適応的に kを決定する ML(k)BiCGStab 法

五条方 昇* 野寺 隆†

1 はじめに

BiCGStab 法 [1] は，非対称行列を係数にした連立 1 次方程式

$$A\boldsymbol{x} = \boldsymbol{b}, \quad A \in \boldsymbol{C}^{n\times n} \quad \boldsymbol{x}, \boldsymbol{b} \in \boldsymbol{C}^n \tag{1}$$

を解くための反復解法の 1 つである．ただし，この係数行列 A は正則とする．

BiCGStab 法はランチョス原理に基づく BiCG 法 [2] から得られる．ここで，この BiCG 法に k個の新しいクリロフ部分空間 $\mathcal{K}(\boldsymbol{q}_1, A^T), \mathcal{K}(\boldsymbol{q}_2, A^T), \ldots, \mathcal{K}(\boldsymbol{q}_k, A^T)$ を組み合わせた新しい算法が ML(k)BiCG 法 [4, 5] である．このような組み合わせをすると，ML(k)BiCG 法はクリロフ部分空間の次元が大きい場合でも，低次元のクリロフ部分空間の集合として計算することができるので，算法の安定性と頑強性を強化させることができる．また，従来の算法よりも少ない反復回数で解を求めることができる．ML(k)BiCGStab 法 [4] は，この ML(k)BiCG 法を改良することで得られる．ML(k)BiCGStab 法の算法を図 1に示す．

しかし，ML(k)BiCGStab 法は kの値が大きいと，1 ステップあたりの計算量が増えてしまう．従って，たとえ反復回数を減らせたとしても，結果的には従来の方法より計算時間が大幅に増加することがある．そこで，本稿では，不適切な kの値を選ぶことによる計算時間の大幅な増加を防ぐために，kの値を反復の途中で自動的に決定する方法について提案する．

2 残差ノルムのスムージング

BiCGStab 法の場合，一般的には残差ノルムは激しく振動し，単調には減少しない．そこで，残差ノルムを単調に減少させる方法について考える．残差ノルムのスムージングの方法にもいろいろあるが，ここでは最小残差ノルムのスムージング [3] を利用する．

今，式 (1) の lステップ後の近似解を$\boldsymbol{x}_l$，lステップ後の残差を$\boldsymbol{r}_l = \boldsymbol{b} - A\boldsymbol{x}_l$ とする．ここで，補助ベクト

