

适于参数识别的多信道 QRD 自适应格型算法¹

欧阳缙 方惠均

(桂林电子工业学院二系 桂林 541004)

摘 要 本文基于格型滤波器的阶递归特性和 Givens 旋转算法的优越数值性能, 推导了两种多信道递归最小二乘格型算法。第一种算法的推导是直接基于对输入数据矩阵进行正交-三角分解, 并利用 Givens 旋转方法来计算其正交-三角分解。首先对输入数据矩阵进行预旋转, 然后重复利用单信道 Givens 格型算法, 便可得到第二种算法。两种算法都具有优越的数值性能, 尤其是对有限字长的稳健性。待估计的滤波器参数矢量可根据算法的内部变量直接提取, 而无需额外的三角阵进行后向代入求解运算。两信道参数识别的计算机模拟结果验证了本文的推导。

关键词 递归最小二乘格型算法, 正交-三角分解, 多信道信号, 自适应参数识别

中图分类号 TN911.72

1 引 言

应用 QR 分解 (QRD) 技术并结合正交 Givens 旋转变换来解最小二乘 (LS) 问题是近年来国际上比较活跃的研究课题。导出的自适应算法称为 QR 分解-递推最小二乘 QRD-RLS 算法, 主要原因是由于 QRD-RLS 算法比常规 RLS 算法有更优越的数值性能, 并十分适合于 VLSI 实现^[1-5]。尤其是无需通过求解自适应系统的抽头系数, 使可直接得到滤波器输出误差。对于只对滤波误差感兴趣的应用场合, 如噪声对消波束成形等特具吸引力。现已提出了几种单信道 QRD-RLS 算法, 并推广应用到了多信道情形^[6-9]。然而, 这些现有的多信道 QRD-RLS 算法都无法直接获得滤波器的抽头系数。如果要获得滤波器的抽头系数, 尚需再实施后向代入求解过程, 这一过程将带来较高的运算复杂度 $O(N^2P^2)$, 其中 P 为信道数, N 为每一信道中滤波器的抽头系数。为了克服这一缺陷, 本文提出两种快速多信道 QRD-RLS 格型算法, 都能有效地直接提取滤波器的抽头系数, 而无需实施后向代入运算, 运算复杂度分别为 $O(NP^3)$ 和 $O(NP^2)$ 。

2 问题描述

考虑有 P 个输入信道的数据序列 $\{x_1(n)\}, \{x_2(n)\}, \dots, \{x_P(n)\}$ 和一个期望数据序列 $\{y(n)\}, (n = 0 \sim k)$ 。在时间 k 时, $Y(k)$ 的 LS 估计由 $x_j(k), (j = 1 \sim P)$ 及其相应的时延形

¹ 1994-02-22 收到, 1994-11-14 定稿
电科院预研基金资助课题

式构成, 并使 LS 的代价函数

$$\xi(k) = \epsilon^T(k)\epsilon(k) \quad (1)$$

为最小。式中“ T ”表示转置,

$$\epsilon(k) = Y(k) - X(k)W(k) \quad (2)$$

为 LS 估计的误差矢量, $W(k)$ 为 $NP \times 1$ 抽头系数,

$$Y(k) = [\lambda^{k/2}y(0), \dots, \lambda^{1/2}y(k-1), y(k)]^T, \quad (3)$$

$$X(k) = \begin{bmatrix} \lambda^{k/2}X(0) & 0 & \dots & 0 \\ \lambda^{(k-1)/2}X(1) & \lambda^{k/2}X(0) & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ X(k) & X(k-1) & \dots & X(k-N+1) \end{bmatrix}, \quad (4)$$

其中 $0 \ll \lambda \leq 1$ 指数遗忘因子, $X(k) = [x_1(k), x_2(k), \dots, x_P(k)]$, 并假定各信道中抽头系数 N 均相等。

在 QR 算法中, 一个 $(k+1) \times (k+1)$ 维正交矩阵 $Q(k) = [Q_1(k)Q_2(k)]^T$ 用来把 $X(k)$ 三角化。由于 $Q(k)$ 是正交矩阵, 使 (1) 式最小, 等价于使 (2) 式的正交变换式的范数最小。用 $Q(k)$ 前乘 (2) 式, 即

$$Q(k)\epsilon(k) = \begin{bmatrix} Y_q(k) \\ Z_q(k) \end{bmatrix} - \begin{bmatrix} R(k) \\ \mathbf{o} \end{bmatrix} W(k), \quad (5)$$

其中 $R(k)$ 是 $NP \times NP$ 的上三角矩阵, 并有

$$X(k)R^{-1}(k) = Q_1(k). \quad (6)$$

由 LS 解满足正规方程和正交矩阵 $Q(k)$ 的性质, 我们不难得到 $Y(k)$ 的估计

$$\hat{Y}(k) = Q_1(k)Q_1^T(k)Y(k) = Q_1(k)Y_q(k) = \sum_{j=0}^{NP-1} q_j(k)y_{q,j}(k), \quad (7)$$

