

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

控制系统计算机辅助设计

第5章: 线性系统的计算机辅助分析

主讲: 修贤超



线性系统的计算机辅助分析

- 5.1 线性系统性质分析
- 5.2 线性系统时域响应解析解法
- 5.3 线性系统的数字仿真分析
- 5.4 根轨迹分析
- 5.5 线性系统频域分析
- 5.6 多变量系统的频域分析



系统的分析方法

- 充分利用计算机对线性系统进行分析，在统一的框架下分析各种线性系统的性质
- 更新传统的系统分析的观念
- 求解传统方法难以求解的问题
 - 离散系统稳定性如何分析？
 - Nyquist图、Nichols图没有频率信息，如何弥补？
 - 高阶系统的根轨迹如何绘制？
 - 多变量系统如何进行频域分析？



线性系统的稳定性分析

- 给定线性系统模型，如何分析稳定性？

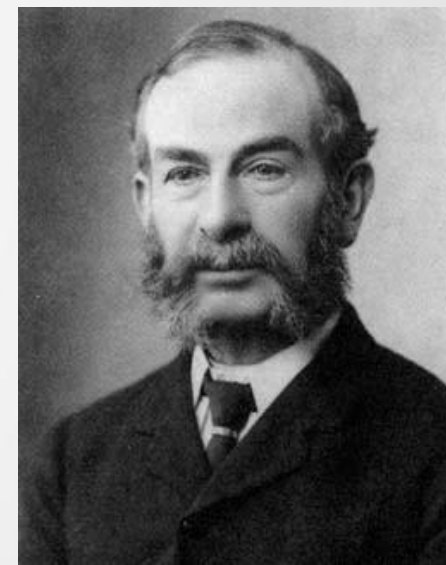
$$G(s) = \frac{10s^4 + 50s^3 + 100s^2 + 100s + 40}{s^7 + 21s^6 + 184s^5 + 870s^4 + 2384s^3 + 3664s^2 + 2496s}$$

- 单位负反馈闭环系统

- 由控制理论可知，用Routh表格可以判定该系统稳定性

- Edward John Routh (1831-1907)

- 历史局限性



Edward John Routh (1831-1907)



状态方程系统的稳定性

➤ 连续线性状态方程

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases}$$

➤ 解析阶

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

➤ 稳定性： \boldsymbol{A} 矩阵的特征根均有负实部



离散系统的稳定性

➤ 离散系统状态方程

$$\begin{cases} \mathbf{x}[(k+1)T] = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}\mathbf{u}(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}\mathbf{u}(kT) \end{cases}$$

➤ 离散系统时域响应解析阶

$$\mathbf{x}(kT) = \mathbf{F}^k \mathbf{x}(0) + \sum_{i=0}^{k-1} \mathbf{F}^{k-i-1} \mathbf{G}\mathbf{u}(iT)$$

➤ 稳定性判定：所有特征根均在单位圆内



Routh 判据的历史局限性

- Routh 判据提出时，没有求多项式根的方法
- 现在求解矩阵特征根、求解多项式方程的根轻而易举，无需间接方法
- Routh 判据只能得出是否稳定，进一步信息得不出来，如系统是否振荡
- 离散系统无法由 Routh 方法直接判定，得借助于 Jury 判据，更复杂
- 稳定性分析方法不统一



基于 MATLAB 的稳定性判定方法

➤ 直接判定

➤ 状态方程模型、传递函数模型等

➤ 由 $\text{eig}(G)$ 可以求出所有特征根

➤ 离散系统: $\text{abs}(\text{eig}(G))$

➤ $\text{isstable}(G)$ 也可以用于系统的稳定性判定

➤ 图解判定法

➤ 连续系统: $\text{pzmap}(G)$

➤ 离散系统: $\text{pzmap}(G)$, 同时画出单位圆



例5-1 连续闭环系统的稳定性分析

- 高阶开环模型，单位负反馈

$$G(s) = \frac{10s^4 + 50s^3 + 100s^2 + 100s + 40}{s^7 + 21s^6 + 184s^5 + 870s^4 + 2384s^3 + 3664s^2 + 2496s}$$

- 直接分析方法

```
>> den=[1,21,184,870,2384,3664,2496,0];  
num=[10,50,100,100,40]; G=tf(num,den); GG=feedback(G,1);  
zpk(GG), pzmap(GG), eig(GG), isstable(GG)
```

- 零极点模型

$$G(s) = \frac{10(s+2)(s+1)(s^2+2s+2)}{(s+6.922)(s+2.635)(s+0.01577)(s^2+4.127s+7.47)(s^2+7.3s+18.62)}$$



例5-2 离散系统的稳定性

- 高阶离散开环模型，单位负反馈系统， $T=0.1$

$$H(z) = \frac{6z^2 - 0.6z - 0.12}{z^4 - z^3 + 0.25z^2 + 0.25z - 0.125}$$

$$G_c(z) = 0.3 \frac{z - 0.6}{z + 0.8}$$

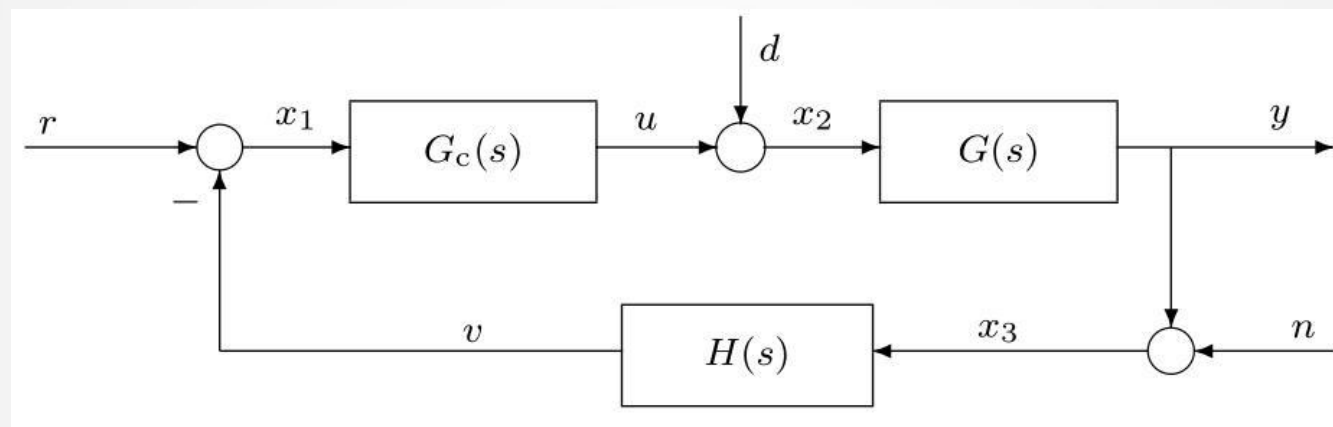
- MATLAB 求解

```
>> den=[1 -1 0.25 0.25 -0.125];  
num=[6 -0.6 -0.12]; H=tf(num,den,'Ts',0.1);  
z=tf('z','Ts',0.1); Gc=0.3*(z-0.6)/(z+0.8);  
GG=feedback(H*Gc,1); pzmap(GG), abs(eig(GG)), isstable(GG)
```



线性反馈系统的内部稳定性

➤ 典型闭环系统模型



- 输入、输出稳定是不够的，因为若内部信号可能过大，对系统作硬件破坏
- 应该引入内部稳定性概念，保证内部信号也是稳定的。



内部稳定性判定

- 由给定输入 r, d, n 到内部信号 x_1, x_2, x_3 都稳定的系统
- 内部稳定系统传递函数矩阵

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \frac{1}{M(s)} \begin{bmatrix} 1 & -G(s)H(s) & -H(s) \\ G_c(s) & 1 & -G_c(s)H(s) \\ G(s)G_c(s) & G(s) & 1 \end{bmatrix} \begin{bmatrix} r \\ d \\ n \end{bmatrix}$$

其中 $M(s) = 1 + G(s)G_c(s)H(s)$

- 内部稳定性定理 `key=intstable(G, Gc, H)`
 - 外部稳定 $1 + H(s)G(s)G_c(s)$
 - 无对消不稳定零极点 $H(s)G(s)G_c(s)$



线性系统的线性相似变换

- 系统的状态方程表示称为系统实现
- 不同状态选择下，状态方程不唯一
- 相似变换
 - 非奇异矩阵 T
 - 状态变换 $z = T^{-1}x$
 - 新状态方程模型
$$\begin{cases} \dot{z}(t) = A_t z(t) + B_t u(t) & z(0) = T^{-1}x(0) \\ y(t) = C_t z(t) + D_t u(t) \end{cases}$$



相似变换及其MATLAB求解

➤ 状态变换公式

$$A_t = T^{-1}AT, \quad B_t = T^{-1}B, \quad C_t = CT, \quad D_t = D$$

➤ MATLAB 直接求解方法

$$G_1 = \text{ss2ss}(G, T)$$

➤ 也可以根据上面公式进行变换



例5-4 系统的相似变换

➤ 已知系统和转换矩阵

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -24 & -50 & -35 & -10 \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} 24 & 24 & 7 & 1 \end{bmatrix} \boldsymbol{x}(t) \end{cases} \quad \boldsymbol{T} = \begin{bmatrix} & & & 1 \\ & & 1 & \\ & & & 1 \\ 1 & & & \end{bmatrix}$$

➤ MATLAB 求解

```
>> A=[0 1 0 0; 0 0 1 0; 0 0 0 1; -24 -50 -35 -10];  
G1=ss(A,[0;0;0;1],[24 24 7 1],0);  
T=flipplr(eye(4)); G2=ss2ss(G1,T)
```



线性系统的可控性分析

- 可控性定义
 - 假设系统由状态方程 (A, B, C, D) 给出
 - 对任意的初始时刻 t_0 ，如果状态中任一状态 $x(t)$ 可以从初始状态 $x(t_0)$ 处，由有界的输入信号 $u(t)$ 的驱动下，在有限时间 t_f 内能够到达任意预先指定的状态 $x(t_f)$ ，则称此状态是可控的。
- 若系统所有状态都可控，则称该系统为完全可控系统
- 通俗地说：系统的可控性就是指系统内部的状态是不是可以由外部输出信号控制的性质



