数理解析研究所講究録
第 880 巻 1994 年 44-52

非対称行列の積型反復解法をめぐって

張　紹良[†]　　　　藤野　清次[††]

[†]名古屋大学工学部　　[††]計算流体力学研究所

Generalizations of Product-type Methods Based on Lanczos Process

SHAO-LIANG ZHANG[†] AND SEIJI FUJINO[††]

[†]FACULTY OF ENGINEERING, NAGOYA UNIVERSITY

[††]INSTITUTE OF COMPUTATIONAL FLUID DYNAMICS

Abstract. Recently Bi-CGSTAB as a variant of Bi-CG has been proposed for solving nonsymmetric linear systems, and its attractive convergence behaviour has been confirmed in many numerical experiments. Bi-CGSTAB can be characterized by its residual polynomial which consists of the product of the Lanczos polynomial form from Bi-CG with other polynomial generated from two-term recurrence relations. In this paper, we propose a unification and generalization of results involing product-type methods for the iterative solution of nonsymmetric linear systems. A characteristic of this class of methods (that includes CGS, Bi-CGSTAB, and Bi-CGSTAB2) is the relationship

$$r_n = H_n(A)R_n(A)r_0$$

where r_n is the residual vector corresponding to the n-th iterate x_n, and R_n is the Lanczos polynomial form. The polynomial H_n in the product $H_n(A)R_n(A)$ is chosen to speed up and/or stabilize convergence, while satisfying a standard three-term recurrence relations. Such product-type methods can be regarded as generalizations of Bi-CGSTAB. From the unification and generalization, we can see how CGS, Bi-CGSTAB, and Bi-CGSTAB2 fit into a more general framework.

Key words. Bi-CG, Bi-CGSTAB, CGS, nonsymmetric linear systems, product-type methods, the Lanczos polynomial form, three-term recurrence relations.

AMS(MOS) subject classification. 65F10

1 まえがき

科学技術計算の分野においては，我々は自然現象を記述する偏微分方程式の解を数値的に求めることにしばしば遭遇する．それは，最終的に大規模でスパースな連立 1 次方程式

(1-1) $$Ax = b$$

を解くことに帰着されることが多い．式 (1-1) の解を精度良く求めることは応用分野で大変重要な位置を占めている．

対象とする問題が大規模になるにつれて，記憶容量の増大，計算時間の長大化などの問題が起き，ガウス消去法をはじめとする直接解法では対応しきれなくなっている．そのため，計算効率と精度のよい反復解法の必要性が唱えられ，研究も活発に行われている．共役勾配系統の反復解法は，スパースな問題を解くときの収束性が優れている [16]. 特に急速に発展を遂げている前処理技術と併用することにより収束性は驚くほど向上する [10, 11, 25]. 近年，非対称問題用のランチョス・プロセス (Lanczos process)[9] に基づく反復解法の開発が盛んに行われている．

本研究では，まず数値計算法で広く利用されるランチョ

ス・プロセスを紹介する．そのランチョス・プロセスに基づいて，対称行列用の共役勾配法 [8] を，それから非対称行列用の双共役勾配法 [3] を導出することについて述べる．そこで，我々はランチョス・プロセスに基づく積型反復解法を一般化する手法を提案する．即ち，ランチョス・プロセスに基づく積型反復解法の定義を与え，積型反復解法の設計方針を定め，CGS 法，Bi-CGSTAB 法を含む積型反復解法のいくつかの実用的なアルゴリズムを導き出す [30]．

2　ランチョス・プロセスと共役勾配法

共役勾配法 (Conjugate Gradient method, CG 法) の導出と言えば，M. R. Hestenes と E. Stiefel によって提案された正値対称行列 A の汎関数

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

の最小点を逐次最小化法で求めるものと連想しがちであるが [8, 12, 15]，ここでは，ランチョス・プロセスを利用して適当な条件のもとで CG 法を導き出すことにする [13, 23, 24]．この導出方法は，CG 法を理解し，それを拡張する点において汎関数からの導出では得られない利点を持っている．

2.1　ランチョス・プロセス

正則な行列 A と非零ベクトル u_0 によって作られたベクトル列 $\{u_0, Au_0, \cdots, A^{n+1}u_0\}$ は一次独立であると仮定する．そのとき，ベクトル列 $\{u_0, Au_0, \cdots, A^{n+1}u_0\}$ にグラム・シュミットの直交化法 (Gram-Schmidt orthogonalization) を施すと，その列で張られたクリロフ部分空間 (Krylov subspace) $\mathrm{Span}\{u_0, Au_0, \cdots, A^{n+1}u_0\}$ の直交系 $v_0, v_1, \cdots, v_{n+1}$ を生成することができる．v_{n+1} は次のように $n+1$ 次多項式 $P_{n+1}(\lambda)$ によって表せるので，

$$v_{n+1} = P_{n+1}(A)u_0, \tag{2-2}$$

容易に確かめられるように，v_{n+1} は次の漸化式によって計算される．

$$v_{n+1} = \alpha_n \left(Av_n - \sum_{k=0}^{n} \frac{(Av_n, v_k)}{(v_k, v_k)} v_k \right). \tag{2-3}$$

ただし，パラメータ α_n は v_{n+1} のスケーリング (Scaling) 係数である．式 (2-3) のように正則な行列 A と非零ベクトル u_0 から直交系を作ることは，アルノルディ・プロセス (Arnoldi process) と呼ばれる [1, 17]．

次に，アルノルディ・プロセスを正値対称行列 A に適用すると，漸化式 (2-3) は次のようになる．

$$v_{n+1} = \alpha_n \left(Av_n - \sum_{k=0}^{n} \frac{(v_n, Av_k)}{(v_k, v_k)} v_k \right). \tag{2-4}$$

