第7卷 第4期　　自 动 化 学 报　　Vol. 7, No. 4
1981年10月　　ACTA AUTOMATICA SINICA　　Oct., 1981

多变量线性系统的递推辨识算法

王秀峰　　卢桂章
（南 开 大 学）

摘　　要

多变量系统的结构辨识是很麻烦的，Guidorzi[1] 提出了一个辨识不变性指标的方法，但必须用矩阵求逆. 本文给出一个递推确定结构指标的算法，不需要求行列式的运算，并且一旦确定了子系统结构，随即得到参数估计值，没有重复的运算. 所需计算量比 Guidorzi 方法[1] 大大减少.

一、引　　言

近几年来多变量线性系统的辨识工作有了很大进展. 从各种角度探讨结构及参数的辨识[1-3]，以 Guidorzi 的结果受到了普遍重视，它给出直接由输入、输出数据辨识系统标准形结构的方法，需要辨识的参数最少. 但这种方法需要反复求矩阵的行列式值，使得计算量相当大. 文献[4,5]对参数辨识导出了递推算法，推广到有色噪声的辨识. 文献[6]用随机逼近的方法进行了讨论. 但这些文献大多是在假定已知系统的结构或假定结构指标相等的条件下，对参数进行估计的一些算法.

本文给出递推确定结构指标的算法，不需求矩阵的行列式值，子系统的结构一旦确定，即可得到参数的估计值.

二、多变量线性系统的输入、输出描述

考虑 r 个输入、m 个输出的多变量线性系统，在可观条件下，同一系统的任何最小实现都等价于 Luenberger 规范形式[1].

$$\begin{aligned} \boldsymbol{x}(k+1) &= A\boldsymbol{x}(k) + B\boldsymbol{u}(k) \\ \boldsymbol{y}(k) &= C\boldsymbol{x}(k). \end{aligned} \tag{1}$$

其中 $\boldsymbol{x}(k)$ 为 n 维状态向量；$\boldsymbol{u}(k)$ 为 r 维输入向量；$\boldsymbol{y}(k)$ 为 m 维输出向量. A, B, C 为相应维数的常数矩阵，其形式参见[1]. 此规范形又完全等价于如下的输入、输出差分方程[1].

$$P(z)\boldsymbol{y}(k) = Q(z)\boldsymbol{u}(k) \tag{2}$$

其中

本文于1980年9月8日收到.

$$P(z)=\begin{pmatrix} p_{11}(z)\cdots p_{1m}(z) \\ \cdots\cdots\cdots\cdots \\ p_{m1}(z)\cdots p_{mm}(z) \end{pmatrix} \tag{3}$$

$$Q(z)=\begin{pmatrix} q_{11}(z)\cdots q_{1r}(z) \\ \cdots\cdots\cdots\cdots \\ q_{m1}(z)\cdots q_{mr}(z) \end{pmatrix} \tag{4}$$

$$p_{ii}(z)=z^{\nu_i}-a_{ii,\nu_i}z^{\nu_i-1}-\cdots-a_{ii,2}z-a_{ii,1} \tag{5}$$

$$p_{ij}(z)=-a_{ij,\nu_{ij}}z^{\nu_{ij}-1}-\cdots-a_{ij,2}z-a_{ij,1} \tag{6}$$

$$q_{ij}(z)=\beta_{(\nu_1+\cdots+\nu_i),j}z^{\nu_i-1}+\cdots+\beta_{(\nu_1+\cdots+\nu_{i-1}+2),j}z+\beta_{(\nu_1+\cdots+\nu_{i-1}+1),j}. \tag{7}$$

$\nu_1, \nu_2, \cdots, \nu_m$ 称为结构指标. ν_i, ν_{ij} 及参数 $a_{ij,k}$ 为一组不变量[7]. 结构指标和系统的阶满足关系式：

$$\sum_{i=1}^{m}\nu_i=n \tag{8}$$

和

$$\begin{aligned} &\nu_{ij}\leqslant\nu_i+1, \quad j<i \\ &\nu_{ij}\leqslant\nu_i. \qquad\quad j\geqslant i \end{aligned} \tag{9}$$

三、Guidorzi 方法简述

由于多变量线性系统规范形的结构完全由结构指标 $\nu_1, \cdots, \nu_m$ 唯一决定,因此,"结构辨识"问题就是决定这些结构指标.

由于规范形 (1) 等价于输入输出描述式 (2), 因此可以由式 (2) 出发, 利用输入、输出数据来确定 $\nu_1, \nu_2, \cdots, \nu_m$. 为此, 将 (2) 式按输出分解为 m 个等式 (每个等式对应于 $P(z)$, $Q(z)$ 的行,可做为多输入单输出的子系统),第 s 个等式为：

$$\sum_{i=1}^{m}p_{si}(z)y_i(k)=\sum_{i=1}^{r}q_{si}(z)u_i(k) \tag{10}$$

其中 $p_{si}(z)$, $q_{si}(z)$ 分别为多项式矩阵 $P(z)$, $Q(z)$ 的第 s 行第 i 列的元素, 如 (5), (6), (7) 式所示. 由此,(10)式可表示为：

$$y_s(k+\nu_s)=\sum_{i=1}^{m}\sum_{j=1}^{\nu_{si}}a_{si,j}y_i(k+j-1)+\sum_{i=1}^{r}\sum_{j=1}^{\nu_s}\beta_{(\nu_1+\cdots+\nu_{s-1}+j),i}u_i(k+j-1) \tag{11}$$

其中 $\nu_{ss}\triangleq\nu_s$.