线性系统的可控性判定

- 理论判定：可控性判定矩阵
 - 构造判定矩阵 $T_c = [B, AB, A^2B, \dots, A^{n-1}B]$
- 若矩阵 T_c 为满秩矩阵，则系统完全可控
- 基于 MATLAB 的判定方法
 - 构造可控性判定矩阵 $T_c = \text{ctrb}(A, B)$
 - 求秩、判定 $\text{rank}(T_c)$



例5-5 离散系统的可控性

➤ 离散状态方程模型

$$\boldsymbol{x}[(k+1)T] = \begin{bmatrix} -2.2 & -0.7 & 1.5 & -1 \\ 0.2 & -6.3 & 6 & -1.5 \\ 0.6 & -0.9 & -2 & -0.5 \\ 1.4 & -0.1 & -1 & -3.5 \end{bmatrix} \boldsymbol{x}(kT) + \begin{bmatrix} 6 & 9 \\ 4 & 6 \\ 4 & 4 \\ 8 & 4 \end{bmatrix} \boldsymbol{u}(kT)$$

➤ MATLAB 求解

```
>> A=[-2.2,-0.7,1.5,-1; 0.2,-6.3,6,-1.5; ...  
      0.6,-0.9,-2,-0.5; 1.4,-0.1,-1,-3.5];  
      B=[6,9; 4,6; 4,4; 8,4]; Tc=ctrb(A,B), rank(Tc)
```

➤ 连续、离散系统同等处理



可控性阶梯分解

- 对于不完全可控的系统阶梯分解
- 阶梯标准型

$$A_c = \begin{bmatrix} \hat{A}_{\bar{c}} & \mathbf{0} \\ \hat{A}_{21} & \hat{A}_c \end{bmatrix}, \quad B_c = \begin{bmatrix} \mathbf{0} \\ \hat{B}_c \end{bmatrix}, \quad C_c = [\hat{C}_{\bar{c}}, \hat{C}_c]$$

- MATLAB 函数调用

$$[A_c, B_c, C_c, T_c] = \text{ctrbf}(A, B, C)$$

- 若原系统状态方程完全可控，则不必分解



例5-6 阶梯标准型处理

➤ 不完全可控系统

$$\mathbf{x}[(k+1)T] = \begin{bmatrix} -2.2 & -0.7 & 1.5 & -1 \\ 0.2 & -6.3 & 6 & -1.5 \\ 0.6 & -0.9 & -2 & -0.5 \\ 1.4 & -0.1 & -1 & -3.5 \end{bmatrix} \mathbf{x}(kT) + \begin{bmatrix} 6 & 9 \\ 4 & 6 \\ 4 & 4 \\ 8 & 4 \end{bmatrix} \mathbf{u}(kT)$$

➤ MATLAB求解

```
>> A=[-2.2,-0.7,1.5,-1; 0.2,-6.3,6,-1.5; ...  
      0.6,-0.9,-2,-0.5; 1.4,-0.1,-1,-3.5];  
B=[6,9; 4,6; 4,4; 8,4]; C=[1 2 3 4];  
[Ac,Bc,Cc,Tc]=ctrbf(A,B,C)
```



线性系统的可观测性分析

- 可观测性定义
 - 假设系统由状态方程 (A, B, C, D) 给出
 - 对任意的初始时刻 t_0 ，若状态空间中任一状态 $x(t)$ 在任意有限时刻 t_f 的状态 $x(t_f)$ 可以由输出信号在这一时间区间内的值精确地确定出来，则称此状态是可观测的
- 若系统所有状态都可观测，则称系统为完全可观测系统
- 系统的可观测性就是指系统内部的状态是不是可以由系统输出信号重建起来的性质



可观测性判定

➤ 判定矩阵

$$T_o = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

➤ 等同于 (A^T, C^T) 系统的可控性判定

➤ 对偶系统: $(A, B, C) \Leftrightarrow (A^T, C^T, B^T)$

➤ MATLAB 求解 $T_o = \text{obsv}(A, C)$, $\text{rank}(T_o)$



Kalman 规范分解

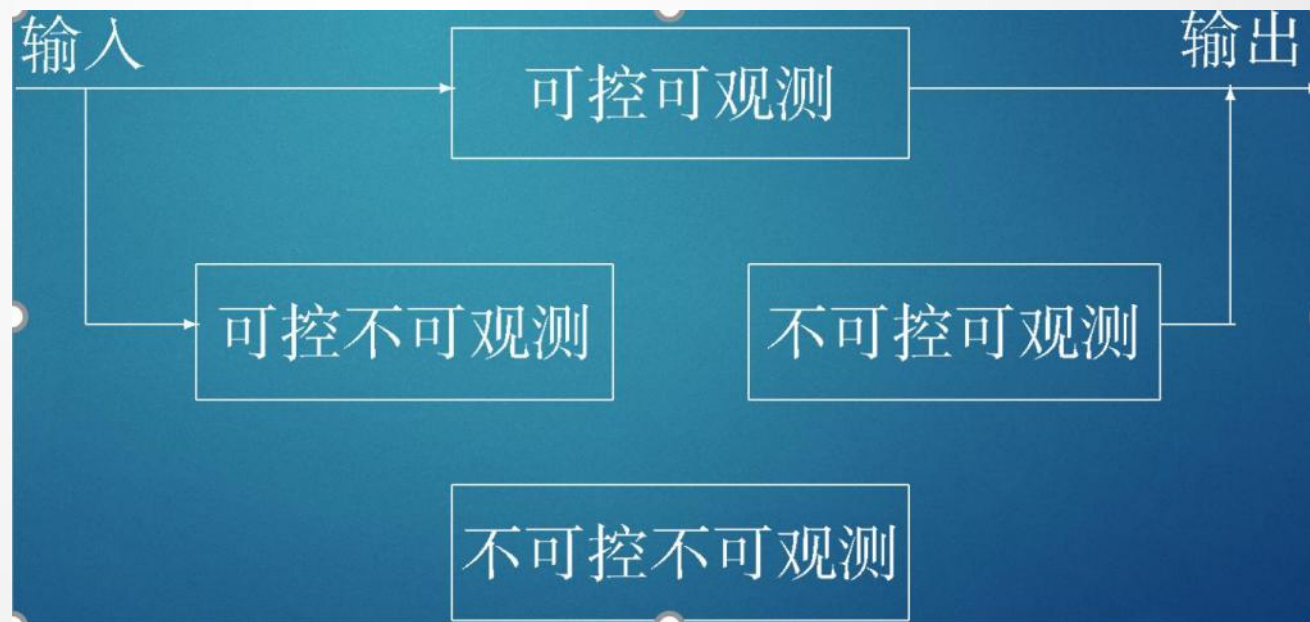
➤ Kalman 规范分解

$$\left\{ \begin{array}{l} \dot{z}(t) = \begin{bmatrix} \hat{A}_{\bar{c},\bar{o}} & \hat{A}_{1,2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{A}_{\bar{c},o} & \mathbf{0} & \mathbf{0} \\ \hat{A}_{3,1} & \hat{A}_{3,2} & \hat{A}_{c,\bar{o}} & \hat{A}_{3,4} \\ \mathbf{0} & \hat{A}_{4,2} & \mathbf{0} & \hat{A}_{c,o} \end{bmatrix} z(t) + \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \hat{B}_{c,\bar{o}} \\ \hat{B}_{c,o} \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} \mathbf{0} & \hat{C}_{\bar{c},o} & \mathbf{0} & \hat{C}_{c,o} \end{bmatrix} z(t) \end{array} \right.$$



四个子空间

- 既不可控又不可观测子空间；可控但不可观测子空间
- 不可控但可观测子空间，可控且可观测子空间
- 示意图
 - 最小实现





连续系统的范数测度及求解

➤ 系统也有范数：范数即测度

➤ \mathcal{H}_2 范数

$$\|G(s)\|_2 = \sqrt{\frac{1}{2\pi j} \int_{-j\infty}^{j\infty} \sum_{i=1}^p \sigma_i^2 [G(j\omega)] d\omega}$$

➤ \mathcal{H}_∞ 范数

$$\|G(s)\|_\infty = \sup_{\omega} \bar{\sigma} |G(j\omega)|$$

➤ \mathcal{H}_∞ 范数是系统频域响应的峰值

➤ MATLAB直接计算



离散系统的范数

➤ 离散系统的范数定义

➤ \mathcal{H}_2 范数

$$\|G(z)\|_2 = \sqrt{\int_{-\pi}^{\pi} \sum_{i=1}^p \sigma_i^2 [G(e^{j\omega})] d\omega}$$

➤ \mathcal{H}_∞ 范数 $\|G(z)\|_\infty = \sup_{\omega} \bar{\sigma} [G(e^{j\omega})]$

➤ 范数的 MATLAB 求解

`norm(G)` or `norm(G,2)`

`norm(G,inf)`



例5-9 离散系统范数计算

➤ 离散系统模型

$$\mathbf{x}[(k+1)T] = \begin{bmatrix} -2.2 & -0.7 & 1.5 & -1 \\ 0.2 & -6.3 & 6 & -1.5 \\ 0.6 & -0.9 & -2 & -0.5 \\ 1.4 & -0.1 & -1 & -3.5 \end{bmatrix} \mathbf{x}(kT) + \begin{bmatrix} 6 & 9 \\ 4 & 6 \\ 4 & 4 \\ 8 & 4 \end{bmatrix} \mathbf{u}(kT)$$
$$y(kT) = [1 \ 2 \ 3 \ 4] \mathbf{x}(kT)$$

➤ 直接计算

```
>> A=[-2.2,-0.7,1.5,-1; 0.2,-6.3,6,-1.5; ...  
      0.6,-0.9,-2,-0.5; 1.4,-0.1,-1,-3.5];  
B=[6,9; 4,6; 4,4; 8,4]; C=[1 2 3 4];  
G=ss(A,B,C,[0 0], 'Ts', 0.1); norm(G,2), norm(G,inf)
```



线性系统的计算机辅助分析

- 5.1 线性系统性质分析
- 5.2 线性系统时域响应解析解法
- 5.3 线性系统的数字仿真分析
- 5.4 根轨迹分析
- 5.5 线性系统频域分析
- 5.6 多变量系统的频域分析



