

国家精品课程/ 国家精品资源共享课程/ 国家级精品教材

国家级十一(二)五规划教材/ 教育部自动化专业教学指导委员会牵头规划系列教材

控制系统仿真与CAD

# 第六章 非线性系统的建模与仿真

## 控制系统的Simulink 建模举例 (上)

Simulink Modeling Examples of Control Systems (I)



主讲：薛定宇教授



# 控制系统的建模与仿真举例

- Simulink 提供的工具可以把控制系统画出来
- 这里将介绍各种控制系统的建模仿真方法
  - 多变量控制系统
  - 计算机控制系统
  - 时变系统
  - 多采样速率系统
  - 延迟系统与变延迟系统
  - 切换系统
  - 随机输入系统



# 控制系统可用的模块

## ➤ 线性系统模块

➤ Simulink自带的Continuous模块组、Discrete模块组

➤ 标准的传递函数、状态方程、零极点、延迟、积分器、PID控制器

➤ Simulink Extras —— 非零初值的传递函数等

➤ 控制系统工具箱模块集——LTI对象模块

## ➤ 非线性系统模块

➤ 下节介绍



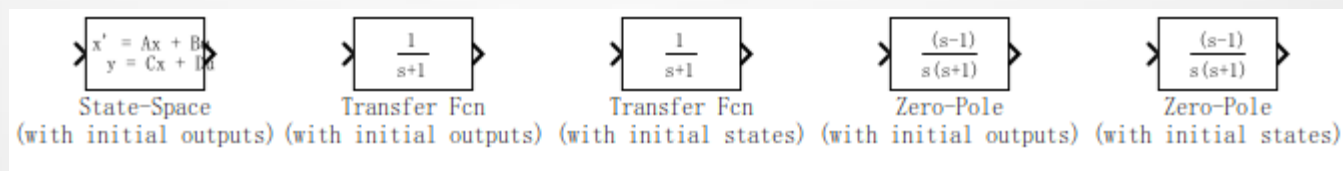
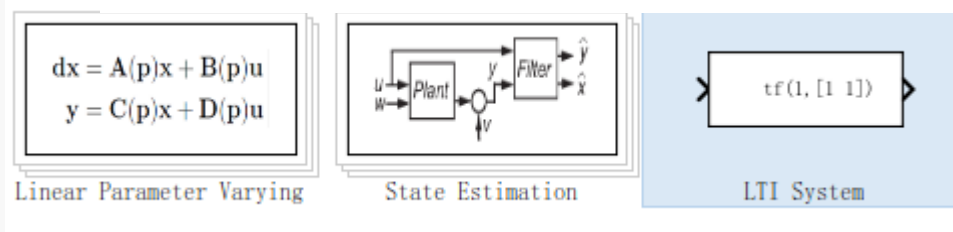
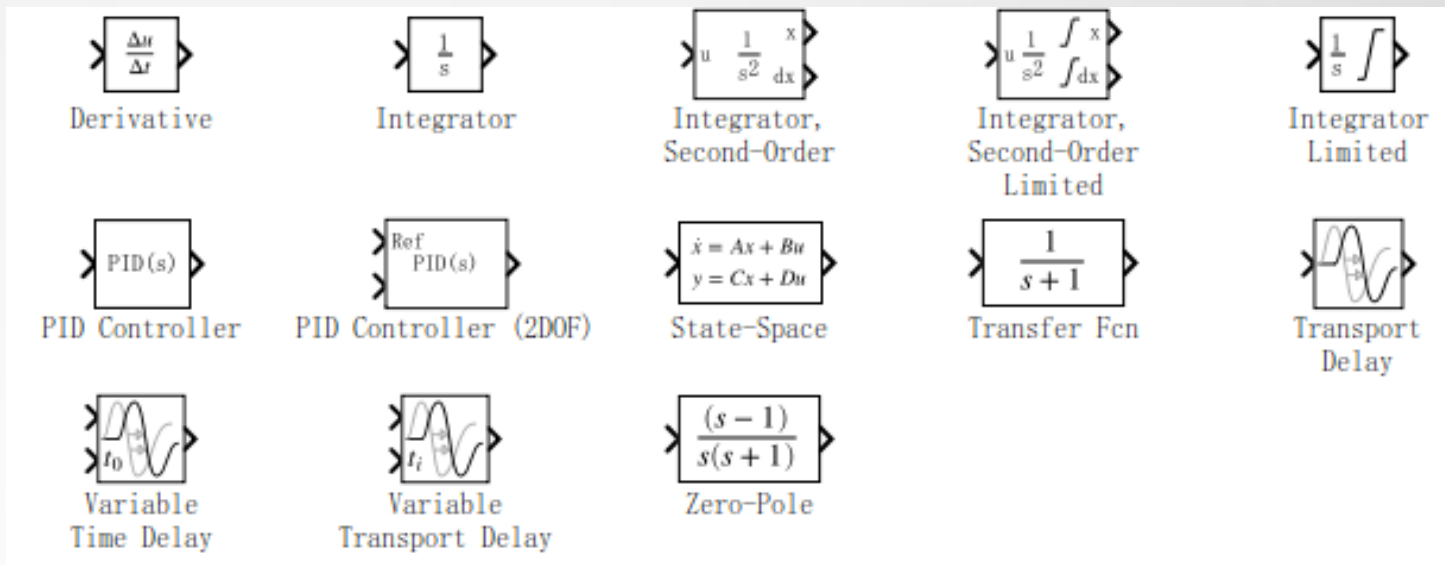
# 线性模块库

