テクニカルノート

リスタートによる疑似残差法の収束性の加速

野寺 隆[†] 稲津隆敏[†]

リスタートを用いて非対称行列系の近似解を求める疑似残差法の収束性を向上させる手法について述べる．これは，リスタートを周期的に用いることで，反復法の収束性を改善させることができ，AP1000 を利用した数値実験とともに報告する．

The Convergence Acceleration of Pseudo Residual Method Using a Restarted Procedure

TAKASHI NODERA[†] and TAKATOSHI INADU[†]

This paper explores the use of a restarted procedure combined with pseudo residual methods for solving nonsymmetric linear systems of equation. Such restarted procedure is easy to employ periodically and well suited to improving the convegence of iterative methods. Finally, the effectiveness of restarted procedure is demonstrated in a variety of numerical experiments on the parallel computer AP1000.

1. はじめに

近年，数理物理学だけでなく，計算機科学や経済学などの分野から生ずる大規模な連立一次方程式

$$Ax = b \tag{1}$$

の近似解を求めることが重要なテーマとなっている．計算機のハードウェアの急速な発展にともない，式 (1) の近似解を求める算法にも様々なものが提案されている[1]．特に，反復解法を使用して近似解を求める場合，係数行列が大型で正定値対称であるときには，共役勾配法（CG 法ともいう）が有効である[1),6)]．しかし，非対称行列の場合には，これが最良の算法というものは存在しない．その理由の 1 つは，非対称行列系の近似解を求める反復解法では，対称行列系に対する共役勾配法のような算法は構成できないことが示されているからである[2]．しかし，最良ではないが有効な算法はいろいろ提案されている[1]．疑似残差法（pseudo residual method, PRES 法ともいう）もその中の 1 つの算法である[3),7)]．

非対称行列系の反復解法の収束を改善する方法の 1 つに，行列の前処理をあげることができる．しかし，メッセージパッシング方式の並列計算機では，ある特殊な行列形式を除いて，有効な行列の前処理はいまだ提案されていない．たとえば，現時点では，対角スケーリングによる前処理，近似逆行列による前処理，ノイマン級数展開を用いた前処理，ヤコビ法による前処理，チェビチェフ多項式などによる前処理などあるが，あまり有効な手法ではない[3]．そこで，前処理なしのオリジナルな算法だけで並列計算機のパワーのみを使って式 (1) の近似解を求めることになる．その場合，使用する算法に修正を加えることにより幾分収束を早めることも可能であるが[5]，記憶領域に制約のある場合には利用できないこともある．

近年，我々は修正を加えないオリジナルな算法を繰り返しリスタートするシンプルな手法が，収束を向上させる有効な手段となることを見いだしたので数値例とともに報告する．

2. リスタートを用いた疑似残差法

大型の正則な非対称行列を係数とする連立一次方程式 (1) の近似解を求める疑似残差法は，次のよう述べることができる[3),4),7)]．

[疑似残差法（PRES Method）]

(1) 任意の初期値 x_0 を選び，初期残差 $r_0 = Ax_0 - b$

† 慶應義塾大学理工学部
Faculty of Science and Technology, Keio University

表1 前処理なしの疑似残差法
Table 1 PRES methods without preconditioning.

算法	σ_k
ORTHORES-E	k+1
ORTHORES-R(σ_{res})	$(k \bmod \sigma_{\mathrm{res}})+1$
ORTHORES-T($\sigma_{\max}$)	$\min(k+1,\sigma_{\max})$
ORTHORES-C($\sigma_{\max},\sigma_{\mathrm{res}}$)	—

を計算する．

(2) 以下の反復を $\|r_{k+1}\|_Z$ が収束するまで繰り返す ($k=1,2,3,\cdots$)．ただし，Z は任意の対称正定値行列とする．

$$d_k = Pr_k$$

（P：右側からの前処理行列）

$$\alpha_{i,k} = -\frac{r_{k+1-i}^T ZAd_k}{r_{k+1-i}^T Zr_{k+1-i}} \quad (1 \le i \le \sigma_k)$$

$$\bar{r}_{k+1} = Ad_k + \sum_{i=1}^{\sigma_k} \alpha_{i,k} r_{k+1-i}$$

（$\bar{r}_{k+1}$：疑似残差ベクトル）

$$\phi_k = 1 \Big/ \sum_{i=1}^{\sigma_k} \alpha_{i,k}$$

$$r_{k+1} = \phi_k \bar{r}_{k+1}$$

$$x_{k+1} = \phi_k \left(d_k + \sum_{i=1}^{\sigma_k} \alpha_{i,k} x_{k+1-i} \right)$$

