

## 树网结构上一种新的矩阵迭代求逆并行算法\*

莫则尧 李晓梅

(国防科技大学电子计算机系 长沙 410073)

**摘要** 运用树网结构可以完成矩阵的并行快速求逆, 其中迭代法是一种非常重要的方法。本文给出了一种新的迭代格式, 对任意非奇异矩阵  $A$ , 运用新的迭代格式对  $A$  求逆相对于经典牛顿迭代法, 在同样精度要求下, 时间可减少一半。

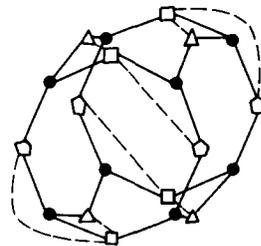
**关键词** 树网, 矩阵求逆, 并行算法

**分类号** O151.21

本文主要讨论  $N \times N \times N$  三维树网结构上对任意  $N \times N$  非奇异矩阵的快速并行求逆算法。一般地, 树网快速求逆法可分为直接法(如 Csanky 算法<sup>[1]</sup>)与迭代法(如经典牛顿迭代法<sup>[2][3]</sup>)。尽管二者时间复杂度均为  $O(\text{Log}^2 N)$ , 但迭代法比直接法优越<sup>[1]</sup>。本文只对迭代法进行讨论。

我们简单回顾一下三维树网结构, 同时作稍微改变以便于以后分析。

$N \times N \times N$  树网是在  $N^3$  个孤立处理机结点上增添一系列的处理机与数据连线构成。这  $N^3$  个基本处理机称为叶子结点(Leaf nodes), 标号分别为  $\{(i, j, k) | 1 \leq i, j, k \leq N\}$ 。对每个  $(j, k)$  ( $1 \leq j, k \leq N$ ), 以  $\{(i, j, k) | 1 \leq i \leq N\}$  为叶子加入另外  $N-1$  个结点和  $2N-2$  条边形成完全二叉树(complete binary tree), 称为一维  $(j, k)$  树; 其根结点记为  $(\tilde{1}, j, k)$ 。类似地, 以  $\{(i, j, k) | 1 \leq j \leq N\}$  为叶子形成  $N^2$  个二维  $(i, k)$  树 ( $1 \leq i, k \leq N$ ), 相应根结点记为  $(i, \tilde{2}, k)$ 。以  $\{(i, j, k) | 1 \leq k \leq N\}$  为叶子构成  $N^2$  个三维  $(i, j)$  树, 根结点记为  $(i, \tilde{3}, k)$ 。称增添的结点为内结点(internal nodes)。这个网络直径为  $6\text{Log} N$ , 折半宽度为  $N^2$ <sup>[1]</sup>。我们知道, 这个网络完成两个  $N \times N$  矩阵相乘只须  $2\text{Log} N + 1$  步<sup>[1]</sup>。为了便于分析, 我们将网络的  $3N^2$  个根结点以一定方式联接起来。具体地将根结点  $(i, \tilde{2}, k)$  分别与根结点  $(\tilde{1}, i, k)$  和根结点  $(i, k, \tilde{3})$  相连, 以便数据传送。我们记这个网络为  $G$ , 以下讨论均针对  $G$  进行。

图1  $2 \times 2 \times 2$  树网

●: 叶子结点, ▲: 一维树内结点, □: 二维树内结点, ⬡: 三维树内结点, ⋯: 另外加入的连接树根结点的线。

\* 1993年7月5日收稿

图 1 给出了一个  $2 \times 2 \times 2$  网络。

## 1 迭代格式

令  $A$  为任意  $N \times N$  非奇异矩阵,  $X_t$  为  $A^{-1}$  第  $t$  次迭代后近似值, 由下面的格式计算  $X_{t+1}$ :

$$X_{t+1} = 2X_t - X_t A X_t \quad (I)$$

这就是典型的矩阵求逆牛顿迭代法。

令剩余矩阵  $R_t = I - X_t A$ , 有:

$$\begin{aligned} R_{t+1} &= I - X_{t+1} A \\ &= I - (2X_t - X_t A X_t) A \\ &= R_t^2 \end{aligned}$$