式 $Av_k = AP_k(A)u_0$ から Av_k は $v_0, v_1, \cdots, v_{k+1}$ の線形結合で表されることがわかる．$v_0, v_1, \cdots, v_k$ は直交系であるため，

$$(v_n, Av_k) = 0, \quad (k \le n-2). \tag{2-5}$$

が成り立つ．したがって，漸化式 (2-4) の右辺は，結局次の 3 項漸化式になる．

$$\begin{aligned} v_{n+1} &= \alpha_n (Av_n - \frac{(v_n, Av_n)}{(v_n, v_n)} v_n \\ &- \frac{(v_n, Av_{n-1})}{(v_{n-1}, v_{n-1})} v_{n-1}). \end{aligned} \tag{2-6}$$

すなわち，v_{n+1} はそれまでのすべての v_k でなく直前の二項だけで求められる．これはアルノルディ・プロセスの特殊なケースであり，ランチョス・プロセス (Lanczos process) と呼ばれる [9]．ランチョス・プロセスは，数値解析の多くの分野において有用であり，固有値解析 (ランチョス・プロセスが発見されたきっかけ)，直交多項式，Stieltjes 連分数などへの応用は，その典型例である [24]．

2.2　ランチョス・プロセスに基づく共役勾配法の導出

この節で，我々はランチョス・プロセスからどのような過程を辿って正値対称問題のための CG 法を導出するかについて述べる．ただし，この節を通じて式 (1-1) の係数行列 A は正値対称とする．

初期近似解を x_0 とし，初期残差 $b - Ax_0$ を r_0 とする．そのとき，ベクトル列 $\{r_0, Ar_0, \cdots, A^{n-1}r_0\}$ を $V_n(A; r_0)$ で表し，初期残差 r_0 から形成されたクリロフ部分空間 $\mathrm{Span}\{r_0, Ar_0, \cdots, A^{n-1}r_0\}$ を $K_n(A; r_0)$ で表す．

ここで次に述べる二つの条件の下で共役勾配法の導出を考える．

条件：クリロフ部分空間 $K_{n+1}(A; r_0)$ から次のように近似解 x_{n+1} を作り出す．

$$x_{n+1} = x_0 + z_{n+1}, \quad z_{n+1} \in K_{n+1}(A; r_0). \tag{2-7}$$

その近似解x_{n+1}に対応する残差r_{n+1}は次のようになる．

$$\text{(2-8)}\qquad r_{n+1} := b - Ax_{n+1} = r_0 - Az_{n+1}.$$

そこで，$r_{n+1} \in K_{n+2}(A;r_0)$であることがわかる．

条件：近似解x_{n+1}に対応する残差r_{n+1}は次のようにガレルキン条件(Galerkin condition)を満たす[17].

$$\text{(2-9)}\qquad r_{n+1} \perp K_{n+1}(A;r_0).$$

式(2-9)は残差列$\{r_0, r_1, \cdots, r_{n+1}\}$がクリロフ部分空間$K_{n+2}(A;r_0)$の直交系であることを意味する．したがって，残差列$\{r_0, r_1, \cdots, r_{n+1}\}$がグラム・シュミットの直交化法によって作られることがわかる．そのとき，$V_{n+2}(A;r_0)$にランチョス・プロセスを施すと，残差r_{n+1}を次の3項漸化式で形成することができる．

$$\begin{aligned}
\text{(2-10)}\qquad r_1 &= \alpha_0\left(Ar_0 - \frac{(r_0, Ar_0)}{(r_0, r_0)} r_0\right),\\
\text{(2-11)}\qquad r_{n+1} &= \alpha_n Ar_n - \alpha_n \frac{(r_n, Ar_n)}{(r_n, r_n)} r_n\\
&- \frac{\alpha_n}{\alpha_{n-1}} \frac{(r_n, r_n)}{(r_{n-1}, r_{n-1})} r_{n-1}.
\end{aligned}$$

ただし，式(2-6)のスケーリング係数α_nは$r_{n+1} = r_0 - Az_{n+1}(z_{n+1} \in K_{n+1}(A;r_0))$の形を満たすように決められ，次の漸化式を満たす．

$$\begin{aligned}
\text{(2-12)}\qquad \alpha_0 &= -\frac{(r_0, r_0)}{(r_0, Ar_0)},\\
\text{(2-13)}\qquad \alpha_n &= -\frac{1}{\frac{(r_n, Ar_n)}{(r_n, r_n)} + \frac{1}{\alpha_{n-1}} \frac{(r_n, r_n)}{(r_{n-1}, r_{n-1})}}.
\end{aligned}$$

そこで，補助ベクトルp_nを次のように導入する．

$$\text{(2-14)}\qquad p_n = (z_{n+1} - z_n)/\alpha_n \in K_{n+1}(A;r_0).$$

そのとき，式(2-8)から隣合う二つの残差の差$r_{n+1} - r_n$は，補助ベクトルp_nで表せることがわかる．

$$\text{(2-15)}\qquad r_{n+1} - r_n = -\alpha_n Ap_n.$$

これを式(2-11)に代入し，式(2-13)を用いて整理すると，ベクトルp_nに関する漸化式

$$\text{(2-16)}\qquad p_{n+1} = r_{n+1} + \beta_n p_n.$$

が得られる．ただし，$\beta_n = \frac{(r_{n+1}, r_{n+1})}{(r_n, r_n)}$とする．

一方，パラメータα_nは補助ベクトルp_nを用いて次のように求めることができる．式(2-9)と式(2-14)により，残差r_{n+1}はp_nと直交することがわかる．したがって，式(2-15)により，α_nの計算式は次のようになる．

