

A Geometrical Interpretation for Fast Numerical Simulation By Electromagnetic-Hydrodynamics

Katsuyoshi Sotani
NEC, Japan

Abstract. The recent advanced development of computers continues to grow rapidly in such fields as Structural Dynamics, Fluid Dynamics, Computational Physics, Computational Chemistry, Atomic Energy, Fusion Science, Space Science, other Big Science, Life Science, Medical Science, and Pharmaceutical Science, the others due to the object. The more computers develop, the more they are required to quickly solve large-scale matrix calculations. Numerical calculation, for example in physics, in such disciplines as theoretical physics, experimental physics, and secondly computational physics that develop in the third part will comprise computational science systems.

In this paper, the object of research in a magneto-fluid 2-fluid hybrid type model represented electromagnetic-hydrodynamics is part of the fundamental fusion that improves numerical calculation inside linear systems and realizes fast numerical solutions. It improves numerical solutions to maintain preconditioned attention on fast calculation for original numerical calculation methods. We report research points from a geometric viewpoint that explain why we can realize high speed.

Keywords : electromagnetic-hydrodynamics simulation, fast numerical solution, computational physics, preconditioning, geometrical expression, eigenvalue, eigenvector

電磁流体力学高速数値シミュレーションの幾何学的解釈

曾谷勝義
NEC

要約 近年の計算機の高度な発達は、構造力学を始め、流体力学、計算物理、計算化学、原子力、核融合、宇宙科学、その他の巨大科学や、生命科学、医学、薬学他に至る迄計算機の対象が広がってきた。計算機が発達すればするほど、より大型な行列計算をより高速に解法させたい要求が出てきている。数値計算は例えば物理学においては、理論物理、実験物理に次いで計算物理として第三の分野として発達し、今や計算科学としての体系が成立しつつある。

本論文では、核融合の基礎編の一部となる電磁流体力学として表される磁気流体二流体混合型モデルを研究対象として、その中の連立一次方程式に関する数値計算を改良して、高速数値解法を実現させた。オリジナルな数値計算方法に対して、高速計算として注目されている前処理を伴った数値解法に改良したが、この時の高速性が何故実現出来るかについて、幾何学的な立場から研究した点を報告する。

1. 電磁流体力学シミュレーション

磁気流体二流体に基づく電磁流体力学シミュレーションは、核融合の基礎研究を行う上で必要とされ、オリジナルの初期バージョンは、ロスアラモス研究所の核融合研究グループによつ

て研究開発された。核融合の数値解析研究グループは、磁気流体二流体の差分形式が連立一次方程式に展開された時に見る特徴ある形式（ノンゼロ要素）を活かして、計算物理に効率的と考えられている ADI 法を用いて数値解を求めている。

1・1．電磁流体力学シミュレーションコード

磁気流体二流体混合型として構成される電磁流体力学シミュレーションコードは、主プラズマが作る物理量（電流等）に対しては、周辺プラズマがその厚みの方向（主プラズマと周辺プラズマを結ぶ磁力線方向）に積分された物理量で応答するモデルとなっている。この時の周辺プラズマは、主プラズマで占められたシミュレーションボックスに対して応答する境界条件としての役割を果たす。この境界条件は主プラズマからの入力量に応じて、周辺プラズマの厚みと垂直な方向（磁力線に対して垂直な方向）の平面内の変形されたディリクレ問題を解くことで決定される。即ち主プラズマで起きた変動は、周辺プラズマからなる境界に入力量の変動と言う形で影響を与え、この入力量の変動に応じて周辺プラズマの振る舞いが、変形ディリクレ問題を解くことによって決定され、この結果得られた周辺プラズマの変動が、電場等の物理量として再び主プラズマに影響を与えるといった、いわゆるフィードバック現象が行われている（Fig.1. 参照）。多量のメッシュ数を持つ変形ディリクレ問題を計算精度良く解く問題は、困難な場合が多い。当プログラムの特徴として、時間発展の計算方法は 2 Step Lax-Wendroff 法、空間微分に関しては中心差分法を適用、時間積分に関しては 4 次の Runge-Kutta-Gill 法に基づいた数式モデルから構成されている。この数値シミュレーションの形式は、時間発展として物理量の変化に基づき、連立一次方程式の係数値を少しずつ変化させて、連立一次方程式を繰り返して解いている（Fig.2. 参照）。

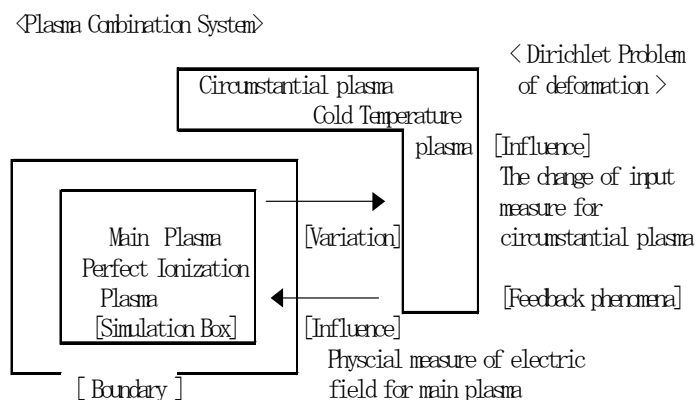


Fig.1. Relation between Main Plasma and Circumstantial Plasma.

1・2．電磁流体力学シミュレーションの方程式

電磁流体力学シミュレーションコードは、主プラズマが作る電磁流体力学の基本方程式系と、周辺プラズマが作る二流体方程式系から構成される。

電磁流体力学シミュレーションの基本方程式系

$$(1.1) \quad \left\{ \begin{array}{l} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \quad \rho \frac{\partial \mathbf{v}}{\partial t} = \mathbf{j} \times \mathbf{B} - \nabla p \\ \frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E}, \quad \mathbf{j} = \frac{1}{4\pi\eta} \nabla \times \mathbf{B} \\ \mathbf{E} = \mathbf{v} \times \mathbf{B} + \eta \mathbf{j}, \quad \frac{\partial p}{\partial t} + \nabla \cdot (p \mathbf{v}) = (\gamma-1)(-p \nabla \cdot \mathbf{v} + \eta \mathbf{j}^2) \end{array} \right.$$

ここで $\mathbf{v}, \mathbf{j}, \mathbf{B}, \mathbf{E}, \rho, p, \eta, \gamma$ は速度、電流、磁場、電場、質量密度、圧力、電気抵抗、断熱定数を表す。

周辺プラズマに対する二流体方程式系

$$(1.2) \quad \frac{\partial n}{\partial t} + \nabla \cdot (n \mathbf{v}_j) = S_j - \alpha (n^2 - n_0^2),$$

$$\mathbf{j} = \mathbf{i} \text{ (ion) or } \mathbf{e} \text{ (electron)}$$

$$(1.3) \quad m_j n \frac{d \mathbf{v}_j}{d t} = q_j n (\mathbf{E} + \mathbf{v}_j \times \mathbf{B}) - \nabla p_j + \mathbf{R}_j,$$

$$\mathbf{j} = \mathbf{i} \text{ (ion) or } \mathbf{e} \text{ (electron)}$$

ここで $\mathbf{v}_j, \mathbf{R}_j, S_j, m_j, q_j, p_j$ は j 種の粒子の速度、衝突などによる摩擦力、電流が流れ込んだり電離する事などによる密度増加の源泉項、質量、電荷、圧力を表す。そして n, n_0, α は数密度、背景に存在するプラズマの数密度、再結合率を表す。

磁場、電場に対して κ を二流体の境界面の単位法線ベクトルとすれば、磁力線は表面に沿っており、磁界の法線成分は0となるため、次の式 (1.4) となる。

$$(1.4) \quad \kappa \mathbf{B} = 0$$

電磁流体力学をメインとして、磁気流体二流体のこれら2つの異なった混合型方程式で表されるそれぞれのシミュレーション領域が磁力線によって結ばれ、物理場の情報を交換するものが、このシステムの基本的数式モデルとなっている。式 (1.1)、式 (1.2) 及び式 (1.3) は離散化された差分系の方程式として展開され、次の連立一次方程式となる。

$$(1.5) \quad \mathbf{A} \mathbf{x} = \mathbf{b}$$

この時の数値シミュレーションは、連立一次方程式をタイムステップとして、ある一定回数 (10,000回) の繰り返し計算を行う。これは物理情報を少しずつ変化させて、係数及びそれに関わる右辺情報を少しずつ変化させて、連立一次方程式を1回毎に解いて、それを一定回数の繰り返しを行い、数値解の変化を見ることになる。そして時間発展後の状態と、その計算途中の最適な値の情報を、これ迄の経験上から検討することにある。具体的には上式を展開し、中性粒子の速度、ホール移動度、ベターセン結合度、及び磁場の平行方向の移動度を検討する。磁気流体二流体のこれら2つの異なった方程式系で表されるそれぞれのシミュレーション領域が磁力線によって結ばれ、物理場の情報を研究する。その全体の流れを次に示す。

2. 高速数値解法への改良

磁気流体二流体混合型方程式は、主プラズマと周辺の関係における変形ディリクレ問題を高精度に求解することを目指して、これを出来る限り正確に反映させる為には、時間ステップ毎に数値シミュレーションを行うことにした。この磁気流体二流体混合型方程式では、数式モデルが連立一次方程式に展開され、その時のノンゼロ要素が対角線にまとまる5点差分形式となる特徴があり、計算物理におけるこの分野では、従来から対角行列の数値解法に高速性をもたらすADI法が重要視され、効果的であると考えられてきた。これに対して当論文では、最近社会で広く使われているBCG法(Biconjugate Gradient Method)に、前処理(Preconditioning)としてILUを用いたILUBCG法(Incomplete LU Biconjugate Gradient Method)を用いる事で、安定した高速数値解法を実現させたことにあり、そしてこの前処理の高速性を幾何学的に研究した点にある。先ずこの章では、前処理の基本的な考え方や各流派の概要を紹介し、実モデルにおける高速数値計算結果を求めた。