易知:

$$R_t = R_0^{2^t}, R_0 \text{ 为初始剩余矩阵。}$$

显然, 若  $X_0$  选择得好,  $R_t$  将迅速收敛于零。特别地, 若令  $X_0 = \frac{1}{m} A^{T[1]}$ , 其中  $m = \text{trace}(A^T A)$ 。则有<sup>[1]</sup>:

$$\|R_0\|_2 = \|I - X_0 A\|_2 \leq 1 - \frac{1}{NK^2} \quad (1)$$

其中  $K = \|A^T\|_2 \cdot \|A^{T^{-1}}\|_2$ 。

一般地, 设  $\|R_0\|_2 \leq 1 - N^{-\alpha}$ , 其中  $\alpha$  依赖于矩阵  $A^T$  的条件数  $K$ , 并满足不等式:

$$1 - \frac{1}{NK^2} \leq 1 - N^{-\alpha}$$

即

$$N^{\alpha-1} \geq K^2 \quad (2)$$

的最小常数。由此可以看出,  $N$  较大时,  $\alpha$  将接近于 1。经过  $T \triangleq (\alpha + \beta) \text{Log} N$  次迭代后, 有

$$\begin{aligned} \|R_T\|_2 &\leq (\|R_0\|_2)^{2^T} \\ &\leq (1 - N^{-\alpha})^{N^{\alpha} 2^T} \\ &\leq e^{-N^{\beta}} \end{aligned}$$

即

$$\|X_T - A^{-1}\|_2 \leq e^{-N^{\beta}} \cdot \|A^{-1}\|_2 \quad (3)$$

其中,  $\beta$  为满足不等式

$$e^{-N^{\beta}} \cdot \|A^{-1}\|_2 \leq 2^{-(N+1)} \quad (4)$$

的最小正常数。由式(3)、(4)我们有:

$$\|X_T - A^{-1}\|_2 \leq 2^{-(N+1)}$$

故经过  $T = (\alpha + \beta) \text{Log} N$  步迭代后,  $A^{-1}$  的每一个元素的前  $N$  位被计算出来 (在范数  $\|\cdot\|_2$  意义下)。由(2)、(4)知, 对一般矩阵,  $N$  较大时,  $\alpha$ 、 $\beta$  均非常接近 1。

以上介绍了经典牛顿迭代法及其收敛情况, 下面讨论新的迭代格式。

新的迭代格式基于下面的引理。

**引理 1** 设  $A$  为  $N \times N$  非奇异矩阵,  $\|A\|_2 < 1$ , 则

$$\lim_{M \rightarrow +\infty} \prod_{i=M}^0 (I - A^{2^i}) = (I - A)^{-1}$$

证明略。且剩余矩阵  $\|R_M\|_2 \leq \|A\|_2^{2^{M-1}}$ 。

由引理 1 可以看出, 若  $\|A\|_2 < 1$ , 则  $R_M = A^{2^{M+1}}$  收敛于零的速度也将很快, 与牛顿迭代法一样, 为指数收敛。由引理 1, 我们可得到下面的引理。

**引理 2** 对于任意非奇异矩阵  $A$ , 若  $A$  满足  $\|I - A\|_2 < 1$ , 则存在迭代格式:

$$\begin{cases} A_{t+1} = (I + (I - A)^{2^t})A_t \\ A_0 = I \end{cases} \quad (\text{I})$$

其中  $A_t$  为第  $t$  次迭代后  $A^{-1}$  的近似值。使得:

$$\lim_{t \rightarrow +\infty} A_t = A^{-1}.$$

**证明** 令矩阵  $B = I - A$ , 则  $I - B = I - (I - A) = A$ 。

故

$$A^{-1} = (I - B)^{-1}$$

对矩阵  $B$  设计新的迭代格式如下。令  $B_t$  为  $t$  次迭代后  $(I - B)^{-1}$  的近似值, 由下面公式计算  $B_{t+1}$ :

$$\begin{cases} B_{t+1} = (I + B^{2^t})B_t \\ B_0 = I \end{cases} \quad (\text{II})$$

考虑到  $\|B\|_2 = \|I - A\|_2 < 1$ , 由引理 1 知:

$$\begin{aligned} \lim_{t \rightarrow +\infty} B_t &= (I - B)^{-1} \\ &= A^{-1} \end{aligned}$$

将  $B = I - A$ ,  $A_t \triangleq B_t$  代入迭代公式 (II) 中得到公式 (I), 且  $\lim_{t \rightarrow \infty} A_t = A^{-1}$ 。

将引理 2 推广到任意  $N \times N$  非奇异矩阵的情况, 我们有下面的重要定理。

**定理** 对任意  $N \times N$  非奇异矩阵  $A$ , 令矩阵  $B = X_0 A$  (其中  $X_0$  为牛顿迭代法的初始近似值), 它使得  $\|I - X_0 A\|_2 < 1$ , 则  $B$  可运用新的迭代格式 (I) 求逆。特别地, 若取  $X_0$

$= \frac{1}{m} A^T$ , 则迭代  $T = (\alpha + \beta) \text{Log} N$  次后, 再经过一个矩阵乘法  $B^{-1} X_0$ , 我们便能将  $A^{-1}$  的每个元素前  $N$  位计算出来 (在  $\|\cdot\|_2$  意义下)。其中  $\alpha, \beta, m$  的定义同前。