系统的时域响应

- 线性系统时域响应的解析解
 - 状态方程的解析解
 - 状态增广方法
 - 直接积分方法
 - 传递函数模型的解析解
 - Laplace 变换与 z 变换, 延迟问题
- 时域响应的数值解
 - 阶跃响应、脉冲响应、任意输入、非零初值问题



直接积分方法

- 状态方程的解析解

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

- 将公式变代码的直接积分语句

$$\mathbf{y} = \mathbf{C} * (\text{expm}(\mathbf{A} * (t - t_0)) * \mathbf{x}_0 + \dots \\ \text{expm}(\mathbf{A} * t) * \text{int}(\text{expm}(-\mathbf{A} * \tau) * \mathbf{B} * \text{subs}(u, t, \tau), \tau, t_0, t))$$

- 得到结果后有必要用simplify()函数化简结果
- 若只需状态变量，则不用乘C



例5-10 状态方程的解析解

➤ 系统的状态方程为

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \begin{bmatrix} -19 & -16 & -16 & -19 \\ 21 & 16 & 17 & 19 \\ 20 & 17 & 16 & 20 \\ -20 & -16 & -16 & -19 \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 2 \end{bmatrix} u(t) \\ y(t) = [2, 1, 0, 0] \boldsymbol{x}(t) \end{cases}$$

➤ 状态变量初值 $\boldsymbol{x}^T(0) = [0, 1, 1, 2]$

➤ 输入信号 $u(t) = 2 + 2e^{-3t} \sin 2t$



方程的直接求解

➤ 系统模型

$$\begin{cases} \dot{\mathbf{x}}(t) = \begin{bmatrix} -19 & -16 & -16 & -19 \\ 21 & 16 & 17 & 19 \\ 20 & 17 & 16 & 20 \\ -20 & -16 & -16 & -19 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 2 \end{bmatrix} u(t) \\ y(t) = [2, 1, 0, 0] \mathbf{x}(t) \end{cases}$$

$$\mathbf{x}^T(0) = [0, 1, 1, 2], u(t) = 2 + 2e^{-3t} \sin 2t$$

➤ MATLAB语句

```
>> syms t tau; u=2+2*exp(-3*t)*sin(2*t);  
A=[-19,-16,-16,-19; 21,16,17,19; 20,17,16,20; -20,-16,-16,-19];  
B=[1; 0; 1; 2]; C=[2 1 0 0]; D=0; x0=[0; 1; 1; 2];  
y=C*(expm(A*t)*x0+expm(A*t)*int(expm(- ...  
A*tau)*B*subs(u,t,tau),tau,0,t)), y=simplify(y)
```



基于状态增广的解析解方法

- 状态方程模型
$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases}$$
- 解析解
$$\boldsymbol{x}(t) = e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t_0) + \int_{t_0}^t e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau$$
- 求解难点
$$\int_{t_0}^t e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau$$
- 想法：如果能用某种方法消去积分项最好



状态增广方法

- 如果输入信号为阶跃信号
- 可以引入增广的状态 $x_{n+1}(t) = u(t)$

$$\begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{x}_{n+1}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ x_{n+1}(t) \end{bmatrix}, \tilde{\mathbf{x}}(0) = \begin{bmatrix} \mathbf{x}(0) \\ 1 \end{bmatrix}$$

- 得出“自治系统”模型 $\tilde{\mathbf{A}}, \tilde{\mathbf{x}}(0)$
- 阶跃响应的解析解 $\tilde{\mathbf{x}}(t) = e^{\tilde{\mathbf{A}}t} \tilde{\mathbf{x}}(0)$



一般输入信号的系统增广

➤ 一般输入信号模型

$$u(t) = u_1(t) + u_2(t) = \sum_{i=0}^m c_i t^i + e^{d_1 t} \left[d_2 \cos(d_4 t) + d_3 \sin(d_4 t) \right]$$

➤ 引入增广状态变量

$$x_{n+1} = e^{d_1 t} \cos(d_4 t)$$

$$x_{n+2} = e^{d_1 t} \sin(d_4 t)$$

$$x_{n+3} = u_1(t), \dots, x_{n+m+3} = u_1^{(m-1)}(t)$$

➤ 增广状态方程模型 $\tilde{x}(t) = e^{\tilde{A}t} \tilde{x}(0)$



增广状态方程的一般形式

- 增广状态方程模型 \tilde{A} , $\tilde{x}(0)$

其中

$$\tilde{A} = \begin{bmatrix} A & d_2 B & d_3 B & B & 0 & \cdots & 0 \\ \hline 0 & d_1 & -d_4 & & & & 0 \\ & d_4 & d_1 & & & & \\ \hline & & & 0 & 1 & \cdots & 0 \\ & & & 0 & 0 & \cdots & 0 \\ 0 & & 0 & \vdots & \vdots & \ddots & \vdots \\ & & & 0 & 0 & \cdots & 0 \end{bmatrix}, \tilde{x}(t) = \begin{bmatrix} x(t) \\ \hline x_{n+1}(t) \\ x_{n+2}(t) \\ \hline x_{n+3}(t) \\ x_{n+4}(t) \\ \vdots \\ x_{n+m+3}(t) \end{bmatrix}, \tilde{x}(0) = \begin{bmatrix} x(0) \\ \hline 1 \\ 0 \\ \hline c_0 \\ c_1 \\ \vdots \\ c_m m! \end{bmatrix}$$

- 解析解 $\tilde{x}(t) = e^{\tilde{A}t} \tilde{x}(0)$

- MATLAB函数 $[G_a, \tilde{x}_0] = \text{ss_augment}(G, c, d, \tilde{x}_0)$

$$c = [c_0, c_1, \cdots, c_m], \text{ and } d = [d_1, d_2, d_3, d_4]$$



例5-11 状态方程的解析解

连续系统模型

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \begin{bmatrix} -19 & -16 & -16 & -19 \\ 21 & 16 & 17 & 19 \\ 20 & 17 & 16 & 20 \\ -20 & -16 & -16 & -19 \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 2 \end{bmatrix} u(t) \\ y(t) = [2, 1, 0, 0] \boldsymbol{x}(t) \end{cases}$$

$$\boldsymbol{x}^T(0) = [0, 1, 1, 2], u(t) = 2 + 2e^{-3t} \sin 2t$$

➤ 由MATLAB求解解析解

```
>> cc=[2]; dd=[-3,0,2,2]; x0=[0; 1; 1; 2];  
A=[-19,-16,-16,-19; 21,16,17,19; 20,17,16,20; -20,-16,-16,-19];  
B=[1; 0; 1; 2]; C=[2 1 0 0]; D=0; G=ss(A,B,C,D);  
[Ga,xx0]=ss_augment(G,cc,dd,x0); Ga.a, xx0'
```



基于Laplace变换的方法求解

➤ 连续系统的解析解法

$$G(s) = \frac{b_1 s^m + b_2 s^{m-1} + \dots + b_m s + b_{m+1}}{s^n + a_1 s^{n-1} + a_2 s^{n-2} + \dots + a_{n-1} s + a_n}$$

➤ 输入信号的 Laplace 变换 $U(s)$

➤ 输出信号的 Laplace 变换 $Y(s) = G(s) U(s)$

➤ 调用MATLAB的 `laplace()` 和 `ilaplace()` 函数可以直接求出系统的解析解



例5-12 阶跃响应解析解

➤ 系统传递函数模型

$$G(s) = \frac{s^3 + 7s^2 + 3s + 4}{s^4 + 7s^3 + 17s^2 + 17s + 6}$$

➤ 输入信号为阶跃信号 $R(s) = 1/s$

➤ 输出信号直接求解——不能使用LTI模型

```
>> syms s;  
G=(s^3+7*s^2+3*s+4)/(s^4+7*s^3+17*s^2+17*s+6);  
Y=G/s; y=ilaplace(Y)
```

➤ 另一组语句

```
>> syms s; U=laplace(sym(1));  
G=(s^3+7*s^2+3*s+4)/(s^4+7*s^3+17*s^2+17*s+6);  
Y=G*U; y=ilaplace(Y)
```



离散系统的解析解方法

- 已知离散传递函数 $G(z)$
 - 输入信号的 z 变换为 $U(z)$, `ztrans()` 函数
 - 计算出输出信号 $Y(z) = G(z)U(z)$
 - z 反变换求解解析解, `iztrans()` 函数
 - 有时可能需要手工化简



例5-13 离散系统阶跃响应

➤ 离散系统模型

$$G(z) = \frac{(z - 1/3)}{(z - 1/2)(z - 1/4)(z + 1/5)}$$

➤ 阶跃响应解析解

➤ 阶跃输入信号的 z 变换为 $z/(z - 1)$

➤ 解析求解

```
>> syms z; G=(z-1/3)/(z-1/2)/(z-1/4)/(z+1/5);  
R=z/(z-1); y=iztrans(G*R)
```



例5-14 有重根系统的解析解

➤ 受控对象模型 $G(z) = \frac{5z - 2}{(z - 1/2)^3(z - 1/3)}$

➤ 直接求解

```
>> syms z; G=(5*z-2)/(z-1/2)^3/(z-1/3);  
R=z/(z-1); y=iztrans(G*R)
```

➤ 手工化简

➤ $nchoosek(n-1,2)$ 是什么? $C_{n-1}^2 = \frac{(n-1)(n-2)}{2}$

```
>> syms n  
y=simplify(subs(y,nchoosek(n-1,2),(n-1)*(n-2)/2))
```



例5-15 连续延迟系统

➤ 传递函数

$$G(s) = \frac{s^3 + 7s^2 + 3s + 4}{s^4 + 7s^3 + 17s^2 + 17s + 6} e^{-2s}$$

➤ 时域响应求解思路 $G(s)e^{-Ls}$

- 先求出 $G(s)$ 在给定输入下的解析解 Y
- 用 $t - L$ 替换得出的 t , $\text{subs}(Y, t, t - L)$
- Heaviside函数的使用



求解方法

➤ 直接求解方法

```
>> syms s t; G=(s^3+7*s^2+3*s+4)/(s^4+7*s^3+17*s^2+17*s+6);  
    Y=G/s; y=ilaplace(Y); y=subs(y,t,t-2)
```

