ポアソン方程式の高収束解に関する最適化ファジィ制御則*

東 尚史^{*1}, 小河原加久治^{*2}, 飯田 誠一^{*2}Optimizations of Fuzzy Controller for Solving Poisson Equations
with Fast Convergence

Hisanobu AZUMA, Kakuji OGAWARA and Sei-ichi IIDA

Fuzzy control was applied to speed up the convergence of a Poisson equation solver for the numerical calculations of incompressible fluid. The optimizations of the control rule were carried out by means of a hybrid method of the conjugate gradient method and the improved random walk method. The residual of the equation and the relaxation parameter are employed as input for the controller. It is found that this method is effective in obtaining fast convergence. Moreover, it is clarified that this control does not affect the bifurcation characteristics of the solution. The changes in optimized relaxation parameter are similar to Bang-Bang control.

Key Words: Numerical Analysis, Fuzzy Control, Finite Difference Method, Conjugate Gradient Method

1. まえがき

非圧縮粘性流体の楕円型方程式を数値的に求める際、ポアソン方程式を解く必要がある。その解法はコレスキ法やクラウト法に代表される直接法⁽¹⁾とSOR法(Successive over relaxation method), CG法(Conjugate gradient method)などの反復法⁽²⁾に大別できる。一般にはコンピュータのメモリ容量の制約によって反復法が用いられることが多く、流れの数値解析に要する計算時間の多くはこの反復計算に費やされる。そのため連立一次方程式の高収束解法について多くの研究⁽³⁾がなされており、高ベクトル化率が得られるアルゴリズムが工夫⁽³⁾されている。特に非定常流れの数値計算を行う際には、ポアソン方程式解法の高速化が必須である。

流れの数値計算にSOR法を用いると、急加速時と解が分岐する際に反復回数が著しく増加し、収束性が悪化する傾向が見られる。CG法に基づくICCG法(Incomplete Choleski conjugate gradient method)を用いて収束速度の鈍化を免れる方法もあるが、ICCG法はSOR法に比較してメモリ容量を多く必要とする欠点がある⁽⁴⁾。精度の高い計算を行おうとすれば格子数は必然的に増え、メモリ容量の制約は厳しいものとなる。メモ

リ容量の確保が困難な場合にはSOR法で効率よく解を求める方策がとられる。SOR法の中でも点SOR法はアルゴリズムが単純でベクトル化も容易に行えるため、広く用いられており、この方法での高速化に関する研究も多い⁽⁵⁾⁽⁶⁾。特に高橋⁽⁶⁾はファジィ推論を用いて緩和係数を制御してその効果を示している。また二相流の数値計算の時間刻み制御にもファジィ推論を適用し、その有効性を明らかにしている⁽⁷⁾。

本研究では非定常計算でのポアソン方程式の高収束解法に主眼を置き、特に計算開始の急加速時および解の分岐に伴う収束性低下の改善を主目的としている。そこで二次元キャビティ流れを例として共役勾配法⁽⁸⁾と改良型ランダム探索法⁽⁹⁾を用いて、ファジィ制御則の同定を行い、ポアソン方程式収束向上のための緩和係数を詳細に調べた。長方形領域のポアソン方程式を等間隔格子を用いて解析する場合、最適緩和係数が最大固有値から理論的に求められる⁽¹⁾ので、理論値を用いた収束計算との比較も合わせて示す。また、緩和係数 ω の操作が解の分岐に与える影響についても確認を行った。一般に、ファジィ制御の入力、すなわち反復計算システムの状態(収束性の良否)を判断するものとして方程式残差値と残差勾配値が使われている⁽⁶⁾。その制御則は残差の減りが鈍くなったなら ω を小さくするというものである。すなわち残差勾配値 σ が負で、絶対値が小さくなったなら ω を減らすという制御アルゴリズムである。ところが人間が制御する場合を考えると、現在の ω 値を念頭においていることは容易

