

求解三对角线性方程组的迭代对角占优算法

李太全^{1*}, 肖柏勋²

(1. 长江大学 物理科学与技术学院, 湖北 荆州 434002; 2. 长江大学 地球物理与石油资源学院, 湖北 荆州 434002)

(* 通信作者电子邮箱 tqli@yangtzeu.edu.cn)

摘要:针对并行求解三对角线性方程组的对角占优(PDD)算法,在系数矩阵为弱对角占优时,近似处理引入误差较大的问题,提出了一种PDD算法的迭代方案。该方案在解的修正值计算中采用迭代方法,计算精度得到了提高;通过对算法的误差分析,导出了算法在给定误差下迭代次数的估算式;数值实验说明了算法的有效性。通过对迭代与非迭代的PDD算法的复杂性分析,迭代算法的计算复杂性增加很小,但通信复杂性随迭代次数成倍增加。

关键词:对角占优算法;迭代;三对角线性方程组;分布式存储;并行计算

中图分类号: TP301.6 **文献标志码:** A

Iterated parallel diagonal dominant algorithm for tridiagonal systems

LI Tai-quan^{1*}, XIAO Bo-xun²

(1. School of Physical Science and Technology, Yangtze University, Jingzhou Hubei 434002, China;

2. School of Geophysics and Oil Resource, Yangtze University, Jingzhou Hubei 434002, China)

Abstract: In parallel solving weak diagonal dominant tridiagonal systems, the approximate error of the Parallel Diagonal Dominant (PDD) algorithm cannot be ignored. An iterated PDD algorithm was presented. In the algorithm, the solution of the correction value was calculated by iterative method, and the computational accuracy was obviously improved. Through error analysis on the algorithm, an estimation formula of iteration number was derived for a given error tolerance. And the numerical experiment shows the validity. Based on the complexity analysis of the iterative and non-iterative PDD algorithm, the increase of iterative algorithm computational complexity is very small, but the communication complexity increases exponentially with the iteration number.

Key words: Parallel Diagonal Dominant (PDD) algorithm; iteration; tridiagonal systems; distributed memory; parallel computing

0 引言

在偏微分方程的数值求解中,具有绝对稳定特性的偏微分方程隐式差分格式需要求解三对角线性方程组。所以,三对角线性方程组的并行求解成为数值并行算法研究中的重要课题。关于三对角方程组的并行求解算法,主要有递推耦合算法^[1-2]、循环约化法^[3-4]、分治法^[5-7]等。针对对角占优的三对角方程组, Sun 等^[8-10]基于分治思想,采用和解方法,提出了对角占优(Parallel Diagonal Dominant, PDD)算法,迟利华等^[11]基于乘分解提出了PPD算法。这些算法均为近似求解算法,在求解强对角占优的三对角方程组时具有良好的计算精度和并行效率。之后,张衡等基于Climent等^[12]提出的重叠并行方法,提出了块重叠分割无通信的高效可扩展并行算法^[13]。

求解对角占优三对角系统的PDD并行算法,将 $N \times N$ 阶的三对角矩阵依照参与计算的进程数 N_p 分解为 N_p 个 $m \times m$ 的子对角矩阵($N = N_p \times m$),每个进程独立求解对应的子对角系统粗略解,并通过相邻进程间的数据交换,计算解的修正值,最后获得系统的近似解。该算法因为其很低的通信复杂性和较低的计算复杂性而被广泛采用。但算法对系数矩阵的对角占优特性要求较苛刻,对于弱对角占优三对角系统,解的误差随着 m 的减小而显著增大^[9],这一特性会限制计算节点

的充分利用。本文在PDD算法的基础上,采用迭代方法计算解的修正值,提高了修正值的精度,改善了PDD算法的性能。

1 PDD 算法

对于由 $N \times N$ 阶实三对角矩阵:

$$A = \begin{bmatrix} b_0 & c_0 & 0 & \cdots & 0 \\ a_1 & b_1 & c_1 & \cdots & 0 \\ 0 & a_2 & b_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & & c_{N-2} \\ 0 & 0 & 0 & a_{N-1} & b_{N-1} \end{bmatrix} \quad (1)$$