[備考：数値解法の英字省略名は、スーパーコンピュータ日本物理学会編で用いられている名称に準じる。]

2-1. 前処理法系の各流派

行列をそのまま解くよりも、前処理をした後に数値解法を行った方が、数値解が速く収束する事が知られる様になった。数値計算における最近の潮流として、前処理法が多く研究されている。ここでは前処理の基本である不完全LU分解や、数値計算の最近の潮流として前処理に対する代表的な流派を紹介し、併せてその動向や効率のとされる考え方を説明する。

それは高速数値解法を目指した数値計算の方法として、LUの不完全分解(ILU)を始め、早くから前処理として研究され、実用化されている代表的なICCG法(Incomplete Cholesky Conjugate Gradient Method)や、最近有望視されているMeijerink流、Gustafsson流、HAYAMI流、更にDOI-HARADA流、他がある。高速計算に向けて、これらはいずれも前処理を重点的に改良提案したものである。これらをここで前処理法系と呼び、各流派の前処理の方法を簡単に説明し、そして今回の磁気流体二流体に適用させた解法を紹介する。

(1) 前処理の基本的な考え方

先ずは、前処理の基本となる不完全なLU分解の考えから入る。 $\mathbf{A}$ をLU分解すると解は求められる。ここで $\mathbf{A}$ が大型疎行列である場合、 $\mathbf{L}$ (下三角行列)を完全に計算しようとすると時間がかかり、 $\mathbf{A}$ が疎である特徴が失われる。そこでLU分解を行う時、 $\mathbf{L}$ の全てを正確に計算せずに、ある場所の成分を強制的に0にする方法を用いる。これは行列 $\mathbf{A}$ の成分が0であった場所をとる事が多い。強制的に0にする場所の集合を $\mathbf{P}_u$ とすると、

$$(2.1) \quad \mathbf{P}_u = \{ (i, j) \mid a_{ij} = 0 \}$$

が定義出来る。計算を実行する時、LU分解アルゴリズムにおいて、左辺に現れる成分の添え字が $\mathbf{P}_u$ に属する時、その成分は計算せずにとばし、右辺に現れる添え字が $\mathbf{P}_u$ に属する時、その成分は0とおいて計算を行う。 $\mathbf{L}$ 、 $\mathbf{U}$ (上三角行列)を $\mathbf{A}$ の完全なLU分解として、 $\tilde{\mathbf{L}}$ と $\tilde{\mathbf{U}}$ がある正則行列として $\mathbf{A}$ の不完全なLU分解とする。次の式

$$(2.2) \quad (\tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1}) (\tilde{\mathbf{U}} \mathbf{x}) = \tilde{\mathbf{L}}^{-1} \mathbf{b}$$

を考える。

$\tilde{\mathbf{L}} \tilde{\mathbf{U}}$ が $\mathbf{A}$ の完全な LU 分解 ($=\mathbf{L}\mathbf{U}$) であれば、式 (2.2) において、 $\tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1}$ が単位行列となる。 $\tilde{\mathbf{L}} \tilde{\mathbf{U}}$ が $\mathbf{A}$ の完全な LU 分解でなくても、LU 分解に近づいたものであれば、これに CG 法 (Conjugate Gradient Method) を適用することで、より少ない反復計算で解が求められ、効率的に計算することが期待できる。

コレスキー分解の場合、LU 分解の代わりに修正コレスキー分解として、不完全 LU 分解と同様の手順で行うことを、不完全コレスキー分解と言われる。次に不完全コレスキー分解の代表である ICCG 法としての前処理を述べる。

(2) ICCG 法

$\mathbf{A}$ は対称正定値行列とする。ICCG 法は不完全コレスキー分解の前処理の後、CG 法で解を求めるものである。この算法の基礎は、次の形となっている。CG 法は n 次元空間でそれぞれ直交するベクトルを定義し、1つの点から次の点へ向かうのに、変化への方向ベクトルに変化係数を乗算した値を加えることで求めている。

$\mathbf{D}$ を $\mathbf{A}$ の対角行列とする。いま式 (1.5) における $\mathbf{A} \mathbf{x} = \mathbf{b}$ に対して、行列 $\mathbf{A}$ の不完全コレスキー分解を行う。行列 $\mathbf{A}$ の不完全コレスキー分解を $\tilde{\mathbf{L}} \tilde{\mathbf{D}} \tilde{\mathbf{L}}^T$ とする。この時、式 (2.2) は

$$(2.3) \quad \left[\left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^{-1} \mathbf{A} \left(\left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^T \right)^{-1} \right] \left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^T \mathbf{x} = \left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^{-1} \mathbf{b}$$

となる。 $\tilde{\mathbf{D}}^{1/2}$ は $\tilde{\mathbf{D}}$ の平方根を成分とする対角行列である。

$$(2.4) \quad \tilde{\mathbf{A}} = \left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^{-1} \mathbf{A} \left(\left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^T \right)^{-1}$$

$$(2.5) \quad \tilde{\mathbf{x}} = \left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^T \mathbf{x}$$

$$(2.6) \quad \tilde{\mathbf{b}} = \left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^{-1} \mathbf{b}$$

とおくことで、式 (2.4)、(2.5)、(2.6) により式 (2.3) は

$$(2.7) \quad \tilde{\mathbf{A}} \tilde{\mathbf{x}} = \tilde{\mathbf{b}}$$

となり。式 (1.5) に対して、式 (2.7) を導くことが出来、この式を解くことになる。つまり不完全コレスキー分解では、式 (2.4) の形である $\tilde{\mathbf{A}} = \left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^{-1} \mathbf{A} \left(\left(\tilde{\mathbf{L}} \tilde{\mathbf{D}}^{1/2} \right)^T \right)^{-1}$

とおくことを $\mathbf{A}$ の前処理としている。これを CG 法にみる計算ステップに準じて進める方法が ICCG 法である。 $\mathbf{A}$ が大規模疎行列であれば、式 (1.5) を式 (2.3) に変換した後に計算する方が、その変換する為の時間を加えても、式 (1.5) を直接計算するよりも、速く収束することが知られている。

(3) Meijerink 流

一般的な前処理としての考え方は、式 (1.5) における $\mathbf{A}$ に対して $\mathbf{A}$ に関する上三角行列 $\mathbf{U}$ (又は下三角行列 $\mathbf{L}$) を採り上げ、この転値及び逆行列を $\mathbf{A}$ の前後から掛ける事により、 $\mathbf{A}$ の条件数を改良する事にある。いま $\mathbf{A}$ に近い正定値行列 $\mathbf{M}$ を選ぶ。この選定の仕方は幾つか考えられるが、ここで前処理としての一般の考えは、この $\mathbf{M}$ には $\mathbf{A}$ の $\mathbf{U}$ を用いて、 $\mathbf{M} = \mathbf{U}^T \mathbf{U}$ とした $\mathbf{U}^T \mathbf{U}$ のコレスキー分解を施す点である。もとの方程式である式 (1.5) に、両辺から $\mathbf{U}^{-T}$ を掛けると

$$(2.8) \quad \mathbf{U}^{-T} \mathbf{A} \mathbf{x} = \mathbf{U}^{-T} \mathbf{b}$$

式 (2.8) に対して $\tilde{\mathbf{x}} = \mathbf{U} \mathbf{x}$ とおけば、

$$(2.9) \quad \mathbf{U}^{-T} \mathbf{A} \mathbf{U}^{-1} \tilde{\mathbf{x}} = \mathbf{U}^{-T} \mathbf{b}$$

$\tilde{\mathbf{A}} = \mathbf{U}^{-T} \mathbf{A} \mathbf{U}^{-1}$ 、 $\tilde{\mathbf{b}} = \mathbf{U}^{-T} \mathbf{b}$ とおく事で、式 (2.9) は式 (2.7) で表現される $\tilde{\mathbf{A}} \tilde{\mathbf{x}} = \tilde{\mathbf{b}}$ の形となる。この $\mathbf{U}^{-T} \mathbf{A} \mathbf{U}^{-1}$ ($= \tilde{\mathbf{A}}$) は、条件数が $\mathbf{A}$ に比較して 1 に近くなる。

この時、Meijerink は $\mathbf{U}^T \mathbf{U}$ ではなく、 $\mathbf{A}$ 自体の特徴を保有するこの対角行列を取り入れた $\mathbf{U}^T \mathbf{D} \mathbf{U}$ のコレスキー分解に注目している。この対角行列 $\mathbf{D}$ を取り入れることにより、前処理として更に計算を効率的にする方法である。つまり固有値は $\mathbf{U}^T \mathbf{U}$ だけの時より、ある対角行列 $\mathbf{D}$ を掛ける事で、更に 1 に近くなる事が分かる。そして $\mathbf{A}$ に Meijerink 流による不完全ガウスの表現を $\mathbf{D}$ に近い $\tilde{\mathbf{D}}$ を用いて、次の形で定義する。

$$(2.10) \quad \mathbf{A} = \mathbf{L} \tilde{\mathbf{D}} \mathbf{U} - \mathbf{R}$$

ここで式 (2.10) から近似される $\mathbf{L} \mathbf{D} \mathbf{U}$ を使用して、不完全 $\mathbf{L} \mathbf{D} \mathbf{U}$ の逆を両辺から掛ける。

$$(2.11) \quad (\mathbf{L} \mathbf{D} \mathbf{U})^{-1} \mathbf{A} \mathbf{x} = (\mathbf{L} \mathbf{D} \mathbf{U})^{-1} \mathbf{b}$$

Meijerink 流では、式 (2.11) に対して BCG 法 (Biconjugate Gradient Method) や、CR 法 (Conjugate Residual Method) を適用する。この時の BCG 法は、汎用性に富み安定解が得られる。この方式による考え方は、スムーズに収束すると多く報告されている。

