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

第四章 线性系统的数学模型

模型降阶方法

Model Reduction Methods

主讲：薛定宇教授

线性系统的模型降阶

- 模型降阶与最小实现
- 传递函数的经典降阶方法
  - 基于Pade近似的降阶方法与实现
  - Routh降阶方法
- 状态方程的降阶
  - 均衡实现的降阶方法
  - 最优Hankel范数的方法

最小实现与模型降阶

- 最小实现是数学意义下的严格化简（没有近似）
- 模型降阶是用低阶模型去逼近高阶模型
  - 最早由Edward J. Davison提出(1966)
- 本节介绍的方法
  - 传递函数的 Padé与 Routh 算法
  - 状态空间的降阶算法

模型降阶的需求

- 已知原始模型

$$G(s)=\frac{b_1 s^m+b_2 s^{m-1}+\cdots+b_m s+b_{m+1}}{a_1 s^n+a_2 s^{n-1}+a_3 s^{n-2}+\cdots+a_n s+a_{n+1}}$$

- 寻求降阶模型

$$G_{r/k}(s)=\frac{\beta_1 s^r+\beta_2 s^{r-1}+\cdots+\beta_{r+1}}{\alpha_1 s^k+\alpha_2 s^{k-1}+\cdots+\alpha_k s+\alpha_{k+1}}$$

- 为简单起见假设 $\alpha_{k+1}=1$

时间矩量的计算

- 展开原模型

$$G(s) = c_0 + c_1 s + c_2 s^2 + \cdots$$

- 其中时间矩量　可以递推求出

$$c_0 = b_{k+1}, \quad \text{and} \quad c_i = b_{k+1-i} - \sum_{j=0}^{i-1} c_j a_{n+1-i+j}, \quad i = 1, 2, \cdots$$

- 若已知状态方程模型

$$c_i = \frac{1}{i!} \frac{\mathrm{d}^i G(s)}{\mathrm{d}s^i}\bigg|_{s=0} = -\boldsymbol{C}\boldsymbol{A}^{-(i+1)}\boldsymbol{B}, \quad i = 0, 1, \cdots$$

Pade降阶的思想

- 时间矩量的MATLAB求解

```
function M=timmomt(G,k)
G=ss(G); C=G.c; B=G.b; iA=inv(G.a);
iA1=iA; M=zeros(1,k);
for i=1:k, M(i)=-C*iA1*B; iA1=iA*iA1; end
```

- Padé降阶思想：保留前 $r+k+1$ 个时间矩量

$$\sum c_i s^i = \frac{\sum \beta_i s^i}{\sum \alpha_i s^i} \longrightarrow \sum c_i s^i \sum \alpha_i s^i = \sum \beta_i s^i$$

Pade降阶模型系数推到

- 对比系数，则

$$
\begin{cases}
\beta_{r+1} = c_0 \\
\beta_r = c_1 + \alpha_k c_0 \\
\qquad\qquad \vdots \\
\beta_1 = c_r + \alpha_k c_{r-1} + \cdots + \alpha_{k-r+1} c_0 \\
0 = c_{r+1} + \alpha_k c_r + \cdots + \alpha_{k-r} c_0 \\
0 = c_{r+2} + \alpha_k c_{r+1} + \cdots + \alpha_{k-r-1} c_0 \\
\qquad\qquad \vdots \\
0 = c_{k+r} + \alpha_k c_{k+r-1} + \cdots + \alpha_2 c_{r+1} + \alpha_1 c_r
\end{cases}
$$

Pade降阶公式

- 这样可以得出

$$\begin{bmatrix} c_r & c_{r-1} & \cdots & . \\ c_{r+1} & c_r & \cdots & . \\ \vdots & \vdots & \ddots & \vdots \\ c_{k+r-1} & c_{k+r-2} & \cdots & c_r \end{bmatrix} \begin{bmatrix} \alpha_k \\ \alpha_{k-1} \\ \vdots \\ \alpha_1 \end{bmatrix} = - \begin{bmatrix} c_{r+1} \\ c_{r+2} \\ \vdots \\ c_{k+r} \end{bmatrix}$$

$$\begin{bmatrix} \beta_{r+1} \\ \beta_r \\ \vdots \\ \beta_1 \end{bmatrix} = \begin{bmatrix} c_0 & 0 & \cdots & 0 \\ c_1 & c_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ c_r & c_{r-1} & \cdots & c_0 \end{bmatrix} \begin{bmatrix} 1 \\ \alpha_k \\ \vdots \\ \alpha_{k-r+1} \end{bmatrix}$$

- 可以得出降阶模型的分子、分母系数

Pade降阶的求解函数

- Pad é降阶求解函数

```
function G_r=pademod(G_Sys,r,k)
c=timmomt(G_Sys,r+k+1); G_r=pade_app(c,r,k);
```

```
function Gr=pade_app(c,r,k)
w=-c(r+2:r+k+1)';
vv=[c(r+1:-1:1)'; zeros(k-1-r,1)];
W=rot90(hankel(c(r+k:-1:r+1),vv));
V=rot90(hankel(c(r:-1:1)));
x=[1 (W\w)']; dred=x(k+1:-1:1)/x(k+1);
y=[c(1) x(2:r+1)*V'+c(2:r+1)];
nred=y(r+1:-1:1)/x(k+1); Gr=tf(nred,dred);
```

例4-26 Padé近似举例

- 原始模型 $G(s)=\dfrac{s^3+7s^2+11s+5}{s^4+7s^3+21s^2+37s+30}$
- Padé近似

```
>> G=tf([1,7,11,5],[1,7,21,37,30]);
   Gr=pademod(G,1,2), step(G,Gr,'--')
```

```
>> bode(G,Gr,'--')
```

