

超並列計算機向き行列解法アルゴリズムの検討

猪貝 光祥 大河内 俊夫* 金野 千里*

(株) 日立超LSIエンジニアリング

* (株) 日立製作所 中央研究所

超並列計算機による大規模連立1次方程式の計算高速化を目的として、Multigrid法、ADI法に関するアルゴリズムの検討と実際の問題への適用を行なった。数値実験の結果、Multigrid法ではスカラ計算機上でもILUBCG法と同程度の収束性能（収束判定値 ϵ 以下までに要する演算量）が得られること、ADI法では特に移流項に対して有効であることが確認できた。両解法の持つ並列性、適用範囲、収束特性を比較検討し、共に超並列計算機の有力な行列解法となる見通しを得ることができた。

Study on the Algorithms for Matrix Solver on Massively Parallel Computer

Mitsuyoshi Igai Toshio Okouchi* Chisato Konno*

Hitachi VLSI Engineering Corp. (Kodaira, Tokyo, 187)

* Central Research Laboratory, Hitachi, Ltd. (Kokubunji, Tokyo, 185)

On the purpose of developing a matrix solver on massively parallel computers(MPC), we have investigated the both Multigrid and ADI method, and had numerical experiments, applying them to practical problems (2-D Poisson equation, etc). We show that Multigrid method is as efficient as ILUBCG on the scalar processor, and ADI method is especially powerful for the convective diffusion problems. We have concluded that they will be useful enough on MPC.

1. はじめに

多くの偏微分方程式の求解においては最終的に大規模な連立一次方程式を解く必要が生じ、その全計算時間に占める割合も極めて高い。連立一次方程式を解く行列解法に要求される性能としては、以下の4つが挙げられる。

- ① 確実性（収束性。正解を確保する安定性）
- ② 高速性
- ③ 少ないワークメモリ
- ④ 計算機特性を活かしたアルゴリズム

これまでのベクトル計算機では共役勾配法、ガウス消去法といった行列解法が、その確実性とも併せて主流となっている。

並列計算機においても、これらの解法を登用しようとする試みはあるが((3)(5))、アルゴリズムに内在する強い逐次性から分散メモリ型の並列計算機の特長を引き出すことは難しいと考える。

並列計算機はプロセッサを結合する通信網、個々のプロセッサ性能等、各々独自のアーキテクチャ構成をとっており、並列計算機向き行列解法の選定には特に④の計算機特性を活かしたアルゴリズムが重要な要素となる。

本報告では、格子点単位の並列粒度を有するMultigrid法、格子線単位の並列粒度を有するADI法（交互方向陰解法）を取り上げ、ポアソン方程式への適用を通して各解法の収束特性、収束性能について明らかにする。

2. 行列解法と並列性

表1に主な行列解法とそのアルゴリズムに内在する並列性をまとめた。

#1～#4の陽的、準陰的解法は、その高並列性から多くの並列計算機（特に、分散メモリ型）へ適用されている((1)(2)(4))。#5, 6は、陰的解法で、陽的、準陰的解法に比べ確実性の点で優れている反面、アルゴリズムに内在する並列性が低く（逐次性が強く）、並列（特に超並列）計算機への適用が困難な解法と言える。

並列計算機では陽解法的な高並列性を維持しつつ、陰解法的な確実性（収束性）を持つ行列解法が、有望な解法と言える。

表1 主な行列解法の並列性

#	主な行列解法 アルゴリズム	特 徴	アルゴリズムに内在する並列性
1	Multigrid法	複数grid ^(a) における反復計算により、消去しづらい低次モード誤差を急速に除去可能。	陽的、R/B SOR等のSOLVERを使用するため、格子点を極の並列性有する。全格子点(行列の各行)について並列処理可能。
2	ヤコビ法	古典的点反復法	R/Bは、陽的の並列性有。SORは格子点の番号付けに依存。
3	Red/Black SOR, SOR	逐次過緩和法	各方向解法時、複数の3項方程式を独立に求解可能。内積、 Δ^2 計算等が並列処理可能。
4	ADI法	交互方向陰解法	強い逐次性があり並列化困難。需行列の特性を利用すると、部分的に並列化可能。
5	PCG, ICCG	共役勾配法 $F(x)=(x, Ax)-2(x, b)$ の最小値問題として解く。	
6	ガウス消去法	直接法	

(注) (a) grid=メッシュ分割した解析領域に含まれる格子点の集合

3. Multigrid法

3.1 古典的反復法の問題点

ヤコビ反復法に代表される陽的解法は、

- (1) 単純なアルゴリズムであり、ワークメモリも必要としない。
- (2) 反復計算は各格子点毎に計算でき、高い並列性を有する。

等の利点がある。しかしその反面、拡散方程式のような単純な問題でも、領域により拡散係数が著しく違ったり、境界条件として固定境界条件が少ない場合、その収束性が悪く、正解を安定に確保できないという問題がある。

