

大規模原子力流体コードにおける省通信 GMRES 法の収束性改善手法の検証

伊奈拓也^{†1} 井戸村泰宏^{†1} 真弓明恵^{†1} 山田進^{†1}

概要 : エクサスケールコンピュータでは逐次演算性能の飛躍的向上により集団通信がボトルネックになることが指摘されている。 Krylov 部分空間法では基底ベクトルの直交化に集団通信を必要とするためエクサスケールコンピュータでストロングスケールを達成することが困難になる。この問題の解決策として省通信 GMRES が提案されている。省通信 GMRES では一度に複数の基底ベクトルを生成してまとめて直交化することで集団通信の回数を削減するが、基底ベクトルの線形独立性が崩れると収束性が悪化する。本研究では基底ベクトルの生成方法と QR 分解の手法を適切に選択することで収束性を改善する。

キーワード : 並列計算, Krylov 部分空間法

Verification of convergence improvement methods for the communication-avoiding generalized minimal residual method on a large scale nuclear fluid code

Takuya Ina^{†1} Yasuhiro Idomura^{†1} Akie Mayumi^{†1} Susumu Yamada^{†1}

Abstract: On exascale computers, collective communications become a major bottleneck due to dramatic improvement of computational performance. Because collective communications are necessary for the orthogonalization process of the basis vectors in Krylov subspace methods, it is difficult to achieve strong scaling on exascale computers. To resolve this issue, the communication avoiding generalized minimal residual method (CA-GMRES) has been proposed. CA-GMRES reduces the number of collective communications by first generating multiple basis vectors and then orthogonalizing them at once. However, when basis vectors lose their linear independence, the convergence becomes worse. This study improves the convergence characteristics for CA-GMRES by a proper choice of generation methods for basis vectors and QR decomposition methods.

Keywords : parallel computing, Krylov subspace

1. はじめに

Krylov 部分空間法では基底ベクトルの直交化のために集団通信を必要とするが、エクサスケールスーパーコンピュータではメニーコアプロセッサによる演算処理の高速化により、集団通信がボトルネックになることが問題になっている。そのため、集団通信の回数を減らすアルゴリズムとして省通信 GMRES 法[1]が提案されている。省通信 GMRES 法は一度に複数の基底ベクトルを生成することで集団通信の回数を減らすアルゴリズムである。しかし、一度に多くの基底ベクトルを生成した場合は線形独立性が崩れやすく収束性を悪化させてしまう問題がある。直交多項式を用いて基底ベクトルを生成することで線形独立性を保ち収束性の悪化を防ぐ方法が知られている。また、省通信 GMRES 法では基底ベクトルの QR 分解を精度よく行うことで直交化したベクトルの線形独立性を確保して収束性の悪化を防ぐことも可能である。大規模原子力流体コード GT5D[2]を対象に上記の基底ベクトル生成手法と QR 分解改善手法を

組み合わせ、収束特性の検証を行う。

2. GT5D

GT5D はプラズマ乱流とプラズマ粒子分布関数の時間発展を移流拡散方程式とポアソン方程式から構成される第一原理モデルに基づいて計算するプログラムである。GT5D において最も計算時間がかかる部分は移流拡散方程式の移流項の計算である。移流項 g の式を示す。

$$g[f] = -\mathbf{U}_1 \cdot \frac{\partial f}{\partial \mathbf{R}} - U_2 \frac{\partial f}{\partial v_{||}} \quad (1)$$

移流項 g は 3 次元の位置微分と 1 次元の速度微分の項で表される。粒子分布関数 f の時間積分に半陰的ルンゲ・クッタ方を適用するため上記移流項が与える非対称疎行列の連立一次方程式を解く必要があり、前処理なし一般化共役残差法を使用している。文献[3]では一般化共役残差法と省通信 GMRES 法の性能比較が行われ、省通信 GMRES 法の優位性が報告されている。

3. 省通信 GMRES 法

省通信 GMRES 法のアルゴリズムを図 1 に示す。省通信

^{†1} 国立研究開発法人日本原子力研究開発機構
Japan Atomic Energy Agency.