$$\text{(2-17)}\qquad \alpha_n = \frac{(p_n, r_n)}{(p_n, Ap_n)}.$$

また，式(2-16)から，$(p_{n+1}, r_{n+1}) = (r_{n+1}, r_{n+1})$が成り立つことがわかり，$\alpha_n$の計算式は次のように書ける．

$$\text{(2-18)}\qquad \alpha_n = \frac{(r_n, r_n)}{(p_n, Ap_n)}$$

α_nとβ_nの計算式を使うと，r_{n+1}に関する3項漸化式は結局，次のようになる：

$$\begin{aligned}
\text{(2-19)}\qquad r_1 &= r_0 - \alpha_0 Ar_0,\\
\text{(2-20)}\qquad r_{n+1} &= \left(1 + \frac{\beta_{n-1}}{\alpha_{n-1}}\alpha_n\right) r_n - \alpha_n Ar_n\\
&- \frac{\beta_{n-1}}{\alpha_{n-1}} \alpha_n r_{n-1}, \quad n = 1, 2, \cdots.
\end{aligned}$$

また，式(2-14)により，近似解x_{n+1}は次の漸化式で計算できる．

$$\text{(2-21)}\qquad x_{n+1} = x_0 + z_{n+1} = x_n + \alpha_n p_n.$$

以上のことをまとめると，正値対称行列Aを係数にもつ連立1次方程式(1-1)の解を求めるCG法のアルゴリズムが得られる．ただしεは収束判定定数とする（以下同じ）．

ALGORITHM 1　CG法

Choose an initial guess x_0,
and set $p_0 = r_0 = b - Ax_0$.
For $n = 0, 1, \cdots$ until $\| r_n \| \le \varepsilon \| b \|$ do :
$\alpha_n = \frac{(r_n, r_n)}{(p_n, Ap_n)}$,
$x_{n+1} = x_n + \alpha_n p_n$,
$r_{n+1} = r_n - \alpha_n Ap_n$,
$\beta_n = \frac{(r_{n+1}, r_{n+1})}{(r_n, r_n)}$,
$p_{n+1} = r_{n+1} + \beta_n p_n$;

CG法のアルゴリズムに現れるベクトル列r_n, p_nに対して，次の重要な性質が成り立つ．

- $(r_i, r_j) = 0,\ i \neq j;$　(orthogonality property)
- $(p_i, Ap_j) = 0,\ i \neq j.$　(conjugacy property)

残差列 $\{r_0, r_1, \cdots,\}$ が直交列であるので，残差 r_N は高々 N 回の反復でゼロになる．これは CG 法が直接解法として収束することを理論的に裏付ける．しかし，実際の応用問題を解くときに，CG 法は前処理技術との併用で反復解法として利用されることが多い．反復解法としての収束性もすでに理論的に解決された [20, 13].

共役勾配法の導出において，残差 r_n と補助ベクトル p_n が多項式 $R_n(\lambda)$ と $P_n(\lambda)$ によって表せることがわかる．

(2-22) $$r_n = R_n(A)r_0, \quad p_n = P_n(A)r_0.$$

特に $R_n(\lambda)$ はランチョス多項式と呼ばれ，次の 3 項漸化式を満たす．[22]

$$\begin{aligned}
&\text{(2-23)} & R_0(\lambda) &= 1, \\
&\text{(2-24)} & R_1(\lambda) &= (1-\alpha_0\lambda)R_0(\lambda), \\
&\text{(2-25)} & R_{n+1}(\lambda) &= (1+\frac{\beta_{n-1}}{\alpha_{n-1}}\alpha_n - \alpha_n\lambda)R_n(\lambda) \\
& & &\quad - \frac{\beta_{n-1}}{\alpha_{n-1}}\alpha_n R_{n-1}(\lambda), \quad n = 1, 2, \cdots.
\end{aligned}$$

ただし，$R_n(0) = 1$．この 3 項漸化式がランチョス・プロセスに基づく積型反復解法の導出において重要な公式であることを注意しておく．また，多項式 $R_n(\lambda)$ と $P_n(\lambda)$ は次の交替漸化式を満たす．

(2-26) $$R_0(\lambda) = 1,\ P_0(\lambda) = 1;$$

(2-27) $$R_{n+1}(\lambda) = R_n(\lambda) - \alpha_n\lambda P_n(\lambda);$$

(2-28) $$P_{n+1}(\lambda) = R_{n+1}(\lambda) + \beta_n P_n(\lambda).$$

3　双共役勾配法

この節では，非対称行列をもつ式 (1-1) の解法の一つ，双共役勾配法 (Bi-Conjugate Gradient method, Bi-CG 法) の導出について述べる．

正値対称問題の CG 法の導出と同様に，まず次の条件を考える．

条件：近似解 x_{n+1} をクリロフ部分空間 $K_{n+1}(A; r_0)$ から作り出す．

(3-1) $$x_{n+1} = x_0 + z_{n+1},\ z_{n+1} \in K_{n+1}(A; r_0).$$

その近似解 x_{n+1} に対応する残差 r_{n+1} は次のようになる．

(3-2) $$r_{n+1} := b - Ax_{n+1} = r_0 - Az_{n+1}.$$

そこで，$r_{n+1} \in K_{n+2}(A; r_0)$ であることがわかる．

クリロフ部分空間 $K_{n+1}(A; r_0)$ から前節のガレルキン条件 (2-8) を満たすような残差列 r_n をアルノルディ・プロセスを用いて作り出すことができる．アルノルディ・プロセスによって作られた反復解法は一般に GMRES 法と呼ばれる．GMRES 法は，CG 法のように簡単かつ明瞭な 3 項漸化式を満たさないため，計算量，記憶容量とも多くなる．しかし，記憶容量を減らすよう考慮したリスタート版 GMRES(k) 法は非対称問題に有効である [18].