この収束性劣下の原因を分析すると、局所的な平衡条件のみを満足するような反復スキームでは反復途中の誤差のうち低周波成分の減衰が遅いことに起因していることがわかる。つまり、計算に入り込んだ誤差の低周波成分が急速に除去できれば、本反復法の収束性も急速に向上することを示唆している。

3.2 Multigrid法概要

反復計算における誤差の低周波成分を除去する手段がMultigrid法(6)(7)(8)(9)である。

Multigrid法の基本的考え方は、ある格子空間上での除去が困難となった誤差をさらに粗い格子空間上に持込んで除去するというものである。図1に示すように $N=8$ のgrid(Ω^h)における波数 $k=2$ の周波数成分を $N=4$ の2倍粗い

grid(Ω^{2h})に写像変換した場合を考える。同じ周波数成分でも各gridで表現できる波数の範囲の違いから粗いgridでは細かいgridよりも高周波成分として捉えることができる。これを利用すれば、反復計算により細かいgridで除去困難なモードは、粗いgridに写像することで高周波成分に変換され、粗いgridにおいて反復計算することで、このモードは簡単に除去できることを示している。

以上述べたことから、連立一次方程式

$$A \cdot U = f \quad (3-1)$$

をMultigrid法により解くスキームを簡単に述べる。本スキームは、coarse grid correction schemeと呼ばれ、後述するMultigrid V-cycle scheme, Full Multigrid schemeを始めとするMultigrid法の種々のスキームの基本となるものである。

coarse grid correction scheme

- ① $A_1 \cdot U_1 = f_1$ を細かいgrid Ω_1 で (ヤコビ法等を用いて) 緩和計算して近似解 U'_1 を求める。
- ② $r_1 = f_1 - A_1 \cdot U'_1$ (残差ベクトル) を求める。
- ③ r_1 を粗いgrid Ω_2 上での残差ベクトル r_2 に変換する。
- ④ $A_2 \cdot e_2 = r_2$ (残差方程式) を Ω_2 で緩和計算して誤差ベクトル e_2 の近似解を求める。
- ⑤ e_2 を Ω_1 上での e_1 に変換し、 $U_1 = U'_1 + e_1$ により近似解 U_1 を補正する。
- ⑥ U_1 を初期値として、①～⑤を収束するまで繰り返し計算する。

本スキームの要点をまとめると、図2にも示すように、①で除去できなかった誤差 (低周波成分) を②～④を通して粗いgridに変換し、粗いgridにおける残差方程式を解いて⑤⑥により近似解を補正する。

ここで、③④を計算する際の、粗いgridへの残差ベクトル及び係数行列の変換法、⑤における細かいgridへの誤差ベクトルの変換法を規定する必要がある。

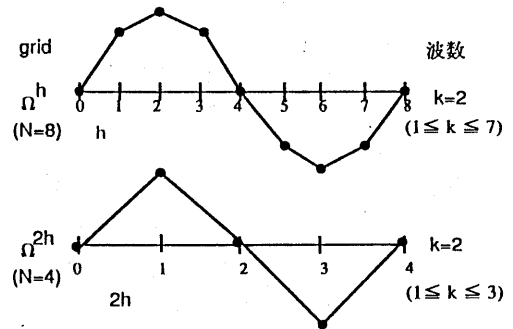


図1 粗密gridにおけるフーリエモード

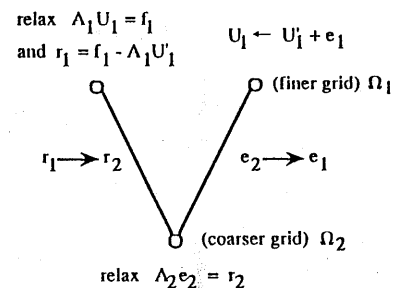


図2 coarse grid correction scheme

3.3 ベクトルおよびマトリクスの変換法

まずベクトルの変換方法について述べる。ベクトルの変換には、細かいgridから粗いgridへの変換とその逆の2ケースがあり、前者をRestriction、後者をInterpolationと呼ぶ。

Restriction としては加重平均方式が一般的であるが、周囲の参照点とその重みの取り方にはいくつかの方法がある。図3は2次元格子における加重平均方式を示したもので(a)は9点参照型、(b)は5点参照型である。また、解析領域の境界の処理にもいくつか方式があり、

