

基于状态空间模型广义预测控制的并行算法

慕德俊 戴冠中

(西北工业大学自动控制系·西安, 710072)

摘要: 本文首先基于脉动(systolic)阵列结构,提出了一种实时参数辨识的并行算法,然后推导出基于状态空间模型广义预测控制(GPC)的两种新算法,这两种算法都可以通过阵列结构并行实现。

关键词: 系统辨识; 广义预测控制; 并行算法; systolic 阵列

1 引言

在信号处理和自校正控制中,往往对参数辨识的实时性要求很高,若采用传统的 Systolic 方法实现 Yang^[1]提出的一种辨识算法,则由于 QR 分解时阵列中的数据流与回代计算时所要求的数据流流向相反,不能使两过程流水线进行. Yang 采用一个附加的线性阵列执行回代计算,但由于数据需从三角阵列转换到线性阵列中,造成吞吐率下降. 本文基于最小平方逆分解思想,给出了一种求解递推最小平方辨识方法. 这种方法在计算时避免了回代过程,与阵列结构^[2]结合并对单元功能作适当修正,可使辨识的实时性得到提高. 文中还推导出基于状态空间模型广义预测控制(GPC)的两种算法,一种是递推法,它将 GPC 问题化为求解离散形式的 Riccati 方程,可通过阵列结构并行实现,分析表明此算法和结构既减少阵列结构的大小又可使计算的实时性得到很大提高. 另一种算法是通过文献[3]算法的改进,可用一组阵列结构并行实现.

2 系统辨识的并行方法

对于 CARIMA 模型:

$$A(q^{-1})y(k) = B(q^{-1})u(k-1) + C(q^{-1})\zeta(k)/\Delta. \quad (1a)$$

式中

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \cdots + a_nq^{-n},$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-2} + \cdots + b_nq^{-n},$$

$$C(q^{-1}) = 1 + c_1q^{-1} + c_2q^{-2} + \cdots + c_nq^{-n}, \quad \Delta = 1 - q^{-1}.$$

(1a)式还可表示为:

$$\Delta y(k) = \Phi^T(k)\theta + \zeta(k). \quad (1b)$$

式中的输入-输出观测向量 Φ 和未知参数向量 θ 分别为:

$$\Phi(k) = [-\Delta y(k-1), \cdots, -\Delta y(k-na), \Delta u(k-1), \cdots,$$

$$\Delta u(k-nb), \zeta(k-1), \cdots, \zeta(k-nc)]^T,$$

$$\theta = [a_1, a_2, \cdots, a_n, b_0, b_1, \cdots, b_n, c_1, c_2, \cdots, c_n]^T,$$

$$\zeta(k) = \Delta y(k) - \Phi^T(k)\hat{\theta}(k-1).$$

(1b)式中未知参数向量 θ 的估计 $\hat{\theta}$ 的递推算式为^[7]:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[y(k) - \Phi^T(k)\hat{\theta}(k-1)],$$

$$K(k) = \frac{P(k-1)\Phi(k)}{\rho + \Phi^T(k)P(k-1)\Phi(k)},$$

$$P(k) = \frac{1}{\rho}[I - K(k)\Phi^T(k)]P(k-1).$$

式中, $0 < \rho < 1$, 称为遗忘因子.

$$P(k) = [[\Phi(1) \ \dots \ \Phi(k-1)][\Phi(1) \ \dots \ \Phi(k-1)^T]]^{-1}.$$

对 $P(k)$ 进行Cholesky逆分解可得 $P^{-1}(k) = R^T(k)R(k)$.

式中, $R(k)$ 为上三角阵.

$$\text{定义} \quad d^T(k) = \Phi^T(k)R^{-1}(k-1)/\sqrt{\rho}, \quad (2)$$

$$Q \begin{bmatrix} -d(k) \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ g(k) \end{bmatrix}. \quad (3)$$

