

反復解法の収束特性と計算効率

張 紹良 藤野 清次

計算流体力学研究所

本研究では、新しく提案された Bi-CGSTAB 法の '生いたち' とアルゴリズムの構成過程を、従来の CG 法、Bi-CG 法、そして CGS 法のアルゴリズムを発展させる形で導く。さらに、典型的なある数値例を通じて、Bi-CGSTAB 法が CGS 法より双直交性の崩れに強いことを実証する。またその収束過程の様子をパラメータ α_k と β_k を追跡することによって明らかにする。

CONVERGENCE BEHAVIOUR AND COMPUTATIONAL EFFICIENCY FOR A CLASS OF ITERATIVE METHODS AS CONJUGATE GRADIENTS TYPE

Shaoliang Zhang Seiji Fujino

Institute of Computational Fluid Dynamics

1-22-3 Haramachi, Meguro-ku, Tokyo 152, Japan

It has been observed that a class of iterative methods as Conjugate Gradients type plays an important role in solving a large system of linear equations. However, due to the rounding error of computer, some iterative methods, e.g., Conjugate Gradients-Squared methods, often show a rather irregular convergence behaviour. Recently, a new iterative method, named Bi-CGSTAB, has been proposed as a fast and smoothly variant of the conjugate gradients type. In this paper, the convergence behaviour of iterative methods involving the Bi-CGSTAB will be illustrated through residual polynomials.

1 まえがき

物理現象を記述する何らかの偏微分方程式を数値的に解いて近似解を求めるとき、これは大型でスパースな連立1次方程式

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

を解くことに帰着される。もとの偏微分方程式が自己随伴のとき、係数行列 A は対称になるが、移流拡散項などがあると、係数行列 A は非対称になる。近年、クリロフ部分空間に帰着される共役勾配法系列の反復解法が盛んに開発され、古典的な反復解法 SOR 法に代わりつつある。その系列の中に、対称用の CG 法、非対称用の Bi-CG 法、そして高収束性をもつ CGS 法などがある。しかし、これらの CG 法系列の解法のほとんどは双直交性 (bi-orthogonalization) に依存した解法である。そのため、丸め誤差の影響で、双直交性が崩れ、理論評価のようにうまく収束しないケースも多い。

本研究では、Van der Vorst[1] によって提案された Bi-CGSTAB 法を解釈し、これを含む CG 法系列の解法の収束特性について、解析的な考察を数値例を通じて行う。

2 共役勾配法系列の反復解法

2.1 共役勾配 (CG: Conjugate Gradient) 法

この章の CG 法、Bi-CG 法、そして CGS 法の導出は、主として文献 [2] に拠っている。

Hestenes と Stiefel によって提案された共役勾配法の原理は、対称正定値行列 (自己随伴) A に関する汎関数

$$f(x) = \frac{1}{2}(x, Ax) - (x, b) \quad (2)$$

を最小にするという大域的な性質に基づく。

いま、対称正定値行列を係数にもつ連立1次方程式 (1) の解を CG 法で求めるとき、そのアルゴリズムは以下ようになる。ただし ε は収束判定値とする。

初期値 x_0 を与える。

$$r_0 = b - Ax_0, \quad p_0 = r_0, \quad k = 0$$

$\|r_k\| \leq \varepsilon \|b\|$ が満たされるまで以下の反復を繰り返す。

$$\alpha_k = (r_k, r_k) / (p_k, Ap_k), \quad x_{k+1} = x_k + \alpha_k p_k, \quad r_{k+1} = r_k - \alpha_k Ap_k,$$

$$\beta_k = (r_{k+1}, r_{k+1}) / (r_k, r_k), \quad p_{k+1} = r_{k+1} + \beta_k p_k,$$

$$k = k + 1$$

アルゴリズム 1 CG 法のアルゴリズム

ここで、残差ベクトル r_k と探索方向ベクトル p_k に関する 2 つの三項漸化式から、残差ベクトル r_k は

$$r_k = R_k(A)r_0 \quad (3)$$

の形で書ける。 $R_k(A)$ は係数行列 A に関する k 次多項式で、残差多項式または r_k の生成多項式と呼ばれる。本報告では、この残差多項式が誤差評価の重要なカギを握っている。CG 法の残差多項式については文献 [3] が詳しいので参照されたい。