- ④ 境界については、そのまま値を残す方式
- ⑤ 境界外の変数値は0として、機械的に図3に示すパターンを使う方式
- ⑥ 境界については重み付けのパターンを境界に合わせて変える方式

等が挙げられる。本報告では、単純かつ高精度な方法として、図3(a)の9点参照方式を使用

し、境界では⑥の方式で機械的に変換する方法を採用する。

Interpolationとしては線形内挿方式が一般的である。Restrictionの逆演算と考えると、図4に示すように、粗いgrid上の格子点が細かいgrid上の格子点でもある場合(①)は、そのままの値を用い、隣接する格子点のうち左右または上下のみが粗いgrid上の格子点の場合(②④)には、それら2点で補間し、他は4点で補間するという方法である。本報告でもこの線形内挿方式を採用する。

尚、Restriction operatorを R 、Interpolation operatorを I としたとき、

$$R = c \cdot (I)^T \quad (c \text{ は定数}) \quad (3-2)$$

となるように選ぶことが重要であるとの報告があり、本報告で採用する2次元gridのベクトル変換方式はこの関係を満足している。これについての詳細は文献(7)(8)を参照されたい。

次にマトリクスの変換方法について述べる。粗いgridへのマトリクスの変換については以下の2つの方法が提唱されている。

(1) A_2 は、粗いgridで、元の偏微分方程式を離散化し、作成する。(9)

(2) A_2 は、ベクトル変換時のoperatorを使い、

$$A_2 = R \cdot A_1 \cdot I \quad (\text{Galerkin approach})$$

により作成する方法する。

(1)の方式は単純な方式で、粗いgridで使用するマトリクスが独立に作成できるという利点を持つ。(2)の方式は、細かいgridのマトリクスから機械的に作成できるため、粗いgrid上での離散化処理が不要となる。しかし、作成されるマトリクスは元々5点差分が9点差分の形式になる等、解に及ぼす影響に理論的な不明点が多い。(これに関する良否は現時点では判断できず、引き続き検討していく。)

本報告では、その単純さの点で、(1)の方式を採用して評価する。

3.4 Multigrid法によるスキーム構成

以上述べたことを基本にして、Multigrid法

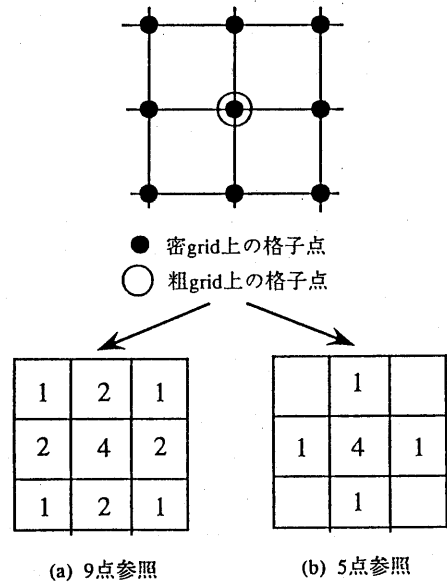


図3 2次元gridにおける加重平均

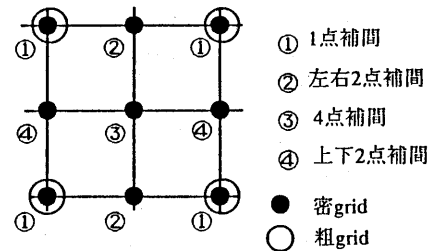


図4 2次元gridにおける線形内挿

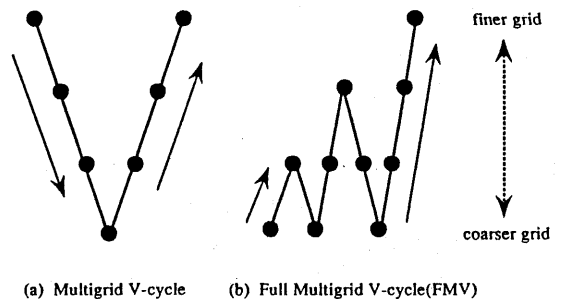


図5 スキーム構成

により実際の連立一次方程式を解くスキームを記述する。

3.2のcoarse grid correction scheme(C G C S)を多段に組み合わせて、種々のスキームが提案されているが、本報告では図5に示す代表的な2つのスキームについて評価を行う。(a)は、V-cycle スキームと呼び、C G C Sの最も単純な拡張である。(b)は、Full Multigrid V-cycle (F M V) スキームと呼び、V-cycleと演算量はほぼ同じであるが粗いgridでの計算を多く含んでいるのが特長であり、低周波成分の除去を頻繁に行う方法と捉えることができる。