* 原稿受付 平成4年1月20日。

^{*1} 正員、北海道大学大学院(〒060 札幌市北区北13条西8丁目)。^{*2} 正員、北海道大学工学部。

に想像がつく。つまり残差勾配が負で絶対値が小さく、 ω の値が小さい場合はわずかに ω を減らせという出力になるであろうし、 ω の値が大きな場合には大きめに ω を減らせという出力になるであろう。SOR法では緩和係数 $\omega < 2.0$ でなければ収束性が数学的に保証されず、 ω が小さすぎても収束性が非常に低下する。そのため本研究では収束計算の安定性を維持するため、 ω 値もファジィ制御の入力として採用し、フィードバック系とした。また残差と残差勾配値の両方をファジィ制御の入力として用いてみたが、残差勾配値の最大値は一般には事前にわからないうえ、繰り返し計算ごとの変動が大きいため、一般化しにくく、残差勾配値は入力としてあまり適当ではないことが判明した。そこで方程式残差と緩和係数の二つを入力とした制御を行い、併せて検討した。

2. ファジィ制御による高速化法

2・1 ファジィ制御則 ファジィ推論法については他の文献^{(4) (10)}に詳しく述べられているので、概略のみを示す。まず n 個の入力 $x_1 \sim x_n$ があるものとする。前件部でメンバーシップ関数を規定し、それら全てに対してファジィ変数を対応付ける(図1)。次に後件部では制御規則によって、それぞれのファジィ変数の組み合わせに対する出力を以下のように決定する。

制御則 i

if x_1 is A_1^i and x_2 is $A_2^i \cdots$ and x_n is A_n^i

then $y_i = f_i(\vec{x}) \cdots \cdots (1)$

ここで「if～」の部分は前件部であり、「then～」の部分は後件部である。また A_n^i はファジィ変数を表し、右上下の添え字 i 、 n はそれぞれ n 番目の入力パラメータの i 番目のファジィ変数を示す。また f_i は

$$f_i(\vec{x}) = a_0^i + a_1^i x_1 + a_2^i x_2 + \cdots + a_n^i x_n \cdots \cdots (2)$$

という線形式である。

入力 $\vec{x}$ に対する前件部の適合度 ω_i は

$$\omega_i = A_1^i(x_1) A_2^i(x_2) \cdots A_n^i(x_n) \cdots \cdots (3)$$

のように積で定義される。(1)式から個々の規則による

図1 ファジィ変数

推論結果を求め、次のように重み付き平均値をとって全体の推論値とする。

$$y_o = \frac{\sum \omega_i f_i(\vec{x})}{\sum \omega_i} \cdots \cdots (4)$$

2・2 ファジィ制御則同定 ファジィ制御則の同定は前件部、後件部それぞれの構造同定とパラメータ同定にわけられる。後件部パラメータ同定には共役勾配法の一つであるFletcher-Reeves法⁽¹¹⁾と改良型ランダム探索法を用いた。前者の方法は局所的な最小値は効率よく求められるが、大域的な最小値への収束は保証されていない。そこで、この短所を補うために改良型ランダム探索法を併用して目的関数の大域的な最小値を求めるハイブリッドアルゴリズム⁽⁹⁾を採用した。ここの目的関数は反復計算の回数に対応しており、これを最小にする式(2)中の a_j^i を探索する。繰り返し計算回数を $N(a_j^i)$ として、Fletcher-Reeves法のアルゴリズムを示す。

Step 1. 初期値 a_0 を定めて、

$$g_0 = \left. \frac{\partial N}{\partial a} \right|_{a=a_0} \text{ を求め、}$$

$$d_0 = -g_0 \text{ とおく。}$$

Step 2. $k=0, 1, \dots, n$ に対して、以下の計算を行う。

$$i) a_{k+1} = a_k + \alpha_k d_k \cdots \cdots (5)$$

ただし α_k は $N(a_k + \alpha_k d_k)$ を最小にするスカラー。

$$ii) g_{k+1} = \nabla N(a_{k+1})^T \cdots \cdots (6)$$

iii) $k=n$ でなければ、

$$d_{k+1} = -g_{k+1} + \beta_k d_k$$

$$(\beta_k = g_{k+1}^T g_{k+1} / g_k^T g_k) \cdots \cdots (7)$$

Step 3. a_{k+1} を a_0 とし、Step 1. へもどる。

Step 2.(i)での一次元探索には黄金探索法⁽¹¹⁾を用いた。改良型ランダム探索法のアルゴリズムについては文献(9)を参照されたい。

このシステムは不等号制約条件付きのシステムである。制約条件については後述することにするが、制約条件のあるシステムの最適化問題はペナルティ関数法や一般縮小勾配法などにより解くことができる⁽⁸⁾。本研究ではペナルティ関数法の外点法を用い、多少のパラメータの調整は必要であるが、計算を破綻させることなく、探索を行うことができた。ペナルティ関数外点法とは、制約条件近くで探索を行う場合(図2)、制約条件を満たさないところではペナルティを課し、最小値を見いだすものである。目的関数を $N(a)$ 、制約条件を $\phi_j < 0.0$ ($j=1, 2, \dots, m$)としたとき、この条件を満たさないときにはペナルティを課した次の式を考え、最小化を行う。