➤ 解的验证

```
>> fplot(y,[0,20])
```

➤ 解的修正

$$y = \begin{cases} 0, & t \leq 2 \\ 2/3 - 9e^{-2(t-2)} + 31e^{-3(t-2)}/12 - e^{-(t-2)}(14(t-2) - 23)/4, & t > 2 \end{cases}$$

$$y(t) = \left(2/3 - 9e^{-2(t-2)} + 31e^{-3(t-2)}/12 - e^{-(t-2)}(14(t-2) - 23)/4 \right) \times 1(t-2)$$

➤ Heaviside函数

```
>> fplot(y*heaviside(t-2),[0,20])
```



非零初值问题的求解

➤ 非零初值问题的数学形式

➤ 传递函数模型

$$G(s) = \frac{b_1 s^m + b_2 s^{m-1} + \cdots + b_m s + b_{m+1}}{s^n + a_1 s^{n-1} + a_2 s^{n-2} + \cdots + a_{n-1} s + a_n}$$

➤ 初始条件 $y(0), y'(0), \cdots, y^{(n-1)}(0)$

➤ 直接还原成微分方程

$$\begin{aligned} & \frac{d^n y(t)}{dt^n} + a_1 \frac{d^{n-1} y(t)}{dt^{n-1}} + \cdots + a_{n-1} \frac{dy(t)}{dt} + a_n y(t) \\ & = b_1 \frac{d^m u(t)}{dt^m} + b_2 \frac{d^{m-1} u(t)}{dt^{m-1}} + \cdots + b_m \frac{du(t)}{dt} + b_{m+1} u(t) \end{aligned}$$



例5-16 非零初值问题求解

➤ 传递函数模型 $G(s) = \frac{s^3 + 7s^2 + 3s + 4}{s^4 + 7s^3 + 17s^2 + 17s + 6}$

➤ 非零初值 $y(0) = 1, y'(0) = 0.5, y''(0) = y'''(0) = 0$

➤ 输入信号 $u(t) = \sin t$

➤ 变换回微分方程模型

$$y^{(4)} + 10y''' + 35y'' + 50y' + 24y = u''' + 7u'' + 24u' + 24u$$

```
>> syms t; u=sin(t);  
uu=diff(u,3)+7*diff(u,2)+24*diff(u)+24*u;  
y=dsolve(['D4y+10*D3y+35*D2y+50*Dy+24*y=', char(uu)], ...  
         'y(0)=1', 'Dy(0)=0.5', 'D2y(0)=0', 'D3y(0)=0')
```



二阶系统的阶跃响应及阶跃响应指标

➤ 二阶系统模型 $G_o(s) = \frac{\omega_n^2}{s(s + 2\zeta\omega_n)}$

➤ 闭环模型 $G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$

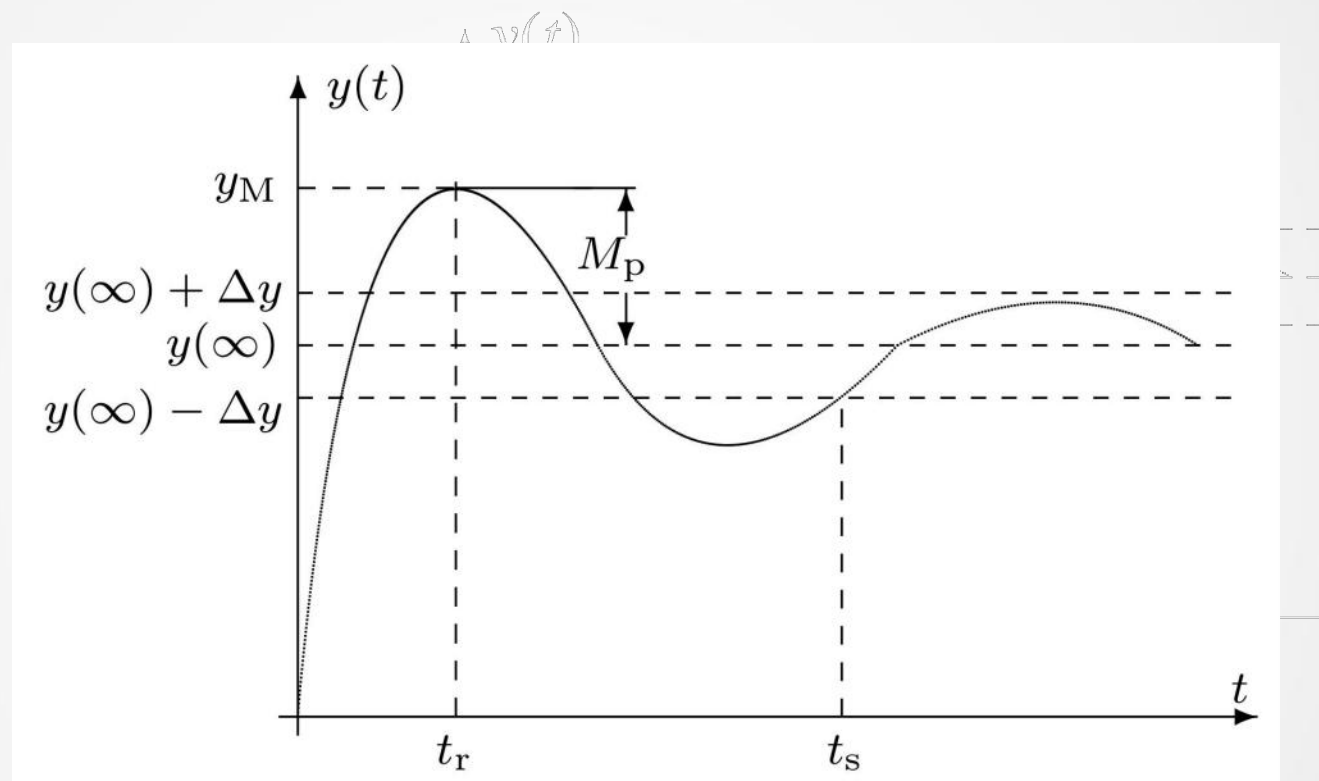
➤ 若 $\omega_n = 1$

```
>> t=[0:0.4:10]'; zet=[0:0.2:1,2,3]; yy=[]; wn=1; s=tf('s');  
for z=zet,  
    G=wn^2/(s^2+2*z*wn*s+wn^2); y=step(G,t); yy=[yy y];  
end  
ribbon(t,yy, 0.2)
```



阶跃响应指标

- 超调量
- 稳态值
- 上升时间
- 调节时间



- 好的伺服控制系统，应该具有稳态误差小或没有稳态误差、超调量小或没有超调量、上升时间短、调节时间短等性能



稳定性分析小结

- 系统稳定性是控制系统最重要的指标
- 如何判定稳定性？
 - 给出三种直接方法： eig, pzmap, isstable
 - 适用于任何线性系统
 - 有了直接方法，似乎没有必要再采用间接方法
- 闭环系统的内部稳定性 intstable



系统分析小结

- 系统的相似变换允许状态方程模型从一种形式变换成另一种形式 `ss2ss`
- 系统的可控性与可观测性定义与判定
 - 所用的函数 `ctrb`, `obsv`, `rank`
 - 阶梯形式 `ctrbf`
- 系统的 \mathcal{H}_2 范数与 \mathcal{H}_∞ 范数 `norm`



状态方程的解析解小结

- 状态方程的解析解是线性系统解析解的一种形式，这里只有连续系统的解
 - 第一种方法是利用 `int` 与 `expm` 等函数直接求解
 - 第二种方法是利用增广状态的方法求解
 - 相应的MATLAB函数 `s_augment`, `expm`
- 可以求解非零初值的系统模型在某些输入信号下时域响应的解析解



线性系统的解析解方法小结

- 传递函数模型
 - 连续系统可以通过 laplace 和 ilaplace 函数求解
 - 离散系统可以通过 ztrans 和 iztrans 函数求解
 - 手工化简 nchoosek
- 延迟函数的求解 subs 函数、heaviside 函数
- 二阶系统的解析解方法：几个指标



Q & A

感谢您的聆听和反馈

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

控制系统计算机辅助设计

第5章: 线性系统的计算机辅助分析

主讲: 修贤超



线性系统的计算机辅助分析

- 5.1 线性系统性质分析
- 5.2 线性系统时域响应解析解法
- 5.3 线性系统的数字仿真分析
- 5.4 根轨迹分析
- 5.5 线性系统频域分析
- 5.6 多变量系统的频域分析



线性系统的数字仿真分析

- 线性系统的解析解可以求解的条件
- 4 阶以上的系统需要求解 4 阶以上的多项式方程，根据 Abel 定理，无解析解。
- 实际应用需要数值解，需要阶跃响应曲线
- 主要内容
 - 线性系统的阶跃响应与脉冲响应
 - 任意输入下系统的响应
 - 非零初始条件下的时域响应



线性系统的阶跃响应与脉冲响应

➤ 阶跃响应曲线绘制函数

$\text{step}(G)$

$[y, t] = \text{step}(G)$

$[y, t] = \text{step}(G, t_f)$

$y = \text{step}(G, t)$

➤ 多系统曲线绘制

$\text{step}(G_1, '- ', G_2, '-.b', G_3, ':r')$



例5-17 延迟系统的阶跃响应

➤ 受控对象模型 $G(s) = \frac{10s + 20}{10s^4 + 23s^3 + 26s^2 + 23s + 10} e^{-s}$

➤ MATLAB 语句

```
>> G=tf([10 20],[10 23 26 23 10],'ioDelay',1);  
step(G,30);
```

➤ 可以从曲线上得到更多的信息，如超调量等

➤ 闭环阶跃响应

```
>> step(feedback(ss(G),1),30);
```



例5-18 离散化系统的响应

- 连续系统模型离散化 $G(s) = \frac{1}{s^2 + 0.2s + 1}e^{-s}$
- 采样周期 $T=0.01, 0.1, 0.5, 1.2$
- 求解

```
>> G=tf(1,[1 0.2 1],'ioDelay',1);  
G1=c2d(G,0.01,'zoh'); G2=c2d(G,0.1);  
G3=c2d(G,0.5); G4=c2d(G,1.2);  
step(G,'-',G2,'--',G3,':',G4,'-.',10)
```