(4) Gustafsson 流

これに対して、 $0 < u < 1$ なる u を用いて、付加項に掛けて不完全 $\mathbf{U}^T \mathbf{D} \mathbf{U}$ を作る。こうした方法で Meijerink の数値計算を改良したものが Gustafsson 流である。Gustafsson はパラメータ u を導入し、更なる高速計算の実現を目指している。

(5) HAYAMI 流

HAYAMI は固有値となる対角行列に着目し、この対角となる行列をあらかじめピックアップして改良した行列を採り上げた。この時の前処理として $\mathbf{A}$ の前後からある対角を掛ける事で、条件数を大幅に改善し、ベクトル計算機に合わせた高速計算をさせる事に成功した。 $\mathbf{A}$ を対称正定値行列とする。HAYAMI は $\mathbf{A}$ の左右から $\mathbf{A}$ をもとにした一つの行列を掛け、条件数を改良する事をスケーリングと定義した。 $\mathbf{A}$ をスケーリングするにおいて、 $\mathbf{A}$ の対角行列に着目する。

$\mathbf{A}$ の対角行列を $\mathbf{D}$ とおくと、ここで $\mathbf{D}^{-\frac{1}{2}}$ を定義し、 $\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} (= \tilde{\mathbf{A}})$ の形に変形する。この時、 $\tilde{\mathbf{A}}$ も対称正定値行列となる。

$$(2.12) \quad \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{D}^{\frac{1}{2}} \mathbf{x} = \mathbf{D}^{-\frac{1}{2}} \mathbf{b}$$

式 (2.12) で $\tilde{\mathbf{x}} = \mathbf{D}^{\frac{1}{2}} \mathbf{x}$ 、 $\tilde{\mathbf{b}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{b}$ とおく事で、式 (2.12) は式 (2.7) で表現される $\tilde{\mathbf{A}} \tilde{\mathbf{x}} = \tilde{\mathbf{b}}$ の形となる。この $\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} (= \tilde{\mathbf{A}})$ は、もとの $\mathbf{A}$ に比べて、条件数がより 1 に近くなる。 $\mathbf{A}$ が対角優位な行列で有れば、これは顕著に現れる。これは SCG 法(Scaled Conjugate Gradient Method)と呼ばれる。HAYAMI 流は、元の行列 $\mathbf{A}$ の前後から効果的な対角行列を掛けることで、行列の条件数を改良し、高速解法を実現している。

(6) DOI・HARADA 流

DOI と HARADA は 5 点や 7 点差分行列において、 $\mathbf{A}$ を対角行列 $\mathbf{D}$ と、 x 方向微分に関する非対角要素からなる行列 $\mathbf{A}_x$ 、 y 方向微分に関する非対角要素からなる行列 $\mathbf{A}_y$ に分解して次式を定義した。

$$(2.13) \quad \mathbf{A} = \mathbf{D} + \mathbf{A}_x + \mathbf{A}_y$$

この条件での 3 つの行列から構成される為、三項対角近似因子分解(Tridiagonal Approximate Factorization Method)と名付けた。この時の前処理行列 $\mathbf{M}_{TF}$ は、次式で定義される。

$$(2.14) \quad \mathbf{M}_{TF} = (\mathbf{D} + \mathbf{A}_x) \mathbf{D}^{-1} (\mathbf{D} + \mathbf{A}_y)$$

式 (2.14) の場合の逆行列計算は、計算上各因子毎に行う。これらを前処理としての基礎反復法を提案した。特徴として規則、疎な非対称行列に対しても、前処理として大きな効果をもたらし、特に大規模な疎行列に対して高速解法が期待出来る。

[磁気流体二流体モデルに適用させた解法]

前処理法系はこれ以外に幾つも提案されており、その基本的な考え方は、元の行列の条件数を改良することに特徴がある。磁気流体二流体におけるオリジナルな解法は、ADI 法により解かれている。これに対して改良版として、高速解法を目指した。その改良版として、まずはクリロフ部分空間解法として高速安定解法が期待出来る BCG 法を選定した。この BCG 法へ導く迄に、前処理を行う事は計算の効率を高めることになる。前処理を含めた解法は、与えられたマトリックスに対して、適切な処理により少しでも密集固有値形に導き、条件数を改良するアプローチ方法である。これには ILU 前処理をした ILUBCG 法を採り上げる。

これを幾何学的に考えると、 n 次元固有値空間における超楕円体が球体により近づく事を意味し、数値解法が容易になってくる。この前処理の後、共役勾配法系を適用することは、与えられた空間の中において最短距離で解に収束させる事を意味している。数値解法では、安定して効果を発揮すると予想出来る。今回の研究の主目的は、磁気流体二流体混合型の方程式が、Fig.3. の形に展開されて、次の章で述べる前処理の効果による高速数値解法における幾何学的解釈にあり、ここでは標準的方法を用いた BCG 法、及び ILUBCG 法を適用して検討する。

(7) BCG 法

BCG 法は、式(1.5)に対して、双対な方程式 $\mathbf{A}^T \mathbf{x}^* = \mathbf{b}^*$ を組み合わせた $2n$ 元連立一次方程式を定義して計算する。 $2n \times 2n$ 行列と $2n$ 次ベクトル

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^T \end{bmatrix}, \quad \tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{x} \\ \mathbf{x}^* \end{bmatrix}, \quad \tilde{\mathbf{b}} = \begin{bmatrix} \mathbf{b} \\ \mathbf{b}^* \end{bmatrix} \quad \text{として、}$$

式 (2.7) の $\tilde{\mathbf{A}} \tilde{\mathbf{x}} = \tilde{\mathbf{b}}$ の形を導くことになる。 BCG 法は、具体的には次のステップとなる。

初期ベクトル $\mathbf{x}_0$, 収束判定値 eps

$$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, \quad \mathbf{p}_0 = \mathbf{r}_0 = \mathbf{p}_0^* = \mathbf{r}_0^*$$

while $\|\mathbf{r}_k\| > \text{eps} * \|\mathbf{b}\|$ do , ($k=0,1,2,\dots$)

$$\alpha_k = (\mathbf{r}_k, \mathbf{r}_k^*) / (\mathbf{A} \mathbf{p}_k, \mathbf{p}_k^*)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k, \quad \mathbf{r}_{k+1}^* = \mathbf{r}_k^* - \alpha_k \mathbf{A}^T \mathbf{p}_k^*$$

$$\beta_k = (\mathbf{r}_{k+1}, \mathbf{r}_{k+1}^*) / (\mathbf{r}_k, \mathbf{r}_k^*)$$

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k, \quad \mathbf{p}_{k+1}^* = \mathbf{r}_{k+1}^* + \beta_k \mathbf{p}_k^*$$

continue

次に、BCG 法に ILU 前処理を施した ILUBCG 法の計算ステップを示す。

(8) ILUBCG 法

行列 $\mathbf{A}$ の不完全分解 $\mathbf{A} = \mathbf{L}\mathbf{U} + \mathbf{R}$ を経た後に BCG 法を適用する。

$\tilde{\mathbf{L}}\tilde{\mathbf{U}}$ を不完全な LU 分解とする。

初期ベクトル $\mathbf{x}_0$, 収束判定値 eps

$$\mathbf{r}_0 = (\tilde{\mathbf{L}}\tilde{\mathbf{U}})^{-1} (\mathbf{b} - \mathbf{A}\mathbf{x}_0), \quad \mathbf{p}_0 = \mathbf{r}_0 = \mathbf{p}_0^* = \mathbf{r}_0^*$$

while $\|\mathbf{r}_k\| > \text{eps} * \|\mathbf{b}\|$ do , ($k=0,1,2,\dots$)

$$\alpha_k = (\mathbf{r}_k, \mathbf{r}_k^*) / ((\tilde{\mathbf{L}}\tilde{\mathbf{U}})^{-1} \mathbf{A} \mathbf{p}_k, \mathbf{p}_k^*)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k (\tilde{\mathbf{L}}\tilde{\mathbf{U}})^{-1} \mathbf{A} \mathbf{p}_k, \quad \mathbf{r}_{k+1}^* = \mathbf{r}_k^* - \alpha_k ((\tilde{\mathbf{L}}\tilde{\mathbf{U}})^{-1} \mathbf{A})^T \mathbf{p}_k^*$$

$$\beta_k = (\mathbf{r}_{k+1}, \mathbf{r}_{k+1}^*) / (\mathbf{r}_k, \mathbf{r}_k^*)$$

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k, \quad \mathbf{p}_{k+1}^* = \mathbf{r}_{k+1}^* + \beta_k \mathbf{p}_k^*$$

continue

2-2. 数値シミュレーション

この節では、ADI 法を基礎に持つオリジナル版と、BCG 法および ILUBCG 法の改良版による数値シミュレーション結果の比較を表示する。

(1) 計算機環境

計算機環境はスーパーコンピュータ SX8R/1CPU、メモリ 256GB、倍精度

(2) 数値シミュレーション結果の比較

オリジナル版(General original version) : ADI法

改良版1(Improved 1 version) : BCG法

改良版2(Improved 2 version) : ILUBCG法

相対残差の打切 ε を 10^{-0} 時点から 10^{-10} 時点迄におけるオリジナル版(General original version)と、改良版1(Improved 1 version)と、改良版2(Improved 2 version)の比較した値を表示する。いずれも10,000回の繰り返し数値計算結果(シミュレーションコードの全体経過時間)である。

Table 1. Comparative Table of Executive Time. (Total Elapse Time)

Relative Residual	General original version(msec)	Improved1 version(msec)	Improved2 version(msec)
10-0	113341	104273	13230
10-1	280996	255706	32587
10-2	429615	382357	48978
10-3	716201	630256	81656
10-4	969720	833959	110189
10-5	1223790	1027983	138758
10-6	1530401	1254928	173219
10-7	1825389	1460311	203071
10-8	2143402	1693287	236016
10-9	2483726	1937306	270223
10-10	2940058	2263844	313276

(注) オリジナル版、改良版1、改良版2において、プログラム構成(ベクトル化率)の面から全てが同一条件ではない。オリジナル版を高速化する上で、改良版1と2では、高速化手段として、ベクトル化率も同時に向上させている。