式中 $g(k) = (1 + d^T(k)d(k))^{1/2}$, 可以证明, 若(3)式成立则有:

$$Q \begin{bmatrix} \frac{1}{\sqrt{\rho}}R^{-T}(k-1) \\ 0 \end{bmatrix} = \begin{bmatrix} R^{-T}(k) \\ f^T(k) \end{bmatrix}. \quad (4)$$

$$\text{式中} \quad f(k) = -\frac{P(k-1)\Phi(k)}{\sqrt{\rho}(\rho + \Phi^T(k)P(k-1)\Phi(k))^{1/2}},$$

$$\text{则} \quad K(k) = -f(k)/g(k),$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) - [y(k) - \Phi^T(k)\hat{\theta}(k-1)]f(k)/g(k). \quad (5)$$

每次辨识需计算(2)~(5)式; 输入 $\Phi^T(k)$, $y(k)$ 与图1所示阵列里的 $R^{-1}(k-1)$, $\hat{\theta}(k-1)$ 乘加, 计算(2)式及 $h(k) = y(k) - \Phi^T(k)\hat{\theta}(k-1)$; 左上方输入1, 与 $d(k)$ 在主对角线的单元上进行QR分解完成(3)式, 从右下方输出 $g(k)$; 旋转参数修正 $R^{-1}(k-1)$ 完成(4)式; 输入 $h(k)$ 及 $g(k)$ 与右边阵列里的 $\hat{\theta}(k-1)$ 一起, 计算(5)式. 连续辨识计算时, 每个采样点只需 $O(n^0)$ 个计算步.

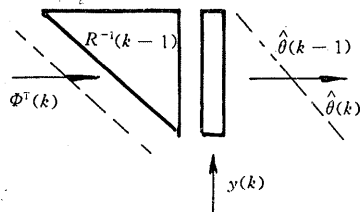


图1 实现辨识的三角阵列结构

3 广义预测控制的并行方法(一)

设下列单输入-单输出的线性离散时间系统:

$$X(k+1) = AX(k) + Bu(k), \quad (6)$$

$$y(k) = CX(k). \quad (7)$$

式中 $X, B, C \in \mathbb{R}^n$, $A \in \mathbb{R}^{n \times n}$.

$$\text{若目标函数取为:} \quad J = \sum_{j=1}^N y^2(k+j) + r \sum_{j=0}^{N-1} u^2(k+j), \quad r \geq 0. \quad (8)$$

应用多值状态空间法^[3], 可得广义预测的当前控制^[4](设预测长度 N 大于未来控制长度 M):

$$u(k) = -ZVX(k). \quad (9)$$

式中

$$Z = e(W^T W + rI)^{-1} W^T, \quad (10)$$

$$W = \begin{bmatrix} CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ CA^{M-1}B & CA^{M-2}B & CA^{M-3}B & \cdots & CB \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ CA^{N-1}B & CA^{N-2}B & CA^{N-3}B & \cdots & CA^{N-M}B \end{bmatrix}, \quad (11)$$

$$V = [(CA)^T, (CA^2)^T, \dots, (CA^N)^T]^T, \quad (12)$$

$$e = [1, 0_{1 \times (M-1)}].$$

基于 LQ 优化的思想^[5], 通过对(10)式递推计算, 将 GPC 问题化为求解离散形式的 Riccati 方程. 这样, 可用类似文献^[6]的方法并行计算. 这种方法稍加修改, 还可对多变量情况下的 GPC 问题进行求解. 同时, 由于这种方法将(10)式的矩阵计算变为子矩阵的递推计算, 因此减少了所需的阵列结构, 这种优势在多变量情况下更加突出. 递推算法的推导中, 定义:

$$C_{i+1} = \begin{bmatrix} C \\ C_i A \end{bmatrix}, \quad C_1 = C, \quad V_i = C_i A, \quad 1 \leq i \leq N-1, \quad (13)$$

$$W_{i+1} = \begin{bmatrix} CB \\ C_i AB \end{bmatrix}, \quad W_1 = CB, \quad 1 \leq i \leq N-M, \quad (14)$$

$$W_{i+1} = \begin{bmatrix} CB & 0 \\ C_i AB & W_i \end{bmatrix}, \quad W_{N+1} = W, \quad i \geq N-M+1. \quad (15)$$

由(13), (15)式得

$$C_{i+1}^T C_{i+1} = C^T C + A^T C_i^T C_i A, \quad (16)$$

$$W_{i+1}^T C_{i+1} = \begin{bmatrix} B^T C^T C + B^T A^T C_i^T C_i A \\ W_i^T C_i A \end{bmatrix}. \quad (17)$$

