

BiCR 法の残差をベースにした BiCRSafe 法の収束性評価

藤野 清次 尾上 勇介 †,††

本論文では、従来のように BiCG 法の残差ベクトルではなく、BiCR 法の残差ベクトルに基づく BiCRSafe 法の変形版を提案する。提案する BiCRSafe 法はよく利用されている積型反復法の拡張とも見做すことができる。この変形版の有効性をいくつかの疎行列に対する実験を通して調べることにする。得られた数値実験結果は、前処理つき BiCRSafe 法が電磁界分野で現れた実際の問題に対して非常に頑丈で効率のよい反復法であることを示している。

Estimation of BiCRSafe method based on residual of BiCR method

SEIJI FUJINO YUSUKE ONUUE †,††

In this paper we propose a variant of BiCGSafe method which is constructed on the residual of BiCR method in place of that of BiCG method. This variant can be also defined to an extension of the useful product-type iterative methods. The effectiveness of this variant is studied through the experiments of a suite of sparse matrices. Our results demonstrate that preconditioned BiCRSafe method is very robust and efficient among the conventional iterative methods for realistic problems in the electromagnetic field.

1. はじめに

$n \times n$ の正則な疎行列 A を係数行列、 n 次元の右辺ベクトル b と解ベクトル x とする連立 1 次方程式

$$Ax = b \quad (1)$$

を反復法で解くことを考える。特に、規模の大きな行列を係数行列として持つ連立一次方程式の求解においては、クリロフ (Krylov) 部分空間法がよく利用される。クリロフ部分空間法に属する反復法には多くのものがあるが⁸⁾、本研究では、Bi-Conjugate Gradient (双共役勾配、以下 BiCG と略す) 法と Bi-Conjugate Residual (双共役残差、BiCR と略す) 法を基本反復法と位置付ける。ここで、「基本」とは、様々な BiCG 法系統の反復法において現れるパラメータ α_k, β_k の値を決めるとき、その決定を元の BiCG 法の残差ベクトル r_k^{BiCG} に基づいて行うのではなく、BiCR 法の残差ベクトル r_k^{BiCR} に基づいて行う、という意味で使用する。一般に、これらの BiCG 法系統の反復法は積型 (product-type) 反復法とも呼ばれる。積型反復法では、その残差ベクトル r_k は、BiCG 法の収束性を加速する加速多項式 $H_k(A)$ とランチョス多項式 $R_k(A)$ との積で定義されることもよく知られている。

最近、阿部らは文献²⁾において、CGS (Squared) 法の残差ベクトル r_k^{CGS} を、BiCG 法の残差ベクトル r_k^{BiCG} から BiCR 法の残差ベクトル r_k^{BiCR} に置き換えた新しい反復法: CRS (Conjugate Residual Squared) 法が考案された。CRS 法は、従来の CGS 法と比較して、収束が速くかつ収束性が安定、という非常に優れた特長を有することが数値実験で明らかにされた。そこで、本論文の目的は、BiCGSafe 法の残差ベクトルに対して、BiCG 法の残差ベクトル r_k^{BiCG} から BiCR 法の残差ベクトル r_k^{BiCR} に置き換えた BiCRSafe 法を導出し、その収束性を数値実験にて明らかにすることである。

本論文は次のように構成されている。第 2 節で、まず積型反復法の概要について記述し、次に積型 BiCR 法におけるパラメータ α_k, β_k の決定法について記述する。第 3 節では、BiCRSafe 法の漸化式の導出を行う。第 4 節では、数値実験で、提案した BiCRSafe 法の収束性を従来の GPBiCG, BiCGStab, BiCGSafe 法と比較し、BiCRSafe 法の収束性の剛健さと効率のよさを明らかにする。最後にまとめを第 5 節で行う。

2. 積型反復法について

2.1 積型 BiCG 法の概要

BiCG 法の残差ベクトル r_k^{BiCG} は
$$r_k^{\text{BiCG}} = R_k(A)r_0 \quad (2)$$

と表せる。残差多項式 $R_k(A)$ は、BiCG 法の 2 つの

† 九州大学情報基盤研究開発センター Research Institute for Information Technology, Kyushu University

†† 九州大学大学院システム情報科学府 Graduate School of Information Science and Electrical Engineering, Kyushu University

