控制与决策
Control and Decision

结构未知系统降维辨识方法-----变量消去算法

陈晶, 程连元, 李俊红, 朱全民

结构未知系统降维辨识方法——变量消去算法

陈 晶[1†], 程连元[1], 李俊红[2], 朱全民[3]

(1. 江南大学 理学院，江苏 无锡 214122；2. 南通大学 电气工程学院，江苏 南通 226019；
3. 西英格兰大学 工业设计和数学系，英国 布里斯托 BS16 1QY)

摘 要: 针对结构未知的系统提出一种新的降维辨识方法. 借助核函数方法,利用一个高维Volterra模型逼近未知系统. 由于Volterra模型未知参数维数较高,为避免高阶矩阵求逆和求特征值,提出变量消去算法,将高维系统的辨识问题转化为两个低维系统辨识问题. 通过理论证明采用降维算法后降维系统信息矩阵条件数变小,参数收敛速度得到提高. 进一步引入Aitken 加速方法提高算法收敛速度,增强算法对步长的鲁棒特性. 最后通过仿真例子验证所提出方法的有效性.

关键词: 参数估计；降维算法；变量消去算法；Volterra模型；结构未知系统；矩阵条件数

中图分类号: TP273 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2022.0596

引用格式: 陈晶,程连元,李俊红,等. 结构未知系统降维辨识方法——变量消去算法[J]. 控制与决策, 2024, 39(1): 171-179.

A reduced-dimension identification algorithm for systems with unknown structures: Variable elimination algorithm

CHEN Jing[1†], *CHENG Lian-yuan*[1], *LI Jun-hong*[2], *ZHU Quan-min*[3]

(1. School of Science，Jiangnan University，Wuxi 214122，China；2. School of Electrical Engineering，Nantong University，Nantong 226019，China；3. Department of Engineering Design and Mathematics，University of the West of England，Bristol BS16 1QY，UK)

Abstract: This study proposes a reduced-dimension identification algorithm for systems with unknown structures. By using the kernel method, the unknown structure systems are approximated by a Volterra model with a high-dimension. To avoid high-order matrix inverse and eigenvalue calculations, a variable elimination algorithm is developed, which decomposes the high-dimension model into two sub-models. The theoretical analysis shows that the variable elimination algorithm has a faster convergence rate than that of the traditional gradient descent algorithm. In addition, an Aitken method is introduced to increase the convergence rates and to make the algorithm be robust to the step-size. The simulation results verify the effectiveness of the algorithm.

Keywords: parameter estimation；reduced-dimension identification algorithm；variable elimination algorithm；Volterra model；unknown structure system；condition number

0 引 言

系统辨识是控制理论和应用的基础,要针对系统设计鲁棒控制器或者预测系统的未来动态,系统的模型必须事先已知[1-3]. 系统辨识主要分为两大类:结构辨识[4-5]和参数辨识[6-7]. 结构辨识是第1步,当结构已知后,再进行参数辨识. 目前广泛应用的最小二乘算法,如梯度算法、极大似然算法以及智能算法通常都是假设系统的结构已知,进而辨识出系统的参数. 然而,随着网络技术和计算机软硬件技术的高速发展,系统结构越来越复杂,被控对象结构已知的假设越来越不现实.

核函数中的Volterra级数方法因其与幂级数有着天然的密切联系和相似之处,广泛应用于对结构未知模型的逼近[8-9]. 不同于泰勒级数, Volterra级数具有记忆项,其具有鲜明的物理意义. 理论上,只要记忆时间足够长,阶数足够大, Volterra级数可以描述

收稿日期: 2022-04-12；**录用日期:** 2022-07-17.

基金项目: 国家自然科学基金项目(61973137)；近地面探测技术重点实验室基金项目(61424140207, 61424140202)；江苏省自然科学基金项目(BK20201339).

责任编委: 柴利.

†通讯作者. E-mail: chenjing1981929@126.com.

任意的非线性动态. 文献[10]针对一类非线性系统利用Volterra模型逼近,由于Volterra模型中参数存在很多稀疏数据,利用正则项进行修正;文献[11]利用Volterra级数对交通流量混沌时间序列进行建模,获得了较好的建模效果;文献[12]针对具有混沌特性的语音信号序列,采用Volttera模型进行拟合. Volterra级数有一个很大的缺点,参数个数随着记忆因子和阶数的增长而急剧增长,从而引起所谓的“维数灾难”.

当待辨识参数维数较大时,传统的最小二乘算法更新参数时需要计算高阶矩阵的逆,这需要很大的计算量,且很多时候高阶矩阵是一个奇异矩阵,此时最小二乘算法不能获取参数估计值[13-14]. 梯度算法无需计算矩阵逆,因此是最小二乘算法一个很好的补充. 梯度算法任意选择一个初始参数估计值,计算出一个负梯度方向,并沿着该方向找到合适的步长,使上一时刻估计值沿着负梯度方向移动合适的步长到达下一个更好的估计值[15-16]. 相比于最小二乘算法,梯度算法收敛速度较慢,要提高收敛速度,方向和步长是两个关键要素. 文献[17]指出,当步长$\gamma = \dfrac{2}{\lambda_{\max}[M]+\lambda_{\min}[M]}$时($M$为被控系统的信息矩阵,$\lambda_{\max}[\cdot]$和$\lambda_{\min}[\cdot]$分别为矩阵的最大和最小特征值),梯度算法的收敛速度达到最快. 因此,需要通过计算矩阵的特征值才能选择合适的步长. 此外,当系统的信息矩阵条件数较大时,无论选择什么样的步长,梯度算法的收敛速度都异常慢[18].

针对以上经典算法处理高维系统辨识的缺点,一个很自然的想法是能否设计这样一种算法:无需求解高阶矩阵逆,无需计算矩阵的特征值,具有较快的收敛速度. 这也是本文的出发点. 借助经典的递阶辨识思想,将高维系统分解为两个低维子系统,利用最小二乘算法获得第1个参数向量关于第2个参数向量的函数,进而达到降维目的. 由于降维后的模型具有更为复杂的结构,为防止降维模型代价函数的导函数方程无解析解,通过梯度算法获得降维模型的参数估计,为提高梯度算法收敛速度,结合计算数学中的Aitken加速思想提高参数收敛速度. 最后,在获得降维模型的最优估计后,采用最小二乘算法获得第1个子系统的参数估计. 本文主要贡献如下:

1) 考虑结构未知的模型辨识问题,应用范围更加广泛;

2) 避免了高阶矩阵求逆以及矩阵特征值求解运算,计算量和计算复杂度得到了降低;

3) 不同于递阶辨识方法利用交互迭代获得参数估计,本文算法首先辨识第2个子系统参数直至获得最优解,再通过最小二乘算法一步获得第1个子系统参数,辨识效率更高;

4) 通过Aitken加速思想提高参数收敛速度,增强对步长的鲁棒性.

1 问题描述

考虑如下模型:

$$y(t) = f(u(t)) + v(t). \tag{1}$$

其中:$y(t)$为系统输出,$u(t)$为系统输入,$v(t)$是均值为零、方差为σ^2的高斯白噪声,$f(\cdot)$为结构未知的函数.