式中 $\{q_j(k)\}$ 是 $Q_1(k)$ 中的元素, 为后向预测误差矢量; $\{y_{q,j}(k)\}$ 是 $Y_q(k)$ 中的元素, 为归一化格型的抽头系数^[10]。

现在要获得抽头系数 $\{y_{q,j}^{(k)}\}$ ($j = 0 \sim NP-1$), 有两条途径, 一是直接对多信道输入数据矩阵进行 QR 分解, 利用扩充维数的前后向预测过程来更新相应的参数; 二是先对多信道输入数据矩阵进行预旋转^[11], 然后直接利用单信道 QRD-RLS 格型算法^[10]的结论。

3 算法推导

为简化起见, 我们仅以两输入信道信号 $P = 2$ 为例进行讨论。

3.1 算法 1

快速 RLS 算法的推导, 充分利用了输入数据矩阵的 Toeplitz 性质。由于多信道输入数据矩阵具有块 Toeplitz 性质, 因此 QR 分解可直接对多信道数据矩阵进行。

3.1.1 后向预测问题 在后向预测问题中, $\mathbf{X}(k-N) = [x_1(k-N), x_2(k-N)]$ 是由 $\mathbf{x}(k), \mathbf{x}(k-1), \dots, \mathbf{x}(k-N+1)$ 来估计的。把 $\mathbf{x}(k-N)$ 附加在数据矩阵 $\mathbf{X}_{2N}(k)$ (下标 $2N$ 表示列数) 的右边而组成

$$\mathbf{X}_{2N+2}(k) = [\mathbf{X}_{2N}(k) \mathbf{X}(k-N)]. \quad (8)$$

对 (8) 式进行 QR 分解可以把 $\mathbf{X}_{2N}(k)$ 变成上三角阵 $\mathbf{R}_{2N}(k)$, 再连乘正交矩阵 $\mathbf{Q}_{2,b}(k), \mathbf{Q}_{1,b}(k)$ 即可完成 (8) 式的三角化过程

$$\mathbf{Q}_{2,b}(k)\mathbf{Q}_{1,b}(k)\mathbf{Q}(k)\mathbf{X}_{2N+2}(k) = \begin{bmatrix} \mathbf{R}_{2N+2}(k) \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{2N}(k) & \mathbf{X}_{1,b}(k) & \mathbf{X}_{2,b}(k) \\ 0 & E_{11b,N}^{1/2}(k) & E_{12b,N}^{1/2}(k) \\ 0 & 0 & E_{22b,N}^{1/2}(k) \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad (9)$$

其中

$$\mathbf{R}_{2N+2}(k) = \begin{bmatrix} E_{11b,0}^{1/2}(k) & E_{12b,0}^{1/2}(k) & r_x & r_x & \dots & \dots \\ 0 & E_{22b,0}^{1/2}(k) & r_x & r_x & \dots & \dots \\ 0 & 0 & E_{11b,1}^{1/2}(k) & E_{12b,1}^{1/2}(k) & \dots & \dots \\ 0 & 0 & 0 & E_{22b,1}^{1/2}(k) & \dots & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & E_{11b,N}^{1/2}(k) & E_{12b,N}^{1/2}(k) \\ 0 & 0 & 0 & 0 & \dots & 0 & E_{22b,N}^{1/2}(k) \end{bmatrix}. \quad (10)$$

(10) 式中 r_x 为讨论中不关心的量。由正交变换的保模性质可得第零阶后向预测误差能量

$$E_{11b,0}(k) = \sum_{i=0}^k \lambda^{k-i} x_1^2(i) \triangleq E_1(k), \quad (11a)$$

$$E_{12b,0}(k) + E_{22b,0}(k) = \sum_{i=0}^k \lambda^{k-i} x_2^2(i) \triangleq E_2(k). \quad (11b)$$

上述过程完成了 $\mathbf{R}_{2N}(k)$ 到 $\mathbf{R}_{2N+2}(k)$ 的阶更新, 实质上得到了预测误差能量的阶更新。这些量在后面的归一化后向预测残差的时间、阶递推和旋转角度更新过程将起重要作用。

3.1.2 前向预测问题 在前向预测中, $\mathbf{X}(k+1) = [x_1(k+1), x_2(k+1)]$ 是由 $\mathbf{X}(k), \mathbf{X}(k-1), \dots, \mathbf{X}(k-N+1)$ 来估计的。把 $\mathbf{X}(k+1)$ 附加在数据矩阵 $\mathbf{X}_{2N}(k)$ 的右边, 可得

$$\mathbf{X}_{2N+2}(k+1) = \begin{bmatrix} \mathbf{X}(k+1) & 0 \dots 0 \\ & \mathbf{X}_{2N}(k) \end{bmatrix}. \quad (12)$$

