1 h - Sessional
week 3

Control Systems - 7^{th} Semester - Week 3

State-space Modeling of Systems

Dr. Salman Ahmed

Model of Systems

Last week, we studied about transfer function models

Last week, we also studied how to obtain poles, zeros, and analyze stability of transfer function model

This week we will learn a new language of modelling which is called as state-space modelling

We will also study the conversion techniques from state-space models to transfer function models (and vise versa)

General template of *ss* model

A system is composed of variables, constants, inputs and outputs.

Among the variables present in a system, we choose some variables as state-space variable (based on certain criteria which we will study later on), and call them state-space variables.

Let x be a vector having all state-space variables and let $\dot{x}$ denote the derivative of state-space variables.

Let $u(t)$ denote the input to a system, and $y(t)$ denote the output of a system.

General template of *ss* model

The standard template for state-space model is as follows:

$$\dot{x} = Ax + Bu(t)$$
$$y = Cx + Du(t)$$

The state-space model is sometimes called as ss model also

Converting ss to tf

The general form or template of ss model is as follows:

$$\dot{x} = Ax + Bu(t)$$
$$y = Cx + Du(t)$$

Let $G(s)$ denote the transfer function after converting to tf domain. The formula is:

$$G(s) = D + C[(sI - A)^{-1}B]$$

Example of conversion from ss to tf

Convert the following state-space model to transfer function

$$\dot{x} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} x + \begin{bmatrix} 5 \\ 6 \end{bmatrix} u(t)$$

$$y = \begin{bmatrix} 1 & 2 \end{bmatrix} x$$

Let us first obtain $(sI - A)^{-1}$

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

$$sI = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix}$$

$$sI - A = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} s-1 & -2 \\ -3 & s-4 \end{bmatrix}$$

Example of conversion from ss to tf

$$sI - A = \begin{bmatrix} s-1 & -2 \\ -3 & s-4 \end{bmatrix}$$

Now let us find $(sI - A)^{-1}$

$$(sI - A)^{-1} = \frac{\text{adjoint}(sI - A)}{\det(sI - A)}$$

$$\text{adjoint}(sI - A) = \begin{bmatrix} s-4 & 2 \\ 3 & s-1 \end{bmatrix}$$

$$\begin{aligned} \det(sI - A) &= (s-1)(s-4) - (-2)(-3) \\ &= (s^2 - 5s + 4) - (6) \\ &= s^2 - 5s + 4 - 6 \\ &= s^2 - 5s - 2 \end{aligned}$$

$$(sI - A)^{-1} = \frac{\text{adjoint}(sI - A)}{\det(sI - A)} = \frac{1}{s^2 - 5s - 2} \begin{bmatrix} s-4 & 2 \\ 3 & s-1 \end{bmatrix}$$

Example of conversion from ss to tf

Next, we post-multiply with matrix B as follows:

$$(sI - A)^{-1} \times B = \frac{1}{s^2 - 5s - 2} \begin{bmatrix} s-4 & 2 \\ 3 & s-1 \end{bmatrix} \times \begin{bmatrix} 5 \\ 6 \end{bmatrix}$$

$$= \frac{1}{s^2 - 5s - 2} \begin{bmatrix} \left((s-4) \times 5\right) + \left(2 \times 6\right) \\ \left(3 \times 5\right) + \left((s-1) \times 6\right) \end{bmatrix}$$

$$= \frac{1}{s^2 - 5s - 2} \begin{bmatrix} 5s - 20 + 12 \\ 15 + 6s - 6 \end{bmatrix}$$

$$= \frac{1}{s^2 - 5s - 2} \begin{bmatrix} 5s - 8 \\ 6s + 9 \end{bmatrix}$$

2

Example of conversion from *ss* to *tf*

Now, let us pre-multiply with matrix C as follows:

$$C(sI-A)^{-1}B = \frac{1}{s^2-5s-2}\begin{bmatrix}1 & 2\end{bmatrix} \times \begin{bmatrix}5s-8\\6s+9\end{bmatrix}$$

$$= \frac{1}{s^2-5s-2}\left[1\times(5s-8)+2\times(6s+9)\right]$$

$$= \frac{1}{s^2-5s-2}\left[5s-8+12s+18\right]$$

$$= \frac{1}{s^2-5s-2}\left[17s+10\right]$$

$$= \frac{17s+10}{s^2-5s-2}$$

Mathematical Model State-space Model Convert state-space to transfer function Conversion from tf to ss MATLAB Code Next week topics Assignment