利用二阶Volterra模型拟合上述结构未知非线性系统

$$y(t) = \sum_{i=1}^{n} h_i u(t-i) + \sum_{i=1}^{n}\sum_{j=1}^{n} h_{i,j} u(t-i)u(t-j) + v(t). \tag{2}$$

由于$h_{i,j}$和$h_{j,i}$对应相同的输入量$u(t-i)u(t-j)$,为降低系统维数,$h_{i,j}$和$h_{j,i}$可用同一个参数$h_{i,j}$表示. 定义系统的参数向量θ和信息向量$\varphi(t)$为

$$\theta = [h_1, \ldots, h_n, h_{1,1}, \ldots, h_{n,n}]^{\mathrm{T}} \in \boldsymbol{R}^{\frac{n^2+3n}{2}},$$

$$\varphi(t) = [u(t-1), \ldots, u(t-n), u^2(t-1), \ldots, u^2(t-n)]^{\mathrm{T}} \in \boldsymbol{R}^{\frac{n^2+3n}{2}}.$$

对于结构未知的系统,Volterra模型的阶数和记忆因子越大,其拟合系统动态效果越好,但会导致参数维数急剧增长. 对于本文的二阶系统,未知参数的维数会随着记忆因子n的增长而平方增长. 例如,当$n=10$时,系统阶数为65,当$n=100$时,系统阶数为5 150.

系统(1)可以转化为如下自回归模型:

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

假设收集了L组数据,并定义

$$\boldsymbol{Y}(L) = [y(L), y(L-1), \ldots, y(1)]^{\mathrm{T}} \in \boldsymbol{R}^{L},$$

$$\boldsymbol{\Phi}(L) = [\boldsymbol{\varphi}(L), \boldsymbol{\varphi}(L-1), \ldots, \boldsymbol{\varphi}(1)]^{\mathrm{T}} \in \boldsymbol{R}^{L\times\frac{n^2+3n}{2}},$$

$$\boldsymbol{V}(L) = [v(L), v(L-1), \ldots, v(1)]^{\mathrm{T}} \in \boldsymbol{R}^{L}.$$

对于L组数据的结构未知系统,等价模型为

$$\boldsymbol{Y}(L) = \boldsymbol{\Phi}(L)\theta + \boldsymbol{V}(L). \tag{3}$$

定义代价函数为

$$J(\theta) = ||\boldsymbol{Y}(L) - \boldsymbol{\Phi}(L)\theta||^2.$$

下面用两种经典的方法辨识系统参数.

1) 传统最小二乘算法(traditional least squares, T-LS).

对代价函数关于参数向量θ求导,得到导函数方程为

$$\boldsymbol{\Phi}^{\mathrm{T}}(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}(L)\theta]=0.$$

上述方程解析解即为参数估计,有

$$\hat{\theta}=[\boldsymbol{\Phi}^{\mathrm{T}}(L)\boldsymbol{\Phi}(L)]^{-1}\boldsymbol{\Phi}^{\mathrm{T}}(L)\boldsymbol{Y}(L).$$

注1　当系统关于参数是线性的,且导函数方程存在解析解时,最小二乘算法能一步获得最优解. 由于最小二乘算法需要求解矩阵逆,当矩阵的维数很大时,其求矩阵逆运算量很大. 例如,当矩阵的阶数为N时,求逆运算计算量为$O(N^3)$[19].

2) 传统梯度算法(traditional gradient descent, T-GD)

假设在第$k-1$次的参数估计值为$\hat{\theta}_{k-1}$,则第k次的方向为

$$\boldsymbol{d}_k=\boldsymbol{\Phi}^{\mathrm{T}}(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}(L)\hat{\theta}_{k-1}].$$

利用梯度算法获得参数估计为

$$\begin{aligned}&\hat{\theta}_k=\hat{\theta}_{k-1}+\gamma\boldsymbol{\Phi}^{\mathrm{T}}(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}(L)\hat{\theta}_{k-1}],\\&0<\gamma<\frac{2}{\lambda_{\max}[\boldsymbol{\Phi}^{\mathrm{T}}(L)\boldsymbol{\Phi}(L)]}.\end{aligned}$$

注2　梯度算法无需求解矩阵逆,无需假设导函数方程存在解析解,因此计算量低,且适合较为复杂的非线性系统(导函数方程无解析解). 然而,梯度算法速度慢,为保证收敛,需要计算矩阵的特征值[20].

由文献[17]可知,当步长

$$\gamma=\frac{2}{\lambda_{\max}[\boldsymbol{\Phi}^{\mathrm{T}}(L)\boldsymbol{\Phi}(L)]+\lambda_{\min}[\boldsymbol{\Phi}^{\mathrm{T}}(L)\boldsymbol{\Phi}(L)]}$$

时,梯度算法收敛速度达到最快,其收敛因子为$\varrho=g(\tau)=\dfrac{\tau-1}{\tau+1}$,$\tau=\dfrac{\lambda_{\max}[\boldsymbol{\Phi}^{\mathrm{T}}(L)\boldsymbol{\Phi}(L)]}{\lambda_{\min}[\boldsymbol{\Phi}^{\mathrm{T}}(L)\boldsymbol{\Phi}(L)]}$为矩阵的条件数,收敛因子越小,收敛速度越快,当$\varrho=0$时,算法收敛速度与最小二乘算法相当. 对于本文的确定系统,其信息矩阵是确定的且函数$g(\tau)$是递增的,当矩阵是奇异或者病态时(奇异时$t\rightarrow\infty$,病态时τ的取值较大),收敛因子近似为1,算法不收敛. 因此,一个自然的想法是对于一个确定的系统,能否通过分解方法降低其信息矩阵条件数,进而提高收敛速度.

2　降维算法

本节介绍两类降维算法: 递阶辨识算法和变量消去算法.

2.1　递阶辨识算法

递阶辨识算法是处理复杂系统或者大系统的一个有效方法,它将复杂系统/大系统分解为几个子系统,辨识其中一个系统参数时需要固定其他子系统的参数. 通过递阶辨识方法可以实现复杂系统/大系统辨识问题的简单化[21-23].

将Volterra模型变为

$$\begin{aligned}&\boldsymbol{Y}(L)=\\&\boldsymbol{\Phi}_1(L)\theta_1+\boldsymbol{\Phi}_2(L)\theta_2+\ldots+\boldsymbol{\Phi}_m(L)\theta_m+\boldsymbol{V}(L).\end{aligned}\tag{4}$$

其中: $\boldsymbol{\Phi}_i(L)\in\boldsymbol{R}^{L\times l_i}$, $i=1,2,\ldots,m,\theta_i\in\boldsymbol{R}^{l_i},l_1+l_2+\ldots+l_m=\dfrac{n^2+3n}{2}$. 原始高维系统可以转化为如下$m$个子系统:

$$\boldsymbol{Y}_i(L)=\boldsymbol{\Phi}_i(L)\theta_i+\boldsymbol{V}(L),$$

$$\boldsymbol{Y}_i(L)=\boldsymbol{Y}(L)-\sum_{j=1,j\neq i}^{m}\boldsymbol{\Phi}_j(L)\theta_j.$$