ベクトル化率、オリジナル版(General original version) : 78.53%

改良版1(Improved 1 version) : 84.95%

改良版2(Improved 2 version) : 85.08%

(3) 収束時間の比較

10,000 回の繰り返し数値計算結果と、そのシミュレーション図を示す。

(Table 1.、Fig.4. 参照)

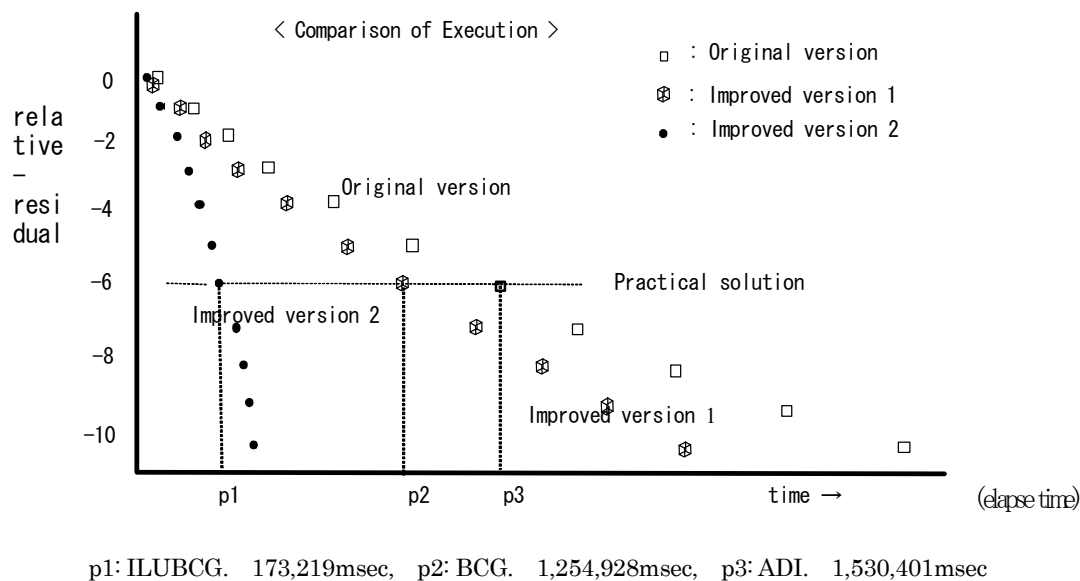


Fig.4. Comparative Illustration.

実用的な打ち切り値 10^{-6} レベルでは、オリジナル版と比較して改良版 1 では 1.22 倍、改良版 2 では 8.84 倍になっている。このシミュレーションコードの構成上、全体の実行時間 (Total Elapse Time) の中で 68% が CPU 時間である。改良版 1 と 2 では、高速化の為にチューニングを実施した。完全な同一条件の比較ではないが、前処理を実施することにより高速計算が評価出来る。上記以外の研究として、行列のサイズを大きくしたりすると、スパース性の比率から前処理としての高速性は顕著になる。

続く章では、前処理をすれば何故これほど速くなるのかについて、前処理の高速性を幾何学的な解釈として述べる。当モデルの最大固有値と最小固有値の比率 (条件数) は、次の値である。

電磁流体力学シミュレーションモデルの条件数 ; 8.53608×10^2

3. 数値計算前処理法系の幾何学的解釈

この章では、今回研究対象とした磁気流体二流体混合型問題の数値解法に、オリジナルな ADI 法に対して、改良版として BCG 法及び ILUBCG 法を適用した結果、特に ILUBCG 法において高速数値計算が実現出来た。前処理の考えのない ADI 法と比較して、更に前処理を装備出来る BCG 法に対しても、この ILU 前処理における高速解法の考え方を、幾何学的立場から説明する。

3-1. 幾何学的説明の前段階、前処理法と効率化

大型行列における効果的解法としての共役勾配法系を考えるにおいて、この解法は計算の理論上 n 回の反復で真の解に到達する事が出来るが、実際の計算過程では丸め誤差の影響もあり、収束の為に n 回以上の反復を必要とする事がある。しかし他の解法と違って、計算の理論上よりも少なく収束する場合もある。それは行列 $\mathbf{A}$ の固有値が重複 (縮退) している場合であり、その重複している分だけ n 回よりも少ない回数で収束することが出来る。これは Fig.3. に表されているマトリックスに対し、適切な前処理を施す事により、少しでも密集固有値形に

導き、条件数を改良するアプローチである。幾何学的には n 次元固有値空間における超楕円体が球体に近づく事を意味し、これにより数値解法が容易になってくる。前処理のあと勾配法を適用する事は、これは最短距離で解に収束させる事を意味しており、これは数値計算としての効率化である。

3-2. Gerschgorin 定理と代表的な前処理法

この節においては、前処理法の効果について幾何学的解釈を用いて紹介する。まずは、幾何学的解釈の基礎となる Gerschgorin 定理について、以下に示す。

〔Gerschgorin 定理〕

行列における固有値問題に関して、

$\mathbf{A} = (a_{ij})$ を n 次の行列とする。(a_{ij} は複素数でも可)
対角要素 a_{ii} を中心に半径 r_i の円を n 個描く。半径は次式で示される。

$$r_i = \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|, \quad (1 \leq i \leq n)$$

この時 $\mathbf{A}$ の固有値は全て n 個の円の共通集合の中に含まれる。

この Gerschgorin 定理の応用として、次を説明したい。前処理法が効果的であるのは、行列 $\mathbf{A}$ の条件数が改良される事にある。この条件数の改良の考えを以下に示す。行列 $\mathbf{A}$ の固有値が密集している時、密集している個数のみ収束が速まる事が知られている。元の行列 $\mathbf{A}$ に対して、ある種の行列（非単位行列）を掛けると、条件数は悪くなり、又もとの行列 $\mathbf{A}$ に対して、ある種の行列（非単位行列）で割る（逆行列を掛ける）と条件数は良くなる。

これは例えば、ある楕円を自乗すると、長軸と短軸の比率はさらに高くなり、ある楕円に対して、近い形の楕円で割ると、長軸と短軸の比率は低くなることは、容易に理解出来る。ここで軸を固有値とすれば、長軸を短軸で割ったものが条件数であり、この条件数が大きくなると数値計算は困難であり、条件数が小さくなるにつれて、数値計算は容易となる事が知られている。この時、長軸と短軸の比率がさらに高くなることは、条件数が悪くなり、長軸と短軸の比率が低くなることで、条件数は良くなることを意味する。つまりもとの行列 $\mathbf{A}$ に対して、ある種の行列で割る（逆行列を掛ける）と、条件数は良くなることになる。

前処理法とは、これは与えられたマトリックスに対し、適切なある処理を施す事により、少しでも密集固有値形に導き、条件数を改良する事を意味している。この適切な処理は、ある種の行列を掛けたりしている。数値計算上、前処理が効果をもたらすのは、反復法に関してであり、その反復法においても、ある点から次への点へは、空間内において直交性を保って探索する考え方として、数値解を進める共役勾配法が有益である。Hestenes と Stiefel が定義したオリジナルの共役勾配法は、対称行列に適合させる事が出来る。

この共役勾配法は、今日多くの改良された枝葉の方法が提案されており、共役勾配法系統として位置づける事が出来る。その多くは、共役勾配法に前処理を用いて効率的に計算を行うものである。前処理を用いた共役勾配法系統全体として様々な手法があり、それらの手法を適用

させることで、行列の形として、密、疎、対称、非対称、規則、不規則等に応じて用いる事が出来る。一般的には、大規模疎行列について適用すれば効果的である事が知られている。良く知られている代表的な前処理法として、前章 2. 高速数値解法への改良で紹介した前処理法系の各流派の他、当モデルに適用させて紹介した ILUBCG 法を始め、MICCG 法、ILUCGS 法、ILUCR 法、MILUCR 法、TFCGS 法、その他がある。この前処理の効率性を、Gerschgorin 定理の応用から幾何学的解釈による方法で説明する。

3-3. 前処理の方法と幾何学的解釈

前章 2. 高速数値解法への改良では、前処理法の各流派について概要を述べているが、この節においては、前処理法の式の具体的な展開を示して、幾何学的に解釈を行う。

行列の前処理法の考え方として、行列 $\mathbf{A}$ に対して $\mathbf{P}$, $\mathbf{Q}$ を $(n \times n)$ 正則行列とする。ここで式 (1.5) に対して、同値な方程式を次に定義する。

$$(3.1) \quad \mathbf{P}^{-1} \mathbf{A} \mathbf{Q}^{-1} (\mathbf{Q} \mathbf{x}) = \mathbf{P}^{-1} \mathbf{b}$$

式 (3.1) で $\mathbf{B} = \mathbf{P}^{-1} \mathbf{A} \mathbf{Q}^{-1}$, $\mathbf{y} = \mathbf{Q} \mathbf{x}$, $\mathbf{c} = \mathbf{P}^{-1} \mathbf{b}$ とおくことで $\mathbf{B} \mathbf{y} = \mathbf{c}$ を解き、次いで $\mathbf{Q} \mathbf{x} = \mathbf{y}$ を解く。この $\mathbf{P}$, $\mathbf{Q}$ は計算の容易性から、一般には対角行列や三角行列が選ばれる。この時 $\mathbf{B}$ の条件数は、元の行列 $\mathbf{A}$ よりも良くなる（条件数の値が小さくなる）行列である。

この条件数の改良は、高速数値計算を行う上で重要なことである。条件数を改良する方法として、行列の適切な分離や、行列の適切な分解等が良く知られている。磁気流体二流体のモデルでは、ILU の前処理のあと BCG 法へ適用させた。ここでは幾何学上、前処理を磁気流体二流体混合型モデル解法として採用した ILU として説明する。

[ILU 前処理]

