

境界適合法による流体解析への 非対称系 PCG 法の応用

秋葉 幸範 , 土肥 俊
日本電気(株) C & C 情報研究所

2次元境界適合法で生じる非対称連立一次方程式の解法として、スーパーコンピュータ向きの前処理反復法(PCG法)を開発した。2次元境界適合法ではポアソン方程式(Navier-Stokes方程式)は大規模で疎な9点(13点)行列となる。この行列の主要5点要素のみで前処理を行えば、ベクトル化に適した三重対角分解前処理法(TF法)が使用できる。当社のスーパーコンピュータSX-2を用いた数値実験により従来のSOR法とPCG法の計算時間・収束性の比較を行なった結果、PCG法の高収束性が確認された。TF法はILU法に比べ反復数が40%程多いが、CPU時間は短く、BCG法はCGS法に比べ反復数・CPU時間ともに30%程度増加することがわかった。

本報告ではまた、線形解法の繰り返しによる楕円型格子生成アルゴリズム、および流体解析法であるMAC法・SMAC法の陰解法化についても提案する。

A PRECONDITIONED CONJUGATE GRADIENT TYPE METHOD SUITED TO FLUID FLOW ANALYSIS ON BOUNDARY-FITTED COORDINATE SYSTEM

Yukinori AKIBA and Shun DOI

C&C Information Technology Research Laboratories
NEC Corporation
4-1-1, Miyazaki, Miyamae-ku, Kawasaki 213, JAPAN

We propose a preconditioned conjugate gradient type method suited to supercomputers for solving nonsymmetric linear systems which arise in the 2D Boundary-Fitted Coordinate analysis. The large sparse 9-point and 13-point FDM coefficient matrices arise from Poisson and Navier-Stokes equations on 2D BFC. By using only the 5-point FDM matrix elements as the preconditioner, these matrices can be efficiently solved by the Tridiagonal Factorization (TF) algorithm, which can be easily vectorized.

Numerical experiments in fluid flow analysis were done on the NEC SX-2 supercomputer in order to compare convergence and computation time of various PCG algorithms with the conventional SOR method. Results show that PCGs show better convergence compared to SOR. The TFCGS/TFBCG require 40% more iterations than ILUs, but are faster in cpu time. The BCGs are about 30% slower in cpu time and convergence compared to CGSs.

In this paper, we also propose to solve linear systems successively in order to generate elliptic grids, and implicit MAC & SMAC schemes.

1. はじめに

ベクトル演算型スーパーコンピュータに代表される高速・高性能な計算機を効果的に利用するためには、ハードウェア性能を十分に引き出すための数値計算アルゴリズムと、膨大なデータを容易に取り扱うための入出力技術の研究開発が重要である。

近年スーパーコンピュータ向きの数値解法として差分法が見直されており、任意形状領域を矩形領域に写像して差分法を適用する境界適合法が注目されている。境界適合法は特に計算流体力学の分野で発展してきた手法であるが、電磁界やデバイスシミュレーションなどにも応用分野が広げられ、今後の発展が期待されている¹⁾。

計算流体力学はスーパーコンピュータの恩恵を最も受けている分野のひとつであろう。高精度なシミュレーションにより、実験では得られない微細な現象をとらえることが可能になってきた。その計算時間の大部分は連立一次方程式の求解に費やされている。例えば2次元圧力ポアソン方程式を古典的な差分法で離散化すると、規則・疎で対称な5点差分行列(=連立一次方程式)の解法に帰着される。ベクトル化に適した対称行列の解法としてICCG法やSCG法などが開発され成果をあげている²⁾。ところが境界適合法でポアソン問題を離散化すると非対称な9点差分行列を生じるため、従来この行列はSOR法やSLOR法などの緩和反復法で解かれていた。これらの緩和反復法は収束性やベクトル化の点で効率的とは言えない。そこで筆者らは、境界適合法で生じる非対称差分行列のスーパーコンピュータ向き求解法として、主要5点差分要素のみを前処理に使用する前処理反復法を開発した³⁾。