通过对 $\begin{bmatrix} 0 \cdots 0 \\ \mathbf{X}_{2N+2}(k+1) \end{bmatrix}$ 进行一系列正交旋转变换后, 可以得到

$$\tilde{\mathbf{Q}}(k) \begin{bmatrix} 0 \cdots 0 \\ \mathbf{X}_{2N+2}(k+1) \end{bmatrix} = \begin{bmatrix} \lambda^{1/2} E_{11f,N}^{1/2}(k) & \lambda^{1/2} E_{12f,N}^{1/2}(k) & 0 \cdots 0 \\ 0 & \lambda^{1/2} E_{22f,N}^{1/2}(k) & 0 \cdots 0 \\ \mathbf{X}_{1f}(k+1) & \mathbf{X}_{2f}(k+1) & \mathbf{R}_{2N}(k) \\ \mathbf{o} & \mathbf{o} & \\ e_{1f,N}(k+1) & e_{2f,N}(k+1) & \mathbf{o} \end{bmatrix}. \quad (13)$$

现在再通过一系列正交旋转变换, 把 $e_{1f,N}(k+1)$, $e_{2f,N}(k+1)$, $\mathbf{X}_{1f}(k+1)$ 和 $\mathbf{X}_{2f,N}(k+1)$ 旋转到 (13) 式的最上面两行, 同时使这些元素处化为零, 即得

$$\begin{bmatrix} 1 & 0 \\ 0 & \mathbf{Q}_{\alpha_2}(k+1) \end{bmatrix} \mathbf{Q}_{\alpha_1}(k+1) \mathbf{Q}_{\varphi}(k+1) (13) \text{ 式} = \begin{bmatrix} \mathbf{R}_{2N+2}(k+1) \\ \mathbf{o} \end{bmatrix}, \quad (14)$$

其中 $\mathbf{Q}_{\varphi}(k+1)$ 为 2 个 2×2 正交 Givens 矩阵, $\mathbf{Q}_{\alpha_2}(k+1)$ 和 $\mathbf{Q}_{\alpha_1}(k+1)$ 均为 $2N$ 个 2×2 正交 Givens 矩阵, 其中之一可表示为

$$\mathbf{Q}_{\alpha_{i,j}}(k+1) = \begin{bmatrix} \cos \alpha_{i,j}(k+1) & \sin \alpha_{i,j}(k+1) \\ -\sin \alpha_{i,j}(k+1) & \cos \alpha_{i,j}(k+1) \end{bmatrix}, \quad (i=1,2; j=1,\dots,2N). \quad (15)$$

通过求解 (14) 式即可得 $\mathbf{Q}_{\varphi}(k+1)$, $\mathbf{Q}_{\alpha_{i,j}}(k+1)$ ($i=1,2; j=1,\dots,2N$). 而 $-\sin \alpha_{i,j}(k+1)$ 即为格型滤波器的反射系数^[10].

3.1.3 更新旋转矩阵 $\hat{\mathbf{Q}}(k+1)$ 在 QR 算法中, 旋转矩阵 $\mathbf{Q}(k+1)$ 可以通过递推求得^[1], 但由于实现递推的具体方法不同, 从而导出了不同的 QRD-RLS 算法^[3~5,10]. 为了获得 $\hat{\mathbf{Q}}(k+1)$, 让我们考虑

$$\begin{aligned} \mathbf{R}_{2N}(k+1) &= \mathbf{Q}(k+1) \mathbf{X}_{2N}(k+1) = \hat{\mathbf{Q}}(k+1) \begin{bmatrix} \mathbf{Q}(k) & 0 \\ 0 & 1 \end{bmatrix} \mathbf{X}_{2N}(k+1) \\ &= \hat{\mathbf{Q}}(k+1) \begin{bmatrix} \lambda^{1/2} E_{11b,0}^{1/2}(k) & \lambda^{1/2} E_{12b,0}^{1/2}(k) & \tau_x & \cdots \\ 0 & \lambda^{1/2} E_{22b,0}^{1/2}(k) & \tau_x & \cdots \\ \vdots & \vdots & \vdots & \cdots \\ x_1(k+1) & x_2(k+1) & \cdots & \cdots \end{bmatrix}. \end{aligned} \quad (16)$$

根据 (16) 式可求得 $\hat{\mathbf{Q}}(k+1)$ 中的 $2N$ 个 2×2 Givens 旋转角, 即完成 $\mathbf{Q}(k+1)$ 的更新。