GMRES 法は一般に知られるリスタート付き GMRES 法と同じ収束性を持つ.一度に複数の基底ベクトルを生成して直交化プロセスで通信削減型の QR 分解を利用することで集団通信回数を削減する.省通信 GMRES 法では QR 分解から得られる上三角行列 R のみが必要であり直交行列 Q は不要である. 解ベクトル x_{i+1} の更新時に $Q_i y_i$ の計算が必要となるが, $Q_i = V_i R_i^{-1}$ の関係を利用することで $t_i = R_i^{-1} y_i$, $V_i t_i$ と 2 回の行列ベクトル積を計算することで Q_i を陽に計算せずに解ベクトル x_{i+1} を更新する.これにより, 演算数とメモリアクセス数を削減する[3].

```

B = [e2, e3, ..., es+1]
for i = 0, ..., until ||r_i|| < ε do

    r_i = b - Ax_i, β = ||r_i||, q_i = r_i / β, ζ = ( z_1, z_2, ..., z_s, z_{s+1} ) = ( β, 0, ..., 0 )

    if( (i ≠ 0) && (NewtonBasis == ON) )
        B = change of basis Matrix (E_{i-1})
        V_j = produces s more basis vectors(q_i)
            = [ρ_0(A)q_i, ρ_1(A)q_i, ..., ρ_s(A)q_i] = [V_i, ρ_s(A)q_i]
    else
        V_j = produces s more basis vectors(q_i)
            = [q_i, Aq_i, ..., A^s q_i] = [V_i, ρ_s(A)q_i]
    endif

    Q_i R_i = QR decomposition(V_j)

    H = R_i B R_i^{-1} = ( H_{1,1} H_{1,2} ... H_{1,s}
                        H_{2,1} H_{2,2} ...   :
                        0      H_{3,2} ...   :
                        ...           ∙   H_{s-1,s-1} H_{s-1,s}
                        H_{s,s-1} H_{s,s}
                        H_{s+1,s} )

    if( NewtonBasis == ON )
        E_i = solving eigenvalues(H)
        E_i = modified_leja_ordering(E_i)
    endif

    for n = 1, ..., s do
        for m = 1, ..., n - 1 do

            ( H_{m,n} ) = ( c_m -s_m ) ( H_{m,n} )
            ( H_{m+1,n} )

        enddo

        c_n = 1 / sqrt(1 + (H_{n+1,n}/H_{n,n})^2)

        s_n = -c_n * H_{n+1,n} / H_{n,n}

        H_{n,n} = c_n H_{n,n} - s_n H_{n+1,n}
        H_{n+1,n} = 0

        ( z_n ) = ( c_n -s_n ) ( z_n )
        ( z_{n+1} ) = ( s_n c_n ) ( z_{n+1} )

    enddo

    y_i = H^{-1} ζ
    x_{i+1} = x_i + Q_i y_i = x_i + V_i R_i^{-1} y_i

```

enddo

図 1 省通信 GMRES 法アルゴリズム

3.1 基底ベクトル

(1) 単基底

単基底では基底ベクトルを

$$K = [q, Aq, A^2q, \dots, A^s q]$$

と生成する. 単基底では $A^s q$ を計算するため数値誤差により基底ベクトルの線形独立性が崩れやすく収束性を悪化させやすい.

(2) ニュートン基底

ニュートン基底では基底ベクトルの生成に前反復のヘッセンベルグ行列 \mathfrak{H} の固有値を用いる. しかし, 非対称実行列の固有値は一般に複素数であるため生成される基底ベクトルが複素数になってしまう. 非対称実行列の固有値は実数あるいは複素共役であるため, Modified Leja ordering[1]により基底生成に用いる固有値の並びを複素共役対が連続に並ぶように並べ替えることで複素数の計算を回避して基底生成を行う.