本報告では2次元境界適合法による非圧縮粘性流体の数値シミュレーションにおいて、第2章では楕円型方程式による格子生成のアルゴリズムと前処理反復法の応用について説明する。第3章では差分スキームでのMAC法・SMAC法の陰解法化を提案する。第4章では連立一次方程式の求解法について、従来法の問題点を明らかにし、主要5点前処理反復法のアルゴリズムを述べる。第5章では本手法と従来法の収束性の比較を行ない、本手法の有効性を確認する。

2. 境界適合法の格子生成法

2.1 格子生成のための基礎方程式⁴⁾

差分法では解析領域に計算のための格子を生成する必要がある。領域形状が複雑な場合、従来の差分法では精度よく近似できないという問題点があった。そこでポアソン方程式を使用して物体に沿った滑らかな格子を生成し、格子に沿った一般曲線座標上で差分法を適用するのが境界適合法である(図

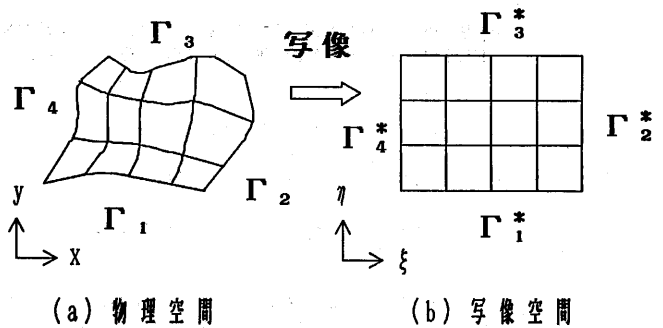


図1 境界適合法

1).

$$\xi_{xx} + \xi_{yy} = P(\xi, \eta), \quad \eta_{xx} + \eta_{yy} = Q(\xi, \eta) \quad (2.1)$$

ここで右辺の外力項 P, Q は格子の分布を制御するための制御関数である。 (x, y) は物理座標系の座標値を表わし、 (ξ, η) は格子に沿った一般座標系の座標値を表わす。物理空間上で図1aのように適当な境界条件を与え、上式を解き、 $\xi, \eta =$ 一定の線を引けば格子が生成できたことになる。

しかし、実際には (ξ, η) に対応する (x, y) 座標値を求めたいので、式(2.1)を物理空間上で解くのではなく、物理座標系(図1a)を (ξ, η) に沿った正規直交格子(図1b)に写像し、その写像空間上で計算をする。その写像変換の行列は次式で与えられる。

$$\text{座標変換: } \begin{pmatrix} \xi \\ \eta \end{pmatrix} = \begin{pmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{J} \begin{pmatrix} y_\eta & -x_\eta \\ -y_\xi & x_\xi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (2.2a)$$

$$J = x_{\xi} y_{\eta} - x_{\eta} y_{\xi} \quad (2.2b)$$

従って、式(2.2)は次の準線形楕円型方程式に変換される。

$$g_{22}(x_{\xi\xi} + P x_{\xi}) + g_{22}(x_{\eta\eta} + Q x_{\eta}) - 2g_{12}x_{\xi\eta} = 0 \quad (2.3a)$$

$$g_{22}(y_{\xi\xi} + P y_{\xi}) + g_{22}(y_{\eta\eta} + Q y_{\eta}) - 2g_{12}y_{\xi\eta} = 0 \quad (2.3b)$$

$$g_{11} = x_{\xi}^2 + y_{\xi}^2, \quad g_{12} = x_{\xi} x_{\eta} + y_{\xi} y_{\eta} \quad (2.4)$$

$$g_{22} = x_{\eta}^2 + y_{\eta}^2, \quad g = J^2 = (x_{\xi} y_{\eta} - x_{\eta} y_{\xi})^2$$