Example of conversion from *ss* to *tf*

MATLAB code for conversion of *ss* to *tf*

```
A=[1 2; 3 4];

B=[5; 6];

C=[1 2];

D=[0];

[num,den] = ss2tf(A,B,C,D);

g=tf(num,den)
```

Mathematical Model State-space Model Convert state-space to transfer function Conversion from tf to ss MATLAB Code Next week topics Assignment

Conversion from *tf* to *ss*

Converting from *tf* to state-space is not a unique process

There are various techniques to convert form transfer function domain to state-space domain

We call each technique as canonical form. Let us study the first canonical form which is topic 3.5 in book

Mathematical Model State-space Model Convert state-space to transfer function Conversion from tf to ss MATLAB Code Next week topics Assignment

Conversion from *tf* to *ss* - Canonical Form 1

For a 2^{nd} order transfer function:

$$G(s) = \frac{b_1 s^1 + b_0}{s^2 + a_1 s + a_0}$$

We write the following state-space model (using Canonical Form 1):

$$A = \begin{bmatrix}0 & 1\\-a_0 & -a_1\end{bmatrix} \qquad B = \begin{bmatrix}b_1\\b_0\end{bmatrix}$$

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

Conversion from *tf* to *ss* - Canonical Form 1

For a 3^{rd} order transfer function:

$$G(s) = \frac{b_2 s^2 + b_1 s^1 + b_0}{s^3 + a_2 s^2 + a_1 s + a_0}$$

We write the following state-space model (using Canonical Form 1):

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{bmatrix} \qquad B = \begin{bmatrix} b_2 \\ b_1 \\ b_0 \end{bmatrix}$$

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

Conversion from *tf* to *ss* - Canonical Form 1

For a 4^{th} order transfer function:

$$G(s) = \frac{b_3 s^3 + b_2 s^2 + b_1 s^1 + b_0}{s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}$$

We write the following state-space mode' (using Canonical Form 1):

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 & -a_3 \end{bmatrix} \qquad B = \begin{bmatrix} b_3 \\ b_2 \\ b_1 \\ b_0 \end{bmatrix}$$

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

Conversion from *tf* to *ss* - Canonical Form 1

For n^{th} order transfer function:

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

We write the following state-space model (using Canonical Form 1):

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 & -a_{n-2} & -a_{n-1} \end{bmatrix} \qquad B = \begin{bmatrix} b_{n-1} \\ b_{n-2} \\ \vdots \\ b_1 \\ b_0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 \end{bmatrix}$$

Conversion from *tf* to *ss* - Canonical Form 1

Example 3.4 Page 128: Convert the following transfer function to state-space domain

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

4

Conversion from tf to ss - Canonical Form 1

Example 3.4 Page 128: Convert the following transfer function to state-space domain

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

Solution: In this example, $a_0 = 24$, $a_1 = 26$, $a_2 = 9$ and $b_0 = 24$, we can obtain the following:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -24 & -26 & -9 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 24 \end{bmatrix}$$

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

Conversion from tf to ss - Canonical Form 1

Example 3.5 Page 128: Convert the following transfer function to state-space domain

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

Conversion from tf to ss - Canonical Form 1

Example 3.5 Page 128: Convert the following transfer function to state-space domain

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

Solution: Solution: In this example, $a_0 = 24$, $a_1 = 26$, $a_2 = 9$ and $b_0 = 2$, $b_1 = 7$ and $b_2 = 1$, we can obtain the following:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -24 & -26 & -9 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 7 \\ 2 \end{bmatrix}$$

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

Conversion from tf to ss - Canonical Form 2

There is another canonical form which is as follows:

$$G(s) = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \ldots b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + a_2 s^2 + \ldots + a_1 s + a_0}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 & -a_{n-2} & -a_{n-1} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} b_0 & b_1 & b_2 & \ldots & b_{n-1} \end{bmatrix}$$

What is the difference between this canonical form and the previous

Conversion from *tf* to *ss* - Canonical Form 2

There is another canonical form which is as follows:

$$G(s) = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + a_2 s^2 + \dots + a_1 s + a_0}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \vdots & \vdots \\ 0 & & & 0 & 1 \\ -a_0 & & \cdots & -a_{n-2} & -a_{n-1} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} b_0 & b_1 & b_2 & \dots & b_{n-1} \end{bmatrix}$$

What is the difference between this canonical form and the previous one?

Matrix B in canonical form 2 seems like transpose of matrix C in the (previous) canonical form 1 and vice versa.