## ➤ 常规模块库

## ➤ 控制系统工具箱

## ➤ 非零初值系统的建模

### ➤ Simulink Extras -> Additional Linear





## 例6-3多变量时间延迟系统的仿真

### ➤ 多变量系统

$$G(s) = \begin{bmatrix} \frac{0.1134e^{-0.72s}}{1.78s^2 + 4.48s + 1} & \frac{0.924}{2.07s + 1} \\ \frac{0.3378e^{-0.3s}}{0.361s^2 + 1.09s + 1} & \frac{-0.318e^{-1.29s}}{2.93s + 1} \end{bmatrix}$$


### ➤ MATLAB仿真方法——局限性

### ➤ Simulink仿真



# MATLAB求解方法

## ➤ 补偿系统的阶跃响应

```
 >> g11=tf(0.1134,[1.78 4.48 1],'ioDelay',0.72);  
      g21=tf(0.3378,[0.361 1.09 1],'ioDelay',0.3);  
      g12=tf(0.924,[2.07 1]);  
      g22=tf(-0.318,[2.93 1],'ioDelay',1.29);  
      G=[g11, g12; g21, g22]; G=ss(G);  
      Kp=[0.1134,0.924; 0.3378,-0.318]; step(G*Kp,15)
```

## ➤ 局限性——如果有非线性环节，无能为力

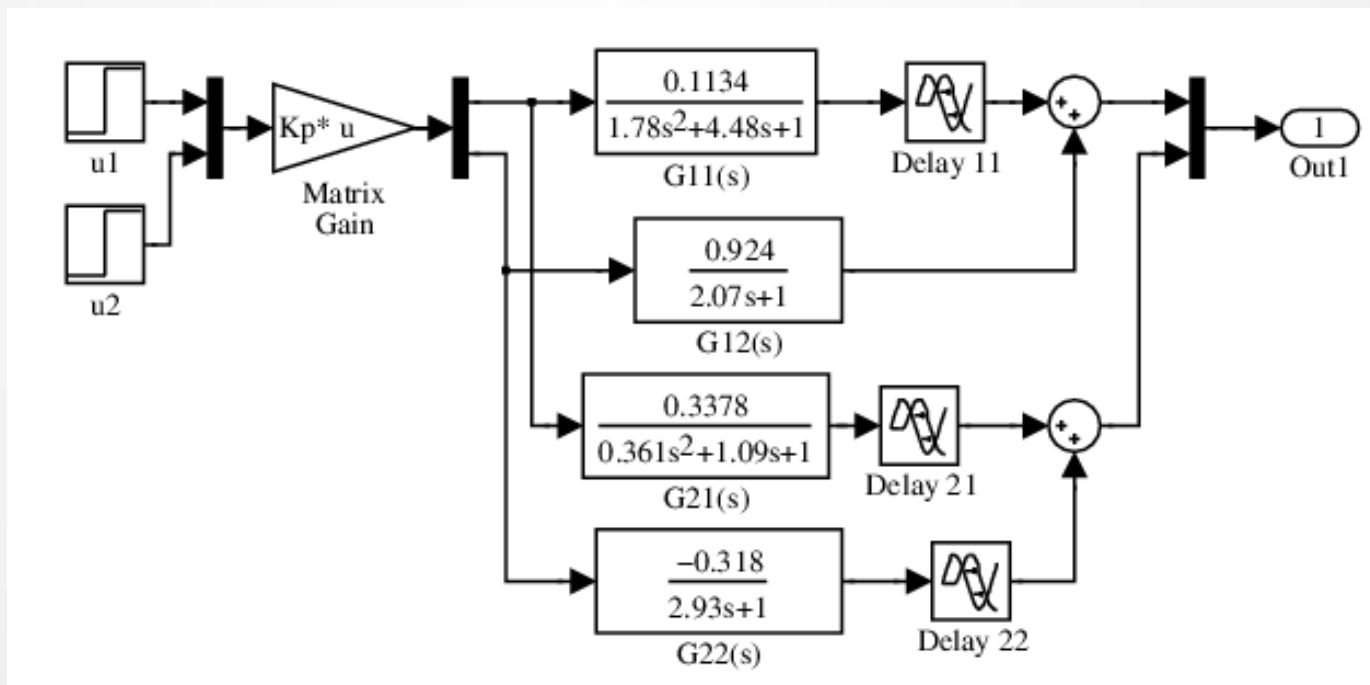


# Simulink仿真方法

## ➤ 早期版本建模方法

➤ 模型文件：c6mmimo.mdl