类似于文献 [10], 对于多信道归一化预测格型, 可得下面的归一化后验误差递推关系式

$$\begin{bmatrix} \bar{\varepsilon}_{1f,i-1}(k+1) \\ \bar{\varepsilon}_{2f,i-1}(k+1) \\ \bar{\varepsilon}_{b,i+1}(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_{2,i}(k+1) & \sin \alpha_{2,i}(k+1) \\ 0 & -\sin \alpha_{2,i}(k+1) & \cos \alpha_{2,i}(k+1) \end{bmatrix} \begin{bmatrix} \cos \alpha_{1,i}(k+1) & 0 & \sin \alpha_{1,i}(k+1) \\ 0 & 1 & 0 \\ -\sin \alpha_{1,i}(k+1) & 0 & \cos \alpha_{1,i}(k+1) \end{bmatrix} \begin{bmatrix} \bar{\varepsilon}_{1f,i}(k+1) \\ \bar{\varepsilon}_{2f,i}(k+1) \\ \bar{\varepsilon}_{b,i-1}(k) \end{bmatrix}. \quad (17)$$

注意到 (17) 式中的 $\bar{e}_{1f,2N}(k+1), \bar{e}_{2f,2N}(k+1)$ 可由前向预测获得。这样根据 k 时的归一化后向预测残差 $\bar{e}_{b,i-1}(k)$, 就可得到 $\bar{e}_{b,i+1}(k+1), (i = 2N, \dots, 1)$, 并有

$$\bar{e}_{b,0}(k+1) = \bar{e}_{1f,0}(k+1), \quad \bar{e}_{b,1}(k+1) = \bar{e}_{2f,0}(k+1). \quad (18)$$

这时似然变量可以得到更新, 即

$$\gamma_{2N-1}(k+1) = [1 - \sum_{i=0}^{2N-1} \bar{e}_{b,i}(k+1)]^{1/2} = \prod_{i=0}^{2N-1} \cos \theta_i(k+1). \quad (19)$$

完整的两信道 QRD-RLS 格型算法概括如表 1 所示, 相应的格型结构如图 1 所示。

表 1 算法 1

(1) 初始化部分:

$$\gamma_{2N-1}(-1) = 1; \theta_j(-1) = x_{1f,j}(-1) = x_{2f,j}(-1) = y_{q,j}(-1) = 0;$$

$$\bar{e}_{b,j}(-1) = 0, j = 0, \dots, 2N-1;$$

$$E_{11f,2N}^{1/2}(-1) = E_{12f,2N}^{1/2}(-1) = E_{22f,2N}^{1/2}(-1) = 0 \text{ (exact start)}$$

$$\text{或 } E_{11f,2N}^{1/2}(-1) = E_{22f,2N}^{1/2}(-1) = \text{小的正常数}; E_{12f,2N}^{1/2}(-1) = 0 \text{ (soft start).}$$

$k > 0$ 信道输入: $x_1(k), x_2(k)$;

$$\text{期望输入: } y(k); e_{1f,0}(k) = x_1(k), e_{2f,0}(k) = x_2(k), e_{q,0}(k) = y(k).$$

预测部分

(2) $j = 0, 1, \dots, 2N-1$

$$\begin{bmatrix} x_{1f,j}(k) & x_{2f,j}(k) \\ e_{1f,j+1}(k) & e_{2f,j+1}(k) \end{bmatrix} = \begin{bmatrix} \cos \theta_j(k-1) & \sin \theta_j(k-1) \\ -\sin \theta_j(k-1) & \cos \theta_j(k-1) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} x_{1f,j}(k-1) & \lambda^{1/2} x_{2f,j}(k-1) \\ e_{1f,j}(k) & e_{2f,j}(k) \end{bmatrix}$$

(3) $j = 2N$

$$\begin{bmatrix} E_{11f,2N}^{1/2}(k) & E_{12f,2N}^{1/2}(k) \\ 0 & E_{22f,2N}^{1/2}(k) \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_2 & \sin \phi_2 \\ 0 & -\sin \phi_2 & \cos \phi_2 \end{bmatrix} \begin{bmatrix} \cos \phi_1 & 0 & \sin \phi_1 \\ 0 & 1 & 0 \\ -\sin \phi_1 & 0 & \cos \phi_1 \end{bmatrix} \begin{bmatrix} \lambda^{1/2} E_{11f,2N}^{1/2}(k-1) & \lambda^{1/2} E_{12f,2N}^{1/2}(k-1) \\ 0 & \lambda^{1/2} E_{22f,2N}^{1/2}(k-1) \\ e_{1f,2N}(k) & e_{2f,2N}(k) \end{bmatrix}$$

