



第5章 辨识的最小二乘法

5.5 广义最小二乘法

5.6 增广最小二乘法

作者：李鹏波、胡德文

单位：国防科技大学机电工程与自动化学院

Email: gaoxia73@163.com

中国水利水电出版社版权所有



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

Home Page

Title Page

◀◀ ▶▶

◀ ▶

Page 1 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

第5章 辨识的最小二乘法

5.5 广义最小二乘法

1. GLS的松弛算法

2. 递推算法

5.6 增广最小二乘法

Home Page

Title Page



Page 2 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法
1. 广义最小二乘...
2. 递推算法(RGLS)
5.6 增广最小二乘法

5.5 广义最小二乘法

我们在讨论最小二乘估计的统计性质时，发现了当系统的噪声满足白噪声的性质时，参数估计值是无偏一致的最小方差估计。但一般情况下，系统的噪声都不是白噪声，为了获得好的估计效果，我们考虑广义最小二乘法、扩展最小二乘法、辅助变量法和相关二步法等参数辨识方法。每种辨识方法都对应着一种特定的噪声模型结构。

广义最小二乘法(Generalized Least Squares, GLS)的基本思想在于对数据先进行一次白化滤波处理，然后利用基本的最小二乘法对滤波后的数据进行辨识。如滤波模型选得合适，对数据进行了较好的白化处理，则直接利用基本的最小二乘法就能得到无偏一致估计。所用的滤波模型实际上是一种动态模型，经过几次迭代调整后，便可对数据进行较好的白化处理。

[Home Page](#)

[Title Page](#)

[◀](#) [▶](#)

[◀](#) [▶](#)

Page 3 of 20

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

设SISO系统的数学模型为

$$A(z^{-1})y(k) = B(z^{-1})u(k) + e(k) \quad (1)$$

假定 $\{e(k)\}$ 为具有有理谱密度的平稳随机序列，对 $\{e(k)\}$ 总可以表示为一个以白噪声序列 $\{\varepsilon(k)\}$ 为输入的线性系统的输出，也即它满足自回归模型。

$$C(z^{-1})e(k) = \varepsilon(k) \quad (2)$$

其中

$$C(z^{-1}) = 1 + c_1 z^{-1} + \dots + c_m z^{-m}$$

若假定模型的阶次 n, m 已经确定，则这类问题的辨识可用广义最小二乘法，以便获得无偏一致估计。

Home Page

Title Page



Page 4 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

令

$$\begin{cases} y_f(k) = C(z^{-1})y(k) \\ u_f(k) = C(z^{-1})u(k) \end{cases} \quad (3)$$

分别为白化处理后的输出和白化处理后的输入。

则式(1)可写为

$$A(z^{-1})y_f(k) = B(z^{-1})u_f(k) + e(k) \quad (4)$$

上式与基本最小二乘法的模型是一样的，若 $C(z^{-1})$ 已知，这样可估计出 \hat{a}_i 和 \hat{b}_i 。若 $C(z^{-1})$ 未知，通常用“松弛算法”来估计参数。

Home Page

Title Page

◀ ▶

◀ ▶

Page 5 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

1. 广义最小二乘法(GLS)的松弛算法

(1) 先猜一个 $C(z^{-1})$ 值, 即设 $C(z^{-1}) = 1$, 利用基本的最小二乘法对式(4)中的 $A(z^{-1})$ 和 $B(z^{-1})$ 进行估计, 得到 $\hat{A}(z^{-1})$ 和 $\hat{B}(z^{-1})$;

(2) 利用式(1)计算 $e(k)$,

$$e(k) = \hat{A}(z^{-1})y(k) - \hat{B}(z^{-1})u(k) \quad (5)$$

(3) 利用式(2)计算 $C(z^{-1})$,

$$C(z^{-1})e(k) = \varepsilon(k) \quad (6)$$

即由最小二乘法估计 $\hat{c}_1, \dots, \hat{c}_m$.