写像は1対1でなければならず、さもなければヤコビアン $J = 0$ の特異性が生じる。一般曲線座標では基底ベクトルは局所的であるので、離散化された各格子点において変換の行列が定義される。ここで制御関数にはつぎの P', Q' を使用し、

$$P' = g_{22} \cdot P / g, \quad Q' = g_{11} \cdot Q / g \quad (2.5)$$

P, Q には指数関数を用いた式を使用した⁴⁾。

$$P = -\sum_{i=1}^n a_i \cdot \text{sgn}(\xi - \xi_i) \cdot \exp(-c_i |\xi - \xi_i|)$$

$$Q = -\sum_{j=1}^m b_j \cdot \text{sgn}(\xi - \xi_j) \cdot \exp(-d_j \cdot ((\xi - \xi_j)^2 + (\eta - \eta_j)^2)^{1/2}) \quad (2.6)$$

式(2.6)において右辺第1項は $\xi = \xi_i$ の線に、第2項は (ξ_j, η_j) の点に格子を引き寄せさせる作用がある。係数 a_i, b_j によって格子を引き寄せさせる強さを制御し、 c_i, d_j によって影響が及ぶ範囲を制御する。 Q についても同様である。

2.2 離散化と計算アルゴリズム³⁾

式(2.3)を解くために差分法を用いる。写像空間の正規直交格子上においては空間刻み幅を $\Delta \xi = \Delta \eta = 1$ とすればよく、微分演算は例えば次のような差分で近似される。

$$f_{\xi} = (f_{i+1,j} - f_{i-1,j}) / 2, \quad f_{\xi\xi} = f_{i+1,j} - f_{i,j} + f_{i-1,j}$$

$$f_{\xi\eta} = (f_{i-1,j-1} - f_{i+1,j-1} - f_{i-1,j+1} + f_{i+1,j+1}) / 4 \quad (2.7)$$

上式を式(2.3)に適用すると非対称・非線形な9点差分方程式が得られる(図2)。従来この方程式の求解にはSOR的緩和反復法が用いられていた。しかしこの方法はベクトル化や収束性の点で有効な解法とは言えない。

そこで筆者らは線形解法の繰り返しによる求解法を提案した³⁾。すなわち式(2.3)の係数を固定して線形化すると連立一次方程式が得られるので、

- ① 初期格子を代数的格子生成法で求める。
- ② 係数を計算する。
- ③ 線形化された方程式を解く。
- ④ 収束条件を満たすまで②-③を繰り返す。

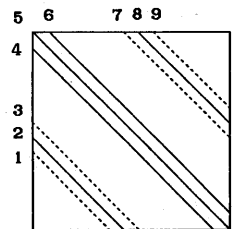
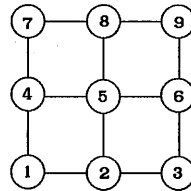


図2 9点差分スキームと行列構造

というアルゴリズムで解を求める。この方法は、ベクトル化に適した行列解法が使用でき、また一外部反復内で変数 (x, y) に関する係数行列は同一であり、それぞれ独立に計算が可能であるなどの特長が得られる。また非常に鋭い角で生じる格子のオーバーラップの問題⁴⁾も、本手法によって解決される。連立一次方程式解法については章4で説明する。

3. 流体の数値解析法

3.1 2次元非圧縮粘性流体の支配方程式⁵⁾

$V(u, v)$ を流体の速度, p を圧力, Re をレイノルズ数とすると, 物理変数を用いた非圧縮粘性流の基本方程式は次のとおりである.

(1) Navier-Stokes方程式 (運動量保存式)

$$\mathbf{V}_t + (\mathbf{V} \cdot \nabla) \mathbf{V} = -\nabla p + \nabla^2 \mathbf{V} / Re \quad (3.1)$$

(2) 連続の式 (質量保存式)

$$\nabla \cdot \mathbf{V} = 0 \quad (3.2)$$

これらの方程式も写像空間上で解かれるので, 格子生成と同様に座標変換される.