パラメータ α_k, β_k から定まり、そしてランチョス多項式と呼ばれる次の 3 項漸化式を満たす。

$$R_0(\lambda) = 1, \quad (3)$$

$$R_1(\lambda) = 1 - \alpha_0 \lambda, \quad (4)$$

$$\begin{aligned} R_{k+1}(\lambda) &= (1 + \alpha_k \frac{\beta_{k-1}}{\alpha_{k-1}} - \alpha_k \lambda) R_k(\lambda) \\ &\quad - \alpha_k \frac{\beta_{k-1}}{\alpha_{k-1}} R_{k-1}(\lambda), \quad k = 1, 2, \dots \end{aligned} \quad (5)$$

BiCG 法の残差列 $r_0^{\text{BiCG}}, r_1^{\text{BiCG}}, \dots, r_k^{\text{BiCG}}, \dots$ と記すとき、 k 次の多項式 $H_k(\lambda)$ を用いて $H_k(A)r_k^{\text{BiCG}}$ が新たな残差ベクトルになるように BiCG 法の収束性を加速したのが積型 BiCG 法である。すなわち、BiCG 法に基づく積型反復法における第 k 回目の反復の残差ベクトル r_k は、BiCG 法の収束性を高めるためのある k 次多項式 $H_k(\lambda)$ とランチョス多項式 $R_k(\lambda)$ 用いて次のように定義する。

$$r_k = H_k(A)R_k(A)r_0.$$

ここで、 r_0 は初期残差ベクトルと呼ばれ、初期近似解ベクトル x_0 、右辺ベクトル b を用いて、 $r_0 := b - Ax_0$ と定義される。ただし、ランチョス多項式 $R_k(\lambda)$ は補助多項式 $P_k(\lambda)$ を用いて次の交代漸化式を満たす。

$$R_0(\lambda) = 1, \quad P_0(\lambda) = 1, \quad (6)$$

$$R_{k+1}(\lambda) = R_k(\lambda) - \alpha_k \lambda P_k(\lambda), \quad (7)$$

$$P_{k+1}(\lambda) = R_{k+1}(\lambda) + \beta_k P_k(\lambda), \quad k = 0, 1, \dots \quad (8)$$

さらに、ランチョス多項式のパラメータ α_k は、双直交条件 $(r_i, r_j) = 0, (i \neq j)$ によって求められる。

$$\alpha_k = \frac{(r_k, r_k^*)}{(Ap_k, p_k^*)} \quad (9)$$

同様に、もう 1 つのパラメータ β_k は、A-共役条件: $(Ap_i, p_j^*) = 0, (i \neq j)$ によって、

$$\beta_k = \frac{(r_{k+1}, r_{k+1}^*)}{(r_k, r_k^*)} \quad (10)$$

と求められる。

2.2 積型 BiCR 法における α_k, β_k

まず、元の BiCR 法の残差ベクトル r_k は

$$r_k = R_k(A^T)r_0 \quad (11)$$

と表せ、BiCR 法によって生成されるベクトル r_k^* は

$$r_k^* = R_k(A^T)r_0^* \quad (12)$$

と表せる。一方、BiCR 法のパラメータ β_k は、

$$\beta_k = \frac{(Ar_{k+1}, r_{k+1}^*)}{(Ar_k, r_k^*)}. \quad (13)$$

と表せる。ここで、次の内積 ρ_k を考える。

$$\begin{aligned} \rho_k &= (AS_k(A)R_k(A)r_0, r_0^*) \\ &= (AR_k(A)r_0, S_k(A^T)r_0). \end{aligned} \quad (14)$$

ただし、

$$S_k(A^T) = \sum_{i=0}^k s_k^i (A^T)^i \quad (15)$$

$$R_k(A) = \sum_{i=0}^k r_k^i (A)^i \quad (16)$$

とおく。このとき、 $AR_k(A)r_0$ は、 $(A^T)^i r_0^* (i < k)$ と直交するので、

$$\begin{aligned} \rho_k &= (AR_k(A)r_0, s_k^k r_0^*) \\ &= (AR_k(A)r_0, \frac{s_k^k}{r_k^k} R_k(A^T)r_0^*) \\ &= \frac{s_k^k}{r_k^k} (AR_k(A)r_0, R_k(A^T)r_0^*). \end{aligned} \quad (17)$$

