国防科技大学学报
JOURNAL OF NATIONAL UNIVERSITY OF DEFENSE TECHNOLOGY
第12卷第3期　1990年9月　Vol.12 No.3

MIMD上三层格式的窗口并行迭代方法

熊　岳　山
（系统工程与应用数学系）

摘　要　本文对一般的三层格式给出了在MIMD机上实现的窗口并行迭代方法，给出了WBJ格式，分析了三层格式的收敛性。文章的结论表明窗口并行迭代法的收敛性与窗口大小无关，窗口大小影响每台处理机的使用效率。

关键词　偏微分方程，差分，迭代法，窗口，窗口块雅可比方法，三层格式
分类号　O241.3

发展方程的三层格式通常归纳为下面三层格式的解[1][2]

$$\begin{cases} My(t_1)=b_1(t_1) \\ My(t_2)=Py(t_1)+b_2(t_2) \\ My(t_r)=Py(t_{r-1})+Qy(t_{r-2})+b(t_r) \qquad r=3,\ \cdots,\ N \end{cases} \tag{1}$$

这里$y(t_i)$，b_i为n维列向量，M，P，Q为$n\times n$矩阵，N为时间层层数。Saltz J H等[3]对二层格式WBJ（窗口块Jacobi）、WBSOR（窗口块SOR）迭代格式，得到了一系列结果，本文拓广和完善了Saltz J H等的方法到三层格式，给出了WBJ迭代格式，证明了三层格式的收敛性。

1　WBJ迭代格式

传统的迭代法关于时间层是每一层单独迭代若干步，直至得到收敛解后再转入下一时间层迭代。若每次关于若干时间层迭代一步后再转入下一次迭代，直到初始层得到收敛解后再转入对下一步的若干时间层进行迭代，直到求出收敛解为止。我们称这些出现的时间层构成的集合为窗口，窗口的大小记为w.

设

$$M=\begin{bmatrix} M_{11} & M_{12}\cdots M_{1q} \\ \vdots & \\ M_{q1} & M_{q2}\cdots M_{qq} \end{bmatrix},\quad P=\begin{bmatrix} P_{11} & P_{12}\cdots P_{1q} \\ \vdots & \\ P_{q1} & P_{q2}\cdots P_{qq} \end{bmatrix},\quad Q=\begin{bmatrix} Q_{11} & Q_{12}\cdots Q_{1q} \\ \vdots & \\ Q_{q1} & Q_{q2}\cdots Q_{qq} \end{bmatrix}$$

则(1)式的 WBJ 算法如下：

1988年12月9日收稿

```
Pardo i=1 to q
  {
      y_i^0(t_1)=y_i(t_0)
  }
For r=1 to N
  Window=min(w, N-r+1)
  Pardo i=1 to q
    {
    Pardo s=r+1 to r+window-1
      {
        y_i^0(t_s)=y_i^0(t_r)
        }
      }
    Beginning with ν=0 increment ν until ‖y^{ν+1}(t_r)-y^ν(t_r)‖<ε
      {
      For s=r to r+window-1
      IF (s=r)
      {
      Solve
```

$$M_{ii}y_i^{\nu+1}(t_s)=-\sum_{j\neq i}M_{ij}y_j^{\nu}(t_s)+b_i(t_s)$$

```
      }
      else
      IF(s=r+1)
      {
      Solve
```

$$M_{ii}y_i^{\nu+1}(t_s)=-\sum_{j\neq i}M_{ij}y_j^{\nu}(t_s)+P_{ii}y_i^{\nu+1}(t_{s-1})+\sum_{j\neq i}P_{ij}y_j^{\nu}(t_{s-1})+b_i(t_s)$$

```
      }
      else
      {

      solve
```

$$M_{ii}y_i^{\nu+1}(t_s)=-\sum_{j\neq i}M_{ij}y_j^{\nu}(t_s)+P_{ii}y_i^{\nu+1}(t_{s-1})+\sum_{j\neq i}P_{ij}y_j^{\nu}(t_{s-1})+Q_{ii}y_i^{\nu+1}(t_{s-2})+\sum_{j\neq i}Q_{ij}y_j^{\nu}(t_{s-2})+b_i(t_s)$$

```
      }
```

```
NEXT S
r⇐r+window
Pardo i=1 to q
  {
    y_i^0(t_r)=y_i^{v+1}(t_{r-1})
  }
NEXT r
Pardo i=1 to q
  {
    y_i(t_r)=y_i^{v+1}(t_r)
  }
```

2　三层格式的收敛性

设　　$M=R_m-S_m,\ P=R_p-S_p,\ Q=R_q-S_q$

R_m非异，对应于WBJ方法，R_m，R_p，R_q分别为M，P，Q的块对角剖分。

此时，WBJ迭代格式具有如下形式：

$$R_m y^{v+1}(t_1)=S_m y^v(t_1)+b(t_1)$$
$$R_m y^{v+1}(t_2)=S_m y^v(t_2)+R_p y^{v+1}(t_1)-S_p y^v(t_1)+b(t_2)$$
$$R_m y^{v+1}(t_r)=S_m y^v(t_r)+R_p y^{v+1}(t_{r-1})-S_p y^v(t_{r-1})+R_q y^{v+1}(t_{r-2})$$
$$-S_q y^v(t_{r-2})+b(t_r),\quad r=3,\cdots,w$$

