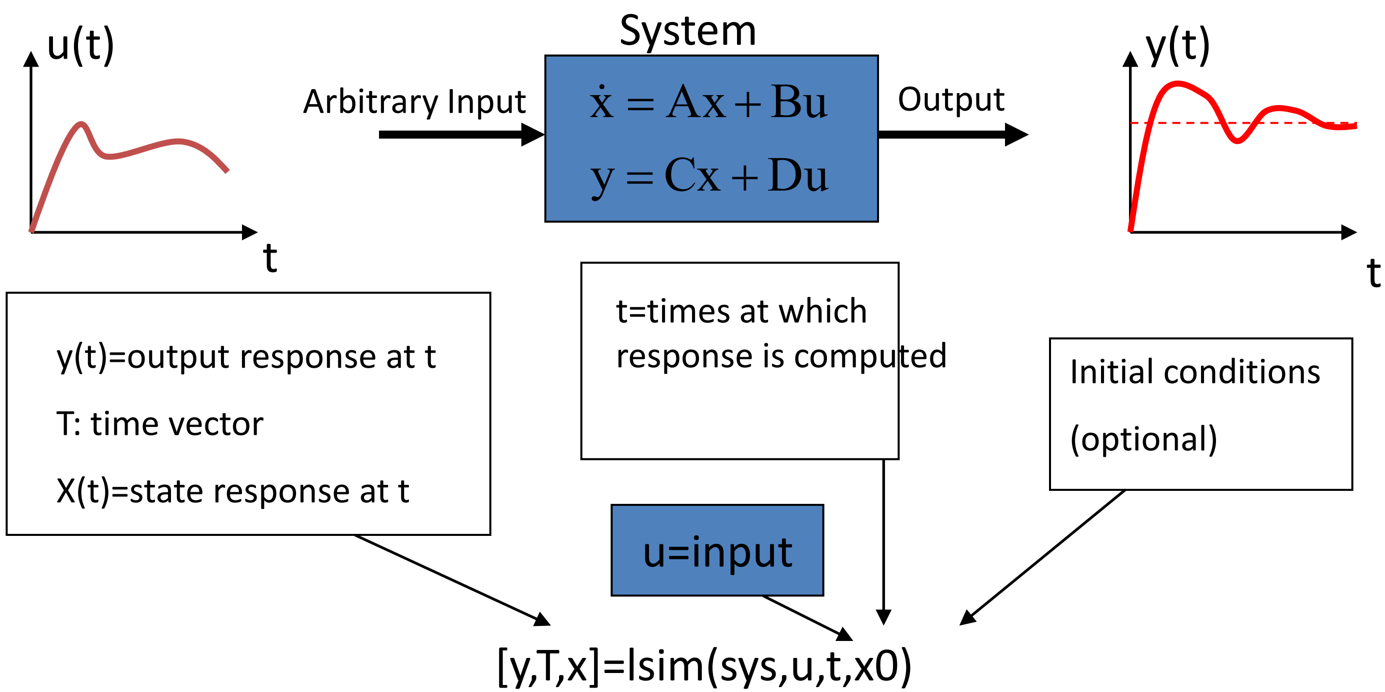
```
>>A=[0 -2;1 -3], dt=0.2; Phi=expm(A*dt)
```

```
Phi =

    0.9671   -0.2968
    0.1484    0.5219
```

The state at t=0.2 is predicted by the state transition method to be

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t=0.2} = \begin{bmatrix} 0.9671 & -0.2968 \\ 0.1484 & 0.5219 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t=0} = \begin{bmatrix} 0.6703 \\ 0.6703 \end{bmatrix}$$

The time response of a system can also be obtained by using **lsim** function. The **lsim** function can accept as input nonzero initial conditions as well as an input function. Using **lsim** function, we can calculate the response for the RLC network as shown below.



Dorf and Bishop, Modern Control Systems