2.2 Bi-CG(Bi-Conjugate Gradient) 法

CG 法は、その原理から非対称行列（自己随伴ではない）には適用できない。そこで、式(1)が非対称行列のとき、それを解くための双共役勾配 (Bi-CG) 法とは、元の方程式とそれと双対な方程式 $A^T x^* = b^*$ を組み合わせて計算する方法である。すなわち、

$$\tilde{H} = \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix}, \quad \tilde{A} = \begin{pmatrix} A & 0 \\ 0 & A^T \end{pmatrix}, \quad \tilde{x} = \begin{pmatrix} x \\ x^* \end{pmatrix}, \quad \tilde{b} = \begin{pmatrix} b \\ b^* \end{pmatrix}$$

とすると、連立1次方程式

$$\tilde{A}\tilde{x} = \tilde{b} \quad (4)$$

を考えることに相当する。 \tilde{A} が \tilde{H} 内積 ($\langle \tilde{x}, \tilde{y} \rangle \equiv (\tilde{x}, \tilde{H}\tilde{y})$) に関して自己随伴

$$\langle \tilde{x}, \tilde{A}\tilde{y} \rangle = \langle \tilde{x}, \tilde{A}\tilde{y} \rangle \quad (5)$$

であるので、式(6)の汎関数

$$f(\tilde{x}) = \frac{1}{2} \langle \tilde{x}, \tilde{A}\tilde{x} \rangle - \langle \tilde{x}, \tilde{b} \rangle \quad (6)$$

に CG 法のアルゴリズムを適用すると、Bi-CG 法のアルゴリズムは以下ようになる。ただし ϵ は収束判定値とする。

初期値 x_0 を与える。

$$r_0 = b - Ax_0, \quad r_0^* = r_0^*, \quad p_0 = r_0, \quad p_0^* = r_0^*, \quad k = 0$$

$\|r_k\| \leq \epsilon \|b\|$ が満たされるまで以下の反復を繰り返す。

$$\begin{aligned} \alpha_k &= (r_k^*, r_k) / (Ap_k^*, p_k), & x_{k+1} &= x_k + \alpha_k p_k, & r_{k+1} &= r_k - \alpha_k Ap_k, & r_{k+1}^* &= r_k^* - \alpha_k A^T p_k^*, \\ \beta_k &= (r_{k+1}^*, r_{k+1}) / (r_k^*, r_k), & p_{k+1} &= r_{k+1} + \beta_k p_k, & p_{k+1}^* &= r_{k+1}^* + \beta_k p_k^*, \\ k &= k + 1 \end{aligned}$$

アルゴリズム 2 Bi-CG 法のアルゴリズム

ここで、 r_k 、 p_k 、 r_k^* 、 p_k^* は CG 法と同様に次の式(7)の形で書ける。

$$r_k = R_k(A)r_0, \quad p_k = P_k(A)r_0, \quad r_k^* = R_k(A^T)r_0^*, \quad p_k^* = P_k(A^T)r_0^* \quad (7)$$

2.3 CG Squared 法 (CGS 法)

自乗共役勾配 (CGS) 法は、Bi-CG 法において残差 $r_0^* = r_0$ としたものに対応する。Bi-CG 法のアルゴリズム中の r_k 、 p_k の残差多項式を $R_k(A)$ 、 $P_k(A)$ として、さらに