ニュートン基底では基底ベクトルを次のように生成する.

$$\begin{aligned}
 K &= [\rho_0(A)q, \rho_1(A)q, \rho_2(A)q, \dots, \rho_s(A)q] \\
 \rho_0(A) &= 1, \quad \rho_1(A) = A\rho_0(A) - \alpha_0\rho_0(A) \\
 \rho_j(A) &= A\rho_{j-1}(A) - \alpha_{j-1}\rho_{j-1}(A) - \beta_{j-2}\rho_{j-2}(A) \\
 \alpha_j &= \text{Re}[\theta_j], \beta_j = \begin{cases} -\text{Im}[\theta_j]^2 & \theta_j = \overline{\theta_{j+1}} \\ 0 & \text{otherwise} \end{cases}
 \end{aligned}$$

基底変換行列 B は

$$B = \begin{pmatrix} \alpha_0 & \beta_0 & & & \\ 1 & \alpha_1 & & & \\ & 1 & \ddots & & \\ & & & \beta_{s-3} & \\ & & & \alpha_{s-2} & \beta_{s-2} \\ & & & 1 & \alpha_{s-1} \\ & & & & 1 \end{pmatrix}$$

である.

ニュートン基底を生成する際にほぼ同値な固有値を連続して用いた場合には, 生成される基底は単基底と同様に線形独立性が崩れやすく収束性を悪化させる. これらの問題を回避するために Modified Leja ordering により固有値の順番を複素平面上の距離が遠くなるように並べ替える必要がある.

Modified Leja ordering のアルゴリズムを図 2 に示す. Modified Leja ordering は並べ替える n 個の固有値に共役複素数を含む場合は虚数部が正の固有値に続いて共役な固有値が並ぶように事前に並べ変えておく必要がある. 加えて固有値の多重度を判定しておく必要がある.

Input : n unique eigenvalue z_1, z_2, \dots, z_n ordered so that any complex eigenvalue only occur consecutively in complex conjugate pairs

$z_k, z_{k+1} = \bar{z}_k$, with $\text{Im}(z_k) > 0$ and $\text{Im}(z_{k+1}) < 0$

Input : Each eigenvalue z_k has multiplicity μ_k

$C = 1$

Let k be the least index j maximizing $|z_j|$

$\theta_1 = z_k$; outList(1) = k

if ($\text{Im}(z_k) \neq 0$) then

$\theta_2 = z_{k+1}$; outList(2) = $k + 1$

$L=2$

else

$L=1$

endif

while $L < n$ do

$C' = C$

$C = \prod_{j=1}^{L-1} |\theta_L - \theta_j|^{\frac{\mu_{\text{outList}(j)}}{L}}$

for $j = 1, \dots, n$ do

$z_j = \frac{z_j}{C'}$

$\theta_j = \frac{\theta_j}{C'}$

end do

Let k be the least index k in $\{1, \dots, n\} \setminus \text{outList}$ maximizing

$\prod_{j=1}^{L-1} |z_k - \theta_j|^{\mu_{\text{outList}(k)}}$

$\theta_{L+1} = z_k$; outList($L + 1$) = k

if ($\text{Im}(z_k) \neq 0$) then

$\theta_{L+2} = z_{k+1}$; outList($L + 2$) = $k + 1$

$L=L+2$

else

$L=L+1$

endif

for $j = 1, \dots, n$ do

$\theta_j = C \theta_j$

end do

end while

図 2 Modified Leja ordering アルゴリズム

3.2 QR 分解

(1) CholeskyQR

CholeskyQR[4] のアルゴリズムを図 3 に示す。CholeskyQR は行列積と Cholesky 分解で構成されるため演算密度の高い QR 分解であるが、得られる直交行列の直交性が悪い。通信回数は $V^T V$ の総和通信 1 回である。

$B = V^T V$
Cholesky decomposition(B)
 $Q = VR^{-1}$

図 3 CholeskyQR アルゴリズム

(2) CholeskyQR2

CholeskyQR2[5] のアルゴリズムを図 4 に示す。CholeskyQR2 は CholeskyQR で得られた直交行列に対して CholeskyQR を行う QR 分解である。CholeskyQR2 は 2 回の CholeskyQR を計算するため CholeskyQR よりも演算量が増加しているが得られる直交行列の直交性が改善する。通信回数は $V^T V$ と $Q_1^T Q_1$ の総和通信 2 回である。