- 得出的曲线可以比较



例5-19 多变量系统响应

➤ 多变量系统，阶跃响应

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

➤ MATLAB 求解语句：开环阶跃响应

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



例5-20 系统的耦合现象

- 系统耦合的概念
- 前置耦合矩阵

$$K_p = \begin{bmatrix} 0.1134 & 0.924 \\ 0.3378 & -0.318 \end{bmatrix}$$

- 补偿后系统的阶跃响应



```
>> Kp=[0.1134,0.924; 0.3378,-0.318];  
step(ss(G)*Kp)
```



例5-21 系统的脉冲响应曲线

- `impulse()` 函数与 `step()` 函数调用结构完全一致

`impulse(G)`, $[y, t] = \text{impulse}(G)$

$[y, t] = \text{impulse}(G, t_f)$, $y = \text{impulse}(G, t)$

- 系统模型 $G(s) = \frac{10s + 20}{10s^4 + 23s^3 + 26s^2 + 23s + 10} e^{-s}$

- MATLAB 求解

```
>> G=tf([10 20],[10 23 26 23 10],'ioDelay',1);  
impulse(G, 30);
```



例5-22 斜坡响应计算

- 可以利用 `step()` 和 `impulse()` 函数求解
 - 输出信号计算 $Y(s) = G(s)R(s)$
- 传递函数模型

$$G(s) = \frac{10s + 20}{10s^4 + 23s^3 + 26s^2 + 23s + 10} e^{-s}$$

- 斜坡响应等效于 $G(s)/s$ 的阶跃响应或 $G(s)/s^2$ 的脉冲响应

```
>> G=tf([10 20],[10 23 26 23 10],'ioDelay',1);  
s=tf('s'); step(G/s,50);
```



任意输入系统的时域响应

➤ 任意输入的时域响应 lsim

➤ 介绍系统辨识时使用过

➤ 调用格式

$$\text{lsim}(G, u, t)$$
$$y = \text{lsim}(G, u, t)$$

➤ 输入信号的表示——列向量构成的矩阵



例5-23 多变量系统任意输入响应

➤ 多变量系统

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

➤ 输入信号

$$u_1(t) = 1 - e^{-t} \sin(3t + 1), \quad u_2(t) = \sin(t) \cos(t + 2)$$

➤ MATLAB 求解

```
>> g11=tf(0.1134,[1.78 4.48 1],'ioDelay',0.72); g12=tf(0.924,[2.07 1]);  
g21=tf(0.3378,[0.361 1.09 1],'ioDelay',0.3);  
g22=tf(-0.318,[2.93 1],'ioDelay',1.29);  
G=[g11, g12; g21, g22]; t=[0:.1:15]';  
u=[1-exp(-t).*sin(3*t+1),sin(t).*cos(t+2)]; lsim(G,u,t);
```



非零初始条件下系统的时域响应

- 线性系统满足叠加原理
- 零初始条件问题 lsim 求解
- 非零初始条件、零输入系统的 initial 函数

$$[y, t] = \text{initial}(G, x_0, t_f)$$

- 叠加原理：将两个响应加起来即为所需



例5-24 任意输入非零初值的响应

➤ 状态方程

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \begin{bmatrix} -19 & -16 & -16 & -19 \\ 21 & 16 & 17 & 19 \\ 20 & 17 & 16 & 20 \\ -20 & -16 & -16 & -19 \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 2 \end{bmatrix} u(t) \\ y(t) = [2, 1, 0, 0] \boldsymbol{x}(t) \end{cases}$$

➤ 输入信号 $u(t) = 2 + 2e^{-3t} \sin 2t$, 初始状态 $\boldsymbol{x}^T(0) = [0, 1, 1, 2]$

➤ MATLAB 求解

```
>> A=[-19,-16,-16,-19; 21,16,17,19;  
      20,17,16,20; -20,-16,-16,-19];  
B=[1; 0; 1; 2]; C=[2 1 0 0]; G=ss(A,B,C,0);  
x0=[0; 1; 1; 2]; [y1,t]=initial(G,x0,10);  
u=2+2*exp(-3*t).*sin(2*t); y2=lsim(G,u,t); plot(t,y1+y2)
```



线性系统的计算机辅助分析

- 5.1 线性系统性质分析
- 5.2 线性系统时域响应解析解法
- 5.3 线性系统的数字仿真分析
- 5.4 根轨迹分析
- 5.5 线性系统频域分析
- 5.6 多变量系统的频域分析



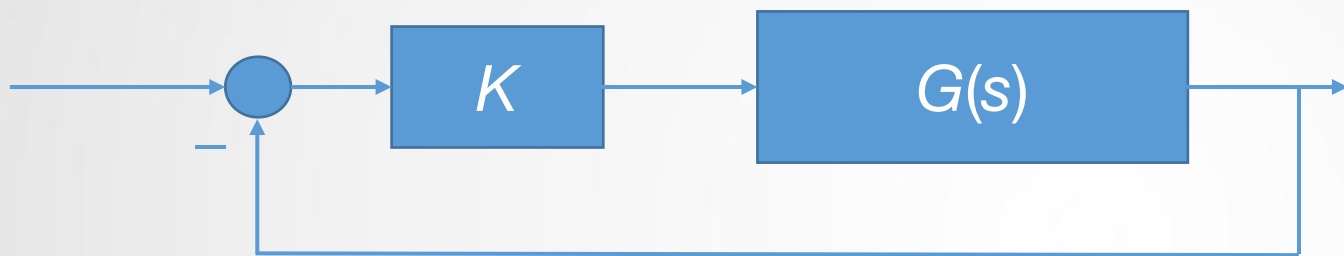
线性系统的根轨迹分析

- 根轨迹的基本概念与作用
- 基于MATLAB的根轨迹绘制方法
 - 连续系统的根轨迹
 - 离散系统的根轨迹
 - 延迟系统的近似根轨迹

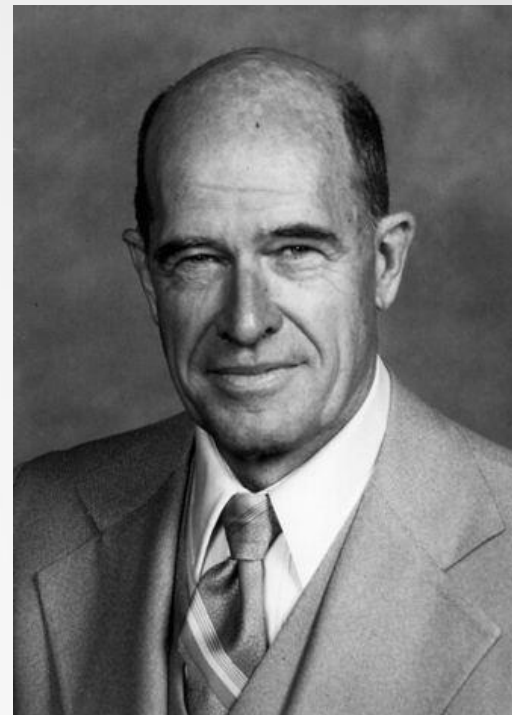


根轨迹分析基本概念

- 单变量开环传递函数 $G(s)$



- 闭环系统特征方程 $1 + KG(s) = 0$
- 对 K 的不同取值，则可能绘制出每个特征根变化的曲线，这样的曲线称为系统的根轨迹。
- 根轨迹用开环信息研究闭环特性



Walter Richard Evans (1920-1999)



根轨迹曲线的MATLAB绘制

➤ MATLAB 求解

`rlocus(G)`

`rlocus(G, K)`

`[R, K]=rlocus(G)`

`rlocus(G1, '-', G2, '-.b', G3, ':r')`

➤ 该函数可以用于单变量系统分析

➤ 单变量不含有时间延迟的连续系统

➤ 离散系统可以带有时间延迟



例5-27 开环系统

➤ 开环传递函数

$$G(s) = \frac{s^2 + 4s + 8}{s^5 + 18s^4 + 120.3s^3 + 357.5s^2 + 478.5s + 306}$$

➤ MATLAB 求解

```
>> num=[1 4 8]; den=[1,18,120.3,357.5,478.5,306];  
G=tf(num,den); rlocus(G)
```

➤ 如何求解临界增益？

➤ 闭环系统稳定性如何变化？



例5-28 基于根轨迹的设计方法

➤ 受控对象模型 $G(s) = \frac{10}{s(s+3)(s^2+2s+4)}$

➤ 根轨迹求解

```
>> s=tf('s'); G=10/(s*(s+3)*(s^2+3*s+4));  
      rlocus(G), grid
```

➤ 求出阻尼在 $\zeta = 0.707$ 处的增益

➤ 临界增益处阶跃响应

```
>> K=0.526; step(feedback(G*K,1))
```



例5-29 离散系统的根轨迹

- 离散系统开环模型， $T = 0.1$

$$G(z) = \frac{-0.95(z+0.51)(z+0.68)(z+1.3)(z^2 - 0.84z + 0.196)}{(z+0.66)(z+0.96)(z^2 - 0.52z + 0.1117)(z^2 + 1.36z + 0.7328)}$$

- 根轨迹绘制

```
>> z=tf('z','Ts',0.1);  
G=-0.95*(z+0.51)*(z+0.68)*(z+1.3)*...  
    (z^2-0.84*z+0.196)/((z+0.66)*(z+0.96)...  
    *(z^2-0.52*z+0.1117)*(z^2+1.36*z+0.7328));  
rlocus(G), grid
```



例5-30 无延迟离散模型

- 离散系统模型， $T = 0.1$

$$G(z) = \frac{0.52(z - 0.49)(z^2 + 1.28z + 0.4385)}{(z - 0.78)(z + 0.29)(z^2 + 0.7z + 0.1586)}$$

- MATLAB 求解

```
>> z=tf('z','Ts',0.1);  
G=0.52*(z-0.49)*(z^2+1.28*z+0.4385)/...  
    ((z-0.78)*(z+0.29)*(z^2+0.7*z+0.1586));  
rlocus(G), grid
```



例5-31 延迟离散系统的根轨迹

➤ 假设延迟为 6 步，则

$$G(z) = \frac{0.52(z - 0.49)(z^2 + 1.28z + 0.4385)}{(z - 0.78)(z + 0.29)(z^2 + 0.7z + 0.1586)} z^{-6}$$

```
>> z=tf('z','Ts',0.1);  
G=0.52*(z-0.49)*(z^2+1.28*z+0.4385)/...  
    ((z-0.78)*(z+0.29)*(z^2+0.7*z+0.1586));  
G.ioDelay=6; rlocus(G)
```