这时(9)式的 ZV 可递推表示如下:

$$Z_i V_i = Z_i C_i A = e_i (W_i^T W_i + rI)^{-1} W_i^T C_i A, \quad (18)$$

$$e_i = 1, \quad 1 \leq i \leq N-M+1, \quad e_i = [1, 0_{1 \times (i-N+M-1)}], \quad i > N-M+1.$$

从(13)~(16)式可以看出:

$$u(k) = -Z_{N+1} C_{N+1} A X(k).$$

由(14), (16), (18)式得

$$Z_{i+1} C_{i+1} = [B^T C^T C B + r + B^T A^T X_i A B]^{-1} [B^T C^T C + B^T A^T X_i A], \quad 1 \leq i \leq N-M. \quad (19)$$

式中

$$X_i = C_i^T C_i,$$

$$X_{i+1} = C_{i+1}^T C_{i+1} = C^T C + A^T X_i A. \quad (20)$$

$i \geq N-M+1$ 时, 应用矩阵的分块求逆定理及(15)式得:

$$\begin{aligned} [W_{i+1}^T W_{i+1} + rI]^{-1} &= \begin{bmatrix} B^T C^T C B + B^T A^T C_i^T C_i A B + r & B^T A^T C_i^T W_i \\ W_i^T C_i A B & W_i^T W_i + rI \end{bmatrix}^{-1} \\ &= \begin{bmatrix} 1 & 0 \\ -[W_i^T W_i + rI]^{-1} W_i^T C_i A B & I \end{bmatrix} \end{aligned}$$

$$\begin{aligned} & \cdot \begin{bmatrix} [B^T C^T C B + r + B^T A^T X_i A B]^{-1} & 0 \\ 0 & [W_i^T W_i + rI]^{-1} \end{bmatrix} \\ & \cdot \begin{bmatrix} 1 & -B^T A^T C_i^T W_i (W_i^T W_i + rI)^{-1} \\ 0 & I \end{bmatrix}, \quad i \geq N - M + 1. \end{aligned} \quad (21)$$

$$X_i = C_i^T C_i - C_i^T W_i (W_i^T W_i + rI)^{-1} W_i^T C_i. \quad (22)$$

式中

将(16), (17)式代入上式得:

$$\begin{aligned} X_{i+1} = & C^T C + A^T X_i A - [C^T C B + A^T X_i A B][B^T C^T C B + r \\ & + B^T A^T X_i A B]^{-1} [B^T C^T C + B^T A^T X_i A]. \end{aligned} \quad (23)$$

将(21), (23)式代入(18)式得:

$$\begin{aligned} Z_{i+1} C_{i+1} = & e_{i+1} (W_{i+1}^T W_{i+1} + rI)^{-1} W_{i+1}^T C_{i+1} = [B^T C^T C B + r \\ & + B^T A^T X_i A B]^{-1} [B^T C^T C + B^T A^T X_i A], \quad i \geq N - M + 1. \end{aligned} \quad (24)$$

由(19), (24)式可以看出, 当前控制量 $u(k)$ 的求解可先通过对(20), (23)式进行递推计算 X_i , 再将求出的 X_N 代入(24)式来完成. 为了并行计算, 对(20), (23)式进行平方根分解, 可以证明存在正交矩阵 Q_1, Q_2 , 使 X_i 的平方根 $X_i^{1/2}$ (上三角阵) 的计算可由下列式子计算产生.

$$Q_1 \begin{bmatrix} C \\ X_{i+1}^{1/2} A \end{bmatrix} = X_{i+1}^{1/2}, \quad 1 \leq i \leq N - M, \quad (25)$$

$$Q_2 \begin{bmatrix} X_i^{1/2} A B & X_i^{1/2} A \\ C B & C \\ r^{1/2} & 0 \end{bmatrix} = \begin{bmatrix} E_i^{1/2} & F_i \\ 0 & X_{i+1}^{1/2} \\ 0 & 0 \end{bmatrix}, \quad i \geq N - M + 1. \quad (26)$$