当$\boldsymbol{Y}_i(L)$已知时,原始的高维辨识问题转化为m个低维辨识问题. 然而,$\boldsymbol{Y}_i(L)$中含有未知的参数向量$\theta_j(j=1,2,\ldots,i-1,i+1,\ldots,m)$,因此必须先求出$\boldsymbol{Y}_i(L)$. 假设在第$k-1$次已经获得了所有参数向量的估计$\hat{\theta}_j^{k-1}(j=1,2,\ldots,m)$,在第$k$次时,第$i$个子系统的输出为

$$\begin{aligned}&\hat{\boldsymbol{Y}}_i^k(L)=\\&\boldsymbol{Y}(L)-\sum_{j=1}^{i-1}\boldsymbol{\Phi}_j(L)\hat{\theta}_j^k-\sum_{j=i+1}^{m}\boldsymbol{\Phi}_j(L)\hat{\theta}_j^{k-1}.\end{aligned}\tag{5}$$

对于每一个子系统,利用梯度算法获得参数估计为

$$\hat{\theta}_i^k=\hat{\theta}_i^{k-1}+\gamma_i\boldsymbol{\Phi}_i^{\mathrm{T}}(L)[\hat{\boldsymbol{Y}}_i^k(L)-\boldsymbol{\Phi}_i(L)\hat{\theta}_i^{k-1}],\tag{6}$$

$$0<\gamma_i<\frac{2}{\lambda_{\max}[\boldsymbol{\Phi}_i^{\mathrm{T}}(L)\boldsymbol{\Phi}_i(L)]}.\tag{7}$$

注3　递阶辨识算法将高维辨识转化为低维辨识,即将高阶矩阵求逆或者求特征值运算转化为低阶矩阵的相应运算,因此计算量和计算复杂度都得到了下降. 然而,递阶辨识算法其子系统的参数向量是交替辨识的,某一个辨识效果差则会导致所有的辨识效果变差,因此相比较于传统的梯度算法,递阶算法的稳定性相对较差.

2.2　变量消去法

变量消去法的核心思想是将系统分解为两个子系统,即

$$\boldsymbol{Y}(L)=\boldsymbol{\Phi}_1(L)\theta_1+\boldsymbol{\Phi}_2(L)\theta_2+\boldsymbol{V}(L).\tag{8}$$

其中: $\boldsymbol{\Phi}_1(L)\in\boldsymbol{R}^{L\times m_1}$, $\boldsymbol{\Phi}_2(L)\in\boldsymbol{R}^{L\times m_2}$, $\theta_1\in\boldsymbol{R}^{m_1}$, $\theta_2\in\boldsymbol{R}^{m_2}$.

不同于递阶辨识法,变量消去法通过最小二乘算法求出参数向量θ_1关于向量θ_2的函数表示. 将系统(8)转化为

$$\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\theta_2=\boldsymbol{\Phi}_1(L)\theta_1+\boldsymbol{V}(L).$$

利用最小二乘算法获得向量θ_1的估计为

$$\theta_1=[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\theta_2].\tag{9}$$

将式(9)代入(8),得到

$$\begin{aligned}\boldsymbol{Y}(L)=&\boldsymbol{\Phi}_1(L)[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\times\\&[\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\theta_2]+\boldsymbol{\Phi}_2(L)\theta_2+\boldsymbol{V}(L)=\\&\boldsymbol{\Phi}_1(L)[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{Y}(L)+\\&\{\boldsymbol{I}-\boldsymbol{\Phi}_1(L)[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\}\times\\&\boldsymbol{\Phi}_2(L)\theta_2+\boldsymbol{V}(L).\end{aligned}\tag{10}$$

令$\{\boldsymbol{I}-\boldsymbol{\Phi}_1(L)[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\}=\boldsymbol{P}_1(L)$,式(10)转化为

$$\boldsymbol{P}_1(L)\boldsymbol{Y}(L)=\boldsymbol{P}_1(L)\boldsymbol{\Phi}_2(L)\theta_2+\boldsymbol{V}(L).\tag{11}$$

定义代价函数

$$J(\theta_2)=||\boldsymbol{P}_1(L)\boldsymbol{Y}(L)-\boldsymbol{P}_1(L)\boldsymbol{\Phi}_2(L)\theta_2||^2.$$

利用梯度算法获得参数估计为

$$\begin{aligned}\hat{\theta}_2^k=&\hat{\theta}_2^{k-1}+\gamma\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1^{\mathrm{T}}(L)\times\\&[\boldsymbol{P}_1(L)\boldsymbol{Y}(L)-\boldsymbol{P}_1(L)\boldsymbol{\Phi}_2(L)\hat{\theta}_2^{k-1}]=\\&\hat{\theta}_2^{k-1}+\gamma\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1^{\mathrm{T}}(L)\boldsymbol{P}_1(L)\times\\&[\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\hat{\theta}_2^{k-1}].\end{aligned}\tag{12}$$

显然,$\boldsymbol{P}_1(L)$是对称矩阵,且

$$\begin{aligned}&\boldsymbol{P}_1^{\mathrm{T}}(L)\boldsymbol{P}_1(L)=\\&\{\boldsymbol{I}-\boldsymbol{\Phi}_1(L)[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\}^2=\\&\boldsymbol{I}-\boldsymbol{\Phi}_1(L)[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)=\boldsymbol{P}_1(L).\end{aligned}$$

于是,式(12)简化为

$$\hat{\theta}_2^k=\hat{\theta}_2^{k-1}+\gamma\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\hat{\theta}_2^{k-1}],\tag{13}$$

$$0<\gamma<\frac{2}{\lambda_{\max}[\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1(L)\boldsymbol{\Phi}_2(L)]}.\tag{14}$$

由梯度算法获得参数向量θ_2的最优估计值$\hat{\theta}_2^{\mathrm{op}}$后,根据式(9)获得$\theta_1$的估计为

$$\hat{\theta}_1=[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\hat{\theta}_2^{\mathrm{op}}].\tag{15}$$

注4 与递阶算法相比,变量消去法有两点不同: 1) 递阶算法使用交互辨识获得参数估计,即$\hat{\theta}_2^0\rightarrow\underline{\underline{\hat{\theta}_1^1}}\rightarrow\hat{\theta}_2^1\rightarrow\underline{\underline{\hat{\theta}_1^2}}\rightarrow\hat{\theta}_2^2\rightarrow\ldots\rightarrow\underline{\underline{\hat{\theta}_1^k}}\rightarrow\hat{\theta}_2^k\ldots$,而变量消去法首先获得$\theta_2$的估计,最后才一步获得$\theta_1$的估计,即$\hat{\theta}_2^0\rightarrow\hat{\theta}_2^1\rightarrow\hat{\theta}_2^2\rightarrow\hat{\theta}_2^k\ldots\rightarrow\hat{\theta}_2^{op}\rightarrow\underline{\underline{\hat{\theta}_1}}$; 2) 利用梯度法求解参数估计时,由于$\hat{\boldsymbol{Y}}_i^k(L)$在每一次迭代过程中是变化的,导致递阶辨识方法求解参数时其迭代方程也是相应变化的,而利用变量消去法中梯度算法更新参数过程中,迭代方程是始终不变的. 综上,变量消去法其稳定性优于递阶辨识算法.

注5 对于一个高维系统,递阶辨识算法可以将高维系统分为多个子系统$(m\geqslant 2)$,进而实现降维;而变量消去法只能将高维系统分为两个子系统,为避免高阶矩阵求逆,一般假设$m_1\leqslant m_2$. 由此带来一个新的问题,当m_2较大时,求解高阶矩阵的特征值比较麻烦. 能否避免求解高阶矩阵特征值,在第3节将给出解决方案.