この算法の中に現れる σ_k を疑似残差法のオーダと呼ぶ．疑似残差法はこのオーダの決定方法により，厳密版，再出発版，打切版，結合版等の算法のバリエーションがある．

前述の算法における行列 Z の決定方法であるが，正定値対称行列となる行列を選べばよい．なお，行列 P の選択をいろいろ変えることで，様々な疑似残差法の算法を導出できる．通常，$P=I$（単位行列）と選ぶと**表1**に示す算法が得られる．また，$P=A^T$ と置くと，ATPRES法（CGNR法）といわれる算法が得られる．

ここで，表1の算法について，少し説明を加えることにする．

(1) **ORTHORES-E**：右側前処理をしない厳密版である．理論的には係数行列 A の次元の反復回数 n で必ず収束する．しかし，残差ベクトルを n 本記憶する必要があるため，現実的な算法とはいえない．

(2) **ORTHORES-R(σ_{res})**：前処理なしの再出発版である．σ_k は最大 σ_{res} の値をとり，σ_{res} 回ごとに残差ベクトルを $r_{k+1}=Ax_{k+1}-b$ により再計算し，x_{k+1} $(=x_0)$ を初期近似解ベクトル，r_{k+1} $(=r_0)$ を初期残差ベクトルとして反復をリスタートする．この算法は σ_{res} 回ごとにリスタートしているので，計算誤差の累積を抑え，計算に必要な記憶領域も少なくてすむ．厳密版と比較して，行列 A の次元の反復回数 n で必ず収束する保証はない．

(3) **ORTHORES-T($\sigma_{\max}$)**：前処理をしない打切版である．σ_k は最大 $\sigma_{\max}$ の値をとり，$k \ge \sigma_{\max}-1$ において最新の $\sigma_{\max}$ 本の残差ベクトルを使って，新しい残差ベクトル r_{k+1} の計算を行う．計算に必要となる記憶領域は，ORTHORES-R(σ_{res}) と同様に少なくてすむが，計算誤差の累積が起る可能性がある．この算法は，つねに $\sigma_{\max}$ 本の残差ベクトルを用いて疑似残差 $\|\bar{r}_{k+1}\|_Z$ を最小化しているので，残差ノルムが発散や振動を起こさなければ，再出発版より少ない反復回数で収束することが多い．

(4) **ORTHORES-C($\sigma_{\max},\sigma_{\mathrm{res}}$)**：この算法は，ORTHORES-T($\sigma_{\max}$) を σ_{res} 回ごとにリスタートする結合版である．この結合版は，打切版の残差ノルムの速い収束性に加え，再出発版の計算誤差の累積を受けにくい性質を持つ．よって，安定した収束をする可能性を有する算法である．

なお，ORTHORES法は基本的に疑似残差を最小とするので，真の残差を最小にはしない．数値例から，真の残差ノルムの収束性を観察すると振動することが分かる．ただし，後で述べる数値例から誤差ノルムは，残差ノルムと比較してほぼ安定した収束をするといえる．

3. 数 値 例

正方形領域 $\Omega=[0,1]\times[0,1]$ における二次元楕円型偏微分方程式のディレクレ境界値問題を考える．

$$u_{xx}+u_{yy}+au_x+bu_y=f(x,y)$$
$$u(x,y)|_{\partial\Omega}=0$$

ただし，a および b は実数係数とする．この方程式を5点中心差分を用いて離散化し，真の解を

$$u(x,y)=(1-e^x)(1-x)y(1-e^{1-y})$$

と設定して右辺を決定し，式(1)を作成する．このようにして連立一次方程式を作成したのは，残差ノルムだけでなく，誤差ノルムの収束性も合わせて観測するためである．メッシュは 256×256 の等間隔格子を用いて，格子点の番号付けは整合順序を用いている．この場合，方程式の次元は65536である．初期近似解は零ベクトルとし，収束判定条件は $\|r_{k+1}\|/\|r_0\| \le 1.0\times10^{-12}$ とする．また，最大の反復回数は10000回とする．ここでは疑似残差法の算法として，$Z=P=I$ と置くORTHORES法を用いて数値実験を行った．なお，すべての計算は64個のプロセッサを持つ富士通AP1000

表 2 ORTHORES 法の数値実験の結果
Table 2 Numerical results of ORTHORES methods.