したがって、積型 BiCR 法の残差ベクトルを、

$$r_k^{\text{PBCR}} := S_k(A)R_k(A)r_0 \quad (18)$$

と定義すると、パラメータ β_k は、

$$\begin{aligned} \beta_k &= \frac{(AR_{k+1}(A)r_0, R_{k+1}(A^T)r_0^*)}{(AR_k(A)r_0, R_k(A^T)r_0^*)} \\ &= \frac{r_{k+1}^{k+1} s_k^k \rho_{k+1}}{s_{k+1} r_k^k \rho_k} \\ &= \frac{\alpha_k (Ar_{k+1}^*, r_0^*)}{\zeta_k (Ar_k^{\text{PBCR}}, r_0^*)} \end{aligned} \quad (19)$$

と表せる。一方、BiCR 法のパラメータ α_k は、

$$\alpha_k = \frac{(Ar_k, r_0^*)}{(Ap_k, A^T p_k^*)}. \quad (20)$$

と表せる。ここで、補助ベクトルを、

$$p_k^{\text{PBCR}} := S_k(A)P_k(A)r_0 \quad (21)$$

と定義する。このとき、積型 BiCR 法のパラメータ α_k は、

$$\begin{aligned} \alpha_k &= \frac{(AR_k(A)r_0, R_k(A^T)r_0^*)}{(AP_k(A)r_0, A^T P_k(A^T)r_0^*)} \\ &= \frac{(AR_k(A)r_0, R_k(A^T)r_0^*)}{(AP_k(A)r_0, A^T R_k(A^T)r_0^*)} \\ &= \frac{(AR_k(A)r_0, S_k(A^T)r_0^*)}{(AP_k(A)r_0, A^T S_k(A^T)r_0^*)} \\ &= \frac{(AR_k(A)r_0, S_k(A)r_0^*)}{(AP_k(A)S_k(A)r_0, A^T r_0^*)} \\ &= \frac{(Ar_k^{\text{PBCR}}, r_0^*)}{(Ap_k^{\text{PBCR}}, A^T r_0^*)}. \end{aligned} \quad (22)$$

で求めることができる。

3. BiCRSafe 法の漸化式の導出

まず、2 つの多項式の積 $S_{k+1}(A)R_{k+1}(A)$ は次のように変形する。ただし、紙面横幅の制約から多項式 $R_k(A)$ は R_k のように略記する。

$$S_{k+1}R_{k+1} = S_k R_k - \alpha_k AS_k P_k - AG_k R_{k+1} \quad (23)$$

また、式 (23) 中の $S_k(A)P_k(A)$, $G_k(A)R_{k+1}(A)$ は次のように変形される。

$$S_k P_k = S_k R_k + \beta_{k-1} (S_{k-1} P_{k-1} - AG_{k-1} R_{k-1}) \quad (24)$$

$$G_k R_{k+1} = \zeta_k S_k R_k + \eta_k G_{k-1} R_k - \alpha_k A G_k P_k \quad (25)$$

次に、以下の4つの補助ベクトルを導入する。

$$\mathbf{p}_k := S_k(A)P_k(A)\mathbf{r}_0 \quad (26)$$

$$\mathbf{u}_k := AG_k(A)P_k(A)\mathbf{r}_0 \quad (27)$$

$$\mathbf{z}_k := G_k(A)R_{k+1}(A)\mathbf{r}_0 \quad (28)$$

$$y_{k+1} := Az_k. \quad (29)$$

これら4つのベクトルから、BiCRSafe法の漸化式に関する部分が以下のように導かれる。

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A\mathbf{p}_k - Az_k \quad (30)$$

$$\mathbf{p}_k = \mathbf{r}_k + \beta_{k-1}(\mathbf{p}_{k-1} - \mathbf{u}_{k-1}) \quad (31)$$

$$\mathbf{u}_k = \zeta_k A\mathbf{p}_k + \eta_k (\mathbf{y}_k + \beta_{k-1}\mathbf{u}_{k-1}) \quad (32)$$

$$z_k = \zeta_k \mathbf{r}_k + \eta_k \mathbf{z}_{k-1} - \alpha_k \mathbf{u}_k \quad (33)$$

$$y_{k+1} = \zeta_k Ar_k + \eta_k y_k - \alpha_k Au_k. \quad (34)$$

次に、パラメータ ζ_k, η_k の求め方について記述する。元のBiCGSafe法およびBiCRSafe法では、残差ベクトルの更新の式(30)の中に、パラメータ ζ_k, η_k は使われていない。そこで、新たな補助ベクトル a_r_k ($:= H_{k+1}(A)R_k(A)\mathbf{r}_0$) を導入する必要がある。この補助ベクトルを準残差ベクトルと呼ぶ。このとき、準残差ベクトル a_r_k を

$$\begin{aligned} H_{k+1}R_k &= (H_k - AG_k)R_k(A) \\ &= H_kR_k - A(\zeta_k H_k + \eta_k G_{k-1})R_k \\ &= H_kR_k - \zeta_k AH_kR_k - \eta_k AG_{k-1}R_k \end{aligned} \quad (35)$$