$B_1 = V^T V$
 $R_1^T R_1 = \text{Cholesky decomposition}(B_1)$
 $Q_1 = VR_1^{-1}$
 $B_2 = Q_1^T Q_1$
 $R_2^T R_2 = \text{Cholesky decomposition}(B_2)$
 $Q_2 = Q_1 R_2^{-1}$
 $V = Q_1 R_1 = Q_2 R_2 R_1 = QR$
 $Q = Q_2$
 $R = R_2 R_1$

図 4 CholeskyQR2 アルゴリズム

(3) TSQR

TQSR[6] はハウスホルダーQR 分解を基にする直交性の良い QR 分解である。ハウスホルダーQR 分解のアルゴリズムを図 5 に示す。桁落ちを防止するために $\|A(i: m, i)\|e_i(i: m)$ の符号を $A(i: m, i)$ と同符号に置き替えるアルゴリズムも存在するが、省通信 GMRES 法は導出の過程で上三角行列の対角成分が正であることを仮定しているため $\|A(i: m, i)\|e_i(i: m)$ の符号は負でなければならない。LAPACK 等の数値計算ライブラリのハウスホルダーQR 分解を利用した場合は上三角行列の対角成分に負の値が混じることが原因で計算が破綻する場合がある。

実装した TSQR は各プロセスが Sequential TSQR でローカルな QR 分解を行い、各プロセスの三角行列を Parallel TSQR で縮約を行う方法である。通信は Parallel TSQR の三角行列の縮約による二分木通信と縮約した三角行列の全プロセスへの送信が必要になる。

ローカルな QR 分解として Sequential TSQR の代わりに一回のハウスホルダーQR 分解を用いることも可能である。Sequential TSQR はブロック化して QR 分解を行うため演算数はハウスホルダーQR 分解よりも増加する。しかし、QR 分解する行列がキャッシュに収まらない場合、ハウスホルダーQR 分解ではメモリアクセス数が Sequential TSQR よりも増加する。

for $i = 1, \dots, n$ do
 $y_i(i: m) = A(i: m, i) - \|A(i: m, i)\|e_i(i: m)$
 $t_i = \frac{2}{(y_i(i: m), y_i(i: m))}$
 $Q_i = (1 - t_i y_i(i: m)) y_i(i: m)^T$
 $A(i: m, i) = Q_i A(i: m, i)$
enddo

図 5 ハウスホルダーQR 分解アルゴリズム

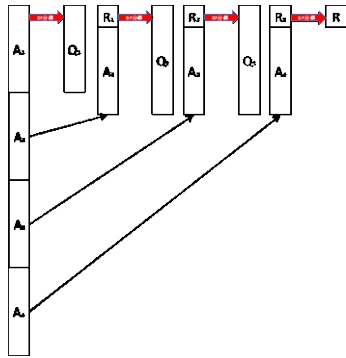


図 6 Sequential TSQR における三角行列の縮約

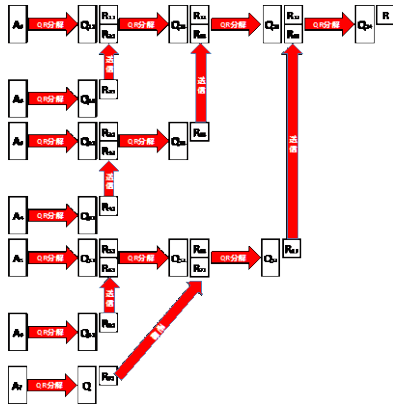


図 7 Parallel TSQR における三角行列の縮約

4. 性能測定

4.1 計算環境

性能測定は日本原子力研究開発機構が所有する大型計算機 ICE-X を利用した。ICE-X の諸元を表 1 に示す。

性能測定は問題サイズ $160 \times 160 \times 32 \times 92$ として ICE-X 2 ノード使用して計測した。

表 1 ICE-X 諸元 (1 ノード)

プロセッサ	Intel Xeon E5-2680 v3 × 2
コア数	12 × 2
メモリ[GB]	64
キャッシュ[MB]	30 × 2
理論演算性能(倍精度) [Gflops]	960
ストリームバンド幅[GB/s]	116.64
アーキテクチャ	Haswell
SIMD 幅[bit]	256
コンパイラ	intel compiler 16.0.1
コンパイラオプション	-O3 -mcmmodel=large -qopenmp -align array64byte -no-prec-div -xHost