(4) $j = 2N, \dots, 2, 1$

$$\begin{bmatrix} E_{11f,j-1}^{1/2}(k) & E_{12f,j-1}^{1/2}(k) \\ 0 & E_{22f,j-1}^{1/2}(k) \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_{2j}(k) & \sin \alpha_{2j}(k) \\ 0 & -\sin \alpha_{2j}(k) & \cos \alpha_{2j}(k) \end{bmatrix} \begin{bmatrix} \cos \alpha_{1j}(k) & 0 & \sin \alpha_{1j}(k) \\ 0 & 1 & 0 \\ -\sin \alpha_{1j}(k) & 0 & \cos \alpha_{1j}(k) \end{bmatrix} \begin{bmatrix} E_{11f,j}^{1/2}(k) & E_{12f,j}^{1/2}(k) \\ 0 & E_{22f,j}^{1/2}(k) \\ x_{1f,j-1}(k) & x_{2f,j-1}(k) \end{bmatrix}$$

(5) $j = 2N, \dots, 2, 1$

$$\begin{bmatrix} \bar{e}_{1f,j-1}(k) \\ \bar{e}_{2f,j-1}(k) \\ \bar{e}_{b,j+1}(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_{2j}(k) & \sin \alpha_{2j}(k) \\ 0 & -\sin \alpha_{2j}(k) & \cos \alpha_{2j}(k) \end{bmatrix} \begin{bmatrix} \cos \alpha_{1j}(k) & 0 & \sin \alpha_{1j}(k) \\ 0 & 1 & 0 \\ -\sin \alpha_{1j}(k) & 0 & \cos \alpha_{1j}(k) \end{bmatrix} \begin{bmatrix} \gamma_{2N-1}(k-1) \sin \phi_1 \\ \gamma_{2N-1}(k-1) \cos \phi_1 \sin \phi_2 \\ \bar{e}_{b,j-1}(k-1) \end{bmatrix}$$

$$\bar{e}_{b,0}(k) = \bar{e}_{1f,0}(k), \bar{e}_{b,1}(k) = \bar{e}_{2f,0}(k)$$

$$\gamma_{2N-1}(k) = \left[1 - \sum_{j=0}^{2N-1} \bar{e}_{b,j}^2(k) \right]^{1/2}$$

(6) $j = 2N - 1, \dots, 1, 0$

$$\begin{bmatrix} \gamma_{j-1}(k) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_j(k) & \sin \theta_j(k) \\ -\sin \theta_j(k) & \cos \theta_j(k) \end{bmatrix} \begin{bmatrix} \gamma_j(k) \\ \bar{e}_{b,j}(k) \end{bmatrix}$$

联合过程部分

(7) $j = 0, 1, \dots, 2N - 1$

$$\begin{bmatrix} y_{q,j}(k) \\ e_{q,j+1}(k) \end{bmatrix} = \begin{bmatrix} \cos \theta_j(k) & \sin \theta_j(k) \\ -\sin \theta_j(k) & \cos \theta_j(k) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} y_{q,j}(k-1) \\ e_{q,j}(k) \end{bmatrix}$$