前章 2 の (1) 前処理の基本的な考え方で述べた如く、 $\mathbf{L}$ と $\mathbf{U}$ を $\mathbf{A}$ の完全な LU 分解として、ある正則行列を $\tilde{\mathbf{L}}$ と $\tilde{\mathbf{U}}$ とする。これは不完全な形の LU 分解とする。前処理として式 (2.2) で表されている形として

$$(3.2) \quad \tilde{\mathbf{A}} = \tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1}$$

とおくことで、 $\tilde{\mathbf{A}}$ を前処理として $\mathbf{A}$ の LU 分解に近づけた形にすることが出来る。式 (3.2) は $\mathbf{A}$ の不完全な LU 分解 $\tilde{\mathbf{L}}$ と $\tilde{\mathbf{U}}$ の逆行列を、 $\mathbf{A}$ の左右から掛けたものである。次に

$$(3.3) \quad \tilde{\mathbf{x}} = (\tilde{\mathbf{U}} \mathbf{x})$$

$$(3.4) \quad \tilde{\mathbf{b}} = \tilde{\mathbf{L}}^{-1} \mathbf{b}$$

とおくことで、式 (2.7) と同じく、 $\tilde{\mathbf{A}} \tilde{\mathbf{x}} = \tilde{\mathbf{b}}$ の形に導くことが出来る。この後、CG 法に結びつけることにより、より少ない反復計算で解が求められ、効率的に計算することが期待出来る。

つまり解くべき式は、式 (2.2) $(\tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1})(\tilde{\mathbf{U}} \mathbf{x}) = \tilde{\mathbf{L}}^{-1} \mathbf{b}$ に示した形として

表現できる。

〔固有値問題〕

ある $n \times n$ の正方行列 $\mathbf{A}$ に関して

$$(3.5) \quad \mathbf{A} \mathbf{x} = \lambda \mathbf{x} \quad , \quad \mathbf{x} \neq \mathbf{0}$$

が成立する。スカラ λ は固有値、列ベクトル $\mathbf{x}$ が固有ベクトルとする。幾何学的には、行列 $\mathbf{A}$ の与える 1 次変換に関して、不変な方向を見つけることにある。 $\mathbf{I}$ を $\mathbf{A}$ と同じ大きさの単位行列とする時、固有値 λ は次の固有方程式として求めることが出来る。

$$(3.6) \quad |\mathbf{A} - \lambda \mathbf{I}| = 0$$

固有値 λ は式 (3.6) の根である。固有値は重複度も入れて、行列の次数だけある。固有方程式の根が全て単根であれば、各固有値に対する固有ベクトルは

$$(3.7) \quad (\mathbf{A} - \lambda \mathbf{I}) \mathbf{x} = \mathbf{0}$$

の解として、一意に定まる。その時、 n 個の一次独立な固有ベクトルが得られる。固有値に重根 (m 個) がある場合でも、その重根に対応して、重根が m 個の一次独立な固有ベクトルを選ぶことが出来る。

〔最大固有値、最小固有値〕

元の行列 $\mathbf{A}$ の最大固有値 $\lambda_{\max}$ 、最小固有値 $\lambda_{\min}$ とし、ある種の行列の最大固有値 λ_1 、最小固有値 λ_n とすれば、2 つの行列を掛ける事は

$$(3.8) \quad \lambda_{\max} \times \lambda_1 / \lambda_{\min} \times \lambda_n = \text{Ratio 1}$$

2 つの行列を割る事は

$$(3.9) \quad \lambda_{\max} \div \lambda_1 / \lambda_{\min} \div \lambda_n = \text{Ratio 2}$$

式 (3.8)、(3.9) から $\text{Ratio 1} > \text{Ratio 2}$ となる事は明らかである。

つまり、ある種の行列で割ること（逆行列を掛ける事）によって、最大最小の比は小さくなる事が分かる。ある種の行列は、元の行列 $\mathbf{A}$ をコレスキー分解した行列や、元の行列 $\mathbf{A}$ の対角行列を取り扱う事が望ましい。

◎ 元の行列

Fig.3. で示した形で、この行列 $\mathbf{A}$ ($6,400 \times 6,400$) が元の問題（行列）である。この行列 $\mathbf{A}$ に対する楕円体は、次式で表現出来る。

$$(3.10) \quad (\mathbf{x}, \mathbf{A} \mathbf{x}) = 1$$

$\mathbf{A}$ の各要素を a_{ij} とする。式 (3.10) は、元の行列に対する n 次元空間における楕円体を形成する。

$$(3.11) \quad (\mathbf{x}, \mathbf{A} \mathbf{x}) = \sum_{i,j=1}^n a_{ij} x_i x_j = 1$$

$$\mathbf{x} = (0, \dots, 0, x_{ii}, 0, \dots, 0)$$

式 (3.11) において、 n 次元空間における楕円体の x_i に対する切片は $\pm 1 / \sqrt{a_{ii}}$ の位

置となる。

ここで線型空間における対称変換の定理の一つとして、行列 $\mathbf{A}$ の最大固有値及び最小固有値は、次式で表現される。

$$(3.12) \quad \lambda_{\max} = \max_{\mathbf{x} \neq \mathbf{0}} \frac{(\mathbf{x}, \mathbf{A} \mathbf{x})}{(\mathbf{x}, \mathbf{x})}$$

$$(3.13) \quad \lambda_{\min} = \min_{\mathbf{x} \neq \mathbf{0}} \frac{(\mathbf{x}, \mathbf{A} \mathbf{x})}{(\mathbf{x}, \mathbf{x})}$$

Gerschgorin 定理によると、 $\mathbf{A}$ の固有値は全て n 個の円の共通集合の中に含まれるところから、 $\mathbf{A}$ が対称正定値行列であれば、固有値 λ を一般式で表現すると、次の和集合 ($\cup$) として定義される。

$$(3.14) \quad \lambda \in \bigcup_{i=1}^n \left\{ a_{ii} - \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|, a_{ii} + \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \right\}$$

従って、 λ は以下の範囲になる。

$$(3.15) \quad \min_i \left\{ a_{ii} - \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \right\} \leq \lambda \leq \max_k \left\{ a_{kk} + \sum_{\substack{l=1 \\ l \neq k}}^n |a_{kl}| \right\}$$

ここで $\mathbf{A}$ の条件数を Cd とすると、条件数の範囲は次式で示される。

$$(3.16) \quad 1 \leq Cd \leq \frac{\max_k \left(a_{kk} + \sum_{\substack{l=1 \\ l \neq k}}^n |a_{kl}| \right)}{\min_i \left(a_{ii} - \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \right)}$$

ここで2次元平面上の幾何学的表現の為、モデルを単純化する。元の行列 $\mathbf{A}$ を、いま (2×2) 行列と仮定する。

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \quad \text{として、X1 軸と X2 軸からなる 2 次元平面を考える。}$$

元の行列 $\mathbf{A}$ に対して、 $\lambda_{\max}$ に対応する固有ベクトルを第1固有ベクトルとして $\mathbf{V}_1$ 、元の行列 $\mathbf{A}$ に対して、 $\lambda_{\min}$ に対応する第2固有ベクトルとして $\mathbf{V}_2$ とする。ここで元の行列 $\mathbf{A}$ に対して、式 (3.10) があり、これは楕円の形として表現できる。 $\mathbf{A}$ が単位行列であれば、式 (3.10) は円になる。

[楕円の長軸が水平軸 X1 とのなす角度]

$\mathbf{A}$ を (2×2) 行列として、式 (3.10) を定義すると、2次元平面上の楕円となる。元の行列において、長軸と X1 軸に対してなす角度を θ_1 とする。ここで固有ベクトルは直交する。

元の行列に対する固有値の大小の比率を考えに入れ、楕円の長軸は傾きを持つと仮定し、その時の長軸の向きは、第一象限、第三象限と仮定する。その時の角度の開きは、固有値の比率として、元の行列では $\lambda_{\min} / \lambda_{\max}$ で表され、 θ_1 の度数の範囲は第一象限で、 $0 \leq \theta_1 \leq 90^\circ$ と仮定する。

元の行列において、最大固有値と最小固有値の関係は、一般には

$$(3.17) \quad \lambda_{\min} \ll \lambda_{\max}$$

で示される。

こうした条件にすると、X1 軸と元の楕円の長軸とのなす角度 θ_1 は、次式で表現することが出来る。

$$(3.18) \quad \theta_1 = \left(\frac{\pi}{2} \right) * (\lambda_{\min} / \lambda_{\max})$$

式 (3.18) で、仮に $\lambda_{\max} = \lambda_{\min}$ とした場合、すなわち条件数が 1 の時、度数表現で $\theta_1 = 90^\circ$ となる。

ここで式 (3.5) の関係式において、 $\lambda_{\max}$ に対する第 1 固有ベクトル $\mathbf{V}_1$ は、楕円と内接する円との接点方向にあり、 $\lambda_{\min}$ に対応する第 2 固有ベクトル $\mathbf{V}_2$ は、楕円と外接する円との接点方向にある。

この時、楕円と X1 軸との交点は $(1 / \sqrt{a_{11}}, 0), (-1 / \sqrt{a_{11}}, 0)$

X2 軸との交点は $(0, 1 / \sqrt{a_{22}}), (0, -1 / \sqrt{a_{22}})$

で示される。

Gerschgorin 定理から、元の行列に対する楕円の式 (3.10) に内接及び外接する円は、次式で表現される。

楕円に内接する円は $(\mathbf{x}, \mathbf{x}) = 1 / \lambda_{\max}$

楕円に外接する円は $(\mathbf{x}, \mathbf{x}) = 1 / \lambda_{\min}$

で示される。

楕円に内接する円の X1 軸との交点は $(1 / \lambda_{\max}, 0), (-1 / \lambda_{\max}, 0)$

X2 軸との交点は $(0, 1 / \lambda_{\max}), (0, -1 / \lambda_{\max})$

楕円に外接する円の X1 軸との交点は $(1 / \lambda_{\min}, 0), (-1 / \lambda_{\min}, 0)$

X2 軸との交点は $(0, 1 / \lambda_{\min}), (0, -1 / \lambda_{\min})$

で示される。

これらから、元の行列により作られる楕円の図を示す (Fig.5. 参照)。

$(\mathbf{x}, \mathbf{A} \mathbf{x}) = 1$: 元の行列の楕円、 $(\mathbf{x}, \mathbf{x}) = 1 / \lambda_{\max}$: 楕円に内接する円、