4.2 測定結果

(1) 基底ベクトル

単基底の省通信 GMRES 法の収束履歴を図 8 に示す。QR 分解として CholeskyQR を使用した。ここで、省通信ステップ数 s は省通信 GMRES 法のリスタート長と等しくなるように設定した。 $s=22$ 以下では単調に残差が減少していくが

$s=23$ 以上では収束が鈍くなる。 $s=30$ では残差が増加する反復も生じており反復回数が $s=22$ よりも増加している。一般的にリスタート付き GMRES 法ではリスタート長を大きくすると収束性が向上するが、単基底の省通信 GMRES 法では $s=23$ 以上で基底ベクトルの線形独立性が崩れてしまい収束性が悪化している。

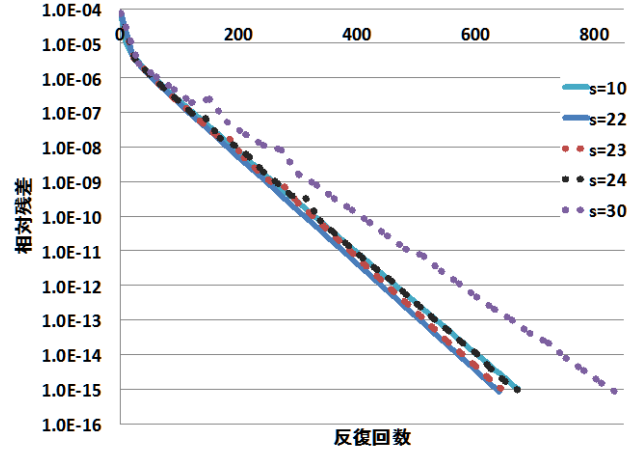


図 8 単基底の収束履歴 (CholeskyQR)

ニュートン基底の省通信 GMRES 法の収束履歴を図 9 に示す。QR 分解として CholeskyQR を使用した。ニュートン基底では Modified Leja ordering による固有値の並べ替えを行わない場合は s が 36 で計算が破綻してしまう。適切に固有値を並べ替えることで線形独立性が崩れにくくなり s が 37 まで大きくなる。固有値の並べ替えをしても s が 38 以上では計算が破綻する。

$s=22$ の 1 回あたりの基底ベクトル生成時間を表 2 に示す。ニュートン基底では単基底と比べて 2 回の DAXPY が増えるため基底生成時間は単基底よりも増加する。

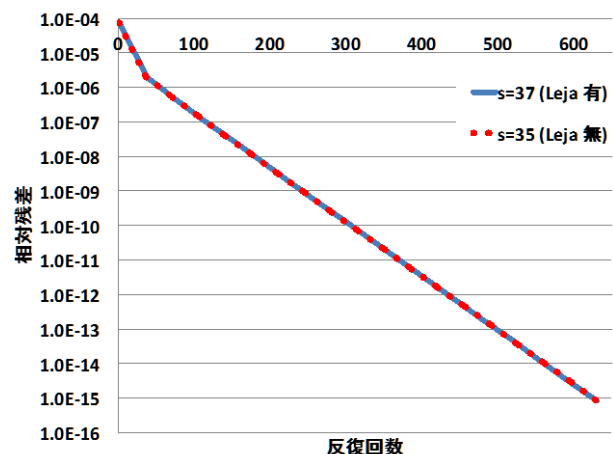


図 9 ニュートン基底の収束履歴 (CholeskyQR)

表 2 1 回あたりの基底ベクトル生成時間($s=22$)

	単基底	ニュートン基底
計算時間 [ms]	414	497

(2) QR 分解

QR 分解の理論性能評価を表 3 に示す。理論性能の評価はループラインモデルを用いた[7]。省通信 GMRES 法では QR 分解で直交行列の計算は不要であるため CholeskyQR では 1 回の行列積で良いが、CholeskyQR2 は 3 回の行列積を計算する必要がある。そのため CholeskyQR2 の理論計算時間は CholeskyQR の 3 倍程度になる。修正グラムシュミットは上三角行列を計算するために直交行列を計算する必要があるため理論計算時間が最も大きくなる。ローカルな QR 分解としてハウスホルダーQR を用いる TSQR ではキャッシュに QR 分解する行列が収まらないためメモリアクセスが Sequential TSQR よりも増加している。