(1) Navier-Stokes方程式

$$u_t + (y_\eta u - x_\eta v) \cdot u_\xi / J + (-y_\xi u + x_\xi v) \cdot u_\eta / J = -(y_\eta p_\xi - y_\xi p_\eta) / J + [g_{22}(u_\xi\xi + P u_\xi) + g_{11}(u_\eta\eta + Q u_\eta) - 2g_{12} \cdot u_{\xi\eta}] / (J^2 \cdot Re) \quad (3.3a)$$

$$v_t + (y_\eta v - x_\eta v) \cdot v_\xi / J + (-y_\xi v + x_\xi v) \cdot v_\eta / J = -(x_\eta p_\xi + x_\xi p_\eta) / J + [g_{22}(v_\xi\xi + P v_\xi) + g_{11}(v_\eta\eta + Q v_\eta) - 2g_{12} \cdot v_{\xi\eta}] / (J^2 \cdot Re) \quad (3.3b)$$

(2) 連続の式

$$(y_\eta u_\xi - y_\xi u_\eta - x_\eta v_\xi + x_\xi v_\eta) / J = 0 \quad (3.4)$$

3.2 数値計算アルゴリズムと差分スキーム⁵⁾⁶⁾

非圧縮流体解析の難しい点は, 各時刻で連続の式を満足しつつ Navier-Stokes方程式を解かなければならないことである. そのため現在までに様々なスキームが提案されている.

筆者らは(a)時間積分には数値的安定性のよい陰解法を使用し, (b)移流項の離散化には数値拡散の小さな風上差分法を使用した方が好ましいと考える.

MAC法・SMAC法は陽解法とされているが, 簡単に陰解法にすることができる. またFractional Step(F.S.)法とSMAC法は等価なスキームであり, F.S.法に圧力の増分値を導入するとSMAC法になる. 陰解法化したMAC法・SMAC法をそれぞれIMAC(Implicit MAC)法・ISMAC(Implicit SMAC)法と呼ぶことにする. 移流項の近似には, 数値的安定性を保ちつつ数値粘性の影響が小さな3次精度風上差分法(K-Kスキーム)を使用する. 以下に3種類の陰解法と3次精度風上差分法のスキームを記す.

(1) IMAC法

$$\begin{cases} \nabla^2 p^{n+1} = \frac{\nabla \cdot \mathbf{V}^n}{\Delta t} \\ \frac{\mathbf{V}^{n+1} - \mathbf{V}^n}{\Delta t} + (\mathbf{V}^{n+1} \cdot \nabla) \mathbf{V}^{n+1} = -\nabla p^{n+1} + \nabla^2 \mathbf{V}^{n+1} / Re \end{cases} \quad (3.5)$$

(2) ISMAC法

$$\begin{cases} \frac{\mathbf{V}^* - \mathbf{V}^n}{\Delta t} = -\nabla p^n \\ \nabla^2 p^* = \frac{\nabla \cdot \mathbf{V}^*}{\Delta t}, \quad p^{n+1} = p^n + p^* \\ \frac{\mathbf{V}^{n+1} - \mathbf{V}^n}{\Delta t} + (\mathbf{V}^{n+1} \cdot \nabla) \mathbf{V}^{n+1} = -\nabla p^{n+1} + \nabla^2 \mathbf{V}^{n+1} / Re \end{cases} \quad (3.6)$$

(3) Crank-Nicholson法

$$\begin{cases} \frac{\mathbf{V}^* - \mathbf{V}^n}{\Delta t} = [- (\mathbf{V}^n \cdot \nabla) \mathbf{V}^n - \nabla p^n + \nabla^2 \mathbf{V}^n / Re] / 2 \\ \nabla^2 p^{n+1} = 2 \frac{\nabla \cdot \mathbf{V}^*}{\Delta t} \\ \frac{\mathbf{V}^{n+1} - \mathbf{V}^*}{\Delta t} = [- (\mathbf{V}^{n+1} \cdot \nabla) \mathbf{V}^{n+1} - \nabla p^{n+1} + \nabla^2 \mathbf{V}^{n+1} / Re] / 2 \end{cases} \quad (3.7)$$

時間に関してIMAC法・ISMAC法は1次の精度であり, Crank-Nicholson(C-N)法は2次の精度である.