构成的三对角线性方程组

$$Ax = d \quad (2)$$

的并行求解,可借助矩阵 A 的分解 $A = \tilde{A} + VE^T$ 实现。其中 $\tilde{A} = \text{diag}[A^{(0)}, \dots, A^{(p)}, \dots, A^{(N_p-1)}]$ 为块对角矩阵,设子块 $A^{(p)}$ 均为 m 阶三对角矩阵,则 $N_p = N/m$,上标 (p) 为块编号;且:

$$V = [a_m e_m, c_{m-1} e_{m-1}, \dots, a_{m(N_p-1)} e_{m(N_p-1)}, c_{m(N_p-1)-1} e_{m(N_p-1)-1}] \quad (3)$$

$$E = [e_{m-1}, e_m, \dots, e_{m(N_p-1)-1}, e_{m(N_p-1)}] \quad (4)$$

其中 e_i 为 N 个元素的列向量,且除第 i 元素为1外,其他元素

收稿日期:2012-04-18;修回日期:2012-05-28。

基金项目:国家自然科学基金资助项目(41140034);湖北省教育厅重点资助项目(20081202)。

作者简介:李太全(1961-),男,湖北松滋人,副教授,博士,主要研究方向:计算电磁学;肖柏勋(1956-),男,湖北新洲人,教授,博士,主要研究方向:地质雷达。

均为0。式(2)的解可以写成^[10]:

$$\mathbf{x} = \tilde{\mathbf{A}}^{-1} \mathbf{d} - \tilde{\mathbf{A}}^{-1} \mathbf{V}(\mathbf{I} + \mathbf{E}^T \tilde{\mathbf{A}}^{-1} \mathbf{V})^{-1} \mathbf{E}^T \tilde{\mathbf{A}}^{-1} \mathbf{d} \quad (5)$$

其中 \mathbf{I} 是单位矩阵,令 $\mathbf{Z} = \mathbf{I} + \mathbf{E}^T \tilde{\mathbf{A}}^{-1} \mathbf{V}$, \mathbf{Z} 为 $2N_p - 1$ 阶的五对角矩阵,可以表示为:

$$\mathbf{Z} = \begin{bmatrix} 1 & w_{m-1}^{(0)} & 0 & & \\ v_0^{(1)} & 1 & 0 & w_0^{(1)} & \\ v_{m-1}^{(1)} & 0 & 1 & w_{m-1}^{(1)} & \\ & & \vdots & & w_0^{(N_p-2)} \\ & & & 0 & 1 & w_{m-1}^{(N_p-2)} \\ & & & & v_0^{(N_p-1)} & 1 \end{bmatrix} \quad (6)$$

其中 $\mathbf{v}^{(p)}, \mathbf{w}^{(p)} (p = 0, 1, \dots, N_p - 1)$ 是方程

$$\begin{cases} \mathbf{A}^{(p)} \mathbf{v}^{(p)} = \mathbf{a}_0^{(p)} \mathbf{e}_0 \\ \mathbf{A}^{(p)} \mathbf{w}^{(p)} = \mathbf{c}_{m-1}^{(p)} \mathbf{e}_{m-1} \end{cases} \quad (7)$$

的解。在矩阵 \mathbf{A} 为对角占优,即 $|b_k| > |a_k| + |c_k|$ 的情况下,对于 $\mathbf{v}^{(p)} = [v_0^{(p)}, v_1^{(p)}, \dots, v_{m-1}^{(p)}]^T$,有 $|v_i^{(p)}| < 1$ 、 $|v_i^{(p)}| > |v_{i+1}^{(p)}|$,而 $\mathbf{w}^{(p)} = [w_0^{(p)}, w_1^{(p)}, \dots, w_{m-1}^{(p)}]^T$,有 $|w_i^{(p)}| < 1$ 、 $|w_i^{(p)}| < |w_{i+1}^{(p)}|$ 。将 \mathbf{Z} 写成 $\mathbf{Z} = \mathbf{Z}_0 + \Delta\mathbf{Z}$,其中 \mathbf{Z}_0 为仅含有 $v_0^{(p)}, w_{m-1}^{(p)}$ 的三对角矩阵,即:

$$\mathbf{Z}_0 = \begin{bmatrix} 1 & w_{m-1}^{(0)} & 0 & & \\ v_0^{(1)} & 1 & 0 & & \\ & 0 & 1 & w_{m-1}^{(1)} & \\ & & \vdots & & \\ & & & 0 & 1 & w_{m-1}^{(N_p-2)} \\ & & & & v_0^{(N_p-1)} & 1 \end{bmatrix} \quad (8)$$

