文章编号:1674-7070(2012)06-0481-15

辨识方法的计算效率(3):信息向量耦合算法

丁锋[1,2,3]

摘要

简单讨论了行列式、矩阵逆和块矩阵逆的计算量;研究了信息向量耦合型多变量系统的子系统递推最小二乘辨识方法,给出了计算量小的联合递推最小二乘辨识算法;研究了部分信息向量耦合型多变量系统的子系统最小二乘辨识算法,提出了计算量小的基于块矩阵求逆的最小二乘辨识算法;给出了部分信息向量耦合型多变量系统的子系统递推最小二乘辨识算法,提出和推导了基于辨识模型分解的递推最小二乘辨识算法,并分别讨论了提出算法的计算量.

关键词

递推辨识; 迭代辨识; 参数估计; FIR 模型; 方程误差模型; CAR 模型; CARMA 模型; CARAR 模型; CARARMA 模型; 输出误差模型; OEMA 模型; OEAR 模型; 辅助模型辨识; 多新息辨识; 递阶辨识; 耦合辨识

中图分类号 TP273

文献标志码 A

收稿日期 2012-11-15

资助项目 国家自然科学基金(61273194);江苏省自然科学基金(BK2012549);高等学校学科创新引智计划(B12018)

作者简介

丁锋,男,博士,教授,博士生导师,主要从事系统辨识、过程建模、自适应控制方面的研究. fding@ jiangnan. edu. cn

1 江南大学 物联网工程学院,无锡,214122

2 江南大学 控制科学与工程研究中心,无锡,214122

3 江南大学 教育部轻工过程先进控制重点实验室,无锡,214122

0 引言

客观事物的运动规律用方程描述就是数学模型. 动态系统数学模型是描述客观事物内部运动或外部运动规律的强有力工具,是从理论上研究事物运动规律的唯一且重要手段. 系统辨识是研究建立(动态)系统数学模型的理论与方法. 系统建模方法或系统辨识方法在各门学科中都有许多重要应用. 开普勒关于行星运动的三大定律就是天文学中行星运动的基本数学模型.

约翰尼斯·开普勒是杰出的德国天文学家,他发现了行星运动的三大定律,分别是轨道定律、面积定律和周期定律. 这三大定律可分别描述为:1)行星运行的轨道为椭圆,太阳居其一焦点,意思是说所有行星分别是在大小不同的椭圆轨道上运行,此即轨道模型;2)行星与太阳连线在等长的时间内扫过相等的面积,意思是说在相同时间里行星向径在轨道平面上所扫过的面积相等,此即面积模型;3)行星绕行太阳一周所需要的时间 T 和行星到太阳的距离 R(椭圆轨道的半长轴)之间关系 T^2/R^3 为定数,意思是说行星公转周期的平方与它同太阳距离的立方成正比,此即周期-距离模型. 这三大模型最终使他赢得了"天空立法者"的美名. 还有古希腊数学家和力学家阿基米德的浮力定律和杠杆原理,英国力学家胡克于 1678 年发现的固体材料的弹性定律,即胡克定律(Hookeś law)(实际上,我国东汉的经学家和教育家郑玄为《考工记·马人》一文的"量其力,有三钧"一句作注解中写到:"假设弓力胜三石,引之中三尺,驰其弦,以绳缓擐之,每加物一石,则张一尺."也揭示了力与形变成正比的关系,郑玄的发现要比胡克早 1 500 年. 郑玄的发现可以说是胡克定律的原始雏形,因此胡克定律应称之为"郑玄-胡克定律"). 还有牛顿的万有引力定律,静止点电荷间相互作用力的库仑定律等,都是描述事物运动规律的简单的代数方程,都是静态数学模型. 这些静态模型只是揭示了事物的本原,只是认识世界,发现客观事物的原形,即只是发现客观事物本身是什么.

随着科学技术的发展,只是认识世界还远远不够,我们还必须改造世界,施加控制作用使事物往人们期望的方向发展. 因此,只有客观事物的静态数学模型远远不够,还必须建立和发展客观事物的动态数学模型,实现对客观事物的控制,达到认识自然,改造自然,为人类造福的目的[1].

特别是在近代控制理论与自动化技术高速发展的年代,在控制

论诞生的半个多世纪以来,在系统辨识正式成为一门学科的半个世纪以来,建立基于微分方程的动态系统数学模型的系统辨识得到了长足发展,通过实验的动态系统建模方法、基于统计数据的动态系统辨识建模方法,以及辨识方法的机理分析和性能分析等都取得了许多辉煌的成就[1-11]. 在新的辨识方法不断问世的时代,最小二乘辨识方法[1]、随机梯度辨识方法[12-13]、牛顿迭代辨识方法和牛顿递推辨识方法[14]、辅助模型辨识方法[5,15-18]、迭代辨识方法[6,19-20]、多新息辨识方法[7,21-24]、递阶辨识方法[8,25-31]、耦合辨识方法[9,32-33]、两阶段辨识方法[8,34-36]等都是典型的辨识方法.

辨识方法包括一次完成算法或直接算法(direct algorithm),即最小二乘辨识算法,递推辨识方法和迭代辨识方法. 文献[10]讨论了线性回归系统、多元线性回归系统、多变量系统的随机梯度辨识算法、最小二乘辨识算法(即一次完成最小二乘算法或直接算法)、递推最小二乘辨识算法的计算量. 文献[11]讨论了最小二乘迭代辨识算法及其计算效率,主要通过块矩阵求逆方法来减小最小二乘迭代算法求逆的计算量,内容包括伪线性回归系统、多元伪线性回归系统、多变量伪线性回归系统的最小二乘迭代辨识算法及其基于块矩阵求逆的最小二乘迭代辨识算法.

一个多变量系统可以按照输出的数目分解为一系列多输入单输出子系统,每个子系统都对应一个辨识模型,根据各子辨识模型的特征,可以分为(部分)信息向量耦合型多变量系统和(部分)参数向量耦合型多变量系统等[1,9]. 本文简单讨论行列式、矩阵逆、块矩阵逆的计算量,仍然采用 flop 数,即浮点运算数表示. 一次加法运算称为一个 flop,一次乘法运算也称为一个 flop[10-11,37]. 除法作为乘法对待,减法作为加法对待. 随后利用联合辨识思想,研究了信息向量耦合型多变量系统辨识算法,推导和分析了联合递推最小二乘辨识算法的计算效率[38],提出了部分信息向量耦合型多变量系统的子系统最小二乘辨识算法、基于块矩阵求逆的最小二乘辨识算法、子系统递推最小二乘辨识算法和基于辨识模型分解的递推最小二乘辨识算法. 尽管部分信息向量耦合型多变量系统的基于块矩阵求逆的最小二乘辨识算法,在形式上很类似于文献[11]中伪线性回归系统的基于块矩阵求逆的最小二乘迭代辨识算法,但是它们有本质区别,前者用于白噪声干扰的多变量线性回归系统,且信息向量是可测的(即信息向量不含未知噪声项),后者用于有色噪声干扰的伪线性回归系统,即信息向量中含有不可测噪声项伪线性标量系统或伪线性多变量系统.

1 行列式与矩阵逆的计算量

1.1 行列式的计算量

方阵 $\boldsymbol{\Lambda}\in\mathbf{R}^{n\times n}$ 的行列式 $\det[\boldsymbol{\Lambda}]:=|\boldsymbol{\Lambda}|$ 有 $n!$ 项求和,每项是不同行和不同列的 n 个元相乘,每项有 $(n-1)$ 次乘法运算,故行列式有 $M_1:=n!\ (n-1)$ 次乘法运算和 $A_1:=(n!\ -1)$ 次加法运算,总 flop 数为 $N_1:=M_1+A_1=n\cdot n!\ -1$. 计算量数量级是 $O(nn!)$,可见这样计算行列式的计算量是很大的.

下面利用行列式的性质来计算,即通过将行列式的一行乘以一个常数加到另一行的办法来计算行列式. 具体步骤如下. 设 $n\times n$ 行列式 $|\boldsymbol{A}|$ 的第 i 行记为 r_i,第 (i,j) 元记为 a_{ij},$i=1,2,\cdots,n$,$j=1,2,\cdots,n$.

1) 首先将第 1 列的 $(n-1)$ 个元变为零. 设 $a_{11}\neq 0$(否则从其下元开始,下同),第 1 行乘以 $-a_{i1}/a_{11}$ 加到第 i 行,即 $(-a_{i1}/a_{11})r_1+r_i\rightarrow r_i\ (i=2,3,\cdots,n)$(因为只关心行列式的值,所以第(1,1)元下方的元不用计算,为 0,以节省运算次数),每行的乘法运算次数为 $1+(n-1)$,加法运算次数为 $n-1$(减法作为加法对待),共有 $n-1$ 行. 操作后的行列式结构形式如下(为了表述方便,操作后的行列式的第 (i,j) 元依然记为 a_{ij}):

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & a_{n2} & a_{n3} & \cdots & a_{nn} \end{vmatrix}.$$

2) 其次把去掉第 1 行和第 1 列的子行列式化为上述形式,即将第(2,2)元之下的 $(n-2)$ 个元变为零. 设 $a_{22}\neq 0$,第 2 行乘以 $-a_{i2}/a_{22}$ 加到第 i 行,即 $(-a_{i2}/a_{22})r_2+r_i\rightarrow r_i\ (i=3,4,\cdots,n)$(因为只关心行列式的值,所以第(2,2)元下方的元不用计算,为 0,以节省运算次数),每行的乘法运算次数为 $1+(n-2)$,加法运算次数为 $n-2$,共有 $n-2$ 行. 操作后的行列式结构形式为

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & \cdots & a_{2n} \\ 0 & 0 & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{n3} & \cdots & a_{nn} \end{vmatrix}.$$

3) 第3行乘以 $-a_{i3}/a_{33}$ 加到第 i 行,即 $(-a_{i3}/a_{33})r_3+r_i\rightarrow r_i$ $(i=4,5,\cdots,n)$ (第(3,3)元下方的元不用计算,为0),每行的乘法运算次数为 $1+(n-3)$,加法运算次数为 $n-3$,共有 $n-3$ 行. 操作后的行列式结构形式为

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & a_{24} & \cdots & a_{2n} \\ 0 & 0 & a_{33} & a_{34} & \cdots & a_{3n} \\ 0 & 0 & 0 & a_{44} & \cdots & a_{4n} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & a_{4n} & \cdots & a_{nn} \end{vmatrix}.$$

4) 以此类推,直到第 $n-1$ 行,$(-a_{n,n-1}/a_{n-1,n-1})r_{n-1}+r_n\rightarrow r_n$,乘法运算次数为 $1+1$,加法运算次数为1,操作后的行列式结构形式为

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & \cdots & a_{2n} \\ 0 & 0 & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & a_{nn} \end{vmatrix}.$$

计算这个行列式有 $(n-1)$ 乘法运算.

至此,操作后的行列式具有上三角形式,其加法运算次数为

$$A_2:=1\cdot 1+2\cdot 2+\cdots+(n-2)\cdot(n-2)+(n-1)\cdot(n-1)=1^2+2^2+\cdots+(n-1)^2=\frac{n(n-1)(2n-1)}{6}\text{flops},$$

乘法运算次数为

$$M_2:=1\cdot(1+1)+2\cdot(1+2)+\cdots+(n-2)[1+(n-2)]+(n-1)[1+(n-1)]+(n-1)=1+2+\cdots+(n-1)+1^2+2^2+\cdots+(n-1)^2+(n-1)=\frac{n(n-1)}{2}+\frac{n(n-1)(2n-1)}{6}+(n-1)=\left(\frac{1}{3}n^3+\frac{2}{3}n-1\right)\text{flops},$$

总 flop 数为

$$N_2:=A_2+M_2=\frac{n(n-1)(2n-1)}{6}+\frac{1}{3}n^3+\frac{2}{3}n-1=\left(\frac{2}{3}n^3-\frac{1}{2}n^2-\frac{5}{6}n-1\right)\text{flops}.$$

这种方法计算行列式计算量数量级为 $O(n^3)$.