一方，ランチョス・プロセスは正値対称行列にしか適用できないため，前章のガレルキン条件 (2-9) を満たすような残差の列 $\{r_n\}$ は，3 項漸化式 (2-20) から生成できない．そこで，新しいクリロフ部分空間 $K_{n+1}(A^T, r_0^*)$ を利用して，次の条件を考える．

条件：近似解 x_{n+1} に対応する残差 r_{n+1} は次のようなガレルキン条件 (2-9) を満たす [9].

(3-3) $$r_{n+1} \perp K_{n+1}(A^T; r_0^*).$$

そうすると，条件 (3-1) と条件 (3-3) を満たす残差の列 r_0, $r_1, \cdots, r_{n+1} \in K_{n+2}(A; r_0)$ は 3 項漸化式 (2-20) によって生成することができる．

そのとき，式 (2-14) と同じように $p_n (\in K_{n+1}(A; r_0))$ を定義すると，3 項漸化式 (2-19)〜(2-20) より，次の漸化式が得られる．

(3-4) $$r_{n+1} = r_n - \alpha_n Ap_n,$$

(3-5) $$p_{n+1} = r_{n+1} + \beta_n p_n.$$

条件 (3-3) より，$((A^T)^n r_0^*, r_{n+1}) = 0$ がわかり，したがって，α_n は次のように表せる．

(3-6) $$\alpha_n = \frac{((A^T)^n r_0^*,\ r_n)}{((A^T)^n r_0^*,\ Ap_n)}$$

また，条件 (3-3) と式 (3-4) より，次の式が成り立つ．

(3-7) $$Ap_n \perp K_n(A^T; r_0^*).$$

したがって，式 (3-5) により，

(3-8) $$\begin{aligned}
0 = ((A^T)^n r_0^*, Ap_{n+1}) &= ((A^T)^{n+1} r_0^*, p_{n+1}) = \\
((A^T)^{n+1} r_0^*, r_{n+1}) &+ \beta_n((A^T)^{n+1} r_0^*, p_n) = \\
((A^T)^{n+1} r_0^*, r_{n+1}) &+ \beta_n((A^T)^n r_0^*, Ap_n).
\end{aligned}$$

が成り立つ．β_nの計算は

$$\beta_n = -\frac{((A^T)^{n+1}r_0^*,\ r_{n+1})}{((A^T)^n r_0^*,\ Ap_n)} \tag{3-9}$$

となり，α_nの計算式 (3-6) を利用すると，

$$\beta_n = -\alpha_n \frac{((A^T)^{n+1}r_0^*,\ r_{n+1})}{((A^T)^n r_0^*,\ r_n)} \tag{3-10}$$

となる．

そこで，クリロフ部分空間$K_{n+2}(A^T;r_0^*)$において，次のように補助ベクトル列$r_0^*, r_1^*, \cdots, r_{n+1}^*$と$p_0^*, p_1^*, \cdots, p_{n+1}^*$を導入する．

$$r_{n+1}^* = r_n^* - \alpha_n A^T p_n^*, \tag{3-11}$$

$$p_{n+1}^* = r_{n+1}^* + \beta_n p_n^*, \tag{3-12}$$

ただし，$p_0^* = r_0^* = r_0$．そのとき，補助ベクトルr_n^*，p_n^*を次のように展開することができる．

$$r_n^* = R_n(A^T)r_0^* = \left((-1)^n \prod_{i=0}^{n-1} \alpha_i\right)(A^T)^n r_0^* + z_1; \tag{3-13}$$

$$p_n^* = P_n(A^T)r_0^* = \left((-1)^n \prod_{i=0}^{n-1} \alpha_i\right)(A^T)^n r_0^* + z_2, \tag{3-14}$$

ただし，z_1，$z_2 \in K_n(A^T, r_0^*)$．したがって，条件 (3-3) と条件 (3-7) より，α_nの計算式 (3-6) とβ_nの計算式 (3-10) は補助ベクトルr_n^*，p_n^*で表現できる．

$$\alpha_n = \frac{(r_n^*,\ r_n)}{(p_n^*,\ Ap_n)}, \quad \beta_n = \frac{(r_{n+1}^*,\ r_{n+1})}{(r_n^*,\ r_n)}. \tag{3-15}$$

以上のことをまとめると，非対称行列のための Bi-CG 法のアルゴリズムが得られる [3].

ALGORITHM 2　Bi-CG 法

Choose an initial guess x_0,
and set $p_0^* = r_0^* = p_0 = r_0 = b - Ax_0$.
For $n = 0, 1, \cdots$ until $\| r_n \| \le \varepsilon \| b \|$ do :
$\alpha_n = \dfrac{(r_n^*, r_n)}{(p_n^*, Ap_n)}$,
$x_{n+1} = x_n + \alpha_n p_n$,
$r_{n+1} = r_n - \alpha_n Ap_n, \quad r_{n+1}^* = r_n^* - \alpha_n A^T p_n^*$,
$\beta_n = \dfrac{(r_{n+1}^*, r_{n+1})}{(r_n^*, r_n)}$,
$p_{n+1} = r_{n+1} + \beta_n p_n, \quad p_{n+1}^* = r_{n+1}^* + \beta_n p_n^*$;

ここで，3 項漸化式 (2-23)~(2-25) と交替漸化式 (2-26)~(2-28) を満す多項式R_nとP_nを使うと，r_n，p_n，r_n^*，p_n^*は CG 法と同様に下記の形で書ける．

$$r_n = R_n(A)r_0, \quad p_n = P_n(A)r_0, \tag{3-16}$$

$$r_n^* = R_n(A^T)r_0^*, \quad p_n^* = P_n(A^T)r_0^*. \tag{3-17}$$

一般的に，Bi-CG 法は，その効率の観点から次の二つ改良すべき点が挙げられる．

- クリロフ部分空間$K_{n+1}(A^T;r_0^*)$ (ベクトル列r_0^*, r_1^*, $\cdots$, r_n^*) を作るために，A^Tとベクトルの積を反復毎に計算しなければならないこと，
- ベクトルr_n^*は，残差r_nとともにゼロに収束するが，そのことはアルゴリズムで直接に生かされていないこと．