性能測定結果を表 4 に示す。基底生成は単基底を使用した。計算時間は CholeskyQR の性能が最も良いが直交性が最も悪い。今回の問題では単基底で $s=22$ でも収束性の悪化はないため直交性は悪いが線形独立性が保たれていると考えられる。最も直交性の良い QR 分解はローカルな QR 分解を Sequential TSQR で計算した TSQR である。直交性が良いため s を増やしても線形独立性が崩れにくく s を伸ばすことができる。計算時間についても CholeskyQR に次いで良い。

図 10 に $s=35$ の場合の省通信 GMRES 法の収束履歴を示す。基底生成は単基底を使用した。CholeskyQR は $s=30$ の場合よりもさらに収束性が悪化している。TSQR (Sequential TSQR) は収束性が悪化することなく収束している。CholeskyQR2 は若干の収束性の悪化がみられるが TSQR (Sequential TSQR) とほぼ同等な収束を示している。

表 3 1 回あたりの QR 分解の理論性能評価($s=22$)

	演算数 [Flop]	メモリ参照量 [Byte]	理論計算時間 [ms]
修正グラムシュミット	1056	8272	2832
CholeskyQR	552	184	85
CholeskyQR2	1656	552	254
TSQR (Sequential TSQR)	1058	184	105
TSQR (ハウスホルダーQR)	1058	7400	2538

表 4 1 回あたりの QR 分解時間($s=22$)

	計算時間 [ms]	直交性 $\ Q^T Q - I\ $
修正グラムシュミット	3336	3.03×10^{-9}
CholeskyQR	89	1.52×10^{-6}
CholeskyQR2	251	6.87×10^{-10}
TSQR (Sequential TSQR)	233	1.47×10^{-10}
TSQR (ハウスホルダーQR)	2733	2.96×10^{-9}

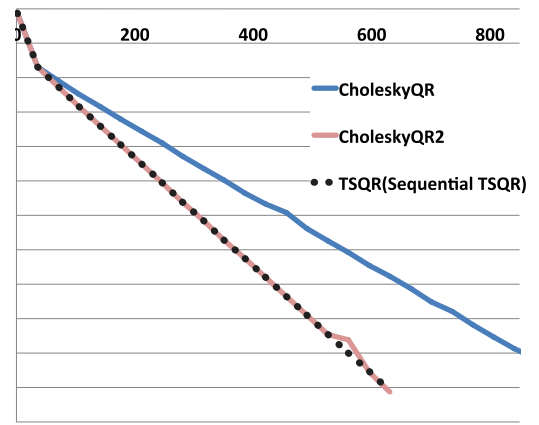


図 10 省通信 GMRES 法の収束履歴(省通信ステップ数 $s=35$)

(3) 線形独立

$s=30$ において生成した基底ベクトル V と直交化した基底ベクトル Q のランクを図 11 に示す。ランクは特異値から算出した。単基底+CholeskyQR では V, Q どちらのランクも基底ベクトルの本数($s+1$)と一致していないため線形独立が崩れている。ニュートン基底+CholeskyQR では V, Q のランクが共に基底ベクトルの本数と一致しているため線形独立が保たれている。単基底+CholeskyQR2 では V の線形独立が崩れているが Q のランクが基底ベクトルの本数と一致するため線形独立が保たれている。同様に単基底+TSQR(Sequential TSQR)においても V の線形独立は崩れているが Q の線形独立は保たれている。