と変形できるので、未確定のパラメータ ζ_k, η_k が準残差ベクトル a_r_k に含まれていることがわかる。さらに、式(35)に、式(28), (29)を代入すると、

$$a_r_k = \mathbf{r}_k - \zeta_k Ar_k - \eta_k y_k \quad (36)$$

と表せる。BiCGSafe法とBiCRSafe法では、準残差ベクトル a_r_k のノルムの最小化から、次のように2つのパラメータ ζ_k, η_k の値は決定される。

$$\|a_r_k\|_2^2 = \min_{\zeta_k, \eta_k} \|\mathbf{r}_k - \zeta_k Ar_k - \eta_k y_k\|_2^2. \quad (37)$$

すなわち、パラメータ ζ_k, η_k の値は、

$$\zeta_k = \frac{(\mathbf{y}_k, \mathbf{y}_k)(Ar_k, r_k) - (\mathbf{y}_k, r_k)(Ar_k, y_k)}{(Ar_k, Ar_k)(y_k, y_k) - (\mathbf{y}_k, Ar_k)(Ar_k, y_k)} \quad (38)$$

$$\eta_k = \frac{(Ar_k, Ar_k)(y_k, r_k) - (\mathbf{y}_k, Ar_k)(Ar_k, r_k)}{(Ar_k, Ar_k)(y_k, y_k) - (\mathbf{y}_k, Ar_k)(Ar_k, y_k)}. \quad (39)$$

で具体的に求められる。この式では、行列ベクトルの積 Ar_k の計算が必要になるが、 Ar_k の計算はBiCRSafe法のアルゴリズムには現れない。そこで、 $A\mathbf{p}_k$ の計算は以下の式(41)で示すように漸化式で求めることにする。その結果、BiCRSafe法のアルゴリズムにおける行列ベクトルの積の計算は、反復中の Au_k の計算およびパラメータ ζ_k, η_k の計算中の Ar_k の計算の合計2回になる。BiCRSafe法のアルゴリズムを以下に示す。

BiCRSafe法のアルゴリズム

Let x_0 be an initial guess, $r_0 = b - Ax_0$

Choose r_0^* such that $(r_0, r_0^*) \neq 0, \beta_{-1} = 0$

$$q_0 = Ar_0$$

For $k = 0, 1, \dots$, Do

$$p_k = \mathbf{r}_k + \beta_{k-1}(\mathbf{p}_{k-1} - \mathbf{u}_{k-1}) \quad (40)$$

$$Ap_k = Ar_k + \beta_{k-1}(A\mathbf{p}_{k-1} - Au_{k-1}) \quad (41)$$

$$\alpha_k = \frac{(Ar_k, r_0^*)}{(Ap_k, Ar_k)} \quad (42)$$

$$\zeta_k = \frac{(\mathbf{y}_k, \mathbf{y}_k)(q_k, r_k) - (\mathbf{y}_k, r_k)(q_k, y_k)}{(q_k, q_k)(y_k, y_k) - (\mathbf{y}_k, q_k)(y_k, q_k)} \quad (43)$$

$$\eta_k = \frac{(q_k, q_k)(y_k, r_k) - (\mathbf{y}_k, q_k)(q_k, r_k)}{(q_k, q_k)(y_k, y_k) - (\mathbf{y}_k, q_k)(y_k, q_k)} \quad (44)$$

$$(\text{if } k = 0, \text{ then } \zeta_0 = \frac{(q_0, r_0)}{(q_0, q_0)}, \eta_0 = 0)$$

$$u_k = \zeta_k A\mathbf{p}_k + \eta_k (\mathbf{y}_k + \beta_{k-1}\mathbf{u}_{k-1}) \quad (45)$$