4. A D I 法 A D I (Alternating Direction Implicit:交互方向陰解)法の概要について述べる。例として2次元ポアソン方程式

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = f \quad (4-1)$$

を考える。Uは未知変数、fはソース項である。加速項 $\omega(U^* - U^0)$ または $\omega(U - U^*)$ を導入し、

$$\text{solve } U^* \text{ of } \frac{\partial^2 U^*}{\partial x^2} + \frac{\partial^2 U^0}{\partial y^2} + \omega(U^* - U^0) = f \quad (4-2)$$

$$\text{solve } U \text{ of } \frac{\partial^2 U^*}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \omega(U - U^*) = f \quad (4-3)$$

を交互に繰り返して解いていく。加速因子 ω は計算が速く進むように選ぶが次のような点に注意しなければならない。

(1) 小さい ω であり、 $U^* \approx U^0$ であれば本来の方程式と等価となる。つまり大きな ω に対しては $U^* \approx U^0$ でも別の解に収束する可能性がある。

(2) $\omega = 0$ とすると各方向の反復後、解が初期値に戻ってしまう。さらに1次元化したために両端が2種境界条件のような問題では方程式が不定となる。

ω の与え方としては固定境界で囲まれたポアソン方程式に対して、以下の方法がある。

④ 一定のパラメータを使用する方法(10)

解析領域を $M \times N$ に等分割した場合、

$R = \max(M, N)$ として、

$$\omega = [16 \cdot \sin^2\{\pi/(2R)\} \cdot \cos^2\{\pi/(2R)\}]^{1/2}$$

とする。

⑤ 反復毎に最適な ω を与え周期的に使用する
方法(11)

$$\alpha = 4 \sin^2\{\pi/(2R)\}, \quad \beta = 4 \cos^2\{\pi/(2R)\},$$

$$q = \ln(\beta/\alpha)/(2 \ln(\gamma)), \quad \gamma = 1/(\sqrt{2} - 1),$$

$$p = (\beta/\alpha)^{1/(2q)} \text{ より、}$$

$$\omega_k = \alpha \cdot p^{2k-1} \quad (k \text{ は正整数で、} 1 \leq k \leq q)$$

となる数列 ω_k を反復毎に変えて、周期的に使用する。

これらの方法は、各方向求解時の係数行列

$$H = \frac{\partial^2}{\partial x^2}, \quad V = \frac{\partial^2}{\partial y^2} \quad (4-4)$$

に対して $HV = VH$ という「可換」の条件下で導出されたものであるが、「非可換」な場合にもその有効性が発揮されるとの報告がある。(11)

5. ポアソン方程式への適用と評価

5.1 評価問題と評価方法

図6に示す2次元熱拡散問題に対して、Multigrid法、A D I法を適用し、その収束特性、収束性能に関する評価を行う。数値実験の目的は以下の点について検討することにある。

① 収束特性

Multigrid法では残差ノルムによる収束状況を測定する。A D I法については解の変動量の追跡とI L U B C G法との比較を行う。

② 収束性能(高速性)

計算はスカラ計算機(HITAC M-680H, ~30MIPS)を使用し、収束までの計算時間を測定する。計算時間についてもI L U B C G法との比較を行う。(スカラ計算機上である程度的高速性が確認できれば、アルゴリズムに内在する並列性を考慮すると、並列化による高加速率が期待できる。)

5.2 Multigrid法による評価結果

本節では、図6の評価問題に対するMultigrid法の適用結果を述べる。評価項目としては、①緩和計算用SOLVER(ヤコビ、R/B SOR)、②拡散係数(問題依存性。均一拡散場と不均一拡散場(2桁違う階段型))、③サイクル方式(V, F M V)とし、収束性がどのような影響を受け

るかを測定する。

さらに評価に際しては以下を設定した。

(1) 最密gridを 33×33 として、以下 17×17 , 9×9 , 5×5 , 3×3 の4つの粗gridを使用する。(等間隔格子)

(2) 3×3 の粗gridでは、ガウス消去法により直接誤差ベクトルを求める。

(3) 残差ベクトル及び誤差ベクトルの変換は、3.3節に従い、9点参照加重平均方式とその逆演算型の線形内挿方式を使用する。マトリクスについては各gridで基本方程式を離散化したものを使用する。

(4) 緩和係数については、ヤコビ法は $\omega = 2/3$ のweighted-Jacobi法を使用し、Red/Black SOR法については、事前の数値実験により、最密gridでの最適値 $\omega = 1.9$ を使用し、粗いgridにおいてもこの値で代用する。