➤ 可以求临界增益，延迟系统临界增益减小



例5-32 正反馈系统的根轨迹

➤ 开环模型

$$G(s) = \frac{s^2 + 5s + 6}{s^5 + 13s^4 + 65s^3 + 157s^2 + 184s + 80}$$

➤ 正反馈系统

➤ 特征方程 $1 - KG(s) = 1 + K[-G(s)] = 0$

➤ 根轨迹绘制 `rlocus(-G)`

```
>> G=tf([1 5 6],[1 13 65 157 184 80]); rlocus(-G)
```



例5-32 延迟连续系统的根轨迹

➤ 连续延迟开环模型 $G(s) = \frac{6s + 4}{s(s^2 + 3s + 1)} e^{-2s}$

➤ 特征方程的推导

$$1 + KG(s) = 1 + K \frac{6s + 4}{s(s^2 + 3s + 1)} e^{-2s} = 0$$

$$s(s^2 + 3s + 1) + K(6s + 4)e^{-2s} = 0$$

➤ 如何求解根轨迹

➤ 用根轨迹做什么？

➤ 如果只求临界增益，则可以考虑使用Pade近似方法



例5-33 延迟系统的近似根轨迹

➤ 开环模型
$$G(s) = \frac{6s + 4}{s(s^2 + 3s + 1)} e^{-2s}$$

➤ 选择同的Pade近似阶次绘制根轨迹

```
>> s=tf('s'); G=(6*s+4)/s/(s^2+3*s+1);  
    G.ioDelay=2; rlocus(pade(G,2))
```

➤ 得出不同近似阶次下的临界增益

➤ 如果值差不多就可以得出近似的临界增益



例5-34 含参数系统的根轨迹

- 给定开环模型 $G(s) = \frac{5(s+5)(s^2+6s+12)}{(s+a)(s^3+4s^2+3s+2)}$
- 绘制关于 a 的根轨迹
- 令 $N_1(s) = 5(s+5)(s^2+6s+12)$, $D_1(s) = s^3+4s^2+3s+2$

➤ 推导

$$1 + \frac{N_1(s)}{(s+a)D_1(s)} = 0.$$

$$N_1(s) + (s+a)D_1(s) = 0$$

$$[N_1(s) + sD_1(s)] + aD_1(s) = 0,$$



含参数系统的根轨迹

➤ 继续推导 $[N_1(s) + sD_1(s)] + aD_1(s) = 0,$

$$1 + a\tilde{G}(s) = 0$$

$$\tilde{G}(s) = \frac{D_1(s)}{N_1(s) + sD_1(s)}.$$

➤ 绘制根轨迹

```
>> s=tf('s');  
N1=5*(s+5)*(s^2+6*s+12); D1=s^3+4*s^2+3*s+2;  
G1=D1/(N1+s*D1); rlocus(G1)
```



线性系统的数值仿真小结

- 线性系统的阶跃响应与脉冲响应
 - 阶跃响应 step 函数，脉冲响应 impulse 函数
 - 可以处理任意的线性模型：连续、离散、多变量、单变量、延迟与内部延迟、传递函数、状态方程、零极点，均可以统一求解
 - 可以同时绘制多个系统的响应，进行比较
- 线性系统任意输入的时域响应：lsim 函数
- 非零初值下系统的响应 initial 函数



系统根轨迹小结

根轨迹的概念

➤ 临界稳定增益，全极点模型的近似比例控制器设计

➤ 根轨迹函数能处理的问题

➤ 单变量系统模型

➤ 连续无延迟的精确根轨迹

➤ 离散系统的根轨迹（可含有延迟）

➤ 延迟模型的近似根轨迹、近似临界增益

➤ 含参数系统的根轨迹



Q & A

感谢您的聆听和反馈

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

控制系统计算机辅助设计

第5章: 线性系统的计算机辅助分析

主讲: 修贤超



线性系统的计算机辅助分析

- 5.1 线性系统性质分析
- 5.2 线性系统时域响应解析解法
- 5.3 线性系统的数字仿真分析
- 5.4 根轨迹分析
- 5.5 线性系统频域分析
- 5.6 多变量系统的频域分析



线性系统频域分析

➤ 频域分析

- Bell 实验室, Nyquist 1932, 反馈放大器稳定性
- Bode、Nichols 提出的新图形方法

➤ 主要内容

- 单变量系统的频域分析
- 利用频率特性分析系统的稳定性
- 系统的幅值裕度和相位裕度
- 多变量系统的频域分析



单变量系统的频域分析

- 用 $j\omega$ 取代 s ，则 $G(j\omega)$ 为复数增益
- 三种表示方法
 - 实部和虚部 $G(j\omega) = P(\omega) + jQ(\omega)$
实部与虚部关系曲线即为 Nyquist 图
Nyquist 图的缺陷：无对应频率信息
 - 幅值与相位 $G(j\omega) = A(\omega)e^{-j\phi(\omega)}$
横轴对数坐标 rad/s，纵轴分贝、度，Bode 图
 - 幅值与相位关系，Nichols 图，缺失频率信息
- 离散系统的频域响应 $z = e^{j\omega T}$



系统的 Nyquist 图

➤ Nyquist 曲线绘制

`nyquist(G)`

`nyquist(G, { ω_m, ω_M })`

`nyquist(G, ω)`

`[R, I, ω] = nyquist(G)`

`nyquist(G1, '- ', G2, '-.b', G3, ':r')`

➤ grid 命令绘制等 M 和等 N 圆



Harry Nyquist (1889–1976)



系统的 Bode 图与 Nichols 图

➤ Bode 图绘制

$\text{bode}(G)$, $\text{bode}(G, \{\omega_m, \omega_M\})$

$\text{bode}(G, \omega)$

$[A, \phi, \omega] = \text{bode}(G)$

$\text{bode}(G_1, '- ', G_2, '-.b', G_3, ':r')$

➤ Nichols 图由 $\text{nichols}()$ 函数绘制

➤ Nathaniel B. Nichols (1914–1997)

➤ 可以同样处理连续、离散、延迟、多变量系统，格式不变



Hendrik Wade Bode (1905-1982)



例5-36 系统的频域分析

➤ 开环传递函数 $G(s) = \frac{s + 8}{s(s^2 + 0.2s + 4)(s + 1)(s + 3)}$

➤ Nyquist 曲线绘制

```
>> s=tf('s');  
G=(s+8)/(s*(s^2+0.2*s+4)*(s+1)*(s+3));  
nyquist(G), grid
```

```
>> ylim([-1.5 1.5])
```

➤ MATLAB 曲线特色 $\omega \in (-\infty, \infty)$

➤ 读取频率信息；频率范围



例5-37 其他频域响应曲线

➤ Bode 图绘制

```
➤ >> bode(G)
```

➤ 快捷菜单读取特性

➤ Nichols 图的绘制

```
➤ >> nichols(G), grid
```

➤ 用鼠标读取频率信息

➤ 弥补了传统 Nichols 图的不同



例5-38 离散化模型与连续模型比较

- 对下面模型离散化， $T=0.1$

$$G(s) = \frac{s + 8}{s(s^2 + 0.2s + 4)(s + 1)(s + 3)}$$

- MATLAB 求解

```
>> s=tf('s'); G=(s+8)/(s*(s^2+0.2*s+4)*(s+1)*(s+3));  
      G1=c2d(G,0.1); bode(G,G1)
```

- 不同采样周期的离散模型 Bode 图

```
>> bode(G), hold on;  
      for T=[0.1:0.2:1], bode(c2d(G,T)); end
```



例5-36 离散系统频域分析

- 离散系统， $T = 0.1$

$$G(z) = \frac{0.2(0.3124z^3 - 0.5743z^2 + 0.3879z - 0.0889)}{z^4 - 3.233z^3 + 3.9869z^2 - 2.2209z + 0.4723}$$

- Nyquist 图与 Nichols 图

```
>> num=0.2*[0.3124 -0.5743 0.3879 -0.0889];  
den=[1 -3.233 3.9869 -2.2209 0.4723];  
G=tf(num,den,'Ts',0.1); nyquist(G); grid
```

```
>> nichols(G), grid
```



例5-37 连续延迟系统Nyquist图

➤ 延迟系统模型

$$G(s) = \frac{1}{s+1} e^{-2s}$$

➤ MATLAB 求解

➤ 不直接绘图，由nyquist() 函数得到数据

➤ 选择密集的频率点，得出光滑曲线

```
>> G=tf(1,[1 1],'ioDelay',2);  
w=logspace(-1,4,2000);  
[x,y]=nyquist(G,w); plot(x(:),y(:))
```



利用频率特性分析系统的稳定性

- Nyquist 定理
 - 如果开环模型含有 m 个不稳定极点
 - 则单位负反馈下单变量闭环系统稳定的充要条件是开环系统的 Nyquist 图逆时针围绕 $(-1, j0)$ 点 m 周。





Nyquist 定理的进一步解释

- 若开环模型 $G(s)H(s)$ 稳定
 - 则当且仅当 $G(s)H(s)$ 不包围 $(-1, j0)$ 点，闭环系统稳定的
 - 若Nyquist包围 $(-1, j0)$ 点 q 次，则有 q 个不稳定极点
- 若系统的开环模型 $G(s)H(s)$ 不稳定
 - 且有 p 个不稳定极点
 - 则当 $G(s)H(s)$ 的 Nyquist图逆时针包围 $(-1, j0)$ 点 p 次，闭环系统稳定
 - 若 Nyquist 图逆时针包围 $(-1, j0)$ 点 q 次，则闭环系统有 $p - q$ 个不稳定极点。



例5-38 闭环系统稳定性分析

➤ 开环模型

$$G(s) = \frac{2.7778(s^2 + 0.192s + 1.92)}{s(s+1)^2(s^2 + 0.384s + 2.56)}$$

➤ Nyquist 图

```
>> s=tf('s');  
G=2.7778*(s^2+0.192*s+1.92)...  
    /(s*(s+1)^2*(s^2+0.384*s+2.56));  
nyquist(G); axis([-2.5,0,-1.5,1.5]); grid
```