假如,取输入、输出数据: $u(k), u(k+1), \cdots, u(k+N+n), y(k), \cdots, y(k+N+n)$, 排成如下形式

$$\left[\begin{array}{cc|c|cc|} y_1(k) & y_1(k+1)\cdots & & y_m(k) & y_m(k+1)\cdots \\ y_1(k+1) & y_1(k+2)\cdots & \cdots & y_m(k+1) & y_m(k+2)\cdots \\ \vdots & \vdots & & \vdots & \vdots \\ y_1(k+N) & y_1(k+N+1)\cdots & & y_m(k+N) & y_m(k+N+1)\cdots \end{array}\right.$$

$$
\left[\left|\begin{matrix} u_1(k) & u_1(k+1)\cdots \\ u_1(k+1) & u_1(k+2)\cdots \\ \vdots & \vdots \\ u_1(k+N) & u_1(k+N+1)\cdots \end{matrix}\right| \cdots \left|\begin{matrix} u_r(k) & u_r(k+1)\cdots \\ u_r(k+1) & u_r(k+2)\cdots \\ \vdots & \vdots \\ u_r(k+N) & u_r(k+N+1)\cdots \end{matrix}\right|\right]
$$

$$
\begin{aligned}
&\triangleq [\,|\bar{\boldsymbol{y}}_1(k)\bar{\boldsymbol{y}}_1(k+1)\cdots|\cdots|\bar{\boldsymbol{y}}_m(k)\bar{\boldsymbol{y}}_m(k+1)\cdots \\
&\quad |\bar{\boldsymbol{u}}_1(k)\bar{\boldsymbol{u}}_1(k+1)\cdots|\cdots|\bar{\boldsymbol{u}}_r(k)\bar{\boldsymbol{u}}_r(k+1)\cdots|\,]
\end{aligned} \tag{12}
$$

则 (11) 式就意味着

$$
\bar{\boldsymbol{y}}_s(k+\nu_s)=\begin{pmatrix} y_s(k+\nu_s) \\ y_s(k+\nu_s+1) \\ \vdots \\ y_s(k+\nu_s+N) \end{pmatrix} \text{是}
$$

$$
\begin{aligned}
&\bar{\boldsymbol{y}}_1(k+\nu_s),\ \cdots,\ \bar{\boldsymbol{y}}_s(k+\nu_s-1),\ \cdots,\ \bar{\boldsymbol{y}}_m(k+\nu_s-1),\cdots,\bar{\boldsymbol{y}}_1(k),\cdots,\bar{\boldsymbol{y}}_m(k) \\
&\bar{\boldsymbol{u}}_1(k+\nu_s-1),\ \cdots,\ \bar{\boldsymbol{u}}_r(k+\nu_s-1),\ \cdots,\ \bar{\boldsymbol{u}}_1(k),\ \cdots,\ \bar{\boldsymbol{u}}_r(k)
\end{aligned}
$$

的线性组合. 因此,可以利用这个关系,通过适当方式,选择输入、输出 $\bar{\boldsymbol{y}}_i(\cdot)$, $\bar{\boldsymbol{u}}_i(\cdot)$, 就可以确定出结构指标 ν_i, $i=1,2,\cdots,m$.

在 (12) 式中按如下顺序选择向量:

$$
\begin{aligned}
&\bar{\boldsymbol{y}}_1(k)\bar{\boldsymbol{y}}_2(k)\cdots\bar{\boldsymbol{y}}_m(k)\ \bar{\boldsymbol{u}}_1(k)\bar{\boldsymbol{u}}_2(k)\cdots\bar{\boldsymbol{u}}_r(k) \\
&\bar{\boldsymbol{y}}_1(k+1)\bar{\boldsymbol{y}}_2(k+1)\cdots\bar{\boldsymbol{y}}_m(k+1)\bar{\boldsymbol{u}}_1(k+1)\,\bar{\boldsymbol{u}}_2(k+1) \\
&\cdots\ \bar{\boldsymbol{u}}_r(k+1)\bar{\boldsymbol{y}}_1(k+2)\cdots
\end{aligned} \tag{13}
$$

当找到某向量 $\bar{\boldsymbol{y}}_s(k+\nu_s)$ 与前面所选出的向量线性相关时 (这时 $\bar{\boldsymbol{y}}_s(k+\nu_s)$ 所在的子矩阵中其余向量也与所选出的向量线性相关), 就得到了 ν_s. 当关于输出的每个子矩阵中都找到这样的向量时, 选择结束. 从每个子矩阵中所选出的向量个数就是所求的 ν_1, ν_2, $\cdots$, ν_m.