$\Delta\mathbf{Z}$ 为仅含有 $v_{m-1}^{(p)}, w_0^{(p)}$ 的剩余部分。且如果 m 足够大,则 $v_{m-1}^{(p)}, w_0^{(p)}$ 可以忽略,矩阵 \mathbf{Z} 能近似表示 \mathbf{Z}_0 ,则方程(2)的近似解便可采用如下步骤求得:

1) 令 $\tilde{\mathbf{x}} = \tilde{\mathbf{A}}^{-1} \mathbf{d}, \mathbf{Y} = \tilde{\mathbf{A}}^{-1} \mathbf{V}$,求解 $\tilde{\mathbf{A}} \tilde{\mathbf{x}} = \mathbf{d}, \tilde{\mathbf{A}} \mathbf{Y} = \mathbf{V}$ 。这里, $\tilde{\mathbf{A}} \mathbf{Y} = \mathbf{V}$ 为式(7)的合并形式。

2) 令 $\mathbf{h} = \mathbf{E}^T \tilde{\mathbf{x}}, \mathbf{y} = \mathbf{Z}^{-1} \mathbf{h}$,取 $\mathbf{Z} = \mathbf{I} + \mathbf{E}^T \mathbf{Y} \approx \mathbf{Z}_0$,求解 $\mathbf{Z}_0 \mathbf{y} = \mathbf{h}$ 。

3) 计算方程解的修正值 $\Delta\mathbf{x} = \mathbf{Y} \mathbf{y}$,从而得到方程的近似解 $\mathbf{x} = \tilde{\mathbf{x}} - \Delta\mathbf{x}$ 。

在上述的求解过程中,由于 $\tilde{\mathbf{A}}, \mathbf{Z}_0$ 均为块对角矩阵,第1)步可直接并行处理,第2)、3)步需要在相互通信的条件下并行计算。

2 迭代 PDD 算法

在上述求解的步骤2)中,由于矩阵 \mathbf{Z} 的近似处理,导致算法存在一定的误差。在矩阵 \mathbf{A} 强对角占优、且子块矩阵 $\mathbf{A}^{(p)}$ 的阶 m 较大时,这一误差可以忽略。但如果矩阵非 \mathbf{A} 强对角占优,则需要 m 足够大,才能保证误差在容许的范围内。过大的 m 可能会限制计算节点的充分利用,为此,将迭代方法引入到PDD算法中,提高PDD算法的精度,以降低对 m 足够大的要求。

考虑 $\mathbf{Z} = \mathbf{Z}_0 + \Delta\mathbf{Z} = \mathbf{Z}_0(\mathbf{I} + \mathbf{Z}_0^{-1} \Delta\mathbf{Z})$,在对角占优的情况下, $\|\mathbf{Z}_0^{-1} \Delta\mathbf{Z}\|_\infty < 1$ ^[9],所以 $(\mathbf{Z}_0 + \Delta\mathbf{Z}) \mathbf{y} = \mathbf{h}$ 的解为:

$$\mathbf{y} = \left(\mathbf{I} + \sum_{i=1}^{\infty} (-1)^i (\mathbf{Z}_0^{-1} \Delta\mathbf{Z})^i \right) \mathbf{Z}_0^{-1} \mathbf{h} \quad (9)$$

式(9)的计算可分解为:

$$\begin{cases} \mathbf{Z}_0 \mathbf{y}^{(0)} = \mathbf{h} \\ \mathbf{Z}_0 \mathbf{y}^{(1)} = -\Delta\mathbf{Z} \mathbf{y}^{(0)} \\ \mathbf{Z}_0 \mathbf{y}^{(2)} = \Delta\mathbf{Z} \mathbf{y}^{(1)} \\ \dots \end{cases} \quad (10)$$

迭代求解(这里上标 (i) 表示迭代次数),得到 \mathbf{y} 的 n 次迭代近似值:

$$\mathbf{y}_n = \sum_{i=0}^n \mathbf{y}^{(i)} \quad (11)$$

取适当的迭代次数 n ,可使式(11)达到要求的精度。由此得到迭代的PDD算法的计算步骤:

1) 求解 $\tilde{\mathbf{A}} \tilde{\mathbf{x}} = \mathbf{d}, \tilde{\mathbf{A}} \mathbf{Y} = \mathbf{V}$;