➤ 闭环阶跃响应

```
>> step(feedback(G,1))
```



系统的幅值裕度和相位裕度

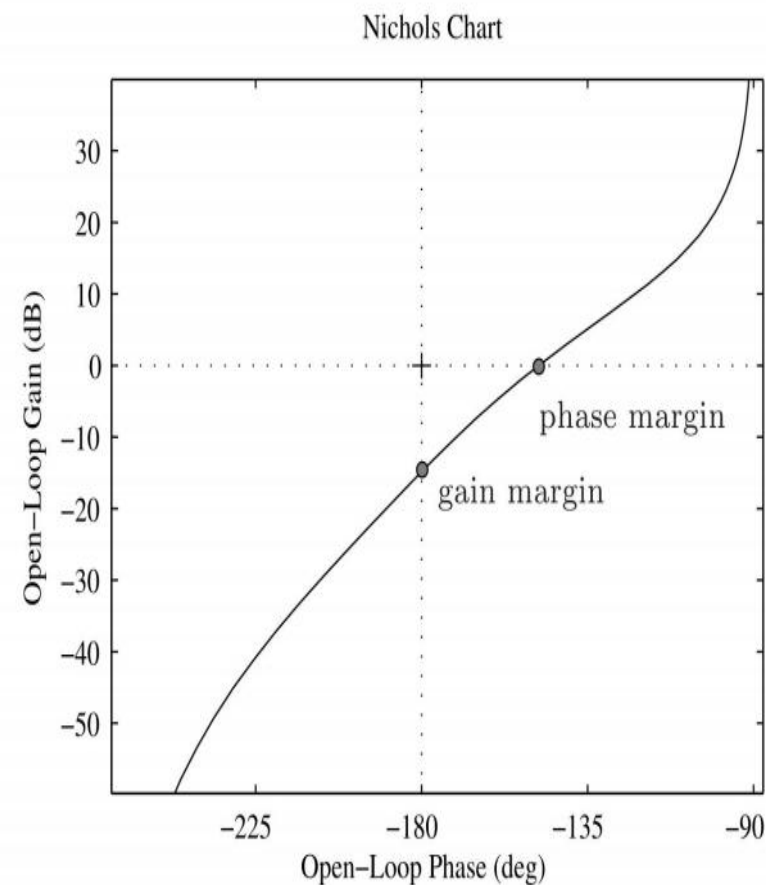
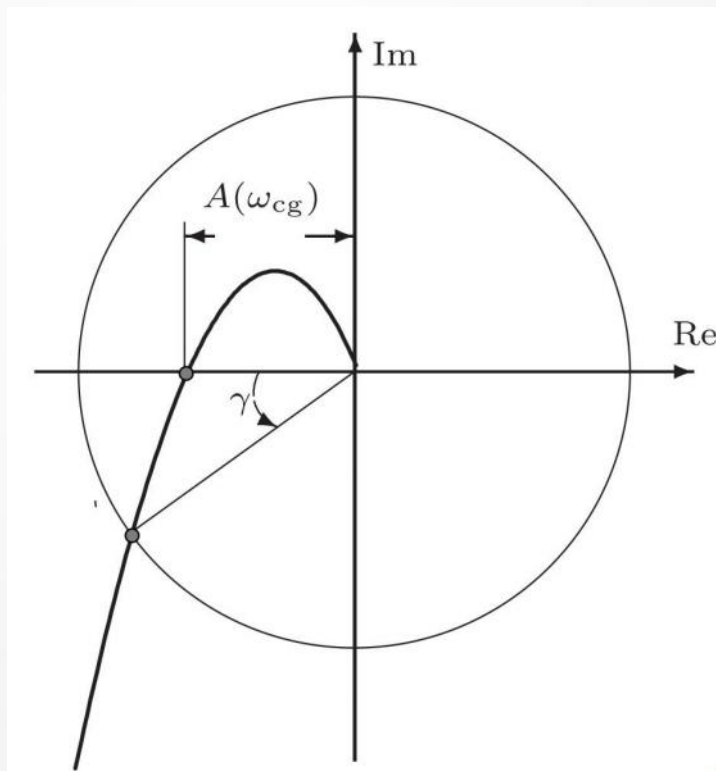
➤ 幅值裕度和相位裕度

➤ 幅值裕度

$$G_m = 1/A(\omega_{cg})$$

➤ 相位裕度

$$\gamma = \phi(\omega_{cp}) - 180^\circ$$





稳定性裕度分析几种特殊情况

- 如果系统的 Nyquist 图不与负实轴相交，则系统的幅值裕度为无穷大
- 如果系统的 Nyquist 图与负实轴在 $(-1, j0)$ 与 $(0, j0)$ 两个点之间有若干个交点，则系统的幅值裕度以离 $(-1, j0)$ 最近的点为准
- 如果系统的 Nyquist 图不与单位圆相交，则系统的相位裕度为无穷大
- 如果系统的 Nyquist 图在第三象限与单位圆有若干个交点，则系统的相位裕度以与离负实轴最近的为准



幅值相位裕度计算

- 如果系统的 Nyquist 图在第三象限与单位圆有若干个交点，则系统的相位裕度以与离负实轴最近的为准。
- MATLAB 求解方法

$$[G_m, \gamma, \omega_{cg}, \omega_{cp}] = \text{margin}(G);$$

- 如果某个裕度为无穷大，则返回 Inf，相应的频率值为 NaN
- 相位裕度——与单位圆没有交点



例5-39 幅值相位裕度计算

➤ 开环模型 $G(s) = \frac{2.7778(s^2 + 0.192s + 1.92)}{s(s+1)^2(s^2 + 0.384s + 2.56)}$

➤ MATLAB 求解

```
>> s=tf('s');  
G=2.7778*(s^2+0.192*s+1.92)/s/(s+1)^2/(s^2+0.384*s+2.56);  
[gm,pm,wg,wp]=margin(G)
```

➤ 幅值裕度为1.105，频率为0.96209rad/s，相位裕度为2.0985度，剪切频率为0.92607rad/s

➤ 由于幅相裕度小，系统闭环响应有强振荡



例5-40 多变量系统的直接分析

➤ 多变量系统模型

$$G(s) = \begin{bmatrix} \frac{0.806s + 0.264}{s^2 + 1.15s + 0.202} & \frac{-15s - 1.42}{s^3 + 12.8s^2 + 13.6s + 2.36} \\ \frac{1.95s^2 + 2.12s + 0.49}{s^3 + 9.15s^2 + 9.39s + 1.62} & \frac{7.15s^2 + 25.8s + 9.35}{s^4 + 20.8s^3 + 116.4s^2 + 111.6s + 18.8} \end{bmatrix}$$

➤ 直接Nyquist图分析

```
>> g11=tf([0.806 0.264],[1 1.15 0.202]);  
g12=tf([-15 -1.42],[1 12.8 13.6 2.36]);  
g21=tf([1.95 2.12 0.49],[1 9.15 9.39 1.62]);  
g22=tf([7.15 25.8 9.35],[1 20.8 116.4 111.6 18.8]);  
G=[g11, g12; g21, g22]; nyquist(G)
```



线性系统的计算机辅助分析

- 5.1 线性系统性质分析
- 5.2 线性系统时域响应解析解法
- 5.3 线性系统的数字仿真分析
- 5.4 根轨迹分析
- 5.5 线性系统频域分析
- 5.6 多变量系统的频域分析



多变量系统的频域分析

- 常规单变量系统的频域分析方法不能直接使用
- 多变量频域分析——现代频域法
 - 英国学派（British School）的贡献
- 多变量频域设计工具箱
 - MFD工具箱简介
 - 多变量系统的模型的标准输入方法



多变量系统的现代频域法

➤ 逆Nyquist阵列方法

➤ Howard Harry Rosenbrock (1920-2010), FRS

➤ 特征轨迹法

➤ Sir Alistair George James MacFarlane (1931-), FRS

➤ 其他设计方法

➤ David Quinn Mayne (1930-), FRS

➤ David H Owens, FREng



Howard Harry Rosenbrock (1920-2010)



Sir Alistair George James MacFarlane (1931-)



MFD 工具箱

- 英国剑桥大学的Boyel与Maciejowski教授开发基于MATLAB 的工具箱
- 多变量系统的描述
 - 可以用传递函数描述，但需要已知公分母
 - 可以求出系统的传递函数矩阵模型
$$[N, d] = \text{mvss2tf}(A, B, C, D)$$
 - 后面还将介绍更容易使用的接口函数
- 提供了大量的多变量系统频域分析与设计函数



例5-37 多变量系统模型

➤ 多变量模型

$$G(s) = \begin{bmatrix} \frac{s+4}{(s+1)(s+5)} & \frac{1}{5s+1} \\ \frac{s+1}{s^2+10s+100} & \frac{2}{2s+1} \end{bmatrix}$$

➤ 传递函数矩阵变换

```
>> s=tf('s');  
G=[(s+4)/((s+1)*(s+5)), 1/(5*s+1);  
    (s+1)/(s^2+10*s+100), 2/(2*s+1)]; G1=ss(G);  
[N,d]=mvss2tf(G1.a,G1.b,G1.c,G1.d)
```



传递函数矩阵的MFD表示方法

➤ 得出公分母

$$d(s) = s^6 + 16.7s^5 + 176.3s^4 + 767.1s^3 + 971.5s^2 + 415s + 50$$

➤ 分子矩阵 $N(s)$

$$\begin{bmatrix} s^5 + 14.7s^4 + 149.9s^3 + 499.4s^2 + 294s + 40 & 0.2s^5 + 3.3s^4 + 34.6s^3 + 146.5s^2 + 165s + 50 \\ s^5 + 7.7s^4 + 16s^3 + 13.4s^2 + 4.6s + 0.5 & s^5 + 16.2s^4 + 168.2s^3 + 683s^2 + 630s + 100 \end{bmatrix}$$

➤ 用这样的方法可以得出传递函数矩阵模型

➤ 可以得出 MFD 能使用的模型



Gershgorin 圆盘定理

➤ Semyon Aranovich Gershgorin (1901 - 1933)