$$\hat{r}_k = R_k^2(A)r_0, \quad \hat{p}_k = P_k^2(A)r_0 \quad (8)$$

を新たに導入し Bi-CG 法を再構成すると、CGS 法が導き出せる。そのアルゴリズムは以下ようになる。ただし ϵ は収束判定値とする。

初期値 \hat{x}_0 を与える。

$$\hat{r}_0 = b - A\hat{x}_0, \quad \hat{p}_0 = e_0 = \hat{r}_0, \quad k = 0$$

$\|\hat{r}_k\| \leq \epsilon \|\hat{b}\|$ が満たされるまで以下の反復を繰り返す。

$$\begin{aligned} \alpha_k &= (\hat{r}_0, \hat{r}_k) / (\hat{r}_0, A\hat{p}_k), & h_{k+1} &= e_k - \alpha_k A\hat{p}_k, & \hat{r}_{k+1} &= \hat{r}_k - \alpha_k A(e_k + h_{k+1}), & \hat{x}_{k+1} &= \hat{x}_k + \alpha_k (e_k + h_{k+1}), \\ \beta_k &= (\hat{r}_0, \hat{r}_{k+1}) / (\hat{r}_0, \hat{r}_k), & e_{k+1} &= \hat{r}_{k+1} + \beta_k h_{k+1}, & \hat{p}_{k+1} &= e_{k+1} + \beta_k (h_{k+1} + \beta_k \hat{p}_k), \\ k &= k + 1 \end{aligned}$$

アルゴリズム 3 CGS 法のアルゴリズム

CGS 法と Bi-CG 法は本質的には同じものである。相違点は Bi-CG 法ではベクトル列 $\{r_k\}$ を計算するのに対して、CGS 法ではベクトル列 $\{\hat{r}_k\}$ を計算するところにある。そのため、もし、Bi-CG 法の残差多項式が

$$\|r_k\| = \|R_k(A)r_0\| \ll \|b\| \quad (9)$$

のようになったとすると、CGS 法の残差ベクトルの大きさは、

$$\|\hat{r}_k\| = \|R_k^2(A)r_0\| < \|R_k(A)r_0\| \quad (10)$$

のように小さくなると期待できる。このため一般には CGS 法の方が Bi-CG 法より速く収束する。

しかし、逆に収束過程において

$$\|r_k\| = \|R_k(A)r_0\| \gg \|b\| \quad (11)$$

になったとき、CGS 法の残差の大きさは、

$$\|\hat{r}_k\| = \|R_k^2(A)r_0\| > \|R_k(A)r_0\| \quad (12)$$

のように、元の $\|R_k(A)r_0\|$ より大きくなる。これが CGS 法の短所といわれている。

2.4 Bi-CGSTAB 法

2.4.1 Bi-CGSTAB 法の残差多項式

この CGS 法の残差多項式が持つ長所と短所の両方の性質を斟酌して、Van der Vorst は次の残差多項式を考案した [1]。

$$\bar{r}_k = Q_k(A)R_k(A)r_0 \quad (13)$$

つまり、まず CGS 法の残差多項式を $R_k^2(A)r_0 = R_k(A) \times R_k(A)r_0$ のように 2 つの積の形に分解する。後半の項 $R_k(A)r_0$ は、Bi-CG 法の残差多項式をそのまま選び、収束性を Bi-CG 法で裏付ける。その上で、新たな前半の項 $R_k(A)$ は、次の多項式 $Q_k(A)$ で評価する。

$$Q_k(A) = (1 - \omega_k A)(1 - \omega_{k-1} A) \cdots (1 - \omega_1 A) \quad (14)$$

これを漸化式の形で表すと、

$$Q_k(A) = (1 - \omega_k A)Q_{k-1}(A) \quad (15)$$

である。

パラメータ $\omega_1, \omega_2, \dots, \omega_k$ の関数である $\|\bar{r}_k\|$ を、全体的に最小化するのは難しいので、次善の策として第 $k-1$ 番目までの反復過程の $Q_{k-1}(A)$ はそのままにして、第 k ステップにおいて追加された $(1 - \omega_k A)$ に関してだけ $\|\bar{r}_k\|$ を最小にするようパラメータ ω を決める。すなわち、Bi-CGSTAB 法の残差多項式は、

$$\bar{r}_k = (1 - \omega_k A)(1 - \omega_{k-1} A) \cdots (1 - \omega_1 A)R_k(A)r_0 \quad (16)$$