図 12 に $s=30$ の省通信 GMRES の収束履歴を示す。直交化した基底ベクトルの線形独立性が保たれていない場合は収束性の悪化が確認できる。

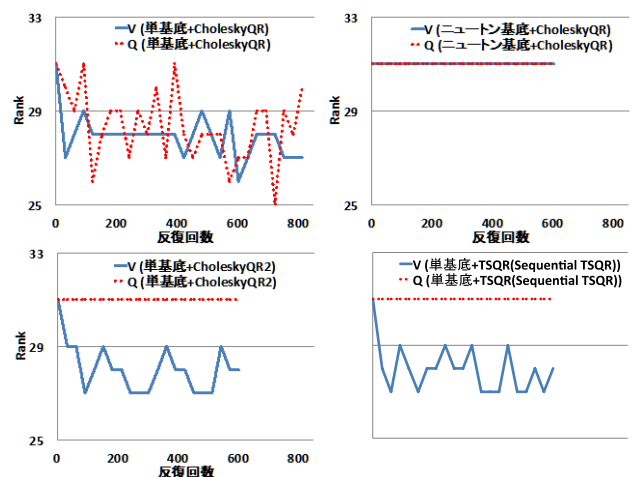


図 11 基底ベクトルのランク(省通信ステップ数 $s=30$)

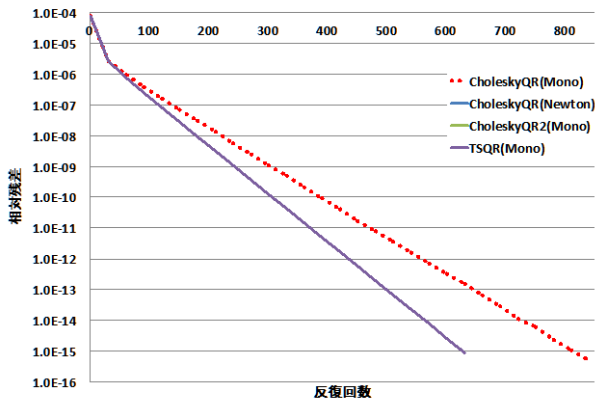


図 12 省通信 GMRES 法の収束履歴(省通信ステップ数 s=30)

省通信 GMRES 法 [ms]	20271	31839	33242
------------------	-------	-------	-------

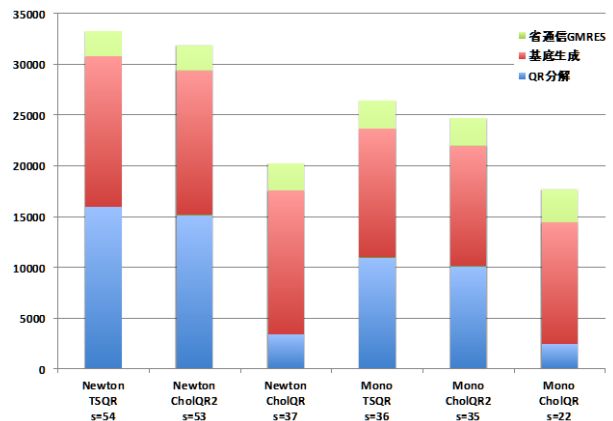


図 13 省通信 GMRES 法性能測定結果

(4) 省通信 GMRES 法

単基底省通信 GMRES 法の性能測定結果を表 5, ニュートン基底省通信 GMRES 法の性能測定結果を表 6, 性能比較結果を図 13, 収束履歴を図 14 に示す. 省通信ステップ数は収束が悪化しない範囲で最大のものを使用した. 単基底と CholeskyQR を組み合わせた省通信 GMRES 法が最も計算時間が短い結果になった. 今回の問題は条件の良い行列であるため収束性の悪化が s=23 まで生じないことに加え, 計算ノードが 4 ノードと少なく集団通信による影響が小さいことが要因である.

条件の悪い行列では収束性の悪化が小さい省通信ステップ数でも生じると予想されるため, ニュートン基底や直交性の良い QR 分解を利用しなければ省通信ステップ数を伸ばすことが難しくなる. 数千ノードを用いる大規模並列計算では集団通信のコストが支配的になるため省通信ステップ数を伸ばすことが難しい単基底や CholeskyQR では性能が頭打ちになることが予想される.