下面是利用行列式性质计算行列式的 Matlab 程序.

```
%————————————————————————————*
% Filename:DeterminantEfficiency1.m *
%      Compute the determinant of matrix A *
%————————————————————————————*
clear; format short g
%——Generate matrix A
n=5;
rand('state',2);
a=round(rand(n,n)*10);
% a=pascal(n);
% a=magic(n);
 a1=a
 fprintf('----Method 1')
 for k=1:n
     for  i=k+1:n
          c=-a1(i,k)/a1(k,k);
          a1(i,:)=c*a1(k,:)+a1(i,:);
     end
end
Uppertriangularmatrix=a1
DetA=1;
for  i=1:n
     DetA=DetA*a1(i,i);
end
[DetA,det(a)]
27  fprintf('----Method 2 with minimimun flops')
a1=a;
for k=1:n
for  i=k+1:n
     c=-a1(i,k)/a1(k,k);
     a1(i,k+1:n)=c*a1(k,k+1:n)+a1(i,k+1:n);
     end
end
NonUppertriangularmatrix=a1
DetA=1;
for  i=1:n
     DetA=DetA*a1(i,i);
end
[DetA,det(a)]
```

1.2 矩阵逆的计算量

如采用伴随矩阵(adjoint matrix) adj[$\boldsymbol{\Lambda}$]计算矩阵 $\boldsymbol{\Lambda}\in\mathbf{R}^{n\times n}$ 的逆,即

$$\boldsymbol{\Lambda}^{-1}=\frac{\text{adj}[\boldsymbol{\Lambda}]}{\det[\boldsymbol{\Lambda}]}\in\mathbf{R}^{n\times n}.$$

伴随矩阵 adj[$\boldsymbol{\Lambda}$] 有 n^2 个元,每个元是一个 $(n-1)$ 阶行列式,有 $(n-1)!$ $(n-2)$ 次乘法运算和

$(n-1)!\ -1$ 次加法运算. 因此,这种方法计算矩阵的逆,其乘法运算次数为

$$M_3:=n!\ (n-1)+n^2[(n-1)!\ (n-2)]=(n^2-n-1)n!\ \text{flops},$$

加法运算次数为

$$A_3:=(n!\ -1)+n^2[(n-1)!\ -1]=[(n+1)!\ -n^2-1]\text{flops}.$$

总 flop 数为

$$N_3:=M_3+A_3=(n^2-n-1)n!\ +(n+1)!\ -n^2-1=(n^2n!\ -n^2-1)\text{flops},$$

这样计算矩阵逆的计算量数量级为 $O(n^2n!)$. 由此可见,利用伴随矩阵求矩阵逆的计算量非常大.

文献[10-11]给出了矩阵乘法、矩阵求逆、块矩阵逆计算量的浮点运算数,即 flop 数. 文献[11]中表 1 给出了计算 $n\times n$ 维矩阵逆的乘法运算次数$\frac{4}{3}n^3+\frac{2}{3}n$ 和加法运算次数$\frac{4}{3}n^3-\frac{3}{2}n^2+\frac{1}{6}n$. 进一步研究表明,矩阵求逆的计算量可以进一步减小:一个 $n\times n$ 矩阵逆的乘法运算次数为 $M_4:=n^3$,加法运算次数为 $A_4:=n^3-n^2$,计算量为 $N_4:=M_4+A_4=(2n^3-n^2)$ flops,数量级为 $O(n^3)$.

1.3 块矩阵逆的计算量

首先介绍块矩阵求逆引理,它可以用来简化耦合多变量系统辨识算法的计算量.

引理 1 (块矩阵求逆引理)(block matrix inversion lemma)假设矩阵 $\boldsymbol{A}\in\mathbf{R}^{m\times m}$,$\boldsymbol{B}\in\mathbf{R}^{m\times n}$,$\boldsymbol{C}\in\mathbf{R}^{n\times m}$,$\boldsymbol{D}\in\mathbf{R}^{n\times n}$,且 $\boldsymbol{A}$ 和 $\boldsymbol{Q}:=\boldsymbol{D}-\boldsymbol{CA}^{-1}\boldsymbol{B}\in\mathbf{R}^{n\times n}$ 是可逆矩阵,那么下列块矩阵逆关系成立,

$$\begin{bmatrix}\boldsymbol{A} & \boldsymbol{B}\\ \boldsymbol{C} & \boldsymbol{D}\end{bmatrix}^{-1}=\begin{bmatrix}\boldsymbol{A}^{-1}+\boldsymbol{A}^{-1}\boldsymbol{BQ}^{-1}\boldsymbol{CA}^{-1} & -\boldsymbol{A}^{-1}\boldsymbol{BQ}^{-1}\\ -\boldsymbol{Q}^{-1}\boldsymbol{CA}^{-1} & \boldsymbol{Q}^{-1}\end{bmatrix}\in\mathbf{R}^{(m+n)\times(m+n)}. \tag{1}$$

引理 2 (块矩阵求逆引理)(block matrix inversion lemma)假设矩阵 $\boldsymbol{A}\in\mathbf{R}^{m\times m}$,$\boldsymbol{B}\in\mathbf{R}^{m\times n}$,$\boldsymbol{C}\in\mathbf{R}^{n\times m}$,$\boldsymbol{D}\in\mathbf{R}^{n\times n}$,且 $\boldsymbol{D}$ 和 $\boldsymbol{Q}:=\boldsymbol{A}-\boldsymbol{BD}^{-1}\boldsymbol{C}\in\mathbf{R}^{m\times m}$ 是可逆矩阵,那么下列块矩阵逆关系成立,

$$\begin{bmatrix}\boldsymbol{A} & \boldsymbol{B}\\ \boldsymbol{C} & \boldsymbol{D}\end{bmatrix}^{-1}=\begin{bmatrix}\boldsymbol{Q}^{-1} & -\boldsymbol{Q}^{-1}\boldsymbol{BD}^{-1}\\ -\boldsymbol{D}^{-1}\boldsymbol{CQ}^{-1} & \boldsymbol{D}^{-1}+\boldsymbol{D}^{-1}\boldsymbol{CQ}^{-1}\boldsymbol{BD}^{-1}\end{bmatrix}\in\mathbf{R}^{(m+n)\times(m+n)}. \tag{2}$$

根据 1.2 节矩阵求逆的计算量 M_4,A_4 和 N_4,以及文献[11]中表 1,可以得到引理 1 中块矩阵求逆计算量,如表 1 所示(在计算运算量时,括号里面的矩阵相乘作为一个整体对待). 式(1)左边矩阵求逆的计算量的总 flop 数为

$$N_5:=2(m+n)^3-(m+n)^2.$$

表 1 给出了式(1)右边表达式的乘法运算次数和加法运算次数,其总 flop 数为

$$N_6:=2(m^3+n^3)+6m^2n+6mn^2-2m^2-4mn.$$

它们之差为

$$N_5-N_6=2(m+n)^3-(m+n)^2-[2(m^3+n^3)+6m^2n+6mn^2-2m^2-4mn]=2(3m^2n+3mn^2)-(6m^2n+6mn^2)-(m+n)^2+2m^2+4mn=m^2+2mn-n^2.$$

表 1 块矩阵求逆的计算量

Table 1 The computational efficiency of the block matrix inversion

变量	计算次序	乘法次数	加法次数
$\boldsymbol{A}_1$	$\boldsymbol{A}_1:=\boldsymbol{A}'-(\boldsymbol{A}'\boldsymbol{B})\boldsymbol{C}_1\in\mathbf{R}^{m\times m}$	m^2n	$m^2n-m^2+n^2$
$\boldsymbol{B}_1$	$\boldsymbol{B}_1:=(\boldsymbol{A}'\boldsymbol{B})\boldsymbol{Q}'\in\mathbf{R}^{m\times n}$	mn^2	mn^2-mn
$\boldsymbol{C}_1$	$\boldsymbol{C}_1:=\boldsymbol{Q}'(\boldsymbol{CA}')\in\mathbf{R}^{n\times m}$	mn^2	mn^2-mn
$\boldsymbol{Q}'$	$\boldsymbol{Q}':=\boldsymbol{Q}^{-1}\in\mathbf{R}^{n\times n}$	n^3	n^3-n^2
$\boldsymbol{A}'$	$\boldsymbol{A}':=\boldsymbol{A}^{-1}\in\mathbf{R}^{m\times m}$	m^3	m^3-m^2
$\boldsymbol{Q}$	$\boldsymbol{Q}=\boldsymbol{D}-(\boldsymbol{CA}')\boldsymbol{B}\in\mathbf{R}^{n\times n}$	mn^2	mn^2
	$\boldsymbol{A}'\boldsymbol{B}\in\mathbf{R}^{m\times n}$	m^2n	m^2n-mn
	$\boldsymbol{CA}'\in\mathbf{R}^{n\times m}$	m^2n	m^2n-mn
	总 数	$(m^3+n^3)+3m^2n+3mn^2$	$(m^3+n^3)+3m^2n+3mn^2-2m^2-4mn$
	总 flop 数	$N_6:=2(m^3+n^3)+6m^2n+6mn^2-2m^2-4mn$	

因此,当 $m>(\sqrt{2}-1)n$ 时,$N_5-N_6>0$. 这说明直接矩阵求逆的计算量比使用块矩阵求逆的计算量大. 例如当 $m=5,n=5$ 时,$N_5=1\ 900$flops,$N_6=1\ 850$flops,$N_5-N_6=50$flops. 当 $m=9,n=1$ 时,$N_5=1\ 900$flops,$N_6=1\ 802$flops,$N_5-N_6=98$flops.

简单起见,当 $m\geqslant n$ 时,采用块矩阵求逆引理 1 求矩阵的逆,计算量小,因为式(1)右边的计算量小于左边直接求逆的计算量;相反,当 $m<n$ 时,采用引理 2 的块矩阵求逆公式,计算量小. 利用块矩阵求逆公式计算对称矩阵的逆,计算量要更小,而直接利用高斯削元法的增广矩阵线性变换求矩阵逆的计算量不会有任何减小,即对称矩阵与非对称矩阵计算量是一样的.

2 信息向量耦合型多变量系统

信息向量耦合型多变量系统包括信息向量全耦合多变量系统和部分信息向量耦合多变量系统. 在不至于混淆的情况下,信息向量全耦合多变量系统简称为信息向量耦合型多变量系统,是指各子系统辨识模型的信息向量都相同.

正如笔者在文献[1,9]中指出的,一个多输入多输出系统,可以分解为 m 个多输入单输出子系统(m 为系统输出的数目),每个子系统对应一个子辨识模型. 根据子辨识模型间的参数向量、信息向量的耦合关系,将多变量辨识模型分为七类,每一类代表一类多变量系统. 这些辨识模型如下:

1) 独立型(子系统参数向量独立、信息向量独立)多变量系统;

2) 信息向量耦合型(子系统信息向量耦合、参数向量独立)多变量系统;

3) 参数向量耦合型(子系统参数向量耦合、信息向量独立)多变量系统;

4) 部分信息向量耦合型(子系统部分信息向量耦合、参数向量独立)多变量系统;

5) 部分参数向量耦合型(子系统部分参数向量耦合、信息向量独立)多变量系统;

6) 部分参数向量和部分信息向量耦合型(子系统部分参数向量和部分信息向量耦合)多变量系统;

7) 部分参数向量耦合、部分信息向量耦合型(子系统部分参数向量耦合、部分信息向量耦合)多变量系统.

考虑下列多变量受控自回归系统(Multivariable Controlled AutoRegressive system, Multivariable CAR 系统)[12-13]

$$\boldsymbol{A}(z)\boldsymbol{y}(t)=\boldsymbol{B}(z)\boldsymbol{u}(t)+\boldsymbol{v}(t),\tag{3}$$

其中 $\boldsymbol{u}(t)=[u_1(t),u_2(t),\cdots,u_r(t)]^{\mathrm{T}}\in\mathbf{R}^r$ 为系统输入向量,$\boldsymbol{y}(t)=[y_1(t),y_2(t),\cdots,y_m(t)]^{\mathrm{T}}\in\mathbf{R}^m$ 为系统输出向量,$\boldsymbol{v}(t)=[v_1(t),v_2(t),\cdots,v_m(t)]^{\mathrm{T}}\in\mathbf{R}^m$ 为零均值随机噪声向量,$\boldsymbol{A}(z)$ 和 $\boldsymbol{B}(z)$ 均为单位后移算子 z^{-1} 的多项式矩阵[$z^{-1}\boldsymbol{y}(t)=\boldsymbol{y}(t-1)$],且

$$\boldsymbol{A}(z):=\boldsymbol{I}+\boldsymbol{A}_1z^{-1}+\boldsymbol{A}_2z^{-2}+\cdots+\boldsymbol{A}_{n_a}z^{-n_a},$$

$$\boldsymbol{B}(z):=\boldsymbol{B}_1z^{-1}+\boldsymbol{B}_2z^{-2}+\cdots+\boldsymbol{B}_{n_b}z^{-n_b}.$$

假设阶次 n_a 和 n_b 已知,$\{\boldsymbol{u}(t),\boldsymbol{y}(t)\}$ 是观测输入输出数据,$\boldsymbol{A}_i\in\mathbf{R}^{m\times m}$ 和 $\boldsymbol{B}_i\in\mathbf{R}^{m\times r}$ 是待辨识的系统参数矩阵.

定义参数矩阵 $\boldsymbol{\theta}$ 和信息向量 $\boldsymbol{\varphi}(t)$ 如下:

$$\boldsymbol{\theta}^{\mathrm{T}}:=[\boldsymbol{A}_1,\boldsymbol{A}_2,\cdots,\boldsymbol{A}_{n_a},\boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_{n_b}]\in\mathbf{R}^{m\times n},$$

$$n:=mn_a+rn_b,$$

$$\boldsymbol{\varphi}(t):=[-\boldsymbol{y}^{\mathrm{T}}(t-1),-\boldsymbol{y}^{\mathrm{T}}(t-2),\cdots,-\boldsymbol{y}^{\mathrm{T}}(t-n_a),$$

$$\boldsymbol{u}^{\mathrm{T}}(t-1),\boldsymbol{u}^{\mathrm{T}}(t-2),\cdots,\boldsymbol{u}^{\mathrm{T}}(t-n_b)]^{\mathrm{T}}\in\mathbf{R}^n.$$

则可得多变量 CAR 系统(3)的辨识模型:

$$\boldsymbol{y}(t)=\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(t)+\boldsymbol{v}(t).\tag{4}$$

因此,多变量受控自回归系统是一个信息向量耦合型多变量系统.