$$U(a, \delta) = N(a) + \delta \sum_{j=1}^m \phi_j(a) \cdots \cdots (8)$$

ここで、第二項目がペナルティ関数（一般に $q=2$ ）である。この方法では制約条件外のところにペナルティを課し最小点を探すものであるが、問題点は δ の値によって制約条件外の値を解としてしまう可能性があることである。そこで図2に示すように δ を大きくしていくと最適値が見いだすことが可能となる。

3. 結果

3.1 計算対象 流れ場は上壁のみが無次元速度1の一定速度で水平方向に移動し、他の壁面は静止している二次元キャビティ流れとした。座標系および境界条件は図3に示すとおりである。格子には格子数 20×30 の等間隔格子を用い、アスペクト比 $\Gamma (=H/W)$ は1.5として計算を行っている。このアスペクト比のもとでは、レイノルズ数の増加とともに一つの渦の状態から、その下部にもう一つの渦を誘起して二つの渦の状態に分岐する現象が現れる^(1,2)。分岐とはパラメータの変化によって解の性質が変化することである。本論文の目的は解が分岐する際のポアソン方程式の収束性悪化の改善と緩和係数の制御が解の分岐に影響を与えるかどうか、すなわち反復計算回数の低減により精度の低下を招くことがないかを探ることにある。そのため、 Γ には分岐が起こりやすい $\Gamma=1.5$ を選択した。

計算方法は差分法を用いた渦度-流れ関数（ $\omega-\psi$ ）法により行った。対流項には一次精度の風上差分を用い、すべての壁面上で滑りなし条件とした。時間微分項には一次精度前進差分を、対流項を除く空間微分項には二次精度中心差分を用いた。初期条件としては流体がすべて静止しているものとしてインパルススタートを用い、時間刻みは $CFL=0.5$ となるように計算を進めた。また、ポアソン方程式は、点SOR法により解き、流れ関数の残差の絶対値が 1.0×10^{-8} 未満となったら収束したものと判断し、繰り返し計算を打ち切った。

図4、図5に $Re=2000$ における等流れ関数線図と計算時間ごとの繰り返し計算回数を示す。図5の実線は $\omega=1.6$ 、破線は $\omega=1.765$ を用いて計算した結果である。後者はSOR反復行列のスペクトル半径を最小化し、反復計

算数を最小とする理論値である。図より無次元時間 $T^*=9.0$ のところで繰り返し計算回数が増大しているのが認められるが、これは一つの渦の状態から二つの渦の状態へ解が分岐したことに起因している。

3.2 後件部の同定 本研究では計算開始直後の急加速時と解の分岐に伴う収束性低下の改善が目的

図3 座標系および使用記号

図4 キャビティ流れ $\Gamma=1.5$, $Re=2000$
等流れ関数線図

図5 キャビティ流れ $\Gamma=1.5$, $Re=2000$
無次元時間 T^* に対する繰り返し計算回数 N の変化

図2 ペナルティ関数

の一つである。制御アルゴリズムは経験などから求めるもので図6に残差、残差勾配、緩和係数を入力とする制御規則表を示す。この制御則アルゴリズムが、図5の $T^*=0.075, 9.0$ の繰り返し計算回数を効率的に低減し得るかどうか確かめておく必要がある。そこで、まず前件部では方程式残差、残差勾配、緩和係数の3個の入力パラメータを $[0, 1]$ の間で正規化し、簡単のため3つの[small, medium, large]のラベルを持つ三角形型ファジィ変数で規定する。方程式残差は $T^*=0.075$ における残差の最大値と収束基準値(1.0×10^{-8})を用い、また残差勾配についても $T^*=0.075$ における最大、最小残差勾配値を用いて正規化を行う。緩和係数については不足緩和の0.8から緩和係数最大値の2.0で正規化することにする。なお、残差勾配値は繰り返し計算ごとの変動が非常に大きいので、3点の残差の勾配を最小二乗法により求めて使用した。