$$E_i^{-1/2} F_i = Z_{i+1} C_{i+1}. \quad (27)$$

式中

$$E_i^{1/2} \in \mathbb{R}, \quad F_i \in \mathbb{R}^n.$$

实现(25), (26)式的计算可通过 $n \times n$ 的三角阵列 1 与 $n+1$ 个单元的线性阵列 2 组成(见图2). 执行(25)式时, 首先输入 $X_{i+1}^{1/2} A$ 和 C 到阵列 1 进行 QR 分解, 在阵列 1 中得到 $X_{i+1}^{1/2}$, 然后输入 A^T 与 $X_{i+1}^{1/2}$ 相乘得到的 $X_{i+1}^{1/2} A$ 再返回到输入端. $N-M$ 次(25)式的计算共需 $(N-M)(2n+1)$ 个计算步.

由阵列 1 和 2 组成的三角阵列可完成(26)式的计算. 将 $X_i^{1/2} A B, X_i^{1/2} A, C B$ 和 C 一起输入阵列, 与预先存放的 $r^{1/2}$ 进行 QR 分解, 在阵列 1 中得到 $X_{i+1}^{1/2}$, 然后输入 $(A B)^T, A^T$ 与 $X_{i+1}^{1/2}$ 相乘得到的 $X_{i+1}^{1/2} A B$ 和 $X_{i+1}^{1/2} A$ 再返回到输入端进行下次迭代计算. M 次(20)式的计算需 $M(2n+3)$ 个计算步.

最后将 $1, 0_{1 \times n}$ 输入阵列 2, 与里面的 $E_N^{1/2}, F_N$ 进行 Gauss 消去计算, 输出 $E_N^{1/2} F_N$ 即 $Z_{N+1} C_{N+1}$, 再与 $A X(k)$ 相乘得 $u(k)$.

由以上分析可算出每个当前控制点需 $N(2n+1) + 2M + 1$ 个计算步, 处理器的利用率约为 $(n+1)(nN+M)/[(n+2)(nN+M+N/2)]$.

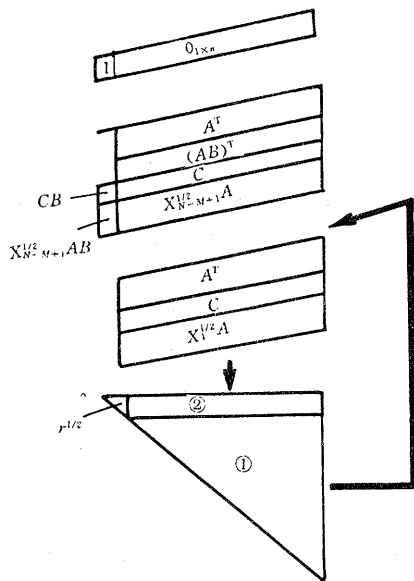


图2 并行GPC的阵列结构及数据流

4 广义预测控制的并行方法(二)

对于(1)式的输入-输出模型,若设 $c(q^{-1}) = 0$ 且目标函数为(8)式,基于状态空间模型算法^[3]的最优控制序列及过程输出可进一步表示如下:

$$U(kN) = D\{B_2 U_1[(k-1)N] + A_2 Y_1[(k-1)N]\}, \quad (28)$$

$$Y(kN) = (I - A_1)^{-1}\{B_3 U_2(kN) + A_2 Y_1[(k-1)N]\}. \quad (29)$$

式中 $Y(kN) = [y(k+1), \dots, y(k+N)]^T$,

$$Y[(k-1)N] = [y(k-N+1), \dots, y(k)]^T,$$

$$U(kN) = [u(k), \dots, u(k+N-1)]^T,$$

$$U[(k-1)N] = [u(k-N), \dots, u(k-1)]^T,$$

$$Y_1[(k-1)N] = [y(k-n+1), \dots, y(k)]^T,$$

$$U_1[(k-1)N] = [u(k-n), \dots, u(k-1)]^T,$$

$$U_2(kN) = [U_1^T[(k-1)N] \quad U^T(kN)]^T,$$

$$D = -(B_1^T L B_1 + rI)^{-1} B_1^T L, \quad L = [(I - A_1)(I - A_1^T)]^{-1},$$

$$A_1 = \begin{bmatrix} 0 & & & & & \\ -a_1 & 0 & & & & \\ \dots & \dots & & & 0 & \\ -a_n & -a_{n-1} & -a_{n-2} & & & \\ 0 & -a_n & -a_{n-1} & \dots & & \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & -a_1 & 0 \end{bmatrix}_{N \times N},$$