になる。

2.4.2 アルゴリズムの構成方法

Bi-CGSTAB 法のアルゴリズムは、以下のように構成される。すなわち、

- (1) $R_k(A)$ を決めるために、Bi-CG 法での定義に基づいてパラメータ α, β を導く
- (2) パラメータ ω は残差ノルム最小の考え方から求める

という方針に沿っておこなう。

まず、Bi-CG 法において、残差ベクトル r_k と探索方向ベクトル p_k に関する三項漸化式から、

$$P_{k+1}(A)r_0 = (R_{k+1}(A) + \beta_k A P_k(A))r_0 \quad (17)$$

$$R_{k+1}(A)r_0 = (R_k(A) - \alpha_k A P_k(A))r_0 \quad (18)$$

の関係が成り立つ。Bi-CGSTAB 法の残差ベクトル \bar{r}_{k+1} を

$$\bar{r}_{k+1} = Q_{k+1}(A)R_{k+1}(A)r_0 \quad (19)$$

とおくとき、

$$\bar{r}_{k+1} = Q_{k+1}(A)R_{k+1}(A)r_0 = (1 - \omega_{k+1}A)Q_k(A)(R_k(A) - \alpha_k A P_k(A))r_0 \quad (20)$$

になる。ここで補助変数 s を導入して、

$$s = (Q_k(A)R_k(A) - \alpha_k A Q_k(A)P_k(A))r_0 = \bar{r}_k - \alpha A \bar{p}_k \quad (21)$$

と定義すると、(19)式の残差ベクトルは

$$\bar{r}_{k+1} = s - \omega_{k+1} A s \quad (22)$$

のような形で表される。

同様に、方向ベクトル $\bar{p}_{k+1} = Q_{k+1}(A)P_{k+1}(A)r_0$ についても、

$$\bar{p}_{k+1} = Q_{k+1}(A)(R_{k+1}(A) + \beta_k A P_k(A))r_0 \quad (23)$$

$$= (Q_{k+1}(A)R_{k+1}(A) + \beta_k(1 - \omega_{k+1}A)Q_k(A)P_k(A))r_0 \quad (24)$$

$$= \bar{r}_{k+1} + \beta_{k+1}(1 - \omega_{k+1}A)\bar{p}_k \quad (25)$$

のように、漸化式の形で書き表わされる。一方パラメータ β_k は、次のように導出できる。

双直交性によって、

$$(r_k^*, r_k) = (P_k(A^T)r_0^*, P_k(A)r_0) = (-1)^k \alpha_1 \alpha_2 \dots \alpha_k ((A^T)^k r_0^*, P_k(A)r_0) \quad (26)$$

$$\rho_k = (r_0^*, \bar{r}_k) = (Q_k(A^T)r_0^*, P_k(A)r_0) = (-1)^k \omega_1 \omega_2 \dots \omega_k ((A^T)^k r_0^*, P_k(A)r_0) \quad (27)$$

がわかる。また Bi-CG 法の β_k の定義から、

$$\beta_k = (r_{k+1}^*, r_{k+1}) / (r_k^*, r_k) \quad (28)$$

$$= (-1)^{\alpha_{k+1}} ((A^T)^{k+1} r_0^*, P_{k+1}(A)r_0) / ((A^T)^k r_0^*, P_k(A)r_0) \quad (29)$$

$$= (\rho_{k+1} / \rho_k) (\alpha_{k+1} / \omega_{k+1}) \quad (30)$$

である。パラメータ α についても同様に導出できる。

一方、パラメータ ω は、次のようにして決められる。いま、 $t = As$ と定義すると、

$$\|\bar{r}_{k+1}\| = \|s - \omega t\| = (s, s) - 2\omega(s, t) + \omega^2(t, t) \quad (31)$$