算法	a	b	反復回数	時間 (sec)	最終残差ノルム $\|r_k\|$
ORTHORES–T(5)	3	5	oscillate	—	4.4×10^{-4}
	3	50	diverge	—	3.9×10^{-4}
	3	500	536	66	1.6×10^{-14}
	3	5000	487	60	3.5×10^{-14}
ORTHORES–R(5)	3	5	oscillate	—	4.1×10^{-2}
	3	50	5288	477	1.3×10^{-13}
	3	500	544	49	4.8×10^{-15}
	3	5000	1417	127	7.0×10^{-14}
ORTHORES–C(5,50)	3	5	3092	381	1.6×10^{-13}
	3	50	1219	152	5.0×10^{-14}
	3	500	491	59	2.8×10^{-14}
	3	5000	606	75	2.7×10^{-14}

を使用した．

表 2 は，$a = 3.0$ と固定し，b の値を $5.0 \leq b \leq 5000.0$ とした場合に，疑似残差法が収束するまでの反復回数，それにともなう計算時間 (sec)，最終残差ノルムの値をまとめた．疑似残差法は σ_k の選び方により，異なる収束をする．しかし，どの算法が最良であるかを一概に判断することはできない．たとえば，打切版である ORTHORES-T(5) 法は，b の値が小さい場合に残差ノルムの振動や発散が起り収束しない．また，再出発版の ORTHORES-R(5) 法は，同様に b の値が小さいときに残差ノルムに振動が発生し，収束しないことがある．

打切版を 50 回反復し，強制的に再出発を繰り返す結合版は，すべての b の値でほぼ安定した収束が得られている．その理由の 1 つには，「打切版の速い収束性」と「再出発版の計算誤差の累積の影響を受けにくい性質」とを兼ね備えているからである．

次に，**図 1** に $a = 3.0$，$b = 5.0$ とした場合の疑似残差法の残差ノルムの収束状況を示した．この図から，結合版は他の算法が収束しない場合に収束する可能性はあるが，リスタートの回数を増やせばよい収束が得られるものではないことが分かる．このように周期的にリスタートを繰り返す算法では，リスタートに入る時期，すなわち σ_{res} の決定がある程度収束の速さを握ることになる．我々の数値実験によれば $\sigma_{\mathrm{res}} = 50$ がかなり好成績をあげている．ただし，リスタートの回数をあまり増やしすぎると，残差ベクトルの再計算にかかる計算時間が増えるので，リスタートの回数はできるだけ少ない方がよい．

図 2 に誤差ノルム[4] $\|e_{k+1}\|/\|e_0\|$ の収束状況を示した．この図からリスタートを用いた結合版を使用する疑似残差法の誤差ノルムはスムーズに収束していることが分かる．

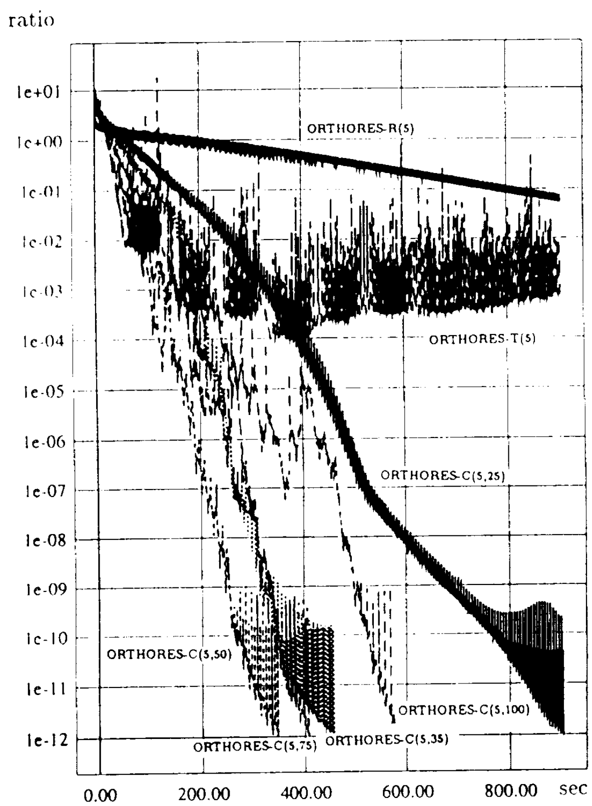

図 1 残差ノルム $\|r_{k+1}\|/\|r_0\|$ の収束性と計算時間 (sec)
Fig. 1 The behaviour of residual norm v.s. computational time (sec).