2.1 子系统递推最小二乘辨识算法

系统的参数矩阵 $\boldsymbol{\theta}$ 是一个 $n\times m$ 矩阵,将其分为 m 个列参数向量 $\boldsymbol{\theta}_i,i=1,2,\cdots,m$,即

$$\boldsymbol{\theta}=[\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_m]\in\mathbf{R}^{n\times m},\boldsymbol{\theta}_i\in\mathbf{R}^n.$$

系统有 m 个输出 $y_i(t)$,则系统(4)可以写为

$$\begin{bmatrix}y_1(t)\\y_2(t)\\\vdots\\y_m(t)\end{bmatrix}=\begin{bmatrix}\boldsymbol{\theta}_1^{\mathrm{T}}\\\boldsymbol{\theta}_2^{\mathrm{T}}\\\vdots\\\boldsymbol{\theta}_m^{\mathrm{T}}\end{bmatrix}\boldsymbol{\varphi}(t)+\begin{bmatrix}v_1(t)\\v_2(t)\\\vdots\\v_m(t)\end{bmatrix}.$$

则上式可以分解为 m 个子系统(子辨识模型):

$$y_i(t)=\boldsymbol{\theta}_i^{\mathrm{T}}\boldsymbol{\varphi}(t)+v_i(t),i=1,2,\cdots,m.\tag{5}$$

其特征是各子系统间信息向量 $\boldsymbol{\varphi}(t)$ 完全相同、参数向量 $\boldsymbol{\theta}_i$ 都不同. 这个模型可称为信息向量耦合型(子系统信息向量耦合、参数向量独立)多变量系统,即各子系统间信息向量全耦合多变量系统.

定义二次准则函数(quadratic criterion function)

$$J_1(\boldsymbol{\theta}_i):=\sum_{j=1}^{t}[y_i(j)-\boldsymbol{\theta}_i^{\mathrm{T}}\boldsymbol{\varphi}(j)]^2,\quad i=1,2,\cdots,m.$$

参考文献[4,9],极小化 $J_1(\boldsymbol{\theta}_i)$ 可以得到计算子系统参数估计 $\hat{\boldsymbol{\theta}}_i(t)$ 的递推最小二乘算法:

$$\hat{\boldsymbol{\theta}}_i(t)=\hat{\boldsymbol{\theta}}_i(t-1)+\boldsymbol{P}_i(t)\boldsymbol{\varphi}(t)[y_i(t)-\hat{\boldsymbol{\theta}}_i^{\mathrm{T}}(t-1)\boldsymbol{\varphi}(t)],$$

$$\hat{\boldsymbol{\theta}}_i(0)=\mathbf{1}_n/p_0, \tag{6}$$

$$\boldsymbol{P}_i^{-1}(t)=\boldsymbol{P}_i^{-1}(t-1)+\boldsymbol{\varphi}(t)\boldsymbol{\varphi}^{\mathrm{T}}(t),\quad \boldsymbol{P}_i(0)=p_0\boldsymbol{I}_n. \tag{7}$$

将矩阵求逆引理(matrix inversion lemma)

$$(\boldsymbol{A}+\boldsymbol{BC})^{-1}=\boldsymbol{A}^{-1}-\boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{I}+\boldsymbol{CA}^{-1}\boldsymbol{B})^{-1}\boldsymbol{CA}^{-1} \tag{8}$$

应用到式(7),便能够得到等价的子系统递推最小二乘(S-RLS)辨识算法:

$$\hat{\boldsymbol{\theta}}_i(t)=\hat{\boldsymbol{\theta}}_i(t-1)+\boldsymbol{P}_i(t)\boldsymbol{\varphi}(t)[y_i(t)-\hat{\boldsymbol{\theta}}_i^{\mathrm{T}}(t-1)\boldsymbol{\varphi}(t)],$$

$$\hat{\boldsymbol{\theta}}_i(0)=\mathbf{1}_n/p_0, \tag{9}$$

$$\boldsymbol{P}_i(t)=\boldsymbol{P}_i(t-1)-\frac{\boldsymbol{P}_i(t-1)\boldsymbol{\varphi}(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{P}_i(t-1)}{1+\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{P}_i(t-1)\boldsymbol{\varphi}(t)},$$

$$\boldsymbol{P}_i(0)=p_0\boldsymbol{I}_n. \tag{10}$$

2.2 联合递推最小二乘辨识算法

当 $i=1,2,\cdots,m$ 时,S-RLS 算法(6)—(7)或(9)—(10)能够计算出每个子系统的参数估计 $\hat{\boldsymbol{\theta}}_i(t)$,但是计算量大,因为通过观察式(7)和(10),可知每个子系统的协方差阵 $\boldsymbol{P}_i(t)$ 是相同的,即 $\boldsymbol{P}_i(t)=\boldsymbol{P}_j(t)$,却在每个子系统中都计算了一次. 将这个共同的协方差阵记作 $\boldsymbol{P}(t)$,则 S-RLS 算法(6)—(7)或(9)—(10)可以合并等价写为下列联合递推最小二乘辨识算法[38]:

$$\hat{\boldsymbol{\theta}}(t)=\hat{\boldsymbol{\theta}}(t-1)+\boldsymbol{P}(t)\boldsymbol{\varphi}(t)[y(t)-\hat{\boldsymbol{\theta}}^{\mathrm{T}}(t-1)\boldsymbol{\varphi}(t)]^{\mathrm{T}},$$

$$\hat{\boldsymbol{\theta}}(0)=\mathbf{1}_{n\times m}/p_0, \tag{11}$$

$$\boldsymbol{P}^{-1}(t)=\boldsymbol{P}^{-1}(t-1)+\boldsymbol{\varphi}(t)\boldsymbol{\varphi}^{\mathrm{T}}(t),\quad \boldsymbol{P}(0)=p_0\boldsymbol{I}_n, \tag{12}$$

或

$$\hat{\boldsymbol{\theta}}(t)=\hat{\boldsymbol{\theta}}(t-1)+\boldsymbol{P}(t)\boldsymbol{\varphi}(t)[\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}^{\mathrm{T}}(t-1)\boldsymbol{\varphi}(t)]^{\mathrm{T}},$$

$$\hat{\boldsymbol{\theta}}(0)=\mathbf{1}_{n\times m}/p_0, \tag{13}$$

$$\boldsymbol{P}(t)=\boldsymbol{P}(t-1)-\frac{\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{P}(t-1)}{1+\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)},$$

$$\boldsymbol{P}(0)=p_0\boldsymbol{I}_n. \tag{14}$$

这个多变量系统联合递推最小二乘辨识算法也可称为多变量系统耦合最小二乘辨识算法.

定义增益向量 $\boldsymbol{L}(t):=\boldsymbol{P}(t)\boldsymbol{\varphi}(t)\in\mathbf{R}^n$,算法(13)—(14)可以简单写为

$$\hat{\boldsymbol{\theta}}(t)=\hat{\boldsymbol{\theta}}(t-1)+\boldsymbol{L}(t)[\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}^{\mathrm{T}}(t-1)\boldsymbol{\varphi}(t)]^{\mathrm{T}},$$

$$\hat{\boldsymbol{\theta}}(0)=\mathbf{1}_{n\times m}/p_0, \tag{15}$$

$$\boldsymbol{L}(t)=\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)[1+\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)]^{-1}, \tag{16}$$

$$\boldsymbol{P}(t)=\boldsymbol{P}(t-1)-\boldsymbol{L}(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{P}(t-1),$$

$$\boldsymbol{P}(0)=p_0\boldsymbol{I}_n. \tag{17}$$

这个算法可以通过极小化下列准则函数得到:

$$J_1(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_m):=\sum_{i=1}^{m}J_1(\boldsymbol{\theta}_i)=$$

$$\sum_{i=1}^{m}\sum_{j=1}^{t}[y_i(j)-\boldsymbol{\theta}_i^{\mathrm{T}}\boldsymbol{\varphi}_i(j)]^2=$$

$$\sum_{j=1}^{t}[\boldsymbol{y}(j)-\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(j)]^{\mathrm{T}}[\boldsymbol{y}(j)-\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(j)].$$

参见文献[10]中表9的多变量递推最小二乘算法的计算量,可知估计式(4)中参数矩阵 $\boldsymbol{\theta}$ 的多变量递推最小二乘算法(Multivariable Recursive Least Squares algorithm,M-RLS 算法)(15)—(17)每递推计算一步的乘法运算次数为 $M_7:=2n^2+2mn+2n$,加法运算次数为 $A_7:=2n^2+2mn$,总 flop 数为 $N_7:=4n^2+4mn+2n$. 而子系统递推最小二乘算法(9)—(10)是独立的,相当于 m 个递推最小二乘算法,参见文献[9]中表3中 RLS 算法每步的计算量,可知子系统递推最小二乘算法(9)—(10)的乘法运算次数为 $M_8:=m(2n^2+4n)$,加法运算次数为 $A_8:=m(2n^2+2n)$,总 flop 数为 $N_8:=m(4n^2+6n)$. 它们之比为

$$\frac{N_7}{N_8}=\frac{4n^2+4mn+2n}{m(4n^2+6n)}=\frac{1}{m}+\frac{2}{2n+3}-\frac{2}{m(2n+3)}.$$

一般 $m<n$,有 $N_7/N_8<2/m$,即多变量递推最小二乘算法(15)—(17)的计算量不及子系统递推最小二乘算法计算量的 $2/m$. 所以说联合递推最小二辨识算法比子系统辨识算法计算量小.

3 部分信息向量耦合型多变量系统

下面研究部分信息向量耦合型(子系统参数向量独立、部分信息向量耦合)多变量系统辨识方法及其计算效率.

部分信息向量耦合型多变量系统的辨识模型为

$$y_i(t)=[\boldsymbol{\phi}^{\mathrm{T}}(t),\boldsymbol{\psi}_i^{\mathrm{T}}(t)]\boldsymbol{\theta}_i+v_i(t),$$

$$i=1,2,\cdots,m, \tag{18}$$

其中 $y_i(t)\in\mathbf{R}$ 为第 i 子系统输出,$v_i(t)\in\mathbf{R}$ 为第 i 子系统的干扰噪声(可以为是零均值、方差为 σ_i^2 的白噪声). 这个多变量系统的特征是,m 个辨识模型的每个子辨识模型拥有部分相同的子信息向量 $\boldsymbol{\phi}(t)\in\mathbf{R}^{n_0}$ 和部分不相同的子信息向量 $\boldsymbol{\psi}_i(t)\in\mathbf{R}^{n_i}$,每个子系统的参数向量 $\boldsymbol{\theta}_i\in\mathbf{R}^{n_0+n_i}$ 是不同的.

部分信息向量耦合型多变量系统的一个例子是3输入2输出系统(参见图1):

$$y_1(t)=G_{11}(z)u_1(t)+G_{12}(z)u_2(t)+v_1(t),$$

$$y_2(t)=G_{22}(z)u_2(t)+G_{23}(z)u_3(t)+v_2(t),$$

其中 $u_j(t)\in\mathbf{R}$ 为第 j 个输入,$G_{ij}(z)\in\mathbf{R}$ 为第 j 个输入 $u_j(t)$ 到第 i 个输出 $y_i(t)$ 的传递函数,为表述方便,这里假设为有限脉冲响应模型:

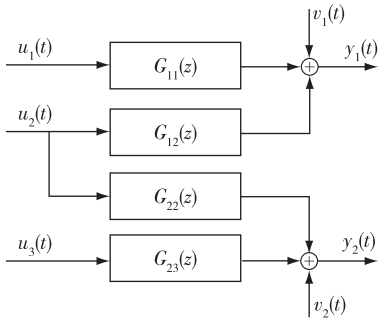


图1 部分信息向量耦合型多变量系统

Fig. 1 The multivariable system with the partially coupled information vectors

$$G_{ij}(z)=b_{ij}(1)z^{-1}+b_{ij}(2)z^{-2}+\cdots+b_{ij}(n_j)z^{-n_j}.$$

系统的两个输出可以写为

$$\begin{aligned}
y_1(t)&=G_{11}(z)u_1(t)+G_{12}(z)u_2(t)+v_1(t)=\\
&[b_{11}(1)z^{-1}+b_{11}(2)z^{-2}+\cdots+b_{11}(n_1)z^{-n_1}]u_1(t)+\\
&[b_{12}(1)z^{-1}+b_{12}(2)z^{-2}+\cdots+b_{12}(n_2)z^{-n_2}]u_2(t)+\\
&v_1(t)=b_{11}(1)u_1(t-1)+b_{11}(2)u_1(t-2)+\cdots+\\
&b_{11}(n_1)u_1(t-n_1)+b_{12}(1)u_2(t-1)+\\
&b_{12}(2)u_2(t-2)+\cdots+b_{12}(n_2)u_2(t-n_2)+\\
&v_1(t)=b_{12}(1)u_2(t-1)+b_{12}(2)u_2(t-2)+\cdots+\\
&b_{12}(n_2)u_2(t-n_2)+b_{11}(1)u_1(t-1)+\\
&b_{11}(2)u_1(t-2)+\cdots+b_{11}(n_1)u_1(t-n_1)+v_1(t)=\\
&[\boldsymbol{\phi}^{\mathrm{T}}(t),\boldsymbol{\psi}_1^{\mathrm{T}}(t)]\boldsymbol{\theta}_1+v_1(t),\\
y_2(t)&=G_{22}(z)u_2(t)+G_{23}(z)u_3(t)+v_2(t)=\\
&[b_{22}(1)z^{-1}+b_{22}(2)z^{-2}+\cdots+b_{22}(n_2)z^{-n_2}]u_2(t)+\\
&[b_{23}(1)z^{-1}+b_{23}(2)z^{-2}+\cdots+b_{23}(n_3)z^{-n_3}]u_3(t)+\\
&v_2(t)=b_{22}(1)u_2(t-1)+b_{22}(2)u_2(t-2)+\cdots+\\
&b_{22}(n_2)u_2(t-n_2)+b_{23}(1)u_3(t-1)+\\
&b_{23}(2)u_3(t-2)+\cdots+b_{23}(n_3)u_3(t-n_3)+v_2(t)=\\
&[\boldsymbol{\phi}^{\mathrm{T}}(t),\boldsymbol{\psi}_2^{\mathrm{T}}(t)]\boldsymbol{\theta}_2+v_2(t),
\end{aligned}$$

