

Lesson 7

BME 444 - Biomedical Controls

Contents

1	Learning Objectives	1
2	Where We Are	2
3	Converting from SSR to TF	2
3.1	Linear Algebra	2
3.2	Example 1	3
3.3	Student Example 1	4
3.4	MATLAB Function	4
3.5	Example 2	4
3.6	Student Example 2	5
4	Converting from TF to SSR	6
4.1	Example 3	6
4.2	Student Example 3	7
4.3	Example 4	8
4.4	Student Example 4	8
5	Converting the TF to a DE and Vice Versa	9
5.1	Example 5	9
5.2	Student Example 5	9

1 Learning Objectives

By the end of this lesson students will be able to:

- Convert a transfer function to state space and vice versa
- Convert a transfer function to a DE and vice versa

→ In Matlab, go to Add-Ons, search for Control Systems Toolbox, and install it.

2 Where We Are

The state-space representation (SSR) is powerful because it can model a wider range of systems than the classical approach that uses transfer functions. However, transfer functions provide great insight into system behavior, so it would be useful if we could convert from SSR to transfer functions and vice versa.

3 Converting from SSR to TF

There are two ways:

1. Linear algebra
2. MATLAB function `ss2tf()`

3.1 Linear Algebra

Remember that the SSR looks like this:

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}$$

$$\mathbf{y} = \mathbf{Cx} + \mathbf{Du}$$

If we take the LaPlace transform of these two equations and combine them, we get the following formula:

$$Y(s) = (C(sI - A)^{-1}B + D)U(s)$$

In this equation, I is the identity matrix. It can vary in size and has 1's in the diagonal. A 3×3 identity matrix looks like:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

If we re-arrange the above equation a little, we get:

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

Which is a transfer function. So, to convert SSR to a transfer function, use the above formula.

3.2 Example 1

The SSR for Example 1 from last class is written below. Convert it to a transfer function using the linear algebra method.

$$\dot{x} = \begin{bmatrix} \dot{v}_{C1} \\ \dot{v}_{C2} \\ \dot{i}_L \end{bmatrix} = \begin{bmatrix} \frac{-1}{RC_1} & \frac{-1}{RC_1} & \frac{1}{C_1} \\ \frac{-1}{RC_2} & \frac{-1}{RC_2} & 0 \\ \frac{-1}{L} & 0 & 0 \end{bmatrix} \begin{bmatrix} v_{C1} \\ v_{C2} \\ i_L \end{bmatrix} + \begin{bmatrix} \frac{1}{RC_1} \\ \frac{1}{RC_2} \\ \frac{1}{L} \end{bmatrix} v(t)$$

```
syms s r l c1 c2
A=[-1/(r*c1) -1/(r*c1) 1/c1; -1/(r*c2) -1/(r*c2) 0; -1/l 0 0]
B=[1/(r*c1); 1/(r*c2); 1/l]
C=[0 1 0]
I=eye(3);
D=0;
ans = C*inv(s*I-A)*B+D;
simplify(ans)
```

3.3 Student Example 1

The SSR for Student Example 1 from last class is written below. Convert it to a transfer function using the linear algebra method.

$$\dot{x} = \begin{bmatrix} \dot{v}_C \\ \dot{i}_L \end{bmatrix} = \begin{bmatrix} \frac{-1}{RC} & \frac{1}{C} \\ \frac{1}{L} & 0 \end{bmatrix} \begin{bmatrix} v_C \\ i_L \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} v(t)$$

Solution

3.4 MATLAB Function

The Control Systems toolbox in MATLAB contains the function `ss2tf()` that will perform the conversion.

- The `ss2tf()` function does not work with variables. You must assign numbers to all R , C , L , or mass, f_v , and k values.

3.5 Example 2

Repeat Example 1 using `ss2tf()`.

```
c1=1; c2=1; r=1; l=1;
A=[-1/(r*c1) -1/(r*c1) 1/c1; -1/(r*c2) -1/(r*c2) 0; -1/l 0 0]
B=[1/(r*c1); 1/(r*c2); 1/l]
C=[0 1 0]
D=0;
[num,den]=ss2tf(A,B,C,D)
vpa(num,6)
vpa(den,6)
```

Here's the output I get:

```
num = 0      1.0000      0.0000      -0.0000
den = 1.0000      2.0000      1.0000      1.0000
```

Is this true? Use `vpa()` to check:

```
ans = [0, 1.0, 6.82455e-16, 9.86076e-32]
ans = [1.0, 2.0, 1.0, 1.0]
```

Looks good. But how do we translate those numbers into a transfer function like we're used to seeing?

3.6 Student Example 2

Repeat Student Example 1 using `ss2tf()`. Assume $R = 1\ \Omega$, $L = 1\ \text{H}$ and $C = 1\ \mu\text{F}$.

Solution

4 Converting from TF to SSR

What if we're given a transfer function and want to calculate the state-space representation? No problem, but remember:

→ The state-space representation (SSR) you calculate from a transfer function will not be the same SSR you get if you start with the physical system and derive it yourself.

Does this matter? Not as long as you keep in mind that both transfer functions are still “correct”; they will give the same outputs for a given input. There is another important thing to remember:

→ You will not know what the state variables represent when you go from TF to SS—they will still be valid state variables, but they will be abstract.

As before, there are two ways to convert TF to SSR:

1. The canonical method
2. MATLAB function `tf2ss()`

4.1 Example 3

Given a transfer function for a system shown below, convert to SSR using the canonical method.

$$G(s) = \frac{s^2 + 7s + 2}{s^3 + 9s^2 + 26s + 24}$$

4.2 Student Example 3

Given a transfer function for a system shown below, convert to SSR using the canonical method.

$$G(s) = \frac{2s + 1}{s^2 + 7s + 9}$$

4.3 Example 4

Given the transfer function below, use `tf2ss()` in MATLAB to convert to SSR.

$$G(s) = \frac{s^2 + 7s + 2}{s^3 + 9s^2 + 26s + 24}$$

```
num=[1 7 2];  
den=[1 9 26 24];  
[A,B,C,D]=tf2ss(num,den)
```

4.4 Student Example 4

Given the transfer function below, use `tf2ss()` in MATLAB to convert to SSR.

$$G(s) = \frac{2s + 1}{s^2 + 7s + 9}$$

Solution

5 Converting the TF to a DE and Vice Versa

The transfer function is just a LaPlace transform of a differential equation written in a non-standard way.

5.1 Example 5

Given the transfer function below, convert it to a differential equation.

$$G(s) = \frac{7s + 4}{2s^2 + 10s + 28}$$

5.2 Student Example 5

Convert the differential equation below to a transfer function.

$$\frac{d^2y}{dt^2} + 8\frac{dy}{dt} + 10y = 4\frac{du}{dt}$$