である。この式は ω の2次方程式とみなせるので、その最小値は

$$\omega = (s, t)/(t, t) \quad (32)$$

のとき得られる。Bi-CGSTAB 法はアルゴリズム 4 のようにまとめられる。ただし ε は収束判定値とする。

初期値 x_0 を与える。

$$r_0 = r_0^* = p_0 = b - Ax_0, \quad \rho_0 = (r_0^*, r_0), k = 0$$

$\|r_k\| \leq \varepsilon \|b\|$ が満たされるまで以下の反復を繰り返す。

$$v_k = Ap_k, \quad \alpha_k = \rho_k / (r_0^*, v_k), \quad s = r_k - \alpha_k v_k, \quad t = As,$$

$$\omega_k = (s, t)/(t, t), \quad x_{k+1} = x_k + \alpha_k p_k + \omega_k s, \quad r_{k+1} = s - \omega_k t,$$

$$\rho_{k+1} = (r_0^*, r_{k+1}), \quad \beta_k = (\rho_{k+1}/\rho_k)(\alpha_k/\omega_k), \quad p_{k+1} = r_{k+1} + \beta_k(p_k - \omega_k v_k),$$

$$k = k + 1$$

アルゴリズム 4 Bi-CGSTAB 法のアルゴリズム

3 残差多項式から見た CG 法系列の反復解法の収束特性

三つの解法、Bi-CG 法、CGS 法、そして Bi-CGSTAB 法のアルゴリズムから、第 k 番目の反復ステップにおけるパラメータ α_k と β_k は、同一のものであることがわかる。しかし、有限桁で演算を行うコンピュータの丸め誤差の影響で、この三つの解法は三者三様の収束過程をたどる。双直交性が崩れやすいものもあるし、なかなか崩れにくいタイプもある。そこで、ある対称行列の場合を例に取り上げ、CGS 法と Bi-CGSTAB 法の双直交性の崩れによる α_k 、 β_k の値を倍精度と 4 倍精度について調べる。

ここで、解析領域 $[0, 1] \times [0, 1]$ において、ポアソン方程式の境界値問題

$$\begin{aligned} U_{xx} + U_{yy} &= -2\pi^2 \sin(\pi(x+y)) & (x, y) \in [0, 1] \times [0, 1] \\ U &= g(x, y) & (x, y) \in \partial([0, 1] \times [0, 1]) \end{aligned} \quad (33)$$

を考える。ただし、解析解を $U = \sin(\pi(x+y))$ とする。

解析領域をそれぞれの方向ともに 26 等分割して、内部の点 $n=625$ 個の未知数をもつ連立 1 次方程式 $Ax = b$ について考える。

このとき、初期残差ベクトル r_0 を行列 A の固有ベクトル $\{u_i\}$ で次のように展開する。

$$r_0 = \sum_{i=1}^n c_i u_i \quad (34)$$

反復の第 k ステップにおいて、GCS 法と Bi-CGSTAB 法の残差ベクトルは

$$\begin{aligned} \|\hat{r}_k\|^2 &= \sum_{i=1}^n c_i^2 \{R_k(\lambda_i)\}^4 \\ \|\bar{r}_k\|^2 &= \sum_{i=1}^n c_i^2 \{\omega_k(\lambda_i)\}^2 \{R_k(\lambda_i)\}^2 \end{aligned} \quad (35)$$

になる。

一方、残差多項式 $R_k(\lambda)$ は Bi-CG 法の性質 $(r_i, r_j) = 0$ から、選点直交多項式系であることがわかり、最終的に

$$R_n(\lambda_i) = 0, \quad 1 \leq i \leq n \quad (36)$$

が成り立つ [3]。すなわち、

$$\sum_{i=1}^n c_i^2 (R_n(\lambda_i))^2 = 0 \quad (37)$$

である。したがって、Hölder不等式によって、

$$\|\bar{r}_n\|^2 \leq \sqrt{\sum_{i=1}^n c_i^2 (R_n(\lambda_i))^2} \sqrt{\sum_{i=1}^n c_i^2 (R_n(\lambda_i))^6} = 0 \quad (38)$$

$$\|\bar{r}_n\|^2 \leq \sqrt{\sum_{i=1}^n c_i^2 (R_n(\lambda_i))^2} \sqrt{\sum_{i=1}^n c_i^2 (\omega_n(\lambda_i))^4 (R_n(\lambda_i))^6} = 0 \quad (39)$$