$$z_k = \zeta_k \mathbf{r}_k + \eta_k \mathbf{z}_{k-1} - \alpha_k \mathbf{u}_k \quad (46)$$

$$y_{k+1} = \zeta_k Ar_k + \eta_k y_k - \alpha_k Au_k \quad (47)$$

$$x_{k+1} = x_k + \alpha_k p_k + z_k \quad (48)$$

$$r_{k+1} = \mathbf{r}_k - \alpha_k A\mathbf{p}_k - y_{k+1} \quad (49)$$

$$q_{k+1} = Ar_{k+1}$$

$$\beta_k = \frac{\alpha_k (q_{k+1}, r_0^*)}{\zeta_k (q_k, r_0^*)} \quad (50)$$

End Do

4. 数値実験

4.1 計算機環境と計算条件

計算機は九州大学情報基盤研究開発センターに設置されたIBM eServer p5(CPU: POWER5, クロック周波数: 1.9GHz, メモリ: 2 Terabytes, OS: IBM AIX 5L)を用いた。コンパイラはIBM XL Fortran Compiler ver. 9.1を用い、最適化オプションは-O3-qarch=pwr5-qtune=pwr5-qhotを使用した。プログラムはFortran90で実装した。計算はすべて倍精度浮動小数点演算を行った。時間の計測にはFortranの時間計測関数nclockを用いた。GPBiCG法、GP-BiCR法、BiCGSafe法、BiCRSafe法の4種類の反復法を用いた。行列は予め対角スケーリングにより対角項をすべて1.0に正規化した。収束判定値は相対残差の2ノルム: $\|r_m\|_2 / \|r_0\|_2 \leq 10^{-7}$ とした。初期近似解 x_0 はすべて0、最大反復回数は10000にした。初期シャドウ残差ベクトル r_0^* には、

- 初期残差ベクトル r_0 ,
- 一様乱数,
- 定数 (=1.0)

の3通りを用いた。前処理は加速係数 γ 付き ILU(0) 分解(以下、加速 ILU(0) 分解と呼ぶ)を用いた。

4.2 テスト問題 1

ここで使用した行列の主な特徴を表 1 に示す。これらの行列は、Florida 大学の疎行列データベースから選出した⁴⁾。

表 1 テスト問題 1 の行列の主な特徴

行列	解析分野	次元数	非零要素数	平均非零要素数
EX10HS	流体解析	2548	57308	22.49
EX19	流体解析	12005	259577	21.62
LI	磁性流体	22695	1215181	53.54

表 2(a)-(c) に、3 つの行列 EX10HS, EX19, LI における加速 ILU(0) 分解つき反復法の収束性を各々示す。表中の第 2 欄の「 r_0^* 」は、初期シャドウ残差ベクトル r_0^* について、 $r_0^* = r_0$ の場合、「rand」は r_0^* に一様乱数を代入した場合、「const」は r_0^* に定数 (= 1.0) を代入した場合、の収束結果を各々表す。また、各段の最終行に、初期シャドウ残差ベクトル r_0^* の 3 つの選択ごとの計算時間の平均値を示した。加速係数 γ は、原則 1.0 から 1.30 まで 0.002 刻みで合計 151 回計算を行った。ただし、行列 LI では、1.17 から 1.30 まで 0.002 刻みで合計 66 回計算した。さらに、「成功度数」とは収束に成功した度数 (= 計算回数)、「成功率 (%)」はその比率を各々表す。また、表中の平均値は収束に成功した度数だけを集計したことを明記する。

表に示した結果から以下のことがわかる。

- 行列 EX10HS では、GPBiCG, BiCRSafe 法の各成功度数は、GPBiCG, BiCGSafe 法のそれらと比較して若干向上した。また、GPBiCR, BiCRSafe 法の平均反復回数と同計算時間もかなり向上したことがわかる。
- 同様に、行列 EX19 でも行列 EX10HS の結果とほぼ同じ傾向が現れている。
- 行列 LI では、GPBiCR, BiCRSafe 法の各成功度数は、GPBiCG, BiCGSafe 法と比較してほとんど変化がない。しかし、GPBiCR, BiCRSafe 法の平均反復回数と同計算時間は、GPBiCG, BiCGSafe 法のそれらと比べて大幅に減っていることがわかる。
- 初期シャドウ残差ベクトル r_0^* の選択について、行列 EX10HS と行列 EX19 では、 $r_0^* = r_0$ の場合、 r_0^* に一様乱数を代入した場合、 r_0^* に定数 (= 1.0) を代入した場合の収束性の差はごく僅かだった。一方、行列 LI では、 r_0^* に定数 (= 1.0) を代入した場合の収束性が他の 2 つの場合に比べて最も優れていた。