$(\mathbf{x}, \mathbf{x}) = 1 / \lambda_{\min}$: 楕円に外接する円

第 1 固有ベクトル軸、及び第 2 固有ベクトル軸の向きは、それぞれ各軸 (X2, X1) から角度 θ_1 の傾きをなす。最大固有値と最小固有値の比(条件数)が大きくなるに従って、角度 θ_1

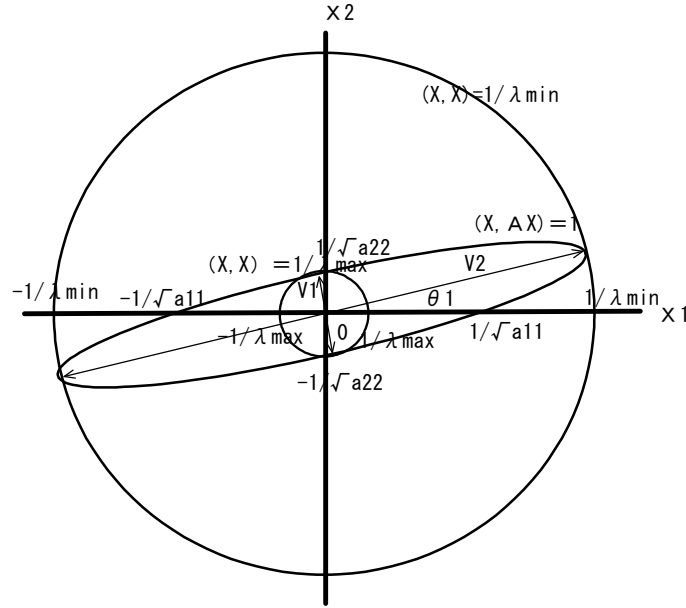


Fig.5. Geometrical Expression for Original Subject.

は小さくなり、固有ベクトルの軸（長軸）は X_1 軸に近くなる。

幾何学的説明の為、Fig.5. の見やすい形の楕円で表現しているが、多くの工学分野における一般的な実モデルでは、条件数がかかなり大きい場合が多く、最大固有値と最小固有値は大きな開きがある。つまり実モデルの楕円体は、かなりひしゃげた細長い形をとっている。

（前章2．高速数値解法への改良、2-2．数値シミュレーション、（3）収束時間の比較、電磁流体力学シミュレーションモデルの条件数、参照）

◎ 前処理後の行列

前処理としての元の行列の変形を、以下の形で行う。 Fig.3. で示した形の元の行列 $\mathbf{A}$ の前処理を行う。ある $\tilde{\mathbf{L}}$ と $\tilde{\mathbf{U}}$ を $\mathbf{A}$ の不完全な LU 分解とし、前処理として式 (3.2) から、式(1.5)を式 (2.2) の形に変形する。

式 (2.2) $(\tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1})(\tilde{\mathbf{U}} \mathbf{x}) = \tilde{\mathbf{L}}^{-1} \mathbf{b}$ より、[ILU 前処理] で述べた如く、式 (3.2)、(3.3)、(3.4) から式 (2.2) は、式 (2.7) $\tilde{\mathbf{A}} \tilde{\mathbf{x}} = \tilde{\mathbf{b}}$ の形として表現される。

この $\tilde{\mathbf{A}}$ は $\mathbf{A}$ に比べて条件数が改良され 1 に近づく。このことを以下に示す。式 (3.2) による前処理した行列 $\tilde{\mathbf{A}}$ の n 次元空間における楕円体は

$$(3.19) \quad (\mathbf{x}, \tilde{\mathbf{A}} \mathbf{x}) = 1$$

として示される。 $\tilde{\mathbf{A}}$ の各要素を $\tilde{a}_{ij}$, $(i = 1, \dots, n)$ とする。式 (3.19) は、次の形である。

$$(3.20) \quad (\mathbf{x}, \tilde{\mathbf{A}} \mathbf{x}) = \sum_{i,j=1}^n \tilde{a}_{ij} x_i x_j = 1$$

$$\mathbf{x} = (0, \dots, 0, x_{ii}, 0, \dots, 0)$$

式(3.20)において、 n 次元空間における前処理後の楕円体の $\mathbf{x}_i$ に対する切片は $\pm 1 / \sqrt{\tilde{\mathbf{a}}_{ii}}$ の位置となる。

次に、前処理した $\tilde{\mathbf{A}}$ で構成される n 次元空間における楕円体を式(3.2)の前処理した形で考える。

$$(3.21) \quad (\mathbf{x}, \tilde{\mathbf{A}} \mathbf{x}) = (\mathbf{x}, \tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1} \mathbf{x}) = 1$$

式(3.21)にみる前処理された行列 $\tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1}$ は、最大固有値と最小固有値の差は縮まり、元の行列 $\mathbf{A}$ からなる式(3.10)の楕円体に比べて、円に近づいた楕円体となる。これは式(3.2)の $\tilde{\mathbf{A}} = \tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1}$ の前処理を行うことで、元の行列 $\mathbf{A}$ より条件数が改良され1に近づく。つまり式(3.10)に比べて、式(3.21)の方が最大固有値と最小固有値の比率が縮まる。式(3.10)では、長軸（最大固有ベクトル軸）が長くなり、細長いひしゃげた形の楕円体となるが、条件数が改良された式(3.21)では、式(3.10)より球に近い形の楕円体となる。従って $\mathbf{x}_i$ に傾きを持った長軸と短軸の差は縮まる。前処理された楕円体は、元の楕円体よりもより球体に近い形を作る。

前処理後を $\tilde{\mathbf{A}}$ として、式(3.5)と同じく、スカラ $\tilde{\lambda}$ は固有値、列ベクトル $\mathbf{x}$ が固有ベクトルとして

$$(3.22) \quad \tilde{\mathbf{A}} \mathbf{x} = \tilde{\lambda} \mathbf{x}, \quad \mathbf{x} \neq \mathbf{0}$$

で表現される式が導かれる。

前処理後の式(3.22)の最大固有値 $\tilde{\lambda}_{\max}$ 、最小固有値 $\tilde{\lambda}_{\min}$ とする。式(3.12)、(3.13)の時と同じく

$$(3.23) \quad \tilde{\lambda}_{\max} = \max_{\mathbf{x} \neq \mathbf{0}} \frac{(\mathbf{x}, \tilde{\mathbf{A}} \mathbf{x})}{(\mathbf{x}, \mathbf{x})}$$

$$(3.24) \quad \tilde{\lambda}_{\min} = \min_{\mathbf{x} \neq \mathbf{0}} \frac{(\mathbf{x}, \tilde{\mathbf{A}} \mathbf{x})}{(\mathbf{x}, \mathbf{x})}$$

となる。条件数の大きい元の行列に対して、式(3.2)にみる $\tilde{\mathbf{A}} = \tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1}$ による前処理によって、式(3.19)の楕円体 $(\mathbf{x}, \tilde{\mathbf{A}} \mathbf{x}) = 1$ は、式(3.10)で表現された元の楕円体 $(\mathbf{x}, \mathbf{A} \mathbf{x}) = 1$ より、改良された楕円体を形成するので、式(3.23)、(3.24)は元の行列の $\lambda_{\max}$ と $\lambda_{\min}$ の比率よりも小さくなり、幾何学的には長軸と短軸の比が1に近づき、より球体に近い楕円体になる（条件数が1に近づく）事を意味する。

Gerschgorin 定理による解説で「元の行列」の欄における $\mathbf{A}$ の固有値は全て n 個の円の共通集合の中に含まれる形に同じく、前処理後の $\tilde{\mathbf{A}}$ の固有値 $\tilde{\lambda}$ は、元の行列 $\mathbf{A}$ からなる式(3.14)の和集合の定義と同形式であり、次式で示される。

$$(3.25) \quad \tilde{\lambda} \in \bigcup_{i=1}^n \left\{ 1 - \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| / \sqrt{a_{ii} a_{jj}}, \quad 1 + \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| / \sqrt{a_{ii} a_{jj}} \right\}$$

従って、 $\tilde{\lambda}$ は以下の範囲になる。

$$(3.26) \quad \min_i \left\{ 1 - \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| / \sqrt{a_{ii} a_{jj}} \right\} \leq \tilde{\lambda} \leq \max_k \left\{ 1 + \sum_{\substack{l=1 \\ l \neq k}}^n |a_{kl}| / \sqrt{a_{kk} a_{ll}} \right\}$$

ここで前処理後の $\tilde{\mathbf{A}}$ の条件数を $\tilde{\text{Cd}}$ とすると、条件数の範囲は次式で示される。

$$(3.27) \quad 1 \leq \tilde{\text{Cd}} \leq \frac{\max_k \left(1 + \sum_{\substack{l=1 \\ l \neq k}}^n |a_{kl}| / \sqrt{a_{kk} a_{ll}} \right)}{\min_i \left(1 - \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| / \sqrt{a_{ii} a_{jj}} \right)}$$

元の行列からなる楕円と同じく、前処理後の行列 $\tilde{\mathbf{A}}$ を単純化して (2×2) 行列にて説明する。

$$\text{この時} \quad \tilde{\mathbf{A}} = \begin{pmatrix} \tilde{a}_{11} & \tilde{a}_{12} \\ \tilde{a}_{21} & \tilde{a}_{22} \end{pmatrix} \quad \text{で表され、この行列 } \tilde{\mathbf{A}} \text{ に対して、式 (3.22) より 最}$$