注意,为了保证选择无关向量, N 必须充分大, $N>n+r\nu_M$, ($\nu_M=\max(\nu_i)$).

为了判断向量的相关性, 引入记号 $R(\delta_1,\cdots,\delta_{m+r})$, 表示从 (12) 式中第 i 个子矩阵中取出前 $\delta_i(i=1,2,\cdots,m+r)$ 列向量构成的矩阵. 即

$$
\begin{aligned}
R(\delta_1,\cdots,\delta_m,\delta_{m+1},\cdots,\delta_{m+r})=[&\bar{\boldsymbol{y}}_1(k)\cdots\bar{\boldsymbol{y}}_1(k+\delta_1-1)\cdots\bar{\boldsymbol{y}}_m(k)\cdots\bar{\boldsymbol{y}}_m(k \\
&+\delta_m-1)\bar{\boldsymbol{u}}_1(k)\cdots\bar{\boldsymbol{u}}_1(k+\delta_{m+1}-1)\cdots\bar{\boldsymbol{u}}_r(k)\cdots\bar{\boldsymbol{u}}_r(k+\delta_{m+r}-1)]
\end{aligned} \tag{14}
$$

由于 R 是长方矩阵, 判断其列的相关性比较麻烦. 利用对任何实矩阵 D, 秩 $(D)=$ 秩 (D^TD) 这一事实,给出一个较方便的算法.

定义.
$$
S(\delta_1,\cdots,\delta_{m+r})=R^T(\delta_1,\cdots,\delta_{m+r})R(\delta_1,\cdots,\delta_{m+r}) \tag{15}
$$

Guidorzi 的方法是依次考查

$$
\begin{aligned}
&S(1,\cdots,1),\ S(2,1,\cdots,1),\ S(2,2,1,\cdots,1),\ S(2,2,\cdots,2), \\
&S(3,2,\cdots,2),\ \cdots
\end{aligned}
$$

是否满秩,当发现 $S(\delta_1,\delta_2,\cdots,\delta_i,\delta_{i+1},\cdots,\delta_{r+m})$ 非奇异, 而 $S(\delta_1,\cdots,\delta_i+1,\delta_{i+1},\cdots,\delta_{r+m})$ $(i\leqslant m)$ 奇异时, 则可断言 $\nu_i=\delta_i$, 这时 $\delta_1,\cdots,\delta_{i-1},\delta_{i+1},\cdots,\delta_m$ 分别为 $\nu_{i1},\cdots,\nu_{i,i-1},\nu_{i,i+1},\cdots,\nu_{im}$; 一直做下去,就可找到全部 $\nu_i(i=1,2,\cdots,m)$. 在此基础上可进行参数辨识:

$$\hat{\boldsymbol{\theta}}_i = S_i^{-1} R_i^T \bar{\boldsymbol{y}}_i(k + \nu_i) \tag{16}$$

其中

$$\begin{aligned}\hat{\boldsymbol{\theta}}_i^T = [\,|a_{i1,1}\cdots a_{i1,\nu_{i1}}|\cdots|a_{ii,1}\cdots a_{ii,\nu_i}|\cdots|a_{im,1}\cdots a_{im,\nu_{im}}|\\ |\beta_{(\nu_1+\cdots+\nu_{i-1}+1),1}\cdots\beta_{(\nu_1+\cdots+\nu_i),1}|\cdots|\beta_{(\nu_1+\cdots+\nu_{i-1}+1),r}\cdots\beta_{(\nu_1+\cdots+\nu_i),r}|\,]\end{aligned} \tag{17}$$

$$R_i = R(\nu_{i1}, \cdots, \nu_{i,i-1}, \nu_i, \nu_{i,i+1}, \cdots, \nu_{im}, \delta_{m+1}\cdots\delta_{m+r}) \tag{18}$$

$$S_i = R_i^T R_i$$

由此可见,用此方法,每确定一个子系统的结构指标,就要多次判断矩阵的奇异性.

四、多变量线性系统辨识的递推算法

1. 递推算法

重新构造矩阵 R, 令 $R^0(\delta_1, \cdots, \delta_{m+r})$ 表示按(13)式的顺序选出且按先后次序排列的矩阵,(注意: 这里的排列次序与 Guidorzi 方法排列次序不同) δ_i 表示从(12)中第 i 个子矩阵中选出了 δ_i 个列向量, $i = 1, 2, \cdots, m + r$. 例如:

$$R^0(1, 1, \cdots, 1) = [\bar{\boldsymbol{y}}_1(k)\bar{\boldsymbol{y}}_2(k)\cdots\bar{\boldsymbol{y}}_m(k)\bar{\boldsymbol{u}}_1(k)\cdots\bar{\boldsymbol{u}}_r(k)]$$

$$\begin{aligned}R^0(2, 1, \cdots, 1) &= [\bar{\boldsymbol{y}}_1(k)\bar{\boldsymbol{y}}_2(k)\cdots\bar{\boldsymbol{y}}_m(k)\bar{\boldsymbol{u}}_1(k)\cdots\bar{\boldsymbol{u}}_r(k)\bar{\boldsymbol{y}}_1(k+1)]\\ &= [R^0(1, \cdots, 1)\ \bar{\boldsymbol{y}}_1(k+1)]\end{aligned}$$

....