図 5 に、行列 LI の場合の加速 ILU(0) 分解つき GPBiCG 法の残差履歴を示す。同様に、図 6 に、加速 ILU(0) 分解つき BiCRSafe 法の残差履歴を示す。

表 2 3 つの行列 (a)EX10HS, (b)EX19, (c)LI における加速 ILU(0) 分解つき反復法の収束性

反復法	r_0^*	成功度数	成功率 (%)	ave. itr.	ave. time
GPBiCG	r_0	83	55	332	0.34
	rand	82	54	480	0.48
	const	83	55	414	0.42
GPBiCR	r_0	83	55	347	0.35
	rand	84	56	347	0.35
	const	84	56	288	0.30
BiCGSafe	r_0	87	58	455	0.45
	rand	85	56	498	0.49
	const	86	57	533	0.53
BiCRSafe	r_0	86	57	474	0.47
	rand	85	56	316	0.32
	const	85	56	322	0.33
r_0^* 每の計算時間の平均値	r_0				0.403
	rand				0.410
	const				0.395

(a) 行列 EX10HS の場合

反復法	r_0^*	成功度数	成功率 (%)	ave. itr.	ave. time
GPBiCG	r_0	94	62	343	1.73
	rand	91	60	743	3.64
	const	88	58	444	2.22
GPBiCR	r_0	95	63	550	2.74
	rand	96	64	542	2.70
	const	96	64	468	2.33
BiCGSafe	r_0	96	64	458	2.24
	rand	92	61	401	1.97
	const	94	62	671	3.25
BiCRSafe	r_0	97	64	484	2.40
	rand	96	64	397	1.99
	const	96	64	426	2.12
r_0^* 每の計算時間の平均値	r_0				2.278
	rand				2.575
	const				2.480

(b) 行列 EX19 の場合

反復法	r_0^*	成功度数	成功率 (%)	ave. itr.	ave. time
GPBiCG	r_0	56	85	649	20.00
	rand	58	88	128	4.69
	const	58	88	125	4.44
GPBiCR	r_0	58	88	70	2.92
	rand	58	88	62	2.66
	const	58	88	51	2.31
BiCGSafe	r_0	58	88	579	17.83
	rand	58	88	130	4.70
	const	58	88	128	4.54
BiCRSafe	r_0	58	88	76	3.10
	rand	58	88	64	2.75
	const	58	88	52	2.35
r_0^* 每の計算時間の平均値	r_0				10.96
	rand				3.70
	const				3.41

(c) 行列 LI の場合

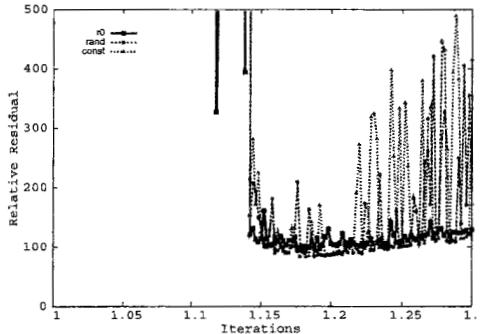


図 1 加速 ILU(0) 分解つき GPBiCG 法の反復回数の加速度係数依存性 (行列 EX19 のとき)

Fig. 1 Variation of iterations of GPBiCG method with accelerated ILU(0) decomposition for matrix EX19.

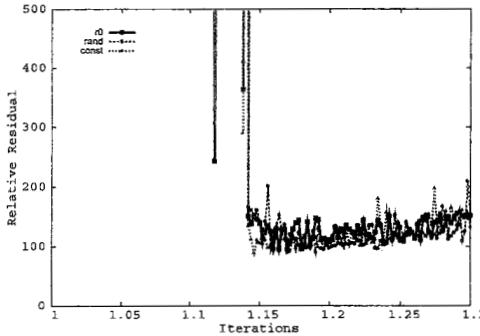


図 2 加速 ILU(0) 分解つき BiCRSafe 法の反復回数の加速度係数依存性 (行列 EX19 のとき)

Fig. 2 Variation of iterations of BiCRSafe method with accelerated ILU(0) decomposition for matrix EX19.