(4) 利用 $C(z^{-1})$ 可计算 $y_f(k)$ 和 $u_f(k)$, 通过式(4)计算 $A(z^{-1})$ 和 $B(z^{-1})$ 新的估计值 $\hat{A}(z^{-1})$ 和 $\hat{B}(z^{-1})$;

(5) 重复步骤(2), 直到估计的精度满足要求为止。

Home Page

Title Page



Page 6 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

所谓精度满足要求，可以看下列不等式是否满足：

$$\max_i \left| \frac{\hat{\theta}_i(k+1) - \hat{\theta}_i(k)}{\hat{\theta}_i(k+1)} \right| < \varepsilon \quad (7)$$

式中 $\hat{\theta}_i(k+1)$ 为参数向量 θ 的第 i 个元素在 $k+1$ 次迭代计算结果。 ε 为给定的表示精度要求的某一正数。

广义最小二乘法就是根据上述过程，反复估计噪声模型参数 $C(z^{-1})$ 与系统模型参数 $A(z^{-1}), B(z^{-1})$ 的一种迭代算法。广义最小二乘法的收敛速度是比较慢的，需要经过多次迭代计算，才能得到较准确的参数估计值。一般情况下，经过多次迭代后，估计值便会收敛到稳态值。但在某些情况下(如信噪比较低时)存在局部极小值，估计值不一定收敛到准则函数的全局极小值上。

Home Page

Title Page

◀ ▶

◀ ▶

Page 7 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

2. 递推算法(RGLS)

为了进行在线辨识, 可采用递推广义最小二乘法。

设前一时刻算出的系统模型参数估计为 $\hat{\theta}_k$, 噪声模型参数估计为 \hat{c}_k (相应的算子多项式为 $\hat{C}_k(z^{-1})$, $\hat{c}_k = [\hat{c}_k(1) \ \hat{c}_k(2) \ \cdots \ \hat{c}_k(m)]^T$), 现时刻采得数据为 y_{k+1}, u_{k+1} 。则一次递推过程的步骤如下

(1) 对新数据进行滤波, 即

$$\begin{cases} y_{k+1}^{(f)} = C_k(z^{-1})y_{k+1} \\ u_{k+1}^{(f)} = C_k(z^{-1})u_{k+1} \end{cases} \quad (8)$$

并构成滤波后的输入输出数据向量

$$\varphi_{k+1}^{(f)} = [y_k^{(f)} \ \cdots \ y_{k-n+1}^{(f)} \ u_k^{(f)} \ \cdots \ u_{k-n+1}^{(f)}]^T$$

Home Page

Title Page

◀ ▶

◀ ▶

Page 8 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

Home Page

Title Page

◀◀ ▶▶

◀ ▶

Page 9 of 20

Go Back

Full Screen

Close

Quit

(2) 对滤波后的输入输出数据作RLS估计，修正系统模型参数，计算公式为

$$\hat{\theta}_{k+1} = \hat{\theta}_k + G_{k+1}^{(f)} [y_{k+1}^{(f)} - \varphi_{k+1}^{(f)T} \hat{\theta}_k] \quad (9)$$

$$G_{k+1}^{(f)} = \frac{P_k^{(f)} \varphi_{k+1}^{(f)}}{1 + \varphi_{k+1}^{(f)T} P_k^{(f)} \varphi_{k+1}^{(f)}} \quad (10)$$

$$P_{k+1}^{(f)} = [I - G_{k+1}^{(f)} \varphi_{k+1}^{(f)}] P_k^{(f)} \quad (11)$$

(3) 由新得到的 $\hat{\theta}_{k+1}$ 计算出新的残差估计值

$$\hat{e}_{k+1} = y_{k+1} - \varphi_{k+1}^T \hat{\theta}_{k+1} \quad (12)$$

式中

$$\varphi_{k+1} = [y_k \cdots y_{k-n+1} \ u_k \cdots u_{k-n+1}]^T$$



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

进而构成残差数据向量

$$\varphi_{k+1}^{(e)} = [-\hat{e}_k \ -\hat{e}_{k-1} \ \cdots \ -\hat{e}_{k-m+1}]^T$$

(4) 对新残差数据作RLS算法估计，修正噪声模型参数 \hat{c}_k ，算法为

$$\hat{c}_{k+1} = \hat{c}_k + G_{k+1}^{(e)} [\hat{e}_{k+1} - \varphi_{k+1}^{(e)T} \hat{c}_k] \quad (13)$$

$$G_{k+1}^{(e)} = \frac{P_k^{(e)} \varphi_{k+1}^{(e)}}{1 + \varphi_{k+1}^{(e)T} P_k^{(e)} \varphi_{k+1}^{(e)}} \quad (14)$$

$$P_{k+1}^{(e)} = [I - G_{k+1}^{(e)} \varphi_{k+1}^{(e)}] P_k^{(e)} \quad (15)$$

Home Page

Title Page

◀ ▶

◀ ▶

Page 10 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

要特别注意 $\varphi_{k+1}^{(f)}$ 与 $\varphi_{k+1}, \varphi_{k+1}^{(e)}$ 的区别。每一次递推过程都包括两次RLS的计算，通过滤波计算和残差计算将它们联系起来。