2.3 稳定性分析

假设递阶辨识算法的代价函数为

$$\begin{aligned}&J(\theta_1,\theta_2,\ldots,\theta_m)=\\&||\boldsymbol{Y}(L)-\boldsymbol{\Phi}_1(L)\theta_1-\boldsymbol{\Phi}_2(L)\theta_2-\ldots-\boldsymbol{\Phi}_m(L)\theta_m||^2.\end{aligned}$$

变量消去法的代价函数为

$$J(\theta_1,\theta_2)=||\boldsymbol{Y}(L)-\boldsymbol{\Phi}_1(L)\theta_1-\boldsymbol{\Phi}_2(L)\theta_2||^2.$$

如下两个定理保证了递阶辨识算法和变量消去法是收敛的.

定理1 对于高维系统(4),每一个子系统输出$\boldsymbol{Y}_i(L)$由式(5)计算得到,每一个子系统参数向量θ_i由式(6)和(7)更新得到,则递阶辨识算法是收敛的.

证明 假设在第$k-1$次,系统的参数估计为

$$\hat{\theta}^{k-1}=[\hat{\theta}_1^{k-1},\hat{\theta}_2^{k-1},\ldots,\hat{\theta}_m^{k-1}].$$

在第k次时,第1个参数向量θ_1的估计值为

$$\hat{\theta}_1^k=\hat{\theta}_1^{k-1}+\gamma_1\boldsymbol{\Phi}_1^{\mathrm{T}}(L)[\hat{\boldsymbol{Y}}_1^k(L)-\boldsymbol{\Phi}_1(L)\hat{\theta}_1^{k-1}],$$

其中$\boldsymbol{\Phi}_1^{\mathrm{T}}(L)[\hat{\boldsymbol{Y}}_1^k(L)-\boldsymbol{\Phi}_1(L)\hat{\theta}_1^{k-1}]$为负梯度方向,能确保第$k-1$的值向真值移动. 步长$\gamma_1$按照式(7)选择,可以保证移动合适的距离,因此可得

$$J(\underline{\underline{\hat{\theta}_1^k}},\hat{\theta}_2^{k-1},\ldots,\hat{\theta}_m^{k-1})\leqslant J(\underline{\underline{\hat{\theta}_1^{k-1}}},\hat{\theta}_2^{k-1},\ldots,\hat{\theta}_m^{k-1}).$$

同理可得

$$\begin{aligned}&J(\hat{\theta}_1^k,\hat{\theta}_2^k,\ldots,\underline{\underline{\hat{\theta}_i^k}},\hat{\theta}_{i+1}^{k-1},\ldots,\hat{\theta}_m^{k-1})\leqslant\\&J(\hat{\theta}_1^k,\hat{\theta}_2^k,\ldots,\underline{\underline{\hat{\theta}_i^{k-1}}},\hat{\theta}_{i+1}^{k-1},\ldots,\hat{\theta}_m^{k-1}).\end{aligned}$$

综上有

$$J(\hat{\theta}_1^k,\hat{\theta}_2^k,\ldots,\hat{\theta}_m^k)\leqslant J(\hat{\theta}_1^{k-1},\hat{\theta}_2^{k-1},\ldots,\hat{\theta}_m^{k-1}).\qquad\square$$

定理2 对于高维系统(8),有

$$\boldsymbol{P}_1(L)=\{\boldsymbol{I}-\boldsymbol{\Phi}_1(L)[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\},$$

参数向量θ_2由式(13)和(14)更新得到,θ_1由式(15)更新得到,则变量消去法收敛.

证明 由式(9)得

$$\begin{aligned}&\theta_1=\\&[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\theta_2]=g(\theta_2).\end{aligned}\tag{16}$$

变量消去法的代价函数可以转化为

$$J(\theta_1,\theta_2)=J(g(\theta_2),\theta_2).$$

假设第$k-1$次时参数估计为$\hat{\theta}_1^{k-1}$和$\hat{\theta}_2^{k-1}$,由式(9)和(16)可得

$$J(g(\hat{\theta}_2^{k-1}),\hat{\theta}_2^{k-1})\leqslant J(\hat{\theta}_1^{k-1},\hat{\theta}_2^{k-1}).$$

由式(13)和(14)可得θ_2的估计为

$$\hat{\theta}_2^k=\hat{\theta}_2^{k-1}+\gamma\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\hat{\theta}_2^{k-1}],$$
$$0<\gamma<\frac{2}{\lambda_{\max}[\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1(L)\boldsymbol{\Phi}_2(L)]}.$$

其中负梯度方向$\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1(L)[\boldsymbol{Y}(L)-\boldsymbol{\Phi}_2(L)\hat{\theta}_2^{k-1}]$确保向真值移动,步长保证移动合适的距离. 因此可得

$$J(g(\hat{\theta}_2^k),\hat{\theta}_2^k)\leqslant J(g(\hat{\theta}_2^{k-1}),\hat{\theta}_2^{k-1}).$$

设第k次的参数估计$\hat{\theta}_1^k$由式(16)求得,则如下等式成立:

$$J(g(\hat{\theta}_2^k),\hat{\theta}_2^k)=J(\hat{\theta}_1^k,\hat{\theta}_2^k).$$

综上可得

$$J(\hat{\theta}_1^k,\hat{\theta}_2^k)\leqslant J(\hat{\theta}_1^{k-1},\hat{\theta}_2^{k-1}),$$

即变量消去法是收敛的. □

3 算法比较及加速算法

对传统梯度算法、递阶辨识算法以及变量消去法进行比较,并分析加速算法的可行性.

3.1 3种算法比较

假设递阶算法、变量消去法将参数向量都分为两个子向量,则代价函数为

$$J(\theta_1,\theta_2)=||\boldsymbol{Y}(L)-\boldsymbol{\Phi}_1(L)\theta_1-\boldsymbol{\Phi}_2(L)\theta_2||^2.$$

传统梯度算法将参数向量θ_1和θ_2看作一个整体,利用梯度算法获得参数估计,此时信息矩阵为

$$\boldsymbol{Q}_{\text{T-GD}}=\begin{bmatrix}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L) & \boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_2(L)\\ \boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L) & \boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{\Phi}_2(L)\end{bmatrix}.$$

利用基于梯度算法的递阶算法,固定一个向量,计算另外一个向量,此时递阶算法的信息矩阵为

$$\boldsymbol{Q}_{\text{HIE}}=\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L),$$

或

$$\boldsymbol{Q}_{\text{HIE}}=\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{\Phi}_2(L).$$

由矩阵论可得,$\lambda_{\max}[\boldsymbol{\Phi}_i^{\mathrm{T}}(L)\boldsymbol{\Phi}_i(L)]\leqslant\lambda_{\max}[\boldsymbol{Q}_{\text{T-GD}}]$且$\lambda_{\min}[\boldsymbol{\Phi}_i^{\mathrm{T}}(L)\boldsymbol{\Phi}_i(L)]\geqslant\lambda_{\min}[\boldsymbol{Q}_{\text{T-GD}}]$. 因此,两种方法信息矩阵的条件数满足

$$\tau_{\text{T-GD}}\geqslant\tau_{\text{HIE}}.$$

注6 通过递阶方法对系统进行降维后,其信息矩阵条件数小于传统梯度方法信息矩阵的条件数,因此,递阶算法收敛速度“应快于”传统梯度算法. 然而,递阶方法存在交互估计,存在一定的不稳定性,导致在实际应用中两种算法的收敛速度无法进行比较.