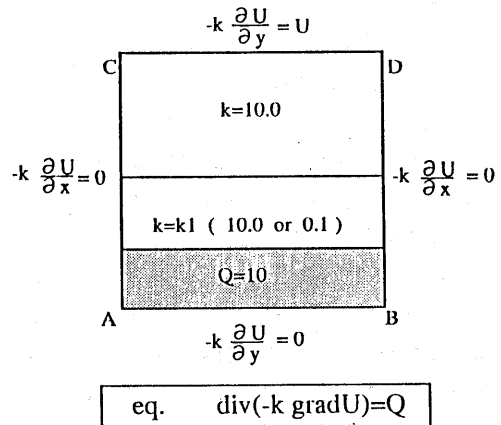


図6 評価問題

表2に評価結果を示す。数値実験は緩和solver、拡散場、サイクル方式に対して各gridでの緩和計算の反復回数(sweep回数。固定打ち)をパラメータにして、残差ノルム($r = \|f - AU\| / \|f\|$)が 10^{-5} 以下となるまでのMultigridサイクル回数、計算時間を測定した。また表中に比較実験としてMultigridを使用せずヤコビ法などの結果、ILUBCG法による計算時間を示した。以下考察する。

収束特性について

① 図7は均一拡散場問題の収束特性である。サイクル方式(V, FMV)と緩和solver(ヤコビ(Ja), R/B SOR(RB))の組合せで、各gridにおける反復回数に対して収束($<10^{-5}$)までに必要なサイクル回数を示している。ヤコビ法の場合にはFMVの方がVに比べ有効となっているが、各gridにおけるヤコビ法自体の収束性が悪いために、Multigridの効果が現れていない。R/B SORの場合は各gridでの収束性が改善されており、十分な緩和を行えば極めて少ないサイクル回数で解は収束する。さらにR/B SORではV-cycleの方が収束までのサイクル回数が少ない結果となった。

表2 Multigrid評価結果

評価項目				測定結果		
緩和 solver	拡散場	リクル法	sweep 回数	リクル数	残差ノルム (order)	CPU(sec) (M680H)
Jacobi	均一	V	2	600	-4	61.37
			30	86	-5	16.11
			50	58	-5	14.45
			100	34	-5	13.66
		FMV	2	365	-5	29.34
			30	65	-5	11.29
			50	48	-5	10.49
			100	33	-5	10.89
	Jacobi only	5000		-1	5.48	
	非均一	V	2	303	-5	31.06
			30	39	-5	7.36
			50	26	-5	6.49
			100	16	-5	6.45
		FMV	2	148	-5	16.46
			30	30	-5	5.27
RB-SOR	均一	V	30	6	-5	1.23
			50	4	-5	1.10
			100	3	-8	1.30
			2	54	-5	6.06
		FMV	30	10	-5	1.83
			50	7	-5	1.63
	非均一	V	100	4	-5	1.41
			2	21	-5	2.26
			30	2	-7	0.49
			50	1	-5	0.36
		FMV	100	1	-12	0.51
			2	35	-5	3.97
高速化	V	30	4	-7	0.80	
		50	2	-6	0.54	
		100	2	-13	0.76	
		70	1	-7	0.37	
ILUBCG					0.33	

② 図8は不均一拡散場の問題である。基本的な収束特性は図7と同様であるが、どのケースも均一場に比べ少ないサイクル回数で収束している。これは陽的な計算が局所的な情報伝達計算であるのに対し、粗いgridでの計算が大域的な伝達となることから判断でき、誤差を粗いgridへ持込むMultigridスキームの有力性が確認できる。

③ 図9は不均一拡散場の問題をR/B SORとV-cycleの組合せで解析したもので、各段の緩和の程度(回数)をふって(iter=30, 50, 100)、 10^{-5} 以下の残差ノルムに対しての収束特性を示している。ここでは各gridでの緩和を充分行うことにより1~2回と非常に少ないMultigrid回数で、 10^{-15} 程度の収束性まで保証できることが明らかとなった。デバイス解析のように高精度の収束性を必要とする分野でもMultigrid法が強力な解法となることが確認できる。

④ 逆に、各段の十分な緩和が本手法のキーで、充分に行わないとMultigridを何度繰り返しても、ある誤差レベルで飽和し、収束解に到らない。

収束性能について

⑤ 計算時間については表2からILUBCG法が0.33secであるのに対しヤコビ系は5~60secと、1~2桁程度遅い。R/B SOR系では計算時間も1/10~1/50に短縮され、ILUBCG法とほぼ同程度のオーダとなった。プログラムを一部修正して各gridのsweep回数をgridサイズに合わせて変化するようにして高速化を図ったところ、表中下2段目のように残差ノルム 10^{-7} 、計算時間0.37secとILUBCG法の性能と比較してほぼ同一となりMultigrid法が強力な解法であることが明らかとなった。