**证明** 因为  $\|I - X_0 A\|_2 < 1$ ,  $B = X_0 A$ , 故  $\|I - B\|_2 < 1$ 。

由引理 2 知, 矩阵  $B$  可运用迭代公式 (I) 求逆。

又  $X_0 A = B$

故  $B^{-1} = A^{-1} X_0^{-1}$ ;  $A^{-1} = B^{-1} X_0$

特别地, 若  $X_0 = \frac{1}{m} A^T$ , 则有

$$\|I - \frac{1}{m} A^T A\|_2 \leq 1 - \frac{1}{NK^2}, \quad K = \|A^T\|_2 \cdot \|(A^{-1})^T\|_2$$

即

$$\|I - B\|_2 \leq 1 - \frac{1}{NK^2} \leq 1 - N^{-\alpha}$$

经过  $T = (\alpha + \beta)\text{Log}N$  次迭代后, 有:

$$I - B_T B = (I - B)^{N^{\alpha+\beta}}$$

其中  $B_T$  为  $B^{-1}$  的  $T$  步迭代后近似值。

将  $B = \frac{1}{m}A^T A$  代入有:

$$I - \frac{1}{m}B_T A^T A = \left(I - \frac{1}{m}A^T A\right)^{N^{\alpha+\beta}}$$

故

$$A^{-1} - \frac{1}{m}B_T A^T = \left(I - \frac{1}{m}A^T A\right)^{N^{\alpha+\beta}} \cdot A^{-1}$$

故

$$\begin{aligned} \|A^{-1} - \frac{1}{m}B_T A^T\|_2 &\leq \|I - \frac{1}{m}A^T A\|_2^{N^{\alpha+\beta}} \cdot \|A^{-1}\|_2 \\ &\leq (1 - N^{-\alpha})^{N^{\alpha+\beta}} \cdot \|A^{-1}\|_2 \\ &\leq e^{-N^\beta} \cdot \|A^{-1}\|_2 \\ &\leq 2^{-(N+1)} \end{aligned}$$

故经过  $T = (\alpha + \beta)\text{Log}N$  次迭代后, 再经过一个矩阵乘法  $\frac{1}{m}B_T \cdot A^T$ ,  $A^{-1}$  的各个元素前  $N$  位便被计算出来 (在  $\|\cdot\|_2$  意义下)。

从定理可以看出, 对任意  $N \times N$  非奇异矩阵, 新的迭代求逆格式 (II) 与牛顿迭代格式 (I) 在同一精度要求下无论在收敛速度, 还是迭代次数方面均是相等的。虽然, 我们只对特殊情形  $X_0 + \frac{1}{m}A^T$  进行了讨论, 不难发现: 对任意其它  $X_0$ , 只要其满足  $\|I - X_0 A\|_2 < 1$  的条件, 两种迭代格式的这个性质始终保持。另外,  $X_0$  的选择对算法的影响是存在的, 尤其当矩阵很特殊时<sup>[2]</sup>。 $X_0$  的影响对二种迭代格式是等同的。基于以上事实, 我们在下一部分中将发现, 在三维树网结构上具体实现时, 新的迭代格式每次迭代时间缩短一半, 意味着新的并行算法在同一精度要求下, 比牛顿迭代法快了一倍。

## 2 迭代格式分析

对于牛顿迭代法 (I), 每次迭代需两个矩阵乘法, 一个矩阵加法, 由于其数据相关性, 两个矩阵乘法与矩阵加法只能顺序执行, 运行时间约为  $4\text{Log}N$  步。对于新的迭代格式, 虽然每次迭代也需二个矩阵乘法 (这是我们将每次迭代所得的  $(I - A)^{2^i}$  存贮起来以备下一次迭代计算  $(I - A)^{2^{i+1}}$ , 一个矩阵加法, 但是仔细观察, 可以将两个乘法在时间上重叠执行, 加法可一步完成, 运行时间约为  $2\text{Log}N$  步, 缩短一半。

具体地, 不妨假设  $\|I - A\|_2 < 1$ ,  $I - A \triangleq (a_{ij})_{N \times N}$ . 令  $A_0 = I$ . 第一步, 将  $(I - A)$  分别输入一维与三维根结点中, 即将  $a_{ij}$  送到根结点  $(\tilde{1}, i, j)$  与  $(i, j, \tilde{3})$  中。第二步, 将它