表 5 単基底省通信 GMRES 法の性能測定結果

	Cholesky QR	Cholesky QR2	TSQR (Sequential TSQR)
省通信ステップ数	22	35	36
反復回数	638	630	648
基底生成 [ms]	12004	11908	12670
QR 分解 [ms]	2431	10094	10957
省通信 GMRES 法 [ms]	17702	24737	26447

表 6 ニュートン基底省通信 GMRES 法の性能測定結果

	Cholesky QR	Cholesky QR2	TSQR (Sequential TSQR)
省通信ステップ数	37	53	54
反復回数	629	636	648
基底生成 [ms]	14132	14239	14819
QR 分解 [ms]	3427	15138	15930

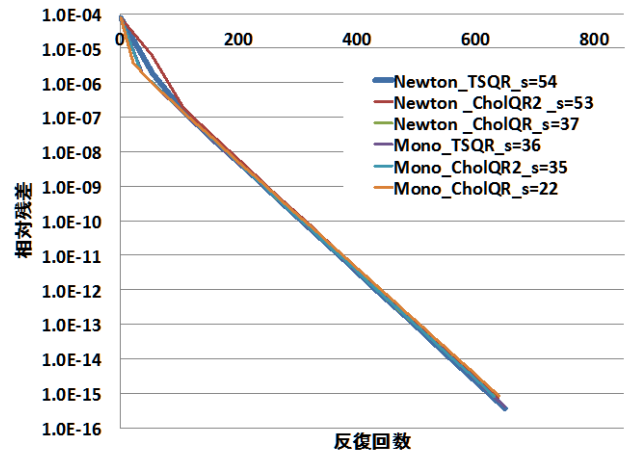


図 14 省通信 GMRES 法収束履歴

5. おわりに

本研究では大規模原子力流体コード GT5D に省通信 GMRES 法を適用し収束性を調査した. ニュートン基底を用いて基底ベクトルを生成する場合に Modified Leja ordering により固有値を並べ替えることで収束性が改善することを示した. QR 分解の直交行列の直交性を向上させることで収束性が改善した. 計算性能は線形独立性が崩れやすい単基底と直交性の悪い CholeskyQR を用いた省通信 GMRES 法が最も良い結果になった. 大規模並列計算では集団通信のコストが支配的になるため線形独立性が保たれやすいニュートン基底や直交性の高い QR 分解を使用した省通信ステップ数 s を伸ばしやすいく省通信 GMRES 法の計算性能がよくなると予想される. 大規模並列計算での性能測定及び収束の加速と省通信 GMRES 法を適用可能な問題を増やすためにリスタート時の初期値改善手法であるデフレーション[8], look-back[9][10]を適用することが今後の課題である.

参考文献

- [1]M. Hoemmen, Communication-avoiding Krylov subspace methods, Ph.D. dissertation, University of California, Berkeley, 2010.
- [2]Y. Idomura, et al., Conservative global gyrokinetic toroidal full-f five-dimensional Vlasov simulation., *Comput. Phys. Commun.* 179 391, 2008.
- [3]Y. Idomura, et al., Application of a communication-avoiding generalized minimal residual method to a gyrokinetic five dimensional Eulerian code on many core platforms., *ScalA 17* 2017
- [4]A. Stathopoulos and K. Wu. A block orthogonalization procedure with constant synchronization requirements. *SIAM J. Sci. Comput.*, 23:2165–2182, 2002.
- [5]T. Fukaya, et al., CholeskyQR2: a simple and communication-avoiding algorithm for computing a tall-skinny QR factorization on a large-scale parallel system., *ScalA 14*, 2014
- [6]J. Demmel, et al., Communication-optimal parallel and sequential QR and LU factorizations, Technical Report, University of California, Berkeley, 2008.
- [7]S. Williams, A. Waterman, and D. Patterson. 2009. Roofline: an insightful visual performance model for multicore architectures. *Commun. ACM* 52, 4 (2009),
- [8]I. Yamazaki et al., Deflation Strategies to Improve the Convergence of Communication-Avoiding GMRES., *ScalA 14*, 2014
- [9]今倉暁, 曾我部知広, 張紹良, 非対称線形方程式のための Look-Back GMRES(m)法, *日本応用数学会論文誌*, Vol.22, No.1, 2012, pp.1-22.
- [10]今倉暁, 曾我部知広, 張紹良, デフレーション型と Look-Back 型のリスタートを併用した GMRES(m)法の収束特性, *日本応用数学会論文誌*, Vol.22, No.3, 2012, pp.117-141.