5.3 ADI法による評価結果

本節では、ADI法の適用結果について述べる。評価問題としては基本的には図6に示すモデルを使用するが、ここではさらに移流項を加えて、

$$\frac{\partial}{\partial x}(-K \frac{\partial U}{\partial x}) + \frac{\partial}{\partial y}(-K \frac{\partial U}{\partial y}) + V \cdot U = Q \quad (5-1)$$

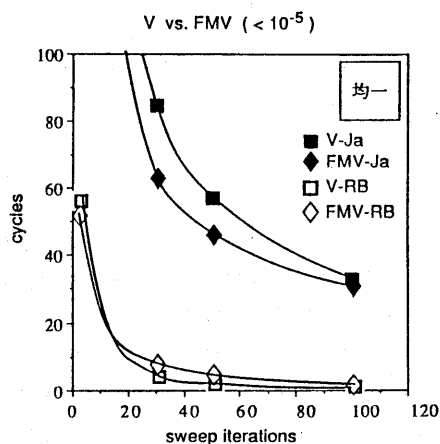


図7 収束特性1

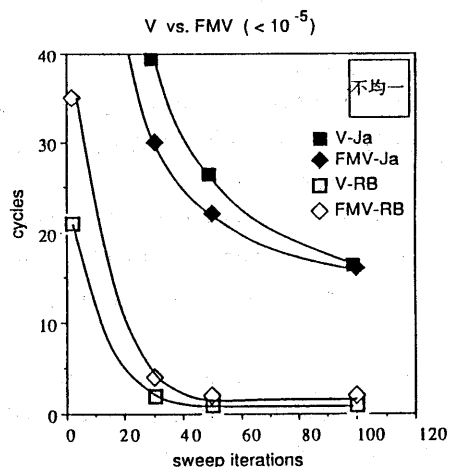


図8 収束特性2

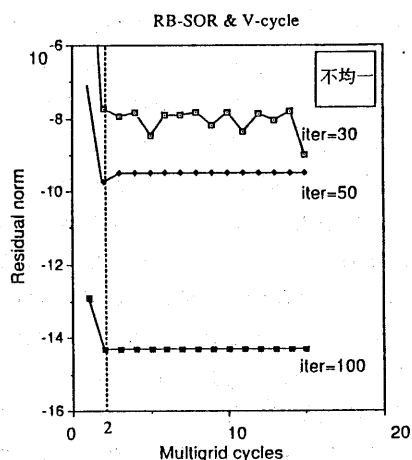


図9 収束特性3

$$V = \begin{cases} V_0 & \text{at 領域上半分} \\ 0 & \text{at 上以外} \end{cases} \quad (5-2)$$

として解析した。またそれに合わせて領域左右の境界条件を、

$$-K \frac{\partial U}{\partial x} + V \cdot U = 0 \quad (5-3)$$

とした。移流項が入ることにより収束性が悪くなることも予想され、ADI法では拡散係数の均一／不均一性と移流項に対する評価を行う。

表3に評価項目一覧を示す。拡散場については図6に従い、移流項については $V_0=0.1, 1.0$ の2種類のパラメータで4つの実験をする。

移流項の強さを示す量としてセルバクレ数 $\lambda = V \cdot dx / (2 \cdot K)$ を導入すると、 33^2 格子点で $\lambda=0.01, 0.1$ の解析となる。各実験に対して以下の2点を主に検討する。

(1) 収束特性

第4章で述べた2つの加速因子（一定値、周期的数列）に対する収束状況を評価する。

(2) 収束性能（高速性）

収束までの計算時間を測定する。計算結果については、ILUBCG法による解（真値と推定できる）と比較して収束の判定をする。

図10～13に、収束特性のグラフを示す（解析メッシュは 33×33 ）。各グラフは、反復による解の前の解との変動量を示している。横軸は、反復回数（ x, y 両方向で全格子点を更新して1回）、縦軸は変動量（1ステップ前の解との差の2乗ノルム）である。変動量では解が飽和することもあると考えられるので、図中に収束ラインとしてILUBCGによる解との差の最大値ノルムの推移を示した。結果をまとめる。

図10（実験1）は、均一拡散場のモデルである。加速因子として一定の ω を用いた場合は、反復回数70回でも正解が得られず、収束しなかった。周期的 ω の場合、収束性は向上し、反復回数37回で変動量 10^{-5} 、ILUBCGによる解との最大値ノルムを計算すると 10^{-5} に収束する。収束状況を見ると、下に凸のピークは加速因子が最大になっている時で、変動量は小さいが過大なソース項が入った効果により本来の方