$$e(k) = \gamma_{2N-1}(k) e_{q,2N}(k)$$

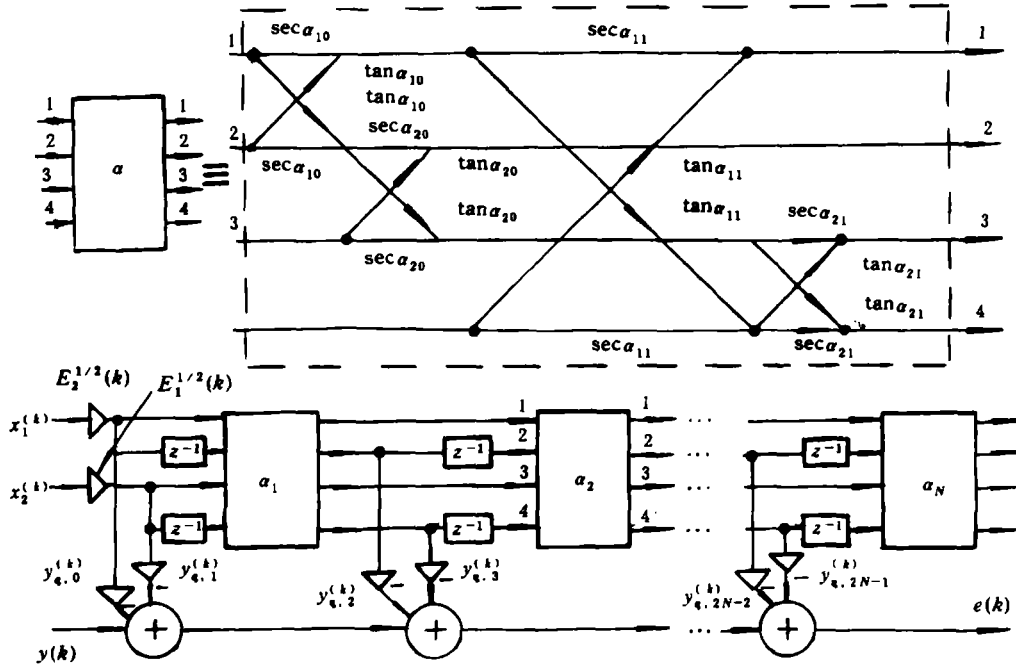


图1 联合过程两信道归一化QRD-RLS格型滤波器

3.2 算法 2

QRD-RLS 算法的一个突出特点, 就是其滤波误差可以直接通过 Givens 旋转得到。这样我们可以像单信道 QRD-RLS 算法中联合过程估计部分那样, 用 $x_1(k)$ 及其时延形式来估计 $x_2(k)$, 我们称之为预旋转^[11]。经这样处理后, 完全可以利用单信道 QRD-RLS 格型算法^[10]的结果。

考虑预旋转

$$\begin{bmatrix} X_2(k+1) \\ e_2(k+1) \end{bmatrix} = \hat{Q}_1(k+1) \begin{bmatrix} \lambda^{1/2} X_2(k) \\ x_2(k+1) \end{bmatrix}, \quad (20)$$

其中 $\hat{Q}_1(k+1)$ 是与第一信道相对应的 N 个 2×2 Givens 旋转矩阵。这样与第二信道有关的前向预测问题可表示为

$$\begin{bmatrix} X_{2f}(k+1) \\ e_{2f}(k+1) \end{bmatrix} = \hat{Q}_2(k) \begin{bmatrix} \lambda^{1/2} X_{2f}(k) \\ e_2(k+1)\gamma_{1,N-1}(k+1) \end{bmatrix}. \quad (21)$$

对于联合过程估计问题, 我们有

$$\begin{bmatrix} Y_q(k+1) \\ e_q(k+1) \end{bmatrix} = \hat{Q}_2(k+1)\hat{Q}_1(k+1) \begin{bmatrix} \lambda^{1/2} Y_q(k) \\ y(k+1) \end{bmatrix}, \quad (22)$$

其中 $\hat{Q}_2(k+1)$, $\hat{Q}_1(k+1)$ 可分别由单信道 QRD-RLS 格型算法更新^[10]。

这时滤波输出误差为

$$e(k+1) = \gamma_{2,N-1}(k+1)\gamma_{1,N-1}(k+1)e_q(k+1). \quad (23)$$

直接结合文献 [10] 的结果, 完整的两信道 QRD-RLS 格型算法概括如表 2 所示。这种方法的详细分析过程可参阅文献 [11]。

表 2 算法 2

(1) 初始化部分同算法 1

$k > 0$ 信道输入: $x_1(k), x_2(k)$; 期望输入: $y(k); e_{1f,0}(k) = x_1(k), e_{2,0}(k) = x_2(k)$

预测部分

$i = 1$

(2) $j = 0, 1, \dots, N-1$

$$\begin{bmatrix} x_{if,j}(k) \\ e_{if,j+1}(k) \end{bmatrix} = \begin{bmatrix} \cos \theta_{i,j}(k-1) & \sin \theta_{i,j}(k-1) \\ -\sin \theta_{i,j}(k-1) & \cos \theta_{i,j}(k-1) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} x_{if,j}(k-1) \\ e_{if,j}(k) \end{bmatrix}$$

(3) $j = N$

$$\begin{bmatrix} E_{if,N}^{1/2}(k) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_{i,N}(k-1) & \sin \theta_{i,N}(k-1) \\ -\sin \theta_{i,N}(k-1) & \cos \theta_{i,N}(k-1) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} E_{if,2N}^{1/2}(k-1) \\ e_{if,N}(k) \end{bmatrix}$$

(4) $j = N, \dots, 2, 1$

$$\begin{bmatrix} E_{if,j-1}^{1/2}(k) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \alpha_{i,j-1}(k) & \sin \alpha_{i,j-1}(k) \\ -\sin \alpha_{i,j-1}(k) & \cos \alpha_{i,j-1}(k) \end{bmatrix} \begin{bmatrix} E_{if,j}^{1/2}(k) \\ x_{if,j-1}(k) \end{bmatrix}$$

$$\bar{e}_{if,N} = \gamma_{i,N-1}(k-1) \sin \theta_{i,N}(k-1)$$

(5) $j = N, \dots, 2, 1$

$$\begin{bmatrix} \bar{e}_{if,j-1}(k) \\ \bar{e}_{ib,j}(k) \end{bmatrix} = \begin{bmatrix} \cos \alpha_{i,j}(k) & \sin \alpha_{i,j}(k) \\ -\sin \alpha_{i,j}(k) & \cos \alpha_{i,j}(k) \end{bmatrix} \begin{bmatrix} \bar{e}_{if,j}(k) \\ \bar{e}_{ib,j-1}(k) \end{bmatrix}$$

$$\bar{e}_{ib,0}(k) = \bar{e}_{if,0}(k)$$

$$\gamma_{i,N-1}(k) = \left[1 - \sum_{j=0}^{N-1} \bar{e}_{ib,j}(k) \right]^{1/2}$$