一般地

$$\begin{aligned}&R^0(\mu_1, \cdots, \mu_i + 1, \mu_{i+1}, \cdots, \mu_{m+r})\\ &= [R^0(\mu_1, \cdots, \mu_i, \mu_{i+1}, \cdots, \mu_{m+r})\ \bar{\boldsymbol{y}}_i(k + \mu_i)]\end{aligned} \tag{19}$$

在递推算法中,为方便起见,采用下面的简化记号. 记 R_i 为递推到第 i 步时的矩阵 R^0, 即

$$R_i = R^0(\mu_1, \mu_2, \cdots, \mu_{m+r})$$

R_{i+1} 为下一步的矩阵 R^0, 即

$$R_{i+1} = [R^0(\mu_1, \mu_2, \cdots, \mu_{m+r})\ \bar{\boldsymbol{y}}_{j,\mu_j}] = [R_i\ \bar{\boldsymbol{y}}_{j,\mu_j}] \tag{20}$$

其中

$$\bar{\boldsymbol{y}}_{j,\mu_j} \triangleq \begin{cases}\bar{\boldsymbol{y}}_j(k + \mu_j), & \text{当 } j = 1, 2, \cdots, m \text{ 时}\\ \bar{\boldsymbol{u}}_j(k + \mu_j), & \text{当 } j = m + 1, \cdots, m + r \text{ 时}\end{cases}$$

注意,这里的 R_i 与文献[1]中的 R_i 排列方法不一样.

令
$$S_i = R_i^T R_i$$

则

$$\begin{aligned}S_{i+1} &= R_{i+1}^T R_{i+1} = [R_i\ \bar{\boldsymbol{y}}_{j,\mu_j}]^T [R_i\ \bar{\boldsymbol{y}}_{j,\mu_j}]\\ &= \begin{bmatrix} R_i^T R_i & R_i^T \bar{\boldsymbol{y}}_{j,\mu_j} \\ \bar{\boldsymbol{y}}_{j,\mu_j}^T R_i & \bar{\boldsymbol{y}}_{j,\mu_j}^T \bar{\boldsymbol{y}}_{j,\mu_j}\end{bmatrix}\end{aligned} \tag{21}$$

只要 $S_i = R_i^T R_i$ 满秩,即 $S_i^{-1} = (R_i^T R_i)^{-1}$ 存在,则有

$$\begin{aligned}\det S_{i+1} &= \det S_i \det [\bar{\boldsymbol{y}}_{j,\mu_j}^T \bar{\boldsymbol{y}}_{j,\mu_j} - \bar{\boldsymbol{y}}_{j,\mu_j}^T R_i S_i^{-1} R_i^T \bar{\boldsymbol{y}}_{j,\mu_j}]\\ &= \det S_i \det [\bar{\boldsymbol{y}}_{j,\mu_j}^T (I - R_i S_i^{-1} R_i^T) \bar{\boldsymbol{y}}_{j,\mu_j}]\end{aligned} \tag{22}$$

如果 $\det S_{i+1}=0$, 则结构指标 $\nu_j=\mu_j$, ν_{ji} 也相应确定, 并且随即得到第 j 个输出分量对应的子系统的参数估计,

$$\hat{\theta}_j = S_i^{-1} R_i^T \bar{\mathbf{y}}_{j,\mu_j} \tag{23}$$

$$\theta_j^T = (a_{j1,1}\cdots a_{j1,\nu_1} \vdots \beta_{(\nu_1+\cdots+\nu_{j-1}+1),1}\cdots\beta_{(\nu_1+\cdots+\nu_j),1}\vdots\cdots) \tag{24}$$

注意,这里的参数向量 θ_j 与 [1] 中的排列顺序也不同.

如果 $\det S_{i+1}\neq 0$, 则可利用下列公式递推求出 S_{i+1}^{-1}.

$$S_{i+1}^{-1}=\begin{bmatrix} R_i^TR_i & R_i^T\bar{\mathbf{y}}_{j,\mu_j} \\ \bar{\mathbf{y}}_{j,\mu_j}^TR_i & \bar{\mathbf{y}}_{j,\mu_j}^T\bar{\mathbf{y}}_{j,\mu_j}\end{bmatrix}^{-1}=\begin{bmatrix} S_i^{-1}+\mathbf{P}_2\bar{\mathbf{y}}_{j,\mu_j}^TR_iS_i^{-1} & -\mathbf{P}_2 \\ -\mathbf{P}_2^T & P_1\end{bmatrix} \tag{25}$$