重新排列整理后得：　　$Ry^{v+1}=Sy^v+B$

这里

$$R=\begin{pmatrix} I & & & & & \\ -R_m^{-1}R_p & I & & & & \\ -R_m^{-1}R_q & -R_m^{-1}R_p & I & & & \\ & \ddots & \ddots & \ddots & & \\ & & -R_m^{-1}R_q & -R_m^{-1}R_p & I \end{pmatrix}_{w\times w},\quad B=\begin{pmatrix} R_m^{-1}\cdot b(t_1) \\ \vdots \\ R_m^{-1}b(t_w) \end{pmatrix}_{w\times 1}$$

$$S=\begin{pmatrix} R_m^{-1}S_m & & & & \\ -R_m^{-1}S_p & R_m^{-1}S_m & & & \\ -R_m^{-1}S_q & -R_m^{-1}S_p & R_m^{-1}S_m & & \\ & \ddots & \ddots & \ddots & \\ & & -R_m^{-1}S_q & -R_m^{-1}S_p & R_m^{-1}S_m \end{pmatrix}_{w\times w}$$

关于WBJ格式有下述收敛性定理。

定理 1　存在R^{-1}使$y^{v+1}=R^{-1}Sy^v+R^{-1}B$，且$P(R^{-1}S)=P(R_m^{-1}S_m)$．若假设$P(R_m^{-1}S_m)<1$，则WBJ方法收敛。

证明　以$w=3$为例证明之。此时

$$R=\begin{pmatrix} I & & \\ -R_m^{-1}R_p & I & \\ -R_m^{-1}R_q & -R_m^{-1}R_p & I \end{pmatrix},\quad R^{-1}=\begin{pmatrix} I & & \\ R_m^{-1}R_p & I & \\ (R_m^{-1}R_p)^2+R_m^{-1}R_q & R_m^{-1}R_p & I \end{pmatrix}$$

满足 $$R^{-1}R=I$$

因R^{-1}，S 为块下三角阵，故

$$R^{-1}S=\begin{pmatrix} R_m^{-1}S_m & & \\ * & R_m^{-1}S_m & \\ * & * & R_m^{-1}S_m \end{pmatrix}$$

所以 $$\det(\lambda I-R^{-1}S)=[\det(\lambda I-R_m^{-1}S_m)]^3,\ \rho(R^{-1}S)=\rho(R_m^{-1}S_m)$$

3 MIMD处理机上WBJ格式的效率分析

考虑单个处理机P，定义

T：完成单个块迭代一步所需的时间；

cT：处理机间的访问时间，c 为访问系数；

T_0：解块迭代所需的时间。

由上述定义，不难得到利用WBJ格式在单个窗口中完成一步迭代至少需$cT+T_0$的时间。

又定义处理机的工作效率

$$U_p=\frac{mT}{T_{总}}$$

这里m为块迭代的总次数，$T_{总}$为完成计算所需的总时间。

关于U_p的上界估计成立下面的关系。

定理 2 $U_p\leqslant 1/\max[1,(c+T_0/T)1/w]$，特别若$c>[w-(T_0/T)]$

则 $$U_p\leqslant w/(c+T_0/T)$$

证明 因 $T_{总}\geqslant\max\left(mT,\ (cT+T_0)\frac{m}{w}\right)$

故 $$\begin{aligned}U_p\leqslant mT/\max(mT,(cT+T_0)m/w)&=1/\max(1,(cT+T_0)1/wT)\\&=1/\max(1,(c+T_0/T)1/w)\end{aligned}$$

若 $$c>\left(w-\frac{T_0}{T}\right),\ 则\left(c+\frac{T_0}{T}\right)\frac{1}{w}>1$$

故 $$U_p\leqslant w/(c+T_0/T)$$

由定理2可知：窗口的大小影响处理机的使用效率。

参考文献

[1] Richtmyer R D, Morton K W. Finite Difference Methods for Initial—value Problem. 2'nd Edition, Interscience.New York, 1967

[2] 郭本瑜. 偏微分方程的差分方法. 北京：科学出版社，1988

[3] Saltz J H,Naik V K.Towards Developing Robust Algorithms for Solving Partial Differential Equations on MIMD Machines. Parallel Computing 1988, (6):19～44

[4] 王嘉谟，沈毅主编. 并行计算方法. 北京：国防工业出版社，1987

（下转第20页）

and q be the number P_I have at the end of the step. Then P_I will win if

$$r \equiv 1 \text{ or } 0 \pmod{2s+2} \text{ if } q \text{ is odd, and}$$

$$r \equiv s+1 \text{ or } s+2 \pmod{2s+2} \text{ if } q \text{ is even}$$

when s is odd and

$$r \equiv 1 \text{ or } 0 \pmod{s+2} \text{ if } q \text{ is odd, and}$$

$$r \equiv s+1 \pmod{s+2} \text{ if } q \text{ is even}$$

when s is even.

The theorem can be proved by induction. As a by-product of the theorem we have

Corollary $\Gamma(n, s)$ is a win for P_{II} iff $n \equiv s+1 \pmod{2s+2}$ when s is odd and $n \equiv s+1 \pmod{s+2}$ when s is even.

Key words: operations research, game theory, strategy, combination, winning strategy

（上接第24页）

Window Parallel Algorithms for Solving Three-level Schemes on MIMD Machines

Xiong Yueshan

(Department of Applied Mathematics and System Engineering)

Abstract

In this paper, the window parallel algorithms for solving general three-level schemes On MIMD machines are proposed, the WBJ schemes are given, and the convergence for three-level schemes is proved. The conclusions of the paper show that the convergence for the window parallel algorithms is independent of the size of the window. Which has influence on the efficiency of each processor.

Key words: partial differential equations, differences, iterative methods, window, WBJ method, three-level schemes