表3 ADI法評価項目一覧

NO.	問題種別	拡散場	移流項(V_0)
1	拡散	均一	0.0
2	拡散	不均一	0.0
3	移流拡散	均一	0.1
4	移流拡散	均一	1.0

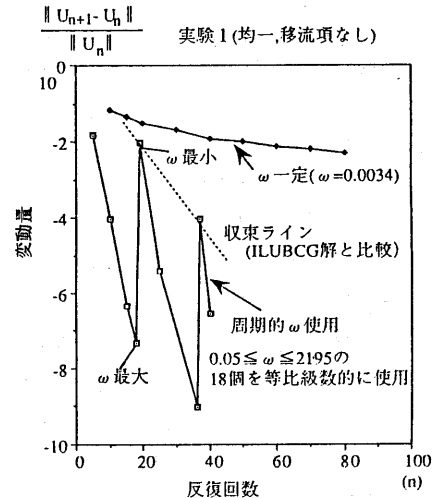


図10 ADI 収束特性 1

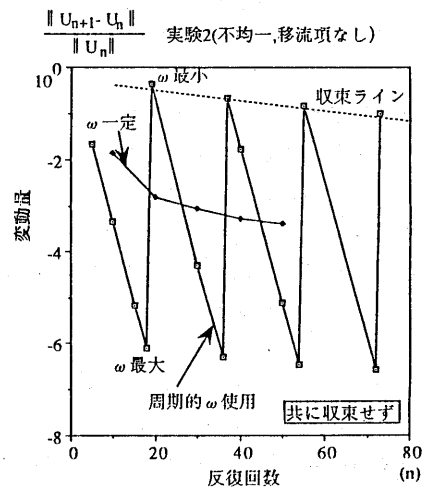


図11 ADI 収束特性 2

程式を満足していない可能性が生じる。グラフの上に凸のピークは加速因子が最小になっているところで、ADIの解はほぼこのピークを結んだ線に沿って収束していくことが判明した。本実験によって、加速因子として周期的パラメータ系列を使う有効性が確認できた。

図11(実験2)は、不均一な拡散場のモデルである。本実験では、2種類の方法とも収束しなかった。

図12(実験3)は、均一な拡散場で、領域上半分に移流項を付加したモデルである(セルバクレ数 $\lambda=0.01$)。加速因子として周期的パラメータ系列を使用することで、反復回数55回で、ILUBCGの解に対して 10^{-6} に収束した。

図13(実験4)は、実験3よりさらに移流項を10倍($\lambda=0.1$)大きくしたモデルである。周期的パラメータ系列により、反復回数55回で、ILUBCGの解に対して 10^{-6} に収束した。

実験3,4に使用した加速因子はADIの定式化で得られるx,y各方向の係数行列が「可換」という条件の下に導出したものである。しかし、実験3,4のような移流項を含んだ「非可換」なモデルに対しても、本加速因子の有効性が確認できた。

表4に各実験に対する収束回数、計算時間の結果を示す。実験は、解析メッシュ数を 33×33 、 65×65 とした2つのモデルに対して測定した。また、先の収束特性で得られた知見により、全て、周期的加速因子を用いた。

結果に対する考察は以下である。

- (1) ADIは、今回の実験において拡散係数の均一なモデル(移流項の有無に関わらず)で、有力性が示される。
- (2) 不均一な拡散係数を与えた問題では、収束性が悪く、正解を得られなかった。本評価問題では、加速因子 $\omega=0$ とするとx方向求解時に方程式が不定となり、さらに変動量が大きいと、 ω が小さくても加速項によるソース項的效果が大きく正解への収束が遅い。Multigrid法の有力性と同様の考えから、ADIスキームの初期値をある程度粗いgridで解いた値で与えることで、改善