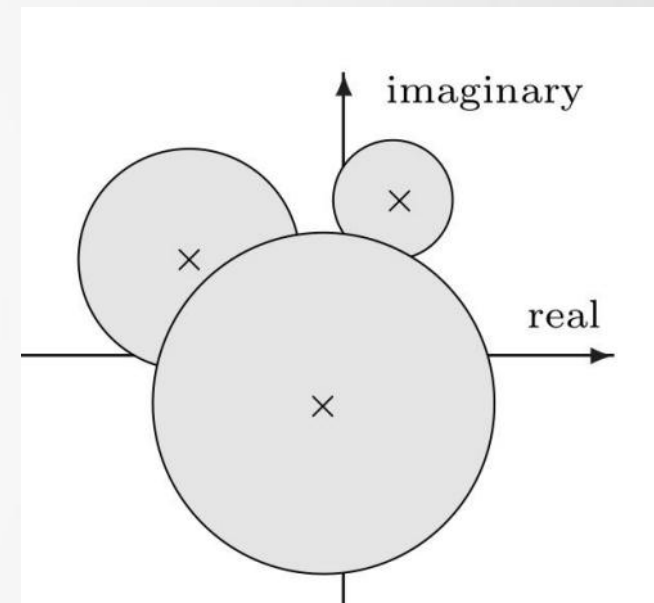
➤ 复数矩阵

$$C = \begin{bmatrix} c_{11} & \cdots & c_{1k} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ c_{k1} & \cdots & c_{kk} & \cdots & c_{kn} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ c_{n1} & \cdots & c_{nk} & \cdots & c_{nn} \end{bmatrix}$$

➤ 矩阵特征值满足

$$|\lambda - c_{kk}| \leq \sum_{j \neq k} |c_{kj}|, \quad |\lambda - c_{kk}| \leq \sum_{j \neq k} |c_{jk}|$$

➤ 对角占优矩阵





MATLAB求解

- MFD 工具箱的频域响应数据

$$H = \text{mv2fr}(N, d, w)$$

$$H = \text{mv2fr}(A, B, C, D, w)$$

- 对每一个频率而言 $G(j\omega)$ 也是复数矩阵

- 我编写的Nyquist 图+Gershgorin带绘制函数

$$\text{gershgorin}(H)$$

$$\text{gershgorin}(H, n_{ij})$$

- 早期版本 inagersh



MATLAB 函数编写

```
function gershgorin(H,key)
if nargin==1, key=0; end
t=[0:.1:2*pi,2*pi]'; [nr,nc]=size(H); nw=nr/nc; ii=1:nc;
for i=1:nc, circles{i}=[]; end
for k=1:nw
    G=H((k-1)*nc+1:k*nc,:);
    if nargin==2 & key==1, G=inv(G); end, H1(:, :,k)=G;
    for j=1:nc, ij=find(ii~=j);
        v=min([sum(abs(G(ij,j))),sum(abs(G(j,ij)))]);
        x0=real(G(j,j)); y0=imag(G(j,j));
        r=sum(abs(v));
        circles{j}=[circles{j} x0+r*cos(t)+sqrt(-1)*(y0+r*sin(t))];
    end, end
end, end
```



MATLAB函数编写

➤ 函数清单续

```
hold off; nyquist(tf(zeros(nc)), 'w'); hold on;
h=get(gcf, 'child'); h0=h(end:-1:2);
for i=ii, for j=ii,
    axes(h0((j-1)*nc+i)); NN=H1(i,j,:); NN=NN(:);
    if i==j, cc=circles{i}(:); x1=min(real(cc)); x2=max(real(cc));
        y1=min(imag(cc)); y2=max(imag(cc)); plot(NN),
        plot(circles{i}), plot(0,0, '+'), axis([x1,x2,y1,y2])
    else, plot(NN), end
end, end
```



例5-38 多变量模型

➤ 多变量模型

$$G(s) = \begin{bmatrix} \frac{0.806s + 0.264}{s^2 + 1.15s + 0.202} & \frac{-15s - 1.42}{s^3 + 12.8s^2 + 13.6s + 2.36} \\ \frac{1.95s^2 + 2.12s + 0.49}{s^3 + 9.15s^2 + 9.39s + 1.62} & \frac{7.15s^2 + 25.8s + 9.35}{s^4 + 20.8s^3 + 116.4s^2 + 111.6s + 18.8} \end{bmatrix}$$

➤ MATLAB 求解

```
>> g11=tf([0.806 0.264],[1 1.15 0.202]);  
g12=tf([-15 -1.42],[1 12.8 13.6 2.36]);  
g21=tf([1.95 2.12 0.49],[1 9.15 9.39 1.62]);  
g22=tf([7.15 25.8 9.35],[1 20.8 116.4 111.6 18.8]);  
  
>> G=[g11, g12; g21, g22]; w=logspace(-2,1.5);  
G=ss(G); H=mv2fr(G.a,G.b,G.c,G.d,w); gershgorin(H);
```



例5-39 补偿后系统的频域响应

➤ 前置补偿矩阵

$$K_p = \begin{bmatrix} 0.3610 & 0.4500 \\ -1.1300 & 1.0000 \end{bmatrix}$$

➤ MATLAB 求解

```
>> Kp=[0.3610,0.4500; -1.1300,1.0000]; G=ss(G*Kp);  
H=mv2fr(G.a,G.b,G.c,G.d,w); gershgorin(H);
```

➤ 对角占优性质明显



其他频域响应数据生成方法

- 串联模型 $G_1(s) G_2(s)$

$$H = \text{fmultf}(w, H_2, H_1)$$

- 并联模型 $G_1(s) + G_2(s)$

$$H = \text{faddf}(w, H_1, H_2)$$

- 其他

$$H = \text{fdly}(w, H_1, D), \quad H = \text{finv}(w, H_1)$$

$$H = \text{fmul}(w, H_1, K), \quad H = \text{fmul}(w, K, H_1)$$



全新的多变量频域分析函数

- 对复杂系统结构原 MFD 函数调用太麻烦
 - 原有的plotnyq、fgersh等函数
- 重新编写MATLAB函数处理多变量问题
 - 利用内部延迟的结构
 - 处理任意复杂度的结构
 - 可以完全取代前面介绍的 MFD 工具箱函数
 - 先得出模型，再用这个函数直接获得频域响应



多变量频域分析函数编写

- MFD工具箱函数使用过于繁琐，应新编程
- 把变换调整到LTI层次，函数清单

```
function H1=mfrd(G,w)
H=frd(G,w); h=H.ResponseData; H1=[];
for i=1:length(w); H1=[H1; h(:, :, i)]; end
```

- 按照MFD工具箱格式要求获得频域响应数据
- 调用格式 $H=mfrd(G, w)$



例5-39 多变量延迟系统

➤ 系统模型

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

➤ MATLAB 求解

```
>> G=[tf(0.1134,[1.78 4.48 1]), tf([0.924],[2.07,1]);  
      tf(0.3378,[0.361,1.09,1]), tf(-0.318,[2.93 1])];  
G=ss(G); D=[0.72 0; 0.3 1.29]; w=logspace(0,1);  
H=mv2fr(G.a,G.b,G.c,G.d,w); H1=fdly(w,H,D); gershgorin(H1);  
  
>> G1=G; G1.ioDelay=D; H=mfrd(G1,w); gershgorin(H1);
```



校正系统的频域响应分析

➤ 前置增益矩阵的计算 $K_p = G^{-1}(0)$

➤ 校正后特性绘制

```
>> H0=mv2fr(G.a,G.b,G.c,G.d,0); Kp=inv(H0);  
      H2=fmul(w,H1,Kp); gershgorin(H2);
```

➤ 简单的计算方法

```
>> H2=mfrd(G1*Kp,w); gershgorin(H2);
```

➤ Gershgorin带的稳定性判定定理



多变量系统的稳定性

➤ 如果 $G(s)$ 为 $m \times m$ 方阵

➤ 第 (i, j) 元素为 $\hat{g}_{ij}(s)$, 存在 $K = \text{diag}(k_1, k_2, \dots, k_m)$

➤ 对所有的 s 均满足

$$|\hat{g}_{ii}(s) + k_i| > \sum_{j \neq i} |\hat{g}_{ij}(s)|$$

➤ 若 $\hat{G}(s)$ 为第 i 个 Gershgorin 带, 且围绕 $(-k_i, j0)$ 点 N_i 次

➤ 当且仅当 $\sum_i N_i = Z_0$ 则回差矩阵 $-G(s)K$ 稳定

➤ 这里 Z_0 是 $G(s)$ 在右半平面的传输零点个数



多变量系统的奇异值曲线绘制

- 单变量系统有 Bode 图
 - 多变量系统能否采用这样的方法分析？
- 传递函数矩阵在 ω 点的奇异值 $\sigma_1(\omega), \sigma_2(\omega), \dots, \sigma_m(\omega)$
- 传递函数矩阵的奇异值可以作为轨迹绘制出来，称为奇异值曲线
- 奇异值曲线是多变量系统鲁棒控制中的重要指标
 - 由 `sigma()` 函数绘制
 - 该函数和 `bode()` 函数调用格式一致



例5-40 多变量延迟模型

➤ 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 绘制奇异值曲线

```
>> G=[tf(0.1134,[1.78 4.48 1],'ioDelay',0.72),...  
      tf([0.924],[2.07,1]);  
      tf(0.3378,[0.361,1.09,1],'ioDelay',0.3), ...  
      tf(-0.318,[2.93 1],'ioDelay',1.29)];  
sigma(G)
```



系统频域分析小结

- 系统频域响应数据的表示方法
 - Bode图 bode
 - Nyquist图 nyquist
 - Nichols图 Nichols
- 延迟模型的频域响应



系统频域响应分析小结

- 利用Nyquist定理也可以分析系统的稳定性
 - 利用系统开环系统模型的不稳定极点个数
 - 与Nyquist图围绕 $(-1, j0)$ 点的周数
 - 分析单位负反馈系统的稳定性或不稳定极点个数
- 引入幅值与相位裕度的概念
 - 几何解释与函数调用 `margin`
 - 从某种程度上分析系统的鲁棒性



多变量系统频域分析小结

- 多变量系统的Nyquist阵列、Gershgorin带
 - MFD工具箱的模型表示: `mv2ss()` 函数
 - 多变量系统频域响应数据求取: `mv2fr()`
 - 频域响应数据计算: `fmul()`、`finv()`等一系列底层函数的调用
 - 新编写的`mfrd()`函数, 避免底层函数调用
 - 带有Gershgorin带的Nyquist图形, 对角占优判定
 - 多变量系统的稳定性分析
- 多变量系统的奇异值曲线绘制 `sigma()`



Q & A

感谢您的聆听和反馈