後件部は前述したハイブリッドアルゴリズムによって最適化を行う。ただし、これはあくまでも前件部の制約下での最適化である。Fletcher-Reeves法の初期ベクトルを決めなければならないが、図6の制御規則表から推論される値を求めて与えた。また、最適化の計算を破綻させずに進めるため、緩和係数が2.0を越えたり、残差勾配が正とならないよう、これらを制約条件として組み入れて、目的関数は以下のように定義した。

$$U(a, \delta_1, \delta_2) = N(a) + \delta_1 \sum_{i=1}^N (\omega_i - 2.0)^2 + \delta_2 \sum_{i=1}^N \sigma_i^2 \dots \dots \dots (9)$$

ここに、 $N(a)$ は繰り返し計算回数を表し、 ω_i, σ_i はそれぞれ繰り返し*i*回目の緩和係数と残差勾配値を表す。また δ_1, δ_2 はそれぞれ $\omega_i > 2.0, \sigma_i > 0$ の時にある定数値となり、それ以外はゼロである。

まず、 $T^*=0.075, 9.0$ の両方に共通する制御則を探るため、以下のような目的関数を用いる。

$$U(a) = U(a) \Big|_{T^*=0.075} + U(a) \Big|_{T^*=9.0} \dots \dots (10)$$

図6 制御規則表(残差 γ , 残差勾配 σ , 緩和係数 ω)

上式は $T^*=0.075, 9.0$ の繰り返し計算回数の和をとる。多目的評価関数である。この関数を最小化した結果を以下に示す。

図7、図8に残差に対する緩和係数の変化と繰り返し計算に対する残差の変化を示す。図7の横軸の残差は対数スケールであり、繰り返し計算回数に対応すると考えてよい。また、残差勾配を計算しているため、最初の5回の計算では緩和係数1.6で一定である。この図で、 $T^*=0.075$ の緩和係数が最初の残差の大きなところでありあまり増加していないことがわかる。これは両者の残差(横軸スケール)を $T^*=0.075$ の最大残差値で正規化したことに起因する。 $T^*=0.075, 9.0$ の両者ともに繰り返し計算の初めから緩和係数を増加させるためには、それぞれ $\log(\gamma) = -5 \sim -6, -6 \sim -7$ で ω を増加せよという出力となる。そうすると、 $T^*=0.075$ の ω は $\log(\gamma) = -5 \sim -7$ まで増え続け、2.0を越えて制約条件外になってしまう。この結果は時間刻みごとにその最大残差値(事前にわかりえないので、繰り返し計算1回目の残差で近似)と収束基準値で残差の正規化を行う必要があることを示唆するものである。そのためには正規化した残差に対する緩和係数の変化が、 $T^*=0.075$ と9.0のそれぞれにほぼ最適である制御則で、ある程度相似性をもっていることが必要と思われる。次に上記のように各々の1回目の繰り返し計算の残差を用いて正規化

図7 残差 γ (対数スケール)に対する緩和係数 ω の変化

図8 繰り返し計算回数*N*に対する残差 γ (対数スケール)の変化

を行う方法に改め、 $T^*=0.075, 9.0$ のそれぞれについて最適化を行った結果を図9、図10に示す。図9の横軸は正規化した残差を、縦軸には緩和係数を示してある。図10は繰り返し計算ごとの対数スケールの残差と $T^*=9.0$ の残差勾配を示す。この図において $T^*=0.075$ の緩和係数が残差の小さなところで、フィードバックのため値が上下している。図10より残差の発振はさほど大きくなく、残差をやや発振ぎみに減らしたほうが速く収束する場合があることがわかる。また、図9からほぼ十分な相似性が得られていると考えられ、これらのことから先に示したファジィ制御則が残差については妥当であることがわかった。残差勾配値については繰り返し計算開始直後の緩和係数増加時と後半の緩和係数減少時に鋭い極値をもつだけであまり変化がない。この二つの極値はそれぞれ緩和係数の増加と減少に起因しており、一概に残差勾配が負でその絶対値が小さいときは緩和係数を減らすという制御則は成り立たないと考えられる。今回の制御則において、残差勾配値は入力として不適当であると思われるが、これをルール修正のパラメータとして用いている例はある⁽¹³⁾。