其中参数向量 $\boldsymbol{\theta}_i$、信息向量 $\boldsymbol{\varphi}(t)$ 和 $\boldsymbol{\psi}_i(t)$ 定义为

$$\begin{aligned}
\boldsymbol{\theta}_1&:=[b_{12}(1),b_{12}(2),\cdots,b_{12}(n_2),b_{11}(1),\\
&\quad b_{11}(2),\cdots,b_{11}(n_1)]^{\mathrm{T}}\in\mathbf{R}^{n_1+n_2},\\
\boldsymbol{\theta}_2&:=[b_{22}(1),b_{22}(2),\cdots,b_{22}(n_2),b_{23}(1),\\
&\quad b_{23}(2),\cdots,b_{23}(n_3)]^{\mathrm{T}}\in\mathbf{R}^{n_2+n_3},\\
\boldsymbol{\phi}(t)&:=[u_2(t-1),u_2(t-2),\cdots,u_2(t-n_2)]^{\mathrm{T}}\in\mathbf{R}^{n_2},\\
\boldsymbol{\psi}_1(t)&:=[u_1(t-1),u_1(t-2),\cdots,u_1(t-n_1)]^{\mathrm{T}}\in\mathbf{R}^{n_1},\\
\boldsymbol{\psi}_2(t)&:=[u_3(t-1),u_3(t-2),\cdots,u_3(t-n_3)]^{\mathrm{T}}\in\mathbf{R}^{n_3}.
\end{aligned}$$

3.1 子系统最小二乘辨识算法

令

$$\boldsymbol{\varphi}_i(t):=\begin{bmatrix}\boldsymbol{\phi}(t)\\ \boldsymbol{\psi}_i(t)\end{bmatrix}\in\mathbf{R}^{n_0+n_i}.$$

式(18)可以写为

$$y_i(t)=\boldsymbol{\varphi}_i^{\mathrm{T}}(t)\boldsymbol{\theta}_i+v_i(t),\quad i=1,2,\cdots,m. \tag{19}$$

设数据长度为 $L(L\gg n)$. 定义堆积输出向量 $\boldsymbol{Y}_i$,堆积信息矩阵 $\boldsymbol{H}_i$ 如下:

$$\boldsymbol{Y}_i:=\begin{bmatrix}y_i(1)\\ y_i(2)\\ \vdots\\ y_i(L)\end{bmatrix}\in\mathbf{R}^L,$$

$$\boldsymbol{H}_i:=\begin{bmatrix}\boldsymbol{\varphi}_i^{\mathrm{T}}(1)\\ \boldsymbol{\varphi}_i^{\mathrm{T}}(2)\\ \vdots\\ \boldsymbol{\varphi}_i^{\mathrm{T}}(L)\end{bmatrix}=\begin{bmatrix}\boldsymbol{\phi}^{\mathrm{T}}(1) & \boldsymbol{\psi}_i^{\mathrm{T}}(1)\\ \boldsymbol{\phi}^{\mathrm{T}}(2) & \boldsymbol{\psi}_i^{\mathrm{T}}(2)\\ \vdots & \vdots\\ \boldsymbol{\phi}^{\mathrm{T}}(L) & \boldsymbol{\psi}_i^{\mathrm{T}}(L)\end{bmatrix}\in\mathbf{R}^{L\times\mu_i},$$

$$\mu_i:=n_0+n_i.$$

定义和极小化准则函数:

$$J_2(\boldsymbol{\theta}_i):=\sum_{j=1}^{L}[y_i(j)-\boldsymbol{\varphi}_i^{\mathrm{T}}(j)\boldsymbol{\theta}_i]^2=\|\boldsymbol{Y}_i-\boldsymbol{H}_i\boldsymbol{\theta}_i\|^2,$$
$$i=1,2,\cdots,m,$$

给出部分信息向量耦合型多变量系统(9)参数向量 $\boldsymbol{\theta}_i$ 的子系统最小二乘(S-LS)辨识算法:

$$\hat{\boldsymbol{\theta}}_i=(\boldsymbol{H}_i^{\mathrm{T}}\boldsymbol{H}_i)^{-1}\boldsymbol{H}_i^{\mathrm{T}}\boldsymbol{Y}_i,\quad i=1,2,\cdots,m. \tag{20}$$

部分信息向量耦合型多变量系统最小二乘算法(20)的计算量如表2所示,其中乘法运算次数为

$$M_9:=\sum_{i=1}^{m}[\mu_i^3+\mu_i^2+(\mu_i^2+\mu_i)L]\text{flops},$$

加法运算次数为

$$A_9:=\sum_{i=1}^{m}[\mu_i^3-\mu_i^2-2\mu_i+(\mu_i^2+\mu_i)L]\text{flops},$$

算法的总 flop 数为

$$N_9:=M_9+A_9=\sum_{i=1}^{m}[2\mu_i^3-2\mu_i+(2\mu_i^2+2\mu_i)L]\text{flops}.$$

3.2 基于块矩阵求逆的最小二乘辨识算法

令

$$\boldsymbol{\Phi}:=\begin{bmatrix}\boldsymbol{\phi}^{\mathrm{T}}(1)\\ \boldsymbol{\phi}^{\mathrm{T}}(2)\\ \vdots\\ \boldsymbol{\phi}^{\mathrm{T}}(L)\end{bmatrix}\in\mathbf{R}^{L\times n_0},\quad \boldsymbol{\Psi}_i:=\begin{bmatrix}\boldsymbol{\psi}_i^{\mathrm{T}}(1)\\ \boldsymbol{\psi}_i^{\mathrm{T}}(2)\\ \vdots\\ \boldsymbol{\psi}_i^{\mathrm{T}}(L)\end{bmatrix}\in\mathbf{R}^{L\times n_i}.$$

则有

$$\boldsymbol{H}_i=[\boldsymbol{\Phi},\boldsymbol{\Psi}_i]\in\mathbf{R}^{L\times(n_0+n_i)}.$$

令 $\boldsymbol{S}:=\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\in\mathbf{R}^{n_0\times n_0}$,$\boldsymbol{R}_i:=\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Psi}_i\in\mathbf{R}^{n_0\times n_i}$,$\boldsymbol{T}_i:=\boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{\Psi}_i\in\mathbf{R}^{n_i\times n_i}$,定义数据乘积矩矩阵(data product moment matrix):

表 2　子系统最小二乘算法的计算量

Table 2　The computational efficiency of the subsystem least squares algorithm

变量	计算次序	乘法次数	加法次数
$\hat{\boldsymbol{\theta}}_i$	$\hat{\boldsymbol{\theta}}_i = \boldsymbol{S}'_i\boldsymbol{\beta}_i \in \mathbf{R}^{\mu_i}$	μ_i^2	$\mu_i(\mu_i-1)$
	$\boldsymbol{S}_i := \boldsymbol{H}_i^{\mathrm{T}}\boldsymbol{H}_i \in \mathbf{R}^{\mu_i\times\mu_i}$	$\mu_i^2 L$	$\mu_i^2(L-1)$
	$\boldsymbol{\beta}_i := \boldsymbol{H}_i^{\mathrm{T}}\boldsymbol{Y}_i \in \mathbf{R}^{\mu_i}$	$\mu_i L$	$\mu_i(L-1)$
	$\boldsymbol{S}'_i := \boldsymbol{S}_i^{-1} \in \mathbf{R}^{\mu_i\times\mu_i}$	μ_i^3	$\mu_i^3-\mu_i^2$
总数		$\sum_{i=1}^{m}[\mu_i^3+\mu_i^2+(\mu_i^2+\mu_i)L]$	$\sum_{i=1}^{m}[\mu_i^3-\mu_i^2-2\mu_i+(\mu_i^2+\mu_i)L]$
总 flop 数		$N_9 := \sum_{i=1}^{m}[2\mu_i^3-2\mu_i+(2\mu_i^2+2\mu_i)L]$	

$$\boldsymbol{S}_i := \boldsymbol{H}_i^{\mathrm{T}}\boldsymbol{H}_i = \begin{bmatrix}\boldsymbol{\Phi}^{\mathrm{T}}\\ \boldsymbol{\Psi}_i^{\mathrm{T}}\end{bmatrix}[\boldsymbol{\Phi},\boldsymbol{\Psi}_i] =$$

$$\begin{bmatrix}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} & \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Psi}_i\\ \boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{\Phi} & \boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{\Psi}_i\end{bmatrix} = \begin{bmatrix}\boldsymbol{S} & \boldsymbol{R}_i\\ \boldsymbol{R}_i^{\mathrm{T}} & \boldsymbol{T}_i\end{bmatrix} \in \mathbf{R}^{(n_0+n_i)\times(n_0+n_i)}.$$

应用块矩阵求逆引理 1,可得

$$\boldsymbol{S}_i^{-1} = \begin{bmatrix}\boldsymbol{S} & \boldsymbol{R}_i\\ \boldsymbol{R}_i^{\mathrm{T}} & \boldsymbol{T}_i\end{bmatrix}^{-1} =$$

$$\begin{bmatrix}\boldsymbol{S}-1+\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1} & -\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}\\ -\boldsymbol{Q}_i^{-1}\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1} & \boldsymbol{Q}_i^{-1}\end{bmatrix} \in \mathbf{R}^{(n_0+n_i)\times(n_0+n_i)},$$

$$\boldsymbol{Q}_i := \boldsymbol{T}_i - \boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1}\boldsymbol{R}_i \in \mathbf{R}^{n_i\times n_i}.$$

令 $\boldsymbol{\gamma}_i := \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i \in \mathbf{R}^{n_0}$, $\boldsymbol{\xi}_i := \boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{Y}_i \in \mathbf{R}^{n_i}$. 注意到 $\boldsymbol{S}=\boldsymbol{S}^{\mathrm{T}}$, $\boldsymbol{T}_i=\boldsymbol{T}_i^{\mathrm{T}}$, $\boldsymbol{Q}_i=\boldsymbol{Q}_i^{\mathrm{T}}$. 则式(20)参数估计可以表示为

$$\hat{\boldsymbol{\theta}}_i = \boldsymbol{S}_i^{-1}\boldsymbol{H}_i^{\mathrm{T}}\boldsymbol{Y}_i =$$

$$\begin{bmatrix}\boldsymbol{S}^{-1}+\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1} & -\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}\\ -\boldsymbol{Q}_i^{-1}\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1} & \boldsymbol{Q}_i^{-1}\end{bmatrix}\begin{bmatrix}\boldsymbol{\varphi}^{\mathrm{T}}\\ \boldsymbol{\Psi}_i^{\mathrm{T}}\end{bmatrix}\boldsymbol{Y}_i =$$

$$\begin{bmatrix}\boldsymbol{S}^{-1}+\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1} & -\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}\\ -\boldsymbol{Q}_i^{-1}\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1} & \boldsymbol{Q}_i^{-1}\end{bmatrix}\begin{bmatrix}(\boldsymbol{\varphi}^{\mathrm{T}}\boldsymbol{Y}_i)\\ (\boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{Y}_i)\end{bmatrix} =$$

$$\begin{bmatrix}(\boldsymbol{S}^{-1}+\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1})(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i)-\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}(\boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{Y}_i)\\ -\boldsymbol{Q}_i^{-1}\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i)+\boldsymbol{Q}_i^{-1}(\boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{Y}_i)\end{bmatrix} =$$

$$\begin{bmatrix}\boldsymbol{S}^{-1}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i)+\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{Q}_i^{-1}[\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i)-(\boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{Y}_i)]\\ -\boldsymbol{Q}_i^{-1}[\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i)-(\boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{Y}_i)]\end{bmatrix} =$$

$$\begin{bmatrix}\boldsymbol{S}^{-1}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i)-\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{\beta}_i\\ \boldsymbol{\beta}_i\end{bmatrix} = \begin{bmatrix}\boldsymbol{S}^{-1}(\boldsymbol{\gamma}_i-\boldsymbol{R}_i\boldsymbol{\beta}_i)\\ \boldsymbol{\beta}_i\end{bmatrix} =$$

$$\begin{bmatrix}\boldsymbol{S}^{-1}\boldsymbol{\gamma}_i-(\boldsymbol{S}^{-1}\boldsymbol{R}_i)\boldsymbol{\beta}_i\\ \boldsymbol{\beta}_i\end{bmatrix} \in \mathbf{R}^{n_0+n_i}, \tag{22}$$