*慶應義塾大学大学院理工学研究科
†慶應義塾大学理工学部

choose x_0 **and** k **vectors** $q_1, q_2, \ldots, q_k$
$r_0 = b - Ax_0$; $g_0 = r_0$;
for $j = 0, 1, 2, \ldots$
  $w_{(j-1)k+k} = Ag_{(j-1)k+k}$;
  $c_{(j-1)k+k} = q_1^T w_{(j-1)k+k}$;
  $\alpha_{jk+1} = q_1^T r_{(j-1)k+k} / c_{(j-1)k+k}$;
  $u_{jk+1} = r_{(j-1)k+k} - \alpha_{jk+1} w_{(j-1)k+k}$;
  $\rho_{j+1} = -u_{jk+1}^T A u_{jk+1} / \|Au_{jk+1}\|$;
  $x_{jk+1} = x_{(j-1)k+k} - \rho_{j+1} u_{jk+1} + \alpha_{jk+1} g_{(j-1)k+k}$;
  $r_{jk+1} = \rho_{j+1} Au_{jk+1} + u_{jk+1}$;
  for $i = 1, 2, \ldots, k$
    $z_d = u_{jk+i}$; $z_g = r_{jk+i}$; $z_w = 0$;
    for $s = i, \ldots, k-1$ **and** $j \geq 1$
      $\beta^{(jk+i)}_{(j-1)k+s} = -q_{s+1}^T z_d / c_{(j-1)k+s}$;
      $z_d = z_d + \beta^{(jk+i)}_{(j-1)k+s} d_{(j-1)k+s}$;
      $z_g = z_g + \beta^{(jk+i)}_{(j-1)k+s} g_{(j-1)k+s}$;
      $z_w = z_w + \beta^{(jk+i)}_{(j-1)k+s} w_{(j-1)k+s}$;
    end
    $\beta^{(jk+i)}_{(j-1)k+k} = -\frac{q_1^T (r_{jk+1} + \rho_{j+1} z_w)}{\rho_{j+1} c_{(j-1)k+k}}$;
    $z_g = z_g + \beta^{(jk+i)}_{(j-1)k+k} g_{(j-1)k+k}$;
    $z_w = \rho_{j+1} \left(z_w + \beta^{(jk+i)}_{(j-1)k+k} w_{(j-1)k+k} \right)$;
    $z_d = r_{jk+1} + z_w$;
    for $s = 1, \ldots, i-1$
      $\beta^{(jk+i)}_{jk+s} = -q_{s+1}^T z_d / c_{jk+s}$;
      $z_d = z_d + \beta^{(jk+i)}_{jk+s} d_{jk+s}$;
      $z_g = z_g + \beta^{(jk+i)}_{jk+s} g_{jk+s}$;
    end
    $d_{jk+i} = z_d - u_{jk+i}$;
    $g_{jk+i} = z_g - z_w$;
    if $i < k$
      $c_{jk+i} = q_{i+1}^T d_{jk+i}$;
      $\alpha_{jk+i+1} = q_{i+1}^T u_{jk+i} / c_{jk+i}$;
      $u_{jk+i+1} = u_{jk+i} - \alpha_{jk+i+1} d_{jk+i}$;
      $x_{jk+i+1} = x_{jk+i} + \rho_{j+1} \alpha_{jk+i+1} g_{jk+i}$;
      $w_{jk+i} = Ag_{jk+i}$;
      $r_{jk+i+1} = r_{jk+i} - \rho_{j+1} \alpha_{jk+i+1} w_{jk+i}$;
    end
  end
end

図 1 ML(k)BiCGSTAB 法の算法

$s_0 = r_0$; $y_0 = x_0$;
for $l = 1, 2, \ldots$
  Compute x_l and r_l;
  Compute η_l by (2);
  $s_l = s_{l-1} + \eta_l (r_l - s_{l-1})$;
  $y_l = y_{l-1} + \eta_l (x_l - y_{l-1})$;
end

図 2 最小残差ノルムのスムージングの算法

ル$\boldsymbol{y}_l$を次のように定義する.

$$\begin{aligned} \boldsymbol{y}_0 &= \boldsymbol{x}_0, \\ \boldsymbol{y}_l &= (1-\eta_l)\boldsymbol{y}_{l-1} + \eta_l \boldsymbol{x}_l, \quad l = 1, 2, \ldots, \end{aligned}$$

ここで，η_lは $\|\boldsymbol{r}_l\|_2 = \|\boldsymbol{b} - A\boldsymbol{y}_l\|_2 = \|\boldsymbol{b} - A((1-\eta_l)\boldsymbol{y}_{l-1} + \eta_l \boldsymbol{x}_l)\|_2$ を最小にするように選ばれる．よって，

$$\eta_l = -\frac{\boldsymbol{s}_{l-1}^T(\boldsymbol{r}_l - \boldsymbol{s}_{l-1})}{\| \boldsymbol{r}_l - \boldsymbol{s}_{l-1} \|_2}, \tag{2}$$

となる．ここで$\boldsymbol{s}_{l-1} = \boldsymbol{b} - A\boldsymbol{y}_{l-1}$である．こうして求められたベクトル$\boldsymbol{y}_l, \boldsymbol{s}_l$をそれぞれ近似解$\boldsymbol{x}_l$，残差$\boldsymbol{r}_l$の代わりに使うことで，なめらかな残差ノルムの値が得られる．最小残差ノルムのスムージングの算法を図 2 に示す.

3 適応的に kを決定する方法

ML(k)BiCGStab 法の kを決定する材料として，残差ノルム $\|\boldsymbol{r}\|_2$を利用する方法を考える．しかし，ML(k)BiCGStab 法の場合，残差ノルムは単調には減少しないので，収束を判定する材料としては利用しにくい．そこで，前節の残差ノルムのスムージングを利用し，スムージングされた残差ノルム $\|\boldsymbol{s}\|_2$を判定条件として利用する方法について考える.