递推过程的初值和停机条件，可参照一般RLS算法。为了防止参数估计值收敛到局部极小值，最好选定初值接近最优解，一般可用最小二乘法的批处理估计值作为初值。如果系统是时变的，或为了克服数据饱和现象，可在两次RLS算法中分别引进遗忘因子。

[Home Page](#)[Title Page](#)[◀◀](#) [▶▶](#)[◀](#) [▶](#)[Page 11 of 20](#)[Go Back](#)[Full Screen](#)[Close](#)[Quit](#)



5.6 增广最小二乘法

增广最小二乘法(Extended Least Squares, ELS)是最小二乘法的一种简单推广，它只是扩充了参数向量 θ 和数据向量 $\varphi(k)$ 的维数，在辨识过程中同时考虑了噪声模型的参数。

设SISO系统采用的数学模型为

$$A(z^{-1})y(k) = B(z^{-1})u(k) + D(z^{-1})\varepsilon(k) \quad (16)$$

式中 $D(z^{-1}) = 1 + d_1z^{-1} + \dots + d_mz^{-m}$.

噪声 $e(k)$ 为MA模型，称 $\{\varepsilon(k)\}$ 为新息序列。即在给定的输出序列 $\{y(k)\}$ 和输入序列 $\{u(k)\}$ 的条件下， $\varepsilon(k)$ 的条件均值为零，把具有这种性质的随机序列称为新息序列。

如果 $\{\varepsilon(k)\}$ 是可量测的，则模型式(16)可写为

$$y(k) = \varphi_k^T \theta + \varepsilon(k) \quad (17)$$

其中

$$\begin{aligned} \varphi_k &= [-y(k-1) \cdots -y(k-n) \ u(k-1) \cdots u(k-n) \ \varepsilon(k-1) \cdots \varepsilon(k-m)]^T \\ \theta &= [a_1 \ a_2 \ \cdots \ a_n \ b_0 \ b_1 \ \cdots \ b_n \ d_1 \ \cdots \ d_m]^T \end{aligned}$$



5.5 广义最小二乘法

1. 广义最小二乘...
2. 递推算法(RGLS)
- 5.6 增广最小二乘法

这时，式(17)可用最小二乘法求参数的估计量。可是，式中 φ_k 的分量 $\varepsilon(k-1), \varepsilon(k-2), \dots$ 是未知的。为了克服这个困难，一个很自然的方法是用 $\hat{\varepsilon}(k)$ 代替 $\varepsilon(k)$ ，借助于过去已知的参数估计量 $\hat{\theta}_{k-1}$ 来计算出误差估计 $\hat{\varepsilon}(k)$ ，即

$$\hat{\varepsilon}(k) = y(k) - \hat{\varphi}_k^T \hat{\theta}_{k-1} \quad (18)$$

其中

$$\hat{\varphi}_k = [-y(k-1) \cdots -y(k-n) \ u(k-1) \cdots u(k-n) \ \hat{\varepsilon}(k-1) \cdots \hat{\varepsilon}(k-m)]^T$$

这样，可采用迭代算法。

迭代初值，取 $\hat{\varepsilon}(k-1) = \cdots = \hat{\varepsilon}(k-m) = 0$ 。

[Home Page](#)[Title Page](#)[◀◀](#) [▶▶](#)[◀](#) [▶](#)[Page 13 of 20](#)[Go Back](#)[Full Screen](#)[Close](#)[Quit](#)



以上介绍了ELS算法。按照递推最小二乘法公式的推导方法，可得到如下的递推增广最小二乘法(RELS)算法。

$$\hat{\theta}_{k+1} = \hat{\theta}_k + G_{k+1}[y_{k+1} - \hat{\varphi}_{k+1}^T \hat{\theta}_k] \quad (19)$$

$$G_{k+1} = \frac{P_k \hat{\varphi}_{k+1}}{1 + \hat{\varphi}_{k+1}^T P_k \hat{\varphi}_{k+1}} \quad (20)$$