其中

$$\begin{aligned}\boldsymbol{\beta}_i &:= -\boldsymbol{Q}_i^{-1}[\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i)-(\boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{Y}_i)] =\\ &-\boldsymbol{Q}_i^{-1}[(\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1})\boldsymbol{\gamma}_i-\boldsymbol{\xi}_i] =\\ &-\boldsymbol{Q}_i^{-1}[\boldsymbol{R}_i^{\mathrm{T}}(\boldsymbol{S}^{-1}\boldsymbol{\gamma}_i)-\boldsymbol{\xi}_i] \in \mathbf{R}^{n_i}.\end{aligned} \tag{23}$$

由以上各式,可以总结出估计部分信息向量耦合型多变量系统(18)参数向量 $\boldsymbol{\theta}_i$ 的基于块矩阵求逆的最小二乘辨识算法(简称 B-LS 算法):

$$\hat{\boldsymbol{\theta}}_i = \begin{bmatrix}\boldsymbol{S}^{-1}\boldsymbol{\gamma}_i-\boldsymbol{S}^{-1}\boldsymbol{R}_i\boldsymbol{\beta}_i\\ \boldsymbol{\beta}_i\end{bmatrix}, \quad i=1,2,\cdots,m, \tag{24}$$

$$\boldsymbol{\beta}_i = -\boldsymbol{Q}_i^{-1}(\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1}\boldsymbol{\gamma}_i-\boldsymbol{\xi}_i), \tag{25}$$

$$\boldsymbol{\gamma}_i := \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i, \quad \boldsymbol{\xi}_i = \boldsymbol{\psi}_i^{\mathrm{T}}\boldsymbol{Y}_i, \tag{26}$$

$$\boldsymbol{R}_i = \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Psi}_i, \quad \boldsymbol{T}_i = \boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{\Psi}_i, \tag{27}$$

$$\boldsymbol{Q}_i = \boldsymbol{T}_i - \boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1}\boldsymbol{R}_i, \tag{28}$$

$$\boldsymbol{Y}_i = [y_i(1),y_i(2),\cdots,y_i(L)]^{\mathrm{T}}, \tag{29}$$

$$\boldsymbol{\Psi}_i = [\boldsymbol{\psi}_i(1),\boldsymbol{\psi}_i(2),\cdots,\boldsymbol{\psi}_i(L)]^{\mathrm{T}}, \tag{30}$$

$$\boldsymbol{\Phi} = [\boldsymbol{\phi}(1),\boldsymbol{\phi}(2),\cdots,\boldsymbol{\phi}(L)]^{\mathrm{T}}, \tag{31}$$

$$\boldsymbol{S} = \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}. \tag{32}$$

基于块矩阵求逆的最小二乘算法(24)—(32)计算参数估计向量 $\hat{\boldsymbol{\theta}}_i$ 的步骤如下.

1) 收集观测数据 $y(t)$, $\boldsymbol{\phi}(t)$, $\boldsymbol{\psi}_1(t)$, $\boldsymbol{\psi}_2(t)$, $\cdots$, $\boldsymbol{\psi}_m(t)$: $t=1,2,3,\cdots,L$.

2) 置 $i=1$,用式(31)构造 $\boldsymbol{\Phi}$,用式(32)计算 $\boldsymbol{S}$.

3) 用式(29)构造 $\boldsymbol{Y}_i$,用式(30)构造 $\boldsymbol{\Psi}_i$.

4) 用式(26)计算 $\boldsymbol{\gamma}_i$ 和 $\boldsymbol{\xi}_i$,用式(27)计算 $\boldsymbol{R}_i$ 和 $\boldsymbol{T}_i$,用式(28)计算 $\boldsymbol{Q}_i$.

5) 用式(25)计算 $\boldsymbol{\beta}_i$,用式(24)计算参数估计 $\hat{\boldsymbol{\theta}}_i$.

6) 如果 $i=m$,就获得参数估计 $\hat{\boldsymbol{\theta}}_1,\hat{\boldsymbol{\theta}}_2,\cdots,\hat{\boldsymbol{\theta}}_m$;否则 i 增 1,转到第 3 步.

B-LS 辨识算法(24)—(32)计算参数估计向量 $\hat{\boldsymbol{\theta}}_i$ 的流程如图 2 所示.

基于块矩阵求逆的最小二乘算法(24)—(32)的计算量如表 3 所示,其中算法的乘法运算次数 M_{10} 和加法运算次数 A_{10} 分别为

$$M_{10} := [n_0^2(n_0+L)+\sum_{i=1}^{m}(n_i+1)\mu_i L+n_0^2 n_i+$$

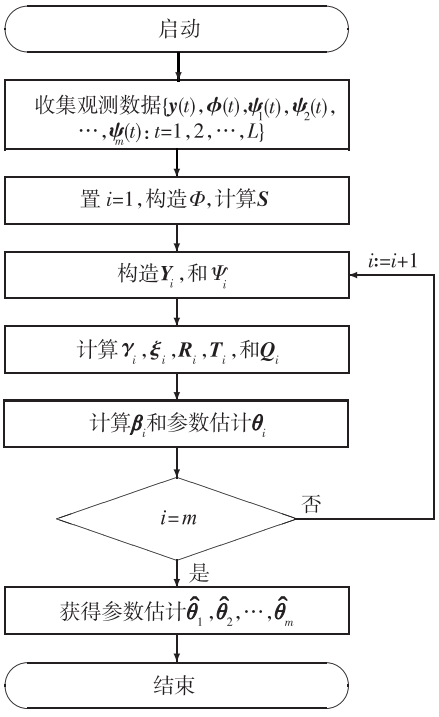


图 2　B-LS 算法计算参数估计向量 $\hat{\boldsymbol{\theta}}_i$ 的流程

Fig. 2　The flowchart of computing the parameter estimates $\hat{\boldsymbol{\theta}}_i$ in the B-LS algorithm

$$n_i^2\,\mu_i + \mu_i^2]\text{flops},$$

$$A_{10} := [n_0^2(n_0 + L - 2) + \sum_{i=1}^{m}(n_i + 1)\mu_i L + n_0^2 n_i + \mu_i^2 + (n_i^2 - 2n_i - 2)\mu_i]\text{flops},$$

算法的总 flop 数为

$$N_{10} := M_{10} + A_{10} = \{2n_0^2(n_0 + L - 1) + 2\sum_{i=1}^{m}[(n_i + 1)\mu_i L + n_0^2 n_i + (n_i^2 + n_0 - 1)\mu_i]\}\text{flops}.$$

为了比较部分信息向量耦合型多变量系统 LS 算法(20)与 B-LS 算法(24)—(32)的计算量,考虑多输入 $m=4$ 输出多变量系统,设 $n_0=6, n_i=6$,即每个子系统有 $\mu_i = n_0 + n_i = 12$ 个参数,取数据长度为 $L=1\,000$. LS 算法的乘法运算次数为 $M_9 = 631\,488$,加法运算次数为 $A_9 = 630\,240$,总 flop 数为 $N_9 = 1\,261\,728$;B-LS 算法的乘法运算次数为 $M_{10} = 375\,384$,加法运算次数为 $A_{10} = 374\,640$,总 flop 数为 $N_{10} = 750\,024$. 两个算法计算量之差为 $\Delta = N_9 - N_{10} = 511\,704$,可见 B-LS 算法比 LS 算法计算量小,也就是 B-LS 算法比 LS 算法计算效率提高了 $\Delta/N_9 = 40.55\%$.

下面的 Matlab 程序计算表 2 中 LS 算法和表 3 中 B-LS 算法的计算量,以及表 4 中 S-RLS 算法和表 5 中基于分解的多变量递推最小二乘(D-RLS)算法每步的计算量.

```
%————————————————————————*
% Filename:ComputationalEfficiency1.m *
% for multivariable identification algorithms *
%————————————————————————*
clear; format long g
m=4; % The number of outputs
n0=6; L=1000; % The data length
n=ones(m,1)*n0; % ni
```

表 3　基于块矩阵求逆的最小二乘算法的计算量

Table 3　The computational efficiency of the LS algorithm based on the block matrix inversion

变量	计算次序	乘法次数	加法次数
$\boldsymbol{S}$	$\boldsymbol{S}=\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\in\mathbf{R}^{n_0\times n_0}$	$n_0^2 L$	$n_0^2(L-1)$
$\boldsymbol{S}'$	$\boldsymbol{S}':=\boldsymbol{S}^{-1}\in\mathbf{R}^{n_0\times n_0}$	n_0^3	$n_0^3-n_0^2$
$\hat{\boldsymbol{\theta}}_i$	$\hat{\boldsymbol{\theta}}_i=\begin{bmatrix}\boldsymbol{S}^{-1}(\boldsymbol{\gamma}_i-\boldsymbol{R}_i\boldsymbol{\beta}_i)\\ \boldsymbol{\beta}_i\end{bmatrix}\in\mathbf{R}^{\mu_i}$	$n_0^2+n_0n_i$	$n_0^2+n_0n_i-n_0$
$\boldsymbol{\beta}_i$	$\boldsymbol{\beta}_i=-\boldsymbol{Q}_i^{-1}(\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}^{-1}\boldsymbol{\gamma}_i-\boldsymbol{\xi}_i)\in\mathbf{R}^{n_i}$	$n_0n_i+n_i^2$	$n_0n_i+n_i^2-n_i$
$\boldsymbol{Q}_i$	$\boldsymbol{Q}_i=\boldsymbol{T}_i-\boldsymbol{R}_i^{\mathrm{T}}(\boldsymbol{S}'\boldsymbol{R}_i)\in\mathbf{R}^{n_i\times n_i}$	$n_0n_i^2$	$n_0n_i^2$
$\boldsymbol{Q}_{i'}$	$\boldsymbol{Q}_i=\boldsymbol{Q}_i^{-1}\in\mathbf{R}^{n_i\times n_i}$	n_i^3	$n_i^3-n_i^2$
$(\boldsymbol{S}^{-1}\boldsymbol{R}_i)$	$\boldsymbol{S}^{-1}\boldsymbol{R}_i=\boldsymbol{S}'\boldsymbol{R}_i=(\boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{S}')^{\mathrm{T}}\in\mathbf{R}^{n_0\times n_i}$	$n_0^2\,n_i$	$(n_0^2-n_0)n_i$
$\boldsymbol{R}_i$	$\boldsymbol{R}_i=\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Psi}_i\in\mathbf{R}^{n_0\times n_i}$	n_0n_iL	$n_0n_i(L-1)$
$\boldsymbol{T}_i$	$\boldsymbol{T}_i=\boldsymbol{\Psi}^{\mathrm{T}}\boldsymbol{\Psi}_i\in\mathbf{R}^{n_i\times n_i}$	$n_i^2\,L$	$n_i^2(L-1)$
$\boldsymbol{\gamma}_i$	$\boldsymbol{\gamma}_i=\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{Y}_i\in\mathbf{R}^{n_0}$	n_0L	$n_0(L-1)$
$\boldsymbol{\xi}_i$	$\boldsymbol{xi}_i=\boldsymbol{\Psi}_i^{\mathrm{T}}\boldsymbol{Y}_i\in\mathbf{R}^{n_i}$	n_iL	$n_i(L-1)$
	总数	M_{10}	A_{10}
	总 flop 数	$2n_0^2(n_0+L-1)+2\sum_{i=1}^{m}[(n_i+1)\mu_iL+n_0^2n_i+\mu_i^2+(n_i^2-n_i-1)\mu_i]$	

```
mu = n0 + n;          % mui

% LS algorithm
M9 = sum( mu. ^3 + mu. ^2 + ( mu. ^2 + mu) * L) ;
A9 = sum( mu. ^3 - mu. ^2 -2 * mu + ( mu. ^2 + mu) * L) ;
N9 = sum(2 * mu. ^3 -2 * mu +2 * ( mu. ^2 + mu) * L) ;
LSalgorithm = [ M9, A9, N9, M9 + A9 - N9]

% B - LS algorithm
m10_1 = n0^2 * L + n0^3;

m10_2 = n0^2 + n0 * n + n0 * n + n. ^2;
m10_3 = n0 * n. ^2 + n. ^3 + n0^2 * n + n0 * n * L + n. ^2
 * L;
m10_4 = n0 * L + n * L;
M10 = m10_1 + sum( m10_2) + sum( m10_3) + sum
(m10_4) ;

a10_1 = n0^2 * ( L -1) + n0^3 - n0^2;

a10_2 = n0^2 + n0 * n + n0 * n + n. ^2 - n0 - n;
a10_3 = n0 * n. ^2 + n. ^3 - n. ^2 + ( n0^2 - n0) * n…
      + n0 * n * ( L -1) + n. ^2 * ( L -1) ;
a10_4 = n0 * ( L -1) + n * ( L -1) ;
A10 = a10_1 + sum( a10_2) + sum( a10_3) + sum( a10_4) ;

N10 = M10 + A10;
BLSalgorithm1 = [ M10, A10, N10]

M10 = n0^2 * ( n0 + L) ;
A10 = n0^2 * ( n0 + L -2) ;
N10 = 2 * n0^2 * ( n0 + L -1) ;
for  i = 1: m
     M10 = M10 + ( n( i) +1) * mu( i) * L + n0^2 * n( i)
          + n( i) ^2 * mu( i)…
     + mu( i) ^2;
     A10 = A10 + ( n( i) +1) * mu( i) * L + n0^2 * n( i)
          + mu( i) ^2…
     + ( n( i) ^2 -2 * n( i) -2) * mu( i) ;
     N10 = N10 +2 * ( n( i) +1) * mu( i) * L +2 * n0^2
          * n( i) +2 * mu( i) ^2…
     +2 * ( n( i) ^2 - n( i) -1) * mu( i) ;
end
BLSalgorithm2 = [ M10, A10, N10]
% BLSalgorithm1 - BLSalgorithm2
Delta = N9 - N10
Ratio = [ Delta/N9, Delta/N10] * 100

% S - RLS algorithm
M11 = sum(2 * mu. ^2 +4 * mu) ;
A11 = sum(2 * mu. ^2 +2 * mu) ;
N11 = sum(4 * mu. ^2 +6 * mu) ;

SRLSalgorithm = [ M11, A11, N11, N11 - M11 - A11]

% D - RLS algorithm
M12 =2 * n0^2 +2 * m * n0 +2 * n0 + sum(2 * n. ^2 +4 * n) ;
A12 =2 * n0^2 +2 * m * n0 + sum(2 * n. ^2 +2 * n) ;
N12 =4 * n0^2 +4 * m * n0 +2 * n0 + sum(4 * n. ^2 +6 * n) ;
DRLSalgorithm = [ M12, A12, N12, N12 - M12 - A12]

Delta = N11 - N12
N 11_N12 =4 * ( m -1) * n0^2 +8 * n0 * sum( n) +2 *
     n0 * ( m -1)
Ratio = [ Delta/N11, Delta/N12] * 100
```