其中

$$P_1^{-1}=\bar{\mathbf{y}}_{j,\mu_j}^T\bar{\mathbf{y}}_{j,\mu_j}-\bar{\mathbf{y}}_{j,\mu_j}^TR_iS_i^{-1}R_i^T\bar{\mathbf{y}}_{j,\mu_j}=\bar{\mathbf{y}}_{j,\mu_j}^T(I-R_iS_i^{-1}R_i^T)\bar{\mathbf{y}}_{j,\mu_j} \tag{26}$$

$$\mathbf{P}_2=S_i^{-1}R_i^T\bar{\mathbf{y}}_{j,\mu_j}P_1 \tag{27}$$

利用 (22), (25), (26), (27) 式就构成了系统辨识的递推算法. 从 $S_1=(R^0(1,0,\cdots,0))^TR^0(1,0,\cdots,0)$ 出发就可递推地确定全部结构指标及所有参数. 当 $\bar{\mathbf{y}}_j(k+\mu_j)$ 添加到 R^0 中使得 $\det(R^{0T}R^0)=0$ 时,就得到 $\nu_j=\mu_j-1$ 及 $\hat{\theta}_j$. 然后去掉 $\bar{\mathbf{y}}_j(k+\mu_j)$, 再将 (12) 式中其他子矩阵中的向量 $\bar{\mathbf{y}}_s(k+\mu_s)$ 添到 R^0 中,…直到全部确定 $\nu_1,\nu_2,\cdots,\nu_m$ 及 $\theta_1,\theta_2,\cdots,\theta_m$ 为止.

为了说明具体步骤,引入下列记号:

$\mathbf{y}(t,l)$ 表示 (12) 式中第 t 个子阵的第 l 列; $\{KK[i];i=1,2,\cdots,m+r\}$, $KK[j]$ 表示从第 j 子阵中取出的向量个数; $\{I_0[i];i=1,2,\cdots,m+r\}$ 是一个标志数组, $I_0[i]=1$ 时,表示不能再从第 i 个子阵中取向量(ν_i 已确定), $I_0[i]=0$ 时, 可从第 i 个子阵中取向量.

具体步骤如下:

1) 置 $t=1,l=1,KK[1]=1$, $KK[j]=0,j=2,3,\cdots,m+r,i=0,R_i=0,S_i^{-1}=0$, $I_0=0$;

2) 取向量 $\mathbf{y}(t,l)$;

3) 算 $P=\mathbf{y}^T(t,l)(I-R_iS^{-1}R_i^T)\mathbf{y}(t,l)$,

如果 $t>m$, 则转 5;

4) 如果 $|P|<\varepsilon$, 则算 $\hat{\theta}_t=S_i^{-1}R_i^T\mathbf{y}(t,l)$, 置 $I_0[t]=1$, $KK[t]\Leftarrow KK[t]-1$,转 6;

5) 做 $R_{i+1}=[R_i\ \mathbf{y}(t,l)]$, 算 $\mathbf{q}=S_i^{-1}R_i^T\mathbf{y}(t,l)$

$$S_{i+1}^{-1}=\begin{bmatrix} S_i^{-1}+\mathbf{q}\mathbf{q}^T/P & -\mathbf{q}/P \\ -\mathbf{q}^T/P & 1/P\end{bmatrix};$$

6) 如果 $\sum_{j=1}^{m} I_0[j]<m$, 则 $i\Leftarrow i+1$, 且按下面顺序考查 $I_0[K]$, $K:t+1,t+2,\cdots,t+m,1,2,\cdots,t$; 若有 $I_0[j]=0$, 则 $t\Leftarrow j$, $l\Leftarrow KK[t]\Leftarrow KK[t]+1$, 转 2; 若 $\sum_{j=1}^{m} I_0[j]=m$, 则结束.

注: 1) 这里不是判断行列式的值是否为零, 而是判断 P 的值. 因为当 R_{i+1} 满秩时

(即列不相关)，由 $R_{i+1}^T R_{i+1}$ 的正定性知 P 必大于零．当 R_i 满秩，而 R_{i+1} 不满秩时 P 必等于零．但由于数据及计算误差，P 不会严格为零，一般可取 ε 为 10^{-3}—10^{-6}，或者先取一个很小的数，例如取 $\varepsilon=10^{-10}$，如不收敛，再逐渐增大 ε，直到收敛为止，就可确定出确切的阶次．

2）由于 P 为标量，因此在整个递推过程中，不需要求行列式值和矩阵求逆运算．

计算框图如图 1 所示．

2. 数值例子

考虑单输入、二输出系统：

$$\begin{bmatrix} z^3+z^2-1 & -z+1 \\ z^2-1 & z^2+2z-1 \end{bmatrix}\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix}=\begin{bmatrix} 1 \\ 1 \end{bmatrix}u(k)$$

取得的数据如表 1 所示，

用 Guidorzi[1] 的方法，首先依据以上数据构造矩阵序列（取 $N=9$）：$S(2,1,1)$，$S(2,2,1)$，$S(2,2,2)$，$S(3,2,2)$，…．然后，依次判别以上矩阵是否满秩，经过五次判别（实际上是求其行列式值），得到第一个奇异矩阵为 $S(3,3,2)$，所以 $\nu_2=3-1=2$．在此基础上继续判别（这时 $S(\cdot,\cdot,\cdot)$ 中第二个标号应固定为 2)，$S(3,2,3)S(4,2,3)$ 得到第二个奇异矩阵为 $S(4,2,3)$，从而 $\nu_1=4-1=3$．结构确定后，再用最小二乘法确定系数．