$$P_{k+1} = [I - G_{k+1} \hat{\varphi}_{k+1}^T] P_k \quad (21)$$

可见，递推增广最小二乘法的算法与RLS的形式是一致的，只是参数向量 θ 与数据向量 φ_k 的维数扩充了 m 维。

5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

Home Page

Title Page

◀ ▶

◀ ▶

Page 14 of 20

Go Back

Full Screen

Close

Quit



增广最小二乘法的估计精度虽比广义最小二乘法低一些，但一般情况下具有一致无偏性，且算法简单，因而在实际中得到了广泛的应用。下面列出带遗忘因子的RELS算法。

$$G_{k+1} = \frac{P_k \hat{\varphi}_{k+1}}{\rho + \hat{\varphi}_{k+1}^T P_k \hat{\varphi}_{k+1}} \quad (22)$$

$$\hat{\theta}_{k+1} = \hat{\theta}_k + G_{k+1} [y_{k+1} - \hat{\varphi}_{k+1}^T \hat{\theta}_k] \quad (23)$$

$$P_{k+1} = \frac{1}{\rho} [I - G_{k+1} \hat{\varphi}_{k+1}^T] P_k \quad (24)$$

$$\hat{\varepsilon}_{k+1} = y_{k+1} - \hat{\varphi}_{k+1}^T \hat{\theta}_{k+1} \quad (25)$$

5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

Home Page

Title Page

◀ ▶

◀ ▶

Page 15 of 20

Go Back

Full Screen

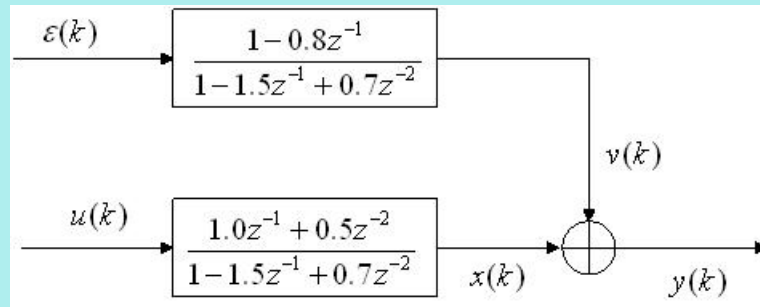
Close

Quit



这里举一个应用RELS算法进行参数估计的仿真例子，并与RLS算法的结果作比较。

例5.1 考虑如图5.1所示的仿真对象。



图中 $\varepsilon(k)$ 是均值为0，方差为1的正态分布近似白噪声序列，由计算机随机数产生。 $u(k)$ 为输入序列，采用m序列，幅度为2。噪声强度系数取为0.9，模型结构选用如下形式

$$y(k) + a_1 y(k-1) + a_2 y(k-2) = b_1 u(k-1) + b_2 u(k-2) + \varepsilon(k) + d_1 \varepsilon(k-1)$$

初始条件取 $P_0 = 10^4 I$, $\hat{\theta}_0 = 10^{-3}$ 。利用RELS与RLS算法估计出的参数如表所示。可见，在 $e(k) = \varepsilon(k) - 0.8\varepsilon(k-1)$ 为有色噪声情况下，RELS算法的估计值比RLS算法的估计精度有所提高。