由式(13)可知,变量消去法的信息矩阵为

$$\boldsymbol{Q}_{\text{VE}}=\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1(L)\boldsymbol{P}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_2(L)=$$
$$\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1(L)\boldsymbol{\Phi}_2(L).\tag{17}$$

为分析变量消去法和传统梯度算法信息矩阵条件数的关系,下面给出两个引理.

引理1 对于一个分块矩阵

$$\boldsymbol{A}=\begin{bmatrix}\boldsymbol{A}_{1,1} & \boldsymbol{A}_{1,2}\\ \boldsymbol{A}_{2,1} & \boldsymbol{A}_{2,2}\end{bmatrix},$$

其逆矩阵为

$$\boldsymbol{A}^{-1}=\begin{bmatrix}\boldsymbol{H}_{1,1} & \boldsymbol{H}_{1,2}\\ \boldsymbol{H}_{2,1} & \boldsymbol{H}_{2,2}\end{bmatrix}.$$

其中

$$\boldsymbol{H}_{1,1}=\boldsymbol{A}_{1,1}^{-1}+\boldsymbol{A}_{1,1}^{-1}\boldsymbol{A}_{1,2}(\boldsymbol{A}_{2,2}-$$
$$\boldsymbol{A}_{2,1}\boldsymbol{A}_{1,1}^{-1}\boldsymbol{A}_{1,2})^{-1}\boldsymbol{A}_{2,1}\boldsymbol{A}_{1,1}^{-1},$$
$$\boldsymbol{H}_{1,2}=-\boldsymbol{A}_{1,1}^{-1}\boldsymbol{A}_{1,2}(\boldsymbol{A}_{2,2}-\boldsymbol{A}_{2,1}\boldsymbol{A}_{1,1}^{-1}\boldsymbol{A}_{1,2})^{-1},$$
$$\boldsymbol{H}_{2,1}=-(\boldsymbol{A}_{2,2}-\boldsymbol{A}_{2,1}\boldsymbol{A}_{1,1}^{-1}\boldsymbol{A}_{1,2})^{-1}\boldsymbol{A}_{2,1}\boldsymbol{A}_{1,1}^{-1},$$
$$\boldsymbol{H}_{2,2}=(\boldsymbol{A}_{2,2}-\boldsymbol{A}_{2,1}\boldsymbol{A}_{1,1}^{-1}\boldsymbol{A}_{1,2})^{-1}.$$

引理2 对于一个对称正定矩阵$\boldsymbol{A}$,其条件数为$\tau_{\boldsymbol{A}}$,逆矩阵为$\boldsymbol{A}^{-1}$,则有

$$\tau_{\boldsymbol{A}}=\tau_{\boldsymbol{A}^{-1}}.$$

引理1和引理2可由矩阵论知识得到,证明略.

为简单起见,下文令$\boldsymbol{\Phi}_1=\boldsymbol{\Phi}_1(L),\boldsymbol{\Phi}_2=\boldsymbol{\Phi}_2(L)$. 根据引理1,可得

$$\boldsymbol{Q}_{\text{T-GD}}^{-1}=$$
$$\begin{bmatrix}\diamond_1 & \diamond_2\\ \diamond_3 & [\boldsymbol{\Phi}_2^{\mathrm{T}}\boldsymbol{\Phi}_2-\boldsymbol{\Phi}_2^{\mathrm{T}}\boldsymbol{\Phi}_1(\boldsymbol{\Phi}_1^{\mathrm{T}}\boldsymbol{\Phi}_1)^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}\boldsymbol{\Phi}_2]^{-1}\end{bmatrix},$$

其中$\diamond_i(i=1,2,3)$是由$\boldsymbol{\Phi}_1$和$\boldsymbol{\Phi}_2$组成的项,不做关注. 由引理2可知$[\boldsymbol{\Phi}_2^{\mathrm{T}}\boldsymbol{\Phi}_2-\boldsymbol{\Phi}_2^{\mathrm{T}}\boldsymbol{\Phi}_1(\boldsymbol{\Phi}_1^{\mathrm{T}}\boldsymbol{\Phi}_1)^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}\boldsymbol{\Phi}_2]^{-1}$与$[\boldsymbol{\Phi}_2^{\mathrm{T}}\boldsymbol{\Phi}_2-\boldsymbol{\Phi}_2^{\mathrm{T}}\boldsymbol{\Phi}_1(\boldsymbol{\Phi}_1^{\mathrm{T}}\boldsymbol{\Phi}_1)^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}\boldsymbol{\Phi}_2]$的条件数相等.

由式(17)可知

$$\boldsymbol{Q}_{\text{VE}}=$$
$$\boldsymbol{\Phi}_2^{\mathrm{T}}(L)\{\boldsymbol{I}-\boldsymbol{\Phi}_1(L)[\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{\Phi}_1(L)]^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}(L)\}\boldsymbol{\Phi}_2(L)=$$
$$\boldsymbol{\Phi}_2^{\mathrm{T}}\boldsymbol{\Phi}_2-\boldsymbol{\Phi}_2^{\mathrm{T}}\boldsymbol{\Phi}_1(\boldsymbol{\Phi}_1^{\mathrm{T}}\boldsymbol{\Phi}_1)^{-1}\boldsymbol{\Phi}_1^{\mathrm{T}}\boldsymbol{\Phi}_2.$$

显然

$$\tau_{\text{VE}}=\tau_{\boldsymbol{Q}_{\text{VE}}}=\tau_{\boldsymbol{Q}_{\text{VE}}^{-1}}.$$

由于$\boldsymbol{Q}_{\text{VE}}^{-1}$是$\boldsymbol{Q}_{\text{T-GD}}^{-1}$的一个对角子矩阵,显然有

$$\tau_{\boldsymbol{Q}_{\text{VE}}^{-1}}\leqslant\tau_{\text{T-GD}}.$$

进一步可得

$$\tau_{\text{VE}}\leqslant\tau_{\text{T-GD}}.$$

注7 上式表明利用变量消去法后得到降维系统的信息矩阵条件数小于原始高维系统信息矩阵的

条件数,且在变量消去法中不存在交互估计. 因此,变量消去法的收敛速度快于传统梯度算法的收敛速度.