・ 3 次精度風上差分⁶⁾と移流項の線形化

$$(\mathbf{v} \cdot \mathbf{u}_x)_i = v_i(-u_{i+2} + 8u_{i+1} - 8u_{i-1} + u_{i-2})/12 \\ + |v_i|(u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2})/4 \quad (3.8)$$

$$(\mathbf{V}^{n+1} \cdot \nabla) \mathbf{V}^{n+1} = (\tilde{\mathbf{V}}^{n+1} \cdot \nabla) \mathbf{V}^{n+1}, \quad \tilde{\mathbf{V}}^{n+1} = 2\mathbf{V}^n - \mathbf{V}^{n-1} \quad (3.9)$$

4. 連立一次方程式解法

4.1 従来法の問題点と主要5点前処理法の提案³⁾⁷⁾

格子は、全ての変数が同じ点で定義される"Reguler Mesh"を使用する。座標変換された支配方程式を差分法により離散化すると、Navier-Stokes方程式は13点、圧力のポアソン方程式は9点の規則・疎な非対称連立一次方程式となる(図2)。従来これらの方程式の求解法に使用されていたSOR法やSLOR法などの緩和反復法には次のような問題点がある。

- ① SOR法は収束性がわるく、SLOR法はベクトル化が難しい。
- ② 問題により最適な緩和係数が異なる。
- ③ スイープの方向により収束性が異なる。特に圧力のポアソン方程式の境界条件の大部分はノイマン条件であるので注意が必要である。

「解析計算に要するCPU時間の大部分が連立一次方程式の求解に費やされる」ことを考えると、より高速な求解法の開発が必要である。そこで最近スーパーコンピュータ向けの解法として、前処理付き反復法が注目されている。いま連立一次方程式が $Au = b$ で表わされているとする。前処理反復法とは、Aの近似行列Mの逆行列を両辺にかけ、係数行列を単位行列に近い方程式に変換し、この方程式に反復法を適用し解を求める手法である。

$$Au = b \rightarrow M^{-1}Au = M^{-1}b, \quad M^{-1}A \sim I \quad (4.1)$$

非対称系の前処理法には、不完全LU分解法(ILU法)、筆者らの開発した三重対角分解法(TP法)⁸⁾などが、基礎反復法には、双対共役勾配法(BCG法)、2乗共役勾配法(CGS法)、共役残差法(CR法)などがある。前処理行列Mの性質として、(a)係数行列Aに対する近似度が高く、(b)逆行列計算が容易で演算の並列性があることが望ましい。

いま $n_x \times n_y$ 元9点連立一次方程式について考えよう。係数行列Aの第i行の非零要素を並べて次のように表記する。

$$A[i] = [b^-, b_i, b^+, c_i, d_i, e_i, f^-, f_i, f^+] \quad (4.2) \\ \text{列} = i-n_x-1 \quad i-n_x \quad i-n_x+1 \quad i-1 \quad i \quad i+1 \quad i+n_x-1 \quad i+n_x \quad i+n_x+1$$

この行列の非零要素すべてを反復法の前処理に使用せず、行列Aの非零要素が

- ① 直交性のよい格子では斜め方向の要素が対角項に比べて小さい ($d \gg b^-, b^+, f^-, f^+$)
- ② なめらかな解では前記係数が相殺し合う ($b^- = f^+, b^+ = f^-, b^- = -b^+, f^- = -f^+$)

ことに着目し、「前処理行列の計算に主要5点差分要素のみを使用する方法」を開発した。

$$A \rightarrow A_5[i] = [b_i, c_i, d_i, e_i, f_i] \rightarrow M \quad (4.3) \\ \text{列} = i-n_x \quad i-1 \quad i \quad i+1 \quad i+n_x$$

前処理行列に関する計算には5点差分のアルゴリズムを使用し、係数行列に関する計算(例えば行列ベクトル積)には9点(13点)差分要素すべてを使用すると、ベクトル化やメモリ容量の点で非常に効率よく計算できる。5点差分行列の前処理には以下のふたつの方法を用いた。