大固有値 $\tilde{\lambda}_{\max}$ に対応する第1固有ベクトル $\tilde{\mathbf{V}}_1$ 、最小固有値 $\tilde{\lambda}_{\min}$ に対応する第2固有ベクトル $\tilde{\mathbf{V}}_2$ とする。X1 軸、X2 軸からなる2次元平面上に、行列を (2×2) として前処理後の楕円を式 (3.19) の形として表現できる。

式 (3.2) による $\tilde{\mathbf{A}} = \tilde{\mathbf{L}}^{-1} \mathbf{A} \tilde{\mathbf{U}}^{-1}$ の前処理を行うことで、元の行列 $\mathbf{A}$ より条件数が改良され、1 に近づく。式 (3.10) に比べて、式 (3.19) の方が最大固有値と最小固有値の比率が縮まる。その比率は次の形である。

式 (3.10) における長軸と短軸の比率（元の行列に対応）

$$(3.28) \quad \text{Ratio O} = \frac{\lambda_{\max}}{\lambda_{\min}}$$

式 (3.19) における長軸と短軸の比率（前処理後の行列に対応）

$$(3.29) \quad \text{Ratio I} = \frac{\tilde{\lambda}_{\max}}{\tilde{\lambda}_{\min}}$$

従って

$$(3.30) \quad \text{Ratio O} > \text{Ratio I}$$

となる。

式 (3.28)、式 (3.29) より、式 (3.10) と式 (3.19) から構成された最大固有値、最小固有値間の比率（元の行列と前処理後の行列に関する比率）

$$(3.31) \quad \text{Ratio T} = \frac{\text{Ratio O}}{\text{Ratio I}} \quad (> 1)$$

となり、前処理された式 (3.29) にみる最大固有値と最小固有値の比率は、式 (3.28) の比率より改良され、その改良された比率は、式 (3.31) で示される。これは円に近い楕円となる。

〔前処理後の楕円の長軸が水平軸 X_1 とのなす角度〕

前処理後の行列は、元の問題に対する最大固有値と最小固有値の差は縮まることで、この比率が元の行列よりも 1 に近づく、式 (3.30) から分かる様に、つまり

$$(3.32) \quad \tilde{\lambda}_{\min} < \tilde{\lambda}_{\max}$$

となることで、式 (3.32) は、式 (3.17) に比べて最大固有値と最小固有値の差が小さいことを意味している。

元の行列の場合と同じく、前処理後の行列に対する楕円の長軸が、水平軸 X_1 とのなす角度を θ_2 とする。元の行列の時と同じく、固有ベクトル軸（長軸）の向きは、第一象限、第三象限と仮定する。 θ_2 の範囲は、第一象限で $0 \leq \theta_2 \leq 90^\circ$ とする。その時の角度の開きは、前処理後の行列では、固有値の比率として $\tilde{\lambda}_{\min} / \tilde{\lambda}_{\max}$ で表すことが出来る。

新しい角度 (X_1 軸と前処理後の楕円の長軸とのなす角度) θ_2 は、式 (3.18) の形と同じく

$$(3.33) \quad \theta_2 = \left(\pi/2 \right) * \left(\tilde{\lambda}_{\min} / \tilde{\lambda}_{\max} \right)$$

で表現される。

式 (3.17) の関係からなる式 (3.18) の θ_1 の角度より、式 (3.32) の関係からなる式 (3.33) の θ_2 の角度の方が大きくなることが分かる。そして長軸と短軸の差は縮まり、新しい前処理後の楕円は、元の楕円より円に近づいた形となる。従って式 (2.2) にみる形で、この時の前処理により高速数値計算が実現出来ることを意味している。

ここで式 (3.22) の関係式において、 $\tilde{\lambda}_{\max}$ に対する第 1 固有ベクトル $\tilde{\mathbf{V}}_1$ は、前処理後の楕円と内接する円との接点方向にあり、 $\tilde{\lambda}_{\min}$ に対応する第 2 固有ベクトル $\tilde{\mathbf{V}}_2$ は、前処理後の楕円と外接する円との接点方向にある。

前処理後の楕円と X_1 軸との交点は $(1 / \sqrt{\tilde{a}_{11}}, 0), (-1 / \sqrt{\tilde{a}_{11}}, 0)$

X_2 軸との交点は $(0, 1 / \sqrt{\tilde{a}_{22}}), (0, -1 / \sqrt{\tilde{a}_{22}})$

で示される。

Gerschgorin 定理から、前処理後の楕円の式 (3.19) に内接及び外接する円は、次式で表現される。

前処理後の楕円に内接する円は $(\mathbf{x}, \mathbf{x}) = 1 / \tilde{\lambda}_{\max}$

前処理後の楕円に外接する円は $(\mathbf{x}, \mathbf{x}) = 1 / \tilde{\lambda}_{\min}$

で示される。

前処理後の楕円に内接する円の X_1 軸との交点は $(1 / \tilde{\lambda}_{\max}, 0), (-1 / \tilde{\lambda}_{\max}, 0)$

X_2 軸との交点は $(0, 1 / \tilde{\lambda}_{\max}), (0, -1 / \tilde{\lambda}_{\max})$

前処理後の楕円に外接する円の X_1 軸との交点は $(1 / \tilde{\lambda}_{\min}, 0), (-1 / \tilde{\lambda}_{\min}, 0)$

X_2 軸との交点は $(0, 1 / \tilde{\lambda}_{\min}), (0, -1 / \tilde{\lambda}_{\min})$

で示される。

これから、前処理後の行列により作られる改良された楕円の図を示す (Fig.6. 参照)。

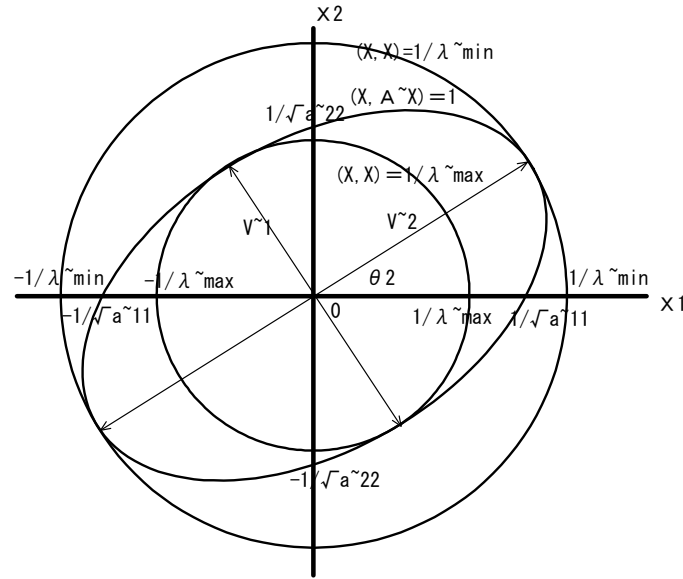


Fig.6. Geometrical Expression for After Preconditioning.

$(\mathbf{x}, \tilde{\mathbf{A}} \mathbf{x}) = 1$: 前処理後の行列の楕円、 $(\mathbf{x}, \mathbf{x}) = 1/\tilde{\lambda}_{\max}$: 前処理後の楕円に内接する円、 $(\mathbf{x}, \mathbf{x}) = 1/\tilde{\lambda}_{\min}$: 前処理後の楕円に外接する円

前処理後の第1固有ベクトル軸、及び第2固有ベクトル軸の向きは、それぞれ各軸 (X2, X1) から角度 θ_2 の傾きをなす。この新しい角度 (前処理された後の固有ベクトル軸) θ_2 は、最大固有値と最小固有値の比 (条件数) が小さくなり、固有ベクトルの軸 (長軸) は X1 軸から離れる。元の問題 $\mathbf{A}$ の固有ベクトル軸から回転し、 θ_2 は θ_1 より高い角度に位置する。Fig.6. に表示している前処理後の図に示す様に、楕円に内接する円が徐々に大きくなれば θ_2 が大きくなる。 θ_2 が 90° となると、この場合は楕円に内接する円は、外接する円と同じ所に位置すると仮定する。これは最大固有値と最小固有値が一致する場合であり、つまり条件数が1となる場合である。

上記説明の如く、1つの数値解から次の数値解へは、共役勾配法に見る直交性を保って、次に進むステップは、大きくひしゃげた楕円体より、球に近い楕円体の方が、数値計算の立場から、速いことが容易に理解できる。概略を次に説明する。

[楕円形から円形に近づく場合の数値計算高速性の解釈]

細長い楕円形より円形に近づくほど速く求解することは、次のことで一般的に説明出来る。ここで前処理法に関して、共役という言葉はないが、細長い楕円形と、円形に近い形の収束状況の理解を助ける為、あえて使用する。数値解法の理解の上で、収束のステップとして用いられる楕円形の収束の代表である一般的な逐次法と、軸 (共役) が直交する円形の収束について次に説明する。以下は、共役勾配法の基礎の理解として、良く用いられている平易な事例である。

いま $\mathbf{A}$ を (2×2) 行列とする。Fig.7. のパターンは細長い楕円であり、直交ではない $\mathbf{u}$ と $\mathbf{v}$ があり、これは $\mathbf{A}$ に関して $\mathbf{u}$ と $\mathbf{v}$ は共役であるとする。(2次元平面上的の動きとして) 逐次法による反復計算では、初期値から水平軸に関して、軸と楕円の接点迄進み、そこから垂直軸方向へ進む。次の楕円の接点で再度水平方向へ進む。その繰り返しが行われ、それが少しずつ小刻みになり、最終的に求める解 $\mathbf{x}$ へと収束する。これは一定の打ち切り収束として $\mathbf{x}$ になり、求める数値解(厳密解)になる。これは楕円の形が、細長ければ長いほど、繰り返しが多くなる意味である。

Fig.8. のパターンは、変換の後、軸が直交して円になった場合である。初期値から水平軸に関して、軸と円の接点迄進みその接点から直交して進む。つまり次に垂直軸方向へ進む。(2次元平面の場合であれば、2回のステップで解 $\mathbf{x}$ に収束する。この $\mathbf{x}$ は打ち切り収束としての数値解である。このことは楕円形から円形に近づくにつれて、繰り返しのステップが短く、解に急速に収束することを意味している。