4　積型反復解法とその一般化

Bi-CG 法を改良する研究において，先駆的な成果の一つとして IDR 法 (Induced Dimension Reducion method) がある [27]．応用問題に効率的でなかったためか，IDR 法はあまり知られていない．しかし，この研究に現れた手法はのちに，積型反復解法の開発にヒントを与え，IDR 法その自身は著名な Bi-CGSTAB 法に再整理された [26]．その後，Bi-CG 法の再構成に当たって，積型反復解法として最も早く脚光を浴びたのは自乗共役勾配法 (Conjugate Gradient-Squared method, CGS 法) である [21]．CGS 法では，残差r_nを構成するために，Bi-CG 法に現れたランチョス多項式$R_n(\lambda)$は次のように使われる．

$$r_n := R_n(A)R_n(A)r_0. \tag{4-1}$$

ここで，$R_n(A)R_n(A)$という二つの多項式の積の形をとることで，Bi-CG 法を構成したときに必要となったクリロフ部分空間$K_n(A^T, r_0^*)$は作らなくてもよいことになった．しかし，CGS 法が高い収束性を持っていることが多くの数値実験で実証されていた一方で [12, 14]，それが不安定なこともよく知られている [26, 29]．このような CGS 法の収束性のよい面/わるい面の両面性が考慮され，Bi-CGSTAB 法と名付けられた積型反復解法は考案された [26]．Bi-CGSTAB 法では，下記の漸化式を満たす多項式$Q_n(A)$

$$Q_{n+1}(\lambda) = (1 - \omega_n \lambda)Q_n(\lambda). \tag{4-2}$$

が新たに導入され，それとランチョス多項式との積から残差r_nは定義された．

$$r_n := Q_n(A)R_n(A)r_0. \tag{4-3}$$

ここで，パラメータω_nは残差r_{n+1}のノルムを最小にするように決められる．Bi-CGSTAB 法は優れた収束性を示している [2, 5, 6, 7, 29].

実際には，Bi-CGSTAB 法の多項式$Q_n(\lambda)$ は GMRES(1) 法の残差多項式であるとでも解釈できる．したがって，$Q_n(\lambda)$ の代わりに GMRES(k) 法の多項式を用いると，Bi-CGSTAB 法を一般化したアルゴリズム Bi-CGSTAB(L) を構成することができる [19]．例えば，GMRES(2) 法の多項式τ_n

(4-4) $\tau_{2n+1}(\lambda) := (1-\chi_n\lambda)\tau_{2n}(\lambda),$

(4-5) $\tau_{2n+2}(\lambda) := (\zeta_n+\eta_n\lambda)\tau_{2n+1}(\lambda) + (1-\zeta_n)\tau_{2n}(\lambda).$

を使うと，Bi-CGSTAB2 法は導出できる．そのとき，残差r_nは次のように定義される [4].

(4-6) $$r_n := \tau_n(A)R_n(A)r_0.$$

Bi-CGSTAB(L) 法は，GMRES(k) 法の多項式を利用するため，GMRES(k) 法と同じように多くの計算量と記憶容量，しかも反復毎に最小二乗法を解く必要がある．

4.1 積型反復解法の定義

そこで，我々は Bi-CGSTAB(L) 法と違ったアプローチで，ランチョス・プロセスに基づく積型反復解法を一般化する手法を次のように提案する．

まず，Bi-CG 法の残差$r_0, r_1, \cdots, r_n$列が収束するとする．そのとき，我々は適当な手続きで多項式列$H_0, H_1, \cdots, H_n$を生成し，$H_0(A)r_0$, $H_1(A)r_1, \cdots, H_n(A)r_n$を用いて，$r_0$, $r_1, \cdots, r_n$の収束の加速をはかることを考える．ただし，H_nは最高次係数がゼロでない n 次多項式で，条件 $H_n(0)=1$ を満たす．

そこで，我々は$H_n(A)r_n$を近似解の残差と特徴付ける解法を積型反復解法と呼ぶ．言い換えれば，積型反復解法の残差r_nを次の式で定義する．

(4-7) $$r_n := R_n(A)H_n(A)r_0 = b - Ax_n.$$

多項式H_nがわかれば，式 (4-7) から残差r_nを計算し，さらに近似解x_nを求めることができる．多項式H_nがR_nになるとき，CGS 法は得られる．多項式H_nがQ_n/τ_nになると，Bi-CGSTAB 法/Bi-CGSTAB2 法は得られる．

しかし，積型反復解法のアルゴリズムを具体的に作る段階で，次の問題点を解決しなければならない．

(A) まず，ランチョス多項式R_nを決定するために必要となるパラメータα_nとβ_nを計算する公式を導く．

(B) そして，多項式列H_nを逐次的に計算する公式を設計する．これが最も重要なことである．

また，より効率のいい積型反復解法を作るために，我々は以下の観点から多項式H_nの設計方針を与える．

(C) 解法の記憶容量，反復毎の演算量を抑える意味で，H_nを短い漸化式で計算でき，さらに安定した収束性能を持つように，多項式H_nを決定するために必要となるパラメータを計算する公式を与える．

4.2 残差多項式の再構成

ランチョス・プロセスに現れた 3 項漸化式は我々に新しい積型反復解法を生み出すヒントを与えてくれた．ここで，我々は二つ独立な変数ζ_nとη_nを導入し，新しい多項式H_nを次の 3 項漸化式を満たすように設計する．

