Journal of Southeast University (English Edition) Vol.19 No.3 Sept. 2003 ISSN 1003—7985

Parallel EOI algorithm with different insertion schemes

Liu Qingfu Zhong Weijun

(College of Economics and Management, Southeast University, Nanjing 210096, China)

Abstract: A parallel embedding overlapped iterative (EOI) algorithm about classic implicit equations with asymmetric Saul'yev schemes (CIS-EOI) to solve one-dimensional diffusion equations is discussed to improve the properties of the segment classic implicit iterative (SCII) algorithm. The structure of CIS-EOI method is given and the stability of scheme and convergence of iteration are proved by matrix method. The property of gradual-approach convergence is also discussed. It has been shown that the convergent rate is faster and the property of gradual-approach convergence also becomes better with the increasing of the net point in subsystems than with the SCII algorithm. The simulation examples show that the parallel iterative algorithm with a different insertion scheme CIS-EOI is more effective.

Key words: diffusion equation; different insertion scheme; convergent rate; property of gradual-approach convergence

The problem is: find the solution $u(x,t)$ in the domain $D:\{0 \leqslant x \leqslant 1\}$ of

$$u_t = u_{xx} \tag{1}$$

with the boundary conditions

$$u(0,t) = g_0(t), u(1,t) = g_1(t) \tag{2}$$

and the initial condition

$$u(x,0) = f(x) \tag{3}$$

Let Δx and Δt be the grid spaces in the directions of x and t, where $\Delta x = 1/m$, m is a positive integer. The approximate value u_i^n of the solution $u(x,t)$ for problems (1) to (3) is to be computed at the grid points (x_i, t_n), where $x_i = i\Delta x, i = 0,1,\cdots,m$; $t_n = n\Delta t, n = 1,2,\cdots$. For simplicity, we denote points (x_i, t_n) by (i,n).

Among the finite difference methods for the numerical solution of problems (1) to (3), a well-known classical implicit scheme is[1,2]

$$-ru_{i-1}^{n+1} + (1+2r)u_i^{n+1} - ru_{i+1}^{n+1} = u_i^n \tag{4}$$

As we know, Eq.(4) is unconditionally stable and has truncation error $O(\Delta t + \Delta x^2)$, in which $r = \Delta t/\Delta x^2$. Applying (4) to all points along $(n+1)$-th time level, it can solve the following tri-diagonal system of equations

$$\bar{\boldsymbol{A}}\boldsymbol{U}^{n+1} = \bar{\boldsymbol{b}} \tag{5}$$

with

$$\bar{\boldsymbol{A}} = \begin{bmatrix} 1+2r & -r & & & \\ -r & 1+2r & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 1+2r & -r \\ & & & -r & 1+2r \end{bmatrix}_{(m-1)\times(m-1)}$$

$$\boldsymbol{U}^n = \{u_1^n, u_2^n, \cdots, u_{m-1}^n\}^{\mathrm{T}}$$

As usual we consider a class of algorithms for solving (5) which is based on the splitting of the matrix $\boldsymbol{A}$ as

$$\boldsymbol{A} = \bar{\boldsymbol{M}} - \bar{\boldsymbol{N}} \tag{6}$$

where

$$\bar{\boldsymbol{M}} = \begin{bmatrix} \boldsymbol{A}_1 & & & \\ & \boldsymbol{A}_2 & & \\ & & \ddots & \\ & & & \boldsymbol{A}_k \end{bmatrix}_{(m-1)\times(m-1)}$$

Received 2002-11-04.

Biographies: Liu Qingfu (1973—), male, graduate; Zhong Weijun (corresponding author), male, doctor, professor, zhong.wj@public1.ptt.js.cn.

$$\bar{\boldsymbol{N}} = \begin{bmatrix} \boldsymbol{0}_1 & \boldsymbol{N}_1 & & & \\ \boldsymbol{N}_1^{\mathrm{T}} & \boldsymbol{0}_2 & \boldsymbol{N}_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \boldsymbol{N}_{k-2}^{\mathrm{T}} & \boldsymbol{0}_{k-1} & \boldsymbol{N}_{k-1} \\ & & & \boldsymbol{N}_{k-1}^{\mathrm{T}} & \boldsymbol{0}_k \end{bmatrix}_{(m-1)\times(m-1)}$$

with

$$\boldsymbol{A}_i = \begin{bmatrix} 1+2r & -r & & & \\ -r & 1+2r & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 1+2r & -r \\ & & & -r & 1+2r \end{bmatrix}_{r_i \times r_i}$$

$$i = 1,2,\cdots,k$$

and

$$\boldsymbol{N}_i = \begin{bmatrix} & & 0 \\ & \iddots & \\ & 0 & \\ r & & \end{bmatrix}_{r_i \times r_{i+1}} \qquad i = 1,2,\cdots,k-1$$

where $\sum_{i}^{k} r_i = m-1$ with $2 \leqslant r_i \leqslant m-1$.