4.2 三重対角分解前処理法(TP法)⁸⁾

筆者らの開発した三重対角分解法はベクトル化に適した前処理法である。5点差分の係数行列Aを、対角行列D・x方向の微分に対応する非対角要素行列 A_x ・y方向の微分に対応する非対角要素行列 A_y に分割すると、

$$A = D + A_x + A_y \quad (4.4)$$

三重対角分解による前処理行列 M_{TF} を次式で定義する。

$$M_{TF} = (D + A_x) D^{-1} (D + A_y) \quad (4.5)$$

この前処理行列 M_{TF} は各因子毎に容易にLU分解でき、計算は軸方向の並列性を持つためベクトル化に適している。

4.3 不完全LU分解前処理法 (ILU法)

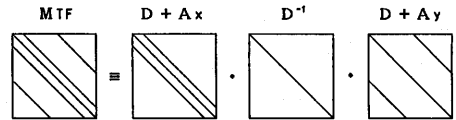
5点差分行列 A の非零要素に対してのみLU分解を行ない、fill-inを無視すると、不完全LU分解前処理行列 M_{ILU} が得られる。

$$M_{ILU} = L D U \quad (4.6)$$

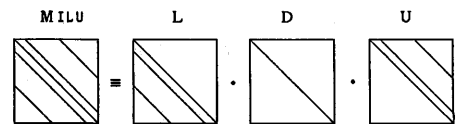
$L \cdot U$ はそれぞれ A の下三角・上三角と同じ構造を持つ下三角行列・上三角行列である。ILU法では、 x 方向の格子数を n_x とするとメモリアクセス間隔は $n_x - 1$ になり、計算は格子に対して斜め方向の並列性を持つ。

格子サイズを $n \times n$ とすると、ILU法の平均ベクトル長が約 $n/2$ であるのに対してTF法のベクトル長は n となる。また9点前処理ILU法およびSOR法のメモリアクセス間隔は $n_x - 2$ になり、平均ベクトル長はさらに短くなる(図3)。

・三重対角分解法 (TF法)



・不完全LU分解法 (ILU法)



メモリアクセスの並列性

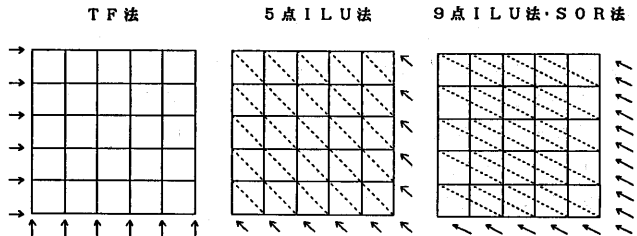


図3 前処理法と並列性

5. 数値実験による解法の比較

当社のスーパーコンピュータ SX-2 を使用して、自動車まわりの流れのシミュレーションにより、従来法と各種前処理反復法の収束性の比較計算を行なった。計算条件は、メッシュ数 = 303×101 , $Re = 1.85E+6$, $\Delta t = 0.05$ で、境界条件は図4のとおりである。スキームにはIMAC法を使用し

た。600ステップ後の1ステップにおける圧力のポアソン方程式について、各種PCG法の反復数とCPU時間の関係を表1に、CPU時間に対する相対残差 $L2ノルム_r$ の収束の様子を図5に示す(収束条件 $r < 1.E-8$)。ここで $MILU \cdot MTF$ の M はGustafsson修正 ($MILU = 0.95, MTF = 0.8$) を表わす。比較計算の結果より、以下のことがわかった。

- (1)SOR法は反復数1000回で収束しなかった ($\omega = 1.8$, 残差 $r = 5.79E-3$)。
- (2)BCG法はCGS法に比べて20~35%程度反復数・CPU時間が多い。
- (3)TF法はILU法に比べ30~45%反復数が多いが、CPU時間はやや短い。
- (4)Gustafsson修正の効果は殆どなく、かえって収束性が悪化する。

圧力ポアソン方程式の境界条件の大部分がノイマン条件であるため、係数行列はill conditionedとなる。そのためSOR法は非常に収束性がわるく実用的ではない。ILU法に比べTF法の反復数が多いのは、前処理行列の近似度が低いと思われる。しかしベクトル化の効果でCPU時間はほぼ同じになっている。収束に際してCGS法は振れ幅が大きい、BCG法よりも早く収束する。