(4-8) $$H_0 := 1,$$

(4-9) $$H_1 := (1-\zeta_0\lambda)H_0,$$

(4-10) $$H_{n+1} := (1+\eta_n-\zeta_n\lambda)H_n - \eta_n H_{n-1}.$$

ただし，ζ_nとη_nとは未確定のパラメータである．

4.3 漸化式による反復ベクトルの計算

交替漸化式 (2-26)~(2-28)，またH_nの定義から，多項式族R_nH_n，P_nH_n，$R_{n+1}H_n$，$R_{n+}H_{n-1}$，$P_{n+1}H_n$ に関する漸化式を作ることができる．

$$
\begin{array}{llcl}
\text{(4-11)} & R_{n+1}H_{n+1} & = & (1+\eta_n)R_{n+1}H_n \\
 & & - & \eta_n R_{n+1}H_{n-1}\zeta_n - \lambda R_{n+1}H_n; \\
\text{(4-12)} & P_{n+1}H_{n+1} & = & R_{n+1}H_{n+1} + \beta_n(1+\eta_n)P_nH_n \\
\text{(4-13)} & & - & \beta_n\eta_n P_nH_{n-1} - \beta_n\zeta_n\lambda P_nH_n; \\
\text{(4-14)} & R_{n+1}H_n & = & R_nH_n - \alpha_n\lambda P_nH_n; \\
\text{(4-15)} & R_{n+1}H_{n-1} & = & R_nH_{n-1} - \alpha_n\lambda P_nH_{n-1}; \\
\text{(4-16)} & P_{n+1}H_n & = & R_{n+1}H_n + \beta_nP_nH_n.
\end{array}
$$

ここで，下記の補助ベクトルを用意し，

$$p_n := P_n(A)H_n(A)r_0, \quad w_n := P_{n+1}(A)H_n(A)r_0,$$
$$t_n := R_{n+1}(A)H_n(A)r_0, \quad s_n := R_{n+1}(A)H_{n-1}(A)r_0,$$

式(4-11)~(4-16)から式(4-7)で定義された積型反復解法の残差r_nを計算する漸化式を得ることができる．

$$\begin{aligned}
&(4\text{-}17)\quad r_{n+1} &=& \ (1+\eta_n)t_n-\eta_n s_n-\zeta_n At_n;\\
&(4\text{-}18)\quad p_{n+1} &=& \ r_{n+1}+\beta_n(1+\eta_n)p_n-\beta_n\eta_n w_{n-1}\\
& &-& \ \beta_n\zeta_n Ap_n;\\
&(4\text{-}19)\quad t_n &=& \ r_n-\alpha_n Ap_n;\\
&(4\text{-}20)\quad s_n &=& \ t_{n-1}-\alpha_n Aw_{n-1};\\
&(4\text{-}21)\quad w_n &=& \ t_n+\beta_n p_n.
\end{aligned}$$

さらに残差r_nの定義式(4-7)と漸化式(4-17)から，近似解x_nの漸化式を得ることができる．

$$\begin{aligned}
(4\text{-}22)\quad x_{n+1} &= -\eta_n(x_{n-1}+\alpha_{n-1}p_{n-1}+\alpha_n w_{n-1})\\
&+ (1+\eta_n)(x_n+\alpha_n p_n)+\zeta_n t_n.
\end{aligned}$$

4.4 α_nとβ_nの計算式

多項式H_nの最高次項の係数が$(-1)^n\prod_{i=0}^{n-1}\zeta_i$であるので，次の式が成立することがわかる．

$$(4\text{-}23)\quad (r_0^*,r_n)=\left((-1)^n\prod_{i=0}^{n-1}\zeta_i\right)((A^T)^n r_0^*,R_n(A)r_0),$$

$$(4\text{-}24)\quad (r_0^*,Ap_n)=\left((-1)^n\prod_{i=0}^{n-1}\zeta_i\right)((A^T)^n r_0^*,AP_n(A)r_0).$$

したがって，式(3-6)と(3-10)から，α_nとβ_nの計算式がわかる．

$$(4\text{-}25)\quad \alpha_n=\frac{(r_0^*,r_n)}{(r_0^*,Ap_n)},\ \beta_n=\frac{\alpha_n}{\zeta_n}\cdot\frac{(r_0^*,r_{n+1})}{(r_0^*,r_n)}.$$

この節では，本研究の目的の一部分，即ち，多項式の設計を完成し，パラメータα_nとβ_nの計算式を与えた．それから，パラメタζ_nとη_nは適当な手段で与えられるとすると，新しい積型反復解法を導き出すことができる．これは第5節で論じる．

5 積型反復解法の様々な変形

5.1 既存解法の再導出

5.1.1 CGS法

$\zeta_n=\alpha_n,\eta_n=\beta_n$のときの積型反復解法を考える．そのとき，$H_n(\lambda)=R_n(\lambda)$となり，残差$r_n=R_n(A)R_n(A)r_0$となる．これは式(4-1)と一致する．ここで，CGS法の導出を割愛するが，詳細な導出については，[15, 21]に参照する．

CGS法は次のようにまとめられる．

ALGORITHM 3　CGS法

Choose an initial guess x_0,
and set $r_0^*=p_0=e_0=r_0=b-Ax_0$.
For $n=0,1,\cdots$ until $\|r_n\|\le\epsilon\|b\|$ do:

$$\begin{aligned}
&\alpha_n=\frac{(r_0^*,r_n)}{(r_0^*,Ap_n)},\\
&h_{n+1}=e_n-\alpha_n Ap_n,\\
&x_{n+1}=x_n+\alpha_n(e_n+h_{n+1}),\\
&r_{n+1}=r_n-\alpha_n A(e_n+h_{n+1}),\\
&\beta_n=\frac{(r_0^*,r_{n+1})}{(r_0^*,r_n)},\\
&e_{n+1}=r_{n+1}+\beta_n h_{n+1},\\
&p_{n+1}=e_{n+1}+\beta_n(h_{n+1}+\beta_n p_n);
\end{aligned}$$