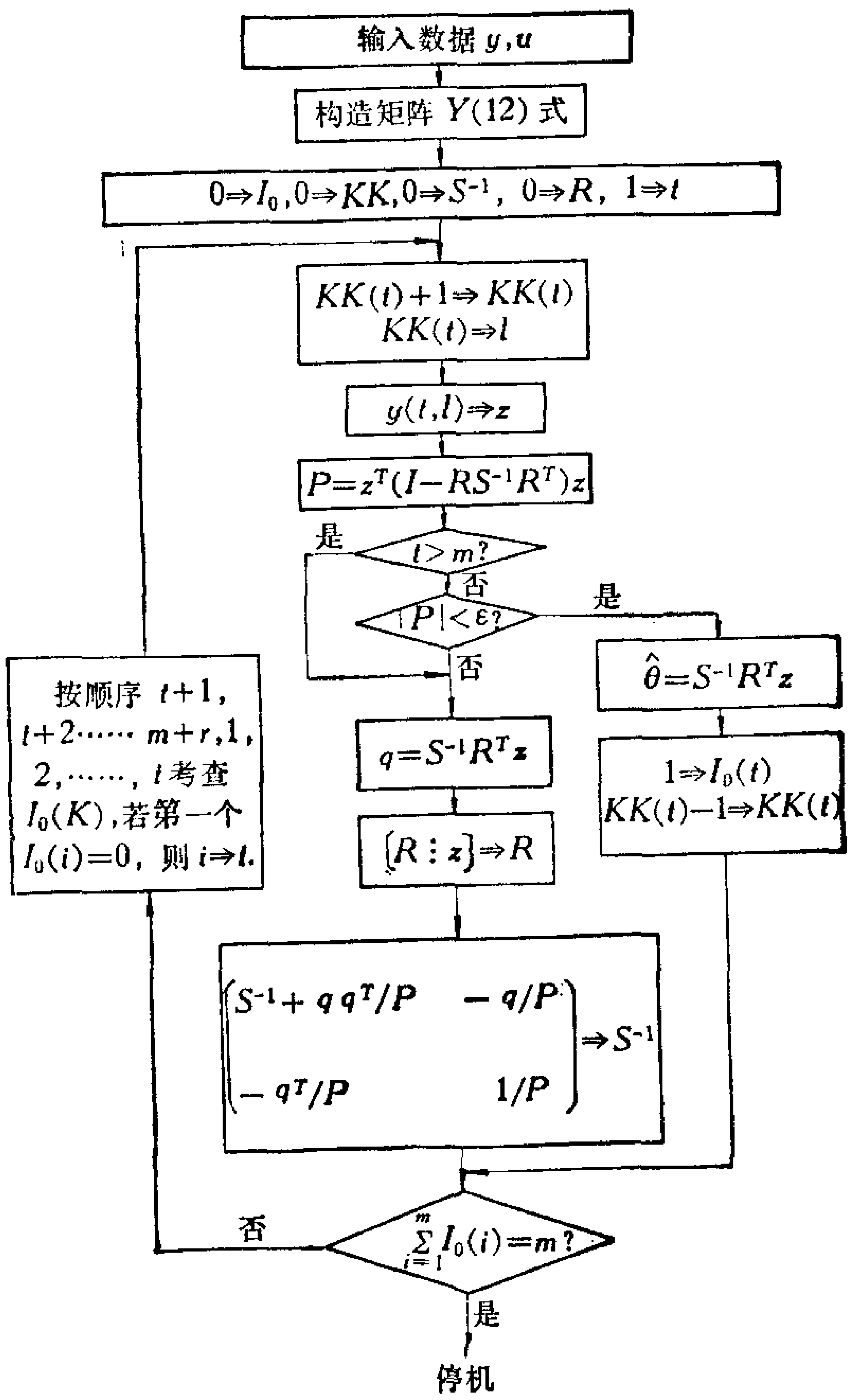

图 1

表 1

K	0	1	2	3	4	5	6	7	8	9	10	11	12
$u(K)$	1	2	4	5	−5	−12	15	50	10	−60	30	0	0
$y_1(K)$	0	0	1	0	2	5	−3	5	−25	56	−3	−27	40
$y_2(K)$	0	0	0	2	−1	4	−9	10	8	−7	10	−4	5

采用递推算法，用上面的数据，取 $\varepsilon=10^{-4}$ 计算结果如表 2 所示．

表 2

	$S(K_1,K_2,K_3)$									
K_1	1	1	1	2	2	2	3	3	3	4
K_2	0	1	1	1	2	2	2	3	2	2
K_3	0	0	1	1	1	2	2	2	3	3
P	3825	2504	2911	1847	29	266	1.65	3.3×10^{-6}	222	-7.6×10^{-5}

从而得到 $\nu_2=2$，$\nu_{21}=3$，$\nu_{23}=2$，$\nu_1=3$，$\nu_{12}=2$，$\nu_{13}=3$．同时得到系数的估计值(表 3)．