$$G(s) = \begin{bmatrix} \frac{0.1134e^{-0.72s}}{1.78s^2 + 4.48s + 1} & \frac{0.924}{2.07s + 1} \\ \frac{0.3378e^{-0.3s}}{0.361s^2 + 1.09s + 1} & \frac{-0.318e^{-1.29s}}{2.93s + 1} \end{bmatrix}$$







# 使用**LTI**模块的建模仿真方法

➤ 直接建模—— c6mmimon.slx

➤ 不同方法比较



```
>> u1=1; u2=0; [t1,a,y1]=sim('c6mmimo',15);  
    u1=0; u2=1; [t2,a,y2]=sim('c6mmimo',15);  
    [y,t]=step(G*Kp,15);
```



```
>> subplot(221), plot(t,y(:,1,1),':',t1,y1(:,1))  
    subplot(222), plot(t,y(:,1,2),':',t2,y2(:,1))  
    subplot(223), plot(t,y(:,2,1),':',t1,y1(:,2))  
    subplot(224), plot(t,y(:,2,2),':',t2,y2(:,2))
```

➤ 对复杂线性系统模型，尽量采用新版的LTI模块





## 例6-4 复杂系统模型的建模

- 复杂系统模型，如何建模？

$$G(s) = \frac{1 + \frac{3e^{-s}}{s+1}}{s+1}$$

- 底层建模

$$G(s) = \frac{1 + \frac{3e^{-s}}{s+1}}{s+1} \rightarrow \frac{1}{s+1} \left( 1 + \frac{3e^{-s}}{s+1} \right)$$

- 如果采用LTI，带有内部延迟的状态方程模型



```
>> s=tf('s'); G1=3/(s+1); G1.ioDelay=3; G=(1+G1)/(s+1)
```