Then the segment classical implicit iterative (SCII) algorithm for solving (1) – (3) can be expressed as

$$\bar{\boldsymbol{M}}\boldsymbol{U}^{[n+1](s+1)} = \bar{\boldsymbol{N}}\boldsymbol{U}^{[n+1](s)} + \bar{\boldsymbol{b}} \qquad s = 0,1,2,\cdots \tag{7}$$

Hence, we get the estimates of the convergent rate and the property of gradual-approach convergence with the increasing of the net point number of SCII as

$$\| \bar{\boldsymbol{M}}^{-1}\bar{\boldsymbol{N}} \|_\infty \leqslant \frac{r}{r+1} \qquad 1 < r_i < m-1 \tag{8}$$

$$\| \bar{\boldsymbol{M}}^{-1}\ \bar{\boldsymbol{N}} \|_\infty \xrightarrow{p\to\infty} \frac{r}{\alpha} \qquad \forall\, r_i = p \tag{9}$$

where $\alpha = r + \dfrac{1+\sqrt{1+4r}}{2}$.

1 The CIS-EOI Method

1.1 Asymmetric schemes[3]

Before reconstructing a better embedding overlapped iterative (EOI) parallel method, we briefly review the asymmetric Saul'yev schemes described as

$$-\frac{r}{2}u_{j-1}^{n+1} + \left(1+\frac{3r}{2}\right)u_j^{n+1} - ru_{j+1}^{n+1} = \frac{r}{2}u_{j-1}^{n} + \left(1-\frac{r}{2}\right)u_j^{n} - ru_{j-1}^{n+1} + \left(1+\frac{3r}{2}\right)u_j^{n+1} - \frac{r}{2}u_{j+1}^{n+1} = \left(1-\frac{r}{2}\right)u_j^{n} + \frac{r}{2}u_{j+1}^{n}$$

Both of which are unconditionally stable and have truncation error $O(\Delta t + \Delta x)$.

1.2 Design of CIS-EOI method

Asymmetric Saul'yev schemes are taken as insertion equations to construct new difference equations as

$$\boldsymbol{A}\boldsymbol{U}^{n+1} = \boldsymbol{b} \tag{10}$$

where

$$
\boldsymbol{A} = \begin{bmatrix} \boldsymbol{B}_1 & \boldsymbol{H}_1 & & & \\ \boldsymbol{H}_1^{\mathrm{T}} & \boldsymbol{B}_2 & \boldsymbol{H}_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \boldsymbol{H}_{k-2}^{\mathrm{T}} & \boldsymbol{B}_{k-1} & \boldsymbol{H}_{k-1} \\ & & & \boldsymbol{H}_{k-1}^{\mathrm{T}} & \boldsymbol{B}_k \end{bmatrix}_{(m-1)\times(m-1)}
$$

$$
\boldsymbol{B}_i = \begin{bmatrix} 1+\frac{3r}{2} & -r & & & \\ -r & 1+2r & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 1+2r & -r \\ & & & -r & 1+\frac{3r}{2} \end{bmatrix}_{r_i\times r_i} \qquad i = 1,2,\cdots,k
$$

$$
\boldsymbol{H}_i = \begin{bmatrix} & & 0 \\ & 0 \;\; \iddots & \\ -\frac{r}{2} & & \end{bmatrix}_{r_i\times r_{i+1}}
$$

where $\boldsymbol{U}^n = \{u_1^n, u_2^n, \cdots, u_{m-2}^n, u_{m-1}^n\}^{\mathrm{T}}$ and $\boldsymbol{b}$ are determined by $\boldsymbol{U}^n$, $g_0(t_{n+1})$ and $g_1(t_{n+1})$.