3・3 ファジィ制御則の最適化 以上の結果より、入力として残差と緩和係数のみを用い、残差勾配を除外した制御規則表を図11に示す。前件部について

は先に述べたとおり、繰り返し計算1回目の残差を用いて正規化し、先に得られた結果を参考にしてファジィ変数を規定した。後件部については図11をもとに式(6)のパラメータを与え、前述のハイブリッド法を用いて最適化を行った。目的関数としては $T^*=0.075, 9.0$ の両方の繰り返し計算回数の和をとり、最小化を行った。その結果を図12に示す。図12はほぼ定常状態まで計算を続けた場合の時間刻みごとの繰り返し計算回数を示すものであり、後件部の最適化を行ったもの、行わないもの、いずれも理論的最適値を用いた場合よりも良い収束性が得られていることがわかる。なお、あらかじめ制御則をマップで与えているため、制御のための計算時間はほとんど要しない。次に $T^*=0.075$ の繰り返し計算ごとの緩和係数と方程式残差の変化を図13、図14に示す。これらの図から、後件部の最適化を行うと緩和係数の変化が急峻になり、Bang-Bang制御（最短時間制御）⁽¹⁴⁾のようになって、高収束率が得られていることがわかる。また、残差がある程度小さいときに、緩和係数を減じると残差が急激に減少するようすが見られる。緩和係数を操作して繰り返し計算回数を減じたときに、計算精度に悪影響をもたらすことがないことを調べるため、流体の運動エネルギーの変化を調べた。緩和係数1.6と固定した場合と比較を行ったと

図9 残差 γ （正規化）に対する緩和係数 ω の変化

図11 制御規則表（残差 γ ，緩和係数 ω ）

図10 繰り返し計算回数 N に対する残差 γ （対数スケール）、残差勾配 σ の変化

図12 無次元時間 T^* に対する繰り返し計算回数 N の変化

図13 繰り返し計算回数Nに対する緩和係数 ω の変化 図14 繰り返し計算回数Nに対する残差 γ (対数スケール) の変化

ころ、結果に全く差異はなく(0.0006%)、数値誤差の範囲程度であった。

図11の制御規則の一般性を探るため、軸対称回転二重円筒間流れに、この制御規則を適用した。不等間隔格子を用いた、差分のSMAC(Simplified Marker and Cell)法により解析を行った。なお、内円筒回転速度が時間とともに変化する非定常流れを考え、収束条件はキャビティ流れよりも厳しいものとした。ファジィ制御の効果は $\omega=1.4$ と比較して、計算時間で約60%程度となり、この制御規則の妥当性が確かめられた。

4. 結論

ポアソン方程式の高収束解法として、点SOR法の緩和係数の制御にファジィ制御を適用した。流れ場は2次元キャビティ流れであり、渦度-流れ関数法によって近似解を求めた。また、ファジィ制御則の後件部はハイブリッド法を用いて最適化を行った。以下に本研究で得られた結果を示す。

(1) 緩和係数にファジィ制御を適用して計算を行った結果、この方法は解の分岐等に影響を与えず(精度の悪化をもたらすことなく)、収束性の向上がはかれることがわかった。

(2) ファジィ制御の入力として方程式残差と緩和係数を用いる方法を示し、その有効性を確認し、残差勾配値を使う方法よりロバスト(強靱)であることがわかった。

(3) 後件部パラメータの最適化を行うことにより、緩和係数は繰り返し計算の後半で小さな値になり、Bang-Bang制御的に変化して高収束率が得られることが判明した。

文 献

- (1) 戸川, マトリクスの数値計算(1986), オーム社, 52.
- (2) 河原, 第4回数値流体シンポジウム講演論文集, (1990), 447.
- (3) 藤野・竹内, 第4回数値流体シンポジウム講演論文集, (1990), 415.
- (4) 矢川ほか, ファジィ推論(1991), 培風館, 56.
- (5) 田中・高橋, 機論, 57-540, B(1990), 2656.
- (6) 高橋, シミュレーション, 7-2(1988), 2.
- (7) 高橋・富山, 日本原子力学会誌, 29-89(1987), 732.
- (8) 赤木, AI技術によるシステム設計論(1988), コロナ社, 153.
- (9) 馬場, FANシンポジウム講演論文集, 910-70(1991), 47.
- (10) 菅野, ファジィ制御(1988), 日刊工業新聞社, 88.
- (11) 馬場・坂和, 数理計画法入門(1989), 共立出版, 100.
- (12) 円山・東・小河原・飯田, 機論, No. 912-1(1991-10), 159.
- (13) 文献(4)の81ページ.
- (14) 文献(4)の38ページ.