3.2 加速算法

将传统梯度算法、递阶辨识算法以及变量消去法的迭代方程分别写为

$$\hat{\theta}_k = f_{\text{T-GD}}(\hat{\theta}_{k-1}) = \hat{\theta}_{k-1} + \gamma \boldsymbol{\Phi}^{\mathrm{T}}(L)[\boldsymbol{Y}(L) - \boldsymbol{\Phi}(L)\hat{\theta}_{k-1}], \tag{18}$$

$$\hat{\theta}_i^k = f_{\text{HIE}}(\hat{\theta}_i^{k-1}) = \hat{\theta}_i^{k-1} + \gamma_i \boldsymbol{\Phi}_i^{\mathrm{T}}(L)[\hat{\boldsymbol{Y}}_i^k(L) - \boldsymbol{\Phi}_i(L)\hat{\theta}_i^{k-1}], \tag{19}$$

$$\hat{\theta}_2^k = f_{\text{VE}}(\hat{\theta}_2^{k-1}) = \hat{\theta}_2^{k-1} + \gamma \boldsymbol{\Phi}_2^{\mathrm{T}}(L)\boldsymbol{P}_1(L)[\boldsymbol{Y}(L) - \boldsymbol{\Phi}_2(L)\hat{\theta}_2^{k-1}]. \tag{20}$$

利用计算数学中的Aitken加速方法对3种算法产生的序列分别进行加速,可得如下结论:

1) 由于 $f_{\text{T-GD}}(\hat{\theta}_{k-1})$ 和 $f_{\text{VE}}(\hat{\theta}_2^{k-1})$ 在整个迭代过程中保持不变,传统梯度算法和变量消去法可以利用Aitken加速法进行加速.

2) 利用递阶辨识方法更新参数的过程中,迭代函数中存在时变向量 $\hat{\boldsymbol{Y}}_i^k(L)$,因此迭代方程 $f_{\text{HIE}}(\hat{\theta}_i^{k-1})$ 是始终变化的. 由文献 [24] 可知, 此时不能利用Aitken方法对递阶辨识方法产生的序列加速,即基于Aitken加速的递阶辨识方法是无效的.

3) 由文献 [24] 可知, 在利用 $f_{\text{T-GD}}(\hat{\theta}_{k-1})$ 和 $f_{\text{VE}}(\hat{\theta}_2^{k-1})$ 加速过程中可以任意选择步长,加速算法始终是收敛的. 即利用Aitken加速后,传统梯度和变量消去法对步长是鲁棒的(无需通过计算特征值选择步长). 因此,利用变量消去法优先选择向量 θ_1 的维数小于 θ_2,可以同时避免求解高阶矩阵逆和高阶矩阵特征值.

4 仿真分析

例1 3种算法收敛速度和时间比较.

为验证变量消去法具有较快的收敛速度,考虑如下一类模型:

$$y(t) = \sum_{i=0}^{10} 0.9^i u(t-i) + v(t),$$

$$\theta_1 = [0.9^0, 0.9^1, 0.9^2, 0.9^3, 0.9^4]^{\mathrm{T}},$$

$$\theta_2 = [0.9^5, 0.9^6, 0.9^7, 0.9^8, 0.9^9, 0.9^{10}]^{\mathrm{T}}.$$

$\{v(t)\}$ 是均值为零、方差为 $\sigma^2 = 0.10^2$ 的白噪声信号,输入 $\{u(t)\}$ 是满足均值为零、方差为1的持续激励信号. 采用传统梯度算法(T-GD)、递阶辨识方法(HIE)和变量消去法(VE)对上述模型进行辨识. 参数估计值和估计误差 $\eta = \|\hat{\theta}^k - \theta\|/\|\theta\|$ 随迭代次数 k 变化的结果如图1和表1所示. 当3种算法参数误差相当时,其收敛时间如表2所示.

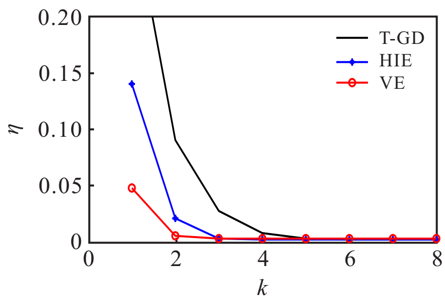

图1 参数误差 η 随 k 变化曲线

表1 参数估计及估计误差

	k	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	θ_9	θ_{10}	θ_{11}	η / %
T-GD	2	0.926 56	0.831 37	0.742 35	0.664 65	0.604 09	0.569 09	0.477 41	0.400 90	0.374 06	0.334 68	0.300 09	9.091 31
	4	0.991 83	0.895 32	0.802 42	0.724 97	0.652 66	0.589 80	0.526 10	0.470 58	0.427 49	0.379 96	0.346 64	0.835 62
	6	0.998 18	0.901 49	0.807 97	0.731 00	0.657 03	0.590 89	0.530 63	0.477 67	0.433 03	0.384 86	0.351 89	0.287 37
	8	0.998 81	0.902 09	0.808 51	0.731 61	0.657 45	0.590 96	0.531 06	0.478 37	0.433 57	0.385 36	0.352 44	0.308 31
HIE	2	0.979 98	0.881 41	0.785 77	0.712 37	0.643 20	0.585 88	0.522 20	0.466 80	0.423 12	0.376 32	0.347 97	2.149 66
	4	0.998 43	0.901 62	0.807 99	0.731 21	0.657 17	0.590 88	0.530 92	0.478 18	0.433 38	0.385 20	0.352 38	0.300 05
	6	0.998 87	0.902 15	0.808 55	0.731 67	0.657 49	0.590 97	0.531 10	0.478 44	0.433 62	0.385 41	0.352 50	0.311 47
	8	0.998 88	0.902 16	0.808 56	0.731 68	0.657 50	0.590 97	0.531 11	0.478 45	0.433 63	0.385 42	0.352 51	0.311 86
VE	2	0.986 25	0.889 04	0.802 88	0.727 97	0.648 12	0.586 60	0.524 80	0.475 08	0.428 42	0.381 55	0.346 54	0.638 52
	4	0.993 71	0.897 96	0.812 11	0.735 98	0.654 91	0.586 68	0.525 35	0.476 24	0.429 48	0.382 40	0.347 65	0.370 99
	6	0.993 79	0.898 06	0.812 21	0.736 08	0.654 98	0.586 68	0.525 36	0.476 26	0.429 50	0.382 41	0.347 67	0.370 90
	8	0.993 79	0.898 06	0.812 21	0.736 08	0.654 99	0.586 68	0.525 36	0.476 26	0.429 50	0.382 41	0.347 67	0.370 90
true values		1.000 00	0.900 00	0.810 00	0.729 00	0.656 10	0.590 49	0.531 44	0.478 30	0.430 47	0.387 42	0.348 68	

表 2　3 种算法的运行时间

算法	运行次数	运行时间 / s
T-GD	5	0.097
HIE	3	0.102
VE	2	0.134

由图 1 和表 1 可知: 变量消去法收敛速度是最快的 (迭代次数最少), 其次是递阶辨识算法, 传统梯度算法的收敛速度最慢. 表 2 表明, 尽管变量消去法迭代次数最少, 但是每一步迭代时间是最长的. 然而, 当系统维数急剧增长时, 由于迭代次数的降低, 且无需计算高维矩阵特征值, 变量消去法其运行时间会逐渐小于传统梯度算法.

进一步, 利用 Aitken 加速方法对 3 种算法产生的序列进行加速, 步长分别选择为: 小步长 $\gamma = \frac{0.1}{\lambda_{\max}[M]}$, 阈值步长 $\gamma = \frac{2}{\lambda_{\max}[M]}$, 大步长 $\gamma = \frac{2.1}{\lambda_{\max}[M]}$. 参数估计误差如图 2 ～ 4 所示. 由图 2 ～ 4 可知:

1) 当选择小步长时, T-GD、HIE 以及 VE 算法都是收敛的, 但收敛速度很慢; 通过 Aitken 加速后, T-GD-Aitken 和 VE-Aitken 算法收敛速度得到明显改善, 而 HIE-Aitken 算法收敛速度变化不大 (见图 2).