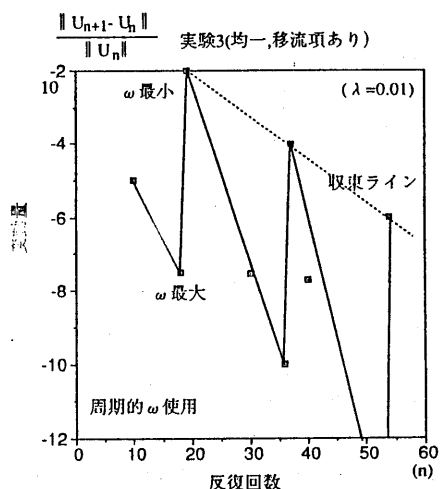


図12 ADI 収束特性 3

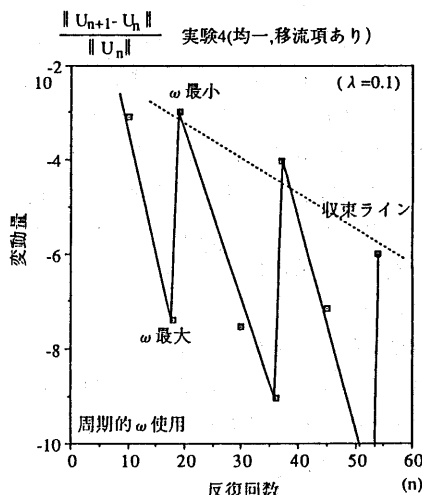


図13 ADI 収束特性 4

表4 ADI 性能結果

解析メッシュ	実験NO.	収束回数(回)	全計算時間(sec)	ILUBCG法計算時間(sec)	備考
33	1	37	1.12	0.33	
	2	収束せず	—	0.33	拡散係数階段型
	3	55	1.87	0.33	移流項(0.1)
	4	55	1.87	0.50	移流項(1.0)
65	1	61	10.90	1.93	
	2	収束せず	—	1.93	拡散係数階段型
	3	61	10.90	1.93	移流項(0.1)
	4	81	14.48	2.17	移流項(1.0)

(注)①表中の実験NO.は表3の実験NO.に対応する。

②全て周期的加速因子を使用。

③計算時間はHITAC-M680H(〜30MIPS)によるCPU時間である。

④収束判定はILUBCGによる解に対して 10^{-6} 以下をもって収束とする。

⑤ILUBCGの収束判定は 10^{-6} とする。

されるのではないかと考えられる。

- (3) 計算時間をILUBCG法と比較すると、ADIは3～6倍程度遅い。この点でADIの性能は劣っているが、移流項に対して有力であるとの結果と、並列計算において、x, y各方向について、独立に複数本の3項方程式が解けるという粒度の大きい並列性を有する点で、並列化による高速化が期待できる。

6. おわりに

超並列計算機向き行列解法アルゴリズムとして、Multigrid法、ADI法を取り上げた。各解法を2次元ポアソン方程式へ適用し収束性に関する数値実験を行った。

結果を以下にまとめる。

- (1) Multigrid法は、Red/Black SORと組合せて使用することで、スカラ計算機上でも(演算量として)ILUBCG法と同程度の性能を得ることができ、有力な解法であることを確認した。
- (2) ADI法は、拡散係数が不連続に変化する問題では収束性が著しく劣下し、正解が得られなかったが、移流拡散方程式に対して、有力性が示された。

各解法の持つ並列性を考えると、Multigrid法は格子点単位、ADI法は格子線単位の並列性があることから超並列計算機における高速計算が期待できる。

謝辞

本稿は、日立製作所における「数値シミュレーション用高水準言語DEQ SOLの研究」の中で、同中央研究所より当社(日立超LSIエンジニアリング)へ委託された内容に関するものである。日立製作所中央研究所の梅谷征雄主任研究員には本検討期間を通して御指導頂き、また同ソフトウェア工場の後保範主任技師には有益な御討論と御助言を頂いた。深謝したい。

参考文献

- [1] 星野 力:「PAX コンピュータ」:オーム社(昭和60-3)
- [2] 谷川 他:「並列計算機ADENA」:情報処理学会 計算機7-技術分科会(1988.7.7)
- [3] Gerard MEURANT:「Multitasking the Conjugate Gradient Method on the CRAY X-MP/48」:Parallel Computing, 1987, Vol.5
- [4] P.S.Tseng:「Iterative Sparse Linear System Solvers on Warp」:ICPP, '88, Vol.1.
- [5] Gregory A.Darmohray:「Gaussian Techniques on Shared Memory Multiprocessor Computers」:SIAM, 1987.12
- [6] J.LINDEN:「Parallel multigrid solution of the Navier-Stokes eq.s on general 2D domains」:Parallel Computing, 1988, Vol.7(p.461)
- [7] William L.Briggs:「A Multigrid Tutorial」:SIAM, Philadelphia, Pennsylvania 1987
- [8] Wolfgang Hackbusch:「Multi-Grid Methods and Applications」:Springer-Verlag, 1985
- [9] Achi Brandt:「Multi-Level Adaptive Solutions to Boundary-Value Problems」:Math. of Computation, 1977, Vol31, NO.138(p.333-390)
- [10] Gordon.D.Smith:「Numerical Solution of Partial Differential Equation」:Oxford University Press, 1965
- [11] R.S.Varga:「Matrix Iterative Analysis」:Prentice-Hall, 1962