が成り立つ。これらの式は、対称行列に対して、CGS法とBi-CGSTAB法が理論上を高々n回の反復で収束することを示している。

しかし、 $R_k(\lambda)$ は、

$$R_k(\lambda) = (-\alpha_{k-1}\lambda + 1 + \frac{\alpha_{k-1}\beta_{k-1}}{\alpha_{k-2}})R_{k-1}(\lambda) - \frac{\alpha_{k-1}\beta_{k-2}}{\alpha_{k-2}}R_{k-2}(\lambda) \quad (40)$$

という漸化式の形で表され、その正確度は二つのパラメータ α_k と β_k の正確度に依存している。したがって、パラメータ α_k と β_k の値のよしあしは、双直交性が成り立っているかどうか大きく影響する。

数値実験は、VP-2600を用いて倍精度で行ったポアソン方程式(33)を解いたときのCGS法とBi-CGSTAB法の収束特性を図1に示す。ただし、初期値 $x_0 = 0$ である。図1において水平方向は反復回数を表わし、垂直方向は相対残差の対数 $\log_{10}(\|r_k\| / \|b\|)$ を表わす。

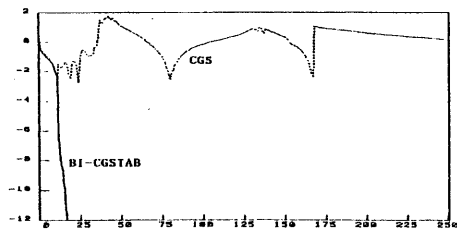


図1 CGS法とBi-CGSTAB法の収束特性

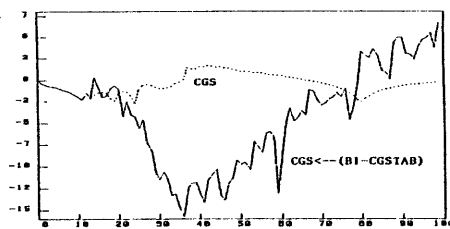


図3 変化したCGS法の収束特性

Bi-CGSTAB法は、17回の反復計算で相対誤差が 10^{-12} まで減少した。一方、CGS法は250回まで反復計算を行ったが、収束しなかった。

倍精度と4倍精度の演算を行ったときの α_k 、 β_k の値を図2に示す。それぞれの精度における α_k 、 β_k を観察すると、11回の反復までは、両解法の α_k 、 β_k はまったく同じである。しかし、第12回目の反復における β_{12} が極端に小さな値になった後、両解法の α_k 、 β_k は全然違った値になり、収束性も非常に異なる。すなわち、両者の相対残差は12回まで同じオーダーであるが、その後CGS法は振動したり、停滞するのに対して、Bi-CGSTAB法はきれいに単調減少する。

Bi-CGSTAB法の計算で使われた α_k 、 β_k の値をもっと詳細に検討するため、以下の実験を行った。

まず、Bi-CGSTAB法で反復計算を行い、全ての反復ステップにおける α_k 、 β_k を記憶させる。次に、CGS法の反復計算を始める。ただしCGS法本来の α_k 、 β_k を使わず、最初に覚えておいたBi-CGSTAB法の α_k 、 β_k を使って反復計算を繰り返した。倍精度の結果を図3に示す。

CGS法本来の α_k 、 β_k を使ったときの結果を点線で表示し、Bi-CGSTAB法の α_k 、 β_k を'借りた'CGS法の結果を実線で表示する。もとのCGS法はほとんど停滞したまま収束しなかったのに対して、後者は、35回の反復で相対残差が 10^{-15} まで減少したことがわかる。