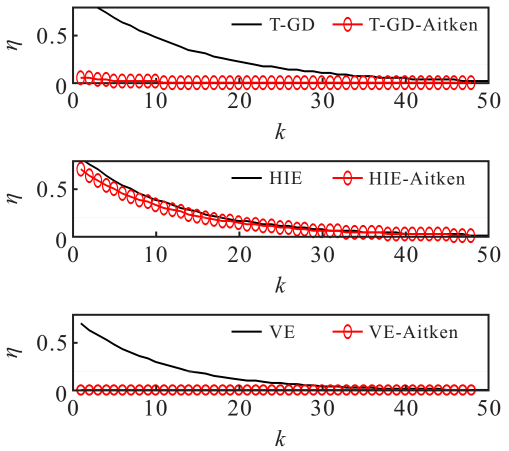

图 2　参数误差 η 随 k 变化曲线 $\left(\gamma = \frac{0.1}{\lambda_{\max}[M]}\right)$

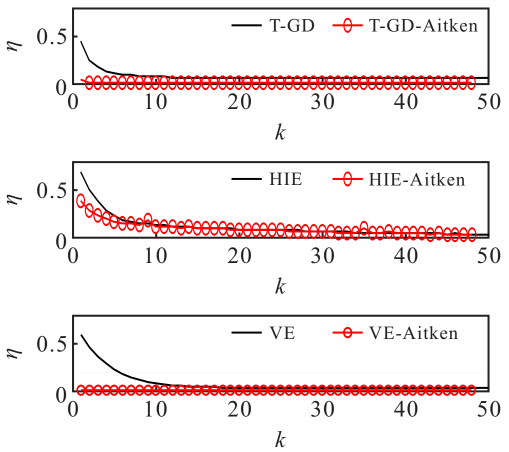

图 3　参数误差 η 随 k 变化曲线 $\left(\gamma = \frac{2}{\lambda_{\max}[M]}\right)$

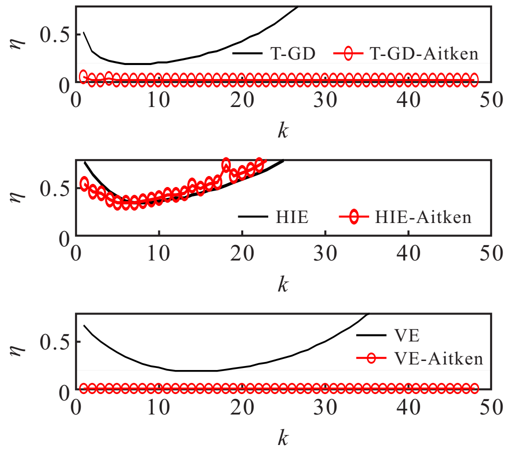

图 4　参数误差 η 随 k 变化曲线 $\left(\gamma = \frac{2.1}{\lambda_{\max}[M]}\right)$

2) 当选择阈值步长时, T-GD、HIE 以及 VE 算法都收敛到一个局部最优. 通过 Aitken 加速后, T-GD-Aitken 和 VE-Aitken 算法收敛速度得到明显改善, 且能收敛到真值, 而 HIE-Aitken 算法收敛速度变化不大, 且同样只能收敛到局部最优值 (见图 3).

3) 当选择超过阈值的步长时 (大步长), T-GD、HIE 以及 VE 算法都发散. 通过 Aitken 加速后, T-GD-Aitken 和 VE-Aitken 算法将发散的序列变为收敛, 而 HIE-Aitken 算法还是发散的 (见图 4).

综上, Aitken 加速算法对 T-GD 以及 VE 算法是有效的 (因为这两种算法的迭代方程是不变的); 而 Aitken 加速算法对 HIE 算法是无效的. 表明 T-GD-Aitken 和 VE-Aitken 算法对步长具有鲁棒特性.

例 2　烘干机模型.

考虑 Matlab 中的烘干机模型, 将系统用 2 阶含有 6 个记忆因子的模型描述为

$$
\begin{aligned}
y(t) &= h_1u(t-1) + \ldots + h_6u(t-6) + \\
&\quad h_{1,1}u^2(t-1) + \ldots + h_{6,6}u^2(t-6) + v(t), \\
\theta &= [h_1, \ldots, h_6, h_{1,1}, h_{1,2}, \ldots, h_{6,6}]^{\mathrm{T}}, \\
\theta_1 &= [h_1, h_2, h_3, h_4, h_5, h_6]^{\mathrm{T}}, \\
\theta_2 &= [h_{1,1}, h_{1,2}, h_{1,3}, h_{1,4}, h_{1,5}, h_{1,6}, h_{2,2}, h_{2,3}, \\
&\quad h_{2,4}, h_{2,5}, h_{2,6}, h_{3,3}, h_{3,4}, h_{3,5}, h_{3,6}, h_{4,4}, \\
&\quad h_{4,5}, h_{4,6}, h_{5,5}, h_{5,6}, h_{6,6}]^{\mathrm{T}}.
\end{aligned}
$$

利用 Matlab 中的命令 load dryer 2 生成数据, 假设噪声 $v(t)$ 满足 $v(t) \sim N(0, 0.1^2)$. 共采集 1 000 组数据, 其中 1 ～ 800 组数据用于建模, 801 ～ 1 000 组用来验证. 3 种算法的估计输出和系统真实输出如图 5 所示, 输出估计逼近率如表 3 所示.

图 5 和表 3 表明, 降维算法 (HIE、VE) 具有较好的辨识效果, 即通过降维可以有效提高辨识效率.

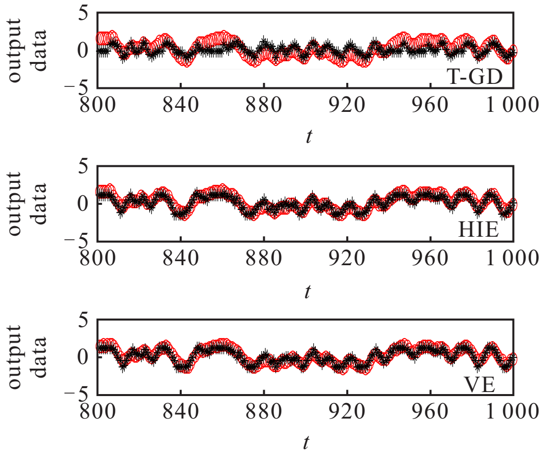

图5 估计输出和真实输出

表3 输出估计预测准确率

算法	τ/%
T-GD	68.87
HIE	74.54
VE	74.38

5 结 论

本文针对结构未知系统,提出了一种新的降维辨识方法——变量消去法. 首先将系统转化为两个子系统,通过最小二乘算法消去其中一个参数向量进而达到降维目的. 利用梯度算法和Aitken加速梯度算法辨识出降维模型的参数,并根据最小二乘算法计算出第一个子系统参数. 相对于传统梯度算法和递阶降维算法,变量消去法收敛速度更快,且可以利用Aitken加速提高参数收敛速度,增强对步长的鲁棒特性.