2) 令 $\mathbf{h} = \mathbf{E}^T \tilde{\mathbf{x}}$,而 $\mathbf{Z} = \mathbf{I} + \mathbf{E}^T \mathbf{Y} = \mathbf{Z}_0 + \Delta\mathbf{Z}$,求解:

$$\begin{cases} \mathbf{Z}_0 \mathbf{y}^{(0)} = \mathbf{h} \\ \mathbf{Z}_0 \mathbf{y}^{(1)} = -\Delta\mathbf{Z} \mathbf{y}^{(0)} \\ \mathbf{Z}_0 \mathbf{y}^{(2)} = \Delta\mathbf{Z} \mathbf{y}^{(1)} \\ \dots \\ \mathbf{Z}_0 \mathbf{y}^{(n)} = \Delta\mathbf{Z} \mathbf{y}^{(n-1)} \end{cases}$$

3) 计算方程解的修正值 $\Delta\mathbf{x} = \mathbf{Y}(\mathbf{y}^{(0)} + \mathbf{y}^{(1)} + \dots + \mathbf{y}^{(n)})$,

从而得到方程的解 $\mathbf{x} = \tilde{\mathbf{x}} - \Delta\mathbf{x}$ 。

3 算法的误差分析与迭代次数的确定

将式(9)表示为:

$$\mathbf{y} = \left(\sum_{i=0}^n (-1)^i (\mathbf{Z}_0^{-1} \Delta\mathbf{Z})^i + \sum_{i=n+1}^{\infty} (-1)^i (\mathbf{Z}_0^{-1} \Delta\mathbf{Z})^i \right) \mathbf{Z}_0^{-1} \mathbf{h} = \tilde{\mathbf{y}} + \delta\mathbf{y} \quad (12)$$

其中 $\tilde{\mathbf{y}} = \left(\sum_{i=0}^n (-1)^i (\mathbf{Z}_0^{-1} \Delta\mathbf{Z})^i \right) \mathbf{Z}_0^{-1} \mathbf{h}$ 为 \mathbf{y} 的 n 次迭代的近似值,

$\delta\mathbf{y} = \left(\sum_{i=n+1}^{\infty} (-1)^i (\mathbf{Z}_0^{-1} \Delta\mathbf{Z})^i \right) \mathbf{Z}_0^{-1} \mathbf{h}$ 为误差。其误差 $\delta\mathbf{y}$ 可表示为:

$$\delta\mathbf{y} = (-\mathbf{Z}_0^{-1} \Delta\mathbf{Z})^{n+1} \mathbf{y} \quad (13)$$

由此可得关于 \mathbf{x} 的 n 次迭代PDD解的误差:

$$\delta\mathbf{x} = -\mathbf{Y} \delta\mathbf{y} \quad (14)$$

由于 \mathbf{Y} 是由 N_p 组 $[\mathbf{v}^{(p)}, \mathbf{w}^{(p)}]$ 构成的块对角矩阵;而 \mathbf{y}_n 是由 N_p 组 $[y_0^{(p)}, y_1^{(p)}]^T$ 构成的块对角矩阵。对应 \mathbf{x} 的第 $i \times p$ 行的误差 $|\delta x_{i \times p, n}| = |v_i^{(p)} \delta y_0^{(p)} + w_i^{(p)} \delta y_1^{(p)}|$,考虑到对角占优的情况下 $|v_i^{(p)}| < 1, |w_i^{(p)}| < 1$,有 $|\delta x_{i \times p, n}| < |\delta y_0^{(p)}| + |\delta y_1^{(p)}|$,所以 $\|\delta\mathbf{x}\|_\infty < \|\delta\mathbf{y}\|_\infty$ 。利用 $\mathbf{E}^T \mathbf{x} = \mathbf{y}$ ^[6]、 $\|\mathbf{E}^T\|_\infty = 1$, n 次迭代求解方程(2)的解的相对误差为:

$$\frac{\|\delta\mathbf{x}\|_\infty}{\|\mathbf{x}\|_\infty} < \|\mathbf{Z}_0^{-1} \Delta\mathbf{Z}\|_\infty^{n+1} \quad (15)$$

对误差的进一步分析需要利用方程组(7)的解。关于方程组(7),考虑矩阵 \mathbf{A} 为Toeplitz三对角矩阵,即矩阵元为:

$$\begin{cases} a_i = a \\ b_i = b; i = 0, 1, \dots, N-1 \\ c_i = c \end{cases} \quad (16)$$