According to the construct above we split the matrix $\boldsymbol{A}$ as

$$
\boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N} \tag{11}
$$

where

$$
\boldsymbol{M} = \begin{bmatrix} \boldsymbol{A}_1 & & & \\ & \boldsymbol{A}_2 & & \\ & & \ddots & \\ & & & \boldsymbol{A}_k \end{bmatrix}_{(m-1)\times(m-1)}
$$

$$
\boldsymbol{N} = \begin{bmatrix} \boldsymbol{0}_1 & \boldsymbol{N}_1 & & & \\ \boldsymbol{N}_1^{\mathrm{T}} & \boldsymbol{0}_2 & \boldsymbol{N}_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \boldsymbol{N}_{k-2}^{\mathrm{T}} & \boldsymbol{0}_{k-1} & \boldsymbol{N}_{k-1} \\ & & & \boldsymbol{N}_{k-1}^{\mathrm{T}} & \boldsymbol{0}_k \end{bmatrix}_{(m-1)\times(m-1)}
$$

with

$$
\boldsymbol{A}_i = \begin{bmatrix} 1+\frac{3}{2}r & -r & & & \\ -r & 1+2r & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 1+2r & -r \\ & & & -r & 1+\frac{3}{2}r \end{bmatrix}_{r_i\times r_i} \qquad i = 1,2,\cdots,k
$$

and

$$N_i = \begin{bmatrix} 0 & & 0 \\ & & \iddots \\ & 0 & \\ \frac{r}{2} & & 0 \end{bmatrix}_{r_i \times r_{i+1}} \qquad i = 1,2,\cdots,k-1$$

where $\sum_{i}^{k} r_i = m - 1$ with $2 \leqslant r_i \leqslant m - 1$. Then the CIS-EOI method can be expressed as

$$\boldsymbol{M}\boldsymbol{U}^{[n+1](s+1)} = \boldsymbol{N}\boldsymbol{U}^{[n+1](s)} + \boldsymbol{b} \qquad s = 0,1,2,\cdots \tag{12}$$

where $\boldsymbol{M}^{-1}\boldsymbol{N}$ is the iteration matrix. Therefore the solution of (12) can be divided into a series of smaller equations as

$$\boldsymbol{A}_i\boldsymbol{U}_i^{[n+1](s+1)} = \boldsymbol{b}_i(\boldsymbol{U}^{[n+1](s)}, r, \boldsymbol{b}) \qquad s = 0,1,2,\cdots \tag{13}$$

1.3 Stability analysis

The matrix description of (10) is defined as

$$(\boldsymbol{I} + r\boldsymbol{G}_1)\boldsymbol{U}^{n+1} = (\boldsymbol{I} - r\boldsymbol{G}_2)\boldsymbol{U}^n + \boldsymbol{b}_1 \qquad n = 0,1,\cdots \tag{14}$$

Supposing

$$\boldsymbol{T} = (\boldsymbol{I} + r\boldsymbol{G}_1)^{-1}(\boldsymbol{I} - r\boldsymbol{G}_2) \tag{15}$$

where

$$\boldsymbol{G}_1 = \begin{bmatrix} \boldsymbol{Q}_1 & \boldsymbol{R}_1 & & & \\ \boldsymbol{R}_1^{\mathrm{T}} & \boldsymbol{Q}_2 & \boldsymbol{R}_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \boldsymbol{R}_{k-2}^{\mathrm{T}} & \boldsymbol{Q}_{k-1} & \boldsymbol{R}_{k-1} \\ & & & \boldsymbol{R}_{k-1}^{\mathrm{T}} & \boldsymbol{Q}_k \end{bmatrix}_{(m-1)\times(m-1)}$$

$$\boldsymbol{G}_2 = \begin{bmatrix} \boldsymbol{W}_1 & \boldsymbol{R}_1 & & & \\ \boldsymbol{R}_1^{\mathrm{T}} & \boldsymbol{W}_2 & \boldsymbol{R}_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \boldsymbol{R}_{k-2}^{\mathrm{T}} & \boldsymbol{W}_{k-1} & \boldsymbol{R}_{k-1} \\ & & & \boldsymbol{R}_{k-1}^{\mathrm{T}} & \boldsymbol{W}_k \end{bmatrix}_{(m-1)\times(m-1)}$$

with

$$\boldsymbol{Q}_i = \begin{bmatrix} \frac{3}{2} & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & \frac{3}{2} \end{bmatrix}_{r_i \times r_i}$$