尽管部分信息向量耦合型多变量系统的基于块矩阵求逆的最小二乘辨识算法(24)—(32),在形式上很类似于文献[11,39]中伪线性回归系统的基于块矩阵求逆的最小二乘迭代辨识算法,但是它们有本质区别,前者算法中下标 i 取值 $i=1,2,\cdots,m$(m 为多变量系统输出的数目),是一次完成最小二乘算法,用于白噪声干扰的多变量线性回归系统;后者算法中下标 k 是迭代变量,取值 $k=1,2,3,\cdots$,是最小二乘迭代辨识算法,用于有色噪声干扰伪线性回归标量系统或多变量系统.

3.3 子系统递推最小二乘辨识算法

考虑式(19)的辨识模型,重写如下:

$$y_i(t)=\boldsymbol{\varphi}_i^{\mathrm{T}}(t)\boldsymbol{\theta}_i+v_i(t),\quad i=1,2,\cdots,m, \tag{33}$$

$$\boldsymbol{\varphi}_i(t)=\begin{bmatrix}\boldsymbol{\phi}(t)\\ \boldsymbol{\psi}_i(t)\end{bmatrix}\in\mathbf{R}^{n},\quad \mu_i:=n_0+n_i. \tag{34}$$

定义和极小化准则函数:

$$J_3(\boldsymbol{\theta}_i):=\sum_{j=1}^{t}[y_i(j)-\boldsymbol{\varphi}_i^{\mathrm{T}}(j)\boldsymbol{\theta}_i]^2,\quad i=1,2,\cdots,m,$$

可以得到估计部分信息向量耦合型多变量系统的子系统递推最小二乘(S-RLS)辨识算法:

$$\hat{\boldsymbol{\theta}}_i(t)=\hat{\boldsymbol{\theta}}_i(t-1)+\boldsymbol{L}_i(t)[y_i(t)-\boldsymbol{\varphi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)],\quad i=1,2,\cdots,m, \tag{35}$$

$$\boldsymbol{L}_i(t)=\boldsymbol{P}_i(t-1)\boldsymbol{\varphi}_i(t)[1+\boldsymbol{\varphi}_i^{\mathrm{T}}(t)\boldsymbol{P}_i(t-1)\boldsymbol{\varphi}_i(t)]^{-1}, \tag{36}$$

$$\boldsymbol{P}_i(t)=\boldsymbol{P}_i(t-1)-\boldsymbol{L}_i(t)[\boldsymbol{\varphi}_i^{\mathrm{T}}(t)\boldsymbol{P}_i(t-1)],\quad \boldsymbol{P}_i(0)=p_0\boldsymbol{I}_{\mu_i}. \tag{37}$$

S-RLS 算法(35)—(37)每步的计算量为 $N_{11}=\sum_{i=1}^{m}(4\mu_i^2+6\mu_i)$flops,如表 4 所示[10].

3.4 基于辨识模型分解的递推最小二乘辨识算法

下面讨论基于辨识模型分解的递推最小二乘辨识算法及其计算效率,基于辨识模型分解的直接最小二乘辨识算法从略.

部分向量耦合型多变量系统辨识模型(33)可以等价写为下列形式:

$$y_i(t)=[\boldsymbol{\phi}^{\mathrm{T}}(t),\boldsymbol{\psi}_i^{\mathrm{T}}(t)]\begin{bmatrix}\boldsymbol{\theta}_i\\ \boldsymbol{\vartheta}_i\end{bmatrix}+v_i(t)=$$

$$\boldsymbol{\phi}^{\mathrm{T}}(t)\boldsymbol{\theta}_i+\boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{\vartheta}_i+v_i(t),\quad i=1,2,\cdots,m,\tag{38}$$

其中 $\boldsymbol{\phi}(t)\in\mathbf{R}^{n_0}$,$\boldsymbol{\psi}_i(t)\in\mathbf{R}^{n_i}$,$\boldsymbol{\theta}_i\in\mathbf{R}^{n_0}$是对应于耦合部分子信息向量$\boldsymbol{\phi}(t)$的参数向量,$\boldsymbol{\vartheta}_i\in\mathbf{R}^{n_i}$是对应于不同子信息向量$\boldsymbol{\psi}_i(t)$的参数向量.

定义两个中间变量:

$$y_{1i}(t):=y_i(t)-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{\vartheta}_i,\tag{39}$$

$$y_{2i}(t):=y_i(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\boldsymbol{\theta}_i.\tag{40}$$

则由(38)得到两个辨识模型:

$$y_{1i}(t):=\boldsymbol{\phi}^{\mathrm{T}}(t)\boldsymbol{\theta}_i+v_i(t),\tag{41}$$

$$y_{2i}(t):=\boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{\vartheta}_i+v_i(t).\tag{42}$$

设 $\hat{\boldsymbol{\theta}}_i(t)\in\mathbf{R}^{n_0}$ 是 $\boldsymbol{\theta}_i\in\mathbf{R}^{n_0}$ 在时刻 t 的估计,$\hat{\boldsymbol{\vartheta}}_i(t)\in\mathbf{R}^{n_0}$是 $\boldsymbol{\vartheta}_i\in\mathbf{R}^{n_i}$在时刻 t 的估计. 使用最小二乘原理[4],定义和极小化两个准则函数:

$$J_3(\boldsymbol{\theta}_i):=\sum_{j=1}^{t}[y_{1i}(j)-\boldsymbol{\phi}^{\mathrm{T}}(j)\boldsymbol{\theta}_i]^2,$$

$$J_{3i}(\boldsymbol{\vartheta}_i):=\sum_{j=1}^{t}[y_{2i}(j)-\boldsymbol{\psi}_i^{\mathrm{T}}(j)\boldsymbol{\vartheta}_i]^2.$$

可以得到下列递推最小二乘算法:

$$\hat{\boldsymbol{\theta}}_i(t)=\hat{\boldsymbol{\theta}}_i(t-1)+\boldsymbol{L}_0(t)[y_{1i}(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)],\quad i=1,2,\cdots,m,\tag{43}$$

$$\boldsymbol{L}_0(t)=\boldsymbol{P}_0(t-1)\boldsymbol{\phi}(t)[1+\boldsymbol{\phi}^{\mathrm{T}}(t)\boldsymbol{P}_0(t-1)\boldsymbol{\phi}(t)]^{-1},\tag{44}$$

$$\boldsymbol{P}_0(t)=[\boldsymbol{I}_{n_0}-\boldsymbol{L}_0(t)\boldsymbol{\phi}^{\mathrm{T}}(t)]\boldsymbol{P}_0(t-1),\quad \boldsymbol{P}_0(0)=p_0\boldsymbol{I}_{n_0},\tag{45}$$

$$\hat{\boldsymbol{\vartheta}}_i(t)=\hat{\boldsymbol{\vartheta}}_i(t-1)+\boldsymbol{L}_i(t)[y_{2i}(t)-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_i(t-1)],\tag{46}$$

$$\boldsymbol{L}_i(t)=\boldsymbol{P}_i(t-1)\boldsymbol{\psi}_i(t)[1+\boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{P}_i(t-1)\boldsymbol{\psi}_i(t)]^{-1},\tag{47}$$

$$\boldsymbol{P}_i(t)=[\boldsymbol{I}_{n_i}-\boldsymbol{L}_i(t)\boldsymbol{\psi}_i^{\mathrm{T}}(t)]\boldsymbol{P}_i(t-1),\quad \boldsymbol{P}_i(0)=p_0\boldsymbol{I}_{n_i}.\tag{48}$$

将式(39)和(40)分别代入式(43)和(46)得到

$$\hat{\boldsymbol{\theta}}_i(t)=\hat{\boldsymbol{\theta}}_i(t-1)+\boldsymbol{L}_0(t)[y_i(t)-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{\vartheta}_i-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)],$$

$$\hat{\boldsymbol{\vartheta}}_i(t)=\hat{\boldsymbol{\vartheta}}_i(t-1)+\boldsymbol{L}_i(t)[y_i(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\boldsymbol{\theta}_i-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_i(t-1)].$$

由于上两式中右边分别包含了未知参数向量 $\boldsymbol{\vartheta}_i$ 和 $\boldsymbol{\theta}_i$,所以不可能计算出参数估计 $\hat{\boldsymbol{\theta}}_i(t)$ 和 $\hat{\boldsymbol{\vartheta}}_i(t)$. 解决的办法是使用递阶辨识原理[25-27]:未知参数向量 $\boldsymbol{\vartheta}_i$ 和 $\boldsymbol{\theta}_i$ 分别用其前一时刻的参数估计 $\hat{\boldsymbol{\vartheta}}_i(t-1)$ 和 $\hat{\boldsymbol{\theta}}_i(t-1)$代替,得到

$$\hat{\boldsymbol{\theta}}_i(t)=\hat{\boldsymbol{\theta}}_i(t-1)+\boldsymbol{L}_0(t)[y_i(t)-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_i(t-1)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)]=\hat{\boldsymbol{\theta}}_i(t-1)+\boldsymbol{L}_0(t)e_i(t),\tag{49}$$

$$\hat{\boldsymbol{\vartheta}}_i(t)=\hat{\boldsymbol{\vartheta}}_i(t-1)+\boldsymbol{L}_i(t)[y_i(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_i(t-1)]=\hat{\boldsymbol{\vartheta}}_i(t-1)+\boldsymbol{L}_i(t)e_i(t),\tag{50}$$

其中

$$e_i(t):=y_i(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_i(t-1)\in\mathbf{R}.\tag{51}$$

定义输出向量$\boldsymbol{y}(t)$和信息向量$\boldsymbol{e}(t)$如下:

$$\boldsymbol{y}(t):=[y_1(t),y_2(t),\cdots,y_m(t)]^{\mathrm{T}}\in\mathbf{R}^m,$$

$$\boldsymbol{e}(t):=[e_1(t),e_2(t),\cdots,e_m(t)]^{\mathrm{T}}\in\mathbf{R}^m.$$

设 $\hat{\boldsymbol{\theta}}(t):=[\hat{\boldsymbol{\theta}}_1(t),\hat{\boldsymbol{\theta}}_2(t),\cdots,\hat{\boldsymbol{\theta}}_m(t)]\in\mathbf{R}^{n_0\times m}$是 $\boldsymbol{\theta}:=[\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_m]\in\mathbf{R}^{n_0\times m}$在时刻 t 的估计,细胞数组 $\hat{\boldsymbol{\vartheta}}(t):=\{\hat{\boldsymbol{\vartheta}}_1(t),\hat{\boldsymbol{\vartheta}}_2(t),\cdots,\hat{\boldsymbol{\vartheta}}_m(t)\}$是 $\boldsymbol{\vartheta}:=\{\boldsymbol{\vartheta}_1,\boldsymbol{\vartheta}_2,\cdots,\boldsymbol{\vartheta}_m\}$在时刻 t 的估计,这细胞数组共有 m 列,其第 i 列的维数为 n_i. 使用 $e_i(t)$的定义,有

$$\boldsymbol{e}(t)=\begin{bmatrix}y_1(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_1(t-1)-\boldsymbol{\psi}_1^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_1(t-1)\\ y_2(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_2(t-1)-\boldsymbol{\psi}_2^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_2(t-1)\\ \vdots\\ y_m(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_m(t-1)-\boldsymbol{\psi}_m^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_m(t-1)\end{bmatrix}=$$