$$A_2 = \begin{bmatrix} -a_n & -a_{n-1} & \dots & -a_1 \\ 0 & -a_n & \dots & -a_2 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & -a_n \\ 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 \end{bmatrix}_{N \times n}, \quad B_2 = \begin{bmatrix} b_n & b_{n-1} & \dots & b_1 \\ 0 & b_n & \dots & b_2 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & b_n \\ 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 \end{bmatrix}_{N \times n},$$

$$B_1 = \begin{bmatrix} b_0 & & & & & \\ b_1 & b_0 & & & & \\ \dots & \dots & & & 0 & \\ b_n & b_{n-1} & b_{n-2} & & & \\ 0 & b_n & b_{n-1} & \dots & & \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & b_1 & b_0 \end{bmatrix}_{N \times N},$$

$$B_3 = [B_2 \quad B_1] = \begin{bmatrix} b_n & b_{n-1} & \dots & b_1 & b_0 & \dots & 0 & 0 \\ 0 & b_n & \dots & b_2 & b_1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & b_n & b_{n-1} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & \dots & b_1 & b_0 \end{bmatrix}_{N \times (N+n)}$$

实现上述算法的 Systolic 结构简图见图 3.

计算 $I = A_2 Y_1[(k-1)N]$ 可通过图 4(a) 所示的一个三角阵列结构来完成, 并将计算结果与 $U_1[(k-1)N]$ 一起输入形如图 4(a) 且初始值为 B_2 元素的三角阵列进行乘加, 完成 $O = A_2 Y_1[(k-1)N] + B_2 U_1[(k-1)N]$ 的计算. 由于对(28)式中 D 阵的计算需反复地进行矩阵求逆及乘加运算, 通过下列变换可使算法利于并行计算.

$$Q \begin{bmatrix} [I - A_1]^{-1} B_1 & [I - A_1]^{-1} O \\ r^{1/2} I_{N \times N} & 0_{N \times 1} \end{bmatrix} = \begin{bmatrix} F_{N \times N} & E_{N \times 1} \\ 0 & * \end{bmatrix}. \quad (30)$$

式中, Q 是使(30)式初始矩阵三角化的正交阵, F, E 分别是变换后得到的上三角阵与向量, 且 F 可逆, $*$ 为无关紧要项. 可以证明: $D = -F^{-1}E$.