$$\boldsymbol{W}_i = \begin{bmatrix} \frac{1}{2} & & & & \\ & 0 & & & \\ & & \ddots & & \\ & & & 0 & \\ & & & & \frac{1}{2} \end{bmatrix}_{r_i \times r_i} \qquad i = 1,2,\cdots,k$$

and

$$\boldsymbol{R}_i = \begin{bmatrix} & & 0 \\ & & \iddots \\ & 0 & \\ -\frac{1}{2} & & \end{bmatrix}_{r_i \times r_{i+1}} \qquad i = 1,2,\cdots,k-1$$

where $\sum_{i}^{k} r_i = m - 1$ with $2 \leqslant r_i \leqslant m - 1$; $\boldsymbol{G}_1$ is nonnegative, so we have $\|(\boldsymbol{I} + r\boldsymbol{G}_1)^{-1}\| \leqslant 1$ and $\|\boldsymbol{I} - r\boldsymbol{G}_2\| \leqslant \left\{ \left|1 - \frac{r}{2}\right|, |1 - r|, 1 \right\}$. Similar to the method of Ref.[4], we have $\|\boldsymbol{T}\| \leqslant 1 \Leftrightarrow r \leqslant 2$. So Eq.(14) is stable if $r \leqslant 2$.

1.4 Analyses of convergent rate and property of gradual-approach convergence

Lemma[5] If $\boldsymbol{M} = (m_{i,j})$ is an $n \times n$ matrix, $\boldsymbol{N} = (n_{i,j})$ is an $n \times m$ matrix, and

$$|m_{ii}| > \sum_{j \neq i} |m_{ij}| \qquad i = 1, 2, \cdots, n$$

Then

$$\|\boldsymbol{M}^{-1}\boldsymbol{N}\|_\infty \leqslant \max_i \left\{ \sum_{j=1}^{m} |n_{i,j}| \Big/ \left(|m_{ii}| - \sum_{j \neq i} |m_{i,j}| \right) \right\} \tag{16}$$

According to the lemma above, we have

$$\|\boldsymbol{M}^{-1}\boldsymbol{N}\|_\infty \leqslant \frac{r}{r + 2} \qquad 1 < r_i < m - 1 \tag{17}$$

Obviously, CIS-EOI algorithm (12) is convergent. Similar to the method adopted in Ref.[2], we have

$$\|\boldsymbol{M}^{-1}\boldsymbol{N}\| \xrightarrow{p \to \infty} \frac{r}{2\alpha - r} \qquad \forall\, r_i = p \tag{18}$$

where $\alpha = r + \dfrac{1 + \sqrt{1 + 4r}}{2}$ and $r \leqslant 2$.

From the results above, it is easy to see that the convergent rate and property of gradual-approach convergence of CIS-EOI is superior to that of SCII. The numerical example shows that the accuracy of CIS-EOI is also better. According to the structure of CIS-EOI algorithm, we can construct a much better computational method in parallel with more accuracy, higher convergent rate and better property of gradual-approach convergence methods.

2 Numerical Example and Experimental Results

Consequently, we provide numerical experiments made for problems (1) - (3) in which[6]

$$f(x) = 4x(1 - x), \; g_0(t) = g_1(t) = 0$$

The exact solution of the problem is

$$U(x, t) = \frac{32}{\pi^3} \sum_{k=1,3,5,\cdots}^{\infty} \frac{1}{k^3} \mathrm{e}^{-k^2\pi^2 t} \sin(k\pi x)$$

The accuracy of SCII and CIS-EOI which is given on Challenge L with 8 CPUs[7] with $r = 1.0$, $\Delta x = 0.01$, $n = 200$ and error $= 10^{-8}$ is compared in Tab.1 by putting the absolute error and percentage error at each point along the mesh line. From Tab.1 we can see that the parallel CIS-EOI method is better than that of SCII. In addition, the property of gradual-approach convergence of CIS-EOI algorithm in Tab.2 is better than SCII algorithm.