表 4 S-RLS 算法每步的计算量

Table 4 The computational efficiency of the S-RLS algorithm for each recursion

变量	计算次序	乘法次数	加法次数
$\hat{\boldsymbol{\theta}}_i(t)$	$\hat{\boldsymbol{\theta}}_i(t)=\hat{\boldsymbol{\theta}}_i(t-1)+\boldsymbol{L}_i(t)e_i(t)\in\mathbf{R}^{\mu_i}$	μ_i	μ_i
	$e_i(t):=y_i(t)-\boldsymbol{\varphi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)\in\mathbf{R}$	μ_i	μ_i
$\boldsymbol{L}_i(t)$	$\boldsymbol{L}_i(t)=\boldsymbol{\zeta}_i(t)/[1+\boldsymbol{\varphi}_i^{\mathrm{T}}(t)\boldsymbol{\zeta}_i(t)]\in\mathbf{R}^{\mu_i}$	$2\mu_i$	μ_i
	$\boldsymbol{\zeta}_i(t):=\boldsymbol{P}_i(t-1)\boldsymbol{\varphi}_i(t)\in\mathbf{R}^{\mu_i}$	μ_i^2	$(\mu_i-1)\mu_i$
$\boldsymbol{P}_i(t)$	$\boldsymbol{P}_i(t)=\boldsymbol{P}_i(t-1)-\boldsymbol{L}_i(t)\boldsymbol{\zeta}_i^{\mathrm{T}}(t)\in\mathbf{R}^{\mu_i\times\mu_i}$	μ_i^2	μ_i^2
	总数	$\boldsymbol{M}_{11}:=\sum_{i=1}^{m}(2\mu_i^2+4\mu_i)$	$\boldsymbol{A}_{11}:=\sum_{i=1}^{m}(2\mu_i^2+2\mu_i)$
	总 flop 数	$\boldsymbol{N}_{11}:=\sum_{i=1}^{m}(4\mu_i^2+6\mu_i)$	

$$\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}^{\mathrm{T}}(t-1)\boldsymbol{\phi}(t)-\boldsymbol{\eta}(t)\in\mathbf{R}^{m},\tag{52}$$

其中

$$\boldsymbol{\eta}(t):=\begin{bmatrix}\boldsymbol{\psi}_1^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_1(t-1)\\ \boldsymbol{\psi}_2^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_2(t-1)\\ \vdots\\ \boldsymbol{\psi}_m^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_m(t-1)\end{bmatrix}\in\mathbf{R}^{m}.\tag{53}$$

当 $i=1,2,\cdots,m$ 时,式(49)可写成式(54)的紧凑形式,联立式(44)—(45),(47)—(48)和式(50)—(53),可得部分信息向量耦合型多变量系统的基于分解的递推最小二乘(D-RLS)辨识算法:

$$\hat{\boldsymbol{\theta}}(t)=\hat{\boldsymbol{\theta}}(t-1)\boldsymbol{L}_0(t)\boldsymbol{e}^{\mathrm{T}}(t),\hat{\boldsymbol{\theta}}(0)=\mathbf{1}_{n_0\times m}/p_0,\tag{54}$$

$$\boldsymbol{L}_0(t)=\boldsymbol{P}_0(t-1)\boldsymbol{\phi}(t)[1+\boldsymbol{\phi}^{\mathrm{T}}(t)\boldsymbol{P}_0(t-1)\boldsymbol{\phi}(t)]^{-1},\tag{55}$$

$$\boldsymbol{P}_0(t)=\boldsymbol{P}_0(t-1)-\boldsymbol{L}_0(t)[\boldsymbol{\phi}^{\mathrm{T}}(t)\boldsymbol{P}_0(t-1)],$$
$$\boldsymbol{P}_0(0)=p_0\boldsymbol{I}_{n_0},\tag{56}$$

$$\hat{\boldsymbol{\vartheta}}_i(t)=\hat{\boldsymbol{\vartheta}}_i(t-1)+\boldsymbol{L}_i(t)e_i(t),i=1,2,\cdots,m,$$
$$\hat{\boldsymbol{\vartheta}}_i(0)=\mathbf{1}_{n_i}/p_0,\tag{57}$$

$$\boldsymbol{L}_i(t)=\boldsymbol{P}_i(t-1)\boldsymbol{\psi}_i(t)[1+\boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{P}_i(t-1)\boldsymbol{\psi}_i(t)]^{-1},\tag{58}$$

$$\boldsymbol{P}_i(t)=\boldsymbol{P}_i(t-1)-\boldsymbol{L}_i(t)[\boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{P}_i(t-1)],$$
$$\boldsymbol{P}_i(0)=p_0\boldsymbol{I}_{n_i},\tag{59}$$

$$\boldsymbol{e}(t)=[e_1(t),e_2(t),\cdots,e_m(t)]^{\mathrm{T}}=$$
$$\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}^{\mathrm{T}}(t-1)\boldsymbol{\phi}(t)-\boldsymbol{\eta}(t),\tag{60}$$

$$\boldsymbol{\eta}(t)=[\boldsymbol{\psi}_1^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_1(t-1),\boldsymbol{\psi}_2^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_2(t-1),\cdots,$$
$$\boldsymbol{\psi}_m^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_m(t-1)]^{\mathrm{T}},\tag{61}$$

$$e_i(t)=y_i(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_i(t-1).\tag{62}$$

式(57)可写成更紧凑的形式:

$$\hat{\boldsymbol{\vartheta}}(t)=\hat{\boldsymbol{\vartheta}}(t-1)+\{\boldsymbol{L}_1(t)e_1(t),\boldsymbol{L}_2(t)e_2(t),$$
$$\cdots,\boldsymbol{L}_i(t)e_m(t)\},\hat{\boldsymbol{\vartheta}}(0)=\{\mathbf{1}\}/p_0.$$

部分信息向量耦合型多变量系统 D-RLS 辨识算法(54)—(62)计算参数估计矩阵 $\hat{\boldsymbol{\theta}}(t)$ 和 $\hat{\boldsymbol{\vartheta}}_i(t)$ 的步骤如下.

1) 初始化 $t=1$,置初值:$\hat{\boldsymbol{\theta}}(0)=\mathbf{1}_{n_0\times m}/p_0$,$\boldsymbol{P}_0(0)=p_0\boldsymbol{I}_{n_0}$,$\hat{\boldsymbol{\vartheta}}_i(0)=\mathbf{1}_{n_i}/p_0$,$\boldsymbol{P}_i(0)=p_0\boldsymbol{I}_{n_i}$,$i=1,2,\cdots,m$,$p_0=10^6$.

2) 收集观测数据 $y(t)$,$\boldsymbol{\phi}(t)$,$\boldsymbol{\psi}_1(t)$,$\boldsymbol{\psi}_2(t)$,$\cdots$,$\boldsymbol{\psi}_m(t)$.

3) 置 $i=1$,用式(55)计算 $\boldsymbol{L}_0(t)$,用式(56)计算 $\boldsymbol{P}_0(t)$,用式(61)和(60)分别计算 $\boldsymbol{\eta}(t)$ 和 $\boldsymbol{e}(t)$,用式(54)刷新参数估计矩阵 $\hat{\boldsymbol{\theta}}(t)$.

4) 用式(58)计算 $\boldsymbol{L}_i(t)$,用式(59)计算 $\boldsymbol{P}_i(t)$,用式(57)刷新参数估计向量 $\hat{\boldsymbol{\vartheta}}_i(t)$.

5) 如果 $i<m$,i 增 1,转到第 4 步;否则 t 增 1,转到第 2 步.

部分信息向量耦合型多变量系统 D-RLS 辨识算法(54)—(62)计算参数估计矩阵 $\hat{\boldsymbol{\theta}}(t)$ 和参数向量 $\hat{\boldsymbol{\vartheta}}_i(t)$ 的流程如图 3 所示. D-RLS 辨识算法(54)—(62)的每步计算量如表 5 所示,其中算法的乘法运算次数 M_{12} 和加法运算次数 A_{12} 分别为

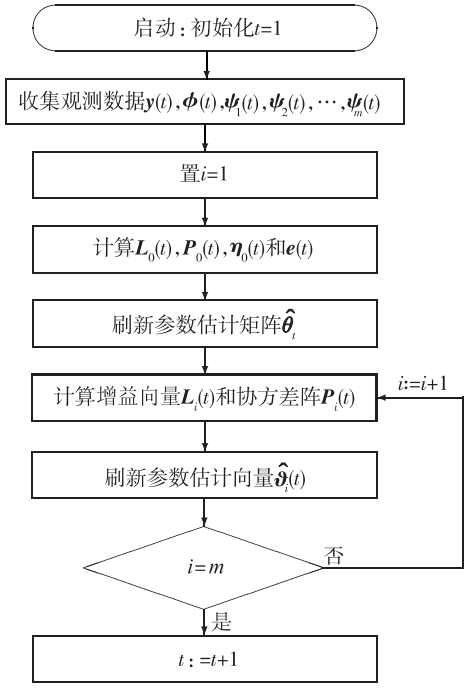


图 3 D-RLS 算法计算参数估计向量 $\hat{\boldsymbol{\theta}}(t)$ 和 $\hat{\boldsymbol{\vartheta}}_i(t)$ 的流程

Fig. 3 The flowchart of computing the parameter estimates $\hat{\boldsymbol{\theta}}(t)$ and $\hat{\boldsymbol{\vartheta}}_i(t)$ in the D-RLS algorithm

$$M_{12}:=[2n_0^2+2mn_0+2n_0+\sum_{i=1}^{m}(2n_i^2+4n_i)]\text{flops},$$

$$A_{12}:=[2n_0^2+2mn_0+\sum_{i=1}^{m}(2n_i^2+2n_i)]\text{flops},$$

算法的总 flop 数为

$$N_{12}:=M_{12}+A_{12}=$$
$$[4n_0^2+4mn_0+2n_0+\sum_{i=1}^{m}(4n_i^2+6n_i)]\text{flops}.$$

因此,部分信息向量耦合型多变量系统 D-RLS 辨识算法(54)—(62)比 S-RLS 算法(35)—(37)的计算量小:

$$N_{11}-N_{12}=\sum_{i=1}^{m}(4\mu_i^2+6\mu_i)-$$
$$\left[4n_0^2+4mn_0+2n_0+\sum_{i=1}^{m}(4n_i^2+6n_i)\right]=$$
$$\sum_{i=1}^{m}[4(n_0+n_i)^2+6(n_0+n_i)]-$$

表 5　基于分解的多变量递推最小二乘算法的计算量

Table 5　The computational efficiency of the multivariable RLS algorithm based on the decomposition

变量	计算次序	乘法次数	加法次数
$\hat{\boldsymbol{\theta}}(t)$	$\hat{\boldsymbol{\theta}}(t)=\hat{\boldsymbol{\theta}}(t-1)+\boldsymbol{L}_0(t)\boldsymbol{e}^{\mathrm{T}}(t)\in\mathbf{R}^{n_0\times m}$	mn_0	mn_0
	$\boldsymbol{e}(t)=[e_i(t)]=\boldsymbol{y}(t)-\hat{\boldsymbol{\theta}}^{\mathrm{T}}(t)\boldsymbol{\phi}(t)-\boldsymbol{\eta}(t)\in\mathbf{R}^m$	mn_0	mn_0+m
	$\boldsymbol{\eta}(t)=[\boldsymbol{\psi}_1^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_1(t-1),\cdots,\boldsymbol{\psi}_m^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_m(t-1)]^{\mathrm{T}}\in\mathbf{R}^m$	$n_1+\cdots+n_m$	$n_1+\cdots+n_m-m$
$\boldsymbol{L}_0(t)$	$\boldsymbol{L}_0(t):=\boldsymbol{\zeta}_0(t)/[1+\boldsymbol{\phi}^{\mathrm{T}}(t)\boldsymbol{\zeta}_0(t)]\in\mathbf{R}^{n_0}$	$2n_0$	n_0
	$\boldsymbol{\zeta}_0(t):=\boldsymbol{P}_0(t-1)\boldsymbol{\phi}(t)\in\mathbf{R}^{n_0}$	n_0^2	$n_0^2-n_0$
$\boldsymbol{P}_0(t)$	$\boldsymbol{P}_0(t)=\boldsymbol{P}_0(t-1)-\boldsymbol{L}_0(t)\boldsymbol{\zeta}_0^{\mathrm{T}}(t)\in\mathbf{R}^{n_0\times n_0}$	n_0^2	n_0^2
$\hat{\boldsymbol{\vartheta}}_i(t)$	$\hat{\boldsymbol{\vartheta}}_i(t)=\hat{\boldsymbol{\vartheta}}_i(t-1)+\boldsymbol{L}_i(t)e_i(t)\in\mathbf{R}^{n_i}$	n_i	n_i
$\boldsymbol{L}_i(t)$	$\boldsymbol{L}_i(t):=\boldsymbol{\zeta}_i(t)/[1+\boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{\zeta}_i(t)]\in\mathbf{R}^{n_i}$	$2n_i$	n_i
	$\boldsymbol{\zeta}_i(t):=\boldsymbol{P}_i(t-1)\boldsymbol{\psi}_i(t)\in\mathbf{R}^{n_i}$	n_i^2	$n_i^2-n_i$
$\boldsymbol{P}_i(t)$	$\boldsymbol{P}_i(t)=\boldsymbol{P}_i(t-1)-\boldsymbol{L}_i(t)\boldsymbol{\zeta}_i^{\mathrm{T}}(t)\in\mathbf{R}^{n_i\times n_i}$	n_i^2	n_i^2
	$e_i(t)=y_i(t)-\boldsymbol{\phi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_i(t-1)-\boldsymbol{\psi}_i^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_i(t-1)\in\mathbf{R}$	n_0+n_i	n_0+n_i
	总数	M_{12}	A_{12}
	总 flop 数	$N_{12}:=4n_0^2+4mn_0+2n_0+\sum_{i=1}^{m}(4n_i^2+6n_i)$	