所提出方法也为强非线性系统[25-26]的辨识提供了一个新的思路,可以将强非线性系统用Volterra模型拟合,进而利用所提出方法获得强非线性系统的等价模型的参数. 尽管所提出方法具有以上优点,但也存在需要进一步解决的问题. 如,当系统存在丢失数据、时延时,降维模型的迭代方程发生变化,此时如何改进本文方法以使其能通过Aitken方法加速以及能否找出递阶辨识算法和本文方法在寻找全局最优点之间的区别等,都是值得研究的课题.

参考文献(References)

[1] Yu C P, Verhaegen M. Subspace identification of individual systems operating in a network (SI^2ON)[J]. IEEE Transactions on Automatic Control, 2018, 63(4): 1120-1125.

[2] Chakrabarti K, Gupta N, Chopra N. Iterative pre-conditioning for expediting the distributed gradient-descent method: The case of linear least-squares problem[J]. Automatica, 2022, 137: 110095.

[3] 崔琳琳, 沈冰冰, 葛志强. 基于混合变分自编码器回归模型的软测量建模方法[J]. 自动化学报, 2022, 48(2): 398-407.
(Cui L L, Shen B B, Ge Z Q. A mixture variational autoencoder regression model for soft sensor application[J]. Acta Automatica Sinica, 2022, 48(2): 398-407.)

[4] 杨心, 张广军, 李学仁, 等. 两个具有耦合时滞的分数阶复杂网络的延迟投影同步与参数辨识[J]. 控制与决策, 2022, 37(6): 1479-1488.
(Yang X, Zhang G J, Li X R, et al. Lag projective synchronization and parameter identification of two fractional-order complex networks with coupling delay[J]. Control and Decision, 2022, 37(6): 1479-1488.)

[5] Bates D M, Watts D G. Nonlinear regression analysis and Its applications[M]. Hoboken: John Wiley & Sons, Inc., 1988.

[6] 顾晓清, 倪彤光, 张聪, 等. 结构辨识和参数优化协同学习的概率TSK模糊系统[J]. 自动化学报, 2021, 47(2): 349-362.
(Gu X Q, Ni T G, Zhang C, et al. Probabilistic TSK fuzzy system in the simultaneous learning of structure identification and parameter optimization[J]. Acta Automatica Sinica, 2021, 47(2): 349-362.)

[7] Chen J, Ding F, Zhu Q M, et al. Interval error correction auxiliary model based gradient iterative algorithms for multirate ARX models[J]. IEEE Transactions on Automatic Control, 2020, 65(10): 4385-4392.

[8] Mu B Q, Chen T S, Ljung L. On asymptotic properties of hyperparameter estimators for kernel-based regularization methods[J]. Automatica, 2018, 94: 381-395.

[9] Birpoutsoukis G, Marconato A, Lataire J, et al. Regularized nonparametric Volterra kernel estimation[J]. Automatica, 2017, 82: 324-327.

[10] Chen T S, Martin S A, et al. System identification via sparse multiple kernel-based regularization using sequential convex optimization techniques [J]. IEEE Transactions on Automatic Control, 2014, 59(11): 2933-2945.

[11] 殷礼胜, 何怡刚, 董学平, 等. 交通流量VNNTF神经网络模型多步预测研究[J]. 自动化学报, 2014, 40(9): 2066-2072.
(Yin L S, He Y G, Dong X P, et al. Research on the multi-step prediction of Volterra neural network for traffic flow[J]. Acta Automatica Sinica, 2014, 40(9): 2066-2072.)

[12] 张玉梅, 胡小俊, 吴晓军, 等. 语音信号序列的Volterra预测模型[J]. 物理学报, 2015, 64(20): 121-133.

(Zhang Y M, Hu X J, Wu X J, et al. Volterra prediction model for speech signal series[J]. Acta Physica Sinica, 2015, 64(20): 121-133.)

[13] Gan M, Zhu H T, Chen G Y, et al. Weighted generalized cross-validation-based regularization for broad learning system[J]. IEEE Transactions on Cybernetics, 2022, 52(5): 4064-4072.

[14] Chen F W, Young P C. A simple robust method of fractional time-delay estimation for linear dynamic systems[J]. Automatica, 2022, 137: 110117.

[15] Ding F, Ma H, Pan J, et al. Hierarchical gradient- and least squares-based iterative algorithms for input nonlinear output-error systems using the key term separation[J]. Journal of the Franklin Institute, 2021, 358(9): 5113-5135.

[16] Zhou Y H, Zhang X, Ding F. Hierarchical estimation approach for RBF-AR models with regression weights based on the increasing data length[J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2021, 68(12): 3597-3601.

[17] Saad Y. Iterative methods for sparse linear systems[M]. New York: Society for Industrial and Applied Mathematics, 2003.

[18] Chen J, Wang D Q, Liu Y J, et al. Varying infimum gradient descent algorithm for agent-sever systems using different order iterative preconditioning methods[J]. IEEE Transactions on Industrial Informatics, 2022, 18(7): 4436-4446.

[19] 丁锋. 系统辨识——辨识方法性能分析[M]. 北京: 科学出版社, 2014.
(Ding F. System identification— Performance analysis for identification methods [M]. Beijing: Science Press, 2014.)

[20] Chen H F, Guo L. Identification and stochastic adaptive control[M]. Boston: Birkhäuser, 1991.

[21] Xu L, Ding F, Zhu Q M. Decomposition strategy-based hierarchical least mean square algorithm for control systems from the impulse responses[J]. International Journal of Systems Science, 2021, 52(9): 1806-1821.

[22] 陈晶, 朱全民. 有理模型辨识的两类新方法——混合迭代与柔性最小二乘法[J]. 控制与决策, 2022, 37(1): 58-66.
(Chen J, Zhu Q M. Two novel identification methods for rational models—Compound iterative algorithm and flexible least squares algorithm[J]. Control and Decision, 2022, 37(1): 58-66.)

[23] 潘雅璞, 谢莉, 杨慧中. 基于KRLS的非均匀采样非线性系统辨识[J]. 控制与决策, 2021, 36(12): 3049-3055.
(Pan Y P, Xie L, Yang H Z. Identification of non-uniformly sampled nonlinear systems based on KRLS[J]. Control and Decision, 2021, 36(12): 3049-3055.)

[24] Chen J, Gan M, Zhu Q M, et al. Robust standard gradient descent algorithm for ARX models using aitken acceleration technique[J]. IEEE Transactions on Cybernetics, 2022, 52(9): 9646-9655.

[25] Bai E W. Identification of linear systems with hard input nonlinearities of known structure[J]. Automatica, 2002, 38(5): 853-860.

[26] Giri F, Bai E W. Block-oriented nonlinear system identification[M]. London: Springer, 2010.

作者简介

陈晶 (1981−), 男, 教授, 博士, 从事系统辨识、机器学习等研究, E-mail: chenjing1981929@126.com;

程连元 (2000−), 男, 硕士生, 从事系统辨识、机器学习等研究, E-mail: chenglianyuan2000@126.com;

李俊红 (1980−), 女, 教授, 博士, 从事系统辨识、机器学习等研究, E-mail: missjunhong@163.com;

朱全民 (1955−), 男, 教授, 博士生导师, 从事有理模型的辨识与控制等研究, E-mail: quan.zhu@uwe.ac.uk.