表 3

	$a_{21,1}$	$a_{21,2}$	$a_{21,3}$	$a_{22,1}$	$a_{22,2}$	$\beta_{4,1}$	$\beta_{5,1}$	$a_{11,1}$	$a_{11,2}$	$a_{11,3}$	$a_{12,1}$	$a_{12,2}$	$\beta_{1,1}$	$\beta_{2,1}$	$\beta_{3,1}$
系数真值	1	0	−1	1	−2	1	0	1	0	−1	−1	1	1	0	0
估计系数	1	6.5×10^{-7}	−1	1	−2	1	5.7×10^{-8}	1	-2×10^{-6}	−1	−1	1	1	3.5×10^{-7}	-2×10^{-7}

从上例可以看出，对于这样一个简单系统用 [1] 中的方法确定其结构就需计算七次行列式的值,行列式的最高阶达到九阶. 而递推算法完全不需要求行列式值的运算,大大减少运算时间,简单易行. 对于变量较多的系统更显出其优越性.

五、数据包含随机噪声的情况

考虑量测噪声,是实际上常见的情形. 假定 $y_j^*(k)$ 是实测的有噪声数据

$$y_j^*(k) = y_j(k) + \varepsilon_j(k)$$
$$u_j^*(k) = u_j(k) + \eta_j(k)$$

其中 $\varepsilon_j(k)$, $\eta_j(k)$ 是与 $y_j(k)$, $u_j(k)$ 不相关的零均值随机噪声. 一般在工程上总可以认为它是平稳遍历的. 在这种情况下，最小二乘估计将是有偏的. 但是只要噪声的统计特性已知就可补偿掉这一偏差，而得到参数的相容估计量.

另一个要解决的问题是如何从有噪声的数据中得到结构辨识所必须的信息 S_i. 由于对噪声特性所作的假定有[1]

$$\operatorname{Plim}_{N\to\infty} \frac{1}{N} S_i^* = \lim_{N\to\infty} \frac{1}{N} S_i + Q_i$$

其中 Q_i 是噪声向量

$$\{\varepsilon_1(k), \varepsilon_1(k+1), \cdots, \varepsilon_1(k+\nu_{i1}) \vdots \cdots \vdots \eta_r(k), \eta_r(k+1), \cdots, \eta_r(k+\nu_i-1)\}$$

的协方差矩阵.

同时有

$$\operatorname{Plim}_{N\to\infty} \frac{1}{N} R_i^{T*} \bar{\boldsymbol{y}}_i^*(k+\nu_i) = \lim_{N\to\infty} \frac{1}{N} R_i^T \bar{\boldsymbol{y}}_i(k+\nu_i) + \boldsymbol{q}_i$$

其中

$$\boldsymbol{q}_i^T = \lim_{N\to\infty} \frac{1}{N} \Big\{ \sum_{j=k}^{N} \varepsilon_1(j)\varepsilon_i(j+\nu_i), \cdots, \sum_{j=k}^{N} \varepsilon_1(j+\nu_{i1}-1)\varepsilon_i(j+\nu_i),$$
$$\cdots, \sum_{j=k}^{N} \eta_r(j+\nu_i-1)\varepsilon_i(j+\nu_i) \Big\}$$

如果噪声统计特性已知,即 Q_i, $\boldsymbol{q}_i$ 已知,即可得到参数的相容估计量

$$\hat{S}_i = S_i^* - NQ_i$$
$$\hat{R}_i^T \hat{\boldsymbol{y}}_i(k+\nu_i) = R_i^{T*} \hat{\boldsymbol{y}}_i^*(k+\nu_i) - N\boldsymbol{q}_i$$

经过这样补偿后的最小二乘估计

$$\hat{\boldsymbol{\theta}}_i = \hat{S}_i^{-1} \hat{R}_i^T \hat{\boldsymbol{y}}_i(k+\nu_i)$$

将是参数的相容估计量. 结构辨识可利用矩阵 $\hat{S}(\mu_1, \cdots, \mu_{m+r})$. 以上事实的详细数学

推证可参看 [8].

下面考虑在实际中最容易检验又便于应用的情形，即对各子系统所加的噪声互不相关,零均值并具有相同的方差,即 $Q_i = \sigma^2 I$, $\boldsymbol{q}_i = 0$. 在这种情况下,上面讨论的关系可简化为

$$P\lim_{N\to\infty} \frac{1}{N} S_i^* = \lim_{N\to\infty} \frac{1}{N} S_i - \sigma^2 I$$

$$P\lim_{N\to\infty} \frac{1}{N} R_i^{T*} \bar{\boldsymbol{y}}_i^*(k + \nu_i) = \lim_{N\to\infty} \frac{1}{N} R_i^T \bar{\boldsymbol{y}}_i(k + \nu_i)$$

所得到的参数估计应为

$$\hat{\boldsymbol{\theta}}_i = (S_i^* - N\sigma^2 I)^{-1} R_i^{T*} \bar{\boldsymbol{y}}_i^*(k + \nu_i)$$

因此，对有噪声的情况只需从有噪声的 S_i^* 中减去 $N\sigma^2 I$ 就可代替无噪声情形的 S_i 去进行结构辨识,有关的递推算法只需作微小的改动即可同样进行. 具体讨论如下:

考察 S_{i+1} 与 S_i 的关系,(为了方便起见将“ * ”号省掉，以下计算中的数据皆指有噪声数据),假定 S_i 已经补偿过了,则

$$S_{i+1} = \begin{bmatrix} S_i & R_i^T \bar{\boldsymbol{y}}_{i,\mu_j} \\ \bar{\boldsymbol{y}}_{j,\mu_j}^T R_i & \bar{\boldsymbol{y}}_{j,\mu_j}^T \bar{\boldsymbol{y}}_{j,\mu_j} - N\sigma^2 \end{bmatrix}$$

因此行列式的递推算法中公式 (22) 修改为

$$\det S_{i+1} = \det S_i \cdot \det [\bar{\boldsymbol{y}}_{j,\mu_j}(I - R_i S_i^{-1} R_i^T)\bar{\boldsymbol{y}}_{j,\mu_j} - N\sigma^2]$$

在结构辨识中对 $\det S_i$ 的考察方法不变.

矩阵求逆公式 (25) 中,只要用

$$P_1^{-1} = \bar{\boldsymbol{y}}_{j,\mu_j}(I - R_i S_i^{-1} R_i^T)\bar{\boldsymbol{y}}_{j,\mu_j} - N\sigma^2$$

代替 (25) 中的 P_1^{-1}, 其他皆不变. 将上面修改过的递推公式换入递推算法的有关步骤,就得到用于有噪声数据的递推算法.

下面提出几点注意事项:

(1) 如果各子系统的随机干扰是互不相关的，只要 Q_i 已知，同样可以按上面的办法对 S_i 进行补偿. 只是补偿不是一个常值,而是将 Q_i 的有关部分补偿进去.

(2) 当所有输入输出量测进行标度变换后，假定同方差在实际上是合理的，σ^2 可通过统计估计得到，也可以对某个假定的较大的阶 μ, 求 $S(\mu, \mu, \cdots, \mu)$ 的最小特征值来得到[1].

(3) 上面的讨论是在样本量 N 充分大时作的近似.

(4) 在确定性情形,从理论上说 $\det S_i$ 只有不等于零和等于零两种情形. 但在实际计算时或在有噪声数据时,行列式的值不会精确为零. 因此给出一个阈值 ε, ε 的大小应与噪声的统计性质及计算误差有关，但是对此进行严格的分析是十分困难的. 因此常常是按照一些经验的办法来取 ε, 具体做法见前面的例注. 这里不检验行列式的值而检验 P_1^{-1} 的值,这样可以消除一些计算误差,同时更突出了新加列向量的性质.

综上所述多变量系统结构辨识是一个十分重要而又复杂的问题. 本文给出的递推算法简单易行,为多变量系统结构辨识提供了有利工具.

对含有随机噪声的数据也给出了相应的补偿计算方法，因而能适用于一般的实际情况.

参 考 文 献

[1] Guidorzi, R., Canonical Structures in the Identification of Multivariable Systems, *Automatica*, 11(1975), 361—374.
[2] Bingulac, S. P., Farias, M. A. C., Identification and Minimal Realization of Multivariable Systems, Proc. of the 4th. IFAC Symp. on Multivariable Technological Systems, (1976), 373—378.
[3] Blessing, P., Parameter Estimation of State Space Models for Multivariable Systems with Correlation Analysis and Method of Least Squares, Proc. of the 4th. IFAC Symp. on Multivariable Technological Systems, (1976), 385—394.
[4] Sinha, N. K., Kwong, Y. H., Recursive Identification of the Parameters of Multivariable Systems, Proc. of the 4th. IFAC Symp. on Multivariable Technological Systems, (1976), 323—328.
[5] Gauthser, A., Landau, I. D., On the Recursive Identification of Multi-input Multi-output Systems, *Automatica*, 14(1978), 609.
[6] Sherief, H. El., Sinha, N. K., Stochastic Approximation Algorithm for the Identification of Linear Multivariable Systems, *IEEE Trans. on Automatic Control*, AC-24(1979), 331—332.
[7] Popov, V. M., Invariant Description of Linear Time-invariant Controllable Systems, *SIAM J. Control*, 10(1972), 252—265.
[8] Bonivento, C., Guidorzi, R., Parametric Identification of Linear Multivariable Systems, Preprint of the J. A. C. C., St. Louis, (1971).

A RECURSIVE IDENTIFICATION ALGORITHM FOR LINEAR MULTIVARIABLE SYSTEMS

WANG XIUFENG LU GUIZHANG
(*Nankai University*)

ABSTRACT

Structural identification of a multivariable system is very troublesome. Guidorzi[1] has proposed a method for identification of a set of invariant indices, but in which the inverse of matrices has to be evaluated. In this paper a recursive algorithm for determinted structural indices is described. In the whole algorithm it is not necessary to evaluate the value of determinant. Once the structure of subsystem is determined, we can obtain the estimation of parameters without repeated calculations and superfluous tasks. So the amount of calculation has reduced greatly.