们沿着树往下送,但  $I-A$  还必须保留在三维树根结点中(要求三维树根结点具有复制数据功能),同时将三维树根结点保留的矩阵  $I-A$  对角元素加 1,并将  $A_0$  输入到一维树根结点中。第三步,  $A_0$  与  $I+(I-A)$  沿着树紧跟  $(I-A)$  往下送。第  $\text{Log}N+1$  步,叶子结点  $(i, j, k)$  接收到数据  $a_{ij}$  与  $a_{jk}$ , 将其相乘并在下一步将乘积沿二维树累加,以求  $(I-A)^2$ 。第  $\text{Log}N+2$  步,叶子结点  $(i, j, k)$  接收到  $I+(I-A)$  的元素  $(i, j)$  及  $A_0$  元素  $(j, k)$ , 执行同样操作,在下一步沿二维树累加,以求  $A_1 = (I+(I-A)) \cdot A_0 = I+(I-A)$ 。第  $2\text{Log}N+1$  步,  $(I-A)^2$  的各个元素出现在二维树的各个相应根结点中,并于第  $2\text{Log}N+2$  步将各个元素播送到一、三维相连的树根结点中。再下一步,  $(I-A)^2$  沿一、三维树往下送,以计算  $(I-A)^4$ 。同时  $(I-A)^2$  保留在三维树根结点中,并变为  $I+(I-A)^2$ 。同时,第  $2\text{Log}N+2$  步,  $A_1$  出现在二维树根结点中,至此第一次迭代结束,只须  $2\text{Log}N+2$  步。为了下一次迭代,第  $2\text{Log}N+3$  步,三维树根结点将  $A_1$  送给一维树根结点。下一步与  $I+(I-A)^2$  一起沿各自树往下送以计算  $A_2 = (I+(I-A)^2) \cdot (I+(I-A))$ , 第二次迭代开始。类似进行下去,  $(\alpha+\beta)\text{Log}N$  次迭代后,出现在二维树根结点中的矩阵即为  $A^{-1}$  所需近似值,它的每个元素与  $A^{-1}$  对应元素前  $N$  位相同。

由以上实现知,新的迭代格式每步约为  $2\text{Log}N$  步,为牛顿迭代法每次迭代的一半,且所需的硬件除了要求三维树根结点具有复制数据功能外,其余均与牛顿迭代法一致。由于在同等精度条件下,新的算法所需迭代次数与牛顿迭代法一致,故在同等精度要求下,新算法在时间上缩短一半。对于  $A$  为任意矩阵情形,由定理知,算法最后只需增加一个矩阵乘法,故可以忽略。故上面结论对任意非奇异矩阵均适应。

### 3 结 论

针对某种应用网络、处理机阵列,我们应当大量研制适应于网络、阵列的高效并行算法,使得每个处理机尽量并行工作,充分发挥效率。三维树网结构上矩阵求逆,迭代法比直接法较优<sup>[1]</sup>。由以上分析,新的迭代格式(II)的并行度比经典牛顿迭代格式(I)要高,能更大地发挥树网求逆的效率,在同样的精度要求下,新算法比牛顿迭代法快了一倍,是一个较优算法。

不难发现,上述迭代法的时间复杂度为  $O(\text{Log}M \text{Log}N)$ ,其中  $M$  为要求的精确位数。特别地,  $M=N$  时,时间复杂度为  $O(\text{Log}^2N)$ 。通过同时求解多个矩阵的逆,我们还可以提高树网效率。但迄今为止,我们还没有发现一个能在  $O(\text{Log}N)$  步内对矩阵求逆的算法<sup>[1]</sup>。这仍然是一个令人关心的问题。

### 参 考 文 献

- 1 Thomson F, Lighton. Introduction to Parallel Algorithms and Architectures; Arrays, Trees, Hypercube. Morgan Kaufmann Publishers, San Mateo, California, 1992
- 2 Isaacson E, Keller H. Analysis of Numerical Methods. Wiley, New York, 1991
- 3 Pan V, Reif J. Efficient parallel solution of linear systems. In: proleedings of the seventeenth Annual ACM symposium on Theory of Computing, pages 143-152, May 1985

- 4 Galil Z, Pan V. Parallel evaluation of the determinant and of the inverse of a matrix. *Information Processing Letters*, 1989, 30(1): 41~45
- 5 李晓梅, 蒋增荣. 并行算法. 长沙: 湖南科技出版社, 1992

## **A New Parallel Algorithm for Matrix Inversion on Mesh**

Mo Zeyao Li Xiaomei

(Department of Computer Science, NUDT, changsha, 410073)

### **Abstract**

Many parallel algorithms can be used to invert a non-singular matrix rapidly on mesh. Among them, the iterative method is very important. This paper presents a new parallel iterative algorithm for arbitrary non-singular matrix inversion on mesh. Meeting the requirement of the same precision, new algorithm can reduce the running time for inversion by half in contrast to the typical Newton iterative algorithm on mesh.

**Key words** mesh, matrix inversion, parallel algorithm