的简化情况。参照文献[9]的方法求解(7),有:

$$\begin{aligned} w_i^{(p)} &= -\frac{(-c/l)^{m-i} [1 - (ac/l^2)^{i+1}]}{1 - (ac/l^2)^{m+1}} \\ v_i^{(p)} &= -\frac{(-a/l)^{i+1} [1 - (ac/l^2)^{m-i}]}{1 - (ac/l^2)^{m+1}}; \end{aligned}$$

$$i = 0, 1, \dots, m-1 \quad (17)$$

其中, $l = b - ac/l$, 关于 l 有两种选择:

$$l = (b \pm \sqrt{b^2 - 4ac})/2$$

但解得的 $w_i^{(p)}$ 、 $v_i^{(p)}$ 相同。

而 $\|Z_0^{-1}\Delta Z\|_\infty$ 的表示式^[9]为:

$$\|Z_0^{-1}\Delta Z\|_\infty = \max_{(p,i)} \{ |\mu_i^{(p)}| + |\mu_i^{\prime(p)}| \} \quad (18)$$

其中 $\mu_i^{(p)}$ 、 $\mu_i^{\prime(p)}$ 是方程 $\begin{bmatrix} 1 & w_{m-1}^{(p)} \\ v_0^{(p+1)} & 1 \end{bmatrix} \begin{bmatrix} \mu_0^{(p)} \\ \mu_1^{(p)} \end{bmatrix} = \begin{bmatrix} 0 \\ w_0^{(p+1)} \end{bmatrix}$ 和

$\begin{bmatrix} 1 & w_{m-1}^{(p)} \\ v_0^{(p+1)} & 1 \end{bmatrix} \begin{bmatrix} \mu_0^{\prime(p)} \\ \mu_1^{\prime(p)} \end{bmatrix} = \begin{bmatrix} v_{m-1}^{(p)} \\ 0 \end{bmatrix}$ 的解。在上述简化情况下, 各

进程 p 所对应方程(7)有相同解, 所以式(18)可表示为:

$$\|Z_0^{-1}\Delta Z\|_\infty = \begin{cases} \frac{|w_0^{(p)}| + |v_0^{(p)}v_{m-1}^{(p)}|}{|1 - v_0^{(p)}w_{m-1}^{(p)}|}, & |a| \leq |c| \\ \frac{|v_{m-1}^{(p)}| + |w_0^{(p)}w_{m-1}^{(p)}|}{|1 - v_0^{(p)}w_{m-1}^{(p)}|}, & |a| > |c| \end{cases} \quad (19)$$

如果设定解的相对误差上限为 ε , 则算法的迭代次数为:

$$n > \frac{\ln \varepsilon}{\ln \|Z_0^{-1}\Delta Z\|_\infty} - 1 \quad (20)$$

对于矩阵 A 的一般情况, 难以取得解析分析, 因此只能作粗略的估计。将方程(2)两边同乘以一系数使得系数 $b'_i = b'(i = 0, 1, \dots, N-1)$, 得到新的方程组:

$$A'x = d' \quad (21)$$

与方程(2)有相同的解。从方程(21)中找出 $|a'_i| + |c'_i|$ 最大(即对角占优最弱)的一组 a'_i, b'_i, c'_i 替代式(16)中的 a, b, c , 利用式(19)、(20)获得 $\|Z_0^{-1}\Delta Z\|_\infty$ 的估计值可以保证算法的精度。

4 数值实验

利用 MPI 编程环境^[14], 在由联想启天 7170 为计算工作站、DES-1024R 为网络交换机组成的基于局域网的分布式存储计算机上实现了上述算法。为了评价算法性能, 使用了一个测试的对角系统, 取方程(2)中的矩阵元 $a_i = -1, b_i = 2 + \delta, c_i = -1, d_i = -1 (i = 0, 1, \dots, N-1), N = 600$ 。采用 12 个进程求解方程组, 每个进程处理 50×50 的子矩阵 $A^{(p)}$, 在 $\delta = 0.0025, \delta = 0.01$ 的情况下, 由式(19)计算 $\|Z_0^{-1}\Delta Z\|_\infty$ 分别为 0.159 和 0.013, 如果要求解的相对误差小于 $\varepsilon = 0.1\%$, 则两情况的迭代次数 n 分别为 3、1。实验测试情况如表 1 所示。在 $\delta = 0.0025$ 情况下, 非迭代的 PDD 算法(0 次迭代)的误差为 15.2%, 经过 2~3 次的迭代便可达到误差小于 0.1% 的要求精度。表 1 中 $\delta = 0.01$ 的 3 次迭代解的误差已小于浮点数的截断误差, 故未给出(用 — 表示)。