4. おわりに

非対称行列を係数とする連立一次方程式の近似解法として反復法を用いることができるが，その収束を加速する手法としてリスタートが有効であることを示した．本稿では，疑似残差法を使って収束の改善を行う有効性を示したが，共役残差法などの他の算法に対してもこの技法は有効である．

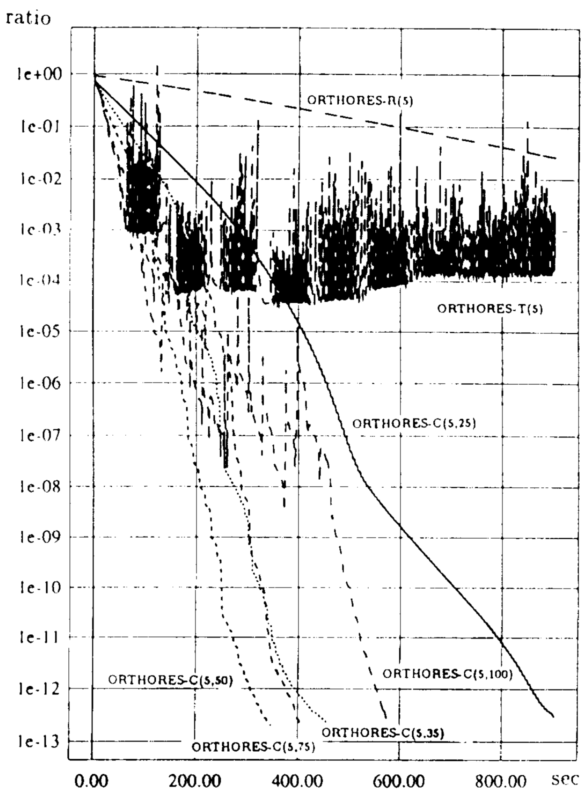

図2 誤差ノルム $\|e_{k+1}\|/\|e_0\|$ の収束性と計算時間 (sec)

Fig. 2 The behaviour of error norm v.s. computational time (sec).

周期的にリスタートを繰り返す手法は，残差ノルムの収束性を向上させる最も単純なアプローチであり，プログラミングも簡単である．しかし，残差ノルムが収束状態に入っているときにリスタートを行うと，収束をいくらか減速してしまうこともある．また，リスタートの回数が多いと計算時間のロスにつながる．リスタートは，残差ノルムが収束が停滞しているときに最も効果的である．そこで，我々は残差ノルムの収束性を観測しながら算法がどうしても必要とするリスタートのみを行う適応的にリスタートする手法を開発した[4]．

本稿で提案したように周期的にリスタートを行って残差ノルムの収束性を向上させることが可能である．さらに効率のよいリスタートを行って反復解法の収束性を向上させるには，算法の並列化とともに最適なリスタートの時期を見つけ，そこで算法を再出発させる適応的なリスタートを行うことも必要である．

参考文献

1) Bruaset, A.M.: A Survey of Preconditioned Iterative Methods, *Pitman Research Notes in Math. Series 328*, Longman Scientific & Technical (1995).
2) Faber, V. and Manteuffel, T.A.: Neccessary and Sufficient Conditions for the Existence of a Conjugate Gardient Method, *SIAM J. Numer. Anal.*, Vol.21, pp.352–362 (1984).
3) 稲津，野寺：非対称行列系に対する疑似残差法と前処理について，情報研報，Vol.95, No.118, pp.19–24 (1995).
4) 稲津，野寺：適応的なリスタートを用いた疑似残差法，情報処理学会論文誌掲載予定.
5) 野寺：オペレータ係数法について，京都大学数理解析研究所講究録，No.880 (1994).
6) Takahasi, H. and Nodera, T.: Variants of the Conjugate Gradient Method, *Keio Math. Sem. Rep.*, No.3, pp.63–68 (1977).
7) Weiss, R.: Properties of Generalized Conjugate Gradient Methods, *Numerical Linear Algebra with Applications*, Vol.1, No.1, pp.45–63 (1994).

(平成8年2月19日受付)
(平成8年4月12日採録)

野寺 隆（正会員）

1982年慶應義塾大学大学院工学研究科博士課程（数理工学専攻）修了．同年，同大学数理科学科助手，1989年講師となり現在に至る．その間，1986年より1年間米国スタンフォード大学客員教授．工学博士．大規模な行列計算の算法の研究開発に従事．ハイパフォーマンス・コンピューティングや文書処理に興味を持つ．著書に『楽々LaTeX』などがある．エッセイスト．SIAM，日本応用数理学会会員．

稲津 隆敏（正会員）

1970年10月25日生．1995年3月慶應義塾大学理工学研究科（数理科学専攻）修了．修士（工学）．同年，東芝入社．行列計算の算法の開発や，ハイパフォーマンス・コンピューティングに興味を持つ．