4.3 テスト問題 2

ここでは、テスト問題として、積層鉄心の表面層を考慮した磁気軸受の渦電流解析から生じた行列を取り扱った⁶⁾⁷⁾。一般に、電気電子機器では、渦電流損を低減するため、非常に薄い鋼板を積層した積層鉄心がよく用いられる。そして、高い設計精度を確保するため、その表面を研磨する必要がある。このとき、場合によっては、層間の絶縁が破れ、この部分に渦電流が流れることがある。したがって、発生する渦電流により被る損失を極力抑えることが重要になる。この非常に薄い表面層のある問題を節要素を用いた有限要素法によりモデル化した場合、要素が扁平な形状になる。その結果、係数行列の性質が悪くなり、反復法が収束しない事態が生じる。

本解析は、次式を基礎方程式とする 3 次元非線形渦電流解析である⁶⁾。ここで、 A , ϕ および J_0 は磁気ベクトルポテンシャル、電気スカラーポテンシャル、強制電流密度を各々表す。また、 ν と σ は磁気抵抗率、導電率を各々表す。

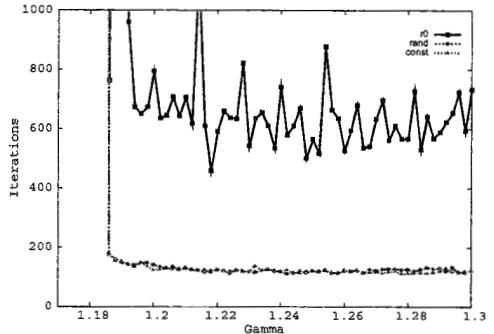


図 3 加速 ILU(0) 分解つき GPBiCG 法の反復回数の加速度係数依存性 (行列 LI のとき)

Fig. 3 Variation of iterations of GPBiCG method with accelerated ILU(0) decomposition for matrix LI.

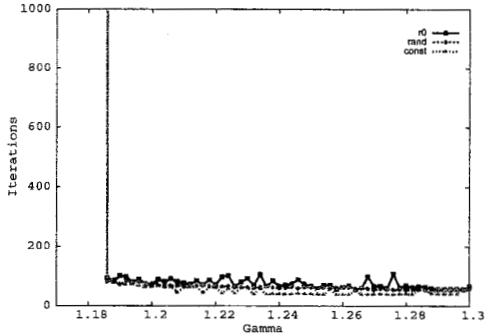


図 4 加速 ILU(0) 分解つき BiCRSafe 法の反復回数の加速度係数依存性 (行列 LI のとき)

Fig. 4 Variation of iterations of BiCRSafe method with accelerated ILU(0) decomposition for matrix LI.

$$w = -\sigma \left(\frac{\partial A}{\partial t} + \text{grad } \phi \right), \quad (51)$$

$$\text{rot} (\nu \text{ rot } A) = J_0 + w, \quad (52)$$

$$\text{div}(w) = 0. \quad (53)$$

表 3 に使用した行列の主な特徴を示す。隙間の大きさを段々狭くする程、行列の性質が悪くなり、反復法は解き難くなる傾向がある。

表 3 テスト問題 2 の行列の主な特徴

行列	次元数	非零要素	平均非零要素	隙間 (mm)
SAGA033	27720	1,665,372	60.08	0.33
SAGA005	27720	1,665,372	60.08	0.05
SAGA001	27720	1,664,326	60.04	0.01

4.4 実験結果

表 4 に、行列 SAGA033 のときの加速 ILU(0) 分解つき反復法の収束性を示す。表中の第 2 棚の「 r_0^* 」は、初期シャドウ残差ベクトル r_0^* について、 $r_0^* = r_0$ の場

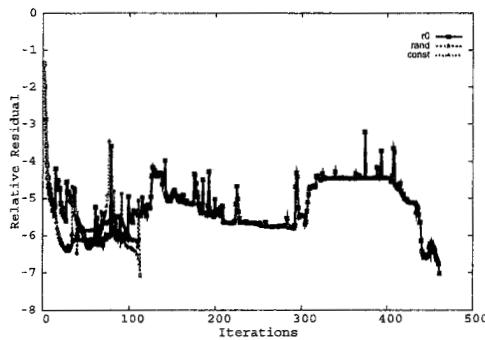


図 5 加速 ILU(0) 分解つき GPBiCG 法の残差履歴 (行列 LI のとき)
Fig. 5 Residual history of GPBiCG method with accelerated ILU(0) decomposition for matrix LI.