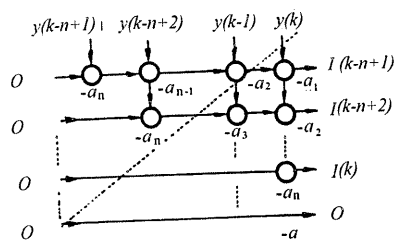


图 4(a) 计算 $I = A_2 Y_1[(k-1)N]$ 的阵列结构

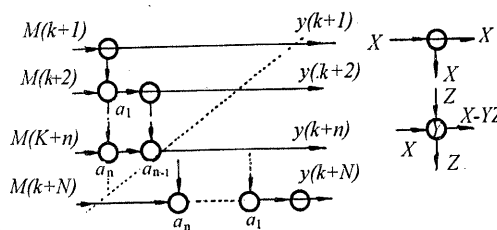


图 4(b) 计算 V, W

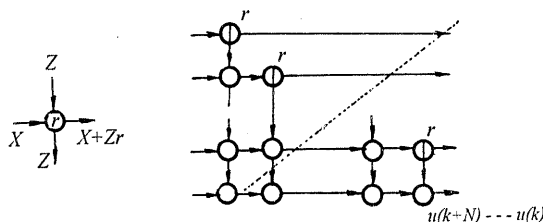


图 4(c) 计算(30)式的阵列结构

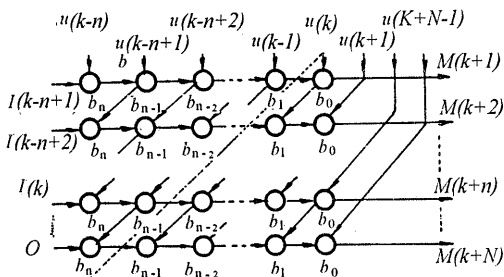


图 4(d) 实现 $y_t(kN)$ 计算的阵列结构

令 $V = [I - A_1]^{-1} B_1, W = [I - A_1]^{-1} O$. 阵列实现时, 将 B_1^T, O^T 输入初始值为 $[I - A_1]$ 的阵列(图 4(b)) 即可得到 V, W . 计算(30)式时, 需将 r 预先存在图 4(c) 所示阵列的对角单元内, 其他单元初始化为零, 对输入的 V, W 进行 QR 分解, 在阵列中得到 F, E . 输入 $-I_{N \times N}, 0_{N \times 1}$ 向量进行 Gauss 消去运算, 将产生 $U(kN)$ 并输出. 最先输出的是当前的控制量 $u(k)$.

计算 $Y(kN)$ 时, 将 I 向量与 $[U_1^T[(k-1)N], U^T(kN)]^T$ 一起输入图 4(d) 所示的阵列进行乘加运算, 得到的 $M = B_3 U_2(kN) + A_2 Y_1[(k-1)N]$ 向量再输入图 4(b) 所示的阵列, $(I - A_1)^{-1} M$ 即 $Y(kN)$ 将输出. $u(k), y(k+1)$ 与过去的控制及输出量组成新的 $U[(k-1)N], Y[(k-1)N]$, 进行下一循环的辨识与控制计算.

通过分析上述辨识和预测控制各阶段的数据流时序可知, 在有效地组织下, 每次控制计算

的周期为 $4N + n + 5$ 个单位时间.

5 结 论

本文给出的算法及阵列结构是一种有效的实时辨识的方法,将 GPC 问题化为求解离散形式的 Riccati 方程,可通过阵列结构并行实现,此算法减少了所需阵列结构的规模;通过对原来串行预测控制算法的重新设计,开发内在的并行性可通过一组流水线阵列结构加以实现;两种并行方法都可使计算的实时性提高很大.

参 考 文 献

- [1] Yang, B. . Rotation-Based RLS Algorithm; Unified Derivations Numerical Properties and Parallel Implementations. IEEE Trans SP, 1992, 40(5): 1151—1166
- [2] Moon, M. . A Systolic array for recursive least squares computations. IEEE Trans SP, 1993, 41(2): 906—911
- [3] PEDRO, A. . On Generalized Predictive Control; Two Alternative Formulations. Automatica, 1989, 25(5): 753—755
- [4] Wook, H. K. . Recursive Solution of Generalized Predictive Control and Its Equivalence to Receding Horizon Tracking Control. Automatica, 1992, 28(6): 1235—1238
- [5] Furuta, K. . Dynamic Compensator Design for Discrete-Time LQG Problem Using Markov Parameters. Proceedings of 32nd Conf on Decision and Control, 1993, 96—101
- [6] 慕德俊, 戴冠中. 脉动阵列执行 LQ 优化. 控制理论及应用年会论文集. 南京, 1992
- [7] 李清泉. 自适应控制系统理论设计与应用. 北京: 科学出版社, 1990

Parallel Algorithms for the Generalized Predictive Control Based on State-Space Model

MU Dejun and DAI Guanzhong

(Department of Automatic control, Northwestern Polytechnical University • Xi'an, 710072, PRC)

Abstract: In this paper, an efficient algorithm and Systolic architecture are proposed for parameter estimation. Two algorithms and Systolic architectures for the generalized predictive control (GPC) based state-space model are developed. A new recursive solution of GPC is developed, in which the control input can be obtained recursively from discrete Riccati equation. The recursive algorithm can be implemented by a small size of array. Another algorithm is mapped to a set of two dimensions arrays for computing predictive control and output. The analysis shows that the methods have high degree of parallelism.

Key words: parameter estimation; generalized predictive control; systolic array; parallel algorithm.

本文作者简介

慕德俊 见本刊 1995 年第 1 期第 127 页.

戴冠中 见本刊 1995 年第 1 期第 127 页.