6. むすび

2次元境界適合法で生じる非対称連立一次方程式のスーパーコンピュータ向き解法とし

て主要5点前処理反復法を開発し、流体の数値シミュレーション実験により高速性・高収束性を確認した。また本手法を応用した楕円型格子生成のアルゴリズムを開発し、2次元非圧縮粘性流体の数値解析法においてMAC法・SMAC法の陰解法化を提案した。

3次元境界適合法の格子生成等では非対称な19点差分行列が生じる。19点すべてを前処理に用いるのは、ベクトル化や処理の複雑さなどを考慮するときわめて困難である。この方程式の解法にも、2次元と同様に主要7点要素を前処理に使用する手法が適用できる。また3次元問題での不完全LU分解法の処理にはリストベクトルを使用しなければならないため、座標軸方向に並列化が可能な三重対角分解法⁸⁾がさらに有効な手法になると考えられる。

文 献

- 1)小瀬,高木,三木,「プラズマイオン源ビーム引出し系の三次元イオン軌道解析手法」, 電学論A, 107巻7号, pp.323-330 昭62.
- 2)Hayami,K. and Harada,N, "The Scaled Conjugate Gradient Method and Vector Processors", Proc. of 1st Int. Conf. on Supercomputing System, pp.213-221 (1985).
- 3)秋葉, 土肥, 原田,「因子分解による前処理付き反復法の境界適合法への応用」, 情処第36回全国大会, Vol.1, pp.25-26 (1988).
- 4)Thompson,J.F., Warsi,Z.U.A. and Mastin,C.W., "Numerical Grid Generation; Foundations and Applications", North-Holland (1985).
- 5)Peyret,R and Taylor,T.D., "Computational Methods for Fluid Flow", Springer-Verlag, (1983).
- 6)Kawamura,T. and Kuwahara,K., "Computation of High Reynolds Number Flow around a Circular Cylinder with Surface Roughness", AIAA paper, 84-0340 (1984).
- 7)戸川隼人,「マトリクスの数値計算法」, オーム社 (1971).
- 8)Doi,S. and Harada,N. "Tridiagonal Factorization Algorithm: A Preconditioner for Nonsymmetric System Solving on Vectorcomputers", J. of Info. Processing, 11. 1. pp.38-46 (1988).