(6) $j = N-1, \dots, 1, 0$

$$\begin{bmatrix} \gamma_{i,j}(k) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_{i,j}(k) & \sin \theta_{i,j}(k) \\ -\sin \theta_{i,j}(k) & \cos \theta_{i,j}(k) \end{bmatrix} \begin{bmatrix} \gamma_{i,j+1}(k) \\ \bar{e}_{ib,j}(k) \end{bmatrix}$$

(7) $j = 0, 1, \dots, N-1$

$$\begin{bmatrix} x_{2,j}(k) \\ e_{2,j+1}(k) \end{bmatrix} = \begin{bmatrix} \cos \theta_{1,j}(k) & \sin \theta_{1,j}(k) \\ -\sin \theta_{1,j}(k) & \cos \theta_{1,j}(k) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} x_{2,j}(k-1) \\ e_{2,j}(k) \end{bmatrix}$$

(8) $i = 2, e_{2f,0}(k) = e_{2,N}(k) \gamma_{1,N-1}(k)$, 重复上述 (2) ~ (6) 的过程。

联合过程部分

$$\begin{bmatrix} y_q(k) \\ e_q(k) \end{bmatrix} = Q_2(k) Q_1(k) \begin{bmatrix} \lambda^{1/2} y_q(k-1) \\ y(k) \end{bmatrix}$$

$$e(k) = e_q(k) \gamma_{2,N-1}(k) \gamma_{1,N-1}(k)$$

4 计算机模拟结果和评价

为了说明算法的参数识别性能, 我们考虑一个两输入变量的系统^[8]

$$y(k) = a_0 x_1(k) + a_1 x_1(k-1) + b_0 x_2(k) + b_1 x_2(k-1) + v(k),$$

其中输入序列 $\{x_1(k)\}$ 和 $\{x_2(k)\}$ 分别是由两个不相关的白噪声激励产生的二阶自回归过程, $v(k)$ 为加性高斯序列。每个输入序列对加性噪声的功率比均为 0dB。 $\lambda = 0.99$, $N = 2$ 。 $a_0 = a_1 = 1$, $b_0 = 0.5$, $b_1 = 1.5$ 。

对于算法 1, LS 估计的抽头系数为 $\hat{a}_j = y_{q,2j}(k)/E_1^{1/2}(k)$, $\hat{b}_j = y_{q,2j+1}(k)/E_2^{1/2}(k)$, ($j = 0, 1, \dots, N-1$)。

对于算法 2, LS 估计的抽头系数为 $\hat{a}_j = y_{q,j}(k)/E_1^{1/2}(k)$, $\hat{b}_j = y_{q,j+N}(k)/E_2^{1/2}(k)$, ($j = 0, 1, \dots, N-1$)。

图 2 示出了 20 次独立运行的平均结果。结果表明, 本文提出的两种多信道 QRD-RLS 算法都较好地完成了抽头系数的 LS 估计问题。

自适应算法的运算复杂度是人们普遍关心的问题之一。本文提出的算法的运算复杂度概括如表 3 (对于 P 个信道)。目前对于多信道算法一般均需 $O(NP^2)$ 运算量。而对于 QRD-RLS 算法, 如要获得 LS 估计的抽头系数, 还需 $O(N^2P^2)$ 运算量用于后向代入求解过程, 以提取 LS 估计的抽头系数, 这对于大的 N (而通常应用正是这种情形), 运算量将很高, 并且这一后向代入求解过程也不利于实时递推。本文的算法 1 运算量为 $O(NP^3)$, 这对于大的 P 复杂度较高, 但对于如图像处理、自适应均衡 ($p = 2$) 等 P 较小的应用场合, 更由于该算法能实时地直接获得抽头系数, 无需后向代入运算过程, 它也还是一种高效的算法。算法 2 具有较低的运算量 $O(NP^2)$, 并能直接提取抽头系数, 但该算法实质上是一种串行运算, 不适合于并行处理。