5.1.2 Bi-CGSTAB法

$\eta_n=0$とおき，ζ_nを残差r_{n+1}のノルムを最小にするように決めることを考える．そのとき，残差r_nは式(4-3)と一致する．ここで，Bi-CGSTAB法の導出を割愛するが，詳細な導出については，[26, 28]に参照する．

Bi-CGSTAB法は次のようにまとめられる．

ALGORITHM 4　Bi-CGSTAB法

Choose an initial guess x_0,
and set $r_0^*=p_0=r_0=b-Ax_0$.
For $n=0,1,\cdots$, until $\|r_n\|\le\epsilon\|b\|$ do:

$$\begin{aligned}
&\alpha_n=\frac{(r_0^*,r_n)}{(r_0^*,Ap_n)},\\
&t_{n+1}=r_n-\alpha_n Ap_n,\\
&\zeta_n=\frac{(t_{n+1},At_{n+1})}{(At_{n+1},At_{n+1})}\\
&x_{n+1}=x_n+\alpha_n p_n+\zeta_n t_{n+1},\\
&r_{n+1}=t_{n+1}-\zeta_n At_{n+1},\\
&\beta_n=\frac{\alpha_n}{\zeta_n}\cdot\frac{(r_0^*,r_{n+1})}{(r_0^*,r_n)},\\
&p_{n+1}=r_{n+1}+\beta_n(p_n-\zeta_n Ap_n);
\end{aligned}$$

5.2 新しい解法の導出

5.2.1 GPBi-CG(ω)法

我々は$\eta_n=\omega$と固定し，ζ_nを残差r_{n+1}のノルムを最小にするように決めることを考える．そのときの積型反復解法をGPBi-CG(ω)法と名付ける．

GPBi-CG(ω)法は次のようにまとめられる．

ALGORITHM 5　GPBi-CG(ω) 法

Choose an initial guess x_0,
and set $r_0^* = p_0 = r_0 = b - Ax_0,\ w_{-1} = t_{-1} = 0$.
For $n = 0, 1, \cdots$ until $\| r_n \| \le \varepsilon \| b \|$ do :

$$\alpha_n = \frac{(r_0^*, r_n)}{(r_0^*, Ap_n)},$$
$$s_n = t_{n-1} - \alpha_n A w_{n-1},$$
$$t_n = r_n - \alpha_n A p_n,$$
$$\zeta_n = \frac{(t_n + \omega(t_n - s_n), At_n)}{(At_n, At_n)},$$
$$x_{n+1} = x_n + \alpha_n p_n + \zeta_n t_{n+1} + \omega(x_n - x_{n-1} + \alpha_n p_n - \alpha_{n-1} p_{n-1} - \alpha_n w_{n-1}),$$
$$r_{n+1} = t_n + \omega(t_n - s_n) - \zeta_n A t_n,$$
$$\beta_n = \frac{\alpha_n}{\zeta_n} \cdot \frac{(r_0^*, r_{n+1})}{(r_0^*, r_n)},$$
$$p_{n+1} = r_{n+1} + \beta_n(p_n - \zeta_n A p_n + \omega(p_n - w_{n-1})),$$
$$w_n = t_n + \beta_n p_n;$$

5.2.2 GPBi-CG 法

我々は式(4-17) に対して残差 r_{n+1} のノルムを最小にするように ζ_n，η_n を決めることを考える．そのときの積型反復解法を GPBi-CG 法と名付ける．

GPBi-CG 法は次のようにまとめられる．

ALGORITHM 6　GPBi-CG 法

Choose an initial guess x_0,
and set $r_0^* = p_0 = r_0 = b - Ax_0,\ w_{-1} = t_{-1} = 0$.
For $n = 0, 1, \cdots$ until $\| r_n \| \le \varepsilon \| b \|$ do :

$$\alpha_n = \frac{(r_0^*, r_n)}{(r_0^*, Ap_n)},$$
$$s_n = t_{n-1} - \alpha_n A w_{n-1},$$
$$t_n = r_n - \alpha_n A p_n,$$
$$\zeta_n = \frac{(s_n - t_n, s_n - t_n)(t_n, At_n) - (s_n - t_n, At_n)(t_n, s_n - t_n)}{(At_n, At_n)(s_n - t_n, s_n - t_n) - (s_n - t_n, At_n)^2},$$
$$\eta_n = \frac{(At_n, At_n)(t_n, s_n - t_n) - (s_n - t_n, At_n)(t_n, At_n)}{(At_n, At_n)(s_n - t_n, s_n - t_n) - (s_n - t_n, At_n)^2},$$
$$x_{n+1} = x_n + \alpha_n p_n + \zeta_n t_n + \eta_n(x_n - x_{n-1} + \alpha_n p_n - \alpha_{n-1} p_{n-1} - \alpha_n w_{n-1}),$$
$$r_{n+1} = t_n + \eta_n(t_n - s_n) - \zeta_n A t_n,$$
$$\beta_n = \frac{\alpha_n}{\zeta_n} \cdot \frac{(r_0^*, r_{n+1})}{(r_0^*, r_n)},$$
$$p_{n+1} = r_{n+1} + \beta_n(p_n - \zeta_n A p_n + \eta_n(p_n - w_{n-1})),$$
$$w_n = t_n + \beta_n p_n;$$

6 あとがき

本研究では，我々は数値解析において最も重要な原理の一つ，ランチョス・プロセスから CG 法と Bi-CG 法の導出について述べた．ランチョス・プロセスに基づく既存解法の得失を評価した上で，我々は積型反復解法を新しく定義し，積型反復解法を一般化する手法を提案した．最後に，著名な CGS 法，Bi-CGSTAB 法を含む積型反復解法の実用的なアルゴリズムをその一般化した観点から導き出した．紙数の制限で，本研究で割愛した前処理付きのアルゴリズムと GPBi-CG 法の改良版を別機会に紹介させていただきたい．

参考文献

[1] Arnoldi, W. E., *The principle of minimized iteration in the solution of the matrix eigenvalue problem*, Quart. Appl. Math., 9(1951), pp. 17-29.