反復 回数	倍精度						4倍精度					
	CGS法			Bi-CGSTAB法			CGS法			Bi-CGSTAB法		
	α_k	β_k	$\log_{10}(\frac{\ r_k\ }{\ r_0\ })$	α_k	β_k	$\log_{10}(\frac{\ r_k\ }{\ r_0\ })$	α_k	β_k	$\log_{10}(\frac{\ r_k\ }{\ r_0\ })$	α_k	β_k	$\log_{10}(\frac{\ r_k\ }{\ r_0\ })$
1	.512168	.277872	0.00	.512168	.277872	0.00	.512168	.277872	0.00	.512168	.277872	0.00
2	.704144	.517553	-0.41	.704144	.517553	-0.50	.704144	.517553	-0.41	.704144	.517553	-0.50
3	.802852	.660594	-0.61	.802852	.660594	-0.73	.802852	.660594	-0.61	.802852	.660594	-0.73
4	.850576	.729661	-0.75	.850576	.729661	-0.88	.850576	.729661	-0.75	.850576	.729661	-0.86
5	.866079	.747695	-0.85	.866079	.747695	-0.99	.866079	.747695	-0.85	.866079	.747695	-0.99
6	.862954	.736461	-0.96	.862954	.736461	-1.10	.862954	.736461	-0.96	.862954	.736461	-1.10
7	.850941	.712209	-1.11	.850941	.712209	-1.21	.850941	.712209	-1.11	.850941	.712209	-1.21
8	.835908	.684325	-1.28	.835908	.684325	-1.33	.835908	.684325	-1.28	.835908	.684325	-1.33
9	.820550	.656632	-1.48	.820550	.656632	-1.48	.820550	.656632	-1.48	.820550	.656632	-1.48
10	.805391	.629320	-1.70	.805391	.629320	-1.68	.805391	.629320	-1.70	.805391	.629320	-1.68
11	.789562	.600309	-1.98	.789562	.600309	-1.96	.789562	.600309	-1.98	.789562	.600309	-1.96
12	.771098	.714603E-15	-2.30	.771098	-.223110E-13	-2.39	.771098	-.777612E-32	-2.33	.771098	-.631371E-29	-2.39
13	.344570	.880167	-1.52	.232898	.135556E+01	-6.49	-.718970	-.538045E+01	-18.63	.260189	-.264727	-23.48
14	.790598	.343243E+01	-1.74	-.151466	-.860118	-8.34	-.111432E+01	-.141189E+01	-16.99	.712976	.184667E+01	-24.90
15	.116782E+01	.140472E+01	-1.69	-.118702E+01	.162713E+01	-8.37	.957566	-.105054	-16.14	.172837	-.197377	-25.36
16	.679878	-.448430	-1.52	-0.110366	-.534097E-02	-9.98	.265273	-.628995	-14.62	.403423	-.110006	-27.13
17	-.710227E-01	-.176000E+01	-1.40	-.101553	-.129298E+01	-11.19	.138478E+01	.325009E+01	-14.52	.262212	-.940203E-01	-28.28
18	-.733333E+01	.448000E+03	-1.49	-.476638E+01	.596301E+01	-12.20	.904695	.661469E-01	-12.11	.665205	-.315176	-29.28
19	.375977E-02	.480519	-2.17	.126919	.102593	-12.76	.743261	.189606E+01	-10.93	.543790	.453726	-29.60
20	.628397E-02	.918919	-2.43	.431978	.693550	-13.44	-.286546E+01	-.105163E+02	-9.57	.714955	.245609E-01	-30.01

図2 各反復ステップの α_k と β_k

4 まとめ

本研究では、新しく提案された Bi-CGSTAB 法の '生いたち' とアルゴリズムの構成過程を、従来の CG 法、Bi-CG 法、そして CGS 法のアルゴリズムを発展させる形で導いた [1]。さらに、典型的なある数値例を通じて、Bi-CGSTAB 法が CGS 法より双直交性の崩れに強いことを、実証した。またその収束過程の様子をパラメータ α_k と β_k を追跡することによって明らかにした。

5 参考文献

1. H.A. Van der Vorst, Bi-CG STAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems, Preprint Nr. 633, University Utrecht, Dept. Math., 1990, SIAM J. Sci. Stat. Compt. (to appear)
2. 村田健郎、名取亮、唐木幸比古著、「大型数値シミュレーション」、岩波書店、1990、pp.95-137
3. 森 正武、ICCG 法と残差多項式、京都大学数理解析研究所 講究録 No.585, 1986, pp.113-129