つまり前処理を行うことにより、細長い楕円形が円形に近づいた形となり、数値解の繰り返しのステップが少なくなる。1つのステップの計算時間は同じであるから、繰り返しは少ないほうが時間は短い。つまり収束が速い。従って楕円形から円形近づくほど、高速計算が出来ることを意味している。

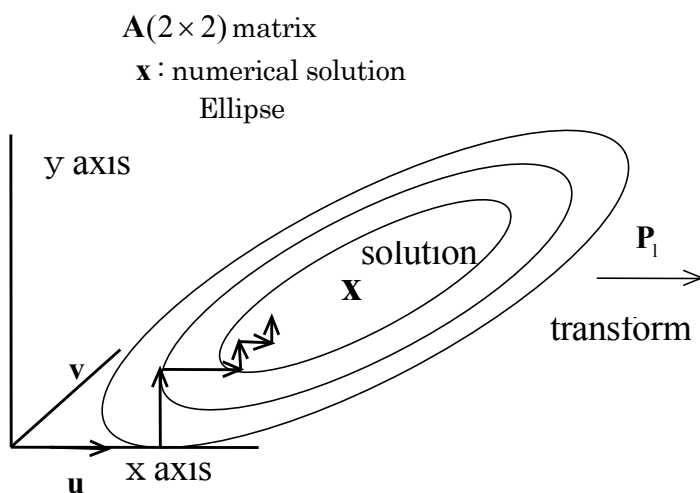


Fig.7. Geometrical Expression for Original Numerical Solution.

$\mathbf{A}$ に関して $\mathbf{u}$ と $\mathbf{v}$ は共役とする。逐次法の形式によるステップの例一般的に数値解の進み方を示す。水平軸 x 方向に進み、そして垂直軸 y 方向に沿った形に進み、更に水平軸 x 方向に、次々に小刻みになって数値解は進み、解 $\mathbf{x}$ に達する。

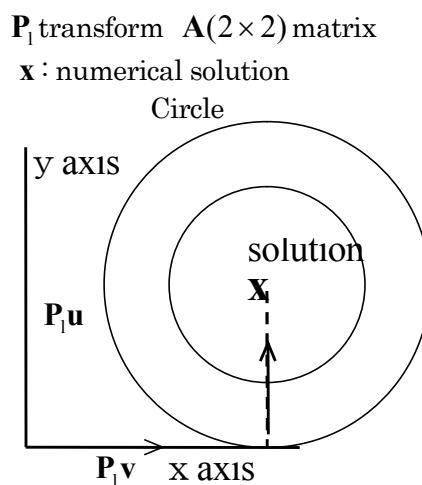


Fig.8. Geometrical Expression for After Orthogonal Transform.

$\mathbf{P}_1$ 変換は共役方向として、直交変換を意味する。 $\mathbf{A}$ に関して $\mathbf{P}_1\mathbf{u}$ と $\mathbf{P}_1\mathbf{v}$ は直交、 2×2 行列の場合、水平軸 x 方向に進み、円との接点で垂直軸 y 方向へ向かう。つまり、数値解は2回の反復で解 $\mathbf{x}$ に達する。つまり円に近づくほど数値解のステップは、短くて済む。

3-4. 高速数値解法に向けての幾何学的解釈結言

この章では、最近研究されてきた前処理法による効率化に関して説明し、Gerschgorin 定理の応用として、固有値、固有ベクトルを使って、これを幾何学的に解釈を進めた。その行列の持つ最大固有値と最小固有値の比率は、これまでも数値計算上影響されることが知られている。この2つの比率が小さくなればなるほど、数値解は求められ易い。幾何学的には最大固有値と最小固有値の比率が、大きくなるとひしやげた楕円形となり、小さくなると円に近い形となる。

Fig.3.に表現されている式を、式 (2.2) の形に前処理として変換し、数値解析を行うことで高速数値解法が実現出来た。つまり元の行列からなる式 (1.5) から構成される式 (3.10) より、式 (2.7) つまり式 (2.2) から構成される式 (3.19) の形の方が、これを幾何学的に解釈すれば、円に近づいた形となることを意味している。前処理とは元の行列にある変換を行うことで、最大固有値と最小固有値の比率が小さくなり、元の行列から構成されるひしやげた楕円形から比較的円に近い楕円形になり、数値解法のステップが少なくなる事を意味している。これを幾何学的立場から紹介し、2次元平面上で説明を行った。

4. まとめ

大規模な連立一次方程式を少しでも速く解法したい。それには前処理と言う方法が存在している。数値解法における前処理の効果とはどのようなものであるか。対象とした磁気流体二流体混合型から構成される電磁流体力学シミュレーションを高速化する上で、前処理を伴った CG 法系に改良すると、明らかに高速性が実現出来た点である。今回の論文では、前処理が高速数値解法を実現する事について、これを視覚に訴える形として幾何学的な立場から研究した。

高速数値解法に関して、最近では前処理を使った多くの解法が研究されており、幾つもの新しい解法が、次々と提案されている。そして前処理を行った後における CG 法系や、CR 法系等による高速計算結果は、多く報告されている。どのモデルにどのような数値解法を当てはめると、正確かつ安定的に収束して効率的であるのかについては、今後の検討課題とされている。それはモデルの持つスパース性や、規則不規則性や、対称非対称や、対角優位性等にも解法の適否がある。

実用面における最適解法の選択において収束に関する件では、行列の持つ条件数や、更に固有値分布状況の良し悪しにも影響され、打ち切り値や相対残差ノルム導入にも依存し、様々な要因が絡んでいる。そうした中であって、高速計算を目指すにおいて、対象とした実モデルが、極端にいびつな性質（条件数が極端に大きい等）でない一般的なものであるとして、それが大規模疎行列であると、適切な前処理を行うことで、確実に高速計算が期待出来る。

そこで何故前処理をすると速くなるのか、このことを知る上で、幾何学的に解釈する事は、数値解析を行う立場から大変重要な意味を持つものと考えられる。大規模疎行列の数値解析を行う上で、前処理を実施して速く収束することは、数値解析に関心の持つ人に向けて、今回研究した前処理の幾何学的解釈の考えが役立つと考えられる。それは今後、様々な工学分野において数値解析が必要とされるが、その数値解析を行う人に対して、その前処理による高速性実現の理解を高めることが、期待出来ると思われる。

参 考 文 献

- [1] A.E. Walter, A. B. Reynolds, "FAST BREEDER REACTORS", Pergamon Press, 1981.
- [2] 浅井忠一他著, 原子力ハンドブック(新版), オーム社, 1989.
- [3] 有馬朗人, スーパーコンピュータ, 日本物理学会編, 培風館, 1985.
- [4] Axelsson, O., Solution of Linear Systems of Equations, Lecture Notes in Mathematics, 572, Springer-Verlag, 1977.
- [5] Doi, S. and Harada, N., Tridiagonal Approximate Factorization Method : A Preconditioning Technique for Solving Nonsymmetric Linear Systems Suitable to Supercomputers, National Aero Space Laboratory, Special Paper 7, 1987. pp.143-149.
- [6] E. Wachspress, Iterative Solution of Elliptic Systems, Prentice-Hall, Englewood, Cliffs, N. J., 1966.
- [7] F. Chatelin, 伊理正夫他訳, 行列の固有値, シュプリンガー・フェアラーク東京, 1993.
- [8] 藤野清次, 竹内敏己, 差分スキームの再考によるベクトル計算機向き不完全LU分解について, 日本応用数理学会論文誌, VOL.4, No.2, 1994.
- [9] 藤野清次, 張紹良, 反復法の数理, 朝倉書店, 1996.
- [10] Gentzsch, W., A fully vectorizable SOR variant, Parallel Computing, 4, 1987.
- [11] Gustafsson, I., A Class of First Order Factorization Method, BIT, No.18, 1978.
- [12] H.A. Van Der Vorst, BiCGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of non-symmetric linear systems, SIAM J. Sci. Comput., 13, 1992.
- [13] I. Gustafsson : BIT, 18(1978), pp.142-156.
- [14] 磯田和男, 大野豊, FORTRANによる数値計算ハンドブック, オーム社, 1982.
- [15] J.J. Dongarra, I.S. Duff, D.C. Sorensen and van der Vorst, H.A., Solving Linear Systems, on Vector and Shared Memory Computers, SIAM, Philadelphia, 1991.
- [16] M.H. Gutknecht, A Completed Theory of the Unsymmetric Lanczos Process and Related Algorithm, Part I, SIAM J. Matrix Anal. Appl., 13, (1992).
- [17] 森正武, 数値解析法, 朝倉書店, 1984.
- [18] 森正武, 数値解析, 共立数学講座 12, 共立出版, 2002.
- [19] 森正武, 杉原正顕, 室田一雄, 線形計算(岩波講座, 応用数学), 岩波書店, 1994.
- [20] M. R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Res. Nat. Bur. Standard vol. 49, (1952), pp.33-53.
- [21] 村田健郎, 小国力, 唐木幸比古, スーパーコンピュータ, 丸善, 1985.
- [22] 村田健郎, 小国力, 三好俊郎, 小柳義夫, 工学における数値シミュレーション, 丸善, 1988.
- [23] 名取亮, 野寺隆, 大規模行列における反復解法, 情報処理, Vol.28, No.11, 1987.
- [24] R. Froehlich, in Mathematical Models and Computational Techniques for Analysis of Nuclear Systems, USAEC CONF-730414-P2, (1973), V.1-1.
- [25] Richard Barrett, 長谷川秀彦, 藤野清次他訳, 反復法 Templates, 朝倉書店, 1996.
- [26] R.S. Varga, Matrix Iterative Analysis, Prentice-Hall, Englewood Cliffs, N.J., 1962.
- [27] 張紹良, 藤野清次, ランチョス・プロセスに基づく積型反復解法, 日本応用数理学会論文誌, VOL.5, No.4, (1995), pp.343-360.