Tab.1 The numberical results of SCII and CIS-EOI algorithms

x_i	Exact result/10^{-1}	Numberical result/10^{-1}		Absolute error/10^{-4}		Percentage error/10^{-4}	
		SCII	CIS-EOI	SCII	CIS-EOI	SCII	CIS-EOI
0.1	2.670 846 5	2.671 671 8	2.671 340 3	0.825 269 7	0.493 804 9	3.089 918 2	1.848 870 3
0.2	5.041 097 2	5.042 234 1	5.041 873 5	1.136 915 0	0.776 272 0	2.255 292 7	1.539 886 9
0.3	6.873 199 8	6.874 127 7	6.873 858 9	0.927 849 2	0.659 040 2	1.349 952 3	0.958 855 1
0.4	8.019 110 9	8.019 686 4	8.019 503 6	0.575 495 7	0.392 668 7	0.717 655 2	0.489 666 2
0.5	8.407 675 6	8.408 095 5	8.407 966 3	0.419 886 9	0.290 652 7	0.499 409 1	0.345 699 2
0.6	8.019 110 9	8.019 686 4	8.019 503 6	0.575 495 7	0.392 668 7	0.717 655 2	0.489 666 2
0.7	6.873 199 8	6.874 127 7	6.873 858 9	0.927 849 2	0.659 040 2	1.349 952 3	0.958 855 1
0.8	5.041 097 2	5.042 234 1	5.041 873 5	1.136 915 0	0.776 272 0	2.255 292 7	1.539 886 9
0.9	2.670 846 5	2.671 671 8	2.671 340 3	0.825 269 7	0.493 804 9	3.089 918 2	1.848 870 3

3 Conclusion

During the process of solving implicit difference equations, we have developed a parallel CIS-EOI algorithm for improving the convergent rate and property of gradual-approach convergence. We have given the convergent rate and property of gradual approach convergence in this paper. From both the proof in theory and examples, we can see that the CIS-EOI algorithm is better in convergent rate and property of gradual-approach convergence than that of SCII algorithm. So the computing time in parallel is greatly shortened. Its numerical results are also more accuracy because of inserting different schemes. Of course, we may consider a good many other effective schemes relative to insertion scheme in parallel computation.

Tab.2 The convergent rate of SCII and CIS-EOI algorithms

r_i	3	5	10	20	25	50
S_{CI}	14	11	8	4	2	1
$S_{CIS\text{-}EOI}$	10	7	5	3	1	1

Note: S_{CI} and $S_{CIS\text{-}EOI}$ are the iterative numbers about SCII and CIS-EOI algorithms, respectively.

References

[1] Liu Qingfu. Parallel iterative algorithms for solving the implicit difference equations [J]. *Guizhou Science*, 2002, **6**(2): 29 - 37. (in Chinese)

[2] Chen Jin, Zhang Baolin. Massively parallel algorithm for parabolic equation: one dimensional problem [J]. *Computational Physics Sinica*, 1994, **4**(5): 372 - 382.

[3] Zhang Baolin, Gu Tongxiang, Mo Zeyao. *Principles and methods of numerical parallel computation* [M]. Beijing: National Defence Industry Press, 1999. 212 - 216. (in Chinese)

[4] Chen Jin, Zhang Baolin. A Class of alternation block crank-nicolson method [J]. *Intern J Computer Math*, 1992, **45**(5): 89 - 112.

[5] Hu Jiagan. The estimates of $\|M^{-1}N\|_\infty$ and the optimally scaled matrix [J]. *Journal of Computational Mathematics*, 1984, **2**(6): 122 - 129.

[6] Zhang Baolin, Su Xiumin. Parallel algorithms for solving the implicit difference equations [J]. *Computational Physics Sinica*, 1992, **9**(3): 250 - 256. (in Chinese)

[7] Zhang Baolin. One finite difference methods for solving initial-boundary value problems in parallel [J]. *Comm on Appl Math and Comput*, 1996, **6**(1): 53 - 65.

一类相异嵌入格式的嵌套迭代并行算法

刘庆富　　仲伟俊

（东南大学经济管理学院，南京 210096）

摘　要　为改善并行迭代算法 SCII 的收敛速度和渐近收敛性质，本文给出了求解一维扩散方程的一类相异嵌入格式的嵌套迭代并行算法 CIS-EOI. 论述了 CIS-EOI 算法的基本构造，并用矩阵理论证明了格式的稳定性；讨论了迭代收敛性和渐近收敛性质. CIS-EOI 算法不仅加快了迭代法的收敛速度、改善了网格加密时的渐近收敛性质，还提高了精确度，比单纯采用 SCII 算法要好. 文中数值例子表明相异嵌入格式的嵌套迭代并行算法 CIS-EOI 是有效的.

关键词　扩散方程；嵌入格式；收敛速度；渐近收敛性质

中图分类号　O241.28