5.5 广义最小二乘法
1. 广义最小二乘...
2. 递推算法(RGLS)
5.6 增广最小二乘法

Home Page

Title Page

◀ ▶

◀ ▶

Page 16 of 20

Go Back

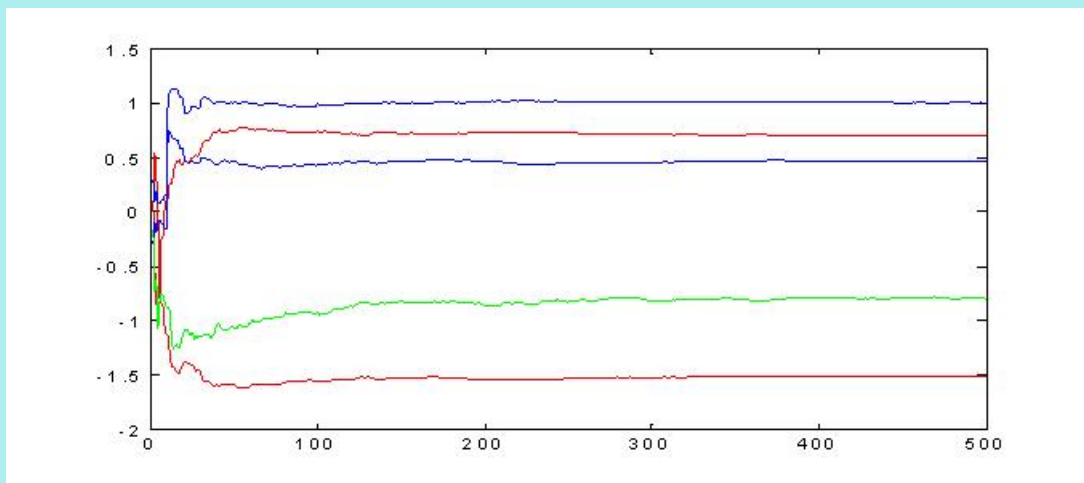
Full Screen

Close

Quit

表5.1 不同噪声模型下RLS和RELS算法的辨识结果

	参数	a_1	a_2	b_1	b_2	d_1
白噪声	真值	-1.5	0.7	1.0	0.5	0
	RLS	-1.4996	0.6955	0.9975	0.5020	
	RELS	-1.4978	0.6945	0.9977	0.5028	0.0109
有色噪声	真值	-1.5	0.7	1.0	0.5	-0.8
	RLS	-1.4666	0.6716	0.9599	0.5526	
	RELS	-1.5148	0.7136	0.9766	0.4872	-0.8807



各参数估计值在递推过程中的收敛情况如图5.2所示。可以看出，各参数估计值在150步左右已收敛。



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

Home Page

Title Page

◀ ▶

◀ ▶

Page 17 of 20

Go Back

Full Screen

Close

Quit



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

例5.1的源程序example5_1.m为

```
clear all; hold off; N=1000;
U = idinput(N,'prbs',[0 1])*2; % 输入伪随机信号
E = NORMRND(0,0.9,N,1); % 随机噪声
ek=zeros(N, 1); y=zeros(N, 1); % 变量初始化
for i=3:N
    ek(i)=[E(i) ek(i-1)]*[1 -0.8]'; % 模拟观测噪声
    y(i)=[-y(i-1) -y(i-2) U(i-1) U(i-2)]*[-1.5 0.7 1.0 0.5]'+ek(i); % 模拟观测值
end
z=iddata(y, U); % 输入输出数据组
Thita=rels(z, 2, 2, 2); % 调用rels函数
x=1:N; plot(x,Thita(1,x),'r'); % 作图显示参数递推值
hold on; plot(x,Thita(2,x),'r');
plot(x,Thita(3,x),'b'); plot(x,Thita(4,x),'b');
plot(x,Thita(5,x),'g'); hold off;
```

Home Page

Title Page

◀ ▶

◀ ▶

Page 18 of 20

Go Back

Full Screen

Close

Quit

用户开发的增广最小二乘法函数rels.m的源程序为

```
function thita=rels(z, na, nb, nc)
%用增广矩阵法估计参数...
if isa(z,'iddata')
    y = pvget(z,'OutputData');
    u = pvget(z,'InputData'); z = [y1,u1];
end
[nz,ns]=size(z); nn=na+nb+nc;
thitak=ones(nn,1)*0.001;
thita=zeros(nn,nz);
p1=eye(nn,nn)*(1.0e6); p2=zeros(nn,nn);
K=zeros(nn,1); e=zeros(nz,1); I=eye(nn,nn);
for i=na+1:nz
    Q=[[-z(i-1:-1:i-na, 1)]',[z(i-1:-1:i-nb, 2)]',[e(i-1:-1:i-nc, 1)]]';
    K=p1*Q'/(Q*p1*Q'+1);
    p2=(I-K*Q)*p1;
    thita(:,i)=thitak+K*(z(i,1)-Q*thitak);
    p1=p2;
    thitak=thita(:,i);
    e(i)=z(i,1)-Q*thitak;
end
```

%函数的注释体
% 输入输出数据

% 数据维数与阶次
% 给定初始条件

%数据向量
%增益因子
%协方差阵
%参数估计值

%预报误差



5.5 广义最小二乘法
1. 广义最小二乘...
2. 递推算法(RGLS)
5.6 增广最小二乘法

[Home Page](#)

[Title Page](#)



Page 19 of 20

[Go Back](#)

[Full Screen](#)

[Close](#)

[Quit](#)



5.5 广义最小二乘法

1. 广义最小二乘...

2. 递推算法(RGLS)

5.6 增广最小二乘法

请采用4阶逆重复m序列作为输入序列，RELS算法中遗忘因子取0.99，仿照本例给出的源程序进行仿真。

Home Page

Title Page



Page 20 of 20

Go Back

Full Screen

Close

Quit