- 用二阶模型近似原始的四阶模型

$$G_{\rm r}(s)=\frac{0.8544s+0.6957}{s^2+1.091s+4.174}$$

例4-27 Pade降阶的反例

- 原始模型

$$G(s)=\frac{0.067s^5+0.6s^4+1.5s^3+2.016s^2+1.55s+0.6}{0.067s^6+0.7s^5+3s^4+6.67s^3+7.93s^2+4.63s+1}$$

$$=\frac{(s+5.92)(s+1.221)(s+0.897)(s^2+0.9171s+1.381)}{(s+2.805)(s+1.856)(s+1.025)(s+0.501)(s^2+4.261s+5.582)}$$

- 稳定模型，但降阶模型不稳定

```
>> num=[0.067,0.6,1.5,2.016,1.66,0.6];
   den=[0.067 0.7 3 6.67 7.93 4.63 1]; G=tf(num,den);
   zpk(G), Gr=pademod(G,1,3); zpk(Gr)
```

Routh 降阶方法与实例

- Routh算法（较烦琐，从略）

```
function G_r=routhmod(G_Sys,nr)
num=G_Sys.num{1}; den=G_Sys.den{1}; n0=length(den); n1=length(num);
a1=den(end:-1:1); b1=[num(end:-1:1) zeros(1,n0-n1-1)];
for k=1:n0-1,
   k1=k+2; alpha(k)=a1(k)/a1(k+1); beta(k)=b1(k)/a1(k+1);
   for i=k1:2:n0-1,
      a1(i)=a1(i)-alpha(k)*a1(i+1); b1(i)=b1(i)-beta(k)*a1(i+1);
end, end
nn=[]; dd=[1]; nn1=beta(1); dd1=[alpha(1),1]; nred=nn1; dred=dd1;
for i=2:nr,
   nred=[alpha(i)*nn1, beta(i)]; dred=[alpha(i)*dd1, 0];
   n0=length(dd); n1=length(dred); nred=nred+[zeros(1,n1-n0),nn];
   dred=dred+[zeros(1,n1-n0),dd];
   nn=nn1; dd=dd1; nn1=nred; dd1=dred;
end
G_r=tf(nred(nr:-1:1),dred(end:-1:1));
```

例4-28 Routh降阶

- 仍考虑稳定模型

$$G(s)=\frac{0.067s^5+0.6s^4+1.5s^3+2.016s^2+1.55s+0.6}{0.067s^6+0.7s^5+3s^4+6.67s^3+7.93s^2+4.63s+1}$$

- MATLAB求解

```
>> num=[0.067,0.6,1.5,2.016,1.66,0.6];
   den=[0.067 0.7 3 6.67 7.93 4.63 1];
   G=tf(num,den); Gr=zpk(routhmod(G,3)), step(G,Gr,'--')
```

```
>> bode(G,Gr,'--')
```

- Routh算法降阶后能保证降阶模型稳定性

状态方程模型的降阶算法

➢ 均衡实现模型的降阶算法

$$\begin{bmatrix} \dot{\boldsymbol{x}}_1 \\ \dot{\boldsymbol{x}}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} \\ \boldsymbol{A}_{21} & \boldsymbol{A}_{22} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \end{bmatrix} u, \quad y = [\boldsymbol{C}_1 \;\; \boldsymbol{C}_2] \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix} + Du$$

令 $\dot{\boldsymbol{x}}_2 = 0$ ↓ 并消去 x_2

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \left(\boldsymbol{A}_{11} - \boldsymbol{A}_{12}\boldsymbol{A}_{22}^{-1}\boldsymbol{A}_{21}\right)\boldsymbol{x}_1 + \left(\boldsymbol{B}_1 - \boldsymbol{A}_{12}\boldsymbol{A}_{22}^{-1}\boldsymbol{B}_2\right)u \\ y = \left(\boldsymbol{C}_1 - \boldsymbol{C}_2\boldsymbol{A}_{22}^{-1}\boldsymbol{A}_{21}\right)\boldsymbol{x}_1 + \left(D - \boldsymbol{C}_2\boldsymbol{A}_{22}^{-1}\boldsymbol{B}_2\right)u. \end{cases}$$

$G_{\rm r}$=`modred(`G`,elim)`

其他降阶模型计算

- MATLAB求解函数
  - Schur降阶

$$G_{\mathrm{r}}=\texttt{schmr}(G,1,k)$$

  - 最优Hankel范数降阶

$$G_{\mathrm{r}}=\texttt{ohklmr}(G,1,k)$$

- 控制系统工具箱函数

例4-34 状态方程降阶举例

- 原始模型

$$G(s)=\frac{1+8.8818s+29.9339s^2+67.087s^3+80.3787s^4+68.6131s^5}{1+7.6194s+21.7611s^2+28.4472s^3+16.5609s^4+3.5338s^5+0.0462s^6}$$

- 降阶模型与比较

```
>> num=[68.6131,80.3787,67.087,29.9339,8.8818,1];
   den=[0.0462,3.5338,16.5609,28.4472,21.7611,7.6194,1];
   G=ss(tf(num,den));
   G1=zpk(schmr(G,1,3)), G2=zpk(ohklmr(G,1,3))
   step(G,G1,'--',G2,':')
```

```
>> bode(G,G1,'--',G2,':')
```

模型降阶方法小结

- 给出了模型降阶的思想
  - 介绍了Pade降阶思想与函数 pademod
  - 介绍了Routh降阶函数 routhmod
- 介绍了鲁棒控制工具箱中的降阶函数
  - 均衡实现的降阶函数 modred
  - Schur降阶算法 schmr
  - 最优Hankel范数降阶算法 ohklmr
- 需要探讨更好的算法，包括可处理延迟的方法

薛定宇教授主讲

控制系统仿真与CAD