表 1 实验测试的迭代次数与误差 %

δ	迭代次数			
	0	1	2	3
0.0025	15.20	2.430	0.3830	0.0579
0.0100	1.16	0.014	0.0006	—

需要说明的是本文没有给出并行求解时间的测试结果, 因为仅对一个方程组的并行求解的通信效率是很低的。将并行求解复杂性分为计算复杂性和通信复杂性, 其计算复杂性为 $17N/p - 4 + n \times 13p$, 其中 p 为并行求解进程数, n 为迭代次数 ($n=0$ 为非迭代的 PDD 算法), $n \times 13p$ 为迭代增加的计算复杂性; 通信复杂性为 $(n+1) \times (2\alpha + 12\beta)$ (α 为通信响应时间, β

为每字节数据的传输时间, 传输数据为单精度实数), 其中 $n \times (2\alpha + 12\beta)$ 为迭代增加的通信复杂性。由于一般情况下 $\alpha \gg \beta$, 所以, 单个数据传输的通信效率很低。实际应用中(如三维 ADI-FDTD 的计算^[15]), 将 k 个三对角方程组分布到各进程同步求解, 可将各方程组的交换数据打包传输, 通信复杂性为 $(n+1) \times (2\alpha + 12k\beta)$, 通信效率将会随着 k 的增大提高。

5 结语

针对 PDD 算法在求解弱对角占优的三对角线性方程组误差较大的问题, 提出关于误差修正的迭代方案, 该方案根据对角占优的强弱设定迭代次数, 确保方程的求解精度。与非迭代的 PDD 算法比较, 复杂性有所提高, 但应用条件变弱。

参考文献:

- [1] SPALETTA G, EVANS D J. The parallel recursive decoupling algorithm for solving tridiagonal linear systems[J]. Parallel Computing, 1993, 19(3): 563-576.
- [2] 方蓉, 赵瑛. 基于递归耦合方法的三对角线性方程组分布式并行算法[J]. 计算机工程与设计, 2006, 27(4): 670-671.
- [3] AMÓDIO P, MASTRONARDI N. A parallel version of the cyclic reduction algorithm on a hypercube[J]. Parallel Computing, 1993, 19(11): 1273-1281.
- [4] MIKKELSEN K C C, KAGSTRÖM B. Parallel solution of narrow banded diagonally dominant linear systems[C]// PARA'10: Proceedings of the 10th International Conference on Applied Parallel and Scientific Computing. Berlin: Springer-Verlag, 2012: 280-290.
- [5] MECHRMANN V. Divide and conquer methods for block tridiagonal systems[J]. Parallel Computing, 1993, 19(2): 257-279.
- [6] McNALLYA J M, GAREYB L E, SHAWB R E. A communication-less parallel algorithm for tridiagonal Toeplitz systems[J]. Journal of Computational and Applied Mathematics, 2008, 212(2): 260-271.
- [7] McNALLYA J M. A fast algorithm for solving diagonally dominant symmetric pentadiagonal Toeplitz systems[J]. Journal of Computational and Applied Mathematics, 2010, 234(4): 995-1005.
- [8] SUN X H, ZHANG H, NI L. Efficient tridiagonal solvers on multicomputers[J]. IEEE Transactions on Computers, 1992, 41(3): 286-296.
- [9] SUN X H. Application and accuracy of the parallel diagonal dominant algorithm[J]. Parallel Computing, 1995, 21(8): 1241-1268.
- [10] SUN X H, ZHANG W. A parallel two-level hybrid method for tridiagonal systems and its application to fast poisson solvers[J]. IEEE Transactions on Parallel and Distributed Systems, 2004, 15(2): 97-106.
- [11] 迟利华, 刘杰, 李晓梅. 三对角线性方程组的一种有效并行算法[J]. 计算机学报, 1999, 22(2): 218-221.
- [12] CLIMENT J J, PEREA C, TORTOSA L, et al. An overlapped two-way method for solving tridiagonal linear systems in a BSP computer[J]. Applied Mathematics and Computation, 2005, 161(2): 475-500.
- [13] 张衡, 张武. 块对角占优块三对角方程组的块重叠分割无通信并行求解方法[J]. 工程数学学报, 2007, 24(6): 1080-1090.
- [14] 都志辉, 李三立. 高性能计算并行编程技术——MPI 并行程序设计[M]. 北京: 清华大学出版社, 2001.
- [15] NAMIKI T. 3-D ADI-FDTD method unconditionally stable time-domain algorithm for solving full vector maxwells equations[J]. IEEE Transactions on Microwave Theory and Techniques, 2000, 48(10): 1743-1747.