[2] Driessen, M. and Van der Vorst, H. A., Bi-CGSTAB *in semiconductor modelling*, in: W. Fichtner(ed.), Simulation of semiconductor devices and processes, 4(1991), pp. 45-54.

[3] Fletcher, R., *Conjugate gradient methods for indefinite systems*, Lecture Notes in Mathematics 506(1976), pp. 73-89.

[4] Gutknecht, M. H., *Variants of Bi-CGSTAB for matrices with complex spectrum*, SIAM J. Sci. Comput., 14(1993), pp. 1020-1033.

[5] 藤野 清次，熱流体解析で現れる非対称連立 1 次方程式の新解法,「第 28 回日本伝熱シンポジウム」講演論文集 (1991), pp. 622-624.

[6] 藤野 清次，松本 直樹，水藤 寛，Bi-CGSTAB 法の流体解析への応用,「第 5 回数値流体力学シンポジウム」講演論文集 (1991), pp. 501-504.

[7] Fujino, S. and Zhang, S. L., *Analysis on convergence behaviour of the* CGS *and* Bi-CGSTAB *method*, Computer Arithmetic and Enclosure Methods (IMACS, 1992) (L. Atanassova, ed.), pp.381-390.

[8] Hestenes, M. R. and Stiefel, E., *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49(1952), pp. 409-435.

[9] Lanczos, C., *Solution of systems of linear equations by minimized iterations*, J. Res. Nat. Bur. Standards, 49(1952), pp. 33-53.

[10] Meijerink, J. A. and Van der Vorst, H. A., *An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix*, Math. Comput., 31(1981), pp. 148-162.

[11] Meijerink, J. A. and Van der Vorst, H. A., *Guidelines for the usage of incomplete decompositions in solving sets of linear equations as they occur in practical problems*, J. Comput. Phys., 44(1981), pp. 134-155.

[12] 村田 健郎, 名取 亮, 唐木 幸比古, 大型数値シミュレーション, 岩波書店 (1990).

[13] 森 正武, 杉原 正顯, 室田 一雄, 線形計算「岩波講座, 応用数学」, 岩波書店 (1994).

[14] Nachtigal, N. M., Reddy, S. C. and Trefethen, L. N., *How fast are nonsymmetric matrix iterations?*, SIAM J. Matrix Anal. Appl., 13(1992), pp. 778-795.

[15] 名取 亮, 数値解析とその応用「コンピュータ数学シリーズ (15)」, コロナ社 (1990).

[16] Reid, J. K., *On the method of conjugate gradients for the solution of large sparse systems of linear equations*, Pro. the Oxford conference of institute of mathematics and its applications(1971), pp. 231-254.

[17] Saad, Y., *Variations on Arnoldi's method for computing eigenelements of large nonsymmetric matrices*, Lin. Alg. Appl., 34(1980), pp. 269-295.

[18] Saad, Y. and Schultz, M. H., GMRES: *A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7(1986), pp. 856-869.

[19] Sleijpen, G.L.G. and Fokkema, D.R., BICGSTAB(L) *for Linear Equations Involving Unsymmetric Matrices with Complex Spectrum*, Electronic Transactions on Numer. Anal. 1(1993), pp. 11-32.

[20] Van der Sluis, A. and Van der Vorst, H. A., *The rate of convergence of conjugate gradients*, Numer. Math. 48(1986), pp. 543-560.

[21] Sonneveld, P., CGS, *A fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 10(1989), pp. 26-52.

[22] Stiefel, E. L., *Kernel polynomial in linear algebra and their numerical applications*, in: Further contributions to the determination of eigenvalues, NBS Applied Math. Ser., 49(1958), pp. 1-22.

[23] 杉原 正顯, CG 法と Gram-Schmidt の直交化法, 筑波大学数値解析研究室輪講資料 (86 年 10 月 7 日).

[24] 高橋 秀俊, Lanczos の原理と数値計算, 数理科学, No.157(1976), pp.25-31.

[25] Van der Vorst, H. A., *Preconditioning by incomplete decompositions*, Ph. D. Thesis, University of Utrecht, The Netherlands(1982).

[26] 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), pp. 631-644.

[27] Wesseling, P., and Sonneveld, P., *Numerical experiments with a multiple grid and a preconditioned Lanczos type method*, in: R. Rautmann(ed.), Approximation methods for Navier-Stokes problems, Lecture Notes in Math.(1980), 543-562.

[28] 張紹良, 藤野清次, 反復解法の収束特性と計算効率, 情報処理学会研究報告 Vol.91, No.61, pp. 91-98.

[29] 張紹良, 藤野清次, 丸め誤差の分離に基づく共役勾配系統の解法の考察, 日本応用数理学会論文誌, 3(1993), pp. 135-146.

[30] Zhang, S. L., GPBi-CG: *Generalized Product-type Methods Based on* Bi-CG *for Solving Nonsymmetric Linear Systems.* Submitted to SIAM J. Sci. Stat. Comput.