$$\left[4n_0^2+4mn_0+2n_0+\sum_{i=1}^{m}(4n_i^2+6n_i)\right]=$$

$$[4(m-1)n_0^2+8n_0(n_1+n_2+\cdots+n_m)+2n_0(m-1)]\text{flops}.$$

对于多输入 $m=4$ 输出部分信息向量耦合型多变量系统,设 $n_0=6$,$n_i=6$,S-RLS 算法每步的乘法运算次数为 $M_{11}=1\ 344$,加法运算次数为 $A_{11}=1\ 248$,总 flop 数为 $N_{11}=2\ 592$;D-RLS 算法每步的乘法运算次数为 $M_{12}=516$,加法运算次数为 $A_{12}=456$,总 flop 数为 $N_{12}=972$. 两个算法计算量之差为 $\Delta=N_{11}-N_{12}=1\ 620$,可见 D-RLS 算法比 SLS 算法计算量小,也就是 D-RLS 算法比 S-RLS 算法效率提高了 $\Delta/N_{11}=62.5\%$.

4　结语

多变量系统可以按照输出的数目分解为多个子系统. 根据子系统辨识模型信息向量之间的关系和参数向量之间的关系,可以分为信息向量耦合型多变量系统、部分信息向量耦合型多变量系统、参数向量耦合型多变量系统、部分参数向量耦合型多变量系统等. 针对信息向量耦合型多变量系统和部分信息向量耦合型多变量系统,研究和提出了子系统(递推)最小二乘辨识、基于块矩阵求逆的最小二乘辨识算法和基于辨识模型分解的递推最小二乘辨识算法. 提出的方法可以推广到参数向量耦合型多变量系统. 进一步可研究滤波辨识算法、两阶段辨识算法或分解辨识算法、牛顿递推辨识算法和牛顿迭代辨识算法的计算效率.

参考文献

References

[1] 丁锋. 系统辨识新论[M]. 北京:科学出版社,2013
DING Feng. System identification:New theory and methods [M]. Beijing:Science Press,2013

[2] 丁锋. 系统辨识(1):辨识导引[J]. 南京信息工程大学学报:自然科学版,2011,3(1):1-22
DING Feng. System identification. Part A:Introduction to the identification [J]. Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2011,3(1):1-22

[3] 丁锋. 系统辨识(2):系统描述的基本模型[J]. 南京信息工程大学学报:自然科学版,2011,3(2):97-117
DING Feng. System identification. Part B:Basic models for system description [J]. Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2011,3(2):97-117

[4] 丁锋. 系统辨识(3):辨识精度与辨识基本问题[J]. 南京信息工程大学学报:自然科学版,2011,3(3):193-226
DING Feng. System identification. Part C:Identification accuracy and basic problems [J]. Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2011,3(3):193-226

[5] 丁锋. 系统辨识(4):辅助模型辨识思想与方法[J]. 南京信息工程大学学报:自然科学版,2011,3(4):289-318
DING Feng. System identification. Part D:Auxiliary model identification idea and methods [J]. Journal of Nanjing University of Information Science and Technology:Natural Science Edition,2011,3(4):289-318

[6] 丁锋. 系统辨识(5):迭代搜索原理与辨识方法[J]. 南京信息工程大学学报:自然科学版,2011,3(6):481-510
DING Feng. System identification. Part E:Iterative search

principle and identification methods [J]. Journal of Nanjing University of Information Science and Technology: Natural Science Edition, 2011, 3(6): 481-510

[7] 丁锋. 系统辨识(6):多新息辨识理论与方法[J]. 南京信息工程大学学报:自然科学版, 2012, 4(1): 1-28
DING Feng. System identification. Part F: Multi-innovation identification theory and methods [J]. Journal of Nanjing University of Information Science and Technology: Natural Science Edition, 2012, 4(1): 1-28

[8] 丁锋. 系统辨识(7):递阶辨识原理与方法[J]. 南京信息工程大学学报:自然科学版, 2012, 4(2): 97-124
DING Feng. System identification. Part G: Hierarchical identification principle and methods [J]. Journal of Nanjing University of Information Science and Technology: Natural Science Edition, 2012, 4(2): 97-124

[9] 丁锋. 系统辨识(8):耦合辨识概念与方法[J]. 南京信息工程大学学报:自然科学版, 2012, 4(3): 193-212
DING F. System identification. Part H: Coupling identification concept and methods [J]. Journal of Nanjing University of Information Science and Technology: Natural Science Edition, 2012, 4(3): 193-212

[10] 丁锋. 辨识方法的计算效率(1):递推算法[J]. 南京信息工程大学学报:自然科学版, 2012, 4(4): 289-300
DING Feng. Computational efficiency of the identification methods. Part A: Recursive algorithms [J]. Journal of Nanjing University of Information Science and Technology: Natural Science Edition, 2012, 4(4): 289-300

[11] 丁锋. 辨识方法的计算效率(2):迭代算法[J]. 南京信息工程大学学报:自然科学版, 2012, 4(5): 385-401
DING Feng. Computational efficiency of the identification methods. Part B: Iterative algorithms [J]. Journal of Nanjing University of Information Science and Technology: Natural Science Edition, 2012, 4(5): 385-401

[12] 丁锋, 杨慧中, 刘飞. 弱条件下随机梯度算法性能分析[J]. 中国科学:信息科学, 2008, 38(12): 2173-2184

[13] Ding F, Yang H Z, Liu F. Performance analysis of stochastic gradient algorithms under weak conditions [J]. Science in China Series: Information Sciences, 2008, 51(9): 1269-1280

[14] Ding F, Liu X P, Liu G. Identification methods for Hammerstein nonlinear systems [J]. Digital Signal Processing, 2011, 21(2): 215-238

[15] Ding F, Chen T. Combined parameter and output estimation of dual-rate systems using an auxiliary model [J]. Automatica, 2004, 40(10): 1739-1748

[16] Ding F, Chen T. Parameter estimation of dual-rate stochastic systems by using an output error method [J]. IEEE Transactions on Automatic Control, 2005, 50(9): 1436-1441

[17] Ding F, Ding J. Least squares parameter estimation with irregularly missing data [J]. International Journal of Adaptive Control and Signal Processing, 2010, 24(7): 540-553

[18] Ding F, Liu G, Liu X P. Parameter estimation with scarce measurements [J]. Automatica, 2011, 47(8): 1646-1655

[19] Ding F, Liu X P, Liu G. Gradient based and least-squares based iterative identification methods for OE and OEMA systems [J]. Digital Signal Processing, 2010, 20(3): 664-677

[20] Ding F, Liu Y J, Bao B. Gradient based and least squares based iterative estimation algorithms for multi-input multi-output systems [J]. Journal of Systems and Control Engineering, 2012, 226(1): 43-55

[21] Ding F, Chen T. Performance analysis of multi-innovation gradient type identification methods [J]. Automatica, 2007, 43(1): 1-14

[22] Ding F, Liu X P, Liu G. Multi-innovation least squares identification for system modeling [J]. IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics, 2010, 40(3): 767-778

[23] Ding F, Liu X P, Liu G. Auxiliary model based multi-innovation extended stochastic gradient parameter estimation with colored measurement noises [J]. Signal Processing, 2009, 89(10): 1883-1890

[24] Ding F. Several multi-innovation identification methods [J]. Digital Signal Processing, 2010, 20(4): 1027-1039

[25] Ding F, Chen T. Hierarchical gradient-based identification of multivariable discrete-time systems [J]. Automatica, 2005, 41(2): 315-325

[26] Ding F, Chen T. Hierarchical least squares identification methods for multivariable systems [J]. IEEE Transactions on Automatic Control, 2005, 50(3): 397-402

[27] Ding F, Chen T. Hierarchical identification of lifted state-space models for general dual-rate systems [J]. IEEE Transactions on Circuits and Systems. I: Regular Papers, 2005, 52(6): 1179-1187

[28] Ding F, Qiu L, Chen T. Reconstruction of continuous-time systems from their non-uniformly sampled discrete-time systems [J]. Automatica, 2009, 45(2): 324-332

[29] Han H Q, Xie L, Ding F, et al. Hierarchical least squares based iterative identification for multivariable systems with moving average noises [J]. Mathematical and Computer Modelling, 2010, 51(9-10): 1213-1220

[30] Zhang Z N, Ding F, Liu X G. Hierarchical gradient based iterative parameter estimation algorithm for multivariable output error moving average systems [J]. Computers and Mathematics with Applications, 2011, 61(3): 672-682

[31] Ding J, Ding F, Liu X P, et al. Hierarchical least squares identification for linear SISO systems with dual-rate sampled-data [J]. IEEE Transactions on Automatic Control, 2011, 56(11): 2677-2683

[32] Ding F, Liu G, Liu X P. Partially coupled stochastic gradient identification methods for non-uniformly sampled systems [J]. IEEE Transactions on Automatic Control, 2010, 55(8): 1976-1981

[33] Ding F. Coupled least squares identification for multivariable systems [J]. IET Control Theory and Applications, 2013, 7(x)

[34] Ding F. Hierarchical multi-innovation stochastic gradient algorithm for Hammerstein nonlinear system modeling [J]. Applied Mathematical Modelling, 2013, 37(4): 1694-1704

[35] Ding F. Two-stage least squares based iterative estimation algorithm for CARARMA system modeling [J]. Applied Mathematical Modelling, 2013. http://dx.doi.org/10.1016/j.apm.2012.10.014

[36] Ding F,Liu X G,Chu J. Gradient based and least squares based iterative algorithms for Hammerstein systems using the hierarchical identification principle [J]. IET Control Theory and Applications,2013,7(x)
[37] Golub G H,Van Loan C F. Matrix Computations [M], 3rd Ed. Baltimore, MD: Johns Hopkins University Press,1996
[38] 丁锋,谢新民. 线性多变量系统的联合辨识算法[J]. 控制理论与应用,1992,9(5):545-550
DING Feng, XIE Xinmin. Combined identification algorithms for linear multivariable systems [J]. Control Theory and Applications,1992,9(5):545-550
[39] Hu H Y,Ding F. An iterative least squares estimation algorithm for controlled moving average systems based on matrix decomposition. Applied Mathematics Letters, 2012,25(12):2332-2338

Computational efficiency of the identification methods. Part C: Coupled information vector algorithms

DING Feng[1,2,3]

1 School of Internet of Things Engineering,Jiangnan University,Wuxi 214122

2 Control Science and Engineering Research Center,Jiangnan University,Wuxi 214122

3 Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education),Jiangnan University,Wuxi 214122

Abstract This paper discusses the computational efficiency of the determinant,the matrix inversion and the block matrix inversion. For the multivariable systems with coupled information vectors,we study the subsystem recursive least squares (RLS) identification algorithm and the joint RLS identification agorithm with less computation are studied. For the multivariable systems the partially coupled information vectors,the subsystem least squares identification algorithm and the block matrix inversion based least squares identification algorithm are presented and the subsystem RLS identification algorithm and the identification model decomposition based RLS identification algorithm are proposed. Finally,the comptutational efficiency of the proposed algorithms is analyzed.

Key words recursive identification;iterative identification; parameter estimation; FIR model;equation error model; CAR model; CARMA model; CARAR model;CARARMA model; output error model; OEMA model; OEAR model;auxiliary model identification; multi-innovation identification;hierarchical identification; coupled identification

编者按:江南大学的丁锋教授从2011年第1期到2012年第6期在本刊刊登了关于系统辨识方面的连载论文共11篇,其中前8篇已由本刊结集编印成《系统辨识论文连载文集》. 以此蓝本为基础,经丁锋教授补充,形成了《系统辨识新论》一书. 该书将由科学出版社出版. 顾名思义,《系统辨识新论》主要讨论系统辨识的新思想、新理论、新原理、新概念与辨识新方法. 以后,本刊将继续刊登系统辨识方面的连载论文.