Conversion from *tf* to *ss* - Canonical Form 2

Example 3.5 Page 128: Convert the following transfer function to state-space domain using canonical form 2

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

Conversion from *tf* to *ss* - Canonical Form 2

Example 3.5 Page 128: Convert the following transfer function to state-space domain using canonical form 2

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

Solution:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -24 & -26 & -9 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 2 & 7 & 1 \end{bmatrix}$$

Controller Canonical Form - Canonical Form 3

There is another canonical form called controller canonical form which is as follows:

$$G(s) = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + a_2 s^2 + \dots + a_1 s + a_0}$$

$$A = \begin{bmatrix} -a_{n-1} & -a_{n-2} & \cdots & -a_1 & -a_0 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} b_{n-1} & b_{n-2} & b_{n-3} & \dots & b_0 \end{bmatrix}$$

(6)

Conversion from tf to ss - Controller Canonical Form

Example 3.5 Page 128: Convert the following transfer function to state-space domain using controller canonical form

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

Conversion from tf to ss - Controller Canonical Form

Example 3.5 Page 128: Convert the following transfer function to state-space domain using controller canonical form

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

Solution:

$$A = \begin{bmatrix} -9 & -26 & -24 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 7 & 2 \end{bmatrix}$$

Controller Canonical Form - Canonical Form 4

Another canonical form is observer canonical form, which is as follows:

$$G(s) = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + ...b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + a_2 s^2 + \ldots + a_1 s + a_0}$$

$$A = \begin{bmatrix} -a_{n-1} & 1 & 0 & 0 & 0 \\ -a_{n-2} & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -a_0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} b_{n-1} \\ b_{n-2} \\ \vdots \\ b_1 \\ b_0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 \end{bmatrix}$$

Conversion from tf to ss - Observer Canonical Form

Example 3.5 Page 128: Convert the following transfer function t[illegible] space domain using observer canonical form

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

Conversion from *tf* to ss - Observer Canonical Form

...ple 3.5 Page 120: Convert the following transfer function to state-... domain using observer canonical form

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

Solution:

$$A = \begin{bmatrix} -9 & 1 & 0 \\ -26 & 0 & 1 \\ -24 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 7 \\ 2 \end{bmatrix}$$

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

MATLAB code for conversion from *tf* to ss

MATLAB code for conversion of *tf* to *ss*

```
num=[1 7 2] ;

den=[1 9 26 24];

[A,B,C,D]=tf2ss(num,den)

Aobsv=A'

Bobsv=B'

Cobsv=C'

Dobsv=D'
```

Common mistake by students using MATLAB code

Students think they are good programmers. They think they may used short variables. So, use the alphabet n for num and d for den.

```
n=[1 7 2] ;

d=[1 9 26 24];

[a,b,c,d]=tf2ss(n,d)
```

Problem in above code: state-space matrix d and transfer function denominator d have same alphabets.

Next week topics

- We already know that stability in *tf* domain is determined by poles.
- How about stability in state-space domain
- How about stability if we do not know the model of a system

Assignment for you

A hard disk drive (HDD) is a data storage device. It is used almost in every computing device including laptops, desktop computers, video game consoles, digital video recorders, mobiles and tablets. A hard disk stores data and nowadays we have too much data to store. Therefore, we require hard disks which can store more data (more data per square inch), which means the storage density of data is high. A hard disk uses magnetic storage system along with electronic hardware to access the data as shown in Figure below

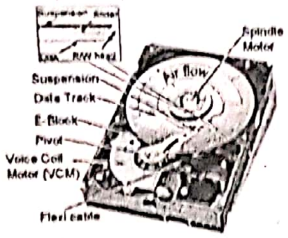

Figure. Hard disk drive schematic

Assignment for you

The electronic circuit of a hard disk consists of a dc motor. A dc motor has the following state-space:

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{i} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{b}{J} & \frac{K}{J} \\ 0 & -\frac{K}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0.5 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ i \end{bmatrix}$$

The above state-space equation is taken from the website
http://ctms.engin.umich.edu/CTMS/index.php?example=MotorPosition&sectio
ControlStateSpace

Using the values of $J = 3.2$, $b = 3.5$, $K = 0.0274$, $R = 4$ and $L = 2.75$, perform the following 3 tasks:

- Convert the state-space model to transfer function
- Check the stability of the hard disk system (in state-space and transfer function)
- Can you compute the range of L such that the hard disk system would be stable