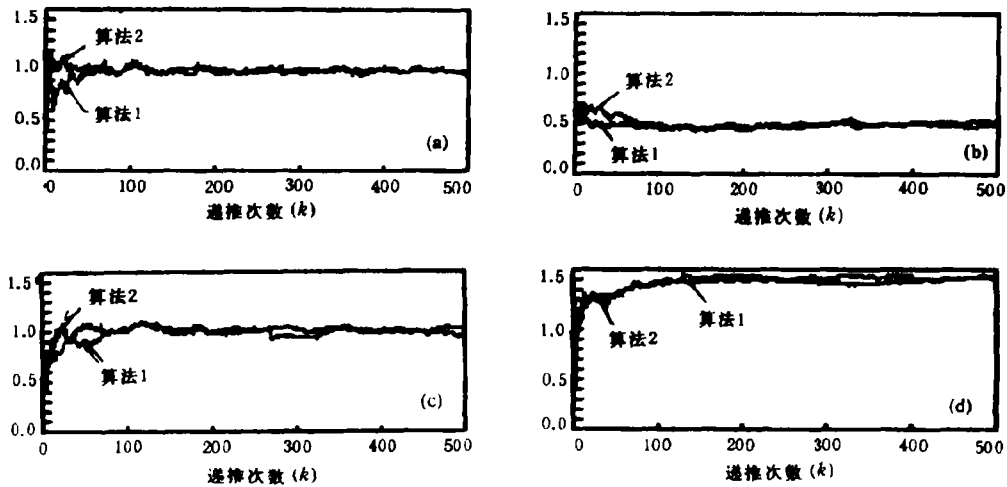


图 2 两信道参数识别的计算机模拟结果

表 3 算法的运算复杂度

运算量	算法 1	算法 2
乘法次数	$(2P^3 + 10P^2 + 4P)N + 4P^2 + 4P$	$2P^2N + 15PN + 2P$
平方根次数	$P^2N + PN + P$	$2PN + P$
除法次数	$P^2N + PN + P$	$2PN$
总的运算量	$(2P^3 + 12P^2 + 6P)N + 4P^2 + 6P$	$2P^2N + 19PN + 3P$

5 结束语

本文从两个不同的角度推导了两种多信道 QRD-RLS 自适应格型算法, 能直接根据算法的内部变量提取滤波器的抽头系数, 从而克服了目前所有的多信道 QRD-RLS 算法尚需额外的三角阵来求抽头系数的缺陷, 因此特别适合于参数识别的应用场合。算法的推导仅利用 Givens 旋转变换直接对输入数据矩阵进行 QR 分解来求解多信道 RLS 问题, 具有优越的数值性能和高度的模块化结构, 既适合于 VLSI 实现, 也适合于用市场上现有的信号处理器实现, 因此算法会有较好的应用前景。

参 考 文 献

- [1] Haykin S. Adaptive Filter Theory. Englewood Cliffs, NJ: Prentice Hall, 1985.
- [2] Liu K R, et al. IEEE Trans. on CAS, 1991, CAS-38(6): 625-636.
- [3] Cioffi J. M. IEEE Trans. on ASSP, 1990, ASSP-38(4): 631-653.
- [4] Ling F. IEEE Trans. on SP, 1991, SP-39(7): 1541-1551.
- [5] Proudler I K, et al. IEE Proc.-F, 1991, 138(4): 341-353.
- [6] Lewis P S. IEEE Trans. on ASSP, 1990, ASSP-38(3): 421-432.
- [7] Zhao K, et al. Multichannel Givens Lattice adaptive algorithm. in Proc. IEEE ICASSP'91, Ontario, Canada: 1991, 1849-1852.

- [8] Ballanger M G, *et al.* Signal Processing, 1991, 22(2): 115–126.
- [9] Proudler I K, *et al.* Computationally efficient QRD-based wideband beamforming. in Proc. IEEE ICASSP'90, New Mexico, USA: 1990, 1799–1802.
- [10] Regalia P A, *et al.* IEEE Trans. on SP, 1991, SP-39(4): 879–891.
- [11] 欧阳缙. 一种多信道 FAR RLS 算法的研究: [硕士论文]. 西安: 西安电子科技大学, 1991, 12.

MULTICHANNEL QRD-BASED ADAPTIVE LATTICE ALGORITHMS FOR THE PARAMETER IDENTIFICATION PROBLEM

Ouyang Shan Fang Huijun

(Guilin Institute of Electronic Technology, Guilin 541004)

Abstract A pair of multichannel recursive least squares (RLS) adaptive lattice algorithms based on the order recursive properties of lattice filters and the superior numerical properties of the Givens algorithms is derived. The derivation of the first algorithm is based on the QR decomposition of the input data matrix directly, and Givens rotations approach is used to compute the QR decomposition. Using first a prerotation of the input data matrix and then a repetition of the single channel Givens lattice algorithm, the second algorithm can be obtained. Both algorithms have superior numerical properties, particularly the robustness to wordlength limitations. The parameter vector to be estimated can be extracted directly from internal variables in the present algorithms without a backsolve operation with an extra triangular array. The results of computer simulation of the parameter identification of a two-channel system are presented to confirm efficiently the derivation.

Key words Recursive least squares lattice algorithm, QR decomposition, Multichannel signals, Adaptive parameter identification

欧阳缙: 男, 1960年生, 讲师, 现主要从事信号与信息处理的研究工作.

方惠均: 男, 1936年生, 教授, 现主要从事信号与信息处理的研究工作.