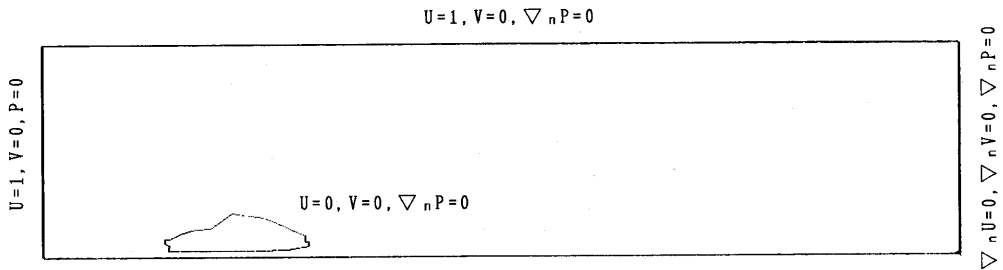


図4a 解析領域と境界条件 $U=1, V=0, \nabla_n P=0$

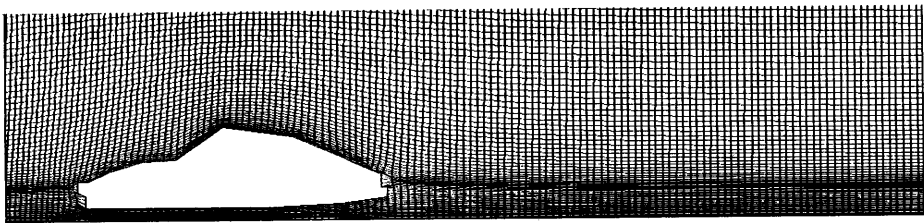


図4b H型格子

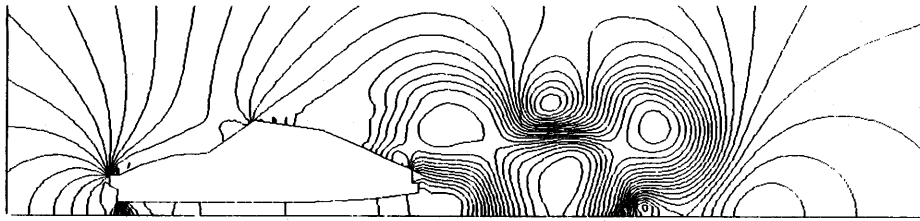


図4c 等圧力線

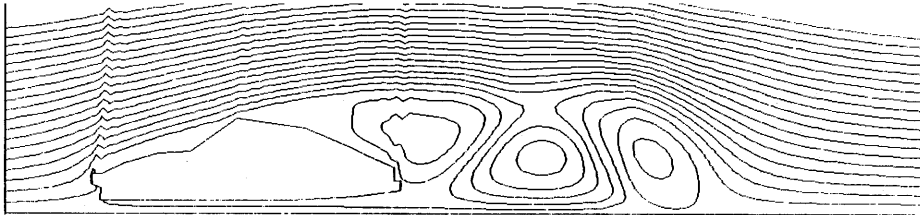


図4d 流線

表1 各解法の比較

解法	反復数 (回)	CPU時間 (SEC)	解法	反復数 (回)	CPU時間 (SEC)
ILUBCG	379	1.93	ILUCGS	294	1.55
MILUBCG	366	2.20	MILUCGS	307	1.70
TFBCG	501	1.96	TFCGS	372	1.44
MTFBCG	534	2.04	MTFCGS	427	1.65
SOR	1000	3.26	※SOR法は収束せず		

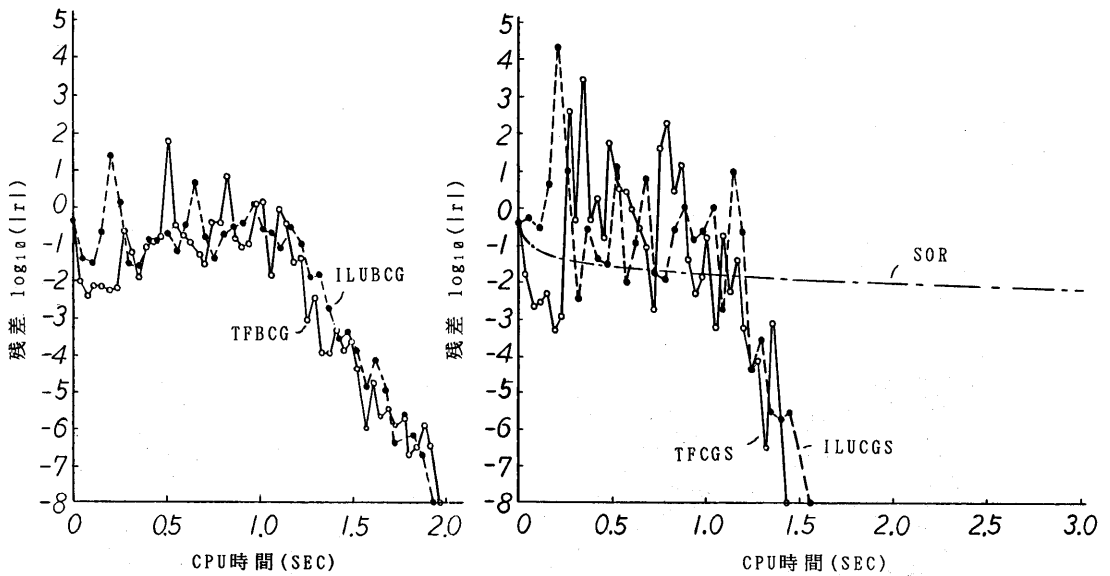


図5 CPU時間に対する収束の様子