始め，kは 1 からスタートする．そして，反復の過程で$\boldsymbol{s}$のノルムを参照して，収束が停滞していると判断されたらkの値を増やす.

収束判定の基準としては，

$$\frac{\left| \| \boldsymbol{s}_l \|_2 - \| \boldsymbol{s}_{l-k} \|_2 \right|}{\| \boldsymbol{s}_l \|_2} < \delta \tag{3}$$

を利用することにする．式 (3) の左辺は，l回目と$l-k$回目の反復における ML(k)BiCGStab 法の相対残差ノルムを表している．これがδよりも小さくなったら残差ノルムの収束が停滞したとみなし，式 (3) が連続してある一定回数満たされたら，この kの値は不適切であると判断して，kの値を 1 つ増やす.

しかし，反復の途中で kの値を変えることは，ML(k)BiCGStab 法の算法を変えることになるので，収束に関して悪影響を及ぼすことが考えられる．そこで，式 (3) の判定はkが適正な値になったと判断されたら以後は行なわないことにする．すなわち，kの適正な値を$\tilde{k}$とすると，$k = \tilde{k}$となった後は ML($\tilde{k}$)BiCGStab 法の反復と同じ計算を行うことになる．これにより，$k = \tilde{k}$となったら残差ノルムのスムージングの計算も行わないことになるので，計算量の減少にもつながる.

また，kの適正な値が大きい場合，その kの値に到達するまでに多くの計算量が必要になることが予想される．たとえば，行列 A の次元を N，非ゼロ要素数を E，停滞判定回数を m とすると，kの値を 1 つ増やすのに必要な計算量は $((k+1)E + (8k^2+17k+9)N)m$ となり，これは kの値が大きくなるとかなりの負担となる．そのため，kの値の増やし方に工夫をする．最も単純なやり方としては，kを 1 ではなく n ずつ増やす方法が考えられる．しかし，この方法では k が変化するとき，ML(k)BiCGStab 法の算法を大きく変えることになる．そのため，良条件問題のような適性な kの値が小さい問題の場合に，必要以上に kの値を増加させてしまう危険性がある．そこで，kをある一定の割合で増加させていく方法を考える．この方法だと kの値が小さいと 1 ずつ増え，kが大きければ増加量も大きくなるので，kが適性の値に到達するまでの時間をある程度短縮させることが可能となる.

4 おわりに

ML(k)BiCGStab 法は，悪条件の問題を解く場合には，BiCGStab 法や GMRES(m) 法などの従来の算法よりも有効である．しかし，そうでない問題の場合，ML(k)BiCGStab 法は逆に従来の算法よりも収束性が悪くなる．これが kの値を適応的に決定することにより改善されるのか．このことに関し，当日は良条件問題，悪条件問題それぞれの数値実験を通じて結果を報告する.

参 考 文 献

[1] H. A. van der Vorst, *Bi-CGSTAB : A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems,* SIAM J. Sci. Statist. Comput., 12 (1992), pp. 631–644.

[2] R. Fletcher, *Conjugate gradient methods for indefinite systems,* in Proceedings of the Dundee Biennal Conference on Numerical Analysis 1974, G. A. Watson, ed., Springer-Verlag, New York, 1975, pp. 73–89.

[3] Lu Zhou and Homer F. Walker, *Residual smoothing techniques for iterative methods,* SIAM J. Sci. Comput. Vol. 15, No. 2, (1994), pp. 297–312.

[4] M. Yeung and T. F. Chan, *ML(k)BiCGSTAB: A BiCGSTAB variant based on multiple Lanczos starting vectors,* SIAM J. Sci. Comput., Vol. 21, No. 4, (1999), pp. 1263–1290.

[5] 五条方, 野寺, ML(k)BiCGSTAB 法の有効性について, 情処研報, Vol. 2001, No. 102, (2001), pp. 49–54.