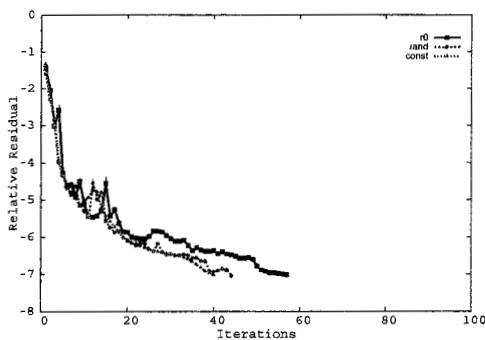


図 6 加速 ILU(0) 分解つき BiCRSafe 法の残差履歴 (行列 LI のとき)
Fig. 6 Residual history of BiCRSafe method with accelerated ILU(0) decomposition for matrix LI.

合、「rand」は r_0^* に一樣乱数を代入した場合、「const」は r_0^* に定数 (= 1.0) を代入した場合、の収束結果を各々表す。「成功度数」とは収束に成功した度数 (= 計算回数)、「成功率 (%)」はその比率を各々表す。また、表中の平均値は、収束に成功した場合の計算時間と反復回数だけを集計した。なお、行列 SAGA005 と行列 SAGA001 の結果については紙面の関係で割愛するが、行列 SAGA033 のときと同様の傾向が観察された。

5. まとめ

提案した BiCR 法の残差をベースにした BiCRSafe 法は、データベースの問題そして実際の解析で現れた行列に対して、従来の反復法よりも優れた収束性を示すことがわかった。

参考文献

- 曾我部知広、杉原正顯、張紹良：共役残差法の非対称行列用への拡張、日本応用数理学会論文誌、

表 4 加速 ILU(0) 分解つき反復法の収束性 (行列 SAGA033 のとき)
Table 4 Convergence of iterative methods with accelerated ILU(0) decomposition for matrix SAGA033.

反復法	r_0^*	成功度数	成功率 (%)	ave. itr.	ave. time
CGS	r_0	0	0	-	-
	rand	0	0	-	-
	const	0	0	-	-
BiCGStab	r_0	44	29	3919	164.64
	rand	26	17	4921	213.23
	const	24	16	4689	185.33
GPBiCG	r_0	108	72	2874	123.68
	rand	108	72	2499	110.24
	const	0	0	-	-
BiCGSafe	r_0	117	77	2364	102.03
	rand	124	82	2188	97.11
	const	123	81	2051	87.17
BiCRSafe	r_0	112	74	2663	113.45
	rand	124	82	2156	95.75
	const	124	82	1920	80.93

- Vol.15, No.3, pp.445-459, 2005.
 2) 阿部邦美、曾我部知広、藤野清次、張紹良：非対称行列用共役残差法に基づく積型反復解法、情報処理学会論文誌 コンピュータシステム、Vol.48 No.SIG 8(ACS18), pp.11-21, 2007.
 3) 藤原牧、吉田正浩、藤野清次：収束の3重の安全鍵を与えるCrout版ILU分解つきBiCGSafe法、情報処理学会論文誌、Vol.47, No.SIG 7, pp.52-60, 2006.
 4) T. Davis: University of Florida Sparse Matrix, <http://www.cise.ufl.edu/research/sparse/matrices/>
 5) Li, N., Saad, Y. and Chow, E.: Crout version of ILU for general sparse matrices, *SIAM J. Sci. Comput.*, **25**(2003), 716-728.
 6) Muramatsu, K., Takahashi, N., et al., Three-dimensional steady-state eddy-current analysis of moving conductor using edge elements and moving-coordinate system, *IEEE Trans. on Magnetics*, **38**(2002), 597-600.
 7) 清水謙規、村松和弘他：箔要素を用いて積層鉄心の表面層を考慮した磁気軸受の渦電流解析、電気学会研究会資料 SA-05-18, (2005), 105-110.
 8) Saad, Y.: Iterative methods for sparse linear systems 2nd edition, SIAM Philadelphia, 2003.
 9) van der Vorst, H.A.: Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.*, **13**(1992), 631-644.
 10) van der Vorst, H.A.: Iterative Krylov preconditionings for large linear systems, Cambridge University Press, Cambridge, 2003.